MITNE-146

UTILITY SYSTEM INTEGRATION AND OPTIMIZATION MODELS FOR NUCLEAR POWER MANAGEMENT

Paul Ferris Deaton

Edward A. Mason

Work Sponsored by
Commonwealth Edison Company
Chicago, Illinois

Department of Nuclear Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts

Vol. 1 & 2

MITNE-146

UTILITY SYSTEM INTEGRATION AND OPTIMIZATION MODELS FOR NUCLEAR POWER MANAGEMENT

by

Paul Ferris Deaton

Supervisor

Edward A. Mason

DEPARTMENT OF NUCLEAR ENGINEERING
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, MASSACHUSETTS

M.I.T. DSR PROJECT NO. 72107

Work Sponsored by

Commonwealth Edison Company
Chicago, Illinois

Issued: June 1973

UTILITY SYSTEM INTEGRATION AND OPTIMIZATION MODELS FOR NUCLEAR POWER MANAGEMENT

by

Paul F. Deaton

Submitted to the Department of Nuclear Engineering in May 1973 in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

ABSTRACT

A nuclear power management model suitable for nuclear utility systems optimization has been developed for use in multi-reactor fuel management planning over periods of up to ten years. The overall utility planning model consists of four sub-models: (1) Refueling and Maintenance Model (RAMM), (2) System Integration Model (SIM), (3) System Optimization Model (SOM), and (4) CORE Simulation and Optimization Models (CORSOM's). The SIM and SOM sub-models were developed in this study and are discussed in detail; full-scale computerized versions of each (SYSINT and SYSOPT, respectively) are evaluated as part of the methods development research.

The RAMM generates feasible, mutually exclusive nuclear refueling-fossil maintenance schedules. These are evaluated in detail by the rest of the model. Using the Booth-Baleriaux probabilistic utility system model, the SIM integrates the characteristics of the utility's plants into a representation which meets the necessary operating constraints. Scheduling of system nuclear production and detailed fossil production is done for each time period (few weeks) making up the multi-year planning horizon.

Utilizing a network programming model, the SOM optimizes the detailed production schedules of the nuclear units so as to produce the required system nuclear energy at minimum system cost. CORSOM's are utilized to optimize reload parameters (batch size and enrichment) and to generate the individual reactor fuel costs and nuclear incremental costs. These incremental costs are then used by the SOM's iterative gradient optimization technique known as the method of convex combinations.

The SYSINT model is shown to be remarkably fast, performing the Booth-Baleriaux simulation for a single time period on a system with over 45 generating units in less than 2.5 seconds on an IBM-370 model 155 computer. SYSOPT converged to optimum solutions in roughly ten iterations. Immediate reduction of iterations by roughly half is estimated by merely increasing piecewise-linearization of the network objective function. Overall model computational requirements are limited by available CORSOM's, which require 99% of the computational effort (over 3 minutes per reactor per SOM iteration).

Nuclear incremental costs (∿ 0.8-1.6 \$/MWH) are shown to be less than fossil incremental costs (> 2.0 \$/MWH) for the foreseeable future. Thus, nuclear power should always be operated so as to supply customer demands with a minimum use of the more expensive fossil energy. For the same reason, the lengthening of nuclear irradiation cycles (in terms of both energy and time) more than pays for itself by reducing the total cost of fossil replacement energy. Idealized nuclear production schedules yield constant nuclear incremental costs regardless of reactor unit and time. One of the key input parameters is the fossil thermal energy cost.

Thesis Supervisor: Edward A. Mason

Title: Professor of Nuclear Engineering

To the 1970-1971 President of the

Massachusetts Institute of Technology's

Technology Dames

my wife

Penelope Craig Deaton

ACKNOWLEDGMENTS

In over five years at MIT, I find myself, at this final moment, indebted to almost everyone, particularly, the Student Loan Office. Consequently, were it not for the generous support of the Atomic Energy Commission (3 years) and the John & Fannie Hertz Foundation (2-1/2 years), I would have been unable to accumulate the vast riches Penny and I now own -- Kimberly Anne and Robyn Michele. And to married students with children, the MIT Westgate Apartments are a real godsend.

With good fortune, I have come to know, during my tenure at MIT, the entire staff of the Department of Nuclear Engineering. To Professor Manson Benedict, I, like many students before me, owe the largest debt of all -- the inspiration to enter my chosen specialty. His sincerity and enthusiasm, both for the subject matter and the students themselves, bespeaks an educator in the truest sense of the word. For the sake of the students to follow, I hope his retirement is in name only. To Professor Edward A. Mason, whose painsaking editorial efforts have attempted to make this document intelligible to others besides myself, I say a profound "Thank You." For providing a receptive ear to many an idea, complaint or just plain trivia, I wish to thank fellow graduate student Joseph P. Kearney.

Many thanks also go to the Commonwealth Edison Company of Chicago for initiating the MIT research project on which this work is based. In particular, I wish to thank W. K. Kiefer and E. F. Koncel for raising many incisive questions and for providing much valuable data and many excellent ideas. Through my association with Commonwealth Edison, it was my pleasure to participate in the stimulating sessions of the Joint Systems Analysis Task Force. SYSINT, in fact, was inspired by the original TVA-ORNL model SYSSIMUL, developed by R. R. Booth.

The Out Of Kilter Network Program was graciously provided by the MIT Flight Transportation Laboratory. All computations ($5000 worth, all provided by Commonwealth Edison) were performed at the MIT Information Processing Center.

To the nine typists who gave of their time to decipher various drafts of this document: "Your accuracy was amazing!" To my new employer, Westinghouse Nuclear Energy Systems, for permitting me the run-of-the-house to complete this document (from drawing the figures to printing the document), I am most deeply indebted.

As for my long-neglected family, to my wife, words cannot express the gratitude for the many large and small things she has done to be both a mother and a father to our children as deadline after deadline slipped by unmet. Finally, to my children, who innumerable times saw Dad leave for the office before they were in bed, I answer, at long last, "Yes" to Kimmy's bedtime question, "Daddy, is your t'esis done?"

TABLE OF CONTENTS

Page

Page

Page

Page

LIST OF FIGURES

Figure | Page

Figure | Page

LIST OF TABLES

Table | Page

Table | Page

CHAPTER 1

AN OVERVIEW

1.1 Historical Perspective of Nuclear Power Management

The advent of commercial nuclear power created new and complex challenges to electric utility management. The utility's staff not only had to resolve difficult questions concerning safety and the environment during a nuclear plant's construction, but also ensure the economical production of energy during the plant's operating life. To aid management in this operation planning, much effort was expended incorporating nuclear power plants into existing utility system optimization models. By making reasonable and convenient assumptions (e.g., base-load operation and annual refuelings), the nuclear fuel cycle cost was determined satisfactorily and allowed a nuclear plant to be treated merely as a "fossil" plant with extremely low fuel cost.

However, as more nuclear plants are added to the grid and nuclear power makes up a larger fraction of the installed capacity, these assumptions become suspect. As a result, operating plans based on them, may be far from optimal. "Traditional methods for planning the operation of a power system cannot adequately consider nuclear fuel economics or fully recognize constraints imposed by the nature of the nuclear fuel cycle (28)."

Thus, current emphasis has shifted to developing utility nuclear power management tools which properly model nuclear plants and the complexity of the nuclear fuel cycle.

1.2 Planning Tools Needed

Utility system planners are faced with four general types of decisions:

(1) scheduling production,

(2) scheduling maintenance and refueling,

(3) purchasing new fuel and

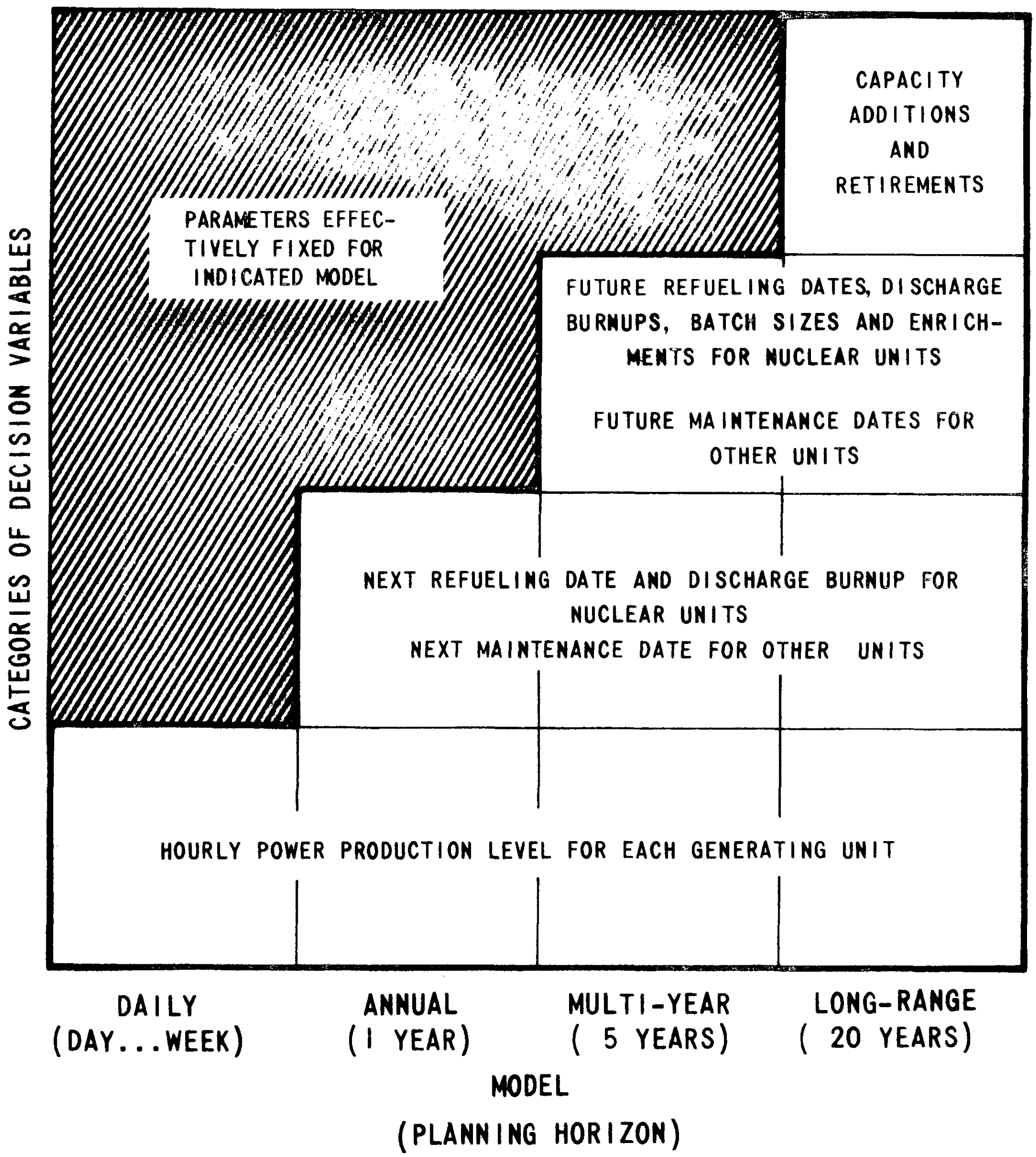
(4) purchasing new capacity.

The above ordering of these decisions is not arbitrary. Each of these problems dominates decision-making on a longer time scale. Conversely, each characteristic time scale imposes a different set of constraints on the options available to the planner. Daily production scheduling must be performed within the context of the yearly maintenance and refueling schedules. Likewise, these scheduled outages must be coordinated with longer term fuel contracts and deliveries. Similarly, long term fuel contracts must be cognizant of future capacity additions and retirements.

The complexities of accurately and efficiently modelling the nuclear fuel cycle for each of these decisions requires four different utility system simulation models (see Figure 1.1):

Figure 1.1

Decision Variables Associated with the Hierarchy of Nuclear Utility System Planning Models



(1) Daily Model: This model deals with the hour-by-hour dispatching of the various generating units. Only a small fraction of the energy potential in the nuclear fuel is released and the sole parameter available for optimization is the power output of each plant.

(2) Annual Model: This model deals with the operation of the nuclear plants between refuelings. The fuel in each reactor cannot be replaced, but the power operation of the reactor, date of the next refueling, and energy potential of the discharge fuel are decision variables for each unit. Widmer's analytical treatment of steady-state nuclear refueling (57, 59) referred to this time scale as "short-range."

(3) Multi-year Model: This model spans the time required for the complete nuclear fuel cycle (on the order of 5 to 10 years). In addition to the variables mentioned for the annual model, this one includes the fuel management reload variables--fuel enrichment and batch size. This time scale plays the determining role in planning for the purchase of fuel and its required processing and fabrication, as well as the financing of all these costs. In the study by Widmer (58, 59) this time scale was referred to as "mid-range."

(4) Expansion Model: This model covers a period of many years--on the order of the expected lifetime of generating stations--and is employed in planning for the addition and retirement of generating equipment. Within the first three models,

certain plants are assumed to exist or to have been ordered so that the type and characteristics of each unit are specified. But in the expansion model, a variety of new energy production equipment is under investigation.

Several considerations pointed to the multi-year model as deserving the initial development effort. Relative to Figure 1.1, such a model ought to have many elements useful in the development of the other three models. At the same time, the multi-year model possesses all of the complex options inherent in nuclear fuel management without the additional complexity of the plant installation decision itself. Finally, multi-year considerations vitally affect decisions regarding long-term fuel financing. Such large dollar commitments hint at large cost savings.

For these reasons, the multi-year nuclear power management model put forth in this work was developed as the first of the Commonwealth Edison-sponsored utility system optimization research projects at the Massachusetts Institute of Technology.

1.3 Introduction to Multi-year Planning

In providing installed capacity to meet the customer loads, a utility relies on up to five different types of generating equipment:

(1) Nuclear units: very large capacity units generating electricity from steam produced via the heat released by a sustained nuclear chain reaction contained within the reactor's core.

(2) Fossil steam units: typically large capacity coal, oil and/or gas-fired boilers producing steam that is expanded in turbine-generators.

(3) Fast-start peaking units: small fossil-fueled jet engine, gas turbine or diesel-driven generators.

(4) Hydro units: Typically medium capacity hydro-electric turbines associated with dams which form water reservoirs.

(5) Pumped-hydro units: similar to hydro except that its dual-purpose turbine may alternately operate as a pump, transferring water from the foot of the dam to the higher reservoir elevation. Like a storage battery, cheap off-peak energy is temporarily stored in another form (water at a height) for retrieval during the peak by reversing the process.

Regardless of the type of unit, certain key information is required by the system planner on each and every unit of the system:

(1) minimum and maximum power level,[1]

(2) fuel consumption rate vs. power level,

(3) fuel cost,

(4) fuel inventory,

---

[1]Throughout this work, all power levels are in units of net MWe delivered to the transmission system busbar. That is, plant auxiliary power requirements (~5%) have already been subtracted from gross generator output, but transmission losses have not been accounted for.

(5) transmission losses,
(6) startup-shutdown data,
(7) maintenance requirements, and
(8) reliability data.

Table 1.1 presents a general summary of these characteristics for each unit type, including capital cost estimates.

With the rates (prices) per unit electricity fixed externally by regulatory commissions and the total amount of electricity determined externally by the customers' demands, the total revenue received by the utility is also fixed (albeit, in a probabilistic sense). By minimizing the revenue required to recover the cost of supplying that electricity, the utility maximizes total profit. Therefore, the utility objective function is the minimizing of the present value of all future required revenue, i.e., the revenue requirement. (Present valuing accounts for the time value of money.) For any project, this sum represents that amount of money which, if received immediately and invested in the company, would just suffice to pay all expenses, as well as permitting a fair return to investors.[2] By including investors' permitted return as another cost component, "revenue requirements" and "total cost" become synonymous.

When considering different operating strategies over a multi-year time horizon (on the order of 5 years), many of the cost components (e.g., capital investment and overhead) are essentially fixed.

The multi-year objective function may, therefore, be reduced to the operating costs directly related to supplying

---

[2] More precisely (55),

"The revenue requirement is that sum of money, which if received as revenue by an investor-owned electric utility at the beginning of the planning horizon and invested in the enterprise, will defray all subsequent fuel cycle costs, the return allowed by regulatory agencies on that portion of the original investment remaining unexpended at any time, and defray all associated income taxes."

Table 1.1

Characteristics of Types of Electric Generating Units

	Dimension	Nuclear Steam (LWR)	Fossil Steam	Fast-Start Peaking	Hydro	Pumped-Hydro
System Use		Base-Load	Base-Load and Cyclical	Peaking	Inventory Dependent	Peaking
Capacity Fact.	Percent	60-90	30-90	Up to 20	Up to 100	Up to 50
Capital Cost	$/kwe	300-450	250-400	100-150	300-500	100-200
Unit Capacity	MW	500-1200	200-1200	10-50	10-600	50-400
Min. Power	% Cap.	10-40	10-50	75-90	0-10	25-40
Avg. Ht. Rate	MBTU/MWH	10.5-11	8.5-14	12-17	N/A	N/A
Avg. Net Energy Conversion Eff.	Percent	31-34	25-40	20-28	85-93	65-80
Fuel Cost	¢/MBTU	16-20	35-80 (Coal) 50-100 (Oil)	50-100	0	Cost of pumping power
Energy Cost	$/MWH	1.7-2.2	3.0-8.4	6.5-20	0	~1.5 X pumping power
Comments on Fuel Inventory		Depends on fuel cycle	Approx. const. at 100 days supply	4-8 hours (Oil)	Depends on season	Depends on operating cycle
Trans. Losses	Percent	Up to 10	Up to 10	Up to 5	Up to 10	Up to 15
SU-SD Ht. Reqt.	MBTU/MW Cap.	3-6	3-8	0-2	~0	~0
Min. SD Time	Hours	<2	2-10	< 0.3	< 0.5	< 0.5
Maint. Reqt.	Week/Year	4-8 wk/refuel	3-5	1-4	1-2	1-2
Forced-Out Rate	Percent	Up to 15	Up to 20	Up to 40	Up to 5	Up to 10
Perf. Prob.	Percent	85-100	80-100	90-100	95-100	95-100

customer loads--fuel consumption within the system and net electricity purchases from neighboring utilities along with the associated taxes and carrying charges.

Adopting the notation that RR(X) is the total revenue requirement related to direct expenditure X,

$$\begin{aligned} RR(X) = &\ \underset{\text{Value}}{\text{Present}}\ (\text{Expenditure X}) \\ &+ \underset{\text{Value}}{\text{Present}} \begin{pmatrix}\text{Taxes associated} \\ \text{with X}\end{pmatrix} \\ &+ \underset{\text{Value}}{\text{Present}} \begin{pmatrix}\text{Carrying charges} \\ \text{associated with X}\end{pmatrix} \end{aligned} \qquad (1.1)$$

Fuel consumption expenditures can be further broken down into:

(1) X_F, fossil fuel related directly to on-line production,

(2) X_N, nuclear fuel related directly to on-line production, and

(3) X_S, fuel related to units' startup-shutdown heat requirements.

Expenditures for electricity purchases from other utilities, X_U, represents both emergency purchases and economy purchases. (Economy purchases are not considered further in this work.)

The standard procedure in performing multi-year optimization is to subdivide the entire planning horizon into Z smaller time periods. In each time period p, expenditures are estimated in undiscounted dollars. Period expenditures are then present-valued at x per year from their mean time $\bar{t}_p$ back to time zero. As Section 1.4 will point out, the addition of nuclear units may prevent immediate evaluation of X_N. [In fact, $RR(X_N)$ or RR_N is determined directly only after all periods have been simulated.]

The equivalent multi-year objective function ORR, the operating revenue requirement, can then be expressed as

$$ORR = RR_F + RR_N + RR_S + RR_U \qquad (1.2)$$

or, in terms of the nonnuclear period expenditures,

$$ORR = \sum^{Z} X_{F_p} \frac{1}{(1+x)^{\bar{t}_p}} + RR_N + \sum^{Z} X_{S_p} \frac{1}{(1+x)^{\bar{t}_p}} + \sum^{Z} X_{U_p} \frac{1}{(1+x)^{\bar{t}_p}} \qquad (1.3)$$

1.4 <u>Complexities of Nuclear Power</u>

The cost of fossil fuel is simply the cost of coal or oil plus shipping charges. Assuming a constant coal stockpile, newly delivered coal is burned immediately. From mine to ash, fossil fuel consumption requires only a matter of some days.

Nuclear fuel, on the other hand, requires years to account for all cost components. Mining, conversion and enrichment begin a year or more before insertion in the reactor. During the three years or more of irradiation, the energy potential is slowly extracted not only from this fuel batch, but also from two or so others in the core. Four months or more after discharge, reprocessing occurs and fissile isotope credits are received. The net result is that $\overline{TC}_r$, the cost of a reactor's fuel over a time span of C cycles, is a nonlinear, nonseparable function of the energy produced in each cycle, E_{rc},

$$\overline{TC}_r = \overline{TC}_r \left(E_{r1}, E_{r2}, \ldots, E_{rC} \right) \tag{1.4}$$

Summing each reactor's total fuel cost (i.e., revenue requirement) yields the system nuclear revenue requirement, RR_N,

$$RR_N \equiv \overline{TC} = \sum^{R} \overline{TC}_r \tag{1.5}$$

Qualitatively, the nonlinearity,

$$\overline{TC}_r \neq c_{r0} + c_{r1} E_{r1} + c_{r2} E_{r2} + \cdots + c_{rC} E_{rC} \tag{1.6}$$

results from the fact that, given the refueling batch fractions, cycle energy is approximately linear in reload enrichment, but the cost of this enrichment (i.e., separative work requirement) is nonlinear.

Preventing a more general uncoupling of the cycle energies,

$$\overline{TC}_r \neq C_{r0} + C_{r1}(E_{r1}) + C_{r2}(E_{r2}) + \dots + C_{rC}E_{rC} \quad (1.7)$$

is the multi-irradiation (multi-zone) nature of today's LWR refueling schemes. The specification of reload enrichments requires not only reactivity allowance for the next cycle, but succeeding ones as well.

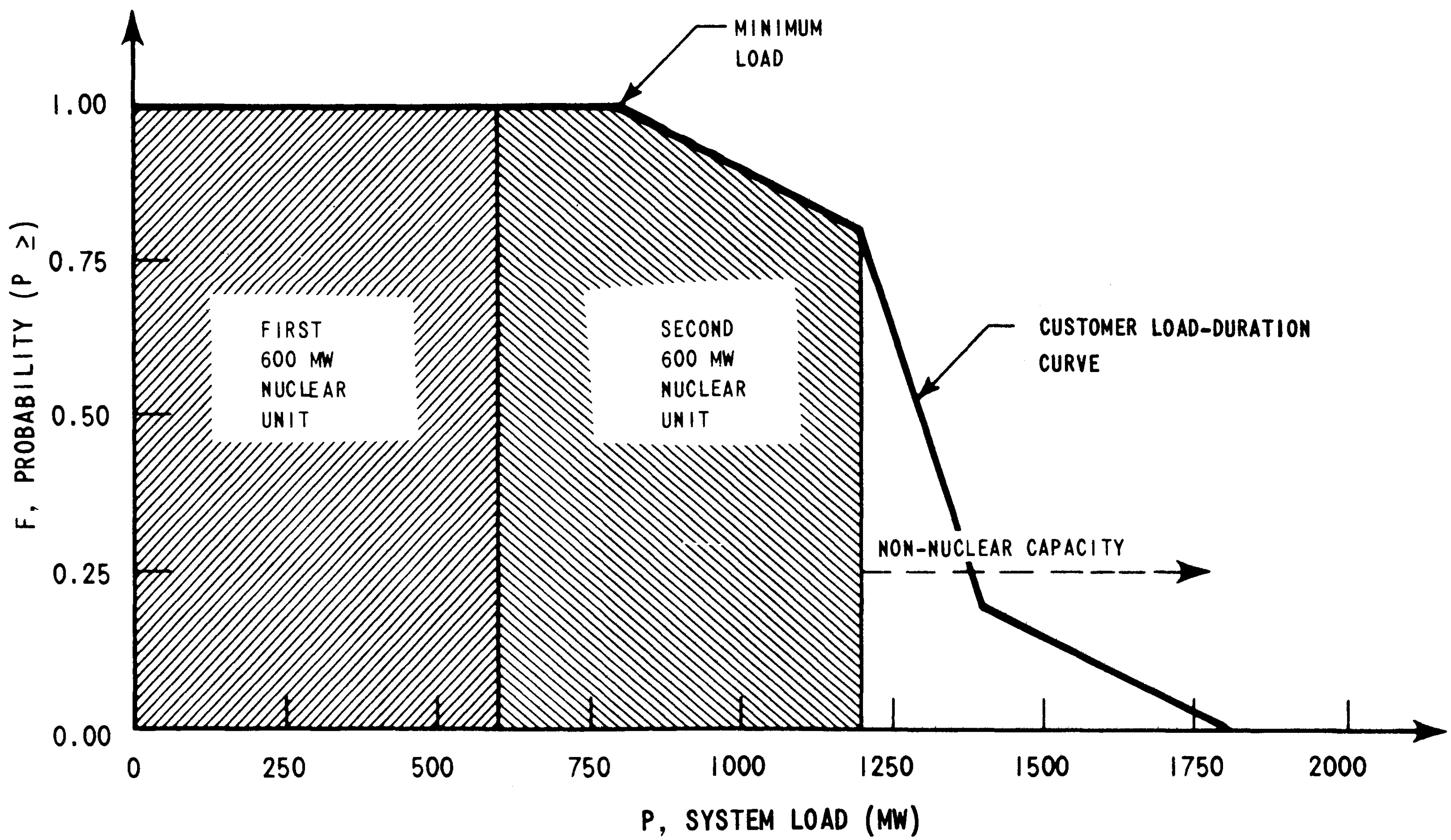
In summary, to calculate nuclear fuel costs, the cycle energies to the horizon of interest must be known.

In the early years of nuclear power, this stringent requirement did not pose a problem for conventional production scheduling models. With only one nuclear plant on a system (see Figure 1.2), base-load operation was possible. That is, nuclear units were operated at full capacity whenever they were available. (In addition, annual refueling meshed nicely with fossil maintenance plans and appeared to be reasonably economical.) For the base-load case (i.e., availability-based capacity factor for unit r, $L_r' = 1$), cycle energy E_{rc} could be immediately determined since

$$E_{rc} = p_r T_{rc}' K_r L_r' \quad (1.8)$$

Figure 1.2

Nuclear Capacity Greater than Minimum Load



where

p_r = estimated probability reactor r is capable of generating energy at random instant of time

T'_{rc} = length of irradiation cycle c for unit r, hours

K_r = rated electric capacity of unit r, MW

If T'_{rc} was constant, the cycles energies to the horizon were the same and reactor steady-state fuel costs could be calculated and used for all cycles.

However, as nuclear capacity on the system increased, two problems became apparent. First, not all nuclear units could be base-loaded if total nuclear capacity was greater than the minimum load (see Figure 1.2). Equation (1.8) was no longer easily evaluated because the nuclear portion of the load-duration curve was no longer equal to 1.0 for all nuclear units ($L'_r = ? < 1$). Which nuclear unit should occupy the base-load position? Intra-nuclear incremental cost competition had surfaced for the first time. Only rough estimates of nuclear fuel costs had been necessary to decide that all nuclear equipment was cheaper than all fossil equipment (22), but very refined costs were now needed to decide nuclear unit A versus nuclear unit B.

Secondly, annual refueling created scheduling problems when each nuclear unit had to be refueled within every calendar scheduling window. Coupled with decreasing nuclear load demand, what was the optimum cycle length for each reactor?

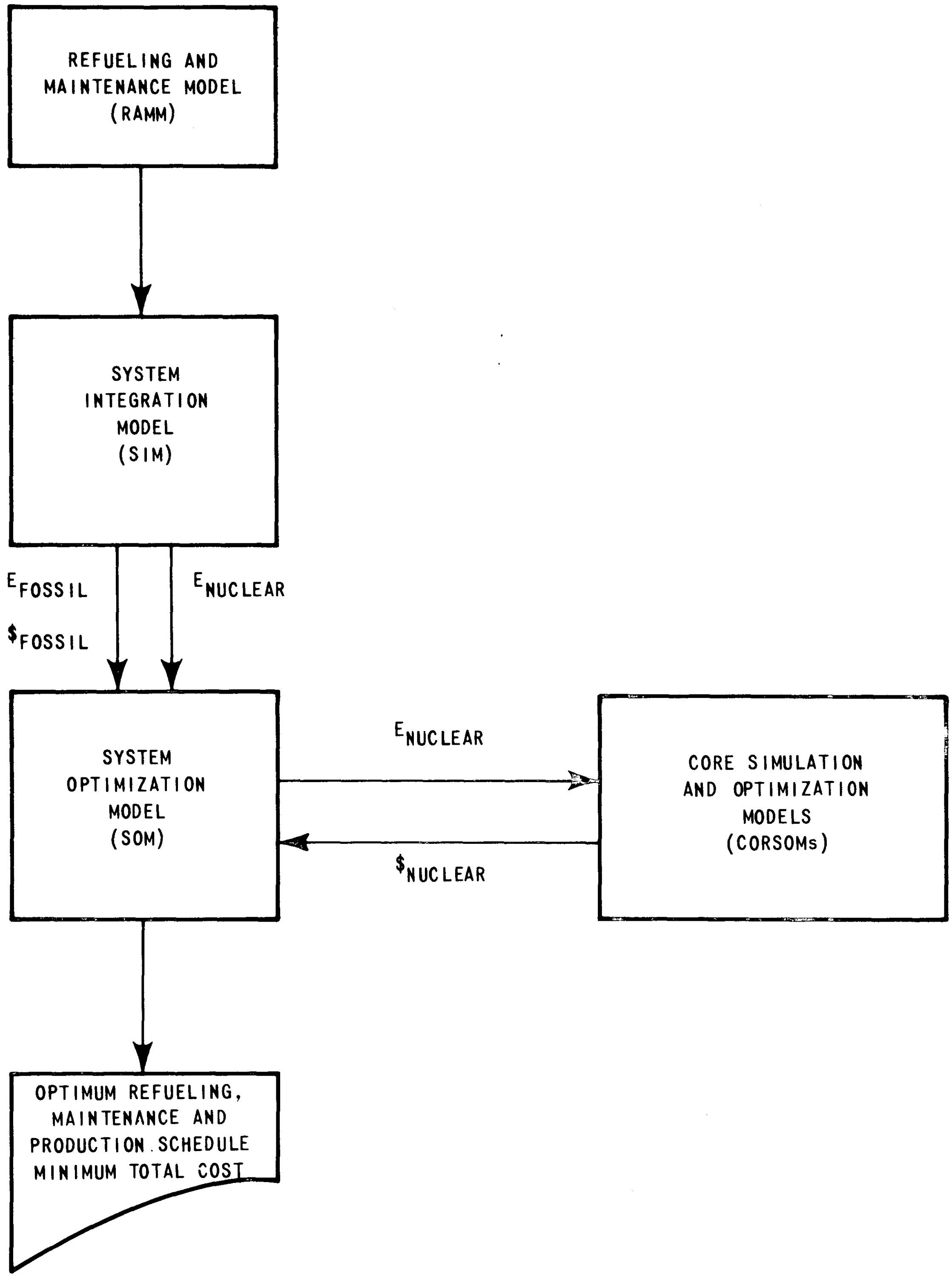
The net result was that cycle energies were no longer easily specified out to the horizon. The nuclear complications rendered previous utility system optimization models obsolete. The nuclear power management model put forth here was developed to provide a modern utility system optimization model capable of handling nuclear plants explicitly. In a utility system containing nuclear powered generating equipment, the planning of the fuel management must be optimized from the system demand viewpoint (cost to utility of supplying all customer loads), not an individual reactor supply viewpoint (cost to utility of supplying power from a particular reactor). The complex interaction between system load and incremental operating costs of the multiplicity of generating units available on a utility system must be considered in optimizing the two nuclear reload design variables--fuel enrichment and batch fraction. The result is that what may appear uneconomical for a particular reactor (e.g., refueling while energy potential remains in the core), may indeed be optimum for the overall system.

1.5 A Nuclear Power Management Multi-year Model

A nuclear power management multi-year model currently under development (23, 34, 41, 55) contains four sub-models as presented in Figure 1.3. The overall model's purpose is to supply the utility system planner with the following outputs:

Figure 1.3

Nuclear Power Management
Multi-Year Model



(1) Optimum schedule for fossil maintenance and nuclear refueling,

(2) Associated optimum production schedule and

(3) The resultant fuel requirements.

Operation of the overall model begins within the Refueling and Maintenance Model (RAMM). Incorporating such inputs as load forecasts, maintenance requirements and scheduling constraints, the RAMM determines a number of feasible multi-year refueling and maintenance schedules. Each schedule is a mutually exclusive, alternative mode of operating the entire system over the multi-year horizon. The purpose of the rest of the overall model is to determine which of the possible alternative strategies results in the minimum total operating revenue requirement, ORR.

The output of the RAMM is accepted by the System Integration Model (SIM) in the form of either a set of downtime dates for each unit on the system or a period-by-period (on the order of one to four weeks per period) maintenance schedule indicating which units are down in each period. Also helpful to the rest of the model is an _a priori_ RAMM ranking of the strategies in order of estimated desirability. That is, "ballpark" estimates by the RAMM of economics and reliability ought to indicate Strategy 1 is most likely to be optimum, while Strategy n (n~100), though feasible, is highly unlikely to be economically attractive and/or a reliable operating scheme. Such a ranking would decrease computing requirements for the overall model by permitting the detailed evaluation of only those strategies with a reasonable chance of competing for the optimum.

Strategy-by-strategy evaluation begins in the System Integration Model (SIM). For each strategy, the SIM integrates the utility's available equipment, operating practices, etc. into a realistic utility simulation model. Since nuclear incremental costs are much less than those of fossil units, production scheduling is optimized so as to meet customer load demand by maximizing nuclear energy and minimizing fossil energy and fossil cost.

The task of the System Optimization Model (SOM) is to then optimize the operation of the nuclear portion of the system (see Figure 1.3) so that the nuclear energy $E_{Nuclear}$ is produced at minimum cost, $\$_{Nuclear}$. To do this, the SOM postulates reactor-by-reactor multi-year production schedules which are then passed to Core Simulation and Optimization Models (CORSOM's) for each reactor unit or type (PWR, BWR, LMFBR, etc.). With each production schedule specified to the horizon, each CORSOM is then able to optimize its reload parameters of batch size and enrichment, minimizing the total fuel revenue requirement for the particular reactor. In addition, the CORSOM calculates nuclear incremental costs for each of the cycles.

With all reactors optimized for the given energy production schedules, the SOM begins a second iteration by using the CORSOM's incremental nuclear energy costs to postulate a better reactor-by-reactor multi-year production schedule.

At each iteration between SOM and the CORSOM's in Figure 1.3, each CORSOM accepts a new set of cycle energies (E's) for its reactor and, in point of fact, the same set of cycle lengths (T's) associated with the particular possible alternative strategy. After simulating core physics-depletion and optimizing the reload parameters (batch size and enrichment), only two specific types of information are returned to the SOM:

(1) the minimum total reactor fuel revenue requirement ($\overline{TC}_r$) and

(2) the $\lambda_{rc}(E_{rc})$ nuclear incremental cost curve for each reactor reload batch,

$$\lambda_{rc}(E_{rc}) = \frac{\partial \overline{TC}_r}{\partial E_{rc}} \qquad (1.9)$$

Specific information about the fuel designs is not needed by the SOM. As long as each CORSOM is properly matched with the reactor unit that it represents, the SOM does not care which units are PWR's, BWR's, HTGR's or fast breeders. Of course, management personnel need fuel design information and it must, therefore, be available in the printed output received directly from the CORSOM (at least, for the final fully-converged iteration).

Iterations between SOM and the CORSOM's continue until the system-wide production schedule converges (see Figure 1.3), giving minimum system nuclear cost $\$_{Nuclear}$. The total system cost for the particular refueling and maintenance strategy under investigation is then merely the sum of $\$_{Fossil}$ and $\$_{Nuclear}$.

After evaluating all possible alternative strategies in this manner, the overall optimum system strategy is the one resulting in the minimum total system operating revenue requirement ORR.

Though the above discussion and, in fact, this entire work assumes only fossil and nuclear equipment exist on the system, the general structure of the overall model holds even if hydro and pumped-hydro equipment have been installed.

The development of the complete nuclear power management multi-year model is a very large task. The four sub-models represent convenient building blocks suitable for somewhat independent development. However, model interface problems must be considered. Ideally, the models ought to be coupled together like the boxcars of a train, not nailed together like the tracks.

In the context of the Commonwealth Edison-sponsored utility system optizimation research project at the Massachusetts Institute of Technology, development of a RAMM was assumed by the project sponsor (20). Development of a pressurized water reactor CORSOM was undertaken at MIT by Kearney (41) and Watt (55). The work reported here deals specifically with the development of the remaining SIM and

SOM. In this regard, Figure 1.4 and the following sections describe these two models.

1.6 The System Integration Model (SIM)

The System Integration Model (SIM) has as its basic purpose the simulation of multi-year utility operation. To do this, it must integrate the following information into a representative utility system model:

(1) Forecasts of customer loads,

(2) Generating equipment characteristics,

(3) Forecasts of fuel costs,

(4) Maintenance schedules, and

(5) Operating constraints.

To portray system operation more accurately, the multi-year horizon is divided into much smaller time periods, on the order of a few weeks. Periods shorter than a week create an undue computational burden. On the other hand, periods longer than a month are precluded by the necessity of discretely representing scheduled maintenance outages which are usually two to four weeks in length.

These time periods are then simulated individually in chronological sequence. Forecasted loads for each period (Item 1 above) are represented by a normalized customer load-duration curve. Thermal energy costs (Item 3) are combined with the characteristics of the generating units to yield unit incremental costs. Any units unavailable due to scheduled maintenance (Item 4) are treated as non-existent for

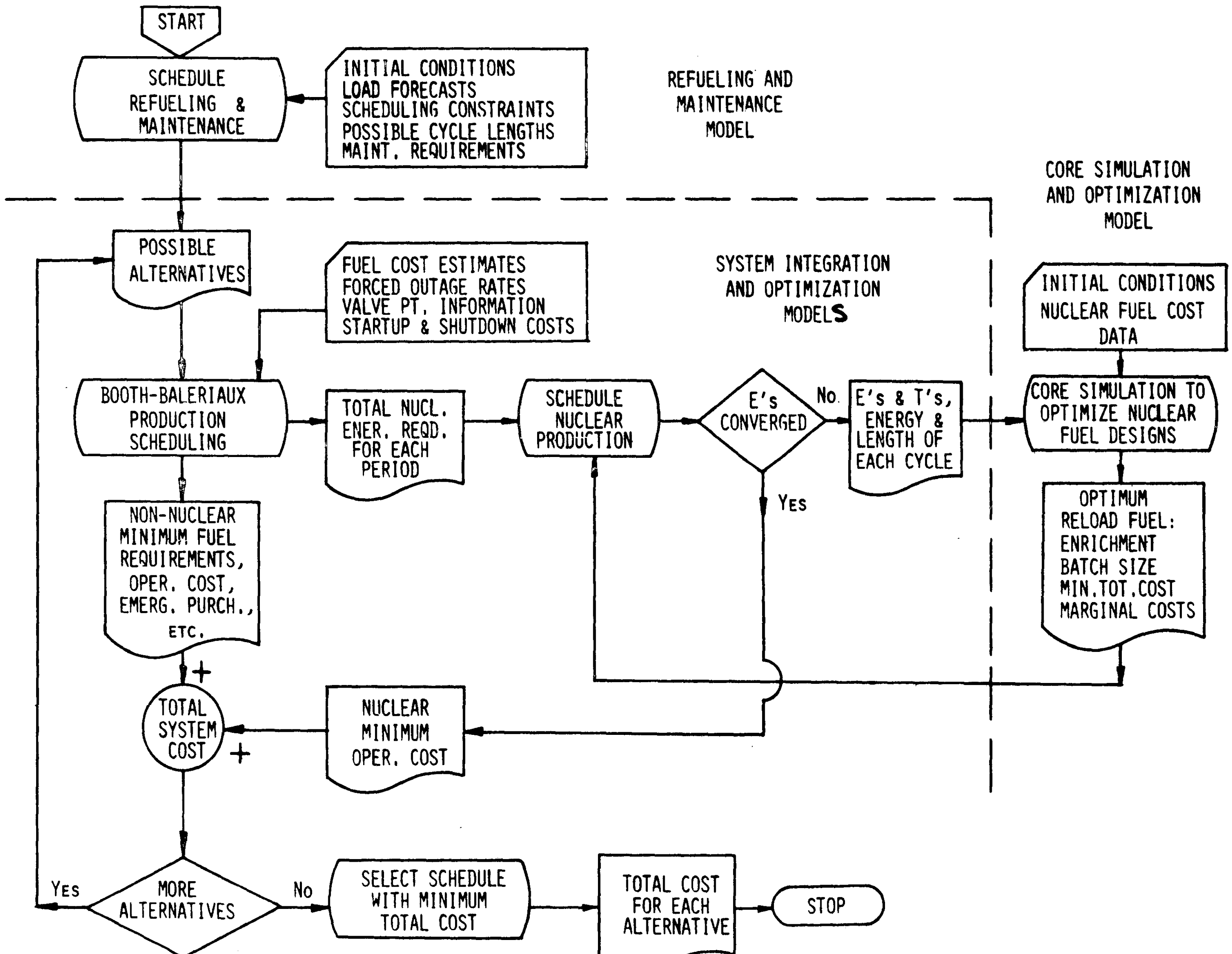


Figure 1.4 NUCLEAR POWER MANAGEMENT MULTI-YEAR MODEL

that period. The next step is the establishment of the startup and loading order for the remaining (on-line) units. It is in this order that various operating constraints (Item 5), such as "spinning reserve" and "zone-loading" requirements are incorporated. Production scheduling of the resulting system representation is performed using the Booth-Baleriaux (10, 19) probabilistic utility system model.

As pointed out earlier (see Section 1.4), the complexities of nuclear power preclude *a priori* knowledge of nuclear fuel costs except for the special case of all nuclear base-load operation. Nevertheless, by incorporating nuclear versus fossil incremental cost arguments (22) to sub-optimize each period, the SIM is able to mark time by calculating in its place, the system nuclear potential (demand) N for each period (a part of the horizon's total $E_{Nuclear}$). The responsibility for optimizing and costing intra-nuclear production of this energy rests with the System Optimization Model (SOM).

Thus, the actual period-by-period output of the SIM consists of:

(1) X_F = Fossil fuel expense related to energy production,

(2) N = Potential nuclear energy production,

(3) X_S = Combined fossil and nuclear startup-shutdown cost, and

(4) X_U = Expense related to emergency energy purchases.

1.6.1 Booth-Baleriaux Probabilistic Utility Simulation Model

The Booth-Baleriaux probabilistic utility simulation model is a recent adaptation of previous deterministic utility models with new emphasis on the field of applied probability theory. Though the original 1967 paper on the subject is a product of Baleriaux, et al., (10) of Belgium, Booth (17-19) of Australia deserves much of the credit for introducing and promoting the model in the United States.

Previous papers reporting on the Booth-Baleriaux model, including the work of Joy and Jenkins (39), have closely followed the development in the original paper. With due respect to these ground-breaking efforts, the following presentation leads to computational savings in terms of time and storage, and also follows a more direct line of reasoning.

The Booth-Baleriaux probabilistic utility model is based on the concept of equivalent system load which embodies not only direct customer demands on a particular unit, but also the indirect demands left unsatisfied by previously loaded units when they are on forced-outages.

The equivalent load P_e may be defined as

$$P_e \equiv P_D + P_O \qquad (1.10)$$

where

P_D = actual direct customer load demand, MW

P_O = system capacity on forced-outage that would be generating energy otherwise, MW

Capacity that is on forced-outage during what would otherwise have been reserve (i.e., economy) shutdown hours anyway is not counted since the outage does not affect system generating operations.

In a probabilistic sense, P_D is a random variable with a complementary cumulative distribution given by $F_D(P_D)$, the normalized customer load-duration curve. Since forced-outages are random, P_O is also a random variable characterized by the performance probabilities of each unit. Thus, P_e is also a random variable and the computation of its complementary cumulative distribution (the equivalent load-duration curve) $F_e(P_e)$ involves the convolution (26) of the distributions of P_D and P_O. The heuristic presentation here is limited to the common two-state model of forced-outages:

State 1: With performance probability p, the unit will perform at any output up to its rated capacity when called upon, and

State 2: With non-performance probability q, the unit will not perform at all when called upon.

Thus,

$$p + q = 1 \qquad (1.11)$$

In accounting for the forced-outages of all of the utility's available generating units (i.e., those not down anyway due to scheduled outages), the approach presented in this work performs the system-wide convolution by sequentially incorporating each unit's contribution to the equivalent load. Referring to Figure 1.5, the general equation for convolving up to the i th increment of unit r into the equivalent load-duration distribution F_{ri}^{wo} can be shown to be as follows,

$$F_{ri}^{w}(P_e) = p_r \cdot F_{ri}^{wo}(P_e) + q_r \cdot F_{ri}^{wo}(P_e - K_{ri}) \quad \text{for all } P_e \tag{1.12}$$

where

F_{ri}^{w} = Equivalent load distribution with the forced-outages of i increments of unit r included.

F_{ri}^{wo} = Equivalent load distribution without the forced-outages of i increments of unit r included

K_{ri} = rated capacity of unit r up to and including i th increment, i.e., magnitude of forced-outage included in P_e when forced-outage occurs (q_r fraction of the time), MW

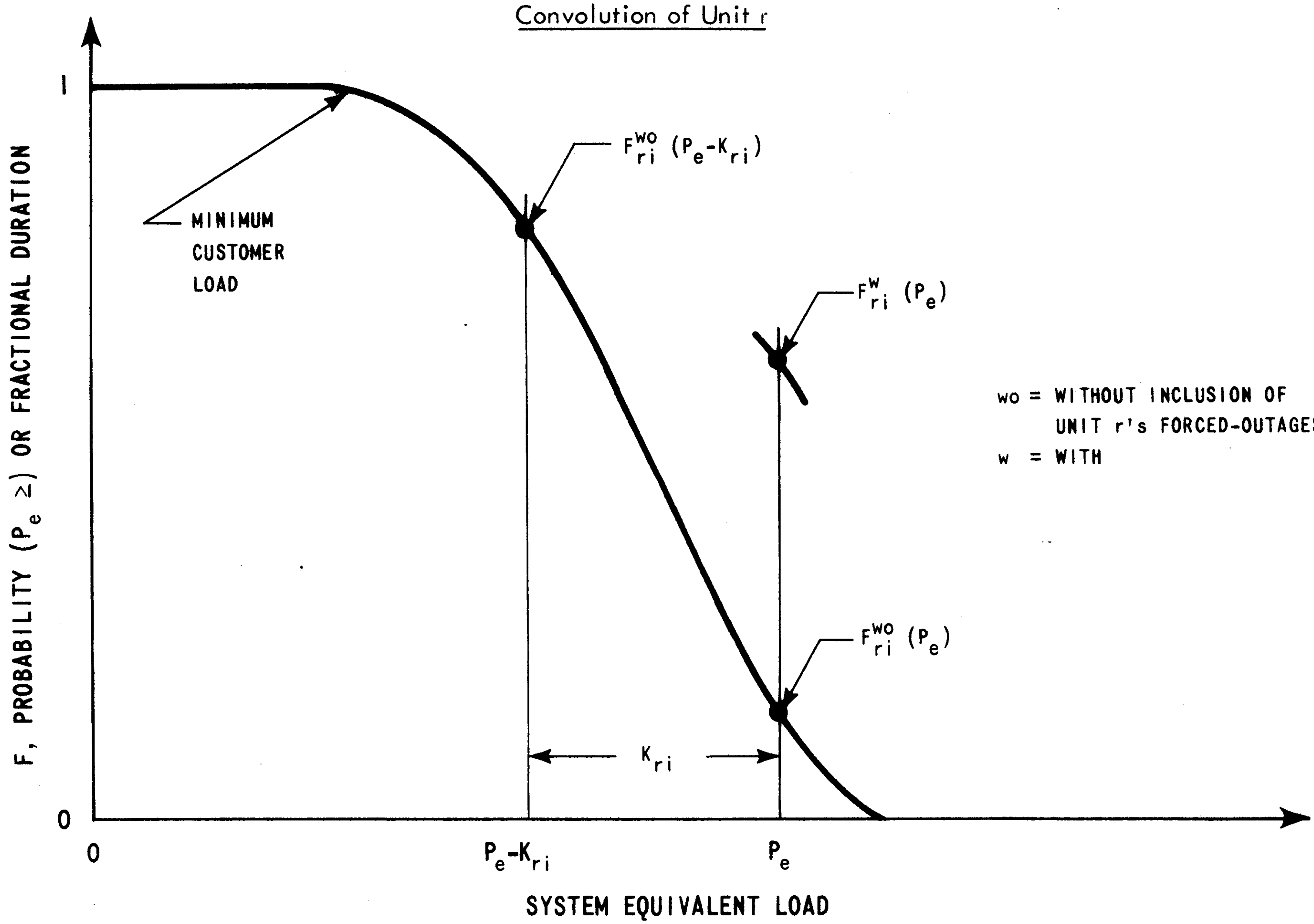
p_r = performance probability of unit r

q_r = $1-p_r$

Due to Equation (1.10), K_{ri} may be less than the K_r maximum rated capacity of unit r because the rest of the unit's capacity is not being used whether on forced-outage or not.

Figure 1.5

Convolution of Unit r



6253-81

Since Equation (1.12) is valid for all P_e, (not merely the single value shown in Figure 1.5), the complete $F^{w}_{ri}(P_e)$ curve can be calculated easily. Two limiting cases are readily apparent. One case is P_e less than the minimum load--each $F^{wo}_{ri}=1$, as does the resulting $F^{w}_{ri}(P_e)$. For very large P_e, each $F^{wo}_{ri}=0$ and, hence, $F^{w}_{ri}(P_e)=0$. Equation (1.12) is the heart and soul of the Booth-Baleriaux model. All subsequent calculations involving F, whether convolutions or deconvolutions (see below) are merely rearrangements of it.

Deconvolution merely refers to reversing the convolution process, subtracting unit r's forced-outages from the equivalent load. That is, given $F^{w}_{ri}(P_e)$, determine $F^{wo}_{ri}(P_e)$. The necessity of performing deconvolutions comes about because:

(1) entire units are not scheduled as single blocks of capacity but as smaller capacity increments due to units' varying incremental costs, and

(2) during the production calculation (see below), increments of the same unit cannot possibly make up for each other's forced-outages since they are all forced offline together (at least, in the simple two-state forced-outage model).

Rearranging Equation (1.12) to the following, deconvolution is accomplished thusly,

$$F^{wo}_{ri}(P_e) = \frac{1}{p_r}\left[F^{w}_{ri}(P_e) - q_r\, F^{wo}_{ri}(P_e - K_{ri})\right] \qquad (1.13)$$

Making use of the fact that $F_{ri}^{wo}(P_e) = 1$ for P_e less than the minimum load, $F_{ri}^{wo}(P_e)$ can be "boot-strapped" from right to left in Figure 1.5 to determine the complete F_{ri}^{wo}.

As illustrated in Figure 1.6, forced-outages of units lower in the loading order increase the demand or duration of load $[F_{ri}^{wo}(P_e) > F_D(P_e)]$ to be satisfied by capacity increments higher in the loading order. However, forced-outages affect not only the demand F_{ri}^{wo} on each increment, but also the increment's energy production E_{ri}. If the unit only performs 90% of the time, then it is expected that only 90% of the production demanded from it will be served. Recalling that p_r is the unit's performance probability, the increments' expected energy production for the period is given by,

$$E_{ri} = T' p_r \int_{P_{ri}^{\circ}}^{P_{ri}^{\circ} + \Delta K_{ri}} F_{ri}^{wo}(P_e)\, dP_e \tag{1.14}$$

where

T' = duration of time period, hours

ΔK_{ri} = i th increment of capacity of unit r, MW

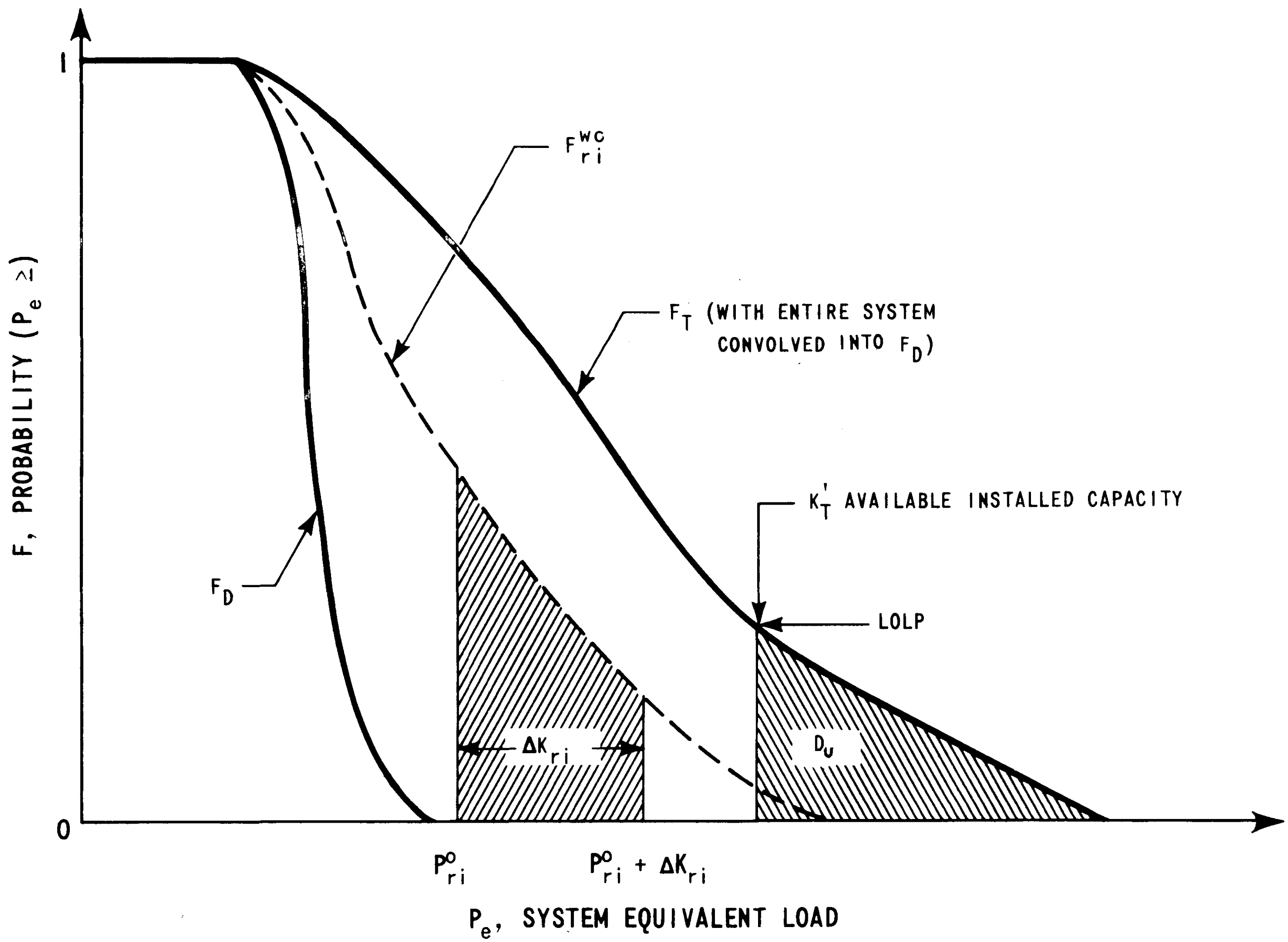
P_{ri}° = system equivalent load when increment i first loaded, i.e., the increment's loading point.

Total unit energy production for the period, E_r, is given by summing E_{ri} over the unit's I increments,

$$E_r = \sum^{I} E_{ri} \tag{1.15}$$

Figure 1.6

Determining Energy Demand on Increment i of Unit r



6253-82

At an average cost of $\bar{e}_{r1}$ for the first increment and incremental costs λ_{ri} for the other increments, the cost of each energy increment is

$$X_{r1} = \bar{e}_{r1} E_{r1} \tag{1.16}$$

$$X_{ri} = \lambda_{ri} E_{ri} \quad \text{for } i > 1 \tag{1.17}$$

and, hence, period production fuel expense X_r for unit r is given by

$$X_r = \sum^{I} X_{ri} \tag{1.18}$$

Recall from Section 1.6, that for nuclear units, the SIM's required period output is not cost, but the system nuclear potential N,

$$N = \sum^{\text{NUCL. UNITS}} E_r \tag{1.19}$$

In Figure 1.6, notice that for the final total system curve, F_T, some indirect customer demand extends beyond the available installed (on-line) capacity,

$$K_T' = \sum^{\text{ON-LINE UNITS}} K_{rI} \tag{1.20}$$

As one measure of system reliability, D_U represents the energy unserved by the system's resources (i.e., wholly owned capacity plus firm purchases),

$$D_U = T' \int_{K'_T}^{\infty} F_T(P_e)\, dP_e \qquad (1.21)$$

"Expected unserviced energy . . . is the expected curtailment or, more realistically, the expected emergency support required during" the time period (49). The determination of the X_U expenditure relative to the D_U emergency electricity purchases from neighboring utilities is straightforward given an $\bar{e}_U$ average cost for this emergency support. The period expenditure is merely,

$$X_U = \bar{e}_U D_U \qquad (1.22)$$

Along with D_U, another measure of the system's reliability is the LOLP "loss-of-load-probability,"

$$LOLP = F_T(K'_T) \qquad (1.23)$$

the fraction of time the utility is unable to serve its customers with its own resources.

With production scheduling completed, only the task of determining the startup-shutdown cost component for the period remains. To accurately calculate the period's X_S, startup-shutdown cost, an hour-by-hour production scheduling model would be required. Having sacrificed detailed chronological load shapes for the more convenient load-duration curves covering much longer periods of time, shutdown costs must be estimated by an approximate technique.

Consider Figure 1.7 [after (18)] which displays qualitatively the approximate relation between Ω, the frequency of startup-shutdowns (per day) and L'_{r1}, the availability-based capacity factor for the unit's first capacity increment. That is,

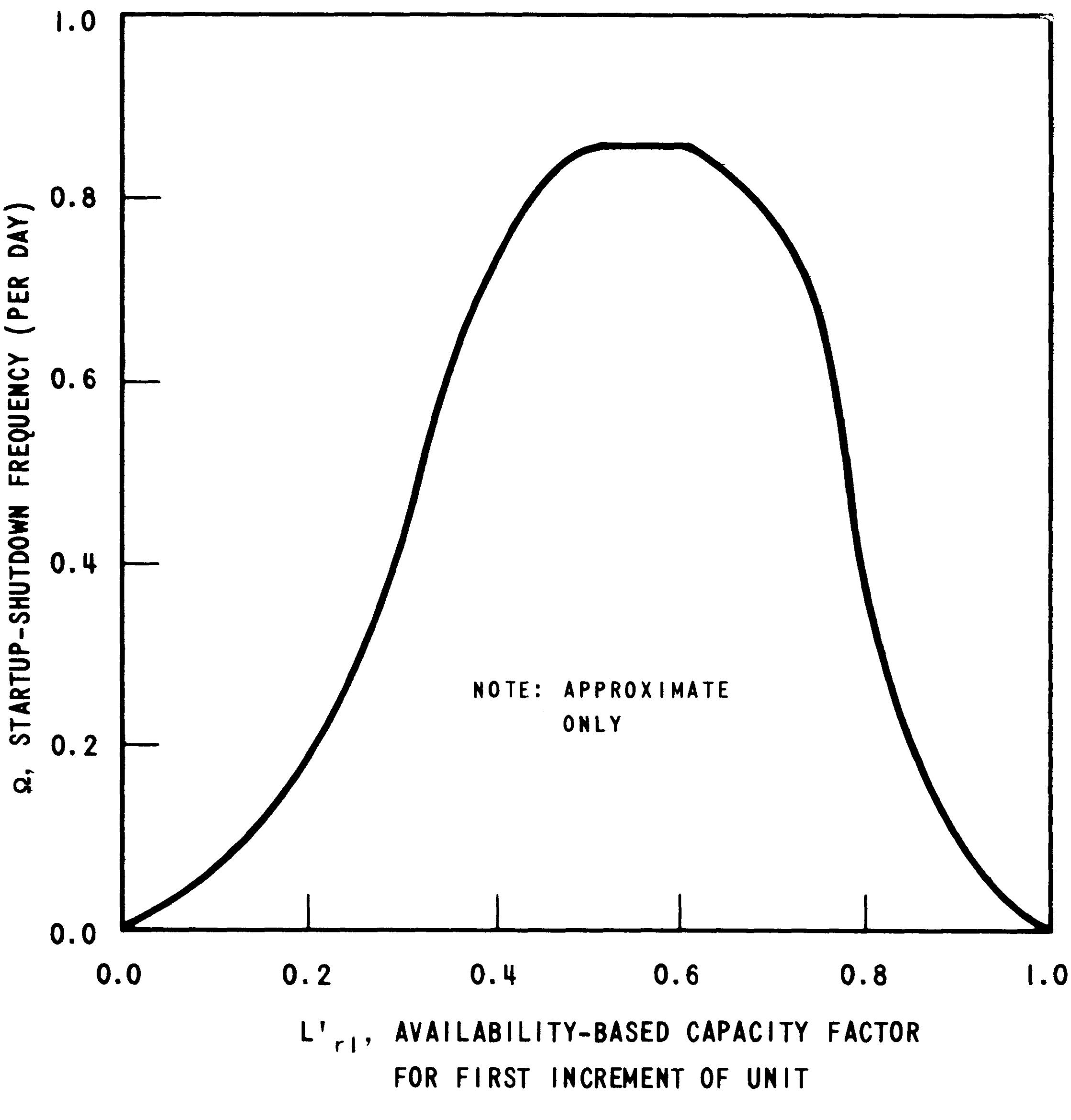
$$L'_{r1} = \frac{1}{K_{r1}} \int_{P^\circ_{r1}}^{P^\circ_{r1} + K_{r1}} F^{w\sigma}_{r1}(P_e)\, dP_e \qquad (1.24)$$

For must-run units, L'_{r1} equals 1 and Ω equals 0. For very expensive peaking units, L'_{r1} approaches 0 and Ω again approaches 0. As expected, units never shutdown and units never started-up incur no startup-shutdown cost. In between are those units started-up and shutdown on a daily basis and, hence, Ω approaches one.

If unit startup-shutdown cost Q_r is specified in time independent units of equivalent thermal energy input, multiplying it by ϕ_r, the unit's thermal energy cost for the period,

Figure 1.7

Example of Startup-Shutdown Frequency versus Availability-Based Capacity Factor [After (18)]



permits escalation in terms of undiscounted dollars. Since L'_{r1} is easily extracted for each unit during the Booth-Baleriaux simulation, the fractional starts per day are easily estimated given the proper dependence of Ω upon L'_{r1}. Thus, a period T'/24 days long, incurs total period startup-shutdown cost amounting to

$$X_S = \frac{T'}{24} \sum^{R} \phi_r Q_r \Omega(L'_{r1}) \quad (1.25)$$

1.6.2 SYSINT, A Computerized Version of the SYStem INTegration Model

SYSINT, a 2000 card Fortran IV version of the SYStem INTegration Model is detailed in Appendix E. This section merely summarizes its capabilities.

The standard two-state forced-outage model (perform or not perform) is employed. A single startup frequency curve Ω (L'_{r1}) is input for the entire horizon. The limitations of the current version, though easily altered, are as follows:

(1) up to 100 units (including retirements and additions),

(2) up to 5 valve points for each unit,

(3) no limit on number of strategies per computer run,

(4) up to 100 time periods per strategy and

(5) up to 25 typical load-duration "shapes," stored in completely normalized form (i.e., peak demand also equals one.)

The multi-period strategy is input for each unit in the following form:

(1) the period installed,

(2) period just prior to retirement and

(3) up to 20 intermediate periods of downtime for maintenance or refueling.

For each period the following data may be input or altered:

(1) Choice of load-duration shape,

(2) Forecasted peak demand,

(3) Expected spinning reserve requirement,

(4) Length of time period,

(5) Average cost of emergency purchase energy,

(6) Fuel cost for each unit (optional initial guess for nuclear units),

(7) Performance probability for each unit, and

(8) Startup order indicating must-run units and peaking equipment.

As for typical running time, each period of a simulation of a utility system containing 40 units with a total of 150 valve points requires approximately 2.5 CPU sec on an IBM 370 Model 155 computer operating in an MVT environment. The code itself requires 108 K bytes of storage, i.e., not including the computer system supervisor. Total core requirements are thus approximately 134 K bytes.

Data transfer from SYSINT to SYSOPT (see Section 4.6 and Appendix F) is completely automated via either disk, magnetic tape or punched cards.

1.7 System Optimization Model (SOM)

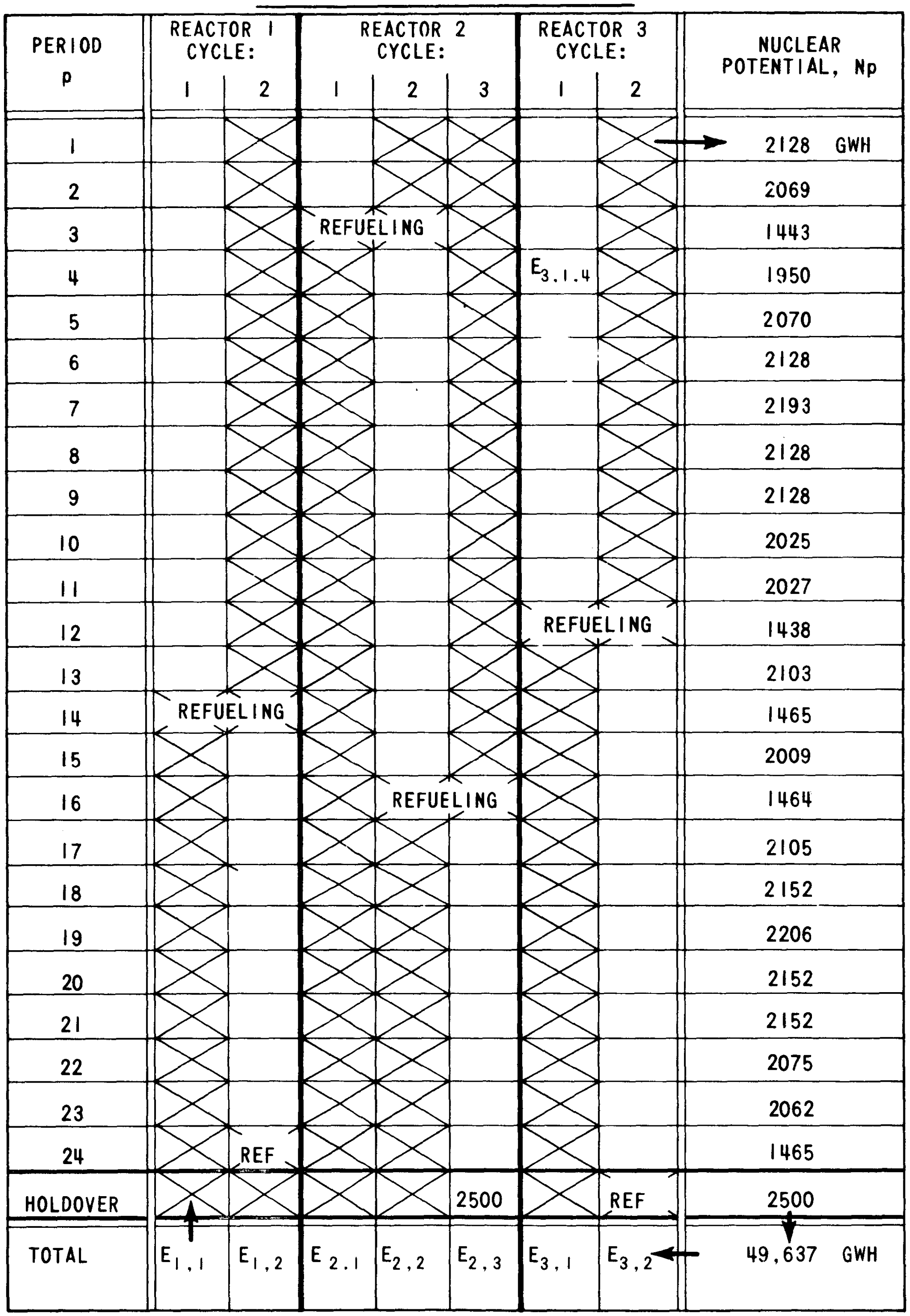
The SOM receives period-by-period information from the SIM relative to the system nuclear energy production potential and each reactor's possible maximum (i.e., if it is the first nuclear unit to be loaded) and minimum (i.e., if last nuclear unit) contribution to it. In addition, the non-nuclear cost totals are entered and later discounted at the appropriate present value rate to yield the total non-nuclear revenue requirement. Optimization itself (see Figure 1.4) begins by utilizing any initial nuclear fuel cost estimates to schedule period-by-period, reactor-by-reactor energy production using network programming (NP).

1.7.1 Nuclear Supply Network Optimization

Since the optimization within the SOM deals with a single commodity (nuclear energy production) in a strict one-to-one (reactor) supply and (customer) demand sense, the production constraints form a (nuclear energy) supply network. Figure 1.8 presents such a network configuration for a 3 reactor, 24 period (month) example. Numbers are displayed for the nuclear potentials N_p to emphasize the fact that these are fixed constraints throughout all of the iterations for a particular refueling and maintenance strategy. Nuclear energy is allocated (i.e., supplied) to each reactor-cycle (E_{rc}). Within each cycle, this energy is allocated to the pertinent

Figure 1.8

Sample Network Configuration



periods (E_{rcp}) so as to satisfy the system nuclear potentials (i.e., demanded).

The objective function for the nuclear supply network optimization is the system nuclear fuel revenue requirement ,

$$\text{minimize } RR_N \equiv \overline{TC} = \sum^{\text{all } r} \overline{TC}_r(E_{r1}, E_{r2}, \ldots) \qquad (1.26)$$

Due to the nonlinearity of Equation (1.26) as discussed in Section 1.4, an iterative gradient optimization technique known as the "method of convex combinations" (54) is employed. With the gradient defined as λ_{rc}, the incremental cost (revenue requirement) of extracting an additional amount of energy in cycle c of reactor r, then

$$\lambda_{rc} = \frac{\partial \overline{TC}_r}{\partial E_{rc}} \qquad (1.27)$$

Denoting the iteration or trials by the superscript t, a Taylor expansion of the objective function about the "current" t set of reactor-cycle energies yields,

$$\text{minimize } \overline{TC}^{t+1} = \overline{TC}^{t} + \sum^{\text{all } r} \sum^{\text{all } c} \int_{E_{rc}^{t}}^{E_{rc}^{t+1}} \lambda_{rc}^{t}(E_{rc})\, dE_{rc} \qquad (1.28)$$

Thus, given the information at the t th iteration, the next iteration determines the t+1 set of E_{rc} so that the double summation term of Equation (1.28) is minimized subject to the constraints indicated in Figure 1.8. Specifically, the sum of any column must equal the energy supplied (or extracted) during that particular reactor-cycle,

$$E_{rc} = \sum^{p \text{ in } c} E_{rcp} \quad \text{for all r and all c} \qquad (1.29)$$

At the same time, the sum of any row must equal the period's required nuclear potential,

$$N_p = \sum^{\text{all } r} E_{rcp} \quad \text{for all p} \qquad (1.30)$$

The range of each E_{rcp} is also constrained ("capacitated") via

$$E_{rcp}^{min} \leq E_{rcp} \leq E_{rcp}^{max} \quad \text{for all r and all p} \qquad (1.31)$$

which is indicative of the minimum and maximum demand in the equivalent load range served by the nuclear units. Representative E_{rcp}^{min} and E_{rcp}^{max} for each E_{rcp} in Figure 1.8 are presented in Table 1.2.

At each iteration, the E_{rc} cycle energy production requirements are passed to the CORSOM's which design the fuel reload batches (batch size and enrichment) to meet the

Table 1.2

Reactor Production Limits for 3 Reactor, 24 Period Example

Period p	Reactor 1 E^{min}_{1cp}	Reactor 1 E^{max}_{1cp}	Reactor 2 E^{min}_{2cp}	Reactor 2 E^{max}_{2cp}	Reactor 3 E^{min}_{3cp}	Reactor 3 E^{max}_{3cp}
1	669	762	629	722	669	762
2	635	760	596	720	635	760
3	687	756	0	0	687	756
4	577	747	540	707	577	747
5	636	760	596	720	636	760
6	669	762	629	722	669	762
7	714	763	674	723	714	763
8	669	762	629	722	669	762
9	669	762	629	722	669	762
10	616	755	577	714	616	755
11	610	759	571	718	610	759
12	718	760	678	720	0	0
13	656	761	617	721	656	761
14	0	0	703	722	743	763
15	610	752	571	712	610	752
16	706	758	0	0	706	758
17	657	761	617	721	657	761
18	686	762	646	722	686	762
19	724	763	684	723	724	763
20	686	762	646	722	686	762
21	686	762	646	722	686	762
22	643	758	604	718	643	758
23	632	759	593	719	632	759
24	0	0	703	722	743	763

All E_{rcp} in GWH

production schedule and refueling dates at minimum reactor cost. Information returned to the SOM is minimum total reactor nuclear fuel revenue requirement $\overline{TC}_r$ (for later summation of total system nuclear costs) and the nuclear incremental cost curve of each reload batch,

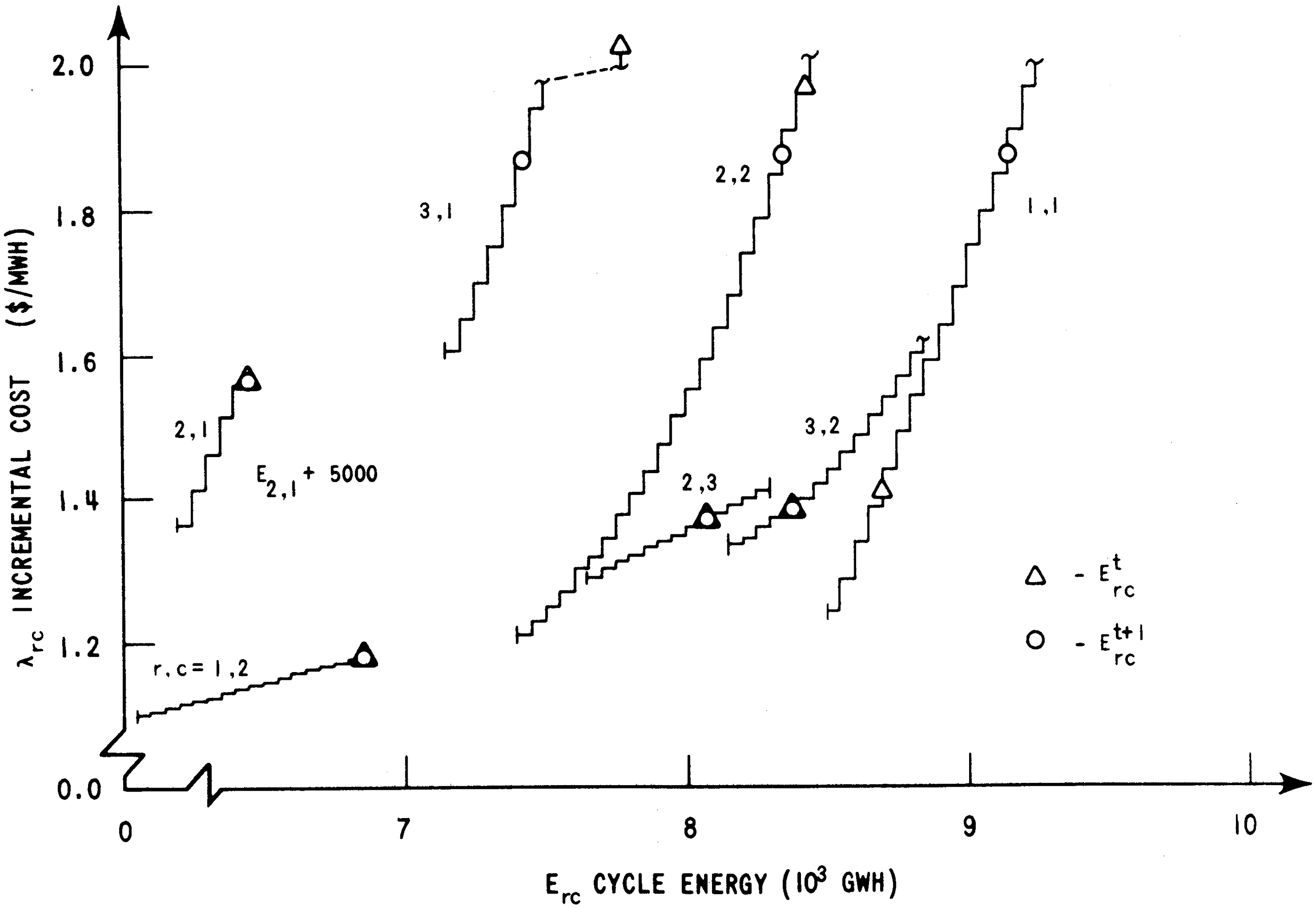
$$\lambda_{rc}(E_{rc}) = \frac{\partial \overline{TC}_r}{\partial E_{rc}} \qquad (1.32)$$

With these incremental costs, the network algorithm reoptimizes nuclear production in order to minimize the objective function [Equation (1.28)]. The result is that all nuclear reload batches are designed at the same incremental cost within the limits of availability and loads (22).

To illustrate a single iteration, consider the 3 reactor, 24 period example of Figure 1.8 and Table 1.2. Figure 1.9 presents a hypothetical set of incremental cost curves returned to the SOM at the end of the previous iteration. The "stair-step" nature of the curves is indicative of the piecewise-linearization of $\overline{TC}$ required to cast the double summation term in Equation (1.28) in an NP format. Note that the NP program effectively seeks to establish equal incremental costs among the reactor-cycles that compete for the nuclear potential (e.g., at the optimum, $\lambda^*_{1,1} = \lambda^*_{2,2} = \lambda^*_{3,1}$). Figure 1.10 presents the complete, optimized period-by-period reactor production schedule for this example.

Figure 1.9

Hypothetical Set of Incremental Cost Curves



6253-44

Figure 1.10

Sample Reactor Production Schedule

PERIOD p	REACTOR I CYCLE: 1	REACTOR I CYCLE: 2	REACTOR 2 CYCLE: 1	REACTOR 2 CYCLE: 2	REACTOR 2 CYCLE: 3	REACTOR 3 CYCLE: 1	REACTOR 3 CYCLE: 2	NUCLEAR POTENTIAL, Np
1	715		722			691		2128 GWH
2	697		720			652		2069
3	722		REFUELING			721		1443
4	661			707		582		1950
5	697			720		653		2070
6	715			722		691		2128
7	738			723		732		2193
8	715			722		691		2128
9	715			722		691		2128
10	685			714		626		2025
11	684			672		671		2027
12	738			700		REFUELING		1438
13	668			674			761	2103
14	REFUELING			703			762	1465
15		752		571			686	2009
16		758		REFUELING			706	1464
17		761			687		657	2105
18		762			704		686	2152
19		763			719		724	2206
20		762			704		686	2152
21		762			704		686	2152
22		758			674		643	2075
23		759			671		632	2062
24		REF			722		743	1465
HOLDOVER					2500		REF	2500
TOTAL	9150	6837	1442	8350	8085	7401	8372	49,637 GWH

In addition to the above network constraint Equations (1.30) and (1.31), which are special cases of linear constraints and can therefore be handled easily by a standard NP code (45), a nonlinear constraint for each period must also be incorporated. In particular, after the iterations are complete, a check must be made to ensure that the optimum E_{rcp} reactor-period energy productions are compatible, or feasible, with regard to shape of the period's equivalent load curve. As illustrated in Figure 1.11, even though Equation (1.30) is satisfied, the set of energy productions for the four nuclear units is not feasible. Within that segment of the equivalent load curve preassigned to the nuclear units (i.e., after the must-run fossil units), the low minimum load permits only one unit A or B to operate as a base-load unit.

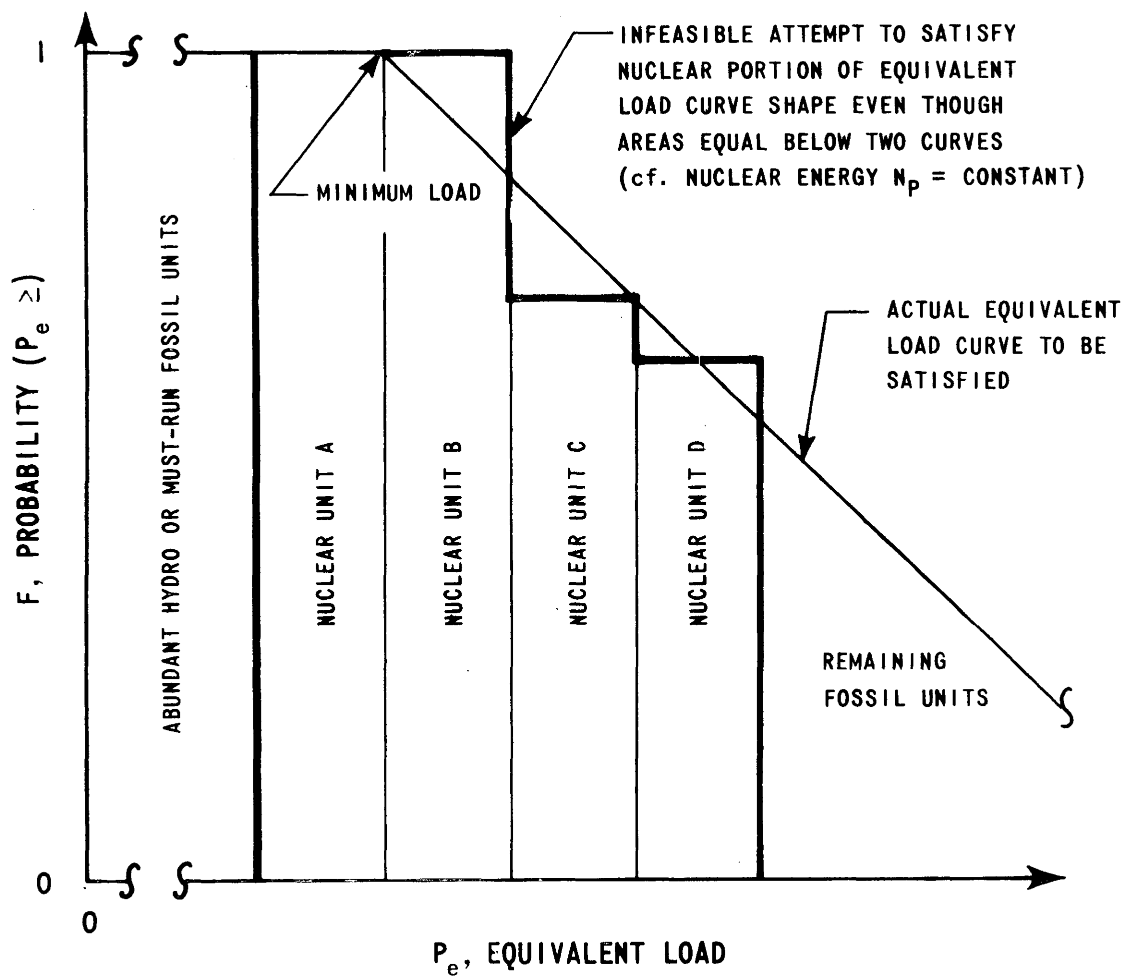
In order to account for this feasibility problem, a shape constraint (similar to a least-squares fitting criterion) was derived that of necessity, included second-order terms in E_{rcp},

$$\sum_{\substack{\text{avail.}\\ \text{units}}} c_{1_{rp}} \cdot E_{rcp} + \sum_{\substack{\text{avail.}\\ \text{units}}} c_{2_{rp}} \cdot E^2_{rcp} \leq c_p \qquad (1.33)$$

The $c_{1_{rp}}$, $c_{2_{rp}}$ and c_p are constants for each reactor r in period p, precalculated by the SOM using the nuclear segment

Figure 1.11

Example of Infeasible Equivalent Load Shape



of the actual equivalent load curve and the performance characteristics of the various nuclear units.

As mentioned above, the nonlinear shape constraint is implemented as a posterior check on the optimized reactor-period production schedules. For each period violating the shape constraint Equation (1.33), the E_{rcp}^{min} and E_{rcp}^{max} of each reactor's production constraint Equation (1.31) are "squeezed" slightly toward their mean so that infeasible schedules (such as in Figure 1.11) are unlikely to occur in that period again. After checking and adjusting the production constraints for all infeasible periods, the revised network is again optimized. Such shape iterations continue until all periods of an optimized schedule satisfy their respective shape constraint.

When iterative convergence and feasibility of the production schedule is realized, overall fossil-nuclear system operation has been optimized for the particular possible alternative maintenance and refueling schedule under investigation.

With the optimization task completed, the resulting (minimum) $\overline{TC}$* represents the total revenue requirement for nuclear fuel RR_N. By present-valuing all of the other period expenditures (received as input from the SIM) according to Equation (1.3), the determination of ORR is complete,

$$ORR = RR_N + \sum^{Z} \frac{1}{(1+x)^{\bar{t}_p}} \left(X_{F_p} + X_{S_p} + X_{U_p} \right) \qquad (1.34)$$

The ORR operating revenue requirement is appropriately stored for later comparison with that of other possible alternative strategies. With the completion of this task, processing of the particular alternative refueling and maintenance strategy is complete. And with completion of the last alternative strategy, selection of the minimum ORR cost strategy becomes possible.

1.7.2 SYSOPT, A Computerized SYStem OPTimization Model

SYSOPT, a 2100 card Fortran IV version of the SYStem OPTimization Model is detailed in Appendix F. SYSOPT is link-edited with the Out of Kilter Network Program (45) which represents an additional 1200 cards in Fortran IV and Assembler Language. Out of Kilter is detailed in Appendix G. This section merely summarizes the capabilities of the current combined version of SYSOPT.

The limitations of the current version of SYSOPT, though easily altered, are as follows:

(1) up to 15 reactors,

(2) up to 15 cycles per reactor within the horizon,

(3) up to 3 cycles per reactor beyond the horizon,

(4) no limit on number of strategies per computer run, and

(5) up to 100 periods per strategy.

Input data for each strategy includes:

(1) Present value rate,

(2) Various convergence criteria, and

(3) Maximum number of iterations to be permitted.

Input data supplied manually for each reactor includes:

(1) Optional initial estimates of λ^*_{rc} or E^*_{rc},

(2) Holdover energy at end of planning horizon, and

(3) Cycle energies and refueling dates beyond planning horizon.

The large volume of SYSINT output required by SYSOPT may be passed either on disk, magnetic tape or punched cards.

As for typical running times on an IBM 370 Model 155 computer (MVT environment), a hypothetical six reactor utility required only 9 CPU seconds per inner iteration (exclusive of time spent in CORSOM's) for strategies 72 periods long and totaling 30 reactor-cycles. The SYSOPT code itself requires 130 K bytes of storage (plus ~26 K for computer supervisor), while the Out-of-Kilter Network Program requires an additional 135 K. Using an overlay structure reduces the 265 K total to 200 K. Execution time is not noticeably increased by the use of the overlay structure.

1.8 Model Evaluation

To properly evaluate the SIM and SOM (or more specifically, the computerized versions SYSINT and SYSOPT, respectively), required interfacing them with a RAMM and CORSOM's to complete the nuclear power management multi-year model of Figure 1.3.

For the purposes of developing and testing a SIM and SOM, the multitude of possible alternative strategies output by a

RAMM were replaced by a few typical strategies developed through simple hand calculations. On the other hand, the on-line iterative nature of the optimization procedure requires computerized CORSOM's. The state of the art, as witnessed by the concurrent methods development research by Kearney (41) and Watt (55), precluded utilization of an established multi-year CORSOM. In order to proceed with the testing of the SIM and SOM, QKCORE, a psuedo-one dimensional, quick core model (performing simulation only), was developed (see Appendix H). The nature of QKCORE necessarily limited the scope of the evaluation to LWR's with the following characteristics:

(1) Modified-scatter refueling with fixed number of zones (e.g., refueling fraction was fixed at one-third),

(2) No plutonium recycle,

(3) No stretchout beyond reactivity-limited energy, and

(4) No cycle-to-cycle optimization (i.e., at each refueling, minimum enrichment chosen regardless of future cycles).

To evaluate the model's usefulness, several sample cases were calculated. An electric utility possessing six 1050 MW PWR's on a 46-unit 11,000 MW system was hypothesized. Minimum customer loads (typically 4000 MW), combined with other system operating constraints, restricted average nuclear availability-based capacity factors to about 80 per cent, i.e., below base-load operation.

Three possible refueling strategies were investigated:

S-1: strictly annual refuelings

S-2: gradual shift to longer (14 month) cycles

S-3: immediate shift to the longer cycles with additional cost of one million dollars for each short notice enrichment change.

Underlying later discussion of the choice from among the several optimized strategies are the properties of the individual strategies themselves. The important numerical properties are convergence, incremental costs and computational requirements. The results (see Table 1.3) of Strategy 2 over a six year horizon will be used for most of the discussion. However, when this Strategy fails to clearly demonstrate a point under discussion, one of the other two will be utilized.

1.8.1 Convergence

Starting from a relatively poor initial guess of equal energy in each cycle regardless of cycle length, the optimization of S-2 required ten cost iterations to converge to the initial optimum $\overline{TC}^*$. The iteration-by-iteration system nuclear fuel cost $\overline{TC}^t$ (i.e., the objective function of the optimization) in presented in Figure 1.12. Since initially 50% of the 72 periods failed their shape constraint, three more iterations were required to produce the feasible optimum. This resulted in a cost increase of only 0.25 (out of nearly 300) million dollars.

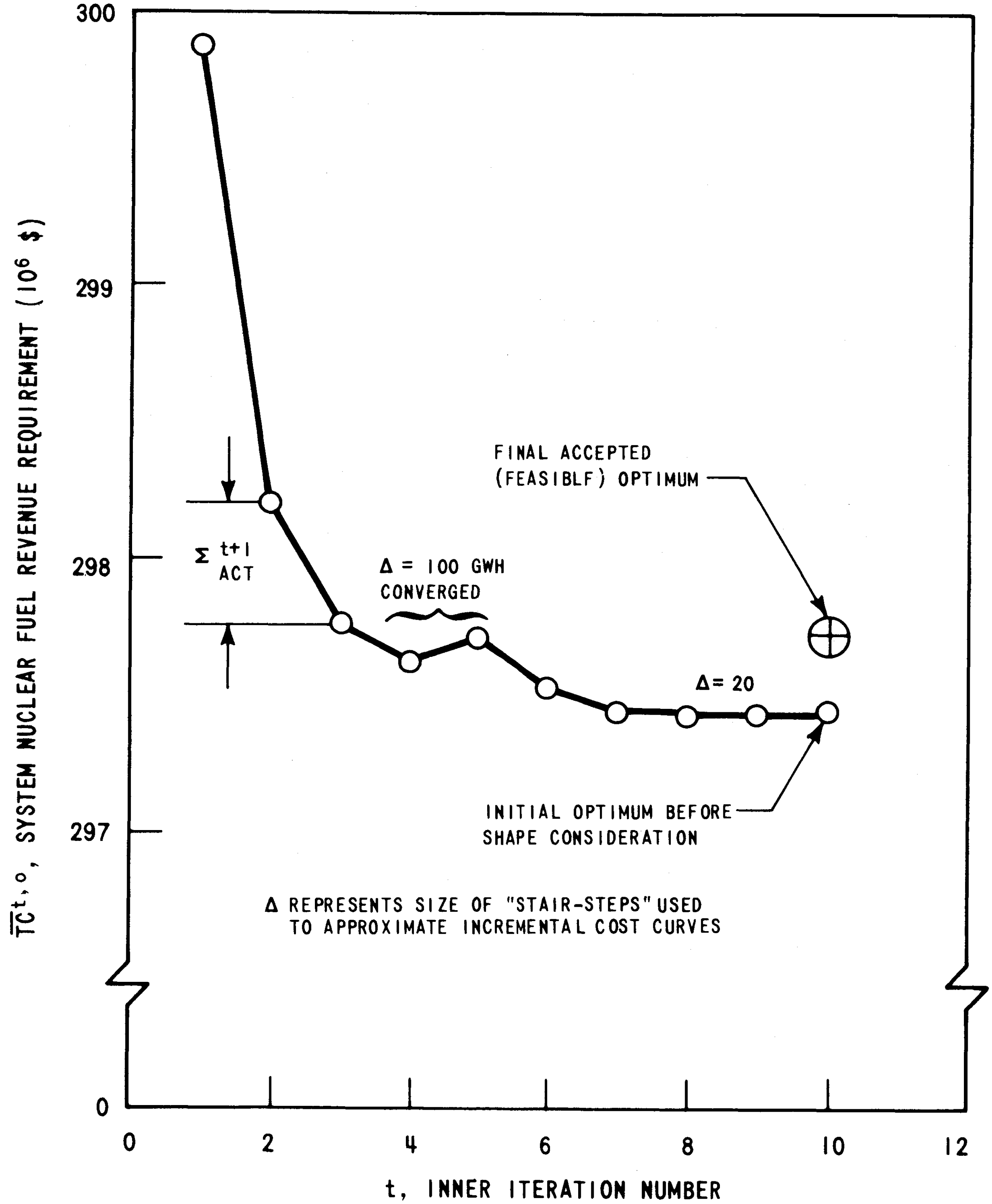
Table 1.3

Revenue Requirements and Undiscounted Energy for Accepted Global Optimum of Strategy 2 over Six Year Horizon

	10^6\$	10^6 MWH
Fossil Fuel	276.583	85.836
Startup-shutdown Cost	1.704	--
Emergency Purchases	0.407	0.048
Non-nuclear Production	278.964	85.884
Nuclear Fuel	297.709	194.077
System Production	576.673	279.961
Fixed Firm Purchase	133.920	81.468
System Total	710.593	361.429

Figure 1.12

Convergence of Inner Cost Iterations for Initial Shape Iteration of Strategy 2 in Case I



The symbol Δ in the Figure represents the energy step size used to segment the continuous incremental cost curves into the stair-step cost functions required by the SOM's NP optimization package. As Δ decreases, the accuracy of the stair-step representation increases as do the computational requirements. Thus, the relatively poor λ_{rc} fits at large Δ were utilized for the initial iterations until either the cycle energies converged (to within a specified percent of Δ, typically 100%) or the objective function itself converged (i.e., the last iteration failed to improve the objective function by more than a required amount, say $2000). In fact, iteration 5 displayed "negative" improvement because piecewise-linearization of $\overline{TC}_r$ prevented the NP program from seeing the smooth increase of λ_{rc} for fractional Δ changes in cycle energy. The net result was that the NP program over-reacted to small differences between various λ_{rc} incremental costs.

After convergence using the first Δ, a second and smaller Δ was utilized and convergence again attained using the same two criteria. This second converged solution was considered to be the initial optimum $\overline{TC}^*$.

From three standpoints, a third Δ choice appeared unwarranted:

(1) With total nuclear fuel cost approaching $300,000,000 for the six year horizon, the fuel cost improvement from the Δ = 100 GWH optimum solution to Δ = 20 was

only $220,000 for the fivefold Δ reduction and would undoubtedly have been much less than that for another fivefold reduction.

(2) At Δ = 20 GWH, cycle energies were already converged to well within 1% (± 50 GWH out of 6000-8000 GWH). and

(3) The fuel cost errors and cycle energy errors both appear to be well within the noise levels of CORSOM errors (> $100,000 per reactor over the planning period) and the errors inherent in forecasting load demands and availabilities (> 1%).

Using the above sequence of the two step sizes, all cases effectively converged (i.e., objective function decreasing insignificantly for Δ = 20 GWH) within ten iterations. Inasmuch as completed CORSOM's are estimated to require over 3 minutes of IBM 370 Model 155 CPU time per reactor strategy per iteration (41), an average six reactor-four iteration solution would involve over an hour and a half of computer time for the CORSOM's alone. The ad hoc simulator QKCORE required less than 3 minutes for all ten iterations.

1.8.2 Nuclear Incremental Costs at the Optimum

An analytical discussion of nuclear utility system optimization similar to that in (22) presents two conclusions relating a strong primary dependence between pertinent cycle incremental costs for each reactor during each period and a weak secondary conclusion relating an idealized state that may not be attainable:

<u>Conclusion I</u>:

At the optimum reactor-cycle energies,

$$\lambda_{N_p} = \frac{\partial \overline{TC}_r}{\partial E_{rc}} \qquad \text{for all r} \quad (1.35)$$

during each period for the pertinent cycle of each reactor.

<u>Conclusion II</u>:

At the optimum reactor-cycle energies,

$$\lambda_N = \frac{\partial \overline{TC}_r}{\partial E_{rc}} \qquad (1.36)$$

for all periods, all cycles and all reactors <u>simultaneously</u>.

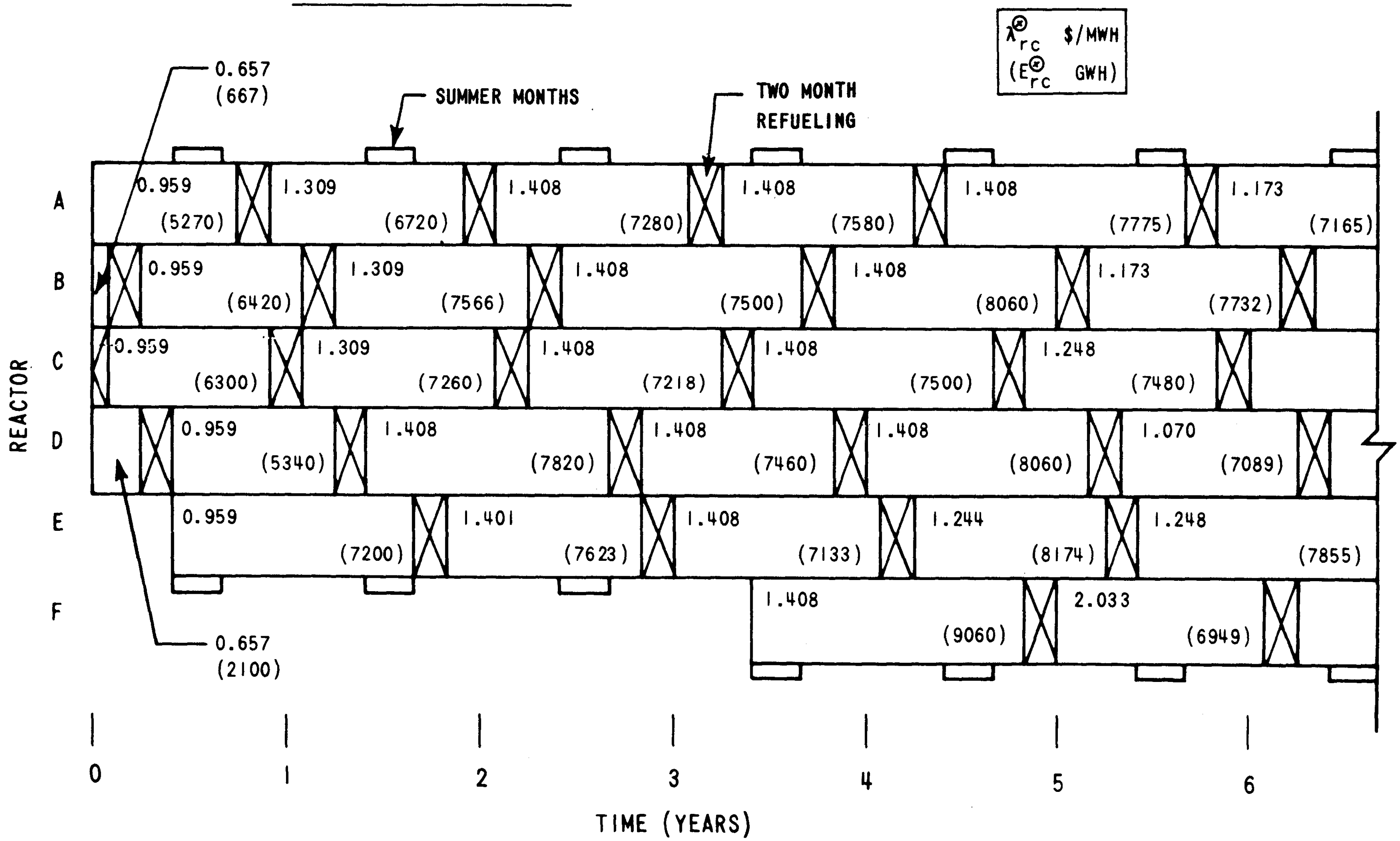
As for typical values of λ_{N_p} and λ_N, the results of Widmer (<u>57</u>), Kearney (<u>51</u>) and Watt (<u>55</u>) indicate optimum mid-range nuclear incremental costs in the range of 0.9 to 1.5 $/MWH.

The terms "strong" and "weak" refer to the number of incremental cost violations anticipated because of over-riding engineering and time constraints.

The λ^*_{rc} cycle-by-cycle incremental costs at the optimum of Strategy 2 are presented in Figure 1.13. In analyzing these values, four important points are to be made. First, the general equality of λ^*_{rc} at each point in time confirms Conclusion I.

Figure 1.13

Incremental Costs and Cycle Energies at Accepted Global Optimum for Strategy-2 in Case I



6253-54

Secondly, incremental costs increase over the first few cycles as the short-range incremental costs of the first year give way to the mid-range incremental costs of later cycles. During the first year, incremental costs are very low because a large proportion of each reactor's cycle costs (e.g., separative work, fabrication and reprocessing) are already spent or committed. Discharge burnup is the only variable. Thus, λ^*_{r1} is Widmer's short-range incremental cost (57, 59). For a cycle further into the future, a larger degree of flexibility is available in the design of the reload batch (size and enrichment) and a larger fraction of total cycle costs can thus be altered. For $c > 2$, λ^*_{rc} becomes Widmer's mid-range incremental cost (58, 59). Thus, short-range incremental costs evolve into mid-range incremental costs.

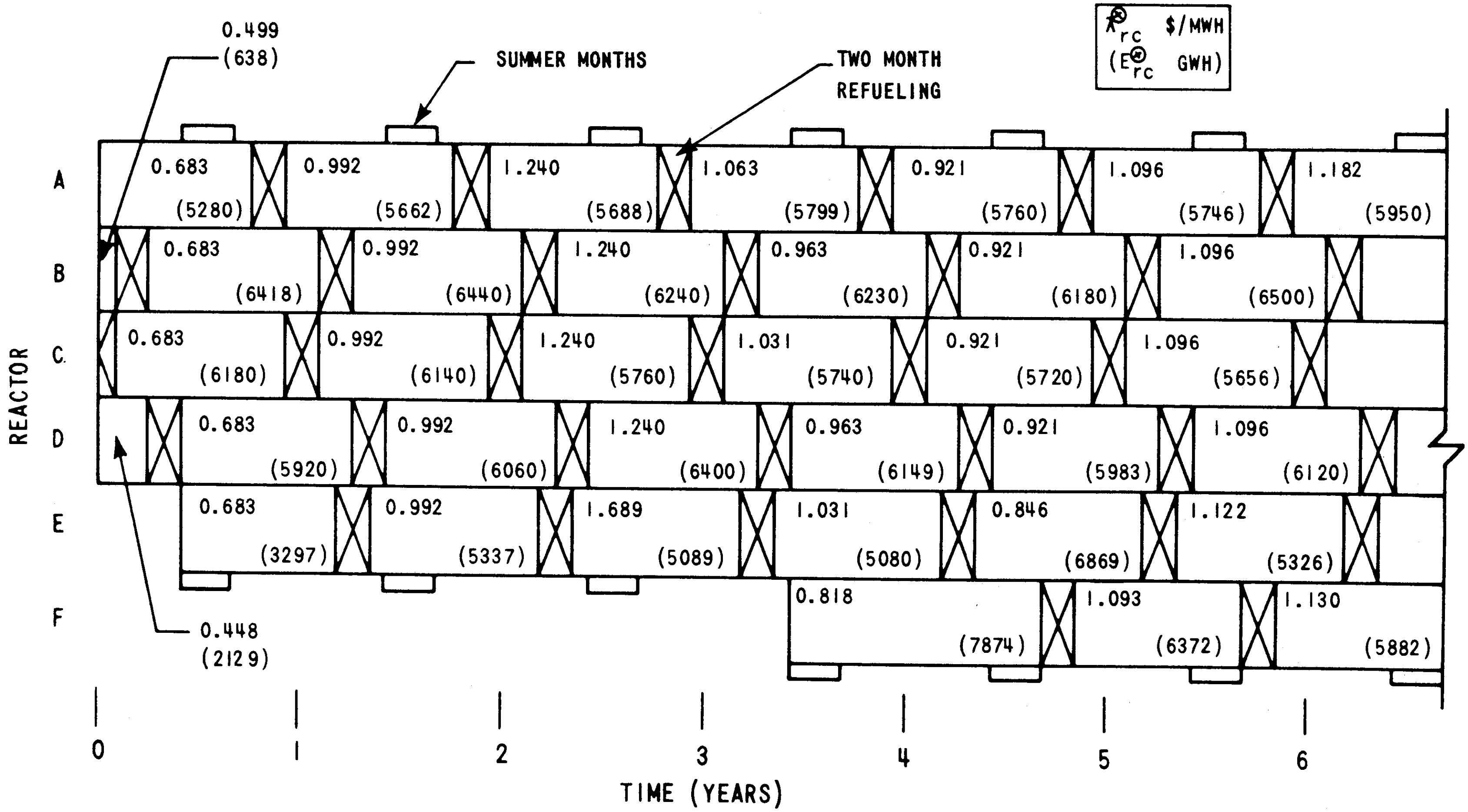
During the middle two to five years of Strategy 2, the constancy of λ^*_{rc} for most reactor-cycles provides ample evidence that Conclusion II is also valid.

Finally, the λ^*_{rc} beyond the fifth year are, indeed, optimal (but erratic) due to the assumed horizon end condition which involved specifying cycle energies beyond the horizon in order to permit cost evaluation of the core contents at the horizon.

Though Figure 1.13 confirmed Conclusion II, the typical λ^*_{rc} optima of the other strategies did not. For example, Figure 1.14 presents λ^*_{rc} for Strategy 1 over the same six

Figure 1.14

Incremental Costs and Cycle Energies at Accepted Global Optimum for Strategy 1 in Case I



6253-53

year horizon. Though Conclusion I continues to be valid with few violations, evidence supporting Conclusion II is non-existent. However, each inconsistency in these incremental costs as cycles begin and end, can be translated directly into the optimal loading order. During reactor-cycle E-3 (with $\lambda^*_{E,3}$ = 1.689 \$/MWH), Reactor E is loaded only after all other nuclear units (with λ^*_{rc} = 1.240 \$/MWH) are fully loaded. Since for economic reasons E-3 is always last, it generates $E^{min}_{E,3,p}$ during each included period of cycle 3 and, hence, $E_{E,3} = E^{min}_{E,3}$. As Figure 1.15 illustrates, this lower limit on cycle energy prevents E-3 from reaching the cost parity of Conclusion I. (If $E_{E,3}$ was less than $E^{min}_{E,2}$, obviously uneconomic fossil energy costing over 2 \$/MWH would be substituted for its 1.7 \$/MWH energy.)

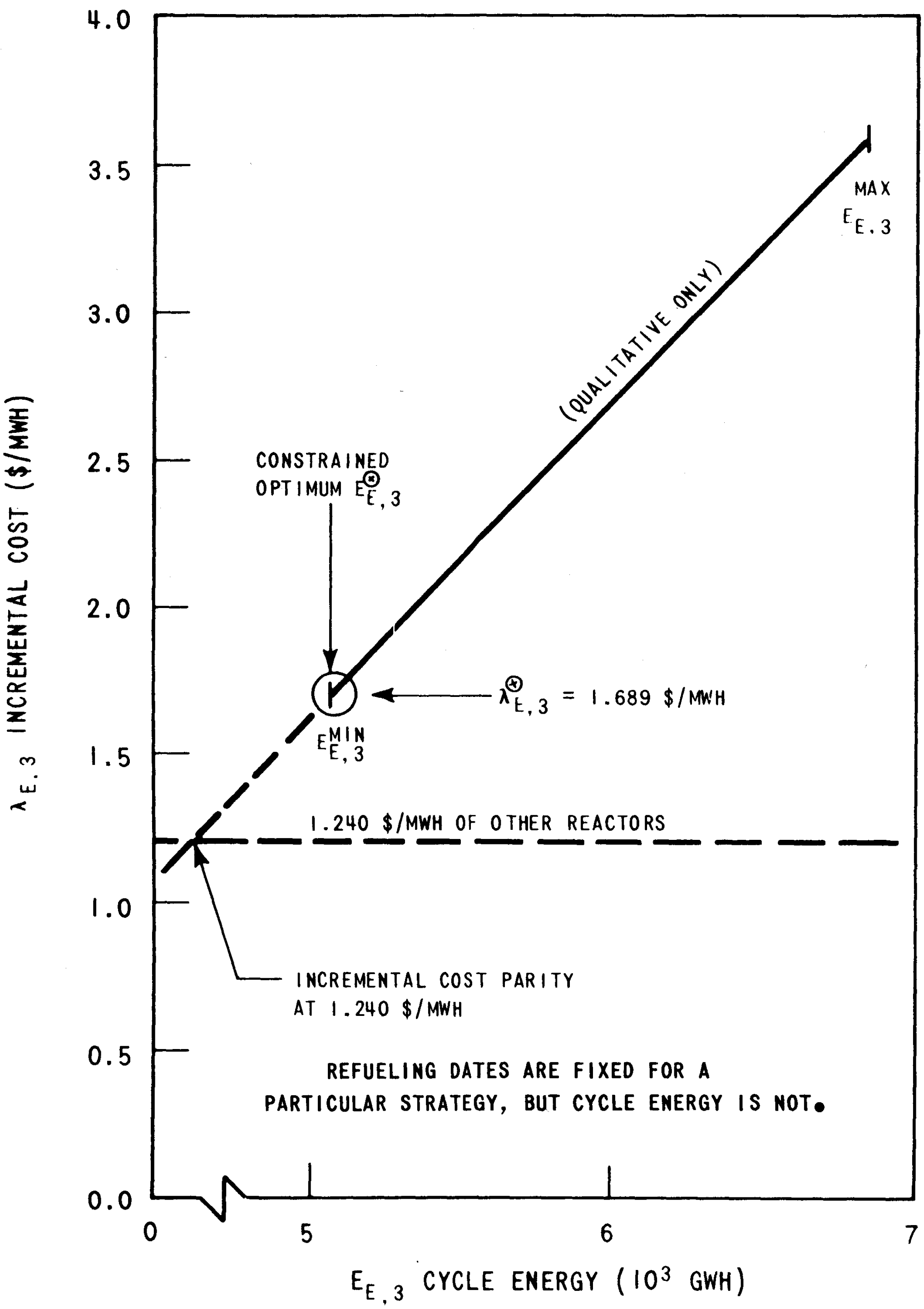
Reactor-cycle F-1 of Figure 1.14 has the opposite problem. With the initial core configuration assumed fixed, $\lambda^*_{F,1}$ is a (cheap) short-range incremental cost. (Cycle burn-up is the only design variable.) Thus, Reactor F is always loaded first, generating $E^{max}_{F,1}$ for the cycle. In an analogous manner, this upper limit on cycle energy can also prevent incremental cost parity.

The other λ^*_{rc} inconsistencies of Figures 1.13 and 1.14 are merely more complicated versions of these two simple cases--reactor-cycles E-3 and F-1. In each instance, the optimal economic period loading order is easily deduced: cheapest first.

Figure 1.15

6253-72

Lower Limit on Cycle Energy Preventing Incremental Cost Parity



Comparing all reactor-cycles of Figures 1.13 and 1.14, λ^*_{rc} is seldom over 1.41 \$/MWH. As Figure 1.2 pointed out, base-loading of a utility system's nuclear reactors may be impossible because the utility's minimum load is too low. However, since λ_N is always much less than λ_F (>2.0 \$/MWH), two possibilities exist for economically utilizing the excess nuclear capacity during the low load periods. One alternative is to sell excess nuclear capacity (i.e., energy) to neighboring utilities at a price greater than its incremental cost. Incorporation of such nuclear economy interchange sales into the SIM and SOM is desirable since this may well become a common utility practice.

The second option is to use the excess capacity on the utility's own system by operating a pumped-hydro station. By pumping during low load hours, $\lambda_P = \lambda_N \leq 1.4$ \$/MWH. Using the stored energy for peak-shaving high cost fossil the next day, $\lambda_G = \lambda_F > \sim 4$ \$/MWH. Even if overall pumped-hydro efficiency is only 67%, total operating revenue requirements are reduced roughly 2 \$/MWH (i.e., 50% of λ_F) for each fossil MWH displaced. Since such a station is also comparatively cheap to install (100-200 \$/kwe), a pumped-hydro station on the grid of a heavily nuclear utility produces startling economies (21, 35). "From a utility's viewpoint, pumped storage is a natural fit with large base-load plants. It can take on load instantly, it uses off-peak power to replenish its resources, and its reliability is second to none (5)."

As pumped-hydro stations become more numerous [~4400 MW installed versus over 8000 MW under construction in entire United States at end of 1972 (5)], the appropriate planning tools must be developed. Thus, it is highly recommended that pumped-hydro units (and hydro units, as well) be incorporated into the SIM.

Underlying the above discussion of incremental costs is the source of those costs--the CORSOM, or specifically, the QKCORE in-core simulator developed merely to test the SOM. By forgoing reload optimization, QKCORE is unable to see some obvious means of saving money. For instance, reactor-cycle E-3 of Figure 1.14 has a very high incremental cost due to energy production requiring 4% enriched reload fuel. Yet, the previous cycle loaded the minimum enrichment allowed (1.5%). If QKCORE allowed early shutdown (reactivity > 0) and optimized the enrichments alone, it might well have loaded 2.5% fuel in E-2, burned only part of the way down and then loaded 3.0% fuel for a complete burn. Indeed, a full-scale CORSOM would be able to optimize reload batch size, as well. What would be the optimum incremental costs for such modes of operation? Obviously, the incorporation of more versatile CORSOM's is a prerequisite to completing a fully operational nuclear power management model.

1.8.3 Computational Requirements

The computational requirements of SYSINT are detailed in Section 1.6.2 while SYSOPT details can be found in Section 1.7.2. However, Table 1.4 presents a summary of computer usage for Strategy 2.

1.8.4 Evaluation of Competing Strategies

Having discussed the properties of a single optimized strategy, it now becomes appropriate to discuss the broader question of strategy versus strategy comparison. In particular, given the same set of input data (i.e., forecasts), which of the individually optimized strategies represents the optimum plan for operating the utility system? How sensitive is this choice to various parameters in the input? To answer these questions, the results for the three Strategies over a four year horizon are presented in Table 1.5.

Recall that S-1 is an annual refueling strategy, S-2 a gradual shift to longer cycles and S-3 an immediate shift to longer cycles.

Of prime importance in correlating the results, is the refueling downtime of each strategy. Naturally, the more rapid the shift to longer cycle lengths, the fewer refuelings that must be scheduled.

With less nuclear downtime, the nuclear energy production increases and fossil energy production decreases by approximately the same amount. Also, startup-shutdown cost is decreased as the fossil units move farther away from nightly

Table 1.4

Computational Requirements for Strategy 2

(Based on IBM 370 model 155 computer operating in MVT environment)

Program	Total Core Storage (Bytes)	CPU Time	Input/ Output Time	Time Units
SYSINT	134 K	2.2	0.5	Sec/period
SYSOPT	246 K with overlay	9	7	Sec/inner iteration
QKCORE	371 K without overlay	13	<1	Sec/inner iteration

TABLE 1.5

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY OVER FOUR YEARS

(48 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
		10^6\$ (10^6 MWH)	
Fossil fuel	184.223 (51.703)	176.348 (50.061)	173.250 (49.390)
Startup-shutdown cost	1.497	1.281	1.227
Emergency purchases	0.464 (0.053)	0.317 (0.036)	0.265 (0.030)
Nonnuclear production	186.184 (51.756)	177.946 (50.097)	174.742 (49.420)
Nuclear fuel	198.267 (118.376)	197.189 (120.035)	199.821 (120.712)
System production	384.451 (170.132)	375.135 (170.132)	374.563 (170.132)
Fixed firm purchase	95.166 (54.312)	95.166 (54.312)	95.166 (54.312)
Penalty for short-notice enrichment changes			2.000
System Total	479.617 (224.444)	470.301 (224.444)	471.729 (224.444)

shutdown. Fewer emergency energy purchases are required due to increased on-line resource margins.

All three components of non-nuclear production cost thus favor reducing downtime. (By looking at the differences in non-nuclear production cost, average long-term levelized replacement energy costs of 5.2-5.7 $/MWH can be calculated.)

As mentioned above, each succeeding strategy is able to increase production because of less refueling downtime. However, the cost of this energy does not increase proportionally. In fact, compared to S-1, S-2 generates more nuclear energy for less money! To explain this anomaly, consider the following:

(1) Less downtime means fewer reloads must be purchased.

(2) Increased average cycle length, however, means increased cycle energy and reload enrichment.

(3) Even with increased batch enrichment cost, the savings due to foregone reloads and the increased energy for amortizing fixed costs, etc., result in a 1.9% decrease in levelized nuclear fuel costs over the four year horizon.

(4) Due to fixed initial conditions and only gradual shift to longer cycles, S-1 and S-2 are very similar in energy production during the first year. At the end of four years, energy production by S-2 is only 1.4% higher. (For longer horizons, the first year

matters less and energy production differences are greater.)

(5) Finally, since the levelized nuclear fuel cost decreases percentagewise more than energy production increases, the net result is more nuclear energy for less money.

Turning to S-3, the immediate shift to longer cycles results not only in increased energy production, but also in increased levelized fuel cost. The result is a return to normalcy--more nuclear energy costs more.

Looking then at system production cost, S-3 saves $570,000 over S-2 and roughly ten million dollars over S-1. This, of course, is not enough to absorb S-3's assumed additional two million dollars in penalties for the two short notice enrichment changes required for the immediate shift to longer cycles. Thus, among the three strategies, S-2 has minimum total system cost.

During the first four years, then, S-2's gradual shift to longer cycles saves 9.3 million dollars compared to the annual cycles of S-1. Such a savings clearly justifies a few hundred thousand dollars in overhead necessary to implement the engineering design changes in the reload fuel specifications.

However, S-2 and S-3 are roughly competitive depending on the magnitude of the enrichment change penalty. Without the penalty S-3 is favored by roughly $600,000. (Of this

$600,000, roughly $95,000 could also be saved by S-2 were it allowed to freely change initial enrichment for two of the reactors.) But after the 2 million dollar penalty, S-3 is 1.4 million dollars more costly.

1.9 Summary

This work presents a multi-reactor, multi-year fuel management model consisting of four sub-models (RAMM, SIM, SOM and CORSOM). The SIM and SOM sub-models have been discussed in some detail. Numerical results were presented as an example of the model's ultimate versatility. Some work remains to be done before the completely computerized nuclear power management multi-year model is ready for implementation on nuclear utility systems. The most severe deficiency is not in either the SIM (SYSINT) or the SOM (SYSOPT), but is due to the large computational requirements of current PWR CORSOM's (estimated at several hours for optimizing a single refueling and maintenance for the entire utility system). In addition, CORSOM's for the other types of reactors are also needed. Acceptable RAMM's already exist [e.g.,(20)] and merely require proper interfacing.

As for the major required improvements in SYSINT and SYSOPT, there are two: (1) addition of hydro and pumped-hydro unit types (likewise, permitting initial cycles of nuclear units to be treated as a scarce-resource initial condition) and (2) on-line sensitivity analysis of the effect on total operating revenue requirement of various forecasting errors, such as incorrect customer load demands or unit performance probabilities.

CHAPTER 2

AN INTRODUCTION TO NUCLEAR POWER MANAGEMENT

2.1 Characteristics of a Utility

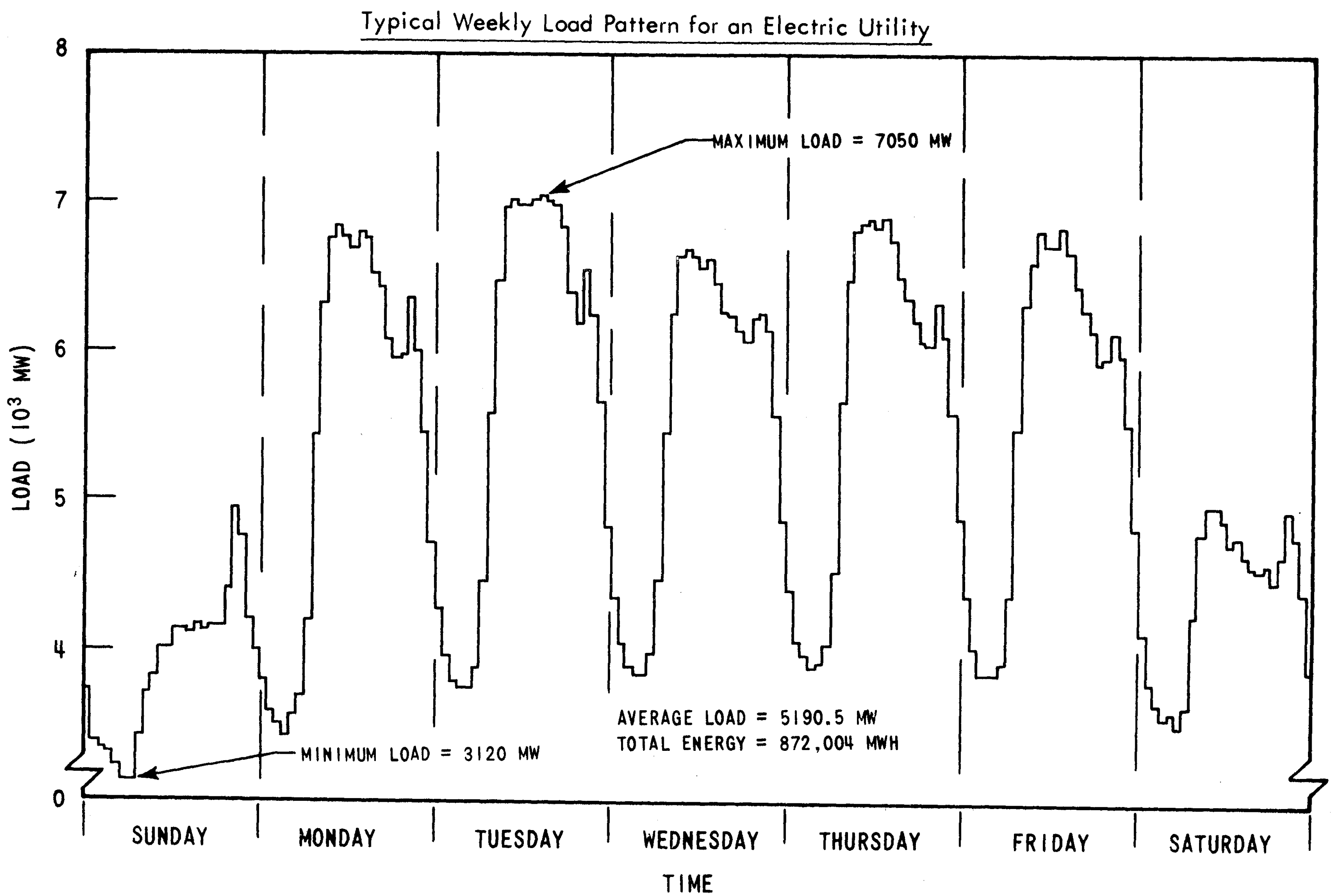
An electric utility, like any other business enterprise, exists because its product fulfills an established need. The utility generates electricity to supply the requirements, or load, demanded by the customers in its geographical service region. The utility's objective is to do so at minimum total cost.

These three characteristics (load demand, power supply and utility objective) must be fully understood before system optimization techniques can be successfully applied to utility management problems.

2.1.1 The Demand: Customer Loads

The load supplied by a utility at any one instant in time is the sum of the individual loads demanded by thousands of customers. These loads range from a residential customer's 40-watt light bulb to a heavy industrial customer's 100 MW's of factory equipment. The statistical nature of the sum of hundreds of thousands of residential customers, thousands of commercial customers and scores of industrial customers makes minute-by-minute load patterns far too cumbersome for even daily management planning work. The typical unit of analysis is the average load during the hour. These hourly loads follow definite daily and weekly patterns for each utility (see Figure 2.1). Minimum loads range from 35% to 60% of peak demand depending on the utility's mix of large round-the-clock heavy industrial customers and small cyclical loads due to residential

Figure 2.1


Typical Weekly Load Pattern for an Electric Utility
8
7
6
5
4
0
LOAD (10^3 MW)
MAXIMUM LOAD = 7050 MW
MINIMUM LOAD = 3120 MW
AVERAGE LOAD = 5190.5 MW
TOTAL ENERGY = 872,004 MWH
SUNDAY
MONDAY
TUESDAY
WEDNESDAY
THURSDAY
FRIDAY
SATURDAY
TIME

and commercial customers. Even for the same utility, seasonal variations and annual load growth affect these patterns.

For daily (or even annual) models, chronological hourly load detail may be appropriate. However, multi-year and long-range models cannot afford to look at each of the 8760 hours in each year. For these models, the load-duration curve is more appropriate. Figure 2.2 presents the load-duration curve for the data of Figure 2.1. The 168 hours in the week are merely rearranged in order of decreasing load demand. Thus, the peak demand occurs during the first hour of the new time scale and the minimum load occurs during the last hour. The interpretation of the new time scale is the number of hours the load was greater than or equal to a specified power level — in short, the load's duration.

The rearrangement of loads results in the complete loss of chronological information, but preserves the more important property that the integral under the curve is the total energy demanded during the week.

Realizing hourly loads are actually averages of a rapidly changing but continuous function, such histograms are usually drawn as smooth curves. In addition, two other changes are made to the load representation throughout the work reported here. First, the axes are reversed so that the power level P is the abscissa and duration d the ordinate (see Figure 2.3). This facilitates mathematical treatment of power level as the independent variable and duration as the dependent variable. The second alteration involves normalizing the duration scale by the total length of the time period T'. The new zero-to-one ordinate scale can be interpreted as not only the fractional duration F but, more importantly, as the probability that the load will be greater than or equal to the specified power level at a random instant of time. From Figure 2.3, the load

Figure 2.2

Load-Duration Curve (Standard Format) for the Typical Week

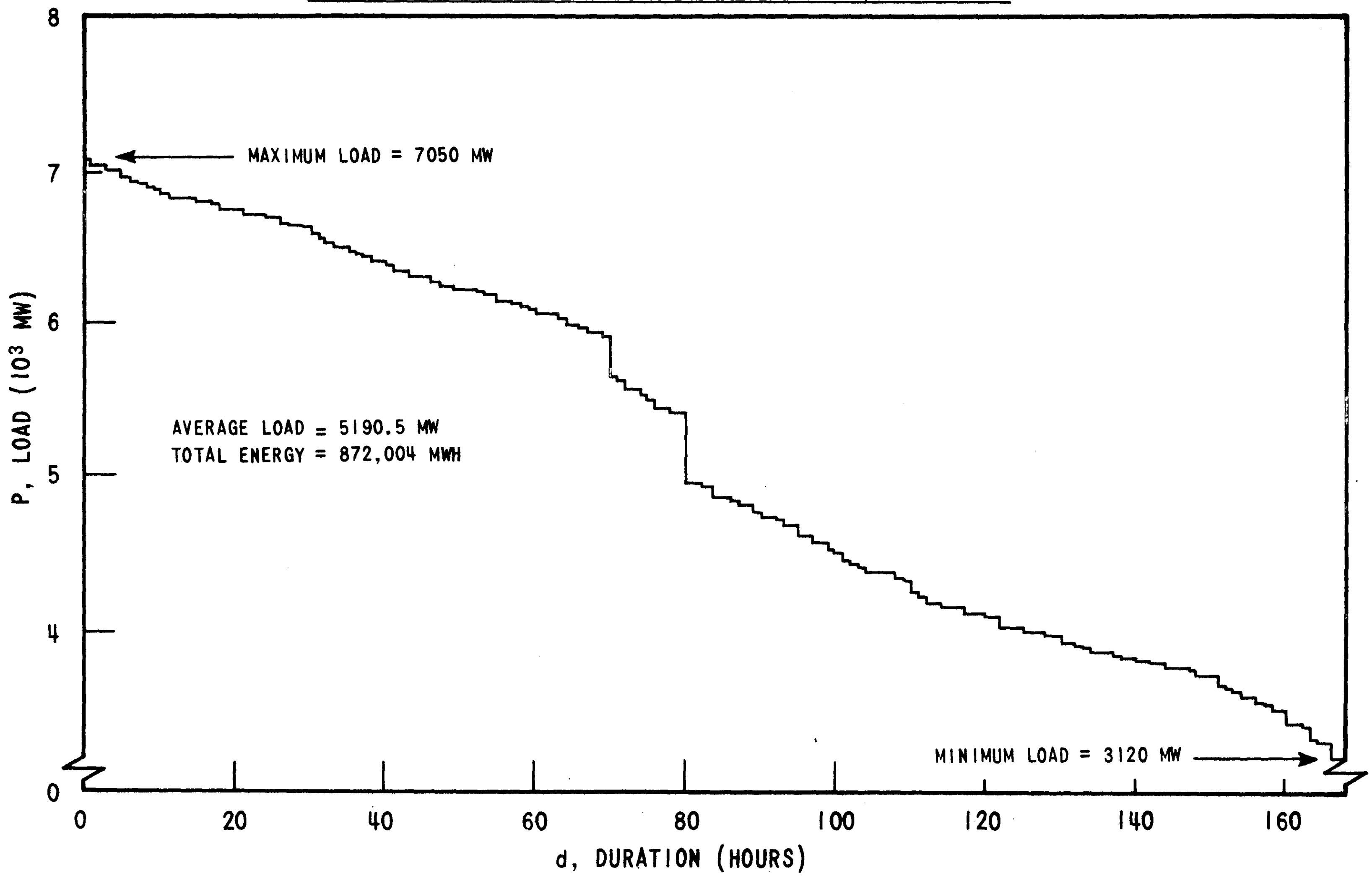
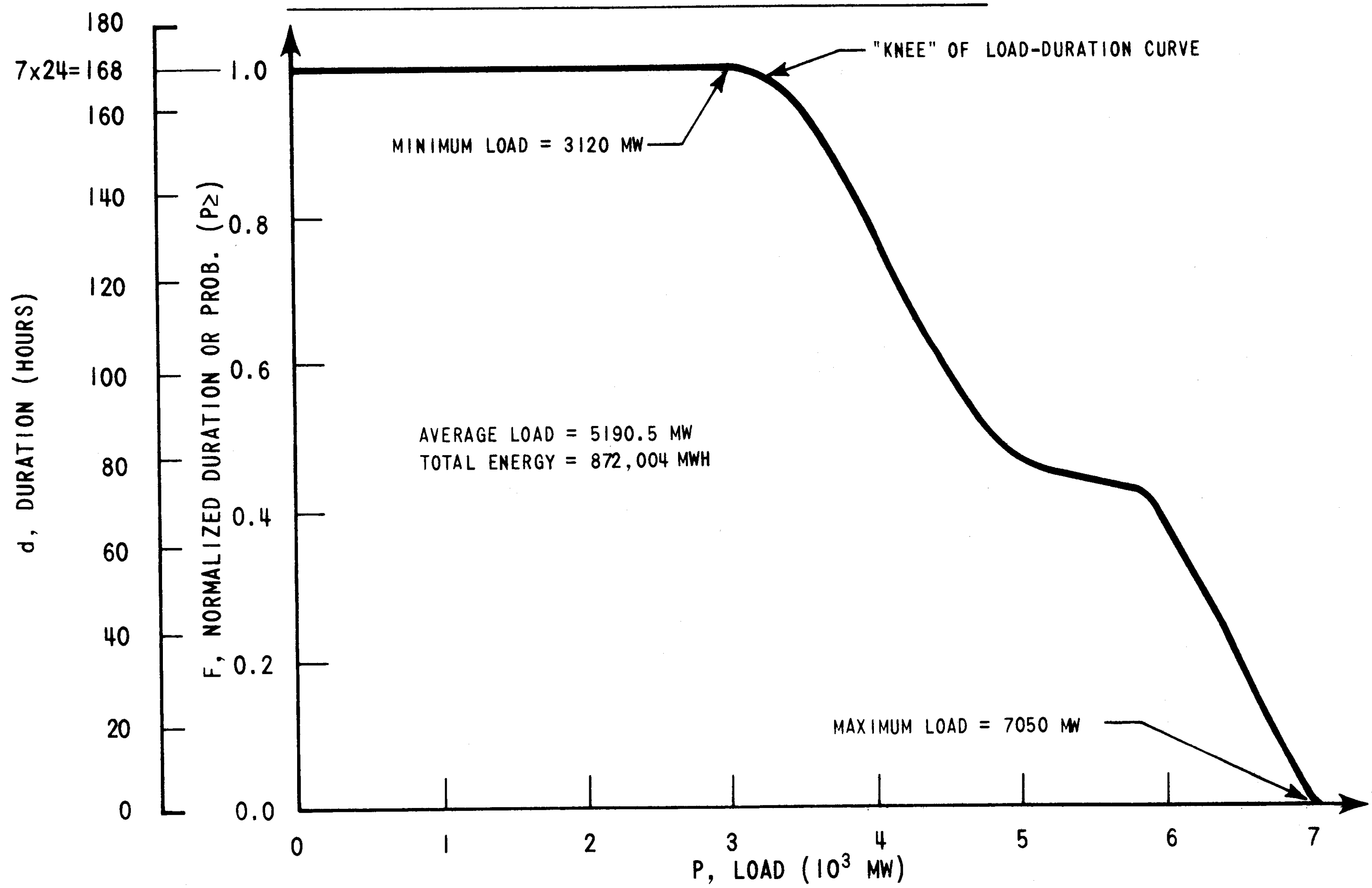


Figure 2.3

Load-Duration Curve (Altered Format) for the Typical Week



was always (100% of the time) greater than or equal to the minimum load of 3120 MW, but never (0% of the time) greater than the peak of 7050 MW.

Neither of these changes alters the basic property that, in the correct units, the integral under the curve is the total energy demanded during the time period,

$$D_T = \int_0^\infty d \cdot dP = T' \int_0^\infty \left(\frac{d}{T'}\right) dP = T' \int_0^\infty F \, dP \tag{2.1}$$

2.1.2 <u>The Supply: Generating Equipment</u>

2.1.2.1 <u>Types</u>

In providing installed capacity to meet the customer loads, a utility relies on up to five different types of generating equipment:

(1) Nuclear units: very large capacity units generating electricity via the heat released by a sustained nuclear chain reaction contained within the reactor's core. If the core coolant exits as a gas or vapor (as in a BWR), it may be expanded directly in turbine-generators. Otherwise, the heat may be first transferred in boilers to produce expandable steam (as with a PWR).

(2) Fossil steam units: typically large capacity coal, oil and/or gas-fired boilers producing high temperature-high pressure steam that is expanded in turbine-generators.

(3) Fast-start peaking units: small fossil-fueled jet engine, gas turbine or diesel-driven generators.

(4) Hydro units: typically medium capacity hydroelectric turbines housed in man-made dams. These dams create the necessary water height differential, or head, by trapping a river's inflows in the reservoir behind the dam.

(5) Pumped-hydro units: similar to hydro except that the dual-purpose turbine may also operate as a pump, transferring water from the foot of the dam to the reservoir. Like a storage battery, excess energy is temporarily stored in another form (water at a height) for later retrieval by reversing the process.

2.1.2.2 Data Required On Each Unit

Regardless of the type of unit, certain key information is required by the system planner on each and every unit of the system:

(1) minimum and maximum power level,[1]

(2) fuel consumption vs. power level,

(3) fuel cost,

(4) fuel inventory,

(5) transmission losses,

(6) startup-shutdown data,

(7) maintenance requirements and

(8) reliability data.

Table 2.1 presents a general summary of these characteristics for each unit type, including capital cost estimates.

The minimum and maximum power levels indicate the lower and upper bounds, respectively, for continuous plant operation. Below the minimum (typically 10 to 50 percent of the maximum), engineering problems, such as boiler flame instability for fossil units, preclude reliable and sustained operation. Similarly, stressing the unit above its maximum power level would be unwise.

[1] Throughout this work, all power levels are in units of net MWe delivered to the transmission system busbar. That is, plant auxiliary power requirements (~5%) have already been subtracted from gross generator output, but transmission losses have not been accounted for.

TABLE 2.1

Characteristics of Types of Electric Generating Units

	Dimension	Nuclear Steam (LWR)	Fossil Steam	Fast-Start Peaking	Hydro	Pumped-Hydro
System Use		Base-Load	Base-Load and Cyclical	Peaking	Inventory Dependent	Peaking
Capacity Fact.	Percent	60-90	30-90	Up to 20	Up to 100	Up to 50
Capital Cost	$/kwe	300-450	250-400	100-150	300-500	100-200
Unit Capacity	MW	500-1200	200-1200	10-50	10-600	50-400
Min. Power	% Cap.	10-40	10-50	75-90	0-10	25-40
Avg. Ht. Rate	MBTU/MWH	10.5-11	8.5-14	12-17	N/A	N/A
Avg. Net Energy Conversion Eff.	Percent	31-34	25-40	20-28	85-93	65-80
Fuel Cost	¢/MBTU	16-20	35-80 (Coal) 50-100 (Oil)	50-100	0	Cost of pumping power
Energy Cost	$/MWH	1.7-2.2	3.0-8.4	6.5-20	0	~1.5 X pumping power
Comments on Fuel Inventory		Depends on fuel cycle	Approx. const. at 100 days supply	4-8 hours (Oil)	Depends on season	Depends on operating cycle
Trans. Losses	Percent	Up to 10	Up to 10	Up to 5	Up to 10	Up to 15
SU-SD Ht. Reqt.	MBTU/MW Cap.	3-6	3-8	0-2	~0	~0
Min. SD Time	Hours	<2	2-10	< 0.3	< 0.5	< 0.5
Maint. Reqt.	Week/Year	4-8 wk/refuel	3-5	1-4	1-2	1-2
Forced-Out Rate	Percent	Up to 15	Up to 20	Up to 40	Up to 5	Up to 10
Perf. Prob.	Percent	85-100	80-100	90-100	95-100	95-100

Fuel consumption data are important in characterizing the unit's thermal efficiency as a function of its power level. Figure 2.4 presents H (heat input rate) versus P (power level) at the valve points typical of a fossil generating unit. Defining $\overline{h}$ and h_{inc} as the average and incremental heat rates, respectively,

$$\frac{H}{P} \equiv \overline{h} \equiv \frac{3.413}{\overline{\eta}} \quad \text{Mega BTU/MWH} \tag{2.2}$$

$$\frac{dH}{dP} \equiv h_{inc} \equiv \frac{3.413}{\eta_{inc}} \quad \text{Mega BTU/MWH} \tag{2.3}$$

During fuel consumption tests, H can only be measured to within a few percent (20). This uncertainty plus the complicated nature of the true H curve (4, 52) make the actual derivative dH/dP impossible to obtain. The result is that $\Delta H/\Delta P$ is usually substituted and treated as a constant for each capacity increment (i.e., between valve points). Figure 2.5 presents $\overline{h}$ and h_{inc} for the data of Figure 2.4. With h_{inc} interpreted as the additional heat input required to generate the next increment of electrical energy, $H(P>K_1)$ can be expressed mathematically as,

$$H(P) = H_1 + \int_{K_1}^{P} \frac{dH}{dP}\, dP = \overline{h}_1 K_1 + \int_{K_1}^{P} h_{inc}(P)\, dP \tag{2.4}$$

In terms of thermal energy, heat rate data can be treated as constant for years at a time. By then applying ϕ time-dependent thermal energy fuel cost, similarly shaped time-dependent incremental energy costs can be calculated,

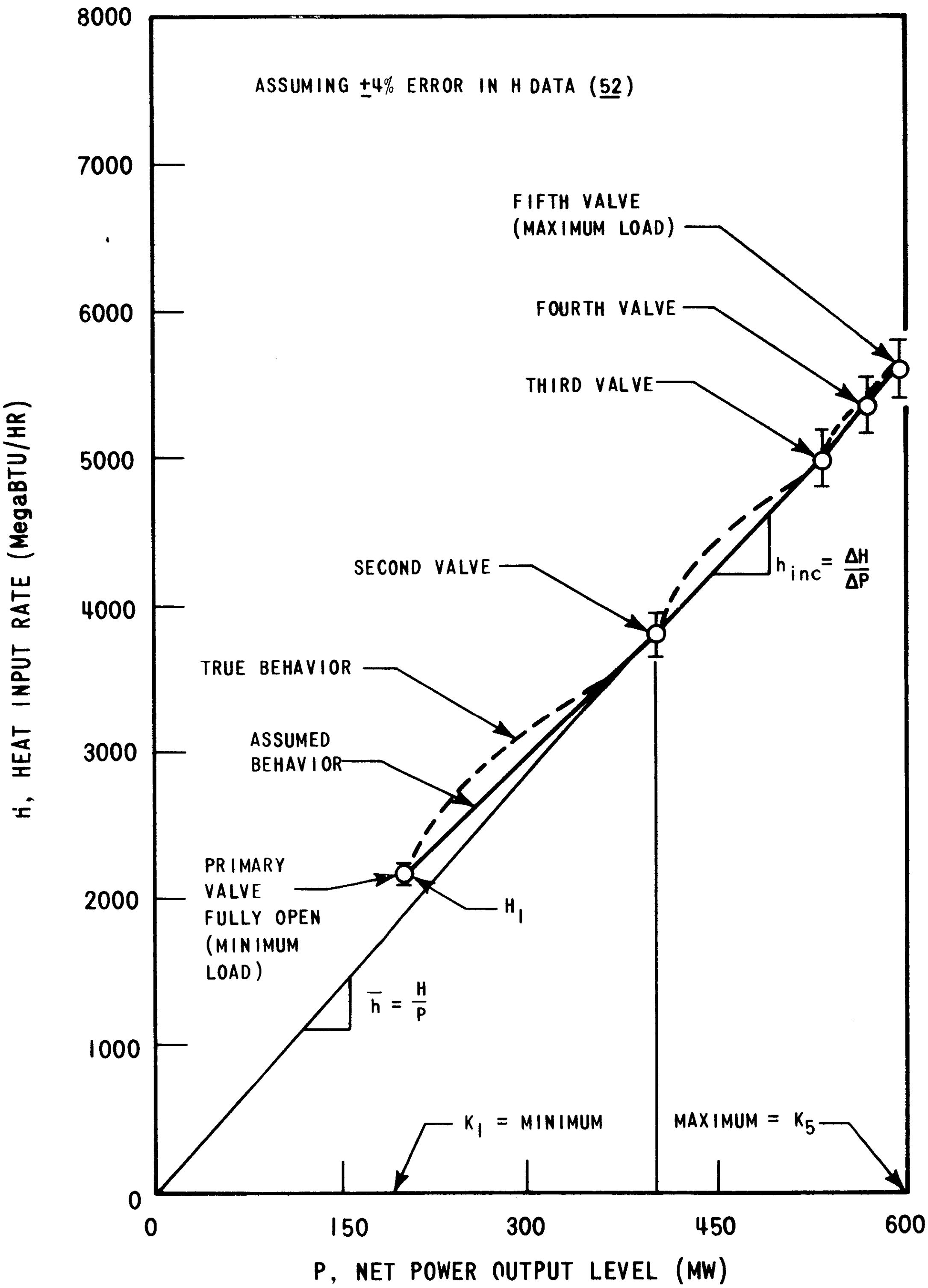
$$\lambda(P,t) = h_{inc}(P)\, \phi(t) \qquad \text{and} \qquad \overline{e}_1 = \overline{h}_1 \;\; \phi(t) \tag{2.5}$$

In the same way that fuel cost has more meaning for a fossil plant than for a hydro unit (where the water is normally assumed to be free), fuel inventory information pertains specifically to the energy-limited type

6253-16

Figure 2.4

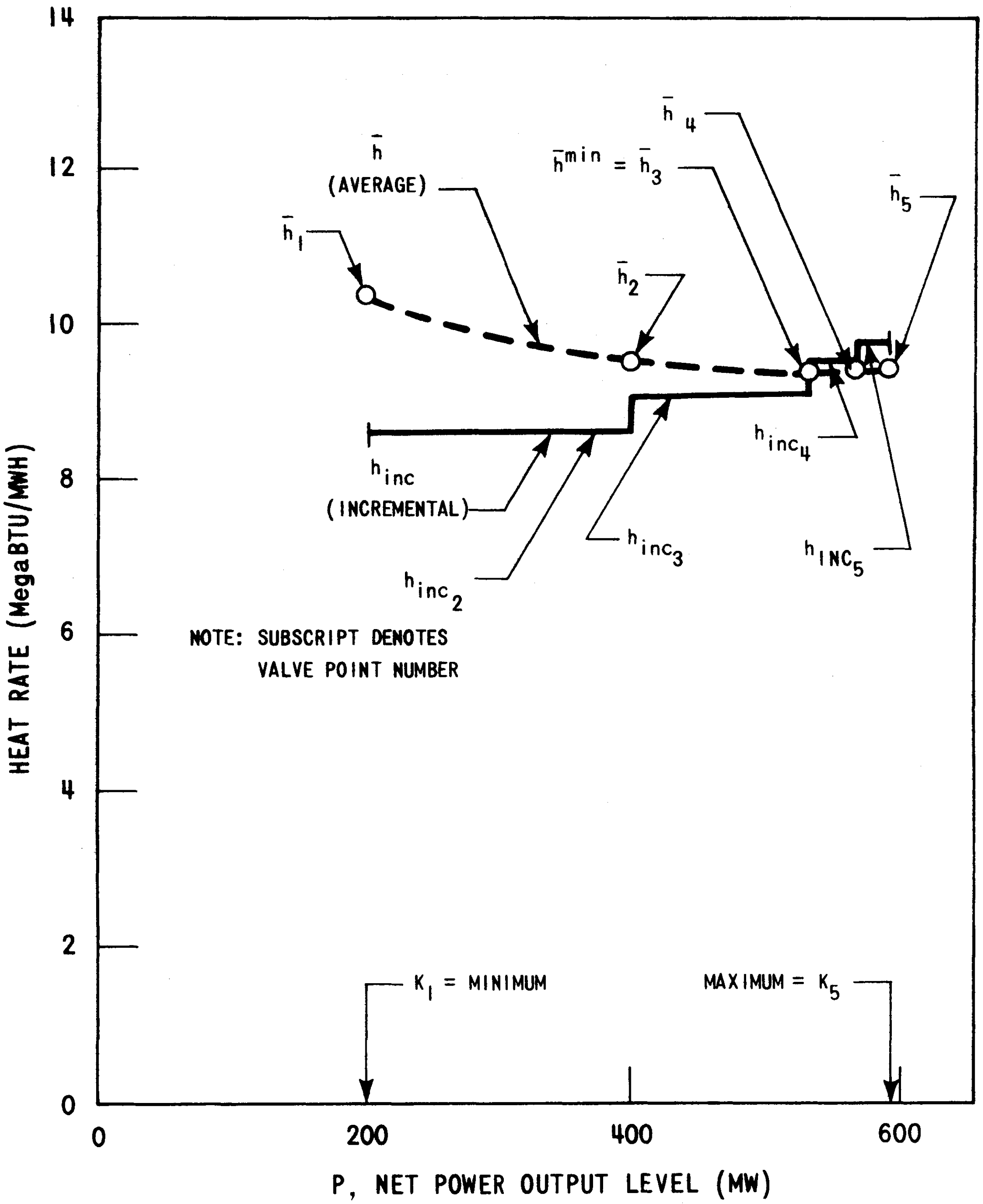
Heat Input Rate versus Net Power Output Level for Typical Fossil Unit [After (37)]



6253-15

Figure 2.5

Heat Rates versus Net Power Output Level for Typical Fossil Unit



of units — nuclear, hydro and pumped-hydro. Fossil fuel inventories are normally maintained at about a 100-day supply (20). Thus, deliveries and consumption can be treated under LIFO last-in, first-out accounting procedures while considering the fuel inventory as an additional initial fixed plant investment. On the other hand, the nature of the nuclear unit's fuel cycle (i.e., core reactivity requirements), the seasonal nature of a hydro unit's river inflows and the weekly pumping-generating cycles of a pumped-hydro unit create situations when there is not enough of the cheap resource to operate the unit at full power all the time. The fuel (or water) becomes a so-called "scarce resource." Generating decisions utilizing scarce resources require a separate method of analysis (see Sections 2.2.2 and 2.2.3).

Transmission losses from the generating unit to the load center must be accounted for. If the customer demands 10 MW, a unit 150 miles away may have to generate 11 MW. Though detailed load flow calculations are required for on-line dispatching (43), more approximate representations are suitable for planning scales on the order of months or years. One of the simplest assumptions is that each unit loses a characteristic percentage of its generation due to this resistance heating. The net MW output for each valve point can then be written down by this percentage so that, just as load demand is in units of MW at the load center, so is unit production. An even simpler assumption (and the one adopted throughout this work) is that transmission losses are negligible or, at least, invariant.

Included in startup-shutdown data are generally three pieces of information: (1) the net cost in time-dependent units of equivalent thermal energy input required for a combined startup-shutdown sequence (see Figure 2.6), (2) the minimum shutdown time (i.e., it is not practical to

Figure 2.6

Startup-Shutdown Cost Data Sheet

GENERATING UNIT SHUTDOWN AND STARTUP COSTS

STATION A UNIT 3 BOILERS ______

GENERATING STATION DATA
(Date & Time)

SHUTDOWN, FROM ______ TO ______

			$ PER:		DOLLARS	
IGNITION FUEL, GAS		Mega BTU	@	=		
IGNITION FUEL, OIL		Mega BTU	@	=		
NORMAL FUEL, GAS	770	Mega BTU	@ .45	=	346.50	
NORMAL FUEL, COAL	337	Mega BTU	@ .50	=	168.50	
ADDITIONAL LABOR	1	HOURS	@ 5.25	=	5.25	
AUXILIARY POWER	5.4	MWH	@ 4.20	=	22.68	
GENERATION (GROSS)	99.6	MWH	@ 4.20	=	- 418.32	CR
			SUB TOTAL		124.61	

IDLE PERIOD, FROM ______ TO ______

			$ PER:		
IGNITION FUEL, GAS		Mega BTU	@	=	
IGNITION FUEL, OIL		Mega BTU	@	=	
NORMAL FUEL, GAS		Mega BTU	@	=	
NORMAL FUEL, OIL		Mega BTU	@	=	
AUXILIARY POWER		MWH	@	=	
			SUB TOTAL		

STARTUP TO NORMAL LOAD LEVEL
FROM ______ TO ______

			$ PER:			
IGNITION FUEL, GAS	67	Mega BTU	@ 1.12	=	75.04	
IGNITION FUEL, OIL		Mega BTU	@	=		
NORMAL FUEL, GAS		Mega BTU	@	=		
NORMAL FUEL, COAL	1098	Mega BTU	@ .50	=	549.00	
ADDITIONAL LABOR	4	HOURS	@ 5.25	=	21.00	
AUXILIARY POWER	5.2	MWH	@ 4.25	=	22.10	
GENERATION (GROSS)	89.8	MWH	@ 4.25	=	- 381.65	CR
GRAND TOTAL:			SUB TOTAL		285.49	

EQUIVALENT NORMAL FUEL 820 Mega BTU @ $.50 = 410.10

ONE HOUR NORMAL OPERATION

NORMAL FUEL		
AUXILIARY POWER		MWH
GENERATION (GROSS)		MWH
RESULTANT HEAT RATE		

GROSS ______ NET ______

SIGNED T. A. Edison DATE 5/3/73

shut down a fossil unit and then have it back on-line within an hour or so even if it were economically attractive), and (3) maximum rate of change of power due to engineering limitations. For a model simulating operation on the order of years, only the startup-shutdown cost is required. For models dealing with day-to-day operating decisions (and restrictions), all three must be included.

Preventive maintenance is performed to keep the units in good operating order. Typically, each unit type has a periodic maintenance requirement, such as two weeks per year. As for scheduling this maintenance, most utilities have an annual peak demand period (frequently the summer months) when scheduled maintenance is prohibited to provide the maximum possible system resources (i.e., wholly-owned generating capacity plus the committed capacity of neighboring utilities) to meet the peak. On a calendar, these taboo periods act as partitions between scheduling windows. It is during these windows that all of the system's required maintenance must be scheduled.

Reliability data account for unscheduled maintenance downtime due to a unit being forced out of service by operating problems, a "forced outage." Normally quoted is the forced-outage rate FOR defined by the Edison Electric Institute (7) (see Figure 2.7) as

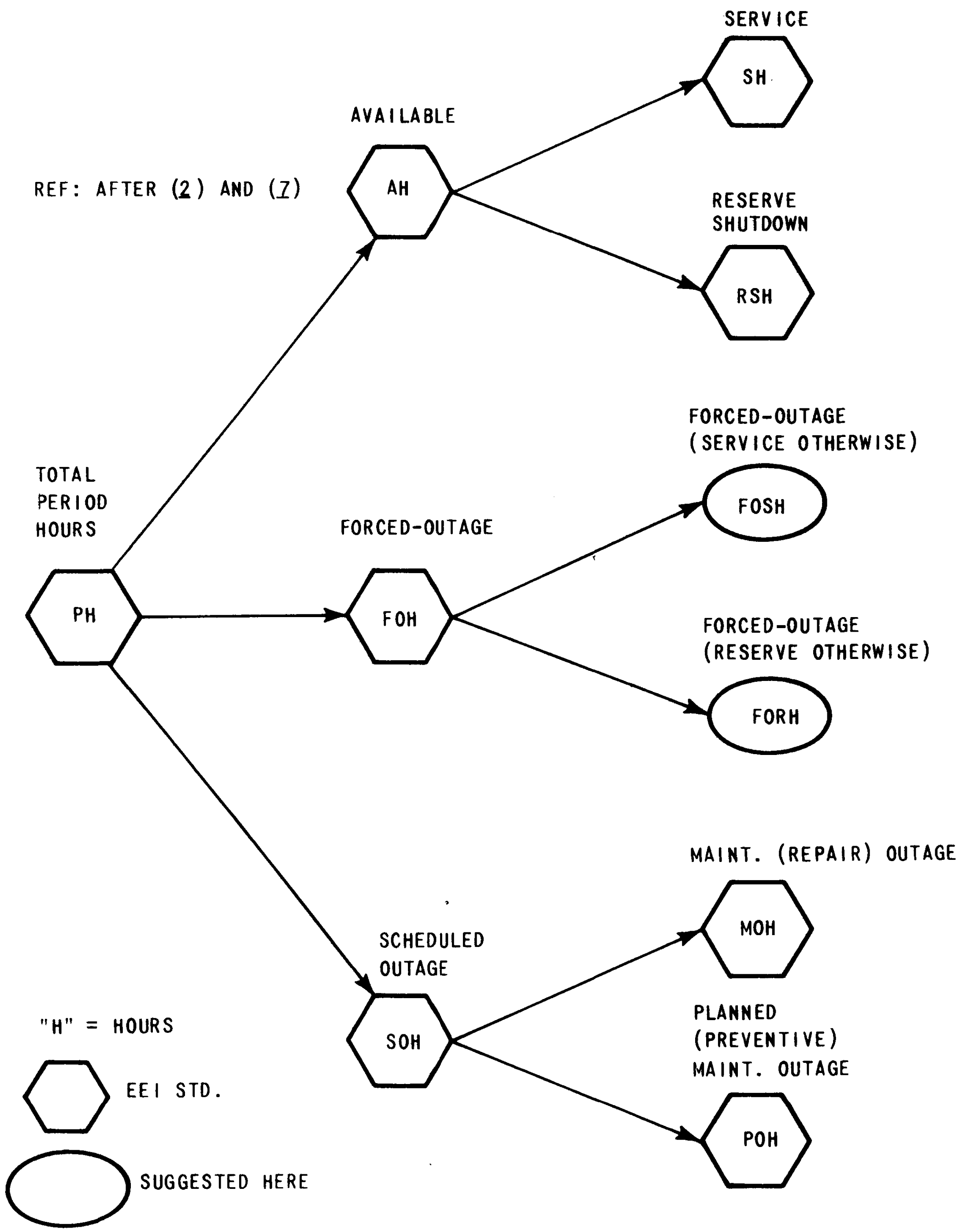
$$FOR \equiv \frac{FOH}{FOH + SH} \qquad (2.6)$$

(Instances of merely derating the unit capability to less than full power due to equipment problems, "forced-deratings," have been ignored.) Currently, the utility industry is continuing (2) to discuss the proper measurement of unit reliability. For this reason, the following detailed discussion is presented.

6253-14

Figure 2.7

Edison Electric Institute Definitions Related to Equipment Availability (Assuming No Forced-Deratings)



Defining the "importance" f as the fraction of forced-outage hours occurring when service was desired (2), the suggested breakdown of FOH in Figure 2.7 becomes

$$\text{FOSH} = f\ \text{FOH} \tag{2.7}$$

$$\text{FORH} = (1-f)\ \text{FOH} \tag{2.8}$$

These additions are required because FOR is not always an accurate indication of how often the unit did not perform when it was called upon. A much better indication of forced-outage effects is q, the nonperformance probability defined as,

$$q \equiv \frac{\text{FOSH}}{\text{FOSH} + \text{SH}} \tag{2.9}$$

Thus the probability that the unit <u>will</u> perform service when called upon, p, can be defined as

$$p \equiv 1 - q \tag{2.10}$$

Returning to Equation (2.9) and utilizing Equation (2.7),

$$q = \frac{f\ \text{FOH}}{f\ \text{FOH} + \text{SH}} \tag{2.11}$$

From Equation (2.6),

$$\text{FOH} = \text{SH}\left(\frac{\text{FOR}}{1 - \text{FOR}}\right) \tag{2.12}$$

Therefore,

$$q = \frac{\text{SH}\left(\frac{\text{FOR}}{1 - \text{FOR}}\right) f}{\text{SH}\left(\frac{\text{FOR}}{1 - \text{FOR}}\right) f + \text{SH}} \tag{2.13}$$

Rearrangement and cancellation lead to the following result,

$$q = \frac{f\ \text{FOR}}{1 - \text{FOR}(1 - f)} \tag{2.14}$$

Figure 2.8 plots the nonperformance probability as a function of the forced-outage rate and the importance. As f approaches 1, q approaches FOR as would be expected for base-load units which are operated whenever possible. On the other hand, forced-outage rate statistics of around 20% to 40% for peaking units make these units appear very unreliable. Considering their low utilizations of around 10%, FOR converts into a respectable 2.5% to 6% nonperformance probability.

2.1.2.3 Five-Unit Reference Utility System

A small Reference Utility System consisting of five units will be used throughout Chapters 2 and 3 for presenting numerical examples designed to assist the reader in understanding the procedures developed here. Quoting Wagner (54), "the manager who resolutely avoids familiarizing himself with the basic mechanism [underlying] his ... application is flirting with trouble. If he really wants to maintain control, he must nurture his insight to the approach."

The pertinent unit data are presented in Table 2.2. The normalized load-duration curve of Figure 2.9 represents the typical month's (730 hour) customer demands. A convenient step size of 100 MW is used for all calculations. A summary of all six examples is presented in Appendix B.

As a final note, a much larger hypothetical utility system consisting of 46 generating units will be used for the nuclear power management model evaluation in Chapter 5. (See Section 5.3.)

Figure 2.8

Non-Performance Probability as a Function of Forced-Outage Rate and Importance

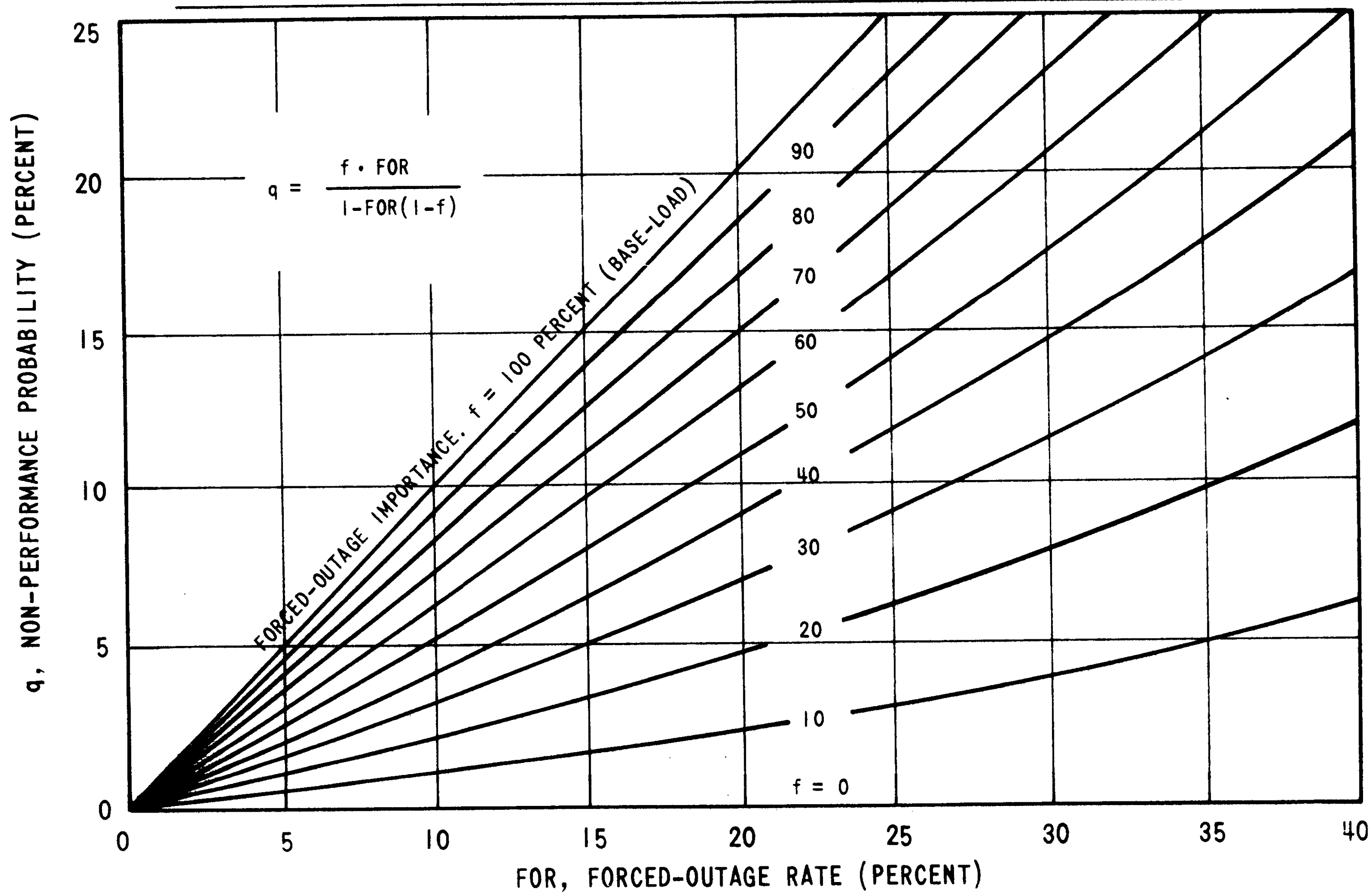


TABLE 2.2

Unit Characteristics for Reference Utility System

Total Capacity = 2000 MW

Unit Name r	Type	Rated Cap. K_r MW	Perf. Prob. p_r %	Fuel Cost ϕ_r ¢/MBTU[3]	SUSD.[1] Heat Q_r MBTU	Valve Point Data K_{r1} MW	$\bar{h}_{r1}$ BTU/kwhe	K_{r2} MW	$h_{inc_{r2}}$ BTU/kwhe
I	P[2]	100	95	90	50	100	18,000	---	----
II	F	200	95	50	800	100	11,000	200	8,500
III	N	300	90	19	1200	100	12,000	300	10,000
IV	F	600	90	40	3600	200	9,800	600	8,300
V	N	800	85	18	2400	300	12,500	800	9,500

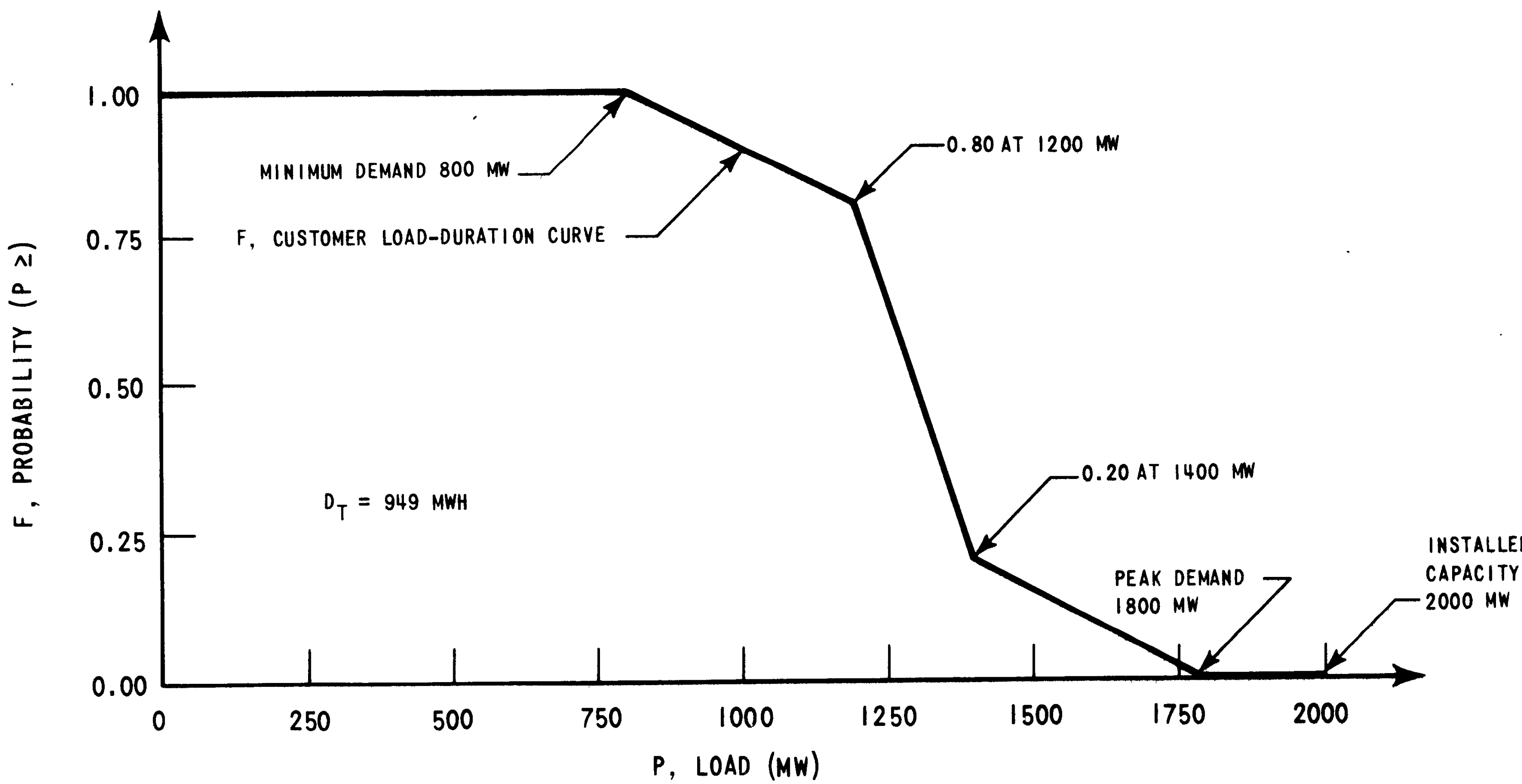
[1] Equivalent startup-shutdown heat requirement

[2] F = Fossil, N = Nuclear, P = Peaking

[3] MBTU = Mega BTU

Figure 2.9

Normalized Customer Load-Duration Curve for 730 Hour Month on Reference Utility System



2.1.3 The Objective: Supply All Demands at Minimum Cost

The electric power supply industry is often chosen as the textbook example of pure monopoly. In fact, electric power is a "natural monopoly" because economies-of-scale with regard to investment in generating and transmission equipment make competition impossible (56). "Recognizing the advantages...of avoiding wasteful duplication and competition, the public [the utility's customers]...grants a utility an exclusive franchise for its particular service in a given geographical region [24]."

As a means of controlling the utility investor's rate-of-return, the Federal Power Commission and state public utilities commissions retain the right to oversee the utility's actions vis-à-vis the public interest. In particular, the local commissions must approve all changes in the electricity rate structure (i.e., prices charged to the utility's customers).

With the rates per unit electricity fixed externally by the regulatory commissions and the total amount of electricity determined externally by the customers' demands, the total revenue received by the utility is also fixed (albeit, in a probabilistic sense). By minimizing the revenue required to recover the cost of supplying that electricity, the utility maximizes total profit. Therefore, the utility objective function is the minimizing of the present value of all future required revenue, i.e., the revenue requirement. (Present valuing accounts for the time value of money.) This sum represents that amount of money which, if received immediately and invested in the company, would just suffice to pay all expenses, as well as permitting a fair return to investors.[2] By including investors' permitted return as another cost component, "revenue requirement" and "total cost" become synonymous. The utility decision-maker is thus responsible for supplying all customer load demands in a reliable manner at minimum total cost.

---

[2]More precisely (55),

> "The revenue requirement is that sum of money, which if received as revenue by an investor-owned electric utility at the beginning of the planning horizon and invested in the enterprise, will defray all subsequent fuel cycle costs, the return allowed by regulatory agencies on that portion of the original investment remaining unexpended at any time, and defray all associated income taxes."

In accounting for all the costs relative to utility operation, revenue is required for the following items:

(1) investment in equipment and facilities,

(2) fuel consumption,

(3) electricity purchases from (less sales to) neighboring utilities,

(4) overhead expenses,

(5) labor and supplies,

(6) maintenance expenses,

(7) taxes and

(8) carrying charges on all of the above.

When considering different operating strategies over a multi-year time horizon (on the order of 5 years), many of the above components are essentially fixed. The long lead times required to effect changes in current equipment installation plans remove item (1) from the multi-year decision-maker's control. On the other hand, total strategy overhead (item 4), labor and supplies (item 5) and maintenance (item 6) are largely invariant though the timing of the latter may be slightly altered by the multi-year strategist.

The multi-year objective function may, therefore, be reduced to the operating costs directly related to supplying customer loads--fuel consumption (item 2) and electricity purchases (item 3) along with the associated taxes (item 7) and carrying charges (item 8).

Adopting the notation that RR(X) is the total revenue requirement related to direct expenditure X,

$$RR(X) = \underset{\text{Value}}{\text{Present}}\ (\text{Expenditure } X) + \underset{\text{Value}}{\text{Present}} \begin{pmatrix}\text{Taxes associated} \\ \text{with } X\end{pmatrix} + \underset{\text{Value}}{\text{Present}} \begin{pmatrix}\text{Carrying charges} \\ \text{associated with } X\end{pmatrix} \quad (2.15)$$

Fuel consumption expenditures can be further broken down into:

(1) X_F, fossil fuel related directly to production,

(2) X_N, nuclear fuel related directly to production, and

(3) X_S, fuel related to startup-shutdown heat requirements.

Expenditures for electricity purchases from other utilities, X_U, represents both emergency purchases and economy purchases. (Economy purchases are not considered further in this work.)

The standard procedure in performing multi-year optimization is to subdivide the horizon into Z smaller time periods. In each time period p, expenditures are estimated in undiscounted dollars. Period expenditures are then present-valued at x per year from their mean time $\bar{t}_p$ back to time zero. As Section 2.3 will point out, the addition of nuclear units may prevent immediate evaluation of X_N. [In fact, $RR(X_N)$ or RR_N is determined directly only after all periods have been simulated.]

The equivalent multi-year objective function ORR, the operating revenue requirement, can then be expressed as

$$ORR = RR_F + RR_N + RR_S + RR_U \quad (2.16)$$

or, in terms of the nonnuclear period expenditures,

$$ORR = \sum^{Z} X_{F_p} \left(\frac{1}{(1+x)}\right)^{\bar{t}_p} + RR_N + \sum^{Z} X_{S_p} \left(\frac{1}{(1+x)}\right)^{\bar{t}_p} + \sum^{Z} X_{U_p} \left(\frac{1}{(1+x)}\right)^{\bar{t}_p} \quad (2.17)$$

2.2 Production Scheduling

Given the predicted customer loads and generating equipment, how are operating expenditures on the Reference System estimated? Much work has been done on modelling utility production scheduling (9, 18, 30, 43, 48, 52, 53). A relatively new technique, the Booth-Baleriaux probabilistic system model (10,19) is rapidly gaining acceptance among utility system planners. The following sections describe qualitatively how the model schedules each type of unit. A quantitative description of the model has been postponed until Chapter 3.

2.2.1 Fossil, Peaking and Nuclear Units

As Section 2.4 will point out, the key element in any utility system optimization is incremental cost. Thus, the first step in any production scheduling technique is surveying the incremental costs of the available units. Using the r^{th} unit and i^{th} increment notation, Equation (2.5) becomes

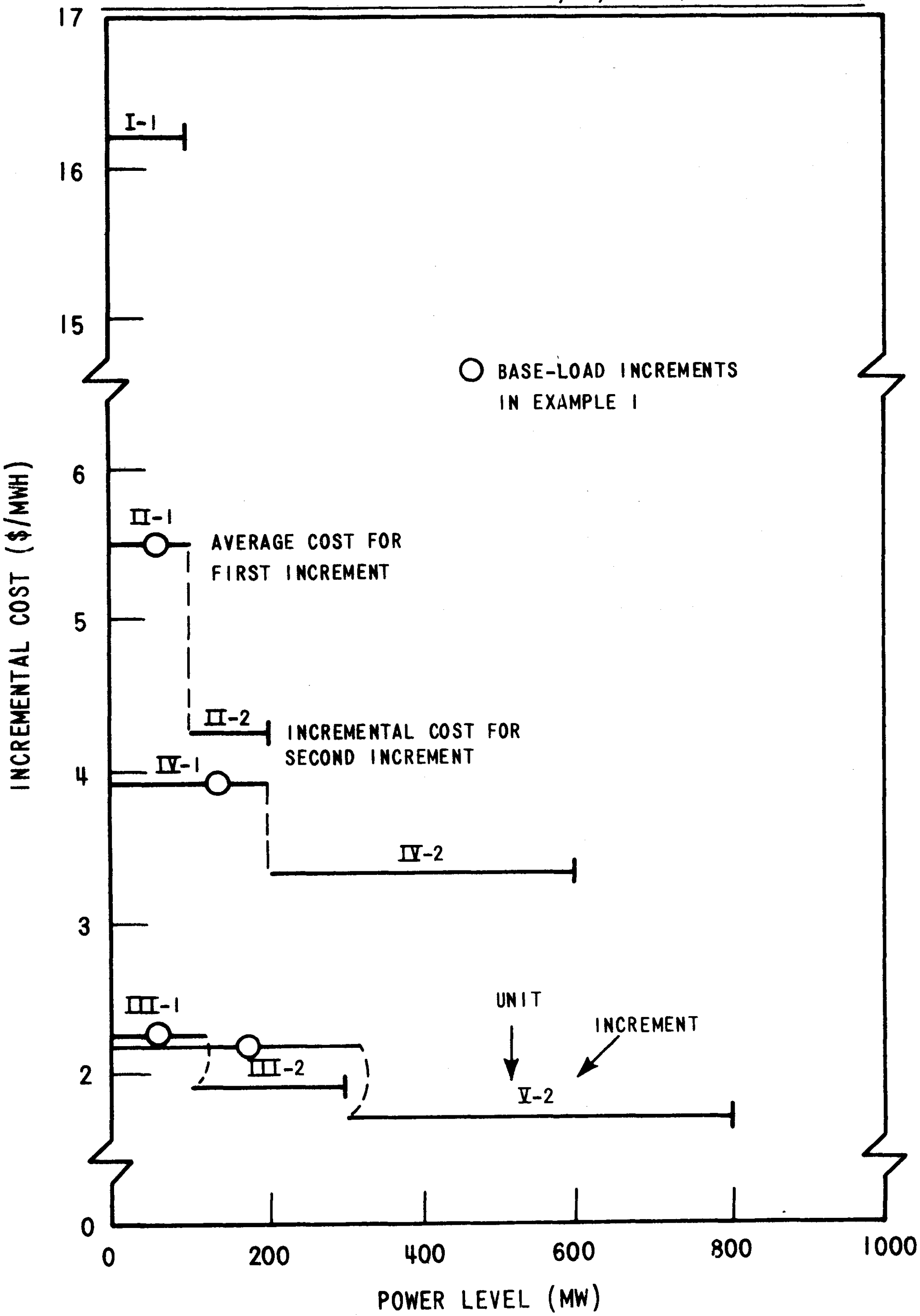
$$\bar{e}_{r1} = \phi_r \bar{h}_{r1} \quad \text{and} \quad \lambda_{ri} = \phi_r h_{inc_{ri}} \quad i > 1 \tag{2.18}$$

Figure 2.10 presents the resulting incremental costs for the Reference System of Section 2.1.2.3. Utilizing these, the order in which the plant increments are started up and loaded (i.e., the startup and loading order) can be established. If all units but Unit I are assumed to be already running at their minimum loads (700 MW in toto), the question is "Which increment should then be loaded when the 701^{st} MW is demanded?" The cheapest unused increment (1.71 \$/MWH per Figure 2.10) is that of Unit V. Thus, it is loaded until total demand reaches 1200 MW. Now Unit III's 1.90 \$/MWH increment should be loaded for the next 200 MW.

6253-18

Figure 2.10

Incremental Costs for Reference Utility System (See Table 2.2)



This procedure of loading in order of increasing incremental cost results in the loading order and system incremental cost curve shown in Figure 2.11. Overlaying this loading order on the customer loads of Figure 2.9 yields the production schedule shown in Figure 2.12. Temporarily assuming all units are always operable (i.e., no forced-outages), energy production by each unit increment E_{ri} equals the total period length T' (the normalizing factor) times the area A_{ri} under that increment's section of the normalized customer load-duration curve,

$$E_{ri} = T' A_{ri} = T' \int_{P^{\circ}_{ri}}^{P^{\circ}_{ri} + \Delta K_{ri}} F(P)\, dP \qquad (2.19)$$

and total unit energy production E_r is given by

$$E_r = \sum^{I} E_{ri} \qquad (2.20)$$

At an average incremental cost of λ_{ri}, the cost of each energy increment is

$$X_{r1} = \tilde{e}_{r1} E_{r1} \quad \text{and} \quad X_{ri} = \lambda_{ri} E_{ri} \quad i > 1 \qquad (2.21)$$

and hence,

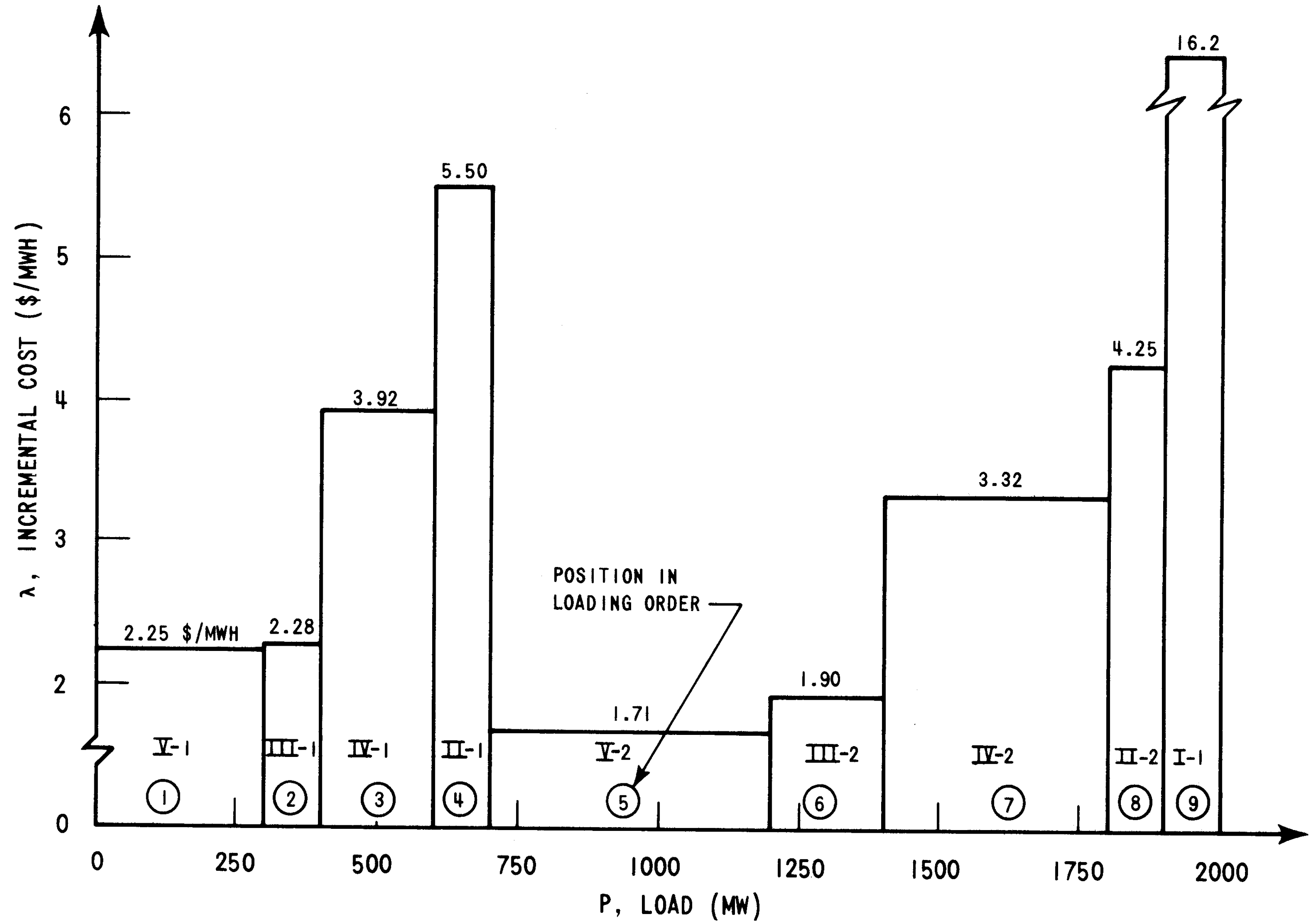
$$X_r = \sum^{I} X_{ri} \qquad (2.22)$$

Table 2.3 summarizes each unit's energy and cost totals for Example 1. (Startup-shutdown costs are ignored throughout this chapter.)

The above description is typical of older, deterministic utility models since all units were assumed always operable with no stochastic forced-outages. Example 2 (see Figure 2.13) portrays the more realistic case where each unit is assumed to have a fixed percentage of random downtime.

Figure 2.11

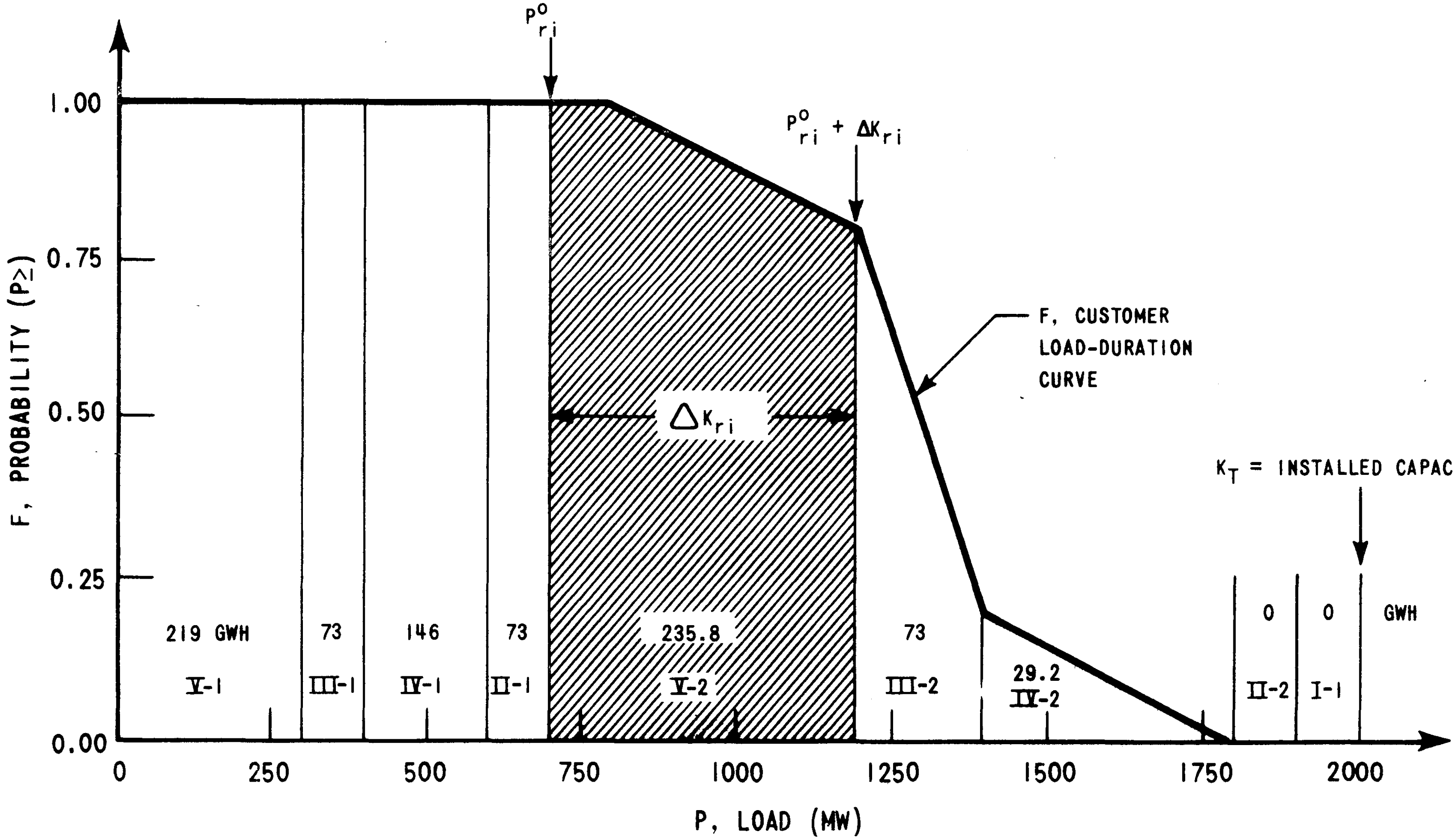
Loading Order and System Incremental Cost for Example 1



6253-25

Figure 2.12

Production Scheduling for Example 1 (No Forced - Outages)



6253-6

TABLE 2.3

Example 1 on Reference Utility System:
"Deterministic Model (No Forced-Outages)"

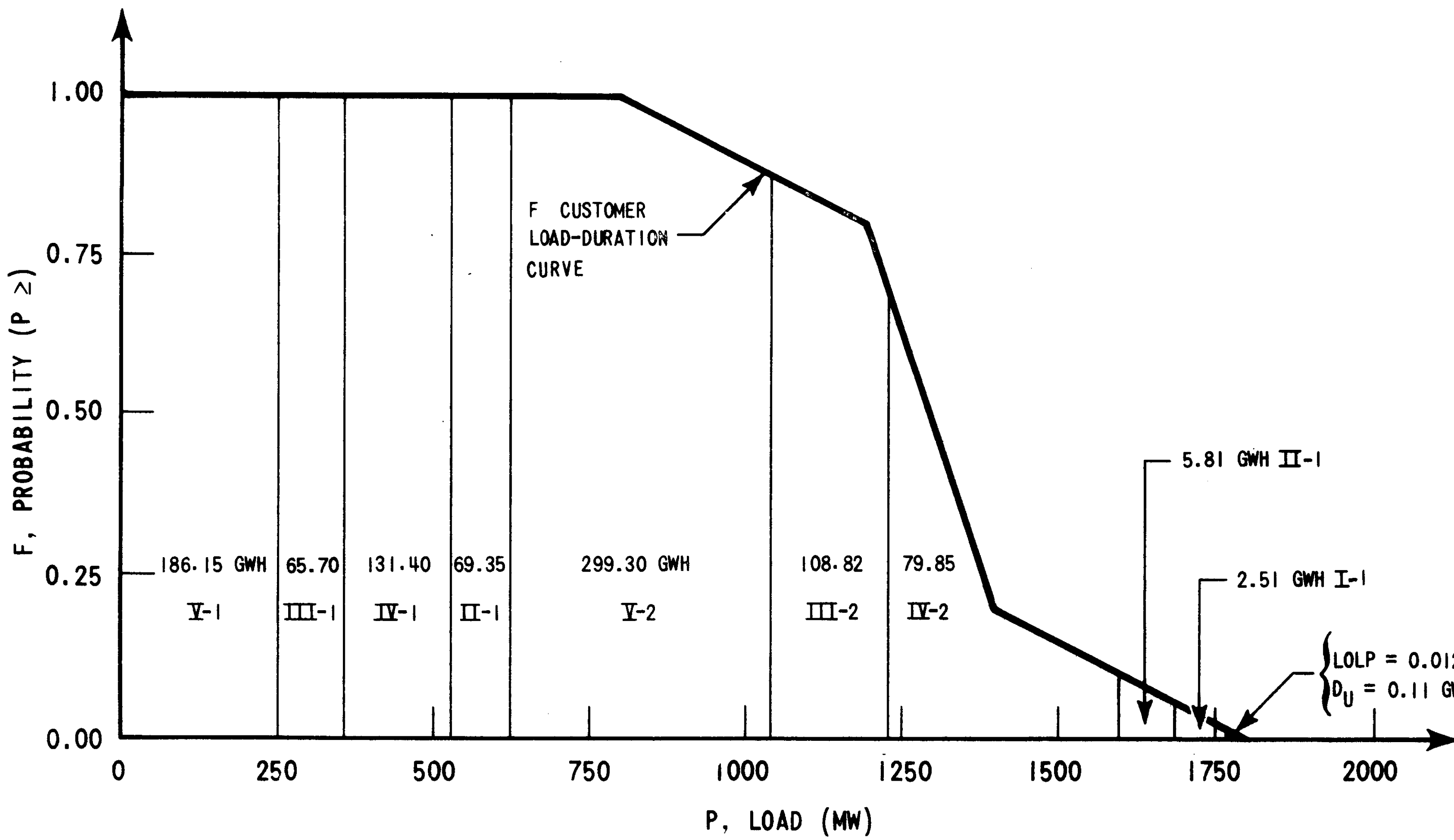
(See Appendix C for further details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 $)
I	1	9 (last)	- 0 -	- 0 -
II	1	4	73.00	401.5
	2	8	- 0 -	- 0 -
III	1	2	73.00	166.4
	2	6	73.00	138.7
IV	1	3	146.00	572.3
	2	7	29.20	97.0
V	1	1 (first)	219.00	492.8
	2	5	335.80	574.2
	Utility Production		949.00	2442.9
	Emergency Purchases (at 10$/MWH)		- 0 -	- 0 -
	Total		949.00	2442.9

Loss-of-Load Probability, LOLP = 0%

Figure 2.13

Production Scheduling for Example 2 (Deterministic Scheduling Using Reduced Rated Capacities)



One of the first attempts at accounting for these forced-outages was to reduce each capacity increment by its nonperformance probability. A 200-MW unit performing 90% of the time was treated as a 180-MW unit performing 100% of the time. Table 2.4 summarizes the energy and cost totals for this example.

A more elegant means of incorporating forced-outages in production scheduling has been developed (10,19) and is portrayed as Example 3 in Figure 2.14. The abscissa has been relabeled the equivalent load P_e signifying the stochastic or random nature of those units on forced-outages. The original normalized customer load-duration curve has been relabeled F_D, the "direct" customer demand to signify that each increment is directly responsible for satisfying customers within its section of the curve. However, if increment V-2 is off-the-line due to a forced-outage, increments of other units higher in the loading order (i.e., to its right) possess excess capacity capable of satisfying the customers V-2 is temporarily failing to serve. These customers are the direct responsibility of V-2 but are also the indirect responsibility of the other units. This additional indirect demand on all partially loaded unit increments is indicated by F_I. The resultant total equivalent demand F_e on each increment (derived in detail in Chapter 3) is given by

$$F_e(P_e) = F_D(P_e) + F_I(P_e) \tag{2.23}$$

Forced-outages affect not only the demand on each increment, but also the increment's production. If the unit only performs 90% of the time, then it is expected that only 90% of its demand will be served. Recalling from Section 2.1.2.2 that p_r is the unit's performance probability, Equation (2.19) becomes,

TABLE 2.4

Example 2 on Reference Utility System:
"Deterministic Model (Reduced Capacities)"

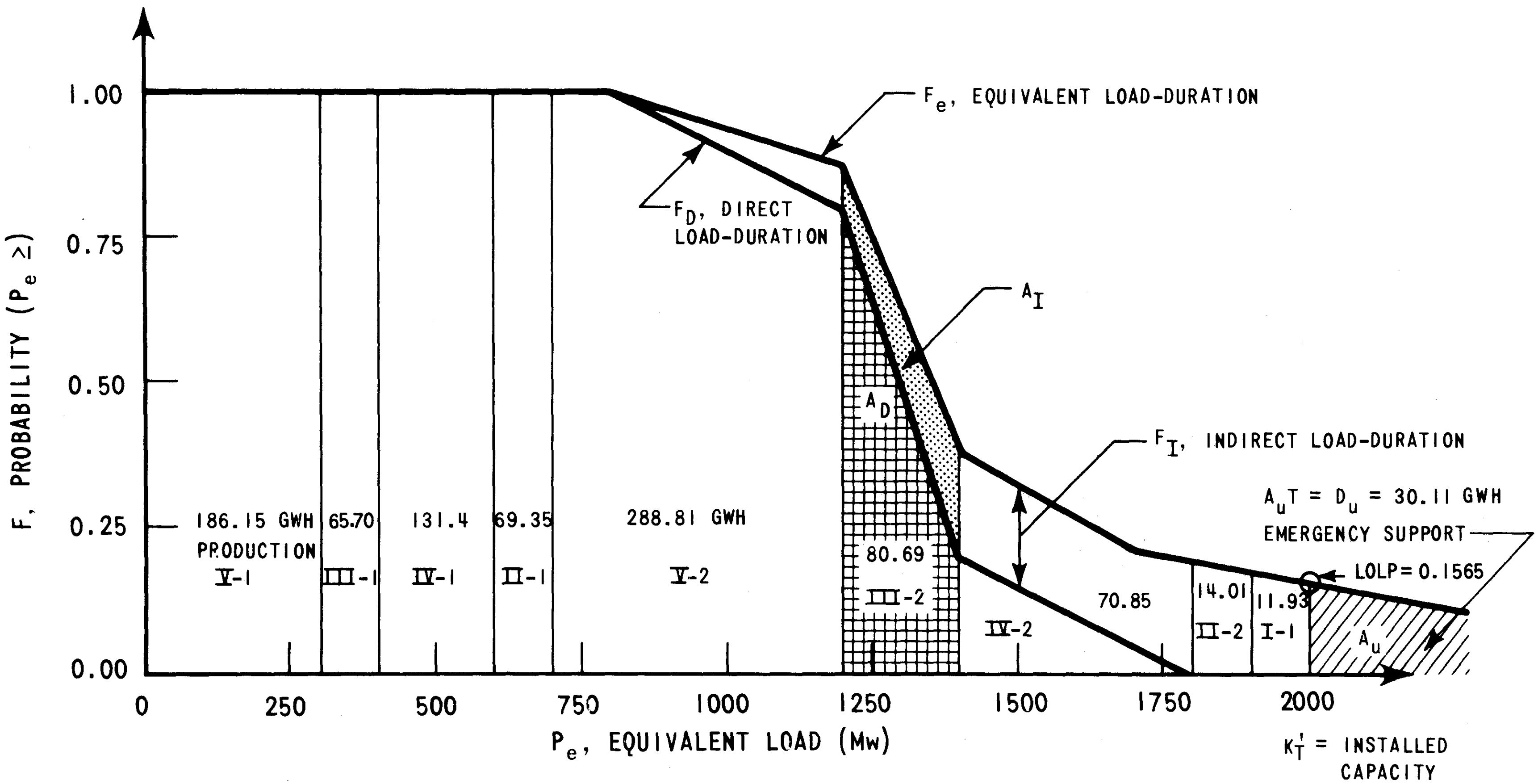
(See Appendix C for further details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 $)
I	1	9	2.51	40.7
II	1	4	69.35	381.4
	2	8	5.81	24.7
III	1	2	65.70	149.8
	2	6	108.82	206.8
IV	1	3	131.40	515.1
	2	7	79.85	265.1
V	1	1	186.15	418.8
	2	5	299.30	511.8
Utility Production			948.89	2514.2
Emergency Purchases (at 10$/MWH)			0.11	1.1
Total			949.00	2515.3

Loss-of-Load Probability, LOLP = 1.25%

Figure 2.14

Production Scheduling for Example 3 (With Forced-Outages)



$$E_{ri} = T' p_r \int_{P^\circ_{ri}}^{P^\circ_{ri} + \Delta K_{ri}} F_e(P_e)\, dP_e \qquad (2.24)$$

For this more general case, Equation (2.24) replaces Equation (2.19) for E_{ri}. However, Equations (2.20) to (2.22) remain unchanged.

Table 2.5 presents the production and cost summary for the Reference System as loaded in Figure 2.14. Notice that, in contrast to Figure 2.12 where peaking Unit I was not utilized to meet any direct demand, in Examples 2 and 3 the unit is subject to some indirect demand due to forced-outages of the other four units. Furthermore, some indirect customer demand extends beyond the available installed (on-line) capacity,

$$K'_T = \sum^{R'} K_{rI} \qquad (2.25)$$

As one measure of system reliability, D_U represents the energy unserved by the system's resources,

$$D_U = T' \int_{K'_T}^{\infty} F_e(P_e)\, dP_e \qquad (2.26)$$

"Expected unserviced energy ... is the expected curtailment or, more realistically, the expected emergency support required during" the time period (49).

Along with D_U, another measure of the system's reliability is the LOLP "loss-of-load-probability,"

$$\text{LOLP} = F_e(K'_T) \qquad (2.27)$$

the fraction of time the utility is unable to serve its customers with its own resources.

TABLE 2.5

Example 3 on Reference Utility System:
"Probabilistic Model (With Forced-Outages)"

(See Appendix C for further details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 $)
I	1	9	11.93	193.3
II	1	4	69.35	381.5
	2	8	14.01	59.5
III	1	2	65.70	149.8
	2	6	80.69	153.3
IV	1	3	131.40	515.1
	2	7	70.85	235.2
V	1	1	186.15	418.8
	2	5	288.81	493.9
Utility Production			918.89	2600.4
Emergency Purchases (at 10$/MWH)			30.11	301.1
Total			949.00	2901.5

Loss-of-Load Probability, LOLP = 15.6%

The quantitative details of Chapter 3 underlying the above discussion center around the calculation of F_e.

Far more germane to the current topic is how other unit types are handled by this model. As for fast-start peaking units, their high fuel cost places them very high in the loading order, but, when their turn finally comes, they are represented exactly like fossil units.

Nuclear units, with very low fuel costs, are also treated like fossil units but they come very early in the loading order, provided each has sufficient reactivity inventory to supply the resulting energy requirements. If not, they are treated like the scarce resource hydro units in the following Section 2.2.2.

2.2.2 Hydro Units

The important characteristic of hydro unit scheduling is making optimum use of a free, but scarce, resource. To do this requires finding that place in the loading order (see Figure 2.15) that utilizes all the available hydro energy while displacing the most costly fossil fuel possible. This is the same process often interpreted as "peak-shaving" the system demand (51).

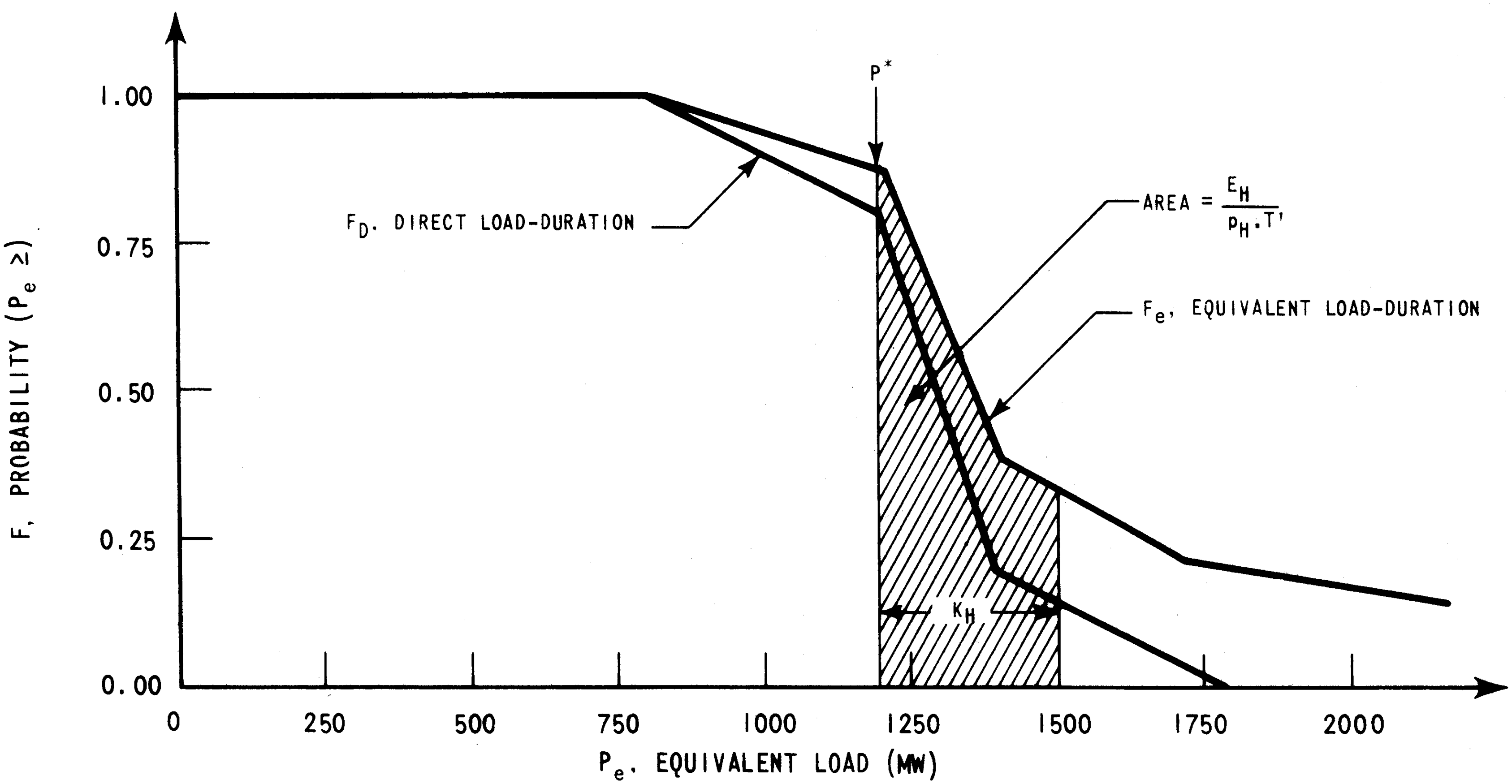
In terms of Equation (2.24), the optimum hydro loading point P^* is determined such that,

$$E_H = T' p_H \int_{P^*}^{P^*+K_H} F_e(P_e)\, dP_e \tag{2.28}$$

The cost of E_H is zero, but by utilizing E_H in this manner, each hydro megawatthour has been used to displace the most expensive fossil energy possible and thereby saving the maximum amount of money.

Figure 2.15

Hydro Unit Production Scheduling



Determining the hydro's position in the loading order given E_H is not difficult. The much more difficult question to answer is how much of the year's forecasted hydro resources to allocate to the period in question – i.e., determining E_H itself. Large scale computer programs (51) are required to tackle this problem on a realistic mixed fossil-hydro system. In order to avoid the hydro complexities in this early nuclear power management development work, hydro units were not included in this study.

2.2.3 Pumped-Hydro Units

The most complicated of all, pumped-hydro unit production scheduling requires not only hydro-type utilization of a fixed energy resource, but also involves the pumping of that resource into the reservoir prior to the generation. Figure 2.16 portrays the situation. Pumping involves an added direct demand on nonfully loaded increments low in the loading order, while generating involves using the stored energy to displace more expensive fossil equipment high in the order. If η_P and η_G are the net efficiencies in the pumping and generating modes, respectively, pumping is continued until the last increment of pumping energy costing λ_P just breaks even displacing an associated increment of generation saving λ_G. That is, pumping continues until,

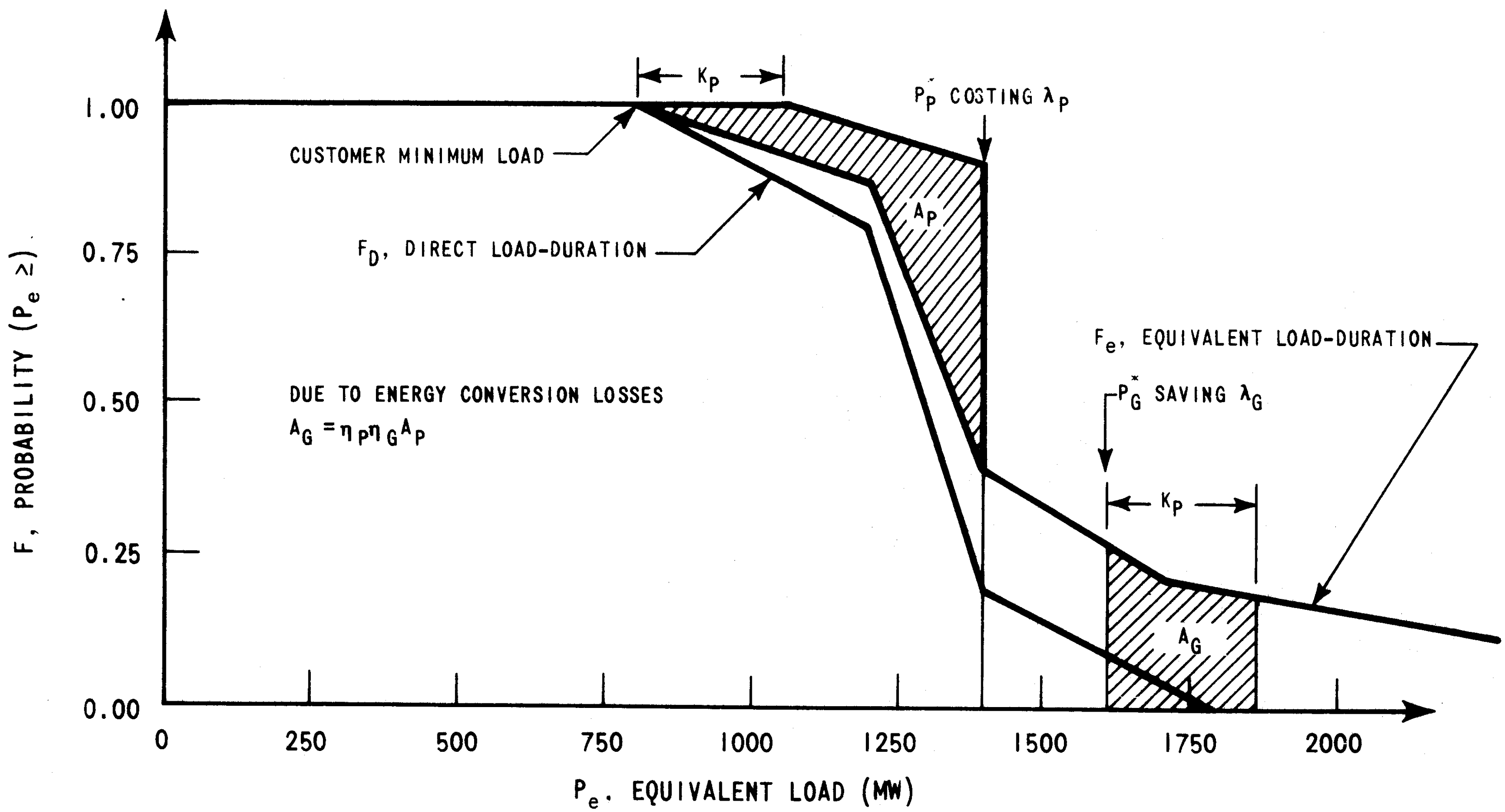
$$\lambda_G = \frac{\lambda_P}{\eta_P \eta_G} \tag{2.29}$$

However, this is subject to the constraint that the upper level reservoir capacity is not exceeded before pumping is terminated.

As with hydro units, pumped-hydro units were not included for further consideration in this initial development effort to avoid unnecessary complexity.

Figure 2.16

Pumped-Hydro Unit Production Scheduling



2.3 Complexities of Nuclear Power

The cost of fossil fuel is simply the cost of coal or oil plus shipping charges. Assuming a constant coal stockpile, newly delivered coal is burned immediately. From mine to ash, fossil fuel consumption requires only a matter of days.

Nuclear fuel, on the other hand, requires years to account for all cost components. Mining and enrichment occur nine months or more before insertion in the reactor. During the three years or more of irradiation, the energy potential is slowly extracted not only from this fuel batch but also from two or so others in the core. Three months or more after discharge, reprocessing occurs and fissile isotope credits are received. (Appendix H treats nuclear fuel cycle costs in more detail.) The net result is that the cost of a reactor's fuel over a time span of C cycles is a nonlinear, nonseparable function of the E_{rc} energy produced in each irradiation cycle,

$$\overline{TC}_r = \overline{TC}_r(E_{r1}, E_{r2}, \ldots, E_{rC}) \tag{2.30}$$

Qualitatively, the nonlinearity,

$$\overline{TC}_r \neq c_{r0} + c_{r1} \cdot E_{r1} + c_{r2} \cdot E_{r2} + \ldots + c_{rC} \cdot E_{rC} \tag{2.31}$$

results from the fact that, given the refueling batch fractions, cycle energy is approximately linear in feed enrichment, but the cost of this enrichment (i.e., separative work requirement) is nonlinear.

Preventing a more general uncoupling of the cycle energies,

$$\overline{TC}_r \neq C_{r0} + C_{r1}(E_{r1}) + C_{r2}(E_{r2}) + \ldots + C_{rC}(E_{rC}) \tag{2.32}$$

is the multi-irradiation (multi-zone) nature of today's LWR refueling schemes. The specification of reload enrichments requires not only reactivity allowance for the next cycle, but succeeding ones as well.

In summary, to calculate nuclear fuel costs, the cycle energies to the horizon of interest must be known.

In the early years of nuclear power, this stringent requirement did not pose a problem for conventional production scheduling models. With only a single nuclear plant on the system (see Figure 2.17), base-load operation was possible. That is, nuclear units were operated at full capacity whenever they were available. In addition, annual refueling meshed nicely with fossil maintenance plans and appeared to be reasonably economical. For the base-load ($F_e = 1$) case, Equation (2.24) reduced to

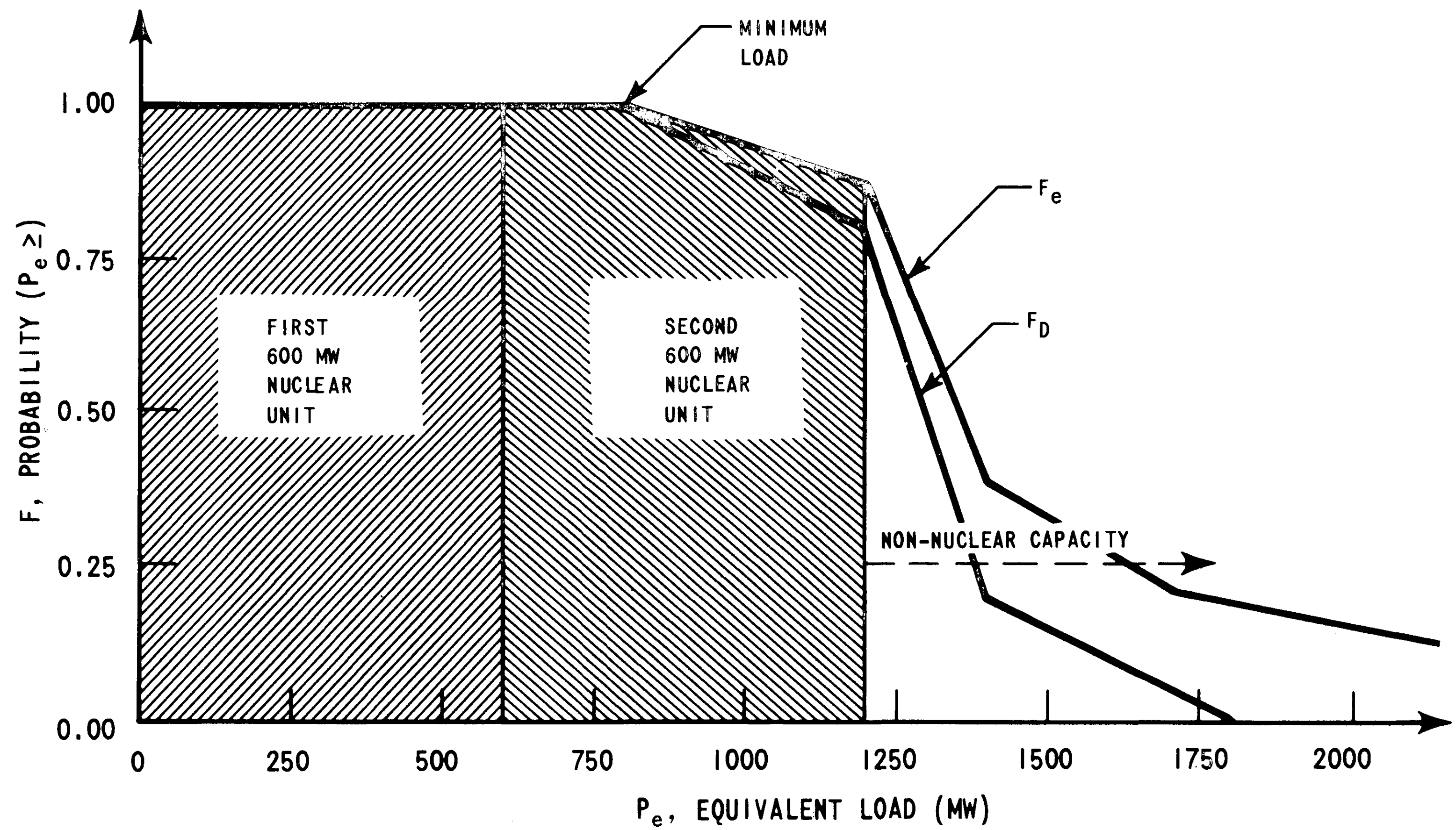
$$E_{rc} = p_r T'_{rc} K_r \tag{2.33}$$

for all cycles. If T'_{rc} was constant, the cycles energies to the horizon were the same and reactor steady-state fuel costs could be calculated and used for all cycles.

However, as nuclear capacity on the system increased, two problems became apparent. First, not all nuclear units could be base-loaded if total nuclear capacity was greater than the minimum load as in Figure 2.17. Equation (2.33) was no longer valid because the nuclear portion of the load-duration curve was no longer equal to 1.0 for all nuclear units. Which nuclear unit should occupy the base-load position? Inter-nuclear incremental cost competition had surfaced for the first time. Only rough estimates of nuclear fuel costs had been necessary to decide that all nuclear equipment was cheaper than all fossil equipment, but very refined costs were now needed to decide nuclear unit A versus nuclear unit B.

Figure 2.17

Nuclear Capacity Greater than Minimum Load



6253-8

Secondly, annual refueling created scheduling problems when each nuclear unit had to be refueled within every scheduling window. Coupled with decreasing nuclear load demand (F_e), what was the optimum cycle length for each reactor?

The net result was that cycle energies were no longer easily specified out to the horizon. The nuclear complications rendered previous utility system optimization models obsolete in the sense that operating plans based on them might be far from optimal.

The nuclear power management model to be put forth in Section 2.5 was developed to provide a modern model for utility system optimization, capable of handling nuclear plants explicitly. To do this, it must accurately predict cycle energies out to the horizon.

2.4 Comparison of Fossil and Nuclear Utility System Optimization

Incremental cost techniques for optimized fossil system dispatching (43,48) have been in use for many years. As Section 2.3 pointed out, nuclear plants present new problems due to the long-range time coupling inherent in the nuclear fuel cycle. Widmer *et al.* (57-59) optimized fossil-nuclear systems using nuclear incremental costs defined much differently from those of fossil plants. This section presents a parallel treatment of both fossil and nuclear incremental costs in order to point out the contrasting assumptions and results.

Consider the following general problem:

> Minimize total system cost (i.e., revenue requirements) from time 0 (zero) to the end of the horizon Z (on the order of ten years) for a system containing R generating units.

Fuel for each unit is assumed to be provided under several consecutive fuel contracts. The objective function is then:

$$\text{Minimize } \overline{TC} = \sum^{R} \overline{TC}_r(\Theta_{r1}, \Theta_{r2}, \Theta_{r3}, \ldots) \tag{2.34}$$

subject to the load constraint,

$$\sum^{R} P_r(t) = P(t) \tag{2.35}$$

If $H_r(P_r)$ represents the instantaneous heat input rate at power level P_r for the r^{th} unit, then from the end of the previous contract, $\tau_{r,c-1}$, to the end of current contract, τ_{rc}, the plant consumes thermal energy equivalent to

$$\Theta_{rc} = \int_{\tau_{r,c-1}}^{\tau_{r,c}} H_r(P_r)\, dt \tag{2.36}$$

2.4.1 Incremental Costs on All Fossil System

For fossil units, two important assumptions come into play:

a) the various fuel supply contracts for each generating unit are uncoupled:

$$\overline{TC}_r(\Theta_{r1}, \Theta_{r2}, \ldots) = \overline{TC}_{r1}(\Theta_{r1}) + \overline{TC}_{r2}(\Theta_{r2}) + \ldots \tag{2.37}$$

and b) the contract total cost $\overline{TC}_{rc}$ is linear in Θ_{rc}:

$$\overline{TC}_{rc} = \overline{TC}^{\circ}_{rc} + \bar{\phi}_{rc} \cdot \Theta_{rc} \tag{2.38}$$

where $\bar{\phi}_{rc}$ = levelized incremental thermal energy unit cost.

For an all fossil system, adding all C contracts for all the R units yields the objective function:

$$\overline{TC} = \sum^{R} \sum^{C} \left\{ \overline{TC}^{\circ}_{rc} + \bar{\phi}_{rc} \int_{\tau_{r,c-1}}^{\tau_{rc}} H_r(P_r)\, dt \right\} \tag{2.39}$$

Since one summation is over all contracts (i.e., cycles), all time from 0 to Z is included and that summation may be replaced by an integral over t. Defining

$$\overline{TC}^{\circ} = \sum^{R} \sum^{C} \overline{TC}^{\circ}_{rc} \tag{2.40}$$

then

$$\overline{TC} = \overline{TC}^{\circ} + \int_0^Z \left\{ \sum^{R} \overline{\phi}_{rc} H_r(P_r) \right\} dt \tag{2.41}$$

or more generally,

$$\overline{TC} = \overline{TC}^{\circ} + \int_0^Z f(t; P_1(t), P_2(t), \ldots) \, dt \tag{2.42}$$

Since the objective function is a definite integral over t, the calculus of variations (32) allows immediate reduction of the problem. Employing the integrand of Equation (2.42) and the load constraint Equation (2.35) to form the auxiliary function ψ_F,

$$\psi_F = f(t; \text{all } P_r; \text{no derivatives } \dot{P}_r) + \lambda_F(t) \left\{ P(t) - \sum^{R} P_r \right\} \tag{2.43}$$

Immediately, the optimum behavior of each $P_r(t)$ is given by Euler's equation:

$$\frac{d}{dt}\left\{ \frac{\partial \psi_F}{\partial \dot{P}_r} \right\} - \frac{\partial \psi_F}{\partial P_r} = 0 \tag{2.44}$$

Since there is no dependence of ψ_F on $\dot{P}_r$, Equation (2.44) reduces to

$$\frac{\partial \psi_F}{\partial P_r} = 0 = \frac{\partial f(\ldots)}{\partial P_r} - \lambda_F(t) \tag{2.45}$$

Substituting for f(...) using Equation (2.41) and rearranging,

$$\lambda_F(t) = \overline{\phi}_{rc} \frac{\partial H_r(P_r)}{\partial P_r} \tag{2.46}$$

Since $\frac{\partial H_r(P_r)}{\partial P_r}$ equals the incremental heat rate at P_r, $h_{inc_r}(P_r)$,

$$\lambda_F(t) = \bar{\phi}_{rc} \cdot h_{inc_r}(P_r) \tag{2.47}$$

for all R units at the same time t, subject to Equation (2.35).

The Lagrangian multiplier $\lambda_F(t)$ represents the time-varying incremental energy cost (i.e., proportional to $\bar{\phi}_{rc}$ discounted dollars over undiscounted energy) at which all fossil units on the system should be operating for minimum system cost. Equation (2.47) is the same result Kirchmayer obtained (43) with the a priori knowledge that instantaneous optimization gave the long-term optimum rather than beginning with the long-term objective function, Equation (2.34).

Typical values for present day fossil systems involve unit fuel costs of 25 to 50 ¢/Mega BTU and incremental heat rates as low as 8000 BTU/kwhe at night to over 15,000 BTU/kwhe (8 to 15 Mega BTU/MWH) during the hours of peak demand. System incremental fossil fuel cost thus varies on a daily basis from 2.0 to 7.5 $/MWH.

2.4.2 Incremental Costs on All Nuclear System

For nuclear reactors, which have coupled, nonlinear cycle costs, the two assumptions made for fossil units [Equations (2.37) and (2.38)] do not hold. However, the data of Figure 2.18 indicates that for today's LWR's, the incremental heat rate of a nuclear plant is approximately constant over the operating range of interest (40% to 100% of full power),

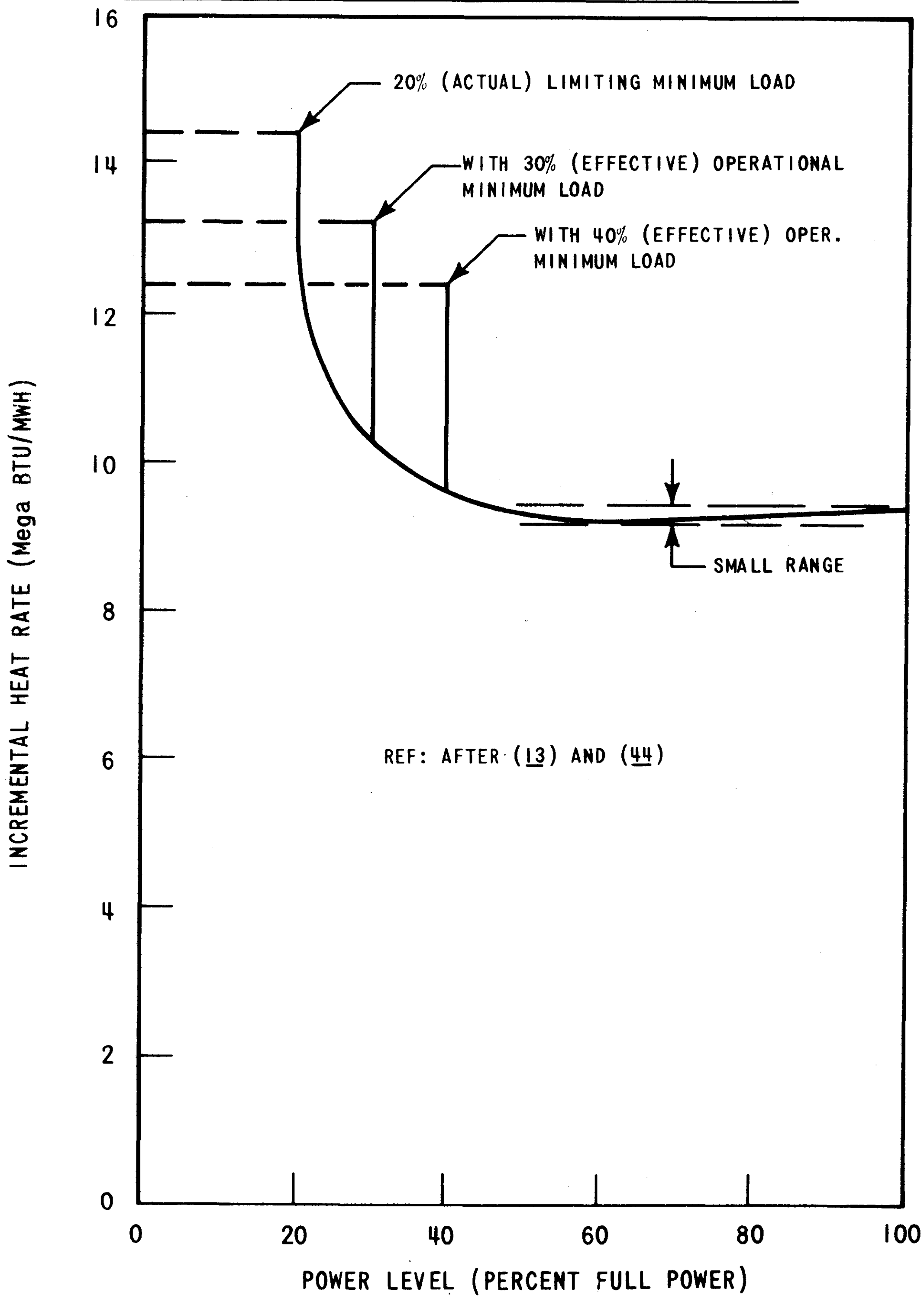
$$h_{inc_r} \neq f(P_r) \tag{2.48}$$

Extrapolating the heat rate curve $H_r(P_r)$ back to $P_r = 0$ at the constant incremental heat rate h_{inc_r},

$$H_r(P_r) = H_r^{\circ} + h_{inc_r} \cdot P_r \quad \text{for } P_r \gg 0 \tag{2.49}$$

Figure 2.18

Incremental Heat Rates for Typical Light Water Reactor



Since P_r (and hence H_r) $\equiv 0$ during the refueling downtime following shutdown at $\tau_{r,c-1}$ (the end of the irradiation cycle), Equation (2.36) need only be integrated over the available generating hours T'_{rc},

$$\Theta_{rc} = \int_{\tau_{rc}-T'_{rc}}^{\tau_{rc}} \left(H_r^\circ + h_{inc_r} \cdot P_r\right) dt \tag{2.50}$$

Assuming the nuclear units to be "must-run" units (see Section 2.4.3), they can be expected to perform at least at minimum load (i.e., $P_r \gg 0$) for $p_r T'_{rc}$ hours.

Hence,

$$\Theta_{rc} = H_r^\circ p_r T_{rc} + h_{inc_r} \int_{\tau_{rc}-T'_{rc}}^{\tau_{rc}} P_r \, dt \tag{2.51}$$

or,

$$\Theta_{rc} = H_r^\circ p_r T'_{rc} + h_{inc_r} E_{rc} \tag{2.52}$$

Since Θ_{rc} is linear in E_{rc}, direct substitution into the objective function is possible:

$$\overline{TC} = \sum^{R} \overline{TC}_r(\Theta_{r1}, \Theta_{r2}, \ldots) = \sum^{R} \overline{TC}_r(E_{r1}, E_{r2}, \ldots) \tag{2.53}$$

In order to transform the customer loads into corresponding energy units, the time horizon is segmented into Z convenient time periods on the order of weeks. Then, the right-hand side of Equation (2.35) is integrated over each time period to yield period energy demand,

$$D_p = \int_{t_{p-1}}^{t_p} P(t) \, dt \tag{2.54}$$

Assuming there are enough nuclear units on the system to prevent loss-of-load, the period energy demand must be generated by the R units in that period,

$$D_p = \sum^{R'} E_{rcp} \tag{2.55}$$

During a particular reactor-cycle, the energy must be the sum of the reactor's production in each of the included periods,

$$E_{rc} = \sum^{p \text{ in } c} E_{rcp} \tag{2.56}$$

Thus, the independent variables in Equation (2.53) can be further subdivided into period energy productions,

$$\overline{TC} = \sum^{R} \overline{TC}_r \left(\{E_{rcp}\}_r\right) \tag{2.57}$$

To form the ψ_N auxiliary function of Equation (2.57), the constraints [Equation (2.55)] are incorporated using a λ_{N_p} Lagrangian constant for each period,

$$\psi_N = \sum^{R} \overline{TC}_r\left(\{E_{rcp}\}_r\right) + \sum^{Z} \lambda_{N_p} \cdot \left(D_p - \sum^{R} E_{rcp}\right) \tag{2.58}$$

which is only a function of the E_{rcp} set, $\{E_{rcp}\}$.

For ψ_N to be a relative minimum (31), the following must hold for all r, all c and all p:

$$\frac{\partial \psi_N}{\partial E_{rcp}} = 0 = \frac{\partial \overline{TC}_r}{\partial E_{rcp}} - \lambda_{N_p} \tag{2.59}$$

Therefore, during each period of the optimum,

$$\lambda_{N_p} = \frac{\partial \overline{TC}_r}{\partial E_{rcp}} \tag{2.60}$$

for the pertinent cycles of each reactor, subject to Equation (2.55).

Since the E_{rcp} sum linearly to give the cycle energy E_{rc} [Equation (2.56)],

$$\frac{\partial(\ldots)}{\partial E_{rcp}} = \frac{\partial(\ldots)}{\partial E_{rc}} \qquad \text{for all p in c} \tag{2.61}$$

the optimality condition Equation (2.60) can be restated as

$$\lambda_{N_p} = \frac{\partial \overline{TC}_r}{\partial E_{rc}} \tag{2.62}$$

The Lagrangian constant λ_{N_p} (with units identical to λ_F, discounted dollars over undiscounted energy) represents the incremental energy cost at which the pertinent refueling cycle of each nuclear unit should be designed and operated. The coupling of nuclear energies in the objective function prevents the simplifications made in the fossil case. However, the approximately constant incremental heat rate of today's nuclear units (above 40% of capacity) permits a different simplification and leads to Equation (2.62).

To contrast Equations (2.47) and (2.62) in more general terms, consider that

$$\lambda_{N_p} = \frac{\partial \overline{TC}_r}{\partial E_{rc}} = \frac{\partial \overline{TC}_r}{\partial \Theta_{rc}} \frac{d\,\Theta_{rc}}{d\,E_{rc}} \tag{2.63}$$

Differentiating Equation (2.52),

$$\frac{d\Theta_{rc}}{dE_{rc}} = h_{inc_r} \tag{2.64}$$

Hence, for nuclear units,

$$\lambda_{N_p} = \frac{\partial \overline{TC}_r}{\partial \Theta_{rc}} \cdot h_{inc_r} \tag{2.65}$$

resulting in nuclear dispatching on a cycle-by-cycle basis using energy-related incremental costs.

Fossil units, on the other hand, are dispatched using instantaneous incremental costs related to power level [Equation (2.47)],

$$\lambda_F(t) = \bar{\phi}_{rc}\, h_{inc_r}(P_r(t)) \tag{2.66}$$

Substituting the definition of h_{inc_r} [Equation (2.3)],

$$\lambda_F(t) = \bar{\phi}_{rc} \cdot \frac{dH_r(P_r)}{dP_r} \tag{2.67}$$

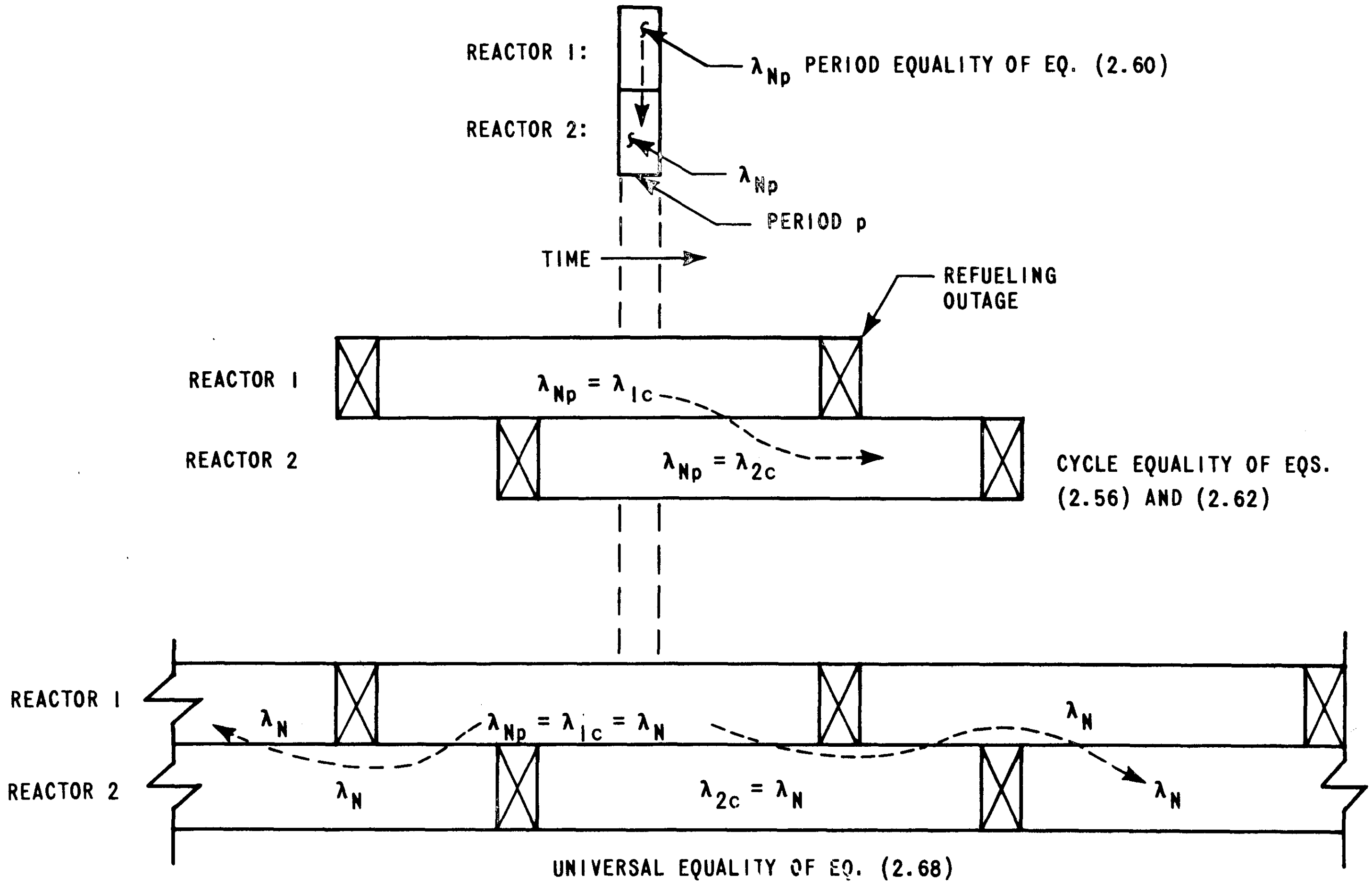
Comparing Equations (2.65) and (2.67), the former is in terms of energy because the "incremental" effect or derivative is in the fuel cost component related to cycle energy, not the incremental heat rate h_{inc_r} which is assumed constant for any power level. The reverse is true for the latter's fossil incremental cost. The λ_F is power level dependent because the h_{inc_r} is recognized as a function of $P_r(t)$; the fuel cost component $\partial\overline{TC}_r/\partial\Theta_{rc}$ is assumed a constant $\bar{\phi}_{rc}$ independent of cycle energy.

Another conclusion regarding nuclear incremental costs can be deduced by considering the cycle-to-cycle overlap of two reactors as in Figure 2.19. In the p^{th} period, both reactors have the same incremental cost per Equation (2.60). Going one step further, Equations (2.56) and (2.62) indicate that within the range of periods in the companion cycles, the incremental cost remains the same. Finally, as the cycle ends for Reactor 1, λ_{N_p} remains at the same level due to Reactor 2. But, Equation (2.62) states that Reactor 1's next cycle should also be designed at this same level to maintain the equality. Thus, the overlapping of reactor-cycles creates a constant λ_{rc} regardless of reactor and cycle. Consequently,

$$\lambda_{N_p} = \lambda_N = \text{constant for all } p \tag{2.68}$$

Figure 2.19

Consequences of Period Incremental Cost Equality



6253-4

and

$$\lambda_N = \frac{\partial \overline{TC}_r}{\partial E_{rc}} \tag{2.69}$$

for all r and all c simultaneously.

A consequence of Equation (2.69) is that steady-state would never be reached. Due to the discounting of dollars, but not energy, it becomes profitable to generate more and more energy in each succeeding cycle, relying on the increasing discount factor to appropriately reduce the additional undiscounted cost. This is the case for cycles 1 through 3 of Figure 2.20. While Equation (2.69) indicates the profitable thing-to-do, it does not indicate how feasible it is. Cycles 4, 5 and 6 of Figure 2.20 are examples of steady-state designs (with decreasing incremental costs) being forced by a constraint, namely, that the capacity factor cannot be greater than one. In other words, generation cannot be postponed. Demand must be satisfied instantaneously, not four years later. Generation can be shifted from one reactor to another on a day-to-day basis but the total production each period must be met [Equation (2.55)].

The net result is the primary Conclusion I [Equation (2.70)], relating a strong dependence between pertinent cycle incremental costs for each reactor during each period and a secondary Conclusion II [Equation (2.71)] relating an idealized state that may not be attainable:

Conclusion I:

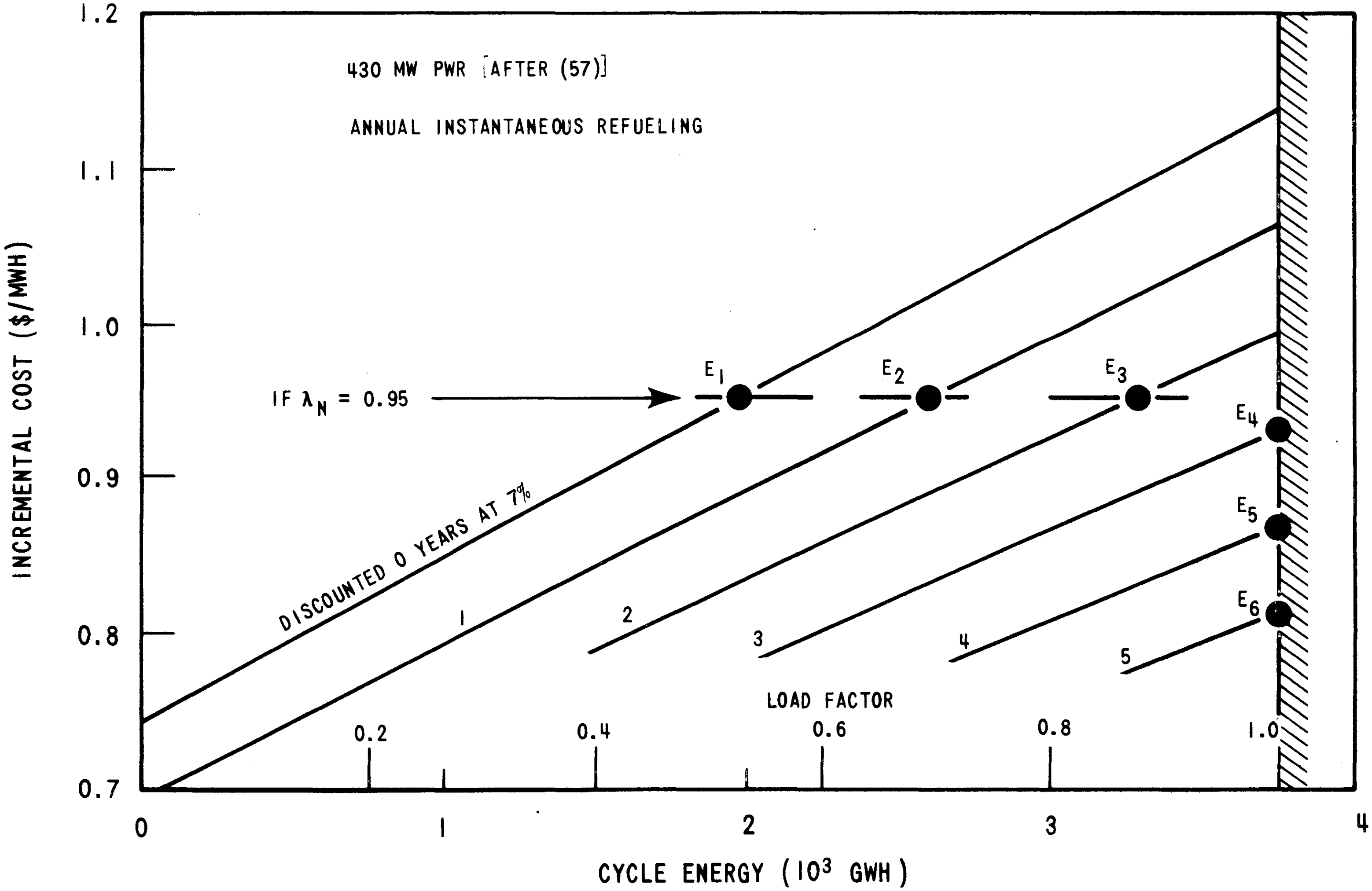
At the optimum reactor-cycle energies,

$$\lambda_{N_p} = \frac{\partial \overline{TC}_r}{\partial E_{rc}} \tag{2.70}$$

during each period for the pertinent cycle of each reactor.

Figure 2-20

Consequences of Conclusion II Incremental Costs versus Cycle Energy



Conclusion II:

At the optimum reactor-cycle energies,

$$\lambda_N = \frac{\partial \overline{TC}_r}{\partial E_{rc}} \tag{2.71}$$

for all periods, all cycle and all reactors simultaneously, subject to physical constraints.

As for typical values of λ_{N_p} and λ_N, the results of Widmer (57), Kearney (41) and Watt (55) as well as Section 5.6.3 indicate optimum mid-range nuclear incremental costs in the range of 0.9 to 1.6 \$/MWH.

2.4.3 Optimization of a Mixed System

The two previous sections have indicated how an all fossil or an all nuclear system would meet the same loads at minimum total system cost. This section endeavors to show the reasoning behind segmenting the more realistic mixed fossil-nuclear system into an equivalent "all fossil plus all nuclear" system such that,

$$D_{T_p} = E_{F_p} + E_{N_p} + D_{U_p} \tag{2.72}$$

Given the normalized customer load-duration curve and the available generating equipment, a startup and loading order is required by the production scheduling model. The first consideration is the placement of unit increments under the "knee" of the load-duration curve, i.e., below the minimum load (see Figure 2.12) where they will be operated even during periods of lowest system demand, such as the early morning hours. These unit increments are typically the minimum loads on all of the large units (e.g., rated capacity ≥ 300 MW). If such units were shut down overnight due to economics alone, minimum shutdown times and other engineering problems might prevent the unit from being in service when it was needed for the next day's peak. Losing such a large unit creates reliability problems. Thus, the operating philosophy is that all large units must be running at least at minimum load if possible. If the minimum load is too low to permit this, either the smallest of the "must-run" units is shut down or its excess capacity is sold to neighboring utilities on an hour-by-hour economy interchange basis.

For a mixed fossil-nuclear system, this must-run philosophy results in grouping all nuclear minimums at the lowest point in the startup and loading order. Next comes the must-run fossil minimums in order of decreasing size. Figure 2.12 portrayed the must-run units in Examples 1 to 3 for a lower limit of 200 MW.

The startup and loading order for the rest of the system is determined by noting two important points. First, on a time scale where reload fuel is being designed, nuclear units are not energy-limited, and nuclear production should not be scheduled as scarce resource. Secondly, even with fossil fuel costing as little as 25¢/MegaBTU, the best-plant fossil incremental costs are at least 2.0 $/MWH (see Section 2.4.1). Since even the highest nuclear incremental fuel costs are less than 1.6 $/MWH (see Section 5.6.3), nuclear power should be operated so as to displace maximum fossil energy. In other words, the greatest potential for cost savings in each period is in maximizing nuclear production E_{N_p} vis-à-vis fossil production E_{F_p}. (D_{U_p} is invariant given the on-line equipment.) Mathematically, total period cost is a minimum when

$$D_{T_p} = E_{F_p}^{min} + E_{N_p}^{max} + D_{U_p}^{\circ} \tag{2.73}$$

The above loading order does just that, maximizing E_{N_p} and resulting in N_p, the system's nuclear potential for the period,

$$N_p = E_{N_p}^{max} \tag{2.74}$$

Thus, after starting up and raising to minimum power the must-run units that are not shut down regularly, all nuclear plants are loaded to full power in accordance with system demands. As demand continues to increase, all the remaining fossil power is loaded in order of increasing incremental cost.

Figure 2.11 portrayed such a startup and loading order applied to the Reference System in Examples 1 to 3. It is now a simple matter to separate the "all nuclear" system from the "all fossil" system. Performing the above for each time period of a study thus separates the fossil and nuclear portions of the system. These two subsystems can then be optimized using the techniques of Sections 2.4.1 and 2.4.2, respectively.

The key assumption leading to the fossil-nuclear dichotomy, bears repeating since it is the basis of the entire nuclear power management model presented in the next section.

$$\lambda_{N_p} < \lambda_F(t) \quad \text{for all } t \text{ and } p \tag{2.75}$$

2.5 A Nuclear Power Management Multi-Year Model

A nuclear power management multi-year model currently under development (23,34) contains four submodels as presented in Figure 2.21. The overall model's purpose is to supply the utility system planner with the following outputs:

(1) Optimum schedule for fossil maintenance and nuclear refueling,

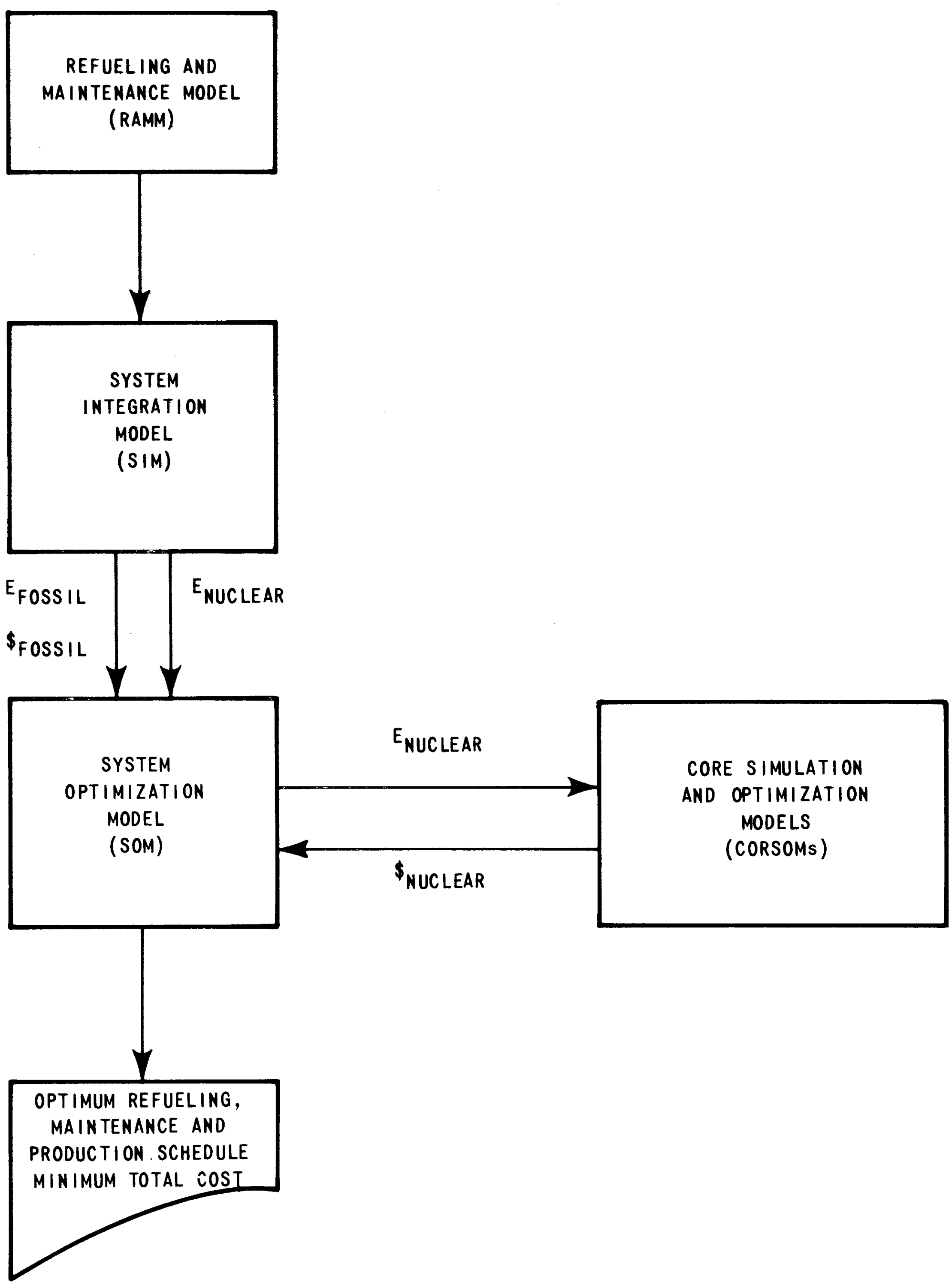
(2) Associated optimum production schedule and

(3) The resultant fuel requirements.

6253-13

Figure 2.21

Nuclear Power Management
Multi-Year Model



Operation of the overall model begins within the Refueling and Maintenance Model (RAMM). Incorporating such inputs as load forecasts, maintenance requirements and scheduling constraints, the RAMM determines a number of feasible multi-year refueling and maintenance schedules. Each schedule is a mutually exclusive, alternative mode of operating the entire system over the multi-year horizon. The purpose of the rest of the overall model is to determine which of the possible alternative strategies results in minimum total cost.

Strategy-by-strategy evaluation begins in the System Integration Model (SIM). For each strategy, the SIM integrates the utility's available equipment, operating practices, etc. into a realistic utility simulation model. Production scheduling is optimized so as to meet customer load demand by maximizing nuclear energy and minimizing fossil energy and fossil cost (see Section 2.4.3).

The task of the System Optimization Model (SOM) is then to optimize the operation of the nuclear portion of the system (see Section 2.4.2) so that the nuclear energy $E_{Nuclear}$ is produced at minimum cost $\$_{Nuclear}$. To do this, the SOM postulates reactor-by-reactor multi-year production schedules which are then passed to Core Simulation and Optimization Models (CORSOM's) for each reactor unit or type (PWR, BWR, LMFBR, etc.). With each production schedule specified to the horizon (see Section 2.3), each CORSOM is then able to optimize its reload parameters of batch size and enrichment, minimizing the total fuel cost for the particular reactor. In addition, the CORSOM calculates nuclear incremental costs for each of the cycles.

With all reactors optimized for the given schedules, the SOM begins a second iteration by using the CORSOM's incremental nuclear energy costs to postulate a better reactor-by-reactor multi-year production schedule. Iterations continue until the system-wide production schedule converges, giving minimum system nuclear cost $\$_{Nuclear}$.

The total system cost for the particular refueling and maintenance strategy under investigation is then merely the sum of $\$_{Fossil}$ and $\$_{Nuclear}$.

After evaluating all possible alternative strategies in this manner, the overall optimum system strategy is the one resulting in the minimum total system cost.

Though the above discussion and, in fact, this entire work assumes only fossil and nuclear equipment exist on the system, the general structure of the overall model holds even if hydro and pumped-hydro equipment have been installed.

The development of the complete nuclear power management multi-year model is a very large task. However, the four submodels represent convenient building blocks suitable for somewhat independent development. However, model interface problems must be considered. Ideally, the models ought to be coupled together like the boxcars of a train, not nailed together like the tracks.

In the context of the Commonwealth Edison-sponsored utility system optimization research project at the Massachusetts Institute of Technology, development of a RAMM was assumed by the project sponsor (20). Development of a pressurized water reactor CORSOM was undertaken at MIT by Kearney (41) and Watt (55). The concluding sections of this chapter emphasize these two models, indicating the important aspects relative to RAMM and CORSOM development and their interfacing with the rest of the model (see Figure 2.22). As the title indicates, the work reported here deals specifically with the development of the remaining SIM and SOM.

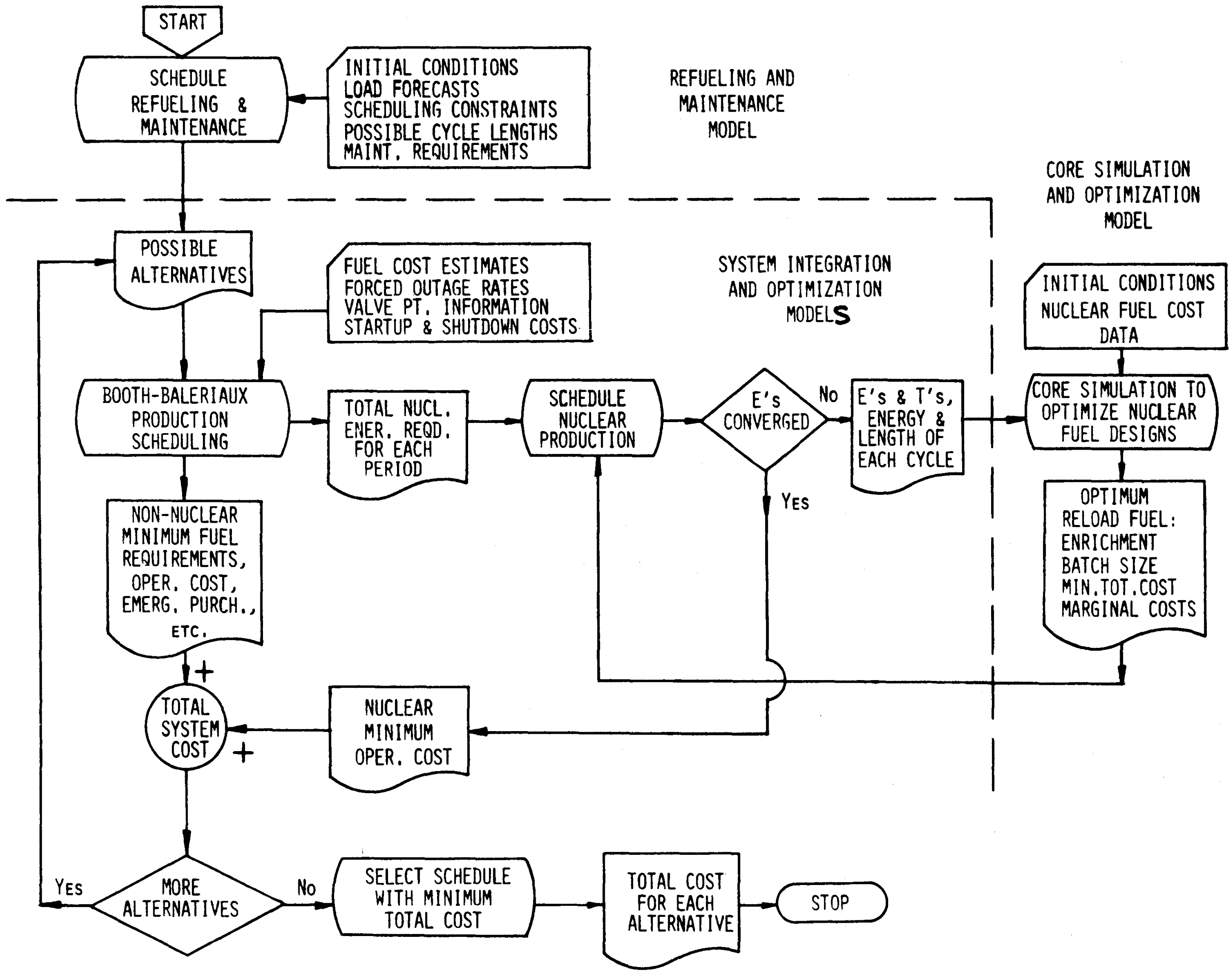


Figure 2.22 NUCLEAR POWER MANAGEMENT MULTI-YEAR MODEL

2.5.1 Refueling and Maintenance Model (RAMM)

Taking due account of the five inputs indicated in Figure 2.22, the RAMM's purpose is to generate possible alternative strategies for further investigation by the rest of the nuclear power management multi-year model.

The output of the RAMM is anticipated by the SIM in the form of either a set of downtime dates for each unit on the system or a period-by-period (on the order of one to four weeks per period) maintenance schedule indicating which units are down in each period.

Also desirable is a RAMM ranking of the strategies in order of anticipated desirability. That is, "ballpark" estimates of economics and reliability ought to indicate Strategy 1 is most likely to be optimum, while Strategy n (n ~ 100), though feasible, is highly unlikely to be economically attractive and/or a reliable operating scheme. Such a ranking would decrease computing requirements by permitting the detailed evaluation of only those strategies with a reasonable chance of competing for the optimum.

With regard to the testing of the nuclear power management model in Chapter 5, Sections 5.2 and 5.3.3 indicate the RAMM utilized in the evaluation.

2.5.2 System Integration Model (SIM)

Chapter 3 is devoted to a detailed discussion of the SIM and, in particular, the Booth-Baleriaux utility model.

2.5.3 System Optimization Model (SOM)

Chapter 4 is devoted to a detailed discussion of the SOM.

2.5.4 Core Simulation and Optimization Model (CORSOM)

At each iteration in Figure 2.22, the CORSOM accepts a new set of cycle energies (E's) for its reactor and, in point of fact, the same set of cycle lengths (T's) associated with the particular possible alternative strategy. After simulating core physics-depletion and optimizing the reload parameters (batch size and enrichment), it is required to return to the SOM only two specific types of information:

(1) the minimum total reactor fuel cost ($\overline{TC}_r$) and

(2) the nuclear incremental cost curve for each reactor reload batch,

$$\lambda_{rc}(E_{rc}) = \frac{\partial \overline{TC}_r}{\partial E_{rc}} \qquad (2.76)$$

Specific information about the fuel designs is not needed by the SOM. As long as each CORSOM is properly matched with the reactor unit index that it represents, the SOM does not care which unit indexes are PWR's, BWR's, HTGR's or fast breeders. Of course, management personnel need fuel design information and it must, therefore, be available in the printed output received directly from the CORSOM (at least, for the final fully-converged iteration).

The details of such a PWR core model can be found in the work of Kearney (41) while the techniques of incremental costing can also be found in the work of Widmer (57) and Watt (55).

With regard to the testing of the nuclear power management model in Chapter 5, Section 5.2 and Appendix H detail the CORSOM utilized in the evaluation.

CHAPTER 3

THE SYSTEM INTEGRATION MODEL

3.1 Overview of the SIM

Many aspects of the System Integration Model (SIM) have already been described in Chapter 2. The emphasis in the current chapter will be on detailing the Booth-Baleriaux probabilistic utility model and describing the calculation of the various cost components.

The SIM has as its basic purpose the simulation of multi-year utility operation. To do this, it must integrate the following information into a representative utility system model:

(1) Forecasts of customer loads,

(2) Generating equipment characteristics,

(3) Forecasts of fuel costs,

(4) Maintenance schedules and

(5) Operating constraints.

To portray system operation more accurately, the multi-year horizon is divided into much smaller time periods, on the order of a few weeks. Periods shorter than a week create an undue computational burden. On the other hand, periods longer than a month are precluded by the necessity of discretely representing scheduled maintenance outages which are usually two to four weeks in length.

These time periods are then simulated individually in chronological sequence. Forecasted loads for each period (Item 1 above) are represented by a normalized customer load-duration curve such as the month on the Reference Utility System presented in Figure 2.9. Thermal energy costs (Item 3) are combined with the characteristics of the generating units per Equation (2.18) to yield unit incremental costs. Any unavailable units down due to scheduled maintenance (Item 4) are treated as non-existent for that period. The next step is the establishment of the startup and loading order (see Section 3.2) for the remaining on-line units. It is in this order that various operating constraints (Item 5), such as "spinning reserve" and "zone-loading" requirements are incorporated. Production scheduling of the resulting system representation is performed using the Booth-Baleriaux probabilistic utility system model (see Section 3.3).

The qualitative discussion of the Booth-Baleriaux model presented in Section 2.2.1 developed cost components for most of the required period expenditures enumerated in Section 2.1.3:

(1) X_F = Fossil fuel expense related to E_F energy production,

(2) X_N = Nuclear fuel expense related to E_N energy production,

(3) X_S = Combined fossil and nuclear startup-shutdown cost (not discussed in Chapter 2) and

(4) X_U = Expense related to D_U emergency energy purchases.

Later, Section 2.3 pointed out that the complexities of nuclear power preclude a priori knowledge of nuclear fuel costs X_N except for the special case of all nuclear base-load operation. Nevertheless, by incorporating the nuclear versus fossil incremental cost argument of Section 2.4.3 to sub-optimize each period, the SIM is able to mark time by calculating in its place the system nuclear potential N for each period. The responsibility for optimizing and costing inter-nuclear production of this energy rests with the System Optimization Model (SOM).

Even an a priori estimate of unit nuclear fuel costs ϕ_{N_r} is sufficiently accurate for the nuclear component of system startup-shutdown costs since $(X_S)_N$ represents only a small fraction of total nuclear production fuel cost X_N,

$$(X_S)_N << X_N \tag{3.1}$$

Furthermore, for nuclear units (all assumed to be must-run units), there are very few startup-shutdowns since the units are always running. Hence, nuclear startup cost is also much less than fossil startup cost,

$$(X_S)_N << (X_S)_F \qquad (3.2)$$

Thus, an initial error in ϕ_{N_r} has a very small effect on total period expenses.

In summary, the actual period-by-period output of the SIM consists of:

(1) X_F = Fossil fuel expense related to E_F^{min} energy production (see Section 3.3),

(2) N = Nuclear potential equal to E_N^{max} energy production (see Section 3.3.3),

(3) X_S = Combined fossil and nuclear startup-shutdown cost (see Section 3.4) and

(4) X_U = Expense related to D_U emergency energy purchases (see Section 3.5).

In addition to these outputs discussed in this chapter, the SOM of Chapter 4 requires various data related to the nuclear potential and each reactor's possible contributions to it. Discussion of these more subtle outputs is postponed until Section 4.2.

3.2 Determining Startup and Loading Order

The Booth-Baleriaux model to be discussed in Section 3.3 is an objective, mathematical algorithm for calculating energy production given a startup and loading order for the capacity increments. Thus, it is in determining this input loading order (sometimes referred to as the "pecking order"), that the more subjective aspects of utility operating practices and constraints must be considered.

The goal is to determine for each period the startup and loading order that meets all operating constraints at minimum total cost. Ironically, startup-shutdown cost itself is not used in the multi-year model for determining the startup order. For one thing, total startup-shutdown cost is rarely as large as 1% of production fuel cost. In addition, accurate startup-shutdown cost prediction requires a daily or hourly model, as in the work of Joy (37, 38). Though this cost component is not considered in determining the loading order prior to the Booth-Baleriaux simulation, Section 3.4 will discuss how X_S is estimated from the model's output.

To determine the unit-by-unit startup order, minimum average fuel costs are determined by inspection of average heat rate data as in Figure 2.5.

$$\bar{e}_r^{min} = \phi_r \, \bar{h}_r^{min} \tag{3.3}$$

A tentative startup order can then be determined by plotting this data in ascending order of cost. Figure 3.1a presents such a startup order for the on-line units of a hypothetical utility system. This order is the most attractive economically (ignoring incremental effects due

Figure 3.1a

Optimum Economic Startup Order

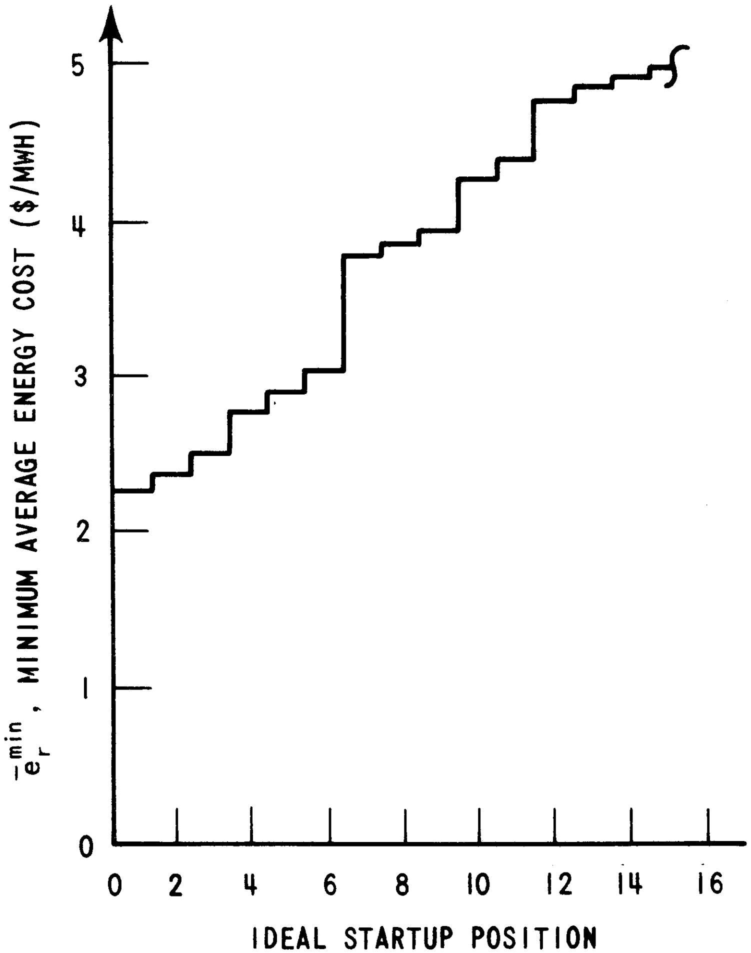
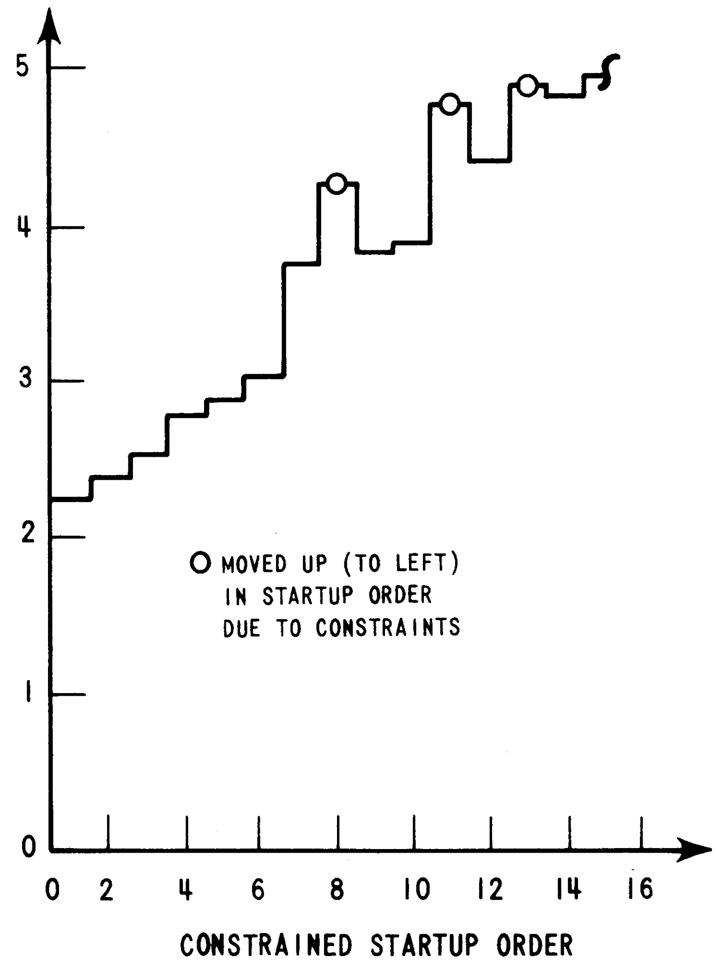


Figure 3.1b

Optimum Constrained Startup Order



6253-21

to startup-shutdown cost itself).

However, various operating constraints alter the order. For instance, engineering and reliability constraints may dictate that some units are must-run units (see Section 2.4.3). Additional constraints related to the distribution of units, loads and transmission lines among geographical regions or zones may impose zone-loading requirements. Such constraints require a unit to be started earlier in the order so that utilization of the entire transmission system will remain approximately balanced. This not only reduces the probability of a transmission system outage, but also reduces the consequences should one occur. Figure 3.1b presents the final constrained startup order for the data of Figure 3.1a.

The first increments in the complete system loading order are, by definition, the minimum power levels of each must-run unit. As Figure 2.12 and Equation (2.33) indicate, the exact order below the minimum system load is arbitrary since all are base-loaded. In fact, the generally low level of nuclear fuel costs coupled with the must-run constraint for such large units is sufficient to permit the assumption that all nuclear minimums are base-loaded. Furthermore, the incremental cost argument of Section 2.4.3 justifies placing all of the upper nuclear increments, as a group, next in the order just to the right of the must-run increments. As it turns out (see Section

3.3.3), the exact intranuclear loading order for these upper increments is arbitrary, relieving the necessity of having precise nuclear incremental costs during the SIM's calculations.

Having assigned all nuclear capacity and all must-run fossil minimums, the incremental cost arguments of Sections 2.2.1 and 2.4.1 determine a complete, but tentative, startup and loading order. For determining the startup of remaining units, $\bar{e}_r^{min}$ represents unit r's opportunity generating cost if the unit is on-line at the power level that minimizes $\bar{h}$. However, costing of the unit's first increment is performed using the $\bar{e}_{r1}$ out-of-pocket average cost [per Equations (2.18) and (2.21)].

$$X_{r1} = \bar{e}_{r1} E_{r1} \tag{3.4}$$

The unit's upper increments are characterized by the usual λ_{ri}.

Given the constrained startup order, the completed loading order is the economic optimum. However, actual operating practices may violate this ordering in the same way that the economic startup order was violated. For instance, a daily practice may involve bringing units up to minimum load a few hours early so that any minor startup problems can be alleviated and their capacity will be available when actually required. Another operating constraint is the requirement for several hundred megawatts of spinning reserve in case a large unit suddenly trips off the line. Spinning reserve represents the readily available (on the order of minutes), uncommitted capacity of turbines already spinning, but generating at less than full capacity. Such a requirement necessitates earlier (uneconomical) startup of some units so that cheaper increments, previously comprising the spinning reserve, may be loaded (see Figure 3.2).

Because of their fast-start capability, peaking units are considered as a separate "stand-by reserve". As such, they need be committed only when their high fuel cost is economically justified.

With such operating constraints properly factored in, the startup and loading order for the period is complete. The evaluation of the period's resulting energy and cost components is the subject of the rest of this chapter.

3.3 Scheduling and Costing Production

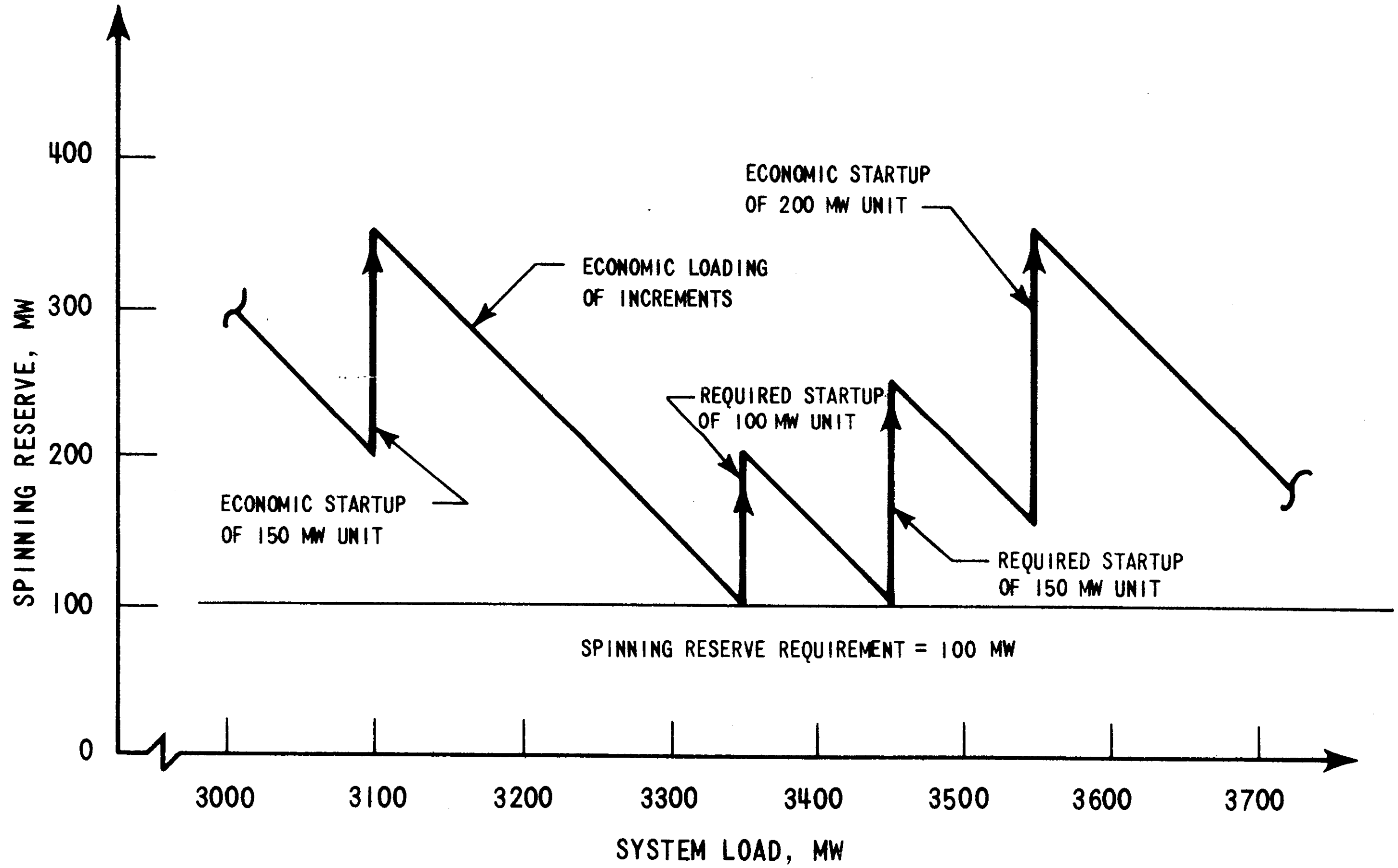
3.3.1 Basics of Booth-Baleriaux Probabilistic Utility Simulation Model

3.3.1.1 Background

The Booth-Baleriaux probabilistic utility simulation model is a recent adaptation of previous deterministic utility models with new emphasis on the field of applied probability theory. Though the original 1967 paper on the

Figure 3.2

Example of Variation of Spinning Reserve as Units are Startedup and Loaded



subject is a product of Baleriaux, et al. (10) of Belgium, Booth (17-19) of Australia deserves much of the credit for introducing and promoting the model in the United States.

Previous papers reporting on the Booth-Baleriaux model, including the work of Joy and Jenkins (39), have closely followed the development in the original paper. With due respect to these ground-breaking efforts, the following presentation leads to computational savings in terms of time and storage, and also follows a more direct line of reasoning.

The Booth-Baleriaux probabilistic utility model is based on the concept of equivalent load which embodies not only direct customer demands on a particular unit, but also the indirect demands left unsatisfied by previously loaded units when they are on forced-outages.

The equivalent load P_e may be defined as

$$P_e \equiv P_D + P_O \tag{3.5}$$

where

P_D = actual direct customer load demand, MW

P_O = system capacity on forced-outage that would be generating energy otherwise, MW

Capacity that is on forced-outage during what would otherwise have been reserve (i.e., economy) shutdown hours anyway is not counted since the outage does not affect system generating operations.

In a probabilistic sense, P_D is a random variable with a complementary cumulative distribution given by $F_D(P_D)$, the normalized customer load-duration curve. Since forced-outages are random, P_O is also a random variable characterized by the performance probabilities of each unit. Thus, P_e is also a random variable and the computation of its required complementary cumulative distribution function $F_e(P_e)$ involves the convolution of the distributions of P_D and P_O (26). Hence, $F_e(P_e)$ is the load-duration curve for the equivalent load P_e. The heuristic presentation here is limited to the common two-state model of forced-outages:

State 1: With probability p, the unit will perform at any output up to its rated capacity when called upon and

State 2: With probability q, the unit will not perform at all when called upon.

Thus,

$$p + q = 1 \tag{3.6}$$

A rigorous treatment of the more general case allowing for forced deratings (i.e., inability of the unit to perform at rated capacity, though partial output is possible), is presented in Appendix A.

To keep the numerical effort to a minimum while illustrating the principle, the detailed numerics of the Booth-Baleriaux convolution algorithm are first presented by way of a simple two-unit, single-increment example. ("Single increment" refers to the fact that each unit is treated as a single block of capacity). This model, the original contribution of Baleriaux, et al. (10), is the so-called "one-piece" Booth-Baleriaux model. Building on this, a more general "multi-piece" procedure (39) permitting the multiple increments to be scheduled separately is presented in Section 3.3.2.

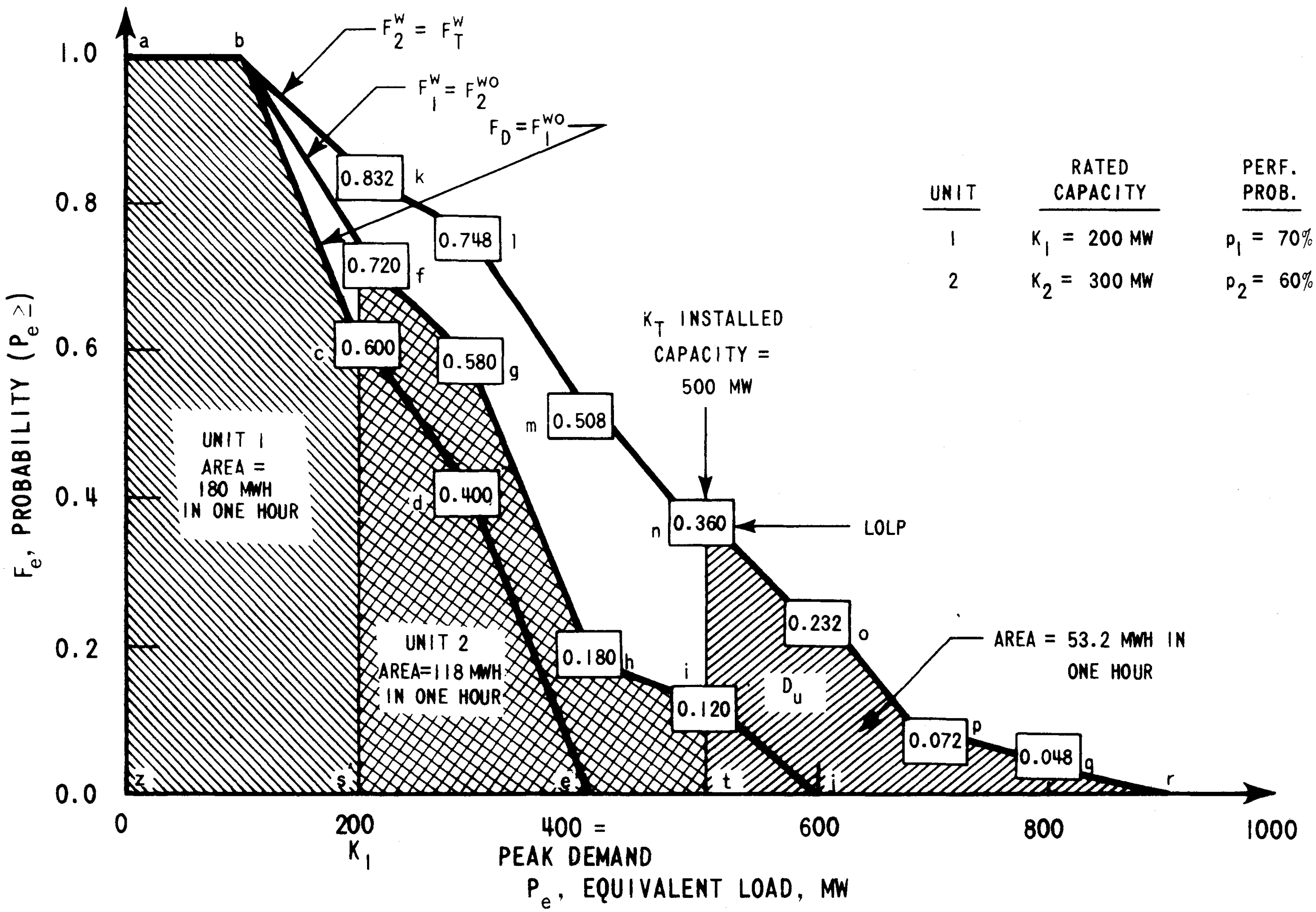
3.3.1.2 Heuristic Derivation of Booth-Baleriaux Convolution using Two Unit, Single Increment Example

In order to derive the basic Booth-Baleriaux convolution equation, consider a 500 MW system consisting of Unit 1 (200 MW with p_1 = 70%) and Unit 2 (300 MW with p_2 = 60%). As displayed in Figure 3.3, the system is attempting to satisfy the indicated F_D customer load-duration curve abcde with a peak demand of 400 MW. For convenience, let the time period duration T' = 1 hour. Hence, total demand D_T = 250 MWH (area zabcdez).

Since Unit 1 is the first to come on line, the first step in the simulation is to compute its loading. Since there are no units to its left, the equivalent load as seen by Unit 1 is merely the direct customer demand F_D. However, the unit performs only 70% of the time. Thus, Unit 1 is

Figure 3.3

Complete Two Unit Booth-Baleriaux Example



6253-23

only able to generate 70% of the energy demanded from it (area zabcsz),

$$E_1 = T' p_1 \int_0^{K_1} F_D(P_e)\, dP_e \qquad (3.7)$$

or

$$E_1 = 1 \text{ hour} \times 0.7 \times 180 \frac{\text{MWH}}{\text{hour}} = 126 \text{ MWH} \qquad (3.8)$$

Unit 1 has been loaded according to F_D, the equivalent load curve F "without" an adjustment for Unit 1's outages ($\equiv F_1^{wo}$). Unit 2, on the other hand, sees not only direct customer demand F_D, but also indirect demand unsatisfied by Unit 1 while it was down due to a forced-outage. Thus, before loading Unit 2, Unit 1's outages must be "convolved" into F_1^{wo} ($\equiv F_D$) to yield F_1^{w} (i.e., "with" an allowance for Unit 1's forced-outages).

To do this, it is necessary to consider the two states:

(1) Unit 1 performs, a state with the probability p_1 (= 0.7), and

(2) Unit 1 fails to perform, a state with the probability $q_1 = 1-p_1$ (= 0.3).

Thus, a particular equivalent load, for example $P_e \geq 300$ MW can be arrived at in only two possible independent ways. The probability that the equivalent load $\geq$ 300 MW is the sum of the probabilities of each of the individual ways. When unit 1 performs, the probability that the equivalent load $P_e \geq 300$ MW is the product of the probability that unit 1 will perform (p_1) and the probability that the equivalent load will exceed 300 MW without an allowance for outage of unit 1 $[F_1^{wo}(P_e)]$, that is $p_1F_1^{wo}(P_e)$.

When unit 1 fails to perform, its forced outage of $K_1 = 200$ MW contributes 200 MW to the equivalent load of 300 MW. Hence, the other probability that the equivalent load $P_e \geq 300$ MW (when Unit 1 fails to perform), is the product of the probability that Unit 1 fails (q_1) and the probability that the equivalent load will exceed $P_e - K_1 = 300 - 200 = 100$ MW without the $K_1 = 200$ MW allowance for the forced-outage of Unit 1 $[F_1^{wo}(P_e-K_1)]$; that is, $q_1F_1^{wo}(P_e-K_1)$.

Hence, the equivalent load curve with allowance for forced-outages of Unit 1, $F_1^{w}(P_e)$, is the sum of the probabilities for states 1 and 2,

$$F_1^{w}(P_e) = p_1F_1^{wo}(P_e) + q_1F_1^{wo}(P_e-K_1) \qquad (3.9)$$

or

$$F_1^{w}(P_e) = 0.7\cdot F_1^{wo}(P_e) + 0.3\cdot F_1^{wo}(P_e-200) \qquad (3.10)$$

For the $P_e = 300$ MW example of Figure 3.3 $F_1^{wo}(300)_{\text{(point d)}} = 0.400$ and $F_1^{wo}(100)_{\text{(point b)}} = 1.00$. Hence

$$F_1^W(300) = 0.7\ \underset{\text{(point d)}}{(0.4)} + 0.3\ \underset{\text{(point b)}}{(1.0)} = \underset{\text{(point g)}}{0.58} \qquad (3.11)$$

Continuing thus for all the points along F_1^W,

$$F_1^W(200) = 0.7 \times \underset{\text{(point c)}}{0.600} + 0.3 \times \underset{\text{(point a)}}{1.0} = \underset{\text{(point f)}}{0.720}$$

$$F_1^W(400) = 0.7 \times \underset{\text{(point e)}}{0.0} + 0.3 \times \underset{\text{(point c)}}{0.600} = \underset{\text{(point h)}}{0.180}$$

(3.12)

$$F_1^W(500) = 0.7 \times \underset{\text{(point t)}}{0.0} + 0.3 \times \underset{\text{(point d)}}{0.400} = \underset{\text{(point i)}}{0.120}$$

$$F_1^W(600) = 0.7 \times \underset{\text{(point j)}}{0.0} + 0.3 \times \underset{\text{(point e)}}{0.0} = \underset{\text{(point j)}}{0.000}$$

In more general terms, any unit r can be convolved into the equivalent load distribution,

$$F_r^W(P_e) = p_r \bullet F_r^{WO}(P_e) + q_r \bullet F_r^{WO}(P_e - K_r) \qquad (3.13)$$

or

$$\begin{bmatrix}\text{Prob.}(P \geq P_e)\\ \text{with outages}\\ \text{of r incl.}\end{bmatrix} = \begin{bmatrix}\text{Prob. r}\\ \text{performs}\end{bmatrix} \bullet \begin{bmatrix}\text{Prob.}(P \geq P_e)\\ \text{w/o outages}\\ \text{of r incl.}\end{bmatrix} + \begin{bmatrix}\text{Prob. r}\\ \text{fails}\end{bmatrix} \bullet \begin{bmatrix}\text{Prob.}(P \geq P_e - K_r)\\ \text{w/o outages of}\\ \text{r included}\end{bmatrix} \qquad (3.14)$$

MW Contribution to Equivalent load:

$$\underbrace{0 + \geq P_e}_{\geq P_e} \qquad \underbrace{K_r + \geq (P_e - K_r)}_{\geq P_e}$$

In deriving Equation (3.13), use was made of the common assumption of statistical independence between the forced-outages of the various units vis-a'-vis each other and the customer demand. Furthermore, Equation (3.13) is valid for all P_e. One limiting case is P_e less than the minimum load where each $F_r^{wo}=1$ as does the resulting $F_r^w(P_e)$. For very large P_e, each $F_r^{wo}= 0$ and, likewise, $F_r^w(P_e) = 0$.

Equation (3.13) is the heart and soul of the Booth-Baleriaux model. All subsequent calculations involving F, whether convolutions or deconvolutions (see Section 3.3.2.1) are merely rearrangements of it.

Returning to the two unit example, Figure 3.3 indicates the resulting F_1^w obtained by applying Equation (3.13) at each multiple of 100 MW. [Equation (3.13) could be applied explicitly at intermediate P_e, but linear interpolation is rigorously correct for this example because the F_D curve consists of straight-line segments.]

Since Unit 2 follows Unit 1 in the loading order, the production of Unit 2 must be determined using an equivalent load curve (F_2^{wo}) that includes not only the direct customer load demands, F_D, but also the forced-outages of units to the left of it in the loading order (i.e., Unit 1). Thus,

$$F_2^{wo}(P_e) = F_1^w(P_e) \tag{3.15}$$

That is, the probability that the equivalent load will exceed a particular value P_e without taking into account forced-outages of Unit 2 equals the same probability taking into account forced-outages of Unit 1.

As with Unit 1, the loading of Unit 2 is determined by multiplying the total demand on the unit (area sfghits) by its performance probability p_2,

$$E_2 = T'p_2 \int_{200}^{500} F_2^{wo}(P_e)dP_e \tag{3.16}$$

$$E_2 = 1 \text{ hour} \times .60 \times 118 \frac{\text{MWH}}{\text{H}} = 70.8 \text{ MWH} \tag{3.17}$$

Rewriting Equation (3.16) in general notation for any Unit r,

$$E_r = T'p_r \int_{P_r^\circ}^{P_r^\circ + K_r} F_r^{wo}(P_e)dP_e \tag{3.18}$$

where P_r° = Loading point for unit r, MW

Now that Unit 2's production has been accounted for, its outages must be convolved into F_2^{wo}. By applying Equation (3.13),

$$F_2^{w}(P_e) = p_2 \cdot F_2^{wo}(P_e) + q_2 \cdot F_2^{wo}(P_e - K_2) \tag{3.19}$$

For example (see Figure 3.3), since $K_2 = 300$ MW and $P_2 = 60\%$,

$$F_2^W(P_e) = 0.6 \times F_2^{WO}(P_e) + 0.4 \times F_2^{WO}(P_e\text{-}300) \qquad (3.20)$$

In particular, at $P_e = 500$ MW (point n)

$$\underset{\text{}}{F_2^W(500)} = 0.6 \times \underset{\text{(point i)}}{.120} + 0.4 \times \underset{\text{(point f)}}{0.720} = \underset{\text{(point n)}}{0.360} \qquad (3.21)$$

Continuing thus,

$$F_2^W(600) = 0.6 \times \underset{\text{(point j)}}{0.0} + 0.4 \times \underset{\text{(point g)}}{0.580} = \underset{\text{(point o)}}{0.232}$$

$$F_2^W(700) = 0.6 \times 0.0 + 0.4 \times \underset{\text{(point h)}}{0.180} = \underset{\text{(point p)}}{0.072}$$

$$(3.22)$$

$$F_2^W(800) = 0.6 \times 0.0 + 0.4 \times \underset{\text{(point i)}}{0.120} = \underset{\text{(point q)}}{0.048}$$

$$F_2^W(900) = 0.6 \times 0.0 + 0.4 \times \underset{\text{(point j)}}{0.0} = \underset{\text{(point r)}}{0.000}$$

Since both of the units on the system have been convolved in via Equation (3.13), the resulting F_2^W equivalent load distribution (see Figure 3.3) includes the entire system, F_T^W.

Hence, the remaining D_U unserved energy (i.e., unserved by the K_T MW of the system's own resources or area tnopqrt) is equal to

$$D_U = T' \int_{K_T}^{\infty} F_T^W(P_e)\,dP_e = 53.2 \text{ MWH} \qquad (3.23)$$

This energy represents the amount of emergency support required from neighboring utilities.

The second measure of system reliability is the LOLP, loss-of-load probability (i.e., percent of time emergency support is required: $P_e \geq K_T$). Hence,

$$LOLP = F_T^W(P_e = 500 \text{ MW}) = 0.360 \qquad (3.24)$$

(point n)

Note that total system production plus emergency purchases have met total customer demand:

$$D_T = E_1 + E_2 + D_U \qquad (3.25)$$

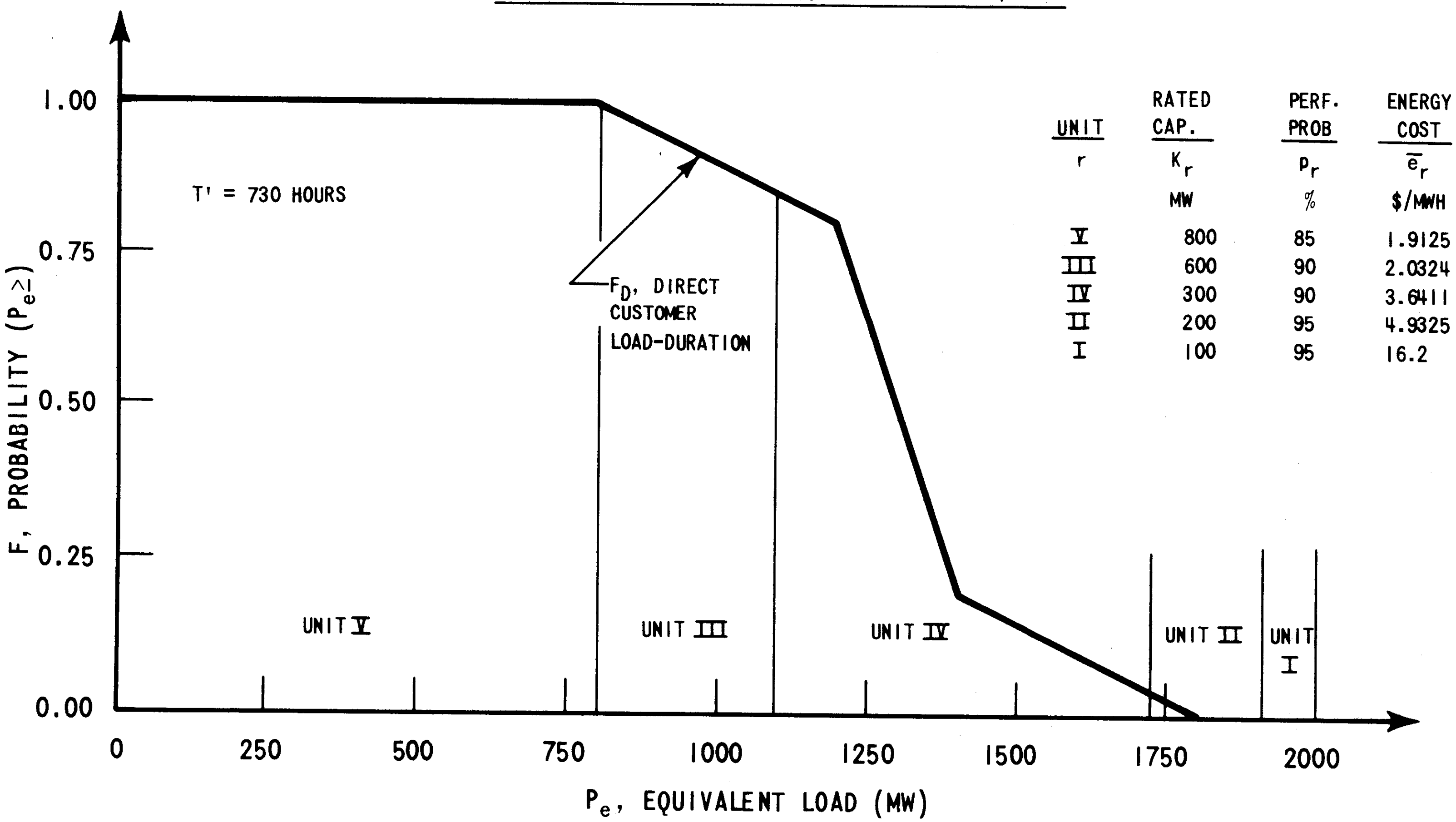
$$250 \text{ MWH} = 126 + 70.8 + 53.2 \text{ MWH} \qquad (3.26)$$

3.3.1.3 Single Increment Example for Reference Utility System

Returning to the original Reference Utility System of Section 2.1.2.3, the customer loads of Figure 2.9 are repeated in Figure 3.4. As for the five generating units, assume the loading order, unit characteristics and average (i.e., equivalent single increment, see Table C.13 in

Figure 3.4

Loading Order and Energy Cost for Example 4



6253-24

Appendix C) costs also indicated in Figure 3.4. This then represents Example 4 on the Reference System.

Applying the load-then-convolve sequence of Section 3.3.1.2, the unit loadings E_r are simulated in order. Table 3.1 presents all of the resulting probability distributions.

When the last unit (Unit I) has been convolved in, the resulting F_I^W distribution includes the entire system F_T^W. Hence,

$$D_U = T' \int_{K_T}^{\infty} F_T^W(P_e)\,dP_e = 30{,}111 \text{ MWH} \qquad (3.\ 27)$$

and

$$\text{LOLP} = F_T^W\ (P_e = K_T) = 15.647\% \qquad (3.28)$$

This completes the Booth-Baleriaux <u>energy</u> calculations for Example 4. Equation (2.21) can then be utilized to determine the cost of each unit's energy production.

$$X_r = \bar{e}_r \cdot E_r \qquad (3.29)$$

Figure 3.5 sketches the complete flow of calculations, including the energy and cost totals (see also Table 3.2).

Table 3.1

Summary of Equivalent Load Distributions for Example 4 with Indication of Segments Used for Loading Each Unit

Unit Loaded, (r)	V	III	IV	II	I	Neigh.Util.
Rated Cap., (K_r)	800MW	300	600	200	100	∞
Perf. Prob., (p_r)	0.85	0.90	0.90	0.95	0.95	1.00
P_e (MW)	$F_D = F_V^{WO}$	$F_V^W = F_{III}^{WO}$	$F_{III}^W = F_{IV}^{WO}$	$F_{IV}^W = F_{II}^{WO}$	$F_{II}^W = F_I^{WO}$	$F_I^W = F_T^W$
0	1.00	1.0000	1.00000	1.00000	1.00000	1.00000
⋮	⋮ V	⋮	⋮	⋮	⋮	⋮
800	1.00	1.0000	1.00000	1.00000	1.00000	1.00000
900	.95	.9575 III	.96175	.96558	.96730	.96893
1000	.90	.9150	.92350	.93115	.93459	.93623
1100	.85	.8725	.88525	.89672	.90017	.90189
1200	.80	.8300	.84275	.85848	.86211	.86401
1300	.50	.5750	.60900	.64810	.66053	.67061
1400	.20	.3200	.37525 IV	.43772	.45876	.46885
1500	.15	.2775	.33275	.39565	.40827	.41080
1600	.10	.2350	.26900	.33445	.33961	.34305
1700	.05	.1850	.19850	.26718	.27360	.27690
1800	.00	.1350	.14925	.21860 II	.22439	.22685

Table 3.1--Continued

P_e (MW)	$F_D = F_V^{WO}$	$F_V^W = F_{III}^{WO}$	$F_{III}^W = F_{IV}^{WO}$	$F_{IV}^W = F_{II}^{WO}$	$F_{II}^W = F_I^{WO}$	$F_I^W = F_T^W$	
1900		.1275*	.13825	.18532	.18942	.19117	
2000		.1200	.12650	.15138	.15474	.15647	LOLP
2100		.0750	.08100	.10618	.11013	.11236	
2200		.0300*	.03975*	.06268	.06711	.06926	
2300		.0225	.03225	.04888	.05174	.05251	D_U
2400		.0150	.02100	.03382	.03527	.03609	
2500		.0075	.00975	.02260	.02391	.02448	
2600		.0000	.00225	.01468	.01563	.01605	
2700			.00150	.00945	.01011	.01038	
2800			.00075	.00465	.00515	.00540	

*Example: 0.03975 = (0.9) (0.0300) + (0.1) (0.1275)

6253-25

Figure 3.5

Calculational Steps for Example 4

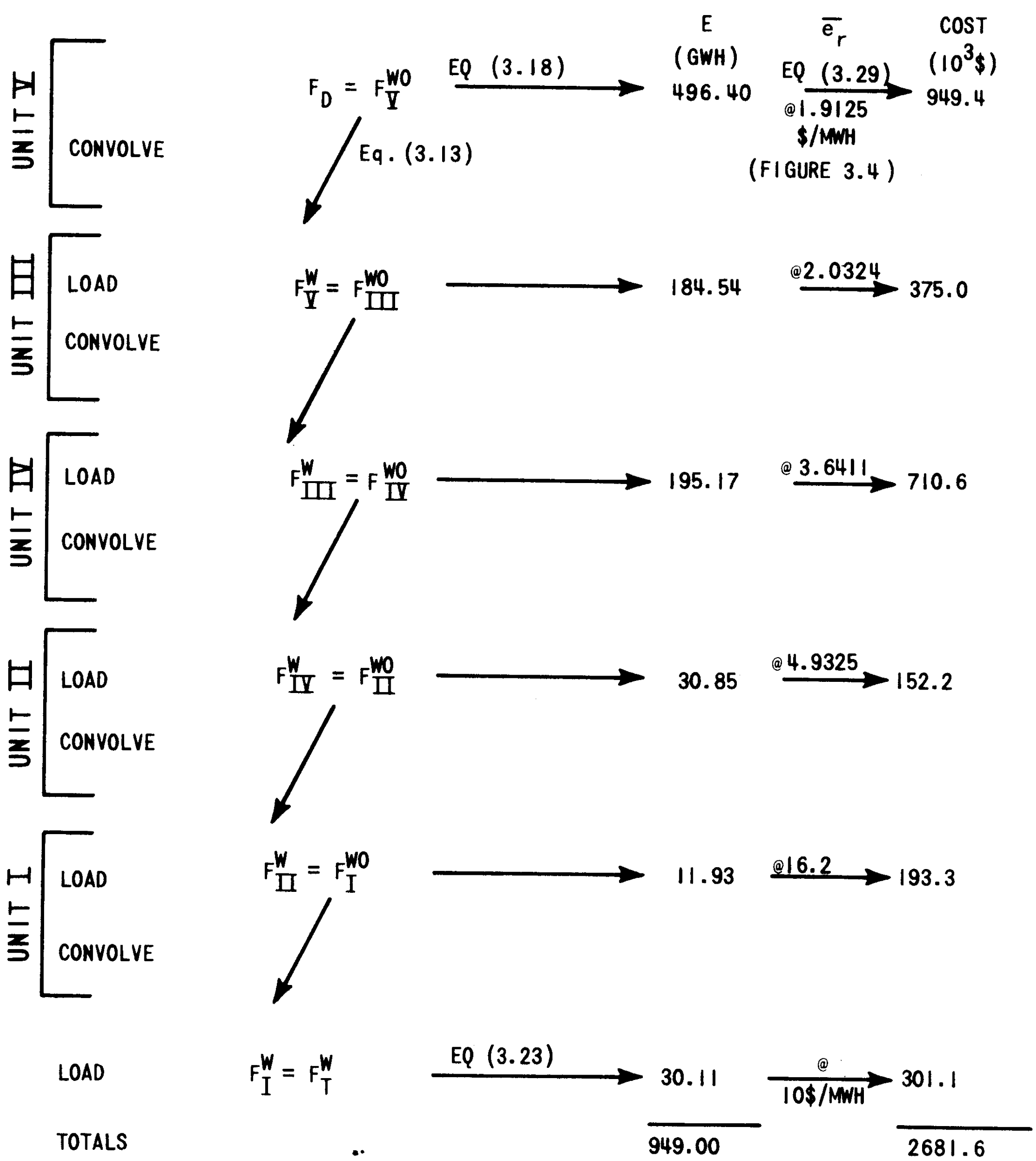


TABLE 3.2

Example 4 on Reference Utility System:

"Single Increment Booth-Baleriaux Model"

(See Appendix C for further details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 \$)
I	1	5	11.93	193.3
II	1, 2	4	30.85	152.2
III	1, 2	2	184.54	375.0
IV	1, 2	3	195.17	710.6
V	1, 2	1	496.40	949.4
	Utility Production		918.89	2380.5
	Emergency Purchases (10 $/MWH)		30.11	301.1
	Total		949.00	2681.6

Loss-of-load Probability, LOLP = 15.6%

3.3.1.4 Single Increment Algorithm

From Figure 3.5, the load-convolve sequence of the single increment Booth-Baleriaux algorithm can be stated as follows:

Step 1: From the specified loading order, label the first unit as unit r. Re-label F_D, the normalized customer load-duration curve so that it becomes the "current" F.

Step 2: Re-label the current F so that it becomes F_r^{wo}.

Step 3: Load unit r by calculating its expected production,

$$E_r = T' p_r \int_{P_r^\circ}^{P_r^\circ + K_r} F_r^{wo}(P_e) dP_e \qquad (3.30)$$

where P_r° = equivalent load level when unit r is at zero power level.

Step 4: Convolve the unit's outages into F_r^{wo} to account for the production unit r was unable to satisfy,

$$F_r^w(P_e) = p_r \cdot F_r^{wo}(P_e) + q_r \cdot F_r^{wo}(P_e - K_r) \qquad (3.31)$$

Step 5: If there are no more units in the loading order, go to Step 6. Otherwise, label the next unit in the specified loading order as unit r. Return to Step 2 and continue.

Step 6: Since there are no more units to be loaded, the current F is for the total system. Label it F_T. Then,

$$LOLP = F_T(P_e = K'_T) \quad (3.32)$$

and

$$D_u = T' \int_{K'_T}^{\infty} F_T(P_e)dP_e \quad (3.33)$$

This completes the Booth-Baleriaux algorithm for one-piece units. Production costing of the energy,

$$X_r = \bar{e}_r E_r \quad (3.34)$$

can be performed either on-line as a second part of Step 3 or off-line after all of the energies have been assigned.

3.3.1.5 Important Numerical Properties

Seven important numerical properties of the Booth-Baleriaux model are worthy of note. The first three relate directly to the computational effort involved while the latter four deal with the more philosophical aspects of the results.

First, by invoking Equation (3.6), the time involved in the convolution of Equation (3.31) can be reduced by almost one-half by rearranging to:

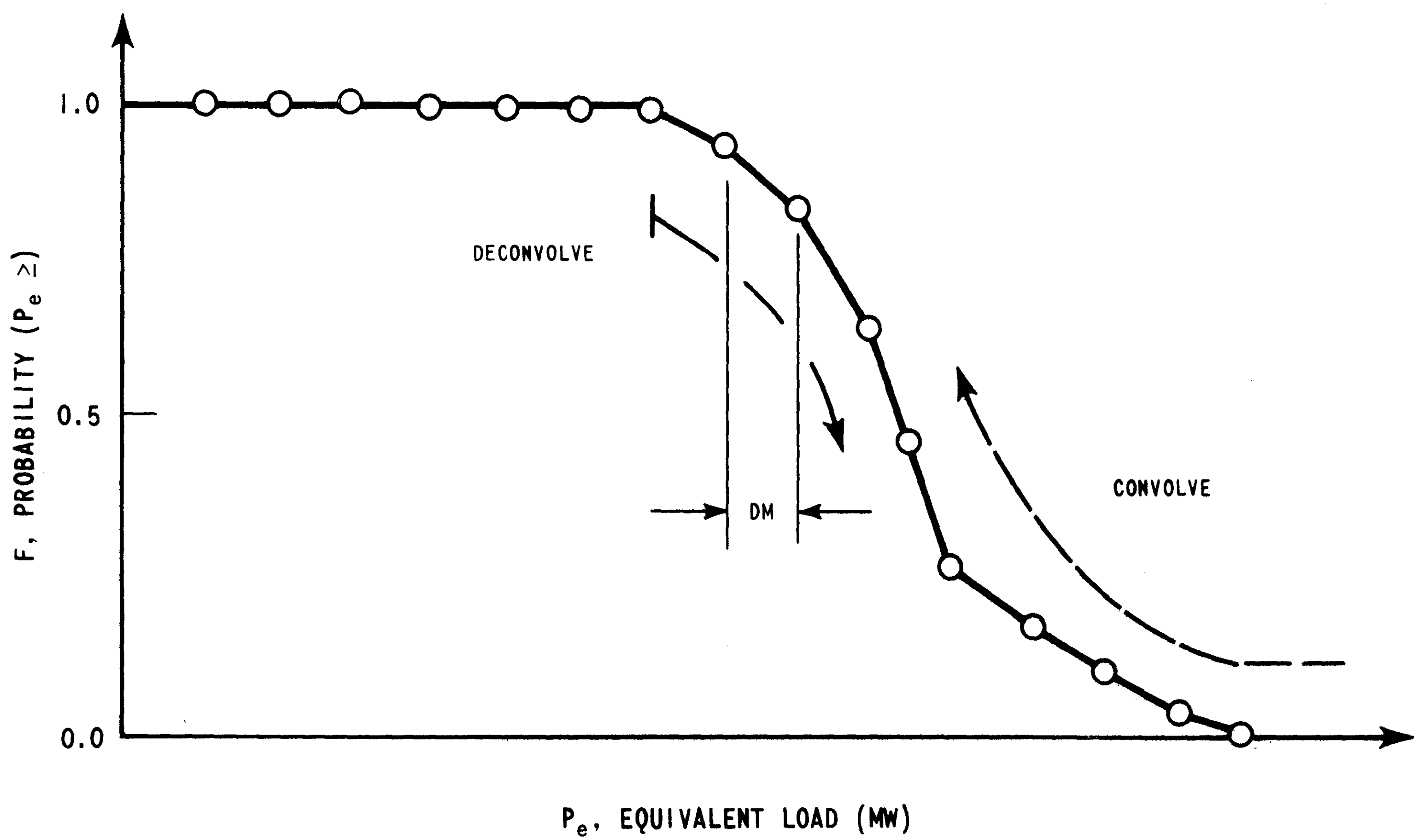
$$F_r^W(P_e) = F_r^{WO}(P_e) + q_r \ [F_r^{WO}(P_e-K_r)-F_r^{WO}(P_e)] \qquad (3.35)$$

Two time-consuming multiplications can be reduced to one. [As a sidelight, F_r^W at P_e never decreases in magnitude as loading proceeds since the second term in Equation (3.35) can never by negative.] Secondly, though Example 4 involved six different F's, only one was required at any one time and, furthermore, none was ever required a second time; the result being that only one array of storage need ever be allocated to F. The array F is stored in the computer as a one dimensional array of equally-spaced points DM MW apart (see Figure 3.6). Thus the 12th array location has stored in it $F(P_e = 12*DM)$. Linear interpolation is assumed between points.

Since the convolution of Equation (3.31) involves only the point of interest (at P_e) and points to its left (specifically, at $P_e - K_r$), it is convenient to begin the convolution of each unit r at the extreme right-hand side of Figure 3.6. Proceeding toward the left, each array location has its current quantity $[F_r^{WO}(P_e)]$ increased by $q_r* \ [F_r^{WO}(P_e-K_r) - F_r^{WO}(P_e)]$ per Equation (3.35). In this manner, F_r^{WO} is convoluted to yield F_r^W. By being identically located, F_r^W automatically becomes F^{WO} for the next unit. The result is that the single F array is kept "current" as the scheduling algorithm

Figure 3.6

Computer Representation of Equivalent Load Curve



proceeds from unit to unit.

The third and final point concerning computational details involves deconvolution. Even if a previous F were needed again, it could be easily restored by reversing Equation (3.31). Such a deconvolving, or stripping out, of the outages of a previously included unit r can thus be achieved by,

$$F_r^{wo}(P_e) = \frac{1}{p_r}\left[F_r^{w}(P_e) - q_r \cdot F_r^{wo}(P_e - K_r)\right] \qquad (3.36)$$

For deconvolution, the direction of calculation would also be reversed, proceeding from left to right of Figure 3.6 so that $F(P_e)$ for P_e to the left of the point of interest would already be F^{wo} as required by Equation (3.36).

The first important philosophical result has already been seen in Section 3.3.1.2: The production of previous increments is unaffected by changes in the loading order of subsequent units. The order of the computations bears this out immediately.

Secondly, with regard to any currently stored F array, it is a function of the units convolved in, but not a function of the <u>order</u> in which they were added. Consider an initial customer demand F_D and the simple two unit utility system (Unit 1 and Unit 2). The task is to prove that $F_T(P_e)$ is identical whether the loading order is (1) Unit 1, then Unit 2 (see Section 3.3.1.2) or (2) Unit 2, then

Unit 1.

Equation (2.31) holds for both cases,

$$F_1^W(P_e) = p_1F_1^{WO}(P_e) + q_1F_1^{WO}(P_e - K_1) \tag{3.37}$$

and

$$F_2^W(P_e) = p_2F_2^{WO}(P_e) + q_2F_2^{WO}(P_e - K_2) \tag{3.38}$$

For Case (1) (Unit 1, then Unit 2),

$$F_1^{WO} = F_D \tag{3.39}$$

$$F_2^{WO} = F_1^W \tag{3.40}$$

and

$$F_T = F_2^W \tag{3.41}$$

Thus,

$$F_T(P_e) = p_2 \; [p_1F_D(P_e) + q_1F_D(P_e - K_1)]$$

$$+ q_2 \; [p_1F_D(P_e - K_2) + q_1F_D(P_e - K_2 - K_1)] \tag{3.42}$$

or finally,

$$F_T(P_e) = p_1p_2F_D(P_e) + q_1p_2F_D(P_e - K_1)$$

$$+ p_1q_2F_D(P_e - K_2) + q_1q_2F_D(P_e - K_1 - K_2) \tag{3.43}$$

For Case (2) (Unit 2, then Unit 1),

$$F_2^{wo} = F_D \tag{3.44}$$

$$F_1^{wo} = F_2^{w} \tag{3.45}$$

and

$$F_T = F_1^{w} \tag{3.46}$$

Thus,

$$\begin{aligned} F_T(P_e) &= p_1 \; [p_2 F_D(P_e) + q_2 F_D(P_e - K_2)] \\ &\quad + q_1 \; [p_2 F_D(P_e - K_1) + q_2 F_D(P_e - K_1 - K_2)] \end{aligned} \tag{3.47}$$

or, rearranging,

$$\begin{aligned} F_T(P_e) &= p_1 p_2 F_D(P_e) + q_1 p_2 F_D(P_e - K_1) \\ &\quad + p_1 q_2 F_D(P_e - K_2) + q_1 q_2 F_D(P_e - K_1 - K_2) \end{aligned} \tag{3.48}$$

Since F_T in Case (1) [Equation (3.43)] is term by term identical with F_T in Case (2) [Equation (3.48)], the proof for the two unit system is complete. The generalization to more units is straightforward, though cumbersome and is not presented formally. In conclusion, each F is a function of the units whose outages have already been included but not a function of their order of inclusion.

The third philosophical point follows immediately from the above. Since F is independent of the order of inclusion, a unit's loading, determined using the F, is also independent of the ordering. However, as with F, it does depend on which units are included.

The fourth and final philosophical point also follows from the second. When all units have been convolved in, the resulting F_T is independent of the loading order. Thus, the LOLP and D_U are not functions of the startup and loading order, but only of the original customer demand and the aggregate system equipment not on scheduled maintenance.

3.3.2 Modifications for Multiple Increments

3.3.2.1 Algorithm Derived

The original single increment Booth-Baleriaux model was a tremendous leap forward in utility system simulation. As Example 3 in Section 2.2.1 pointed out, not only was the production of peaking equipment more accurately predicted, but the model was also better able to estimate the LOLP and unserved energy D_U by the same technique. One large stumbling block remained--how to accurately represent the interweaving of the multiple increments of the various units. Units are not scheduled as single blocks of capacity, not only because of economics, but also because of spinning reserve requirements.

To handle this more general case rigorously, only a slight modification of the single increment algorithm is required. The load-convolve pattern is replaced with a deconvolve-load-convolve sequence.

To derive the algorithm, after loading the first increments of several units, assume (1) the next increment in the loading order is ΔK_{ri} (the i th increment of unit r), (2) that i > 1 and (3) that the current F, ($F^{w}_{r,i-1}$) already includes unit r's increments up to $K_{r,i-1}$. If ΔK_{ri} was mistakenly loaded using $F^{w}_{r,i-1}$ itself, the i th increment would, in essence, be meeting demands due to (1) customers, (2) the forced-outages of increments of other units already loaded and (3) the forced-outages of its own lower (i-1) increments. However, the latter is an impossibility. If the lower increments are down on forced-outage, so is ΔK_{ri}. (The converse is not necessarily true. See Appendix A.)

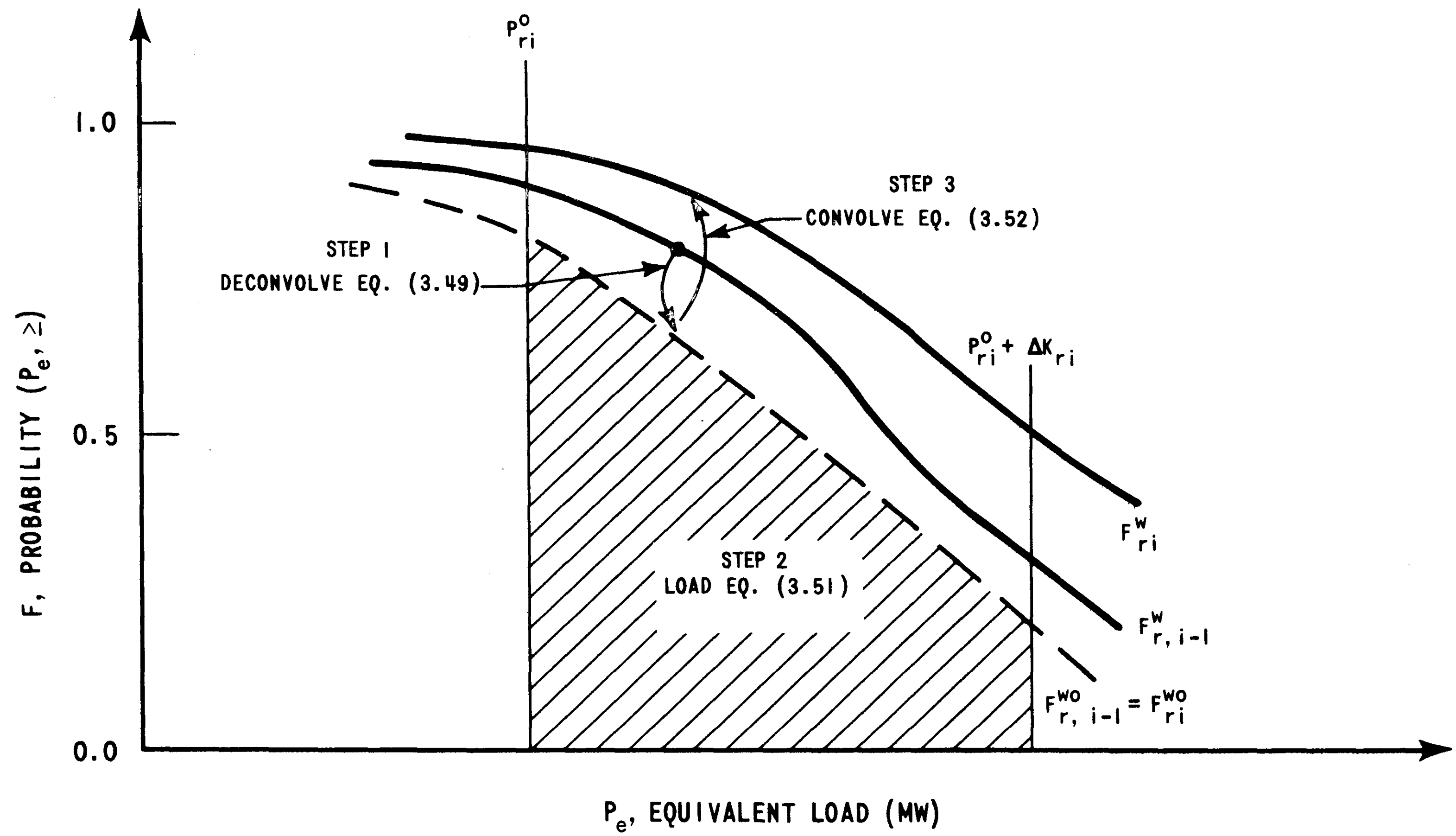
Thus, to load ΔK_{ri} properly (see Figure 3.7), the previously convolved forced-outages of unit r ($K_{r,i-1}$ MW at p_r percent) must be stripped out of F to yield $F^{wo}_{r,i-1}$.

Equation (3.36) does just that,

$$F^{wo}_{r,i-1}(P_e) = \frac{1}{p_r}\left[F^{w}_{r,i-1}(P_e) - q_r F^{wo}_{r,i-1}(P_e - K_{r,i-1})\right] \tag{3.49}$$

Figure 3.7

Deconvolve-Load-Convolve Scheme for Multiple Increments



6253-27

After deconvolution,

$$F^{wo}_{r,i-1} = F^{wo}_{ri} \qquad (3.50)$$

and

$$E_{ri} = T' p_r \int_{P^\circ_{ri}}^{P^\circ_{ri} + \Delta K_{ri}} F^{wo}_{ri}(P_e)dP_e \qquad (3.51)$$

Once the i th increment itself has been loaded, the outages of all the i increments can be convolved into F^{wo}_{ri} at one time,

$$F^{w}_{ri}(P_e) = p_r \cdot F^{wo}_{ri}(P_e) + q_r \cdot F^{wo}_{ri}(P_e - K_{ri}) \qquad (3.52)$$

The resulting deconvolve-load-convolve sequence of Figure 3.7 can be applied successively to each increment in the loading order.

Using the indicated multiple increment loading order (Units III-V must run; 80 MW spinning reserve), Table 3.3 presents the results for this Example 5 on the Reference Utility System. Table 3.4 presents a summary comparison of Examples 1 through 5. The D_T, D_U and LOLP are reassuringly equal for all three probabilistic examples. Furthermore, the multiple increment Example 5 does save $123,000 in production costs over the less economical (early startup

TABLE 3.3

Example 5 on Reference Utility System:

"Multiple Increment Booth-Baleriaux Model (V-2, then III-2)"

(Among Nuclear Upper Increments V-2, then III-2)
(See Appendix C for further details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 $)
I	1	9	11.93	193.3
II	1	6	36.71	201.9
	2	8	14.01	59.5
III	1	2	65.70	149.8
(Nuclear)	2	5	103.90	197.4
IV	1	3	131.40	515.1
	2	7	70.85	235.2
V	1	1	186.15	418.8
(Nuclear)	2	4	298.24	510.0
	Utility Production		918.89	2481.0
	Emergency Purchases (10 $/MWH)		30.11	301.1
	Total		949.00	2782.1

Loss-of-load Probability, LOLP = 15.6%

Table 3.4

Comparison of Examples 1 to 5 on Reference Utility System

Example	Remarks	D_T (GWH)	D_U (GWH)	System Production Fuel Cost (10^6 \$)	LOLP (%)	Reference Table
1	Deterministic (No Forced Outages)	949	0.00	2.443	0.00	2.3
2	Deterministic (with Reduced Capacities)	949	0.11	2.514	1.25	2.4
3	Probabilistic, Multiple Increment; Early Startup of II	949	30.11	2.604	15.65	2.5
4	Probabilistic, Single Increment; No Must-Run, No Spin Res.	949	30.11	2.380[1]	15.65	3.2
5	Probabilistic, Multiple Increment; Operating Constraints App'd. to Econ. order	949	30.11	2.481[2]	15.65	3.3

[1] Lower limit if all operating constraints are violated.

[2] Lower limit if all operating constraints are satisfied.

of Unit II) but practical (spinning reserve satisfied) multiple increment Example 3. The low cost of Example 4 is misleading because the must-run status of Unit IV and the system spinning reserve requirement were ignored, rendering the single increment loading order infeasible (i.e., the system operating constraints were violated).

Before formally stating the steps of the more general multiple increment Booth-Baleriaux algorithm in the next Section, two important points need to be made to justify that generality. First, the method is valid even if i = 1. For then,

$$K_{r,i-1} = K_{r,0} \equiv 0 \qquad (3.53)$$

and the deconvolution of Equation (3.49) reduces to

$$F^{wo}_{r0}(P_e) = \frac{1}{p_r}\left[F^{w}_{r0}(P_e) - q_r F^{wo}_{r0}(P_e - 0)\right] \qquad (3.54)$$

Utilizing Equation (3.6), $F^{wo}_{r0} \equiv F^{w}_{r0}$. That is, if no increments of the unit have been previously loaded, straightforward application of Equation (3.49) correctly deconvolves zero MW.

The second point also involves a limiting condition. Suppose all the multiple increments for a given unit happen to be scheduled adjacent to each other. This case ought to revert to the results of the single increment model. Indeed, each "convolution to left; deconvolution to the right" sequence returns F to the identical F_r^{wo}. In fact, this was the actual scheme used to calculate Example 4 of Section 3.3.1.3 (see Appendix C).

3.3.2.2 Multiple Increment Algorithm

The deconvolve-load-convolve sequence of the more general, multiple increment Booth-Baleriaux algorithm is stated as follows:

Step 1: From the specified loading order, label the first unit increment as unit r, increment i $(i \equiv 1)$. Re-label F_D the normalized customer load-duration curve so that it becomes $F^w_{r,i-1}$.

Step 2: Deconvolve the i-1 previously loaded increments of unit r which cannot create indirect demand on the current increment,

$$F^{wo}_{r,i-1}(P_e) = \frac{1}{p_r}\left[F^w_{r,i-1}(P_e) - q_r F^{wo}_{r,i-1}(P_e - K_{r,i-1})\right] \tag{3.55}$$

and re-label the result F^{wo}_{ri}.

Step 3: Load the unit increment by calculating its expected production,

$$E_{ri} = T' p_r \int_{P^\circ_{ri}}^{P^\circ_{ri}+\Delta K_{ri}} F^{wo}_{ri}(P_e)\,dP_e \tag{3.56}$$

where P°_{ri} = equivalent load level when the unit increment is at zero power level.

Step 4: Convolve the outages of the unit's increments loaded thus far (K_{ri}) into F^{wo}_{ri} to account for the production unit r has thus far been unable to satisfy,

$$F^{w}_{ri}(P_e) = p_r \cdot F^{wo}_{ri}(P_e) + q_r \cdot F^{wo}_{ri}(P_e - K_{ri}) \quad (3.57)$$

Step 5: If there are no more unit increments, go to Step 6. Otherwise, label the next unit increment in the specified loading order as unit r, increment i. Re-label the current F so that it becomes $F^{w}_{r,i-1}$. Return to Step 2 and continue.

Step 6: Since there are no more increments to be loaded, the current F is for the total system. Label it F_T. Then,

$$LOLP = F_T(P_e = K'_T) \quad (3.58)$$

and

$$D_U = T' \int_{K'_T}^{\infty} F_T(P_e)\,dP_e \quad (3.59)$$

This completes the Booth-Baleriaux multiple increment algorithm. Comparing it with the single increment algorithm of Section 3.3.1.4, only Step 2 is significantly different. Instead of immediately re-labeling the current F to F_r^{wo}, a deconvolution must first be performed to ensure that no outages of unit r are included.

As before, production costing of the energy increment,

$$X_{r1} = \bar{e}_{r1}\, E_{r1}, \text{or}, X_{ri} = \lambda_{ri}\, E_{ri} \text{ for } i>1 \qquad (3.60)$$

can be performed either on-line as a second part of Step 3 or off-line after all of the energies have been assigned.

3.3.3 Constancy of Nuclear Potential

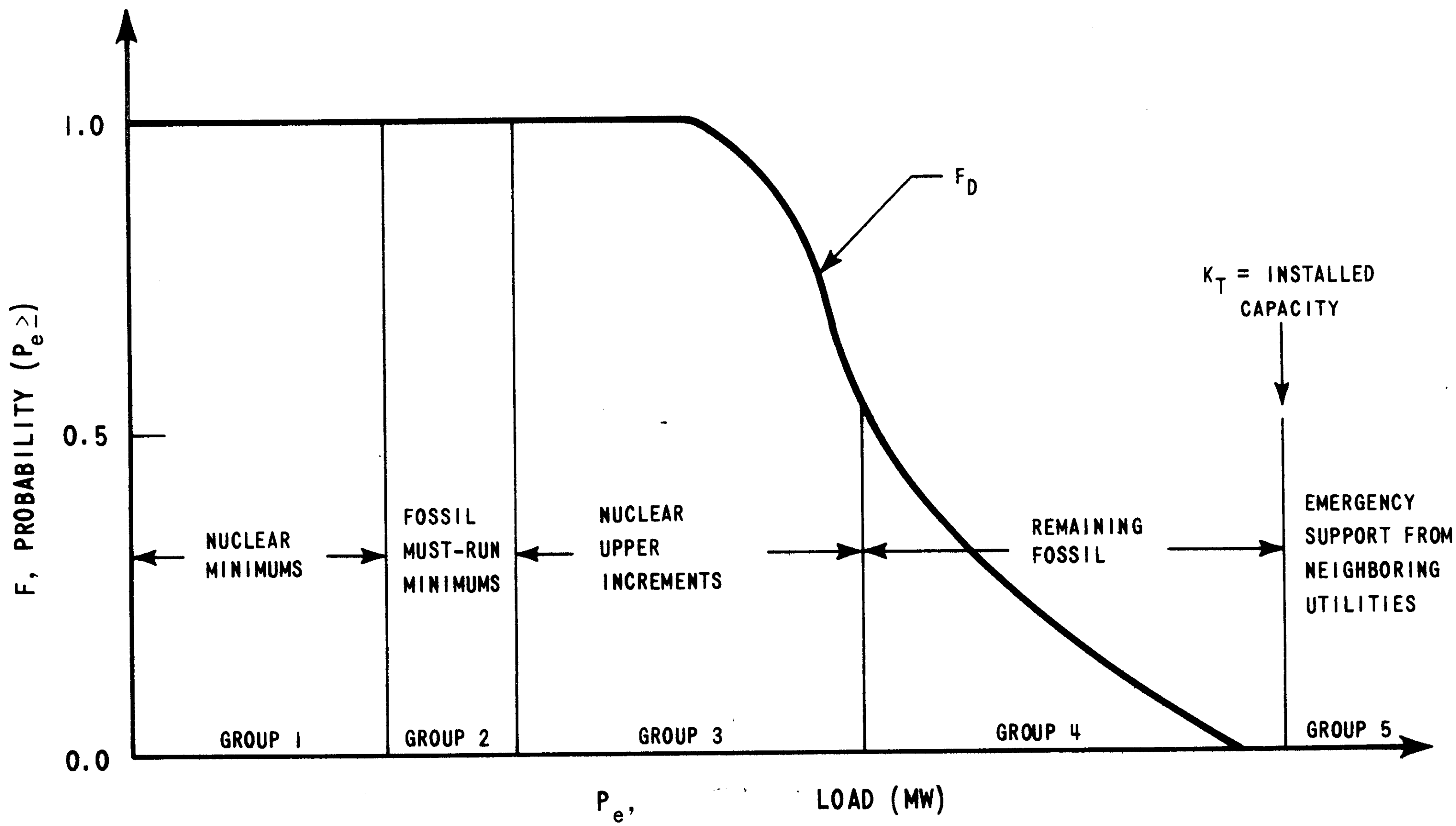
An extremely important conclusion regarding nuclear energy production can be deduced by combining the simple logic of the optimized loading order presented in Sections 2.4.3 and 3.2 and the purely mathematical properties of the Booth-Baleriaux model as discussed in Section 3.3.1.5.

Conclusion: Irrespective of the the intra-group loading order of the nuclear increments, the period's nuclear potential N_p is a constant.

Consider Figure 3.8 which presents a typical period load-duration curve being satisfied by a nuclear utility system using a loading order as suggested in Section 3.2. Proceeding from left to right through the startup and loading order, the first two groups of increments are the nuclear minimums (group 1) and the fossil minimums (group 2) for the

Figure 3.8

Loading Order Sub-Groups



must-run units. Since today's nuclear units all possess incremental costs on the order 0.9 to 1.5 $/MWH, next comes an amorphous block of capacity comprised of all the nuclear upper increments (group 3). (It is assumed that there are units in group 2. Otherwise, groups 3 and 4 must be mixed in order to provide spinning reserve.) After group 3 comes the well-ordered, but much more expensive, remaining fossil equipment (group 4) costing from 2 $/MWH on up. Beyond this installed capacity, are the emergency resources of neighboring utilities (group 5).

The conclusion is postulated as follows:

Given two alternative loading orders for group 3 ($g = 3A$ and $g = 3B$), show that the nuclear potentials are equal:

$$N_{g=3A} = N_{g=3B} \tag{3.61}$$

The other group loading orders remain the same. For instance,

$$g = 4A \equiv 4B \tag{3.62}$$

Since,

$$N \equiv E_{g=1} + E_{g=3} \tag{3.63}$$

The question becomes,

$$E_{g=1A} + E_{g=3A} \overset{?}{=} E_{g=1B} + E_{g=3B} \tag{3.64}$$

Since groups 1 and 2 remain the same and precede group 3, the conclusions of Section 3.3.1.5 dictate that those groups produce the same energy. Dropping the "g =" notation for convenience,

$$E_{1A} = E_{1B} \tag{3.65}$$

and

$$E_{2A} = E_{2B} \tag{3.66}$$

Moving through group 3, the first increment of group 4 is loaded utilizing the F curve remaining after the last nuclear increment has been convolved in. Since all of the nuclear increments have been convolved in, the current F must be identical for the two alternatives since the order they were included is immaterial. Thus, all of the Booth-Baleriaux calculations for group 4 will be identical and,

$$E_{4A} = E_{4B} \tag{3.67}$$

As for D_U ($\equiv E_{g=5}$), Section 3.3.1.5 already stated that it is invariant. Thus,

$$E_{5A} = E_{5B} \tag{3.68}$$

Since the same customer demand is satisfied for both alternatives,

$$\begin{array}{ccccccccccc} D_T & = & E_{1A} & + & E_{2A} & + & E_{3A} & + & E_{4A} & + & E_{5A} \\ \| & & \| & & \| & & & & \| & & \| \\ D_T & = & E_{1B} & + & E_{2B} & + & E_{3B} & + & E_{4B} & + & E_{5B} \end{array} \quad (3.69)$$

With four of the five components on the right-hand side being equal, the remaining components must also be equal,

$$E_{3A} = E_{3B} \quad (3.70)$$

and Equation (3.64) is, in fact, true.

Therefore,

$$E_{g=1} + E_{g=3} = N = \text{constant} \quad (3.71)$$

independent of the intra-nuclear loading order.

Q.E.D.

As a matter of fact, a much more general conclusion can be proven in an analogous manner: Each sub-group of unit increments produces the same energy regardless of the intra-group loading orders, provided that the inter-group loading order remains the same.

Example 6 on the Reference System is presented in Table 3.5. It involves the rearrangement of nuclear upper increments V-2 and III-2 with respect to Example 5 of Table 3.3. In both examples, the two upper nuclear increments produced a total of 402.14 GWH and a system nuclear potential of 653.99 GWH.

TABLE 3.5

Example 6 on Reference Utility System:

"Multiple Increment Booth-Baleriaux Model (III-2, then V-2)"

(Among Nuclear Upper Increments III-2, then V-2)

(See Appendix C for further details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 $)
I	1	9	11.93	193.3
II	1	6	36.71	201.9
	2	8	14.01	59.5
III	1	2	65.70	149.8
(Nuclear)	2	4	131.40	249.7
IV	1	3	131.40	515.1
	2	7	70.85	235.2
V	1	1	186.15	418.8
(Nuclear)	2	5	270.74	463.0
Utility Production			918.89	2486.3
Emergency Purchases (10 $/MWH)			30.11	301.1
Total			949.00	2787.4

Loss-of-load Probability, LOLP = 15.6%

The conclusion concerning constant nuclear potential is extremely important to the structure of the nuclear power management model of Figure 2.21 because the Booth-Baleriaux simulation in the SIM does not require detailed reactor-by-reactor nuclear incremental costs. (Recall that "ballpark" nuclear incremental costs were, nonetheless, useful in establishing the loading order groups.) Any intra-nuclear order is as good as any other for calculating the system nuclear potential. The model merely picks an arbitrary order for the amorphous nuclear group (g=3), simulates the system and totals the nuclear production to get the constant nuclear potential.

Furthermore, after all periods have been simulated by the SIM, the SOM begins optimizing the intra-nuclear production of the nuclear potentials. Since period nuclear potential is a constant regardless of the various detailed incremental costs (i.e., loading orders) calculated at each iteration by the CORSOM's (see Section 2.5), the iterations in Figure 2.21 need not loop back through the SIM. All of the above, make this an extremely important conclusion.

3.4 Estimating Startup-Shutdown Cost

To accurately calculate the startup-shutdown cost component of operating revenue requirements, an hour-by-hour production scheduling model is required. Having sacrificed the detailed chronological load shapes for the more convenient load-duration curves (see Section 2.1.1) covering much longer periods of time, it becomes necessary to estimate startup-shutdown costs by an approximate technique.

Consider Figure 3.9 [after (18)] which displays qualitatively the approximate relation between Ω, the frequency of startup-shutdowns (per day) and L'_{r1}, the availability-based capacity factor for the unit's first increment. That is,

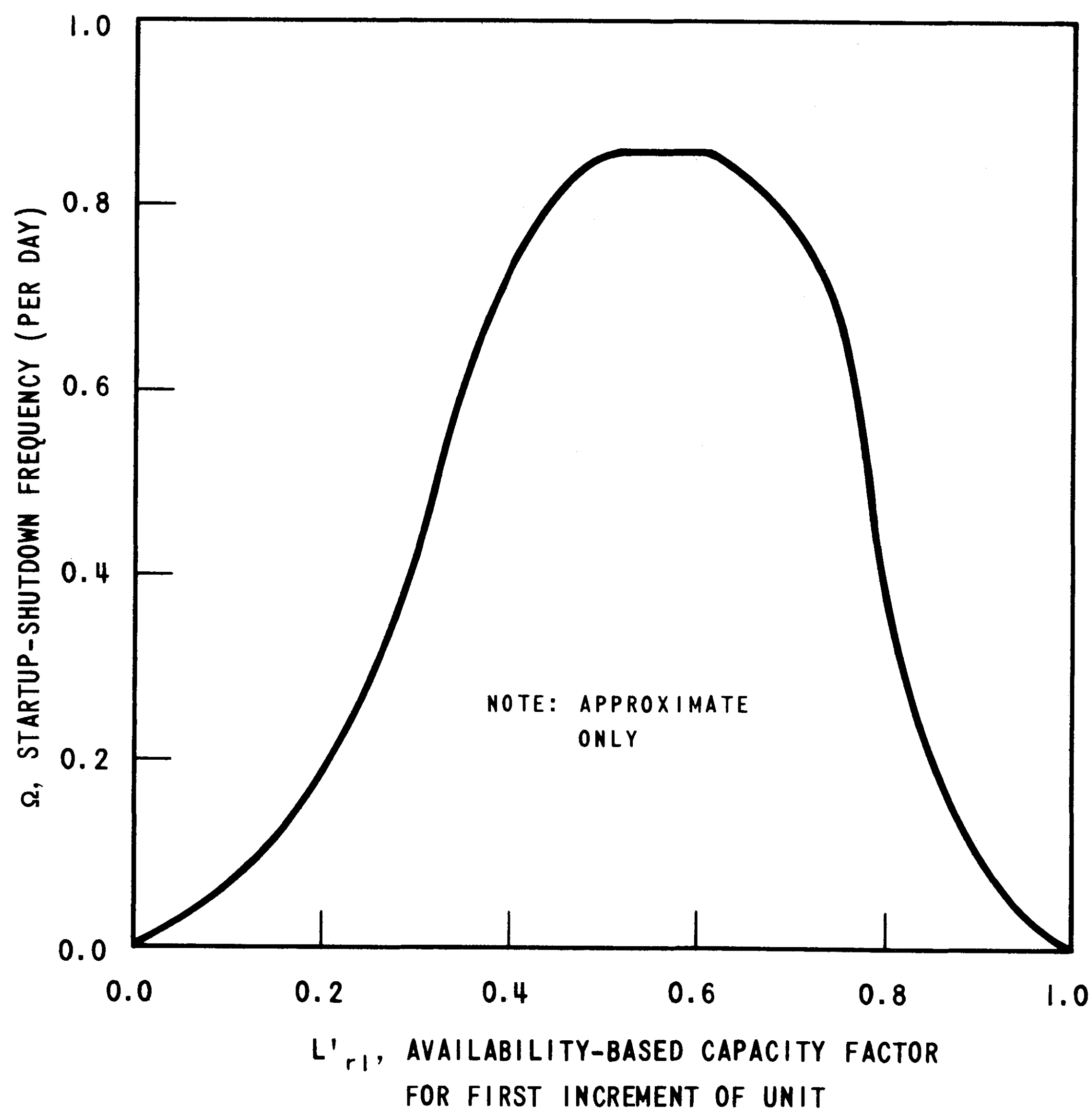
$$L'_{r1} = \frac{1}{K_{r1}} \int_{P^{\circ}_{r1}}^{P^{\circ}_{r1}+K_{r1}} F^{wo}_{r1}(P_e)\,dP_e \qquad (3.72)$$

For must-run units, L'_{r1} equals 1 and Ω equals 0. For very expensive peaking units, L'_{r1} approaches 0 and Ω again approaches 0. As expected, units never shutdown and units never started-up incur no startup-shutdown cost. In between are those units started-up and shutdown on a daily basis and, hence, Ω approaches one.

Since unit startup-shutdown cost Q_r is specified in time independent units of equivalent thermal energy input, multiplying it by ϕ_r, unit thermal energy cost for the time period, permits escalation in terms of undiscounted dollars. Since L'_{r1} is easily extracted for each unit during the Booth-Baleriaux simulation, the fractional starts per

Figure 3.9

Example of Startup-Shutdown Frequency versus Availability-Based Capacity Factor [After (18)]



day are easily estimated given the proper dependence of Ω upon L'_{r1}. Thus, a period T'/24 days long, incurs total startup-shutdown cost amounting to

$$X_S = \frac{T'}{24} \sum^{R} \phi_r \; Q_r \; \Omega(L'_{r1}) \qquad (3.73)$$

Table 3.6 presents the detailed calculation of unit startup-shutdown costs for Example 5 which was presented in Table 3.3.

3.5 Determining Cost of Emergency Purchases

The determination of expenditures relative to D_U emergency electricity purchases from neighboring utilities is straight-forward once the SIM has been given an $\bar{e}_U$ average cost for this emergency support. The total expenditure is merely,

$$X_U = \bar{e}_U \cdot D_U \qquad (3.74)$$

3.6 SYSINT, A Computerized Version of the SYStem INTegration Model

SYSINT, a 2000 card Fortran IV version of the SYStem INTegration Model is detailed in Appendix E. This section merely summarizes its capabilities.

The standard two-state forced-outage model (performs or fails) is employed. A single startup frequency curve $\Omega(L'_{r1})$ is input for the entire horizon. The limitations of the current version, though easily altered, are as follows:

Table 3.6

Calculation of Startup-Shutdown Costs for Example 5 on Reference Utility System

Unit r	Fuel Cost ϕ_r ¢/MegaBTU	Susd. Heat Reqt. Q_r MegaBTU	Avl.-bsd. Capacity Factor L'_{r1}	Susd. Frequency $\Omega(L'_{r1})$[1] per day	Unit Susd. Cost $\phi_r Q_r$ $	Daily Susd. Cost $\phi_r Q_r \Omega$ $/day	Period Susd. Cost $\phi_r Q_r \Omega T'/24$ $
I	90	50	.172	.152	45	6.85	208
II	50	800	.529	.860	400	344.00	10,460
III	19	1200	1.000	-0-	228	-0-	-0-
IV	40	3600	1.000	-0-	1440	-0-	-0-
V	18	2400	1.000	-0-	432	-0-	-0-

Total Startup-Shutdown Cost~$12,670

[1]See Figure 3.9

(1) up to 100 units (including retirements and additions),

(2) up to 5 valve points for each unit,

(3) no limit on number of strategies per computer run

(4) up to 100 time periods per strategy and

(5) up to 25 typical load-duration "shapes", stored in completely normalized form (i.e., peak demand also equals one).

The multi-period strategy is input for each unit in the following form:

(1) the period installed,

(2) period just prior to retirement and

(3) up to 20 intermediate periods of downtime for maintenance or refueling.

For each period the following data may be input or altered:

(1) Choice of load-duration shape,

(2) Forecasted peak demand,

(3) Expected spinning reserve requirement,

(4) Length of time period,

(5) Average cost of emergency purchase energy,

(6) Fuel cost for each unit (optional initial guess for nuclear units),

(7) Performance probability for each unit and

(8) Startup order indicating must-run units and peaking equipment.

As for typical running time, each period of a simulation of a utility system containing 40 units with a total of 150 valve points requires approximately 2.5 CPU sec on an IBM 370 model 155 computer in an MVT environment. The code itself requires 108 K bytes of storage, i.e., not including the computer system supervisor. Total core requirements are thus approximately 134 K bytes.

Data transfer from SYSINT to SYSOPT (see Section 4.6 and Appendix F) is completely automated via either disk, magnetic tape or punched cards.

3.7 Summary of the SIM

For each multi-year refueling and maintenance strategy, the SIM performs period-by-period detailed production scheduling utilizing the Booth-Baleriaux probabilistic utility system model. Besides, calculating the system nuclear potential N (shown to be a constant), the model outputs the following system cost components:

(1) X_F, the fossil fuel expense related to electricity production,

(2) X_S, the startup-shutdown cost and

(3) X_U, the cost of emergency energy purchases.

This and other data are then passed to the SOM of Chapter 4 for iterative optimization of the production of the nuclear potential and present-valuing of all the cost components to obtain the final ORR for the given strategy.

CHAPTER 4

THE SYSTEM OPTIMIZATION MODEL

4.1 Overview of the SOM

The System Optimization Model (SOM), shown schematically in Figure 2.22 performs two tasks for each of the possible alternative refueling and maintenance strategies under investigation. The first, and most difficult, is optimizing each reactor's energy output so as to produce the required system nuclear potential for each period with a minimum total revenue requirement for nuclear fuel over the multi-year horizon (see Section 4.2.1). The SOM receives, as input, the period-by-period results (see Section 3.7) of the System Integration Model (SIM) which are used to formulate the constraints on this optimization (see Sections 4.2.2 to 4.2.4). Interfacing with a CORe Simulation and Optimization Model (CORSOM) for each reactor (see Section 2.5.4), the SOM passes a set of reactor-cycle energies and receives the minimum total reactor fuel revenue requirement to the horizon and the partial derivatives of this cost with respect to each of the cycle energies. These nuclear incremental cost data are then used to iterate toward the optimum set of cycle energies (see Section 4.4).

When the system nuclear fuel revenue requirement has been thus minimized, the supervisory task commences. This second task (see Section 4.5) merely involves present-valuing the non-nuclear period expenses and adding in the total nuclear revenue requirement to determine the total operating revenue requirement for the particular possible alternative strategy under investigation. With the completion of this task, processing of the refueling and maintenance strategy is complete and optimization of the next such strategy may begin.

4.2 Elements of Optimization Problem in SOM

The following sections outline the elements of the SOM's optimization problem. In Section 4.2.1, the objective function of the optimization is first presented straightforwardly as the total system nuclear fuel revenue requirement. Then assumptions and simplifications are made to reduce the objective function to a form readily solvable by an iterative gradient technique. Next, the constraints on the optimization are discussed in detail.

Reviewing the context of this optimization, the principal SIM result passed to the System Optimization Model is the nuclear potential, N_p, which is equal to the sum of the subset of reactor period energy productions, E_{rcp}, for each period. As indicated in Section 3.3.3, each N_p value is independent of the detailed loading order of the nuclear increments. Hence, for each period p subset of E_{rcp}, there exist many possible combinations of each reactor's E_{rcp} which will satisfy N_p. The SOM is able to determine these additional possible subsets of E_{rcp} more rapidly than if the SIM is used repeatedly, thus eliminating the need for more than one SIM calculation per period. The object of the SOM is then to determine, subject to certain feasibility constraints, which combination of these subsets of E_{rcp} for each period in the entire planning horizon results in the minimum system revenue requirement.

The first constraint (see Section 4.2.2) ensures that each system production subset of E_{rcp} satisfies the nuclear potential, N_p, that was calculated by the SIM for that period. Next, the reactor production constraints (see Section 4.2.3.1) put limits on each reactor's maximum and minimum period energy production. These represent the SIM cases when each nuclear unit's total upper capacity was loaded first or last, respectively, within the system upper nuclear capacity. Finally, a shape constraint (see Section 4.2.4) is used to select subsets of reactor-period productions, E_{rcp}, which are compatible with the shape of the equivalent load curve.

4.2.1 Objective Function

The optimization seeks to minimize $\overline{\overline{TC}}$ ($\equiv RR_N$), the system nuclear fuel revenue requirement, over the multi-year horizon as a function of $\mathcal{E}$, the set of all E_{rcp},

$$\text{minimize } \overline{\overline{TC}} = \overline{\overline{TC}} \; (\, \mathcal{E} \,) \qquad (4.1)$$

Since $\overline{\overline{TC}}$ is the sum of the various reactor fuel costs $\overline{\overline{TC}}_r$ calculated by the CORSOM's, which, in turn, are really functions of the E_{rc} cycle energies,

$$\overline{TC}\ (\mathcal{E}) = \sum^{R} \overline{TC}_r\ (E_{r1}, E_{r2}, \ldots) \quad (4.2)$$

As Section 2.3 pointed out, $\overline{TC}_r$ is non-linear and non-separable. However, since $\overline{TC}_r$ has been minimized by the CORSOM for the given set of E_{rc}, it must be well-behaved in the sense that it is continuous and unimodal, increasing with increasing E_{rc}. Hence $\overline{TC}_r$ is differentiable and

$$\lambda_{rc} \equiv \frac{\partial \overline{TC}_r}{\partial E_{rc}} > 0 \quad (4.3)$$

Equation (4.3) permits taking the total differential of Equation (4.2),

$$d\overline{TC} = \sum^{R} \sum^{C} \frac{\partial \overline{TC}_r}{\partial E_{rc}}\, dE_{rc} \quad (4.4)$$

Since $\overline{TC}$ is a point function, given a cost $\overline{TC}^t$ at $\mathcal{E}^t$ trial set of E_{rcp}, the cost $\overline{TC}^{t+1}$ at any other set $\mathcal{E}^{t+1}$ can be obtained by integrating Equation (4.4),

$$\overline{TC}^{t+1} - \overline{TC}^{t} = \int_{\mathcal{E}^t}^{\mathcal{E}^{t+1}} \sum^{R} \sum^{C} \left(\frac{\partial \overline{TC}_r}{\partial E_{rc}} \right)_{\mathcal{E}^t \to \mathcal{E}^{t+1}} dE_{rc} \quad (4.5)$$

(Section 5.6.1 of Chapter 5 refers to the integral on the right-hand side as the actual or true difference between $\overline{TC}^t$ and $\overline{TC}^{t+1}$, $\sum_{act}^{t+1}$.)

To be rigorously accurate, the line integral must follow a tortuous route through the multi-dimensional space from

$\mathcal{E}^{t}$ to $\mathcal{E}^{t+1}$. Thus, each partial derivative must be calculated along a different line segment connecting two adjacent intermediate points along the route. It is far easier to calculate each partial derivative only about the current trial point $\mathcal{E}^{t}$ itself, $(\partial TC_r/\partial E_{rc})_{\mathcal{E}^t}$. If these derivatives are used to replace those in Equation (4.5), an error term δ^{t+1} must be included to correct for the approximation,

$$\overline{TC}^{t+1} = \overline{TC}^{t} + \delta^{t+1} + \int_{\mathcal{E}^t}^{\mathcal{E}^{t+1}} \sum^{R} \sum^{C} \left(\frac{\partial \overline{TC}_r}{\partial E_{rc}}\right)_{\mathcal{E}^t} dE_{rc} \quad (4.6)$$

Since each differential is only about its own E_{rc}^{t}, the integral limits reduce to E_{rc}^{t} to E_{rc}^{t+1} and the two summations may be taken outside,

$$\overline{TC}^{t+1} = \overline{TC}^{t} + \delta^{t+1} + \sum^{R} \sum^{C} \int_{E_{rc}^t}^{E_{rc}^{t+1}} \left(\frac{\partial \overline{TC}_r}{\partial E_{rc}}\right)_{\mathcal{E}^t} dE_{rc} \quad (4.7)$$

or

$$\overline{TC}^{t+1} = \overline{TC}^{t} + \delta^{t+1} + \sum^{R} \sum^{C} \int_{E_{rc}^t}^{E_{rc}^{t+1}} \lambda_{rc}^{t} \, dE_{rc} \quad (4.8)$$

Defining the double summation term as Σ_{EST}^{t+1}, the estimated change in $\overline{TC}^{t}$,

$$\overline{TC}^{t+1} = \overline{TC}^{t} + \delta^{t+1} + \Sigma_{EST}^{t+1} \quad (4.9)$$

Provided that the error in the approximation or estimation δ^{t+1} is sufficiently small (see Section 5.6.1), Equation (4.9) provides an excellent basis for re-formulating the non-linear objective function and hence, the optimization, into an iterative procedure:

Given a trial point $\mathcal{E}^t$ with cost $\overline{TC}^t$ and incremental costs λ^t_{rc}, the next feasible trial point $\mathcal{E}^{t+1}$ is determined that minimizes $\overline{TC}^{t+1}$.

Since $\overline{TC}^t$ is constant within the iteration, the minimization of $\overline{TC}^{t+1}$ may be replaced by the approximately equivalent minimization of Σ^{t+1}_{EST}. Using the new $\mathcal{E}^{t+1}$, the CORSOM's can then generate the corresponding $\overline{TC}^{t+1}_r$ and λ^{t+1}_{rc}. The next SOM iteration then seeks to minimize Σ^{t+2}_{EST}, and so on.

In general, convergence of $\mathcal{E}$ and $\overline{TC}$ may occur but globality of the optimum $\mathcal{E}^*$ and $\overline{TC}^*$ cannot be guaranteed. However, for the special case of a convex $\overline{TC}$ ($\mathcal{E}$), both convergence and globality are guaranteed (54). That is,

$$\frac{\partial^2 \overline{TC}_r}{\partial E^2_{rc}} \text{ must be } \geq 0 \qquad (4.10)$$

or

$$\frac{\partial}{\partial E_{rc}}\left(\frac{\partial \overline{TC}_r}{\partial E_{rc}}\right) = \frac{\partial \lambda_{rc}}{\partial E_{rc}} \text{ must be } \geq 0 \qquad (4.11)$$

The work of Widmer (57) and Watt (55) have shown that this is a reasonable assumption--the nuclear incremental cost λ_{rc} increases or, at least, does not decrease with the cycle energy E_{rc}. That is, each additional increment of cycle energy (i.e., reload enrichment) costs at least as much as the previous increment.

To summarize, given that $\overline{TC}$ ($\mathcal{E}$) is convex, the iterative optimization will converge to the global optimum using as the objective function,

$$\text{minimize} \quad \sum_{EST}^{t+1}(\mathcal{E}) = \sum^{R}\sum^{C}\int_{E_{rc}^{t}}^{E_{rc}^{t+1}} \lambda_{rc}^{t}\, dE_{rc} \tag{4.12}$$

The above objective function is actually not a function of the period productions, but only of the cycle subtotals, the E_{rc} cycle energies. However, all of the various constraints on the optimization, discussed in the following Sections 4.2.2-4.2.4, are period constraints and involve E_{rcp} explicitly.

4.2.2 System Production Constraint

The constraint on system production requires that in each period the reactors produce sufficient energy to meet the nuclear potential,

$$\sum^{R} E_{rcp} = N_p \quad \text{for all p} \tag{4.13}$$

Calculation of N_p has already been discussed in Section 3.3.3.

4.2.3 Reactor Production Constraint

There are two types of reactor production constraints. The first, discussed in Section 4.2.3.1 brackets the permissible values of each reactor's production for each of the Z periods within the planning horizon,

$$E_{rcp}^{min} \le E_{rcp} \le E_{rcp}^{max} \quad \text{for all r and p} \qquad (4.14)$$

The second, discussed in Section 4.2.3.2, specifies the reactor energy production beyond the planning horizon. These horizon end conditions permit the CORSOM's to evaluate and cost (at least approximately) the reactivity requirements of cycles beyond the end of the planning horizon. The goal is to normalize strategy vs. strategy horizon end effects. To accomplish this,

$$E_{rC} = E_{rCp} + E_{r,C,Z+1} \quad \text{for all r} \qquad (4.15)$$

where $E_{r,C,Z+1}$ = energy held over for production by reactor r beyond the horizon cycle C (in fictitious period Z+1). In addition, $E_{r,C+1}$, $E_{r,C+2}$, etc. are specified.

4.2.3.1 Typical Period

The reactor period production constraint [Equation (4.14)] merely establishes the limits on each reactor's production. For the trivial case when unit r is down for refueling in period p,

$$E_{rcp}^{min} = E_{rcp}^{max} = 0 \qquad (4.16)$$

The SOM pre-calculates the other minimums and maximums using results from SIM. Two important load-duration curves, (F^{min} and F^{max}), not previously discussed, are among these results (see Figure 4.1).

The F^{min} was the SIM's current F immediately prior to the deconvolution required to load the first nuclear upper increment of group 3 (see Figure 3.8). That is, F^{min} includes forced-outage allowances for all of the nuclear minimums (group 1) plus any must-run fossil minimums (group 2). This curve is used to determine the E_{rcp}^{max} since the maximum energy a reactor's upper increments can produce occurs when all of its remaining capacity,

$$k_r \equiv K_{rI} - K_{r1} = \dot{K}_r - K_{r1} \qquad (4.17)$$

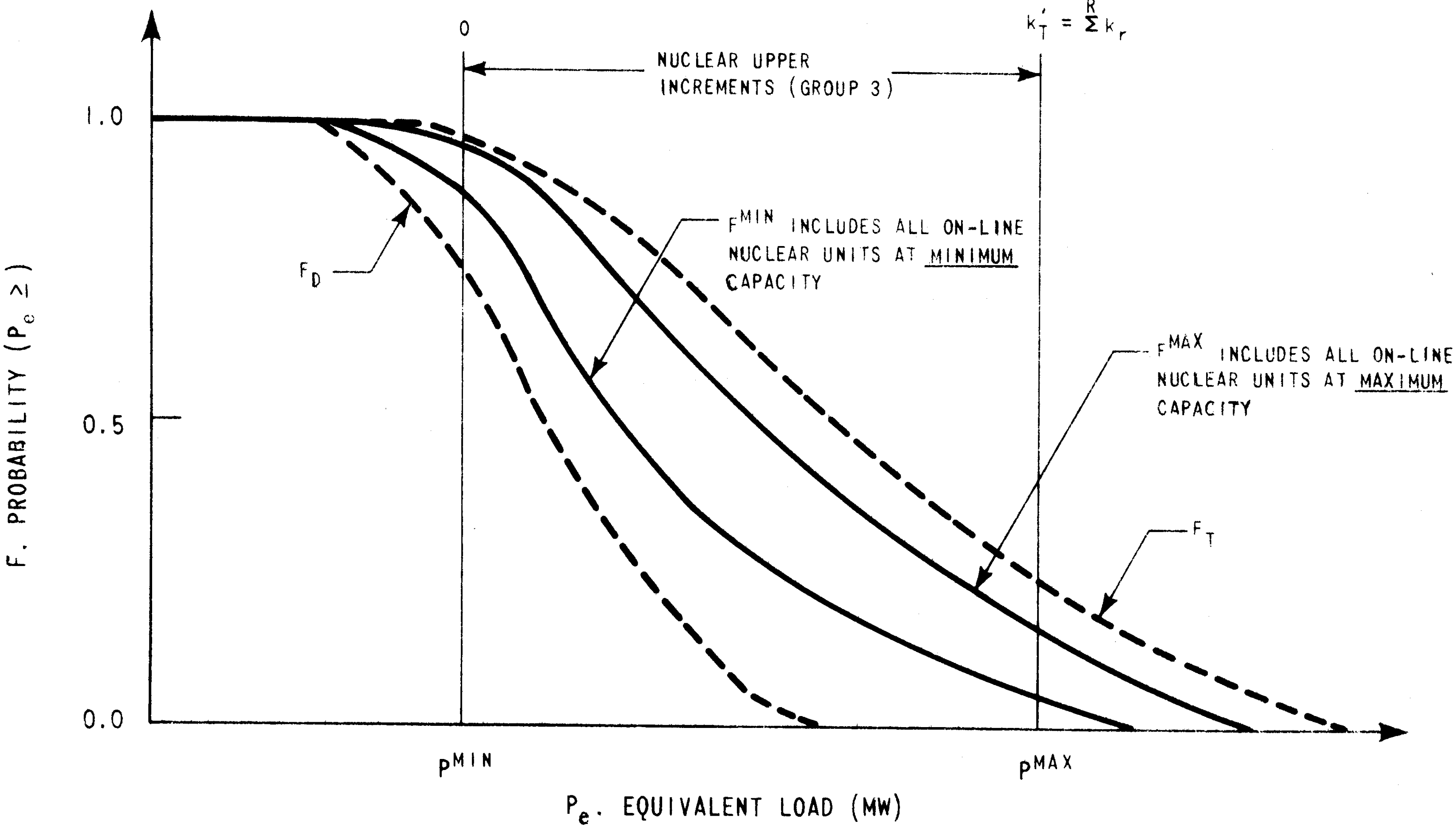
is loaded at the very beginning of this group 3.

Thus to determine E_{rcp}^{max}, the following two step procedure is performed (see Figure 4.2) for each on-line reactor:

<u>Step 1</u>: From F^{min}, which includes all on-line nuclear minimums, deconvolve the initial increment of unit r,

Figure 4.1

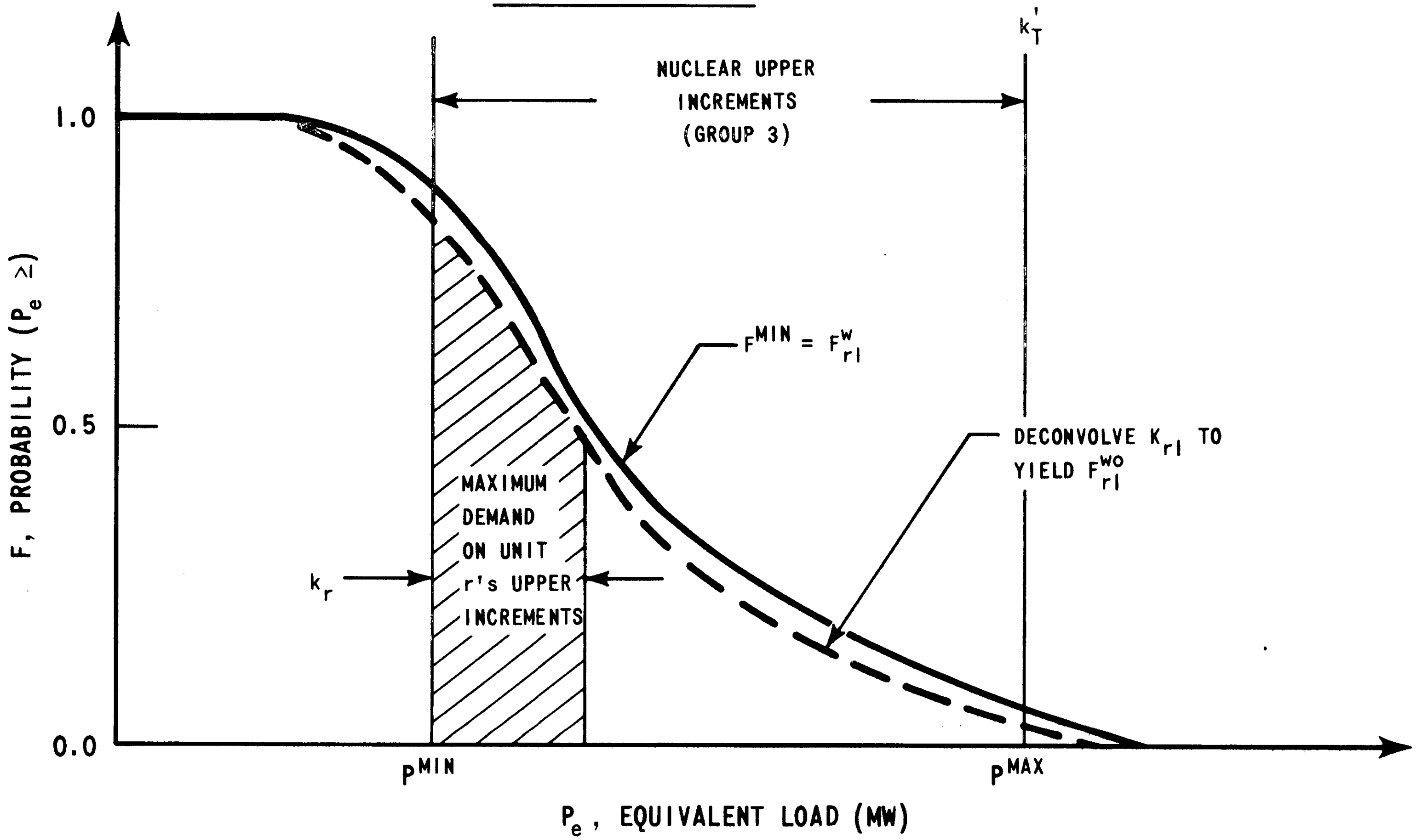
F^{min} and F^{max} Load-Duration Curves Required by SOM



6253-30

Figure 4.2

Calculation of E_{rcp}^{max}



6253-31

$$F_{r1}^{wo}(P_e) = \frac{1}{p_r}\left[F^{min}(P_e) - q_r F_{r1}^{wo}(P_e - K_{r1})\right] \tag{4.18}$$

<u>Step 2:</u> Since F_{r1}^{wo} is the proper curve for loading the remaining k_r MW in order to maximize E_{rcp},

$$E_{rcp}^{max} = E^{\circ}_{rcp} + T' p_r \int_{P^{min}}^{P^{min} + k_r} F_{r1}^{wo} \, dP_e \tag{4.19}$$

where E°_{rcp} is the invariant energy production of the unit's first K_{r1} MW.

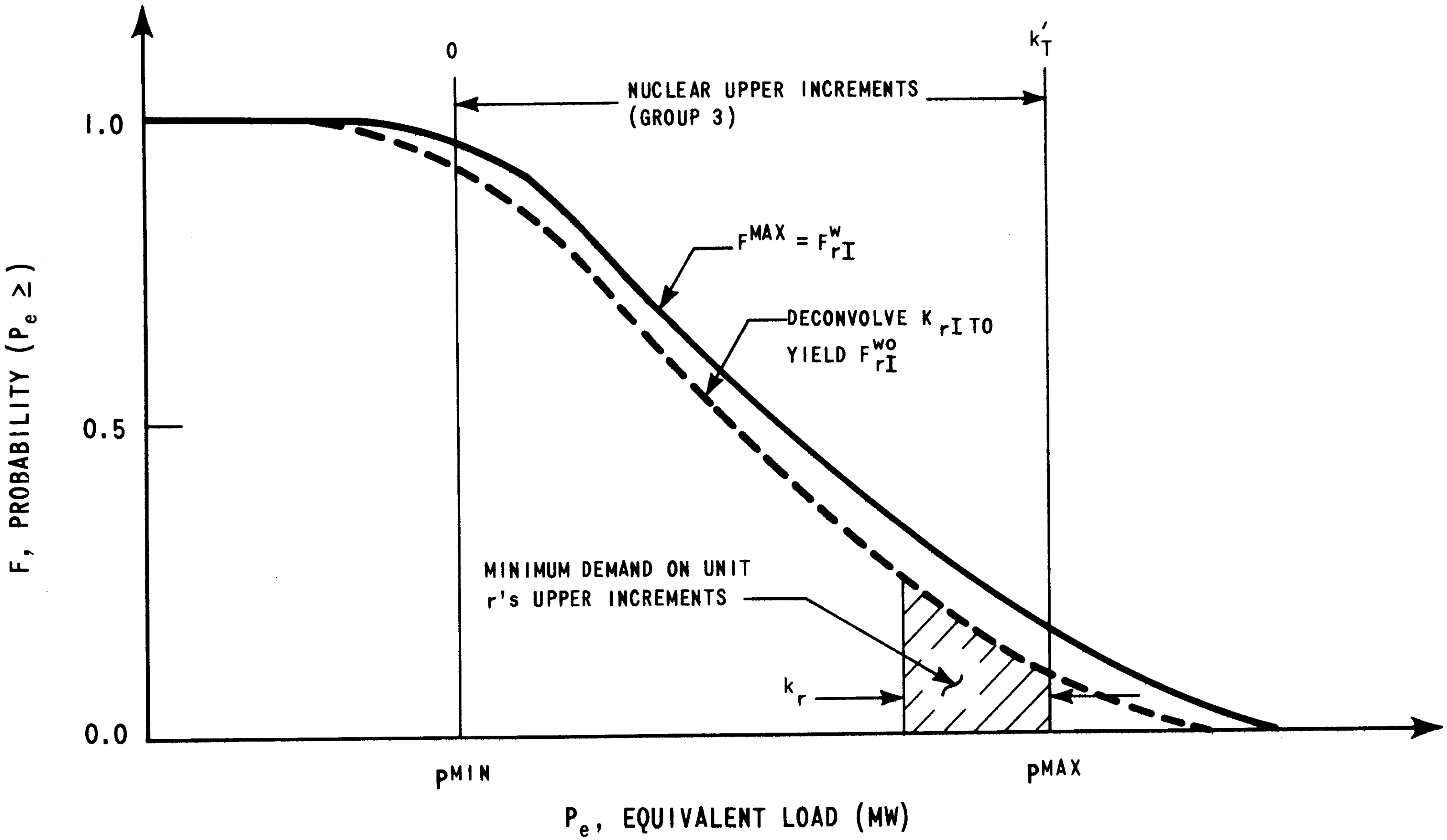
To determine E_{rcp}^{min} requires the F^{max} of Figure 4.1, which represents the SIM's current F after the last nuclear upper increment of group 3 has been convolved in. That is, F^{max} includes any fossil must-run minimums plus all of the nuclear maximums. Whereas E_{rcp} was maximized when k_r MW were first in group 3, minimum reactor energy production for the period occurs when unit r's k_r MW are the very last in group 3 to be loaded. Thus, the following two step procedure is applied to F^{max} for each reactor (see Figure 4.3):

<u>Step 1:</u> From F^{max}, which includes all on-line nuclear units at their maximum capacity, deconvolve the entire K_{rI} MW of unit r,

$$F_{rI}^{wo} = \frac{1}{p_r}\left[F^{max}(P_e) - q_r F_{rI}^{wo}(P_e - K_{rI})\right] \tag{4.20}$$

Figure 4.3

Calculation of E_{rcp}^{min}



6253-32

Step 2: Since F_{rI}^{wo} is the proper curve for loading the remaining k_r MW in order to minimize E_{rcp},

$$E_{rcp}^{min} = E_{rcp}^{o} + T'p_r \int_{P^{max}-k_r}^{P^{max}} F_{rI}^{wo} \, dP_e \tag{4.21}$$

4.2.3.2 Horizon End Condition

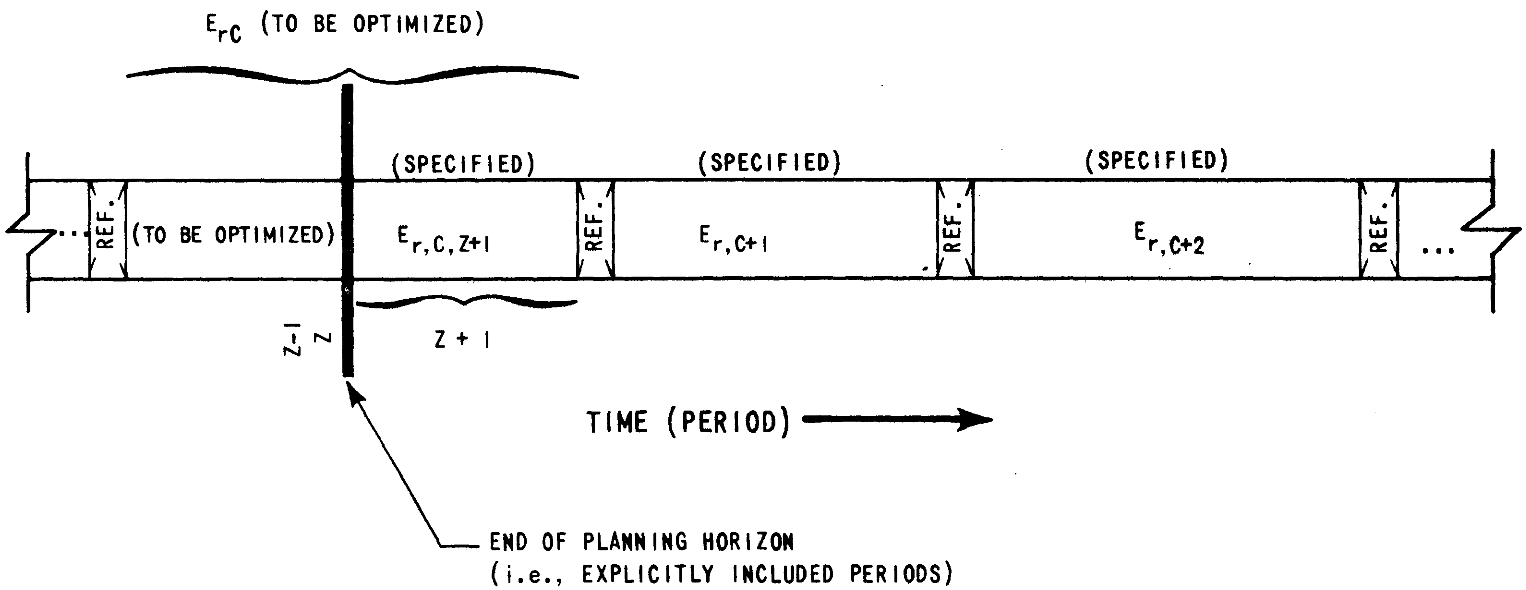
To properly evaluate fuel cycle costs (i.e., reload requirements and discharge characteristics) incurred within the planning horizon, each reactor's CORSOM must receive not only the energy of each of the C "included" cycles within the horizon, but also estimated cycle energies for several "excluded" cycles beyond the horizon. The specified end condition should match as closely as possible the same general operating philosophy (i.e., capacity factor) anticipated for the strategy's included cycles. That is, excluded cycles continue with similar cycle lengths in both energy and time as those within the horizon, not return to some arbitrary state, regardless of the particular included strategy.

To effect this requires an estimate of $E_{r,C,Z+1}$, the amount of cycle C energy held over beyond the horizon (for fictitious period Z+1) for production before the next refueling (see Figure 4.4),

$$E_{rC} = \sum^{Z} E_{rCp} + E_{r,C,Z+1} \tag{4.22}$$

Figure 4.4

Horizon End Condition



6253-33

In addition, several completely excluded cycle energies are estimated ($E_{r,C+1}$, $E_{r,C+2}$, etc.). Total system nuclear production from all reactors during the excluded cycles should be held constant for all refueling and maintenance strategies to ensure similar system-wide core energy content at the end of the planning horizon. Recall that the goal is to normalize strategy vs. strategy horizon end effects.

Since the end condition exists only in deference to the CORSOM's calculational requirements, it is not included explicitly in the mathematical formulation of the SOM's optimization problem summarized in Section 4.3.

4.2.4 Shape Constraint

The shape constraint is used to guarantee that the reactor energy productions within the period are, in the aggregate, compatible with the given equivalent load shape. In the Booth-Baleriaux calculations of the SIM, the various increments of each unit are assigned various segments of the equivalent load curve on a MW for MW basis. Summing the I increments of energy production E_{ri} for each unit,

$$E_r = \sum^{I} E_{ri} \tag{4.23}$$

These E_r represent each unit's energy production for the period using the specified increment-by-increment loading order. By the nature of the SIM calculation, any detailed loading order specifies a set of feasible E_r's for the period (i.e., a set of E_r's which are compatible with the shape of the equivalent load curve).

However, the optimization variable in SOM is not the detailed loading order, but each nuclear unit's period production E_{rcp}. Thus, the shape compatibility question becomes: "For a given subset of reactor-period energy productions (E_{rcp} for all r at p) whose sum equals the required period nuclear potential N_p from SIM, could a corresponding detailed reactor loading order be found that satisfies the period's equivalent load shape (calculated by SIM) yet results in the SOM's postulated E_{rcp}?" The shape constraint attempts to quantify the feasibility of finding such a loading order (yet circumvents actually having to perform the search or SIMulation).

The general form of the shape constraint will be shown to be second-order,

$$\sum^{R'_p} c_{1rp} E_{rcp} + \sum^{R'_p} c_{2rp} E_{rcp}^2 \leq c_p \tag{4.24}$$

where c_{1rp}, c_{2rp} and c_p are constants pre-calculated by the SOM from SIM results. While the system and reactor production constraints [Equations (4.13) and (4.14), respectively] are linear, (i.e., first order), the shape constraint Equation (4.24) is non-linear. As with all but the most trivial problems in operations research, non-linearities greatly complicate the optimization algorithm (see Section

4.4.3). The current discussion, however, concentrates solely on understanding "why" and "how" the shape constraint is formulated in the first place.

4.2.4.1 Purpose

To understand why the shape constraint is necessary, consider the following example which would otherwise be permitted by the SOM as a feasible solution. Assume the customer loads remain as on the Reference Utility System in Figure 2.9. However, assume for the sake of this example that the utility system itself consists of only six identical 400 MW nuclear reactors which, for simplicity in the example, have no forced-outages (p_r = 100%) and no minimum load constraint; therefore, $F_D = F_e$. Figure 4.5 portrays system production calculated by the SIM for the specified startup and loading order. Note that for this feasible production schedule, the SIM results indicate nuclear system production of

$$N_p = D_T = 949 \text{ GWH} \quad (4.25)$$

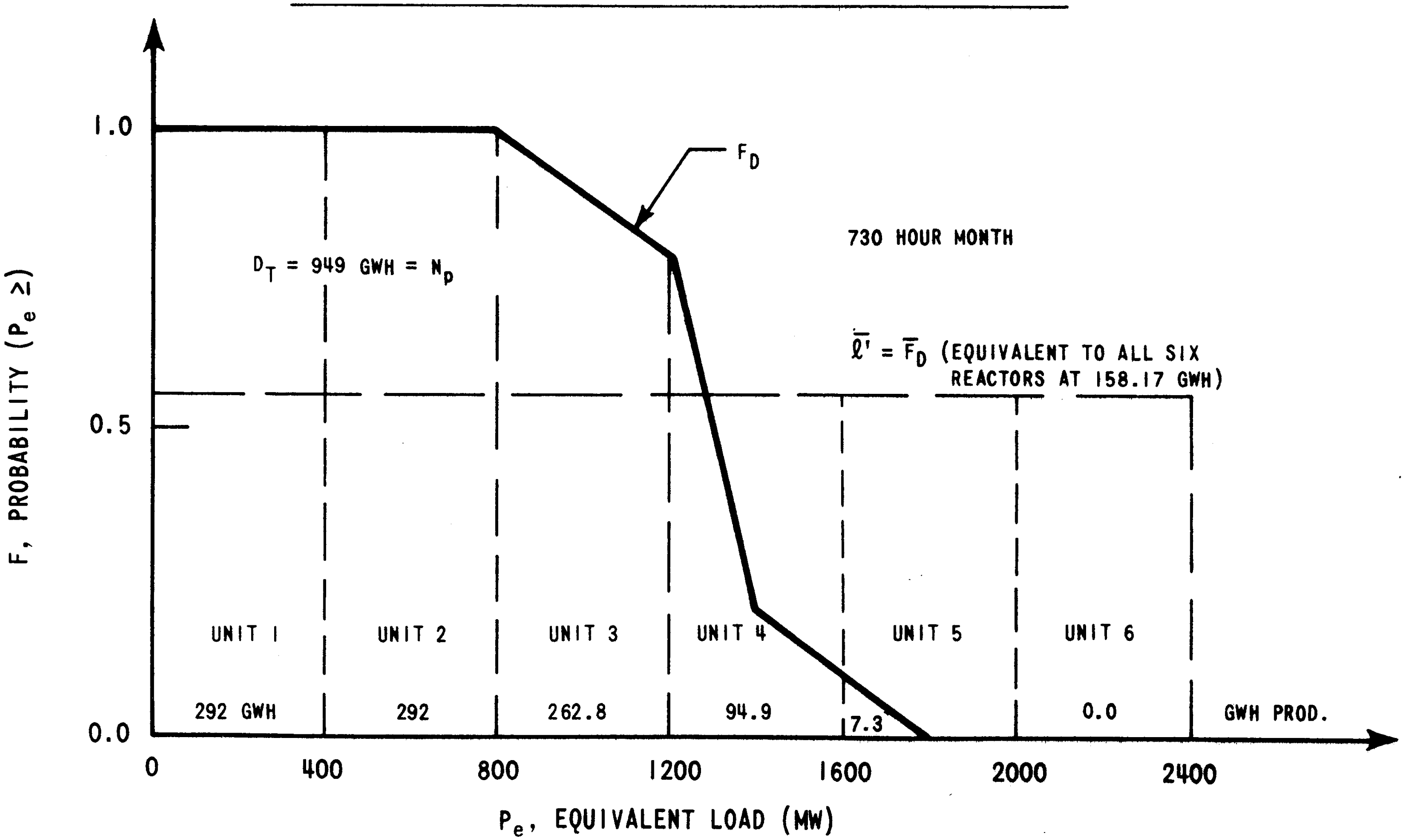
and reactor production limits equivalent to

$$E_{rcp}^{max} = 292 \text{ GWH} \quad (4.26)$$

$$E_{rcp}^{min} = 0 \text{ GWH} \quad (4.27)$$

Figure 4.5

Six Identical Reactors versus Reference Utility Customer Demand



Inserting these values in the two production constraints [Equation (4.13) and (4.14)] and ignoring any shape constraint, the SOM would be perfectly justified in postulating the production schedule shown in Figure 4.6 since the desired total energy N_p (proportional to area under the curve) is supplied. Comparing this production shape with that of the customers (F_D), the shape infeasibility is readily apparent since production never reaches a power level greater than 1400 MW while the customer demand is greater than that 20% of the time.

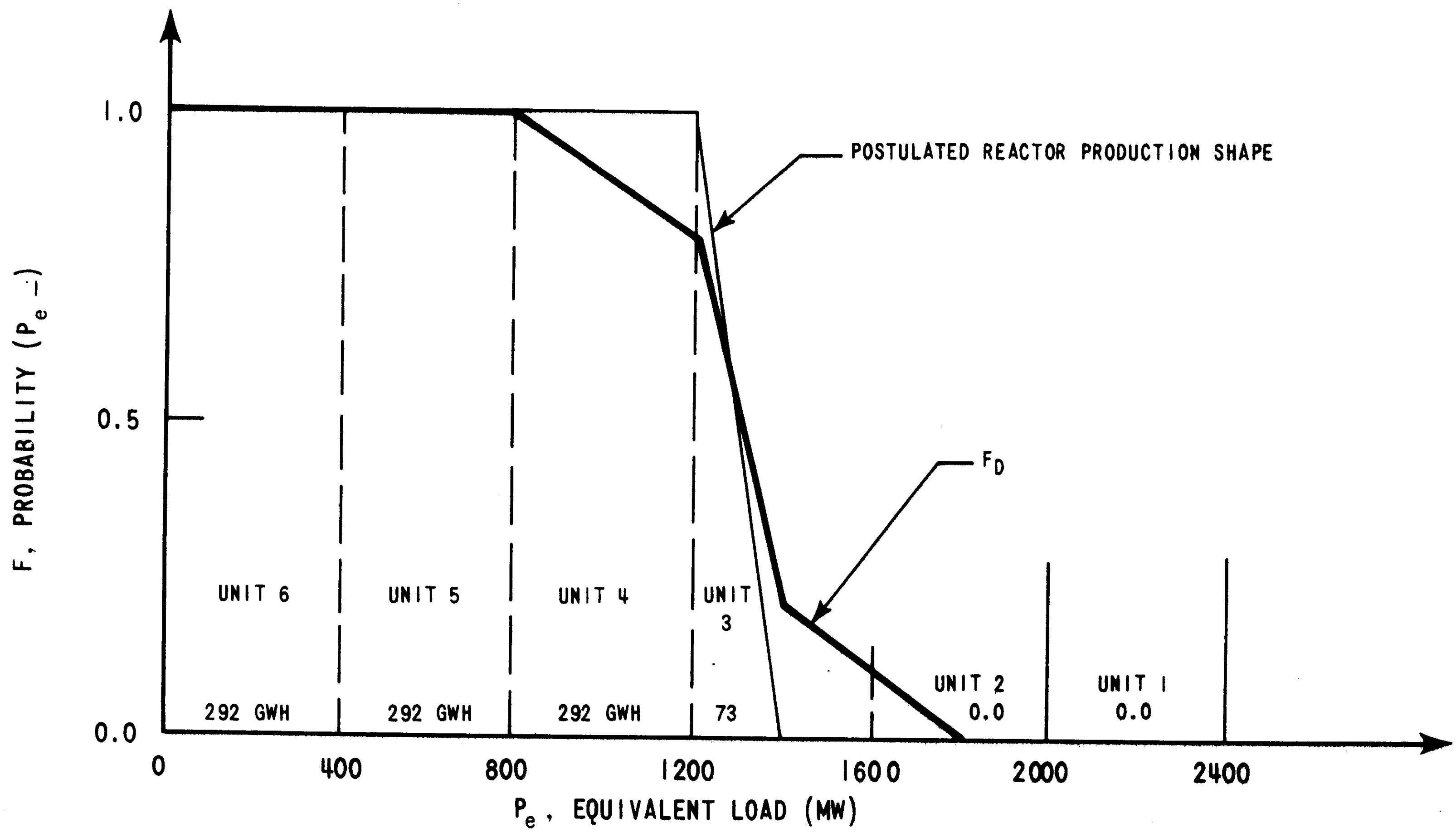
Thus, the optimization model must include either (a) some method of forcing each subset of E_{rcp} derived in the SOM to satisfy the load shape, or (b) include a constraint, or posteriori check, which rejects from further consideration any subsets of E_{rcp} which cannot satisfy the load shape. The latter method, referred to as a "shape constraint," is utilized in the model presented here.

Having established the necessity of a shape constraint, how might the "shape" be quantified?

First of all, the shape most indicative of the demands to be satisfied by each nuclear unit is not the direct customer load-duration curve F_D (unless all p_r are actually equal to 100%), but the equivalent load-duration curve F_e, which includes not only direct, but also indirect, customer loads. (Section 4.2.4.3 discusses the practical means by which the SOM determines F_e given F^{min} and F^{max}.) Furthermore, by focusing attention only on the nuclear units and assuming their size and economics make them all must-run

Figure 4.6

Infeasible Six Reactor Production Schedule



units, the pertinent range of F_e can be reduced to that segment served by the nuclear upper increments of the R' available (on-line) nuclear units (group 3 of Figure 4.1). Henceforth, the term "system shape" and symbol F_e refer to that segment of the equivalent load curve over the range of loads running from zero MW upper nuclear capacity to the system total availability-based nuclear upper increment capacity k_T' (i.e., each unit's first increment is excluded from the discussion since all K_{r1} MW are base-loaded),

$$k_T' = \sum^{R'} k_r = \sum^{R'} (K_{rI} - K_{r1}) \qquad (4.28)$$

In order to characterize the production schedule in terms of the optimization variables E_{rcp}, consider the capacity factors of the units. (For convenience, the E_{rcp} notation is shortened to E_r since the same period p applies to all reactors and cycle c is immaterial to the current discussion.) As Widmer (57, 58) stated with elegant simplicity,

$$E = KLT \qquad (4.29)$$

where

E = electric energy production

K = rated electric capacity

L = average capacity factor

T = total length of time (i.e., including all outages)

Equation (4.29) actually serves to define L,

$$L \equiv \frac{E}{KT} \tag{4.30}$$

With the current discussion limited to any time period of length T' during which the unit (with a performance probability p) is never down for scheduled maintenance or refueling, a more meaningful parameter is the availability-based capacity factor L'

$$E = KL'T'p \tag{4.31}$$

or

$$L' \equiv \frac{E}{KT'p} \tag{4.32}$$

In words, L' represents the capacity factor the unit experienced during the period's pT' available hours that it was not down due to maintenance or refueling (T-T') or forced-outages [(1-p)T']. By comparing Equation (4.31) with Equation (2.24) integrated over the appropriate segment of the complete F_e,

$$K_r L'_r T' p_r = T' p_r \int_0^{K_r} F_e(P_e)\, dP_e \tag{4.33}$$

or,

$$L'_r = \frac{1}{K_r} \int_0^{K_r} F_e(P_e)\, dP_e \tag{4.34}$$

Hence, L_r' represents the average value of F_e in those segments placing demand on unit r.

Since the discussion is limited to the nuclear unit's upper (i.e., I-1) increments, define ℓ_r' as the availability-based increment capacity factor for unit r. Thus,

$$E_r = \sum^{I} E_{ri} = E_{r1} + k_r \ell_r' \, T' \, p_r \tag{4.35}$$

or

$$\ell_r' \equiv \frac{E_r - E_{r1}}{k_r \, T' \, p_r} = \alpha_r E_r - \beta_r \tag{4.36}$$

where

$$\alpha_r \equiv 1/k_r T' p_r \tag{4.37}$$

$$\beta_r \equiv E_{r1}/k_r T' p_r \tag{4.38}$$

Given each reactor's postulated production, E_{rcp}, (the p subset for all r resulting in N_p <u>in</u> <u>toto</u>), each ℓ_r' can be calculated and then ordered and plotted in decreasing magnitude. The resulting curve, whose abscissa is defined as P_r, is labeled the "average reactor shape" $\bar{F}_r$ in Figure 4.7c.

Using Figure 4.7a as an illustration, the segments of F_e used for loading each reactor's upper capacity k_r can be replotted separately as in Figure 4.7b. The average F_e for each reactor's upper increments is then ℓ_r'. Reordering the

Figure 4.7 Decomposition and Reordering of System Shape

Figure 4.7a System Shape Example

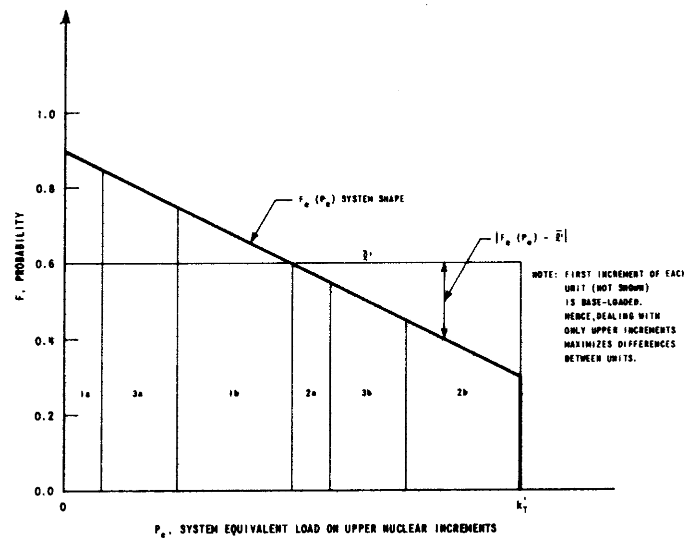


Figure 4.7b Individual Reactor Shapes

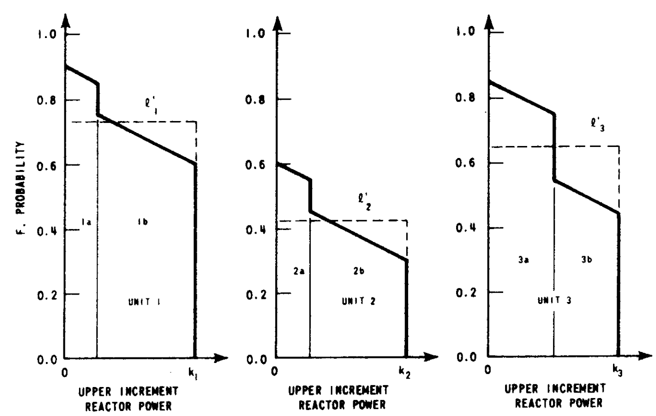
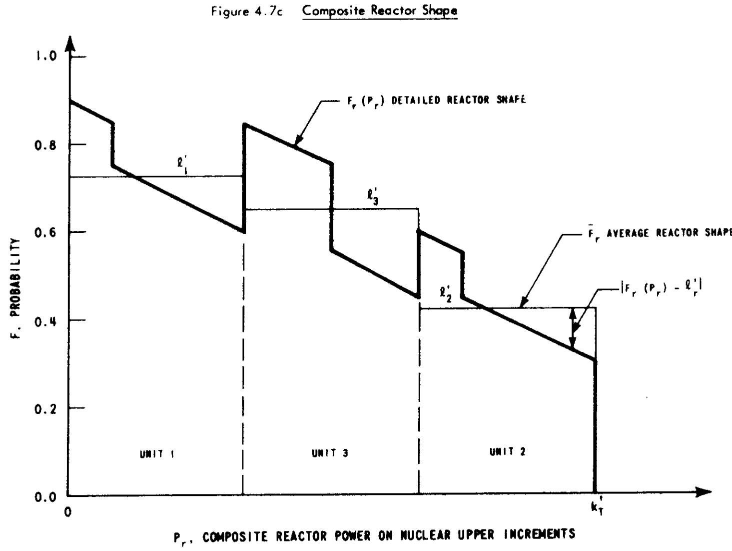


Figure 4.7c Composite Reactor Shape



reactor segments of Figure 4.7b with the largest ℓ_r' first, the detailed reactor shape F_r of Figure 4.7c results, defining the abscissa P_r, the composite reactor upper increment power. The system shape $F_e(P_e)$ of Figure 4.7a has merely been segmented and then reordered into the detailed reactor shape of $F_r(P_r)$ of Figure 4.7c on the basis of the average demand on each unit's upper increments ℓ_r'. Mathematically speaking, $F_r(P_r)$ is a one-to-one mapping of $F_e(P_e)$ since for every element of (point along) F_e at P_e, there exists a corresponding element of (point along) F_r at P_r. (However, in general, $P_e \neq P_r$.) Thus, the total area under the three shapes (i.e., for all k_T' MW of on-line nuclear upper increment capacity) is the same,

$$\int_0^{k_T'} F_e(P_e)\,dP_e = \int_0^{k_T'} F_r(P_r)\,dP_r = \int_0^{k_T'} \bar{F}_r(P_r)\,dP_r \tag{4.39}$$

The example in Figure 4.7 is, by definition, feasible since the detailed upper increment loading order resulting in each E_r (recall that each K_{r1} MW are base-loaded) and N_p in toto is clearly specified in Figure 4.7a. However, recall that in the SOM, only the F_e system shape to be satisfied and a postulated subset of E_r's are specified (not the detailed loading order). Hence, too little information is known to determine the detailed $F_r(P_r)$ as in Figure 4.7c. Nonetheless, the $\bar{F}_r$ average reactor shape can be determined for the postulated subset of E_r's. By applying Equation (4.36), each reactor's ℓ_r' can be calculated and placed in descending order, resulting in the desired $\bar{F}_r$.

The question of feasibility can then be stated as follows:

Given a postulated subset of E_r's (and the resulting "postulated" average reactor shape $\bar{F}_r$ on the upper increments), does there exist at least one intra-nuclear upper increment loading order such that the on-line reactors can indeed satisfy the given detailed system shape F_e?

A detailed loading order need not be determined, merely its existence established. If one exists, the postulated set of E_r represent a feasible means of operating the nuclear units; if none exists, then the postulated schedule is infeasible.

Two methods were considered for determining the existence of such a loading order: (1) area method and (2) variance method. The area method (see Appendix B), though rigorous (i.e., necessary and sufficient), involved an inordinate amount of computer data handling and storage and, therefore, was not implemented.

Utilizing the other (approximate) variance method, the shape constraint (derived in Section 4.2.4.2 and implemented per Section 4.2.4.3) is used to eliminate postulated subsets of E_r's which result in infeasible shapes by comparing a single parameter, the "variance" of the shape produced by the postulated E_r's against a similar parameter for the

SIM-calculated system shape F_e.

4.2.4.2 Mathematical Basis

To derive the shape constraint, consider the $F_e(P_e)$ system shape on the upper nuclear increments shown in Figure 4.7a. As a measure of the system shape, compare the shape with its mean $\bar{\ell}'$,

$$\bar{\ell}' \equiv \frac{1}{k_T'} \int_0^{k_T'} F_e(P_e)\,dP_e \tag{4.40}$$

Defining S^2 as the "variance" of the system shape compared with its mean,

$$S^2 \equiv \frac{1}{k_T'} \int_0^{k_T'} (F_e - \bar{\ell}')^2\,dP_e \tag{4.41}$$

For a known feasible solution, the S^2 variance will be the same whether integrated directly from 0 to k_T' (see Figure 4.7a), or first segmented into the respective detailed MW-by-MW reactor load shapes, reordered and then integrated (see Figure 4.7c).

$$S^2 = \frac{1}{k_T'} \int_0^{k_T'} (F_r - \bar{\ell}')^2\,dP_r \tag{4.42}$$

Breaking this integral into a sum over each of the R' on-line reactors,

$$S^2 = \frac{1}{k'_T} \sum^{R'} \int_{P_r^\circ}^{P_r^\circ + k_r} (F_r - \bar{\ell}')^2 \, dP_r \tag{4.43}$$

Adding and subtracting ℓ'_r inside the integrals of the summation,

$$S^2 = \frac{1}{k'_T} \sum^{R'} \int_{P_r^\circ}^{P_r^\circ + k_r} (F_r - \ell'_r + \ell'_r - \bar{\ell}')^2 \, dF_r \tag{4.44}$$

or

$$S^2 = \frac{1}{k'_T} \sum^{R'} \int_{P_r^\circ}^{P_r^\circ + k_r} \left[(F_r - \ell'_r)^2 + (\ell'_r - \bar{\ell}')^2 + 2(F_r - \ell'_r)(\ell'_r - \bar{\ell}') \right] dP_r \tag{4.45}$$

The third term inside the brackets vanishes since $(\ell'_r - \bar{\ell}')$ equals a constant and

$$\ell'_r \equiv \frac{1}{k_r} \int_{P_r^\circ}^{P_r^\circ + k_r} F_r(P_r) \, dP_r \tag{4.46}$$

for then

$$\int_{P_r^\circ}^{P_r^\circ + k_r} 2(F_r - \ell'_r)(\ell'_r - \bar{\ell}') \, dP_r = 2(\ell'_r - \bar{\ell}') \underbrace{\int_{P_r^\circ}^{P_r^\circ + k_r} (F_r - \ell'_r) \, dP_r}_{=0} \tag{4.47}$$

Thus,

$$S^2 = \frac{1}{k'_T}\sum^{R'}\int_{P_r^\circ}^{P_r^\circ + k_r}(F_r - \ell'_r)^2\,dP_r + \frac{1}{k'_T}\sum^{R'} k_r\,(\ell'_r - \bar{\ell}')^2 \qquad (4.48)$$

$$S^2 = V^2 + W^2 \qquad (4.49)$$

where V^2 = total internal variance of sub-segments of F_r for each reactor (i.e., requires detailed loading order)

W^2 = weighted sum of squares of reactor average versus system average of F_e (i.e., not dependent on MW-by-MW loading order, only average F_r over each k_r MW)

For a feasible E_r subset, V^2 must be non-negative since the integrand is squared. Therefore, if V^2 is negative for some other postulated production schedule when calculated by taking the difference in the calculated values of S^2 [Equation (4.41)] and W^2 [Equation (4.48)], that postulated schedule is clearly infeasible. Note that the converse is not true. If $(S^2 - W^2)$ is greater than or equal to zero, feasibility is not guaranteed. The following Section 4.2.4.3 discusses the practical implementation of this approximate constraint.

Typical values of S^2 calculated in this study are on the order of 0.01 to 0.03, while the theoretical maximum value is 0.25 for the pathological case of

$$F_e(P_e) = 1 \qquad 0 \leq P_e < 0.5\,k_T' \qquad (4.50)$$
$$= 0 \qquad 0.5\,k_T' < P_e \leq k_T'$$

For the infeasible example of Figure 4.6, $S^2 = 0.201$ while the reactor summation term W^2 has a value of 0.217. Thus, $V^2 = S^2 - W^2 = -0.016$, a highly infeasible value.

4.2.4.3 Practical Implementation

Practical implementation of Equation (4.49) as the SOM's shape constraint involves (1) determining the system shape F_e given the SIM's F^{min} and F^{max} (see Figure 4.1) and (2) incorporating a V^2_{REJ} rejection level on V^2 to allow flexibility in the model's handling of the constraint that $V^2 \equiv S^2 - W^2 \geq 0$.

The practical definition of F_e is the demand curve used for loading each MW according to Equation (2.24). For the first MW of the nuclear upper increments, the deconvolve-load-convolve sequence of the multiple increment algorithm of Section 3.3.2.2 must be applied to F^{min}. Since the identity of the first nuclear upper increment to be loaded is arbitrary at this point, a hypothetical unit with the average values of p_r and K_{r1} which gives the same average MW of outage would appear to be useful,

$$\overline{p_r^{min}} = \sum^{R'} p_r K_{r1} \Big/ \sum^{R'} K_{r1} \tag{4.51}$$

$$\overline{K_{r1}} = \frac{1}{R'} \sum^{R'} K_{r1} \tag{4.52}$$

Deconvolving this unit per Equation (3.55),

$$F_{\overline{r1}}^{wo}(P_e) = \frac{1}{\overline{p_r^{min}}} \left[F^{min}(P_e) - (1 - \overline{p_r^{min}})\, F_{\overline{r1}}^{wo}(P_e - \overline{K_{r1}}) \right] \tag{4.53}$$

This $F_{\overline{rI}}^{wo}$ is the average curve used to load the first MW of the nuclear upper increments. In a similar manner, an $F_{\overline{rI}}^{wo}$ can be determined from F^{max} that estimates the curve used for loading the last MW of the nuclear upper increments:

$$\overline{p_r^{max}} = \sum^{R'} p_r K_{rI} \Big/ \sum^{R'} K_{rI} \tag{4.54}$$

$$\overline{K_{rI}} = \frac{1}{R'} \sum^{R'} K_{rI} \tag{4.55}$$

and

$$F_{\overline{rI}}^{wo}(P_e) = \frac{1}{\overline{p_r^{max}}} \left[F^{max}(P_e) - (1 - \overline{p_r^{max}})\, F_{\overline{rI}}^{wo}(P_e - \overline{K_{rI}}) \right] \tag{4.56}$$

Figure 4.8 presents $F^{wo}_{\overline{r1}}$ and $F^{wo}_{\overline{rI}}$ for the F^{min} and F^{max} of Figure 4.1. Since each F^{wo} is equal to F_e at a particular point of application,

Point A: $$F^{wo}_{\overline{r1}}(0) = F_e(0) \tag{4.57}$$

Point B: $$F^{wo}_{\overline{rI}}(k'_T) = F_e(k'_T) \tag{4.58}$$

then $F_e(P_e)$ must trace a path connecting points A and B of Figure 4.8. Thus, F_e can be simply approximated by interpolation over the range $0 \leq P_e \leq k'_T$,

$$F_e(P_e) = \left(1 - \frac{P_e}{k'_T}\right) F^{wo}_{\overline{r1}}(P_e) + \frac{P_e}{k'_T} F^{wo}_{\overline{rI}}(P_e) \tag{4.59}$$

With F_e approximated, $\overline{\ell}'$ and S^2 are easily calculated (see Figure 4.9),

$$\overline{\ell'} \equiv \frac{1}{k'_T} \int_0^{k'_T} F_e(P_e)\, dP_e \tag{4.40}$$

$$S^2 \equiv \frac{1}{k'_T} \int_0^{k'_T} \left(F_e - \overline{\ell'}\right)^2 dP_e \tag{4.41}$$

With $\overline{\ell}'$ and S^2 pre-calculated by the SOM before the iterative optimization procedure begins (see Section 4.4), Equation (4.49) is implemented as the shape constraint on

Figure 4.8

Approximation of F_e

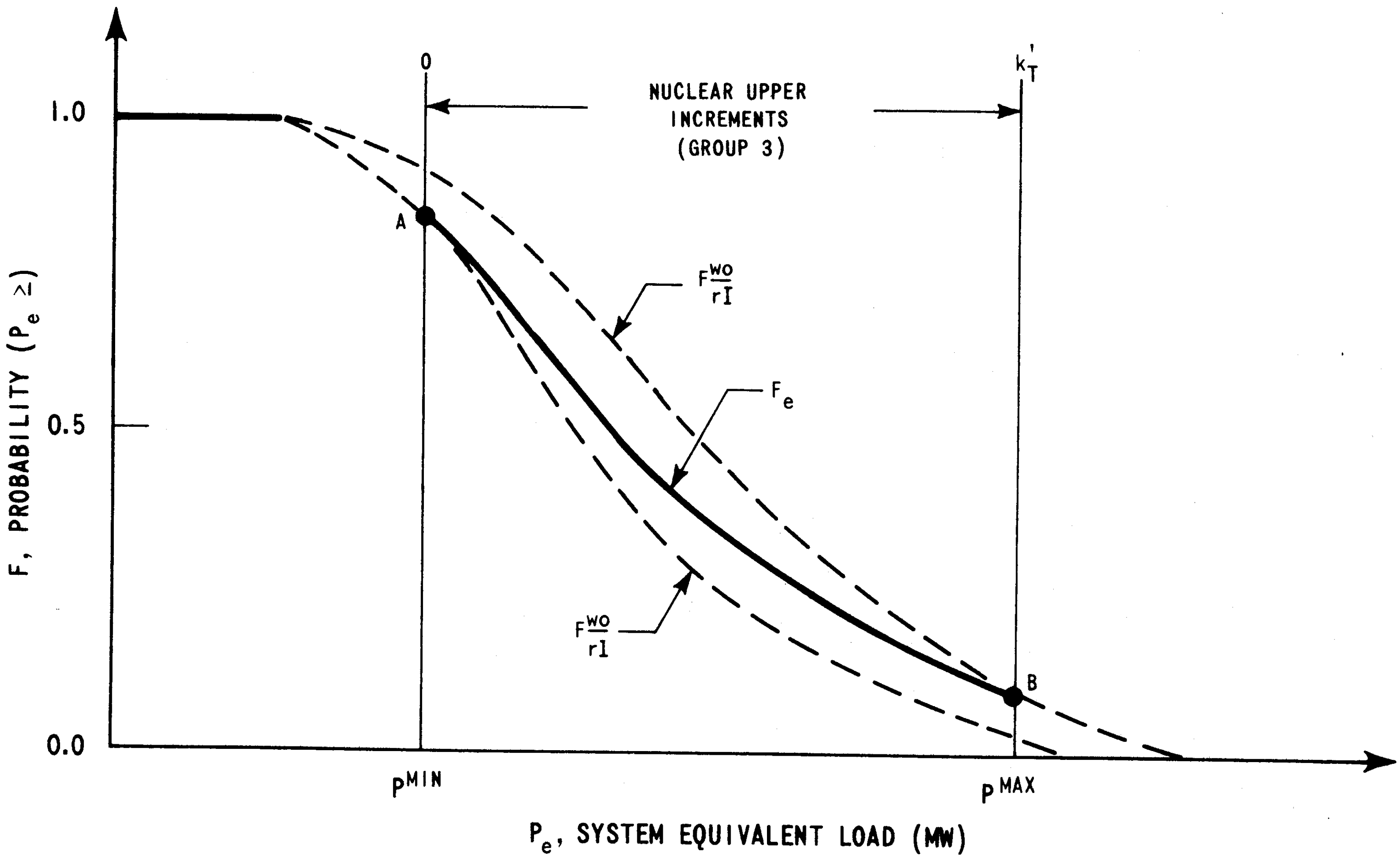
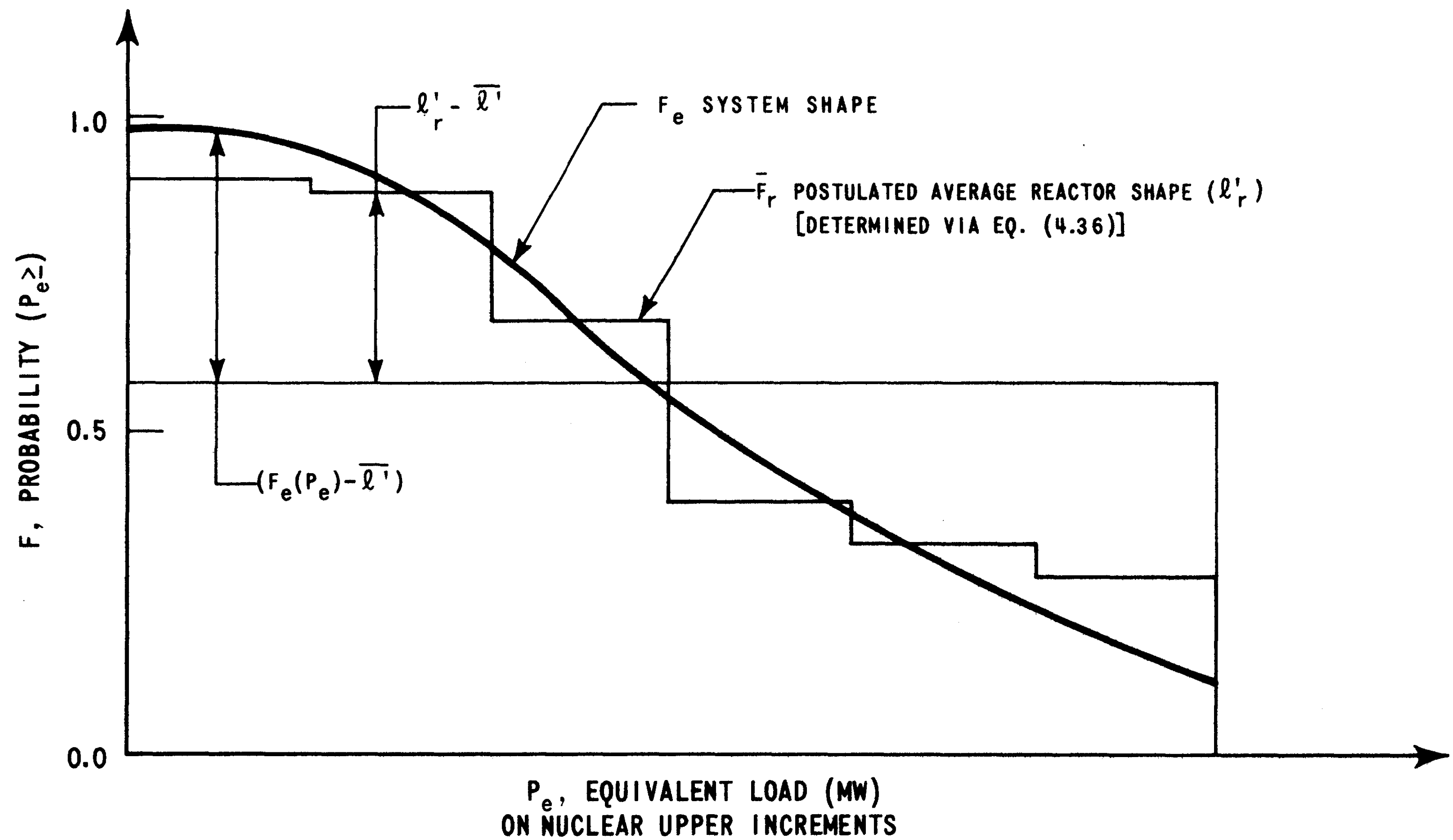


Figure 4.9

Comparison of System Shape and Postulated Average Reactor Shape



6253-37

each iteration's postulated set of E_r. This involves (1) using Equation (4.36) to calculate ℓ'_r for each postulated E_r,

$$\ell'_r = \alpha_r E_r - \beta_r \tag{4.36}$$

(2) calculating W^2 from the resulting ℓ'_r (see Figure 4.9),

$$W^2 \equiv \frac{1}{k'_T} \sum^{R'} k_r (\ell'_r - \bar{\ell}')^2 \tag{4.48}$$

and (3) testing the resultant V^2 ($\equiv S^2 - W^2$) versus a V^2_{REJ} rejection level designed to establish feasibility, not merely infeasibility, as discussed below.

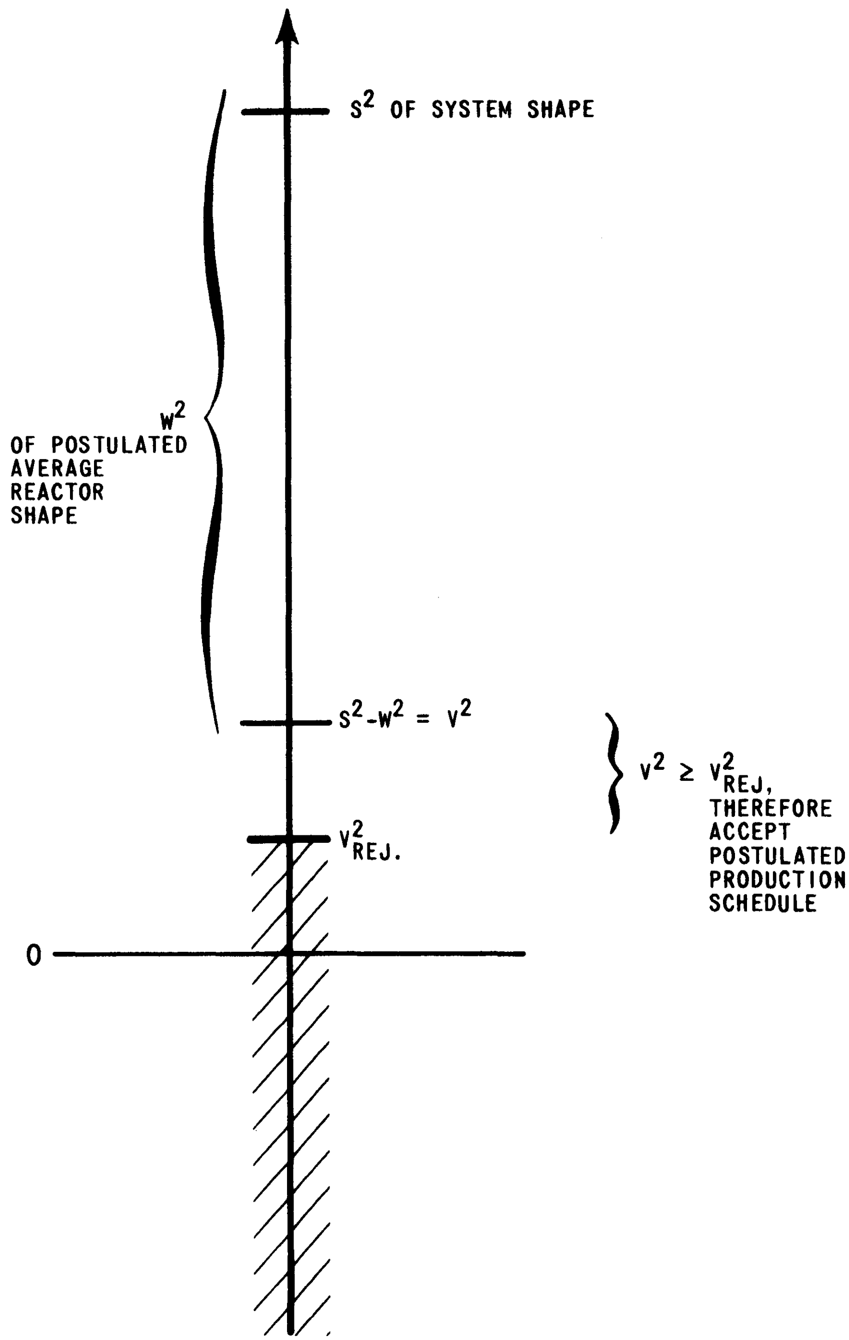
Rearranging Equation (4.49),

$$V^2 = S^2 - W^2 \tag{4.60}$$

This is the convenient form of Equation (4.49) since determining V^2 by difference does not required a detailed loading order (which may not even exist). For $V^2 < 0$, the postulated production schedule is infeasible; $V^2 \sim 0$, may be infeasible; $V^2 >> 0$, almost certainly feasible. To implement the constraint, a V^2_{REJ} rejection level is introduced such that if $V^2 \leq V^2_{REJ}$, the postulated schedule is rejected as probably infeasible. Figure 4.10 presents a visual interpretation of the implementation.

Figure 4.10

Implementation of V^2_{Rej} Period Shape Test



Note the flexibility of a model allowing V^2_{REJ} as an input parameter:

(1) If $V^2_{REJ} = 0$, Equation (4.48) holds directly with $V^2 > 0$ being required, or (2) If $V^2_{REJ} \leq -0.25$, the shape constraint is effectively nullified. To be accepted W^2 must be $\leq S^2 - V^2_{REJ} = S^2 + 0.25$. Theoretically, $(W^2)^{max} = 0.25$ (see Section 4.2.4.2) and $(S^2)^{min} = 0$. Thus, W^2 is always $\leq S^2 + 0.25$ and, hence, always accepted.

To summarize the complete formulation of the shape constraint for period p, the E_r notation returns to E_{rcp}. Hence, a postulated period production schedule is not rejected as infeasible if

$$W_p^2 \leq S_p^2 - V_{REJ}^2 \tag{4.61}$$

or

$$\frac{\sum_{r=1}^{R'_p} k_r \left(\alpha_{rp} E_{rcp} - \beta_{rp} - \bar{\ell}'_p\right)^2}{\sum_{r=1}^{R'_p} k_r} \leq S_p^2 - V_{REJ}^2 \tag{4.62}$$

Note the existence of second-order terms (E^2_{rcp}) as was indicated in Equation (4.24).

4.3 Mathematical Statement of Optimization Problem

Summarizing the elements of the optimization problem formulated in Section 4.2, the problem can be stated succinctly as,

$$\text{minimize} \quad \overline{TC}(\mathcal{E}) = \sum^{R} \overline{TC}_r(\{E_{rc}\}_r) \tag{4.1}$$

or equivalently

$$\text{minimize} \quad \Sigma_{EST}^{t+1}(\mathcal{E}) = \sum^{R}\sum^{C} \int_{E_{rc}^{t}}^{E_{rc}^{t+1}} \lambda_{rc}^{t}\, dE_{rc} \tag{4.12}$$

such that the following period constraints are met for

System Production:

$$\sum^{R} E_{rcp} = N_p \quad \text{for all } p \tag{4.13}$$

Reactor Production:

$$E_{rcp}^{min} \leq E_{rcp} \leq E_{rcp}^{max} \quad \text{for all } r \text{ and } p \tag{4.14}$$

and Shape:

$$\frac{\sum_{r=1}^{R'_p} k_r(\alpha_{rp}E_{rcp} - \beta_{rp} - \overline{\ell'_p})^2}{\sum_{r=1}^{R'_p} k_r} \leq S_p^2 - V_{REJ}^2 \tag{4.62}$$

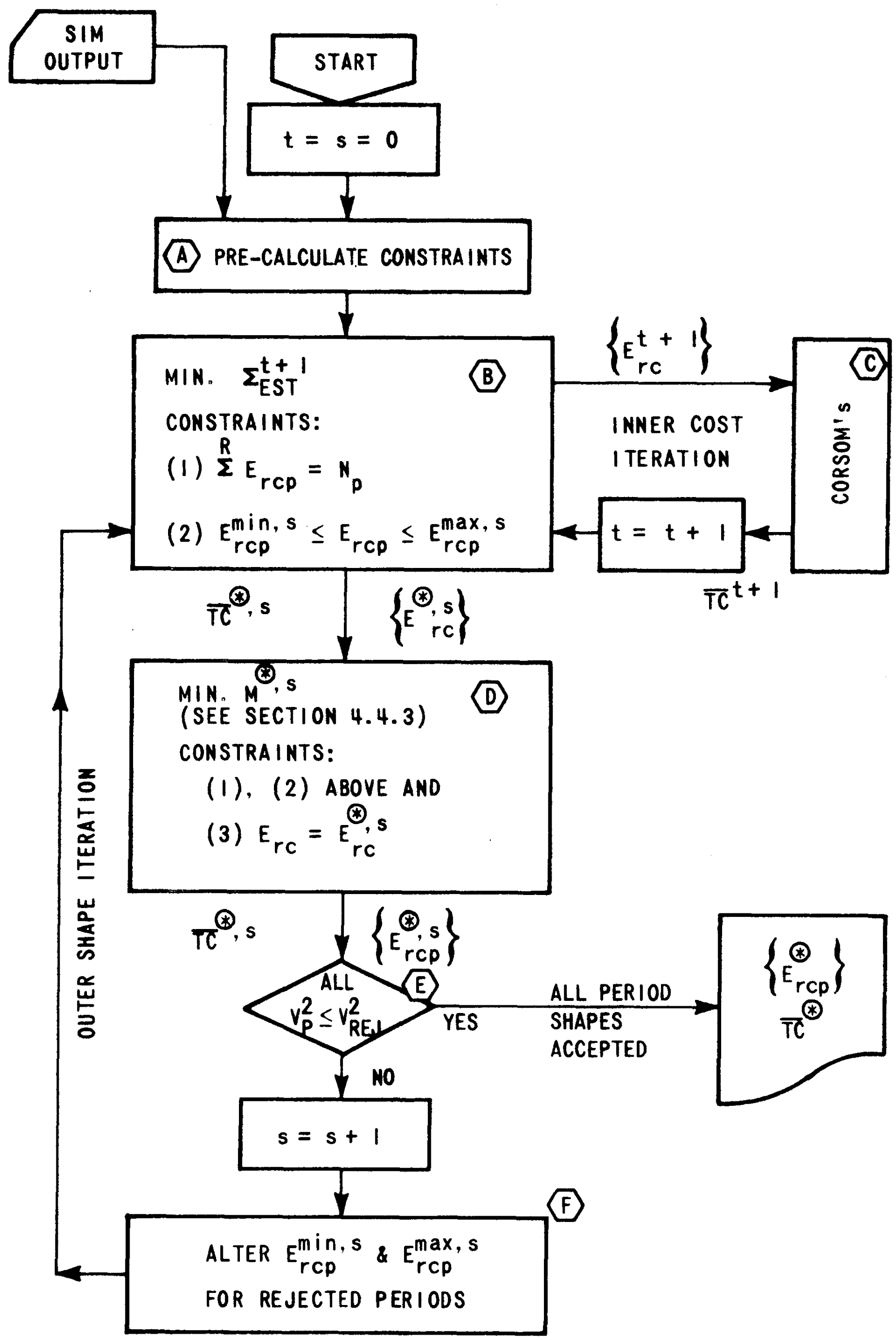
4.4 Method of Optimization

In choosing a method of optimization, the size of the problem itself must be considered. Suppose a utility with eight reactors desires to optimize the system refueling strategy over the next six years using time periods two weeks long. Then, there will be Z ~ 150 time periods, each of which has two constraints [Equations (4.13) and (4.62), one of which is non-linear]. Each of the R·Z = 1200 optimization variables in $\mathcal{E}$ has a lower and an upper limit (2400 more constraints). The final total: 1200 variables to be optimized subject to 2700 constraints--a very large optimization problem, particularly if solved in an iterative fashion.

The schematic diagram of a two-stage iterative optimization procedure is shown in Figure 4.11. The optimization is initiated by the precalculation of constraint limits (Block A) based on the output supplied by the SIM. Then for each outer shape iteration, s, the inner cost iteration loop, consisting of the network program without any shape constraints (Block B) and the CORSOM's (Block C), operates within the remaining constraints. The inner loop's output is a complete set of optimized reactor-cycle energies, $E_{rc}^{*,s}$, which results in the minimum nuclear fuel revenue requirement for the system, $\overline{TC}^{*,s}$. In the second stage, the network program of Block D is used to apportion each reactor-cycle energy in this set among the various reactor-periods

Figure 4.11

SOM Optimization Scheme



making up a reactor-cycle. The objective is to minimize the likelihood that the shape constraint for any period will be violated, $M^{*,s}$. Then, Block E compares the "variance", V_p^2, for each period of the resulting set of reactor-period energy productions, $\{E_{rcp}^{*,s}\}$, with the preselected shape rejection criterion, V_{REJ}^2. If the shape of any period violates the criterion, another outer shape iteration is begun by decreasing the range of the permissible reactor-period energy productions for all reactors supplying energy in each rejected period. When all period shapes are accepted, the optimization of the SOM is complete. The resulting optimized (i.e., minimized) nuclear fuel revenue requirement, $\overline{TC}^{\bullet}$, is combined with the non-nuclear operating revenue requirements to produce the system's total optimized operating revenue requirement (as shown in Figure 2.22) for the particular alternative refueling and maintenance strategy under investigation.

While many iterative, non-linear optimization techniques seek the global optimum by operating within the feasible $\mathcal{E}$ hyperspace, this two-stage technique approaches the optimum from without, i.e., from the infeasible region. Consequently, instead of each iteration decreasing the objective function, the objective function increases as feasibility is approached, giving a lower bound for the more feasible solution at the next iteration (see Section 5.6.2).

4.4.1 Concept of Nuclear Energy Supply Network

Since the only non-linear constraint [Equation (4.62)] is not considered explicitly in either sub-optimization of Figure 4.11, the remaining constraints are linear. In fact, because the resulting sub-optimizations deal with a single commodity (nuclear energy production) in a strict one-to-one (reactor) supply and (customer) demand sense, the constraints form a nuclear energy supply network. Figure 4.12 presents such a network configuration for a 3 reactor,24 period (month) example. (Numbers are displayed for the nuclear potentials to emphasize the fact that these are fixed constraints throughout all of the iterations for a particular refueling and maintenance strategy.) Nuclear energy is allocated (supplied) to each reactor-cycle. Within each cycle, the energy is allocated to the pertinent periods so as to satisfy the system nuclear potentials (demanded). The sum of any column must equal the energy supplied (or extracted) during that particular reactor-cycle while the sum of any row must equal its required nuclear potential [Equation (4.13)]. The range of each E_{rcp} is also constrained via Equation (4.14) (presented in Table 4.1 but not shown explicitly on Figure 4.12) leading to the term "capacitated" network.

Each of the sub-optimizations in the following sections thus seeks to determine that $\mathcal{E}$ set of E_{rcp} that satisfies these network constraints, yet minimizes its repective objective function.

Figure 4.12

Sample Network Configuration

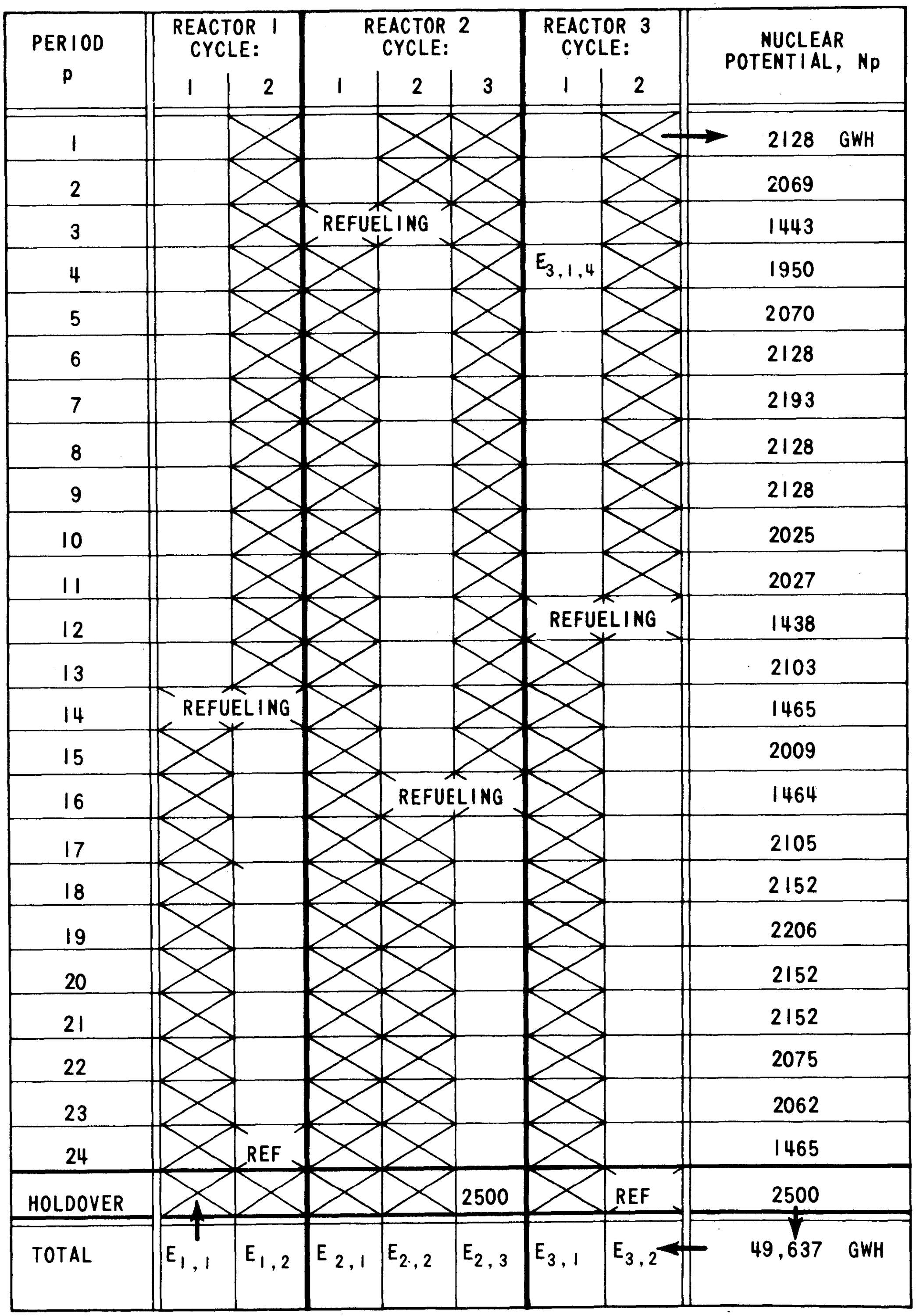


Table 4.1

Reactor Production Limits for 3 Reactor, 24 Period Example

Period p	Reactor 1 E^{min}_{1cp}	Reactor 1 E^{max}_{1cp}	Reactor 2 E^{min}_{2cp}	Reactor 2 E^{max}_{2cp}	Reactor 3 E^{min}_{3cp}	Reactor 3 E^{max}_{3cp}
1	669	762	629	722	669	762
2	635	760	596	720	635	760
3	687	756	0	0	687	756
4	577	747	540	707	577	747
5	636	760	596	720	636	760
6	669	762	629	722	669	762
7	714	763	674	723	714	763
8	669	762	629	722	669	762
9	669	762	629	722	669	762
10	616	755	577	714	616	755
11	610	759	571	718	610	759
12	718	760	678	720	0	0
13	656	761	617	721	656	761
14	0	0	703	722	743	763
15	610	752	571	712	610	752
16	706	758	0	0	706	758
17	657	761	617	721	657	761
18	686	762	646	722	686	762
19	724	763	684	723	724	763
20	686	762	646	722	686	762
21	686	762	646	722	686	762
22	643	758	604	718	643	758
23	632	759	593	719	632	759
24	0	0	703	722	743	763

All E_{rcp} in GWH

4.4.2 <u>Inner Iteration on Nuclear Cost</u>

Each inner cost iteration of Figure 4.11 solves the following sub-optimization problem:

$$\text{minimize} \quad \Sigma^{t+1}_{EST}(\mathcal{E}^{t+1}) = \sum^{R} \sum^{C} \int_{E^{t}_{rc}}^{E^{t+1}_{rc}} \lambda^{t}_{rc} \, dE_{rc} \qquad (4.12)$$

such that

$$\sum^{R} E_{rcp} = N_p \quad \text{for all } p \qquad (4.13)$$

and

$$E^{min,s}_{rcp} \leq E_{rcp} \leq E^{max,s}_{rcp} \qquad (4.14)$$

Inner iterations continue until $\mathcal{E}^{t+1}$ converges to $\mathcal{E}^{*,s}$. Critical to the minimization of Equation (4.12) is the representation of the incremental cost curve λ^{t}_{rc} as a function of E^{t}_{rc}. Figure 4.13 presents a typical true incremental cost curve and two approximations to it:

(1) linear approximation,

$$\lambda^{t}_{rc} = a^{t}_{rc} E^{t}_{rc} + b^{t}_{rc} \qquad (4.63)$$

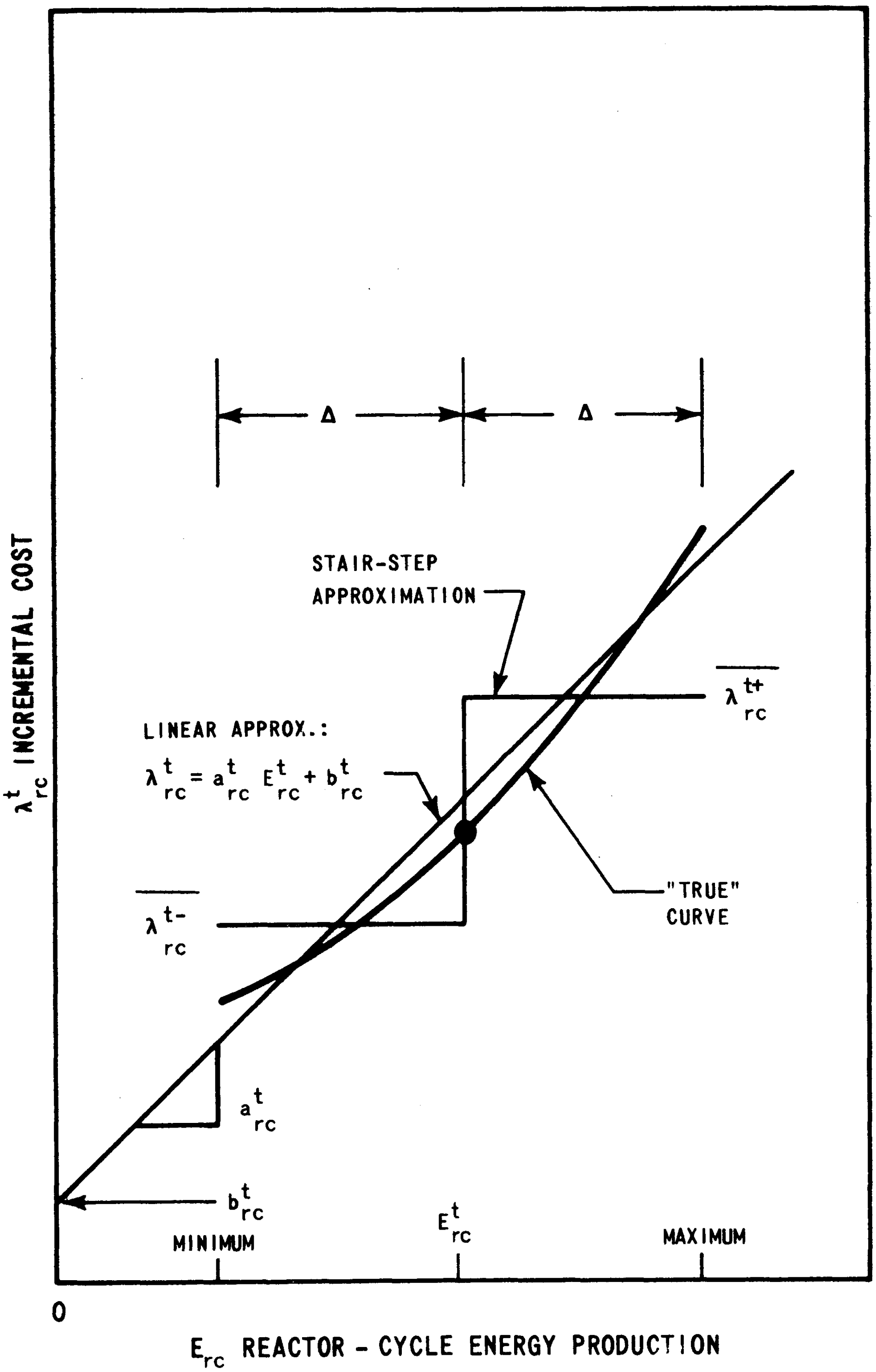
and (2) a "stair-step" approximation having the same areas as the Δ GWH segments of the true curve,

$$\begin{aligned} \lambda^{t}_{rc} &= \overline{\lambda^{t-}_{rc}} & E^{t}_{rc} - \Delta \leq E_{rc} < E^{t}_{rc} \\ &= \overline{\lambda^{t+}_{rc}} & E_{rc} < E_{rc} \leq E^{t}_{rc} + \Delta \end{aligned} \qquad (4.64)$$

Figure 4.13

Typical Incremental Cost Curve and Approximations

6253-43



Performing the integration of Equation (4.12), the linear approximation results in a quadratic programming (QP) problem,

$$\text{minimize } \Sigma_{EST}^{t+1} = \sum^{R}\sum^{C}\left\{\frac{a_{rc}^{t}}{2}\left[\left(E_{rc}^{t+1}\right)^{2}-\left(E_{rc}^{t}\right)^{2}\right]+b_{rc}^{t}\left(E_{rc}^{t+1}-E_{rc}^{t}\right)\right\} \tag{4.65}$$

subject to the capacitated supply network constraints of Equations (4.13) and (4.14).

On the other hand, the stair-step approximation leads to a linear programming (LP) problem utilizing the method of "convex combinations" (54) of E_{rc}^{t} and E_{rc}^{t+1}. In fact, since the model's context is a supply network and the objective function is linear, this special LP problem reduces to a network programming (NP) problem,

$$\text{minimize } \Sigma_{EST}^{t+1} = \sum^{R}\sum^{C} \overline{\lambda_{rc}^{t\pm}}\cdot\left(E_{rc}^{t+1}-E_{rc}^{t}\right) \tag{4.66}$$

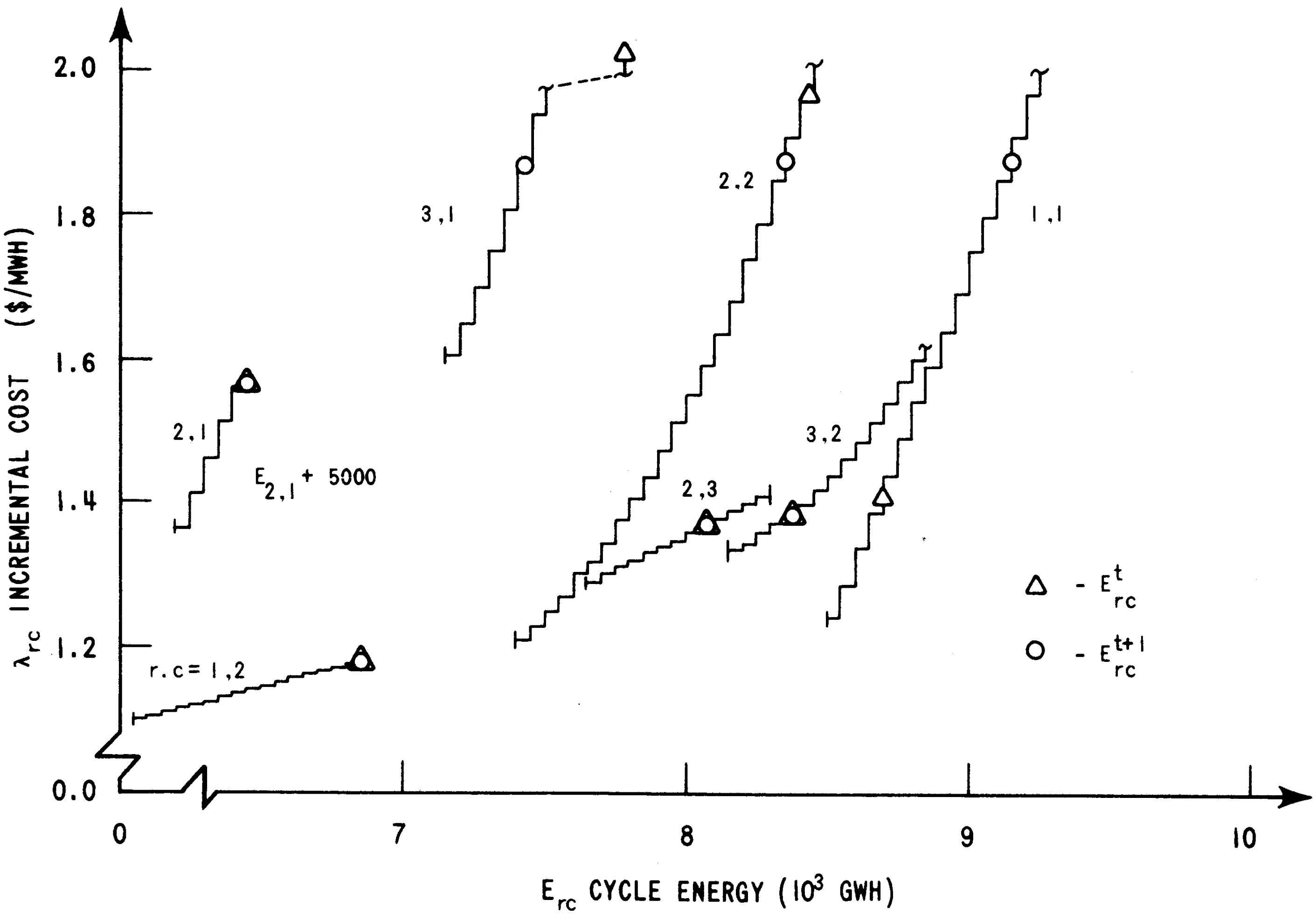
Considering only the accuracy of the underlying approximations, a QP code package ought to be favored over a NP package for achieving the sub-optimization. However, even the example optimization problem of Section 4.4 (with 1200 primary variables subject to 2700 constraints), is too large for a typical generalized QP package (6) which permits only 1100 variables (including slack variables) and 800 constraints.

Investigating the stair-step approximation further by decreasing Δ and increasing the number of steps, Equation (4.66) becomes a "piecewise-linear" (54) NP problem and the second approximation approaches the first with regard to accuracy. [This piecewise-linearization refers to $\overline{TC}_r$ and is made possible by the separability of the equivalent objective function Equation (4.12)]. Furthermore, specialized NP packages tailored to capacitated networks (27, 45) are available that can readily handle up to 10,000 primary capacitated variables and up to 5000 system production-type constraints (see Appendix G). Such capabilities easily permit the additional variables introduced during the piecewise-linearization.

To illustrate a single inner iteration consider the 3 reactor, 24 period example of Figure 4.12 and Table 4.1. Figure 4.14 presents a hypothetical set of incremental cost curves returned to the SOM at the end of the previous iteration. These are taken with respect to changes about the indicated E^t_{rc}. Also indicated is the next trial set E^{t+1}_{rc} resulting from the single inner optimization. Note that (1) the NP program seeks to establish equal nuclear incremental costs (see Section 2.4.2) among the reactor-cycles that compete for the nuclear potential (e.g., $\lambda_{1,1} = \lambda_{2,2} = \lambda_{3,1}$) and (2) the total increase in cycle energies in a given trial equals the total decrease in cycle energies in that trial since the total nuclear

Figure 4.14

Hypothetical Set of Incremental Cost Curves



6253-44

potential, of course, does not change from iteration to iteration. Figure 4.15 presents the complete period-by-period reactor production schedule for t+1.

4.4.3 Outer Iteration on Shape Misfit Potential

As outlined in Section 4.4 and Figure 4.11, inner cost iterations continue until the $\{E_{rc}^{t,s}\}$ converges to $\{E_{rc}^{*,s}\}$ at which time the outer iteration commences. The objective function $M^{*,s}$ of the outer shape iteration is based on the key fact that if all $\ell'_{rp} = \bar{\ell}'_p$, then $W_p^2 = 0$ [from Equation (4.48)]. Hence, $V_p^2 = S_p^2$ and consequently, all periods are feasible since $V_p^2 >> V_{REJ}^2$ (see Figure 4.10). Furthermore, any deviation of ℓ'_{rp} from $\bar{\ell}'_p$ increases the likelihood of ultimate period rejection.

Each outer shape iteration of Figure 4.11 thus solves the following sub-optimization problem:

$$\text{minimize} \quad M^{*,s}(\mathcal{E}) \equiv \sum^{R} \sum^{Z} \int_{E_{rcp} \text{ corres. to } \bar{\ell}'_p}^{E_{rcp}} m_{rp}(E_{rcp})\, dE_{rcp} \tag{4.67}$$

such that

$$\sum^{R} E_{rcp} = N_p \quad \text{for all } p \tag{4.13}$$

$$E_{rcp}^{min,s} \leq E_{rcp} \leq E_{rcp}^{max,s} \quad \text{for all } r \text{ and } p \tag{4.14}$$

$$E_{rc} \equiv \sum^{p \text{ in } c} E_{rcp} = E_{rc}^{*,s} \quad \text{for all } r \text{ and } c \tag{4.68}$$

Figure 4.15

Sample Reactor Production Schedule

PERIOD p	REACTOR 1 CYCLE: 1	REACTOR 1 CYCLE: 2	REACTOR 2 CYCLE: 1	REACTOR 2 CYCLE: 2	REACTOR 2 CYCLE: 3	REACTOR 3 CYCLE: 1	REACTOR 3 CYCLE: 2	NUCLEAR POTENTIAL, Np
1	715		722			691		2128 GWH
2	697		720			652		2069
3	722		REFUELING			721		1443
4	661			707		582		1950
5	697			720		653		2070
6	715			722		691		2128
7	738			723		732		2193
8	715			722		691		2128
9	715			722		691		2128
10	685			714		626		2025
11	684			672		671		2027
12	738			700		REFUELING		1438
13	668			674			761	2103
14	REFUELING			703			762	1465
15		752		571			686	2009
16		758		REFUELING			706	1464
17		761			687		657	2105
18		762			704		686	2152
19		763			719		724	2206
20		762			704		686	2152
21		762			704		686	2152
22		758			674		643	2075
23		759			671		632	2062
24		REF			722		743	1465
HOLDOVER					2500		REF	2500
TOTAL	9150	6837	1442	8350	8085	7401	8372	49,637 GWH

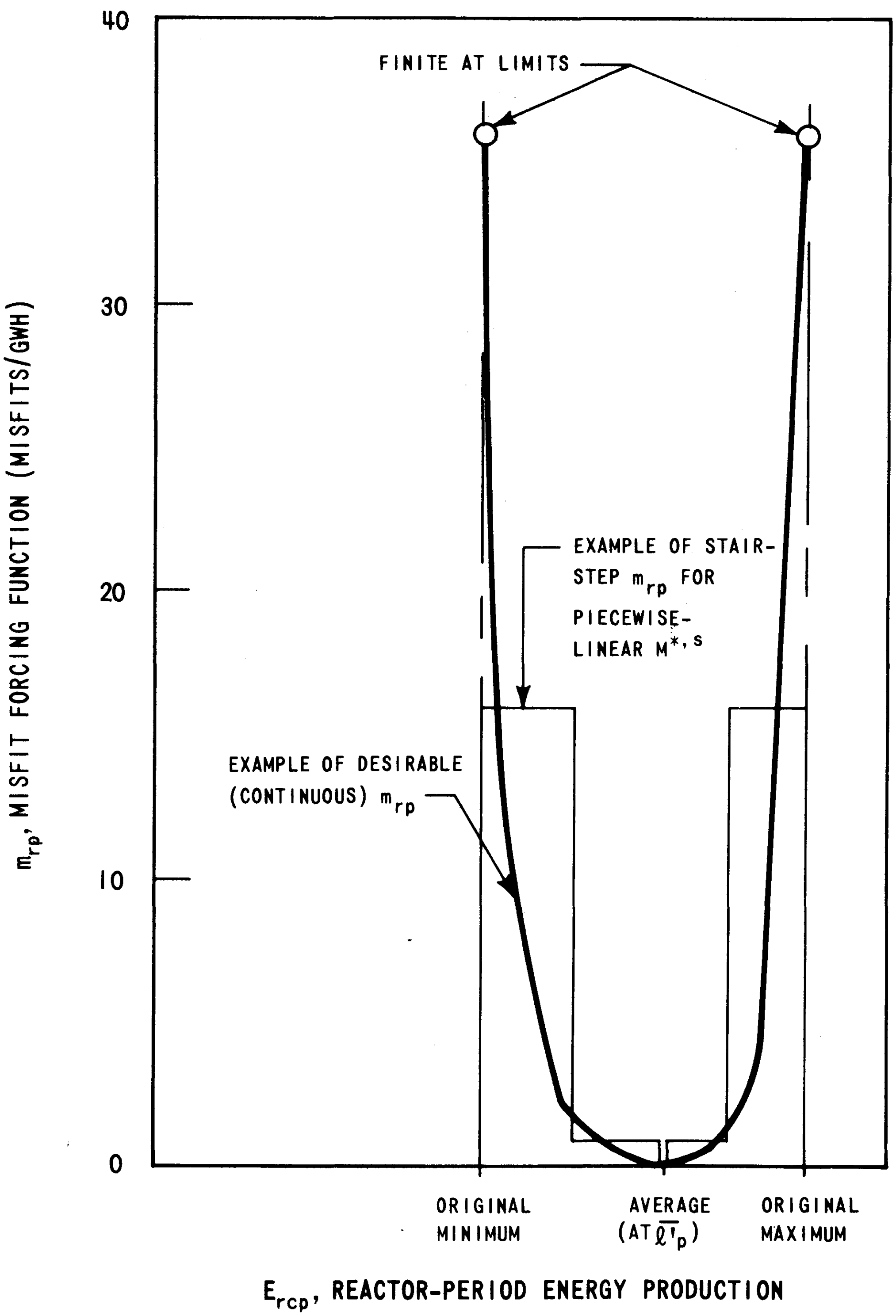
The $M^{*,s}$ system misfit potential, defined by Equation (4.67), merely represents a mathematical "gimmick" designed to force $\mathcal{E}$ (i.e., the set of all E_{rcp}) into the feasible region, minimizing the number of period shapes later rejected due to misfitting shapes [Equation (4.62)]. The all-important misfit forcing function, m_{rp}, though arbitrary, should possess the properties indicated in Figure 4.16. At E_{rcp} corresponding to $\ell'_{rp} = \overline{\ell}'_p$, $m_{rp} = 0$; for deviations in either direction from this E_{rcp}, m_{rp} increases rapidly; and for the end points $E_{rcp}^{min,0}$ and $E_{rcp}^{max,0}$, which are especially vulnerable to rejection, m_{rp} should still be finite since the extremums are not unacceptable <u>per se</u>. The optimization of Equations (4.67) to (4.68) thus attempts to force each E_{rcp} to the bottom of the resulting "trough" of m_{rp} subject to the various constraints, such as fixed reactor-cycle energy.

Since $M^{*,s}$ is defined (via the m_{rp}) to be separable and convex, the methods of piecewise-linearization and convex combinations can again be applied as was done for the inner cost iterations of Section 4.4.2. Note that given the typical, but arbitrary stair-step m_{rp} curve of Figure 4.16, the linearized $M^{*,s}$ optimization of the capacitated supply network is <u>not</u> iterative in nature--the complete optimization of $\mathcal{E}^{*,s}$ occurs in one pass through the NP package. The actual "iteration" involves checking resulting period shape acceptabilities and appropriately altering the reactor production constraints for the next

6253-45

Figure 4.16

Misfit Forcing Function versus Reactor-Period Production



set of inner cost iterations (see Figure 4.11).

Looking at each optimized period in turn (the notation is shortened to E_r for convenience), the variance test of Equation (4.62) is applied. If $S^2-W^2 \geq V^2_{REJ}$, the period is accepted and processing moves on to the next period. If the test fails, then V^2 ($\equiv S^2-W^2$) $\leq V^2_{REJ}$. Defining σ as following measure of infeasibility,

$$\sigma \equiv \sqrt{V^2_{REJ}-V^2} \tag{4.69}$$

σ represents the average change of each ℓ'_r (toward $\bar{\ell}'$) required before the postulated production shape would pass the test. If a fraction γ of this average reduction is applied to each reactor's limiting values of ℓ'_r (see Figure 4.17), then from Equation (4.35),

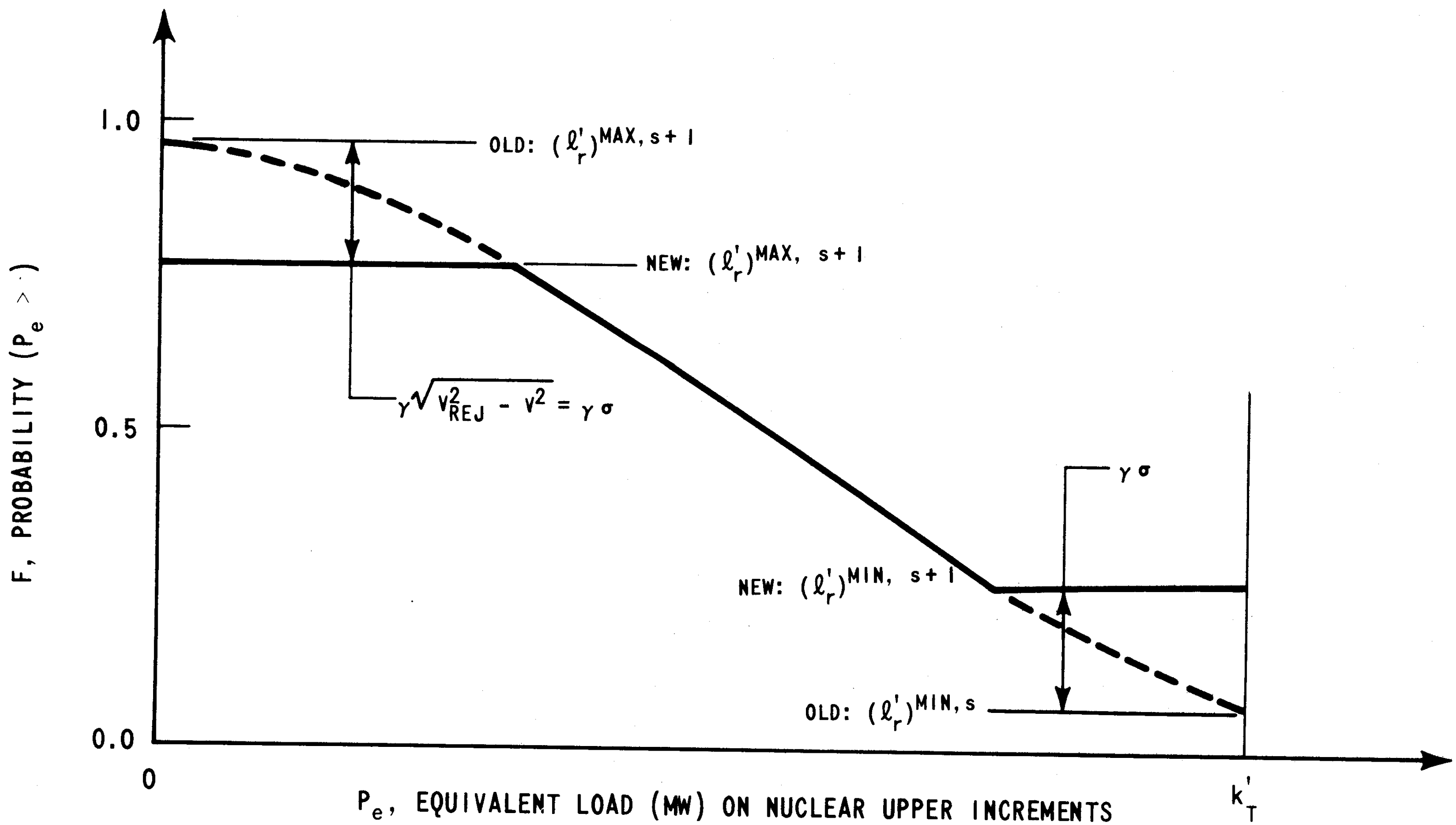
$$E_r^{min,s+1} = E_{r1} + k_r T' p_r \left[(\ell'_r)^{min,s} + \gamma\sigma \right] \tag{4.70}$$

$$E_r^{max,s+1} = E_{r1} + k_r T' p_r \left[(\ell'_r)^{max,s} - \gamma\sigma \right] \tag{4.71}$$

When all periods have been tested thusly, and/or the appropriate limits altered, the outer shape iteration terminates and inner cost iterations begin on the new subproblem. The shape iteration which results in all period shapes being accepted, terminates the entire optimization at the feasible global optimum $\mathcal{E}^{*}$ and minimum total

Figure 4.17

"Squeezing" Permissible Reactor Production Shapes



6253-46

cost $\overline{TC}^{\circledast}$. [Note that if $V^2_{REJ} \leq -0.25$ (see Section 4.2.4.3), all period shapes are acceptable regardless of feasibility. Hence, $\mathcal{E}^{*,0} = \mathcal{E}^{\circledast}$ and $\overline{TC}^{*,0} = \overline{TC}^{\circledast}$ immediately.]

4.5 Completion of Supervisory Task

With the optimization task completed, the resulting feasible optimum $\overline{TC}^{\circledast}$ represents the total revenue requirement for nuclear fuel RR_N. By present-valuing all of the other period expenditures (received as input from the SIM) according to Equation (2.17),

$$ORR = RR_N + \sum^{Z} \frac{1}{(1+x)^{\bar{t}_p}} \left(X_{F_p} + X_{S_p} + X_{U_p}\right) \qquad (4.72)$$

The ORR operating revenue requirement is appropriately stored for later comparison with that of other possible alternative strategies. With the completion of this task, processing of the particular alternative strategy is complete. And with completion of the last alternative strategy, selection of the minimum ORR cost strategy becomes possible (see Section 2.5.1).

4.6 SYSOPT, A Computerized SYStem OPTimization Model

SYSOPT, a 2100 card Fortran IV version of the SYStem OPTimization Model is detailed in Appendix F. SYSOPT is link-edited with the Out of Kilter Network Program (45) which represents an additional 1200 cards in Fortran IV and Assembler Language. Out of Kilter is detailed in

Appendix G. This section merely summarizes the capabilities of the current combined version of SYSOPT.

The limitations of the current version of SYSOPT, though easily altered, are as follows:

(1) up to 15 reactors,

(2) up to 15 cycles per reactor within the horizon,

(3) up to 3 cycles per reactor beyond the horizon,

(4) no limit on number of strategies per computer run and

(5) up to 100 periods per strategy.

Input data for each strategy includes:

(1) Present value rate,

(2) Various convergence criteria,

(3) Various Δ for linearizing λ_{rc} of inner iterations,

(4) Maximum total number of inner iterations to be permitted,

(5) Number of linearized segments in m_{rp} (up to 10) and

(6) V^2_{REJ} and γ of the shape iteration.

Input data supplied manually for each reactor includes:

(1) Optional initial estimates of λ^*_{rc} or E^*_{rc},

(2) Holdover energy at end of planning horizon, $E_{r,C,Z+1}$ and

(3) Cycle energies and refueling dates beyond planning horizon.

The large volume of SYSINT output required by SYSOPT may be passed either on disk, magnetic tape or punched cards.

As for typical running times on an IBM 370 model 155 computer (MVT environment), the cases presented in Chapter 5 for a hypothetical six reactor utility required only 9 CPU seconds per inner iteration (exclusive of time spent in CORSOM's) for strategies 72 periods long and totaling 30 reactor-cycles. The SYSOPT code itself requires 130 K bytes of storage (plus ~ 26 K for computer supervisor) while the Out-of-Kilter Network Program requires an additional 135 K. Using an overlay structure reduces the 265 K total to 200 K. [When link-edited with QKCORE (see Appendix H) to complete the overall nuclear power management model (see Section 5.2), the code storage requirement increases to 345 K without overlay or 220 K with overlay (exclusive of computer supervisor).] Execution time is not noticeably increased by the use of the overlay structure.

4.7 Summary

For each multi-year refueling and maintenance strategy, the SOM receives period-by-period system nuclear energy production requirements and system non-nuclear operating costs. The SOM performs a two-stage iterative optimization in conjunction with the necessary CORSOM's to produce the required nuclear energy at minimum total nuclear cost. The optimized final nuclear cost is then added to the present-value of all the other operating expenses to determine the total ORR operating revenue requirement for the strategy. It is this final total cost which is used to rank the alternatives economically.

CHAPTER 5

EVALUATION OF THE SYSTEM INTEGRATION AND OPTIMIZATION MODELS

5.1 Purpose of Evaluation: Critical Questions

When pursuing research in "methods development," important questions must be answered. These critical questions revolve around the characteristics of the numerical method and the model itself:

(1) To what problems is the model applicable?

(2) What assumptions are required?

(3) Does the method converge to an optimum?

(4) Is it the global optimum?

(5) How accurate are the results?

(6) What are the computational requirements?

Once these questions have been answered satisfactorily, research interest shifts from the methodology to the impact of its results.

Since the main thrust of the work reported here is methods development, the purpose of the evaluation is to aid and abet further development by searching for the answers to these basic questions. After a brief discussion of the hypothetical utility system studied (Section 5.3), the detailed discussion of results is presented. Section 5.8 concludes the chapter with a summary of the findings with respect to each of the critical questions.

5.2 Completion of Nuclear Power Management Multi-year Model

To properly evaluate the SIM and SOM (or more specifically, the computerized versions SYSINT and SYSOPT, respectively), requires interfacing them with a RAMM and CORSOM's to complete the nuclear power management multi-year model of Figure 2.21.

For the purposes of developing and testing a SIM and SOM, the multitude of possible alternative strategies output by a RAMM may be replaced by a few typical strategies developed through simple hand calculations (see Section 5.3.3). On the other hand, the on-line iterative nature of the optimization procedure requires computerized CORSOM's. The state of the art, as witnessed by the concurrent methods development research by Kearney (41) and Watt (55), precluded utilization of an established multi-year CORSOM. In order to proceed with the testing of the SIM and SOM, QKCORE, a pseudo-one dimensional, quick core model (performing simulation only) was developed (see Appendix H). The nature of QKCORE necessarily limited the scope of the evaluation to LWR's with the following characteristics:

(1) Modified-scatter refueling with fixed number of zones (e.g., refueling fraction was fixed at one-third),

(2) No plutonium recycle,

(3) No optional stretchout beyond reactivity-limited energy and

(4) No cycle-to-cycle optimization

(i.e., at each refueling, minimum enrichment chosen regardless of future cycles).

Nevertheless, QKCORE is a key element in the success of the SYSOPT evaluation. By generating coupled and well-behaved physics data, the resultant total costs and marginal costs passed to SYSOPT are also well-behaved. It provides all of this at a very high speed. On an IBM 370 model 155 computer, less than 15 milliseconds (CPU time) per reactor cycle were required to choose the proper refueling enrichment to yield the required cycle energy, deplete the core and calculate the cost of that energy. On the same computer, a simplified two dimensional FLARE-type model requires on the order of seconds to perform the depletion task alone--an increase of at least two orders of magnitude.

5.3 Hypothetical Utility System Studied

An 11,000 MW (~ 45% nuclear) utility was hypothesized in order to confirm the nuclear power management multi-year model's applicability to large utility systems. To properly represent scheduled downtime and, at the same time, keep computation costs within a development budget, one month was chosen as the length of each time period. Customer loads (see Section 5.3.1) were forecast for six calendar years on this monthly basis. With respect to generating equipment, the utility's forty fossil generating units (see Section 5.3.2) were chosen so as to have a representative span of sizes and heat rates. With respect to nuclear equipment, four 1050 Mwe PWR's were assumed to be on the system initially with two more to be commissioned on specific dates within the planning horizon. These additions, plus typical fossil additions and retirements were taken as fixed for the multi-year horizon.

Assuming negligible (or invariant) transmission costs and with all alterations to system generating capacity completely specified, only the operating revenue requirements need be considered when comparing alternative refueling and maintenance strategies (see Section 2.1.3). Three such possible alternative strategies (see Section 5.3.3) were developed for satisfying the customer load demands and the generating equipment maintenance requirements.

The model's behavior for a typical strategy (see Section 5.6) and the relative economics of the three strategies (see Section 5.7) form the data base for all of the evaluations in this chapter.

5.3.1 Customer Loads

Representation of monthly customer loads required three pieces of information:

(1) a load-duration curve, normalized on both scales,

(2) a normalizing factor for the load scale (P_D^{max} MW peak load) and

(3) a normalizing factor for the duration scale (T' hours in the time period)

Utilizing Commonwealth Edison data covering several recent years, the four normalized load-duration curves presented in Figure 5.1 where chosen to represent obvious seasonal variations.

A typical set of twelve monthly peaks (see Figure 5.2) was assembled for the first year with an overall peak of 10,000 MW occurring in July. The resultant monthly minimum loads are also presented in Figure 5.2. Note that what may appear at first glance in Figure 5.1 to be seasonal variations in the minimum load are actually the result of variations in the peak loads, i.e., the normalizing factors. In fact, the non-seasonal nature of the nightly minimum load components results in remarkably constant monthly minimum loads.

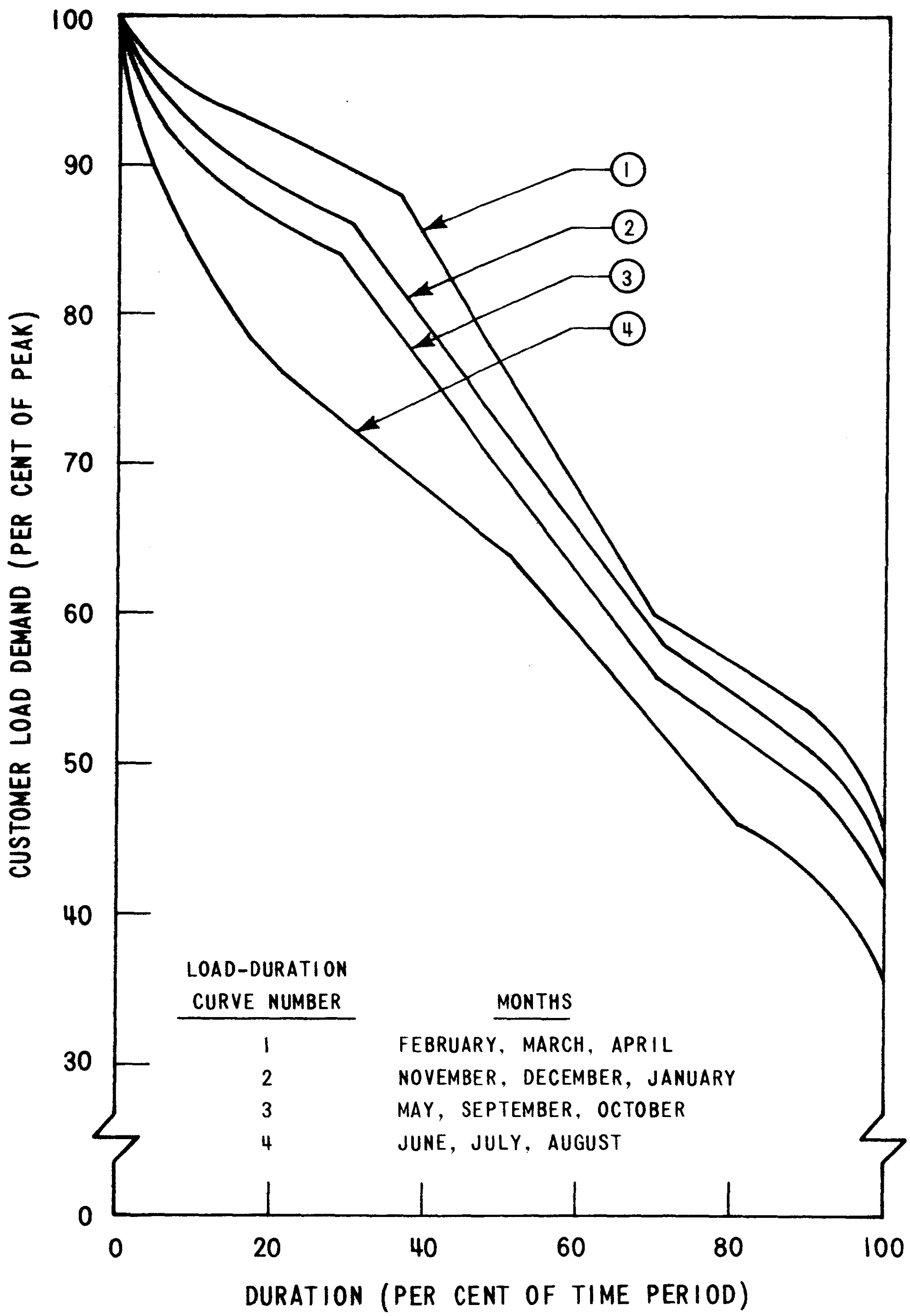
For the remaining five years in the planning horizon, monthly peaks (see Table 5.1) were forecast using 7% annual growth (rounded to 10 MW). As for time period duration, all months were assumed to be 730 hours (30.4 days) in length.

Having specified the required three pieces of information for each period, customer loads had been forecast six years in the future. One of the current model's shortcomings is that it assumes these are perfect forecasts, which, therefore, are treated as deterministic. The

6253-47

Figure 5.1

Input Load-Duration Curves



6253-48

Figure 5.2

Forecasted Monthly Minimum and Maximum Loads for First Year

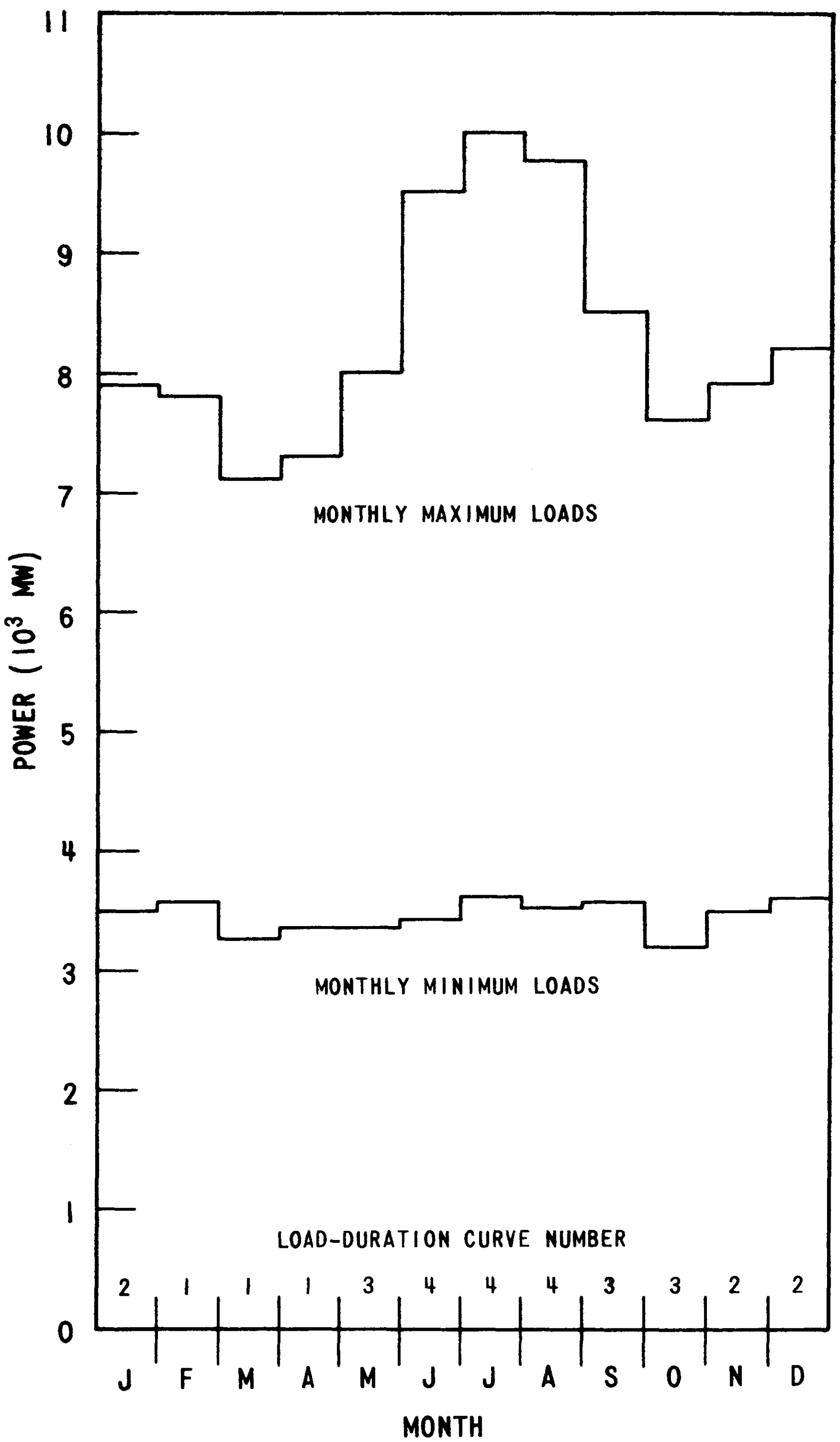


Table 5.1

Forecast of Monthly Peak Loads

(in Megawatts)

Month	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
January	7,900	8,450	9,050	9,680	10,350	11,070
February	7,800	8,350	8,940	9,560	10,210	10,910
March	7,100	7,600	8,130	8,700	9,300	9,950
April	7,300	7,810	8,360	8,950	9,560	10,220
May	8,000	8,560	9,160	9,800	10,490	11,200
June	9,500	10,180	10,890	11,640	12,450	13,300
July	10,000	10,700	11,450	12,250	13,100	14,000
August	9,750	10,430	11,170	11,950	12,780	13,650
September	8,500	9,100	9,730	10,410	11,130	11,900
October	7,600	8,130	8,700	9,310	9,960	10,640
November	7,900	8,450	9,050	9,680	10,350	11,070
December	8,200	8,770	9,400	10,050	10,740	11,490

significant probabilistic nature of the Booth-Baleriaux model derives from the simulation of each unit's stochastic forced-outages, not customer's stochastic demands. Though errors in forecasting monthly peaks can be incorporated into the model (18), the truly difficult uncertainties, such as incorrect load-duration shape, have not been adequately investigated. Research into this area is needed to establish the sensitivity of various results to such uncertainties and to develop means of incorporating them directly so that the model yields not only a numerical answer, but also a confidence interval around it.

5.3.2 Generating Equipment

Again relying on Commonwealth Edison Company data, a representative mix of fossil generating equipment was assembled (see Table 5.2). For reliability, units greater than 300 MW were considered must-run units (i.e., at least at minimum load) provided enough demand was present for the must-run units themselves.

Also presented in Table 5.2 are unit heat rate characteristics for each of the nuclear plants. Because of their size and economics, these six units are also treated as must-run units. All have high heat rates characteristic of light water reactors. The two nuclear units (E and F) under-construction at time-zero are assumed to have only 70% performance probabilities for the first twelve months of commercial service. After this shakedown period, they are

TABLE 5.2
GENERATING UNIT DATA

Unit No. r	Name	Rated Cap. K_r MW	Type	SU-SD Reqt. Q_r MegaBTU	Performance Probability p_r	Valve Point by Valve Point Heat Rate Data Capacity in MW and Heat Rate in BTU/kwh K_{r1}	$\overline{h}_{r1}$	K_{r2}	$h_{inc_{r2}}$	K_{r3}	$h_{inc_{r3}}$	K_{r4}	$h_{inc_{r4}}$	K_{r5}	$h_{inc_{r5}}$
1	A100	100	Fossil	510	0.87	10	26800.	45	12500.	65	13420.	90	13700.	100	14540.
2	A120	120	F	850	0.90	30	12780.	70	10640.	110	12190.	120	13400.	***********	
3	B120	120	F	340	0.91	35	12410.	70	10980.	120	12480.	***********		***********	
4	A140	140	F	250	0.85	30	12380.	90	9860.	120	10140.	140	11340.	***********	
5	B140	140	F	500	0.94	30	13190.	75	9080.	100	9120.	125	9350.	140	9520.
6	C140	140	F	1050	0.90	40	12770.	90	10500.	120	11100.	130	11350.	140	11950.
7	A160	160	F	690	0.89	25	13220.	90	9470.	110	10040.	135	10610.	160	11560.
8	B160	160	F	520	0.92	35	13360.	115	8750.	145	9230.	160	9850.	***********	
9	C160	160	F	700	0.94	45	11040.	80	7950.	130	8410.	150	8610.	160	8750.
10	D160	160	F	1130	0.90	45	11030.	75	8260.	115	8430.	150	8480.	160	9590.
11	A220	220	F	660	0.85	70	10110.	150	8270.	185	8900.	205	9120.	220	10100.
12	B220	220	F	660	0.90	70	10110.	150	8270.	185	8900.	205	9120.	220	10100.
13	C220	220	F	1460	0.91	60	9450.	180	8020.	210	8260.	220	8570.	***********	
14	D220	220	F	1460	0.89	60	9450.	180	8020.	210	8260.	220	8570.	***********	
15	A320	320	F	1750	0.87	140	9670.	180	7890.	280	8120.	310	8460.	320	8880.
16	A600	600	F	3360	0.85	200	10320.	400	8600.	530	9050.	570	9510.	600	9670.
17	B600	600	F	4160	0.87	200	9760.	390	7930.	530	8270.	580	8680.	600	8740.
18	A650	650	F	3500	0.88	300	10320.	400	8600.	530	9050.	590	9510.	650	9670.
19	B650	650	F	4500	0.90	300	9760.	390	7930.	530	8270.	600	8680.	650	8740.
20	A830	830	F	5980	0.89	450	9960.	650	8310.	750	8510.	830	8710.	***********	
21	B830	830	F	5980	0.91	450	9360.	650	8310.	750	8510.	830	8710.	***********	
22	PK 1	60	Peaking	66	0.95	54	17000.	60	19500.	***********		***********		***********	
23	PK 2	60	P	66	0.95	54	17000.	60	19500.	***********		***********		***********	
24	PK 3	60	P	66	0.95	54	17000.	60	19500.	***********		***********		***********	
25	PK 4	60	P	66	0.95	54	17000.	60	19500.	***********		***********		***********	
26	PK 5	60	P	66	0.95	54	17000.	60	19500.	***********		***********		***********	
27	PK 6	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
28	PK 7	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
29	PK 8	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
30	PK 9	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
31	PK10	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
32	PK11	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
33	PK12	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
34	PK13	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
35	PK14	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
36	PK15	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
37	PK16	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
38	PK17	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
39	PK18	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
40	PK19	100	P	110	0.95	90	17000.	100	19500.	***********		***********		***********	
41	A	1050	Nuclear	6400	0.95	400	12425.	620	9910.	830	9570.	1050	10390.	***********	
42	B	1050	N	6400	0.95	400	12178.	620	9470.	830	9090.	1050	10130.	***********	
43	C	1050	N	6400	0.95	400	12425.	620	9910.	830	9570.	1050	10390.	***********	
44	D	1050	N	6400	0.95	400	12178.	620	9470.	830	9090.	1050	10130.	***********	
45	E	1050	N	6400	0.70	400	12178.	620	9470.	830	9090.	1050	10130.	***********	
46	F	1050	N	6400	0.70	400	12178.	620	9470.	830	9090.	1050	10130.	***********	

assumed to perform 95% of the time. The physics characteristics of the reactors are detailed in Appendix H.

In order to impose a more severe test of the nuclear planning ability of the model, the dispatcher's opportunities to base-load the nuclear capacity were decreased by adding an admittedly artificial constraint--a long-term contract with a neighboring utility for 1550 MW capacity with 100% guaranteed availability.

The schedule for installing and retiring utility equipment to keep pace with load growth is presented in Table 5.3. All plants not specifically mentioned exist both before and after the time span of interest. Note the typical trend of retiring smaller (and older) equipment with high heat rates in favor of larger, more efficient units. The system characteristics are summarized in Table 5.4. (The term "system resources" refers to wholly-owned capacity plus firm purchases). A typical summer and non-summer month on the hypothetical system are shown in Figures 5.3 and 5.4, respectively. The difficulty in base-loading the nuclear plants is readily apparent.

5.3.3 Maintenance and Refueling Strategies

While developing maintenance and refueling strategies, various scheduling constraints, maintenance requirements and initial conditions had to be considered. Due to summer peak loads, reliability considerations were assumed to dictate that no scheduled maintenance was to be performed during

Table 5.3

Additions and Retirements of Equipment

Year	Additions: First Month	(Period)	Unit Name	Retirements: Last Month	(Period)	Unit Name
First	June	(6)	Reactor E PK-15	August	(8)	A-160
Second	June	(18)	A-600		NONE	
Third	June	(30)	A-830 PK-16	August	(32)	B-160
Fourth	June	(42)	Reactor F	August	(44)	A-100
Fifth	June	(54)	C-220 B-600 PK-17 PK-18	August	(56)	A-120
Sixth	June	(66)	D-220 B-830 PK-19	August	(68)	B-120

Table 5.4

Summary of System Characteristics

I. Customer Loads:

Load-Duration Curves	See Figure 5.1
Monthly Peaks for First Year	See Figure 5.2
Monthly Peaks for Six Years	See Table 5.1

II. Generating Equipment:

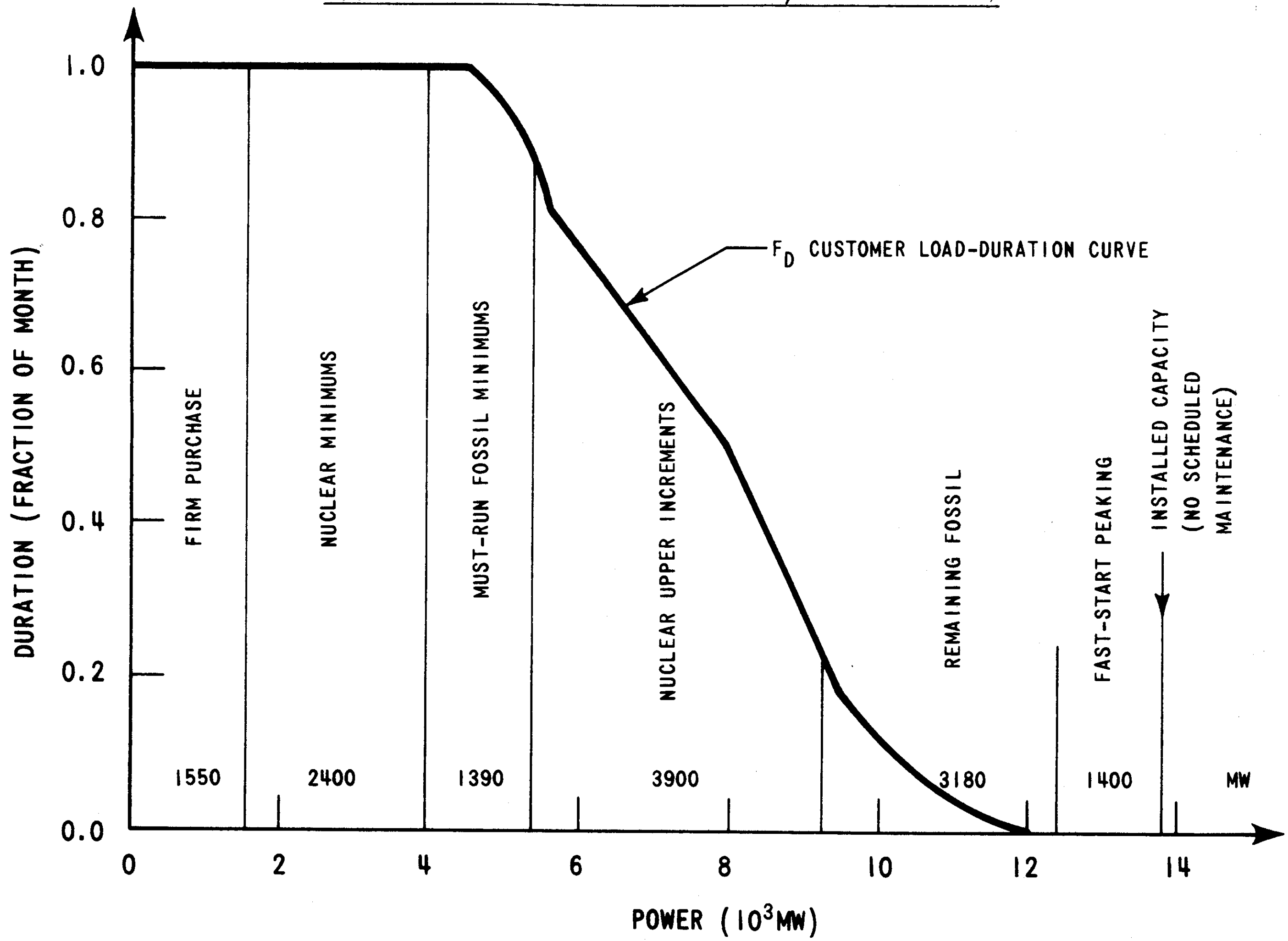
Unit Data	See Table 5.2
Additions and Retirements	See Table 5.3

III. Resulting System Configuration:

Equipment Type	Per Cent of System Resources
Fossil (Non-Peaking)	29-36
Fast-Start Peaking	10-11
Nuclear	41-46
Firm Purchases (1550 MW)	11-14
Total System Resources	100%
Annual Peak Demand	88-89
Resource Margin	11-12%

Figure 5.3

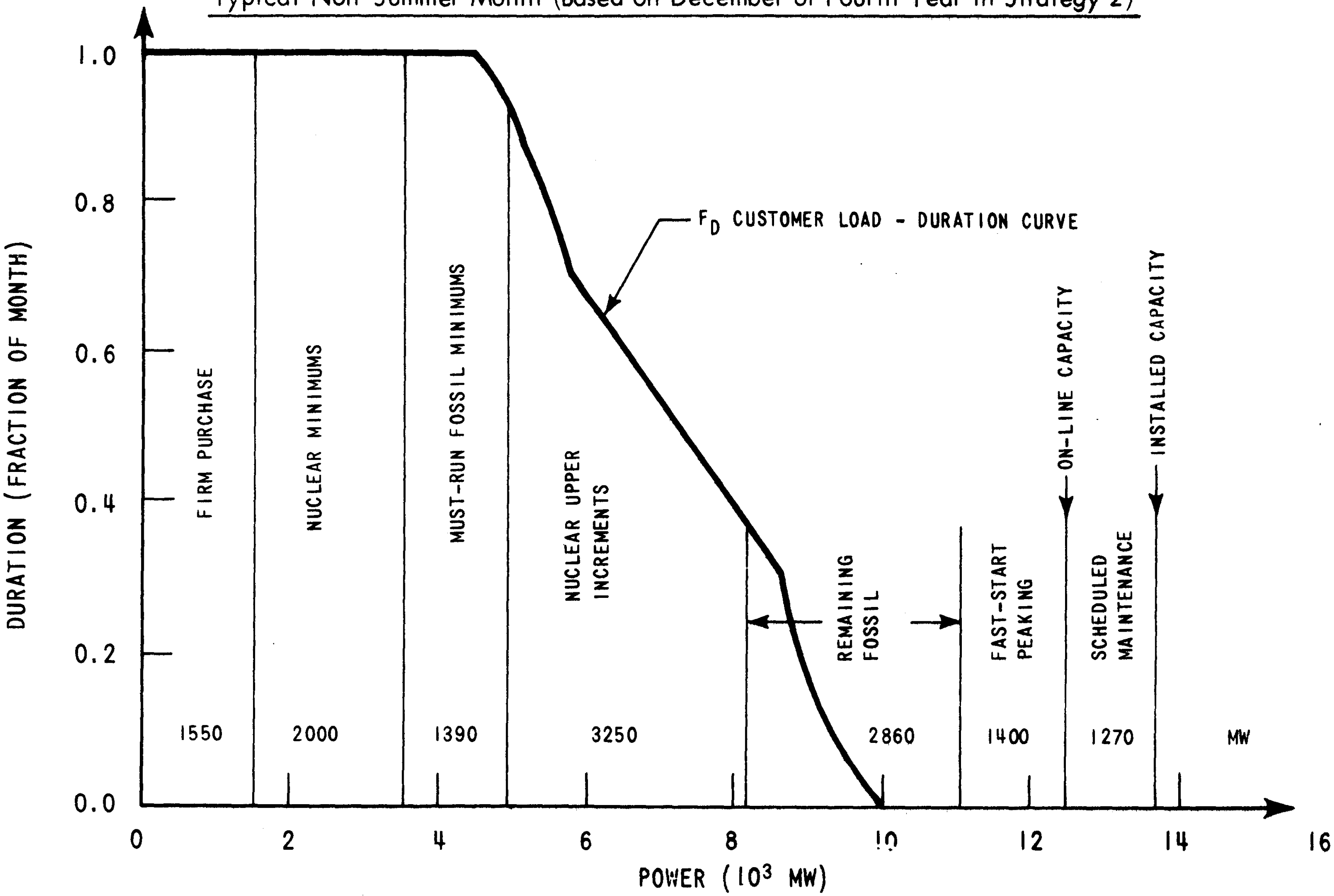
Typical Summer Month (Based on July of Fourth Year)



6253-49

Figure 5.4

Typical Non-Summer Month (Based on December of Fourth Year in Strategy 2)



June, July or August. This typical constraint provided a convenient way of looking at schedules--as nine month "windows" between two summers. Maintenance requirements for (non-peaking) fossil equipment were set at one month per year while fast-start peaking equipment was assumed to be maintained during off-line hours. Two months downtime was assumed for each nuclear refueling. The initial conditions of each reactor core are indicated in Table 5.5.

Within this context, the following three nuclear refueling schedules were postulated:

S-1 : Strictly annual refueling

S-2 : Gradual shift to longer cycles (14 months) to increase cycle energy production

S-3 : Immediate shift to the longer cycles.

These schedules are presented graphically in Figures 5.5, 5.6 and 5.7, respectively. For each of these strategies, fossil maintenance was then scheduled so as to level-off the monthly capacity margin. Figures 5.8, 5.9 and 5.10 present detailed views of the maintenance and refueling schedules for each of the strategies during the first full scheduling window. Note that during the window, each strategy, in turn, refuels one less reactor (i.e., 2100 MW-months less downtime). Thus, the average monthly resource margin during the nine month window increases by 233 MW.

Before considering Strategy 3 further, note that due to the immediate shift to longer cycles, two initial conditions must be violated--namely, the enrichments already

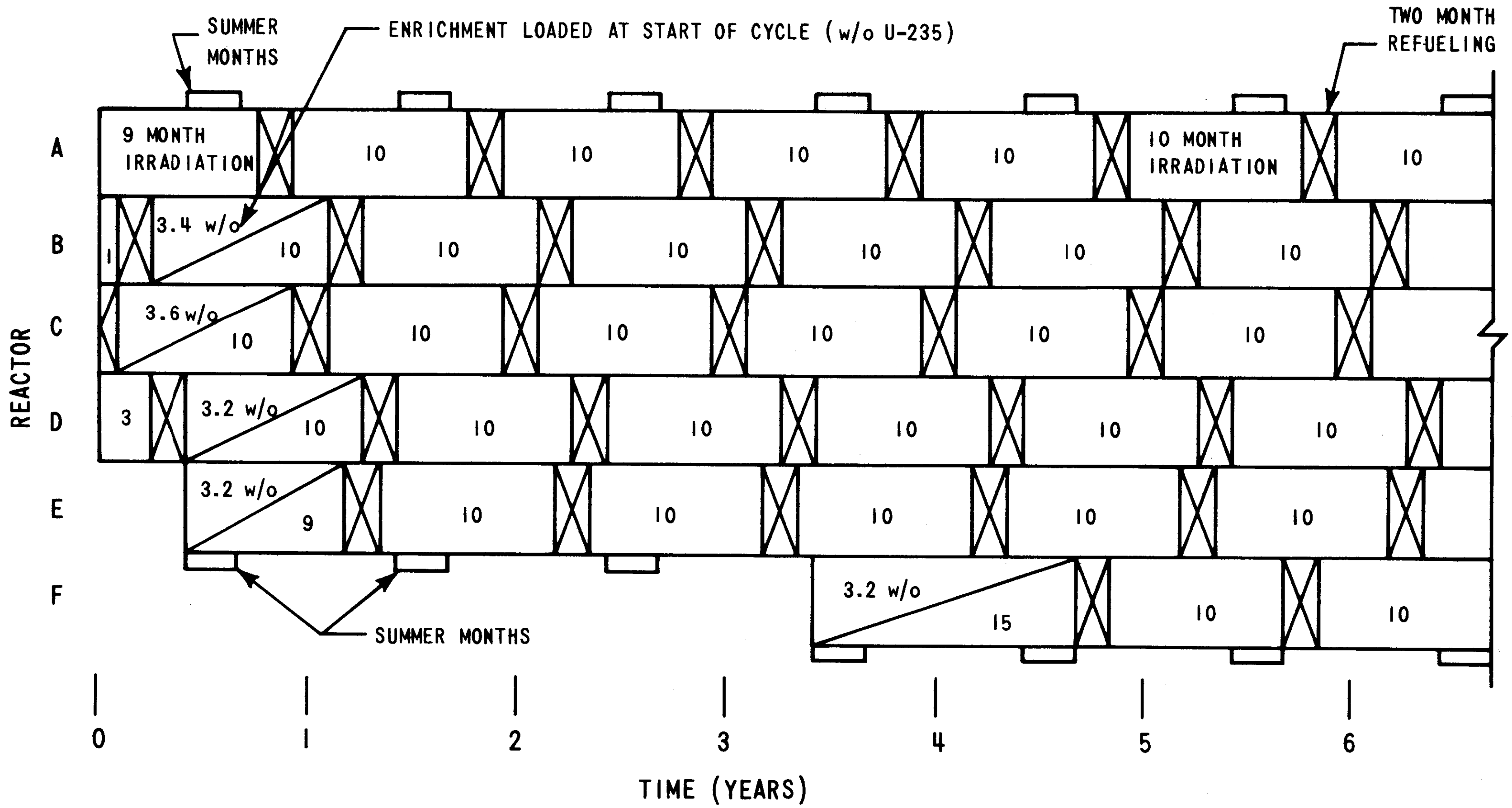
Table 5.5

Initial Conditions of Nuclear Reactors

Reactor	Current Status	Scheduled Refueling During First Year	Enrichment Ordered (w/o U-235)	Zone	Current Core Contents: Enrich. (Fab.) (w/o U-235)	Current Core Contents: Current Burnup (MWD/Kg)
A	Generating	October-November	Open	1	3.3	1.5
				2	3.3	11.5
				3	3.3	20.5
B	Generating	February-March	3.4	1	3.4	9.0
				2	3.2	19.0
				3	3.2	28.0
C	Down for Refueling	January (Current)	3.6	1	3.3	10.0
				2	3.3	20.0
				3	N/A	N/A
D	Generating	April-May	3.2	1	3.2	7.0
				2	3.2	17.0
				3	3.2	27.0
E	Under-Constr. (On-line June First year)	Open	Open	1	3.2	0.0
				2	2.7	0.0
				3	2.2	0.0
F	Under-Constr. (On-line June Fourth year)	Open	Open	1	3.2	0.0
				2	2.7	0.0
				3	2.2	0.0

Figure 5.5

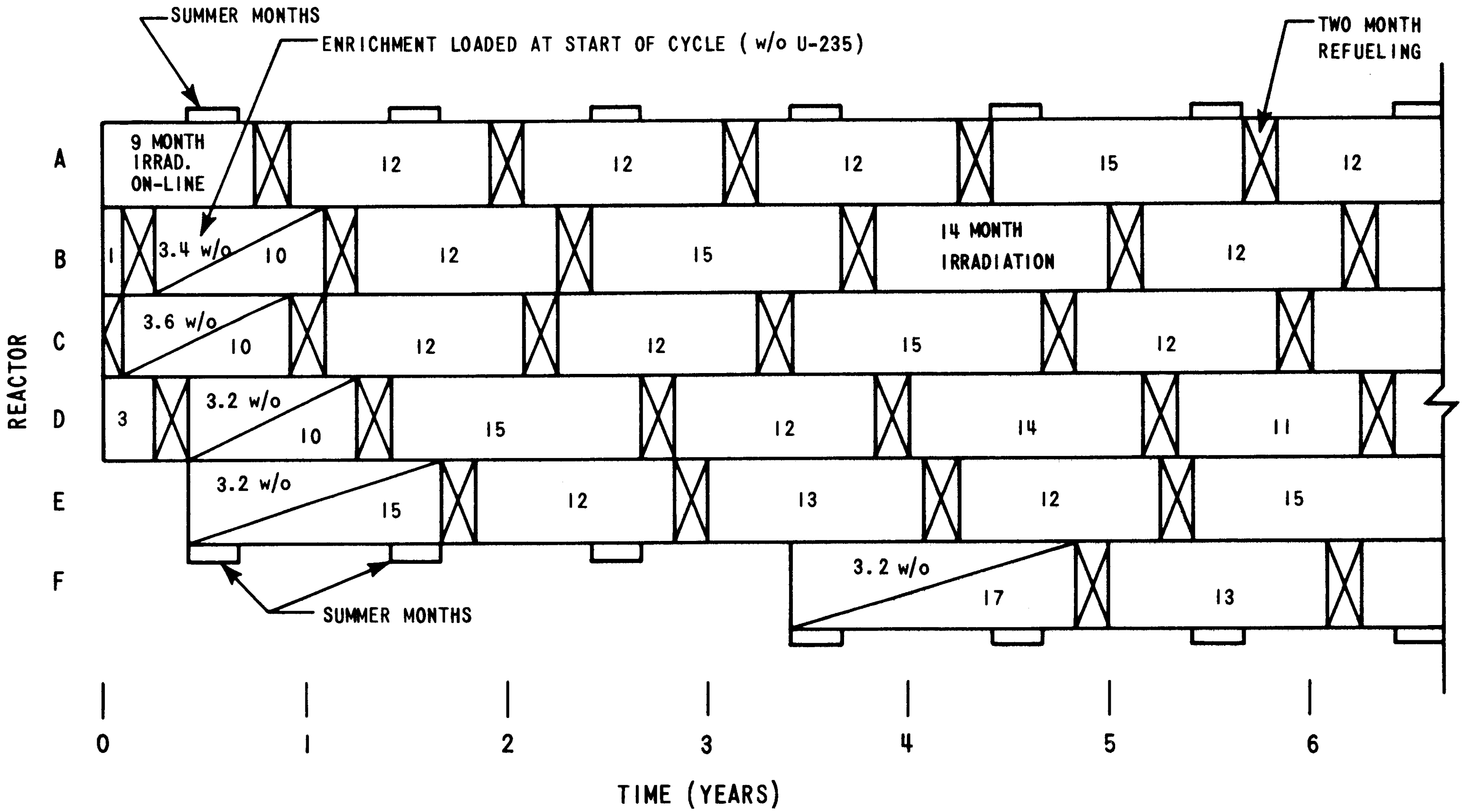
Strategy 1: Annual Refueling (12 Month Cycles = 10 Up + 2 Down)



6253-52

Figure 5.6

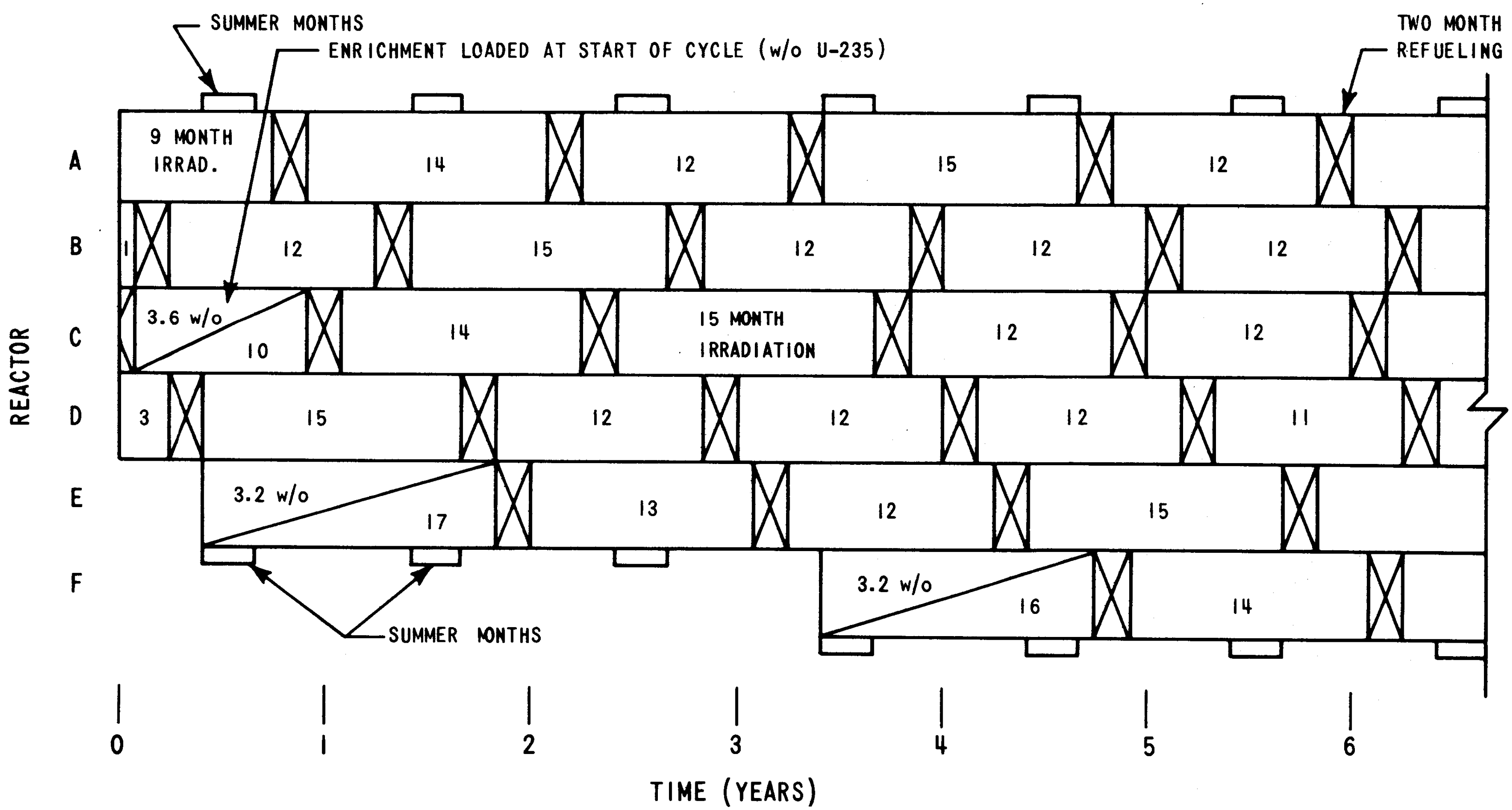
Strategy 2: Gradual Shift to Longer Cycles (14 Months when Possible = 12 Up + 2 Down)



6253-56

Figure 5.7

Strategy 3: Immediate Shift to Longer Cycles (14 Months when Possible = 12 Up + 12 Down)



6253-58

6253-59

Figure 5.8

Resource Commitment for S-1 During First Scheduling Window

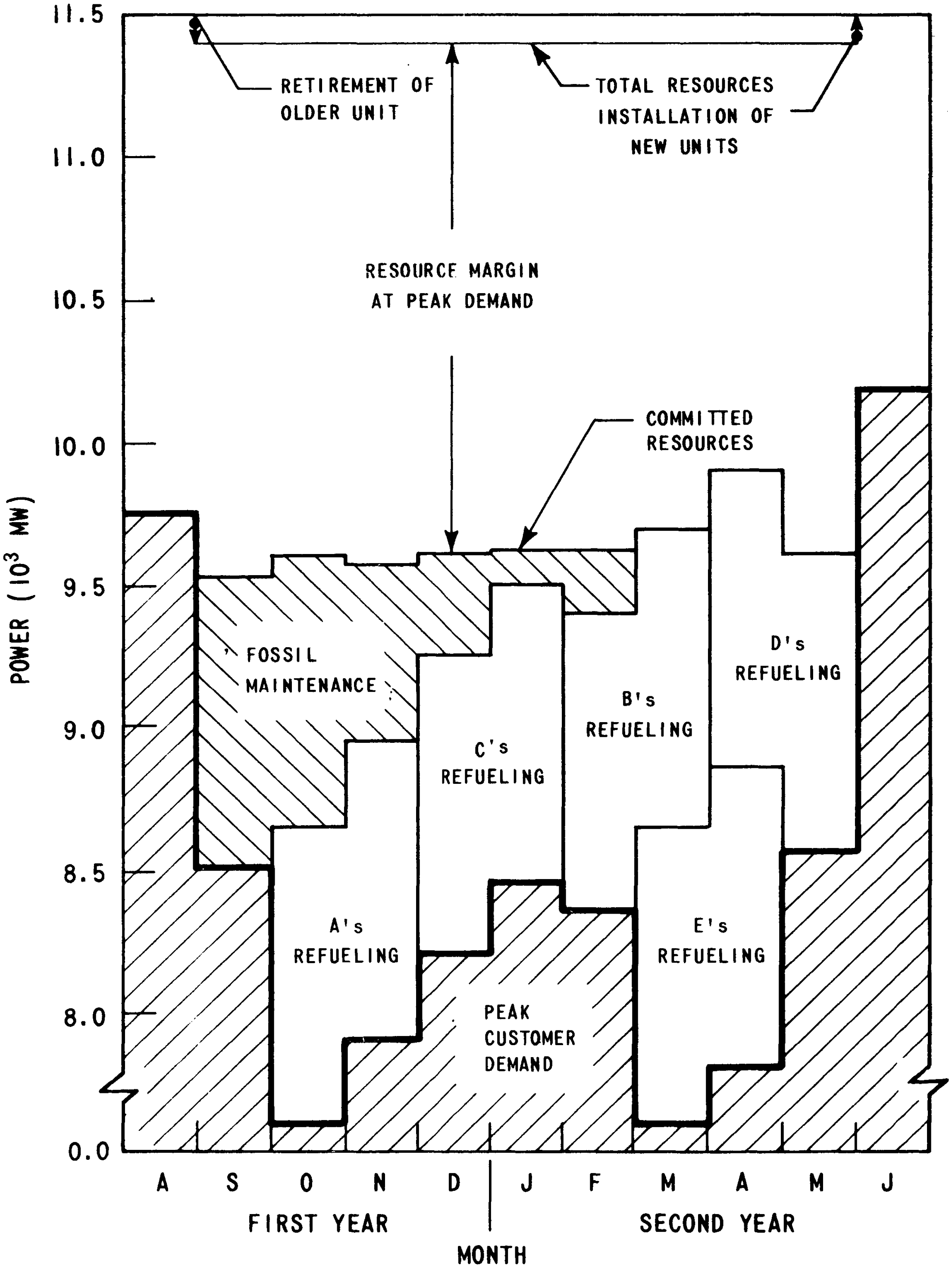
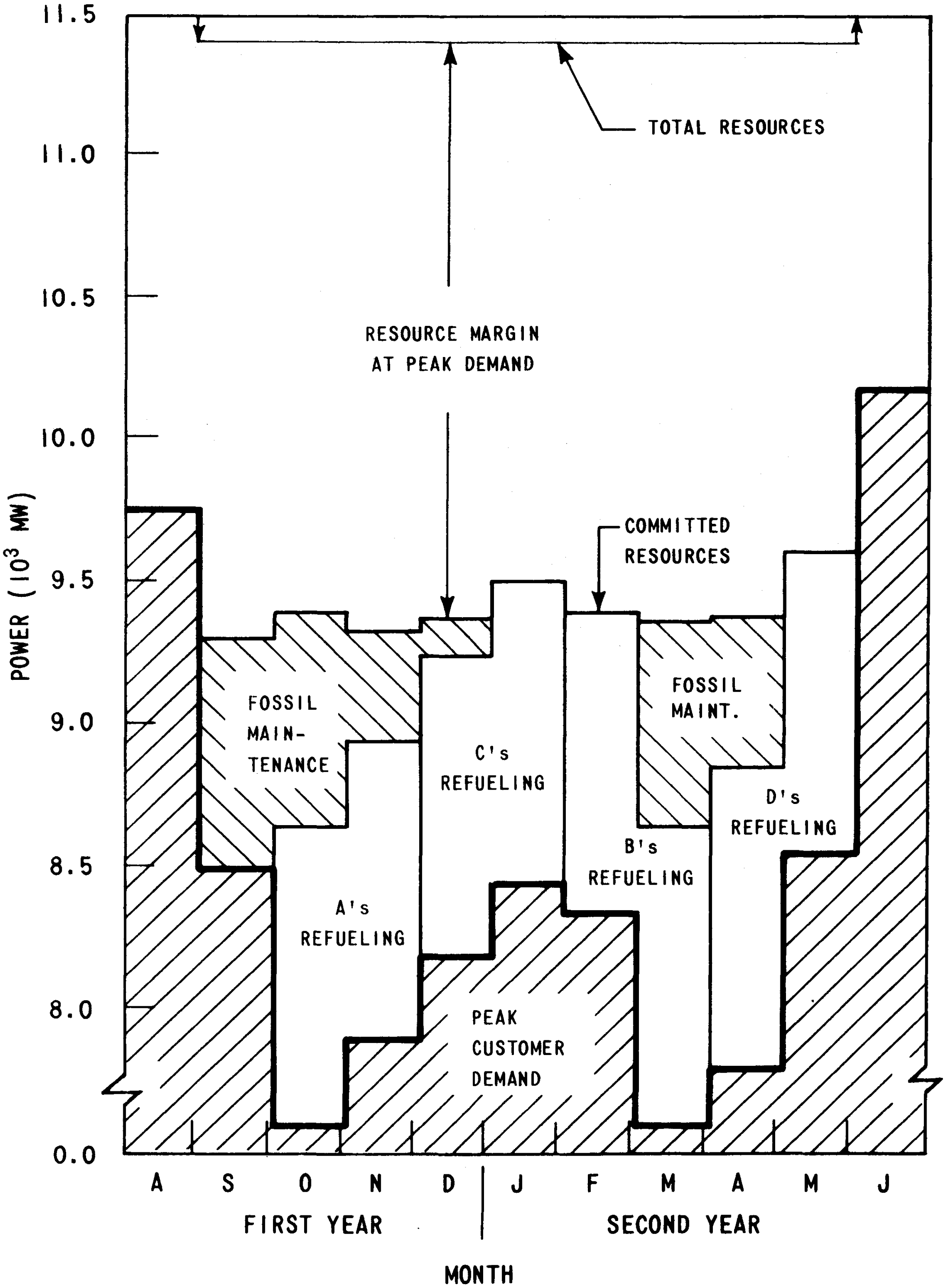


Figure 5.9

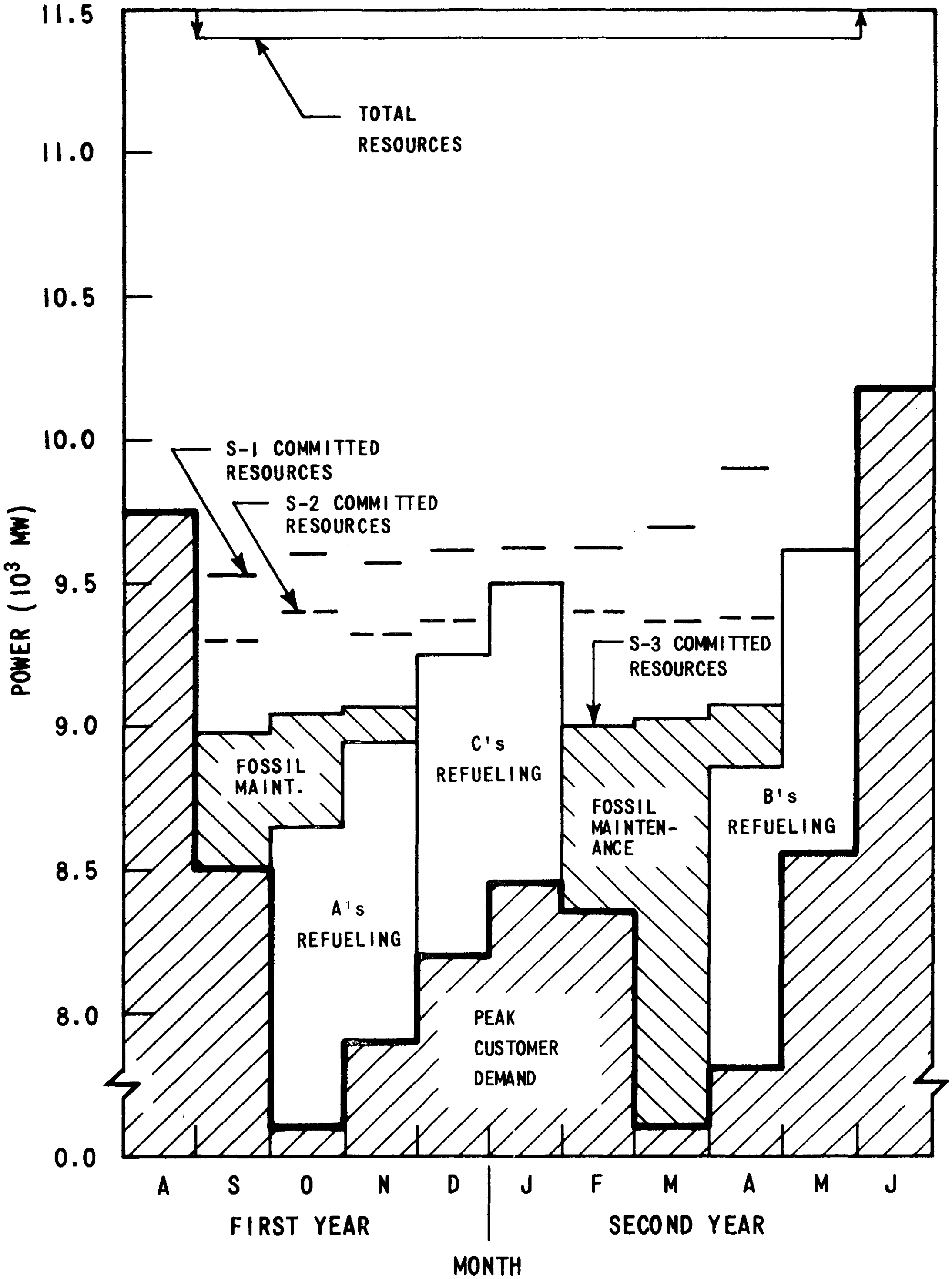
Resource Commitment for S-2 During First Scheduling Window



6253-61

Figure 5.10

Resource Commitment for S-3 During First Scheduling Window



scheduled for Reactors B and D. [Kearney (41) noted the same infeasibility for abrupt large energy changes in the initial cycles.] These longer cycles require increased beginning-of-cycle reactivity to generate the additional energy. Because the QKCORE simulation model required constant refueling batch size for each unit throughout the horizon, the only alternative was to refuel with a higher enrichment. However, the minimum notice for changing reload enrichments is about nine months (20). In order to permit evaluation of S-3, a one million dollar penalty (roughly one year's carrying charges on the unused reload batch) was assessed for changing a batch enrichment on less than nine months notice. This raised new questions: Could S-3 pay such a penalty and still be economically attractive? How much of a penalty could it afford to pay?

The ability of the nuclear power management model to answer such "What if . . . ?" questions is but one indication of the model's versatility and usefulness as a utility management planning tool.

5.4 Remaining Parameters of Interest

In addition to the customer load demand, utility generating equipment and feasible maintenance and refueling schedules, other operating and cost information must be provided. Some of these inputs were arbitrarily fixed at reasonable values (see Table 5.6) throughout the evaluation. Other inputs were adjusted from case to case to evaluate the

Table 5.6

Input Parameters Fixed Throughout Evaluation

	Value	Dimensions
Startup-Shutdown Frequency Curve	See Figure 3.9	
Spinning Reserve Requirement	600	MW
Fossil Fuel Cost	40	¢/MegaBTU
Peaking Fuel Cost	90	¢/MegaBTU
Emergency Energy Purchase	10	$/MWH
Firm Energy Purchase	2	$/MWH
Tax Rate	52	per cent
Refueling downtime	2	months/ refueling
Nuclear Data:		
Enrichment Feed Assay	0.711	w/o U-235
Enrichment Tails Assay	0.25	w/o U-235
Pre-Irradiation Investment Lead Time	0.5	year
Post-Irradiation Credit Lag Time	0.6	year
Delay Time From Yellowcake to UF_6	0.123	year
Processing Yields:		
Conversion	0.995	
Fabrication	0.99	
Reprocessing	0.99	
Re-conversion	0.995	

model's performance (see Tables 5.7 and 5.8).

From a computational viewpoint, note that the six cases per strategy represent perturbations of only SYSOPT's input. Thus only one reference 72 period SYSINT run was required per strategy. Furthermore, because many of SYSINT's unit costs were fixed per Table 5.6, the effect of varying cost parameters could be determined by hand calculation.

5.5 Numerical Results

With all the pertinent information specified for each of the eighteen optimizations, the necessary computer runs were carried out. The revenue requirements and undiscounted energy totals up to the end of specified planning horizon are tabulated for each of the cases in subsequent sections where appropriate to the particular discussion. These tables are cross-referenced in Table 5.8 for ease in locating the results of the six cases.

In addition to these results, Appendix D also presents more detailed numerical results relative to each reactor-cycle (e.g., cycle energy, average energy cost, incremental energy cost and reload enrichment).

The discussion of the results of the cases is the subject of the remainder of this chapter.

5.6 Numerical Evaluation of an Optimized Strategy

Underlying later discussion of the choice from among several optimized strategies are the properties of the individual strategies themselves. The important numerical

Table 5.7

Nuclear Fuel Cycle Unit Costs

Cost Component	Dimensions	Notation		
		Low (75% Reference)	Reference (12)	High (125% Reference)
Yellowcake	\$/lb U_3O_8	6.00	8.00	10.00
Conversion to UF_6	\$/Kg U	1.72	2.30	2.88
Separative Work	\$/Kg SWU	24.00	32.00	40.00
Fabrication	\$/Kg U	52.50	70.00	87.50
Ship. and Reproc.	\$/Kg (U + Pu)	26.25	35.00	43.75
Re-conversion	\$/Kg U	4.20	5.60	7.00
Pu Credit[1]	\$/gm. Fis. Pu	9.38	7.50	5.62

[1] Note that since plutonium is a credit, it is changed in the opposite direction.

Table 5.8

Structure of Case Study

All three Strategies (S-1, S-2 and S-3) were optimized for each set of input parameters comprising Cases I through VI.

Case Number	Shortened Case Notation[1]	Horizon Length (months)	Present Value Rate (%)	Nuclear Unit Costs[2]	Shape Rej. Criterion[3]	For Results See
I	72M, 7%, R, O	72	7	Reference	0.0	Table 5.12
II	48M, 7%, R, N	48	7	Reference	N.A.	Table 5.14 Table 5.16 Table 5.19
III	48M, 0%, R, N	48	0	Reference	N.A.	Table 5.15
IV	48M, 12%, R, N	48	12	Reference	N.A.	Table 5.17
V	48M, 7%, L, N	48	7	Low	N.A.	Table 5.18
VI	48M, 7%, H, N	48	7	High	N.A.	Table 5.20

[1] Refers to parameter values in next four columns.

[2] See Table 5.7.

[3] Per Section 4.4.3, if $V^2_{REJ} < -0.25$, all period production shapes are accepted regardless of feasibility. Thus, "N.A." represents "Not Applied."

properties are cost convergence, shape convergence, incremental costs and computational requirements. The results (see Table 5.9) of Strategy 2 in Case I (i.e., S-2 with 72 month horizon, 7% present value rate, Reference nuclear unit costs and zero rejection level) will be used for most of the discussion. However, when this strategy fails to clearly demonstrate a point under discussion, another will be utilized.

5.6.1 Convergence of Inner Cost Iterations

Starting from a relatively poor initial guess of equal energy in each cycle regardless of cycle length, the initial (s=0) shape iteration of S-2 in Case I required ten inner cost iterations to converge to $\overline{TC}^{*,0}$ (see Section 4.4.2 and Figure 4.11). The system nuclear fuel cost $\overline{TC}^{t}$ (i.e., the objective function of the optimization) for each iteration is presented in Figure 5.11. The revenue requirements and undiscounted energy for this converged solution are shown in Table 5.10.

The symbol Δ in Figure 5.11 represents the energy step size used to segment the incremental cost curves into the stair-step cost functions required by the NP optimization package (see Figure 4.13). As Δ decreases, the accuracy of the piecewise-linear representation increases as does the computational requirement. Thus, a relatively coarse piecewise fit for λ_{rc} at large Δ was utilized for the initial iterations until either the cycle energies

Table 5.9

Revenue Requirements and Undiscounted Energy for Accepted Global Optimum of Strategy 2 in Case I (72M, 7%, R,0)

	10^6\$	10^6 MWH
Fossil Fuel	276.583	85.836
Startup-shutdown Cost	1.704	--
Emergency Purchases	0.407	0.048
Non-nuclear Production	278.964	85.884
Nuclear Fuel	297.709	194.077
System Production	576.673	279.961
Fixed Firm Purchase	133.920	81.468
System Total	710.593	361.429

6253-62

Figure 5.11

Convergence of Inner Cost Iterations for Initial Shape Iteration of Strategy 2 in Case I

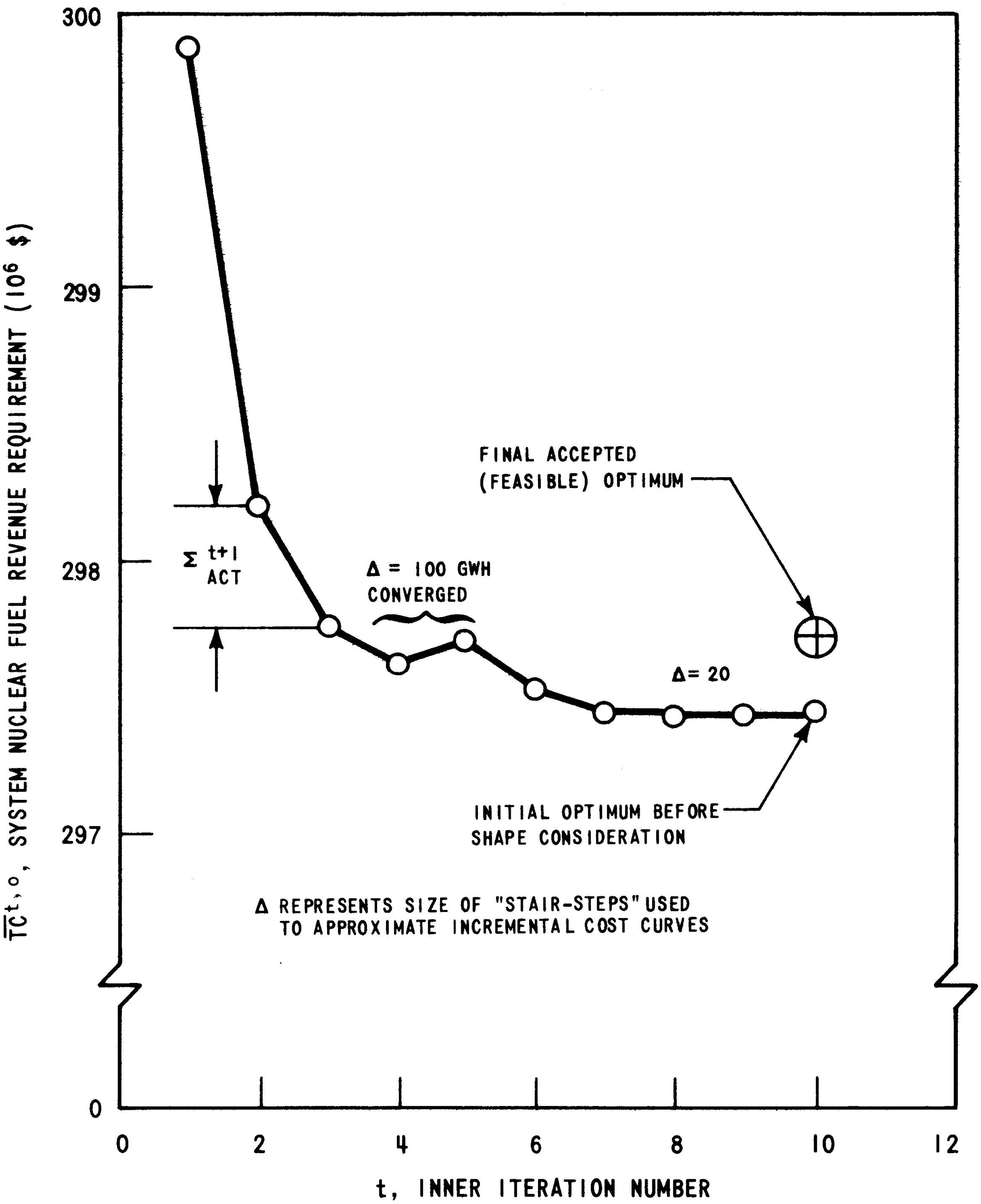


Table 5.10

Revenue Requirements and Discounted Energy For Converged Initial[1] Shape Iteration of Strategy 2 in Case I (72M, 7%, R,0)

	10^6\$	10^6MWH
Fossil Fuel	276.853	85.836
Startup-shutdown Cost	1.704	--
Emergency Purchases	0.407	0.048
Non-nuclear Production	278.964	85.884
Nuclear Fuel	297.456	194.077
System Production	576.420	279.961
Fixed Firm Purchase	133.920	81.468
System Total	710.340	361.429

[1]Per Section 4.4.3, these results also apply for the global optimum for the following input set: 72M, 7%, R,N (cf. Table 5.8).

converged (to within a specified per cent of Δ, typically 100%) or the objective function itself converged (i.e., Σ_{EST}^{t+1} of the last iteration failed to improve the objective function by more than a required amount, say $2000). In fact, iteration 5 displayed "negative" improvement because piecewise-linearization of $\overline{TC}_r$ prevented the NP program from seeing the smooth increase of λ_{rc} for fractional Δ changes in cycle energy. The net result was that the NP program over-reacted to small differences between the incremental costs λ_{rc}.

After convergence using the first Δ, a second and smaller Δ was utilized and convergence again attained using the same two criteria. This second converged solution was considered to be the inner optimum $\overline{TC}^{*,0}$.

From three standpoints, a third Δ choice appeared unwarranted:

(1) With the total nuclear fuel revenue requirement approaching $300,000,000, the fuel cost improvement from the Δ = 100 GWH optimum solution to Δ = 20 was only $220,000 for the fivefold Δ reduction and would undoubtedly have been much less than that for another fivefold reduction.

(2) At Δ = 20 GWH, cycle energies were already converged to well within 1% (± 50 GWH out of 6000-8000 GWH), and

(3) The fuel cost errors and cycle energy errors both appear to be well within the noise levels of CORSOM errors [> $100,000 per reactor over five years (55)] and the errors inherent in forecasting load demands and availabilities (> 1%).

Using the above sequence of the two step sizes for all cases, the initial shape iteration was effectively converged (i.e., objective function decreasing insignificantly for Δ = 20 GWH) within ten inner iterations. In as much as completed CORSOM's are estimated to require over 3 minutes of IBM 370 model 155 CPU time per reactor strategy per iteration (41), a six reactor-ten iteration solution would involve over 3 hours of computer time for the CORSOM's alone. (The *ad hoc* simulator QKCORE required less than 3 minutes for all ten iterations.) Since each iteration of the SOM [using roughly 9 seconds (see Section 4.6)] involves another 20 minutes of CORSOM time, further investigation is recommended into improving the SOM's NP convergence and decreasing the number of iterations required.

Returning to Figure 5.11, a detailed analysis of the iteration-to-iteration improvement in the objective function is warranted. Recalling the development of the cost objective function Σ_{EST}^{t+1} in Section 4.2.1, Equation (4.8) stated that

$$\overline{TC}^{t+1} = \overline{TC}^{t} + \delta^{t+1} + \underbrace{\sum^{R}\sum^{C}\int_{E_{rc}^{t}}^{E_{rc}^{t+1}} \lambda_{rc}^{t}\, dE_{rc}}_{\Sigma_{EST}^{t+1}} \qquad (5.1)$$

Since,

$$\overline{TC}^{t+1} = \overline{TC}^{t} + \Sigma_{ACT}^{t+1} = \overline{TC}^{t} + \delta^{t+1} + \Sigma_{EST}^{t+1} \qquad (5.2)$$

Therefore,

$$\delta^{t+1} = \Sigma_{ACT}^{t+1} - \Sigma_{EST}^{t+1} \qquad (5.3)$$

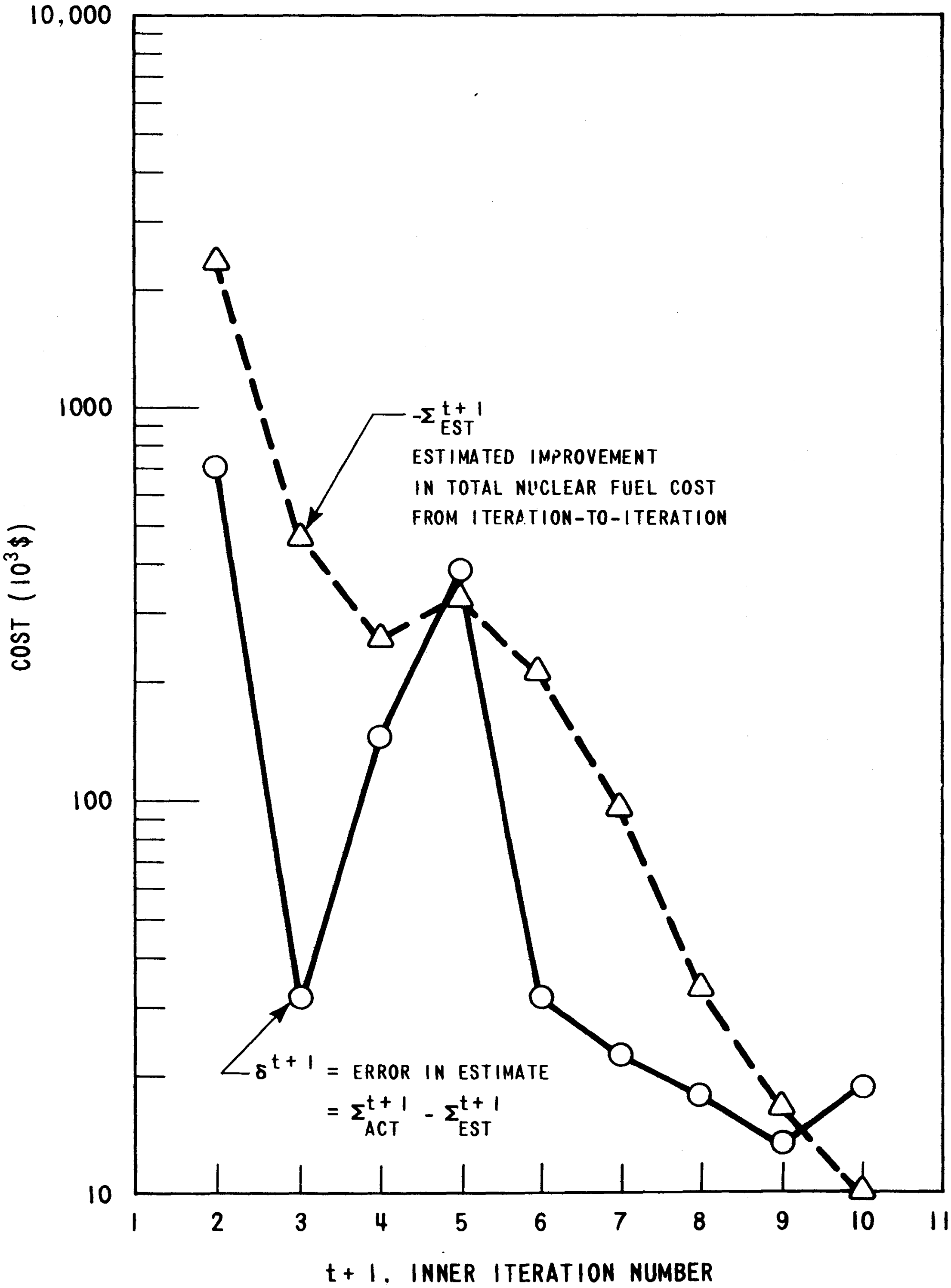
Both Σ_{EST}^{t+1} and δ^{t+1} are presented in Figure 5.12.

Section 4.2.1 postulated simplification of the objective function [Equation (4.12)] based on the assumption that the resulting error δ^{t+1} was much less than the projected improvement, which is the case seven out of nine times. The two failures are a combination of (1) the actual error in the simplification and (2) the NP program's over-reaction to small differences in incremental costs.

By plotting δ^{t+1} versus the average (root-mean-square) energy change for all reactor-cycles altered between the two iterations, Figure 5.13 results. Intuitively, such behavior was to be expected--namely, δ^{t+1} tends to grow large for large shifts in energy. The cluster of data representing less than

Figure 5.12

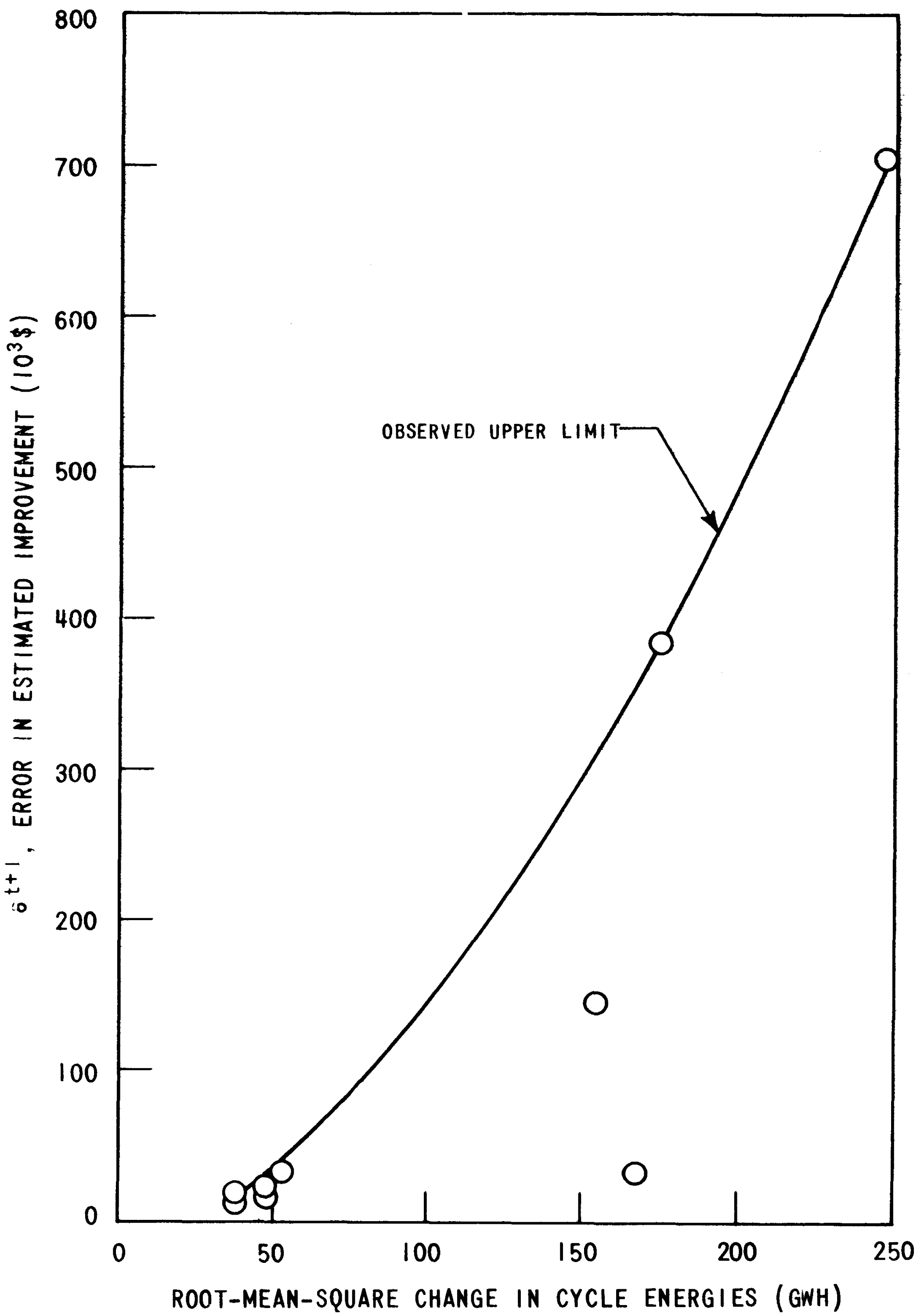
Change in Total Nuclear Fuel Cost During Inner Iterations of Initial Shape Iteration of Strategy 2 in Case 1



6253-64

Figure 5.13

Error in Estimated Improvements versus Change in Cycle Energies
(Strategy 2 in Case 1)



\$30,000 errors for changes on the order of 50 GWH provides adequate justification that the assumption in Chapter 4 can be applied for small changes in energy. The fact that even the largest δ^{t+1} still permits a net improvement indicates, though somewhat less convincingly, an even larger range of applicability.

In summary, the validity of the δ^{t+1} assumption of Section 4.2.1 has been established. The inner NP optimization based on it converged adequately with regard to both cycle energies and total system nuclear fuel cost. However, as previously mentioned, the rate of convergence left something to be desired.

5.6.2 Convergence of Outer Shape Iterations

Strategy 2 in Case I (72M, 7%, R, 0) required four outer shape iterations to achieve the acceptable optimum $\overline{\overline{TC}}^{\circledast}$ by the method described in Section 4.4.3 using the "stairstep" m_{rp} of Figure 4.16. Figure 5.14 plots the progress at each outer s shape iteration of $\overline{\overline{TC}}^{*,s}$ and the number of rejected periods versus the average rejected V_p^2. Convergence is rapid in the sense that the early iterations greatly reduce the average V_p^2 while the later iterations reduce the number of periods that must be included in the average.

Also presented in Figure 5.14 are similar data provided by a separate computer run in which the V_{REJ}^2 was raised from 0.00 to 0.01. Table 5.11 presents a summary of the

Figure 5.14

6253-6!

Outer Shape Iteration Convergence

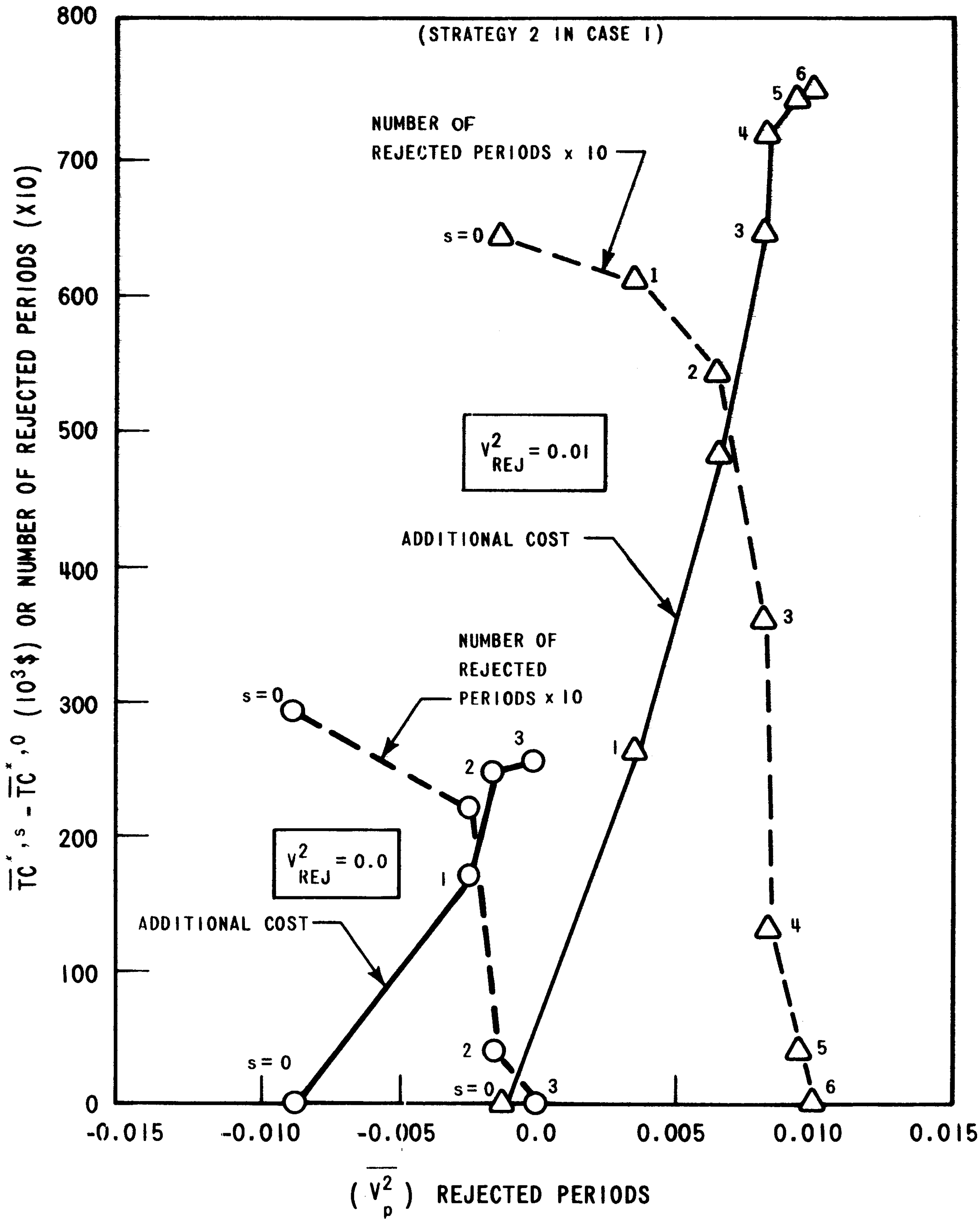


Table 5.11

Results at End of Outer Shape Iterations
(Strategy 2 in Case I)

s	$\overline{TC}^{*,s}$ (10^3\$)	Lowest V_p^2 ($x10^3$)	Average Rejected V_p^2 ($x10^3$)	Number of Periods Rejected
		** V^2_{REJ} = 0.0 **		
0	297,457	-12.31	-8.66	29
1	297,627	- 4.03	-2.35	22
2	297,701	- 4.15	-1.50	4
3	297,709	+ 0.09	0.0	0
		** V^2_{REJ} = 0.01 **		
0	297,457	-12.31	-1.12	64
1	297,717	- 8.03	+3.65	61
2	297,938	- 4.73	+6.63	54
3	298,098	+ 2.75	+8.20	36
4	298,173	+ 6.29	+8.39	13
5	298,199	+ 9.00	+9.46	4
6	298,205	+10.03	10.00	0

important results at the end of each shape iteration for both runs.

During the outer iterations, reactor production limits of each rejected period are "squeezed" toward each other to decrease the likelihood of further rejection (See Section 4.4.3 and Figure 4.17). When the final iteration reaches the global optimum, a distribution of the Z = 72 periods versus the percent original energy range remaining can be plotted as in Figure 5.15. For the run with $V^2_{REJ} = 0$, 42 of the 72 periods required no reduction in energy range (i.e., 100% remaining since never rejected) and the maximum reduction for any single period was 22% (78% remaining). The much stiffer requirements imposed by $V^2_{REJ} = 0.01$ (S^2_p was only ~ 0.02), resulted in only 3 unaltered periods and 45 periods with reductions of 25% or more.

As for the proper choice of V^2_{REJ} itself, Figures 5.16 to 5.18 present system and average reactor shapes yielding the indicated values of S^2_p and V^2_p. Visual inspection indicates the infeasibility of Figure 5.16 and the acceptability of the other two periods. Furthermore, the system shape itself is not an ironclad constraint from the standpoint that the information it contains is the result of many forecasts (customer load-duration shape and performance probabilities), not of well-defined engineering constraints such as are found in deterministic optimization problems (e.g., optimum heat exchanger design). The net result is a

Figure 5.15. Distribution of 72 Period Energy Ranges Remaining at Accepted Optimum (Strategy 2 in Case I)

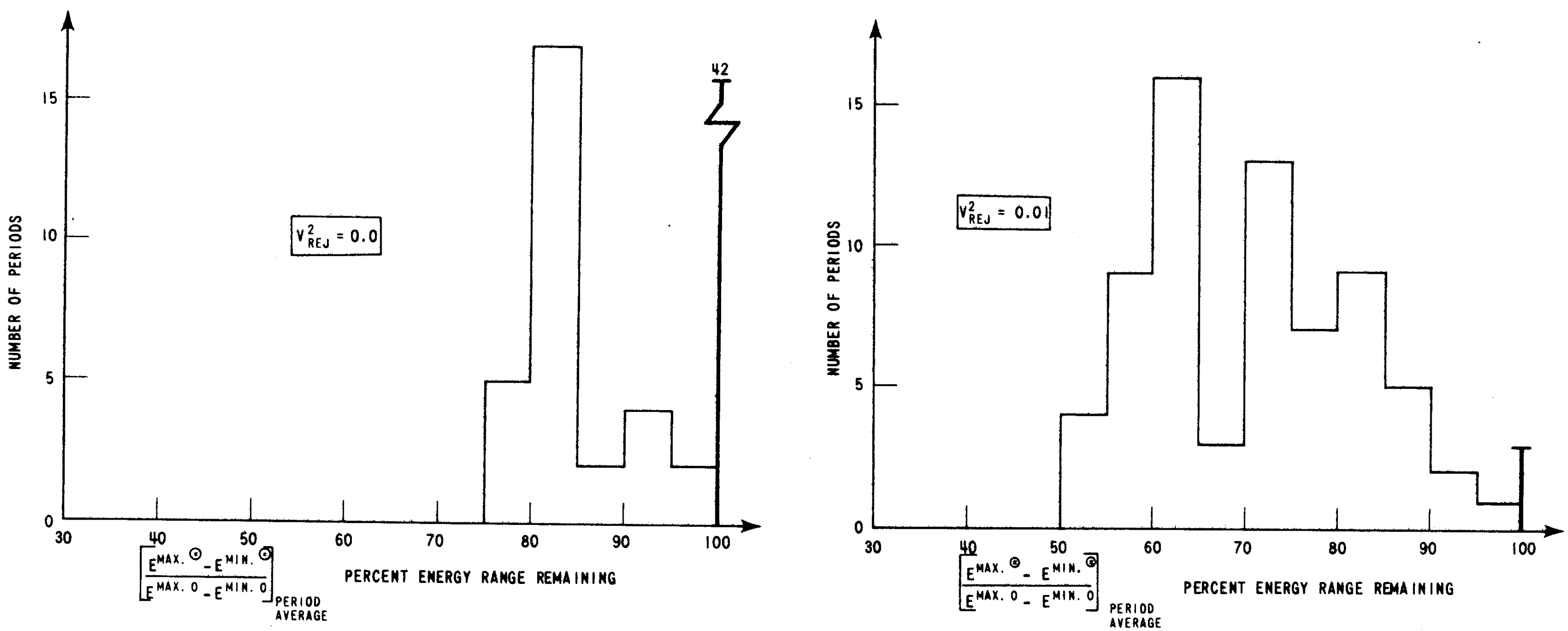
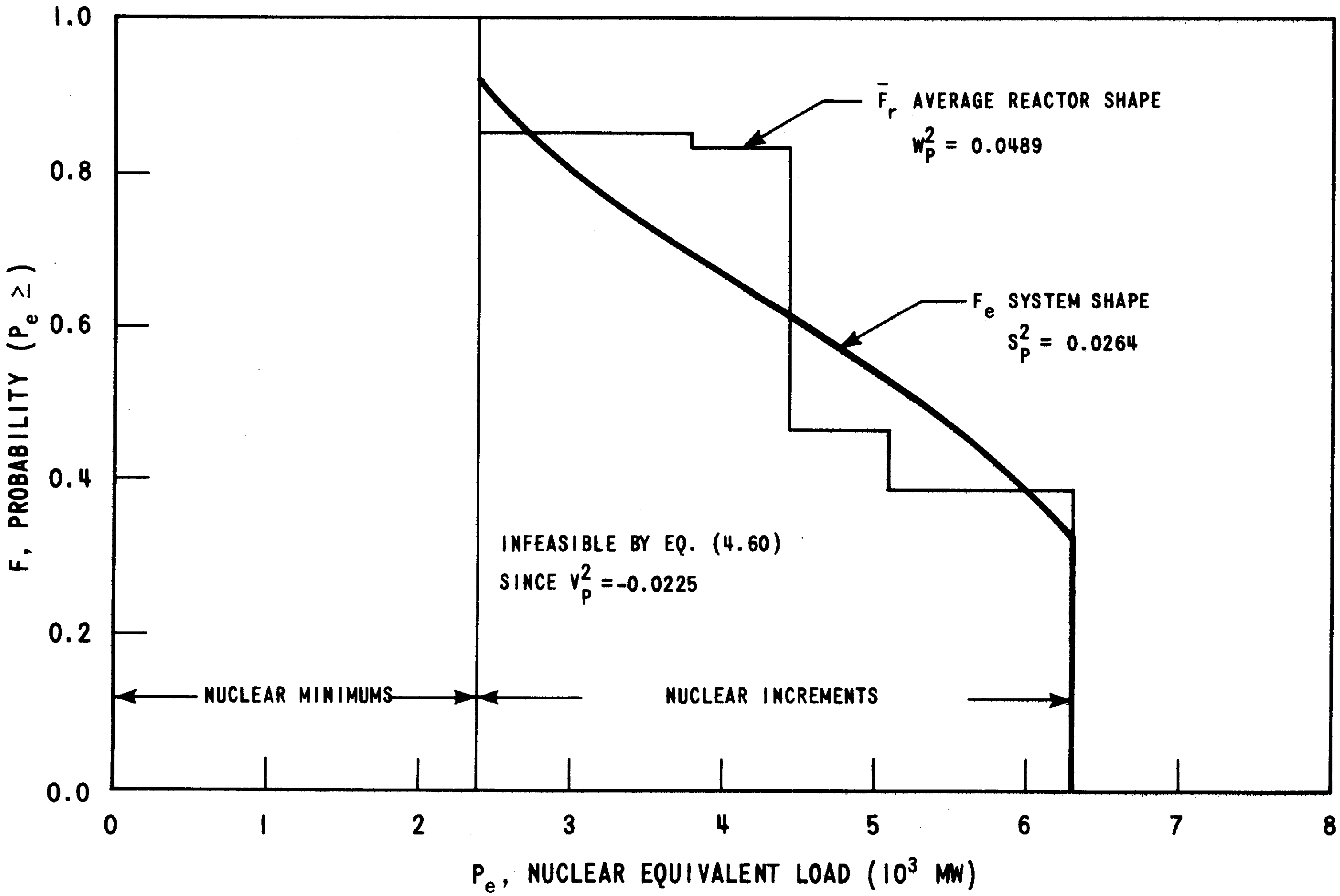


Figure 5.16

Typical Period with Infeasible Postulated Average Reactor Shape ($V^2 < 0$)



-317-

6253-67

Figure 5.17

Typical Period Giving Shape Test V_p^2 Near Zero

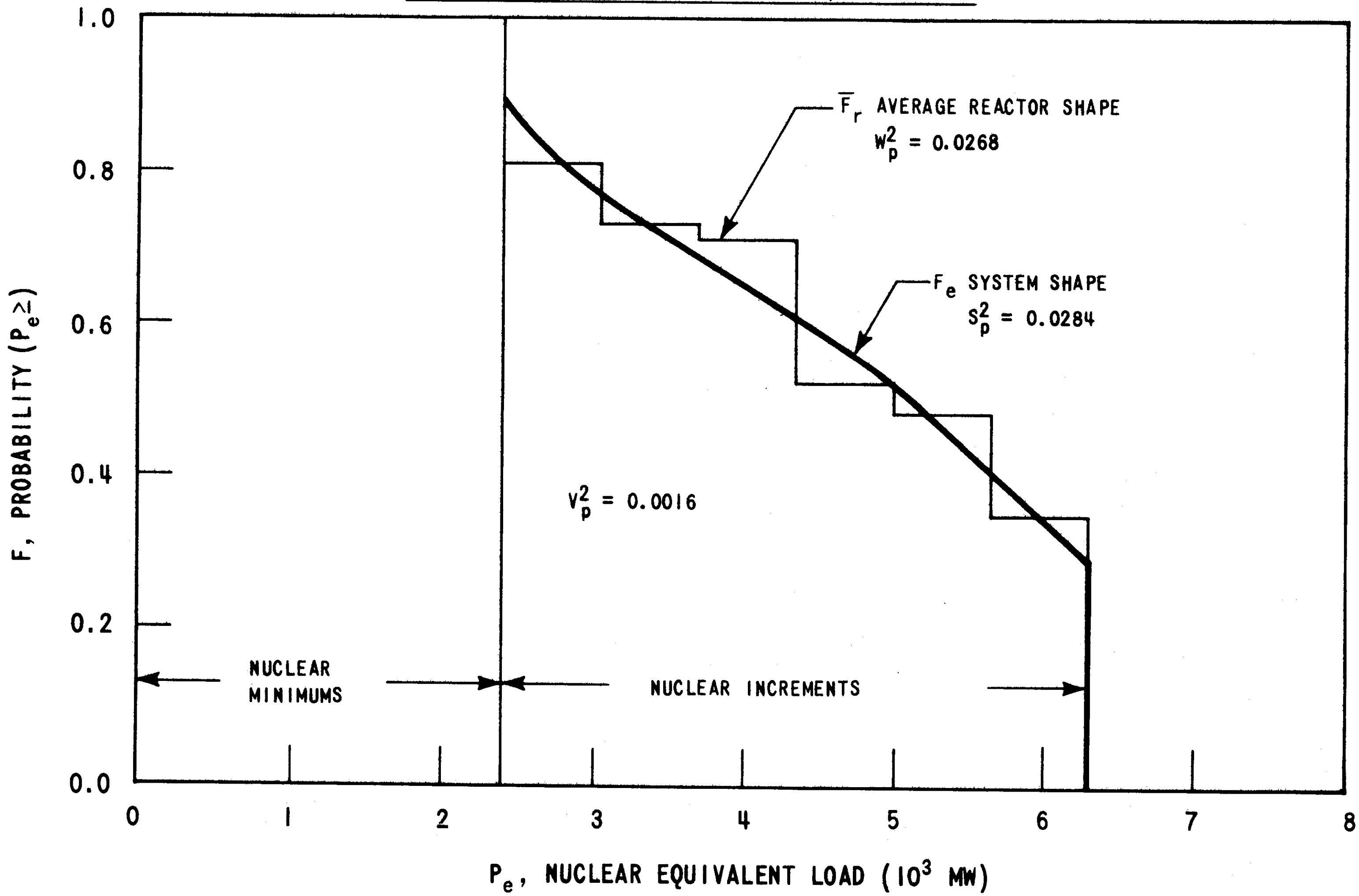
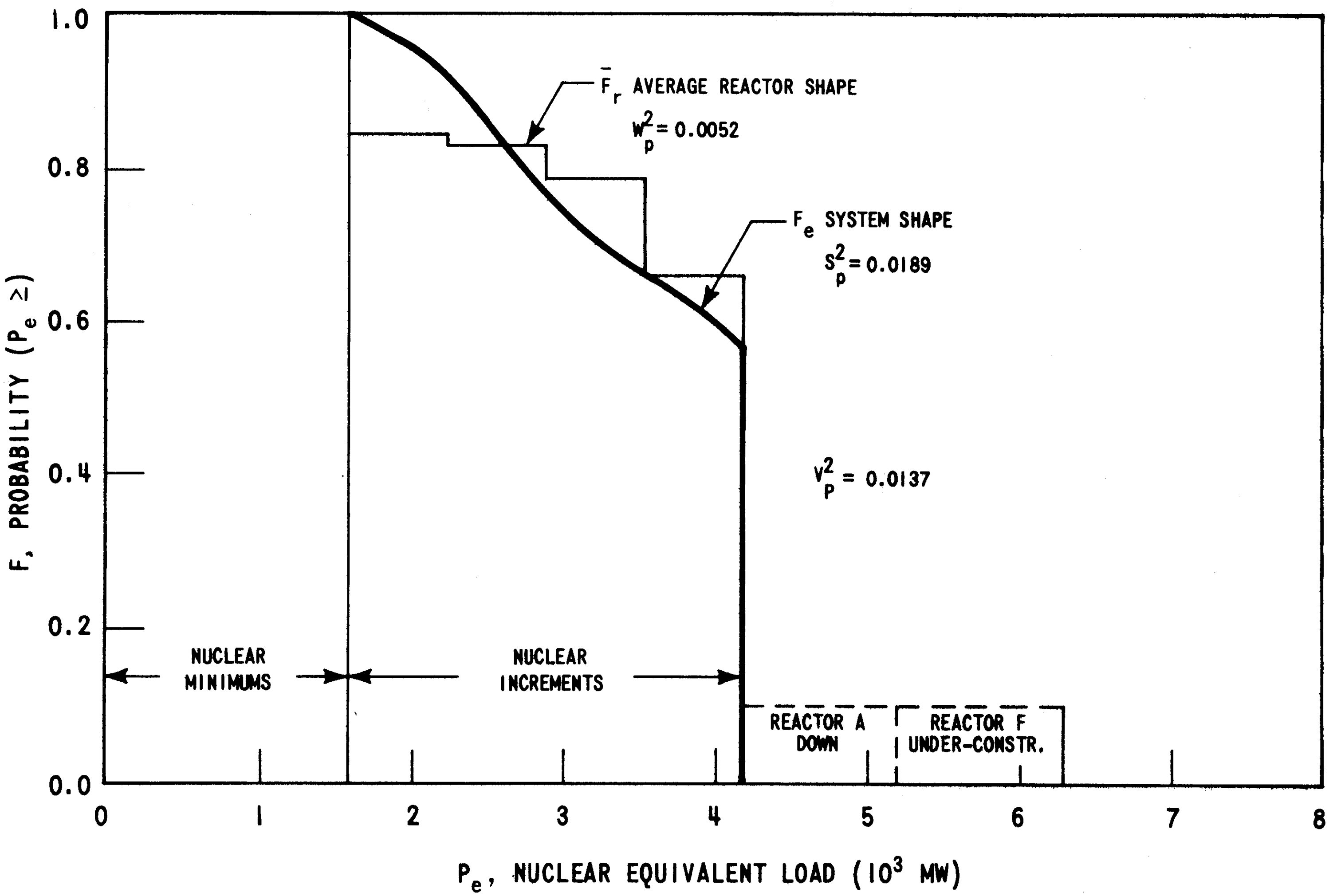


Figure 5.18

Typical Period Giving Shape Test V_p^2 Much Greater than Zero



6253-69

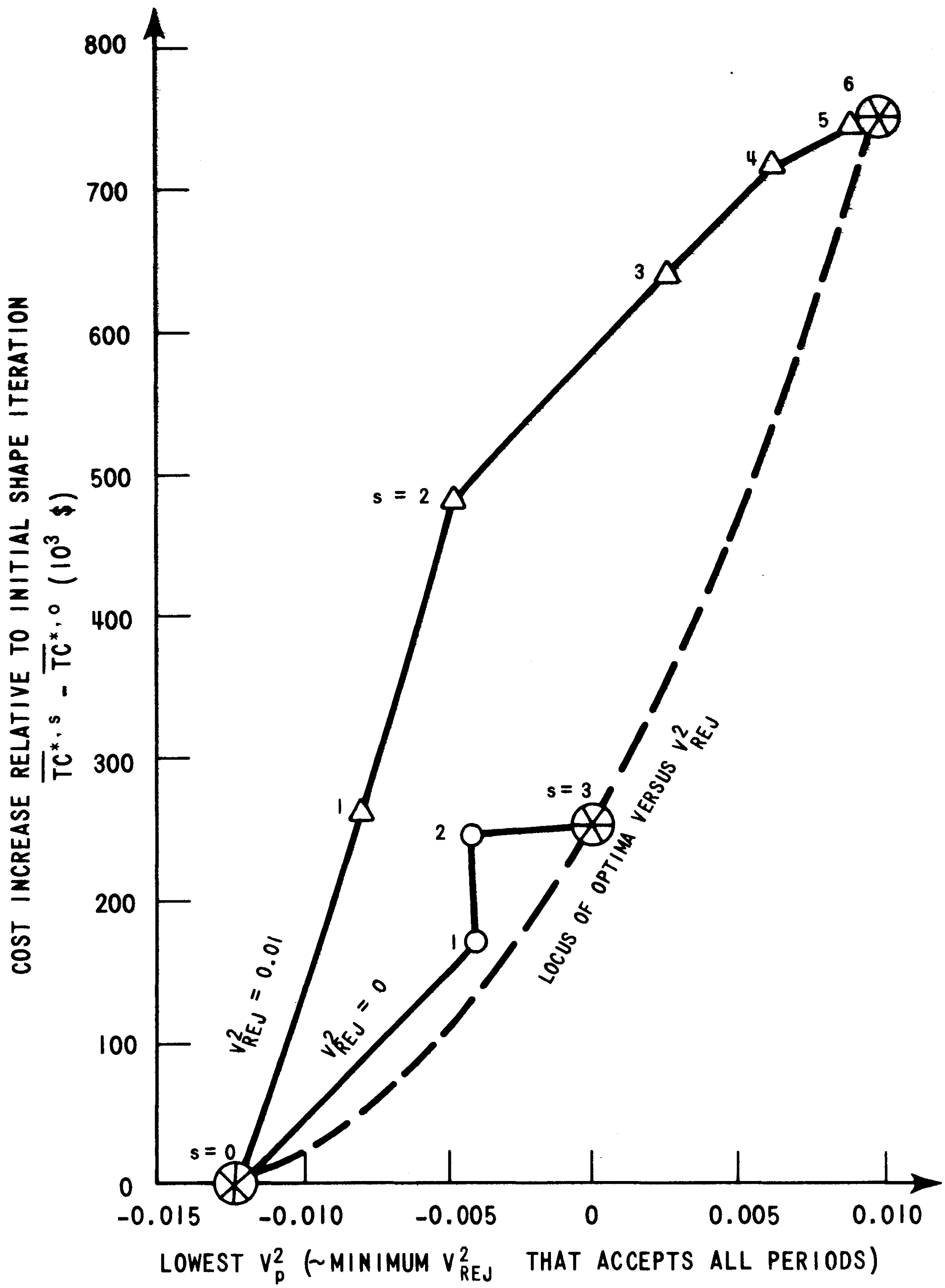
recommendation that $V^2_{REJ} \sim 0$ is satisfactory for planning purposes.

Figure 5.19 presents the iterative progress of $\overline{TC}^{*,s}$ for Strategy 2 in Case I versus the lowest V^2_p (i.e., V^2_p, for the period failing the criterion by the largest amount or equivalently, the V^2_{REJ} that would have accepted all periods). Since both solid curves begin from the same point, but are not co-linear, $\overline{TC}^{*,s}$ is only valid as a measure of minimum system nuclear cost at the final optimum $\overline{TC}^{\circledast}$ for each V^2_{REJ}. In other words, the outer iterations reach their respective global optimums by a sequence of non-optimum iterations. The means of increasing the rate of outer shape convergence, as with inner cost convergence, lies merely in increasing the number of steps used in the piecewise-linearization of the objective functions.

Another input parameter affecting the outer shape iterations is the fraction γ of the σ ($\equiv \sqrt{V^2_{REJ} - V^2}$) actually applied to the reactor production limits [Equations (4.70) and (4.71)]. Figure 5.20 presents a plot of all three optimizations in Case I ($V^2_{REJ} = 0$) as a function of the γ used to achieve the global optimization. The ordinate represents the increase of $\overline{TC}^{\circledast}$ over $\overline{TC}^{*,0}$, absolute minimum cost when all shape constraints are ignored (i.e., ignoring feasibility). (The revenue requirements and undiscounted energy totals for Case I are presented in Table 5.12.)

Figure 5.19

Strategy Cost versus V^2_{REJ} (Strategy 2 in Case I)



6253-71

Figure 5.20

Accepted Optimum for Case I versus γ Correction Factor

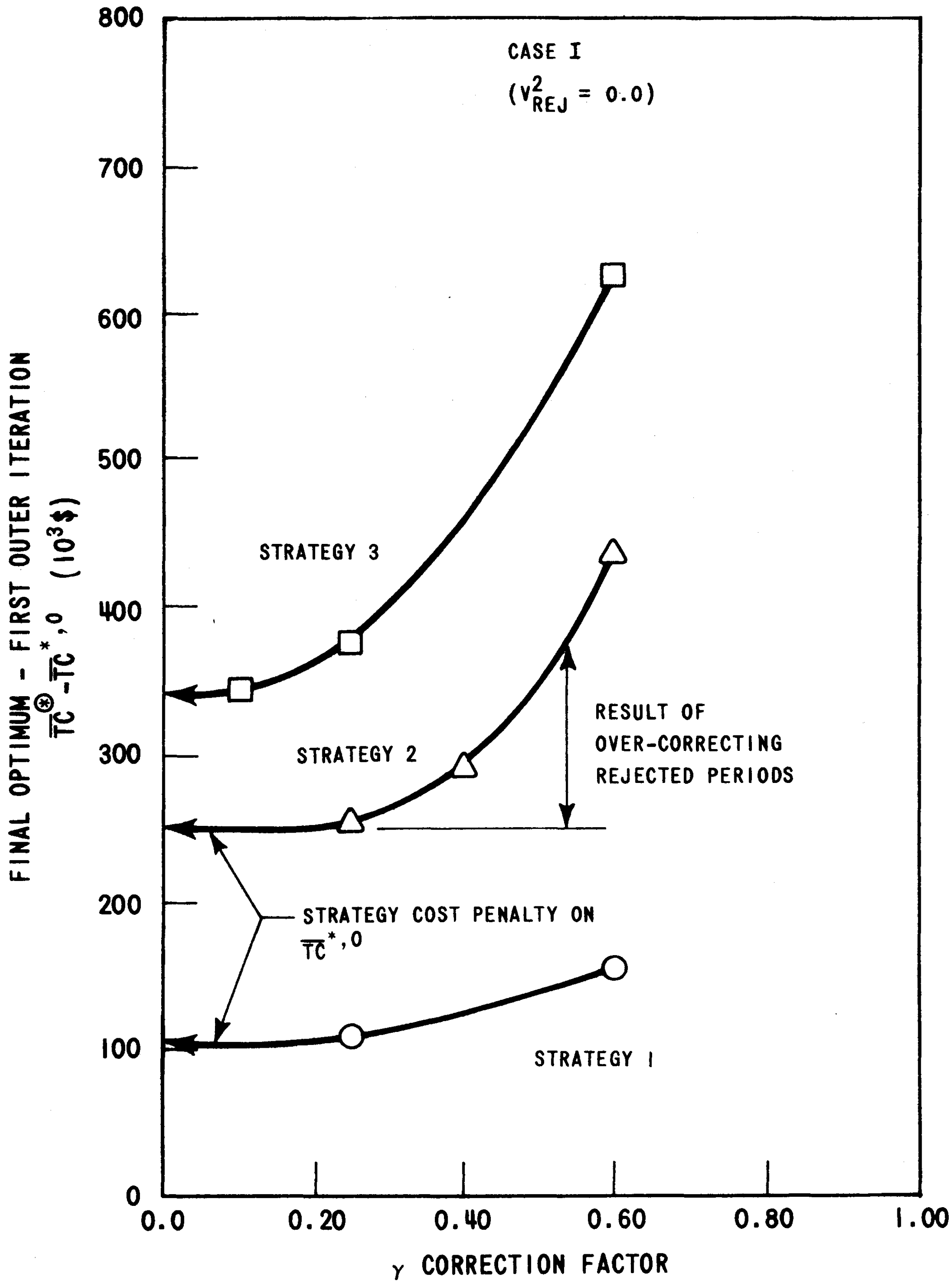


TABLE 5.12

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE I

(72 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, 0.0 Shape Rejection Criterion)
Direct Calculation Using γ = 0.25

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	62	51	49
Average cycle length (months)	12	14.9	15.2
System nuclear capacity factor	0.642	0.656	0.658
		10^6\$ (10^6 MWH)	
Fossil fuel	293.205	276.853	274.082
	(90.068)	(85.836)	(85.196)
Startup-shutdown cost	2.022	1.704	1.650
Emergency purchases	0.655	0.407	0.363
	(0.079)	(0.048)	(0.043)
Nonnuclear production	295.882	278.964	276.095
	(90.147)	(85.884)	(85.239)
Nuclear fuel	294.690	297.709	300.137
	(189.814)	(194.077)	(194.722)
System production	590.572	576.673	576.232
	(279.961)	(279.961)	(279.961)
Fixed firm purchase	133.920	133.920	133.920
	(81.468)	(81.468)	(81.468)
Penalty for short-notice enrichment changes			2.000
System Total	724.492	710.593	712.152
	(361.429)	(361.429)	(361.429)

Two points are worthy of note. First, $\gamma \sim 0.1$ to 0.3 appears optimal since for γ smaller, a larger number of outer iterations (>10) would be required (i.e., slower convergence) while for γ larger, the method over-corrects the offending periods causing an additional cost penalty. Secondly, for scoping purposes only (i.e., when only ORR is required for the comparison of many strategies and the feasibility of $\mathcal{E}^{\circledast}$ is not important for actual production purposes), the additional computations required in attaining an acceptable optimum for each and every run may not be required. (However, if the convergence of SYSOPT is accelerated, the additional shape computations may be easily tolerated in the first place.) Since the strategy versus strategy "cost of feasibility" differences are small (<\$100,000 for S-3 vs. S-2) relative to overall cost differences (~\$1,400,000), a single benchmark run is sufficient for determining the appropriate strategy cost penalty. Adding this to each $\overline{TC}^{*,0}$ eliminates the need for any further outer shape iterations (for scoping purposes only).

The results of Cases II through VI presented in Section 5.7 represent such $\overline{TC}^{*,0}$ solutions (i.e., ignoring all shape considerations). By applying the cost penalties indicated in Figure 5.20, they can be approximately converted to $\overline{TC}^{\circledast}$ (however, $\mathcal{E}^{*,0} \neq \mathcal{E}^{\circledast}$).

5.6.3 Comparison of Theory and Result: Incremental Costs

The analytical discussion of utility system optimization in Section 2.4.2 presented two conclusions:

Conclusion I: The strong conclusion [Equation (2.70)] that all reactor-cycles generating energy during the same time period should be designed at the same incremental cost, and

Conclusion II: The weak conclusion [Equation (2.71)] that all reactor-cycles should simultaneously be designed at the same incremental cost.

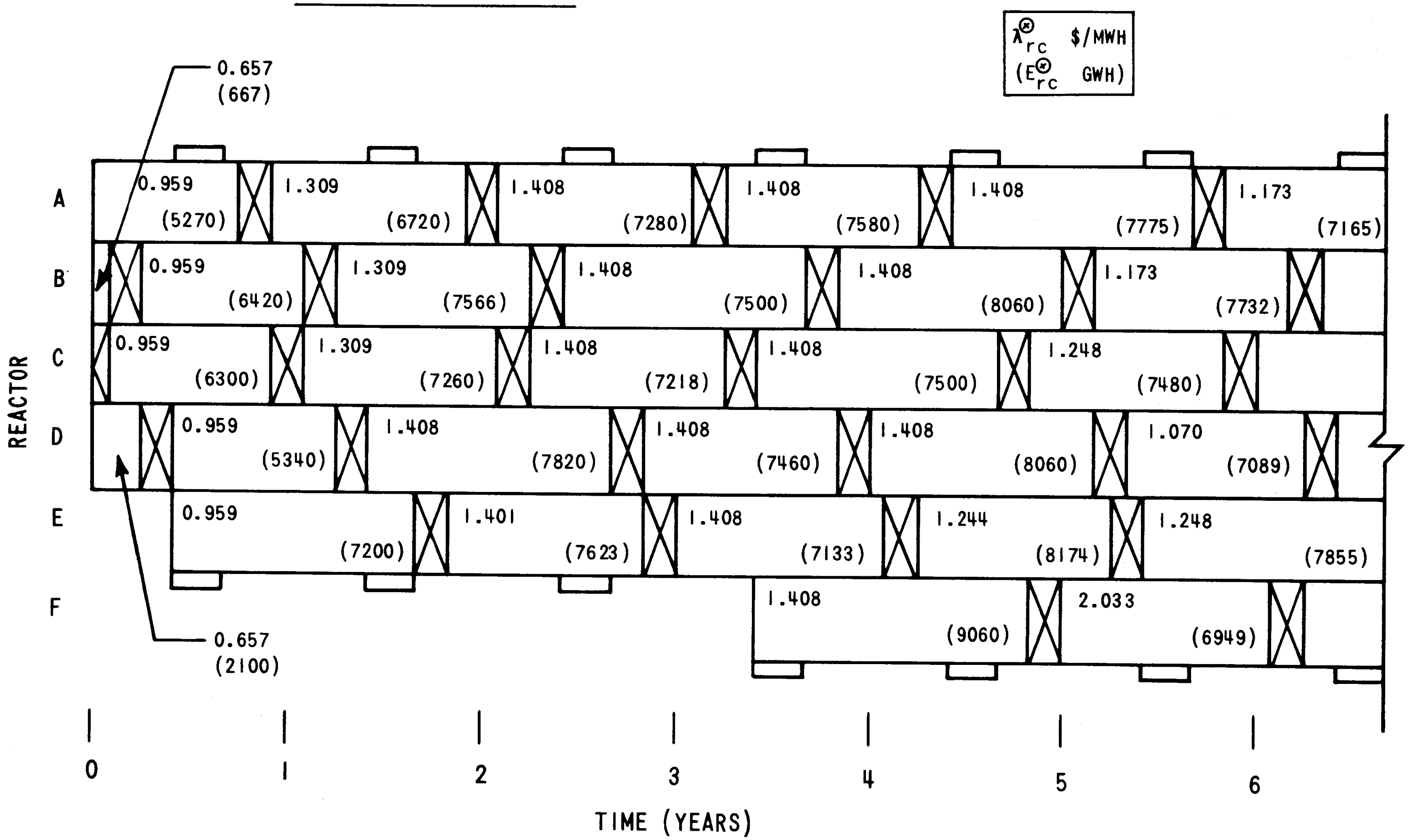
Recall that "strong" and "weak" refer to the number of incremental cost violations anticipated because of over-riding engineering and time constraints.

The $\lambda^{\bullet}_{rc}$ cycle-by-cycle incremental costs at the optimum of Strategy 2 in Case I are presented in Figure 5.21. In analyzing these values, four important points are to be made. First, the general equality of $\lambda^{\bullet}_{rc}$ at each point in time confirms Conclusion I that

$$\lambda_{N_p} = \frac{\partial \overline{TC}_r}{\partial E_{rc}} = \text{constant for all } r \text{ at } \underline{\text{each}}\ p \qquad (5.4)$$

Secondly, incremental costs increase over the first few cycles as the short-range incremental costs of the first year give way to the mid-range incremental costs of later cycles. During the first year, incremental costs are very low because a large proportion of each reactor's cycle costs

Figure 5.21
Incremental Costs and Cycle Energies at Accepted Global Optimum for Strategy 2 in Case I



6253-54

(e.g., separative work, fabrication and reprocessing) are already spent or committed. Discharge burnup is the only variable. Thus, $\lambda^{\bullet}_{r1}$ is Widmer's short-range incremental cost (57, 59). For a cycle further into the future, a larger degree of flexibility is available in the design of the reload batch (size and enrichment) and a larger fraction of total cycle costs can thus be altered. For $c > 2$, $\lambda^{\bullet}_{rc}$ becomes Widmer's mid-range incremental cost (57, 58). Thus, short-range incremental costs evolve into mid-range incremental costs.

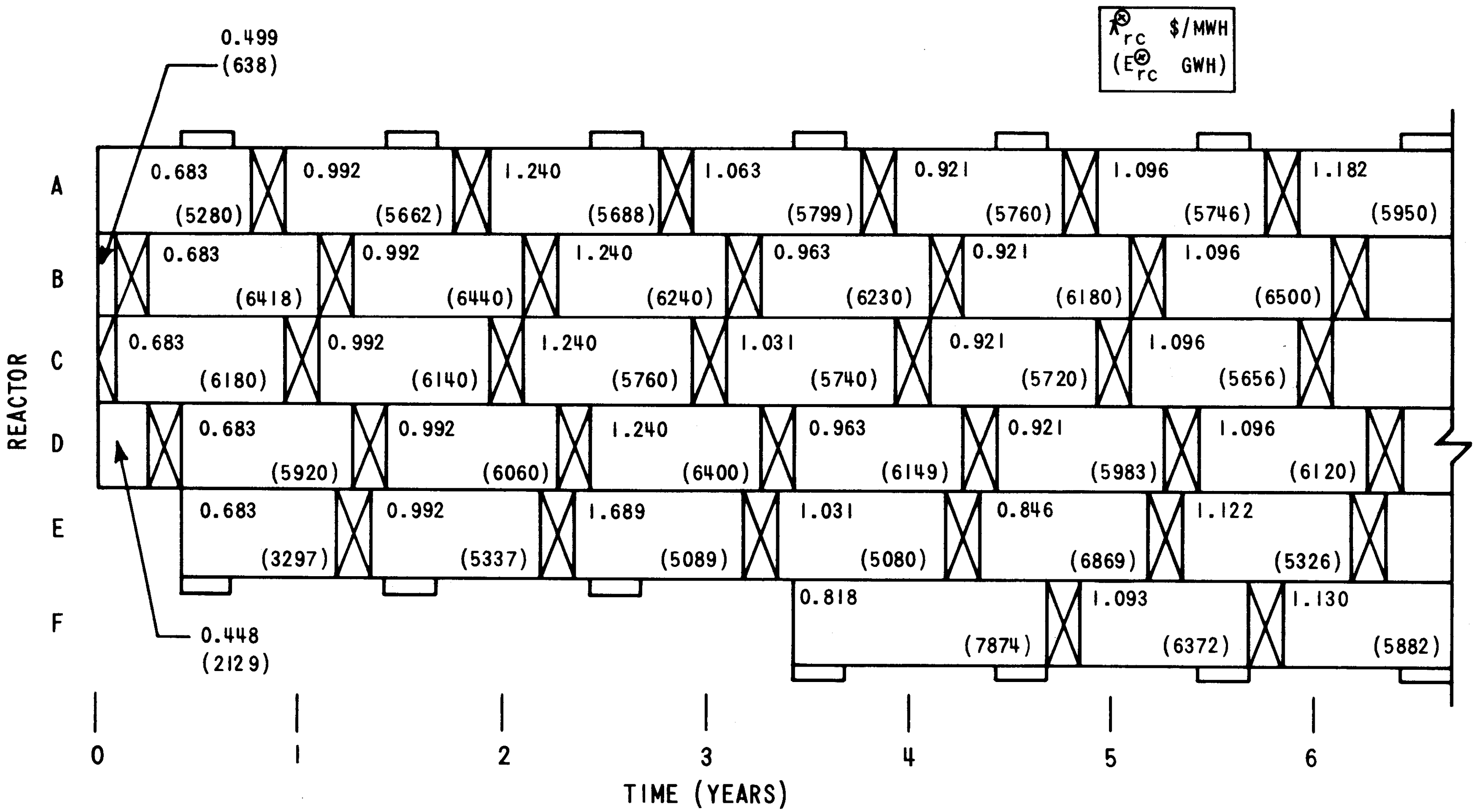
During the middle two to five years of Strategy 2 (see Figure 5.21), the constancy of $\lambda^{\bullet}_{rc}$ for most reactor-cycles provides ample evidence that Conclusion II is also valid.

Finally, the $\lambda^{\bullet}_{rc}$ beyond the fifth year are optimal (but erratic) for the fixed horizon end condition of Section 4.2.3.2. Further investigation into the ideal end condition for each reactor and each strategy are recommended.

Though Figure 5.21 confirmed Conclusion II, the typical $\lambda^{\bullet}_{rc}$ optima of the other strategies did not. For example, Figure 5.22 presents $\lambda^{\bullet}_{rc}$ for Strategy 1 in Case I. Though Conclusion I continues to be valid with few violations, the results do not support Conclusion II.

Figure 5.22

Incremental Costs and Cycle Energies at Accepted Global Optimum for Strategy 1 in Case I



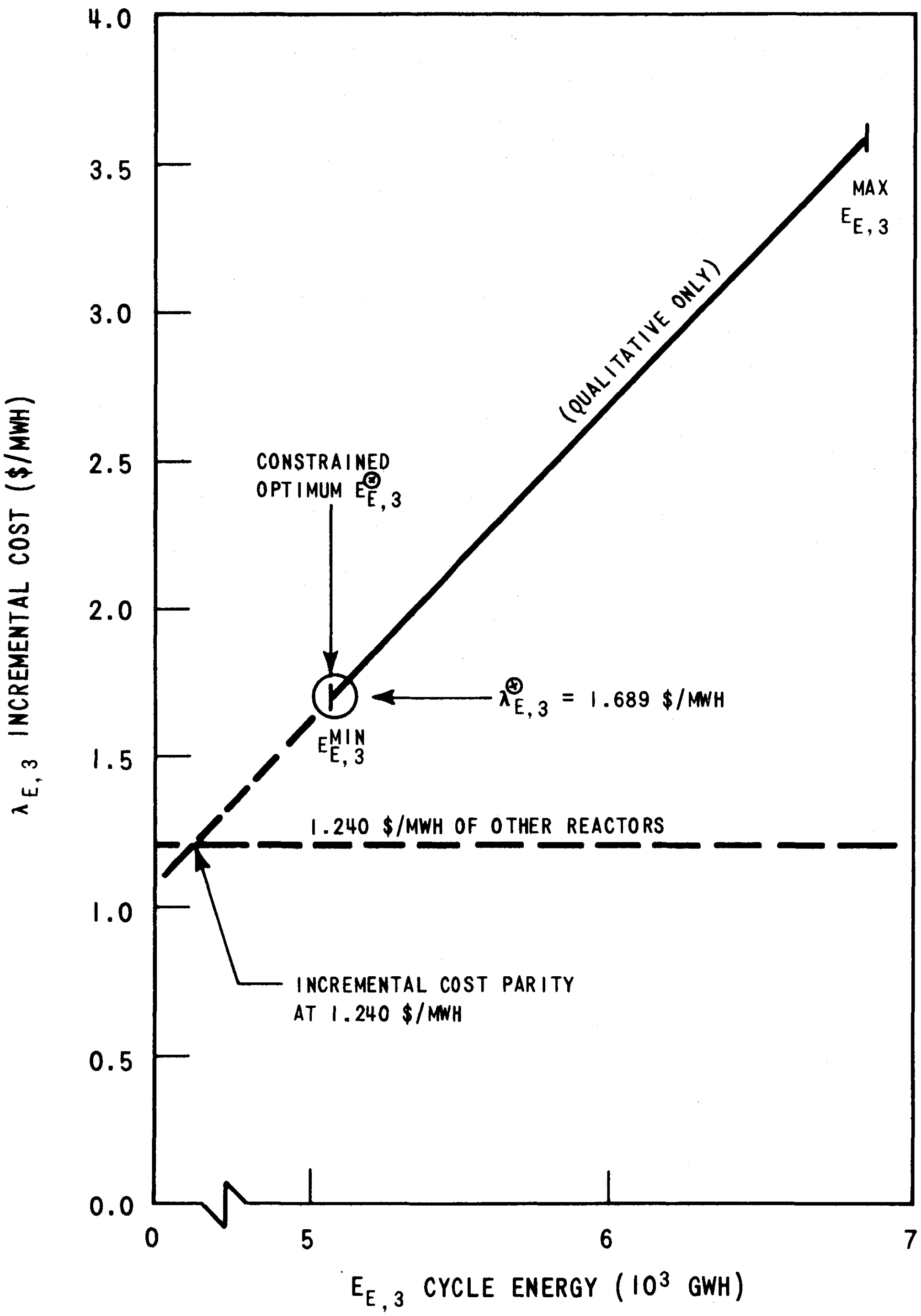
Underlying any discussion of incremental costs is the source of those costs--the CORSOM, or specifically, the QKCORE in-core simulator developed merely to test the SOM. By foregoing an internal optimization, QKCORE is unable to see some obvious means of saving money. For instance, reactor-cycle E-3 of Figure 5.22 has a very high incremental cost due to energy production requiring 4% enriched reload fuel (see Appendix D, Table D.8). Yet, the previous cycle loaded the minimum enrichment allowed (1.5%). If QKCORE allowed early shutdown (reactivity > 0) and optimized the enrichments alone, it might well have loaded 2.5% fuel in E-2, burned only part of the way down and then loaded 3.0% fuel for a complete burn. Indeed, a full-scale CORSOM should be able to optimize reload batch size, as well. The development and incorporation of more versatile CORSOM's is a prerequisite to completing a fully operational nuclear power management model as in Figure 2.21.

Each inconsistency in incremental costs as cycles begin and end, can be translated directly into the optimal loading order (see Figure 5.22). During reactor-cycle E-3 (with $\lambda^{\circledast}_{E,3}$ = 1.689 \$/MWH), Reactor E is loaded only after all other nuclear units (with $\lambda^{\circledast}_{rc}$= 1.240 \$/MWH) are fully loaded. Since E-3 is always loaded last, it generates $E^{min}_{E,3,p}$ during each included period of cycle 3 and, hence, $E_{E,3} = E^{min}_{E,3}$. As Figure 5.23 illustrates, this lower limit on cycle energy prevents E-3 from reaching the cost parity

Figure 5.23

6253-72

Lower Limit on Cycle Energy Preventing
Incremental Cost Parity



of Conclusion I. (If $E_{E,3}$ was less than $E^{min}_{E,3}$, obviously uneconomic fossil energy costing over 2 $/MWH would be substituted for its 1.7 $/MWH energy.)

Reactor-cycle F-1 of Figure 5.22 fails to establish cost parity for the opposite reason. With the initial core configuration assumed fixed, $\lambda^{\circledast}_{F,1}$ is a cheap (0.818 $/MWH) short-range incremental cost. (Cycle burnup is the only design variable.) Thus, Reactor F is always loaded first, generating $E^{max}_{F,1}$ for the cycle. As Figure 5.24 indicates, this upper limit on cycle energy can also prevent incremental cost parity.

The other $\lambda^{\circledast}_{rc}$ inconsistencies of Figures 5.21 and 5.22 are merely more complicated versions of these two simple cases--reactor-cycles E-3 and F-1. In each instance, the optimal economic period loading order is easily deduced: cheapest first.

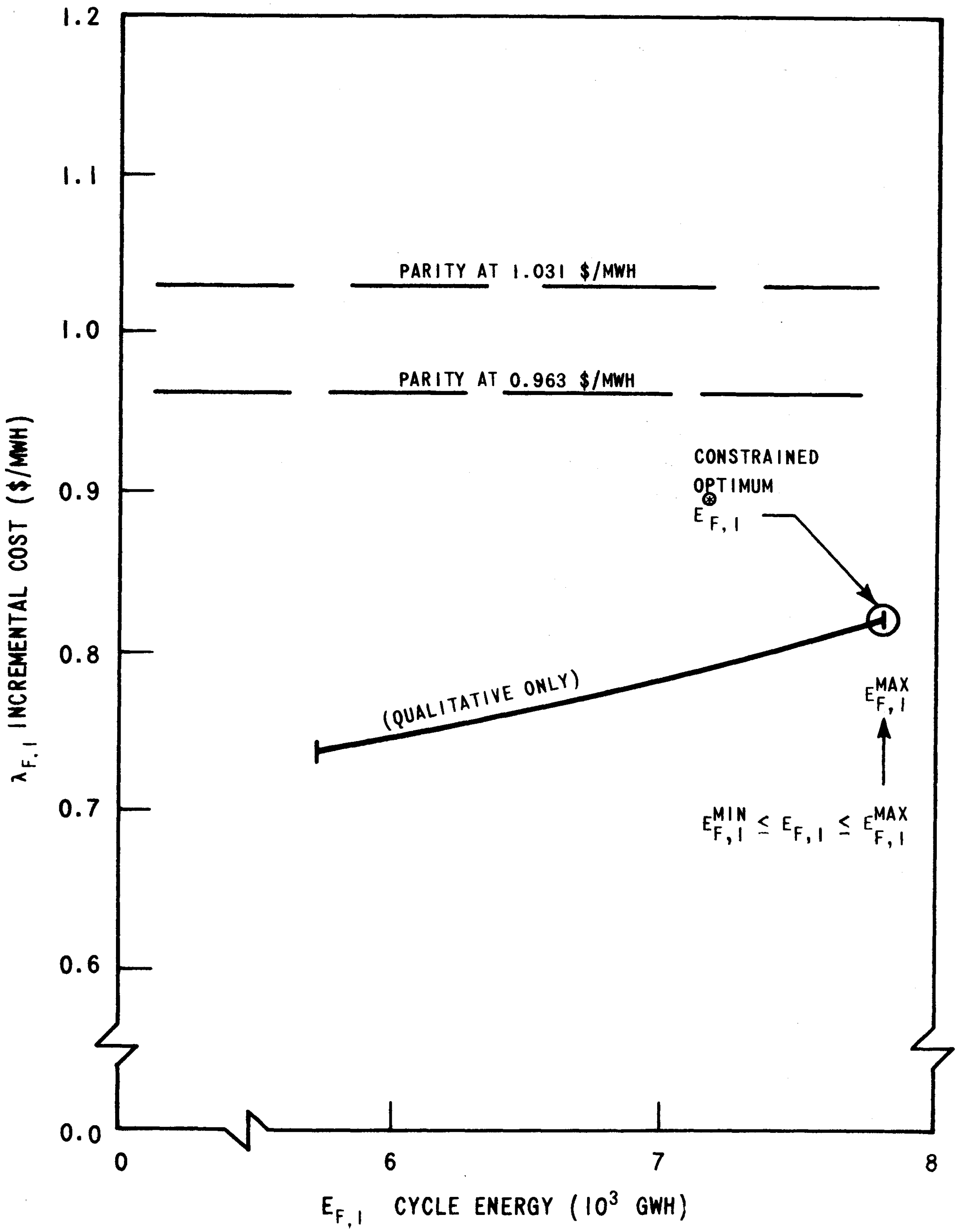
Comparing all reactor-cycles of Figures 5.21 and 5.22, λ_{rc} is seldom greater than 1.41 $/MWH. This observed upper limit on the mid-range incremental cost of nuclear power for an optimized utility system is typical of the individual reactor incremental costs observed by others (41, 55, 57, 58), especially since the Reference nuclear unit cost set (12) is also representative of typical "current" economic parameters.

As Figures 5.3 and 5.4 pointed out, base-loading of the hypothetical utility system's six nuclear reactors is

6253-73

Figure 5.24

Upper Limit on Cycle Energy Preventing Incremental Cost Parity



impossible because the utility's minimum load is too low. However, since λ_N is always much less than λ_F (>2.0 \$/MWH), two possibilities exist for economically utilizing the excess nuclear capacity during the low load periods to decrease system operating revenue requirements. One alternative is to sell excess nuclear capacity (i.e., energy) to neighboring utilities at any price greater than its incremental cost. Incorporation of such nuclear economy interchange sales into the SIM and SOM is recommended since this may well become a common utility practice.

The second option is to use the excess capacity on the utility's own system by operating a pumped-hydro station (see Section 2.2.3). By pumping during low load hours, $\lambda_P = \lambda_N \leq 1.4$ \$/MWH. Using the stored energy for peak-shaving high cost fossil the next day, $\lambda_G = \lambda_F > \sim 4$ \$/MWH. With overall pumped-hydro efficiency typically 67%, total operating revenue requirements are reduced roughly 2 \$/MWH (i.e., 50% of λ_F) for each fossil MWH displaced [Equation (2.29)]. Since such a station is also comparatively cheap to install (See Table 2.1), a pumped-hydro station on the grid of a utility unable to base-load its nuclear capacity produces startling economies (21, 35). "From a utility's viewpoint, pumped storage is a natural fit with large base-load plants. It can take on load instantly, it uses off-peak power to replenish its resources, and its reliability is second to none [5]."

As pumped-hydro stations become more numerous [~ 4400 MW installed versus over 8000 MW under construction in entire United States at end of 1972 (5)], the appropriate planning tools must be developed. Thus, it is highly recommended that pumped-hydro units (and hydro units, as well) be incorporated into the SIM.

5.6.4 Computational Requirements

The computational requirements of SYSINT are detailed in Section 3.6 and Appendix E, while SYSOPT details can be found in Section 4.6 and Appendices F and G. However, Table 5.13 presents a summary of computer usage for Strategy 2 in Case I.

5.7 Evaluation of Competing Strategies

Having discussed the properties of a single optimized strategy, it now becomes appropriate to discuss the broader question of strategy versus strategy comparison. In particular, given the same set of input data (i.e., forecasts), which of the individually optimized strategies represents the optimum plan for operating the utility system? How sensitive is this choice to various parameters in the input? To answer these questions, first the results for Case II will be presented in Section 5.7.1. Later sections will then discuss the other Cases and the optimum strategy choice with respect to horizon length (Section 5.7.2), present value rate (Section 5.7.3), nuclear unit costs

Table 5.13

Computational Requirements For Stragegy 2 in Case I

(Based on IBM 370 model 155 computer operating in MVT environment)

Program	Total Core Storage (Bytes)	CPU Time	Input/ Output Time	Time Units
SYSINT	134 K	2.2	0.5	Sec/period
SYSOPT	246 K with overlay	9	7	Sec/inner iteration
QKCORE	371 K without overlay	13	<1	Sec/inner iteration

(Section 5.7.4) and non-nuclear unit costs (Section 5.7.5).

5.7.1 Comparing Strategies in a Single Case

The optimized results for the three strategies (S-1, S-2 and S-3) in Case II are presented in Table 5.14. Recall from Section 5.3.3 that S-1 is an annual refueling strategy, S-2 a gradual shift to longer cycles and S-3 an immediate shift to longer cycles.

Of prime importance in correlating the results, is the refueling downtime of each strategy. Naturally, the more rapid the shift to longer cycle lengths, the fewer refuelings that must be scheduled.

With less nuclear downtime, the nuclear energy production increases and fossil energy production decreases by approximately the same amount. Also, startup-shutdown cost is decreased as the fossil units move farther away from nightly shutdown. Fewer emergency energy purchases are required due to increased on-line resource margins (see Section 5.3.3).

All three components of non-nuclear production cost thus favor reducing downtime. (By looking at the differences in non-nuclear production cost, average long-term levelized replacement energy costs of 5.2.-5.7 \$/MWH can be calculated.)

As mentioned above, each succeeding strategy is able to increase production because of less refueling downtime. However, the cost of this energy does not increase

TABLE 5.14

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE II

(48 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Strategy	**S-1**	**S-2**	**S-3**
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
		10^6\$ (10^6 MWH)	
Fossil fuel	184.223 (51.703)	176.348 (50.061)	173.250 (49.390)
Startup-shutdown cost	1.497	1.281	1.227
Emergency purchases	0.464 (0.053)	0.317 (0.036)	0.265 (0.030)
Nonnuclear production	186.184 (51.756)	177.946 (50.097)	174.742 (49.420)
Nuclear fuel	198.267 (118.376)	197.189 (120.035)	199.821 (120.712)
System production	384.451 (170.132)	375.135 (170.132)	374.563 (170.132)
Fixed firm purchase	95.166 (54.312)	95.166 (54.312)	95.166 (54.312)
Penalty for short-notice enrichment changes			2.000
System Total	479.617 (224.444)	470.301 (224.444)	471.729 (224.444)

proportionally. In fact, compared to S-1, S-2 generates more nuclear energy for less money! To explain this anomaly, consider the following:

(1) Less downtime means fewer reloads must be purchased.

(2) Increased average cycle length, means increased cycle energy and reload enrichment.

(3) Even with increased batch enrichment cost, the savings due to foregone reloads and the increased energy for amortizing fixed costs, etc., result in a 1.9% decrease in levelized nuclear fuel costs over the four year horizon.

(4) Due to fixed initial conditions and only gradual shift to longer cycles, S-1 and S-2 are very similar in nuclear energy production during the first year. At the end of four years, nuclear production by S-2 is only 1.4% higher. (For longer horizons, the first year matters less and nuclear energy production differences are greater.)

(5) Finally, since the levelized nuclear fuel cost decreases percentagewise more than nuclear production increases, the net result is more nuclear energy for less money.

Turning to S-3, the immediate shift to longer cycles results not only in increased energy production, but also in increased levelized fuel cost. The result is a return to normalcy--more nuclear energy costs more.

Looking then at system production cost over the 48 month horizon, S-3 saves $570,000 over S-2 and roughly ten million dollars over S-1. This, of course, is not enough to absorb S-3's assumed additional two million dollars in penalties for the two short-notice enrichment changes. Thus, among the three strategies, S-2 has minimum total system cost.

During the first four years, then, S-2's gradual shift to longer cycles saves 9.3 million dollars compared to the annual cycles of S-1. Such a savings would clearly justify a few hundred thousand dollars necessary to implement the engineering design changes in the reload fuel specifications. In fact, the savings is large enough to perpetuate S-1's poor showing in all six Cases of the input parameters (see Table 5.8 and Appendix D). (Strategy 2 is always cheaper by at least 6.7 million dollars.)

However, S-2 and S-3 are roughly competitive depending on the magnitude of the enrichment change penalty. Without the penalty S-3 is favored by roughly $600,000.[1] But after the 2 million dollar penalty, it is 1.4 million dollars more costly. This competitiveness is used to advantage in the following sections where the sensitivity study is presented as a comparison of S-2 vs. S-3 directly (i.e., without

---

[1]Of this $600,000, roughly $95,000 could also be saved by S-2 were it allowed to freely change initial enrichment for Reactors B and D.

any penalty) and with penalties of a half or one million dollars per change.

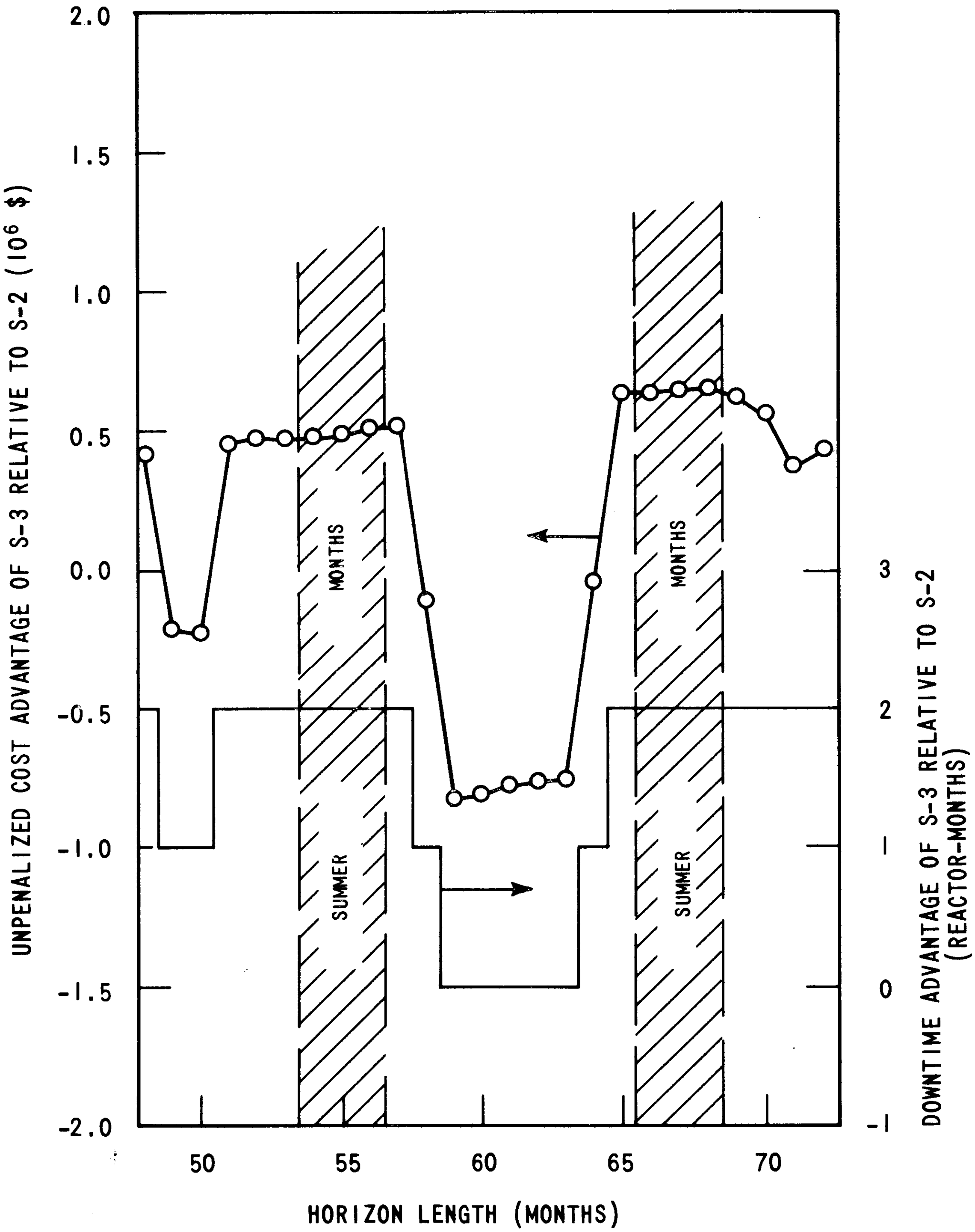
5.7.2 Sensitivity to Horizon Length

Ideally, a management planning tool should yield consistent results whether the planning horizon is taken to be four, five or six years into the future. To test this aspect of the model, the results in Figure 5.25 were produced using the Case I (see Table 5.8) detailed optimized solutions for Strategies 2 (see Figure 5.6) and 3 (see Figure 5.7). However, the operating revenue requirement summation [Equation (2.17)] for the 72 months covered by the horizon of Case I was only carried up to and including the horizon indicated on the abscissa (enrichment change penalties were not included). The disturbing oscillatory nature of the comparison is almost identically matched by the shifts in downtime advantages which are also presented. In a particular period, if an additional reactor is down for refueling in Strategy S-3, then S-3 will lose a reactor-month of downtime advantage. More importantly each nuclear MWH foregone must be made up with fossil replacement energy. Thus, each month of downtime means roughly 300 GWH (discounted) of short-term replacement energy at 4.0 \$/MWH versus nuclear average costs of 2.0 \$/MWH. The net result: each reactor-month of downtime five years in the future costs roughly \$600,000.

Figure 5.25

6253-74

Cost and Downtime Advantage of S-3 versus S-2 as Function of Horizon Length

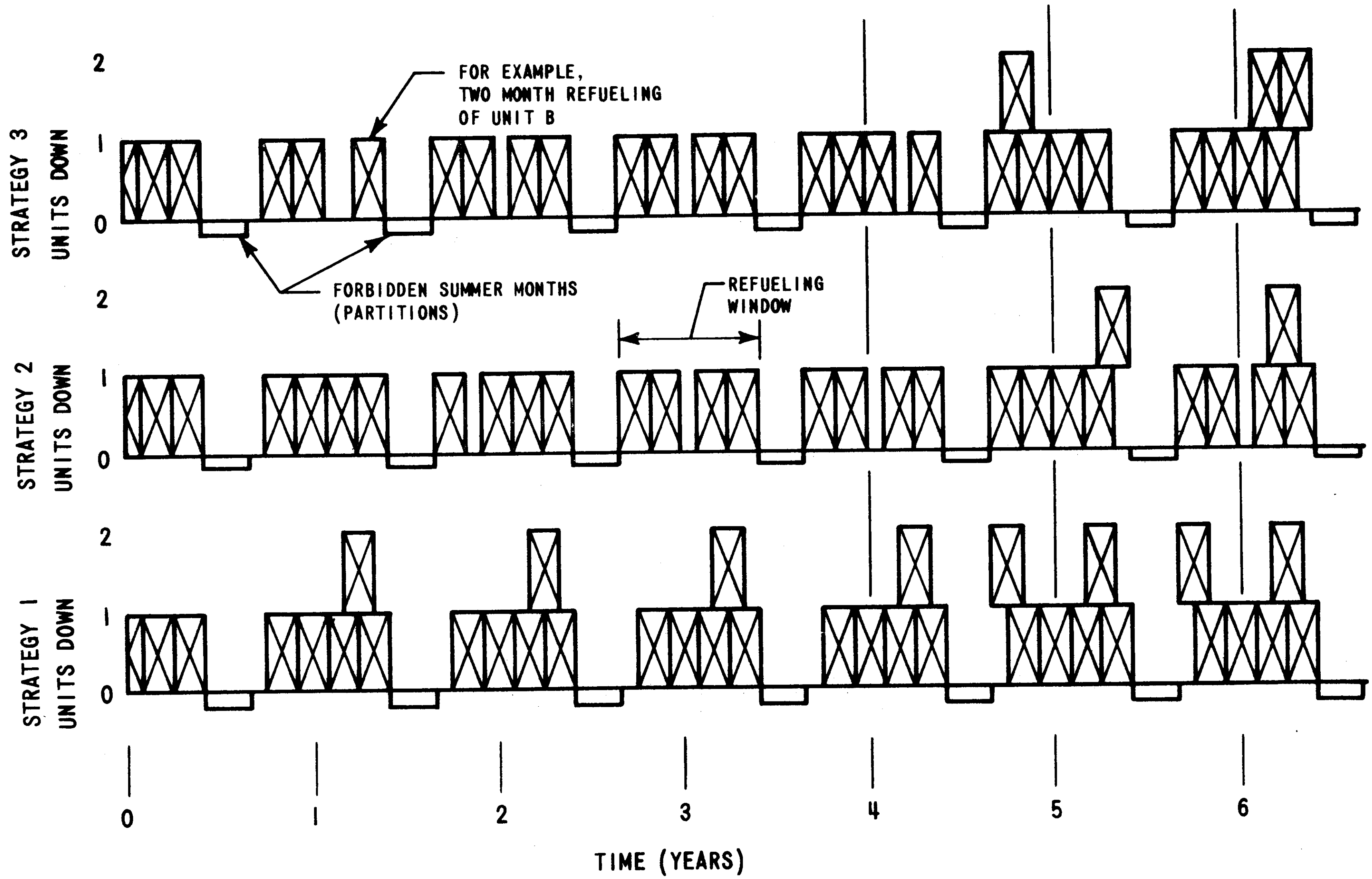


The next question is "What causes these shifts in downtime advantage?" The answer is given in Figure 5.26, a composite of the two month refueling outages in each strategy presented in Figures 5.5 to 5.7. [Note the regularity of S-1's annual refuelings and the fact that every refueling window involves at least two months of simultaneous or "stacked" refuelings. S-2 and S-3, by selectively skipping over a window with different reactors (see Section 5.3.3), are able to avoid simultaneous refuelings until the fifth year.] S-3's two reactor-month downtime advantage at 48 months can be pin-pointed as actually occurring during the first full window of the first year when S-3's immmediate shift to longer cycles dictated immediately skipping a summer. Further note that although the four year horizon ends exactly after a refueling for both S-2 and S-3, S-2 shifts the next refueling back one month. This causes the temporary one reactor-month shift in downtime advantage just after four years.

At the six year horizon, shown on Figure 5.26, note both S-2 and S-3 conveniently terminate exactly after a refueling. Now consider the relative position of their simultaneous refueling with respect to a five year horizon. In S-3, it occurs before the five year cutoff, but in S-2, it is postponed until just before the summer. The window, as a whole, involves no shift in downtime advantages, but if the horizon occurs within the window (e.g., 5 year horizon) an anomalous one million dollar added advantage may

Figure 5.26

Strategy Composite



6253-75

accrue to S-2. Since no refuelings occur during the summer and, in fact, the summers represent the partitions between the windows, it is recommended that a single horizon coinciding with one of these partitions be chosen. Note that if the horizon occurs in any of the six summer months appearing in Figure 5.25, S-3 is cheaper by roughly $700,000 (if no enrichment change penalty is applied).

In the absence of utility refueling constraints (e.g., no refuelings in summer) that create the computationally convenient windows and partitions, a single, long horizon could still be calculated in detail. However, prudence would dictate developing shorter horizon results such as those in Figure 5.25 to permit a more intelligent evaluation of strategy cost differences.

Though the above horizon-at-partition conclusion is presented with verification, a solid conclusion concering <u>which</u> partition must await the second generation nuclear power management model possessing detailed CORSOM's. As an interim rule of thumb, intuition suggests that the horizon ought to include a complete core of freely specified enrichments for each reactor. In other words, the horizon should be far enough into the future to predict completely the discharge characteristics of the next reload enrichment to be finalized (i.e., actually ordered from vendor) for each reactor.

In summary, choice of a proper horizon is imperative, but not difficult. If the worst comes to the worst, a long horizon evaluated per Figure 5.25 would always be valid and helpful. In any event, for planning horizons on the order of five or six years, differences in total system cost under a few hundred thousand dollars are best viewed as insignificant (see Section 5.8.5). Such dilemmas ought to be reconciled based on other criteria--e.g., the most flexible, the easiest to implement or the most reliable strategy.

5.7.3 Sensitivity to Present Value Rate

The optimized results for the three Cases with different present value rates are presented in Table 5.15 for Case III (0%), Table 5.16 for Case II (7%) and Table 5.17 for Case IV (12%).

By recognizing three general cost components of each strategy, much insight can be gained. They are (1) all fossil fuel related costs, (2) direct nuclear outlays and (3) carrying charges on the nuclear outlays. At a 7% present value rate, nuclear carrying charges are ~ 25% of nuclear outlays while fossil carrying charges are relatively insignificant.

As the present value rate increases, the revenue requirements for (1) and (2) decrease slowly while those for component (3) rise sharply. The result is that as the present value rate increases, the heavier a strategy's reliance on nuclear energy, the less advantageous that strategy becomes. The optimum choice may not change, but

TABLE 5.15

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE III

(48 Month Horizon, 0% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
		10^6\$ (10^6 MWH)	
Fossil fuel	212.434	203.326	199.928
	(51.703)	(50.061)	(49.390)
Startup-shutdown cost	1.684	1.430	1.373
Emergency purchases	0.528	0.355	0.299
	(0.053)	(0.036)	(0.030)
Nonnuclear production	214.646	205.111	201.600
	(51.756)	(50.097)	(49.420)
Nuclear fuel	158.416	153.987	154.678
	(118.376)	(120.035)	(120.712)
System production	373.062	359.098	356.278
	(170.132)	(170.132)	(170.132)
Fixed firm purchase	108.624	108.624	108.624
	(54.312)	(54.312)	(54.312)
Penalty for short-notice enrichment changes			2.000
System Total	481.686	467.722	466.902
	(224.444)	(224.444)	(224.444)

TABLE 5.16

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE II

(48 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
	10^6\$ (10^6 MWH)		
Fossil fuel	184.223 (51.703)	176.348 (50.061)	173.250 (49.390)
Startup-shutdown cost	1.497	1.281	1.227
Emergency purchases	0.464 (0.053)	0.317 (0.036)	0.265 (0.030)
Nonnuclear production	186.184 (51.756)	177.946 (50.097)	174.742 (49.420)
Nuclear fuel	198.267 (118.376)	197.189 (120.035)	199.821 (120.712)
System production	384.451 (170.132)	375.135 (170.132)	374.563 (170.132)
Fixed firm purchase	95.166 (54.312)	95.166 (54.312)	95.166 (54.312)
Penalty for short-notice enrichment changes			2.000
System Total	479.617 (224.444)	470.301 (224.444)	471.729 (224.444)

TABLE 5.17

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE IV

(48 Month Horizon, 12% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Strategy	**S-1**	**S-2**	**S-3**
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
		$10^6$$ (10^6 MWH)	
Fossil fuel	167.908 (51.703)	160.762 (50.061)	157.850 (49.390)
Startup-shutdown cost	1.388	1.194	1.142
Emergency purchases	0.427 (0.053)	0.294 (0.036)	0.245 (0.030)
Nonnuclear production	169.723 (51.756)	162.250 (50.097)	159.237 (49.420)
Nuclear fuel	220.395 (118.376)	221.107 (120.035)	224.731 (120.712)
System production	390.118 (170.132)	383.357 (170.132)	383.968 (170.132)
Fixed firm purchase	87.340 (54.312)	87.340 (54.312)	87.340 (54.312)
Penalty for short-notice enrichment changes			2.000
System Total	477.458 (224.444)	470.697 (224.444)	473.308 (224.444)

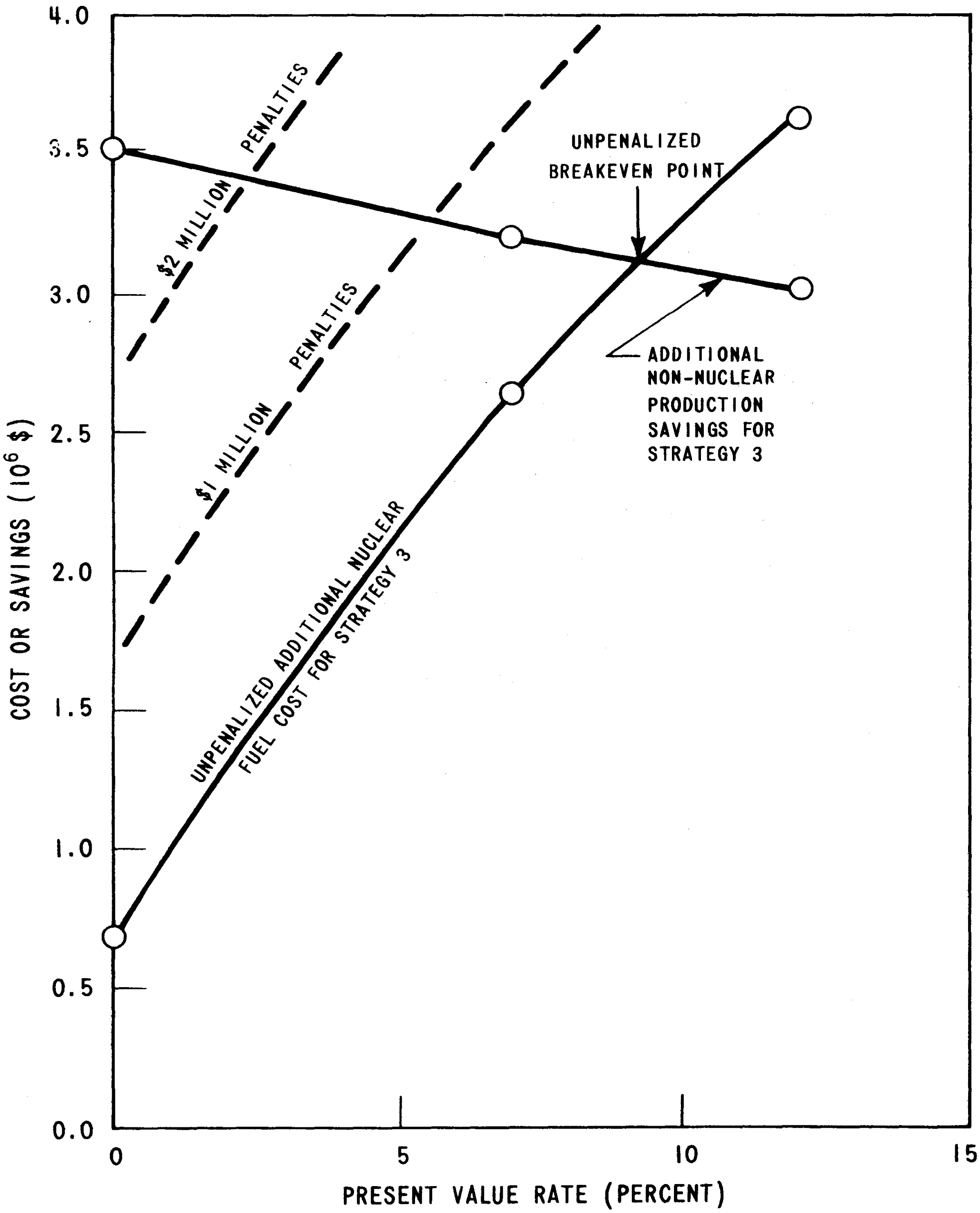
the advantage will decrease. For example, comparing S-1 (the annual strategy) and S-2 (the gradual shift to longer cycles), S-2 is always favored but the savings decreases from 14.0 to 6.7 million dollars as the rate goes from 0 to 12 per cent.

To investigate such changes in more detail, Figure 5.27 presents a cost comparison of S-2 (gradual shift) and S-3 (immediate shift) for the three rates involved. S-3 uses more nuclear energy and less fossil. Therefore, it possesses a non-nuclear savings of 3.5 million dollars at 0 per cent. However, as a result of nuclear carrying charges, S-3's added nuclear cost increases six times as fast as the fossil advantage itself decreases! On an unpenalized basis, S-3 is the optimum at a 7% present value rate, but S-2 is optimum at 12 per cent. The break-even point is 9-1/4 per cent. Naturally, the higher the penalty, the more S-3 must have saved prior to applying the penalty. The result: one million dollars in penalties breaks even at 5-1/2% while two million requires 2-1/4%. With any reasonable penalty and present value rate, S-2 is clearly optimum over both S-1 and S-3.

An interesting question is now posed: Suppose a mythical fourth strategy differed from S-2 by only $500,000. What size error in forecasting the present value rate would completely mask this difference? Using the slope from Figure 5.27, an error of approximately 1-3/4% in the present

Figure 5.27

Non-Nuclear Savings and Nuclear Cost for S-3 versus S-2 as Function of Present Value Rate



value rate would shift the total cost advantage $500,000. Such a forecasting error is not altogether improbable. Thus, as standard practice, all near optimal policies should be evaluated and ranked at several additional present value rates (say, the nominal ± 2%), not at the nominal rate alone. In this manner, strategies extremely sensitive to the present value rate may be eliminated.

In the above recommendation, note the word "evaluated", not "re-optimized". All of the results quoted in this Section are for re-optimized solutions using the specified present value rate. Practically speaking, the computer expense of re-optimizing the Case II solutions was not necessary. Re-optimization saved less than $90,000 each on five out of the six cases involved [S-3 saved $275,000 if there was no time value of money (0%)].

5.7.4 Sensitivity to Nuclear Unit Costs

The optimized results for the cases involving Low, Reference, and High nuclear unit costs (see Table 5.7) are presented in Table 5.18 for Case V (Low), Table 5.19 for Case II (Reference) and Table 5.20 for Case VI (High). From a total cost standpoint, S-2 remained the optimum choice. The trends in the S-2 vs. S-3 comparison are portrayed in Figure 5.28.

Of course, variations in nuclear costs do not affect S-3's 3.2 million dollar fossil savings. But S-3's increased nuclear energy does result in increased separative

TABLE 5.18

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE V

(48 Month Horizon, 7% P.V. Rate, Low Nuclear Unit Costs, No Shape Constraints)

Strategy	**S-1**	**S-2**	**S-3**
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
		10^6\$ (10^6 MWH)	
Fossil fuel	184.223 (51.703)	176.348 (50.061)	173.250 (49.390)
Startup-shutdown cost	1.497	1.281	1.227
Emergency purchases	0.464 (0.053)	0.317 (0.036)	0.265 (0.030)
Nonnuclear production	186.184 (51.756)	177.946 (50.097)	174.742 (49.420)
Nuclear fuel	141.229 (118.376)	141.156 (120.035)	143.463 (120.712)
System production	327.413 (170.132)	319.102 (170.132)	318.205 (170.132)
Fixed firm purchase	95.166 (54.312)	95.166 (54.312)	95.166 (54.312)
Penalty for short-notice enrichment changes			2.000
System Total	422.579 (224.444)	414.268 (224.444)	415.371 (224.444)

TABLE 5.19

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE II

(48 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Strategy	**S-1**	**S-2**	**S-3**
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
		10^6\$ **(10^6 MWH)**	
Fossil fuel	184.223 (51.703)	176.348 (50.061)	173.250 (49.390)
Startup-shutdown cost	1.497	1.281	1.227
Emergency purchases	0.464 (0.053)	0.317 (0.036)	0.265 (0.030)
Nonnuclear production	186.184 (51.756)	177.946 (50.097)	174.742 (49.420)
Nuclear fuel	198.267 (118.376)	197.189 (120.035)	199.821 (120.712)
System production	384.451 (170.132)	375.135 (170.132)	374.563 (170.132)
Fixed firm purchase	95.166 (54.312)	95.166 (54.312)	95.166 (54.312)
Penalty for short-notice enrichment changes			2.000
System Total	479.617 (224.444)	470.301 (224.444)	471.729 (224.444)

TABLE 5.20

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE VI

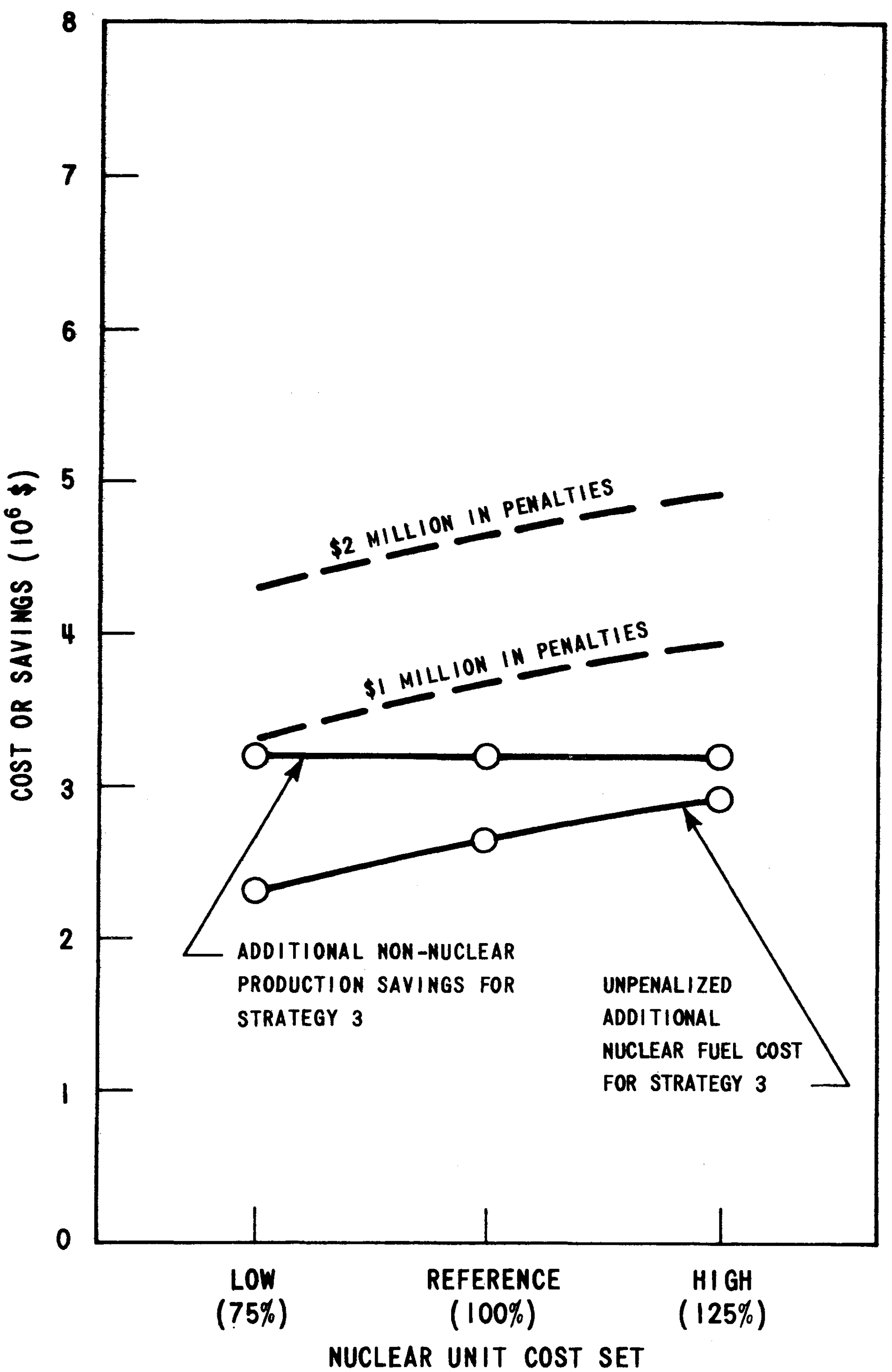
(48 Month Horizon, 7% P.V. Rate, High Nuclear Unit Costs, No Shape Constraints)

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
		10^6\$ (10^6 MWH)	
Fossil fuel	184.223 (51.703)	176.348 (50.061)	173.250 (49.390)
Startup-shutdown cost	1.497	1.281	1.227
Emergency purchases	0.464 (0.053)	0.317 (0.036)	0.265 (0.030)
Nonnuclear production	186.184 (51.756)	177.946 (50.097)	174.742 (49.420)
Nuclear fuel	255.223 (118.376)	253.211 (120.035)	256.169 (120.712)
System production	441.407 (170.132)	431.157 (170.132)	430.911 (170.132)
Fixed firm purchase	95.166 (54.312)	95.166 (54.312)	95.166 (54.312)
Penalty for short-notice enrichment changes			2.000
System Total	536.573 (224.444)	526.323 (224.444)	528.077 (224.444)

Figure 5.28

6253-77

Non-Nuclear Savings and Nuclear Cost for S-3 versus S-2 as Function of Nuclear Unit Costs



work requirements. These, in turn, cause S-3 to suffer a larger disadvantage as unit costs increase. Unpenalized, S-3 is able to maintain at least a $300,000 advantage in the entire range investigated. However, even one million dollars in penalties turns the choice around for the same range.

As for the forecasting error that results in $500,000 closer competition, a 40% change in Reference nuclear unit costs is required. This would appear to border on the improbable. However, the characteristics of the six PWR reactors comprising the hypothetical utility are so similar, that generalizations to all types of nuclear reactors are impossible. A utility possessing a broad mix of reactor types (PWR, BWR, HTGR, LMFBR, GCFR, etc.) and sizes would very likely find that small shifts within various unit cost components would alter the reactor loading order. For instance, rising plutonium value decreases LWR fuel costs as a credit, but increases LMFBR fuel costs. Such an investigation is clearly beyond the scope of the current nuclear power management model because of QKCORE's inherent limitations (see Section 5.2). In the future, this may well be the most interesting investigation of all.

A word about re-optimizing the Case II solutions is again in order. With the qualifications just mentioned regarding other reactor types, re-optimization, though performed, was not necessary. Since the reactors were nearly identical, energy was not re-optimized significantly. The nuclear cost was merely re-evaluated. The average cost savings for each of the six perturbed solutions was less than $15,000.

5.7.5 Sensitivy to Non-Nuclear Costs

To evaluate the non-nuclear cost components, the results of Case II in Table 5.14 are used. Since the non-nuclear cost components only affect SYSINT results directly, parameterization of these costs did not require further SYSOPT runs.

Cursory examination of Table 5.15 indicates immediately that startup-shutdown cost and emergency power purchases do not vary by more than $300,000 from strategy to strategy. On the other hand, fossil fuel cost can vary by 10 million dollars or more. On account of their relative size and absolute size with respect to various forecasting and core modeling errors, the comparison is more convenient if all non-nuclear components are lumped together. The obvious parameter is ϕ_F cents per MegaBTU for fossil fuel. If this were to increase, startup-shutdown costs would increase proportionally since the major cost component is incurred due to sensible heat requirements during startup (see Figure 2.6). Emergency power purchases should also be proportional to fossil fuel cost if the neighbor supplying the energy relies on fossil fueled equipment to generate it.

With these assumptions, Figure 5.29 is presented indicating breakeven points for S-2 (gradual shift) versus S-3 (immediate shift) as a function of fossil fuel cost. The higher the cost of fossil fuel, the larger the fossil savings of S-3 and the larger penalty it can successfully absorb. Unpenalized, S-3 breaks even at 33¢/MegaBTU. Each one million dollars in penalties requires another 12-1/2¢/MegaBTU. Thus, with any reasonable penalty, S-2 is again the optimum.

More importantly, note the forecasting error required to equalize a $500,000 difference--merely 6-1/4¢/MegaBTU. Given the realities of today's fossil fuel marketplace and the environmental concern, forecasting fossil fuel costs five or six years into the future within 6¢/MegaBTU is a near impossible task. This forecast very likely could turn out to be the critical item in the overall model input. The models of interfuel competition currently under development in many institutions [e.g., (11)] may aid in pinpointing, or at least bracketing more closely, the future trends in fossil fuel costs.

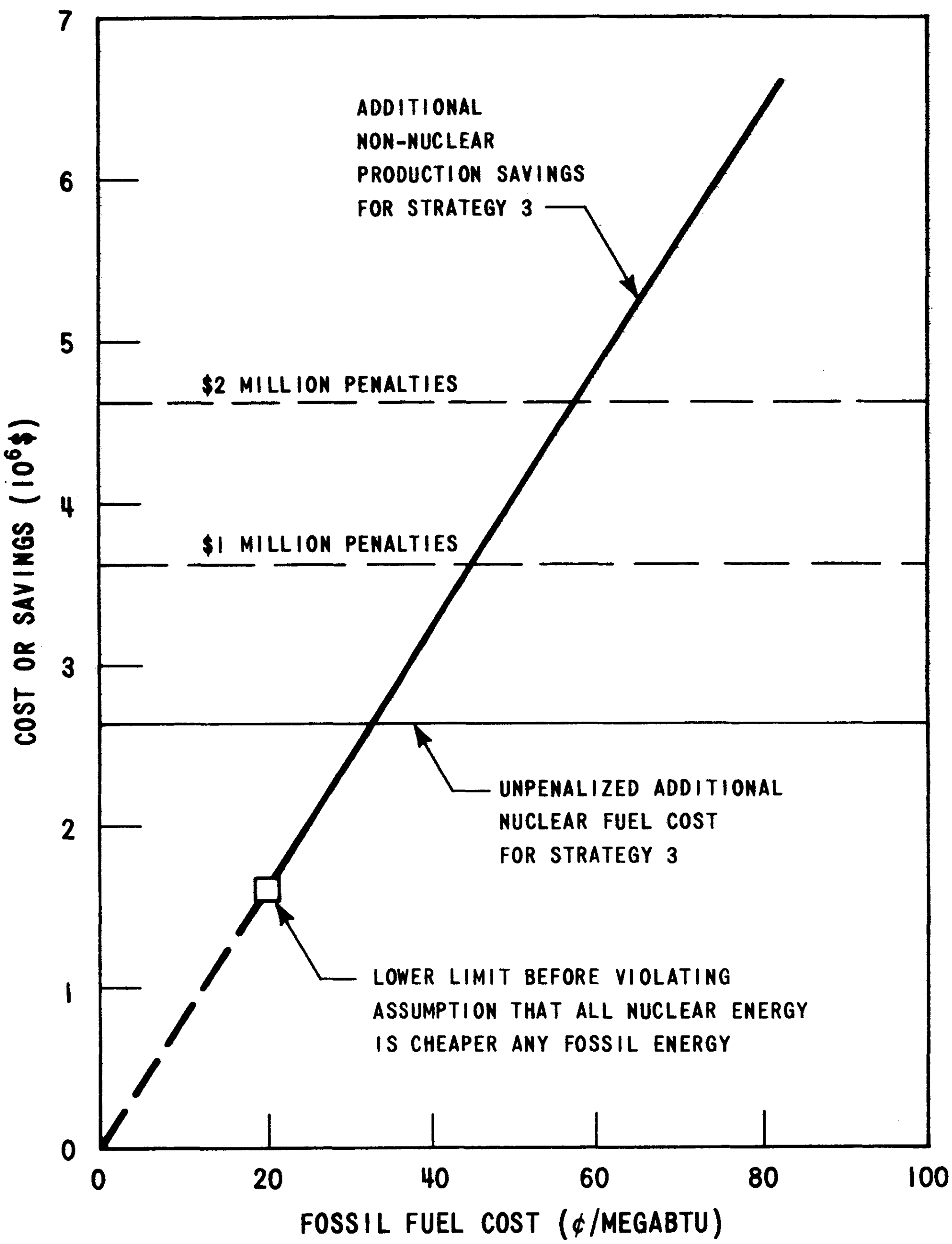
In short, fossil fuel thermal energy cost appears to be one of the critical input data.

5.8 Critical Questions Revisited

Section 5.1 posed six critical questions pertinent to the development of any management planning tool. The following sections provide a summary of their answers as they apply

Figure 5.29

Non-Nuclear Savings and Nuclear Cost for S-3 versus S-2 as Function of Fossil Fuel Cost



to the current nuclear power management multi-year model and, in particular, to the SIM and SOM developed in this work.

5.8.1 To What Problems is the Model Applicable?

The complete model of Figure 2.21 applies to the multi-year management of utility systems possessing any types and amounts of fossil, nuclear, hydro and pumped-hydro equipment. As implemented in the SYSINT and SYSOPT computer models of the SIM and SOM, respectively, only fossil and nuclear equipment are currently permitted. Addition of the other two types should receive a high priority. A computerized RAMM should be interfaced with the models to permit the investigation of many strategies. Development of detailed CORSOM's for each reactor type are required to replace the limited test simulator QKCORE.

Once these improvements have been made, the scope of the problems the model could analyze are almost numberless. Input to the model consists of forecasts, operating constraints, initial conditions, unit costs, etc. The optimized outputs include period production schedules, fossil maintenance-nuclear refueling schedules and nuclear reload parameters. The combination and permutations of altered inputs affecting outputs generates an enormous number of possibilities.

5.8.2 <u>What Assumptions are Required?</u>

Though the current computerized version of the nuclear power management model contains several simplifying assumptions, only one of the assumptions is actually inherent in the model of Figure 2.21. The others, enumerated below, could be relaxed by reprogramming the affected portions.

The pivotal assumption involves the permanent relationship between nuclear and fossil fuel costs. Namely, nuclear incremental costs are sufficiently less than even the best fossil incremental costs, that for the foreseeable future, nuclear energy will be utilized so as to displace as much fossil energy as possible. This maximization of nuclear energy dictates the SIM's loading order segregation into must-run fossil minimums, nuclears and remaining fossils (see Section 3.2) regardless of intra-nuclear cost differences. The SOM then minimizes the cost of producing this nuclear energy.

The SOM's inner iterative procedure involves passing cycle energy vectors to the CORSOM's and receiving cost information as a feedback loop to test for convergence and determine the cycle energy vectors for the next iteration. If the key assumption were to be relaxed or should it become invalid due to unforeseen price shifts, the termination of the feedback loop would have to be shifted to the SIM.[2] For then, changes in nuclear incremental costs would

---

[2]The ORSIM model, currently under development at Oak Ridge National Laboratory (<u>14</u>), is of this more general type.

also alter the fossil-nuclear competition (i.e., loading order), resulting in varying amounts of fossil energy and fossil cost at each iteration. The objective function in the SOM would become the total system cost directly, not merely the nuclear cost as at present.

Though the nuclear-vs.-fossil cost assumption does restrict the model's generality, the prospects of violating it are low and the computational savings may be significant.

The following additional assumptions were made in order to simplify programming the models:

(1) At time zero, none of the nuclear cores is so depleted as to represent a scarce resource. When further development enables the SIM to handle scarce resource hydro units, this assumption may be relaxed by treating energy-short nuclear plants similarly.

(2) All forecasts (even six years into the future) are 100% accurate (i.e., a deterministic future). As recommended in Section 5.3.1, much work needs to be done in this area with regard to confidence limits on the various results.

(3) For such a non-expansion planning model, only operating costs need be included in the objective function since capital costs and related carrying charges are already fixed by the additions and

retirements specified and held constant for all strategies (see Section 2.1.3). The addition of these and other cost components to the model would complete a useful tool for multi-year or longer planning.

(4) The incremental heat rate of each nuclear plant was assumed constant by the SOM over the operating range of interest. As Section 2.4.2 pointed out, proprietary data on today's PWR's and BWR's confirm the assumption. Future plant types, as well as newer generations of the above, may force re-evaluation of this assumption.

(5) The utility system contains enough must-run fossil equipment to provide sufficient spinning reserves to permit all nuclear upper increment capacity (group 3 in Figure 3.8) to be scheduled as a single, continuous block of capacity. In other words, spinning reserve requirements do not make it necessary to mix groups 3 and 4 (remaining fossil capacity). This condition appears likely to prevail for many years, i.e., as long as the system contains large fossil units that cannot be shut-down and then started up readily and reliably.

(6) All incremental cost curves are continuous and monotonically increasing. All data produced by the simple QKCORE model bore out this assumption.

Such behavior assures convexity of the SOM's operating cost objective function and permits the use of a standard NP optimization package.

(7) Finally, all nuclear minimums are base-loaded. One implied result is that there are no nuclear startup-shutdowns. In addition, this assumption coupled with assumption (4) allows the analytical simplifications that lead to Equation (2.52) relating thermal and electrical energy directly. This same simplification facilitates the interfacing of the SIM and SOM, but, as with the other six simplifications, it could be relaxed.

As for recommendations concerning further development, numbers (1) and (2) ought to have high priority; (3) through (5), medium priority; (6) and (7), low priority.

5.8.3 Does the Method Converge to an Optimum?

As the discussion in Section 5.6.1 pointed out, the inner iteration on system cost did converge. Considering the other errors inherent in the models (see Section 5.8.5), convergence can be called complete. Convergence was, however, slow. This prompted the recommendation to study the problem further.

Convergence of the outer shape iterations (see Section 5.6.2) was obtained with only slight increases in predicted system total cost. However, outer convergence was also slow.

Increasing the amount of piecewise-linearization would aid both the inner and outer convergence rates.

5.8.4 Is it the Global Optimum?

Globality hinges on two key issues:

(1) Was the globally optimal strategy even included as a possible alternative?

(2) Did the SOM achieve the minimum system cost for each and every strategy that was evaluated?

The answer to the first question depends on the completeness of the RAMM. As for the second question, assumption (5) of Section 5.8.2, relative to the incremental cost curves, guaranteed convexity of the objective function (see Section 4.4.2). And this, in turn, guaranteed the minimization of each strategy subject only to a posterior feasibility check.

Barring decreasing incremental cost curves, globality thus depends solely on providing a suitable RAMM.

5.8.5 How Accurate are the Results?

The forecast error analysis of Sections 5.7.3 to 5.7.5, combined with the work of Watt (55), indicate that strategy versus strategy total cost differences are probably accurate only to within a minimum of \$500,000 when compared with the actual (versus calculated) total costs realized over five or six years (on the order of \$500,000,000). The major contributions to this error are CORSOM inaccuracies (>\$100,000 per reactor) and poor forecasts regarding fossil fuel costs, present value rate, customer load demands and unit availabilities. The latter two forecasting errors have been totally ignored in this initial modeling work and should, therefore, be high on the list for future development effort.

5.8.6 What are the Computational Requirements?

Computational requirements have been previously discussed in Sections 3.6, 4.6 and 5.6.4.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

6.1 Summary

This work has presented a nuclear power management multi-year model suitable for 5 to 10 year multi-reactor fuel management studies. The overall model consists of four sub-models:

(1) Refueling and Maintenance Model (RAMM),

(2) System Integration Model (SIM),

(3) System Optimization Model (SOM), and

(4) CORe, Simulation and Optimization Model (CORSOM) for each reactor type.

The SIM and SOM sub-models have been developed in this study and are discussed in detail. Computerized versions of these (SYSINT and SYSOPT, respectively), were programmed and tested. Numerical results were presented not only to evaluate the models, but also as examples of the overall model's versatility. As an aid in further model development, the following sections summarize the main conclusions and recommendations. (All computation times given below are in terms of an IBM 370 model 155 computer.)

6.2 Conclusions

(1) While fossil unit instantaneous power levels are chosen so as to maintain equal fossil incremental costs, the nuclear unit period energy production schedules should be

chosen so that all reactors are operating at the same nuclear incremental cost.

(2) The overlapping of irradiation cycles for the various reactors plus Conclusion (1) above leads to idealized production schedules yielding a constant nuclear incremental cost regardless of time. However, such production schedules may not be feasible. The computer code SYSOPT determines the optimum feasible production schedule that approaches this ideal as closely as possible (i.e., with minimum total system revenue requirement).

(3) While nuclear average fuel costs are on the order of 1.8 to 2.2 $/MWH, the incremental system cost of designing more nuclear energy into a given cycle is on the order of 0.8 to 1.6 $/MWH. During nightly low load periods, it would be economical to sell power to neighboring utilities in this lower price range. In fact, it is even more advantageous to use excess nuclear capacity for pumping at a stored-hydro station.

(4) Even with fossil fuel costing as little as 25¢/Mega-BTU (and rising), the best-plant fossil incremental cost is at least 2.0 $/MWH. Considering that even the highest nuclear incremental fuel costs today are less than 1.6 $/MWH, the conclusion is that nuclear incremental costs will be less than fossil incremental costs for the foreseeable future.

(5) As a result of Conclusion (4) above, nuclear power should always be operated so as to displace maximum fossil energy.

(6) Another conclusion based on Conclusion (4) above is that an economic incentive exists for lengthening nuclear irradiation cycles in terms of both energy and time. Increasing nuclear incremental costs are more than justified by the reduction in average annual fossil replacement energy required during refueling downtime. In addition, minimum total system nuclear downtime (subject to burnup constraints) appears to be a good a priori measure of the ranking of various refueling and maintenance strategies.

(7) One of the key input parameters was shown to be the fossil thermal energy cost. A small forecasting error in this number alone (roughly 6 out of 40 ¢/MegaBTU) altered example four year strategy cost differences by $500,000 (out of a total difference of $1,500,000).

(8) Using the latest in a PWR in-core model (41) and assuming convergence in five iterations, computation costs are on the order of 300 to 500 $ per strategy for a utility system possessing five nuclear reactors. Assuming a 1% annual savings in nuclear fuel revenue requirements alone, roughly $500,000 per year would be saved. Thus, scores of strategies could be run each year in order to up-date the current operating strategy, specify the next set of reload enrichments or, more importantly, re-optimize the strategy to account for large perturbations from the intended production or refueling and maintenance schedule. For example, how does the AEC's 1973 step price increase in enrichment charges from $32 to $38.50 per kg SWU (1) affect the

current operating strategy. The nuclear power management model's ability to quantify the complex utility system trade-offs (not only nuclear-vs-nuclear, but also, nuclear-vs-fossil) make it an indispensable planning tool for nuclear utility decision-makers.

(9) The reactor-by-reactor nuclear energy allocation problem may be cast as a network supply problem, permitting the use of network programming rather than the more general (and computationally difficult) linear programming.

(10) In addition, the Out of Kilter Network Program (45) was demonstrated to be sufficiently flexible to permit piecewise-linearization of the nuclear system optimization to an extent approaching quadratic programming in accuracy and exceeding it in the size of the problem solved.

(11) Several instances were encountered where strategy reoptimization was not necessary in order to evaluate the effect of various input data changes on previously optimized solutions. The capability to merely re-evaluate several previously optimized solutions eliminates the need for more than a single iteration per strategy and thus, reduces computational costs further.

(12) On a multi-year basis (~5 to 7 years), strategy-vs-strategy cost differences are estimated to be accurate only to within $200,000 per 1000 MW reactor (out of roughly $50,000,000) given perfect (i.e., deterministic) load and unit reliability forecasts. Estimates of the additional

cost inaccuracies incurred due to errors in these forecasts form part of the Recommendations.

(13) The multi-year planning horizon ought to include a full core of freely specified enrichments for each reactor. In other words, the horizon should be far enough into the future to completely predict the discharge characteristics of the next reload enrichment to be finalized (i.e., actually ordered from a vendor) for each reactor. In addition, it is convenient to place the planning horizon in a forbidden maintenance period in order to minimize distortion of strategy-vs-strategy cost differences due to horizon end effects. Beyond the planning horizon, cycle energies should be postulated so as to maintain the individual operating philosophy ("character") of each strategy, not return to an arbitrary final state.

6.3 Recommendations

(1) The Booth-Baleriaux probabilistic utility model within SYSINT represents the latest in utility system simulation. The current model is capable of simulating a 100 unit utility system (with up to 5 valve points per unit) for up to 100 time periods. Since nuclear, fossil and peaking equipment are currently included, the addition of hydro and pumped-hydro equipment (i.e., types involving scarce resource utilization) is highly recommended in order to complete the range of possibilities.

(2) The Booth-Baleriaux model's accuracy has been established by others (19, 36, 49) based on the reproduction of historical data. However, little if any testing has been done of the model's ability to project future production given forecasted loads and unit reliability data. Research into this area is needed to establish the sensitivity of the various results to unavoidable forecasting errors. Ultimately, the nuclear power management model should yield not only a numerical answer, but also a confidence interval around it.

(3) As a further refinement of the Booth-Baleriaux model, the two-state forced-outage model ought to be replaced with a more general model permitting unit derating (See Appendix A).

(4) The principal recommendation for SYSOPT model improvement is expansion of the network structure to permit decreased cycle energy step size (i.e., increased total cost linearization) and, hence, provide a closer approximation to quadratic programming (QP). (Due to problem size, the direct inclusion of a general QP model is out of the question.) Each iteration of SYSOPT (itself using less than 10 seconds for a six reactor utility system) requires another 20 minutes of computer time within even advanced in-core models (41). The reduction in step size is aimed at decreasing the number of iterations required to reach an acceptable optimum nuclear production schedule (hopefully, to as few as three of four).

(5) Other suggested improvements to SYSOPT include the capability to optimize nuclear units with varying incremental heat rates and to handle core reactivity stretch-out (i.e., allowance for reduced plant capacity). The inclusion of capital and other nonoperating revenue requirements in the total cost would complete a useful tool for multi-year (or longer) planning horizons.

(6) Relative to completion of the overall nuclear power management model put forth in this work, acceptable RAMM's already exist. The most severe deficiency is not due to either the SIM (SYSINT) or SOM (SYSOPT), but to a lack of computationally efficient CORSOM's for each reactor type. These in-core models represent the critical sub-models requiring the greatest development effort. The PWR in-core model recently developed by Kearney (41), though a great leap forward in nuclear in-core simulation and optimization, still requires over 3 minutes per reactor per SYSOPT iteration. CORSOM's an order of magnitude faster are desired so that computation costs can be rendered truly insignificant compared with system savings.

APPENDIX A

BOOTH-BALERIAUX EQUATIONS FOR GENERAL FORCED-OUTAGE MODELS

A.1 Forced-Outage Models

Presented in this Appendix are derivations of the most general forms of the Booth-Baleriaux deconvolve-load-convolve Equations (3.55), (3.56) and (3.57) of the multiple increment algorithm of Section 3.3.2.2. Whereas, Chapter 3 dealt exclusively with the two-state forced-outage model, this Appendix extends the model to premit derating of a unit. That is, a unit may be unable to produce at full capacity, yet be capable of operating at 90% of capacity--a 10% derating.

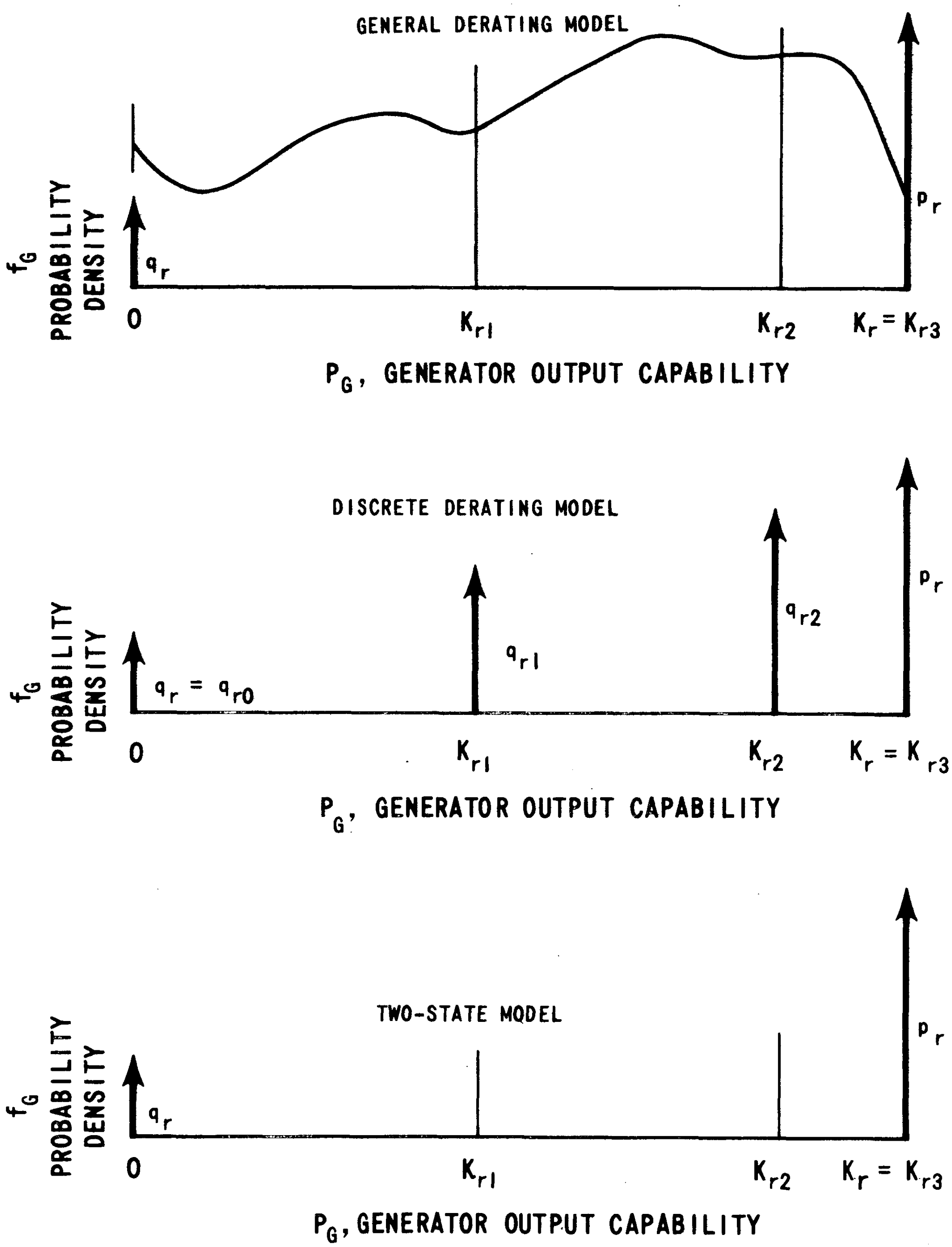
To distinguish the more general unit performance models from the simpler two-state model requires introducing their probability density functions (26) f_G as a function of P_G, the generating unit output power capability. Thus, $f_G(P_G)dP_G$ represents the probability that, at a random instant of time, the unit's capability is limited to a range of dP_G about P_G. For the two-state model (See Figure A.1), f_G is one impulse (q_r) at $P_G=0$ and another (p_r) at $P_G=K_r$ since the unit is assumed not operable at all $(P_G=0)$ or operable over the entire range to rated power $(P_G=K_r)$.

The probability density functions f_G for the general unit performance models are also shown in Figure A.1. With probability p_r, unit r is capable of full power operation at

6253-84

Figure A.1

Probability Density Functions of Unit Capability



$P_G = K_r (\equiv K_{rI})$MW. Conversely, with probability q_r the unit is not capable of producing any power at all ($P_G=0$). For the "general derating" model, any fraction of capacity may be derated and, hence, f_G may have any shape between $0 < P_G < K_r$ so long as the standard probability density function requirement is met,

$$\int_{-\infty}^{+\infty} f_G(P_G)\,dP_G = 1 \qquad \text{(A.1)}$$

More specifically,

$$p_r + q_r + \int_{0^+}^{K_r^-} f_G(P_G)\,dP_G = 1 \qquad \text{(A.2)}$$

In the second "discrete derating" model, only whole increments of capacity may be derated and f_G is restricted to a probability mass function with each q_{ri} coinciding with the K_{ri} capacity increments. For $i = 0$, $q_{ri} = q_{r0} = q_r$ and

$$p_r + \sum_{i=0}^{I-1} q_{ri} = 1 \qquad \text{(A.3)}$$

Finally, for the special case $q_{ri} = 0$ for all $i > 0$, the discrete derating model reduces to the original (all-or-nothing) "two-state" model of Chapter 3,

$$p_r + q_r = 1 \qquad \text{(A.4)}$$

The symbol $\wp$ is used to denote the complementary cumulative distribution function for f_G,

$$\wp(P_G) = 1 - \int_{-\infty}^{P_G} f_G(P)\,dP \tag{A.5}$$

Thus, $\wp(P_G)$ represents the probability that the unit is capable of generating P_GMW or more at any random instant of time.[1] Figure A.2 presents typical $\wp$ for the three models.

When performing each convolution or deconvolution, the pertinent portion of the K_{rI}MW unit may be temporarily treated as a smaller "sub-unit" of $K_{r\ell}$MW. Derived in this manner, the following equations are the most general.

For this smaller unit, $\wp(P_G)$, by definition, falls to zero just beyond $K_{r\ell}$MW. In addition, f_G for the sub-unit is most easily viewed as the probability masses and derivative of this truncated $\wp(P_G)$,

$$f_G(P_G) = -\frac{d\,\wp(P_G)}{dP_G} \tag{A.7}$$

---

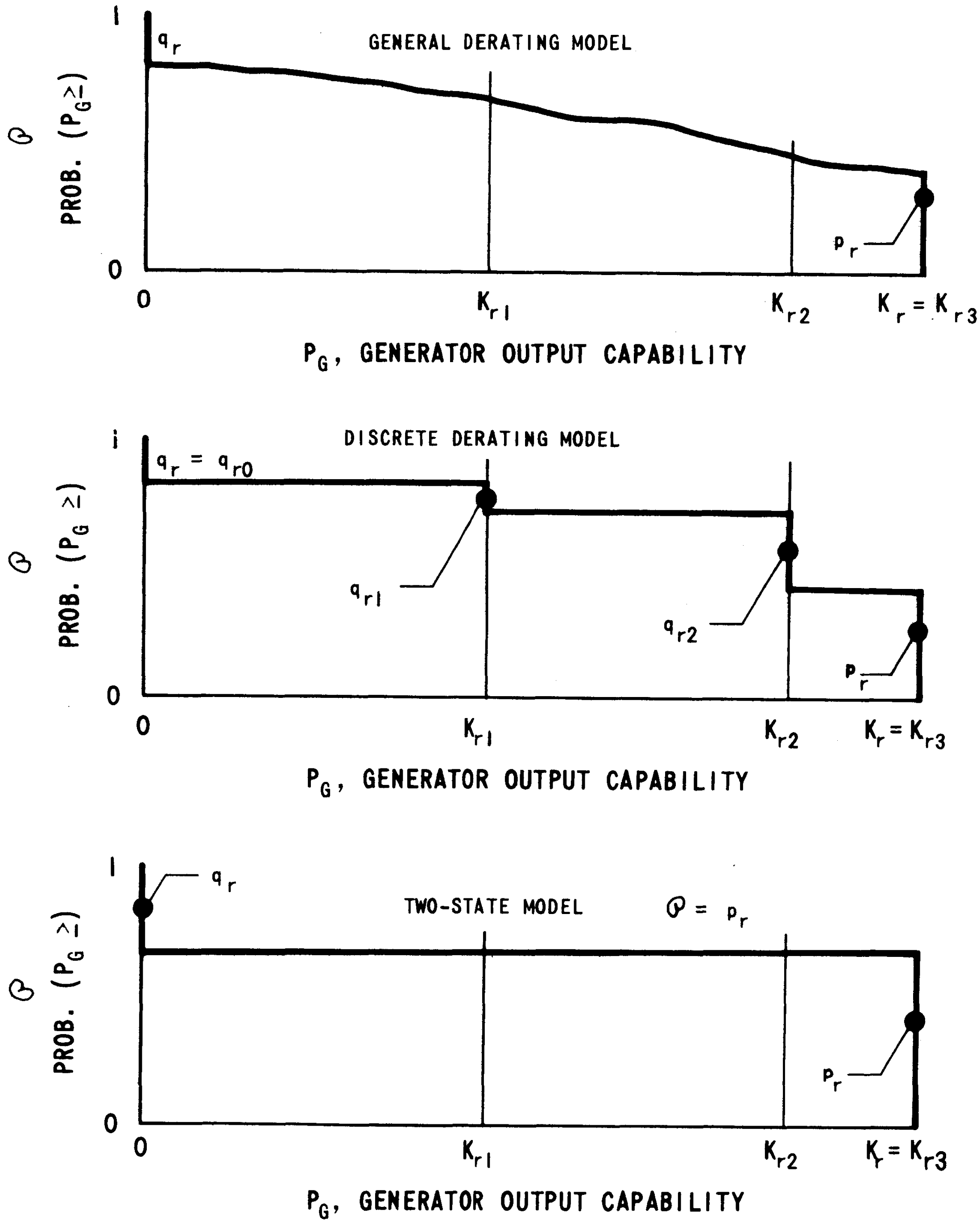
[1]Note that in this work, the complementary cumulative distribution function is defined to include the equality at the upper limit of the integral, in contrast to the usual (26) placement of the equality with the cumulative distribution function itself,

$$\underbrace{\text{Prob.}(P<P_G) + \overbrace{\text{Prob.}(P=P_G)}_{}}_{\text{usual C.D.F.}}\overbrace{{} + \underbrace{\text{Prob.}(P>P_G)}_{\text{usual C.C.D.F.}}}^{\wp(P_G)} = 1 \tag{A.6}$$

The distinction is purely academic as applied in this work.

Figure A.2

Performance Probability Functions of Unit Capability



Of more immediate use than f_G in determining equivalent load distribution is the forced-outage distribution $f_O(P_O)$ since only the unit's forced-outages contribute to the equivalent load [See Equation (3.5)]. To derive $f_O(P_O)$, use is made of the fundamental applied probability equation for changing random variables in a density function,

$$f_O(P_O)dP_O = f_G(P_G)dP_G \tag{A.8}$$

or

$$f_O(P_O) = f_G(P_G)\left|\frac{dP_G}{dP_O}\right| \tag{A.9}$$

Since,

$$P_G + P_O = K_{r\ell} (= K_{ri} + P_{O_i} \text{ for the discrete case}) \tag{A.10}$$

$$\left|\frac{dP_G}{dP_O}\right| = \left| -1 \right| = 1 \tag{A.11}$$

Hence,

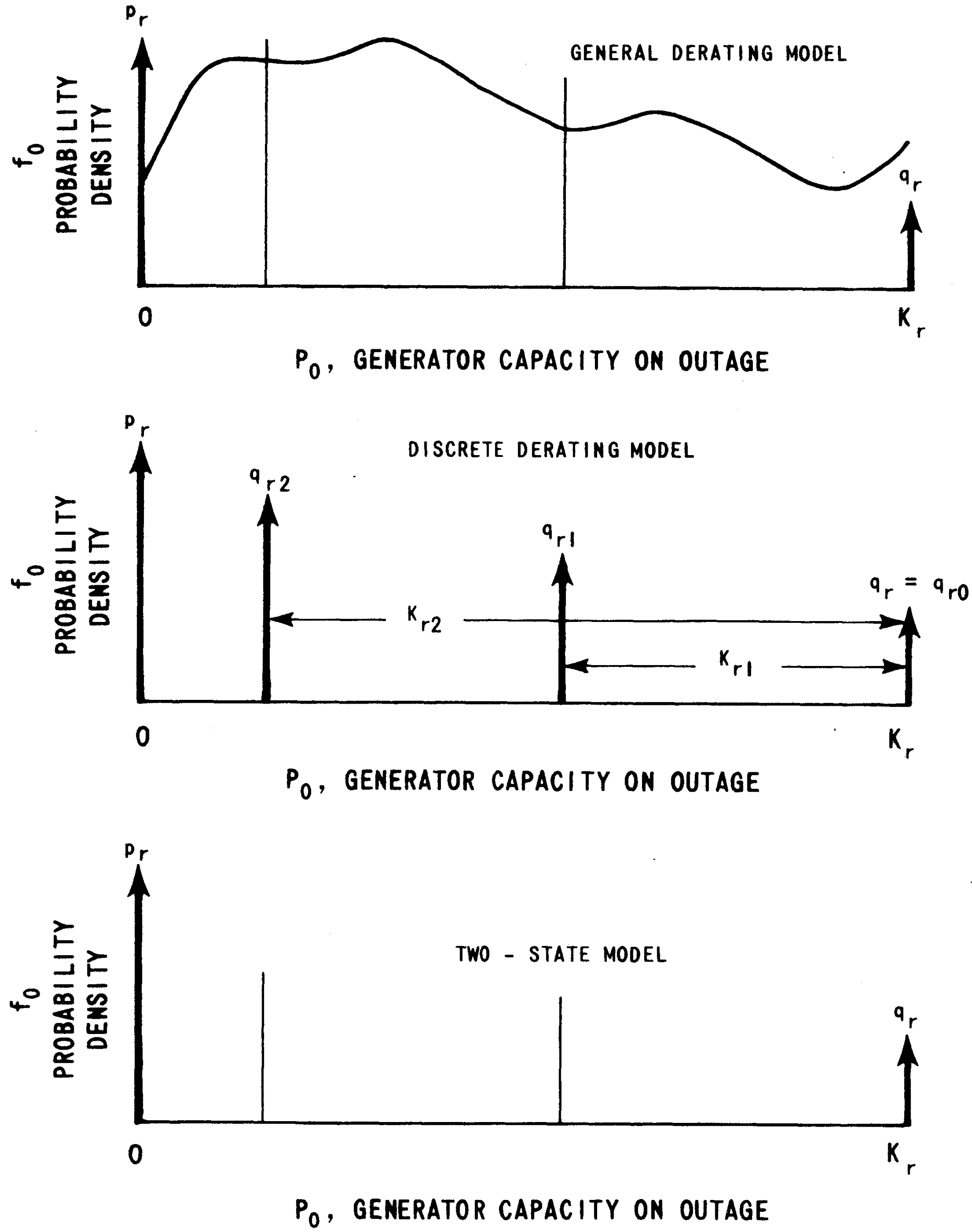
$$f_O(P_O) = f_G(K_{r\ell}-P_O) \tag{A.12}$$

and f_G is merely reversed (i.e., rotated about 0.5* $K_{r\ell}$). Figure A.3 presents typical $f_O(P_O)$ for the three models.

6253-86

Figure A.3

Probability Density Functions of Unit Capacity on Outage



A.2 Convolution

As in Chapter 3, convolution is presented first since deconvolution is most easily expressed as the reverse of convolution. The aim of the convolution is to calculate an $F^{w}_{r\ell}$ which includes (i.e., superscript w = with) unit r's forced-outages (up to $K_{r\ell}$MW). The starting point is (1) the current equivalent load curve $F^{wo}_{r\ell}$ that does not include any allowance for the outages of unit r (i.e., wo = without) and (2) the sub-unit's own forced-outage distribution $f_O(P_O)$. (Since all references to F are for the same unit increment $r\ell$, the notation is shortened to F^{w} and F^{wo}).

From the equivalent load definition Equation (3.5), the notation becomes

$$P_e = P_D + (P_O)_{\substack{\text{Other}\\ \text{Units}}} + (P_O)_{\text{Unit } r} \tag{A.13}$$

$$\downarrow \qquad \underbrace{\qquad\qquad\qquad}_{} \qquad \downarrow$$

$$P^{w}_{e} = P^{wo}_{e} + P_O \tag{A.14}$$

The equivalent load curve F^{wo} is the complementary cumulative density function of f^{wo} or the probability that $P_e \geq P^{wo}_e$,

$$F^{wo}(P^{wo}_e) = 1 - \int_{-\infty}^{P^{wo}_e} f^{wo}(P)\,dP \tag{A.15}$$

Convolution is performed in the manner of Drake (26) using Figure A.4. Thus, $F^W(P_e^W)$ represents the complementary cumulative distribution function of $f_{e,O}^{WO}$ (i.e., below and to the left of the P_e^W = constant line),

$$F^W(P_e^W) = 1 - \int_{-\infty}^{+\infty} \int_{-\infty}^{(P_e^W - P_O)} f_{e,O}^{WO}(P_e^{WO}, P_O)\, dP_e^{WO} dP_O \tag{A.16}$$

Assuming the usual statistical independence between equivalent load (f^{WO}) and un-included unit forced-outages (f_O),

$$f_{e,O}^{WO}(P_e^{WO}, P_O) = f^{WO}(P_e^{WO}) \cdot f_O(P_O) \tag{A.17}$$

Hence,

$$F^W(P_e^W) = 1 - \int_{-\infty}^{+\infty} \int_{-\infty}^{(P_e^W - P_O)} f^{WO}(P_e^{WO}) \cdot f_O(P_O)\, dP_e^{WO} dP_O \tag{A.18}$$

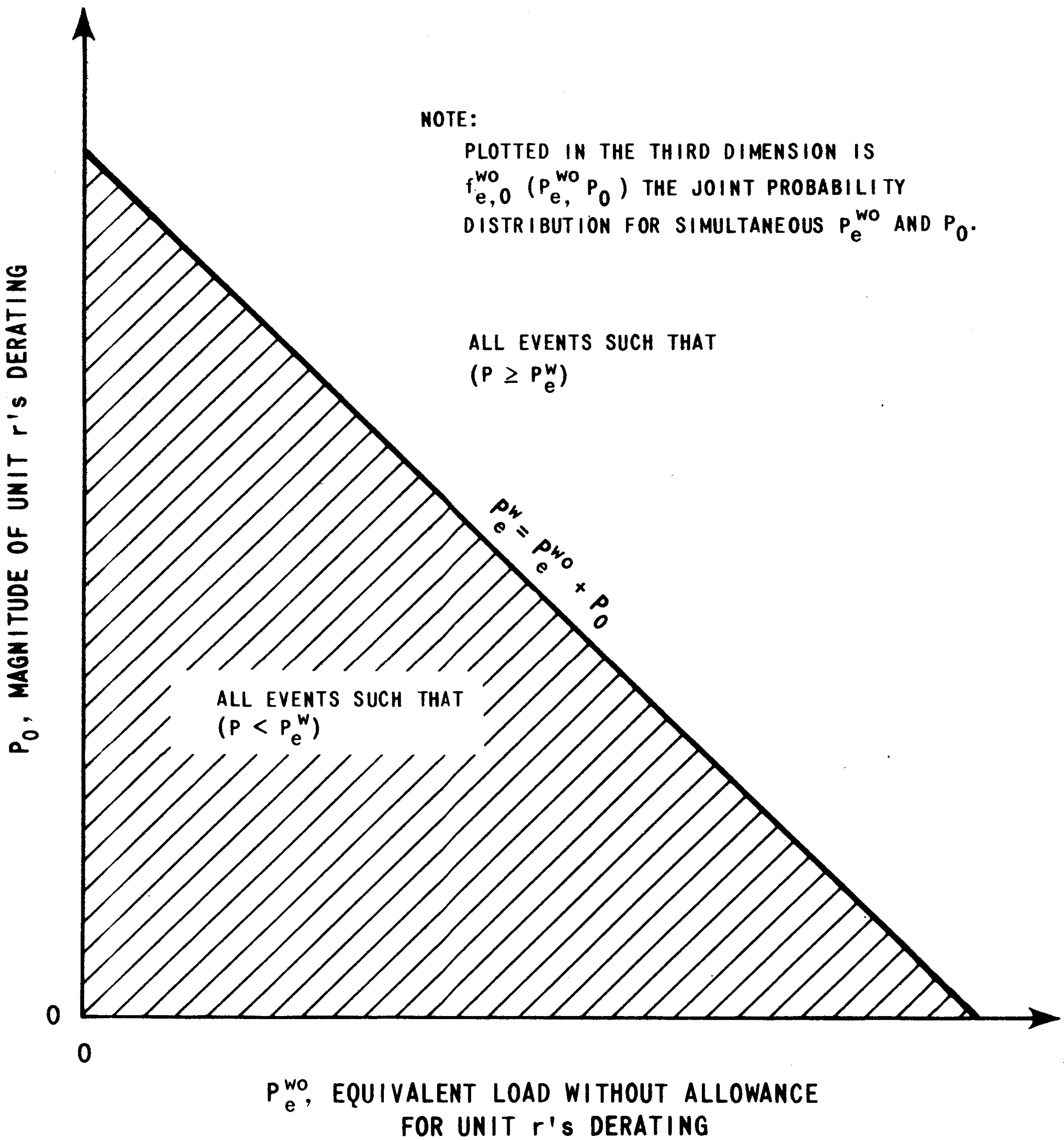
Since

$$1 = \int_{-\infty}^{+\infty} f_O(P_O)\, dP_O \tag{A.19}$$

Figure A.4

6253-87

Event Space Interpretation of Convolution



Equation (A.18) can be factored into

$$F^{W}(P_e^{W}) = \int_{-\infty}^{+\infty} f_O(P_O)\left[1 - \int_{-\infty}^{(P_e^{W}-P_O)} f^{WO}(P_e^{WO})\,dP_e^{WO}\right] dP_O \tag{A.20}$$

Since the bracketed term is, by definition Equation (A.15), the complementary cumulative distribution function of f^{WO}, i.e., $F^{WO}(P_e^{W}-P_O)$, then

$$F^{W}(P_e^{W}) = \int_{-\infty}^{+\infty} f_O(P_O)F^{WO}(P_e^{W}-P_O)\,dP_O \tag{A.21}$$

Reducing the P_e^{W} notation to merely P_e, the result is the convolution of the general derating model,

$$F^{W}(P_e) = \int_{-\infty}^{+\infty} f_O(P_O)\cdot F^{WO}(P_e-P_O)\,dP_O \tag{A.22}$$

For the discrete derating model of Equation (A.3), this reduces to

$$F^{W}(P_e) = p_r F^{WO}(P_e) + \sum_{i=o}^{\ell-1} q_{ri}\cdot F^{WO}(P_e-K_{r\ell}-K_{ri}) \tag{A.23}$$

Finally, the two-state model of Equation (A.4) yields the original Equation(3.57),

$$F^{W}(P_e) = p_r F^{WO}(P_e) + q_r F^{WO}(P_e-K_{r\ell}) \tag{A.24}$$

A.3 Deconvolution

Deconvolution seeks to regain F^{wo} given F^{w}. That is, it strips out the forced-outages of the $K_{r\ell}$MW unit.

Performing the integration of Equation (A.22) from $-\infty$ to 0^{+} (See Figure A.1),

$$F^{w}(P_e) = p_r F^{wo}(P_e) + \int_{0^+}^{\infty} f_0(P_0) F^{wo}(P_e - P_0)\, dP_0 \tag{A.25}$$

Solving for $F^{wo}(P_e)$, deconvolution for the general derating model becomes

$$F^{wo}(P_e) = \frac{1}{p_r}\left[F^{w}(P_e) - \int_{0^+}^{\infty} f_0(P_0) F^{wo}(P_e - P_0)\, dP_0\right] \tag{A.26}$$

For the discrete derating model, Equation (A.23) rearranges into

$$F^{wo}(P_e) = \frac{1}{p_r}\left[F^{w}(P_e) - \sum_{i=o}^{\ell-1} q_{ri}\, F^{wo}(P_e - K_{r\ell} + K_{ri})\right] \tag{A.27}$$

Likewise, the two-state model of Equation (3.55) may be obtained from Equation (A.24),

$$F^{wo}(P_e) = \frac{1}{p_r}\left[F^{w}(P_e) - q_r F^{wo}(P_e - K_{r\ell})\right] \tag{A.28}$$

A.4 Loading

In performing the expected loading calculation of Figure A.5, the statistical independence is again invoked,

$$\text{Prob.}\begin{bmatrix}\text{Energy increment}\\ \text{at } P_e = P^o_{ri} + \delta K\\ \text{is generated}\end{bmatrix} = \text{Prob.}\begin{bmatrix}\text{Energy}\\ \text{demanded}\\ \text{at } P_e\end{bmatrix} \times \text{Prob.}\begin{bmatrix}\delta K \text{ increment of}\\ \text{capacity is}\\ \text{operable, i.e.,}\\ P_G \geq K_{r,i-1} + \delta K\end{bmatrix} \tag{A.29}$$

$$= F^{wo}(P^o_{ri} + \delta K) \cdot \mathcal{P}(K_{r,i-1} + \delta K) \tag{A.30}$$

Integrating from $\delta K = 0$ to $\delta K = \Delta K_{ri}$ and multiplying by T', the length of the time period, the general derating model is loaded according to

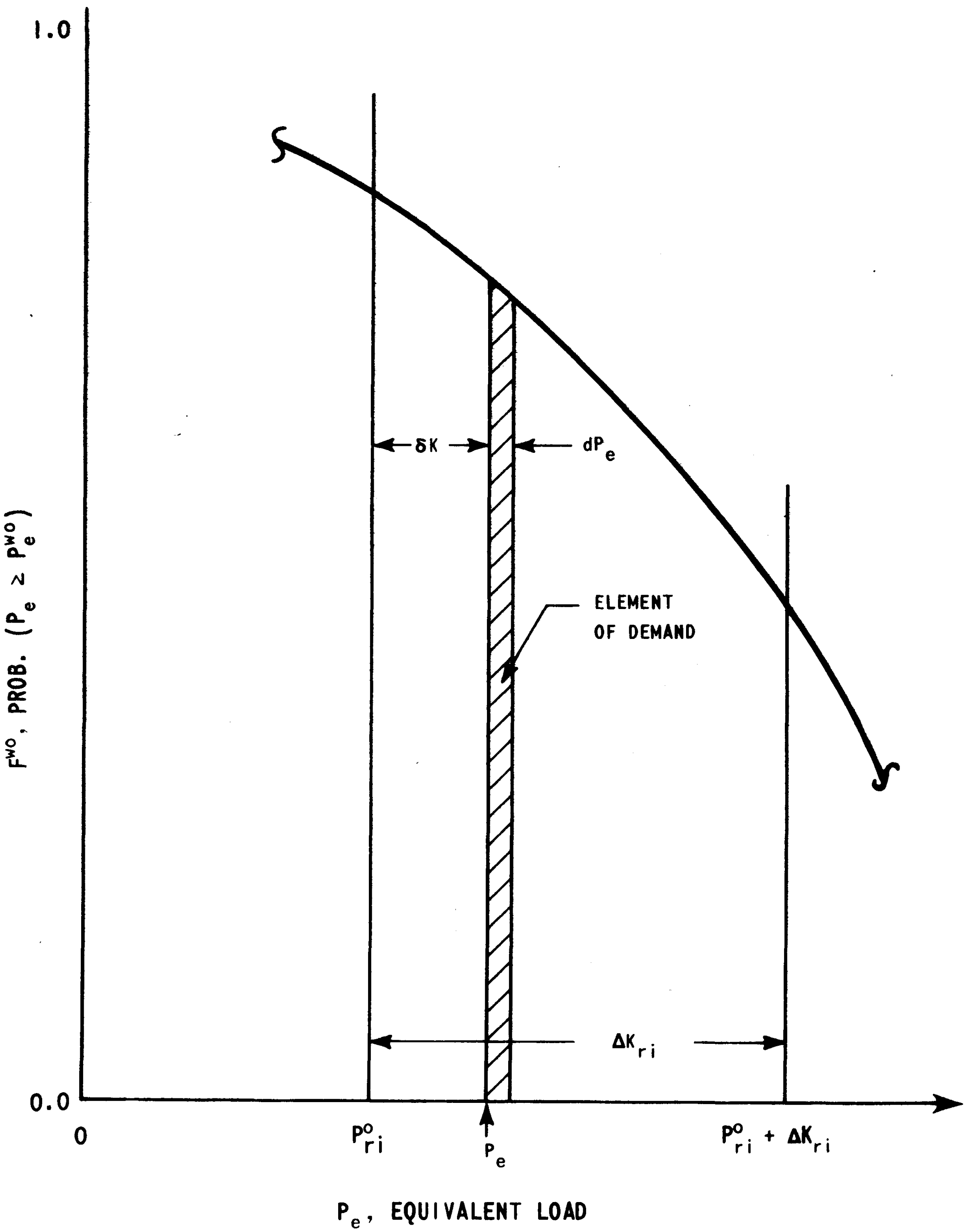
$$E_{ri} = T' \int_{P^o_{ri}}^{P^o_{ri} + \Delta K_{ri}} F^{wo}(P_e)\, \mathcal{P}(K_{r,i-1} + P_e - P^o_{ri})\, dP_e \tag{A.31}$$

For the discrete derating model (See Figure A.2),

$$\mathcal{P}(K_{r,i-1} + \delta K) \equiv \mathcal{P}_{ri} = \text{constant for } 0 \leq \delta K \leq \Delta K_{ri} \tag{A.32}$$

Figure A.5

Load Demanded of ΔK_{ri} Unit Increment



and, hence,

$$E_{ri} = T'\mathcal{P}_{ri}\int_{P_{ri}^{o}}^{P_{ri}^{o}+\Delta K_{ri}} F^{wo}(P_e)\,dP_e \qquad (A.33)$$

The two-state model ($\mathcal{P}_{ri} = p_r$) reduces to Equation (3.56),

$$E_{ri} = T'p_r\int_{P_{ri}^{o}}^{P_{ri}^{o}+\Delta K_{ri}} F^{wo}(P_e)\,dP_e \qquad (A.34)$$

A.5 Summary

Table A.1 presents a summary of the deconvolve-load-convolve sequence of calculations for each forced-outage models.

Table A.1

Summary of Booth-Baleriaux Equations for Various Forced-Outage Models

Model	Step	Equation
GENERAL DERATING MODEL	D	$F^{w\bar{o}}(P_e) = \frac{1}{p_r}\left[F^{w}(P_e) - \int_{0^+}^{\infty} f_0(P_0)\,F^{w\bar{o}}(P_e-P_0)\,dP_0\right]$
	L	$E_{ri} = T'\int_{P_{ri}^\circ}^{P_{ri}^\circ+\Delta K_{ri}} F^{w\bar{o}}(P_e)\,\mathcal{P}(P_e-P_{ri}^\circ+K_{r,i-1})\,dP_e$
	C	$F^{w}(P_e) = \int_{-\infty}^{+\infty} f_0(P_0)\,F^{w\bar{o}}(P_e-P_0)\,dP_0$
DISCRETE DERATING MODEL	D	$F^{w\bar{o}}(P_e) = \frac{1}{p_r}\left[F^{w}(P_e) - \sum_{i=0}^{\ell-1} q_{ri}\,F^{w\bar{o}}(P_e-K_{r\ell}+K_{ri})\right]$
	L	$E_{ri} = T'\mathcal{P}_{ri}\int_{P_{ri}^\circ}^{P_{ri}^\circ+\Delta K_{ri}} F^{w\bar{o}}(P_e)\,dP_e$
	C	$F^{w}(P_e) = p_r F^{w\bar{o}}(P_e) + \sum_{i=0}^{\ell-1} q_{ri}\,F^{w\bar{o}}(P_e-K_{r\ell}+K_{ri})$
TWO-STATE MODEL	D	$F^{w\bar{o}}(P_e) = \frac{1}{p_r}\left[F^{w}(P_e) - q_r\,F^{w\bar{o}}(P_e-K_{r\ell})\right]$
	L	$E_{ri} = T' p_r \int_{P_{ri}^\circ}^{P_{ri}^\circ+\Delta K_{ri}} F^{w\bar{o}}(P_e)\,dP_e$
	C	$F^{w}(P_e) = p_r F^{w\bar{o}}(P_e) + q_r F^{w\bar{o}}(P_e-K_{r\ell})$

NOTES:

(1) D = DECONVOLVE, L = LOAD, C = CONVOLVE

(2) IDENTITY OF SUB-UNIT $K_{r\ell}$ CHANGES BETWEEN DECONVOLUTION AND CONVOLUTION STEPS SINCE $\ell_{CONVOLVE} \neq \ell_{DECONVOLVE}$ TO ACCOUNT FOR $\Delta K_{r\ell}$ MW JUST LOADED.

(3) IN ACCORDANCE WITH EQUATION (A.7) AND NOTE (2), p_r FOR SUB-UNIT $K_{r\ell}$ IS ACTUALLY ORIGINAL $\mathcal{P}(P_G)$ (FOR ENTIRE UNIT K_r) EVALUATED AT $P_G = K_{r\ell}$.

APPENDIX B

AREA METHOD OF FORMULATING SHAPE CONSTRAINT

Section 4.2.4 explained the need for a shape constraint in the SOM and derived an approximate variance method for establishing the feasibility of postulated $\bar{F}_r$ shapes. This Appendix presents the rigorous (i.e., necessary and sufficient) but cumbersome, area method. Recall that given an F_e system shape (cf. Figure 4.9 and Figure B.1) over the system nuclear upper increment capacity from 0 to k_T', the problem is to determine if a set of postulated period energies E_r (that resulted in the $\bar{F}_r$ postulated average reactor shape) could be satisfied by a feasible detailed loading order.

The area method is based on an observation relative to the mapping process of Figure 4.7. That is, over the range from 0 to any equivalent load P, it is impossible to reorder $F_e(P_e)$ into a detailed $F_r(P_r)$ such that the resulting $\bar{F}_r(P_r)$ contains more energy than the original $F_e(P_e)$. In other words, there can be no pre-production of equivalent load energy. Thus,

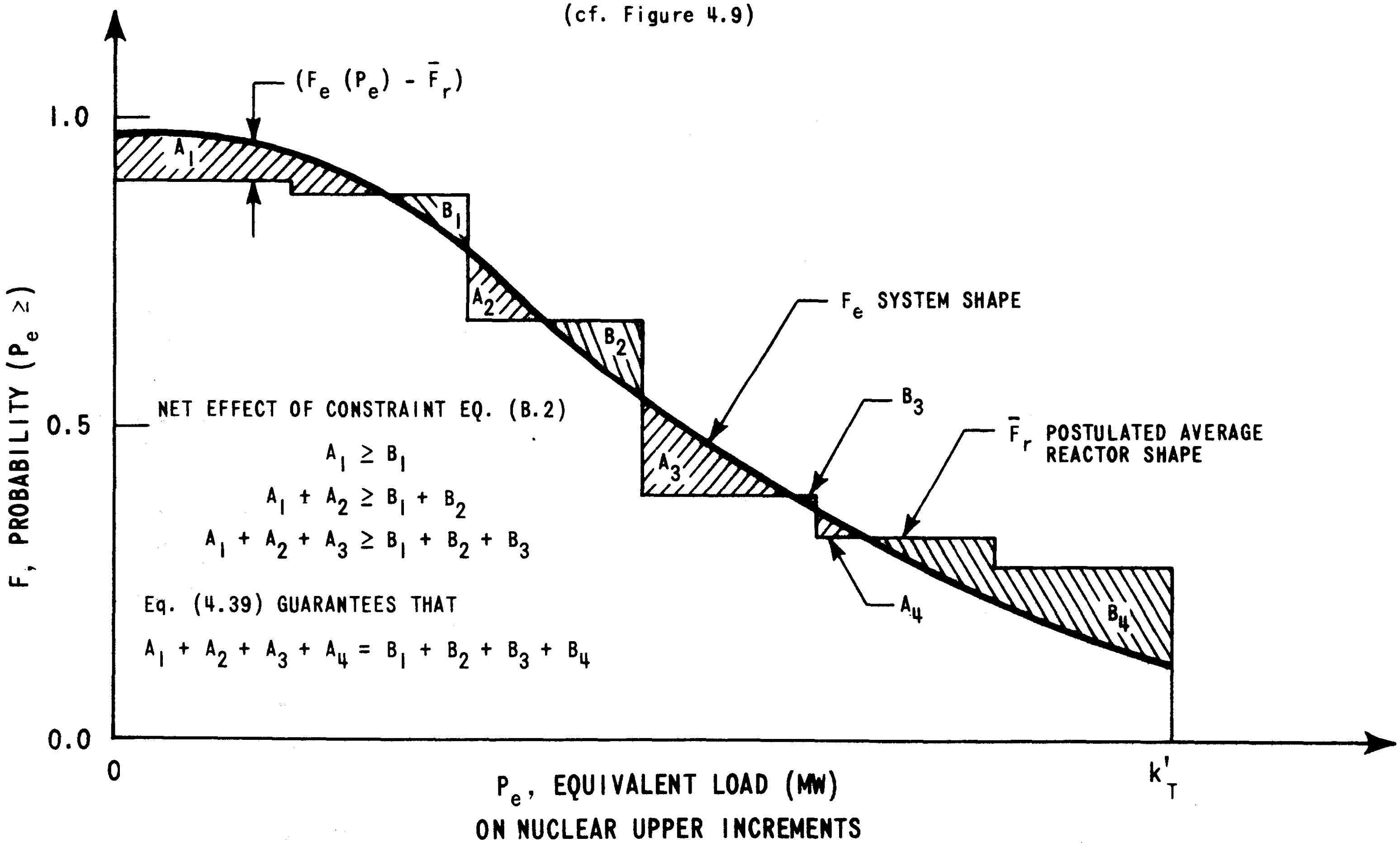
$$\int_0^P \bar{F}_r dP_r \leq \int_0^P F_e(P_e) dP_e \qquad \text{(B.1)}$$

or

Figure B.1

Area Method for Determining Feasibility

(cf. Figure 4.9)



6253-38

$$0 \leq \int_{0}^{P} (F_e(P_e) - \overline{F}_r) dP_e \qquad (B.2)$$

Hence, the net area between $F_e(P_e)$ and $\overline{F}_r(P_r)$ from 0 to any P must be positive (See Figure B.1).

If the inequality of Equation (B.1) or (B.2) does not hold at any single P, the required detailed loading order does not exist (e.g., see Figure 4.6). Herein, lies the difficulty with the area method: it must be checked at every P or at least at several well-chosen ones. Though the method is rigorous, the amount of computer data handling and storage are unwieldy even using a linear approximation to F_e.

APPENDIX C

REFERENCE UTILITY SYSTEM EXAMPLES

Section 2.1.2.3 presented the Five-Unit Reference Utility System. Unit characteristics were detailed in Figure 2.2. Table C.1 summarizes the data for each valve point. Figure C.1 repeats the F_D customer load-duration curve of Figure 2.9 for the 730 hour month.

Table C.2 presents a SYSINT Fortran-to-text symbol cross-reference table. The following Tables C.3 to C.20 present the numerical data of SYSINT's Booth-Baleriaux model for each of the six Examples, in turn. (Section E.3 presents the computer input decks actually used in executing the Examples.)

Table C.1

<u>Unit Characteristics for Reference Utility System</u>

Total Capacity - 2000 MW

Unit Name r	Type	Rated Cap. K_r MW	Perf. Prob. p_r %	SUSD. Cost $	Valve Point Data			
					K_{r1} MW	$\bar{e}_{r1}$ $/MWH	K_{r2} MW	λ_{r2} $/MWH
I	Peaking	100	95	45	100(95)*	16.20	-----	------
II	Fossil	200	95	400	100(95)	5.50	200(190)	4.25
III	Nuclear	300	90	228	100(90)	2.28	300(270)	1.90
IV	Fossil	600	90	1440	200(180)	3.92	600(540)	3.32
V	Nuclear	800	85	432	300(255)	2.25	800(680)	1.71

*(95MW) = 0.95 x 100 MW = K_{ri} for Example 2 only.

$$\underset{p_r}{0.95} \times \underset{K_{ri}}{100\ \text{MW}}$$

6253-17

Figure C.1

Normalized Customer Load-Duration Curve for 730 Hour Month on Reference Utility System

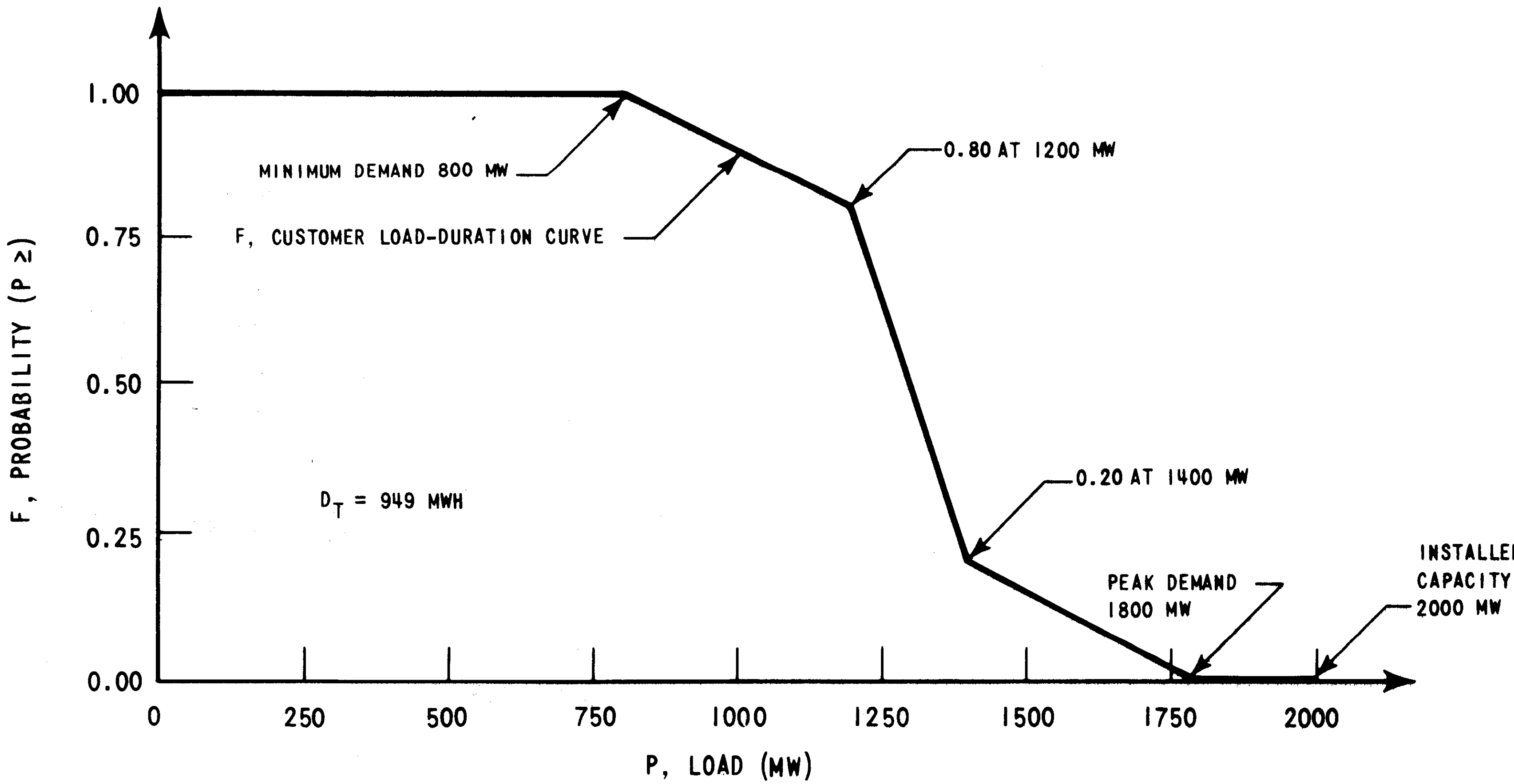


Table C.2

SYSINT Fortran-to-Text Symbol Cross-Reference Table

SYSINT Fortran Symbol	Text Symbol	Description
AVPROB	$\frac{1}{\Delta K_{ri}} \int_{P^{o}_{ri}}^{P^{o}_{ri}+\Delta K_{ri}} F^{wo}_{ri}(P_e)dP_e = \frac{E_{ri}}{K_{ri}\ T' \rho_r} \equiv L'_{ri}$	Average availability-based capacity factor for the capacity increment
DELGWH	E_{ri}	Increment energy production, GWH
DM		Spacing of F array stored in PROB.
EXPGWH	$\sum^{I} E_{ri}$	Cumulative increment production, GWH
IDNO		Unit identification number
IEMAX		PROB storage location of peak equivalent load, PROB(IEMAX) $\equiv$ 0.0
L	r	Unit index = order unit data read in = order final unit results presented
MWADD	ΔK_{ri}	Increment of capacity being loaded for unit-of-interest r

Table C.2--Continued

SYSINT Fortran Symbol	Text Symbol	Description
MWIN	$K_{r,i-1}$	Unit r capacity previously loaded.
MWTOT	K_{ri}	Unit r capacity now loaded
PE	$P^{\circ}_{ri} + \Delta K_{ri}$	Equivalent load after loading increment
PROB(K)	$F_{ri}(P_e = K*DM)$	Current F equivalent load-duration curve
	ℓ	Position in loading order

Table C.3

Example 1 on Reference Utility System:
"Deterministic Model (No Forced-Outages)"

(See Sect. 2.2.1 for further details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 $)
I	1	9 (last)	- 0 -	- 0 -
II	1	4	73.00	401.5
	2	8	- 0 -	- 0 -
III	1	2	73.00	166.4
	2	6	73.00	138.7
IV	1	3	146.00	572.3
	2	7	29.20	97.0
V	1	1 (first)	219.00	492.8
	2	5	335.80	574.2
Utility Production			949.00	2442.9
Emergency Purchases (at 10$/MWH)			- 0 -	- 0 -
Total			949.00	2442.9

Loss-of-Load Probability, LOLP = 0%

Table C.4

Example 1 : SYSINT Output Totals

STRATEGY ID = 1 TITLE :" SAMPLE SYSINT RUN PERFORMING CALCS. FOR EXAMPLES 1 & 2 "

PERIOD NUMBER = 1 TITLE :" EXAMPLE NO. 1 : DETERMINISTIC MODEL (NO FORCED-OUTAGES) "

					STARTUPS & SHUTDOWNS			EXPECTED PRODUCTION			TOTALS		
INDEX	IDNO	NAME	LD FACT	OPER HRS	NUMBER	MEGABTU	COST($)	ELECT(GWH)	MEGABTU	COST($)	MEGABTU	COST($)	INDEX
1	101	I	0.0	0.0	0.0	0.	0.	0.0	0.	0.	0.	0.	1
2	202	II	0.500000	730.0000	0.0000	0.	0.	73.00000	803000.	401500.	933000.	401500.	2
3	303	III	0.666667	730.0000	0.0000	0.	0.	146.00000	1606000.	305140.	1606000.	305140.	3
4	404	IV	0.400000	730.0000	0.0000	0.	0.	175.20000	1673160.	669264.	1673160.	669264.	4
5	505	V	0.950000	730.0000	0.0000	0.	0.	554.80000	5927600.	1066968.	5927600.	1066968.	5

```
P O W E R :                                 MEGAWATTS
        INSTALLED CAPACITY                    2000
        ON-LINE   CAPACITY                    2000
        PEAK LOAD FORECAST                    1800
        ON-LINE MARGIN @ PEAK                  200
        SPINNING RESERVE                         0
    LOSS-OF-LOAD PROBABILITY              0.0

E N E R G Y :                                 GWH
        EXPECTED DEMAND                     949.0000
        EXPECTED PRODUCTION                 949.0000
                ( NUCLEAR                   700.8000)
                (NON-NUCLEAR                248.2000)
        EXPECTED EMERG PURCH                  0.0000
        (UNSERVED BY DIRECT CALC              0.0    )

D O L L A R   C O S T :                     SYSTEM        NUCLEAR      NON-NUCLEAR
         PRODUCTION FUEL                  2442872.       1372108.       1070764.
         STARTUPS & SHUTDOWNS           _________0.    _________0.    _________0.
              SUB-TOTALS                  2442872.       1372108.       1070764.
         EMERG.PURCH.@ 10.00 $/MWH_________0.
              TOTAL                       2442872.
```

Table C.5

Example 1 : SYSINT Detailed Calculations

```
   L  IDAC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
*********************************************************************************************************************

WITHOUT PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19    PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
(1) L  IDAC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    5   505   300.     0   300   300 1.00000000  219.000000  219.000000    5
WITH    PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19    PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
*********************************************************************************************************************

WITHOUT PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19    PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
(2) L  IDAC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    3   303   400.     0   100   100 1.00000000   73.000000   73.000000    3
WITH    PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19    PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
*********************************************************************************************************************

WITHOUT PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19    PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
(3) L  IDAC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    4   404   600.     0   200   200 1.00000000  146.000000  146.000000    4
WITH    PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19    PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
*********************************************************************************************************************

WITHOUT PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19    PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
(4) L  IDAC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    2   202   700.     0   100   100 1.00000000   73.000000   73.000000    2
WITH    PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19    PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
*********************************************************************************************************************

WITHOUT PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18    PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
(5) L  IDAC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    5   505  1200.   300   500   800 0.92000000  335.800000  554.800000    5
WITH    PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18    PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
*********************************************************************************************************************

WITHOUT PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18    PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
(6) L  IDAC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    3   303  1400.   100   200   300 0.90000000   73.000000  146.000000    3
WITH    PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18    PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
*********************************************************************************************************************

WITHOUT PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18    PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
(7) L  IDAC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    4   404  1800.   200   400   600 0.10000000   29.200000  175.200000    4
WITH    PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18    PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
*********************************************************************************************************************

WITHOUT PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18    PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
(8) L  IDAC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    2   202  1500.   100   100   200 0.0          0.0         73.000000    2
WITH    PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18    PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
*********************************************************************************************************************

WITHOUT PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18    PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
(9) L  IDAC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    1   101  2000.     0   100   100 0.0          0.0         0.0          1
WITH    PLANT OF INTEREST       PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18    PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
```

Table C.6

Example 2 on Reference Utility System:
"Deterministic Model (Reduced Capacities)"

(See Sect. 2.2.1 for further details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 $)
I	1	9	2.51	40.7
II	1	4	69.35	381.4
	2	8	5.81	24.7
III	1	2	65.70	149.8
	2	6	108.82	206.8
IV	1	3	131.40	515.1
	2	7	79.85	265.1
V	1	1	186.15	418.8
	2	5	299.30	511.8
Utility Production			948.89	2514.2
Emergency Purchases (at 10$/MWH)			0.11	1.1
Total			949.00	2515.3

Loss-of-Load Probability, LOLP = 1.25%

Table C.7

Example 2 : SYSINT Output Totals

STRATEGY ID = 1 TITLE :" SAMPLE SYSINT RUN PERFORMING CALCS. FOR EXAMPLES 1 & 2 "

PERIOD NUMBER = 1 TITLE :" EXAMPLE NO. 2 : DETERMINISTIC MODEL (REDUCED CAPACITIES) "

					STARTUPS & SHUTDOWNS			EXPECTED PRODUCTION			TOTALS		
INDEX	LOAD	NAME	LD FACT	OPER HRS	NUMBER	MEGABTU	COST($)	ELECT(GWH)	MEGABTU	COST($)	MEGABTU	COST($)	INDEX
1	101	I	0.036250	26.4625	0.4410	22.	20.	2.51394	45251.	40726.	45273.	40746.	1
2	202	II	0.541875	730.0000	0.0000	0.	0.	75.15806	812219.	406109.	812219.	406109.	2
3	303	III	0.685443	730.0000	0.0000	0.	0.	174.52019	1876602.	356554.	1876602.	356554.	3
4	404	IV	0.535891	730.0000	0.0000	0.	0.	211.24831	1950461.	780184.	1950461.	780184.	4
5	505	V	0.577932	730.0000	0.0000	0.	0.	485.44544	5170182.	930633.	5170182.	930633.	5

P O W E R :	MEGAWATTS
INSTALLED CAPACITY	1775
ON-LINE CAPACITY	1775
PEAK LOAD FORECAST	1800
ON-LINE MARGIN @ PEAK	-25
SPINNING RESERVE	0
LOSS-OF-LOAD PROBABILITY	0.012500

E N E R G Y :	GWH
EXPECTED DEMAND	949.0000
EXPECTED PRODUCTION	948.8859
(NUCLEAR	659.9656)
(NON-NUCLEAR	288.9203)
EXPECTED EMERG PURCH	0.1141
(UNSERVED BY DIRECT CALC	0.1141)

D O L L A R C O S T :	SYSTEM	NUCLEAR	NON-NUCLEAR
PRODUCTION FUEL	2514207.	1287187.	1227019.
STARTUPS & SHUTDOWNS	20.	0.	20.
SUB-TOTALS	2514226.	1287187.	1227039.
EMERG.PURCH.@ 10.00 $/MWH	1141.		
TOTAL	2515367.		

Table C.8

Example 2 : SYSINT Detailed Calculations

```
  L  IDNC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L

**************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19     PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
(1) L  IDNC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    5   505   255.     0   255   255  1.000000000  186.150000  186.150000   5
WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19     PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0

**************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19     PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
(2) L  IDNC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    3   301   345.     0    90    90  1.000000000   65.700000   65.700000   3
WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19     PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0

**************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19     PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
(3) L  IDNC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    4   404   525.     0   180   180  1.000000000  131.400000  131.400000   4
WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19     PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0

**************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19     PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0
(4) L  IDNC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    2   202   420.     0    95    95  1.000000000   69.350000   69.350000   2
WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    19     PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0

**************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18     PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
(5) L  IDNC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    5   505  1045.   255   425   680  0.96469118   299.295437  485.445437   5
WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18     PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0

**************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18     PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
(6) L  IDNC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    3   303  1225.    50   180   270  0.82015512   108.820108  174.520108   3
WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18     PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0

**************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18     PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
(7) L  IDNC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    4   404  1565.   180   360   540  0.30303661    79.848312  211.248313   4
WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18     PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0

**************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18     PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
(8) L  IDNC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    2   202  1680.    95    95   190  0.08375000     5.800062   75.150062   2
WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18     PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0

**************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18     PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
(9) L  IDNC    PE    MWIN MWADD MWTOT   AVPROB       DELGWH      EXPGWH    L
    1   101  1775.     0    95    95  0.03625000     2.513938    2.513938   1
WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =    18     PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0
```

Table C.9

Example 3 on Reference Utility System:
"Probabilistic Model (With Forced-Outages)"

(See Sect. 2.2.1 for further details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 $)
I	1	9	11.93	193.3
II	1	4	69.35	381.5
	2	8	14.01	59.5
III	1	2	65.70	149.8
	2	6	80.69	153.3
IV	1	3	131.40	515.1
	2	7	70.85	235.2
V	1	1	186.15	418.8
	2	5	288.81	493.9
Utility Production			918.89	2600.4
Emergency Purchases (at 10$/MWH)			30.11	301.1
Total			949.00	2901.5

Loss-of-Load Probability, LOLP = 15.6%

Table C.10

Example 3 : SYSINT Output Totals

STRATEGY IC = 2 TITLE :" SAMPLE SYSINT FUN PERFCRMING CALCS. FOR EXAMPLES 3 THRU 5 "

PERICC NUMBER = 1 TITLE :" EXAMPLE NO. 3 : PROBABILISTIC MODEL (WITH FORCED-CUTAGES) "

					STARTUPS & SHUTDOWNS			EXPECTED PRODUCTION			TOTALS		
INDEX	IDNO	NAME	LD FACT	CPER HRS	NUMBER	MEGABTU	CCST($)	ELECT(GWH)	MEGABTU	COST($)	MEGABTU	COST($)	INDEX
1	101	I	0.163473	119.3353	4.59C1	230.	2C7.	11.93353	214EC4.	193323.	215033.	19353C.	1
2	2C2	II	0.57C932	653.5000	0.0000	0.	0.	83.35610	881902.	440951.	381902.	440551.	2
3	303	III	0.668428	657.0C00	C.CCCO	0.	0.	146.38576	1595258.	303099.	1595258.	303095.	3
4	4C4	IV	0.461762	657.0C00	0.0000	0.	0.	2C2.25171	1875789.	750316.	1875789.	750316.	4
5	505	V	0.613251	620.5000	C.0000	0.	0.	474.96171	5070586.	912706.	5070586.	912706.	5

P O W E R :	MEGAWATTS
INSTALLED CAPACITY	2000
ON-LINE CAPACITY	2C00
PEAK LOAD FORECAST	1800
ON-LINE MARGIN @ PEAK	200
SPINNING RESERVE	0
LOSS-OF-LOAD PROBABILITY	0.156470

E N E R G Y :	GWH
EXPECTED DEMAND	949.0000
EXPECTED PRODUCTION	918.8888
(NUCLEAR	621.3475)
(NON-NUCLEAR	297.5413)
EXPECTED EMERG PURCH	30.1112
(UNSERVED BY DIRECT CALC	30.1112)

D O L L A R C O S T :	SYSTEM	NUCLEAR	NON-NUCLEAR
PRODUCTION FUEL	2600394.	1215804.	1384590.
STARTUPS & SHUTDOWNS	207.	0.	207.
SUB-TOTALS	26006C1.	1215804.	1384796.
EMERG.PURCH.@ 10.00 $/MWH	301112.		
TOTAL	2901713.		

Table C.11

Example 3 : SYSINT Detailed Calculations

```
   L  IDAC    PE    PMIN PWADD MWTOT   AVPPCB        DELGMH       EXPGMH    L

*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   19     PEMAX =   1900.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.950000000 0.900000000
 0.850000000 0.800000000 0.500000000 0.200000000 0.150000000 0.100000000 0.050000000 0.0         0.0

(1) L  IDAC    PE    PMIN PWADD MWTOT   AVPPCB        DELGMH       EXPGMH    L
    5   505   300.     0   300   300 1.00000003  186.149992   186.149992    5

WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   21     PEMAX =   2100.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.957500002 0.915000004
 0.872500005 0.822500005 0.560000014 0.257500023 0.247500023 0.160000014 0.072500005 0.022500005 0.015000004 0.007500002
 0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   21     PEMAX =   2100.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.957500002 0.915000004
 0.872500005 0.822500005 0.560000014 0.257500023 0.247500023 0.160000014 0.072500005 0.022500005 0.015000004 0.007500002
 0.0

(2) L  IDNO    PE    PMIN PWADD MWTOT   AVPPCB        DELGMH       EXPGMH    L
    3   303   400.     0   100   100 1.00000000   65.699998    65.699998    3

WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   22     PEMAX =   2200.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.961750003 0.915250004
 0.876350006 0.827500007 0.586250020 0.323750029 0.252500024 0.168750017 0.081250008 0.027500007 0.015750004 0.008250007
 0.000750000 0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   22     PEMAX =   2200.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.961750003 0.915250004
 0.876350006 0.827500007 0.586250020 0.323750029 0.252500024 0.168750017 0.081250008 0.027500007 0.015750004 0.008250002
 0.000750000 0.0

(3) L  IDAC    PE    MMIN MWADD MWTOT   AVPHOR        DELGMH       EXPGMH    L
    4   404   600.     0   200   200 1.00000000  131.399997   131.399997    4

WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   24     PEMAX =   2400.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.965575003 0.927325006
 0.885250008 0.836675009 0.615300025 0.374125038 0.285875032 0.184250322 0.098375314 0.041625011 0.022300006 0.010175003
 0.002250001 0.000825000 0.000075000 0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   24     PEMAX =   2400.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.965575003 0.927325006
 0.885250008 0.836675009 0.615300025 0.374125038 0.285875032 0.184250322 0.098375314 0.041625011 0.022300006 0.010175003
 0.002250001 0.000825000 0.000075000 0.0

(4) L  IDAC    PE    PMIN PWADD MWTOT   AVPHCB        DELGMH       EXPGMH    L
    2   202   700.     0   100   100 1.00000000   69.349999    69.349999    2

WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   25     PEMAX =   2500.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.967296254 0.925237506
 0.887353758 0.839103759 0.626360777 0.386183791 0.290287533 0.189331274 0.102668765 0.044462512 0.023266256 0.010781253
 0.002446251 0.000856250 0.000112500 0.000003750 0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   22     PEMAX =   2200.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.961525003 0.918750004
 0.867475004 0.817500004 0.575125017 0.301250023 0.197250012 0.121250006 0.067625004 0.017500004 0.005975002 0.000750000
 0.000025000 0.0

(5) L  IDNC    PE    MMIN MWADD MWTOT   AVPPCB        DELGMH       EXPGMH    L
    5   505  1200.   300   500   800 0.93090000  288.811714   434.561706    5

WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   30     PEMAX =   3000.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.967296254 0.925237506
 0.887353758 0.844875010 0.638856279 0.406062545 0.317662539 0.253062536 0.201710036 0.152187536 0.135233313 0.123262533
 0.084290023 0.045187514 0.024587509 0.018187505 0.010143753 0.002625001 0.000896250 0.000112500 0.000003750 0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   29     PEMAX =   2900.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.963662503 0.925612506
 0.881125007 0.840625009 0.616437526 0.382687541 0.310437537 0.246687535 0.196712535 0.147462535 0.133837532 0.122387530
 0.082312521 0.041062512 0.028312508 0.017062505 0.009375003 0.001875001 0.000787500 0.000037500 0.0

(6) L  IDAC    PE    PMIN PWADD MWTOT   AVPPCB        DELGMH       EXPGMH    L
    3   303  1400.   100   200   300 0.61400693   83.685761   146.385759    3

WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   32     PEMAX =   3200.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.967296254 0.932871257
 0.894812509 0.852928762 0.647335031 0.432731299 0.363456257 0.283662543 0.215310040 0.161760019 0.145122535 0.125550032
 0.088827524 0.050340017 0.037690013 0.023587508 0.012541754 0.004518752 0.002415001 0.000971251 0.000187500 0.000078750
 0.000003750 0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   33     PEMAX =   3300.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.963662503 0.925612506
 0.887162508 0.844875010 0.620687527 0.386937541 0.334875042 0.272187540 0.202025036 0.151712516 0.138800013 0.127037511
 0.083275022 0.041812513 0.032625010 0.021562537 0.013312503 0.002625001 0.001537501 0.000787510 0.000037500 0.0

(7) L  IDAC    PE    PMIN PWADD MWTOT   AVPPCB        DELGMH       EXPGMH    L
    4   404  1800.   200   400   600 0.26940316   73.851710   232.251736    4

WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   36     PEMAX =   3600.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.967296254 0.932471257
 0.898444260 0.860381513 0.658610783 0.448243032 0.397751803 0.337510052 0.270538800 0.221028030 0.186988799 0.153072538
 0.108435030 0.066850021 0.049565017 0.034577513 0.023161259 0.015671257 0.005711255 0.004800001 0.001296252 0.000715625
 0.000131251 0.000282500 0.000193750 0.000078750 0.000033750 0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   35     PEMAX =   3500.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.965575003 0.931150007
 0.896325010 0.858475012 0.648100031 0.437725050 0.395450052 0.334450051 0.267175049 0.218600049 0.185325046 0.151375037
 0.104175025 0.062675020 0.048875016 0.033825012 0.022625009 0.016475007 0.009450005 0.003725002 0.002225002 0.001200001
 0.000675001 0.000225000 0.000150000 0.000075000 0.0

(8) L  IDAC    PE    MMIN MWADD MWTOT   AVPPCB        DELGMH       EXPGMH    L
    2   202  1950.   100   100   200 0.20158255   14.306102    85.356102    2

WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   37     PEMAX =   3700.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.967296254 0.934592507
 0.900167910 0.862101763 0.660531283 0.458762554 0.408272554 0.339613802 0.273598800 0.224592551 0.189417545 0.154716289
 0.110132930 0.067110022 0.051740018 0.035267513 0.023913760 0.015632507 0.010107535 0.003515125 0.003356252 0.002227501
 0.001087501 0.000318750 0.000191250 0.000087500 0.000007500 0.000003750 0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   37     PEMAX =   3700.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.967296254 0.934592507
 0.900167910 0.862101763 0.660531283 0.458762554 0.408272554 0.339613802 0.273598800 0.224592551 0.189417545 0.154716289
 0.110132930 0.067110022 0.051740018 0.035267513 0.023913760 0.015632507 0.010107535 0.003515125 0.003356252 0.002227501
 0.001087501 0.000318750 0.000191250 0.000087500 0.000007500 0.000003750 0.0

(9) L  IDAC    PE    MMIN MWADD MWTOT   AVPPCB        DELGMH       EXPGMH    L
    1   101  2600.     0   100   100 0.17207692   11.933934    11.933934    1

WITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :          DM =   100.00     IEMAX =   38     PEMAX =   3800.0010
 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 1.000000000 0.968971441 [illegible]
 0.901881361 0.864011701 0.670610160 0.468850992 0.410797055 0.343066741 0.276859551 0.226857466 0.191186[illegible] 0.15587[illegible]
 0.112362719 0.069261148 0.052538518 0.036291137 0.024681467 0.016066970 0.010144755 0.005599065 0.004617[illegible] [illegible]
 0.001144501 0.000353188 0.000197625 0.000087938 0.000011250 0.000003938 0.000000148 0.0
```

Table C.12

Example 4 on Reference Utility System:

"Single Increment Booth-Baleriaux Model"

(See Sect. 3.3.1.3 forfurther details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 $)
I	1	5	11.93	193.3
II	1 2	4	30.85	152.2
III	1 2	2	184.54	375.0
IV	1 2	3	195.17	710.6
V	1 2	1	496.40	949.4
	Utility Production		918.89	2380.5
	Emergency Purchases (10 $/MWH)		30.11	301.1
	Total		949.00	2681.6

Loss-of-load Probability, LOLP = 15.6%

Table C.13

Example 4 : SYSINT Output Totals

STRATEGY ID = 2 TITLE :" SAMPLE SYSINT RUN PERFORMING CALCS. FOR EXAMPLES 3 THRU 5 "

PERIOD NUMBER = 2 TITLE :" EXAMPLE NO. 4 : SINGLE INCREMENT BOOTH-BALERIAUX MODEL "

					STARTUPS & SHUTDOWNS			EXPECTED PRODUCTION			TOTALS		
INDEX	IDNO	NAME	LD FACT	OPER HRS	NUMBER	MEGABTU	COST($)	ELECT(GWH)	MEGABTU	COST($)	MEGABTU	COST($)	INDEX
1	101	I	0.163473	119.3353	4.5901	230.	207.	11.93353	214804.	193323.	215033.	193530.	1
2	202	II	0.211304	168.4425	8.1273	6502.	3251.	30.85035	304339.	152169.	310840.	155420.	2
3	303	III	0.842625	643.0387	0.3878	465.	88.	184.53487	1973956.	375052.	1974422.	375140.	3
4	404	IV	0.445594	522.2739	12.1473	43730.	17492.	195.17007	1776594.	710638.	1820324.	728130.	4
5	505	V	0.850000	620.5000	0.0000	0.	0.	496.39998	5274250.	949365.	5274250.	949365.	5

```
P O W E R :                        MEGAWATTS
        INSTALLED CAPACITY           2000
        ON-LINE   CAPACITY           2000
        PEAK LOAD FORECAST           1800
        ON-LINE MARGIN @ PEAK         200
        SPINNING RESERVE       **********
    LOSS-OF-LOAD PROBABILITY     0.156470

E N E R G Y :                        GWH
        EXPECTED DEMAND           949.0000
        EXPECTED PRODUCTION       918.8888
                 ( NUCLEAR        680.9348)
                 (NON-NUCLEAR     237.9540)
        EXPECTED EMERG PURCH       30.1112
        (UNSERVED BY DIRECT CALC   30.1112)

D O L L A R   C O S T :           SYSTEM       NUCLEAR    NON-NUCLEAR
         PRODUCTION FUEL          2380547.     1324417.     1056130.
         STARTUPS & SHUTDOWNS     ____21028.   _______88.   ____20950.
              SUB-TOTALS          2401585.     1324505.     1077080.
         EMERG.PURCH.@ 10.00 $/MWH___301112.
              TOTAL               2702697.
```

Table C.14

Example 4 : SYSINT Detailed Calculations

```
  L  ICNC    PE    MWIN MWADD MWTOT    AVPRCB        DELGWH      EXPGWH     L
*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   19     PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0

(1)  L  ICNC    PE    MWIN MWADD MWTOT    AVPRCB        DELGWH      EXPGWH     L
     5   505   300.     0   300   300  1.00000000   186.149492  186.149492    5

WITH      PLANT OF INTEREST      PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   21     PEMAX =    2100.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.957500002  0.915000004
 0.872500005  0.822500005  0.560000014  0.297500023  0.247500023  0.165000016  0.372500035  0.022500005  0.015000004  0.007500002
 0.0
*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   18     PEMAX =    1800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0

(2)  L  ICNC    PE    MWIN MWADD MWTOT    AVPROB        DELGWH      EXPGWH     L
     5   505   800.   300   500   800  1.00000000   310.249987  456.359979    5

WITH      PLANT OF INTEREST      PROB(K),K=1,IEMAX :            DP =   100.00      IEMAX =   26     PEMAX =    2600.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.957500002  0.915000004
 0.872500005  0.830000007  0.575000018  0.320000029  0.277500030  0.235000032  0.185000032  0.135000032  0.127500030  0.120000029
 0.075000018  0.030000007  0.022500005  0.015000004  0.007500002  0.0
*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   26     PEMAX =    2600.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.957500002  0.915000004
 0.872500005  0.830000007  0.575000018  0.320000029  0.277500030  0.235000032  0.185000032  0.135000032  0.127500030  0.120000025
 0.075000018  0.030000007  0.022500005  0.015000004  0.007500002  0.0

(3)  L  IDNC    PE    MWIN MWADD MWTOT    AVPRCB        DELGWH      EXPGWH     L
     3   303   900.     0   100   100  0.57875000    64.303873   64.303873    3

WITH      PLANT OF INTEREST      PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   27     PEMAX =    2700.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.961750003  0.915250004
 0.876150006  0.834250008  0.600500023  0.345500034  0.281750031  0.239250033  0.190000033  0.140000033  0.128250031  0.120750029
 0.075900020  0.034500009  0.023250006  0.015750004  0.008250002  0.000750000  0.0
*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   26     PEMAX =    2600.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.957500002  0.915000004
 0.872500005  0.830000007  0.575000018  0.320000029  0.277500030  0.235000032  0.185000032  0.135000032  0.127500030  0.120000029
 0.075000018  0.030000007  0.022500005  0.015000004  0.007500002  0.0

(4)  L  IDNO    PE    MWIN MWADD MWTOT    AVPROB        DELGWH      EXPGWH     L
     3   303  1100.   100   200   300  0.91500000   120.230297  184.534871    3

WITH      PLANT OF INTEREST      PROB(K),K=1,IEMAX :            DP =   100.00      IEMAX =   29     PEMAX =    2900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.961750003  0.923500005
 0.885250008  0.842750010  0.609000025  0.375250039  0.332750041  0.269000039  0.198500035  0.149250035  0.138250033  0.126500031
 0.081000021  0.039750012  0.032250010  0.021000006  0.009750003  0.002250001  0.001500001  0.000750000  0.0
*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :            DP =   100.00      IEMAX =   29     PEMAX =    2900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.961750003  0.923500005
 0.885250008  0.842750010  0.609000025  0.375250039  0.332750041  0.269000039  0.198500035  0.149250035  0.138250033  0.126500031
 0.081000021  0.039750012  0.032250010  0.021000006  0.009750003  0.002250001  0.001500001  0.000750000  0.0

(5)  L  IDNC    PE    MWIN MWADC MWTOT    AVPRCB        DELGWH      EXPGWH     L
     4   404  1300.     0   200   200  0.75453751   104.454786  104.454786    4

WITH      PLANT OF INTEREST      PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   31     PEMAX =    3100.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.931150007
 0.892900009  0.850825011  0.636625029  0.422000048  0.360375046  0.279625041  0.211925039  0.161225039  0.144275035  0.128775032
 0.086125023  0.048425016  0.037125012  0.022875007  0.012000004  0.004125002  0.002325001  0.001900000  0.001500000  0.000375000
 0.0
*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   29     PEMAX =    2900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.961750003  0.923500005
 0.885250008  0.842750010  0.609000025  0.375250039  0.332750041  0.269000039  0.198500035  0.149250035  0.138250033  0.126500031
 0.081000021  0.039750012  0.032250010  0.021000006  0.009750003  0.002250001  0.001500001  0.000750000  0.0

(6)  L  IDNC    PE    MWIN MWADC MWTOT    AVPRCB        DELGWH      EXPGWH     L
     4   404  1700.   200   400   600  0.34518754    93.715282  155.170069    4

WITH      PLANT OF INTEREST      PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   35     PEMAX =    3500.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.931150007
 0.896725010  0.858475012  0.648100031  0.437725050  0.395650052  0.334450051  0.267175049  0.218600049  0.185325044  0.151375037
 0.108175029  0.062675020  0.048875016  0.031825012  0.022600009  0.014675007  0.009450005  0.004650002  0.003225002  0.002100001
 0.000975001  0.000225000  0.000150000  0.000075000  0.0
*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   35     PEMAX =    3500.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.931150007
 0.896725010  0.858475012  0.648100031  0.437725050  0.395650052  0.334450051  0.267175049  0.218600049  0.185325044  0.151375037
 0.108175029  0.062675020  0.048875016  0.031825012  0.022600009  0.014675007  0.009450005  0.004650002  0.003225002  0.002100001
 0.000975001  0.000225000  0.000150000  0.000075000  0.0

(7)  L  IDNO    PE    MWIN MWADD MWTOT    AVPRCB        DELGWH      EXPGWH     L
     2   202  1800.     0   100   100  0.24281755    16.844251   16.844251    2

WITH      PLANT OF INTEREST      PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   36     PEMAX =    3600.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.967296256  0.931871257
 0.858446260  0.800387513  0.658618783  0.448243802  0.397753031  0.337510052  0.276538800  0.221028800  0.186988794  0.153072538
 0.108435030  0.064050021  0.045565017  0.034577513  0.023161250  0.015071257  0.009711255  0.004890033  0.003296252  0.002154251
 0.001031251  0.000262500  0.000153750  0.000078750  0.000003750  0.0
*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   35     PEMAX =    3500.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.931150007
 0.896725010  0.858475012  0.648100031  0.437725050  0.395650052  0.334450051  0.267175049  0.218600049  0.185325044  0.151375037
 0.108175029  0.062675020  0.048875016  0.031825012  0.022600009  0.014675007  0.009450005  0.004650002  0.003225002  0.002100001
 0.000975001  0.000225000  0.000150000  0.000075000  0.0

(8)  L  IDNC    PE    MWIN MWADD MWTOT    AVPRCB        DELGWH      EXPGWH     L
     2   202  1900.   100   100   200  0.20156255    14.006102   30.850354    2

WITH      PLANT OF INTEREST      PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   37     PEMAX =    3700.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.967296256  0.934592507
 0.900167910  0.862108763  0.660531283  0.458762554  0.408272554  0.339613802  0.273598830  0.224392551  0.189417545  0.154736284
 0.110132530  0.067110022  0.051740018  0.035267513  0.023913760  0.015632507  0.010107505  0.005151255  0.003536252  0.002227501
 0.001087501  0.000318750  0.000191250  0.000082500  0.000007500  0.000003750  0.0
*****************************************************************************************************

WITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :            DM =   100.00      IEMAX =   37     PEMAX =    3700.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.967296256  0.934592507
 0.900167910  0.862108763  0.660531283  0.458762554  0.408272554  0.339613802  0.273598830  0.224392551  0.189417545  0.154736284
 0.110132530  0.067110022  0.051740018  0.035267513  0.023913760  0.015632507  0.010107505  0.005151255  0.003536252  0.002227501
 0.001087501  0.000318750  0.000191250  0.000082500  0.000007500  0.000003750  0.0

(9)  L  ICNC    PE    MWIN MWADC MWTOT    AVPRCB        DELGWH      EXPGWH     L
     1   101  2000.     0   100   100  0.17207692    11.933534   11.933534    1

WITH      PLANT OF INTEREST      PROB(K),K=1,IEMAX :            DP =   100.00      IEMAX =   38     PEMAX =    3800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.968931441  0.938227695
 0.901888761  0.864011701  0.670610160  0.468850992  0.410797059  0.343046741  0.276899551  0.226457864  0.191166245  0.156476352
 0.112362719  0.069261148  0.052508518  0.036091139  0.024481447  0.016046570  0.010183755  0.005399065  0.003617002  0.002292939
 0.001144501  0.000357188  0.000197625  0.000087938  0.000011250  0.000003938  0.000000188  0.0
```

Table C.15

Example 5 on Reference Utility System:

"Multiple Increment Booth-Baleriaux Model (V-2, then III-2)"

(Among Nuclear Upper Increments V-2, then III-2)

(See Sect. 3.3.2.1 for further details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 $)
I	1	9	11.93	193.3
II	1	6	36.71	201.9
	2	8	14.01	59.5
III	1	2	65.70	149.8
(Nuclear)	2	5	103.90	197.4
IV	1	3	131.40	515.1
	2	7	70.85	235.2
V	1	1	186.15	418.8
(Nuclear)	2	4	298.24	510.0
	Utility Production		918.89	2481.0
	Emergency Purchases (10 $/MWH)		30.11	301.1
	Total		949.00	2782.1

Loss-of-load Probability, LOLP = 15.6%

Table C.16

Example 5 : SYSINT Output Totals

STRATEGY ID = 2 TITLE :" SAMPLE SYSINT RUN PERFORMING CALCS. FOR EXAMPLES 3 THRU 5 "

PERIOD NUMBER = 3 TITLE :" EXAMPLE NO. 5 : MULTIPLE INCREMENT BOOTH-BALERIAUX MODEL (V-2,THEN III-2) "

					STARTUPS & SHUTDOWNS			EXPECTED PRODUCTION			TOTALS		
INDEX	IDNO	NAME	LD FACT	OPER HRS	NUMBER	MEGABTU	COST($)	ELECT(GWH)	MEGABTU	COST($)	MEGABTU	COST($)	INDEX
1	1C1	I	0.163473	119.3353	4.5501	230.	207.	11.93353	214804.	193323.	215033.	193530.	1
2	2C2	II	0.347356	367.0782	26.1583	20927.	10463.	50.71393	522838.	261419.	543765.	271892.	2
3	3C3	III	0.774412	657.0000	0.0000	0.	0.	169.59633	1927363.	347199.	1827363.	347199.	3
4	404	IV	0.461762	657.0000	0.0000	0.	0.	202.25171	1875789.	750316.	1375789.	750316.	4
5	5C5	V	0.829441	620.5000	0.0000	0.	0.	484.39331	5163186.	929834.	5163186.	929834.	5

P O W E R :	MEGAWATTS
INSTALLED CAPACITY	2000
ON-LINE CAPACITY	2000
PEAK LOAD FORECAST	1800
ON-LINE MARGIN @ PEAK	200
SPINNING RESERVE	80
LOSS-OF-LOAD PROBABILITY	0.156470

E N E R G Y :	GWH
EXPECTED DEMAND	949.0000
EXPECTED PRODUCTION	918.8888
(NUCLEAR	653.9896)
(NON-NUCLEAR	264.8992)
EXPECTED EMERG PURCH	30.1112
(UNSERVED BY DIRECT CALC	30.1112)

D O L L A R C O S T :	SYSTEM	NUCLEAR	NON-NUCLEAR
PRODUCTION FUEL	2481090.	1276033.	1205058.
STARTUPS & SHUTDOWNS	10670.	0.	10670.
SUB-TOTALS	2491760.	1276033.	1215728.
EMERG.PURCH.@ 10.00 $/MWH	301112.		
TOTAL	2792872.		

Table C.17

Example 5 : SYSINT Detailed Calculations

```
   L  IDAC    PE     PMIN PWADD PWTOT    AVPPCB         DELGWH        EXPGWH     L

*********************************************************************************************************

 WITHOUT PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     19      PEMAX =     1900.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
  0.850000000  0.800000000  0.550000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0

①  L  IDAC    PE     PMIN PWADD PWTOT    AVPPCB         DELGWH        EXPGWH     L
   5   505   100.      0   500   500  1.00000000  186.149992    186.149992     5

 WITH    PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     21      PEMAX =     2100.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.957500002  0.915000004
  0.872500005  0.822500005  0.560000014  0.257500023  0.247500023  0.160000014  0.072500005  0.022500005  0.015000004  0.007500002
  0.0

*********************************************************************************************************

 WITHOUT PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     21      PEMAX =     2100.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.957500002  0.915000004
  0.872500005  0.822500005  0.560000014  0.257500023  0.247500023  0.160000014  0.072500005  0.022500005  0.015000004  0.007500002
  0.0

②  L  IDAC    PE     PMIN PWADD PWTOT    AVPPCB         DELGWH        EXPGWH     L
   3   303   400.      0   100   100  1.00000000   65.659998     65.659998     3

 WITH    PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     22      PEMAX =     2200.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.961750003  0.915250004
  0.876750006  0.827500007  0.586250020  0.323750029  0.252500024  0.168750017  0.081250008  0.027500007  0.015750004  0.008250002
  0.000750000  0.0

*********************************************************************************************************

 WITHOUT PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     22      PEMAX =     2200.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.961750003  0.915250004
  0.876750006  0.827500007  0.586250020  0.323750029  0.252500024  0.168750017  0.081250008  0.027500007  0.015750004  0.008250002
  0.000750000  0.0

③  L  IDAC    PE     PMIN PWADD PWTOT    AVPPCB         DELGWH        EXPGWH     L
   4   404   800.      0   200   200  1.00000000  131.399997    131.399997     4

 WITH    PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     24      PEMAX =     2400.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.927325006
  0.885250008  0.836675009  0.615300025  0.374125038  0.285675032  0.184250022  0.098375014  0.041625011  0.022100006  0.010175003
  0.002250001  0.000825000  0.000075000  0.0

*********************************************************************************************************

 WITHOUT PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     21      PEMAX =     2100.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.959500002  0.914500003
  0.849500004  0.815000004  0.562500014  0.287500020  0.192500011  0.117500005  0.365000004  0.315000004  0.005500001  0.000500000
  0.0

④  L  IDAC    PE     PMIN PWADD PWTOT    AVPPCB         DELGWH        EXPGWH     L
   5   505  1100.    300   500   800  0.96130000  298.243313    464.393505     5

 WITH    PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     29      PEMAX =     2900.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.927325006
  0.885250008  0.842750010  0.628125028  0.394375043  0.313625038  0.249875036  0.199175035  0.149925036  0.134425032  0.122675030
  0.084375022  0.043125013  0.028875008  0.017625005  0.009750003  0.002250001  0.000825000  0.000075000  0.0

*********************************************************************************************************

 WITHOUT PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     28      PEMAX =     2800.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.961750003  0.923500005
  0.881000007  0.838500009  0.604750024  0.371000039  0.307250036  0.243500034  0.194250034  0.145000035  0.133250032  0.121500029
  0.080250020  0.035000011  0.027750008  0.016500004  0.009000003  0.001500001  0.000750000  0.0

⑤  L  IDAC    PE     PMIN PWADD PWTOT    AVPPCB         DELGWH        EXPGWH     L
   3   303  1300.    100   200   300  0.79068751  103.896136    169.556335     3

 WITH    PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     31      PEMAX =     3100.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.931150007
  0.892900009  0.853825011  0.636625029  0.422000048  0.360375046  0.279625041  0.211925039  0.161225035  0.144275035  0.128775032
  0.086725023  0.048425016  0.037125012  0.022875007  0.012000004  0.004125002  0.002325001  0.000900000  0.000150000  0.000075000
  0.0

*********************************************************************************************************

 WITHOUT PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     31      PEMAX =     3100.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.931150007
  0.892900009  0.853825011  0.636625029  0.422000048  0.360375046  0.279625041  0.211925039  0.161225035  0.144275035  0.128775032
  0.086725023  0.048425016  0.037125012  0.022875007  0.012000004  0.004125002  0.002325001  0.000900000  0.000150000  0.000075000
  0.0

⑥  L  IDAC    PE     PMIN PWADD PWTOT    AVPPCB         DELGWH        EXPGWH     L
   2   202  1400.      0   100   100  0.52931254   36.707824     36.707824     2

 WITH    PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     32      PEMAX =     3200.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.967296256  0.932871257
  0.854812509  0.852928762  0.647335031  0.432731299  0.363456297  0.283662543  0.215310040  0.163763339  0.145122535  0.125990032
  0.088827524  0.050140017  0.037690013  0.023587508  0.012543354  0.004518752  0.002415001  0.000971251  0.000187500  0.000078750
  0.000003750  0.0

*********************************************************************************************************

 WITHOUT PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     34      PEMAX =     3400.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.963862503  0.925612506
  0.887162508  0.844875010  0.620687527  0.386937561  0.334875042  0.272187540  0.202025036  0.156712536  0.138800033  0.121037531
  0.081275022  0.041812513  0.032625010  0.021562507  0.010312503  0.002625001  0.001537501  0.000787501  0.000337500  0.0

⑦  L  IDAC    PE     PMIN PWADD PWTOT    AVPPCB         DELGWH        EXPGWH     L
   4   404  1800.    200   400   600  0.26960316   70.851710    202.251706     4

 WITH    PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     36      PEMAX =     3600.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.967296256  0.932871257
  0.858446260  0.860381513  0.658618783  0.448243802  0.397753803  0.337513352  0.270538833  0.221129833  0.186988796  0.153072516
  0.128435030  0.066850021  0.049565017  0.034577513  0.023161259  0.015071257  0.009711255  0.006890003  0.003296252  0.002196251
  0.001031251  0.000262500  0.000153750  0.000078750  0.000001750  0.0

*********************************************************************************************************

 WITHOUT PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     35      PEMAX =     3500.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.931150007
  0.896725010  0.858475012  0.648100031  0.437125050  0.395650052  0.334450051  0.267125049  0.218600049  0.185325044  0.151375037
  0.106175029  0.062675020  0.048875016  0.033825012  0.027600009  0.014675007  0.005400005  0.004650002  0.003225002  0.002133001
  0.000975001  0.000225000  0.000150000  0.000075000  0.0

⑧  L  IDAC    PE     PMIN PWADD PWTOT    AVPPCB         DELGWH        EXPGWH     L
   2   202  1900.    100   100   200  0.20196255   14.336102     50.713927     2

 WITH    PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     37      PEMAX =     3700.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.967296256  0.934592507
  0.900167510  0.862106763  0.660531283  0.458762554  0.408272554  0.339613802  0.273598800  0.224592551  0.189417565  0.154736765
  0.116132530  0.067110022  0.051740018  0.035267513  0.023913760  0.015632507  0.010107505  0.005151251  0.003536252  0.002227501
  0.001087501  0.000318750  0.000191250  0.000082500  0.000007500  0.000003750  0.0

*********************************************************************************************************

 WITHOUT PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     37      PEMAX =     3700.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.967296256  0.934592507
  0.900167510  0.862106763  0.660531283  0.458762554  0.408272554  0.339613802  0.273598800  0.224592551  0.189417565  0.154736765
  0.116132530  0.067110022  0.051740018  0.035267513  0.023913760  0.015632507  0.010107505  0.005151251  0.003536252  0.002227501
  0.001087501  0.000318150  0.000191250  0.000082500  0.000007500  0.000003750  0.0

⑨  L  IDAC    PE     PMIN PWADD PWTOT    AVPPCB         DELGWH        EXPGWH     L
   1   101  2000.      0   100   100  0.17207692   11.933534     11.933534     1

 WITH    PLANT OF INTEREST          PROB(K),K=1,IEMAX :             DM =    100.00      IEMAX =     38      PEMAX =     3800.0010
  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.968931441  0.938227695
  0.901888761  0.866011701  0.670610160  0.468850192  0.410797055  0.343046741  0.276699551  0.226852064  0.191166245  0.156670352
  0.112362719  0.065261148  0.052508518  0.036091139  0.024481467  0.016046570  0.010383755  0.005399065  0.003617302  0.002292939
  0.001144501  0.000393188  0.000197625  0.000087938  0.000011250  0.000001938  0.000000188  0.0
```

Table C.18

Example 6 on Reference Utility System:

"Multiple Increment Booth-Baleriaux Model (III-2, then V-2)"

(Among Nuclear Upper Increments III-2, then V-2)

(See Sect.3.3.3 for further details.)

Unit r	Increment i	Position in Loading Order	Increment Energy E_{ri} (GWH)	Increment Cost X_{ri} (10^3 $)
I	1	9	11.93	193.3
II	1	6	36.71	201.9
	2	8	14.01	59.5
III	1	2	65.70	149.8
(Nuclear)	2	4	131.40	249.7
IV	1	3	131.40	515.1
	2	7	70.85	235.2
V	1	1	186.15	418.8
(Nuclear)	2	5	270.74	463.0
	Utility Production		918.89	2486.3
	Emergency Purchases (10 $/MWH)		30.11	301.1
	Total		949.00	2787.4

Loss-of-load Probability, LOLP = 15.6%

Table C.19

Example 6 : SYSINT Output Totals

STRATEGY ID = 2 TITLE :" SAMPLE SYSINT RUN PERFORMING CALCS. FOR EXAMPLES 3 THRU 5 "

PERIOD NUMBER = 4 TITLE :" EXAMPLE NO. 6 : MULTIPLE INCREMENT BOOTH-BALERIAUX MODEL (III-2,THEN V-2) "

					STARTUPS & SHUTDOWNS			EXPECTED PRODUCTION			TOTALS		
INDEX	ICNO	NAME	LD FACT	OPER HRS	NUMBER	MEGABTU	COST($)	ELECT(GWH)	MEGABTU	COST($)	MEGABTU	COST($)	INDEX
1	101	I	0.163473	119.3353	4.5901	230.	207.	11.93353	214834.	193323.	215[illegible]33.	1[illegible]353[illegible].	1
2	202	II	0.347356	367.0782	26.1563	20927.	10463.	50.71393	522838.	261419.	543765.	271342.	2
3	303	III	0.900000	657.0000	0.0000	0.	0.	157.09999	2102400.	399456.	2102400.	399456.	3
4	404	IV	0.461762	657.0000	0.0000	0.	0.	202.25171	1875789.	750316.	1[illegible]75745.	750316.	4
5	505	V	0.782345	620.5000	0.0000	0.	0.	456.89964	4898902.	481302.	45[illegible]8902.	[illegible]81802.	5

```
P O W E R :                                  MEGAWATTS
        INSTALLED CAPACITY                     2000
        ON-LINE   CAPACITY                     2000
        PEAK LOAD FORECAST                     1800
        ON-LINE MARGIN @ PEAK                   200
        SPINNING RESERVE                   **********
    LOSS-OF-LOAD PROBABILITY               0.156470

E N E R G Y :                                GWH
        EXPECTED DEMAND                    949.0000
        EXPECTED PRODUCTION                918.8888
              (  NUCLEAR                   653.5896)
              (NON-NUCLEAR                 264.8552)
        EXPECTED EMERG PURCH                30.1112
        (UNSERVED BY DIRECT CALC            30.1112)

D O L L A R   C O S T :                     SYSTEM        NUCLEAR      NON-NUCLEAR
         PRODUCTION FUEL                   2486316.       1281258.      1205058.
         STARTUPS & SHUTDOWNS          ____10670.     _________0.    ____10670.
              SUB-TOTALS                   2496986.       1281258.      1215728.
         EMERG.PURCH.@ 10.00 $/MWH_____301112.
              TOTAL                        2798098.
```

Table C.20
Example 6 : SYSINT Detailed Calculations

```
  L  IDNC    PE     PMIN PMADD MWTOT    AVPROB        DELGWH       EXPGWH     L

*****************************************************************************************************

WITHOUT PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    19     PEMAX =    1900.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.950000000  0.900000000
 0.850000000  0.800000000  0.500000000  0.200000000  0.150000000  0.100000000  0.050000000  0.0          0.0

(1)  L  IDNC    PE     PMIN PMADD MWTOT    AVPROB        DELGWH       EXPGWH     L
     5   505   300.      0   300   300  1.00000000   186.149992   186.149992     5

WITH    PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    21     PEMAX =    2100.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000010  1.000000000  0.957500002  0.915000004
 0.872500005  0.822500005  0.560000014  0.297500023  0.267500023  0.160000014  0.072500005  0.022500005  0.015000004  0.007500002
 0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    21     PEMAX =    2100.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.957500002  0.915000004
 0.872500005  0.822500005  0.560000014  0.297500023  0.267500023  0.160000014  0.072500005  0.022500005  0.015000004  0.007500002
 0.0

(2)  L  IDNC    PE     PMIN PMADD MWTOT    AVPROB        DELGWH       EXPGWH     L
     3   303   400.      0   100   100  1.00000000    65.699998    65.699998     3

WITH    PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    22     PEMAX =    2200.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000030  0.961750003  0.915250004
 0.876750006  0.827500007  0.586250023  0.323750027  0.252500024  0.168750017  0.081250008  0.027500007  0.015750004  0.008250002
 0.006750000  0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    22     PEMAX =    2200.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000030  0.961750003  0.919250004
 0.876750006  0.827500007  0.586250023  0.323750029  0.252500024  0.168750017  0.081250008  0.027500007  0.015750004  0.008250002
 0.006750000  0.0

(3)  L  IDNC    PE     PMIN PMADD MWTOT    AVPROB        DELGWH       EXPGWH     L
     4   404   600.      0   200   200  1.00000000   131.399997   131.399997     4

WITH    PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    24     PEMAX =    2400.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.923325006
 0.885250008  0.836675005  0.615300025  0.374125038  0.285875032  0.184250022  0.098375014  0.041625011  0.022300006  0.010175001
 0.002250001  0.000825000  0.000075000  0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    23     PEMAX =    2300.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000030  0.961750003  0.923500005
 0.881000007  0.831750007  0.591250021  0.350000034  0.278750030  0.173750018  0.090000011  0.036250010  0.020750005  0.005000003
 0.006150001  0.000750000  0.0

(4)  L  IDNC    PE     PMIN PMADD MWTOT    AVPROB        DELGWH       EXPGWH     L
     3   303   800.    100   200   300  1.00000000   131.399997   157.659995     3

WITH    PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    26     PEMAX =    2600.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.931150007
 0.852500009  0.844750010  0.624475027  0.403100044  0.336050041  0.215500329  0.116000023  0.063500017  0.036350013  0.017100005
 0.004575002  0.002750001  0.000900000  0.000150000  0.000075000  0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    23     PEMAX =    2300.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.959500002  0.915000004
 0.874000005  0.824500006  0.572500017  0.323000027  0.247500023  0.152500015  0.080000008  0.027500007  0.015500004  0.000000002
 0.001000000  0.000500000  0.0

(5)  L  IDNC    PE     PMIN PMADD MWTOT    AVPROB        DELGWH       EXPGWH     L
     5   505  1300.    300   500   800  0.87265001   270.719653   456.885645     5

WITH    PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    31     PEMAX =    3100.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.965575003  0.931150007
 0.892900009  0.850425011  0.636625029  0.422000048  0.360375046  0.279625041  0.211925039  0.161225039  0.144275035  0.128775032
 0.086325023  0.048425016  0.037125012  0.022875007  0.012000004  0.004125002  0.002325001  0.000900000  0.000150000  0.000075000
 0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    31     PEMAX =    3100.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000030  0.965575003  0.931150007
 0.852500009  0.850825011  0.636625029  0.422000048  0.360375046  0.279625041  0.211925035  0.161225035  0.144275035  0.128775032
 0.086325023  0.048425016  0.037125012  0.022875007  0.012000004  0.004125002  0.002325001  0.000900000  0.000150000  0.000075000
 0.0

(6)  L  IDNC    PE     PMIN PMADD MWTOT    AVPROB        DELGWH       EXPGWH     L
     2   202  1400.      0   100   100  0.52531254    36.707824    36.707824     2

WITH    PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    32     PEMAX =    3200.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.967296254  0.932871257
 0.894812509  0.852926762  0.647335031  0.432731299  0.363456257  0.283652543  0.215310040  0.163760039  0.145122535  0.129550032
 0.088827524  0.050340317  0.037690013  0.023587508  0.012563754  0.004518752  0.002415001  0.000971251  0.000187500  0.000078750
 0.000003750  0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    30     PEMAX =    3000.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.963667503  0.929612506
 0.887162508  0.844875010  0.620687527  0.396937541  0.334375042  0.272187543  0.202325036  0.151712536  0.138833333  0.127387531
 0.083275022  0.041812513  0.032625010  0.021562507  0.010312503  0.002625001  0.001537501  0.000787500  0.000037500  0.0

(7)  L  IDNC    PE     PMIN PMADD MWTOT    AVPROB        DELGWH       EXPGWH     L
     4   404  1600.    200   400   600  0.26563316    70.851710   202.251706     4

WITH    PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    36     PEMAX =    3600.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  0.967296254  0.932871257
 0.898446260  0.860387513  0.658610783  0.448243802  0.397753803  0.337510052  0.270538800  0.221028800  0.186988794  0.153072538
 0.108435030  0.066485021  0.049565017  0.034577513  0.023161259  0.015071257  0.005711255  0.004890000  0.003296252  0.002156251
 0.001031251  0.000262500  0.000153753  0.000078750  0.000003750  0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    35     PEMAX =    3500.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000010  0.965575003  0.931150007
 0.896125010  0.858475012  0.648100031  0.437725050  0.395650052  0.334450051  0.267175045  0.218625045  0.185325044  0.151175037
 0.106175029  0.062675020  0.048875016  0.033875012  0.027600009  0.014675007  0.009450005  0.004650007  0.001225012  0.000210001
 0.000575001  0.000225000  0.000150000  0.000075000  0.0

(8)  L  IDNC    PE     PMIN PMADD MWTOT    AVPROB        DELGWH       EXPGWH     L
     2   202  1900.    100   100   200  0.20196255    14.006102    50.713927     2

WITH    PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    37     PEMAX =    3700.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000010  0.967296254  0.934592507
 0.900167510  0.862108763  0.660531283  0.458762554  0.408272554  0.339613802  0.273598800  0.226192551  0.189417545  0.154716259
 0.110132530  0.067110022  0.051740018  0.035267513  0.023913760  0.015632507  0.010107505  0.005151251  0.001536252  0.000227501
 0.001087501  0.000318750  0.000191250  0.000082500  0.000007500  0.000003750  0.0

*****************************************************************************************************

WITHOUT PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    37     PEMAX =    3700.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000010  0.967296254  0.934592507
 0.900167510  0.862108763  0.660531283  0.458762554  0.408272554  0.339613802  0.273598800  0.226192551  0.189417545  0.154716269
 0.110132530  0.067110022  0.051740018  0.035267513  0.023913760  0.015632507  0.010107505  0.005151251  0.003516252  0.002277501
 0.001087501  0.000318750  0.000191250  0.000082500  0.000007500  0.000003750  0.0

(9)  L  IDNC    PE     PMIN PMADD MWTOT    AVPROB        DELGWH       EXPGWH     L
     1   101  2000.      0   100   100  0.17207692    11.933534    11.933534     1

WITH    PLANT OF INTEREST         PROB(K),K=1,IEMAX :          DP =   100.00     IEMAX =    38     PEMAX =    3800.0010
 1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000000  1.000000010  0.968431441  0.936862765
 0.901808761  0.864011701  0.670610160  0.468851192  0.410797655  0.343346741  0.276895551  0.226852866  0.151166255  0.156470157
 0.112362719  0.069261148  0.052908518  0.036491139  0.024401447  0.016046570  0.010183755  0.005399065  0.003617002  0.002292515
 0.001144501  0.000353128  0.000197625  0.000097938  0.000011250  0.000003938  0.000000938  0.0
```

APPENDIX D

NUMERICAL RESULTS FOR CASES I THROUGH VI ON HYPOTHETICAL UTILITY SYSTEM OF CHAPTER 5

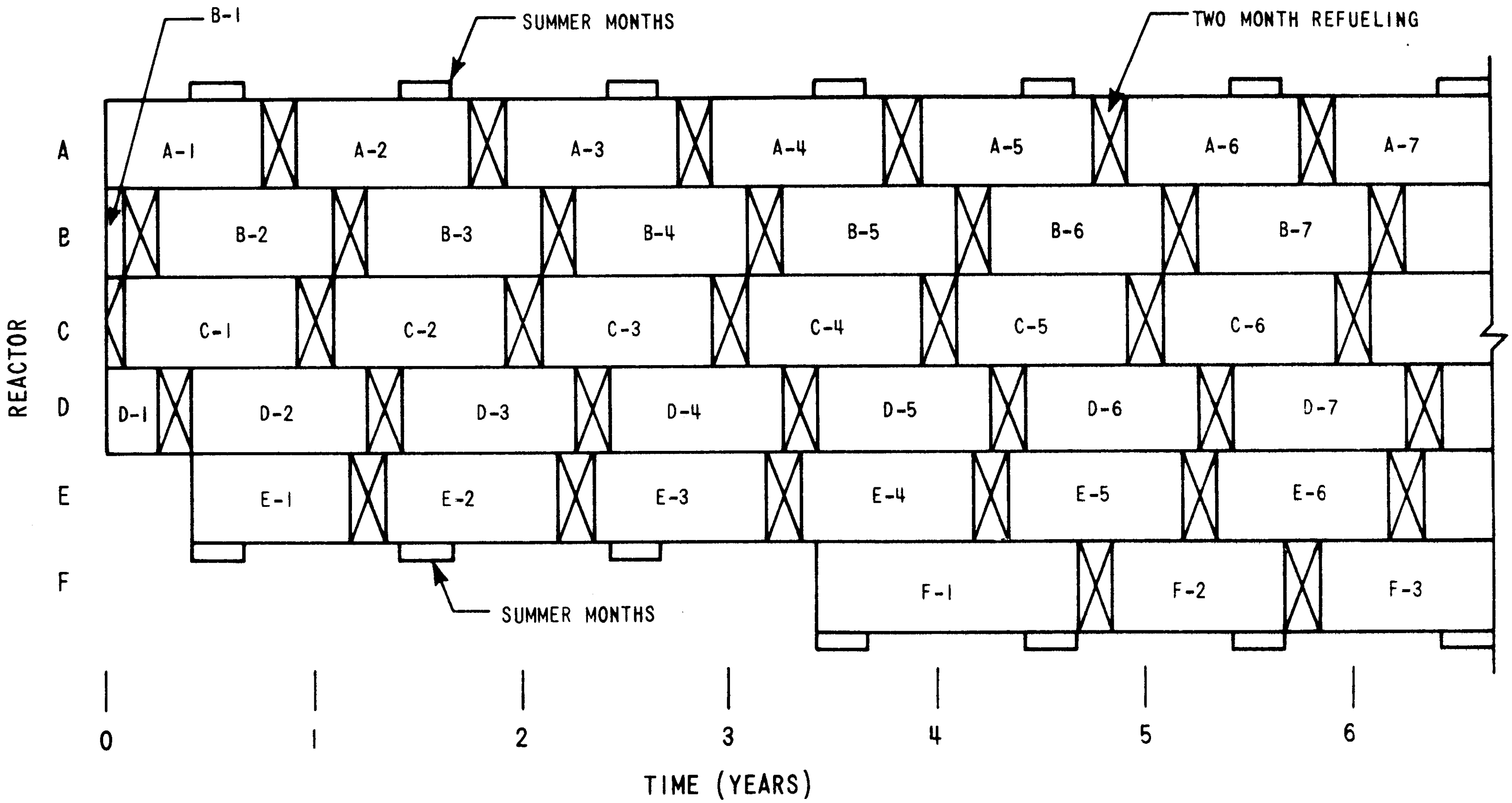
Section 5.3 presented the customer loads, generating equipment and the three maintenance and refueling strategies investigated. (Figures D.1 to D.3 present the reactor-cycle notation used in tabulating the results for each strategy). Section 5.4 indicated the values chosen for the remaining parameters of interest. Table 5.8 presented the structure of the Case I through VI studies.

Tables D.1 through D.6 present the same Case-by-Case results presented throughout Chapter 5. In addition, Table D.7 presents the Case I results at the end of the first shape iteration when $\overline{\overline{TC}} = \overline{\overline{TC}}^{*,0}$. These results differ from Case II input only with respect to the planning horizon (72 month rather than 48 month as in Case II).

Tables D.8-D.25 present strategy-by-strategy, Case-by-Case detailed results for each reactor-cycle. In addition, Tables D.26-D.28 present Case I strategy-by-strategy data at the end of the first shape iteration.

Figure D.1

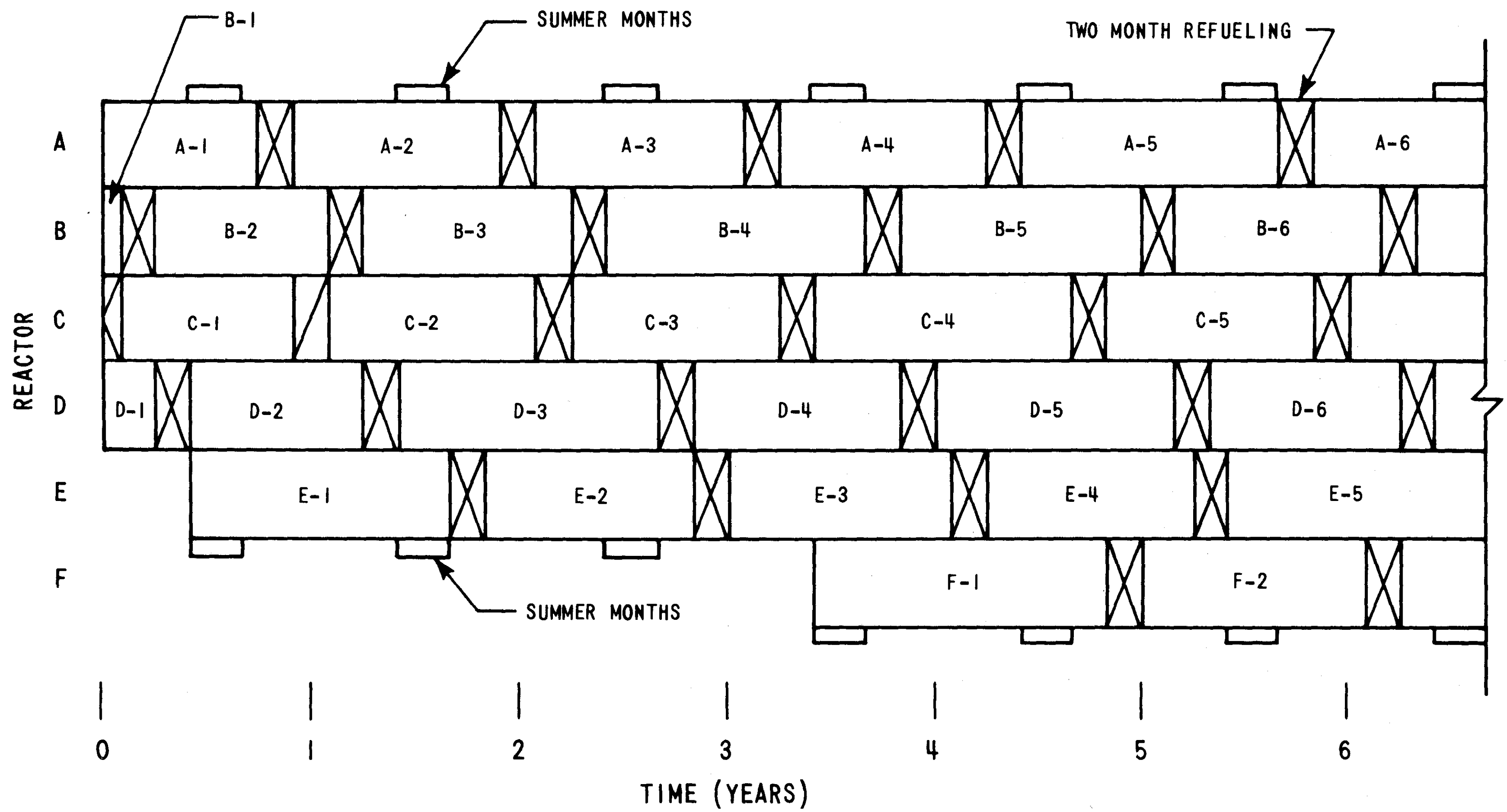
Reactor-Cycle Notation for Strategy 1 (Annual Refuelings)



6253-51

Figure D.2

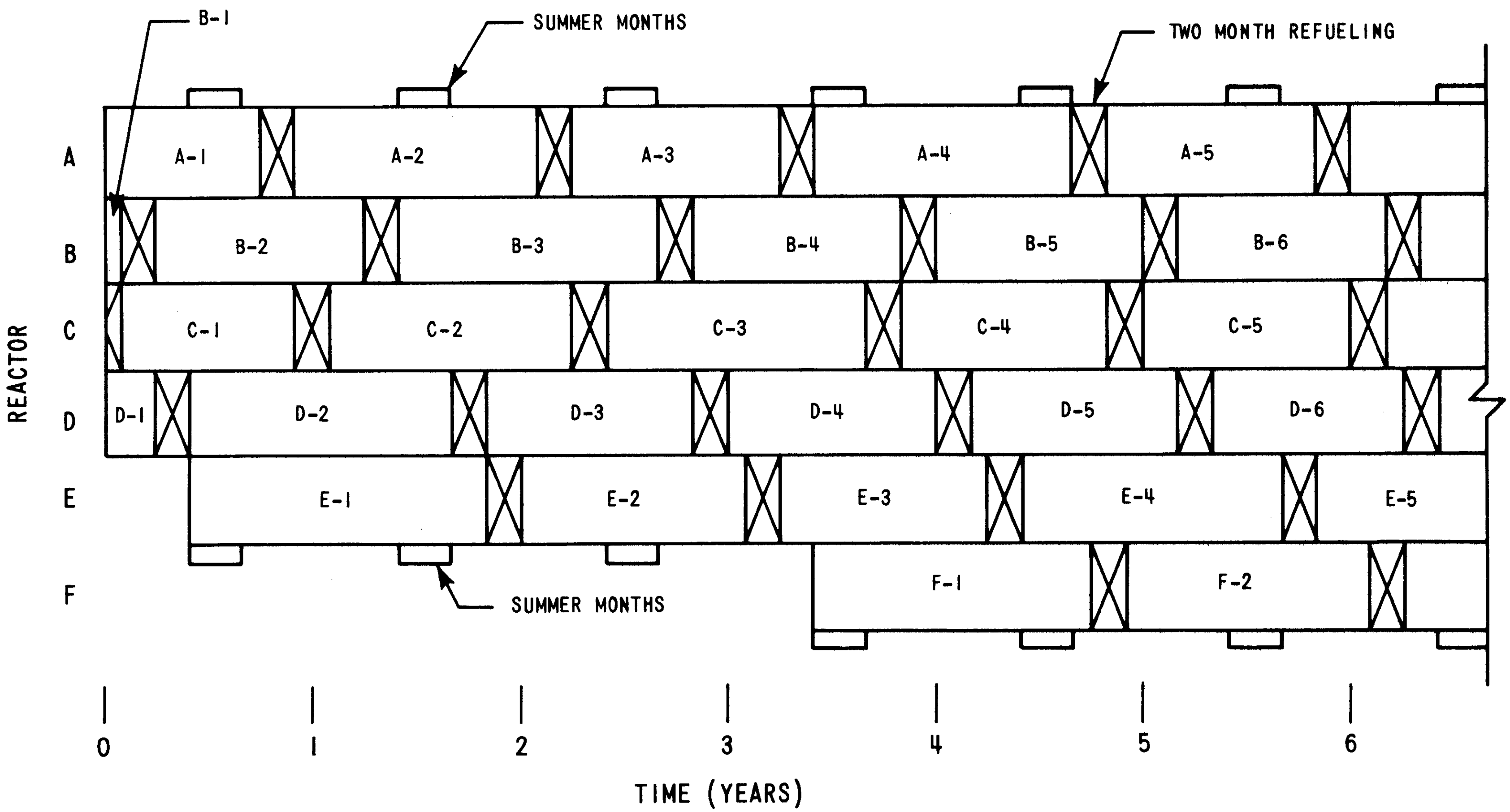
Reactor-Cycle Notation for Strategy 2 (Gradual Shift to Longer Cycles)



6253-55

Figure D.3

Reactor-Cycle Notation for Strategy 3 (Immediate Shift to Longer Cycles)



6253-57

TABLE D.1

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE I

(72 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, 0.0 Shape Rejection Criterion)
Direct Calculation Using γ = 0.25

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	62	51	49
Average cycle length (months)	12	14.9	15.2
System nuclear capacity factor	0.642	0.656	0.658
		10^6\$ (10^6 MWH)	
Fossil fuel	293.205	276.853	274.082
	(90.068)	(85.836)	(85.196)
Startup-shutdown cost	2.022	1.704	1.650
Emergency purchases	0.655	0.407	0.363
	(0.079)	(0.048)	(0.043)
Nonnuclear production	295.882	278.964	276.095
	(90.147)	(85.884)	(85.239)
Nuclear fuel	294.690	297.709	300.137
	(189.814)	(194.077)	(194.722)
System production	590.572	576.673	576.232
	(279.961)	(279.961)	(279.961)
Fixed firm purchase	133.920	133.920	133.920
	(81.468)	(81.468)	(81.468)
Penalty for short-notice enrichment changes			2.000
System Total	724.492	710.593	712.152
	(361.429)	(361.429)	(361.429)

TABLE D.2

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE II

(48 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
	10^6\$ (10^6 MWH)		
Fossil fuel	184.223 (51.703)	176.348 (50.061)	173.250 (49.390)
Startup-shutdown cost	1.497	1.281	1.227
Emergency purchases	0.464 (0.053)	0.317 (0.036)	0.265 (0.030)
Nonnuclear production	186.184 (51.756)	177.946 (50.097)	174.742 (49.420)
Nuclear fuel	198.267 (118.376)	197.189 (120.035)	199.821 (120.712)
System production	384.451 (170.132)	375.135 (170.132)	374.563 (170.132)
Fixed firm purchase	95.166 (54.312)	95.166 (54.312)	95.166 (54.312)
Penalty for short-notice enrichment changes			2.000
System Total	479.617 (224.444)	470.301 (224.444)	471.729 (224.444)

TABLE D.3

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE III

(48 Month Horizon, 0% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
		10^6\$ (10^6 MWH)	
Fossil fuel	212.434 (51.703)	203.326 (50.061)	199.928 (49.390)
Startup-shutdown cost	1.684	1.430	1.373
Emergency purchases	0.528 (0.053)	0.355 (0.036)	0.299 (0.030)
Nonnuclear production	214.646 (51.756)	205.111 (50.097)	201.600 (49.420)
Nuclear fuel	158.416 (118.376)	153.987 (120.035)	154.678 (120.712)
System production	373.062 (170.132)	359.098 (170.132)	356.278 (170.132)
Fixed firm purchase	108.624 (54.312)	108.624 (54.312)	108.624 (54.312)
Penalty for short-notice enrichment changes			2.000
System Total	481.686 (224.444)	467.722 (224.444)	466.902 (224.444)

TABLE D.4
REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE IV

(48 Month Horizon, 12% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
		10^6\$ (10^6 MWH)	
Fossil fuel	167.908 (51.703)	160.762 (50.061)	157.850 (49.390)
Startup-shutdown cost	1.388	1.194	1.142
Emergency purchases	0.427 (0.053)	0.294 (0.036)	0.245 (0.030)
Nonnuclear production	169.723 (51.756)	162.250 (50.097)	159.237 (49.420)
Nuclear fuel	220.395 (118.376)	221.107 (120.035)	224.731 (120.712)
System production	390.118 (170.132)	383.357 (170.132)	383.968 (170.132)
Fixed firm purchase	87.340 (54.312)	87.340 (54.312)	87.340 (54.312)
Penalty for short-notice enrichment changes			2.000
System Total	477.458 (224.444)	470.697 (224.444)	473.308 (224.444)

TABLE D.5

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE V

(48 Month Horizon, 7% P.V. Rate, Low Nuclear Unit Costs, No Shape Constraints)

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
		10^6\$ (10^6 MWH)	
Fossil fuel	184.223	176.348	173.250
	(51.703)	(50.061)	(49.390)
Startup-shutdown cost	1.497	1.281	1.227
Emergency purchases	0.464	0.317	0.265
	(0.053)	(0.036)	(0.030)
Nonnuclear production	186.184	177.946	174.742
	(51.756)	(50.097)	(49.420)
Nuclear fuel	141.229	141.156	143.463
	(118.376)	(120.035)	(120.712)
System production	327.413	319.102	318.205
	(170.132)	(170.132)	(170.132)
Fixed firm purchase	95.166	95.166	95.166
	(54.312)	(54.312)	(54.312)
Penalty for short-notice enrichment changes			2.000
System Total	422.579	414.268	415.371
	(224.444)	(224.444)	(224.444)

TABLE D.6

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE VI

(48 Month Horizon, 7% P.V. Rate, High Nuclear Unit Costs, No Shape Constraints)

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	38	33	31
Average cycle length (months)	12	14.5	15.2
System nuclear capacity factor	0.638	0.647	0.651
		10^6\$ (10^6 MWH)	
Fossil fuel	184.223 (51.703)	176.348 (50.061)	173.250 (49.390)
Startup-shutdown cost	1.497	1.281	1.227
Emergency purchases	0.464 (0.053)	0.317 (0.036)	0.265 (0.030)
Nonnuclear production	186.184 (51.756)	177.946 (50.097)	174.742 (49.420)
Nuclear fuel	255.223 (118.376)	253.211 (120.035)	256.169 (120.712)
System production	441.407 (170.132)	431.157 (170.132)	430.911 (170.132)
Fixed firm purchase	95.166 (54.312)	95.166 (54.312)	95.166 (54.312)
Penalty for short-notice enrichment changes			2.000
System Total	536.573 (224.444)	526.323 (224.444)	528.077 (224.444)

TABLE D.7

REVENUE REQUIREMENTS AND UNDISCOUNTED ENERGY FOR CASE I AT END OF FIRST SHAPE ITERATION

(72 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Strategy	S-1	S-2	S-3
Downtime to horizon (reactor-months)	62	51	49
Average cycle length (months)	12	14.9	15.2
System nuclear capacity factor	0.642	0.656	0.658
		10^6\$ (10^6 MWH)	
Fossil fuel	293.205 (90.068)	276.853 (85.836)	274.082 (85.196)
Startup-shutdown cost	2.022	1.704	1.650
Emergency purchases	0.655 (0.079)	0.407 (0.048)	0.363 (0.043)
Nonnuclear production	295.882 (90.147)	278.964 (85.884)	276.095 (85.239)
Nuclear fuel	294.583 (189.814)	297.456 (194.077)	299.761 (194.722)
System production	590.465 (279.961)	576.420 (279.961)	575.856 (279.961)
Fixed firm purchase	133.920 (81.468)	133.920 (81.468)	133.920 (81.468)
Penalty for short-notice enrichment changes			2.000
System Total	724.385 (361.429)	710.340 (361.429)	711.776 (361.429)

TABLE D. 8

REACTOR-CYCLE RESULTS FOR STRATEGY 1 IN CASE I

(72 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, V^2_{REJ} = 0)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (|$|/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5280	1.703	.683	–
A-2	10	5662	1.896	.992	2.876
A-3	10	5688	1.935	1.240	3.164
A-4	10	5799	1.931	1.063	3.178
A-5	10	5760	1.905	.921	3.036
A-6	10	5746	1.924	1.096	3.153
A-7	10	5950	1.909	1.182	3.226
B-1	1*	638	1.845	.499	–
B-2	10	6418	1.832	.683	3.4†
B-3	10	6440	1.771	.992	2.907
B-4	10	6240	1.854	1.240	3.447
B-5	10	6230	1.825	.963	3.123
B-6	10	6180	1.815	.921	3.113
B-7	10	6500	1.834	1.096	3.471
C-1	10†	6180	1.875	.683	3.6†
C-2	10	6140	1.845	.992	2.786
C-3	10	5760	1.944	1.240	3.296
C-4	10	5740	1.936	1.031	3.096
C-5	10	5720	1.904	.921	2.983
C-6	10	5656	1.942	1.096	3.177
D-1	3*	2129	1.465	.448	–
D-2	10	5920	1.822	.683	3.2†
D-3	10	6060	1.817	.992	3.040
D-4	10	6400	1.830	1.240	3.377
D-5	10	6149	1.827	.963	3.118
D-6	10	5983	1.830	.921	3.013
D-7	10	6120	1.852	1.096	3.255
E-1	9	3297	3.437	.683	3.2†
E-2	10	5337	2.086	.992	1.5**
E-3	10	5089	2.183	1.689	4.012
E-4	10	5080	2.091	1.031	3.168
E-5	10	6869	1.718	.846	2.661
E-6	10	5326	1.967	1.122	3.331
F-1	15	7874	2.175	.818	3.2†
F-2	10	6372	1.847	1.093	3.410
F-3	10	5882	1.840	1.130	3.086

* Fractional cycle
† Fixed initial condition
** 1.5 w/o U-235 was lower limit permitted by QKCORE.

TABLE D. 9
REACTOR-CYCLE RESULTS FOR STRATEGY 2 IN CASE I

(72 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, $V^2_{REJ} = 0$)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (|$|/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5270	1.690	.959	–
A-2	12	6720	1.913	1.309	3.592
A-3	12	7280	1.900	1.408	3.927
A-4	12	7580	1.883	1.408	3.936
A-5	15	7775	1.979	1.408	3.966
A-6	12	7165	1.884	1.173	3.497
B-1	1*	667	1.802	.657	–
B-2	10	6420	1.819	.959	3.4†
B-3	12	7566	1.798	1.309	3.650
B-4	15	7500	1.942	1.408	3.965
B-5	14	8060	1.872	1.408	3.984
B-6	12	7732	1.808	1.173	3.689
C-1	10†	6300	1.844	.959	3.6†
C-2	12	7260	1.873	1.309	3.620
C-3	12	7218	1.902	1.408	3.863
C-4	15	7500	1.988	1.408	3.875
C-5	12	7480	1.878	1.248	3.760
D-1	3*	2100	1.481	.657	–
D-2	10	5340	1.905	.959	3.2†
D-3	15	7820	1.894	1.408	3.841
D-4	12	7460	1.844	1.408	3.928
D-5	14	8060	1.872	1.408	3.975
D-6	11	7089	1.786	1.070	3.243
E-1	15	7200	2.284	.959	3.2†
E-2	12	7623	1.843	1.401	3.838
E-3	13	7133	1.907	1.408	3.990
E-4	12	8174	1.781	1.244	3.822
E-5	15	7855	1.916	1.248	3.880
F-1	17	9060	2.053	1.408	3.2†
F-2	13	6949	2.045	2.033	4.632

* Fractional cycle

† Fixed initial condition

TABLE D. 10

REACTOR-CYCLE RESULTS FOR STRATEGY 3 IN CASE I

(72 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, $V^2_{REJ} = 0$)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5460	1.660	1.397	–
A-2	14	7206	1.994	1.905	4.101
A-3	12	7652	1.887	1.577	3.998
A-4	15	7406	1.974	1.228	3.626
A-5	12	6960	1.861	1.158	3.451
B-1	1*	710	1.735	.795	–
B-2	12	7265	1.820	1.397	3.718**
B-3	15	8026	1.912	1.905	4.140
B-4	12	7280	1.822	1.092	3.479
B-5	12	7400	1.817	1.092	3.546
B-6	12	7220	1.842	1.158	3.620
C-1	10†	6473	1.817	1.312	3.6†
C-2	14	7740	1.953	1.905	4.119
C-3	15	7960	1.998	1.905	4.243
C-4	12	7320	1.876	1.092	3.504
C-5	12	7042	1.892	1.158	3.465
D-1	3*	2057	1.486	1.045	–
D-2	15	8445	2.023	1.600	5.0**
D-3	12	7880	1.881	1.905	4.242
D-4	12	8076	1.765	1.055	3.465
D-5	12	7480	1.817	1.092	3.550
D-6	11	7225	1.813	1.144	3.651
E-1	17	8295	2.159	1.418	3.2†
E-2	13	7120	1.959	1.905	4.230
E-3	12	6924	1.848	1.092	3.464
E-4	15	7584	1.895	1.185	3.533
F-1	16	8599	2.097	.436	3.2†
F-2	14	8174	2.035	1.158	5.0

* Fractional cycle

† Fixed initial condition

** Short notice enrichment change (5.0 w/o U-235 was upper limit permitted by QKCORE).

TABLE D. 11
REACTOR-CYCLE RESULTS FOR STRATEGY 1 IN CASE II

(48 Month Horizon, 7% P.V. Rate,
Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	4960	1.752	0.632	–
A-2	10	5740	1.877	0.915	2.736
A-3	10	5880	1.948	1.227	3.465
A-4	10	5840	1.917	1.245	3.098
A-5	10	5919	1.887	1.339	3.083
B-1	1*	638	1.842	0.446	–
B-2	10	6500	1.819	0.630	3.4†
B-3	10	6520	1.769	0.915	3.005
B-4	10	6420	1.836	1.227	3.492
B-5	10	6725	1.804	1.243	3.402
C-1	10†	5640	1.960	0.632	3.6†
C-2	10	6654	1.803	0.889	2.765
C-3	10	5740	1.957	1.229	3.431
C-4	10	5800	1.922	1.245	3.061
D-1	3*	2129	1.457	0.442	–
D-2	10	6349	1.765	0.630	3.2†
D-3	10	5520	1.857	0.917	2.941
D-4	10	6614	1.825	1.227	3.494
D-5	10	6402	1.819	1.243	3.270
E-1	9	3697	3.194	0.632	3.2†
E-2	10	5186	2.073	0.917	1.5**
E-3	10	5073	2.190	1.627	4.023
E-4	10	5088	2.099	1.313	3.152
F-1	15	7139	2.311	1.395	3.2†

* Fractional cycle
† Fixed initial condition
** 1.5 w/o U-235 was lower limit permitted by QKCORE.

TABLE D. 12

REACTOR-CYCLE RESULTS FOR STRATEGY 2 IN CASE II

(48 Month Horizon, 7% P.V. Rate,
Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost ($/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5400	1.671	0.924	–
A-2	12	6760	1.920	1.339	3.710
A-3	12	7240	1.892	1.474	3.812
A-4	12	7580	1.884	1.476	3.941
B-1	1*	710	1.747	0.614	–
B-2	10	6580	1.795	0.924	3.4†
B-3	12	7422	1.810	1.339	3.674
B-4	15	7480	1.936	1.478	3.888
B-5	14	8524	1.889	1.476	4.374
C-1	10†	6220	1.855	0.924	3.6†
C-2	12	7420	1.865	1.339	3.678
C-3	12	7280	1.902	1.474	3.912
C-4	15	8129	1.999	1.674	4.346
D-1	3*	2057	1.494	0.636	–
D-2	10	5308	1.914	1.043	3.2†
D-3	15	7820	1.892	1.474	3.802
D-4	12	7440	1.816	1.476	3.944
E-1	15	7080	2.308	0.926	3.2†
E-2	12	7609	1.838	1.374	3.743
E-3	13	6940	1.919	1.478	3.912
F-1	17	8834	2.084	1.258	3.2†

* Fractional cycle

† Fixed initial condition

TABLE D. 13
REACTOR-CYCLE RESULTS FOR STRATEGY 3 IN CASE II

(48 Month Horizon, 7% P.V. Rate,
Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5560	1.647	1.309	–
A-2	14	7120	2.001	1.837	4.105
A-3	12	7704	1.882	1.601	4.009
A-4	15	8129	1.976	1.574	4.182
B-1	1*	710	1.735	0.772	–
B-2	12	7260	1.820	1.311	3.715**
B-3	15	7980	1.912	1.837	4.105
B-4	12	7392	1.786	1.302	3.561
C-1	10†	6609	1.799	1.240	3.6†
C-2	14	7560	1.964	1.837	4.081
C-3	15	7940	1.992	1.833	4.189
C-4	12	7004	1.883	1.336	3.311
D-1	3*	2057	1.487	0.970	–
D-2	15	8373	2.032	1.492	5.0**
D-3	12	7865	1.874	1.833	4.175
D-4	12	8228	1.764	1.290	3.584
E-1	17	8405	2.142	1.313	3.2†
E-2	13	6960	1.966	1.835	4.177
E-3	12	7020	1.843	1.302	3.508
F-1	16	7634	2.238	1.242	3.2†

* Fractional cycle
† Fixed initial condition
** Short-notice enrichment change (5.0 w/o U-235 was upper limit permitted by OKCORE).

TABLE D. 14
REACTOR-CYCLE RESULTS FOR STRATEGY 1 IN CASE III

(48 Month Horizon, 0% P.V. Rate,
Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5060	1.301	0.665	–
A-2	10	5880	1.290	0.799	2.886
A-3	10	6040	1.315	1.025	3.452
A-4	10	6047	1.308	1.170	3.200
A-5	10	5968	1.296	1.218	3.087
B-1	1*	660	1.264	0.552	–
B-2	10	6800	1.233	0.653	3.4†
B-3	10	6397	1.231	0.799	3.114
B-4	10	6480	1.250	1.025	3.402
B-5	10	6660	1.252	1.170	3.375
C-1	10†	5680	1.341	0.655	3.6†
C-2	10	6654	1.284	0.779	2.792
C-3	10	5720	1.318	1.027	3.389
C-4	10	5700	1.320	1.172	3.006
D-1	3*	2107	1.129	0.552	–
D-2	10	6338	1.220	0.653	3.2†
D-3	10	5681	1.249	0.825	3.030
D-4	10	6414	1.265	1.025	3.312
D-5	10	6360	1.261	1.170	3.292
E-1	9	3268	2.445	0.761	3.2†
E-2	10	5008	1.583	0.877	1.5**
E-3	10	5073	1.453	1.294	3.854
E-4	10	5039	1.403	1.237	3.220
F-1	15	7139	1.672	1.444	3.2†

* Fractional cycle
† Fixed initial condition
** 1.5 w/o U-235 was lower limit permitted by QKCORE.

TABLE D. 15
REACTOR-CYCLE RESULTS FOR STRATEGY 2 IN CASE III

(48 Month Horizon, 0% P.V. Rate,
Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5342	1.272	0.756	–
A-2	12	6702	1.271	0.981	3.628
A-3	12	7300	1.269	1.227	3.907
A-4	12	7600	1.260	1.367	3.949
B-1	1*	710	1.239	0.566	–
B-2	10	6816	1.221	0.698	3.4†
B-3	12	7300	1.220	0.981	3.746
B-4	15	7780	1.224	1.369	4.022
B-5	14	8582	1.233	1.350	4.367
C-1	10†	6080	1.308	0.756	3.6†
C-2	12	7600	1.264	0.979	3.706
C-3	12	7141	1.261	1.229	3.838
C-4	15	8129	1.279	1.516	4.354
D-1	3*	2057	1.152	0.665	–
D-2	10	5308	1.246	0.927	3.2†
D-3	15	7980	1.239	1.227	3.917
D-4	12	7487	1.230	1.367	3.948
E-1	15	7042	1.664	0.798	3.2†
E-2	12	7609	1.280	1.094	3.716
E-3	13	6980	1.235	1.369	3.963
F-1	17	8288	1.566	1.367	3.2†

* Fractional cycle
† Fixed initial condition

TABLE D. 16
REACTOR-CYCLE RESULTS FOR STRATEGY 3 IN CASE III

(48 Month Horizon, 0% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5400	1.266	1.088	–
A-2	14	7325	1.282	1.438	4.154
A-3	12	7704	1.263	1.423	4.047
A-4	15	8129	1.256	1.375	4.143
B-1	1*	710	1.230	0.748	–
B-2	12	7580	1.221	1.088	3.947**
B-3	15	8080	1.218	1.438	4.108
B-4	12	7740	1.204	1.244	3.725
C-1	10†	6440	1.289	1.088	3.6†
C-2	14	7780	1.275	1.438	4.125
C-3	15	7640	1.259	1.438	3.992
C-4	12	7382	1.242	1.197	3.589
D-1	3*	2057	1.145	0.904	–
D-2	15	8373	1.333	1.237	5.0**
D-3	12	7780	1.240	1.436	4.109
D-4	12	8228	1.192	1.189	3.596
E-1	17	8414	1.556	1.090	3.2†
E-2	13	6820	1.285	1.438	4.073
E-3	12	7377	1.223	1.242	3.768
F-1	16	6551	1.717	1.453	3.2†

* Fractional cycle

† Fixed initial condition

** Short-notice enrichment change (5.0 w/o U-235 was upper limit permitted by QKCORE).

TABLE D. 17
REACTOR-CYCLE RESULTS FOR STRATEGY 1 IN CASE IV

(48 Month Horizon, 12% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	4840	2.099	0.569	–
A-2	10	5780	2.284	0.954	2.690
A-3	10	5620	2.421	1.314	3.343
A-4	10	6002	2.348	1.295	3.274
A-5	10	5919	2.293	1.336	3.013
B-1	1*	638	2.250	0.421	–
B-2	10	6320	2.271	0.567	3.4†
B-3	10	6366	2.151	0.954	2.803
B-4	10	6340	2.286	1.312	3.605
B-5	10	6749	2.208	1.263	3.433
C-1	10†	5420	2.463	0.569	3.6†
C-2	10	6654	2.160	0.882	2.622
C-3	10	5546	2.458	1.314	3.451
C-4	10	6080	2.348	1.293	3.282
D-1	3*	2129	1.694	0.362	–
D-2	10	6368	2.153	0.567	3.2†
D-3	10	5440	2.303	0.958	2.901
D-4	10	6540	2.230	1.297	3.465
D-5	10	6544	2.220	1.262	3.387
E-1	9	4198	3.499	0.567	3.2†
E-2	10	5380	2.353	0.956	1.5**
E-3	10	5073	2.750	1.923	4.207
E-4	10	5088	2.596	1.312	3.065
F-1	15	7139	2.770	1.336	3.2†

* Fractional cycle
† Fixed initial condition
** 1.5 w/o U-235 was lower limit permitted by QKCORE.

TABLE D. 18

REACTOR-CYCLE RESULTS FOR STRATEGY 2 IN CASE IV

(48 Month Horizon, 12% P.V. Rate,
Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5380	1.965	1.005	–
A-2	12	6722	2.385	1.504	3.668
A-3	12	7160	2.347	1.553	3.773
A-4	12	7691	2.336	1.551	4.056
B-1	1*	638	2.237	0.592	–
B-2	10	6660	2.189	1.005	3.4†
B-3	12	7380	2.239	1.504	3.672
B-4	15	7460	2.448	1.555	3.866
B-5	14	8582	2.360	1.504	4.429
C-1	10†	6040	2.292	1.005	3.6†
C-2	12	7500	2.278	1.502	3.604
C-3	12	7160	2.375	1.553	3.906
C-4	15	8129	2.519	1.716	4.365
D-1	3*	2129	1.711	0.564	–
D-2	10	5308	2.388	1.023	3.2†
D-3	15	7640	2.370	1.555	3.702
D-4	12	7520	2.236	1.553	4.021
E-1	15	7200	2.740	1.007	3.2†
E-2	12	7560	2.251	1.551	3.792
E-3	13	7140	2.400	1.555	4.008
F-1	17	8834	2.480	1.173	3.2†

* Fractional cycle
† Fixed initial condition

TABLE D. 19
REACTOR-CYCLE RESULTS FOR STRATEGY 3 IN CASE IV

(48 Month Horizon, 12% P.V. Rate,
Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5574	1.924	1.409	–
A-2	14	7080	2.520	2.066	4.083
A-3	12	7704	2.326	1.710	4.010
A-4	15	8129	2.492	1.620	4.191
B-1	1*	710	2.097	0.805	–
B-2	12	7100	2.256	1.451	3.603**
B-3	15	7956	2.410	2.066	4.132
B-4	12	7320	2.209	1.332	3.541
C-1	10†	6609	2.170	1.350	3.6†
C-2	14	7560	2.459	2.066	4.081
C-3	15	8029	2.514	1.995	4.264
C-4	12	7004	2.346	1.370	3.304
D-1	3*	2057	1.733	0.955	–
D-2	15	8373	2.533	1.558	5.0**
D-3	12	7880	2.327	1.995	4.187
D-4	12	8228	2.175	1.317	3.582
E-1	17	8551	2.531	1.451	3.2†
E-2	13	6920	2.469	2.064	4.245
E-3	12	7092	2.286	1.332	3.518
F-1	16	7634	2.686	1.166	3.2†

* Fractional cycle
† Fixed initial condition
** Short-notice enrichment change (5.0 w/o U-235 was upper limit permitted by QKCORE).

TABLE D. 20
REACTOR-CYCLE RESULTS FOR STRATEGY 1 IN CASE V

(48 Month Horizon, 7% P.V. Rate,
Low Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	4740	1.302	0.426	–
A-2	10	5800	1.379	0.627	2.642
A-3	10	5706	1.410	0.875	3.462
A-4	10	6020	1.347	0.856	3.246
A-5	10	5919	1.325	0.930	2.986
B-1	1*	680	1.007	0.320	–
B-2	10	6320	1.325	0.424	3.4†
B-3	10	6477	1.269	0.627	2.884
B-4	10	6260	1.305	0.873	3.490
B-5	10	6749	1.274	0.846	3.453
C-1	10†	5440	1.416	0.426	3.6†
C-2	10	6654	1.289	0.590	2.635
C-3	10	5540	1.399	0.875	3.432
C-4	10	6080	1.348	0.854	3.287
D-1	3*	2087	0.987	0.320	–
D-2	10	6246	1.286	0.424	3.2†
D-3	10	5509	1.337	0.665	2.865
D-4	10	6560	1.285	0.858	3.532
D-5	10	6506	1.282	0.852	3.343
E-1	9	4400	2.081	0.424	3.2†
E-2	10	5180	1.454	0.665	1.5**
E-3	10	5073	1.525	1.226	4.168
E-4	10	5088	1.472	0.877	3.072
F-1	15	7139	1.662	0.989	3.2†

* Fractional cycle
† Fixed initial condition
** 1.5 w/o U-235 was lower limit permitted by QKCORE.

TABLE D. 21

REACTOR-CYCLE RESULTS FOR STRATEGY 2 IN CASE V

(48 Month Horizon, 7% P.V. Rate, Low Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5400	1.211	0.676	–
A-2	12	6722	1.408	0.962	3.682
A-3	12	7180	1.368	1.025	3.778
A-4	12	7604	1.342	1.025	3.981
B-1	1*	667	1.006	0.450	–
B-2	10	6580	1.290	0.676	3.4†
B-3	12	7380	1.300	0.962	3.629
B-4	15	7520	1.390	1.027	3.943
B-5	14	8440	1.347	1.025	4.301
C-1	10†	6120	1.321	0.676	3.6†
C-2	12	7500	1.333	0.960	3.662
C-3	12	7220	1.357	1.025	3.903
C-4	15	8129	1.432	1.173	4.349
D-1	3*	2100	0.995	0.450	–
D-2	10	5308	1.385	0.711	3.2†
D-3	15	7760	1.363	1.025	3.775
D-4	12	7500	1.286	1.025	3.988
E-1	15	7180	1.644	0.678	3.2†
E-2	12	7609	1.321	1.016	3.814
E-3	13	7080	1.365	1.027	3.963
F-1	17	8834	1.496	0.898	3.2†

* Fractional cycle
† Fixed Initial condition

TABLE D. 22

REACTOR-CYCLE RESULTS FOR STRATEGY 3 IN CASE V

(48 Month Horizon, 7% P.V. Rate,
Low Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5560	1.192	0.930	–
A-2	14	7100	1.468	1.306	4.089
A-3	12	7704	1.360	1.117	4.013
A-4	15	8129	1.422	1.108	4.187
B-1	1*	710	0.975	0.548	–
B-2	12	7200	1.310	0.932	3.673**
B-3	15	7960	1.384	1.306	4.106
B-4	12	7432	1.275	0.889	3.601
C-1	10†	6609	1.269	0.883	3.6†
C-2	14	7560	1.415	1.306	4.081
C-3	15	8020	1.431	1.302	4.256
C-4	12	7004	1.357	0.953	3.304
D-1	3*	2057	0.998	0.684	–
D-2	15	8373	1.457	1.051	5.0**
D-3	12	7865	1.359	1.302	4.175
D-4	12	8228	1.268	0.876	3.584
E-1	17	8465	1.532	0.932	3.2†
E-2	13	6920	1.418	1.304	4.186
E-3	12	6980	1.316	0.889	3.466
F-1	16	7634	1.609	0.875	3.2†

* Fractional cycle
† Fixed initial condition
** Short-notice enrichment change (5.0 w/o U-235 was upper limit permitted by QKCORE).

TABLE D. 23
REACTOR-CYCLE RESULTS FOR STRATEGY 1 IN CASE VI
(48 Month Horizon, 7% P.V. Rate,
High Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5080	2.206	0.837	–
A-2	10	5766	2.374	1.199	2.824
A-3	10	5840	2.484	1.552	3.348
A-4	10	5840	2.484	1.653	3.126
A-5	10	5968	2.446	1.741	3.145
B-1	1*	638	2.650	0.636	–
B-2	10	6769	2.288	0.835	3.4†
B-3	10	6260	2.289	1.199	3.002
B-4	10	6554	2.369	1.552	3.526
B-5	10	6720	2.336	1.651	3.394
C-1	10†	5680	2.522	0.837	3.6†
C-2	10	6654	2.315	1.155	2.792
C-3	10	5720	2.529	1.554	3.389
C-4	10	5707	2.495	1.653	3.011
D-1	3*	2129	1.936	0.588	–
D-2	10	6349	2.255	0.835	3.2†
D-3	10	5560	2.373	1.201	2.966
D-4	10	6540	2.369	1.552	3.428
D-5	10	6500	2.355	1.651	3.354
E-1	9	3268	4.346	0.884	3.2†
E-2	10	5380	2.667	1.201	1.5**
E-3	10	5073	2.865	2.106	4.012
E-4	10	5039	2.734	1.742	3.138
F-1	15	7139	2.959	1.802	3.2†

* Fractional cycle
† Fixed initial condition
** 1.5 w/o U-235 was lower limit permitted by QKCORE

TABLE D. 24

REACTOR-CYCLE RESULTS FOR STRATEGY 2 IN CASE VI

(48 Month Horizon, 7% P.V. Rate,
High Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (|$|/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5360	2.140	1.157	–
A-2	12	6700	2.430	1.698	3.639
A-3	12	7320	2.426	1.924	3.913
A-4	12	7560	2.426	1.928	3.914
B-1	1*	710	2.508	0.747	–
B-2	10	6660	2.289	1.157	3.4†
B-3	12	7420	2.327	1.698	3.726
B-4	15	7480	2.478	1.930	3.848
B-5	14	8566	2.432	1.927	4.407
C-1	10†	6218	2.400	1.157	3.6†
C-2	12	7482	2.392	1.696	3.722
C-3	12	7257	2.448	1.926	3.878
C-4	15	8129	2.564	2.175	4.342
D-1	3*	2057	1.985	0.820	–
D-2	10	5308	2.440	1.343	3.2†
D-3	15	7841	2.423	1.924	3.817
D-4	12	7400	2.345	1.928	3.910
E-1	15	7042	2.966	1.208	3.2†
E-2	12	7609	2.357	1.765	3.716
E-3	13	6880	2.471	1.930	3.888
F-1	17	8834	2.673	1.618	3.2†

* Fractional cycle
† Fixed initial condition

TABLE D. 25

REACTOR-CYCLE RESULTS FOR STRATEGY 3 IN CASE VI

(48 Month Horizon, 7% P.V. Rate,
High Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5560	2.102	1.683	–
A-2	14	7120	2.534	2.364	4.105
A-3	12	7704	2.404	2.089	4.009
A-4	15	8129	2.530	2.038	4.182
B-1	1*	710	2.495	0.993	–
B-2	12	7320	2.330	1.685	3.757**
B-3	15	8000	2.438	2.364	4.105
B-4	12	7380	2.296	1.718	3.539
C-1	10†	6609	2.329	1.597	3.6†
C-2	14	7560	2.513	2.364	4.081
C-3	15	7900	2.552	2.362	4.156
C-4	12	7004	2.410	1.720	3.316
D-1	3*	2057	1.976	1.231	–
D-2	15	8373	2.606	1.897	5.0**
D-3	12	7860	2.389	2.362	4.171
D-4	12	8228	2.260	1.706	3.585
E-1	17	8345	2.757	1.687	3.2†
E-2	13	6985	2.510	2.364	4.156
E-3	12	7032	2.372	1.718	3.532
F-1	16	7634	2.868	1.609	3.2†

* Fractional cycle
† Fixed initial condition
** Short-notice enrichment change (5.0 w/o U-235 was upper limit permitted by QKCORE)

TABLE D. 26

REACTOR-CYCLE RESULTS FOR STRATEGY 1 IN CASE I
AT END OF FIRST SHAPE ITERATION

(72 Month Horizon, 7% P.V. Rate,
Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5280	1.703	0.682	–
A-2	10	5660	1.896	0.997	2.875
A-3	10	5680	1.937	1.244	3.160
A-4	10	5690	1.937	1.032	3.107
A-5	10	5700	1.914	0.918	3.033
A-6	10	5740	1.928	1.102	3.168
A-7	10	5933	1.908	1.179	3.204
B-1	1*	638	1.845	0.495	–
B-2	10	6420	1.832	0.682	3.4†
B-3	10	6420	1.772	0.995	2.895
B-4	10	6200	1.856	1.242	3.427
B-5	10	6316	1.823	0.990	3.190
B-6	10	6220	1.809	0.916	3.109
B-7	10	6520	1.830	1.100	3.470
C-1	10†	6200	1.872	0.682	3.6†
C-2	10	6140	1.847	0.995	2.798
C-3	10	5740	1.944	1.244	3.269
C-4	10	5620	1.943	0.992	3.025
C-5	10	5676	1.913	0.918	3.001
C-6	10	5700	1.941	1.102	3.212
D-1	3*	2129	1.465	0.454	–
D-2	10	5900	1.825	0.684	3.2†
D-3	10	6080	1.815	0.997	3.042
D-4	10	6419	1.829	1.240	3.394
D-5	10	6260	1.822	0.990	3.182
D-6	10	6020	1.824	0.918	3.000
D-7	10	6141	1.850	1.100	3.260
E-1	9	3346	3.405	0.686	3.2†
E-2	10	5320	2.084	0.997	1.5**
E-3	10	5073	2.185	1.678	4.004
E-4	10	4980	2.099	0.992	3.102
E-5	10	6962	1.721	0.872	2.753
E-6	10	5316	1.960	1.113	3.280
F-1	15	8057	2.146	0.813	3.2†
F-2	10	6260	1.859	1.100	3.445
F-3	10	5857	1.836	1.125	3.014

* Fractional cycle
† Fixed initial condition
** 1.5 w/o U-235 was lower limit permitted by QKCORE

TABLE D. 27

REACTOR-CYCLE RESULTS FOR STRATEGY 2 IN CASE I AT END OF FIRST SHAPE ITERATION

(72 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5191	1.702	0.967	–
A-2	12	6780	1.907	1.347	3.583
A-3	12	7220	1.904	1.380	3.907
A-4	12	7540	1.883	1.380	3.910
A-5	15	7660	1.979	1.380	3.890
A-6	12	7218	1.850	1.159	3.567
B-1	1*	667	1.803	0.666	–
B-2	10	6400	1.822	0.967	3.4†
B-3	12	7602	1.797	1.347	3.661
B-4	15	7500	1.942	1.382	3.966
B-5	14	8021	1.871	1.380	3.950
B-6	12	7860	1.808	1.205	3.786
C-1	10†	6300	1.845	0.967	3.6†
C-2	12	7220	1.874	1.347	3.591
C-3	12	7140	1.903	1.380	3.816
C-4	15	7440	1.990	1.382	3.854
C-5	12	7513	1.880	1.264	3.807
D-1	3*	2100	1.478	0.666	–
D-2	10	5480	1.882	0.969	3.2†
D-3	15	7640	1.905	1.382	3.811
D-4	12	7460	1.840	1.380	3.896
D-5	14	7980	1.873	1.380	3.943
D-6	11	7238	1.784	1.079	3.361
E-1	15	7217	2.281	0.969	3.2†
E-2	12	7609	1.844	1.375	3.840
E-3	13	7100	1.906	1.382	3.959
E-4	12	8240	1.781	1.266	3.878
E-5	15	7940	1.916	1.262	3.936
F-1	17	9340	2.016	1.378	3.2†
F-2	13	6740	2.067	2.001	4.636

* Fractional cycle
† Fixed initial condition

TABLE D. 28
REACTOR-RECYCLE RESULTS FOR STRATEGY 3 IN CASE I AT END OF FIRST SHAPE ITERATION

(72 Month Horizon, 7% P.V. Rate, Reference Nuclear Unit Costs, No Shape Constraints)

Reactor-Cycle	Cycle Length (Months on-line)	Cycle Energy (GWH)	Average Cycle Energy Cost (I$I/MWH)	Incremental Cycle Energy Cost ($/MWH)	Reload Enrichment (w/o U-235)
A-1	9*	5603	1.642	1.322	–
A-2	14	7060	2.006	1.853	4.088
A-3	12	7704	1.883	1.630	4.001
A-4	15	7275	1.978	1.196	3.543
A-5	12	6960	1.865	1.160	3.490
B-1	1*	710	1.735	0.814	–
B-2	12	7260	1.820	1.397	3.715**
B-3	15	7980	1.911	1.853	4.105
B-4	12	7305	1.820	1.101	3.502
B-5	12	7440	1.816	1.101	3.574
B-6	12	7267	1.839	1.160	3.635
C-1	10†	6580	1.803	1.322	3.6†
C-2	14	7580	1.961	1.853	4.074
C-3	15	7960	1.993	1.849	4.217
C-4	12	7320	1.877	1.101	3.528
C-5	12	7040	1.892	1.160	3.464
D-1	3*	2057	1.487	1.008	–
D-2	15	8373	2.032	1.537	5.0**
D-3	12	7885	1.874	1.849	4.191
D-4	12	8200	1.763	1.099	3.563
D-5	12	7540	1.813	1.101	3.564
D-6	11	7225	1.808	1.134	3.616
E-1	17	8391	2.144	1.399	3.2†
E-2	13	6960	1.964	1.851	4.168
E-3	12	6980	1.843	1.101	3.485
E-4	15	7519	1.897	1.170	3.504
F-1	16	8712	2.081	0.436	3.2†
F-2	14	8082	2.043	1.160	5.0

* Fractional cycle
† Fixed initial condition
** Short-notice enrichment change (5.0 w/o U-235 was upper limit permitted by QKCORE).

APPENDIX E

S Y S I N T

E.1 SYSINT Discussion

E.1.1 Introduction

SYSINT is a computerized version of the SYStem INTegration Model (SIM) discussed in Chapter 3. A summary of SYSINT characteristics was presented in Section 3.6.

SYSINT performs (1) the Booth-Baleriaux probabilistic utility system simulation for each time period in the planning horizon, (2) estimates all of the required cost components and, (3) outputs data for SYSOPT, the computerized SYStem OPTimization Model (SOM) of Chapter 4 and Appendix F.

E.1.2 Code Structure and Mode of Operation

Table E.1 presents a summary of SYSINT subroutine information while Figure E.1 portrays the general sequence of operations occurring in a SYSINT production run. (Table E.2 presents information relative to possible error messages printed by subroutine ERRMSG.)

The input to SYSINT is modularized into three separate datasets to permit maximum flexibility in changing parameters with a minimum number of input cards: strategy data (alternative maintenance schedules) change often, period data (e.g., load forecasts and fossil fuel costs) less often, unit data (heat rates) seldom.

Table E.1

Summary of SYSINT Subroutines

Name	Called By	Calls	Purpose
SYSINT (Main)	------	SUPSIM STRTIM	Main Program Prints Title
BLOCK DATA	------	------	Initializes data in COMMON areas
SUPSIM (QUIT)	SYSINT	BASIC PERIOD STRATG PRESIM PUNCHR ERRMSG CMPTIM ERASE	Supervises entire SYSINT simulation; Reads Control cards; Has ENTRY QUIT to terminate execution if severe error occurs.
BASIC	SUPSIM	PRPNDX ERRMSG	Reads basic system information (unit data)
PERIOD	SUPSIM	INNDEX ERRMSG	Reads period data and stores it on direct acess device
INNDEX (PRPNDX)	BASIC PERIOD STRATG LDGORD COMPRS	ERRMSG ERASE	Determines INDEX corresponding to a particular IDNO; Has ENTRY PRPNDX to initialize procedure.

Table E.1--Continued

Name	Called By	Calls	Purpose
STRATG	SUPSIM	INNDEX ERRMSG ERASE	Reads refueling and maintenance strategy input
PRESIM	SUPSIM	NUSCAL LDGORD SYSGEN GWHNRG PUNCHR CMPTIM ERASE	Performs pre-simulation data manipulation for each period
NUSCAL	PRESIM	GWHNRG ERRMSG	Changes spacing of PROB from that determined by input PKMW to DM
LDGORD	PRESIM	INNDEX COMPRS RETMRG MERGER ERRMSG ERASE	Optimizes loading order according to NORDOP and encodes order as 1000*NPT + INDEX

Table E.1--Continued

Name	Called By	Calls	Purpose
COMPRS (RETMRG)	LDGORD	INNDEX ERRMSG ERASE	Performs STATUS vs. IDNO check and then compresses and transfers NORDER into NTEMP; Alters incremental cost curves and optimizes startup order; Has ENTRY RETMRG to return incremental cost curves to original values.
MERGER	LDGORD	ERRMSG	Merges newly started plant increments with those of previously started plants
SYSGEN	PRESIM	SUBPLT GWHNRG ADDPLT PROBX SUSDNO PUNCHR ERRMSG ERASE	Supervises actual simulation; Calculates costs, etc.; Prints period output.
SUBPLT	SYSGEN	ERRMSG	Subtracts outages of plant-of interest from PROB
GWHNRG	PRESIM NUSCAL SYSGEN	------	Calculates energy under section of PROB

Table E.1--Continued

Name	Called By	Calls	Purpose
ADDPLT	SYSGEN	ERRMSG	Adds outages of plant-of-interest into PROB
PROBX	SYSGEN	------	Linearly interpolates PROB at a particular equivalent load
SUSDNO	SYSGEN	------	Estimates number of startup-shutdowns during the period
PUNCHR	SUPSIM	ERRMSG DAYTIM WHEN*	Performs output operations for SYSINT-to-SYSOPT output; Dependent upon IBM Data Utility Program IEBUPDTE (Release 20) (3)
ERRMSG	SUPSIM BASIC PERIOD INNDEX STRATG NUSCAL LDGORD COMPRS MERGER SYSGEN SUBPLT ADDPLT PUNCHR	QUIT	Prints error messages; Chooses to terminate execution if severe error occurs (see Table E.2)

Table E.1--Continued

Name	Called By	Calls	Purpose
CMPTIM (STRTIM) (DAYTIM)	SYSINT SUPSIM PRESIM PUNCHR	WHEN* TIMING*	Calls MIT internal clock routines to monitor execution time; Prints subroutine-to-subroutine transfer times; Has ENTRY STRTIM to start clock and ENTRY DAYTIM to print calendar date and time.
ERASE	SUPSIM PRPNDX STRATG PRESIM LDGORD COMPRS SYSGEN	------	MIT Assembler Language program that sets arrays to zeroes rapidly

*WHEN and TIMING ARE MIT internal clock subroutines

Figure E.1

SYSINT Flowchart

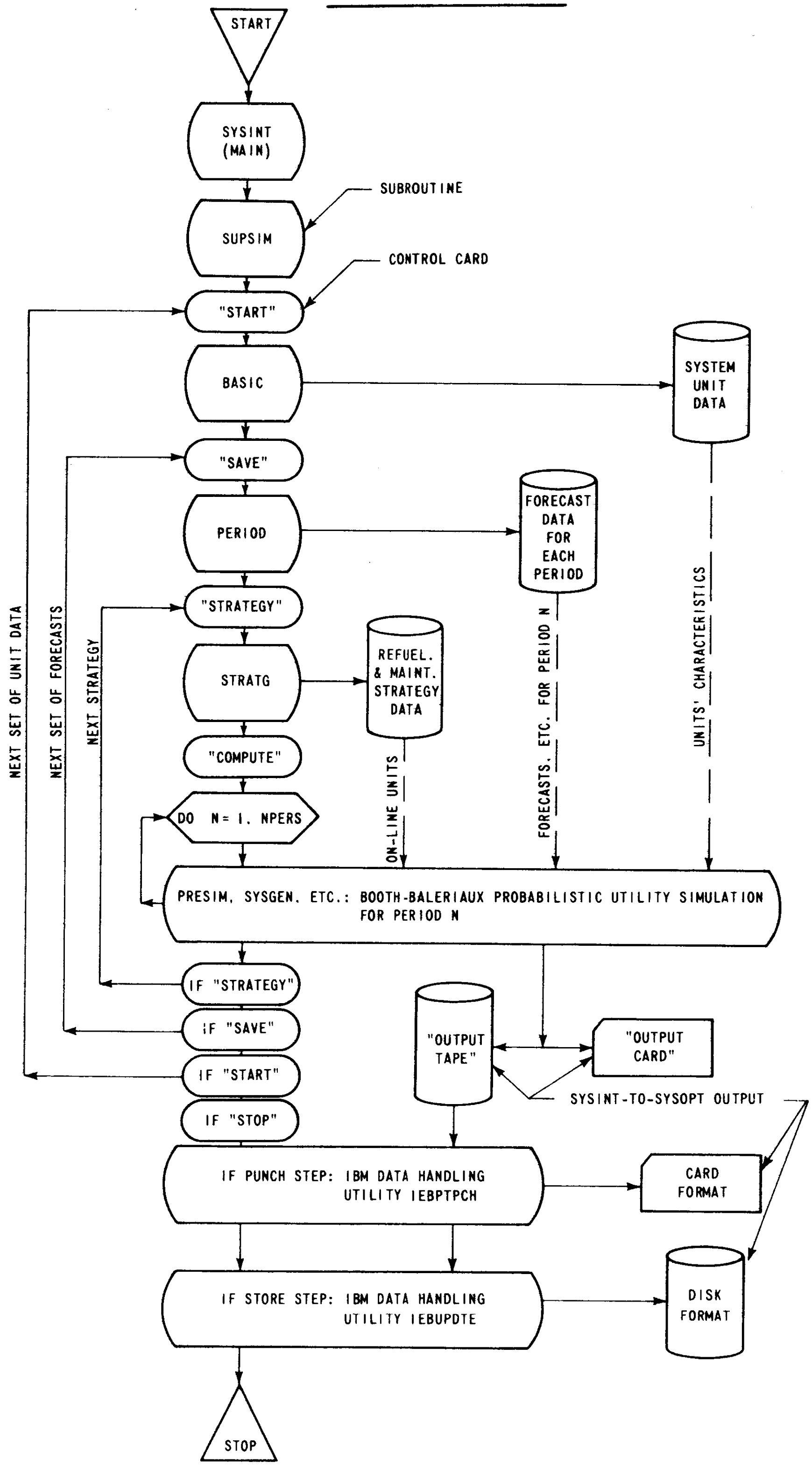


Table E.2

SYSINT Error Messages Printed by ERRMSG

Number*	Source	Action after Printing	Error
1	NUSCAL	CALL QUIT	PROB not dimensioned large enough
2	SUBPLT ADDPLT	CALL QUIT	Capacity of unit greater than minimum load
3	NUSCAL	RETURN	Warning of large error in changing PROB spacing
4	NUSCAL	CALL QUIT	New PROB violate properties of probability function
5	SYSGEN	RETURN	Warning of large error in total area under PROB
6	SUPSIM BASIC PERIOD STRATG	CALL QUIT	Input deck has improper sequence &/or card
7	INNDEX	CALL QUIT	Invalid or inconsistent IDNO encountered
8	SUPSIM	STOP	"STOP" Control Card; many small errors; etc.
9	LDGORD COMPRS MERGER PUNCHR	CALL QUIT	Input NORDER is improper
A(=10)	QUIT	STOP	QUIT executed "RETURN" to ERRMSG
B(=11)	PUNCHR	RETURN	Nuclear upper increments not consecutive
C(=12)	PUNCHR	RETURN	Nuclear minimums not base-loaded

*The error number initiating the ERRMSG print appears as the rightmost digit in the accumulated ERRCOD which is printed as part of the message.

E.1.3 SYSINT-to-SYSOPT Output Data Transfer

SYSINT-to-SYSOPT output can be obtained in either disk, magnetic tape or punched card format. All are in card image form with LRECL=80. Table E.3 summarizes the control cards and output modes.

Figure E.2 portrays accumulation of SYSINT strategy output during a single CALC step (see Section E.3, Figure E.5) in the computer run. After terminating the CALC step, the output must be separated by a STORE step or by hand for input into SYSOPT. As an example of the volume of output data involved, each of the three strategies of Chapter 5 (72 time periods each) produced 2164 punched cards. Figure E.3 presents the punched output of the sample SYSINT run shown in Figure E.5 of Section E.3.

Each strategy output deck begins with "./ADD NAME=" and "---BEGIN" cards and ends with a "---ABORT" or "---END" card followed by two blank cards. The ADD NAME card is used as input to the IBM utility IEBUPDTE (3) in the STORE step. [The IBM utility IEBPTPCH, used for printing and/or punching datasets in the PUNCH step, is also detailed in (3).] The ABORT card signifies abnormal termination of SYSINT-to-SYSOPT output due to SYSINT execution errors. The END card signifies normal (successful) completion of all SYSINT calculations and output.

Table E.3

SYSINT-to-SYSOPT Output Modes

Mode	"OUTPUT" Control Card 12	PUNCH Step Result (if included) (See Section E.3, Figure E.5)	STORE Step Result (if included) (See Section E.3, Figure E.5)	Comments
I	"NO TAPE"	....	....	No SYSINT-to-SYSOPT output
II	"CARD"	....	....	Punched Cards only
III	"TAPE"	Card Output	Disk Output	Most Versatile

Notes:

1. Card Output:
   - (a) No limit on number of strategies in one SYSINT run.
   - (b) May be put through later STORE step to create Disk output.
   - (c) Strategies may be separated and input directly into SYSOPT.
2. Tape Output:
   - (a) May be temporary direct access dataset on SYSDA device if STORE step used immediately to create Disk output with no limit on number of strategies per run.
   - (b) If actually a (backup) tape, may be put through later STORE step to create Disk output with no limit on number of strategies per run.
   - (c) May be input directly into SYSOPT but limit of one strategy per SYSINT run (i.e., per tape file).
3. Disk output: Preferred SYSOPT input since Disk output is on-line, provides faster data transfer and does not idly tie up tape drive during subsequent SYSOPT execution.

Figure E.2 6253-89

SYSINT - to - SYSOPT Output Data Transfer

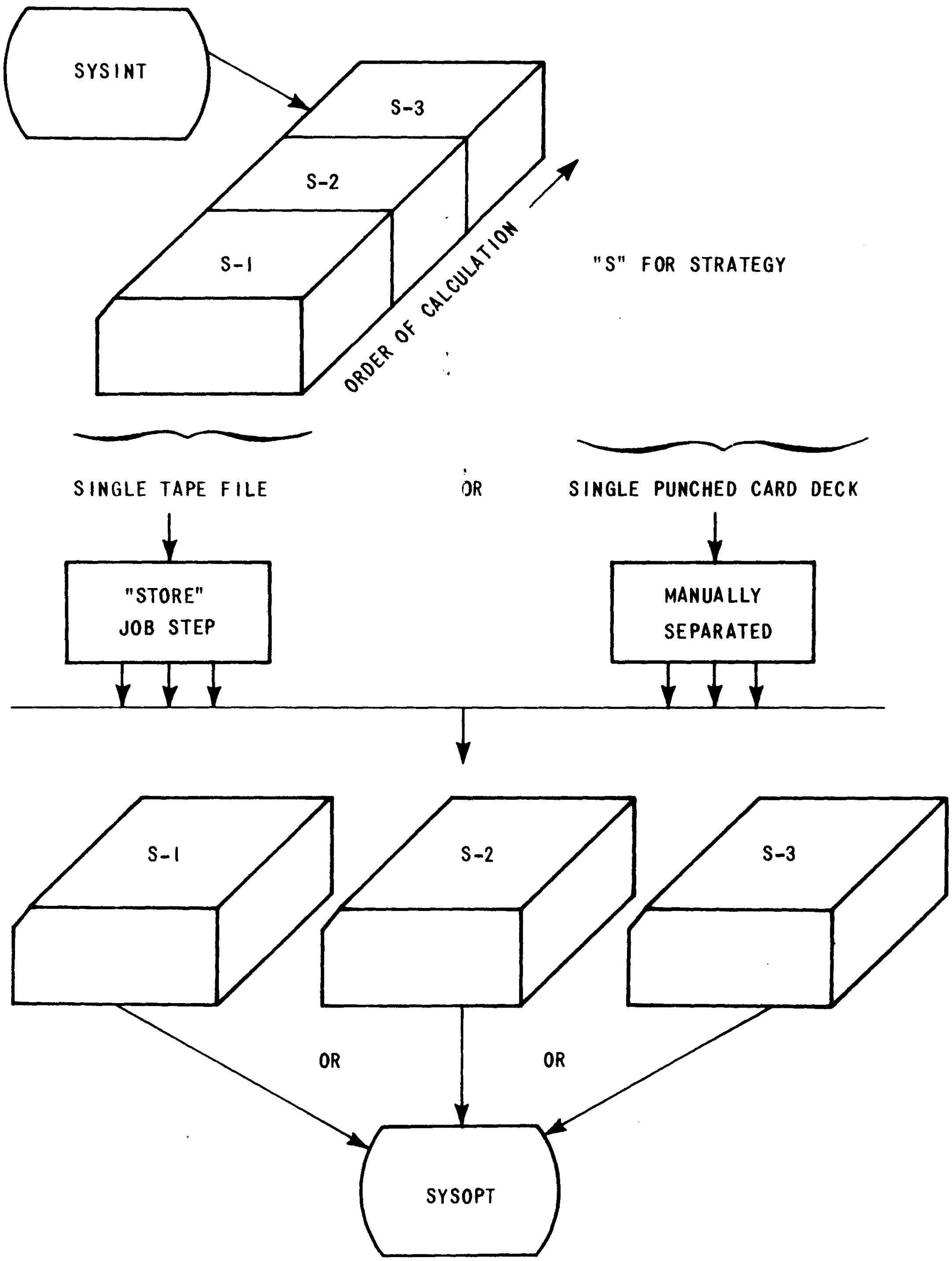


Figure E.3

Example of SYSINT-to-SYSOPT Output Data

```
./   ADD   NAME=T1000002,LEVEL=1,LIST=ALL
----------BEGIN  STRATEGY WITH NAME=T1000002 ON  2/18/73 AT    1*19*59.01----------
  T1     2      SAMPLE SYSINT RUN PERFORMING CALCS. FOR EXAMPLES   3 THRU 5
    2
  303  III  100  300            3
  505    V  300  800            5
NUKES* MAINT.DATA FOR     4 PERIODS    (       1 VALUES)
22220
22220
.............  1TH PERIOD TO FOLLOW SIMULATED   2/18/73  1*19*59.01.............
  EXAMPLE NO.  3  :  PROBABILISTIC MODEL ( WITH FORCED-OUTAGES )
        1  100.0000   730.0000    10.0000
MIN    700    8      8.967296254.929237506.887353758.839103759.626368777.386183791
.290287533.189331274
    0    0  657.0000  620.5000
MAX   1400    8      8.967296254.932671257.894812509.852920762.647335031.432731299
.363456297.283662543
 &FNLTOT
 MWINST=         2000,MWONLN=        2000,MWPEAK=         1800,MWMRGN=          200,
 MWSPIN=              0,PLOFL= 0.15647030     ,EXPDEM=  948.9999999999999     ,
 EXPGEN=  918.8888063575763     ,XNKGEN=  621.3474646260496     ,IDUM1=      0,
 XNNGEN=  297.5413417315266     ,EXPEMR=  30.11119364242359     ,IDUM2=      0,
 UNSRVD=  30.11119364242339     ,PRODS=  2600394.297870291     ,IDUM3=      0,
 $NKPRD=  1215804.453797403     ,$NNPRD=  1384589.844072888     ,IDUM4=      0,
 SUSD$=  205.5548917086381     ,$NKSUS= 0.42810199829546030-11,IDUM5=      0,
 $NNSUS=  206.5548917086338     ,$SHTOT=  2600600.852761999     ,IDUM6=      0,
 $NKTOT=  1215804.453797403     ,$NNTOT=  1384796.398964597     ,IDUM7=      0,
 EMRP$=  301111.9364242359     ,TOTAL$=  2901712.789186235
 &END
 CSTATU   AVLBTY   ENERGY    EXPHRS            EXPGWH            EXPBTU          COST
 19.0000 90.0000     0.0     657.000          146.38576          1595258.        303099.
 18.0000 85.0000     0.0     620.500          474.96171          5070586.        912706.
.............  3TH PERIOD TO FOLLOW SIMULATED   2/18/73  1*19*59.01.............
 EXAMPLE NO.  5  :  MULTIPLE INCREMENT BOOTH-BALERIAUX MODEL (V-2,THEN III-2)
        3  100.0000   730.0000    10.0000
MIN    600    8      7.965575003.927325006.885250008.836675009.615300025.374125038
.285875032
    0    0  657.0000  620.5000
MAX   1300    8      7.965575003.931150007.892900009.850825011.636625029.422000048
.360375046
 &FNLTOT
 MWINST=         2000,MWONLN=        2000,MWPEAK=         1800,MWMRGN=          200,
 MWSPIN=            80,PLOFL= 0.15047030     ,EXPDEM=  948.9999999999999     ,
 EXPGEN=  918.8888063575763     ,XNKGEN=  653.9896396680502     ,IDUM1=      0,
 XNNGEN=  264.8991666895261     ,EXPEMR=  30.11119364242359     ,IDUM2=      0,
 UNSRVD=  30.11119364242339     ,PRODS=  2481090.463857027     ,IDUM3=      0,
 $NKPRD=  1276032.582515141     ,$NNPRD=  1205057.881341865     ,IDUM4=      0,
 SUSD$=  10669.88822504196     ,$NKSUS= 0.36379788070917130-11,IDUM5=      0,
 $NNSUS=  10669.88822504196     ,$SHTOT=  2491760.352082068     ,IDUM6=      0,
 $NKTOT=  1276032.582515141     ,$NNTOT=  1215727.769566927     ,IDUM7=      0,
 EMRP$=  301111.9364242359     ,TOTAL$=  2792872.288506304
 &END
 CSTATU   AVLBTY   ENERGY    EXPHRS            EXPGWH            EXPBTU          COST
 19.0000 90.0000     0.0     657.000          169.59633          1827363.        347199.
 18.0000 85.0000     0.0     620.500          484.39331          5160186.        928834.
.............  4TH PERIOD TO FOLLOW SIMULATED   2/18/73  1*19*59.01.............
 EXAMPLE NO.  6  :  MULTIPLE INCREMENT BOOTH-BALERIAUX MODEL (III-2,THEN V-2)
        4  100.0000   730.0000    10.0000
MIN    600    8      7.965575003.927325006.885250008.836675009.615300025.374125038
.285875032
    0    0  657.0000  620.5000
MAX   1300    8      7.965575003.931150007.892900009.850825011.636625029.422000048
.360375046
 &FNLTOT
 MWINST=         2000,MWONLN=        2000,MWPEAK=         1800,MWMRGN=          200,
 MWSPIN=-2000000000,PLOFL= 0.15047030     ,EXPDEM=  948.9999999999999     ,
 EXPGEN=  918.8889063575763     ,XNKGEN=  653.9896396680501     ,IDUM1=      0,
 XNNGEN=  264.8991666895261     ,EXPEMR=  30.11119364242364     ,IDUM2=      0,
 UNSRVD=  30.11119364242339     ,PRODS=  2486316.159291511     ,IDUM3=      0,
 $NKPRD=  1281258.277949625     ,$NNPRD=  1205057.881341865     ,IDUM4=      0,
 SUSD$=  10669.88822504196     ,$NKSUS= 0.36379788070917130-11,IDUM5=      0,
 $NNSUS=  10669.88822504196     ,$SHTOT=  2496986.047516552     ,IDUM6=      0,
 $NKTOT=  1281258.277949625     ,$NNTOT=  1215727.769566927     ,IDUM7=      0,
 EMRP$=  301111.9364242364     ,TOTAL$=  2798097.983940789
 &END
 CSTATU   AVLBTY   ENERGY    EXPHRS            EXPGWH            EXPBTU          COST
 19.0000 90.0000     0.0     657.000          197.09999          2102400.        399456.
 18.0000 85.0000     0.0     620.500          456.88964          4898902.        881802.
----------END OF STRATEGY WITH NAME=T1000002 ON  2/18/73 AT    1*19*59.01----------
```

E.1.4 Altering Dataset Reference Numbers

Table E.4 presents the dataset reference number for each input/output device, their meaning and instructions for altering them for other computer installations.

Table E.4

SYSINT Dataset Reference Numbers

Fortran Symbol	Meaning	Current Value	Instructions for Altering
RD	Card Reader	5	See BLOCK DATA subroutine
WT	Output Printer	6	See BLOCK DATA subroutine
CARD	Card Punch	7	See BLOCK DATA subroutine
TAPE	Output tape or disk for SYSINT-to-SYSOPT output	8	See BLOCK DATA subroutine and any //G.FT08F001 Data Definition cards (see Section E.3)
------	Temporary direct access device storing period-by-period forecasts	9	See PERIOD and PRESIM subroutines and any //G.FT09F001 Data Definition cards (see Section E.3)
------	Final Disk dataset	SYSUT2	See Section E.3, Figure E.5

E.2 SYSINT Input Specifications

Table E.5 presents complete input specifications for SYSINT. The "START" Card 1 heads the plant data input module (Cards 2-10). The "SAVE" Card 11 heads the period data input module (Cards 12-20). Likewise, the "STRATEGY" Card 21 heads the maintenance strategy input module (Cards 22-24). "Compute" Cards 25-26 determine which periods of the strategy are executed. If no other modules are to be input and/or executed, a "STOP" Card 27 terminates SYSINT calculations.

Table E.5

SYSINT Input Specifications

Variable	Columns	Format	Description
Card 1			
...	1-5	...	"START" Control card initiates input processing for plant data, normalized startup-shutdown frequency function and load-duration shapes
Card 2			
...	1-10	...	"PLANT DATA" Header card for plant data
Card 3			
NOSTNS	1-5	I5	Number of units (stations) to be read in, $1 \leq$ NOSTNS ≤ 100

Note: For each of NOSTNS, a Card 4 of unit data is read in.

Variable	Columns	Format	Description
Card 4			
IDNO	1-4	I4	Unique unit identification number
NAME	5-8	A4	Unit name
TYPE	10	1X,A1	Type of Unit: F= Fossil H= Hydro (not currently used) N= Nuclear P= Peaking S= Pumped-storage (not currently used)
SUSDHT	11-20	F10.0	Q_r unit startup-shutdown equivalent heat requirement, MegaBTU
PNOM	21-29	F9.5	p_r unit performance probability, fraction
NPTS	30	I1	I total number of capacity increments, $1 \leq I \leq 5$

Variable	Columns	Format	Description
MWPT	...	I4	K_{ri} cumulative unit capacity, MW
HTRAT	...	F6.0	$h_{inc_{ri}}$ incremental heat rate, BTU/kwhe

Note: Continue (MWPT, HTRAT) sets until all I increments have been read in.

Card 5

...	1-20	...	"NORMALIZED SUSD DATA" Header card for normalized startup-shutdown frequency function

Note: There are three Card 6's required to read in the 20 Ω values

Card 6

F(1) to F(20)	1-80	8F10.4	$\Omega(L'_{r1})$ normalized startup-shutdown frequency function at increments of 0.05 of L'_{r1} $(.05 \leq L'_{r1} \leq 1.00)$; $\Omega(0) \equiv 0$; linear interpolation between points; per day

Card 7

...	1-10	...	"LOAD TYPES" Header card for load-duration shapes

Card 8

LDTYPS	1-5	I5	Total number of normalized load-duration shapes, $1 \leq$ LDTYPS ≤ 25

Note: For each of LDTYPS, Cards 9 and 10 of load shape data are read in.

Card 9

LDTYPE	1-5	I5	Unique load-duration shape identification number, $1 \leq$ LDTYPE ≤ 25

Variable	Columns	Format	Description
NUMONE	6-10	I5	Number of 1's to be prefixed to load shape data on Card 10, $0 \leq$ NUMONE ≤ 49

Note: There are [(50-NUMONE + 7)/8] Card 10's to be read in for this LDTYPE

Card 10

Variable	Columns	Format	Description
PROB (NUMONE + 1) to PROB (50)	1-80	8F10.4	F_D completely normalized customer load-duration curve from minimum load to peak demand where $F_D = 0$ for first time. (Usually $F_D = 0$ at PROB (50) resulting in spacing of 2% of PKMW.), fractional duration

Card 11

Variable	Columns	Format	Description
...	1-4	...	"SAVE" Control card signifying previous data on Cards 1-10 to be saved.

Card 12

Variable	Columns	Format	Description
...	1-7	...	"OUTPUT ", Control card for large volume of data to be transferred to SYSOPT
...	8-11	...	"TAPE", SYSINT data output to temporary dataset with Dataset Reference Number = TAPE (See BLOCK DATA subroutine and Section E.1.3)
...	8-11	...	"CARD", SYSINT data output to card punch with Dataset Reference Number = CARD (See BLOCK DATA Subroutine and Section E.1.3)
...	8-14	...	"NO TAPE", SYSINT-to-SYSOPT data not desired

Note: Choose one or include all three types of Card 12 with only last one read being valid

Variable	Columns	Format	Description
Card 13			
...	1-14	...	"OUTPUT PRINT ", Control card for printed output on Data-set Reference Number = WT (See BLOCK DATA Subroutine and Section E.1.4)
...	15-18	...	"MINI" prints input edit, unit incremental costs, unit production totals and system totals
...	15-18	...	"MIDI" prints MINI plus Unit increment loading during load step of Booth-Baleriaux algorithm
...	15-18	...	"MAXI" prints MIDI plus all F's calculated at each convolve or deconvolve operation (Warning: This option should be used only for very small problems.)

Note: Choose one or include all three types of Card 13 with only last one read being valid.

Note: A set of Cards 14-20 is included for each of NPERS (up to 100) periods desired in planning horizon. Each NPER need not be entered in numerical order.

Variable	Columns	Format	Description
Card 14			
...	1-6	...	"PERIOD" Header card
...	7-10	A4	Free for Comments
Card 15			
PDTITL	1-80	10A8	Period title
Card 16			
NPER	1-4	I4	Period number, $1 \leq$ NPER ≤ 100
LDTYPE	5-8	I4	Load shape desired, $1 \leq$ LDTYPE ≤ 25

Variable	Columns	Format	Description
PKMW	9-16	F8.0	Peak customer demand, MW (The resulting minimum load must not be less than largest unit on the system.)
SPNRES	17-24	F8.0	Spinning reserve requirement, (Expected) MW
DM	25-32	F8.0	Desired equivalent load curve spacing, should be 1 to 4% of PKMW if PROB (49) ≠ 0 (Card 10), MW
DT	33-40	F8.0	T', Duration of period, hours
CSTEMR	41-48	F8.0	$\bar{e}_U$, Average cost of emergency purchases, $/MWH
CSTFOS*	49-56	F8.0	ϕ_F, Cost of fossil fuel for all fossil units, ¢/MegaBTU
CSTNUK*	57-64	F8.0	ϕ_N, Cost of nuclear fuel for all nuclear units, ¢/MegaBTU
CSTPKG*	65-72	F8.0	Cost of fuel for all peaking units, ¢/MegaBTU
AVLALL*	73-80	F8.0	p_r, Performance probability for all units, (If 0.0 or blank, 100*$PNOM_r$ used for each unit for first period read.), per cent

* Requires non-zero, non-negative entry to be effective. To input zero, use 1.E-50. If left blank, has no effect on data remaining after previous period was processed.

Note: Card 17 included for each unit whose data is to be altered from current values (i.e., last period processed plus effects of CSTFOS, CSTNUK, CSTPKG or AVLALL for this period).

Card 17 (Optional)

...	1-5	...	"ALTER" Control Card
ID	17-20	11X,14	IDNO for unit whose data is to be altered

Variable	Columns	Format	Description
CST*	21-30	F10.4	ϕ_r unit fuel cost, ¢/MegaBTU
AVL*	31-40	F10.4	p_r unit performance probability, per cent
ENER*	41-50	F10.4	Scarce-resource energy available (not currently used)

Note: Cards 18-20 optional if period is to use same startup-shutdown data remaining after previous period was processed.

Card 18

Variable	Columns	Format	Description
...	1-9	...	"SUSD DATA" Control Card

Card 19

Variable	Columns	Format	Description
NORDOP	1-5	I5	Loading order optimization option: =1 , no optimization, NORDER used as input. Each of NOENTY represents next increment of that unit.(SPNRES, NOBASE and NOPEAK ignored). =2 , Base group order as is; Intermediate group started in given order for either economic or spinning reserve reasons; Peaking group started in economic order after all of increments in Intermediate group. =3 , Same as 2 but Intermediate group started in economic order =4 , Same as 3 but Peaking group competes economically after last unit of Intermediate group is started.
NOENTY	6-10	I5	Number of NORDER entries to be read
NOBASE	11-15	I5	The first NOBASE entries in NORDER form the Base group of increments and are started in the order specified (i.e., the must-run increments)

Variable	Columns	Format	Description
NOPEAK	16-20	I5	The last NOPEAK entries in NORDER form the Peaking group regardless of unit TYPE. The Intermediate (central) group is made up of the remaining NOENTY-NOBASE-NOPEAK entries in NORDER.

Note: There are [(NOENTY + 15)/16] Card 20's to be read in.

Card 20

Variable	Columns	Format	Description
NORDER (1) to NORDER (NOENTY)	1-80	16I5	Input startup-shutdown order, unit (increment) IDNO. SYSINT automatically strips out off-line units and, therefore, it is wise to include all units in NORDER since various strategies will have different off-line units in the same period.

Note: A set of Cards 21-26 is included for each strategy (no limit on number of strategies) to be calculated.

Card 21

Variable	Columns	Format	Description
...	1-8	...	"STRATEGY" Control Card

Note: Card 13 required here if this is not first strategy.

Card 22

Variable	Columns	Format	Description
NPM	3	2X,L1	Nuclear power management assumption check option, =F, SYSINT-SYSOPT assumptions concerning nuclear plant utilization not checked (That is, base-load nuclear minimums and all nuclear upper increments consecutive) ="T", Assumptions checked
IPLACE	4	I1	Version of strategy if same strategy was run previously

Variable	Columns	Format	Description
IDSTRG	5-10	I6	Strategy identification number. If < 0, skips SYSGEN calculations for input check

Note: SYSINT-to-SYSOPT output data stored using 8 alphameric character membername = NPM + 10^6 *IPLACE + IDSTRG which should be unique to save old results with same IDSTRG

Variable	Columns	Format	Description
SGTITL	11-80	10A7	Strategy Title

Card 23

Variable	Columns	Format	Description
...	1-11	...	"MAINT. DATA" Header card

Note: Card 24 must appear for each of NOSTNS.

Card 24

Variable	Columns	Format	Description
ID	1-4	I4	Unit IDNO for which maintenance data card applies
NAM	5-8	A4	Unit NAME (optional)
NOTZRO(1)	11-15	2X,I5	Unit installed just prior to period NOTZRO(1). If blank or zero, unit already installed before beginning strategy
NOTZRO(2)	16-20	I5	Unit retired after period NOTZRO(2). If blank or zero, unit not retired during strategy
NDOWN(1) to NDOWN (20)	21-80	20I3	Period number during which unit off-line for maintenance (or refueling). If blank, zero or > NPERS has no effect

Note: If "COMPUTE" Card 25 omitted, only checks input of strategy and/or periods.

Card 25

Variable	Columns	Format	Description
...	1-8	...	"COMPUTE "Control Card initiates computation of strategy for all indicated periods
...	9-12	...	"SOME" (optional) only some of NPERS to be calculated

Variable	Columns	Format	Description

Note: Card 26 included only if "SOME" included on "COMPUTE" Card 25. Then, there must be [(NPERS + 79)/80] Card 26's.

Card 26

Variable	Columns	Format	Description
DOPERD(1) to DOPERD (NPERS)	1-80	80L1	Calculate period NPER = Card Column? "T" = Yes "F" or blank = No

Note: Next card may be "START" Card 1, "SAVE" Card 11, "STRATEGY" Card 21 or "STOP" Card 27. Control reverts back to that point in Card input sequence.

Card 27

Variable	Columns	Format	Description
...	1-4	...	"STOP" Control card to terminate SYSINT execution for this computer run.

E.3 <u>SYSINT Sample Problems</u>

Two sample problem input decks are presented in Figures E.4 and E.5. The deck in Figure E.4 was actually used to generate Reference Utility System Examples 1 and 2 (see Appendix C). The deck in Figure E.5 was likewise used for Examples 3 to 6 and to produce the SYSINT-to-SYSOPT output deck example in Figure E.3.

Figure E.4
SYSINT Sample Problem Input Deck I

```
//  'DEATON',CLASS=A,REGION=140K
/*MITID  USER=(M7894,6948)
/*SRI LOW
/*MAIN LINES=10,CARDS=00,TIME=1
//CALC   EXEC FORG,PROG='USERFILE.M7894.6948.LOAD.SYSINT(GO)',
//G.FT08F001 DD UNIT=SYSDA,DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600),
//   SPACE=(CYL,(3,3)),DISP=(MOD,PASS)
//G.FT09F001 DD UNIT=SYSDA,DCB=BUFNO=2,SPACE=(4000,(100))
//G.SYSIN DD *
START
PLANT DATA
    5
 101    I P        50.       .95 1 100 18000
 202   II F       800.       .95 2 100 11000 200   8500
 303  III N      1200.       .90 2 100 12000 300 10000
 404   IV F      3600.       .90 2 200  9800 600  8300
 505    V N      2400.       .85 2 300 12500 800  9500
NORMALIZED SUSD DATA
      .02       .07       .12       .19       .28       .44       .62       .74
      .83       .86       .86       .86       .83       .77       .66       .37
      .19       .09       .03       .00
LOAD TYPES
    1
    1    8
       .95       .90       .85       .80       .50       .20       .15       .10
       .05       .00       .00       .00       .00       .00       .00       .00
       .00       .00       .00       .00       .00       .00       .00       .00
       .00       .00       .00       .00       .00       .00       .00       .00
       .00       .00       .00       .00       .00       .00       .00       .00
       .00       .00
SAVE
OUTPUT NO TAPE
OUTPUT PRINT  MAXI
PERIOD  1
  EXAMPLE NO.  1  :  DETERMINISTIC MODEL  ( NO FORCED-OUTAGES )
   1   1  1800.      0.     100.     730.      10.      40.      18.      90.     100.
ALTER              303        15.
ALTER              202        50.
SUSD DATA
    3    5    4    1
  505  303  404  202  101
STRATEGY
  F1      1       SAMPLE SYSINT RUN PERFORMING CALCS. FOR EXAMPLES 1 & 2
MAINT. DATA
 505
 404
 303
 202
 101
COMPUTE SOME
TT
START
PLANT DATA
    5
 101    I P        50.       .95 1  95 16000
 202   II F       800.       .95 2  95 11000 190   8500
 303  III N      1200.       .90 2  90 12000 270 10000
 404   IV F      3600.       .90 2 180  9800 540  8300
 505    V N      2400.       .85 2 255 12500 680  9500
NORMALIZED SUSD DATA
      .02       .07       .12       .19       .28       .44       .62       .74
      .83       .86       .86       .86       .83       .77       .66       .37
      .19       .09       .03       .00
LOAD TYPES
    1
    1    8
       .95       .90       .85       .80       .50       .20       .15       .10
       .05       .00       .00       .00       .00       .00       .00       .00
       .00       .00       .00       .00       .00       .00       .00       .00
       .00       .00       .00       .00       .00       .00       .00       .00
       .00       .00       .00       .00       .00       .00       .00       .00
       .00       .00
SAVE
OUTPUT NO TAPE
OUTPUT PRINT  MAXI
PERIOD  1
  EXAMPLE NO.  2  :  DETERMINISTIC MODEL  ( REDUCED CAPACITIES )
   1   1  1800.      0.     100.     730.      10.      40.      18.      90.     100.
ALTER              303        15.
ALTER              202        50.
SUSD DATA
    3    5    4    1
  505  303  404  202  101
STRATEGY
  F2      1       SAMPLE SYSINT RUN PERFORMING CALCS. FOR EXAMPLES 1 & 2
MAINT. DATA
 101
 202
 303
 404
 505
COMPUTE
STOP
/*
```

Figure E.5

SYSINT Sample Problem Input Deck II

```
// 'DEATON',CLASS=A,REGION=140K
/*MITID , USER=(47894,6948)
/*SRI LOW
/*MAIN LINES=10,CARDS=0,TIME=1
//CALC  EXEC FORG,PROC=USERFILE.M7894.6948.LOAD.SYSINT(GO),
//G.FT08F001 DD UNIT=SYSDA,DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600),
//   SPACE=(CYL,(3,3)),DISP=(NEW,PASS)
//G.FT09F001 DD UNIT=SYSDA,DCB=BUFNO=2,SPACE=(4000,(100))
//G.SYSIN DD *
START
PLANT DATA
    5
 101   I P        50.      .95 1 100 18000
 202  II F       800.      .95 2 100 11000 200  8000
 303 III N      1250.      .90 2 100 12000 300 10000
 404  IV F      3600.      .90 2 200  9800 600  8300
 505   V N      2400.      .85 2 300 12500 800  9500
NORMALIZED SUSD DATA
     .02        .07        .12        .19        .28        .44        .62        .74
     .83        .86        .86        .86        .83        .77        .66        .37
     .19        .09        .03        .00
LOAD TYPES
    1
    1    9
       .95        .90        .85        .80        .50        .20        .15        .10
       .05        .00        .00        .00        .00        .00        .00        .00
       .00        .00        .00        .00        .00        .00        .00        .00
       .00        .00        .00        .00        .00        .00        .00        .00
       .00        .00        .00        .00        .00        .00        .00        .00
       .00        .00
SAVE
OUTPUT TAPE
OUTPUT PRINT  MAXI
PERIOD  1
  EXAMPLE NO.  3  :  PROBABILISTIC MODEL ( WITH FORCED-OUTAGES )
   1   1   1800.        0.     100.     730.      10.      40.      18.      90.
ALTER             303         15.
ALTER             202         50.
SUSD DATA
   3    5    4    1
  505  303  404  202  101
PERIOD  2
  EXAMPLE NO.  4  :  SINGLE INCREMENT BOOTH-BALERIAUX MODEL
   2   1   1800.        0.     100.     730.      10.
SUSD DATA
   1    9    9    0
  505  505  303  303  404  404  202  202  101
PERIOD  3
  EXAMPLE NO.  5  :  MULTIPLE INCREMENT BOOTH-BALERIAUX MODEL (V-2,THEN III-2)
   3   1   1800.       80.     100.     730.      10.
SUSD DATA
   3    5    3    1
  505  303  404  202  101
PERIOD  4
  EXAMPLE NO.  6  :  MULTIPLE INCREMENT BOOTH-BALERIAUX MODEL (III-2,THEN V-2)
   4   1   1800.        0.     100.     730.      10.
SUSD DATA
   1    9    9    0
  505  303  404  303  505  202  404  202  101
STRATEGY
  T1    2       SAMPLE SYSINT RUN PERFORMING CALCS. FOR EXAMPLES  3 THRU 5
MAINT. DATA
 202
 404
 101
 505
 303
COMPUTE SOME
T TT
STOP
/*
//PUNCH  EXEC  PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=*.CALC.G.FT08F001,DISP=(OLD,PASS)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
    PUNCH  MAXFLDS=1
   RECORD  FIELD=(80)
/*
//STORE EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=USERFILE.M7894.6948.RESULTS.SYSINT,
//           DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600),
//           SPACE=(1600,(1,1,2),,,ROUND),
//           VOL=REF=RENTDISK,DISP=(NEW,CATLG)
//SYSIN DD DSN=*.CALC.G.FT08F001,DISP=(MOD,DELETE),
//  VOLUME=REF=*.CALC.G.FT08F001
/*
```

E.4 SYSINT Source Listing

The following is a Fortran IV source listing of the SYSINT code (included only in MIT library copies).

APPENDIX F

S Y S O P T

F.1 SYSOPT Discussion

F.1.1 Introduction

SYSOPT is a computerized version of the SYStem OPTimization Model (SOM) discussed in Chapter 4. A summary of SYSOPT characteristics was presented in Section 4.6.

SYSOPT performs the nuclear system optimization in conjunction with CORSOM's (specifically QKCORE of Appendix H using the Out-of-Kilter (O-O-K) Network Program of Appendix G. Input is accepted in the form of output from SYSINT (See Section E.1.3) as well as SYSOPT's own card input.

F.1.2 Code Structure and Mode of Operation

Table F.1 presents a summary of SYSOPT subroutines while Figure F.1 portrays the general sequence of operations occurring in a SYSOPT production run. (Table F.2 presents information relative to possible error messages printed by subroutine OPERR.)

In interfacing with the off-line code SYSINT, the SYSINT-to-SYSOPT output is transferred per Section E.1.3.

To be operational, SYSOPT must be link-edited with O-O-K since variables are transferred into and out of O-O-K's storage on-line by SYSOPT. The structure of the network itself and the resulting arc "Types" are indicated in Figure F.2.

Table F.1

Summary of SYSOPT Subroutines

Name	Called By	Calls	Purpose
SYSOPT (Main)	------	RDOPTN RDSTRG RDPERS ASMTYS WTPERS SETUPN SETUPT CONVRG CHKSHP EDTSHP OPTMUM LOC OPERR CMPTIM STRTIM ICNPUT[1]	Oversees entire SYSOPT optimization; Calls ICNPUT to permit INCORE Model to read input
BLOCK DATA	------	------	Initializes data in Common areas; Dimensions Out-of-Kilter (O-O-K) Network Program arrays
RDOPTN	SYSOPT	PVINIT OPERR	Reads in data directly pertinent to SYSOPT

[1] INCORE Model subroutines (see QKCORE, Appendix H)

Table F.1--Continued

Name	Called By	Calls	Purpose
RDSTRG	SYSOPT	OPERR ERASE	Reads SYSINT-to-SYSOPT information relative to maintenance and refueling strategy
RDPERS	SYSOPT	PDCALC OPERR	Reads SYSINT-to-SYSOPT information relative to period results
PDCALC	RDPERS	SUBPLT GWHNRG PROBX OPERR	Performs various pre-calculations for each period; Sets up some costs and limits for network arcs
SUBPLT	PDCALC	OPERR	Subtracts plant-of-interest from PROB; Similar to SUBPLT of SYSINT
GWHNRG	PDCALC	------	Calculates energy under section of PROB; Identical to GWHNRG of SYSINT
PROBX	PDCALC	------	Linearly interpolates PROB at a particular equivalent load; Identical to PROBX of SYSINT
ASMTYS	SYSOPT	PVPER$	Assembles various calendar dates to and beyond horizon

Table F.1--Continued

Name	Called By	Calls	Purpose
WTPERS	SYSOPT	OPERR ERASE	Writes out input for the various periods and system horizon totals
SETUPN	SYSOPT	ONLY$$ LOC ERASE	Sets up costs and limits for remaining arcs in the network
SETUPT	SYSOPT	OPERR	Sets up input tape for Out of Kilter (O-O-K) Network Program
CONVRG	SYSOPT	CALSHP ARCPRT SETELE NEWMRG PVPER$ LOC OPERR ERASE OOKMAN[1] INCORE[2]	Supervises inner cost convergence between OOKMAN (O-O-K Main Program) and INCORE Model

[1]Out of Kilter (O-O-K) Network Program subroutines (see Appendix G)

[2]INCORE Model subroutines (see QKCORE, Appendix H)

Table F.1--Continued

Name	Called By	Calls	Purpose
CALSHP	CONVRG CHKSHP	LOC	Calculates shape criterion for each period
ARCPRT	CONVRG	LOC OPERR	Prints O-O-K arcs after inner cost iteration
SETELE	CONVRG OPTMUM	ERASE	Sets up new table of E's to be investigated by INCORE Model
NEWMRG	CONVRG	LOC OPERR	Sets up new table of λ's to be used by O-O-K
PVPER$ (PVINIT)	RDOPTN ASMTYS CONVRG	------	Calculates present (at base date) value of one dollar; Has ENTRY PVINIT to initialize present value rate; Identical to QKCORE version
CHKSHP (ONLY$$)	SYSOPT SETUPN	CALSHP ARCPRT SQUEEZ LOC ERASE OOKMAN[1]	Performs outer shape iteration and checks shape criteria to evaluate acceptability; Has ENTRY ONLY$$ to change objective function of O-O-K from shape to cost

[1]Out of Kilter (O-O-K) Network Program subroutines (see Appendix G)

Table F.1--Continued

Name	Called By	Calls	Purpose
SQUEEZ	CHKSHP	LOC	Squeezes reactor period energy production range
EDTSHP	SYSOPT	LOC	Edits shape information and prints final altered energy limits
OPTMUM	SYSOPT	SETELE INCORE[1]	Supervises printing of optimum solution; Calls INCORE Model to get final nuclear costs at optimum; Totals all operating revenue requirements
LOC	SYSOPT SETUPN CONVRG CALSHP ARCPRT NEWMRG CHKSHP SQUEEZ EDTSHP	------	Calculates pointer to desired network arc

[1]INCORE Model subroutines (see QKCORE, Appendix H)

Table F.1--Continued

Name	Called By	Calls	Purpose
OPERR	SYSOPT RDOPTN RDSTRG RDPERS PDCALC SUBPLT WTPERS SETUPT CONVRG ARCPRT NEWMRG	ICERRS[1]	Prints error messages and chooses to terminate execution if severe error occurs (see Table F.2); Calls ICERRS to get final INCORE Model error edit
CMPTIM (STRTIM) (DAYTIM)	SYSOPT	WHEN[2] TIMING[2]	Calls MIT internal clock routines to monitor execution time; Prints subroutine-to-subroutine transfer times; Has ENTRY STRTIM to start clock and ENTRY DAYTIM to print calendar date and time
ERASE	RDSTRG WTPERS SETUPN ERASE SETELE CHKSHP	------	MIT Assembler Language program that sets arrays to zeroes rapidly

[1]INCORE Model subroutines (see QKCORE, Appendix H)

[2]WHEN and TIMING are MIT internal clock subroutines

Figure F.1

SYSOPT Flowchart

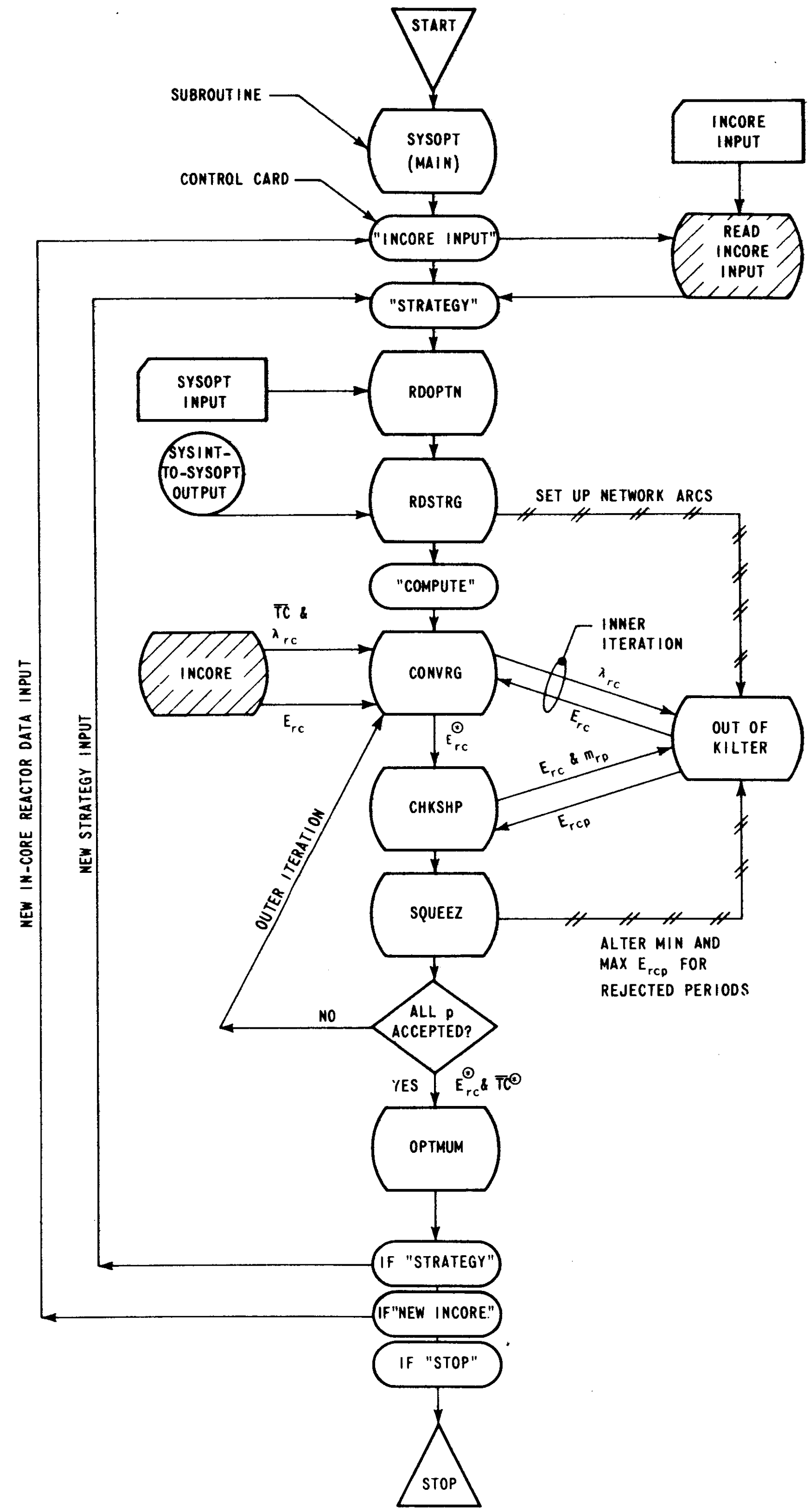


Table F.2

SYSOPT Error Messages Printed by OPERR

Number*	Source	Action after Printing	Error
1	RDPERS	Terminate	PROB Data inconsistent
2	PDCALC SUBPLT	Terminate	Nuclear upper increment not consecutive or unit capacity ≥ minimum load
3	RDSTRG	Terminate	Reactor or Strategy IDNO's do not agree
4	SETUPT	Terminate	Number of arcs input to O-O-K and equation in Figure F.2 do not agree
5	NEWMRG	RETURN	Incremental cost curve not monotonically increasing
6	SYSOPT RDOPTN WTPERS	Terminate	Improper input sequence and/or card; Input option outside limits
7	CONVRG	RETURN	MXITER reached without complete convergence
8	SYSOPT	Terminate	"STOP" Card 10 encountered in input or other severe error
9	CONVRG	RETURN	$\overline{TC}$ converged within TH$CON

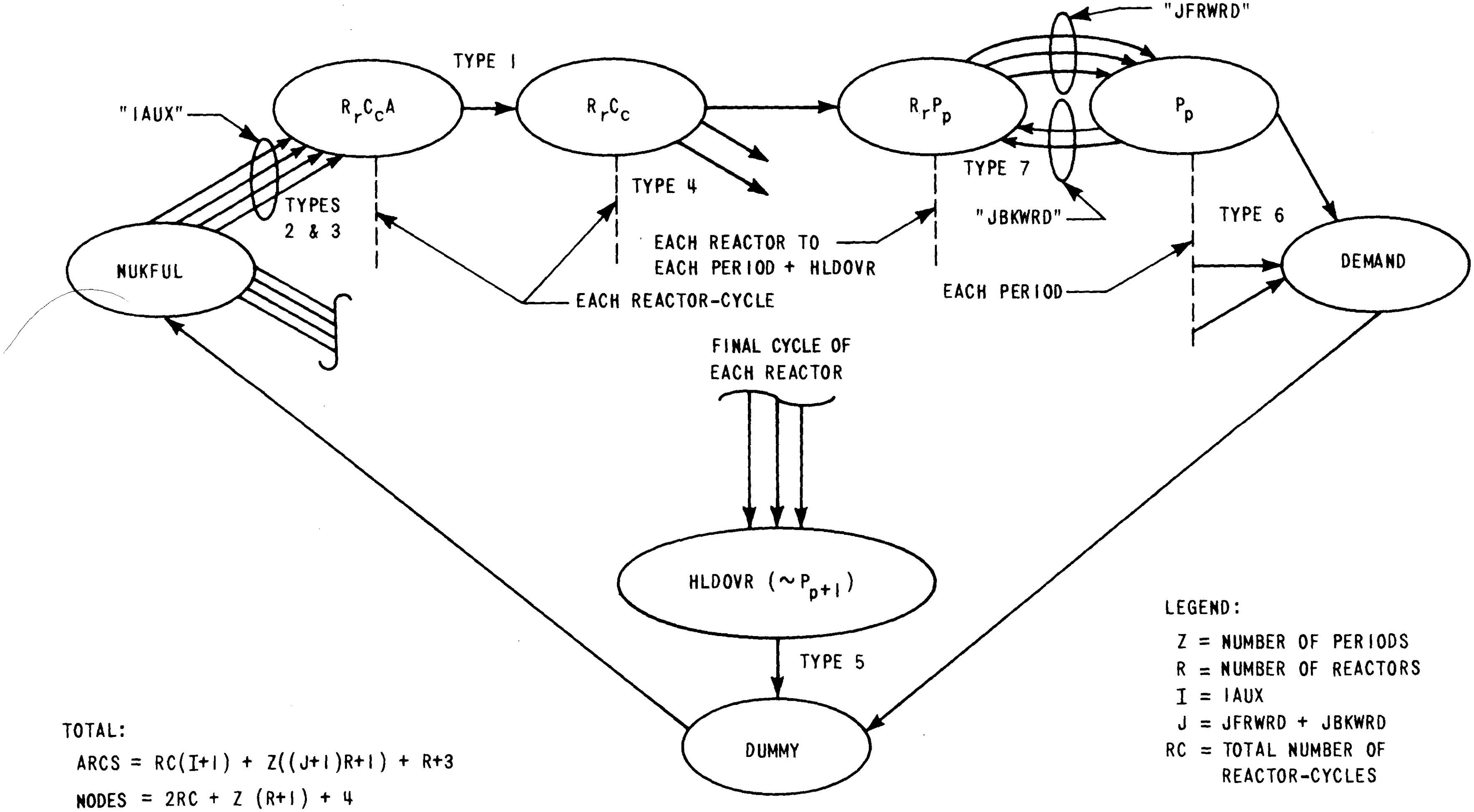
*The error number initiating the OPERR print appears as the rightmost digit in the accumulated ERRCOD (which is printed as part of the message).

Table F.2--Continued

Number*	Source	Action after Printing	Error
A(=10)	CONVRG	Terminate	INCORE and SYSOPT using different present value rates
B(=11)	ARCPRT	Terminate	No feasible solution to O-O-K problem
C(=12)	RDSTRG RDPERS WTPERS	Terminate	Premature end to SYSINT data; some periods not read in
D(=13)	NEWMRG	RETURN	Cycle energy greater than its upper stretchout limit

Figure F.2

Nuclear Energy Network Structure



6253-90

Relative to INCORE interfacing, only four distinct points of SYSOPT-INCORE contact are necessary to ensure compatibility with general CORSOM's:

(1) SYSOPT itself calls ICNPUT [if an "INCORE INPUT" Control Card 1 is encountered (See Section F.2)] to permit an INCORE Model to read any data required by it (e.g., core initial conditions and cost parameters).

(2) Subroutine CONVRG calls INCORE subroutine with the arguments specified in Table F.3. This call is executed many times as this is the actual inner iteration. The important results are $\overline{TC}_r$ (returned as RTC) and the λ_{rc} (appearing "sandwiched" between the pertinent E_{rc} and $E_{rc}+\Delta$ in array ELAME as in Section H.1.3.

(3) Subroutine OPTIMUM also calls INCORE subroutine per Table F.3, but only to evaluate the final optimum reload designs in more detail. COMMON area /PRINTS/ is used for passing any print options or dataset reference numbers.

(4) Finally, subroutine OPERR calls INCORE error subroutine ICERRS to permit printing final edit of any INCORE Model errors encountered during the SYSOPT optimization.

When SYSOPT and O-O-K have been link-edited with the particular simulator QKCORE, core storage requirements (See Section 4.6) can be reduced by 125 K bytes of storage or

Table F.3

Interfacing of SYSOPT and an INCORE Model

(SYSOPT) Variable	Supplied By	Description
IDNUM	SYSOPT	Unit IDNO
NCYCIN	SYSOPT	Number of cycles at least partially within horizon
NCYCXS	SYSOPT	Number of whole cycles specified beyond horizon
NCYCTO	SYSOPT	=NCYCIN + NCYCXS = total
TSY(1) to TSY(NCYCTO)	SYSOPT	Calendar time at start of cycle, years
TEY(1) to TEY(NCYCTO)	SYSOPT	Calendar time at end of cycle, years
NECBAL(1) to NECBAL(NCYCTO)	SYSOPT	Position of key E_{rc} within ELAME representing E^t_{rc} (See Section H.1.3)
ELAME(1,1) to ELAME($2n_\lambda$+1,NCYCTO)	E by SYSOPT λ by INCORE	E_{rc} cycle energy and λ_{rc} incremental costs (See Section H.1.3)
MXESX2	SYSOPT	n_λ number of Δ stair-steps in each λ_{rc} incremental cost curve
ECHDOV	SYSOPT	Holdover energy, GWHe

Table F.3--Continued

(SYSOPT) Variable	Supplied By	Description
RTC	INCORE	Total nuclear fuel cost including appropriate fraction of cost of cycle split by horizon, 10^3 dollars
PVR	INCORE	x, present value rate used by INCORE, fraction per year
YBS	INCORE	Calendar time base date of present valuing in INCORE, years
ECUPLM(1) to ECUPLM(NCYCTO)	INCORE	Upper limit of energy extractable from each cycle that has reload enrichment fixed, GWHe
TOY(1) to TOY(NCYCTO)	SYSOPT	Length of time that unit is operating during cycle, years

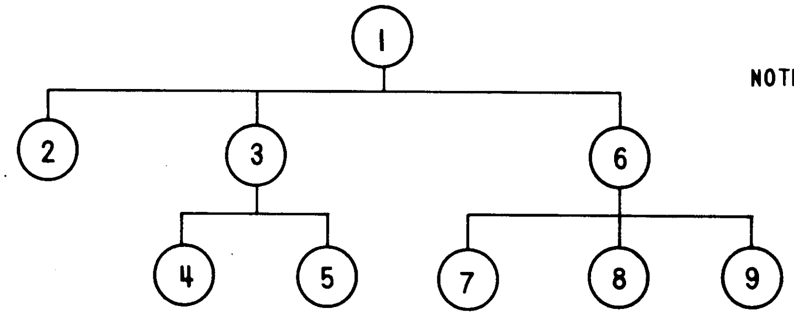
one-third of total (with negligible increase in computing time) by using the overlay structure of Figure F.3.

F.1.3 Altering Dataset Reference Numbers

Table F.4 presents the dataset reference numbers for each input/output device, their meaning and instructions for altering them for other computer installations.

Figure F.3

"SYSOPT + Out of Kilter + QKCORE" Overlay Structure



NOTE: EITHER SEGMENT 4 OR 8 DETERMINES MAXIMUM STORAGE REQUIREMENT.

SEGMENT 1

Subrs.	Commons
MAIN(SYSOPT)	/OPTLIM/
LOC	/RCRDAT/
OPERR	/FINALS/
CMPTIM	/PDPERM/
INCORE	/KC/
UNTCOS	/KU/
UF6VAL	/KL/
ICERRS	/OOKCOM/
PVPER$	/FXDDAT/
[SYSTEMS	/ARDATA/
ROUTINES]	/PRINTS/
	/CHKSHP/

SEGMENT 2

RDOPTN
RDSTRG
REDCOR

This is subroutine not a commons

SEGMENT 3

/PDTEMP/

SEGMENT 4

RDPERS
PDCALC
SUBPLT
GWHNRG
PROBX
/PROB/

SEGMENT 5

ASMTYS
WTPERS
SETUPN
SETUPT

SEGMENT 6

Subrs.	Commons	
CONVRG	/KX/	/LC/
CALSHP	/NL/	/KA/
ARCPRT	/NN/	/IFIN/
SETELE	/NP/	/KI/
NEWMRG	/IJ/*	/KO/
SQUEEZ	/IL/*	/KQ/
FULSIM	/JL/*	/K/
CONSTS	/JI/	/SHPINF/
NXTIRR	/M/	
IRRDAT	/N/	
CSTBAT	/LER/	
PRTTOP	/KAT/	
EMPRCL	/KOR/	
MAINE	/KTER/	
PREDAT	/MINE/	
ASSEM1		
ASSEM2		

* COMMON AREAS /IJ/, /IL/, /JL/ MUST OCCUPY CONSECUTIVE CORE STORAGE (SEE SECTION G.2)

This is a commons not a subroutine

SEGMENT 7

ARCASY
MAKEJL
NODASY

SEGMENT 8

READER
TRANSL
KILTER
OUTPUT

SEGMENT 9

EDTSHP
OPTMUM

6253-91

Table F.4

SYSOPT Dataset Reference Numbers

Fortran Symbol	Meaning	Current Value	Instructions for Altering
RD	Card Reader	5	See BLOCK DATA subroutine
WT	Output Printer	6	See BLOCK DATA subroutine
SIOT	SYSINT-to-SYSOPT Output	8	Input Card 4 and any //G.FT08F001 Data Definition Cards (see Figure F.4)
NPIN	Network Program Input	9	Input Card 4 and any //G.FT09F001 Data Definition Cards (see Figure F.4)
NPOT	Network Program Output	10	Input Card 4 and any //G.FT10F001 Data Definition Cards (see Figure F.4)

F.2 SYSOPT Input Specifications

Table F.5 presents complete SYSOPT input specifications. "NEW" Card 1 signals a call to ICNPUT to read the INCORE Model data module. After the INCORE input, "STRATEGY" Card 2 heads the SYSOPT input data module (Cards 3-8). The next module read is SYSINT-to-SYSOPT output whether on disk, tape or card. A "COMPUTE" Card 9 initiates the optimization. If no other modules are to be input and/or executed, a "STOP" Card 10 terminates SYSOPT execution.

Table F.5

SYSOPT Input Specifications

Variable	Columns	Format	Description
Card 1			
...	1-12	...	"INCORE INPUT" Control card signifies following group of cards intended as input to INCORE Model
Note: Input deck for INCORE Model is inserted here.			
Card 2			
...	1-8	...	"STRATEGY" Control Card signifies SYSOPT input to follow
Card 3			
NPM	3	2X,L1	Nuclear power management strategy? (See Card 22 of SYSINT Input Specifications, Table E.5)
IDSTRG	4-10	I7	IPLACE *10 + IDSTRG of SYSINT (See Card 22 of SYSINT Input Specifications, Table E.5)
Note: These 8 alphameric characters must match membername of SYSINT-to-SYSOPT output which, likewise, must match membername on SIOT Data Definition Card (See Figure F.4).			
NRCRS	11-15	I5	Number of reactors in SYSINT strategy, ≤ 15
Card 4			
SIOT	1-5	I5	Dataset reference number for SYSINT-to-SYSOPT output, $\neq$ WT
NPIN	6-10	I5	Dataset reference number for 0-0-K Network Program input, $\neq$ RD or WT
NPOT	11-15	I5	Dataset reference number for O-O-K Network Program output $\neq$ RD

Variable	Columns	Format	Description
PARCAL	16-20	I5	Last arc type printed for all inner SYSOPT iterations (See Figure F.2), ≥ 0
PARCON	21-25	I5	Last arc type printed for converged inner iteration (See Figure F.2), ≥ 0
PARCOP	26-30	I5	Last arc type printed for accepted global optimum (See Figure F.2), ≥ 0
CORDTL	31-35	I5	INCORE detailed output desired for accepted global optimum? 0 = No 1 = Yes
OPRCOR(1) to OPRCOR(6)	36-41	6L1	INCORE print parameters for use by OPTMUM (See Card 2, QKCORE Input Specifications, Table H.6) F = No T = Yes Fortran symbol in SYSOPT — QKCORE OPRCOR(1) = RELCST OPRCOR(2) = INCCST OPRCOR(3) = BALCST OPRCOR(4) = NBLCST OPRCOR(5) = PIRDAT OPRCOR(6) = PBATCS

Card 5

Variable	Columns	Format	Description
PVRATE	1-7	F7.0	x, present value rate, fraction per year
YBASE	8-14	F7.0	Calendar time base date for present valuing, years
YSTART	15-21	F7.0	Calendar time at start of Period 1, years
PCONVG	22-28	F7.0	$100*\left|E_{rc}^{t+1} - E_{rc}^{t}\right| / \Delta \leq$ PCONVG, cycle energy convergence criteria, per cent

Variable	Columns	Format	Description
TH$CON	29-35	F7.0	$\overline{TC}^t - \overline{TC}^{t+1} <$ TH$CON, total nuclear fuel cost convergence criterion, 10^3\$
PCDELA	36-42	F7.0	γ, fraction of estimated correction applied to reactor production limits, per cent
REJLVL	43-50	F8.0	V^2_{REJ}, shape rejection criterion for $S^2 - W^2$
NPERS	51-55	I5	Z, number of periods of SYSINT strategy to be included in horizon, $\leq$ NPERS in SYSINT ≤ 100
GESFRS	56-60	I5	Initial guess option for starting optimization: =0, No guess at all (No Card 6's) =1, Use SYSINT output E_{rcp} (No Card 6's) =2, λ_{rc} entered on Card 6's =3, Estimated E_{rc} entered on Card 6's =4, Previous E^*_{rc} solution entered on Card 6's
MXITER	61-65	I5	Maximum total number of inner iterations to be allowed, ≤ 100
IAUX	66-70	I5	Total number of auxiliary arcs (Types 2 and 3 of Figure F.2) per reactor-cycle, used to form stair-step λ_{rc} curve, $3 \leq$ IAUX ≤ 19
JFRWRD	71-75	I5	Number of forward arcs (part of Type 7) per reactor per period, $2 \leq$ JFRWRD ≤ 6

Variable	Columns	Format	Description
JBKWRD	76-80	I5	Number of backward arcs (rest of Type 7) per reactor per period, For balance, JBKWRD= JFRWRD-1 is best, $1 \leq$ JBKWRD ≤ 6

Note: Total number of network arcs (See Figure F.2) cannot exceed MXARCS (=3500). Total number of network nodes cannot exceed MXNODS (=700).

Note: If GESFRS ≥ 2, there must be NRCRS of Card 6, one for each reactor.

Card 6 (if GESFRS = 2)

Variable	Columns	Format	Description
ELAME(NR,1) to ELAME (NR,NCYCIN)	1-80	20F4.0	λ_{rc}, incremental cost guess, \$/GWH $\equiv$ \$/MWH x 10^3

Card 6 (if GESFRS > 2)

Variable	Columns	Format	Description
ELAME(NR,1) to ELAME (NR,NCYCIN)	1-80	20I4	E_{rc} cycle energy guess or solution, GWH

Card 7

Variable	Columns	Format	Description
NMESH	1-5	I5	Number of different Δ energy increment (step size) to be used in approaching $TC^{*,0}$, $1 \leq$ NMESH ≤ 15
MESH(1) to MESH(NMESH)	6-80	15I5	Δ energy increment (step size), largest first, GWH

Note: There must be NRCRS of Card 8, one for each reactor.

Card 8

Variable	Columns	Format	Description
IDNO	1-4	I4	Reactor IDNO, must agree with SYSINT's IDNO for same unit.
INSTAT	5-7	I3	Initial state of unit, i.e., maintenance status during "period" immediately preceding first period of strategy =0 , did not exist =1 , down for refueling =2 , on-line

Variable	Columns	Format	Description
CYCXS	8-10	I3	Number of excess cycles included beyond horizon
GWHOLD	11-15	I5	$E_{r,C,Z+1}$ Cycle energy held over beyond horizon for split cycle, GWH
DYHOLD	16-22	F7.4	T'_{Z+1} Time remaining to next refueling beyond horizon for split cycle, years
DYDWN	...	F6.4	Downtime between excess cycles, years
DYUP	...	F6.4	Uptime for this excess cycle, years
GWHXS	...	I6	$E_{r,C+1}$ Excess cycle energy, GWH

Note: Continue until CYCXS number of excess cycles have been specified.

Note: SYSINT-to-SYSOPT output is inserted here if SIOT = RD = 5 at MIT (See Section E.1.3).

Note: "COMPUTE" Control Card 9 may be omitted to only check input of strategy or obtain present value of SYSINT cost results.

Card 9

...	1-7	...	"COMPUTE" Control Card initiates optimization

Note: Next card may be "INCORE INPUT" Card 1, "STRATEGY" Card 2 or "STOP" Card 10 with input sequence reverting to appropriate point in card sequence.

Card 10

...	1-4	...	"STOP" Control Card to terminate execution of SYSOPT for this computer run.

F.3 SYSOPT Sample Problem

Figure F.4 presents the input deck used for optimizing Strategy 2 in Case I of Chapter 5. SYSINT-to-SYSOPT output is provided on Disk.

Figure F.4

SYSOPT Sample Problem Input Deck

```
// 'DEATON','REGION=250K,CLASS=C
/*MITID  USER=(M7894,6948)
/*MAIN    LINES=20,TIME=5
/*SRT   LOW
//CALCGG EXEC FORG,PROG='USERFILE.M7894.6948.LOAD.SYSOPT(GO)'
//G.FT08F001 DD DSN=USERFILE.M7894.6948.RESULTS.SYSINT(T2045002),
//          LABEL=(,,,IN),DISP=OLD
//G.FT09F001 DD DSN=&NPIN,UNIT=SYSDA,SPACE=(80,(1000,300)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=2000)
//G.FT10F001 DD DSN=&NPOT,UNIT=SYSDA,SPACE=(133,(1000,300)),
//          DCB=(RECFM=FB,LRECL=133,BLKSIZE=3325)
//G.SYSIN DD *,DCB=(RECFM=FB,LRECL=80,BLKSIZE=2000)
NEW INCORE DATA
INCORE INPUT
    1    6    3    2 T
    TYPICAL SET OF ECONOMIC DATA  REF: MH(22.27 NOTES) & EAM (NN 2/71)
.00711     .0025      .52        .07        0.0        .500       .60        0.123
   8.00
   2.30                          0.995
  32.00
  70.00                          0.993
  35.00                          0.990
   5.60                          0.995
   7.50
  100 NK-1 1050     3     1     3  28300.   .306        6000.      500.
 0
  3.3       1.5       66.60      2.00      .3282
  3.3      11.5       42.50     15.00      .3519
  3.3      20.5       20.60     29.00      .3199
  200 NK-2 1050     1     1     3  28300.   .316         700.      500.
 1   3.4
  3.4       9.0       48.80     12.00      .3282
  3.2      19.0       24.40     25.00      .3519
  3.2      28.0        2.68     37.00      .3199
  300 NK-3 1050     3     1     3  28300.   .306        0.0001     500.
 1   3.6
  3.3      10.0       47.00     13.00      .3333
  3.3      20.0       23.00     26.00      .3333
  .9223    .000001    00.00     41.16      .3334
  400 NK-4 1050     1     1     3  28300.   .316        2000.      500.
 1   3.2
  3.2       7.0       51.20     10.00      .3282
  3.2      17.0       26.90     29.00      .3519
  3.2      27.0        5.10     35.00      .3199
  500 NK-5 1050     1     1     3  28300.   .316        0.0001     500.
 1   3.2
  2.7      .000001    72.29     30.00      .0001
  2.2      .000001    71.73     33.00      .0001
  .9223    .000001    00.00     41.16      .9998
  600 NK-6 1050     1     1     3  28300.   .316        0.0001     500.
 1   3.2
  2.7      .000001    72.29     30.00      .0001
  2.2      .000001    71.73     33.00      .0001
  .9223    .000001    00.00     41.16      .9998
REACTOR DATA FOR 1100 MWE ZICN CLASS; 3 ZONE; NO PU RECYCLE      COMPUTER VERS.
.141076      .0000218686 -.343574E-09-1.70111     -17.7720     .000106261  KANEW
5.69086      -4.58587     1.13417      -.0967594    8.52622      -17.8407    PHI

REACTOR DATA FOR 1100 MWE ZICN CLASS; 3 ZONE; NO PU RECYCLE      SLIDE RULE VERS.
.20963       .000017424                -2.8845                               KBNEW
2.570870     -1.465217    .1782607     0.0          4.34783      0.0         PHI

EXTRA .1 W/O FOR 1100 MWE ZICN CLASS; 3 ZONE; NO PU RECYCLE;ARB. BAD NEUT.ECON.
.1509        .0000218686 -.343574E-09-1.70111     -17.7720     .000106261  KBNEW
5.69086      -4.58587     1.13417      -.0967594    8.52622      -17.8407    PHI

   FUEL DATA FOR 1100 MWE ZICN CLASS; 3 ZONE; NO PU RECYCLE      COMPUTER VERS.
 .805642E 00 .195080E 00-.153501E-01-.148402E-01 .162489E-02-.257056E-03A1&A2 KB
 .265350E-03-.862559E-04 .132249E-04 1.00                                A3KBA1UR
-.189080E-02 .217089E-03-.333955E-04 .180377E-04-.724569E-05 .100522E-05A2&A3 UR
 .114056E 00-.385956E-01 .425073E-02-.128230E-02 .644474E-03-.858913E-04A1&A2 WO
 .311900E-04-.147188E-04 .185895E-05 .322341E-02 .173157E-02-.577577E-04A3WOA1PU
-.384453E-04 .463393E-04-.584225E-05 .165077E-05-.165659E-05 .252965E-06A2&A3 PU
 .112485E-02-.205317E-03 .220933E-04 .228005E 00-.703667E-01 .773171E-02UL&PL PU
.053238      .017860                                                     SIGA
   FUEL DATA FOR 1100 MWE ZICN CLASS; 3 ZONE; NO PU RECYCLE      SLIDE RULE VERS.
.955         .090                      -.00897                           A1&A2 KB
                                       1.0                               A3KB A1U
-.00137                                                                  A2&A3 UR
.0652        -.0100                    80.      F-06                     A1&A2 WO
                                       .00365       .00156               A3URA1PU
                                                                         A2&A3 PU
.00071                                 0.0872                            UL&PL PU
.0532        .0179                                                       SIGA
END OF INCORE INPUT .
STRATEGY
  T2045002    6
    8    9   10    2    2    6     1111111
  0.07  0.00   0.00   100.   2.0     25.    0.0E-03   72    4   30    7    3    2
5600750075007500750075007500750
 700750075007500750075007500750
6600750075007500750075007500750
2000750075007500750075007500750
9000750075007500750075007500750
8500750075007500750075007500750
    2  100   20
 100  2  2 5800  .033  .167  1.00   7000  .167  1.00  7000
 200  2  2 1160  .167  .167  1.00   7000  .167  1.00  7000
 300  1  2    0  .000  .167  1.00   7000  .167  1.00  7000
 400  2  2 1740  .250  .167  1.00   7000  .167  1.00  7000
 500  0  2 4000  .667  .167  1.00   7000  .167  1.00  7000
 600  0  2  560  .083  .167  1.00   7000  .167  1.00  7000
COMPUTE
STOP
/*
```

F.4 SYSOPT Source Listing

The following is a Fortran IV source listing of the SYSOPT code (included only in MIT library copies).

APPENDIX G

Out of Kilter Network Program

G.1 Out of Kilter Discussion

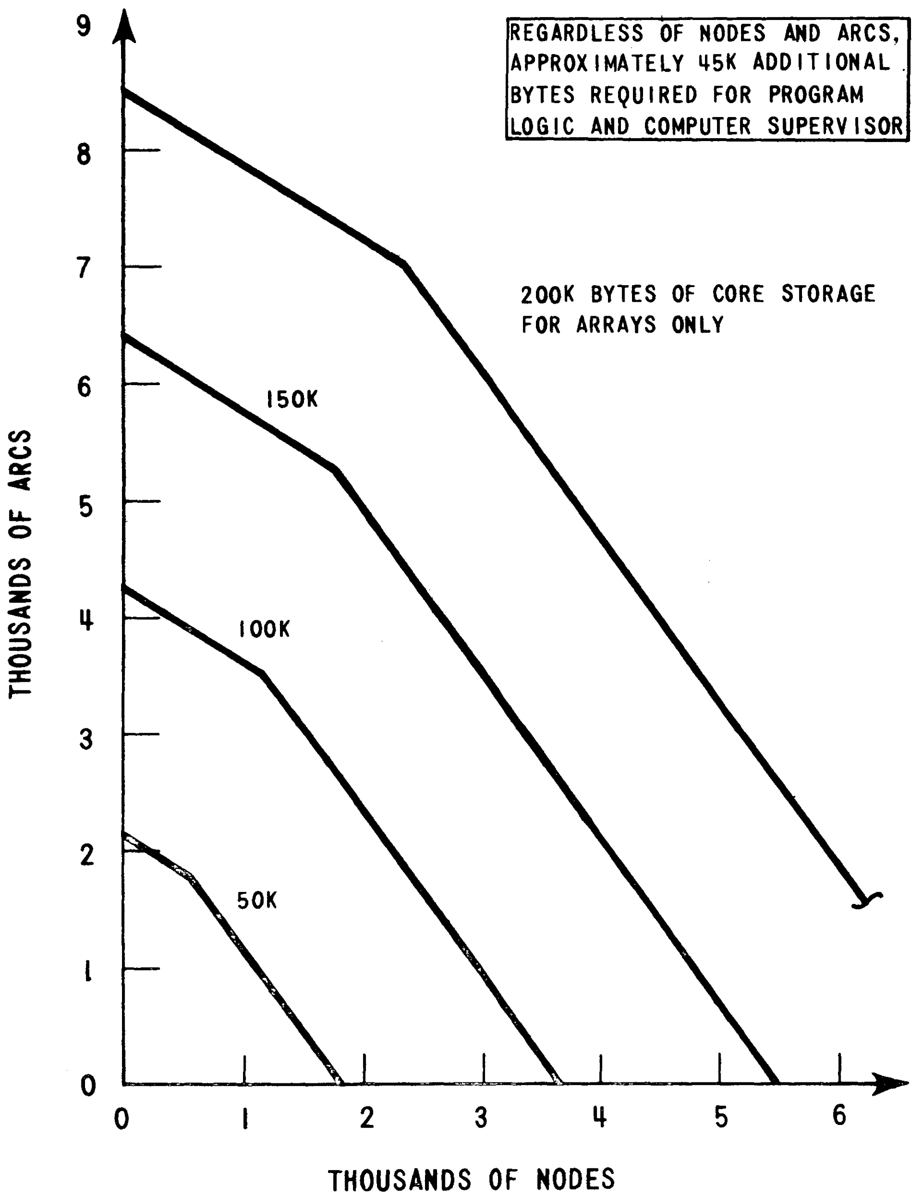
The complete Out of Kilter Network Program was graciously provided to the author by the Flight Transportation Laboratory at MIT. Only minor modifications were made to the program to facilitate on-line merging with SYSOPT. These modifications are transparent to any user interested only in the Out of Kilter Program itself, i.e., for solving network programming problems in other contexts. Figure G.1 is provided as a guide to the computer storage requirements necessary to run the progam for various size problems (see Sub-section 13 of Section G.2).

Because of the program's generality, the original input manual (45) is included here with only minor editorial revisions.

Figure G.1

6253-98

Core Storage Requirements for Out-of-Kilter Network Program



G.2 Out of Kilter Input Specifications

IBM /360 OUT OF KILTER NETWORK FLOW ROUTINE

DESCRIPTION FOR THE USER

Table of Contents

Section

1. Introductory Notes

This writeup is intended for the user of the "Out of Kilter" program which has been written for the IBM system 360 model 65. The program has been successfully run at the MIT Computation Center.

Both the program and the writeup are based on the SHARE routine RS OKF1 and its corresponding writeup.

The FORTRAN subprograms are written in FORTRAN IV (G level). The assembly language subprograms use the extended mnemonic branching instruction codes and the macros SAVE and RETURN.

The program and this writeup were prepared by Amos Levin, Flight Transportation Laboratory, MIT, August 1967.

2. Formulation

A Computer routine for the solution of "network flow" programs -- problems of finding those flows of an homogeneous commodity through a capacitated network minimizing the sum of the linear costs of flow through each arc -- is herein described. The computational algorithm employed is described in the book "Flows in Networks", L.R. Ford and D.R. Fulkerson, Princeton University Press, 1962, pp.162-169.

The network in question consists of nodes designated by i or j, and a certain collection of arcs joining pairs of nodes. The arc $\widehat{ij}$ is thought of as directed from i to j. With each arc in the network is associated the following four integer quantities.

c_{ij} the cost of one unit of flow from i to j along arc $\widehat{ij}$;

u_{ij} the upper bound on the amount of flow along the arc $\widehat{ij}$;

ℓ_{ij} the lower bound on the amount of flow along the arc $\widehat{ij}$;

x_{ij} the quantity of flow along the arc $\widehat{ij}$

The network flow problem is that of determining x_{ij} (for all arcs $\widehat{ij}$ of the network) such that

(1) $\ell_{ij} \leq x_{ij} \leq u_{ij}$ (all arcs $\widehat{ij}$),

(2) the net flow into any node (generally zero) remains fixed throughout the solution of the problem, and

(3) $\sum_{\widehat{ij}} c_{ij}x_{ij}$ is minimized

3. Data

Data Format

A node may be represented by any combination of six Hollerith characters (at least one of which is neither zero nor blank); i and j below are such combinations. (Note that for node names a blank is a character, and different from a zero.) The numerical data above are represented as right-justified integers in the appropriate fields. All data pertaining to one arc are entered on one card as follows:

1..6	7..12	13..18	19,20	21..30	31..40	41..50	51..60	61..80
blank	i	j	free to use	c_{ij}	u_{ij}	ℓ_{ij}	x_{ij}	free to use

Leading zeros in the numeric fields need not be entered, nor need any figures where zero is desired.

Of course, fields 7-50 contain constants for the stated problem. Entry of the "x_{ij}" is optional, constituting only an initial guess at the solution.

An optional initial set of node prices π_i may be entered. These are entered one per card as follows:

1 .. 6	7 .. 12	13 ... 20	21 ... 30	31 80
blank	i	free to use	π_i	free to use

Assembly of Data

The data just described is put together in the following way:

1) All arcs $\widehat{ij}$ having a given first node i must be adjacent in the deck. (No other requirement on their order is made.)

2) The arc cards are preceded by two cards, the first being the title card and the second bearing the word "ARCS" in the field 1-4. The title card should be blank in column 1 and may have any Hollerith punches in columns 2-80.

3) If no node prices are given, the arc cards are followed by a card bearing "END" in 1-3.

4) If node prices are given, the arcs are followed by a card bearing "NODES" in 1-5; the node cards follow this, and all the cards are followed by the END card of (3).

4. Control Cards for Standard Run

Input, computation, and output are effected by control cards whose punching in the field 1-12 controls the operation of the routine. Punching always begins in column 1, and there is one blank between English words. The first card of the deck which follows the program deck must be the control card

READY

Following the "READY" card must be one of the two control cards

CARDS or TAPE

If "TAPE", the assembled data described in the previous section should be on the reserved input tape. If "CARDS", the assembled data should immediately follow this control card.

Next may be placed any combination of the three output control cards

OUTPUT TAPE

OUTPUT PRINTER

OUTPUT PUNCH

which will cause the types of output described in Section 8. At least one OUTPUT control card must be included in the data set.

Next is placed the card

COMPUTE

which causes computation to begin.

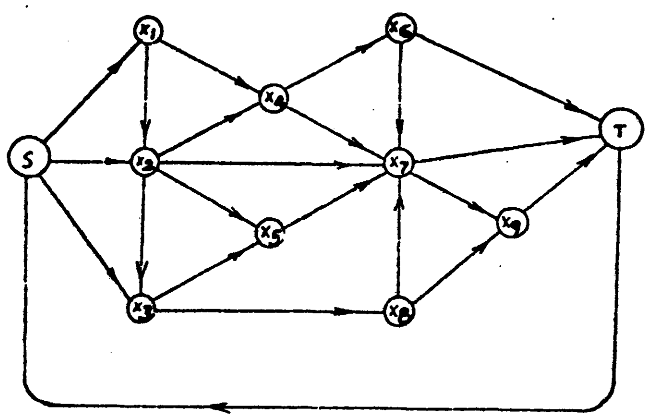
The last card in the deck must be the control card

PAUSE

which terminates the job.

5. Example

The example which follows is a modification of the one given in the book " Flows In Networks", L.R. Ford and D.R. Fulkerson, Princeton University Press, 1962, pp.123-127. Costs and bounds for the arcs can be found in the data listing on the next page. Since the cost on the arc $\widehat{TS}$ is very low (negative) compared to the costs on the other arcs, the routine finds the maximal flow that minimizes costs from S to T.



```
READY
CARDS
   F. AND F. EXAMPLE 1
ARCS
        S       X1            3        50       35        0
        S       X2            6        30        0        0
        S       X3            8        15        0        0
        X1      X2            2        50        0        0
        X1      X4            2        25        0        0
        X2      X3            2        15        0        0
        X2      X4            1        45        0        0
        X2      X5            3        10       10        0
        X2      X7            8        15        0        0
        X3      X5            1        10        0        0
        X3      X8            3        20        0        0
        X4      X6            9        90        0        0
        X4      X7            8        10        0        0
        X5      X7            5        60        0        0
        X6      X7            1        10        7        0
        X6      T             2        10        0        0
        X7      X9            1        10        0        0
        X7      T             4        80        0        0
        X8      X7            2        20        0        0
        X8      X9            3        10        0        0
        X9      T             3        10        0        0
        T       S        -10000        85       25        0
END
OUTPUT PRINTER
OUTPUT TAPE
COMPUTE
PAUSE
```

6. Jobs with More than One Run.

The control card setup described in Section 3 applies to jobs with only one run. By a "job", we mean all that is done in one pass at the computer; that is, any work that can be done without manual interference with the computer and, in addition, without inputting the program instructions into the computer more than once. By a "run", we mean that which is involved in the solution of one problem.

For multiple run jobs, the standard input for each run is as described in Section 3 with the "PAUSE" card removed. Runs may be stacked one after another. Only one "PAUSE" card may be used, and it is always placed after the "COMPUTE" card of the final run.

Each run begins with a "READY" card or a "SAVE" card as described in Section 6. Each run ends with a "COMPUTE" card. The job ends with a "PAUSE" card.

7. Save and Alter Run

In Section 3, the standard run beginning with the " READY" card was described. In Section 5, it was noted that these runs may be stacked, one after another. Frequently it is desired to execute a run in which only relatively few c_{ij}, u_{ij} or ℓ_{ij} are changed, but in which the arc configuration remains the same. In this event, a "Save and Alter" procedure may be followed. A "Save and Alter" run may be any run except the first. The control card setup for this type of run is as follows.

The first card of the run must be the control card

SAVE

which initiates a new run without destroying the results of the previous run.

The second card is the title card, which may have any Hollerith punches in it, except that column 1 should be blank.

Next are placed the "OUTPUT" cards as mentioned in Section 3 and described in Section 8.

Next are placed any number of "ALTER" cards. Each "ALTER" card has the following format:

1...6	7..12	13..18	19,20	21..30	31..40	41..50	51..60	61..80
ALTER	i	j	n_{ij}	c_{ij}	u_{ij}	ℓ_{ij}	Δf_{ij}	free to use

i and j are the source and sink nodes of an arc which is in core storage; that is, one which was used on the preceding "READY" run. n_{ij} may be left blank if there is only one arc $\widehat{ij}$. If there is more than one arc $\widehat{ij}$, then n_{ij} gives the number of this arc as to whether it was the 1st, 2nd, 3rd, etc. arc $\widehat{ij}$ which was read into memory in the applicable "READY" run. c_{ij}, u_{ij} and ℓ_{ij} are the new values of these same quantities for this arc.

Δf_{ij} is usually zero (or blank). It is the change in the flow out of node i and into node j. Note that inputting a new x_{ij} is meaningless, since x_{ij} on input is a guess, and guessing a value of x_{ij} on an alter run would only upset the conservation of flow from the nodes. Hence inputting a non-zero Δf_{ij} is a means of deliberately upsetting the flow conservation. It will change x_{ij} to $x_{ij} + \Delta f_{ij}$.

The last card of the run must be the control card

COMPUTE

which causes computation to begin.

Note that any number of "Save and Alter" runs may follow one "READY" run. The effects of each "Save and Alter" are cumulative.

The program also allows "ALTER" cards to be placed after the "OUTPUT" cards and before the "COMPUTE" card on a "READY" run. This "Ready and Alter" run is useful when data is on tape and a few changes in the value of c, u, and ℓ are needed before the run is to be executed.

8. Other Program Options

In the standard run, the program requires that every node be a first node for some arc and be a second node for some other arc. This is the standard network problem. Another type of problem allows arcs to end at nodes at which no arcs begin. These sinks are designated by the program as "dead end arcs." There may also be source nodes at which no arc terminates. This type of problem is designated a "transportation" problem and the requirement that at least one arc begin at each node and end at each node is ignored by the program.

The reserved input tape may have data for several jobs stacked on it. There are no ends of file on this tape except at the end of all data; the program knows when it is at the end of the data for one run by sensing the "END" card record. In certain cases, it may be desirable to pass over some data packages while processing a job. In this event, the control card "SKIP" is used.

The general "READY" type run is now described.

The first card must be the control card

READY

An optional card which must follow the "READY" card if this is a transportation problem, is the control card

TRANSPORTATION

Also optional is the control card

SKIP

which is used to cause the <u>reserved input tape</u> to skip one package of assembled data. As many "SKIP" cards are used as are needed to skip the desired number of packages of assembled data. The "SKIP" cards and the "TRANSPORTATION" card may be in any order immediately following the "READY" card.

Following the above cards must be one of the two control cards

CARDS or TAPE

These cards are as described in Section 3.

The data package follows the "CARDS" control card. Following the data package, or the control card "TAPE" where there is no data package with the control cards, may be an optional title card. If this is included, it supersedes the title card on the data package.

Next may be placed any number of "OUTPUT" cards as described in Section 8.

Next may be placed any number of "ALTER" cards as described in Section 6.

The last card in the run must be the control card

COMPUTE

which causes computation to begin.

9. Output

The type of output is controlled by one or more of four control cards. The four control cards are

a) OUTPUT PRINTER
b) OUTPUT TAPE
c) OUTPUT PUNCH
d) OUTPUT NODES

The "OUTPUT PRINTER" control card causes output to be written on the system output device.This output is written for printing on the peripheral printer under program control. The system output device is denoted in the program by the symbol "KO", and KO has the value 6 in the version of the program submitted. The data for each arc are printed horizontally on the page. The data for one arc, $\widehat{ij}$, are printed in the following order:

1) node name i
2) node name j
3) c_{ij}, the unit cost of arc $\widehat{ij}$
4) u_{ij}, the upper bound of the quantity of flow through arc $\widehat{ij}$
5) ℓ_{ij}, the lower bound of the qunatity of flow through arc $\widehat{ij}$
6) x_{ij}, the quantity of flow in the arc $\widehat{ij}$
7) "FLOW" $= c_{ij}\, x_{ij}$, the total cost of x_{ij} units at the cost c_{ij}
8) π_i, the node price of node i
9) π_j, the node price of node j
10) $\bar{c}_{ij}$, the quantity $\pi_i + c_{ij} - \pi_j$.
11) The letter "K", the letter "N" or nothing. The letter "K" is printed if all the arcs are in kilter. The letter "N" is printed if this arc could not be brought into kilter,

indicating that the problem has
no feasible solution. Nothing is
printed in all other cases.

The "OUTPUT TAPE" control card causes output to be written on the reserved output tape. This output may be printed peripherally using single space (or double space) control. It may also be punched peripherally, and the cards gotten thereby will be substantially the same as the cards gotten from the "OUTPUT PUNCH" option described below. The information from the "OUTPUT TAPE" option is the same as that from the "OUTPUT PRINTER" option, except that items 8), 9), and 10), are not output. This output is compatible with the input "TAPE" option.

The "OUTPUT PUNCH" option gives items 1) through 7) on the on-line punch. This option is generally very time consuming except on short problems.

Any of the above three options may be used in combination on any one problem. At least one OUTPUT control card must be included in each data set.

The "OUTPUT NODES" option will output a list of node prices in addition to the arc information on the tape or punch options. This option will have no effect on the printer output option.

All of the output on the reserved output tape and on the punch is compatible with the input to the problem. The "OUTPUT PRINTER" output is not compatible with the input.

In addition to the above, all control card information is written on the peripheral printer device, with the exeption of the

"COMPUTE" control card for which is substituted a count of the arcs and the nodes. The messages in Section 9 are all written on the system output device also.

On the following two pages are shown the "OUTPUT PRINTER" results of the example given in Section 4. "Flow" is $c_{ij}\ x_{ij}$. "Total system contribution" is the optimal value of the objective function $\sum c_{ij}\ x_{ij}$. Note that the first page contains information that would be on the system output device regardless of whether "OUTPUT PRINTER" is requested.

```
READY
CARDS
   F. AND F. EXAMPLE 1
ARCS
OUTPUT PRINTER
OUTPUT TAPE
NO OF ARCS=    22 NO OF NODES=    11
THIS RUN OUTPUT TO TAPE
```

F. AND F. EXAMPLE 1

ARCS		COST	UPPER	LOWER	X	FLOW	PI1	PI2	CBAR	
S	X1	3	50	35	50	150	13	17	-1	K
S	X2	6	30	0	20	120	13	19	0	K
S	X3	8	15	0	15	120	13	24	-3	K
X1	X2	2	50	0	25	50	17	19	0	K
X1	X4	2	25	0	25	50	17	20	-1	K
X2	X3	2	15	0	15	30	19	24	-3	K
X2	X4	1	45	0	5	5	19	20	0	K
X2	X5	3	10	10	10	30	19	25	-3	K
X2	X7	8	15	0	15	120	19	30	-3	K
X3	X5	1	10	0	10	10	24	25	0	K
X3	X8	3	20	0	20	60	24	28	-1	K
X4	X6	9	90	0	20	180	20	29	0	K
X4	X7	8	10	0	10	80	20	30	-2	K
X5	X7	5	60	0	20	100	25	30	0	K
X6	X7	1	10	7	10	10	29	30	0	K
X6	T	2	10	0	10	20	29	34	-3	K
X7	X9	1	10	0	0	0	30	31	0	K
X7	T	4	80	0	75	300	30	34	0	K
X8	X7	2	20	0	20	40	28	30	0	K
X8	X9	3	10	0	0	0	28	31	0	K
X9	T	3	10	0	0	0	31	34	0	K
T	S	-10000	85	25	85	-850000	34	13	-9979	K

END

TOTAL SYSTEM CONTRIBUTION = -848525

NO OF BREAKTHRUS= 12, NO OF NONBREAKTHRUS= 11, NO OF X CHANGES= 67
NO OF NODES FROM WHICH LABELING WAS DONE= 131

PAUSE

RESERVED TAPE HAS BEEN WRITTEN

10. Program Messages

One exception to the previous formats is permitted. If the "READY" of "SAVE" card is not the first card in a run this is not considered to be an error, but it is assumed that these are comment cards. The contents of columns 7-72 of all cards in a run (if any) which precede the "READY" or "SAVE" card plus columns 7-72 of the "READY" or "SAVE" card itself are written on the system output device.Thus only columns 1-6 of the "READY" and the "SAVE" card are fixed in format, the rest of the card may be used for comments. The above is also applicable to the "PAUSE" card.

Below is given a list of comments which may be written on the system output device.

Comments 3),4), 5), 6), 7), 8), 9), 12), and 13), denote errors in data set-up that were caught by the pre-processing routines. Conditions 10) and 11) are considered to be errors only if no "TRANSPORTATION" control card was present. Whenever any of the above error conditions are present, the run is terminated.

Comment 18) is given to convey information but is not regarded as an error.

Comment 17) denotes a trivial infeasibility--in this case the algorithm is not executed.

Comment 2) is written if the algorithm computation was started but not finished. Comment 1) will be present when comment 2) is written.

OFF LINE PROGRAM COMMENTS

1)	OVERFLOW IN NODE PRICES	A node price is greater than 100,000,000. Costs should be rescaled to run job.
2)	RUN TERMINATED AT ARC___ ___	Gives the arc at which run was terminated due to the reason stated above the comment.
3)	RUN TERMINATED DUE TO ERRORS IN THE DATA	Self - explanatory
4)	TOO MANY NODES IN THIS RUN	
5)	TOO MANY ARCS IN THIS RUN	
6)	CARD PUNCHING ERROR IN ARC CARD NO. ______	These comments are self-explanatory
7)	CARD PUNCHING ERROR IN NODE CARD NO. _______	
8)	THE ARC IN THE ABOVE ALTER CARD IS NOT IN CORE	
9)	SOURCE NODES ARE NOT ADJACENT, ARC ___ ____	All arcs having similar first nodes must be adjacent. This comment gives an arc which is separated from another arc having the same first node.
10)	ARC ___ ___ IS A DEAD END ARC	The second node of this arc does not appear anywhere as a first node.
11)	NO ARC ENDS AT NODE___ ___	Self-explanatory
12)	CARD ___NODE___NOT IN ARCS	A node card appears on which the node is not represented in any arc.
13)	ILLEGAL CONTROL CARD(___ ___ ___)	The control card just read into core is not able to be interpreted by the program.

14)	OUTPUT CONTROL CARD MISSING OR OUR OF SEQUENCE	Self - explanatory
15)	RESERVED TAPE HAS BEEN WRITTEN	This comment states whether an output has been written on a tape other than the system device (as requested by an " OUTPUT TAPE" control card).
16)	NO RESERVED TAPE HAS BEEN WRITTEN	
17)	ARC ___ HAS LOWER BOUND GREATER THAN UPPER BOUND	Self-explanatory
18)	NODE ___ NON-CONSERVATIVE, NET FLOW=____	Node has a finite net flow. Negative flow denotes source node.
19)	THIS RUN OUTPUT TO TAPE	These comments state where the output to this run may be found.
20)	THIS RUN OUTPUT PUNCH	
21)	______ARCS ARE OUT OF KILTER	This run was completed, but there is no feasible solution. As many as 100 arcs are marked with an "N" on the output. "N" denotes that these arcs are not in kilter .

11. Program Operation Notes

The I/O device reference numbers the program uses are given below. Since these numbers vary from installation to installation, they can be changed as indicated in Section 13.

I/O Device	Fortram Symbol	Reference Numbers
System input device-- all control cards and data packages of the "CARDS" variety	KI	5
System output device-- general editing output and "OUTPUT PRINTER" option	KO	6
Card punch-- "OUTPUT CARD" output	KQ(1)	7
Reserved output tape-- "OUTPUT TAPE" output	KQ(2)	3
Reserved input tape-- data packages of "TAPE" variety	KQ(3)	2

System control cards must be included in the deck whenever the reserved tapes are used. The reference numbers 2 and 3 for the reserved input and output tapes, respectively, were arbitrarily chosen. These numbers can be changed, but they must correspond to the tape numbers specified on the system control cards.

For a reserved output tape the following two control cards must be included:

```
//G.FT03F001 DD UNIT=TAPE9,LABEL=(1,NL),                              X
//     VOLUME=SER=tapeid,DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000)
```

These cards should immediately precede the data. When the job is run under the ASP system (at the MIT Computation Center), the following card must also be included:

```
/*SETUP DDNAME=FT03F001,DEVICE=2400-9,ID=(tapeid,RING,SAVE,NL)
```

This card should immediately follow the job card. Note that "tapeid" is an identification number assigned to the tape by the MIT Computation Center. Three similar control cards must be included whenever a reserved input tape is used, but FT03 should be changed to FT02. The OS/360 user's manual contains more details concerning the use of reserved tapes.

The sequence of operations by the computer when it is doing one problem is as follows:

First the "READY" card is looked for.

Next the data package is read.

Next comes the generation of the output. When outputting is finished, the next run (if any) will be started.

The running time for this program, of course, varies considerably from problem to problem. The input and output time will be roughly proportional to the number of arcs. The execution of the algorithm is the most variable part of the problem, and its duration will depend on the type of problem considered. At the end of "PRINTER" output, the number of non-breakthroughs that were obtained are written. Also it writes the "number of X changes," which is the sum of the number of arcs in each breakthrough chain, and "number of nodes from which labeling was done," which is the sum of the number of nodes scanned on each labeling operation.

As an example, a problem was run that gave the following statistics:

Number of arcs	414
Number of nodes	348
Number of breakthroughs	40
Number of non-breakthroughs	179
Number of X-changes	1915
Number of nodes from which labeling was done	7550

The upper bounds on the elapsed times were:

Program compilation	2.2 min.
Data preprocessing	3.3 sec.
Algorithm computations	3.6 sec.

12. Structure of the Program

A. Main Program

1) Sets up I/O device numbers and dimensions

2) Calls MAINE

B. Subroutine MAINE (with ENTRY OOKMAN for on-line linking and execution)

1) Calls the preprocessing routines

PREDAT
ARCASY
MAKEJL
NODASY
READER
TRANSL

2) Calls the subroutine KILTER once for each arc.

3) Calls the postprocessing routine OUTPUT.

The routine also processes certain error and infeasibility conditions.

C. Subroutine PREDAT looks for a control card of the type "READY", "SAVE", or "PAUSE". If it finds a "READY" card, core is cleared and it looks for a control card of the type "CARDS", "TAPE", "SKIP", or "TRANSPORTATION". After it finds a "CARD" or "TAPE" control card, it then looks for the control card "ARCS" on the appropriate input device.

If a "SAVE" card is found the program returns control to the main program and control is passed next to the subroutine READER.

If a "PAUSE" card is found, the end-of-job instructions are executed.

D. Subroutine ARCASY reads arc record after arc record into storage until it comes to a record with "END" or one with "NODES"

The ℓ_{ij}, u_{ij}, c_{ij}, and x_{ij} information is stored in the KL, KU, KC, and KX blocks, respectively. The BCD names of the first nodes are stored in NN, and the BCD names of the second nodes are stored in IJ.

E. Subroutine MAKEJL sets up lists in IL and JL storage. These lists are cumulative counts of the arcs beginning and ending at the nodes. The subroutine also replaces the IJ names by numbers.

F. Subroutine NODASY reads in the node prices, if any.

G. Subroutine READER reads the OUTPUT, ALTER, and COMPUTE control cards.

H. Subroutine TRANSL performs the final operations before going to the Out of Kilter algorithm.

I. Subroutine KILTER tests the arc presented to see if it is in kilter. If it is not in kilter, the assembly language subroutine LABELN is called. Depending on a flag set in LABELN, the KILTER subroutine then calls either UPNOPR or BREAKT. When the arc has been brought into kilter or when it is determined that the arc cannot be brought into kilter, the control passes back to MAINE.

J. Subroutine OUTPUT generates the output required for the run.

K. ASSEM1 routine includes:

1) Assembly language subroutine LABELN performs the labeling operation. If a breakthrough results, the next subroutine called by KILTER will be BREAKT. If a non-breakthrough results, the next subroutine called by KILTER will be UPNOPR.

2) Assembly language subroutine BREAKT alters the quantities of flow in the cycle generated by LABELN.

3) Assembly language subroutine UPNOPR raises the node prices of the labeled nodes by the appropriate amount.

4) Assembly language function NODENO returns the number of the node that has the name presented.

L. ASSEM2 routine includes:

1) Assembly language function LADDR returns the rightmost 16 bits of the word presented as a 32-bits FORTRAN integer.

2) Assembly language function LDECR returns the leftmost 16 bits of the word presented as a 32-bits FORTRAN integer.

3) Assembly language subroutine PLACE stores the rightmost 16 bits of the first full-word argument in the leftmost 16 bits of the second full-word argument.

13. Compiling the Program

In order to change the I/O device numbers of the program, only the MAIN program need be compiled. The I/O device numbers are the first items to be defined by the program. The symbols assigned to the devices are as follows:

KI	=	System input device
KO	=	System output device
KQ(1)	=	Punch card device
KQ(2)	=	Reserved output tape
KQ(3)	=	Reserved input tape

In order to change the dimensions of the program, it is necessary to change the dimensions of all the FORTRAN subprograms and also the numeric values of the symbols KQ(4) and KQ(5). The assembly language subprograms need not be changed since they do not contain dimensions information.

Let "a" be the maximum number of arcs allowed in the program and "n" the maximum number of nodes allowed. Then, KQ(4) = a, and KQ(5) = n in the main routine. The storage which must be allocated for each symbol is as follows:

	SYMBOL	DIMENSION
	KL	a
	KC	a
	KU	a
	KX	a
	NL	n
	NN	2n
	NP	n
Must occupy at least "a" words of consecutive storage	IJ	n
	IL	n + 1
	JL	maximum $\begin{pmatrix} n+1 \\ a-2n-1 \end{pmatrix}$
	JI	a

Total storage for above symbols = 5a + 4n + max (a,3n+2)

A total of 108,000 four-byte words were available for dimensions when the program was tested on the IBM 360 model 65 computer. One can choose a and n to be any positive integers as long as

$$5a + 4n + \max(a, 3n+2) \leq \text{full-word storage available for dimensions.}$$

G.3 Out of Kilter Sample Problem

Sub-section 5 of Section G.2 contains a sample problem input listing.

G.4 Out of Kilter Source Listing

The following is a source listing of the Out of Kilter Network Program (included only in MIT library copies).

APPENDIX H[1]

Q K C O R E

H.1 QKCORE Discussion

H.1.1 Introduction

As was pointed out in Section 5.2, development of QKCORE, a QuicK in-CORE empirical fuel cost simulator (See Figure H.1) was undertaken to allow completion and evaluation of the nuclear power management model of Figure 2.21. To provide maximum flexibility, QKCORE is programmed as a separate "stand-alone" code suitable for independent fuel management studies.

A pseudo-1D nodal model of LWR reactor core physics is used (See Section H.1.2). Each cycle of a multi-cycle planning horizon may operate in one of three modes:

(1) With reload (i.e., freshly fabricated) enrichment ϵ_f specified, irradiate to reactivity-limited cycle energy E_{rc}. This mode is representative of normal fuel-depletion code operation.

(2) With cycle energy E_{rc} specified, determine reload enrichment ϵ_f required at start of cycle to generate reactivity-limited E_{rc}. This mode is required by SYSOPT.

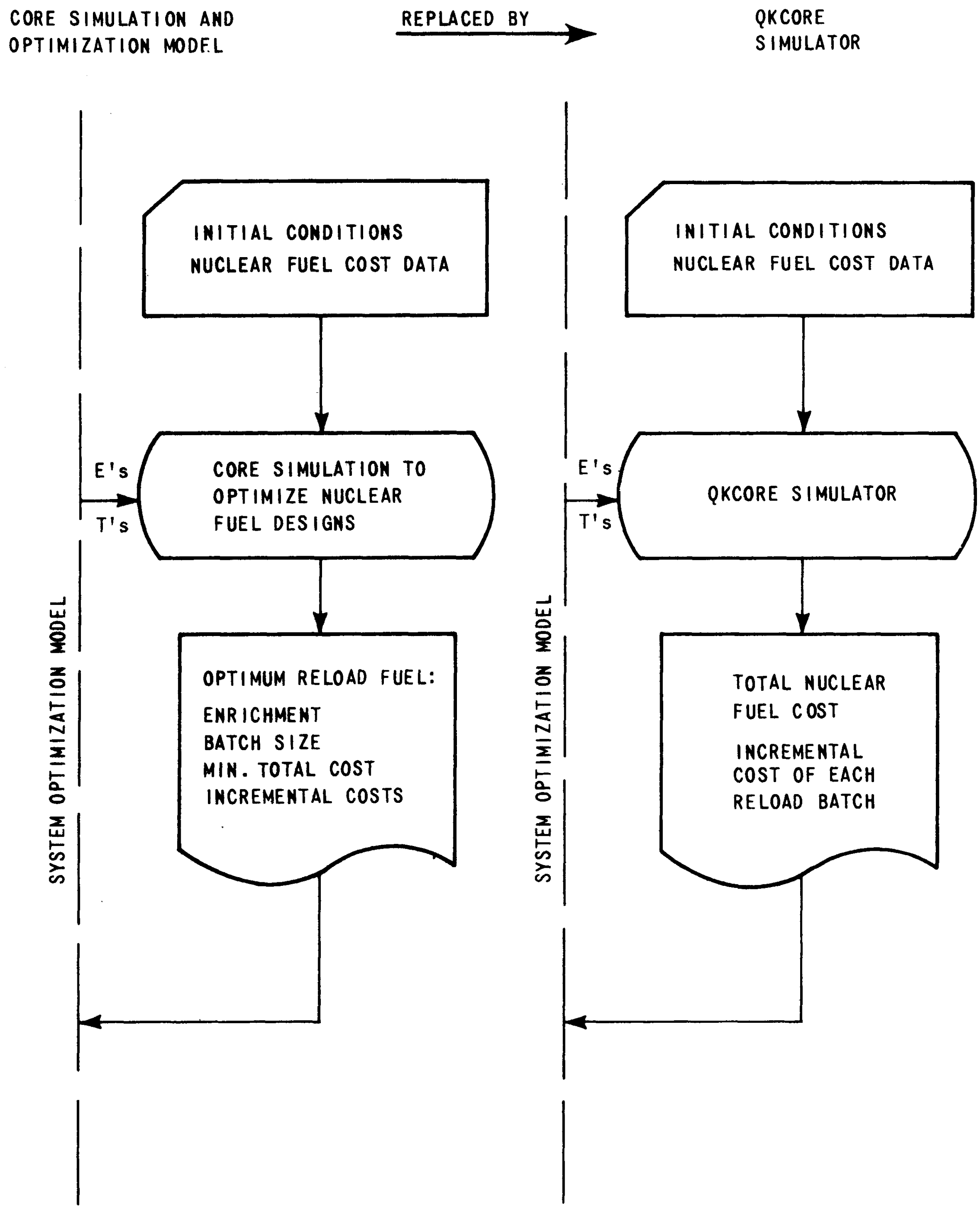
---

[1]Notation in this Appendix is defined specifically in context rather than in Nomenclature of Appendix I.

Figure H.1

6253-92

Compatibility of the Fuel Cost Simulator



(3) If both ϵ_f and E_{rc} specified, determine amount of early shutdown or stretchout required. This model represents a compromise where first few cycles of horizon have enrichment fixed and specific cycle energy required.

Total and incremental fuel costs for each cycle are determined on-line as indicated in Section H.1.2.

The limitations of the code are as follows:

(1) modified-scatter refueling with fixed number of zones ($1 \leq$ NOZONE ≤ 10),

(2) no plutonium recycle,

(3) up to 20 cycles considered,

(4) up to 15 different sets of nuclear generating unit characteristics may be retained simultaneously,

(5) each nuclear unit may have a different set of empirical core physics constants,

(6) up to 5 different sets of empirical fuel constants and

(7) the cost of each operation in the nuclear fuel cycle may be escalated using an input quadratic equation.

H.1.2 Computational Model

The computational model is based on (1) empirical fuel equations (See Table H.1) which represent homogenized unit fuel cell data as a function of fabricated

Table H.1

QKCORE Empirical Fuel Simulator Equations

I. $k_\infty = K8 = (F_1 + F_2\varepsilon_f + F_3\varepsilon_f^2) + (F_4 + F_5\varepsilon_f + F_6\varepsilon_f^2)\ B + (F_7 + F_8\varepsilon_f + F_9\varepsilon_f^2)\ B^2$

II. $KGU = (F_{10} + F_{11}\varepsilon_f + F_{12}\varepsilon_f^2) + (F_{13} + F_{14}\varepsilon_f + F_{15}\varepsilon_f^2)\ B + (F_{16} + F_{17}\varepsilon_f + F_{18}\varepsilon_f^2)\ B^2$

III. $\varepsilon = ENRICH = \varepsilon_f \cdot e^{-\alpha_1 B}$

where

$\alpha_1 = (F_{19} + F_{20}\varepsilon_f + F_{21}\varepsilon_f^2) + (F_{22} + F_{23}\varepsilon_f + F_{24}\varepsilon_f^2)\ B + (F_{25} + F_{26}\varepsilon_f + F_{27}\varepsilon_f^2)\ B^2$

IV. $KGPU = \alpha_2\ (e^{-\alpha_3 B} - e^{-\alpha_4 B})$

where

$\alpha_2 = (F_{28} + F_{29}\varepsilon_f + F_{30}\varepsilon_f^2) + (F_{31} + F_{32}\varepsilon_f + F_{33}\varepsilon_f^2)\ B + (F_{34} + F_{35}\varepsilon_f + F_{36}\varepsilon_f^2)\ B^2$

$\alpha_3 = F_{37} + F_{38}\varepsilon_f + F_{39}\varepsilon_f^2$

$\alpha_4 = F_{40} + F_{41}\varepsilon_f + F_{42}\varepsilon_f^2$

Table H.1--Continued

V. Σ_a = SIGA = $F_{43} + F_{44}\varepsilon_f$

Units:

F_i = FULCON(I)

ε_f = as-fabricated enrichment, w/o U-235

ε = current (i.e., at burnup B) enrichment, w/o U-235

B = average zone burnup, MWD/kg

KGU = uranium inventory, kg U/kg U fab.

KGPU = fissile plutonium inventory, kg fissile Pu/kg U fab

Σ_a = macroscopic absorption cross section, cm^{-1}

enrichment ϵ_f and current burnup B and (2) empirical reactor equations (See Table H.2) which mockup zone-by-zone irradiation during each cycle.

To facilitate explanation of the model, assume that all the required coefficients in Tables H.1 and H.2 are known a priori. In the first operating mode (See Section H.1.1), the purpose of the model is to answer the following question:

Given the as-fabricated enrichments ϵ_{f_i} and average zone burnups B_i for non-fresh fuel (i=2 to n) in an n-zone core, what must be the fresh fuel (i.e., B_1=0), enrichment ϵ_{f_1} loaded to give a cycle electrical energy production of E_c?

First, the electrical energy production E_c must be converted to thermal energy θ_c. Using a previous assumption (See Section 2.4.2) of constant nuclear incremental efficiency η_{inc}, Equation (2.52) yields

$$\theta_c = H° \; T_{op} + \frac{E_c}{\eta_{inc}} \qquad \text{(H.1)}$$

where

$H°$ = fixed heat consumption rate during operation

T_{op} = time of operation

The next step is the determination of $k_{\infty_{INNER}}$ as an index of the reactivity remaining in the core. Assuming three-zone modified-scatter refueling,

Table H.2

QKCORE Reactor Irradiation Empirical Equations

VI. $$k_{\infty_{NEW}} = K8NEW = 1 + R_1 + R_2\theta c + R_3\theta c^2 + (R_4 + R_5\ \delta k_{INNER} + R_6\theta c)\ \delta k_{INNER}$$

where $\delta k_{INNER} = k_{\infty INNER} - 1$

$$k_{\infty INNER} = \frac{\sum_{i=2}^{n} k_{\infty}\ (\varepsilon_{f_i}\ ,\ B_i)}{n-1}$$

VII. $$\Phi = \frac{1}{1 + R_7 + R_8\varepsilon_f + R_9\varepsilon_f^{\ 2} + R_{10}\varepsilon_f^{\ 3} + R_{11}\ \delta k_{INNER} + R_{12}\ \delta k^2_{\ INNER}}$$

Units:

R_i = RCRCON(I)

θ_c = Cycle thermal energy, GWHt

n = n-zone core (NOZONE)

ε_f = w/o U-235 as-fabricated

$$k_{\infty INNER} = \frac{k_\infty(\epsilon_{f_2}, B_2) + k_\infty(\epsilon_{f_3}, B_3)}{2} \qquad (H.2)$$

Using this index and θ_c the required energy production, Equation VI of Figure H.2 gives the fresh fuel k_∞ needed,

$$k_{\infty NEW} = k_{\infty NEW}(\theta_c, k_{\infty INNER}) \qquad (H.3)$$

The fresh (B_1=0) fuel enrichment is then determined by applying the quadratic equation to

$$k_{\infty NEW} = k_\infty(\epsilon_{f_{NEW}}, 0) = F_1 + F_2\,\epsilon_{f_{NEW}} + F_3\,\epsilon_{f_{NEW}}^2 \qquad (H.4)$$

and solving for $\epsilon_{f_{NEW}} (\equiv \epsilon_{f_1})$.

Burnup increments for each zone must now be calculated by predicting power-sharing.

Since,

$$\Sigma_f \equiv \left(\frac{\Sigma_a}{\nu}\right)\left(\frac{\nu\Sigma_f}{\Sigma_a}\right) \propto \Sigma_a k_\infty \qquad (H.5)$$

where ν = average number of neutrons per fission

Σ_f = macroscopic fission cross section, cm^{-1}

then

$$\Delta B_i \propto \Sigma_{f_i} \phi_i t \propto (\phi \Sigma_a k_\infty)_i \qquad (H.6)$$

Since inner zones 2 and 3 see the same flux $(\phi_2 = \phi_3)$, a single fit of outer zone 1 flux ϕ_1 normalized to that of the inner zones suffices to allow a determination of power sharing:

$$\text{Fraction of Cycle Energy } \theta_c \text{ supplied by ith zone} = \frac{(\phi \Sigma_a k_\infty)_i}{\Phi \Sigma_{a_1} k_{\infty_1} + \Sigma_{a_2} k_{\infty_2} + \Sigma_{a_3} k_{\infty_3}} \qquad \text{(H.7)}$$

where

$$\phi_1 = \Phi(\epsilon_{f_{NEW}}, k_{\infty_{INNER}}) \text{ of Equation VII}$$

$$\phi_2 = \phi_3 \equiv 1$$

After the burnup increments are determined for each zone, simulation of one irradiation is complete. Refueling is then represented by discharging zone 3 and renumbering zones 1 and 2 to 2 and 3, respectively. Clearly, the next irradiation can now be simulated by repeating all of the above steps. And so on, for all the cycles of interests. (The other operating cycle modes of Section H.1.1 are easily handled within this framework.)

When all fed and discharged fuel characteristics (ϵ_f, B_{FINAL}) have been determined, application of the uranium inventory Equation II (See Table H.1), current enrichment Equation III, and fissile plutonium inventory Equation IV provides pertinent mass balance data.

Reload batch fuel cost is then calculated using the simple, straightforward, but approximate equation:

$$\begin{pmatrix}\text{Batch Revenue Requirement}\\ \text{Present Valued to}\\ \text{Middle of Irradiation}\end{pmatrix} = \begin{pmatrix}\text{Batch Initial}\\ \text{Investment}\\ \text{Cost}\end{pmatrix}\left(1 + y\left(T_{pre} + \frac{T}{2}\right)\right) - \begin{pmatrix}\text{Batch}\\ \text{Salvage}\\ \text{Value}\end{pmatrix}\left(1 - y\left(T_{pst} + \frac{T}{2}\right)\right) \tag{H.8}$$

where $y = \frac{x}{1-\tau}$ = average cost of money (before taxes), per year

x = present value rate, per year

τ = income tax rate, fraction of taxable income

T = total in-core time, years

T_{pre} = pre-irradiation lead time for fuel purchases, years

T_{pst} = post-irradiation lag time for receipt of fuel credit, years

All batches are then present-valued to the study's base date to yield $\overline{TC}_r^p$, the total nuclear fuel revenue requirement for the "path" p of cycle energies (E_{r1}, E_{r2}, E_{r3},...) to the horizon. A second path p', equal to the first in all but one cycle (E_{r1}, $E_{r2}+\Delta$, E_{r3},...), can also be evaluated. Then, the λ_{r2} incremental cost for that cycle becomes simply

$$\lambda_{r2} \equiv \frac{\partial \overline{TC}_r}{\partial E_{r2}} \approx \frac{\overline{TC}_r^{p'} - \overline{TC}_r^{p}}{\Delta} \tag{H.9}$$

Returning to the question of determining the proper empirical coefficients, data points can be easily generated by a suitable physics-depletion code set such as CELL-CORE (40,41) or even LASER-FLARE (25,50). Multiple regression techniques (15) can be applied directly to the unit fuel cell data with a minimum of pre-fit data handling. On the other hand, the reactor irradiation data is best utilized in terms of the parameters of interest (e.g., power-sharing) as opposed to the physics quantities represented (e.g., flux ratios). In other words, the interpretation of Φ is qualitatively based on a flux ratio, but the actual Φ (to be used as input to any data-fitting package) is more appropriately backed-out of the actual power-sharing data using the empirical value of k_{∞_i} and Σ_{a_i} calculated for the same reactor core conditions.

Sample results for a Zion class 1100 MW PWR are shown in Figure H.2. Coefficients were fitted to Zion data output by CELL-CORE. Cost calculations are all based on annual refuelings with four week outages using unit costs representative of 1975 startup (46).

As an indication (See Table H.3) of simulator accuracy, in attempting to reproduce one of the fitted data points, QKCORE end of cycle burnups were in error by less than 0.6 per cent compared to CORE results (118 out of 19149 MWD/T at the end of second irradiation);

Figure H.2

QKCORE Fuel Simulator Results for 1100 MWe PWR at Steady-State

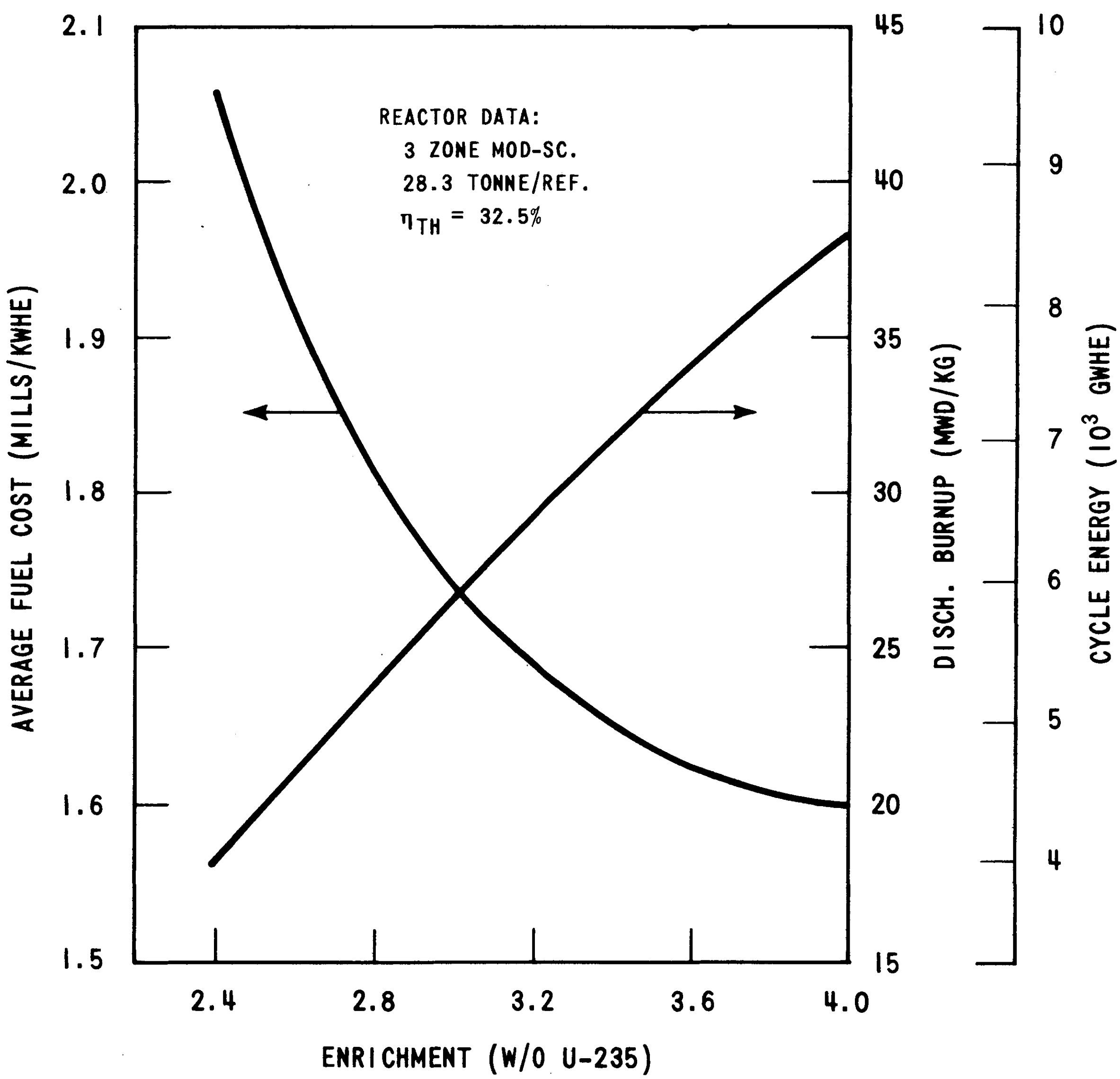


Table H.3

Comparison of QKCORE versus CORE results for 3.2% U-235 at Steady-state

NOTE: All burnups in MWD/T
30.1 Metric tonnes loaded at each refueling

	Average Zone Burnup				ERROR: QKCORE vs. CORE			
	CORE		QKCORE		End of Cycle		Cycle Increment	
					Absolute	Percent	Absolute	Percent
Initial	0.0		0.0					
Increment		9173		9200			27	.294
At End of Cycle 1	9173		9200		27	.294		
Increment		9976		10067			91	.912
At End of Cycle 2	19149		19267		118	.616		
Increment		9294		9163			-131	1.410
At End of Cycle 3 (Discharge)	28443		28430		-13	.046		

errors in cycle incremental burnups were higher but still less than 1.5 per cent (131 out of 9294 MWD/T).

Programming the empirical model and its associated cost calculations resulted in the 1300 card Fortran IV program QKCORE which requires 80K bytes of computer memory (plus 26 K for computer supervisor). Less than 0.2 sec of CPU time on an IBM 370 model 155 is required to simulate ten irradiation cycles including costing for each batch.

H.1.3 Code Structure and Mode of Operation

Table H.4 presents a summary of QKCORE subroutines while Figure H.3 portrays the general sequence of operations occurring in a QKCORE production run. (Table H.5 presents information relative to possible error messages printed by subroutine ICERRS.)

In order to calculate incremental costs ($\partial \overline{TC}_r / \partial E_{rc}$), an ELAME table (See Figure H.4) is passed to INCORE. The key path p of cycle energies is evaluated first. Then, each cycle, in turn, (last cycle first) is altered to a p' with a non-key E_{rc}, holding all others constant at their key value. Equation (H.9) is then used to determine λ_{rc} which is then "sandwiched" between the two pertinent cycle energies that differ (See Figure H.4).

Table H.4

Summary of QKCORE Subroutines

Name	Called BY	Calls	Purpose
QKCORE (Main)	------	INCORE ICNPUT ICERRS ERASE	Reads QKCORE input, then calls INCORE (see Table F.3)
INCORE (ICNPUT)	QKCORE (Main)	REDCOR FULSIM INIT3 EMPRCL ICERRS	Supervises in-core simulation; Has ENTRY ICNPUT to initiate reading of input data by subroutine REDCOR
REDCOR	ICNPUT	INIT2 UF6VAL SETUVL PVINIT ICERRS ERASE	Reads input data for INCORE
FULSIM	INCORE	CONSTS NXTIRR FRSIRR CSTBAT PRTTOP PRTBTM ERASE	Supervises fuel irradiation simulation for all E's

Table H.4--Continued

Name	Called By	Calls	Purpose
CONSTS	FULSIM	UNTCOS PVPER$	Supervises calculation of unit ($/Kg) cost for all batches
NXTIRR (FRSIRR)	FULSIM	FK8 FSIGA FEPF FK8NEW FPHI FECOUT ICERRS	Performs simulations of next irradiation; Has ENTRY FRSIRR for initial split cycle
CSTBAT (INIT3)	INCORE FULSIM	FKGUR FEPB FKGPU UF6VAL PVPER$ ERASE	Calculates cost of batch of fuel; Has ENTRY INIT3 for initialization
PRTTOP (PRTBTM)	FULSIM	------	Prints top of FULSIM result table; Has ENTRY PRTBTM to print bottom of table

Table H.4--Continued

Name	Called By	Calls	Purpose
EMPRCL (FK8, FKGUR, FEPB, FKGPU FSIGA, FEPF, FK8NEW, FPHI, FECOUT)	INCORE NXTIRR FRSIRR CSTBAT	------	Initializes empirical equations; Has multiple ENTRY points for each equation
UNTCOS (INIT2)	REDCOR CONSTS	------	Calculates escalated unit ($/Kg) costs; Has ENTRY INIT2 to initialize excalation constants
UF6VAL (SETUVL)	REDCOR CSTBAT	PVPER$	Calculates value of enriched uranium ($/Kg UF_6); Has ENTRY SETUVL to pre-calculate constants in value equation
PVPER$ (PVINIT)	REDCOR CONSTS CSTBAT SETUVL	------	Calculates present (at base date) value of one dollar; Has ENTRY PVINIT to initialize present value rate; Identical to SYSOPT version (see Appendix F)
ICERRS	QKCORE (Main) INCORE REDCOR NXTIRR FRSIRR	------	Prints error messages and choses to terminate execution if severe error occurs (see Table H.2)

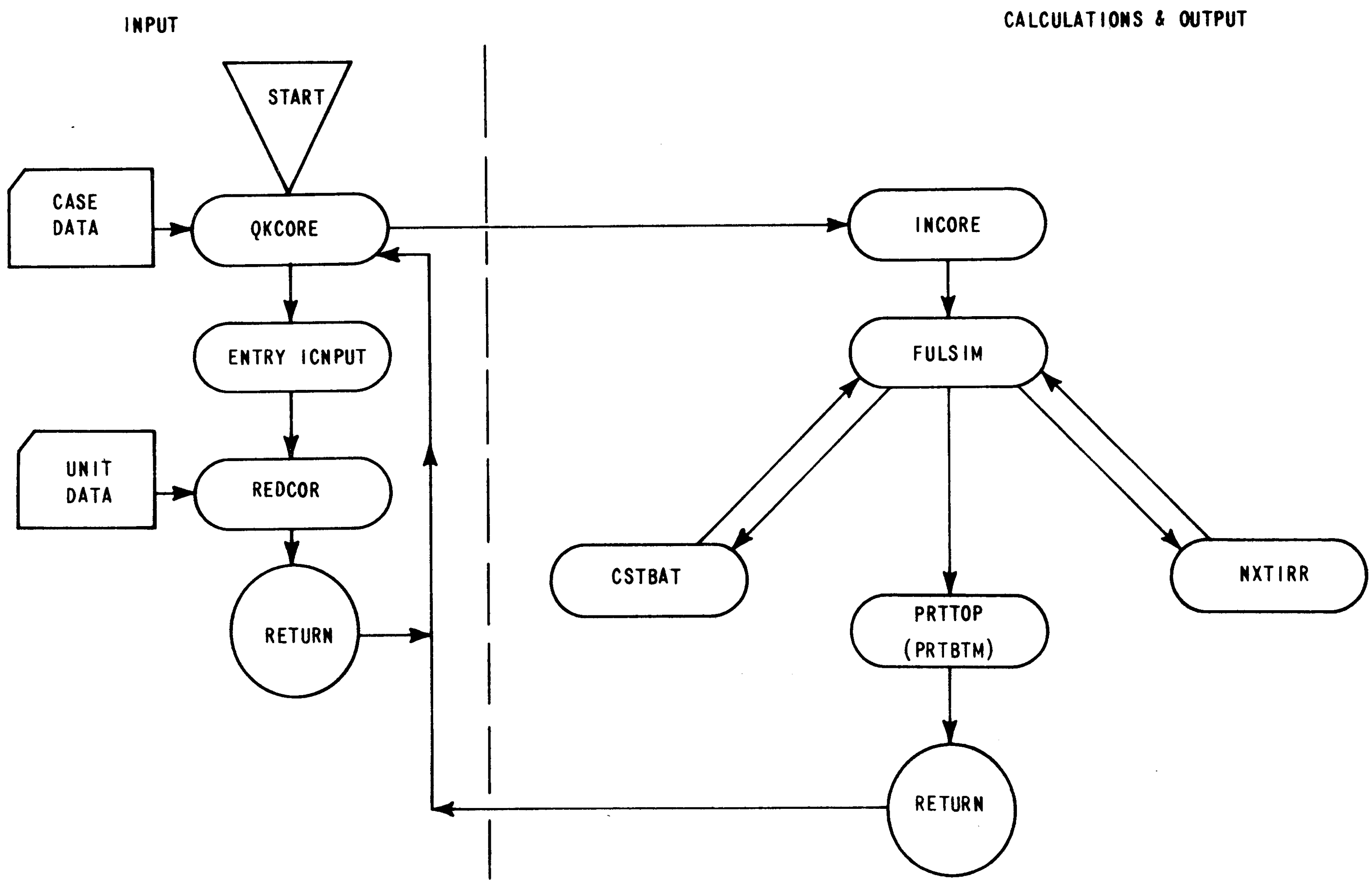
Table H.4--Continued

Name	Called By	Calls	Purpose
ERASE	QKCORE (Main) REDCOR FULSIM CSTBAT	------	MIT Assembler Language program that sets arrays to zeroes rapidly

Note: Computer installation-dependent dataset reference numbers for RD and WT may be altered in ICNPUT.

Figure H.3

QKCORE Flowchart



6353-94

Table H.5

QKCORE Error Messages Printed by ICERRS

Number*	Source	Action after Printing	Error
1	NXTIRR	RETURN	Cycle energy stretched-out more than 25% of reactivity-limited energy
2	NXTIRR	RETURN	Cycle energy less than 75% of reactivity-limited energy
3	{ QKCORE(MAIN) { REDCOR	Terminate	Input deck has improper sequence and/or card
4	INCORE	Terminate	Array G in subroutine INCORE too small for problem
5	{ INCORE { REDCOR	Terminate	One or more inputs are outside permissible limits
6	INCORE	RETURN	NCYCTO $\neq$ NCYCIN + NCYCXS when subroutine INCORE entered
7	INCORE	Terminate	Data for unit IDNUM not read in
8	QKCORE	Terminate	"Stop" Card 27 or severe error encountered
9	REDCOR	RETURN	Power-sharing fractions (see Card 15 of Section H.2) do not sum within 1 ± 10^{-5}
A(=10)	QKCORE	Terminate	Too many cycle-energies being investigated
B(=11)	NXTIRR	RETURN	Needs reload enrichment < 1.5 w/o U-235 or > 5.0 w/o U-235
C(=12)	NXTIRR	Terminate	NXTIRR improperly called instead of FRSIRR

*The error number initiating the ICERR print appears as the rightmost digit in the accumulated ERRCOD (which is printed as part of the message).

Figure H.4

ELAME Table

CYCLE c

INDEX I	1	2	3	4	5
1	E_{r1}	E_{r2}	E_{r3}	E_{r4}	E_{r5}
2	λ_{r1}		λ_{r3}		
3	$E_{r1} + \Delta$		$E_{r3} + \Delta$	$E_{r4} + \Delta$	
4	λ_{r1}		λ_{r3}		
5	$E_{r1} + 2\Delta$		$E_{r3} + 2\Delta$		
6	λ_{r1}	PATH P	λ_{r3}		
7	$E_{r1} + 3\Delta$		$E_{r3} + 3\Delta$		$E_{r5} + 3\Delta$
8	λ_{r1}		λ_{r3}		
9	$E_{r1} + 4\Delta$	$E_{r2} + 4\Delta$	$E_{r3} + 4\Delta$		

PATH P'

C = CYCLE c

$$ELAME\ (I.C) = E_{rc} + [(I-1)/2] * \Delta \quad \text{IF I ODD}$$

$$= \lambda_{rc} \quad \text{IF I EVEN}$$

H.2 QKCORE Input Specifications

Table H.6 presents the complete input specifications for QKCORE. "INCORE" Card 1 initiates reading of INCORE input data. Card 2 indicates the amount of input data and print options desired. A single set of economic parameters (with quadratic escalation permitted) are input on Cards 3-11. Reactor unit initial conditions and thermal efficiencies appear on Cards 12-15. Card 16-17 contain sets of reactor empirical constants while sets of fuel empirical constants are input on Cards 18-19. "END " Card 20 indicates end of INCORE input. Then, Card 21 "CASE" enters case data on Cards 22-25. Another"CASE" can then be entered, or a "NEW " Card 26 enters any new INCORE data (back to Card 1). Finally a "STOP" Card 27 terminates QKCORE execution.

Table H.6

QKCORE Input Specifications

Variable	Columns	Format	Description
Card 1			
...	1-12	...	"INCORE INPUT" Control Card initiates input of INCORE data
...	13-80	17A4	Free for comments
Card 2			
NUECON	1-5	I5	Control parameter for new economic data: if: =0 , Cards 3 to 11 not to be read in =1 , Cards 3 to 11 to be read in
NURCRS	6-10	I5	Number of individual reactors (i.e., nuclear units) for which data to be read in, $0 \le$ NURCRS $\le$ MXRCRS (=15)
NURCRK	11-15	I5	Number of sets of reactor empirical constants for which data to be read in, $0 \le$ NURCRK $\le$ MXRCRK (=15)
NUFULK	16-20	I5	Number of sets of fuel empirical constants for which data to be read in, $0 \le$ NUFULK $\le$ MXFULK (=5)
RELCST	21	L1	Print option for relative cost results $(\overline{TC}_r^{p'} - \overline{TC}_r^{p})$ in in ELAME table, F = No T = Yes
INCCST	22	L1	Print option for incremental cost λ_{rc} results in ELAME table, F = No T = Yes

Table H.6--Continued

Variable	Columns	Format	Description
BALCST	23	L1	Print option for batch costs of key cycle energy path p F = No T = Yes
NBLCST	24	L1	Print option for batch costs at all cycle energy paths p' F = No T = Yes
PIRDAT	25	L1	Print option for irradiation data of all paths, F = No T = Yes
PBATCS	26	L1	Print option for detailed batch cost data of all paths, F = No T = Yes

Note: Cards 3 to 11 may be omitted from subsequent INCORE INPUT blocks if no changes in previous economic data read in. Then, NUECON = 0. If QKCORE used in SYSOPT overlay structure (See Section F.1.2), always use NUECON = 1.

Card 3

Variable	Columns	Format	Description
ECTITL	1-80	20A4	Title for economic data

Card 4

Variable	Columns	Format	Description
XF	1-10	F10.3	Enrichment of diffusion plant feed material (yellowcake), weight fraction U-235
XW	11-20	F10.3	Enrichment of diffusion plant tails, weight fraction U-235
TXRATE	21-30	F10.3	τ, income tax rate, fraction of taxable income

Table H.6--Continued

Variable	Columns	Format	Description
PVRATE	31-40	F10.3	x, present value rate, fraction per year
TBASE	41-50	F10.3	Calendar base data for present valuing, years
DTPRE	51-60	F10.3	T_{pre}, pre-irradiation lead time for fuel purchases, years
DTPST	61-70	F10.3	T_{pst}, post-irradiation lag time for receipt of fuel credit, years
DTY2F6	71-80	F10.3	Effective delay time from yellowcake to UF_6, years
Card 5			
A0(1)	1-10	F10.3	Constant term in yellowcake unit cost escalation, \$/lb U_3O_8
A1(1)	11-20	F10.3	Linear coefficient in yellowcake unit cost escalation, \$/lb U_3O_8/year
A2(1)	21-30	F10.3	Quadratic coefficient in yellowcake unit cost escalation, \$ /lb U_3O_8/year^2
Card 6			
A0(2)	1-10	F10.3	Constant term in uranium conversion unit cost escalation, \$/kgU
A1(2)	11-20	F10.3	Linear coefficient, \$/kgU/year
A2(2)	21-30	F10.3	Quadratic coefficient, \$/kgU/year^2
FCOR	31-40	F10.3	Yield in uranium conversion step, fraction

Table H.6--Continued

Variable	Columns	Format	Description
Card 7			
A0(3)	1-10	F10.3	Constant term in separative work unit cost escalation, \$/kg SWU
A1(3)	11-20	F10.3	Linear coefficient, \$/kg SWU/year
A2(3)	21-30	F10.3	Quadratic coefficient, \$/kg SWU/year^2
Card 8			
A0(4)	1-10	F10.3	Constant term in fabrication unit cost escalation, \$/kg Fab.
A1(4)	11-20	F10.3	Linear coefficient, \$/kg Fab./year
A2(4)	21-30	F10.3	Quadratic coefficient, \$/kg Fab./year^2
FFAB	31-40	F10.3	Yield in fabrication step, fraction
Card 9			
A0(5)	1-10	F10.3	Constant term in shipping and reprocessing unit cost escalation, \$/kg S&R (U+Pu)
A1(5)	11-20	F10.3	Linear Coefficient, \$/kg S&R (U+Pu)/year
A2(5)	21-30	F10.3	Quadratic Coefficient, \$/kg S&R(U+Pu)/year^2
FSAR	31-40	F10.3	Yield in reprocessing step, fraction
Card 10			
A0(6)	1-10	F10.3	Constant term in uranium reconversion unit cost escalation, \$/kg U.
A1(6)	11-20	F10.3	Linear coefficient, \$/kg U/year

Table H.6--Continued

Variable	Columns	Format	Description
A2(6)	21-30	F10.3	Quadratic coefficient, \$/kg U/year^2
FCRE	31-40	F10.3	Yield in uranium reconversion step, fraction

Card 11

Variable	Columns	Format	Description
A0(7)	1-10	F10.3	Constant term in fissile plutonium value escalation, \$/gm fis.Pu
A1(7)	11-20	F10.3	Linear Coefficient, \$ /gm fis. Pu/year
A2(7)	21-30	F10.3	Quadratic coefficient, \$ /gm fis.Pu/year^2

Note: There must be NURCRS sets of Cards 12 to 15, one for each nuclear unit. If no change in previous NRCRS (nuclear unit data read in previously), NURCRS may equal zero. However, if QKCORE used in SYSOPT overlay structure (See Section F.1.2), always use NURCRS > 0

Card 12

Variable	Columns	Format	Description
IDNO	2-5	1X,I4	Unique unit identification number
NAME	7-10	1X,A4	Unit name
MWCAP	11-15	I5	Unit net capacity, MW
IRCRKA	16-20	I5	Pointer to set of reactor empirical constants to be used for unit, 1 ≤ IRCRKA ≤ NRCRK
IFULKA	21-25	I5	Pointer to set of fuel empirical constants to be used for unit, 1 ≤ IFULKA ≤ NFULK
NOZONE	26-30	I5	n, number of refueling zones in units' fuel management scheme, 1 ≤ NOZONE ≤ 10
ZONKG	31-40	F10.2	Mass of uranium fabricated for placement in units' outer zone, kg

Table H.6--Continued

Variable	Columns	Format	Description
EFFNET	41-50	F10.2	Average net thermal efficiency for unit, fraction
DECRIT	51-60	F10.2	Energy remaining in split cycle (at start of simulation) until reactivity-limited burnup reached, GWHe
DESTCH	61-70	F10.2	Maximum stretchout permitted in cycle with fixed reload enrichment, GWHe
EFFINC	71-80	F10.2	Incremental net thermal efficiency for unit, fraction If = 0 or blank, EFFINC set equal to EFFNET internally.

Card 13

Variable	Columns	Format	Description
N	1-2	12	Number of entries to follow for EPFFX $0 \leq N \leq$ MXCYTO (=20) - NCYCXS
EPFFX(1) to EPFFX(8)	3-80	F8.3, 7F10.3	ϵ_f Refueling enrichment already ordered for reactor, w/o U-235 if < 0, $\lvert \epsilon_f \rvert$ is enrichment loaded at that refueling with reactivity-limited energy to be determined. if =0 (or blank), enrichment not ordered; free to choose reload enrichment to give reactivity-limited energy desired. if > 0, ϵ_f enrichment ordered, extract cycle energy (regardless of reactivity-limited energy).

Note: If N > 8, there must be [(N-1)/8] Card 14's for remaining EPFFX

Table H.6--Continued

Variable	Columns	Format	Description
Card 14			
EPFFX(9) to EPFFX(N)	1-80	8F10.3	Remaining EPFFX (see Card 13)

Note: There must be NOZONE Card 15, one for each zone of the reactor. First Card 15 is for Zone 1 (freshest fuel), while last Card 15 is for Zone NOZONE (about to be discharged).

Variable	Columns	Format	Description
Card 15			
EPFSRT	1-10	F10.3	ε_{f_i} As-fabricated enrichment w/o U-235
BSRT	11-20	F10.3	B_i Current average burnup at start of simulation, MWD/kg U fab.
FABINV	21-30	F10.3	Remaining book value of fabrication to be depreciated before discharge, $/kg U fab.
SRCINV	31-40	F10.3	Current book value of shipping, reprocessing and reconversion (to be appreciated before discharge), $/kg (U+Pu) disch.
POWFRC	41-50	F10.3	Power-sharing for this zone during this initial split cycle, fraction of total core output

$$\left| \sum_{i=1}^{NOZONE} POWFRC_i - 1 \right| \text{ must be } < 10^{-5}$$

Note: If simulation does not start with split cycle, zone parameters for last Card 15 should be chosen judiciously since instantaneous depreciation of FABINV and appreciation of SRCINV can result in error in total cost (incremental costs are not affected). (Subroutine CSTBAT currently assumes the initial cycle is a split cycle.) Try EPFSRT = 1.0, FABINV = 0.0 and SRCINV = AO(5) + AO(6) to net error to zero.

Table H.6--Continued

Variable	Columns	Format	Description

Note: There must be NURCRK sets of Cards 16 and 17, one set for each set of reactor empirical constants. If no change in NRCRK sets of constants read in previously, NURCRK may equal zero. However, if QKCORE used in SYSOPT overlay structure (see Section F.1.2), always use NURCRK > 0.

Card 16

Variable	Columns	Format	Description
RCRKTL	1-80	20A4	Title card for set of reactor empirical constants

Note: There must be three Card 17's to accommodate the 18 constants in each set.

Card 17

Variable	Columns	Format	Description
RCRCON(1) to RCRCON(18)	1-80	3(6E12.6)	R_i, Reactor empirical constants, 12 constants currently used (see Table H.2)

Note: There must be NUFULK sets of Cards 18 and 19, one set for each set of fuel empirical constants. If no change in NFULK sets of constants read in previously, NUFULK may equal zero. However, if QKCORE used in SYSOPT overlay structure (see Section F.1.2), alway use NUFULK > 0.

Card 18

Variable	Columns	Format	Description
FULKTL	1-80	20A4	Title card for set of fuel empirical constants

Note: There must be eight Card 19's to accommodate the 48 constants in each set.

Card 19

Variable	Columns	Format	Description
FULCON(1) to FULCON(48)	1-80	8(6E12.6)	F_i, Fuel empirical constants, 44 currently used (see Table H.1)

Card 20

Variable	Columns	Format	Description
. . .	1-4	. . .	"END "Control card signifying end of REDCOR input.

Table H.6--Continued

Variable	Columns	Format	Description
. . .	5-80	19A4	Free for comments
Card 21			
. . .	1-4	. . .	"CASE" Control card indicating case data to be read by QKCORE
. . .	5-80	19A4	Free for comments
Card 22			
CATITL	1-80	20A4	Case title card
Card 23			
NCYCIN	1-10	I10	Number of cycles involved in horizon (initial cycle assumed split and final cycle may be split)
NCYCXS	11-20	I10	Number of complete extra (excess) cycles beyond horizon (=NOZONE-1)
Note:	NCYCTO = NCYCIN + NCYCXS $\leq$ MXCYTO (=20)		
IDNUM	27-30	6X,I4	IDNO of unit being input (used to retrieve unit data input by REDCOR)
ECHDOV	31-40	F10.2	Energy held over beyond horizon in split cycle, 0 $\leq$ ECHDOV, GWHe
Note:	There must be NCYCTO sets of Card 24 and 25, one set for each cycle in simulation.		
Card 24			
I	1-10	I10	Cycle number, 1 $\leq$ I $\leq$ NCYCTO
NECBAL	11-20	I10	Position of key cycle energy on Card 25, 1 $\leq$ NECBAL $\leq$ NES
TS	21-30	F10.4	Calendar time at start of irradiation cycle, years
TE	31-40	F10.4	Calendar time at end of irradiation cycle, years

Table H.6--Continued

Variable	Columns	Format	Description
NES	41-50	I10	Number of cycle energies to be read in on Card 25, $1 \leq NES \leq [(MXESX2+1)/2]=25$
TO	51-60	F10.4	Length of time unit operated during cycle, years $TO \leq TE-TS$

Note: There must be [(NES + 7)/8] of Card 25 to accommodate the NES cycle energies.

Card 25

Variable	Columns	Format	Description
ERC(1) to ERC(NES)	1-80	8F10.4	Alternative cycle energies for cycle I, GWHe [If I=1 and not split cycle, ERC(1) = 0.03]

Note: Next card may be "NEW" Card 26, "CASE" Card 21 or "STOP" Card 27 with input sequence reverting to appropriate point.

Card 26

Variable	Columns	Format	Description
. . .	1-4	. . .	"NEW "Control card initiates input of new INCORE data. Revert to Card 1 in input sequence.
. . .	5-80	19A4	Free for comments

Card 27

Variable	Columns	Format	Description
. . .	1-4	. . .	"STOP" Control card to terminate execution of QKCORE for this computer run.
. . .	5-80	19A4	Free for comments

H.3 QKCORE Sample Problem

Figure H.5 presents a QKCORE Sample Problem input deck which is, in fact, part of (i.e., Reactor 2) the SYSOPT Sample Problem in Figure F.4. Figure H.6 presents a summary of QKCORE output for the Sample Problem.

FIGURE H.5

QKCORE SAMPLE PROBLEM INPUT DECK

```
// 'DEATON',CLASS=B,REGION=96K
/*MITID  USER=(M7894,6948)
/*MAIN LINES=20,CARDS=30,TIME=2
/*SRI LOW
//CALCGO EXEC FORG,PROG='USERFILE.M7894.6948.LOAD.QKCORE(GO)'
//G.SYSIN DD *,DCB=(RECFM=FB,LRECL=80,BLKSIZE=2000)
INCORE INPUT
    1    1    2    2TTTTTT
    TYPICAL SET OF ECONOMIC DATA  REF: MB(22.27 NOTES) & EAM (NN 2/71)
.00711     .002       .50        .97        0.0        .329       .60        0.123
 8.00
 2.52                            .997
 28.70
 70.00                           .998
 45.00                           .99
 3.00                            .997
 7.50
  200 NK-2 1050    1    1    3  28300.  .316        700.      500.
 1   3.4
  3.4       9.0       48.80      12.00      .3282
  3.2      19.0       24.40      25.00      .3519
  3.2      28.0        2.68      37.00      .3199
REACTOR DATA FOR 1100 MWE ZION CLASS; 3 ZONE; NO PU RECYCLE     COMPUTER VERS.
.141076     .0000218686 -.343574E-09-1.70111    -17.7720    .000105261  K8NEW
5.69086     -4.58587    1.13417     -.0967594   8.52622     -17.8407    PHI

REACTOR DATA FOR 1100 MWE ZION CLASS; 3 ZONE; NO PU RECYCLE     SLIDE RULE VERS.
.20943      .000017424              -2.8845                              K8NEW
2.570870    -1.465217   .1782609    0.0         4.34783     0.0         PHI

   FUEL DATA FOR 1100 MWE ZION CLASS; 3 ZONE; NO PU RECYCLE     COMPUTER VERS.
 .805642E 00 .195080E 00-.153501E-01-.148402E-01 .162489E-02-.257056E-03A1&A2 K8
 .206350E-03-.862559E-04 .132240E-04 1.00                               A3K8A1UR
-.189080E-02 .217089E-03-.333958E-04 .180377E-04-.724569E-05 .100522E-05A2&A3 UR
 .114056E 00-.385956E-01 .429073E-02-.128230E-02 .644474E-03-.858913E-04A1&A2 WO
 .311900E-04-.147188E-04 .189895E-05 .322341E-02 .173157E-02-.577577E-04A3WOA1PU
-.384453E-04 .463393E-04-.584225E-05 .165077E-05-.165659E-05 .252965E-06A2&A3 PU
 .112485E-02-.205317E-03 .220933E-04 .228005E 00-.703667E-01 .773171E-02UL&PL PU
.053238     .017860                                                     SIGA
   FUEL DATA FOR 1100 MWE ZION CLASS; 3 ZONE; NO PU RECYCLE     SLIDE RULE VERS.
.955        .090                    -.00897                             A1&A2 K8
                                    1.0                                 A3K8 A1U
-.00137                                                                 A2&A3 UR
.0652       -.0100                  80.     E-06                        A1&A2 WO
                                    .00365      .00156                  A3URA1PU
                                                                        A2&A3 PU
.00071                              0.0872                              UL&PL PU
.0532       .0179                                                       SIGA
END OF INCORE INPUT
CASE
   REACTOR 2 UNDER STRATEGY 2 AT A FEW REPRESENTATIVE CYCLE ENERGIES
        6         2       200    1160.00
        1         3    0.0000     0.0833         3     0.0792
     500.      600.      700.
        2         1    0.2500     1.0833         1     0.7916
    7200.
        3         2    1.2500     2.2500         3     0.9500
    7400.     7500.     7600.
        4         1    2.4167     3.6667         1     1.1875
    7500.
        5         1    3.8333     5.0000         1     1.1083
    7700.
        6         1    5.1667     6.1667         1     0.9500
    7500.
        7         1    6.3333     7.3333         1     0.9500
    7000.
        8         1    7.5001     8.5001         1     0.9500
    7000.
STOP
/*
```

Figure H.6

QKCORE Sample Problem Output

```
INDEX=  1 IDNO=  200    * * * * * INCREMENTAL REACTOR TOTAL COST  (P.V.$/MWHE) * * * * *

        REACTOR TOTAL COST FOR BALANCED EC'S (ECBAL)  =   52762.571  10**3P.V.$

 ECBAL     700.0    7200.0    7500.0    7500.0    7700.0    7500.0

EDUPLM    1200.0    7310.0       0.0       0.0       0.0       0.0

 CYCLE       1         2         3         4         5         6

  EC      500.00  7200.00   7400.00   7500.00   7700.00   7500.00
INCCST    0.8354*********   1.6805***************************
 ETC.     600.00     0.0    7500.00      0.0       0.0       0.0
          0.8618   0.0       1.7834    0.0       0.0       0.0
          700.00     0.0    7600.00      0.0       0.0       0.0
       *********   0.0    *********    0.0       0.0       0.0
```

H.4 QKCORE Source Listing

The following is a Fortran IV source listing of the QKCORE code (included only in MIT library copies).

APPENDIX I

NOMENCLATURE AND ACRONYMS

Symbol	Description	Dimension[1]
A	Area under fractional load-duration curve	MW
a	Coefficient of cycle energy in linear approximation to λ	$\frac{\$}{(MWH)^2}$
AH	Available Hours, those during which a unit is available (7)	hours
b	Constant term in linear approximation to λ	$\frac{\$}{MWH}$
C	(See Subscripts)	
c	Numerical constant	
CORSOM	CORe Simulation and Optimization Model	
D	Customer electric energy demand	MWH
d	Duration of load, amount of time that load $\geq$ specified power level	hours
DM	Equivalent load spacing along F curves	MW
E	Electric energy produced	MWH
$\mathcal{E}$	Set of all E_{rcp} or $\{E_{rcp}\}$	MWH
e	Electric energy unit cost	$\frac{\$}{MWH} \equiv \frac{mills}{kwhe}$
F	Fractional load-duration, probability that load $\geq$ specified power level at random instant	fraction of period

[1]The symbol \$ represents present-valued or discounted dollars while |\$| represents absolute-value or non-discounted dollars. All MW are in net megawatts electric.

Symbol	Description	Dimension[1]
f_G	Probability density function of unit performing (capable of P_G MW)	per MW
f_O	Probability density function of unit not performing (derated P_O MW)	per MW
$\mathcal{f}$	Forced-outage importance, fraction of FOH actually affecting system generating operations	(None)
FOH	Forced-Outage Hours, those during which a unit was unavailable due to a forced-outage (7)	hours
FOR	Forced-Outage Rate (7), See Equation (2.6)	(None)
FORH	Forced-Outage Reserve Hours, those during which a unit was unavailable due to a forced-outage, but would have been in reserve shutdown status if available.	hours
FOSH	Forced-Outage Service Hours, those during which a unit was unavailable due to a forced-outage, but would have been in service status if available	hours
g	(See Subscripts)	
H	Heat input rate	$\frac{\text{MegaBTU}}{\text{hour}}$
h	Heat rate	$\frac{\text{MegaBTU}}{\text{MWH}}$
I	(See Subscripts)	
K	Unit capacity	MW
k	Unit capacity above minimum	MW
L	Capacity factor	(None)

Symbol	Description	Dimension[1]
ℓ	Increment capacity factor, i.e., above minimum	(None)
LIFO	Last-In, First-Out inventory accounting	
LOLP	Loss-Of-Load Probability	fraction of period
LP	Linear Programming	
M	Misfit potential, objective function for outer shape iterations	"misfits"
m	Misfit forcing function	$\frac{\text{"misfits"}}{\text{MWH}}$
MOH	Maintenance Outage Hours, those during which a unit is unavailable due to a postponed repair maintenance outage (7)	hours
N	Nuclear Potential	MWH
NP	Network Programming	
O-O-K	Out-Of-Kilter Network Program	
ORR	Operating Revenue Requirement to the horizon	$
P	Power or load level	MW
$\mathcal{P}$	Probability unit capable of generating P_G MW or more when called upon	(None)
p	Performance probability, probability unit capable of generating K MW when called upon	(None)
PH	Period Hours, total hours in the period (7)	hours
POH	Planned Outage Hours, those during which a unit is unavailable due to a planned preventive maintenance outage (7)	hours

Symbol	Description	Dimension[1]
PV	Present Value of stream of expenditures within horizon	$
Q	Quantity of equivalent thermal energy input during a startup-shutdown sequence	MegaBTU
q	Non-performance probability, probability unit will not perform when called upon	(None)
QKCORE	QuicK in-CORE nuclear reactor core simulator and cost accounting computer code	
QP	Quadratic Programming	
R	(See Subscripts)	
R'	(See Subscripts)	
RAMM	Refueling And Maintenance Model	
RR	Revenue Requirement to the horizon associated with a direct expense	$
RSH	Reserve Shutdown Hours, those during which a unit is off-line due to economy or similar reasons but is available as reserves (7)	hours
S	Strategy or schedule of system refueling and maintenance outages	
s^2	Variance of F_e equivalent load-duration shape (Nuclear upper increments only)	(None)
SH	Service Hours, those during which a unit is "actually operated with breakers closed to station bus" (7)	hours
SIM	System Integration Model	
SOH	Scheduled Outage Hours, those during which a unit is unavailable due to mainetnance and planned outages (7)	hours

Symbol	Description	Dimension
SOM	System Optimization Model	
SYSINT	SYStem INTegration model computer code.	
SYSOPT	SYStem OPTimization model computer code	
T	Duration of a time interval extending over several time periods	hours
T'	Duration of a time period	hours
t	Time, calendar time	hours
$\overline{TC}$	Total Cost (i.e., revenue requirement) to horizon	$
V^2	Total internal variance of mean availability-based increment capacity factors (Nuclear upper increments only)	(None)
W^2	Weighted sum of squares reactor average versus system average availability-based increment capacity factors (Nuclear upper increments only)	(None)
X	Expenditures during period	|$|
x	Present value rate ≡ discount rate ≡ effective cost of money	$\frac{\text{fraction}}{\text{year}}$
Z	Time at end of planning horizon (See also Subscripts)	hours
α	Coefficient of E in Equation (4.36)	$\frac{\text{fraction}}{\text{MWH}}$
β	Constant term in Equation (4.36)	(None)
γ	Fraction of σ applied to limits on availability-based increment capacity factors	(None)
Δ	Energy step size for segmenting incremental cost curves	MWH
ΔK	Capacity of Increment	MW

Symbol	Description	Dimension
δ	Error in estimated objective function for next SOM iteration	\$
δK	Power level within ΔK capacity increment	MW
η	Energy conversion efficiency	$\frac{MW(e)}{MW(t)}$
θ	Thermal energy consumption	MegaBTU
λ	Incremental energy cost	$\frac{\$}{MWH} \equiv \frac{mills}{kwhe}$
Σ	Change in ORR at next trial solution	\$
σ	Average reduction in $\ell'_r - \bar{\ell}'$ required to pass shape test	(None)
τ	Time at end of cycle	hours
ϕ	Incremental fossil thermal energy cost during the period	$\frac{\|\$\|}{MegaBTU}$
$\bar{\phi}$	Levelized incremental fossil thermal energy unit cost	$\frac{\$}{MegaBTU}$
ψ	Lagrangian auxiliary function	\$
Ω	Frequency of startup-shutdown sequence	per day
\$	Same as RR; the units of present-valued or discounted dollars	\$
|\$|	The units of absolute value or non-discounted dollars	|\$|
{...}	Set of all ...	

Subscripts

ACT	ACTual
C	Cycle number at end of planning horizon
c	Cycle or contract
D	Direct demand
e	Equivalent or expected
EST	ESTimated
F	Fossil
G	Generating mode
g	Ordered sub-group of unit increments
H	Hydro
I	Indirect demand
I	Total number of capacity increments for unit
$\mathcal{I}$	Total number of capacity increments currently being considered for unit
i	Increment of unit capacity
inc	incremental
N	Nuclear
O	Outage
P	Pumped-hydro or pumping mode
p	Period number
R	Number of reactors or generating units
R'	Number of on-line reactors
r	Reactor or generating unit
REJ	REJection level
S	Startup-shutdown

Subscripts

T	Total for utility system
U	Unserved (energy), urgent or emergency (purchases)
Z	Total number of periods in planning horizon
Z+1	Fictitious holdover period beyond planning horizon

Superscripts

max	maximum
min	minimum
o	Out, as in without
s	Shape interation
t	Trial or inner total cost iteration
w	With
wo	without
$\overline{\ \ }$	Average; levelized
$*$	At the optimum
$\circledast$	At the acceptable optimum
$\circ$	At zero; is invariant
$'$	Availability-based

APPENDIX J

REFERENCES

(1) "AEC to Raise Enrichment Prices in August and Institute an Annual 2% Hike," Nucleonics Week, 14, 7, February 15, 1973.

(2) "A Four-state Model for Estimation of Outage Risk for Units in Peaking Service," IEEE Task Group on Models for Peaking Service Units, IEEE Trans. on Power App. and Sys., PAS-91, 618-627 (March/April 1972).

(3) "IBM System/360 Operating System Utilities," Manual C28-6586-8, IBM Corporation (1968).

(4) "Present Practices in the Economic Operation of Power Systems," IEEE Trans. on Power App. and Sys., PAS-90, pp. 1768-1775 (July/August 1971).

(5) "Pumped Storage comes to the Rescue," Electrical World, p. 106, June 15, 1972.

(6) "Quadratic Programming System," MIT Information Processing Center, Applications Program Manual AP-53, May 22, 1970.

(7) "Report on Equipment Availability for the Nine-Year Period 1960-1968," Report of EEI Prime Movers Committee, EEI Publication 69-33, September 1969.

(8) "Zeroing Arrays," MIT Information Processing Center, Applications Program Manual, AP-39 Revision 1, March 1, 1972.

(9) Anderson, S. W., R. T. Jenkins, D. S. Joy, E. T. Merrill and T. D. Wolsko, "Methods of Planning Generating Unit Operation," Proc. Am. Power Conf., April 1970.

(10) Baleriaux, H., E. Jamoulle and Fr. Linard deGuertechin, "Simulation de l'exploitation d'un parc de machines thermiques de production d'électricité couplé à des stations de pompage," Revue E (edition S.R.B.E.), V, No. 7, pp. 3-24 (1967).

(11) Baughman, M. L., "Dynamic Energy System Modeling--Interfuel Competition," Energy Analysis and Planning Group, MIT, Report No. 72-1, September 1972.

(12) Benedict, M., "Economics of Nuclear Power," 22.27 Course Notes, MIT Dept. of Nuclear Engineering, Spring 1971.

(13) Benedict, M. of MIT, personal communication, October 26, 1972.

(14) Bennett, L. L. and J. Turnage of ORNL, personal communication, October 12, 1972.

(15) Bevington, P. R., "Data Reduction and Error Analysis for the Physical Sciences," McGraw-Hill, New York (1969).

(16) Billinton, R., "Bibliography on the Application of Probability Methods in Power System Reliability Evaluation," IEEE Trans. on Power App. and Sys., PAS-91, pp. 649-660 (March/April 1972).

(17) Booth, R. R., "Notes on Computer Programs Developed During Period of Stay with T.V.A.: Jan.-Feb.-March 1970," State Electricity Commission of Victoria, Australia, April 1970.

(18) Booth, R. R., "The A-B-C of Probabilistic Simulation," State Electricity Commission of Victoria, Australia, February 1971.

(19) Booth, R. R., "Power System Simulation Model Based on Probability Analysis," IEEE Trans. on Power App. and Sys., PAS-91, 62-69 (January/February 1972).

(20) Commonwealth Edison Company of Chicago, personal communications with the Staff, November 1969 to February 1973.

(21) Day, J. T., "Factors Affecting the Justification of Nuclear Units in Generation Expansion," Trans. Am. Nucl. Soc., 13, 770 (1970).

(22) Deaton, P. F. and E. A. Mason, "Parallel Derivation of Marginal Costs Pertinent to Utility System Optimization," Trans. Am. Nucl. Soc., 15, 375 (1972).

(23) Deaton, P. F. and E. A. Mason, "A System Integration and Optimization Model for Nuclear Power Management Planning," Trans. Am. Nucl. Soc., 15, 373-375 (1972).

(24) DeGarmo, E. P., "Engineering Economy," Fourth edition, p. 441, Macmillan, Toronto (1967).

(25) Delp, D. L., D. L. Fischer, J. M. Harriman and M. J. Stedwell, "FLARE--A Three-Dimensional Boiling Water Reactor Simulator," USAEC Report GEAP-4598, Atomic Power Equipment Department, General Electric Company, July 1964.

(26) Drake, A. W., "Fundamentals of Applied Probability Theory," McGraw-Hill, New York (1967).

(27) Ford, L. R. Jr. & D. R. Fulkerson, "Flows in Networks," Princeton University Press, Princeton, New Jersey (1962).

(28) Gilleland, J. E., W. B. Behnke and M. J. Whitman, "Integrating Nuclear Power into Planning for Power-System Operation," Peaceful Uses of Atomic Energy, Vol. 4, p. 105, Proc. of Fourth International Conf., Geneva, September 1971.

(29) Gilleland, J. E., R. E. Hoskins, W. M. Kiefer and E. F. Koncel, "Fitting Nuclear Capacity into Utility Planning," Proc. Am. Power Conf., April 1970.

(30) Hicks, K. L., "Theory of Economic Selection of Generating Units," AIEE Trans. on Power App. and Sys., PAS-79, pp. 1794-1800 (February 1960).

(31) Hildebrand, F. B., "Advanced Calculus for Applications," Prentice-Hall, Englewood Cliffs, New Jersey, pp. 354, 360 (1962).

(32) Hildebrand, F. B., "Methods of Applied Mathematics," Prentice-Hall, Englewood Cliffs, New Jersey, Second edition, p. 125 (1965).

(33) Hoskins, R. E., N. B. Hughes, W. M. Kiefer and E. F. Koncel, "Nuclear Fuel and Power Management," Trans. Am. Nucl. Soc., 15, 596 (1972).

(34) Hoskins, R. E., E. F. Koncel, S. Strauch, "Planning through Simulation of Operating with Nuclear and Fossil Plants," Proc. Am. Power Conf., April 1972.

(35) Houser, H. G. and D. A. Nordman, "Nuclear Plant Evaluation More Complex in Phase II," Electrical World, May 23, 1966.

(36) Jenkins, R. T., of TVA, personal communication, September 1970.

(37) Joy, D. S., "A Dynamic Programming Solution of the Electric Generating System Unit Commitment Problem," ORNL-TM-2881 (April 1970).

(38) Joy, D. S., "A Dynamic Programming Approach for Determining the Optimal Dispatching Sequence for a Combined Hydro-Thermal-Pumped-Storage Utility System," ORNL-4624 (November 1970).

(39) Joy, D. S. and R. T. Jenkins, "A Probabilistic Model for Estimating the Operating Cost of an Electric Power Generating System," ORNL-TM-3549 (October 1971).

(40) Kearney, J. P., "Summary Description of the Fuel Depletion Code CELL-MOVE," MIT Dept. of Nuclear Engineering, August 1971.

(41) Kearney, J. P., "Simulation and Optimization Techniques of Nuclear In-core Fuel Management Decisions," Ph.D. Thesis, Dept. of Nuclear Engineering, MIT, January 1973.

(42) Kiefer, W. M. and E. F. Koncel, "Scheduling Generation on Systems with Fossil and Nuclear Units," Trans. Am. Nucl. Soc., 13, 768 (1970).

(43) Kirchmayer, L. K., "Economic Operation of Power Systems," Wiley, New York, p. 40 (1958).

(44) Koncel, E. F., to M. Benedict, personal communication, October 11, 1972.

(45) Levin, A., "IBM/360 Out of Kilter Network Flow Routine: Description for the User," Flight Transportation Laboratory, MIT, August 1967.

(46) Mason, E. A., "An Overall View of the Nuclear Fuel Cycle," Nuclear News, 14, No. 2, 35, February 1971.

(47) McCoy, N. H., "Introduction to Modern Algebra," Allyn and Bacon, Boston (1968).

(48) Noakes, F. and A. Arismunandar, "Bibliography on Optimum Operation of Power Systems: 1919-1959," AIEE Trans. on Power App. and Sys.," PAS-82, pp. 864-871 (February 1963).

(49) Peschon, J., "Program Assesses Fuel Cost and Security," Electrical World, p. 42, May 1, 1972.

(50) Poncelet, C. G., "LASER--A Depletion Program for Lattice Calculations Based on MUFT and THERMOS," USAEC Report WCAP-6073, Westinghouse Electric Corporation, September 1963.

(51) Rees, F. J. and R. E. Larson, "Computer-Aided Dispatching and Operations Planning for an Electric Utility with Multiple Types of Generation," IEEE Trans. on Power App. and Sys., PAS-90, pp. 891-899 (March/April 1971).

(52) Skrotski, B. G. A., and W. A. Vopat, "Power Station Engineering and Economy," McGraw-Hill, New York (1960).

(53) Steinberg, M. J. and T. H. Smith, "Economic Loading of Steam Power Plants and Electric Systems," Wiley, New York (1943).

(54) Wagner, H. M., "Principles of Operations Research," Prentice-Hall, Englewood Cliffs, New Jersey, pp. 96, 551, 580 (1969).

(55) Watt, H. Y., "Incremental Costs and Optimization of In-core Fuel Management of Nuclear Power Plants," Ph.D. Thesis, Dept. of Nuclear Engineering, MIT, June 1973.

(56) Weiss, L. W., "Case Studies in American Industry," Wiley, New York, p. 89 (1967).

(57) Widmer, H., "The Optimal Utilization of a Nuclear Power Plant with Fixed Fueling Schedule," Ph.D. Thesis, Dept. of Nuclear Engineering, MIT, September 1970.

(58) Widmer, H., E. A. Mason and P. W. MacAvoy, "Mid-Range Optimization of a Nuclear Power Plant," Trans. Am. Nucl. Soc., 13, 784 (1970).

(59) Widmer, H., E. A. Mason and P. W. MacAvoy, "Short-Range Optimization of Use of a Nuclear Power Plant," Trans. Am. Nucl. Soc., 13, 785 (1970).

BIOGRAPHY OF Paul F. Deaton

PERSONAL INFORMATION

Birth date: June 22, 1944 — 5' 8"; 170 pounds
Married; Two children — Health: excellent

VOCATIONAL OBJECTIVE: Employment in the area of nuclear economics and system analysis, particularly in power management, fuel management or fuel cycle analysis.

EDUCATION

Massachusetts Institute of Technology, Cambridge, Massachusetts
Currently completing PhD thesis in the Department of Nuclear Engineering: "A System Integration and Optimization Model for Nuclear Power Management", Adviser: Professor E.A. Mason. Working closely with large Midwest utility sponsoring the research, thesis involves application of operations research methods to simulate and optimize their electric generating system which possesses both fossil and nuclear power plants. Graduate courses include nuclear physics, reactor engineering, reactor physics, nuclear chemical engineering, economics of nuclear power and space applications of nuclear energy. Minor includes managerial accounting, financial management and economics of fuel and power.

Cumulative grade average: 5.0 (5.0 = A)
Expected degree and date: PhD; January 1973
AEC Special Fellowships in Nuclear Science (1967, 1968, 1969).
John & Fannie Hertz Foundation Fellowships (1970, 1971, 1972).
Member: American Nuclear Society; President, Westgate Community Association (Married students in MIT housing).

University of Cincinnati, Cincinnati, Ohio
Majored in Chemical Engineering. Courses included process economics and control, physical and chemical rate processes, material and energy balances, organic and physical chemistry, and modern physics.

Rank in class: 1st in 223
Cumulative grade average: 3.91 (4.0 = A)
Degree and date: B.S.Ch.E. (with High Honors); June 1967
Four scholarships
Member: American Institute of Chemical Engineers, Varsity Baseball
Honoraries: Vice-president, Tau Beta Pi; Omicron Delta Kappa; Phi Eta Sigma; Phi Lambda Upsilon.

Tecumseh High School, New Carlisle, Ohio: Diploma, 1962
(Valedictorian) Majored in College Preparatory Course.

WORK EXPERIENCE

Stone & Webster Engineering Corporation, Boston, Mass. (1969): Summer work in Nuclear Division's new fuel management group adapting computer codes to in-house IBM/360 computer. Familiar with: 2DB, ANISN, FLARE, CELL-MOVE, GGC-3, CINCAS and COBRA. Programming capabilities include Fortran II, Fortran IV, BASIC, MAD, Assembler and Job Control languages.

Raphael Katzen Associates, Cincinnati, Ohio (1967): Part-time work during senior year of college, organizing the data-files and design notebooks of this chemical engineering consulting firm.

Bauer Bros. Company, Springfield, Ohio (1962 to 1966): Co-operative work experience included 10 months in engineering department as draftsman and checker, 7 months as metallurgical laboratory technician using classical methods of analysis, and 7 months as a technician in an industrial demonstration laboratory.

PUBLICATIONS

"A System Integration and Optimization Model for Nuclear Power Management Planning," P.F. Deaton and E.A. Mason, Trans. Am. Nucl. Soc., 15, 373 (1972).

"Parallel Derivation of Marginal Costs Pertinent to Utility Optimization," P.F. Deaton and E.A. Mason, Trans. Am. Nucl. Soc., 15, 375 (1972).

MILITARY STATUS: First Lieutenant in U.S. Army Reserves with 3-year Ready Reserve obligation remaining.

BACKGROUND AND INTERESTS: Born and raised in small Ohio town; active in sports; hobbies include bridge, chess and flying.

```
C*************************************************************************
C*                                                                       *
C*    S Y S I N T   :   AN ELECTRIC UTILITY SYSTEM INTEGRATION MODEL    *
C*                         WRITTEN BY PAUL F. DEATON                     *
C*                   M.I.T. DOCTORAL THESIS,    MARCH  1973              *
C*                                                                       *
C*************************************************************************
C     MAIN PROGRAM
C     SYSINT VERSION  1-01-73
      COMMON/INTEGR/RD,WT
      INTEGER RD,WT
      CALL STRTIM
      WRITE(WT,900)
      CALL SUPSIM
      STOP
  900 FORMAT(T31,72('*')/T31,'*',T102,'*'/T31,'*',T37,'S Y S I N T   :
     $ AN ELECTRIC UTILITY SYSTEM INTEGRATION MODEL',T102,'*'/
     $T31,'*',T64,'WRITTEN BY PAUL F. DEATON',T102,'*'/
     $T31,'*',T58,'M.I.T. DOCTORAL THESIS,    MARCH  1973',T102,'*'/
     $T31,'*',T102,'*'/T31,72('*')//
     $T56,'VERSION  1-01-73')
      END
      BLOCK DATA
C     SYSINT VERSION 10-15-71
C     INITIALIZES CONSTANT DATA IN COMMON AREAS
C     ***************************************************************
      IMPLICIT REAL*8 (A-H,O-$)
C     COMMON VARIABLES
C     VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS
      COMMON/PLTDAT/IDNO(100),NAME(100),TYPE(100),SUSDHT(100),PNOM(100),
     $NPTS(100),MWPT(5,100),HTRAT(5,100)
      COMMON/PERDAT/AVLBTY(100),CSTBTU(100),STATUS(100),EXPHRS(100),
     $EXPBTU(100),EXPGWH(100),NORDER(500),COST(100),ENERGY(100),
     $SUPCST(100),MRGCST(5,100)
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
```

```
      COMMON/FLOAT/EPS,TRACE,PKMW,SPNRES,CSTEMR
      CCMMON/TITLE/SGTITL(10),PDTITL(10)
      COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1
     $,IDSTRG,PCHMIN,PCHMAX,MBRNUM
      CCMMON/LDGNFO/LDTYPE,LDTYPS,LOAD(50,25),NORDOP,NOENTY,NOBASE,
     $NOPEAK,NNORD
      COMMON/MAXMUM/IDIMEN,MAXPLT,MAXPER,MAXNPT
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI
      COMMON/LOGICL/MINI,MIDI,MAXI,NPM,PCHING
      COMMON/SUSDF/F(20)
      COMMON/MAINT/MAINT(100,20)
C     MAINT IS DIMENSIONED (MAXPLT,MAXPER/5)  THE 5 IS 5I1/INTEGER*2
      COMMON/MURGER/CTEMP(500),NEWCOD(5),NEWCST(5),MPTS,IFRST,ILAST
C     NEWCST & NEWCOD ARE DIMENSIONED MAXNPT;CTEMP (MAXPLT*MAXNPT)
      REAL*4 SUSDHT,PNOM,HTRAT
      REAL*4 SUPCST,MRGCST
      REAL*4 CTEMP,NEWCST
      REAL*8 MILLI
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX
      INTEGER*4 NEWCOD
      INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD
      LOGICAL*1 MINI,MIDI,MAXI,NPM,PCHING
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES
      REAL*8 EPS/1.D-3/
      REAL*8 TRACE/1.D-10/
      INTEGER RD/5/,WT/6/,CARD/7/,TAPE/8/
      INTEGER IDIMEN/500/
      INTEGER MAXPLT/100/
      INTEGER MAXPER/100/
      INTEGER MAXNPT/5/
      INTEGER NPER1/1/
      REAL*8 ZERO/0.0D0/,ONE/1.0D0/,TWO/2.0D0/,HALF/0.5D0/,TEN/1.D1/,
     $TENTH/1.D-1/,HUNDRD/1.D2/,CENTI/1.D-2/,THOUS/1.D3/,MILLI/1.D-3/
      END
      SUBROUTINE SUPSIM
C     SUPERVISOR OF ENTIRE SYSINT SIMULATION
```

```
C     SYSINT VERSION 11-2-71                                            SINT0073
C     ************************************************************      SINT0074
C     DEFINITION OF IMPORTANT VARIABLES ******************************   SINT0075
C     AVLBTY = PERFORMANCE PROBABILITY (PER CENT)                        SINT0076
C     CARD   = UNIT NUMBER FOR COMPUTER CARD PUNCH DEVICE                SINT0077
C     COST   = EXPECTED COST (DOLLARS)                                   SINT0078
C     CSTBTU = COST OF FUEL (CENTS/MEGABTU)                              SINT0079
C     CSTEMR = COST OF EMERGENCY ENERGY PURCHASES ($/MWH)                SINT0080
C     DAYS   = DURATION OF PERIOD (DAYS)                                 SINT0081
C     DM     = EQUIVALENT LOAD CURVE SPACING (MW)                        SINT0082
C     DT     = DURATION OF PERIOD (HOURS)                                SINT0083
C     EMRP$  = TOTAL COST OF EMERGENCY ENERGY PURCHASES (DOLLARS)        SINT0084
C     ENERGY = ENERGY AVAILABLE AS A SCARCE RESOURCE (GWH)               SINT0085
C     EPS    = MINIMUM SEPARATION OF IEMAX*DM AND PEMAX (MW)             SINT0086
C     ERRCOD = ACCUMULATED ERROR CODE                                    SINT0087
C     EXPBTU = EXPECTED FUEL CONSUMPTION (MEGABTU)                       SINT0088
C     EXPDEM = EXPECTED ENERGY DEMAND (GWH)                              SINT0089
C     EXPEMR = EXPECTED EMERGENCY ENERGY PURCHASES (GWH)                 SINT0090
C     EXPGEN = EXPECTED SYSTEM GENERATION (GWH)                          SINT0091
C     EXPGWH = EXPECTED PLANT  GENERATION (GWH)                          SINT0092
C     EXPHRS = EXPECTED HOURS OF OPERATION                               SINT0093
C     F      = NORMALIZED STARTUP-SHUTDOWN FREQUENCY FUNCTION (PER DAY)  SINT0094
C     GWHPER = ENERGY PER UNIT AREA UNDER LOAD CURVE (GWH) = DM*DT/1000  SINT0095
C     HTRAT  = INCREMENTAL HEAT RATE (BTU/KWH)                           SINT0096
C     IDIMEN = MAXIMUM NUMBER OF POINTS ALLOWED IN PROB ARRAY            SINT0097
C     IDNO   = PLANT IDENTIFICATION NUMBER                               SINT0098
C     IDSTRG = STRATEGY ID                                               SINT0099
C     IEMAX  = PROB ARRAY LOCATION OF MAXIMUM LOAD                       SINT0100
C     IEMIN  = PROB ARRAY LOCATION OF MINIMUM LOAD                       SINT0101
C     INDEX  = SEQUENTIAL ORDER OF PLANT AS READ IN                      SINT0102
C     LDTYPE = TYPE OF LOAD CURVE TO BE USED IN THIS PERIOD              SINT0103
C     LDTYPS = TOTAL NUMBER OF LOAD CURVES INPUT                         SINT0104
C     LOAD   = NORMALIZED LOAD-DURATION CURVES (10**-4)                  SINT0105
C     MAINT  = NUMERICALLY-PACKED MAINTENANCE STATUS                     SINT0106
C     MAXI   = OPTION FOR MAXIMUM PRINTOUT                               SINT0107
C     MAXNPT = MAXIMUM NUMBER OF VALVE POINTS ALLOWED                    SINT0108
```

```
C      MAXPER = MAXIMUM NUMBER OF PERIODS ALLOWED                          SINT0109
C      MAXPLT = MAXIMUM NUMBER OF PLANTS ALLOWED                           SINT0110
C      MIDI   = OPTION FOR MEDIUM VOLUME PRINTOUT                          SINT0111
C      MINI   = OPTION FOR MINIMUM PRINTOUT                                SINT0112
C      MRGCST = MARGINAL COST ($/MWH)                                      SINT0113
C      MWPT   = VALVE POINT RATING (MW)                                    SINT0114
C      NAME   = PLANT NAME                                                 SINT0115
C      NNORD  = NUMBER OF VALVE POINTS USED IN NORDER                      SINT0116
C      NOBASE = NUMBER OF ENTRIES IN NORDER IN BASE PORTION                SINT0117
C      NOENTY = NUMBER OF ENTRIES TO NORDER                                SINT0118
C      NOPEAK = NUMBER OF ENTRIES IN NORDER TREATED AS PEAKERS             SINT0119
C      NORDER = LOADING ORDER CODED AS 1000*NPT+INDEX                      SINT0120
C      NORDOP = STARTUP ORDER OPTION DESIRED                               SINT0121
C      NOSTNS = NUMBER OF STATIONS FOR WHICH DATA READ IN                  SINT0122
C      NPER   = NUMBER OF THIS PERIOD                                      SINT0123
C      NPERS  = TOTAL NUMBER OF PERIODS READ IN                            SINT0124
C      NPER1  = ASSOCIATED VARIABLE FOR DIRECT ACCESS DEVICE;  NPER1=NPER  SINT0125
C      NPM    = NUCLEAR POWER MANAGEMENT CPTION                            SINT0126
C             = (.TRUE.=N.P.M. PROBLEM,  .FALSE.=SIMULATION ONLY)          SINT0127
C      NPTS   = NUMBER OF VALVE POINTS OR CAPACITY INCREMENTS              SINT0128
C      PCHMAX = NORDER POINT WHEN PROB PUNCHED AT MAX.NUKES                SINT0129
C      PCHMIN = NORDER POINT WHEN PROB PUNCHED AT MIN.NUKES                SINT0130
C      PDTITL = PERIOD TITLE                                               SINT0131
C      PEMAX  = MAXIMUM EQUIVALENT LOAD (MW)                               SINT0132
C      PEMIN  = MINIMUM EQUIVALENT LOAD (MW)                               SINT0133
C      PKMW   = FORECAST PEAK LOAD FOR THE PERIOD (MW)                     SINT0134
C      PNCM   = PLANT NOMINAL AVAILABILITY FRACTION                        SINT0135
C      PROB   = EQUIVALENT LOAD CDF                                        SINT0136
C      PROD$  = TOTAL SYSTEM PRODUCTION FUEL COST (DOLLARS)                SINT0137
C      PUNCH  = OUTPUT DEVICE TO BE USED FOR PUNCHED OUTPUT                SINT0138
C      RD     = UNIT NUMBER OF COMPUTER INPUT READING DEVICE               SINT0139
C      SGTITL = STRATEGY TITLE                                             SINT0140
C      SPNRES = SPINNING RESERVE REQUIREMENT (MW)                          SINT0141
C      STATUS = MAINTENANCE STATUS                                         SINT0142
C             = (0=NON-EXISTENT,1=DOWN,2=ON-LINE)                          SINT0143
C      SUPCST = STARTUP-SHUTDOWN COST (DOLLARS)                            SINT0144
```

PAGE 4

```
C      SUSDHT = PLANT STARTUP & SHUTDOWN HEAT REQUIREMENT (MEGABTU)
C      SUSD$  = TOTAL SYSTEM STARTUP-SHUTDOWN COST (DOLLARS)
C      TAPE   = UNIT NUMBER FOR COMPUTER TAPE DEVICE
C      TOTAL$ = TOTAL SYSTEM COST (DOLLARS)
C      TRACE  = LOWER LIMIT OF PROB PROCESSING
C      TYPE   = PLANT TYPE
C             = (F=FOSSIL,H=HYDRO,N=NUCLEAR,P=PEAKING,S=PUMPED-STORAGE)
C      WT     = UNIT NUMBER OF COMPUTER OUTPUT PRINTING DEVICE
C      END OF DEFINITIONS ***********************************************
       IMPLICIT REAL*8 (A-H,O-$)
C      COMMON VARIABLES
C      VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS
       COMMON/PLTDAT/IDNO(100),NAME(100),TYPE(100),SUSDHT(100),PNOM(100),
      $NPTS(100),MWPT(5,100),HTRAT(5,100)
       COMMON/PERDAT/AVLBTY(100),CSTBTU(100),STATUS(100),EXPHRS(100),
      $EXPBTU(100),EXPGWH(100),NORDER(500),COST(100),ENERGY(100),
      $SUPCST(100),MRGCST(5,100)
C      OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
       COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
       COMMON/FLOAT/EPS,TRACE,PKMW,SPNRES,CSTEMR
       COMMON/TITLE/SGTITL(10),PDTITL(10)
       COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1
      $,IDSTRG,PCHMIN,PCHMAX,MBRNUM
       COMMON/LDGNFO/LDTYPE,LDTYPS,LOAD(50,25),NORDOP,NOENTY,NOBASE,
      $NOPEAK,NNORD
       COMMON/MAXMUM/IDIMEN,MAXPLT,MAXPER,MAXNPT
       COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI
       COMMON/LOGICL/MINI,MIDI,MAXI,NPM,PCHING
       COMMON/SUSDF/F(20)
       COMMON/MAINT/MAINT(100,20)
C      MAINT IS DIMENSIONED (MAXPLT,MAXPER/5)  THE 5 IS 5I1/INTEGER*2
       COMMON/MURGER/CTEMP(500),NEWCOD(5),NEWCST(5),MPTS,IFRST,ILAST
C      NEWCST & NEWCOD ARE DIMENSIONED MAXNPT;CTEMP (MAXPLT*MAXNPT)
       REAL*4 SUSDHT,PNOM,HTRAT
       REAL*4 SUPCST,MRGCST
       REAL*4 CTEMP,NEWCST
```

```
      REAL*8 MILLI                                                      SINT0181
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX                SINT0182
      INTEGER*4 NEWCOD                                                  SINT0183
      INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD            SINT0184
      LOGICAL*1 MINI,MIDI,MAXI,NPM,PCHING                               SINT0185
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES                   SINT0186
      DATA $SUPSI/'SUPSIM'/                                             SINT0187
      INTEGER KEYWRD(7)/'STAR','SAVE','OUTP','PERI','STRA','COMP','STOP'SINT0188
     $/,$PRIN$/'PRIN'/,$CARD$/'CARD'/,$TAPE$/'TAPE'/,$MINI$/'MINI'/,    SINT0189
     $$MIDI$/'MIDI'/,$MAXI$/'MAXI'/,$BSOM$/' SOM'/                      SINT0190
      LOGICAL*1 DOPERD(100)                                             SINT0191
C     DOPERD DIMENSIONED BY MAXPER                                      SINT0192
      DEFINE FILE 9(100,1000,U,NPER1)                                   SINT0193
C     IN DEFINE FILE STATEMENT,    100 IS MAXPER & 1000 IS 10*MAXPLT    SINT0194
      MINI=.TRUE.                                                       SINT0195
      ASSIGN 10 TO NEXT                                                 SINT0196
      ERRCOD=0                                                          SINT0197
   10 MIDI=.TRUE.                                                       SINT0198
   15 KEY=0                                                             SINT0199
   20 KEY=KEY+1                                                         SINT0200
   30 READ(RD,900) KEY1,I,KEY2,J,KEY3                                   SINT0201
      WRITE(WT,910) $SUPSI,KEY1,I,KEY2,J,KEY3                           SINT0202
   40 IF(KEY1.EQ.KEYWRD(KEY)) GO TO (50,20,60,80,90,100,140),KEY        SINT0203
      KEY=KEY+1                                                         SINT0204
      IF(KEY.GE.8) CALL ERRMSG('SUPSIM',6)                              SINT0205
      GO TO 40                                                          SINT0206
C     START    CONTROL CARD READ                                        SINT0207
   50 ERRCOD=0                                                          SINT0208
      CALL CMPTIM('SUPSIM','BASIC ')                                    SINT0209
      CALL BASIC                                                        SINT0210
      CALL CMPTIM('BASIC ','SUPSIM')                                    SINT0211
      GO TO 20                                                          SINT0212
   60 IF(KEY2.EQ.$PRIN$) GO TO 70                                       SINT0213
C     OUTPUT TAPE  OR  OUTPUT CARD   CONTROL CARD READ                  SINT0214
      PUNCH=0                                                           SINT0215
      IF(KEY2.EQ.$TAPE$) PUNCH=TAPE                                     SINT0216
```

```
      IF(KEY2.EQ.$CARD$) PUNCH=CARD
      PCHING=PUNCH.GT.0
      GO TO 30
C     OUTPUT PRINT   CONTROL CARD READ
   70 MIDI=.FALSE.
      MAXI=.FALSE.
      IF(KEY3.EQ.$MAXI$.OR.KEY3.EQ.$MIDI$) MIDI=.TRUE.
      IF(KEY3.EQ.$MAXI$) MAXI=.TRUE.
      GO TO 30
C     PERIOD   CONTROL CARD READ
   80 DO 85 I=1,NCSTNS
   85 AVLBTY(I)=HUNDRD*PNOM(I)
      CALL ERASE(CSTBTU,2*MAXPLT,ENERGY,2*MAXPLT,NORDER,MAXPLT*MAXNPT/2)
      IF(MIDI) CALL CMPTIM('SUPSIM','PERIOD')
      CALL PERIOD
      IF(MIDI) CALL CMPTIM('PERIOD','SUPSIM')
C     STRATEGY CONTROL CARD READ
   90 IF(MIDI) CALL CMPTIM('SUPSIM','STRATG')
      CALL STRATG
      IF(MIDI) CALL CMPTIM('STRATG','SUPSIM')
      GO TO 20
C     COMPUTE  CONTROL CARD READ
  100 IF(KEY2.NE.$BSOM$) GO TO 104
      READ(RD,915) (DOPERD(J),J=1,NPERS)
      DO 102 N=1,NPERS
      IF(DOPERD(N)) WRITE(WT,916) N
  102 CONTINUE
      GO TO 108
  104 DO 106 N=1,NPERS
  106 DOPERD(N)=.TRUE.
  108 KEY2=ERRCOD
      CALL CMPTIM('      ','COMPUT')
C     WRITE BASIC PLANT INFO FOR THIS STRATEGY
      WRITE(WT,920)
      WRITE(WT,930)  IDSTRG,SGTITL,NPM,MBRNUM
      IF(NPM) WRITE(WT,935)
```

```
      IF(PCHING) CALL PUNCHR(1)
      WRITE(WT,940) NOSTNS
      WRITE(WT,950)(J,IDNO(J),NAME(J),MWPT(NPTS(J),J),TYPE(J),SUSDHT(J),
     $PNCM(J),NPTS(J),(MWPT(I,J),HTRAT(I,J),I=1,MAXNPT),J=1,NOSTNS)
      WRITE(WT,970)(I,I=1,9)
      KEY1=(NPERS+4)/5
      DO 110 I=1,NOSTNS
  110 WRITE(WT,971) I,IDNO(I),(MAINT(I,J),J=1,KEY1)
      WRITE(WT,960) F
      IF(PUNCH.LT.0)  GO TO 130
      ASSIGN 120 TO NEXT
      DO 120 N=1,NPERS
      IF(.NOT.DOPERD(N)) GO TO 120
      NPER=N
      NPER1=NPER
      ERRCOD=KEY2
      IF(MIDI) CALL CMPTIM('SUPSIM','PRESIM')
      CALL PRESIM
      IF(MIDI) CALL CMPTIM('PRESIM','SUPSIM')
      IF(PCHING) CALL PUNCHR(5)
      IF(.NOT.MINI) GO TO 135
  120 CONTINUE
      ASSIGN 10 TO NEXT
      ERRCOD=KEY2
  130 CALL CMPTIM('COMPUT','      ')
      GO TO 15
      ENTRY QUIT
  135 IF(PCHING) CALL PUNCHR(6)
      GO TO NEXT,(10,120)
C     STOP     CCNTROL CARD READ
  140 CALL ERRMSG('SUPSIM',8)
      RETURN
  900 FORMAT(2(A4,A3),3A4)
  910 FORMAT(/T12,'KEY1   KEY2   KEY3'/2X,A6,' : ',2(A4,A3),3A4)
  915 FCRMAT(80L1)
  916 FORMAT(' SIMULATE PERIOD',I4)
```

```
  920 FORMAT('1'/30('0'/),4(' ',132('0')/'+',132('*')/)                      SINT0289
     $         /30('0'/),6(' ',132('0')/'+',132('*')/))                      SINT0290
  930 FORMAT('0STRATEGY ID = ',I6,5X,'TITLE :"',10A7,'"',3X,                 SINT0291
     $'PUNCH NAME=',L1,I7)                                                   SINT0292
  935 FORMAT('0',T25,'* * * * *  N U C L E A R   P O W E R   M A N A G',     SINT0293
     $' E M E N T   S T U D Y  * * * * *')                                   SINT0294
  940 FORMAT('0','                         PLANT DATA FOR',I4,' STATIONS'//  SINT0295
     $' INDEX  IDNO  NAME  MAXMW  TYPE  SUSDHT(MEGABTU)  PNOM  NPTS',        SINT0296
     $   '  MWPT(I,INDEX),HTRAT(I,INDEX),I=1,NPTS  ',                        SINT0297
     $'MWPT IN MW & HTRAT IN BTU/KWH'/)                                      SINT0298
  950 FORMAT((I4,I8,A6,I6,5X,A1,F14.2,F11.5,I3,5(2X,I4,F7.0)))               SINT0299
  960 FORMAT(//'                       NORMALIZED STARTUP & SHUTDOWN',       SINT0300
     $' FUNCTION :'/(8F10.6))                                                SINT0301
  970 FORMAT(//,T20,' MAINTENANCE STRATEGY BY PERIOD AND INDEX',             SINT0302
     $'  (0=NON-EXISTENT;1=DOWN;2=ON-LINE)'//T115,'1',T62,'PERIOD'/          SINT0303
     $15X,9I10,9X,'0'/' INDEX IDNO',4X,10('1234567890')/)                    SINT0304
  971 FORMAT(I4,I7,4X,20I5)                                                  SINT0305
      END                                                                    SINT0306
      SUBROUTINE BASIC                                                       SINT0307
C     SYSINT VERSION 10-31-71                                                SINT0308
C     READS BASIC SYSTEM INFORMATION                                         SINT0309
C     ***********************************************************            SINT0310
      IMPLICIT REAL*8 (A-H,O-$)                                              SINT0311
C     COMMON VARIABLES                                                       SINT0312
C     VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS      SINT0313
      COMMON/PLTDAT/IDNO(100),NAME(100),TYPE(100),SUSDHT(100),PNOM(100),     SINT0314
     $NPTS(100),MWPT(5,100),HTRAT(5,100)                                     SINT0315
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES                          SINT0316
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)        SINT0317
      COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1     SINT0318
     $,IDSTRG,PCHMIN,PCHMAX,MBRNUM                                           SINT0319
      COMMON/LDGNFO/LDTYPE,LDTYPS,LOAD(50,25),NORDOP,NOENTY,NOBASE,          SINT0320
     $NOPEAK,NNORD                                                           SINT0321
      COMMON/MAXMUM/IDIMEN,MAXPLT,MAXPER,MAXNPT                              SINT0322
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI     SINT0323
      COMMON/SUSDF/F(20)                                                     SINT0324
```

```
      REAL*4 SUSDHT,PNOM,HTRAT                                          SINT0325
      REAL*8 MILLI                                                      SINT0326
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX                SINT0327
      INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD            SINT0328
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES                   SINT0329
      DATA $BASIC/' BASIC'/                                             SINT0330
      INTEGER $PLAN$/'PLAN'/,$NORM$/'NORM'/,$LOAD$/'LOAD'/,             SINT0331
     $  $SAVE$/'SAVE'/                                                  SINT0332
   10 READ(RD,900) KEY1,(PROB(I),I=1,6)                                 SINT0333
      WRITE(WT,910) $BASIC,KEY1,(PROB(I),I=1,6)                         SINT0334
      IF(KEY1.EQ.$PLAN$) GO TO 20                                       SINT0335
      IF(KEY1.EQ.$NORM$) GO TO 50                                       SINT0336
      IF(KEY1.EQ.$LOAD$) GO TO 60                                       SINT0337
      IF(KEY1.EQ.$SAVE$) RETURN                                         SINT0338
      CALL ERRMSG(' BASIC',6)                                           SINT0339
C     READ PLANT DATA                                                   SINT0340
   20 READ (RD,920) NOSTNS                                              SINT0341
      WRITE(WT,930) NOSTNS                                              SINT0342
      READ (RD,940)(IDNO(J),NAME(J),TYPE(J),SUSDHT(J),PNOM(J),NPTS(J),  SINT0343
     $(MWPT(I,J),HTRAT(I,J),I=1,MAXNPT),J=1,NOSTNS)                     SINT0344
      DO 40 J=1,NOSTNS                                                  SINT0345
      I=NPTS(J)                                                         SINT0346
   30 IF(I.EQ.MAXNPT) GO TO 40                                          SINT0347
      I=I+1                                                             SINT0348
      MWPT(I,J)=30000                                                   SINT0349
      HTRAT(I,J)=1.E20                                                  SINT0350
      GO TO 30                                                          SINT0351
   40 CONTINUE                                                          SINT0352
      WRITE(WT,950)(J,IDNO(J),NAME(J),MWPT(NPTS(J),J),TYPE(J),SUSDHT(J),SINT0353
     $PNOM(J),NPTS(J),(MWPT(I,J),HTRAT(I,J),I=1,MAXNPT),J=1,NOSTNS)     SINT0354
      I=PRPNDX(J)                                                       SINT0355
      GO TO 10                                                          SINT0356
C     READ NORMALIZED STARTUP & SHUTDOWN FUNCTION                       SINT0357
   50 READ(RD,960) F                                                    SINT0358
      WRITE(WT,970) F                                                   SINT0359
      GO TO 10                                                          SINT0360
```

```
C       READ LOAD TYPES
   60 TEMP=TEN**4
      READ(RD,920) LDTYPS
      WRITE(WT,980) LDTYPS
      IF(LDTYPS.GT.25) CALL ERRMSG('BASIC',6)
      DO 90 I=1,LDTYPS
      READ(RD,920)  LDTYPE,NUMONE
      WRITE(WT,921) LDTYPE,NUMONE
      IF(NUMONE.LE.0) GO TO 75
      DO 70 J=1,NUMONE
   70 PROB(J)=ONE
   75 KEY1=NUMONE+1
      READ(RD,960)(PROB(J),J=KEY1,50)
      WRITE(WT,990)(PROB(J),J=1,50)
      IF(PROB(50).GT.ZERO) WRITE(WT,991)
C       STORE LOAD TYPES IN UNITS OF 10**-4 (SAVES STORAGE)
      DO 80 J=1,50
   80 LOAD(J,LDTYPE)=PROB(J)*TEMP+HALF
   90 CONTINUE
      GO TO 10
  900 FORMAT(2(A4,A3),3A4)
  910 FORMAT(//T12,'KEY1   KEY2   KEY3'/2X,A6,' : ',2(A4,A3),3A4)
  920 FORMAT(16I5)
  921 FORMAT(/,2I5)
  930 FORMAT('1','BASIC NOW READING PLANT DATA FOR',I4,' STATIONS'//
     $' INDEX  IDNO  NAME  MAXMW  TYPE  SUSDHT(MEGABTU)  PNOM  NPTS',
     $   '  MWPT(I,INDEX),HTRAT(I,INDEX),I=1,NPTS  ',
     $'MWPT IN MW & HTRAT IN BTU/KWH'/)
  940 FORMAT((I4,A4,1X,A1,F10.0,F9.5,I1,5(I4,F6.0)))
  950 FORMAT((I4,I8,A6,I6,5X,A1,F14.2,F11.5,I3,5(2X,I4,F7.0)))
  960 FORMAT(8F10.4)
  970 FORMAT (/' BASIC NOW READING NORMALIZED STARTUP & SHUTDOWN',
     $' FUNCTION :'/(8F10.6))
  980 FORMAT (/' BASIC NOW READING',I3,' LOAD TYPES'/' LDTYPE NUMONES'/)
  990 FORMAT(10F10.4)
  991 FORMAT('+',T104,'<---- PRESIM WILL LINEARIZE'/
```

```
$ T110,'THIS NON-ZERO END POINT')
      END
      SUBROUTINE PERIOD
C     SYSINT VERSION 10-29-71
C     READS PERIOD DATA AND STORES IT ON DIRECT ACCESS DEVICE
C     ***********************************************************
      IMPLICIT REAL*8 (A-H,O-$)
C     COMMON VARIABLES
C     VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS
      COMMON/PLTDAT/IDNO(100),NAME(100),TYPE(100),SUSDHT(100),PNOM(100),
     $NPTS(100),MWPT(5,100),HTRAT(5,100)
      COMMON/PERDAT/AVLBTY(100),CSTBTU(100),STATUS(100),EXPHRS(100),
     $EXPBTU(100),EXPGWH(100),NORDER(500),COST(100),ENERGY(100),
     $SUPCST(100),MRGCST(5,100)
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
      COMMON/FLOAT/EPS,TRACE,PKMW,SPNRES,CSTEMR
      COMMON/TITLE/SGTITL(10),PDTITL(10)
      COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1
     $,IDSTRG,PCHMIN,PCHMAX,MBRNUM
      COMMON/LDGNFO/LDTYPE,LDTYPS,LOAD(50,25),NORDOP,NOENTY,NOBASE,
     $NOPEAK,NNORD
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI
      REAL*4 SUSDHT,PNOM,HTRAT
      REAL*4 SUPCST,MRGCST
      REAL*8 MILLI
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX
      INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES
      REAL*8 BTUCST(3)
      INTEGER*2 TEST(3)/'F','N','P'/
      LOGICAL*1 CHGCST(3),CHGAVL
      EQUIVALENCE (BTUCST,CSTFOS),(BTUCST(2),CSTNUK),(BTUCST(3),CSTPKG)
      DATA $STRAT,$PERIO,$SUSDB,$ALTER/'STRAT','PERIO','SUSD ','ALTER'/
      DATA STARS/1.D50/
      WRITE(WT,900)
```

```
      NPERS=0
C     PERIOD CONTROL CARD READ
   10 READ(RD,910) PDTITL
      READ(RD,920) NPER,LDTYPE,PKMW,SPNRES,DM,DT,CSTEMR,CSTFOS,CSTNUK,
     $CSTPKG,AVLALL
      NPERS=NPERS+1
      IF(SPNRES.LT.ZERO) SPNRES=ZERO
      DO 12 K=1,3
      CHGCST(K)=BTUCST(K).GT.ZERO
      IF(.NOT.CHGCST(K)) BTUCST(K)=STARS
   12 CONTINUE
      CHGAVL=AVLALL.GT.ZERO.AND.AVLALL.LT.HUNDRD+ONE
      IF(.NOT.CHGAVL) AVLALL=STARS
      WRITE(WT,930) PDTITL,NPER,LDTYPE,PKMW,SPNRES,DM,DT,CSTEMR,CSTFOS,
     $CSTNUK,CSTPKG,AVLALL
      IF(.NOT.CHGAVL) GO TO 16
      DO 14 I=1,NOSTNS
   14 AVLBTY(I)=AVLALL
   16 DO 20 K=1,3
      IF(.NOT.CHGCST(K)) GO TO 20
      DO 18 I=1,NOSTNS
      IF(TYPE(I).EQ.TEST(K)) CSTBTU(I)=BTUCST(K)
   18 CONTINUE
   20 CONTINUE
   30 READ(RD,940)$KEY1,$KEY2,ID,CST,AVL,ENER
      IF($KEY1.EQ.$STRAT.OR.$KEY1.EQ.$PERIO) GO TO 50
      IF($KEY1.EQ.$SUSDB) GO TO 40
      IF($KEY1.EQ.$ALTER) GO TO 31
      WRITE(WT,950) $KEY1,$KEY2,ID,CST,AVL,ENER
      CALL ERRMSG('PERIOD',6)
C     ALTER CARD WAS READ
   31 INDEX=INNDEX(ID)
      IF(CST.NE.ZERO) GO TO 32
      CST=STARS
      GO TO 33
   32 CSTBTU(INDEX)=CST
```

```
   33 IF(AVL.GT.ZERO) GO TO 34
      AVL=STARS
      GO TO 35
   34 AVLBTY(INDEX)=AVL
   35 IF(ENER.GT.ZERO)GO TO 36
      ENER=STARS
      GO TO 37
   36 ENERGY(INDEX)=ENER
   37 WRITE(WT,950) $KEY1,$KEY2,ID,CST,AVL,ENER
      GO TO 30
C     SUSD DATA CONTROL CARD READ
   40 WRITE(WT,951)$KEY1,$KEY2
      READ(RD,970) NORDOP,NOENTY,NOBASE,NOPEAK
      WRITE(WT,960) NORDOP,NOENTY,NOBASE,NOPEAK
      READ(RD,970) (NORDER(I),I=1,NOENTY)
      WRITE(WT,970)(NORDER(I),I=1,NOENTY)
      GO TO 30
   50 WRITE(WT,980)(I,IDNO(I),NAME(I),CSTBTU(I),AVLBTY(I),ENERGY(I),
     $I=1,NOSTNS)
      WRITE(WT,951)$KEY1,$KEY2
      NPER1=NPER
      WRITE(9'NPER1)PDTITL,NPER,LDTYPE,PKMW,SPNRES,DM,DT,CSTEMR,NORDOP,
     $NOENTY,NOBASE,NOPEAK,CSTBTU,AVLBTY,NORDER,ENERGY
      IF($KEY1.EQ.$PERIO) GO TO 10
C     STRATEGY CONTROL CARD READ
      RETURN
  900 FORMAT('OPERIOD NOW READING PER PERIOD DATA & STORING ON DIRECT'
     $,' ACCESS DEVICE'/)
  910 FORMAT(10A8)
  920 FORMAT(2I4,9F8.0)
  930 FORMAT('1PERIOD TITLE :"',10A8,'"'/T83,'(CENTS PER MEGABTU)'/
     $'   NPER   LDTYPE   PKMW(MW)   SPNRES(MW)   DM(MW)   DT(HRS)',
     $T63,'CSTEMR($/MWH)    CSTFOS   CSTNUK   CSTPKG    AVLALL(%)'/
     $I6,I8,F12.0,F11.0,F12.2,F9.2,F13.3,8X,3(F6.3,3X),F9.4/
     $'OSPECIFIC CHANGES INPUT ON ALTER CARDS :'
     $/T18,'IDNO   CSTBTU    AVLBTY    ENERGY')
```

```
 940 FORMAT(2A5,I10,3F10.4)
 950 FORMAT(' ',2A5,I10,3F10.4)
 951 FORMAT(//T12,'$KEY1$KEY2'/' PERIOD :   ',2A5/)
 960 FORMAT(
    $' STARTUP ORDER OPTION                  = NORDOP =',I5/
    $' NUMBER OF ENTRIES IN NORDER           = NOENTY =',I5/
    $' NUMBER OF ENTRIES IN BASE PORTION     = NOBASE =',I5/
    $' NUMBER OF ENTRIES IN PEAK PORTION     = NOPEAK =',I5/
    $' NORDER(I),I=1,NOENTY :')
 970 FORMAT(16I5)
 980 FORMAT(//' FINAL KEY PERIOD INFO:'/
    $' INDEX  IDNO  NAME     CSTBTU     AVLBTY   ENERGY'/
    $(I4,I8,A6,2X,3F10.4))
     END
     FUNCTION INNDEX(ID)
C    SYSINT VERSION  1-01-73
C    FINDS INDEX CORRESPONDING TO A PARTICULAR IDNO
C    ************************************************************
     IMPLICIT REAL*8 (A-H,O-$)
C    COMMON VARIABLES
C    VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS
     COMMON/PLTDAT/IDNO(100),NAME(100),TYPE(100),SUSDHT(100),PNOM(100),
    $NPTS(100),MWPT(5,100),HTRAT(5,100)
C    OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
     COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1
    $,IDSTRG,PCHMIN,PCHMAX,MBRNUM
     REAL*4 SUSDHT,PNOM,HTRAT
     INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX
     INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD
C    END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES
     INTEGER*2 ID2NDX(100)
C    DIMENSION 100 ALLOWS FOR ALL TWO-DIGIT NUMBERS
     IF(ID.LT.0.OR.ID.GT.9999) GO TO 20
     IDNO(NOSTNS+1)=ID
     I=ID2NDX(ID/100+1)-1
  10 I=I+1
```

```
      IF(ID.EQ.IDNO(I)) GO TO 30
      GO TO 10
   20 I=NOSTNS+1
   30 IF(I.GT.NOSTNS) GO TO 50
      INNDEX=I
      RETURN
C     PREPARES ID2NDX FOR FASTER SEARCH BY LATER CALLS TO INNDEX
      ENTRY PRPNDX(JDUMMY)
      PRPNDX=JDUMMY
      CALL ERASE(ID2NDX,100/2)
      DO 40 I=1,NOSTNS
      KEYID=IDNO(I)/100+1
      IF(ID2NDX(KEYID).EQ.0) ID2NDX(KEYID)=I
   40 CONTINUE
      RETURN
   50 WRITE (WT,900) ID
      CALL ERRMSG('INNDEX',7)
      INNDEX=I
      RETURN
  900 FORMAT(T10,'INVALID IDNO = ',I10)
      END
      SUBROUTINE STRATG
C     SYSINT VERSION 10-15-71
C     READS STRATEGY INPUT AND FORMS MAINTENANCE CODE
C     ************************************************************
      IMPLICIT REAL*8 (A-H,O-$)
C     COMMON VARIABLES
C     VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS
      COMMON/PLTDAT/IDNO(100),NAME(100),TYPE(100),SUSDHT(100),PNOM(100),
     $NPTS(100),MWPT(5,100),HTRAT(5,100)
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
      COMMON/TITLE/SGTITL(10),PDTITL(10)
      COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1
     $,IDSTRG,PCHMIN,PCHMAX,MBRNUM
      COMMON/MAXMUM/IDIMEN,MAXPLT,MAXPER,MAXNPT
      LOGICAL*1 MINI,MIDI,MAXI,NPM,PCHING
```

```
      COMMON/MAINT/MAINT(100,20)                                        SINT0577
C     MAINT IS DIMENSIONED (MAXPLT,MAXPER/5)  THE 5 IS 5I1/INTEGER*2    SINT0578
      REAL*4 SUSDHT,PNOM,HTRAT                                          SINT0579
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX                SINT0580
      INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD            SINT0581
      COMMON/LOGICL/MINI,MIDI,MAXI,NPM,PCHING                           SINT0582
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES                   SINT0583
      INTEGER*2 M(100),NOTZRO(2),NDOWN(20)                              SINT0584
C     DIMENSION M(MAXPER)                                               SINT0585
      INTEGER $MAIN$/'MAIN'/,$BLANK/'    '/,NOT/'NOT '/                 SINT0586
      READ(RD,910) NPM,IPLACE,IDSTRG,SGTITL                             SINT0587
      WRITE(WT,920)IDSTRG,SGTITL                                        SINT0588
      IF(IPLACE.LE.0) IPLACE=9                                          SINT0589
      MBRNUM=1000000*IPLACE+IDSTRG                                      SINT0590
      IF(IDSTRG.LT.0) MBRNUM=9999999                                    SINT0591
      KEY1=NOT                                                          SINT0592
      IF(NPM) KEY1=$BLANK                                               SINT0593
      WRITE(WT,925) KEY1,NPM,MBRNUM                                     SINT0594
      READ(RD,930) KEY1,KEY2,KEY3                                       SINT0595
      WRITE(WT,940)KEY1,KEY2,KEY3                                       SINT0596
      IF(KEY1.NE.$MAIN$) CALL ERRMSG('STRATG',6)                        SINT0597
      LMAX=(NPERS+4)/5                                                  SINT0598
      CALL ERASE(MAINT,MAXPLT*MAXPER/10)                                SINT0599
      DO 50 I=1,NOSTNS                                                  SINT0600
      READ(RD,950) ID,NAM,NOTZRO,NDOWN                                  SINT0601
      INDEX=INNDEX(ID)                                                  SINT0602
      IF(NAM.NE.NAME(INDEX).AND.NAM.NE.$BLANK) CALL ERRMSG('STRATG',7)  SINT0603
      IF(NOTZRO(1).LE.0) NOTZRO(1)=1                                    SINT0604
      IF(NOTZRO(2).LE.0.OR.NOTZRO(2).GT.NPERS) NOTZRO(2)=NPERS          SINT0605
      WRITE(WT,960) INDEX,IDNO(INDEX),NAME(INDEX),NOTZRO,NDOWN          SINT0606
      CALL ERASE(M,MAXPER/2)                                            SINT0607
      NOT1=NOTZRO(1)                                                    SINT0608
      NOT2=NOTZRO(2)                                                    SINT0609
      DO 10 L=NOT1,NOT2                                                 SINT0610
   10 M(L)=2                                                            SINT0611
      DO 20 L=1,20                                                      SINT0612
```

```
      IF(NDOWN(L).LT.NOT1.OR.NDOWN(L).GT.NOT2) GO TO 30                 SINT0613
   20 M(NDOWN(L))=1                                                     SINT0614
   30 DO 40 N=1,NPERS,5                                                 SINT0615
   40 MAINT(INDEX,(N+4)/5)=                                             SINT0616
     $M(N+4)+10*(M(N+3)+10*(M(N+2)+10*(M(N+1)+10*M(N))))                SINT0617
   50 CONTINUE                                                          SINT0618
      WRITE(WT,970)(I,I=1,9)                                            SINT0619
      DO 60 I=1,NOSTNS                                                  SINT0620
   60 WRITE(WT,971) I,IDNO(I),(MAINT(I,J),J=1,LMAX)                     SINT0621
      RETURN                                                            SINT0622
  910 FORMAT(L3,I1,I6,10A7)                                             SINT0623
  920 FORMAT('1 STRATG NOW PROCESSING STRATEGY DATA FOR IDSTRG =',I10/  SINT0624
     $'0 STRATEGY TITLE :"',10A7,'"'//)                                 SINT0625
  925 FORMAT('0*****',A6,'A NUCLEAR POWER MANAGEMENT STRATEGY *****',   SINT0626
     $' NAME=',L1,I7,' FOR PUNCH OPTION *****'//)                       SINT0627
  930 FORMAT(3A4)                                                       SINT0628
  940 FORMAT(' KEY1'/' ',3A4//                                          SINT0629
     $' INDEX  IDNO  NAME STARTUP RETIRE        DOWN FOR REFUELING &/OR' SINT0630
     $,' MAINTENANCE'/T29,'AFTER')                                      SINT0631
  950 FORMAT(I4,A4,2X,2I5,20I3)                                         SINT0632
  960 FORMAT(I4,I8,A6,I5,I8,6X,20I4)                                    SINT0633
  970 FORMAT(//,T20,' MAINTENANCE STRATEGY BY PERIOD AND INDEX',        SINT0634
     $'  (0=NON-EXISTENT;1=DOWN;2=ON-LINE)'//T115,'1',T62,'PERIOD'/     SINT0635
     $15X,9I10,9X,'0'/' INDEX IDNO',4X,10('1234567890')/)               SINT0636
  971 FORMAT(I4,I7,4X,20I5)                                             SINT0637
      END                                                               SINT0638
      SUBROUTINE PRESIM                                                 SINT0639
C     SYSINT VERSION  1-01-73                                           SINT0640
C     PERFORMS PRE-SIMULATION DATA MANIPULATION FOR EACH PERIOD         SINT0641
C     **************************************************************    SINT0642
      IMPLICIT REAL*8 (A-H,O-$)                                         SINT0643
C     COMMON VARIABLES                                                  SINT0644
C     VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS SINT0645
      COMMON/PLTDAT/IDNO(100),NAME(100),TYPE(100),SUSDHT(100),PNOM(100), SINT0646
     $NPTS(100),MWPT(5,100),HTRAT(5,100)                                SINT0647
      COMMON/PERDAT/AVLBTY(100),CSTBTU(100),STATUS(100),EXPHRS(100),    SINT0648
```

602

```
     $EXPBTU(100),EXPGWH(100),NORDER(500),COST(100),ENERGY(100),
     $SUPCST(100),MRGCST(5,100)
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
      COMMON/FLOAT/EPS,TRACE,PKMW,SPNRES,CSTEMR
      COMMON/TITLE/SGTITL(10),PDTITL(10)
      COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1
     $,IDSTRG,PCHMIN,PCHMAX,MBRNUM
      COMMON/LDGNFO/LDTYPE,LDTYPS,LOAD(50,25),NORDOP,NOENTY,NOBASE,
     $NOPEAK,NNORD
      COMMON/MAXMUM/IDIMEN,MAXPLT,MAXPER,MAXNPT
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI
      COMMON/LOGICL/MINI,MIDI,MAXI,NPM,PCHING
      COMMON/MAINT/MAINT(100,20)
C     MAINT IS DIMENSIONED (MAXPLT,MAXPER/5)  THE 5 IS 5I1/INTEGER*2
      REAL*4 SUSDHT,PNOM,HTRAT
      REAL*4 SUPCST,MRGCST
      REAL*8 MILLI
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX
      INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD
      LOGICAL*1 MINI,MIDI,MAXI,NPM,PCHING
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES
      LOGICAL*1 PRINT
      EQUIVALENCE (PRINT,MIDI)
      REAL*4 TEMP4
      FIND(9'NPER1)
C     TRANSLATE MAINTENANCE CODE INTO STATUS
      J=(NPER+4)/5
      I=NPER+5-J*5
      IDUM=10**(5-I)
      DO 110 K=1,NOSTNS
  110 STATUS(K)=MOD(MAINT(K,J)/IDUM,10)
C     RETRIEVE PERIOD INFO FROM DIRECT ACCESS DEVICE
      READ (9'NPER1)PDTITL,NPER,LDTYPE,PKMW,SPNRES,DM,DT,CSTEMR,NORDOP,
     $NOENTY,NOBASE,NOPEAK,CSTBTU,AVLBTY,NORDER,ENERGY
C     RESCALE LOAD-DURATION CURVE & CONVERT FROM DL SPACING (2% PKMW)
```

```
C     TO DESIRED DM
      CALL ERASE(PROB,2*IDIMEN)
      TEMP=1.D-4
      IEMIN=0
      DO 10 J=1,50
      PROB(J)=LOAD(J,LDTYPE)*TEMP
      IF(PROB(J).GT.ONE-TRACE) IEMIN=J
      IF(PROB(J).LE.ZERO) GO TO 20
   10 CONTINUE
      J=50
   20 IEMAX=J
      DL=PKMW*ONE/IEMAX
      PEMAX=IEMAX*DL+EPS
      PEMIN=IEMIN*DL
      GWHPER=DL*DT*MILLI
      DAYS=DT/24.D0
      DMTEMP=DM
      DM=DL
      IF(.NOT.PRINT) GO TO 30
      WRITE(WT,930) IDSTRG,SGTITL
      WRITE(WT,940) NPER,PDTITL
      WRITE(WT,920) DM,IEMAX,PEMAX,(PROB(K),K=1,IEMAX)
      TEMP=GWHNRG(ZERO,PEMAX)
      WRITE(WT,901) TEMP
   30 CALL NUSCAL(DL,DMTEMP)
C     ADJUST FINAL POINT SO LATER LINEAR INTERPOLATION GIVES PROPER
C     AREA UNDER THE CURVE (I.E., EXPECTED VALUE)
      PROB(IEMAX)=PROB(IEMAX)*HALF*(ONE+PEMAX/DM-IEMAX)
      PROB(IEMAX+1)=ZERO
      IEMAX=IEMAX+1
      PEMAX=IEMAX*DM+EPS
      IF(.NOT.PRINT) GO TO 40
      WRITE(WT,920) DM,IEMAX,PEMAX,(PROB(K),K=1,IEMAX)
      TEMP=GWHNRG(ZERO,PEMAX)
      WRITE(WT,902) TEMP
   40 DO 50 I=1,NCSTNS
```

```
      TEMP4=CSTBTU(I)*CENTI
      SUPCST(I)=SUSDHT(I)*TEMP4
      DO 50 J=1,MAXNPT
   50 MRGCST(J,I)=HTRAT(J,I)*.001*TEMP4
C     WRITE FINAL PERIOD CONFIGURATION
      WRITE(WT,930) IDSTRG,SGTITL
      WRITE(WT,940) NPER,PDTITL
      IF(NORDOP.EQ.1) SPNRES=-2.D9
      WRITE(WT,950) PKMW,SPNRES,DT,LDTYPE
      WRITE(WT,920) DM,IEMAX,PEMAX,(PROB(K),K=1,IEMAX)
      WRITE(WT,960) CSTEMR
      WRITE(WT,970) (I,IDNO(I),NAME(I),MWPT(NPTS(I),I),TYPE(I),STATUS(I)
     $,AVLBTY(I),CSTBTU(I),SUPCST(I),ENERGY(I),NPTS(I),(MWPT(J,I),
     $MRGCST(J,I),J=1,MAXNPT),I,I=1,NOSTNS)
      IF(MIDI) CALL CMPTIM('PRESIM','LDGORD')
      CALL LDGORD
      IF(MIDI) CALL CMPTIM('LDGORD','PRESIM')
      IF(PCHING) CALL PUNCHR(2)
      CALL CMPTIM('PRESIM','SYSGEN')
      CALL SYSGEN
      CALL CMPTIM('SYSGEN','PRESIM')
      RETURN
  901 FORMAT (//10X,'GWHNRG(0,PEMAX) AT POINT 1=',F15.8)
  902 FORMAT (//10X,'GWHNRG(0,PEMAX) AT POINT 2=',F15.8)
  920 FORMAT('0' ,10X,'DM = ',F10.4,10X,'IEMAX = ',I5,10X,'PEMAX = ',
     $F12.4,//,10X,'PROB(K),K=1,IEMAX ',/,(1X,10F13.9))
  930 FORMAT('1'/'OSTRATEGY ID = ',I10,10X,'TITLE :"',10A7,'"')
  940 FORMAT('OPERIOD NUMBER =',I9,10X,'TITLE :"',10A8,'"')
  950 FORMAT('0',T10,'PKMW',T22,'SPNRES(MW)',T39,'DT(HRS)',T54,'LDTYPE'/
     $F15.2,7X,F7.2,F15.2,I13)
  960 FORMAT('OCOST OF EMERGENCY POWER =',F8.4,' $/MWH')
  970 FORMAT(/'OINDEX IDNO NAME MAXMW TYPE STAT.AVLBTY CSTBTU SUPCST',
     $' ENERGY NPTS',T68,'(MWPT,MRGCST) IN UNITS OF (MW,$/MWH)',
     $T129,' INDEX'//(I4,I7,A5,I5,3X,A1,I6,F8.3,F7.3,F6.0,F8.1,I3,I5,F7.3
     $,4(2X,I4,F7.3),I4))
      END
```

```
      SUBROUTINE NUSCAL(DMOLD,DMNEW)                                    SINT0757
C     SYSINT VERSION 10-15-71                                           SINT0758
C     CHANGES SPACING OF PROB FROM DMOLD TO DMNEW                       SINT0759
C     ************************************************************      SINT0760
      IMPLICIT REAL*8 (A-H,O-$)                                         SINT0761
C     COMMON VARIABLES                                                  SINT0762
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES                     SINT0763
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)   SINT0764
      COMMON/FLOAT/EPS,TRACE,PKMW,SPNRES,CSTEMR                         SINT0765
      COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1SINT0766
     $,IDSTRG,PCHMIN,PCHMAX,MBRNUM                                      SINT0767
      COMMON/MAXMUM/IDIMEN,MAXPLT,MAXPER,MAXNPT                         SINT0768
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLISINT0769
      COMMON/LOGICL/MINI,MIDI,MAXI,NPM,PCHING                           SINT0770
      REAL*8 MILLI                                                      SINT0771
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX                SINT0772
      LOGICAL*1 MINI,MIDI,MAXI,NPM,PCHING                               SINT0773
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES                   SINT0774
      IF(DMOLD.EQ.DMNEW) RETURN                                         SINT0775
      PDUM=PEMAX                                                        SINT0776
      IDUM=IEMAX+1                                                      SINT0777
      GOAL=GWHNRG(ZERO,PEMAX)                                           SINT0778
C     GOAL =  EXPECTED DEMAND UNDER PROB VS DMOLD                       SINT0779
      IF(.NOT.MIDI) GO TO 5                                             SINT0780
      WRITE(WT,1)                                                       SINT0781
      WRITE(WT,910) GOAL                                                SINT0782
      WRITE(WT,920) DM,IEMAX,PEMAX,(PROB(I),I=1,IEMAX)                  SINT0783
    5 DM=DMNEW                                                          SINT0784
      GWHPER=DM*DT*MILLI                                                SINT0785
      IEMAX=PEMAX/DM                                                    SINT0786
      IF(IEMAX+IDUM.GT.IDIMEN) CALL ERRMSG('NUSCAL',1)                  SINT0787
      TEMP=IEMAX*DM+EPS                                                 SINT0788
      IF(TEMP.GT.PEMAX)PEMAX=TEMP                                       SINT0789
      IEMIN=IEMIN*DMOLD/DMNEW+EPS                                       SINT0790
      PEMIN=IEMIN*DM                                                    SINT0791
      DO 10 I=1,IEMIN                                                   SINT0792
```

```
   10 PROB(I+IDUM)=ONE
      JLOW=IEMIN+1
      TEMP=(IDUM-1)*DMOLD
      JHI=TEMP/DM+ONE
      IF(JHI.GT.IEMAX) GO TO 30
      TEMP=PROB(IDUM-1)/(PDUM-TEMP)
      DO 20 I=JHI,IEMAX
   20 PROB(I+IDUM)=TEMP*(PDUM-I*DM)
   30 JHI=JHI-1
C     FIRST APPROX :  PROB(INEW)=LINEAR INTERPOLATION  OF OLD PROB
C     AT INEW*DMNEW
      TEMP=DMNEW/DMOLD
      DO 40 I=JLOW,JHI
      FB=I*TEMP
      ILO=FB
      FB=FB-ILO
      IHI=ILO+1
   40 PROB(I+IDUM)=PROB(ILO)+FB*(PROB(IHI)-PROB(ILO))
      DO 50 I=1,IEMAX
   50 PROB(I)=PROB(I+IDUM)
      I=IEMAX
      IF(PROB(I).GT.ZERO) GO TO 59
      I=I-1
      IF(PROB(I).GT.ZERO) GO TO 58
   51 I=I-1
      IF(PROB(I).LE.ZERO) GO TO 51
      IEMAX=I+1
   58 PEMAX=IEMAX*DM+EPS
   59 TEST=GWHNRG(ZERO,PEMAX)
C     TEST = EXPECTED DEMAND UNDER FIRST APPROX
      TRUERR=GOAL-TEST
      RELERR=DABS(TRUERR)/GOAL
      IF(RELERR.LT.TRACE) GO TO 100
      IF(RELERR.GT.MILLI) CALL ERRMSG('NUSCAL',3)
      IF(.NOT.MAXI) GO TO 60
      WRITE(WT,910) GOAL,TEST,TRUERR,RELERR
```

```
      WRITE(WT,920) DM,IEMAX,PEMAX,(PROB(I),I=1,IEMAX)
C     SECOND APPROX : ADJUST INTERIOR POINTS UP OR DOWN EQUAL AMOUNT DP
   60 ILO=IEMIN+2
      IHI=IEMAX-1
      DP=TRUERR/(GWHPER*(IHI-ILO+1))
      IF(DABS(DP).GT.MILLI) CALL ERRMSG('NUSCAL',3)
      DO 70 I=ILO,IHI
   70 PROB(I)=PROB(I)+DP
      IF(.NOT.MAXI) GO TO 75
      TEST=GWHNRG(ZERO,PEMAX)
      WRITE(WT,910) GOAL,TEST
      WRITE(WT,930) DP
      WRITE(WT,920) DM,IEMAX,PEMAX,(PROB(I),I=1,IEMAX)
C     CHECK TO SEE IF VIOLATE CDF PROPERTIES AT ENDS
C     THIRD APPROX : AVERAGE POINTS IN VIOLATION AND CHECK TO SEE THAT
C     THEY ARE LESS THAN 1 AND GREATER THAN 0
   75 IF(PROB(ILO).LE.PROB(ILO-1)) GO TO 90
      PROB(ILO)=HALF*(PROB(ILO)+PROB(ILO-1))
      PROB(ILO-1)=PROB(ILO)
      IF(PROB(ILO).LT.ONE) GO TO 100
   80 CALL ERRMSG('NUSCAL',4)
      WRITE(WT,920) DM,IEMAX,PEMAX,(PROB(I),I=1,IEMAX)
      RETURN
   90 IF(PROB(IHI).GT.PROB(IEMAX)) GO TO 100
      PROB(IHI)=HALF*(PROB(IHI)+PROB(IEMAX))
      PROB(IEMAX)=PROB(IHI)
      IF(PROB(IHI).LE.ZERO) GO TO 80
C     EXIT IF REASONABLE NEW PROB OBTAINED
  100 IF(.NOT.MIDI) RETURN
      TEST=GWHNRG(ZERO,PEMAX)
      WRITE(WT,910) GOAL,TEST
      WRITE(WT,920) DM,IEMAX,PEMAX,(PROB(I),I=1,IEMAX)
      RETURN
    1 FORMAT('1   NUSCAL ENTERED TO CHANGE SPACING OF PROB')
  910 FORMAT(5(/),T7,'GOAL',T22,'TEST',T37,'TRUERR',T52,'RELERR',/,
     $3F15.6,E15.6)
```

```
  920 FORMAT(/,10X,'DM = ',F10.4,10X,'IEMAX = ',I5,10X,'PEMAX = ',       SINT0865
     $F12.4,//,10X,'PROB(I),I=1,IEMAX ',/,(1X,10F13.9))                  SINT0866
  930 FORMAT(/,T11,'DP = ',F12.10,//)                                     SINT0867
      END                                                                 SINT0868
      SUBROUTINE LDGORD                                                   SINT0869
C     SYSINT VERSION 11-2-71                                              SINT0870
C     SETS UP NORDER FOR THE SPECIFIED OPTION NORDOP                      SINT0871
C     ************************************************************        SINT0872
      IMPLICIT REAL*8 (A-H,O-$)                                           SINT0873
C     COMMON VARIABLES                                                    SINT0874
C     VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS   SINT0875
      COMMON/PLTDAT/IDNO(100),NAME(100),TYPE(100),SUSDHT(100),PNOM(100),  SINT0876
     $NPTS(100),MWPT(5,100),HTRAT(5,100)                                  SINT0877
      COMMON/PERDAT/AVLBTY(100),CSTBTU(100),STATUS(100),EXPHRS(100),      SINT0878
     $EXPBTU(100),EXPGWH(100),NORDER(500),COST(100),ENERGY(100),          SINT0879
     $SUPCST(100),MRGCST(5,100)                                           SINT0880
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES                       SINT0881
      COMMON/FLOAT/EPS,TRACE,PKMW,SPNRES,CSTEMR                           SINT0882
      COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1  SINT0883
     $,IDSTRG,PCHMIN,PCHMAX,MBRNUM                                        SINT0884
      COMMON/LDGNFO/LDTYPE,LDTYPS,LOAD(50,25),NORDOP,NOENTY,NOBASE,       SINT0885
     $NOPEAK,NNORD                                                        SINT0886
      COMMON/MAXMUM/IDIMEN,MAXPLT,MAXPER,MAXNPT                           SINT0887
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI  SINT0888
      COMMON/LOGICL/MINI,MIDI,MAXI,NPM,PCHING                             SINT0889
      COMMON/MURGER/CTEMP(500),NEWCOD(5),NEWCST(5),MPTS,IFRST,ILAST       SINT0890
C     NEWCST & NEWCOD ARE DIMENSIONED MAXNPT;CTEMP (MAXPLT*MAXNPT)        SINT0891
      REAL*4 SUSDHT,PNOM,HTRAT                                            SINT0892
      REAL*4 SUPCST,MRGCST                                                SINT0893
      REAL*4 CTEMP,NEWCST                                                 SINT0894
      REAL*8 MILLI                                                        SINT0895
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX                  SINT0896
      INTEGER*4 NEWCOD                                                    SINT0897
      INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD              SINT0898
      LOGICAL*1 MINI,MIDI,MAXI,NPM,PCHING                                 SINT0899
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES                     SINT0900
```

```
      DATA $ONLY/1.D0/                                                  SINT0901
      INTEGER*2 NTEMP(500),MWSPIN(500)                                  SINT0902
C     NTEMP AND MWSPIN ARE DIMENSIONED MAXPLT*MAXNPT                    SINT0903
      NAMELIST/ERRDAT/NORDOP,NOENTY,NOBASE,NOPEAK,IFRST,ILAST,INDEX,NPT, SINT0904
     $MPTS,ID,I,SPIN,NORDER,NTEMP                                       SINT0905
      CALL COMPRS(NTEMP)                                                SINT0906
      WRITE(WT,940) NORDOP,NOENTY,NOBASE,NOPEAK,(NTEMP(I),I=1,NOENTY)   SINT0907
C     ENCODE THOSE VALVE POINTS IN BASE PORTION                         SINT0908
      ISWTCH=NOENTY-NOPEAK+1                                            SINT0909
      NOBASP=NOBASE+1                                                   SINT0910
      IFRST=NOBASP                                                      SINT0911
      ILAST=NOBASE                                                      SINT0912
      CTEMP(ILAST+1)=1.E50                                              SINT0913
      NORDER(ILAST+1)=1001                                              SINT0914
      SPINXS=-SPNRES                                                    SINT0915
      DO 120 INDEX=1,NOSTNS                                             SINT0916
      NPT=0                                                             SINT0917
      ID=IDNO(INDEX)                                                    SINT0918
      DO 70 NORD=1,NOBASE                                               SINT0919
      IF(NTEMP(NORD).NE.ID) GO TO 70                                    SINT0920
      NPT=NPT+1                                                         SINT0921
      NORDER(NORD)=1000*NPT+INDEX                                       SINT0922
      CTEMP(NORD)=MRGCST(NPT,INDEX)                                     SINT0923
   70 CONTINUE                                                          SINT0924
      IF(NPT.EQ.0) GO TO 120                                            SINT0925
      MPTS=NPTS(INDEX)-NPT                                              SINT0926
      IF(MPTS) 80,105,90                                                SINT0927
C     ANY ONE OF SEVERAL ERRORS                                         SINT0928
   80 WRITE(WT,ERRDAT)                                                  SINT0929
      CALL ERRMSG('LDGORD',9)                                           SINT0930
   90 SPINXS=SPINXS+                                                    SINT0931
     $CENTI*AVLBTY(INDEX)*(MWPT(NPTS(INDEX),INDEX)-MWPT(NPT,INDEX))     SINT0932
      DO 100 I=1,MPTS                                                   SINT0933
      MPT=NPT+I                                                         SINT0934
      NEWCOD(I)=1000*MPT+INDEX                                          SINT0935
  100 NEWCST(I)=MRGCST(MPT,INDEX)                                       SINT0936
```

```
      ILAST=ILAST+MPTS                                                  SINT0937
      CTEMP(ILAST+1)=1.E50                                              SINT0938
      NORDER(ILAST+1)=1001                                              SINT0939
      CALL MERGER                                                       SINT0940
  105 IF(NOBASE.EQ.NOENTY) GO TO 120                                    SINT0941
      DO 110 I=NOBASP,NOENTY                                            SINT0942
      IF(NTEMP(I).EQ.ID) GO TO 80                                       SINT0943
  110 CONTINUE                                                          SINT0944
  120 CONTINUE                                                          SINT0945
      IF(NOBASE.EQ.NOENTY) GO TO 205                                    SINT0946
C     STARTUP INTERMEDIATE PLANTS ACCORDING TO SPINNING RESERVE REQ.    SINT0947
C     OR ECONOMICS                                                      SINT0948
      IPTR=NOBASP                                                       SINT0949
      NEXTID=NTEMP(IPTR)                                                SINT0950
      NXTNDX=INNDEX(NEXTID)                                             SINT0951
      REASON=ZERO                                                       SINT0952
      K=0                                                               SINT0953
  140 IF(ILAST-IFRST+1) 80,141,142                                      SINT0954
  141 K=1                                                               SINT0955
  142 NPT=NORDER(IFRST)/1000                                            SINT0956
      INDEX=NORDER(IFRST)-NPT*1000                                      SINT0957
      DSPIN=CENTI*AVLBTY(INDEX)*(MWPT(NPT,INDEX)-MWPT(NPT-1,INDEX))     SINT0958
      IF(DSPIN.GT.SPINXS+HALF) GO TO 150                                SINT0959
C     SPINNING RESERVE OK WITH PLANTS ALREADY STARTED                   SINT0960
      IF(MRGCST(1,NXTNDX).LT.CTEMP(IFRST)) GO TO 150                    SINT0961
C     NEXT VALVE POINT LESS EXPENSIVE THAN NEXT PLANT                   SINT0962
      SPINXS=SPINXS-DSPIN                                               SINT0963
      IFRST=IFRST+1                                                     SINT0964
      GO TO 140                                                         SINT0965
C     START UP NEXT PLANT                                               SINT0966
  150 IF(IPTR.NE.ISWTCH) GO TO 170                                      SINT0967
C     FIRST PEAKING PLANT ABOUT TO BE STARTED                           SINT0968
      IF(NOPEAK.EQ.0) GO TO 205                                         SINT0969
      IF(REASON.EQ.$ONLY) GO TO 170                                     SINT0970
      SPINXS=10.D6                                                      SINT0971
      REASON=$ONLY                                                      SINT0972
```

```
      IF(NORDOP.EQ.4) GO TO 140
C     NORDOP=4  PEAKERS COMMITTED ECONOMICALLY AFTER LAST INTERMEDIATE
C     PLANT STARTED
C     NORDOP<4  PEAKERS COMMITTED ECONOMICALLY AFTER ALL  INTERMEDIATE
C     EQUIPMENT
      IFRST=ILAST+1
      K=1
  170 IF(IPTR.GT.NOENTY) GO TO 205
      CTEMP(IFRST)=CTEMP(IFRST)-2.E-5
      ID=NTEMP(IPTR)
      NEXTID=NTEMP(IPTR+1)
      NXTNDX=INNDEX(NEXTID)
      IPTR=IPTR+1
      INDEX=INNDEX(ID)
      MPTS=NPTS(INDEX)
      SPINXS=SPINXS+
     $CENTI*AVLBTY(INDEX)*(MWPT(MPTS,INDEX)-MWPT(1,INDEX))
      I=2
      IF(MPTS.EQ.1) GO TO 200
      DO 180 I=2,MPTS
      NEWCOD(I-K)=1000*I+INDEX
  180 NEWCST(I-K)=MRGCST(I,INDEX)
      IF(K.EQ.1) GO TO 202
      DO 190 I=2,MPTS
      NEWCOD(I-1)=NEWCOD(I)
      NEWCST(I-1)=NEWCST(I)
      IF(NEWCST(I).GE.CTEMP(IFRST)) GO TO 200
  190 CONTINUE
      I=MPTS+1
  200 NEWCST(I-1)=CTEMP(IFRST)
      NEWCOD(I-1)=NORDER(IFRST)
  202 NORDER(IFRST)=1000+INDEX
      CTEMP(IFRST)=MRGCST(1,INDEX)
      IFRST=IFRST+1
      ILAST=ILAST+MPTS
      CTEMP(ILAST+1)=1.E50
```

```
      NORDER(ILAST+1)=1001                                              SINT1009
      MPTS=MPTS-K                                                       SINT1010
      CALL MERGER                                                       SINT1011
      IF(MPTS.GT.0) K=0                                                 SINT1012
      GO TO 140                                                         SINT1013
  205 NNORD=ILAST                                                       SINT1014
      CALL RETMRG                                                       SINT1015
      SPIN=ZERO                                                         SINT1016
      CALL ERASE(NTEMP,MAXPLT*MAXNPT/2)                                 SINT1017
      DO 230 I=1,NNORD                                                  SINT1018
      IF(NORDER(I).LE.1000) GO TO 80                                    SINT1019
      NPT=NORDER(I)/1000                                                SINT1020
      INDEX=NORDER(I)-NPT*1000                                          SINT1021
      IF(NPT-1.NE.NTEMP(INDEX)) GO TO 80                                SINT1022
      NTEMP(INDEX)=NPT                                                  SINT1023
      IF(NPT.EQ.1) GO TO 220                                            SINT1024
      SPIN=SPIN-                                                        SINT1025
     $CENTI*AVLBTY(INDEX)*(MWPT(NPT,INDEX)-MWPT(NPT-1,INDEX))           SINT1026
      GO TO 230                                                         SINT1027
  220 SPIN=SPIN+                                                        SINT1028
     $CENTI*AVLBTY(INDEX)*(MWPT(NPTS(INDEX),INDEX)-MWPT(1,INDEX))       SINT1029
  230 MWSPIN(I)=SPIN                                                    SINT1030
      JJ=(NNORD+4)/5                                                    SINT1031
      I=JJ*5-NNORD                                                      SINT1032
      IF(I.EQ.0) GO TO 250                                              SINT1033
      DO 240 J=1,I                                                      SINT1034
      NORDJ=NNORD+J                                                     SINT1035
      NORDER(NORDJ)=0                                                   SINT1036
      MWSPIN(NORDJ)=-10000                                              SINT1037
  240 CTEMP(NORDJ)=-1.E30                                               SINT1038
  250 JJ5=JJ*5                                                          SINT1039
      DO 255 J=1,JJ5                                                    SINT1040
  255 NTEMP(J)=J                                                        SINT1041
      WRITE(WT,920) NNORD                                               SINT1042
      WRITE(WT,930)(J,NORDER(J),CTEMP(J),MWSPIN(J),(NTEMP(J+I*JJ),      SINT1043
     $NORDER(J+I*JJ),CTEMP(J+I*JJ),MWSPIN(J+I*JJ),I=1,4),J=1,JJ)        SINT1044
```

```
      IF(DABS(SPIN).GT.HALF) GO TO 80
      RETURN
  920 FORMAT(/'OLOADING ORDER (NORDER) AS (1000*NPT + INDEX) :',10X,I5
     $,' VALID ENTRIES'//1X,5('|    J  NORDER MRGCST MWSPN'),'|')
  930 FORMAT((' ',5('|',I5,I6,F9.4,I5),'|'))
  940 FORMAT(//'OSTARTUP ORDER :',10X,'WITH NORDOP=',I2,6X,'NOENTY=',I4,
     $6X,'NOBASE=',I3,6X,'NOPEAK=',I3/(20I5))
      END
      SUBROUTINE COMPRS(NTEMP)
C     SYSINT VERSION 10-31-71
C     PERFORM STATUS:IDNO CHECK AND THEN COMPRESS AND TRANSFER NORDER
C     INTO NTEMP;  ALTER MARGINAL COST CURVES AND OPTIMIZE STARTUP ORDER
C     ************************************************************
      IMPLICIT REAL*8 (A-H,O-$)
C     COMMON VARIABLES
C     VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS
      COMMON/PLTDAT/IDNO(100),NAME(100),TYPE(100),SUSDHT(100),PNOM(100),
     $NPTS(100),MWPT(5,100),HTRAT(5,100)
      COMMON/PERDAT/AVLBTY(100),CSTBTU(100),STATUS(100),EXPHRS(100),
     $EXPBTU(100),EXPGWH(100),NORDER(500),COST(100),ENERGY(100),
     $SUPCST(100),MRGCST(5,100)
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
      COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1
     $,IDSTRG,PCHMIN,PCHMAX,MBRNUM
      COMMON/LDGNFO/LDTYPE,LDTYPS,LOAD(50,25),NORDOP,NOENTY,NOBASE,
     $NOPEAK,NNORD
      COMMON/MAXMUM/IDIMEN,MAXPLT,MAXPER,MAXNPT
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI
      COMMON/LOGICL/MINI,MIDI,MAXI,NPM,PCHING
      REAL*4 SUSDHT,PNOM,HTRAT
      REAL*4 SUPCST,MRGCST
      REAL*8 MILLI
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX
      INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD
      LOGICAL*1 MINI,MIDI,MAXI,NPM,PCHING
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES
```

```
      REAL*4 TEMP4
      INTEGER*2 NTEMP(1),IPJ(3)
C     CHECK CONSISTENCY OF NOBASE,NOPEAK & NOENTY
    5 IF(NORDOP.EQ.1.OR.NOBASE.GT.NOENTY) NOBASE=NOENTY
      IF(NOBASE.LE.0) NOBASE=1
      NOPEAK=MIN0(NOPEAK,NOENTY-NOBASE)
      NTEMP(NOENTY+1)=IDNO(1)
      IF(NORDER(1).EQ.0) GO TO 80
      CALL ERASE(NTEMP,MAXPLT*MAXNPT/2)
C     FLAG OFF-LINE PLANTS & CHECK THAT EACH ON-LINE PLANT MENTIONED
      DO 8 I=1,NOSTNS
      ID=IDNO(I)
      IS=STATUS(I)
      IF(IS.NE.2) IS=1
      DO 7 J=1,NOENTY
      IF(ID.EQ.NORDER(J)) GO TO (6,8),IS
      GO TO 7
    6 NORDER(J)=0
    7 CONTINUE
      IF(IS.EQ.2) WRITE(WT,911)ID
      IF(IS.EQ.2) CALL ERRMSG('COMPRS',9)
    8 CONTINUE
C     CONTROL SEGMENT OF NORDER COMPRESSED INTO NTEMP
      IP=0
      J=1
   10 GO TO (20,30,40,45),J
   20 ILO=1
      IHI=NOBASE
      GO TO 50
   30 ILO=IHI+1
      IHI=NOENTY-NOPEAK
      IF(IHI.LT.ILO) GO TO 70
      GO TO 50
   40 ILO=IHI+1
      IHI=NOENTY
      GO TO 50
```

```
   45 NOBASE=IPJ(1)                                                    SINT1117
      NOPEAK=IPJ(3)-IPJ(2)                                             SINT1118
      NOENTY=IPJ(3)                                                    SINT1119
      NORDER(1)=0                                                      SINT1120
      GO TO 5                                                          SINT1121
C     PERFORM COMPRESSION AND TRANSFER OF A SEGMENT                    SINT1122
   50 IF(ILO.GT.NOENTY) GO TO 70                                       SINT1123
      DO 60 I=ILO,IHI                                                  SINT1124
      IF(NORDER(I).EQ.0) GO TO 60                                      SINT1125
      IP=IP+1                                                          SINT1126
      NTEMP(IP)=NORDER(I)                                              SINT1127
   60 CONTINUE                                                         SINT1128
   70 IPJ(J)=IP                                                        SINT1129
      J=J+1                                                            SINT1130
      GO TO 10                                                         SINT1131
C     ALTER MARGINAL COST CURVES                                       SINT1132
   80 IF(MIDI) WRITE(WT,901)                                           SINT1133
      DO 61 I=1,NCSTNS                                                 SINT1134
      JJ=NPTS(I)                                                       SINT1135
C     PUT MINIMUM AVERAGE COST IN MRGCST(1,I)                          SINT1136
      TEMP4=MRGCST(1,I)*MWPT(1,I)                                      SINT1137
      IF (JJ.EQ.1) GO TO 61                                            SINT1138
      DO 1 J=2,JJ                                                      SINT1139
      TEMP4=TEMP4+MRGCST(J,I)*(MWPT(J,I)-MWPT(J-1,I))                  SINT1140
    1 MRGCST(1,I)=AMIN1(MRGCST(1,I),TEMP4/MWPT(J,I))                   SINT1141
      SUM=ZERO                                                         SINT1142
C     LEVELIZE DECREASING MARGINAL COST CURVES                         SINT1143
   11 IF(JJ.LT.3) GO TO 55                                             SINT1144
      DO 51 J=3,JJ                                                     SINT1145
      IF(MRGCST(J,I).GE.MRGCST(J-1,I)) GO TO 51                        SINT1146
      SUM=ZERO                                                         SINT1147
      DO 31 K=2,J                                                      SINT1148
   31 SUM=SUM+MRGCST(K,I)*(MWPT(K,I)-MWPT(K-1,I))/(MWPT(J,I)-MWPT(1,I))SINT1149
      DO 41 K=2,J                                                      SINT1150
   41 MRGCST(K,I)=SUM                                                  SINT1151
      GO TO 11                                                         SINT1152
```

```
   51 CONTINUE                                                          SINT1153
   55 IF(MINI) GO TO 61                                                 SINT1154
      WRITE(WT,910) I,IDNO(I),NAME(I),(MWPT(K,I),MRGCST(K,I),K=1,JJ)    SINT1155
      IF(SUM.NE.ZERO) WRITE(WT,920)                                     SINT1156
   61 CONTINUE                                                          SINT1157
      IF(NORDOP.LT.3) GO TO 170                                         SINT1158
C     OPTIMIZE STARTUP ORDER                                            SINT1159
      NO=NOENTY-(NOBASE+NOPEAK)                                         SINT1160
      IDUM=NOBASE                                                       SINT1161
      IPJ(3)=2                                                          SINT1162
      IF(NO.NE.0) GO TO 100                                             SINT1163
   90 NO=NOPEAK                                                         SINT1164
      IDUM=NOENTY-NOPEAK                                                SINT1165
      IPJ(3)=3                                                          SINT1166
      IF(NO.EQ.0) GO TO 150                                             SINT1167
  100 DO 110 J=1,NO                                                     SINT1168
      ID=NTEMP(IDUM+J)                                                  SINT1169
      NORDER(J)=INNDEX(ID)                                              SINT1170
  110 NORDER(NO+J)=ID                                                   SINT1171
C     START UP UNITS IN ORDER OF INCREASING MINIMUM AVERAGE COST        SINT1172
      IF(NO.EQ.1) GO TO 150                                             SINT1173
      DO 140 J=2,NO                                                     SINT1174
      IPJ(1)=NORDER(J)                                                  SINT1175
      IPJ(2)=NORDER(NO+J)                                               SINT1176
      IP=J                                                              SINT1177
  120 IP=IP-1                                                           SINT1178
      IF(IP.EQ.0) GO TO 130                                             SINT1179
      IF(MRGCST(1,IPJ(1)).GE.MRGCST(1,NORDER(IP))) GO TO 130            SINT1180
      NORDER(IP+1)=NORDER(IP)                                           SINT1181
      NORDER(NO+IP+1)=NORDER(NO+IP)                                     SINT1182
      GO TO 120                                                         SINT1183
  130 NORDER(IP+1)=IPJ(1)                                               SINT1184
      NORDER(NO+IP+1)=IPJ(2)                                            SINT1185
  140 CONTINUE                                                          SINT1186
  150 DO 160 J=1,NO                                                     SINT1187
  160 NTEMP(IDUM+J)=NORDER(NO+J)                                        SINT1188
```

```
      IF(IPJ(3).NE.3) GO TO 90
  170 CALL ERASE(NORDER,MAXPLT*MAXNPT/2)
      RETURN
C     RETURN MARGINAL COST CURVES TO ORIGINAL VALUES
      ENTRY RETMRG
      DO 210 I=1,NOSTNS
      TEMP4=CSTBTU(I)*1.E-5
      DO 210 J=1,MAXNPT
  210 MRGCST(J,I)=HTRAT(J,I)*TEMP4
      RETURN
  901 FORMAT('1 COMPRS WILL TEMPORARILY LEVELIZE DECREASING MARGINAL',
     $' COST CURVES TO ALLOW PROPER INCREMENTAL LOADING.'/'  IN ADDITION
     $, MINIMUM AVERAGE COST WILL BE PLACED IN MRGCST(1,I).   THUS,'//
     $T5,'I',T8,'IDNO',T14,'NAME',T21,'(MWPT,MINAVGCST)',T50,'INCREASING
     $ MARGINAL COST CURVE')
  910 FORMAT (I5,I6,A6,5('    (',I4,',',F9.5,')'))
  911 FORMAT(///,' UNLISTED IDNO OF ON-LINE PLANT=',I5)
  920 FORMAT('+',T122,'LEVELIZED')
      END
      SUBROUTINE MERGER
C     SYSINT VERSION 10-31-71
C     MERGES NEWLY STARTED PLANT WITH PREVIOUSLY STARTED ONES
C     ***********************************************************
      IMPLICIT REAL*8 (A-H,O-$)
C     COMMON VARIABLES
C     VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS
      COMMON/PERDAT/AVLBTY(100),CSTBTU(100),STATUS(100),EXPHRS(100),
     $EXPBTU(100),EXPGWH(100),NORDER(500),COST(100),ENERGY(100),
     $SUPCST(100),MRGCST(5,100)
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
      COMMON/MAXMUM/IDIMEN,MAXPLT,MAXPER,MAXNPT
      COMMON/MURGER/CTEMP(500),NEWCOD(5),NEWCST(5),MPTS,IFRST,ILAST
C     NEWCST & NEWCOD ARE DIMENSIONED MAXNPT:CTEMP (MAXPLT*MAXNPT)
      REAL*4 SUPCST,MRGCST
      REAL*4 CTEMP,NEWCST
      INTEGER*4 NEWCOD
```

```
      INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD             SINT1225
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES                    SINT1226
      IF(MPTS.EQ.0) RETURN                                               SINT1227
      I=ILAST+1-MPTS                                                     SINT1228
   10 I=I-1                                                              SINT1229
      IP=I+MPTS                                                          SINT1230
      NORDER(IP)=NORDER(I)                                               SINT1231
      CTEMP(IP)=CTEMP(I)                                                 SINT1232
      IF(I.GT.IFRST) GO TO 10                                            SINT1233
      CTEMP(ILAST+1)=1.E50                                               SINT1234
      NORDER(ILAST+1)=1001                                               SINT1235
      IF(ILAST.GE.MAXPLT*MAXNPT) CALL ERRMSG('MERGER',9)                 SINT1236
      IP=IFRST                                                           SINT1237
      I=IFRST+MPTS                                                       SINT1238
      DO 40 M=1,MPTS                                                     SINT1239
   20 IF(NEWCST(M).LT.CTEMP(I)) GO TO 30                                 SINT1240
      CTEMP(IP)=CTEMP(I)                                                 SINT1241
      NORDER(IP)=NORDER(I)                                               SINT1242
      I=I+1                                                              SINT1243
      IP=IP+1                                                            SINT1244
      GO TO 20                                                           SINT1245
   30 CTEMP(IP)=NEWCST(M)                                                SINT1246
      NORDER(IP)=NEWCOD(M)                                               SINT1247
   40 IP=IP+1                                                            SINT1248
      RETURN                                                             SINT1249
      END                                                                SINT1250
      SUBROUTINE SYSGEN                                                  SINT1251
C     SYSINT VERSION  1-01-73                                            SINT1252
C     SIMULATES SYSTEM GENERATION FOR ONE TIME PERIOD                    SINT1253
C     ***********************************************************        SINT1254
      IMPLICIT REAL*8 (A-H,O-$)                                          SINT1255
C     COMMON VARIABLES                                                   SINT1256
C     VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS  SINT1257
      COMMON/PLTDAT/IDNO(100),NAME(100),TYPE(100),SUSDHT(100),FNCM(100), SINT1258
     $NPTS(100),MWPT(5,100),HTRAT(5,100)                                 SINT1259
      COMMON/PERDAT/AVLBTY(100),CSTBTU(100),STATUS(100),EXPHRS(100),     SINT1260
```

```
     $EXPBTU(100),EXPGWH(100),NORDER(500),COST(100),ENERGY(100),
     $SUPCST(100),MRGCST(5,100)
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
      COMMON/FLOAT/EPS,TRACE,PKMW,SPNRES,CSTEMR
      COMMON/TITLE/SGTITL(10),PDTITL(10)
      COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1
     $,IDSTRG,PCHMIN,PCHMAX,MBRNUM
      COMMON/LDGNFO/LDTYPE,LDTYPS,LOAD(50,25),NORDOP,NOENTY,NOBASE,
     $NOPEAK,NNORD
      COMMON/MAXMUM/IDIMEN,MAXPLT,MAXPER,MAXNPT
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI
      COMMON/LOGICL/MINI,MIDI,MAXI,NPM,PCHING
      REAL*4 SUSDHT,PNOM,HTRAT
      REAL*4 SUPCST,MRGCST
      REAL*8 MILLI
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX
      INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD
      LOGICAL*1 MINI,MIDI,MAXI,NPM,PCHING
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES
C     IDUM'S USED TO MAKE NAMELIST OUTPUT MORE READABLE
      NAMELIST /FNLTOT/MWINST,MWONLN,MWPEAK,MWMRGN,MWSPIN,PLOFL,
     $EXPDEM,EXPGEN,XNKGEN,IDUM1   ,XNNGEN,EXPEMR,IDUM2   ,UNSRVD,PROD$,
     $IDUM3   ,$NKPRD,$NNPRD,IDUM4   ,SUSD$,$NKSUS,IDUM5   ,$NNSUS,$SBTOT,
     $IDUM6   ,$NKTOT,$NNTOT,IDUM7   ,EMRP$,TOTAL$
      INTEGER*2 IDUM1,IDUM2,IDUM3,IDUM4,IDUM5,IDUM6,IDUM7
      DATA IDUM1,IDUM2,IDUM3,IDUM4,IDUM5,IDUM6,IDUM7/7*0/
      INTEGER*2 NUCL/'N'/
      REAL*4 PLOFL
C     IDSTRG.LT.0 IS OPTIONAL RETURN TO CHECK INPUT
      IF(IDSTRG.LT.0) RETURN
      IF(MIDI)  WRITE(WT,930)
      CALL ERASE(EXPBTU,2*MAXPLT,EXPGWH,2*MAXPLT,EXPHRS,2*MAXPLT)
      EXPDEM=GWHNRG(ZERO,PEMAX)
      PE=ZERO
      EXPOUT=ZERO
```

```
C       DOUBLE CHECK TO AVOID INADVERTENT PUNCHING
        IF(NPM.AND.PCHING) GO TO 40
        PCHMIN=-1
        PCHMAX=-1
C       DO LOOP TO BUILD UP EQUIVALENT LOAD CDF
     40 DO 50 J=1,NNORD
        L1=NORDER(J)
        NPT=L1/1000
        L=L1-NPT*1000
        IF(STATUS(L).LE.1) GO TO 50
        P=AVLBTY(L)*1.D-2
        MWIN=0
        IF(NPT.GT.1)MWIN=MWPT(NPT-1,L)
        MWTOT=MWPT(NPT,L)
        HTRATE=HTRAT(NPT,L)
        MWADD=MWTOT-MWIN
        EXPOUT=EXPOUT+(ONE-P)*MWADD
C       SUBTRACT PLANT OF INTEREST
        CALL SUBPLT(MWIN,P)
        IF(MAXI) WRITE(WT,921) DM,IEMAX,PEMAX,(PROB(K),K=1,IEMAX)
        TEMP= PE+MWADD
C       EVALUATES INCREMENT OF EXPECTED PRODUCTION
        ENERGE=P*GWHNRG(PE,TEMP)
        PE=TEMP
C       ADD THE PLANT OF INTEREST BACK IN
        CALL ADDPLT(MWTOT,P)
C       EVALUATE & ACCUMULATE IMPORTANT PRODUCTION INFO
        IF(NPT.EQ.1) EXPHRS(L)=ENERGE*THOUS/MWPT(1,L)
        EXPBTU(L)=EXPBTU(L)+ENERGE*HTRATE
        EXPGWH(L)=EXPGWH(L)+ENERGE
        IF(J.EQ.PCHMIN.OR.J.EQ.PCHMAX) CALL PUNCHR(IDINT(PE))
        IF(.NOT.MIDI) GO TO 50
        AVPROB=1.D20
        IF(MWADD.GT.0) AVPROB=ENERGE*THOUS/(P*DT*MWADD)
        IF(MAXI) WRITE(WT,931)
        WRITE(WT,940)L,IDNO(L),PE,MWIN,MWADD,MWTOT,AVPROB,ENERGE,
```

```
     $EXPGWH(L),L
      IF(MAXI) WRITE(WT,922) DM,IEMAX,PEMAX,(PROB(K),K=1,IEMAX)
   50 CONTINUE
      TEMP=GWHNRG(ZERO,PEMAX)
      TEMP=TEMP-EXPDEM
      APXOUT=TEMP*THOUS/DT
      TEMP=(EXPOUT-APXOUT)*HUNDRD/(EXPOUT+1.D-20)
      IF(DABS(TEMP).GT.CENTI) CALL ERRMSG('SYSGEN',5)
      IF(.NOT.MIDI) GO TO 60
      WRITE(WT,910) EXPOUT,APXOUT,TEMP
      WRITE(WT,920) DM,IEMAX,PEMAX,(PROB(K),K=1,IEMAX)
   60 UNSRVD=GWHNRG(PE,PEMAX)
      PLOFL=PROBX(PE)
      MWONLN=PE+EPS
      MWPEAK=PKMW
      MWMRGN=MWONLN-MWPEAK
      MWSPIN=SPNRES
      MWINST=0
      PROD$=ZERO
      EXPGEN=ZERO
      SUSD$=ZERO
      XNNGEN=ZERO
      $NNPRD=ZERO
      $NNSUS=ZERO
      TEMP=HUNDRD/DT
      WRITE (WT,950) IDSTRG,SGTITL,NPER,PDTITL
C     EVALUATE AND PRINT FINAL PER PLANT RESULTS
      DO 70  J=1,NOSTNS
      IF(STATUS(J).GE.1) MWINST=MWINST+MWPT(NPTS(J),J)
      FACT=EXPGWH(J)*THOUS/(MWPT(NPTS(J),J)*DT)
      SUSDS=SUSDNO(EXPHRS(J)*TEMP/AVLBTY(J))
      SUBTU=SUSDS*SUSDHT(J)
      $SUSD=SUBTU*CSTBTU(J)*1.D-2
      SUSD$=SUSD$+$SUSD
      PRDBTU=EXPBTU(J)
      EXPBTU(J)=EXPBTU(J)+SUBTU
```

```
      $PROD=PRDBTU*CSTBTU(J)*1.D-2
      COST(J)=$PROD+$SUSD
      PROD$=PROD$+$PROD
      EXPGEN=EXPGEN+EXPGWH(J)
      IF(TYPE(J).EQ.NUCL) GO TO 65
      XNNGEN=XNNGEN+EXPGWH(J)
      $NNPRD=$NNPRD+$PROD
      $NNSUS=$NNSUS+$SUSD
   65 WRITE (WT,960) J,IDNO(J),NAME(J),FACT,EXPHRS(J),SUSDS,SUBTU,$SUSD,
     $EXPGWH(J),PRDBTU,$PROD,EXPBTU(J),CCST(J),J
   70 CONTINUE
C     EVALUATE AND PRINT FINAL SYSTEM RESULTS
      XNKGEN=EXPGEN-XNNGEN
      $NKPRD=PROD$-$NNPRD
      $NKSUS=SUSD$-$NNSUS
      $NKTOT=$NKPRD+$NKSUS
      $NNTOT=$NNPRD+$NNSUS
      $SBTOT=$NKTOT+$NNTOT
      EXPEMR=EXPDEM-EXPGEN
      EMRP$=EXPEMR*THOUS*CSTEMR
      TOTAL$=PROD$+SUSD$+EMRP$
      WRITE(WT,970) MWINST,MWONLN,MWPEAK,MWMRGN,MWSPIN,PLOFL
      WRITE(WT,980) EXPDEM,EXPGEN,XNKGEN,XNNGEN,EXPEMR,UNSRVD
      WRITE(WT,990) PROD$,$NKPRD,$NNPRD,SUSD$,$NKSUS,$NNSUS,
     $$SBTOT,$NKTOT,$NNTOT,CSTEMR,EMRP$,TOTAL$
      IF(PCHING) WRITE(PUNCH,FNLTOT)
      RETURN
  910 FORMAT(//T10,'TRUE EXP. OUTAGE      =',F8.2,' MW'/
     $T10,'APPROX. EXP. OUTAGE   =',F8.2,' MW'/
     $T10,'ERROR IN APPROX.      =',F9.5,' %'//
     $' FINAL EQUIVALENT LOAD CDF:')
  920 FORMAT(/10X,'DM = ',F10.4,10X,'IEMAX = ',I5,10X,'PEMAX = ',
     $F12.4,//,10X,'PROB(K),K=1,IEMAX ',/,(1X,10F13.9))
  921 FORMAT('0',132('*')/
     $       'OWITHOUT PLANT OF INTEREST        PROB(K),K=1,IEMAX :
     $       DM = ',F8.2,5X,'IEMAX = ',I5,5X,'PEMAX = ',F12.4/(10F13.9))
```

```
922 FORMAT('OWITH    PLANT OF INTEREST        PROB(K),K=1,IEMAX :
   $        DM = ',F8.2,5X,'IEMAX = ',I5,5X,'PEMAX = ',F12.4/(10F13.9))
930 FORMAT('1',T5,'L  IDNO    PE    MWIN MWADD MWTOT   AVPROB',
   $T54,'DELGWH      EXPGWH    L')
931 FORMAT('0',T5,'L  IDNO    PE    MWIN MWADD MWTOT   AVPROB',
   $T54,'DELGWH      EXPGWH    L')
940 FORMAT(I5,I6,F7.0,3I6,F12.8,2F12.6,I5)
950 FORMAT('1'/'OSTRATEGY ID = ',I10,10X,'TITLE :"',10A7,'"'/
   $        'OPERIOD NUMBER =',I9,10X,'TITLE :"',10A8,'"'///
   $T45,'STARTUPS & SHUTDOWNS',T75,'EXPECTED PRODUCTION',T112,'TOTALS'
   $,/,' INDEX  IDNO  NAME  LD FACT   OPER HRS    NUMBER  MEGABTU',
   $T60,'COST($)',T70,' ELECT(GWH)   MEGABTU   COST($)',
   $T108,'MEGABTU   COST($)   INDEX'/)
960 FORMAT(I4,I8,A6,F10.6,2F10.4,F10.0,F8.0,F14.5,2F10.0,4X,2F10.0,I6)
970 FORMAT(////,T22,'P O W E R :',T59,'MEGAWATTS',/
   $T26,'    INSTALLED CAPACITY',T56,I10,/
   $T26,'    ON-LINE   CAPACITY',T56,I10,/
   $T26,'    PEAK LOAD FORECAST',T56,I10,/
   $T26,'    ON-LINE MARGIN @ PEAK',T56,I10,/
   $T26,'    SPINNING RESERVE'  ,T56,I10,/
   $T26,'LOSS-OF-LOAD PROBABILITY',T56,F10.6)
980 FORMAT(//T22,'E N E R G Y :',T60,'GWH',/
   $T30,'EXPECTED DEMAND',     T54,F12.4,/
   $T30,'EXPECTED PRODUCTION', T54,F12.4,/
   $T38,'(  NUCLEAR  ',T54,F12.4,')'/
   $T38,'(NON-NUCLEAR',T54,F12.4,')'/
   $T30,'EXPECTED EMERG PURCH',T54,F12.4,/
   $T30,'(UNSERVED BY DIRECT CALC',T54,F12.4,')')
990 FORMAT(//,T22,'D O L L A R   C O S T :',T59,'SYSTEM',T74,'NUCLEAR'
   $,T86,'NON-NUCLEAR'/
   $T31,'PRODUCTION FUEL',T54,F12.0,2F15.0/
   $T31,'STARTUPS & SHUTDOWNS',T54,F12.0,2F15.0/
   $'+',T56,10('_'),T71,10('_'),T86,10('_')/
   $T31,'      SUB-TOTALS',T54,F12.0,2F15.0/
   $T31,'EMERG.PURCH.@',F6.2, ' $/MWH',T56,F10.0,/'+',
   $T56,'__________',/,T36,'TOTAL',T54,F12.0)
```

```
      END
      SUBROUTINE SUBPLT(MW,P)
C     SYSINT VERSION  1-01-73
C     SUBTRACTS PLANT OF MW MEGAWATTS AND P FRACTIONAL AVAILABILITY
C     FROM PROB, THE EQUIVALENT LOAD CDF
C     NOTE:    MW  MUST BE LESS THAN OR EQUAL TO PEMIN
C     ************************************************************
      IMPLICIT REAL*8 (A-H,O-$)
C     COMMON VARIABLES
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
      COMMON/FLOAT/EPS,TRACE,PKMW,SPNRES,CSTEMR
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI
      REAL*8 MILLI
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES
      IF(MW.LE.0) RETURN
      IF(MW.LE.PEMIN) GO TO 10
      CALL ERRMSG('SUBPLT',2)
      RETURN
   10 ILOW=IEMIN+1
      FB=MW/DM
      INT=FB
      FB=FB-INT
      OVP=ONE/P
      Q=ONE-P
      QFB=Q*FB
      GAMMA=ONE/(ONE-QFB)
      IF(INT.GT.0) GO TO 60
C     LOOP TO UNCONVOLVE PLANT IF MW.LT.DM
      DO 20 J=ILOW,IEMAX
   20 PROB(J)=GAMMA*(PROB(J)-QFB*PROB(J-1))
C     FIND NEW PEMAX AND IEMAX
   30 J=IEMAX
   40 IF(PROB(J).GT.TRACE) GO TO 50
      PROB(J)=ZERO
      J=J-1
```

```
      GO TO 40
   50 IF(IEMAX.EQ.J) RETURN
      IEMAX=J+1
      PEMAX=IEMAX*DM+EPS
      RETURN
C     LOOP TO UNCONVOLVE PLANT IF MW.GE.DM
   60 DO 70 J=ILOW,IEMAX
      JINT=J-INT
   70 PROB(J)=OVP*(PROB(J)-Q*(PROB(JINT)+FB*(PROB(JINT-1)-PROB(JINT))))
      GO TO 30
      END
      FUNCTION GWHNRG(XLOWER,XUPPER)
C     SYSINT VERSION 10-15-71
C     CALCULATES GWH OF ENERGY UNDER PORTION OF PROB, THE CDF OF
C     EQUIVALENT LOAD, BY INTEGRATING FROM XLOWER TO XUPPER ASSUMING
C     LINEAR INTERPOLATION BETWEEN ARRAY POINTS
C     ************************************************************
      IMPLICIT REAL*8 (A-H,O-$)
C     COMMON VARIABLES
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI
      REAL*8 MILLI
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES
      XLO=XLOWER
      XUP=XUPPER
      GWHNRG=ZERO
      SUM=ZERO
      IF(XLO.GE.XUP) RETURN
      IBELO=XLO/DM
      ILAST=XUP/DM
      IF(IBELO.LE.0.OR.ILAST.GE.IEMAX) GO TO 50
C     STANDARD CASE WITH BOTH POINTS WITHIN NON-ZERO ARRAY POINTS
    5 IFRST=IBELO+1
      IABOV=ILAST+1
      IFRSTP=IFRST+1
```

```
      ILASTM=ILAST-1
      ICASE=IABOV-IBELO
      RLO=IFRST-XLO/DM
      RUP=XUP/DM-ILAST
      PLO=PROB(IFRST)+(PROB(IBELO)-PROB(IFRST))*RLO
      PUP=PROB(IABOV)+(PROB(ILAST)-PROB(IABOV))*(ONE-RUP)
      GO TO (10,20,30,40),ICASE
   40 DO 35 I=IFRSTP,ILASTM
   35 SUM=SUM+PROB(I)
   30 SUM=SUM+HALF*(PROB(IFRST)+PROB(ILAST))
   20 SUM=SUM+HALF*(RLO*(PLO+PROB(IFRST))+RUP*(PUP+PROB(ILAST)))
   15 GWHNRG=SUM*GWHPER
      RETURN
   10 SUM=SUM+(XUP-XLO)*(PLO+PUP)*HALF/DM
      GO TO 15
C     SPECIAL CASES INVOLVING ONE OR BOTH END POINTS
   50 IF(XUP.LE.ZERO.OR.XLO.GE.PEMAX) RETURN
      IF(XLO.LT.ZERO) XLO=ZERO
      IF(XUP.GT.PEMAX) XUP=PEMAX
      IBELO=XLO/DM
      ILAST=XUP/DM
      JCASE=1
      IF(ILAST.GT.0)      JCASE=JCASE+1
      IF(ILAST.EQ.IEMAX)  JCASE=JCASE+1
      IF(IBELO.GT.0)      JCASE=JCASE+1
      IF(IBELO.EQ.IEMAX)  JCASE=JCASE+1
      GO TO (101,102,102,104,105),JCASE
  101 GWHNRG=(XUP-XLO)*GWHPER/DM
      RETURN
  102 SUM=ONE-XLO/DM
      XLO=DM
      IBELO=1
      IF(JCASE.EQ.2) GO TO 5
  104 XO=IEMAX*DM
      PUP=PROB(IEMAX)*(ONE-(XUP-XO)/(PEMAX-XO))
      SUM=SUM+(XUP-XO)*HALF*(PUP+PROB(IEMAX))/DM
```

```
      XUP=XO                                                            SINT1549
      ILAST=IEMAX-1                                                     SINT1550
      GO TO 5                                                           SINT1551
  105 XO=IEMAX*DM                                                       SINT1552
      PUP=PROB(IEMAX)*(ONE-(XUP-XO)/(PEMAX-XO))                         SINT1553
      PLO=PROB(IEMAX)*(ONE-(XLO-XO)/(PEMAX-XO))                         SINT1554
      GWHNRG=(XUP-XLO)*(PLO+PUP)*HALF*GWHPER/DM                         SINT1555
      RETURN                                                            SINT1556
      END                                                               SINT1557
      SUBROUTINE ADDPLT(MW,P)                                           SINT1558
C     SYSINT VERSION  1-01-73                                           SINT1559
C     ADDS PLANT OF MW MEGAWATTS AND P FRACTIONAL AVAILABILITY TO PROB, SINT1560
C     THE EQUIVALENT LOAD CDF                                           SINT1561
C     NOTE:    MW  MUST BE LESS THAN OR EQUAL TO PEMIN                  SINT1562
C     **********************************************************        SINT1563
      IMPLICIT REAL*8 (A-H,O-$)                                         SINT1564
C     COMMON VARIABLES                                                  SINT1565
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES                     SINT1566
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)   SINT1567
      COMMON/FLOAT/EPS,TRACE,PKMW,SPNRES,CSTEMR                         SINT1568
      COMMON/MAXMUM/IDIMEN,MAXPLT,MAXPER,MAXNPT                         SINT1569
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI SINT1570
      REAL*8 MILLI                                                      SINT1571
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES                   SINT1572
      IF(MW.LE.0) RETURN                                                SINT1573
      IF(MW.LE.PEMIN) GO TO 5                                           SINT1574
      CALL ERRMSG('ADDPLT',2)                                           SINT1575
      RETURN                                                            SINT1576
    5 TEMP=PEMAX                                                        SINT1577
      PRTEMP=PROB(IEMAX)                                                SINT1578
      IDUX=IEMAX                                                        SINT1579
      IDUM=IEMAX+1                                                      SINT1580
      Q=ONE-P                                                           SINT1581
      FB=MW/DM                                                          SINT1582
      INT=FB                                                            SINT1583
      FB=FB-INT                                                         SINT1584
```

```
C     CALCULATE NEW VALUES AT POINTS ON UPPER END OF PROB AND
C     FIND NEW PEMAX AND IEMAX
      PEMAX=PEMAX+MW
      IEMAX=PEMAX/DM
      DO 20 J=IDUX,IEMAX
      JINT=J-INT
      IF(JINT.EQ.IDUM) GO TO 10
      PRJINT=PROB(JINT)
      IF(JINT.EQ.IDUX) PRJINT=PRTEMP
      PROB(J)=PROB(J)+Q*(PRJINT-PROB(J)+FB*(PROB(JINT-1)-PRJINT))
      GO TO 15
   10 PROB(J)=Q*PRTEMP*(TEMP/DM-IDUM+FB)/(TEMP/DM-IDUM+ONE)
   15 IF(J.LT.IEMAX) PROB(J+1)=ZERO
      IF(PROB(J).LE.TRACE) GO TO 30
   20 CONTINUE
      TEMP=IEMAX*DM+EPS
      IF(TEMP.GT.PEMAX) PEMAX=TEMP
      GO TO 40
   30 PROB(J)=ZERO
      IEMAX=J
      PEMAX=IEMAX*DM+EPS
   40 IF(IEMAX.GT.IDIMEN) CALL ERRMSG('ADDPLT',1)
      J=IDUX
      JINT=J-INT
C     LOOP TO CONVOLVE IN NEW PLANT
   50 J=J-1
      IF(J.LE.IEMIN) RETURN
      JINT=JINT-1
      PROB(J)=PROB(J) +
     $ Q*(PROB(JINT)-PROB(J)+FB*(PROB(JINT-1)-PROB(JINT)))
      GO TO 50
      END
      FUNCTION PROBX(X)
C     SYSINT VERSION 10-15-71
C     EVALUATES PROB AT A PARTICULAR VALUE OF X MW
C     ************************************************************
```

```
C     COMMON VARIABLES                                                  SINT1621
      IMPLICIT REAL*8 (A-H,O-$)                                         SINT1622
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES                     SINT1623
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)   SINT1624
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI SINT1625
      REAL*8 MILLI                                                      SINT1626
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES                   SINT1627
      PROBX=ONE                                                         SINT1628
      IF(X.LE.PEMIN) RETURN                                             SINT1629
      PROBX=ZERO                                                        SINT1630
      IF(X.GE.PEMAX) RETURN                                             SINT1631
      FB=X/DM                                                           SINT1632
      ILO=FB                                                            SINT1633
      FB=FB-ILO                                                         SINT1634
      IF(ILO.GE.IEMAX) GO TO 10                                         SINT1635
      PROBX=PROB(ILO)+FB*(PROB(ILO+1)-PROB(ILO))                        SINT1636
      RETURN                                                            SINT1637
   10 PROBX=PROB(IEMAX)*(PEMAX-X)/(PEMAX-IEMAX*DM)                      SINT1638
      RETURN                                                            SINT1639
      END                                                               SINT1640
      FUNCTION SUSDNO(AVBSLF)                                           SINT1641
C     SYSINT VERSION 10-15-71                                           SINT1642
C     APPROXIMATES NUMBER OF STARTUPS AND SHUTDOWNS DURING THE PERIOD   SINT1643
C     AS A FUNCTION OF THE AVAILABILITY-BASED LOAD FACTOR, AVBSLF       SINT1644
C     ************************************************************      SINT1645
      IMPLICIT REAL*8 (A-H,O-$)                                         SINT1646
C     COMMON VARIABLES                                                  SINT1647
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES                     SINT1648
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)   SINT1649
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI SINT1650
      COMMON/SUSDF/F(20)                                                SINT1651
      REAL*8 MILLI                                                      SINT1652
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES                   SINT1653
      IF(AVBSLF.GE.ONE) AVBSLF=ONE-1.D-10                               SINT1654
      FB=20.D0*AVBSLF                                                   SINT1655
      ILO=FB                                                            SINT1656
```

```
      FB=FB-ILO                                                         SINT1657
      IF(AVBSLF.LT.0.05D0) GO TO 10                                     SINT1658
      SUSDNO=DAYS*(F(ILO)+FB*(F(ILO+1)-F(ILO)))                         SINT1659
      RETURN                                                            SINT1660
   10 SUSDNO=DAYS*FB*F(1)                                               SINT1661
      RETURN                                                            SINT1662
      END                                                               SINT1663
      SUBROUTINE PUNCHR(MODE)                                           SINT1664
C     SYSINT VERSION 11-2-71                                            SINT1665
C     PERFORMS PUNCHING OPERATIONS                                      SINT1666
C                                                                       SINT1667
C     NOTE THAT:                                                        SINT1668
C     1.  FOR PROGRAMMING MODULARITY, THIS SUBROUTINE PERFORMS PUNCHING SINT1669
C         OF OUTPUT, WHETHER ON CARDS, TAPE OR DIRECT ACCESS DEVICE.    SINT1670
C         THE ONLY EXCEPTION IS THE FINAL TOTALS NAMELIST /FNLTOT/      SINT1671
C         PUNCHED BY THE SYSGEN SUBROUTINE.                             SINT1672
C     2.  THIS SUBROUTINE IS DEPENDENT UPON THE IBM/360 UTILITY PROGRAM SINT1673
C         "IEBUPDTE"  (RELEASE 20).                                     SINT1674
C                                                                       SINT1675
C     **********************************************************        SINT1676
      IMPLICIT REAL*8 (A-H,O-$)                                         SINT1677
C     COMMON VARIABLES                                                  SINT1678
C     VARIABLES DIMENSIONED IN MULTIPLES OF MAXPLT, MAX.NO. OF STATIONS SINT1679
      COMMON/PLTDAT/IDNO(100),NAME(100),TYPE(100),SUSDHT(100),PNOM(100),SINT1680
     $NPTS(100),MWPT(5,100),HTRAT(5,100)                                SINT1681
      COMMON/PERDAT/AVLBTY(100),CSTBTU(100),STATUS(100),EXPHRS(100),    SINT1682
     $EXPBTU(100),EXPGWH(100),NORDER(500),COST(100),ENERGY(100),        SINT1683
     $SUPCST(100),MRGCST(5,100)                                         SINT1684
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES                     SINT1685
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)   SINT1686
      COMMON/FLOAT/EPS,TRACE,PKMW,SPNRES,CSTEMR                         SINT1687
      COMMON/TITLE/SGTITL(10),PDTITL(10)                                SINT1688
      COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1SINT1689
     $,IDSTRG,PCHMIN,PCHMAX,MBRNUM                                      SINT1690
      COMMON/LDGNFO/LDTYPE,LDTYPS,LOAD(50,25),NORDOP,NOENTY,NOBASE,     SINT1691
     $NOPEAK,NNORD                                                      SINT1692
```

```
      COMMON/CONSTS/ZERO,ONE,TWO,HALF,TEN,TENTH,HUNDRD,CENTI,THOUS,MILLI          SINT1693
      COMMON/LOGICL/MINI,MIDI,MAXI,NPM,PCHING                                     SINT1694
      COMMON/MAINT/MAINT(100,20)                                                  SINT1695
C     MAINT IS DIMENSIONED (MAXPLT,MAXPER/5)  THE 5 IS 5I1/INTEGER*2              SINT1696
      REAL*4 SUSDHT,PNOM,HTRAT                                                    SINT1697
      REAL*4 SUPCST,MRGCST                                                        SINT1698
      REAL*8 MILLI                                                                SINT1699
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX                          SINT1700
      INTEGER*2 IDNO,TYPE,NPTS,MWPT,NORDER,STATUS,MAINT,LOAD                      SINT1701
      LOGICAL*1 MINI,MIDI,MAXI,NPM,PCHING                                         SINT1702
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES                             SINT1703
      INTEGER*2 NTEST(100),NDXS(100),$N$/'N'/,LSTMOD/0/                           SINT1704
C     NTEST & NDXS DIMENSIONED MAXPLT                                             SINT1705
      REAL*4 A(5)                                                                 SINT1706
      IF(.NOT.PCHING) RETURN                                                      SINT1707
      MOD=MODE                                                                    SINT1708
      IF(MOD.LE.6) GO TO 10                                                       SINT1709
      IF(LSTMOD.NE.2.AND.LSTMOD.NE.3) GO TO 10                                    SINT1710
      MW=MODE                                                                     SINT1711
      MOD=LSTMOD+1                                                                SINT1712
   10 GO TO (100,200,300,400,500,600),MOD                                         SINT1713
C     STRATEGY INFORMATION                                                        SINT1714
  100 NUKES=0                                                                     SINT1715
      DO 110 N=1,NOSTNS                                                           SINT1716
      IF(TYPE(N).NE.$N$) GO TO 110                                                SINT1717
      NUKES=NUKES+1                                                               SINT1718
      NDXS(NUKES)=N                                                               SINT1719
  110 CONTINUE                                                                    SINT1720
      IF(NUKES.GT.0) GO TO 130                                                    SINT1721
C     SINCE NO NUKES, PUNCH ALL PLANTS                                            SINT1722
      DO 120 N=1,NOSTNS                                                           SINT1723
  120 NDXS(N)=N                                                                   SINT1724
      NUKES=NOSTNS                                                                SINT1725
      NPM=.FALSE.                                                                 SINT1726
  130 JMAINT=(NPERS+4)/5                                                          SINT1727
      CALL WHEN(A)                                                                SINT1728
```

```
      CALL DAYTIM
      N=1
      IPLACE=MBRNUM/1000000
      WRITE(PUNCH,911) NPM,MBRNUM,N,NPM,MBRNUM,A
      WRITE(PUNCH,912) NPM,IPLACE,IDSTRG,SGTITL,NUKES
      WRITE(PUNCH,913) (IDNO(NDXS(I)),NAME(NDXS(I)),MWPT(1,NDXS(I)),
     $MWPT(NPTS(NDXS(I)),NDXS(I)),NDXS(I),I=1,NUKES)
      WRITE(PUNCH,914) NPERS,JMAINT
      DO 140 N=1,NUKES
  140 WRITE(PUNCH,915) (MAINT(NDXS(N),J),J=1,JMAINT)
      GO TO 800
C     PERIOD INFORMATION
C     N.P.M. CHECK OF NORDER AND SET PCHMIN AND PCHMAX
  200 PCHMIN=-1
      PCHMAX=-1
      IF(.NOT.NPM) GO TO 260
      NSUM=0
      MSUM=0
      DO 210 NK=1,NUKES
      NDX=NDXS(NK)
      NTEST(NK)=1000*NPTS(NDX)+NDX
      IF(STATUS(NDX).NE.2) GO TO 210
      IF(NTEST(NK).GT.2000) NSUM=NSUM+NTEST(NK)
      MSUM=MSUM+1000+NDX
  210 CONTINUE
      NPMFAL=0
      NPMDEL=0
      DO 220 J=1,NNORD
      N=NORDER(J)
      IF(N.GT.2000) GO TO 230
      IF(TYPE(N-1000).EQ.$N$) MSUM=MSUM-N
  220 CONTINUE
  230 JLOW=J
      J=J-1
      IF(MSUM.NE.0) NPMFAL=100
      IF(NSUM.EQ.0) GO TO 250
```

```
      DO 240 J=JLOW,NNORD                                               SINT1765
      N=NORDER(J)                                                       SINT1766
      M=N-(N/1000)*1000                                                 SINT1767
      IF(TYPE(M).NE.$N$) NPMDEL=20                                      SINT1768
      DO 240 NK=1,NUKES                                                 SINT1769
      IF(N.EQ.NTEST(NK)) NSUM=NSUM-NTEST(NK)                            SINT1770
      IF(NSUM.EQ.0) GO TO 250                                           SINT1771
  240 CONTINUE                                                          SINT1772
      CALL ERRMSG('PUNCHR',9)                                           SINT1773
  250 PCHMAX=J                                                          SINT1774
      PCHMIN=JLOW-1                                                     SINT1775
      NPMFAL=NPMFAL+NPMDEL                                              SINT1776
      IF(NPMFAL.EQ.0) GO TO 260                                         SINT1777
      WRITE(WT,921) NPMFAL                                              SINT1778
      CALL ERRMSG('PUNCHR',11)                                          SINT1779
  260 WRITE(PUNCH,922) NPER,A,PDTITL,NPER,DM,DT,CSTEMR                  SINT1780
      IF(.NOT.NPM) MOD=4                                                SINT1781
      GO TO 800                                                         SINT1782
C     PROB AT NUCLEAR MINIMUMS                                          SINT1783
  300 M=MW                                                              SINT1784
      NTBSLD=0                                                          SINT1785
      DO 310 NK=1,NUKES                                                 SINT1786
      N=NDXS(NK)                                                        SINT1787
      IF(EXPHRS(N)+MILLI.LT.DT*CENTI*AVLBTY(N)) NTBSLD=NTBSLD+1         SINT1788
  310 M=M+MWPT(NPTS(N),N)-MWPT(1,N)                                     SINT1789
      LPTS=MAX0(IDINT(M/DM)-IEMIN+2,1)                                  SINT1790
      LPTS=MIN0(LPTS,IEMAX-IEMIN)                                       SINT1791
      IF(NTBSLD.EQ.0.OR.MW.LE.PEMIN) GO TO 320                          SINT1792
      NPMFAL=NPMFAL+3                                                   SINT1793
      WRITE(WT,932)   NPMFAL,NTBSLD,(EXPHRS(NDXS(NK)),NK=1,NUKES)       SINT1794
      CALL ERRMSG('PUNCHR',12)                                          SINT1795
  320 WRITE(PUNCH,931) MW,IEMIN,LPTS,(PROB(IEMIN+I),I=1,LPTS)           SINT1796
      WRITE(PUNCH,933)NPMFAL,NTBSLD,(EXPHRS(NDXS(NK)),NK=1,NUKES)       SINT1797
      GO TO 800                                                         SINT1798
C     PROB AT NUCLEAR MAXIMUMS                                          SINT1799
  400 LPTS=MAX0(IDINT(MW/DM)-IEMIN+2,1)                                 SINT1800
```

```
      LPTS=MINO(LPTS,IEMAX-IEMIN)                                       SINT1801
      WRITE(PUNCH,941) MW,IEMIN,LPTS,(PROB(IEMIN+I),I=1,LPTS)           SINT1802
      GO TO 800                                                         SINT1803
C     FINAL PERIOD RESULTS                                              SINT1804
C                                                                       SINT1805
C     NOTE SUBROUTINE SYSGEN HAS ALREADY PUNCHED THE FINAL TOTALS       SINT1806
C                                                                       SINT1807
  500 WRITE(PUNCH,951) (CSTBTU(NDXS(I)),AVLBTY(NDXS(I)),ENERGY(NDXS(I)),SINT1808
     $EXPHRS(NDXS(I)),EXPGWH(NDXS(I)),EXPBTU(NDXS(I)),COST(NDXS(I)),    SINT1809
     $I=1,NUKES)                                                        SINT1810
      MOD=1                                                             SINT1811
      IF(NPER.LT.NPERS) GO TO 800                                       SINT1812
      WRITE(PUNCH,952) NPM,MBRNUM,A                                     SINT1813
      GO TO 700                                                         SINT1814
C     ABORT CAUSED BY DETECTION OF SEVERE ERROR                         SINT1815
  600 IF(LSTMOD.GT.0) WRITE(PUNCH,961)NPM,MBRNUM,A                      SINT1816
      PUNCH=PUNCH-1000                                                  SINT1817
      PCHING=.FALSE.                                                    SINT1818
  700 LSTMOD=0                                                          SINT1819
      RETURN                                                            SINT1820
  800 LSTMOD=MOD                                                        SINT1821
      RETURN                                                            SINT1822
  911 FORMAT('./   ADD   NAME=',L1,I7,',LEVEL=',Z2,',LIST=ALL'/         SINT1823
     $'----------BEGIN  STRATEGY WITH NAME=',L1,I7,' ON ',2A4,' AT ',   SINT1824
     $3A4,'----------')                                                 SINT1825
  912 FORMAT(L3,I1,I6,10A7/I5)                                          SINT1826
  913 FORMAT(I5,A5,2I5,I10)                                             SINT1827
  914 FORMAT('NUKES'' MAINT.DATA FOR',T22,I4,' PERIODS   (',T41,I3,     SINT1828
     $' VALUES)')                                                       SINT1829
  915 FORMAT(16I5)                                                      SINT1830
  921 FORMAT('ONPMFAL=',I3,4X,'(100=FIRST REASON, 20=SECOND REASON,',   SINT1831
     $' OR 120=BOTH REASONS FOR ERROR 11 (HEXADECIMAL B)')              SINT1832
  922 FORMAT(13('.'),I3,'TH PERIOD TO FOLLOW SIMULATED  ',5A4,13('.')/  SINT1833
     $10A8/I10,3F10.4)                                                  SINT1834
  931 FORMAT('MIN  ',3I5,6F10.9/(8F10.9))                               SINT1835
  932 FORMAT('ONPMFAL=',I3,6X,'NTBSLD=',I3/                             SINT1836
```

```
     $' EXPHRS(NK),NK=1,NUKES AT CALL TO PCHMIN :',8F10.4/(12F10.4))
  933 FORMAT(2I5,(7F10.4))
  941 FORMAT('MAX  ',3I5,6F10.9/(8F10.9))
  951 FORMAT(' CSTBTU  AVLBTY  ENERGY   EXPHRS',T42,'EXPGWH',T58,'EXPBTU
     $',T75,'COST'/(2F8.4,F8.2,F10.3,F16.5,2F15.0))
  952 FORMAT(
     $'---------END OF STRATEGY WITH NAME=',L1,I7,' ON ',2A4,' AT ',
     $3A4,'---------'//)
  961 FORMAT(
     $'---------ABORT  STRATEGY WITH NAME=',L1,I7,' ON ',2A4,' AT ',
     $3A4,'---------'//)
      END
      SUBROUTINE ERRMSG(SUBR,JERR)
C     SYSINT VERSION  1-01-73
C     WRITES OUT ALL ERROR MESSAGES
C     ***********************************************************
      IMPLICIT REAL*8 (A-H,O-$)
C     COMMON VARIABLES
C     OTHER VARIABLES COMMON TO SEVERAL SUBROUTINES
      COMMON/INTEGR/RD,WT,PUNCH,CARD,TAPE,ERRCOD,NOSTNS,NPER,NPERS,NPER1
     $,IDSTRG,PCHMIN,PCHMAX,MBRNUM
      INTEGER RD,WT,PUNCH,CARD,TAPE,ERRCOD,PCHMIN,PCHMAX
C     END OF STATEMENTS COMMON TO SEVERAL SUBROUTINES
      DATA NPRINT/0/,$QUIT$/' QUIT'/
      IERR=JERR
  100 ERRCOD=16*ERRCOD+IERR
      IF(ERRCOD.GT.8*16**6) IERR=8
      NPRINT=NPRINT+1
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12),IERR
    1 WRITE(WT,901) SUBR,ERRCOD,NPRINT
      GO TO 1000
    2 WRITE(WT,902) SUBR,ERRCOD,NPRINT
      GO TO 1000
    3 WRITE(WT,903) SUBR,ERRCOD,NPRINT
      RETURN
    4 WRITE(WT,904) SUBR,ERRCOD,NPRINT
```

```
      GO TO 1000
    5 WRITE(WT,905) SUBR,ERRCOD,NPRINT
      RETURN
    6 WRITE(WT,906) SUBR,ERRCOD,NPRINT
      GO TO 1000
    7 WRITE(WT,907) SUBR,ERRCOD,NPRINT
      GO TO 1000
    8 WRITE(WT,908) SUBR,ERRCOD,NPRINT,NPRINT
      STOP
    9 WRITE(WT,909) SUBR,ERRCOD,NPRINT
      GO TO 1000
   10 WRITE(WT,910) SUBR,ERRCOD,NPRINT
      IERR=8
      GO TO 100
   11 WRITE(WT,911) SUBR,ERRCOD,NPRINT
      RETURN
   12 WRITE(WT,912) SUBR,ERRCOD,NPRINT
      RETURN
 1000 NPRINT=NPRINT+1
      WRITE(WT,999) NPRINT
      CALL QUIT
      SUBR=$QUIT$
      IERR=10
      GO TO 100
  901 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'  IEMAX GREATER THAN DIMENSION OF PROB ARRAY   ',
     $T131,'*',/,' ',130('*'),I2)
  902 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'  CAPACITY OF A PLANT GREATER THAN ',
     $'MINIMUM LOAD',
     $T131,'*',/,' ',130('*'),I2)
  903 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'WARNING -   RELERR &/OR |DP| GREATER THAN 0.001',
     $T131,'*',/,' ',130('*'),I2)
  904 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'  NEW PROB VIOLATES PROPERTIES OF A CDF        ',
```

```
     $T131,'*',/,' ',130('*'),I2)
 905 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $' WARNING : ERROR IN EXPECTED MW OUTAGES GREATER',
     $' THAN 0.01% ',
     $T131,'*',/,' ',130('*'),I2)
 906 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'INPUT DECK HAS IMPROPER SEQUENCE &/OR CARD     ',
     $T131,'*',/,' ',130('*'),I2)
 907 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'INVALID OR INCONSISTENT IDNO ENCOUNTERED       ',
     $T131,'*',/,' ',130('*'),I2)
 908 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'SUPSIM ENCOUNTERED STOP CARD, ERRMSG CALLED ONCE TOO OFTEN OR O',
     $'THER FATAL ERROR',  T131,'*'/' * DURING THIS ENTIRE RUN, ERRMSG',
     $' PRINTED A TOTAL OF ',I3,' ERROR MESSAGES JUST LIKE (AND ',
     $'INCLUDING) THIS ONE',
     $T131,'*',/,' ',130('*'),I2)
 909 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'INPUT NORDER ERROR SUCH AS TOO FEW/MANY VALVE',
     $' POINTS, BAD IDNO OR UNLISTED ON-LINE PLANT',
     $T131,'*',/,' ',130('*'),I2)
 910 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'  "QUIT" EXECUTED A RETURN TO "ERRMSG"        ',
     $T131,'*',/,' ',130('*'),I2)
 911 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'BASE NUCL. W/IN NUCL. NON-MINIMUMS OR NON-MIN',
     $'IMUM NON-NUCL.V.PTS. PRECEDE SOME NUCL.V.PTS.',
     $T131,'*',/,' ',130('*'),I2)
 912 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'MINIMUM LOAD TOO LOW TO KEEP NUKES ON ALL THE TIME',
     $T131,'*',/,' ',130('*'),I2)
 999 FORMAT(/' ',130('*')/,' * PREVIOUS ERROR SEVERE ENOUGH TO',
     $' INVALIDATE FURTHER COMPUTATIONS. THEREFORE,RETURNING',
     $' CONTROL TO SUPSIM.',
     $T131,'*',/,' ',130('*'),I2)
      END
```

```
      SUBROUTINE CMPTIM(LV,ENT)
C     SYSINT VERSION 10-15-71
C     PRINTS TIME OF INTRA-SUBROUTINE TRANSFERS OR DATE&TIME
C     "TIMING" IS AN M.I.T. INTERNAL SUBROUTINE THAT RETURNS THE CPU TIME
C     IN HUNDREDTHS OF SECONDS.
C     "WHEN" IS AN M.I.T. INTERNAL SUBROUTINE THAT RETURNS THE DATE AND
C     TIME IN THE FOLLOWING 5A4 FORMAT:  MM/DD/YY HR*MI*SS.FF
      COMMON/INTEGR/RD,WT
      INTEGER RD,WT
      DIMENSION A(5)
      DOUBLE PRECISION LV,ENT
      INTEGER  TNOW,TSTART,TREL
      CALL TIMING(TNOW)
      TREL=TNOW-TSTART
      IF(TREL.LT.0)   TREL=TREL+8640000
      TI=TREL/100.
      WRITE(WT,10)LV,ENT,TI
      RETURN
      ENTRY STRTIM
      CALL TIMING(TSTART)
      ENTRY DAYTIM
      CALL WHEN(A)
      WRITE(WT,20) A
      RETURN
   10 FORMAT(/,T103,29('*'),/,T103,'* LV.  ',A6,T131,'*',/,
     $T103,'* ENT. ',A6,' @',F7.2,' SEC. *',/,T103,29('*'),/)
   20 FORMAT(/T103,29('*')/T103,'*    DATE =    ',2A4,T131,'*'/
     $T103,'*    TIME = ',  3A4,T131,'*'/T103,29('*')/)
      END
********************************************************************** 00000000
*                                                                    * 00000010
*                  ASSEMBLER LANGUAGE SUBROUTINE ERASE               * 00000011
*                        WRITTEN BY JOHN W. KIDSON                   * 00000012
*                      MIT DEPARTMENT OF METEOROLOGY                 * 00000014
*                                                                    * 00000016
*       TO SET ELEMENTS OF REAL OR INTEGER ARRAYS TO ZERO. A1,A2,... * 00000020
```

```
*         ARE ARRAY NAMES AND N1,N2,... ARE INTEGER VALUES OR           * 00000030 SINT1981
*         EXPRESSIONS GIVING THE ARRAY SIZES.                           * 00000040 SINT1982
**        I.E. - CALL ERASE(C,26*31,N,7*31,E,254)                      ** 00000050 SINT1983
*                                                                       * 00000060 SINT1984
*********************************************************************** 00000070 SINT1985
ERASE     START 0                                                         00000080 SINT1986
          SAVE  (14,12),,*                                                00000090 SINT1987
          BALR  12,0                                                      00000100 SINT1988
          USING *,12                                                      00000110 SINT1989
          SR    0,0                                                       00000120 SINT1990
          SR    2,2             PARAMETER LIST INDEX=0                    00000130 SINT1991
          L     6,=F'4'                                                   00000140 SINT1992
E1        L     3,0(2,1)        LOAD 3 WITH ARRAY ADDRESS                 00000150 SINT1993
          L     4,4(2,1)        LOAD 4 WITH ADDRESS OF ARRAY LENGTH       00000160 SINT1994
          L     7,0(4)          LOAD 7 WITH ARRAY LENGTH-1 TIMES 4        00000170 SINT1995
          SLA   7,2                                                       00000180 SINT1996
          SR    7,6                                                       00000190 SINT1997
          SR    5,5                                                       00000200 SINT1998
E2        ST    0,0(5,3)        STORE ZERO                                00000210 SINT1999
          BXLE  5,6,E2                                                    00000220 SINT2000
          LTR   4,4             TEST FOR LAST ARGUMENT IN LIST            00000230 SINT2001
          BM    RETN                                                      00000240 SINT2002
          A     2,=F'8'                                                   00000250 SINT2003
          B     E1              PICK UP NEXT ARGUMENT PAIR                00000260 SINT2004
RETN      RETURN (14,12),T                                                00000270 SINT2005
          END                                                             00000280 SINT2006
*********************************************************************** 00000290 SINT2007
```

```
C*************************************************************************
C*                                                                       *
C*    S Y S O P T   :   AN ELECTRIC UTILITY SYSTEM OPTIMIZATION MODEL    *
C*                          WRITTEN BY PAUL F. DEATON                    *
C*                      M.I.T. DOCTORAL THESIS,    MARCH  1973           *
C*                                                                       *
C*************************************************************************
C     SYSOPT MAIN PROGRAM
C     SYSOPT VERSION 12-16-72
C*********** DEFINITIONS OF IMPORTANT VARIABLES ***************************
C     $NKPRD = DIRECT NUCLEAR PRODUCTION FUEL COST (10**3 $)
C     $NKSUS = NUCLEAR STARTUP & SHUTDOWN COST (10**3 $)
C     $NKTOT = TOTAL COST OF NUCLEAR PRODUCTION =$NKPRD+$NKSUS  (10**3 $)
C     $NNPRD = DIRECT NON-NUCL. PRODUCTION FUEL COST (10**3 $)
C     $NNSUS = NON-NUCL. STARTUP & SHUTDOWN COST (10**3 $)
C     $NNTOT = TOTAL COST OF NON-NUCL. PROD. = $NNPRD + $NNSUS (10**3 $)
C     $SBTOT = PROD$ + SUSD$ , TOTAL COST OF PRODUCTION W/IN SYSTEM
C     ALPHA  = LINEAR SOLUTION VARIANCE PARAMETER (PER GWHE)
C     AVL    = REACTOR PERFORMANCE PROBABILITY (PER CENT)
C     BASCFA = AVAIL.-BASED CAP. FACTOR FOR REACTOR BASE PORTION ( % )
C     BASVAR = BASE VARIANCE FOR SOLUTION COMPARISON
C     BETAP  = CONSTANT SOLUTION VARIANCE PARAMETER
C     CAVG   = AVERAGE C.D.F. VALUE FOR PERIOD
C     CDFAVG = AVERAGE C.D.F.
C     CDFMAX = C.D.F. AT LVLMAX
C     CDFMIN = C.D.F. AT LVLMIN
C     CORDTL = OPTIMUM IN-CORE DETAIL PRINT OPTION (0=NO,1=YES)
C     CYCNUM = CUMULATIVE CYCLE NUMBER FOR GIVEN R-C
C     CYCRMX = MAXIMUM R-C FOR REACTOR
C     CYCRNG = RANGE OF PERIODS COVERED BY EACH R-C
C     CYCXS  = NUMBER OF EXCESS CYCLES BEYOND HORIZON OF INTEREST
C     DELTAL = DELTA CAPACITY FACTOR LIMITS = SQRT(-SLNCRT)
C     DM     = DMW
C     DMW    = EQUIVALENT LOAD STEP SIZE (MW)
C     DT     = DTH
C     DTH    = PERIOD DURATION (HOURS)
                                                          PAGE   1
```

```
C       DYDWN  = DOWN TIME FOR EXCESS CYCLE (YEARS)                          SOPT0037
C       DYHOLD = POST-HORIZON TIME UNTIL END OF SPLIT CYCLE (YEARS)          SOPT0038
C       DYUP   = UP   TIME FOR EXCESS CYCLE (YEARS)                          SOPT0039
C       ECS    = EMERGENCY POWER COST ($/MWHE)                               SOPT0040
C       ECUPLM = UPPER LIMIT ON EC'S IMPOSED BY IN-CORE MODEL (GWHE)         SOPT0041
C       ELAME  = SANDWICHED TABLE OF EC'S, LAMBDAS & EC'S (GWHE,$/MWHE)      SOPT0042
C       EMRP$  = COST OF EMERGENCY POWER PURCHASES (10**3 $)                 SOPT0043
C       EXPDEM = EXP. CUSTOMER ENERGY DEMAND (GWHE)                          SOPT0044
C       EXPEMR = EXP. EMERGENCY ENERGY PURCHASED (GWHE)                      SOPT0045
C       EXPGEN = EXP. UTILITY TOTAL GENERATION (GWHE)                        SOPT0046
C       EXPGWH = SYSINT EXPECTED GENERATION BY EACH REACTOR (GWHE)           SOPT0047
C       FINTST = FINE-GRAINED SHAPE TEST FOR THE PERIOD                      SOPT0048
C       FINVAR = FINE-GRAINED VARIANCE FOR THE PERIOD (THESIS S**2)          SOPT0049
C       GESFRS = FIRST GUESS OPTION(0=NONE,1=SYSINT,2=MRGCST,3=CA.EC,4=EC)   SOPT0050
C       GMESH  = INCREMENTAL SPACING USED FOR ARC TYPES 2 & 3 (GWHE)         SOPT0051
C       GWHOLD = GWHE HELD OVER FOR LATER PRODUCTION IN SPLIT CYCLE          SOPT0052
C       GWHPER = GWHE PER UNIT DM UNDER CDF                                  SOPT0053
C       GWHXS  = EC FOR EXCESS CYCLE (GWHE)                                  SOPT0054
C       IAUX   = TOTAL NUMBER OF ARCS TO AUXILIARY R-C NODE                  SOPT0055
C       IAUXM  = IAUX-1                                                      SOPT0056
C       IDNO   = REACTOR I.D. NUMBER                                         SOPT0057
C       IDSTRG = STRATEGY I.D. NUMBER                                        SOPT0058
C       IEMAX  = PEMAX/DM                                                    SOPT0059
C       IEMIN  = PEMIN/DM                                                    SOPT0060
C       INSTAT = INITIAL STATE OF REACTOR AT START OF PERIOD 1 (CF. 'S' )    SOPT0061
C       ITER   = INNER COST ITERATION NUMBER                                 SOPT0062
C       JBKWRD = NUMBER OF BACKWARD ARCS OF TYPE 7                           SOPT0063
C       JFRPBK = JFRWRD + JBKWRD                                             SOPT0064
C       JFRWRD = NUMBER OF FORWARD  ARCS OF TYPE 7                           SOPT0065
C       KC     = UNIT TRANSPORTATION COST ACROSS ARC ($/GWHE)                SOPT0066
C       KL     = ARC CAPACITY LOWER LIMIT  (GWHE)                            SOPT0067
C       KU     = ARC CAPACITY UPPER LIMIT  (GWHE)                            SOPT0068
C       KX     = ARC CAPACITY USED         (GWHE)                            SOPT0069
C       LVLMIN = LVLMN                                                       SOPT0070
C       LVLMN  = POWER LEVEL AT END OF MINIMUMS                              SOPT0071
C       LVLMAX = LVLMX                                                       SOPT0072
```

```
C       LVLMX  = POWER LEVEL AT END OF MAXIMUMS                                SOPT0073
C       MAX    = REACTOR-TO-PERIOD MAX. GWHE CONTRIB. TO NUCL. POTENTIAL       SOPT0074
C       MAXTST = MAXIMUM POSSIBLE SHAPE TEST FOR THE PERIOD                    SOPT0075
C       MAXVAR = MAXIMUM POSSIBLE VARIANCE FOR THE PERIOD                      SOPT0076
C       MESH   = SEQUENCE OF GMESH VALUES TO BE USED IN CONVERGENCE (GWHE)     SOPT0077
C       MIDCYC = REACTOR IN MID-CYCLE AT START OF PERIOD 1 ?                   SOPT0078
C       MIN    = REACTOR-TO-PERIOD MIN. GWHE CONTRIB. TO NUCL. POTENTIAL       SOPT0079
C       MWD    = INCREMENT OF CAPACITY AVAILABLE FOR LOAD FOLLOWING (MW)       SOPT0080
C       MWINST = UTILITY INSTALLED CAPACITY (MW)                               SOPT0081
C       MWMAX  = REACTOR MAXIMUM LOAD (MW)                                     SOPT0082
C       MWMIN  = REACTOR MINIMUM LOAD (MW)                                     SOPT0083
C       MWMRGN = ON-LINE CAPACITY MARGIN ABOVE FORECAST PEAK (MW)              SOPT0084
C       MWONLN = UTILITY  ON-LINE  CAPACITY (MW)                               SOPT0085
C       MWPEAK = FORECAST PEAK CUSTOMER DEMAND (MW)                            SOPT0086
C       MWSPIN = SPINNING RESERVE REQUIREMENT (MW)                             SOPT0087
C       MXARCS = MAXIMUM ALLOWED NUMBER OF ARCS  IN O-O-K                      SOPT0088
C       MXESX2 = FIRST DIMENSION OF ELAME = (MAX.NO.EC'S IN COL.) * 2          SOPT0089
C       MXITER = MAXIMUM ITERATIONS TO BE ATTEMPTED                            SOPT0090
C       MXNODS = MAXIMUM ALLOWED NUMBER OF NODES IN O-O-K                      SOPT0091
C       MXNPER = MAXIMUM ALLOWED NUMBER OF PERIODS IN SYSOPT STUDY             SOPT0092
C       MXRCRS = MAXIMUM ALLOWED NUMBER OF REACTORS IN STRATEGY                SOPT0093
C       MXRCYC = MAXIMUM ALLOWED CYCLES FOR A SINGLE REACTOR                   SOPT0094
C       NCYCT  = CUMULATIVE NUMBER OF R-C'S                                    SOPT0095
C       NMESH  = NUMBER OF MESHES TO BE READ IN                                SOPT0096
C       NP     = PERIOD INDEX                                                  SOPT0097
C       NPERIN = NUMBER OF PERIODS IN SYSINT SIMULATION OUTPUT                 SOPT0098
C       NPERS  = NUMBER OF CYCLES COMPRISING TIME HORIZON OF INTEREST          SOPT0099
C       NPERSP = NPERS + 1                                                     SOPT0100
C       NPIN   = COMPUTER DEVICE NUMBER FOR NET.PROG. INPUT                    SOPT0101
C       NPM    = NUCLEAR POWER MANAGEMENT STUDY ?                              SOPT0102
C       NPMFAL = SYSINT ERROR INDICATION THAT SYSOPT N.P.M. MAY FAIL           SOPT0103
C       NPOT   = COMPUTER DEVICE NUMBER FOR NET.PROG. OUTPUT                   SOPT0104
C       NR     = REACTOR INDEX                                                 SOPT0105
C       NRCRS  = NUMBER OF REACTORS IN THE STRATEGY                            SOPT0106
C       NTBSLD = NUMBER OF REACTORS NOT BASE-LOADED IN THE PERIOD              SOPT0107
C       OPHRS  = SYSINT OPERATING HOURS FOR REACTOR                            SOPT0108
```

```
C       CPRCCR = IN-CORE PRINT OPTIONS TO BE USED FOR OPTIMUM SOLUTION           SOPT0109
C       PARCAL = ARC TYPES PRINTED FOR ALL O-O-K SOLUTIONS                       SOPT0110
C       PARCON = ARC TYPES PRINTED FOR CONVERGED O-O-K SOLUTIONS                 SOPT0111
C       PARCOP = ARC TYPES PRINTED FOR OPTIMUM O-O-K SOLUTION                    SOPT0112
C       PCONVG = PER CENT GMESH USED FOR CONVERGENCE TEST                        SOPT0113
C       PCDELA = PERIOD CAP. FACT. RANGE CORRECTION (PER CENT DELTAL)            SOPT0114
C       PDELIM = PERIOD DELIMITING CARD                                          SOPT0115
C       PDTITL = PERIOD TITLE CARD                                               SOPT0116
C       PEMAX  = MAXIMUM EQUIVALENT LOAD CONSIDERED  (MW)                        SOPT0117
C       PEMIN  = MINIMUM EQUIVALENT LOAD             (MW)                        SOPT0118
C       PLOFL  = PROBABILITY OF LOSS OF LOAD (FRACTION)                          SOPT0119
C       PROB   = CUMULATIVE DENSITY FUNCTION (C.D.F.) FOR EQUIVALENT LOAD        SOPT0120
C       PROD$  = DIRECT PRODUCTION FUEL COST (10**3 $)                           SOPT0121
C       PVFACT = MID-PERIOD PRESENT VALUE FACTOR (FRACTION)                      SOPT0122
C       PVRATE = PRESENT VALUE RATE (FRACTION PER YEAR)                          SOPT0123
C       RC     = REACTOR-CYCLE (R-C) INDEX                                       SOPT0124
C       RD     = COMPUTER DEVICE NUMBER FOR CARD READER                          SOPT0125
C       RDFACT = ROUND-OFF CORRECTION FACTOR FOR O-O-K 'S INTEGER EC'S           SOPT0126
C       REJLVL = REJECTION LEVEL FOR FINVAR-SLNWSR                               SOPT0127
C       S      = REACTOR STATUS DURING PERIOD (0=NONE,1=DOWN,2=UP)               SOPT0128
C       SGTITL = STRATEGY TITLE                                                  SOPT0129
C       SIOT   = COMPUTER DEVICE NUMBER FOR SYSINT OUTPUT                        SOPT0130
C       SLNCRT = SOLUTION SHAPE CRITERION = FINVAR-SLNWSR-REJLVL (.GE.0)         SOPT0131
C       SLNWSR = SOLUTION WTD. SUM OF SQUARES OF RESIDUALS (THESIS W**2)         SOPT0132
C       SP$    = PRESENT VALUE SUMS OF VARIOUS PERIOD COSTS                      SOPT0133
C       SUSD$  = SYSTEM STARTUP & SHUTDOWN COST (10**3 $)                        SOPT0134
C       TEY    = TIME AT  END  OF CYCLE (YEARS)                                  SOPT0135
C       TH$CON = CONVERGENCE CRITERION ON SYSTEM NUCLEAR COST (10**3 $)          SOPT0136
C       TOTAL$ = TOTAL SYSTEM COST = $NKTOT + $NNTOT + EMRP$    (10**3 $)        SOPT0137
C       TOY    = OPERATING TIME OF CYCLE (YEARS)                                 SOPT0138
C       TSY    = TIME AT START OF CYCLE (YEARS)                                  SOPT0139
C       UNSRVD = SECOND ESTIMATE OF UNSERVED ENERGY,EXPEMR (GWHE)                SOPT0140
C       WT     = COMPUTER DEVICE NUMBER FOR PRINTER                              SOPT0141
C       XNKGEN = EXP. NUCLEAR GENERATION (GWHE)                                  SOPT0142
C       XNNGEN = EXP. NON-NUCL. GENERATION (GWHE)                                SOPT0143
C       YBASE  = BASE YEAR FOR PRESENT VALUING                                   SOPT0144
```

```
C      YEND   = END POINT OF PERIOD (YEARS)
C      YMID   = MID-POINT OF PERIOD (YEARS)
C      YSTART = YEAR OF START OF FIRST PERIOD IN THE STRATEGY
C*********** END OF DEFINITIONS ****************************************
       IMPLICIT INTEGER(C,G)
       REAL*8 RDFACT,SGTITL
       COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCCNVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
     $,GESFRS,ECUPLM(18),CORDTL,OPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
       INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
       LOGICAL NPM,OPRCOR
       LOGICAL OPTRCH,SHPSOK
       REAL*8 $NKPRD
       DIMENSION X(20)
       DATA $STOP$,$STRA$,$NEWB$,$COMP$/'STOP','STRA','NEW ','COMP'/
       WRITE(WT,903)
       WRITE(WT,900)
    10 CALL STRTIM(WT)
    20 READ(RD,901) X
       WRITE(WT,902)X
       IF(X(1).EQ.$STOP$) CALL CPERR('SYSOPT',8)
       IF(X(1).EQ.$STRA$) GO TO 30
       IF(X(1).EQ.$COMP$) GO TO 40
       IF(X(1).NE.$NEWB$) CALL CPERR('SYSOPT',6)
       CALL CMPTIM('SYSOPT','ICNPUT')
       CALL ICNPUT
       CALL CMPTIM('ICNPUT','SYSOPT')
       GO TO 20
    30 CALL CMPTIM('      ','INPUT ')
       CALL RDOPTN
       CALL RDSTRG
       X(1)=LOC(10,0,0,0)
       CALL RDPERS
       CALL ASMTYS
```

```
      CALL WTPERS
      CALL SETUPN
      CALL SETLPT
      GO TO 20
   40 CALL CMPTIM('INPUT ','CALCS ')
      $NKPRD=0.0D0
   50 CALL CONVRG(OPTRCH,$NKPRD)
      CALL CHKSHP(SHPSOK)
      IF(ITER.LT.MXITER.AND.OPTRCH.AND..NOT.SHPSOK) GO TO 50
      CALL EDTSHP(SHPSOK)
      CALL OPTMUM(OPTRCH,$NKPRD)
      CALL CMPTIM('CALCS ','      ')
      IF(.TRUE.) GO TO 10
      STOP
  900 FORMAT(T31,72('*')/T31,'*',T102,'*'/T31,'*',T37,'S Y S O P T    :
     $ AN ELECTRIC UTILITY SYSTEM OPTIMIZATION MODEL',T102,'*'/
     $T31,'*',T64,'WRITTEN BY PAUL F. DEATON',T102,'*'/
     $T31,'*',T58,'M.I.T. DOCTORAL THESIS,   MARCH  1973 ',T102,'*'/
     $T31,'*',T102,'*'/T31,72('*')//
     $T56,'VERSION 12-16-72')
  901 FORMAT(20A4)
  902 FORMAT('0SYSOPT READ : ',1H',20A4,1H')
  903 FORMAT('0'/'0'/'0')
      END
      BLOCK DATA
C     INITIALIZES COMMON BLOCKS AND DIMENSIONS O-O-K ARRAYS
C     SYSOPT VERSION 12-16-72
      IMPLICIT INTEGER(C,G)
      REAL*8 RDFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
     $,GESFRS,ECUPLM(18),CORDTL,CPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
      LOGICAL NPM,CPRCOR
```

```
      LOGICAL MIDCYC
      INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX
      COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),
     $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),IDNO(15),GWHOLD(15),MWD(15)
     $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)
     $,DYHOLD(15),TOY(18,15)
      COMMON/OCKCOM/KIX,KOX,KQ1X,KQ2X,KQ3X,KQ4X,KQ5X
      COMMON /KL/KL/KC/KC/KU/KU/KX/KX/NL/NL/NN/NN/NP/NP/IJ/IJ/IL/IL
      COMMON /JL/JL/JI/JI
      DIMENSION KL(3500),KC(3500),KU(3500),KX(3500),NL( 700)
      DIMENSION NN(1400),NP( 700),IJ(2100),IL( 701),JL( 701),JI(3500)
      COMMON/PDPERM/S(100,15),ALPHA(100,15),BETAP(100,15),FINVAR(100)
      INTEGER*2 S
      COMMON/PDTEMP/NPMFAL(100),NTBSLD(100),OPHRS(100,15),LVLMN(100),
     $LVLMX(100),PDELIM(20,100),PDTITL(20,100),DMW(100),DTH(100),ECS(100
     $),R4(13,100),R8(12,100),YMID(100),YEND(100),PVFACT(100),AVL(100,
     $15),EXPGWH(100,15),CAVG(100),BASVAR(100),FINTST(100),MAXVAR(100),
     $MAXTST(100),MIN(100,15),MAX(100,15),BASCFA(100,15)
      REAL MAXVAR,MAXTST,CAVG
      REAL*8 PVFACT,R8
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
     $,LVLMIN,LVLMAX
      REAL*8 DM,DT,GWHPER,DAYS,PEMIN,PEMAX,PROB
      COMMON/FINALS/S4,SA4,SP4,SL4,SP8
      REAL*8 S4(13),SA4(13),SP4(13),SL4(13),SP8(13)
      COMMON/PRINTS/RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS,KRD,KWT
      LOGICAL RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS
      COMMON/SHPINF/SLNCRT(100),SLNWSR(100),ITRSHP,PDWSBD(100)
      LOGICAL PDWSBD
      DATA MXESX2/40/
      DATA MXRCYC/18/
      DATA MXNPER/100/
      DATA MXRCRS/15/
      DATA MXARCS/3500/
      DATA MXNODS/ 700/
      DATA RD/5/
```

```
      DATA WT/6/
      END
      SUBROUTINE RDOPTN
C     READS IN DATA PERTINENT DIRECTLY TO SYSOPT
C     SYSOPT VERSION 12-16-72
      IMPLICIT INTEGER(C,G)
      REAL*8 RDFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPOT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
     $,GESFRS,ECUPLM(18),CORDTL,CPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
      LOGICAL NPM,OPRCOR
      LOGICAL MIDCYC
      INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX
      COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),
     $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),IDNO(15),GWHOLD(15),MWD(15)
     $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)
     $,DYHOLD(15),TOY(18,15)
      READ (RD,901) NPM,IDSTRG,NRCRS
      WRITE(WT,911) NPM,IDSTRG,NRCRS
      READ (RD,907) SIOT,NPIN,NPOT,PARCAL,PARCON,PARCOP,CORDTL,OPRCOR
      WRITE(WT,912) SIOT,NPIN,NPOT,PARCAL,PARCON,PARCOP,CORDTL,OPRCOR
      READ (RD,903) PVRATE,YBASE,YSTART,PCONVG,TH$CON,PCDELA,REJLVL,
     $NPERS,GESFRS,MXITER,IAUX,JFRWRD,JBKWRD
      CALL PVINIT(PVRATE)
      WRITE(WT,913) PVRATE,YBASE,YSTART,PCONVG,TH$CON,PCDELA,REJLVL,
     $NPERS,GESFRS,MXITER,IAUX,JFRWRD,JBKWRD
      JFRPBK=JFRWRD+JBKWRD
      IF(GESFRS-2) 20,5,12
    5 WRITE(WT,916) (I,I=2,MXRCYC)
      DO 10 NR=1,NRCRS
      READ(RD,906) (ELAME(NR,I),I=1,MXRCYC)
   10 WRITE(WT,917) NR,(ELAME(NR,I),I=1,MXRCYC)
      GO TO 20
```

```
 12 WRITE(WT,918) (I,I=2,MXRCYC)
    DO 15 NR=1,NRCRS
    READ(RD,906) (ELAME(NR,I),I=1,MXRCYC)
 15 WRITE(WT,919) NR,(ELAME(NR,I),I=1,MXRCYC)
 20 READ (RD,902) NMESH,(MESH(N),N=1,NMESH)
    WRITE(WT,914) NMESH,(MESH(N),N=1,NMESH)
    READ (RD,905)(IDNO(NR),INSTAT(NR),CYCXS(NR),GWHOLD(NR),DYHOLD(NR),
   $(DYDWN(C,NR),DYUP(C,NR),GWHXS(C,NR),C=1,3),NR=1,NRCRS)
    WRITE(WT,915)(NR,IDNO(NR),INSTAT(NR),CYCXS(NR),GWHOLD(NR),DYHOLD
   $(NR),(DYDWN(C,NR),DYUP(C,NR),GWHXS(C,NR),C=1,3),NR=1,NRCRS)
    IAUXM=IAUX-1
    NPERSP=NPERS+1
    IF(NRCRS.GT.MXRCRS.OR.NPERS.GT.MXNPER.OR.IAUX.GT.(MXESX2/2-1).OR.
   $IAUX.LT.3.OR.JFRWRD.GT.6.OR.JFRWRD.LT.2.OR.JBKWRD.GT.5.OR.
   $JBKWRD.LT.1.OR.NMESH.GT.15)     CALL OPERR('RDOPTN',6)
    RETURN
901 FORMAT(L3,I7,I5)
902 FORMAT(16I5)
903 FORMAT(6F7.0,F8.0,6I5)
905 FORMAT((I4,2I3,I5,F7.4,3(2F6.4,I6)))
906 FORMAT(20F4.0)
907 FORMAT(7I5,6L1)
911 FORMAT('1',10X,'SYSOPT INPUT READ BY RDOPTN :'/
   $'0       NPM IDSTRG    NRCRS'/9X,L1,I7,I6)
912 FORMAT('0       SIOT       NPIN       NPOT       PARCAL     PARCON     PA
   $RCCP     CCRDTL     CPRCOR'/7I10,6X,6L1)
913 FORMAT('0      PVRATE          YBASE        YSTART     PCCNVG',
   $                              '  TH$CON  PCDELA  REJLVL          NPERS
   $    GESFRS     MXITER       IAUX        JFRWRD     JBKWRD'/F13.6,2F11.4,
   $F10.2,F8.3,F7.0,1PE10.1,6I10)
914 FORMAT('ONMESH',9X,'MESH(I),I=1,NMESH'/I5,5X,24I5)
915 FORMAT('0  NR   IDNO   INSTAT   CYCXS  GWHOLD  DYHOLD        DYDWN1
   $DYUP1   GWHXS1',6X,'DYDWN2  DYUP2   GWHXS2',6X,'DYDWN3  DYUP3   GW
   $HXS3'/(I5,2I6,I8,I9,F9.4,3(F13.4,F8.4,I7)))
916 FORMAT('0     INITIAL GUESS OF REACTOR-CYCLE MARGINAL COSTS :'/
   $'  NR  RC:   1',(17I7)/(4X,18I7))
```

```
 917 FORMAT(I4,3X,-3P18F7.3/(5X,-3P18F7.3))
 918 FORMAT('0    INITIAL GUESS OF REACTOR-CYCLE ENERGIES, EC''S :'/
   $'  NR  RC:  1',(17I7)/(4X,18I7))
 919 FORMAT(I4,3X,18F7.0/(5X,18F7.0))
     END
     SUBROUTINE RDSTRG
C    READS STRATEGY INFO. OUTPUT BY SYSINT
C    SYSOPT VERSION 12-16-72
     IMPLICIT INTEGER(C,G)
     REAL*8 RDFACT,SGTITL
     COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
    $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
    $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
    $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
    $,GESFRS,ECUPLM(18),CORDTL,CPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
     INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
     LOGICAL NPM,CPRCOR
     LOGICAL MIDCYC
     INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX
     COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),
    $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),IDNO(15),GWHOLD(15),MWD(15)
    $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)
    $,DYHOLD(15),TOY(18,15)
     COMMON/PDPERM/S(100,15),ALPHA(100,15),BETAP(100,15),FINVAR(100)
     INTEGER*2 S
     DIMENSION IDNUM(15),NAME(15),INDEX(15)
     LOGICAL*1 AL(26)/'A','B','C','D','E','F','G','H','I','J','K','L',
    $'M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z'/,NPM1
     REAL*8 DASHES/'--------'/
   5 READ(SIOT,901,END=9)  SGTITL
     WRITE(WT,902) SGTITL
     IF(SGTITL(1).NE.DASHES) GO TO 5
     READ(SIOT,903,END=9) NPM1,IDSTG1,SGTITL
     WRITE(WT,904)        NPM1,IDSTG1,SGTITL
     IF((NPM.AND..NOT.NPM1).OR.(.NOT.NPM.AND.NPM1).OR.
    $(IDSTRG.NE.IDSTG1)) CALL OPERR('RDSTRG',3)
```

```
      READ(SIOT,905,END=9) NRCRS,(IDNUM(I),NAME(I),MWMIN(I),             SOPT0361
     $MWMAX(I),INDEX(I),I=1,NRCRS)                                       SOPT0362
      WRITE(WT,906) NRCRS,(I,AL(I),IDNUM(I),NAME(I),MWMIN(I),            SOPT0363
     $MWMAX(I),INDEX(I),I=1,NRCRS)                                       SOPT0364
      READ(SIOT,907,END=9) NPERIN                                        SOPT0365
      WRITE(WT,970) (I,I=1,9)                                            SOPT0366
      CALL ERASE(CYCNUM,MXRCYC*NRCRS/2,CYCRNG,MXRCYC*NRCRS)              SOPT0367
      CTOT=0                                                             SOPT0368
      MXRCMX=0                                                           SOPT0369
      DO 30 NR=1,NRCRS                                                   SOPT0370
      IF(IDNO(NR).NE.IDNUM(NR)) CALL OPERR('RDSTRG',3)                   SOPT0371
      MWD(NR)=MWMAX(NR)-MWMIN(NR)                                        SOPT0372
      READ(SIOT,908,END=9) (S(I,NR),I=1,NPERIN)                          SOPT0373
      WRITE(WT,909) NR,IDNO(NR),(S(I,NR),I=1,NPERIN)                     SOPT0374
      MIDCYC(NR)=.FALSE.                                                 SOPT0375
      IF(INSTAT(NR)+S(1,NR).EQ.4) MIDCYC(NR)=.TRUE.                      SOPT0376
      J=1                                                                SOPT0377
      K=0                                                                SOPT0378
      INIFLG=3                                                           SOPT0379
      CTOT=CTOT+1                                                        SOPT0380
      CYCNUM(1,NR)=CTOT                                                  SOPT0381
      CYCRNG(1,CTOT)=1                                                   SOPT0382
      DO 20 I=1,NPERS                                                    SOPT0383
      IF(K.EQ.2.OR.S(I,NR).NE.2) GO TO 10                                SOPT0384
      INIFLG=INIFLG-1                                                    SOPT0385
      IF(INIFLG.EQ.2) GO TO 10                                           SOPT0386
      J=J+1                                                              SOPT0387
      CTOT=CTOT+1                                                        SOPT0388
      CYCNUM(J,NR)=CTOT                                                  SOPT0389
      CYCRNG(1,CTOT)=I                                                   SOPT0390
   10 CYCRNG(2,CTOT)=I                                                   SOPT0391
   20 K=S(I,NR)                                                          SOPT0392
      MXRCMX=MAX0(MXRCMX,J)                                              SOPT0393
   30 CYCRMX(NR)=J                                                       SOPT0394
      NCYCT=CTOT                                                         SOPT0395
      WRITE(WT,910) (I,I=1,NRCRS)                                        SOPT0396
```

```
      DO 40 IC=1,MXRCMX
   40 WRITE(WT,911) IC,(CYCNUM(IC,IR),IR=1,NRCRS)
      WRITE(WT,912)  (CYCRMX(IR),IR=1,NRCRS)
      WRITE(WT,913)  (IC,CYCRNG(1,IC),CYCRNG(2,IC),IC=1,NCYCT)
      GO TO 50
    9 CALL OPERR('RDSTRG',12)
   50 RETURN
  901 FORMAT(10A8)
  902 FORMAT('1RDSTRG READ : ',1H',10A8,1H')
  903 FORMAT(L3,I7,10A7)
  904 FORMAT('0',10X,'NPM+IDSTRG =',L2,I7,5X,
     $'STRATEGY TITLE : ',1H',10A7,1H')
  905 FORMAT(I5/(I5,1X,A4,2I5,I10))
  906 FORMAT('0DATA FOR THE ',I3,' REACTORS :'/'    NR    AL IDNO NAME MWM
     $IN MWMAX INDEX IN SYSINT'/(I5,4X,A1,I5,A5,I5,I6,I10))
  907 FORMAT(21X,I4)
  908 FORMAT(80I1)
  909 FORMAT(I5,I6,4X,100I1/(15X,100I1))
  910 FORMAT('0    CYCNUM(RC,NR) :'/'OR.CYCLE',T19,'NR  REACTOR INDEX'/
     $'  INDEX',30I4/(9X,30I4))
  911 FORMAT('0',I4,3X,30I4/(10X,30I4))
  912 FORMAT('0CYCRMX ',30I4/(10X,30I4))
  913 FORMAT('0CYCRNG AS   (CYCNUM,FRSPRD,LSTPRD)  :'/
     $(1X,10('(',I3,2I4,')')))
  970 FORMAT(//,T20,' MAINTENANCE STRATEGY BY PERIOD AND INDEX',
     $'  (0=NON-EXISTENT;1=DOWN;2=ON-LINE)'//T115,'1',T62,'PERIOD'/
     $15X,9I10,9X,'0'/'    NR  IDNO',4X,10('1234567890')/)
      END
      SUBROUTINE RDPERS
C     READS PERIOD INFO OUTPUT BY SYSINT
C     SYSOPT VERSION 12-16-72
C     IDUM'S USED TO MAKE NAMELIST OUTPUT MORE READABLE
      IMPLICIT INTEGER(C,G)
      REAL*8 RDFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
```

```
$MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
$PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
$,GESFRS,ECUPLM(18),CORDTL,OPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
 INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
 LOGICAL NPM,OPRCOR
 COMMON/PDTEMP/NPMFAL(100),NTBSLD(100),OPHRS(100,15),LVLMN(100),
$LVLMX(100),PDELIM(20,100),PDTITL(20,100),DMW(100),DTH(100),ECS(100
$),R4(13,100),R8(12,100),YMID(100),YEND(100),PVFACT(100),AVL(100,
$15),EXPGWH(100,15),CAVG(100),BASVAR(100),FINTST(100),MAXVAR(100),
$MAXTST(100),MIN(100,15),MAX(100,15),BASCFA(100,15)
 REAL MAXVAR,MAXTST,CAVG
 REAL*8 PVFACT,R8
 COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
$,LVLMIN,LVLMAX
 REAL*8 DM,DT,GWHPER,DAYS,PEMIN,PEMAX,PROB
 NAMELIST /FNLTOT/MWINST,MWONLN,MWPEAK,MWMRGN,MWSPIN,PLOFL,
$EXPDEM,EXPGEN,XNKGEN,IDUM1  ,XNNGEN,EXPEMR,IDUM2  ,UNSRVD,PROD$,
$IDUM3  ,$NKPRD,$NNPRD,IDUM4  ,SUSD$,$NKSUS,IDUM5  ,$NNSUS,$SBTOT,
$IDUM6  ,$NKTOT,$NNTOT,IDUM7  ,EMRP$,TOTAL$
 REAL*8 PROD$,$NKPRD,$NNPRD,SUSD$,$NKSUS,$NNSUS,$SBTOT,$NKTOT,
$$NNTOT,EMRP$,TOTAL$
 DATA $DASH$,$DOTS$/'----','....'/
 REAL*8 CDFMIN(500),CDFMAX(500)
 DIMENSION X(20),Y(20)
10 READ(SIOT,901,END=9) X
 IF(X(1).EQ.$DASH$) GO TO 100
 IF(X(1).NE.$DOTS$) GO TO 10
 READ(SIOT,902,END=9) Y,NPER,DM,DT,DC
 IF(NPER.GT.NPERS) GC TC 10
 IF(.NOT.NPM) GO TO 20
 READ(SIOT,903,END=9) LVLMN(NPER),N1,L1,(CDFMIN(N1+I),I=1,L1)
 READ(SIOT,904,END=9) NPMFAL(NPER),NTBSLD(NPER),(OPHRS(NPER,I),
$I=1,NRCRS)
 LPTS=L1
 NUMONE=N1
 READ(SIOT,903,END=9) LVLMX(NPER),N1,L1,(CDFMAX(N1+I),I=1,L1)
```

```
      IF(N1.NE.NUMONE.OR.LPTS.LT.L1.OR.LVLMN(NPER).GE.LVLMX(NPER))      SOPT0469
     $     CALL OPERR('RDPERS',1)                                       SOPT0470
   20 READ(SIOT,FNLTOT,END=9)                                           SOPT0471
      READ(SIOT,905,END=9) (AVL(NPER,I),EXPGWH(NPER,I),I=1,NRCRS)       SOPT0472
      DO 30 I=1,20                                                      SOPT0473
      PDELIM(I,NPER)=X(I)                                               SOPT0474
   30 PDTITL(I,NPER)=Y(I)                                               SOPT0475
      DMW(NPER)=DM                                                      SOPT0476
      DTH(NPER)=DT                                                      SOPT0477
      ECS(NPER)=DC                                                      SOPT0478
      GWHPER=DM*DT*1.D-3                                                SOPT0479
      IEMIN=NUMONE                                                      SOPT0480
      IEMAX=NUMONE+L1                                                   SOPT0481
      PEMIN=NUMONE*DM                                                   SOPT0482
      PEMAX=IEMAX*DM+1.D-3                                              SOPT0483
      LVLMIN=LVLMN(NPER)                                                SOPT0484
      LVLMAX=LVLMX(NPER)                                                SOPT0485
      DO 40 I=1,NUMONE                                                  SOPT0486
      CDFMIN(I)=1.0D0                                                   SOPT0487
   40 CDFMAX(I)=1.0D0                                                   SOPT0488
      R4( 1,NPER)=NPER                                                  SOPT0489
      R4( 2,NPER)=MWINST                                                SOPT0490
      R4( 3,NPER)=MWONLN                                                SOPT0491
      R4( 4,NPER)=MWPEAK                                                SOPT0492
      R4( 5,NPER)=MWMRGN                                                SOPT0493
      R4( 6,NPER)=MWSPIN                                                SOPT0494
      R4( 7,NPER)=PLOFL                                                 SOPT0495
      R4( 8,NPER)=EXPDEM                                                SOPT0496
      R4( 9,NPER)=EXPGEN                                                SOPT0497
      R4(10,NPER)=XNKGEN                                                SOPT0498
      R4(11,NPER)=XNNGEN                                                SOPT0499
      R4(12,NPER)=EXPEMR                                                SOPT0500
      R4(13,NPER)=UNSRVD                                                SOPT0501
      R8( 1,NPER)=NPER                                                  SOPT0502
      R8( 2,NPER)=PROD$                                                 SOPT0503
      R8( 3,NPER)=$NKPRD                                                SOPT0504
```

```
      R8( 4,NPER)=$NNPRD                                                SOPT0505
      R8( 5,NPER)=SUSD$                                                 SOPT0506
      R8( 6,NPER)=$NKSUS                                                SOPT0507
      R8( 7,NPER)=$NNSUS                                                SOPT0508
      R8( 8,NPER)=$SBTOT                                                SOPT0509
      R8( 9,NPER)=$NKTOT                                                SOPT0510
      R8(10,NPER)=$NNTOT                                                SOPT0511
      R8(11,NPER)=EMRP$                                                 SOPT0512
      R8(12,NPER)=TOTAL$                                                SOPT0513
      CALL PDCALC(NPER,CDFMIN,CDFMAX)                                   SOPT0514
      GO TO 10                                                          SOPT0515
  100 READ(SIOT,906,END=50)                                             SOPT0516
    9 CALL OPERR('RDPERS',12)                                           SOPT0517
   50 RETURN                                                            SOPT0518
  901 FORMAT(20A4)                                                      SOPT0519
  902 FORMAT(20A4/I10,3F10.4)                                           SOPT0520
  903 FORMAT(3X,I7,2I5,6F10.9/(8F10.9))                                 SOPT0521
  904 FORMAT(2I5,(7F10.4))                                              SOPT0522
  905 FORMAT(/(8X,F8.4,18X,F16.5))                                      SOPT0523
  906 FORMAT(///)                                                       SOPT0524
      END                                                               SOPT0525
      SUBROUTINE PDCALC(NP,CDFMIN,CDFMAX)                               SOPT0526
C     PERFORMS VARIOUS PRE-CALCS. FOR EACH PERIOD                       SOPT0527
C     SYSOPT VERSION 12-16-72                                           SOPT0528
      IMPLICIT INTEGER(C,G)                                             SOPT0529
      REAL*8 RDFACT,SGTITL                                              SOPT0530
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART, SOPT0531
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,    SOPT0532
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,   SOPT0533
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER      SOPT0534
     $,GESFRS,ECUPLM(18),CORDTL,CPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK   SOPT0535
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP                           SOPT0536
      LOGICAL NPM,OPRCOR                                                SOPT0537
      LOGICAL MIDCYC                                                    SOPT0538
      INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX                              SOPT0539
      COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),       SOPT0540
```

```
     $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),ICNO(15),GWHOLD(15),MWD(15)
     $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)
     $,DYHOLD(15),TOY(18,15)
      COMMON/PDPERM/S(100,15),ALPHA(100,15),BETAP(100,15),FINVAR(100)
      INTEGER*2 S
      COMMON/PCTEMP/NPMFAL(100),NTBSLD(100),OPHRS(100,15),LVLMN(100),
     $LVLMX(100),PDELIM(20,100),PDTITL(20,100),DMW(100),DTH(100),ECS(100
     $),R4(13,100),R8(12,100),YMID(100),YEND(100),PVFACT(100),AVL(100,
     $15),EXPGWH(100,15),CAVG(100),BASVAR(100),FINTST(100),MAXVAR(100),
     $MAXTST(100),MIN(100,15),MAX(100,15),BASCFA(100,15)
      REAL MAXVAR,MAXTST,CAVG
      REAL*8 PVFACT,R8
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
     $,LVLMIN,LVLMAX
      REAL*8 DM,DT,GWHPER,DAYS,PEMIN,PEMAX,PROB
      REAL*8 CDFLPR(1),CDFMIN(1),CDFMAX(1)
      REAL*8 CAVE,CMIN,CMAX,GWHBAS,CI,GWH,GWHNRG,F,TEMP
      EQUIVALENCE(CDFLPR(1),PROB(1))
      GWH(MWLO,MWHI)=GWHNRG(DFLOAT(MWLO),DFLOAT(MWHI))
C     MIN & MAX REACTOR CONTRIBUTIONS TO NUCLEAR POTENTIAL
      NRON=0
      MWDMIN=100000
      MWDTOT=0
      IMX=0
      IMN=0
      SPMX=0.0
      SPMN=0.0
      DO 30 NR=1,NRCRS
      IF(S(NP,NR).NE.2) GO TO 20
      NRON=NRON+1
      MWDTOT=MWDTOT+MWD(NR)
      MX=MWMAX(NR)
      MN=MWMIN(NR)
      P=AVL(NP,NR)*0.01
      IMX=IMX+MX
      IMN=IMN+MN
```

```
      SPMX=SPMX+P*MX
      SPMN=SPMN+P*MN
      MWDMIN=MINO(MWDMIN,MWD(NR))
      GWHBAS=OPHRS(NP,NR)*MN*0.001
      BASCFA(NP,NR)=100.*OPHRS(NP,NR)/(DT*P)
      ALPHA(NP,NR)=1000./(MWD(NR)*P*DT)
      BETAP(NP,NR)=GWHBAS*ALPHA(NP,NR)
      CALL SUBPLT(MN,P,CDFMIN)
      MAX(NP,NR)=GWHBAS+P*GWH(LVLMIN,LVLMIN+MWD(NR))+0.5
      CALL SUBPLT(MX,P,CDFMAX)
      MIN(NP,NR)=GWHBAS+P*GWH(LVLMAX-MWD(NR),LVLMAX)+0.5
      GO TO 30
   20 MIN(NP,NR)=0
      MAX(NP,NR)=0
      BASCFA(NP,NR)=-100.
   30 CONTINUE
      IF(MWDTOT.NE.LVLMAX-LVLMIN) CALL OPERR('PDCALC',2)
C     CALCULATE CDFLPR AND CAVG
      IF(MWDTOT.LE.0) GO TO 36
      P$MX=SPMX/IMX
      P$MN=SPMN/IMN
      MXBAR=FLOAT(IMX)/NRON+0.5
      MNBAR=FLOAT(IMN)/NRON+0.5
      CALL SUBPLT(MNBAR,P$MN,CDFMIN)
      DO 32 I=1,IEMAX
   32 CDFMIN(I)=PROB(I)
      CALL SUBPLT(MXBAR,P$MX,CDFMAX)
      DO 34 I=1,IEMAX
   34 CDFMAX(I)=PROB(I)
   36 ILO=(LVLMIN-.01)/DM
      IF(ILO.LE.IEMIN) ILO=IEMIN+1
      IHI=(LVLMAX+.01)/DM+1
      TEMP=1./MWDTOT
      DO 38 I=ILO,IHI
      F=(I*DM-LVLMIN)*TEMP
   38 CDFLPR(I)=CDFMIN(I)+F*(CDFMAX(I)-CDFMIN(I))
```

```
      CAVE=GWH(LVLMIN,LVLMAX)/(MWDTOT*DT*0.001)                         SOPT0613
      CAVG(NP)=CAVE                                                     SOPT0614
      CMAX=PROBX(DFLOAT(LVLMIN))                                        SOPT0615
      CMIN=PROBX(DFLOAT(LVLMAX))                                        SOPT0616
      IF(NRON.LE.0.OR.CMIN.GE.1.D0) GO TO 60                            SOPT0617
C     BASVAR                                                            SOPT0618
      VAR=0.0                                                           SOPT0619
      LVL=LVLMIN                                                        SOPT0620
      KBLKS=(MWDTOT-1)/MWDMIN+1                                         SOPT0621
      TEMP=1000./(MWDMIN*DT)                                            SOPT0622
      DO 40 K=1,KBLKS                                                   SOPT0623
      CI=GWH(LVL,LVL+MWDMIN)*TEMP                                       SOPT0624
      LVL=LVL+MWDMIN                                                    SOPT0625
   40 VAR=VAR+(CI-CAVE)**2                                              SOPT0626
      BASVAR(NP)=VAR/KBLKS                                              SOPT0627
C     MAXVAR                                                            SOPT0628
      F=(CAVE-CMIN)/(CMAX-CMIN)                                         SOPT0629
      MAXVAR(NP)=F*(CMAX-CAVE)**2+(1.-F)*(CAVE-CMIN)**2                 SOPT0630
      MAXTST(NP)=MAXVAR(NP)/BASVAR(NP)                                  SOPT0631
C     FINVAR                                                            SOPT0632
      DO 50 I=IEMIN,IEMAX                                               SOPT0633
   50 CDFLPR(I)=CDFLPR(I)**2                                            SOPT0634
      FINVAR(NP)=GWH(LVLMIN,LVLMAX)/(MWDTOT*DT*0.001)-CAVE**2           SOPT0635
      FINTST(NP)=FINVAR(NP)/BASVAR(NP)                                  SOPT0636
      GO TO 70                                                          SOPT0637
   60 MAXVAR(NP)=0.0                                                    SOPT0638
      MAXTST(NP)=0.0                                                    SOPT0639
      FINVAR(NP)=0.0                                                    SOPT0640
      FINTST(NP)=0.0                                                    SOPT0641
      BASVAR(NP)=1.E15                                                  SOPT0642
   70 RETURN                                                            SOPT0643
      END                                                               SOPT0644
      SUBROUTINE SUBPLT(MW,P,CDF)                                       SOPT0645
C     SUBTRACTS PLANT OF MW MEGAWATTS AND P FRACTIONAL AVAILABILITY     SOPT0646
C     FROM PROB, THE EQUIVALENT LOAD CDF                                SOPT0647
C     SYSOPT VERSION 03-06-72                                           SOPT0648
```

```
C      NOTE:    MW  MUST BE LESS THAN PEMIN                                   SOPT0649
       IMPLICIT REAL*8 (A-H,O-$)                                               SOPT0650
       COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)         SOPT0651
      $,LVLMIN,LVLMAX                                                          SOPT0652
       REAL*8 ZERO/0.0D0/,ONE/1.0D0/,TWO/2.0D0/,HALF/0.5D0/,TEN/1.D1/,         SOPT0653
      $TENTH/1.D-1/,HUNDRD/1.D2/,CENTI/1.D-2/,THOUS/1.D3/,MILLI/1.D-3/         SOPT0654
       DIMENSION CDF(1)                                                        SOPT0655
       DATA EPS,TRACE/1.D-3,1.D-10/                                            SOPT0656
       DO 10 J=1,IEMAX                                                         SOPT0657
    10 PROB(J)=CDF(J)                                                          SOPT0658
       IF(MW.LE.0) RETURN                                                      SOPT0659
       IF(MW.GE.PEMIN) CALL CPERR('SUBPLT',2)                                  SOPT0660
       ILOW=IEMIN+1                                                            SOPT0661
       FB=MW/DM                                                                SOPT0662
       INT=FB                                                                  SOPT0663
       FB=FB-INT                                                               SOPT0664
       OVP=ONE/P                                                               SOPT0665
       Q=ONE-P                                                                 SOPT0666
       QFB=Q*FB                                                                SOPT0667
       GAMMA=ONE/(ONE-QFB)                                                     SOPT0668
       IF(INT.GT.0) GO TO 60                                                   SOPT0669
C      LOOP TO UNCONVOLVE PLANT IF MW.LT.DM                                    SOPT0670
       DO 20 J=ILOW,IEMAX                                                      SOPT0671
    20 PROB(J)=GAMMA*(PROB(J)-QFB*PROB(J-1))                                   SOPT0672
C      FIND NEW PEMAX AND IEMAX                                                SOPT0673
    30 J=IEMAX                                                                 SOPT0674
    40 IF(PROB(J).GT.TRACE) GO TO 50                                           SOPT0675
       PROB(J)=ZERO                                                            SOPT0676
       J=J-1                                                                   SOPT0677
       GO TO 40                                                                SOPT0678
    50 IF(IEMAX.EQ.J) RETURN                                                   SOPT0679
       IEMAX=J+1                                                               SOPT0680
       PEMAX=IEMAX*DM+EPS                                                      SOPT0681
       RETURN                                                                  SOPT0682
C      LOOP TO UNCONVOLVE PLANT IF MW.GE.DM                                    SOPT0683
    60 DO 70 J=ILOW,IEMAX                                                      SOPT0684
```

```
      JINT=J-INT
   70 PROB(J)=CVP*(PROB(J)-Q*(PROB(JINT)+FB*(PROB(JINT-1)-PROB(JINT))))
      GO TO 30
      END
      FUNCTION GWHNRG(XLOWER,XUPPER)
C     CALCULATES GWH OF ENERGY UNDER PORTION OF PROB, THE CDF OF
C     EQUIVALENT LOAD, BY INTEGRATING FROM XLOWER TO XUPPER ASSUMING
C     LINEAR INTERPOLATION BETWEEN ARRAY POINTS
C     SYSOPT VERSION 03-06-72
      IMPLICIT REAL*8 (A-H,O-$)
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
     $,LVLMIN,LVLMAX
      REAL*8 ZERO/0.0D0/,ONE/1.0D0/,TWO/2.0D0/,HALF/0.5D0/,TEN/1.D1/,
     $TENTH/1.D-1/,HUNDRD/1.D2/,CENTI/1.D-2/,THOUS/1.D3/,MILLI/1.D-3/
      XLO=XLOWER
      XUP=XUPPER
      GWHNRG=ZERO
      SUM=ZERO
      IF(XLO.GE.XUP) RETURN
      IBELO=XLO/DM
      ILAST=XUP/DM
      IF(IBELO.LE.0.OR.ILAST.GE.IEMAX) GO TO 50
C     STANDARD CASE WITH BOTH POINTS WITHIN NON-ZERO ARRAY POINTS
    5 IFRST=IBELO+1
      IABOV=ILAST+1
      IFRSTP=IFRST+1
      ILASTM=ILAST-1
      ICASE=IABOV-IBELO
      RLO=IFRST-XLO/DM
      RUP=XUP/DM-ILAST
      PLO=PROB(IFRST)+(PROB(IBELO)-PROB(IFRST))*RLO
      PUP=PROB(IABOV)+(PROB(ILAST)-PROB(IABOV))*(ONE-RUP)
      GO TO (10,20,30,40),ICASE
   40 DO 35 I=IFRSTP,ILASTM
   35 SUM=SUM+PROB(I)
   30 SUM=SUM+HALF*(PROB(IFRST)+PROB(ILAST))
```

```
   20 SUM=SUM+HALF*(RLC*(PLO+PROB(IFRST))+RUP*(PUP+PROB(ILAST)))              SOPT0721
   15 GWHNRG=SUM*GWHPER                                                        SOPT0722
      RETURN                                                                   SOPT0723
   10 SUM=SUM+(XUP-XLO)*(PLO+PUP)*HALF/DM                                      SOPT0724
      GO TO 15                                                                 SOPT0725
C     SPECIAL CASES INVOLVING ONE OR BOTH END POINTS                           SOPT0726
   50 IF(XUP.LE.ZERO.OR.XLO.GE.PEMAX) RETURN                                   SOPT0727
      IF(XLO.LT.ZERO) XLO=ZERO                                                 SOPT0728
      IF(XUP.GT.PEMAX) XUP=PEMAX                                               SOPT0729
      IBELO=XLO/DM                                                             SOPT0730
      ILAST=XUP/DM                                                             SOPT0731
      JCASE=1                                                                  SOPT0732
      IF(ILAST.GT.0)      JCASE=JCASE+1                                        SOPT0733
      IF(ILAST.EQ.IEMAX) JCASE=JCASE+1                                         SOPT0734
      IF(IBELO.GT.0)      JCASE=JCASE+1                                        SOPT0735
      IF(IBELO.EQ.IEMAX) JCASE=JCASE+1                                         SOPT0736
      GO TO (101,102,102,104,105),JCASE                                        SOPT0737
  101 GWHNRG=(XUP-XLO)*GWHPER/DM                                               SOPT0738
      RETURN                                                                   SOPT0739
  102 SUM=ONE-XLO/DM                                                           SOPT0740
      XLO=DM                                                                   SOPT0741
      IBELO=1                                                                  SOPT0742
      IF(JCASE.EQ.2) GO TO 5                                                   SOPT0743
  104 XO=IEMAX*DM                                                              SOPT0744
      PUP=PROB(IEMAX)*(ONE-(XUP-XO)/(PEMAX-XO))                                SOPT0745
      SUM=SUM+(XUP-XO)*HALF*(PUP+PROB(IEMAX))/DM                               SOPT0746
      XUP=XO                                                                   SOPT0747
      ILAST=IEMAX-1                                                            SOPT0748
      GO TO 5                                                                  SOPT0749
  105 XO=IEMAX*DM                                                              SOPT0750
      PUP=PROB(IEMAX)*(ONE-(XUP-XO)/(PEMAX-XO))                                SOPT0751
      PLO=PROB(IEMAX)*(ONE-(XLO-XO)/(PEMAX-XO))                                SOPT0752
      GWHNRG=(XUP-XLO)*(PLO+PUP)*HALF*GWHPER/DM                                SOPT0753
      RETURN                                                                   SOPT0754
      END                                                                      SOPT0755
      FUNCTION PROBX(X)                                                        SOPT0756
```

```
C     EVALUATES PROB AT A PARTICULAR VALUE OF X MW
C     SYSOPT VERSION 03-06-72
      IMPLICIT REAL*8 (A-H,O-$)
      COMMON/PROB/DM,DT,GWHPER,DAYS,IEMIN,IEMAX,PEMIN,PEMAX,PROB(500)
     $,LVLMIN,LVLMAX
      DATA ZERO,ONE/0.0D0,1.0D0/
      PROBX=ONE
      IF(X.LE.PEMIN) RETURN
      PROBX=ZERO
      IF(X.GE.PEMAX) RETURN
      FB=X/DM
      ILO=FB
      FB=FB-ILO
      IF(ILO.GE.IEMAX) GO TO 10
      PROBX=PROB(ILO)+FB*(PROB(ILO+1)-PROB(ILO))
      RETURN
   10 PROBX=PROB(IEMAX)*(PEMAX-X)/(PEMAX-IEMAX*DM)
      RETURN
      END
      SUBROUTINE ASMTYS
C     ASSEMBLES TSY'S AND TEY'S
C     SYSOPT VERSION 12-16-72
      IMPLICIT INTEGER(C,G)
      REAL*8 RDFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPOT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
     $,GESFRS,ECUPLM(18),CORDTL,OPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
      LOGICAL NPM,OPRCOR
      LOGICAL MIDCYC
      INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX
      COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),
     $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),IENO(15),GWHOLD(15),MWD(15)
     $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)
```

```
     $,DYHCLD(15),TOY(18,15)
      COMMON/PDPERM/S(100,15),ALPHA(100,15),BETAP(100,15),FINVAR(100)
      INTEGER*2 S
      COMMON/PDTEMP/NPMFAL(100),NTBSLD(100),OPHRS(100,15),LVLMN(100),
     $LVLMX(100),PDELIM(20,100),PDTITL(20,100),DMW(100),DTH(100),ECS(100
     $),R4(13,100),R8(12,100),YMID(100),YEND(100),PVFACT(100),AVL(100,
     $15),EXPGWH(100,15),CAVG(100),BASVAR(100),FINTST(100),MAXVAR(100),
     $MAXTST(100),MIN(100,15),MAX(100,15),BASCFA(100,15)
      REAL MAXVAR,MAXTST,CAVG
      REAL*8 PVFACT,R8
      LOGICAL WASUP,DWNOW
      INTEGER RC
      PVP(Y)=PVPER$(Y,YBASE)
      TEMP=0.5/8760.
      YEND(1)=YSTART+DTH(1)/8760.
      YMID(1)=(YSTART+YEND(1))*0.5
      PVFACT(1)=PVP(YMID(1))
      DO 10 NP=2,NPERS
      X=TEMP*DTH(NP)
      YMID(NP)=YEND(NP-1)+X
      YEND(NP)=YMID(NP)+X
   10 PVFACT(NP)=PVP(YMID(NP))
      DO 50 NR=1,NRCRS
      IC=1
      TOY(IC,NR)=0.0
      IF(MIDCYC(NR)) IC=0
      CLIM=CYCRMX(NR)
      DO 30 RC=1,CLIM
      CYC=CYCNUM(RC,NR)
      NPF=CYCRNG(1,CYC)
      NPL=CYCRNG(2,CYC)
      IC=IC+1
      TOY(IC,NR)=0.0
      WASUP=.FALSE.
      DO 20 NP=NPF,NPL
      TOY(IC,NR)=TOY(IC,NR)+OPHRS(NP,NR)/8760.
```

```
      DWNCW=S(NP,NR).NE.2                                               SOPT0829
      IF(WASUP.AND.DWNCW) GO TO 30                                      SOPT0830
      IF(WASUP.AND..NOT.DWNCW) GO TO 20                                 SOPT0831
      IF(.NOT.WASUP.AND.DWNCW) GO TO 20                                 SOPT0832
      WASUP=.TRUE.                                                      SOPT0833
      TSY(IC,NR)=YEND(NP-1)                                             SOPT0834
      IF(NP.EQ.1) TSY(IC,NR)=YSTART                                     SOPT0835
   20 TEY(IC,NR)=YEND(NP)                                               SOPT0836
   30 CONTINUE                                                          SOPT0837
      TEY(IC,NR)=TEY(IC,NR)+DYHOLD(NR)                                  SOPT0838
      TOY(IC,NR)=TOY(IC,NR)+DYHOLD(NR)*AVL(NPERS,NR)*0.01               SOPT0839
      IF(MIDCYC(NR)) GO TO 35                                           SOPT0840
      TEY(1,NR)=TSY(2,NR)                                               SOPT0841
      TSY(1,NR)=TSY(2,NR)-1.E-4                                         SOPT0842
   35 NCYCXS=CYCXS(NR)                                                  SOPT0843
      IF(NCYCXS.LT.1) GO TO 50                                          SOPT0844
      DO 40 I=1,NCYCXS                                                  SOPT0845
      IC=IC+1                                                           SOPT0846
      TSY(IC,NR)=TEY(IC-1,NR)+DYDWN(I,NR)                               SOPT0847
      TOY(IC,NR)=DYUP(I,NR)*AVL(NPERS,NR)*0.01                          SOPT0848
   40 TEY(IC,NR)=TSY(IC,NR)+DYUP(I,NR)                                  SOPT0849
   50 CONTINUE                                                          SOPT0850
      RETURN                                                            SOPT0851
      END                                                               SOPT0852
      SUBROUTINE WTPERS                                                 SOPT0853
C     WRITES INFO. FOR THE VARIOUS PERIODS                              SOPT0854
C     SYSOPT VERSION 12-16-72                                           SOPT0855
      IMPLICIT INTEGER(C,G)                                             SOPT0856
      REAL*8 RDFACT,SGTITL                                              SOPT0857
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART, SOPT0858
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,    SOPT0859
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,   SOPT0860
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER      SOPT0861
     $,GESFRS,ECUPLM(18),CORDTL,CPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK   SOPT0862
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP                           SOPT0863
      LOGICAL NPM,OPRCOR                                                SOPT0864
```

```
      LOGICAL MIDCYC                                                    SOPT0865
      INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX                              SOPT0866
      COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),       SOPT0867
     $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),IDNO(15),GWHOLD(15),MWD(15)SOPT0868
     $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)  SOPT0869
     $,DYHOLD(15),TOY(18,15)                                            SOPT0870
      COMMON/PDPERM/S(100,15),ALPHA(100,15),BETAP(100,15),FINVAR(100)   SOPT0871
      INTEGER*2 S                                                       SOPT0872
      COMMON/PDTEMP/NPMFAL(100),NTBSLD(100),OPHRS(100,15),LVLMN(100),   SOPT0873
     $LVLMX(100),PDELIM(20,100),PDTITL(20,100),DMW(100),DTH(100),ECS(100SOPT0874
     $),R4(13,100),R8(12,100),YMID(100),YEND(100),PVFACT(100),AVL(100,  SOPT0875
     $15),EXPGWH(100,15),CAVG(100),BASVAR(100),FINTST(100),MAXVAR(100), SOPT0876
     $MAXTST(100),MIN(100,15),MAX(100,15),BASCFA(100,15)                SOPT0877
      REAL MAXVAR,MAXTST,CAVG                                           SOPT0878
      COMMON/FINALS/S4,SA4,SP4,SL4,SP8                                  SOPT0879
      REAL*8 S4(13),SA4(13),SP4(13),SL4(13),SP8(13)                     SOPT0880
      REAL*8 S8(13),SA8(13),SL8(13),SPV,PV,PVFACT,R8                    SOPT0881
      LOGICAL*1 AL(26)/'A','B','C','D','E','F','G','H','I','J','K','L', SOPT0882
     $'M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z'/          SOPT0883
      CALL ERASE(S4,26,SA4,26,SP4,26,SL4,26,S8,26,SA8,26,SP8,26,SL8,26) SOPT0884
      SPV=0.0D0                                                         SOPT0885
      DO 20 I=1,NPERS                                                   SOPT0886
      IF(R4(1,I).NE.I) CALL OPERR('WTPERS',12)                          SOPT0887
      PV=PVFACT(I)                                                      SOPT0888
      SPV=SPV+PV                                                        SOPT0889
      DO 10 J=2,12                                                      SOPT0890
      S4(J)=S4(J)+R4(J,I)                                               SOPT0891
      S8(J)=S8(J)+R8(J,I)                                               SOPT0892
      SP4(J)=SP4(J)+PV*R4(J,I)                                          SOPT0893
   10 SP8(J)=SP8(J)+PV*R8(J,I)                                          SOPT0894
      S4(13)=S4(13)+R4(13,I)                                            SOPT0895
   20 SP4(13)=SP4(13)+PV*R4(13,I)                                       SOPT0896
      DO 30 J=2,12                                                      SOPT0897
      SA4(J)=S4(J)/NPERS                                                SOPT0898
      SL4(J)=SP4(J)/SPV                                                 SOPT0899
      SA8(J)=S8(J)/NPERS                                                SOPT0900
```

```
30  SL8(J)=SP8(J)/SPV                                                  SOPT0901
    SA4(13)=S4(13)/NPERS                                               SOPT0902
    SL4(13)=SP4(13)/SPV                                                SOPT0903
    WRITE(WT,901) (I,(PDELIM(J,I),J=1,20),I=1,NPERS)                   SOPT0904
    WRITE(WT,902) (I,(PDTITL(J,I),J=1,20),I=1,NPERS)                   SOPT0905
    WRITE(WT,903) (I,DMW(I),ECS(I),DTH(I),YMID(I),YEND(I),PVFACT(I),   SOPT0906
   $NPMFAL(I),NTBSLD(I),LVLMN(I),LVLMX(I),I=1,NPERS)                   SOPT0907
    WRITE(WT,915) SPV,(I,I=1,MXRCYC)                                   SOPT0908
    DO 35 NR=1,NRCRS                                                   SOPT0909
    CLIM=CYCRMX(NR)+CYCXS(NR)                                          SOPT0910
    IF(.NOT.MIDCYC(NR)) CLIM=CLIM+1                                    SOPT0911
    IF(CLIM.GT.MXRCYC) CALL OPERR('WTPERS',6)                          SOPT0912
    WRITE(WT,916) NR,(TSY(I,NR),I=1,CLIM)                              SOPT0913
    WRITE(WT,919) (TOY(I,NR),I=1,CLIM)                                 SOPT0914
35  WRITE(WT,917) (TEY(I,NR),I=1,CLIM)                                 SOPT0915
    WRITE(WT,911)                                                      SOPT0916
    WRITE(WT,904) (I,I=1,NRCRS)                                        SOPT0917
    WRITE(WT,905) (AL(I),I=1,NRCRS)                                    SOPT0918
    DO 40 I=1,NPERS                                                    SOPT0919
40  WRITE(WT,906) I,(AVL    (I,NR),NR=1,NRCRS)                         SOPT0920
    WRITE(WT,912)                                                      SOPT0921
    WRITE(WT,904) (I,I=1,NRCRS)                                        SOPT0922
    WRITE(WT,905) (AL(I),I=1,NRCRS)                                    SOPT0923
    DO 50 I=1,NPERS                                                    SOPT0924
50  WRITE(WT,906) I,(OPHRS (I,NR),NR=1,NRCRS)                          SOPT0925
    WRITE(WT,918)                                                      SOPT0926
    WRITE(WT,904) (I,I=1,NRCRS)                                        SOPT0927
    WRITE(WT,905) (AL(I),I=1,NRCRS)                                    SOPT0928
    DO 55 I=1,NPERS                                                    SOPT0929
55  WRITE(WT,906) I,(BASCFA(I,NR),NR=1,NRCRS)                          SOPT0930
    WRITE(WT,913)                                                      SOPT0931
    WRITE(WT,904) (I,I=1,NRCRS)                                        SOPT0932
    WRITE(WT,905) (AL(I),I=1,NRCRS)                                    SOPT0933
    DO 60 I=1,NPERS                                                    SOPT0934
60  WRITE(WT,906) I,(EXPGWH(I,NR),NR=1,NRCRS)                          SOPT0935
    WRITE(WT,907) ((R4(I,J),I=1,13),J=1,NPERS)                         SOPT0936
```

```
      WRITE(WT,908) (S4(I),I=2,13),(SA4(I),I=2,13),(SP4(I),I=2,13),
     $(SL4(I),I=2,13)
      WRITE(WT,909) ((R8(I,J),I=1,12),J=1,NPERS)
      WRITE(WT,910) (S8(I),I=2,12),(SA8(I),I=2,12),(SP8(I),I=2,12),
     $(SL8(I),I=2,12)
      WRITE(WT,914) (I,BASVAR(I),FINVAR(I),FINTST(I),MAXVAR(I),
     $MAXTST(I),CAVG(I),I=1,NPERS)
      RETURN
  901 FORMAT('1      PERIOD',40X,'DELIMITER CARD'/(I10,10X,1H',20A4,1H'))
  902 FORMAT('1      PERIOD',40X,'PD. TITLE CARD'/(I10,10X,1H',20A4,1H'))
  903 FORMAT('1PERIOD  DMW      ECS     DTH     YMID      YEND      PVFACT
     $NPMFAL  NTBSLD  LVLMN  LVLMX'/(I5,F8.1,F8.3,F7.1,F9.4,F9.4,F10.6,
     $I6,I7,I10,I7))
  904 FORMAT(T8,'NR: ',I1,14I8/(10X,15I8))
  905 FORMAT('    NP  AL: ',A1,14(7X,A1)/(10X,15(7X,A1)))
  906 FORMAT(I5,1X,15F8.2)
  907 FORMAT('1          --------- M E G A W A T T S ----------  FRACT.',
     $4X,'------------------ GEGAWATT-HOURS ELECTRIC -----------------'
     $      /' PERIOD   MWINST  MWONLN  MWPEAK  MWMRGN  MWSPIN   PLOFL',
     $4X,'EXPDEM     EXPGEN     XNKGEN     XNNGEN     EXPEMR     UNSRVD'
     $  /(F6.0,2X,5F8.0,F8.4,6F11.2))
  908 FORMAT('OTOTAL :',5F8.0,F8.4,6F11.2/
     $       'OAVG.  :',5F8.C,F8.4,6F11.2/
     $       'OPVTOTL:',5F8.C,F8.4,6F11.2/
     $       'OLVAVG.:',5F8.C,F8.4,6F11.2/)
  909 FORMAT('1',T30,'ALL COSTS IN THOUSANDS OF DOLLARS AT MIDDLE OF PER
     $IOD'/' PERIOD      PRCD$      $NKPRD      $NNPRD      SUSD$      $NKS
     $LS      $NNSUS      $SBTCT      $NKTOT      $NNTCT      EMRP$      TOTA
     $L$'     /(0PF6.0,2X,-3P11F11.2))
  910 FORMAT('OTOTAL :',-3P11F11.2/'OAVG.  :',-3P11F11.2/
     $       'OPVTOTL:',-3P11F11.2/'OLVAVG.:',-3P11F11.2/)
  911 FORMAT('1',T20,'AVAILABILITY (PER CENT)'/)
  912 FORMAT('1',T20,'OPERATING HOURS'/)
  913 FORMAT('1',T20,'EXP. PRODUCTION (GWHE) FROM SYSINT'/)
  914 FORMAT('1',T20,'SHAPE VARIANCES AND TESTS'/
     $'0 PERIOD     BASVAR',10X,'FINVAR',9X,'FINTST',9X,'MAXVAR',9X,
```

```
     $'MAXTST',8X,'CAVG'/(I6,4X,E12.6,E15.6,F14.6,E19.6,F11.6,F13.6))
  915 FORMAT('0',T24,'SUMMATION OF PVFACT =',F12.6/'1',T10,'STARTING AND
     $ ENDING TIMES OF REACTOR CYCLES AS PASSED TO IN-CORE MODEL :'/
     $'0  RC:',I5,17I7/(I8,17I7))
  916 FORMAT('0',I3,' = REACTOR INDEX, NR'/
     $       '  TSY :',18F7.3/(F8.3,17F7.3))
  917 FORMAT('  TEY :',18F7.3/(F8.3,17F7.3))
  918 FORMAT('1',T20,'PER CENT CAPACITY FACTOR FOR BASE PORTION (AVAILAB
     $ILITY-BASED)'/)
  919 FORMAT('  TOY :',18F7.3/(F8.3,17F7.3))
      END
      SUBROUTINE SETUPN
C     SETS UP COSTS AND LIMITS OF REMAINING ARCS IN THE NETWORK
C     SYSOPT VERSION 12-16-72
      IMPLICIT INTEGER(C,G)
      REAL*8 RDFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
     $,GESFRS,ECUPLM(18),CCRDTL,CPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
      LOGICAL NPM,OPRCOR
      LOGICAL MIDCYC
      INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX
      COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),
     $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),IDNO(15),GWHOLD(15),MWD(15)
     $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)
     $,DYHOLD(15),TOY(18,15)
      COMMON/PDPERM/S(100,15),ALPHA(100,15),BETAP(100,15),FINVAR(100)
      INTEGER*2 S
      COMMON/KC/KC(1)/KU/KU(1)/KL/KL(1)
      COMMON/PDTEMP/NPMFAL(100),NTBSLD(100),OPHRS(100,15),LVLMN(100),
     $LVLMX(100),PDELIM(20,100),PDTITL(20,100),DMW(100),CTH(100),ECS(100
     $),R4(13,100),R8(12,100),YMID(100),YEND(100),PVFACT(100),AVL(100,
     $15),EXPGWH(100,15),CAVG(100),BASVAR(100),FINTST(100),MAXVAR(100),
```

```
     $MAXTST(100),MIN(100,15),MAX(100,15),BASCFA(100,15)                SOPT1009
      REAL MAXVAR,MAXTST,CAVG                                           SOPT1010
      REAL*8 PVFACT,R8                                                  SOPT1011
      INTEGER RC                                                        SOPT1012
      DATA LARGE/2000000000/                                            SOPT1013
      REAL*8 SUMD                                                       SOPT1014
      LOGICAL GESEQ1,GESEQ2,GESGT2                                      SOPT1015
      CALL ERASE(KC,MXARCS,KU,MXARCS,KL,MXARCS)                         SOPT1016
      SUMD=0.0D0                                                        SOPT1017
      KSUM=0                                                            SOPT1018
      L=LOC(6,0,0,1)-1                                                  SOPT1019
      DO 6 NP=1,NPERS                                                   SOPT1020
      LSUM=0                                                            SOPT1021
      DO 4 NR=1,NRCRS                                                   SOPT1022
    4 LSUM=LSUM+MAX(NP,NR)                                              SOPT1023
C     TYPE 6                                                            SOPT1024
      N=L+NP                                                            SOPT1025
      KU(N)=R4(10,NP)+0.5                                               SOPT1026
      IF(KU(N).GT.LSUM) KU(N)=LSUM                                      SOPT1027
      KL(N)=KU(N)                                                       SOPT1028
      KSUM=KSUM+KU(N)                                                   SOPT1029
    6 SUMD=SUMD+R4(10,NP)                                               SOPT1030
      RDFACT=SUMD/KSUM                                                  SOPT1031
      WRITE(WT,900) RDFACT                                              SOPT1032
      GDEL=10                                                           SOPT1033
      IF(GESFRS.EQ.4) GDEL=0                                            SOPT1034
      GESEQ1=GESFRS.EQ.1                                                SOPT1035
      GESEQ2=GESFRS.EQ.2                                                SOPT1036
      GESGT2=GESFRS.GT.2                                                SOPT1037
      DO 30 NR=1,NRCRS                                                  SOPT1038
      L=LOC(4,NR,0,NPERSP)                                              SOPT1039
C     TYPE 4    HLDOVR                                                  SOPT1040
      KU(L)=GWHOLD(NR)                                                  SOPT1041
      KL(L)=KU(L)                                                       SOPT1042
      L=LOC(4,NR,1,1)-1                                                 SOPT1043
      CLIM=CYCRMX(NR)                                                   SOPT1044
```

```
      DO 30 RC=1,CLIM                                                   SOPT1045
      C=CYCNUM(RC,NR)                                                   SOPT1046
      ILO=CYCRNG(1,C)                                                   SOPT1047
      IHI=CYCRNG(2,C)                                                   SOPT1048
      CMIN=0                                                            SOPT1049
      CMID=0                                                            SOPT1050
      CMAX=0                                                            SOPT1051
      DO 10 J=ILO,IHI                                                   SOPT1052
C     TYPE 4    PERIODS                                                 SOPT1053
      N=L+J                                                             SOPT1054
      KL(N)=MIN(J,NR)                                                   SOPT1055
      KU(N)=MAX(J,NR)                                                   SOPT1056
      CMID=CMID+EXPGWH(J,NR)+0.5                                        SOPT1057
      CMIN=CMIN+KL(N)                                                   SOPT1058
   10 CMAX=CMAX+KU(N)                                                   SOPT1059
      IF(IHI.NE.NPERS) GO TO 20                                         SOPT1060
      CMIN=CMIN+KL(L+NPERSP)                                            SOPT1061
      CMID=CMID+KU(L+NPERSP)                                            SOPT1062
      CMAX=CMAX+KU(L+NPERSP)                                            SOPT1063
C     TYPE 1                                                            SOPT1064
   20 KL(C)=CMIN                                                        SOPT1065
      KU(C)=CMAX                                                        SOPT1066
C     TYPE 2                                                            SOPT1067
      IF(GESGT2) GO TO 26                                               SOPT1068
      IF(GESEQ1) GO TO 25                                               SOPT1069
      IF(GESEQ2) KC(C+NCYCT)=ELAME(NR,RC)                               SOPT1070
      KL(C+NCYCT)=CMIN/10*10                                            SOPT1071
      KU(C+NCYCT)=(9+CMAX)/10*10                                        SOPT1072
      GO TO 30                                                          SOPT1073
   25 KL(C+NCYCT)=CMID-5                                                SOPT1074
      KU(C+NCYCT)=CMID+5                                                SOPT1075
      GO TO 30                                                          SOPT1076
   26 KU(C+NCYCT)=ELAME(NR,RC)+GDEL                                     SOPT1077
      KL(C+NCYCT)=ELAME(NR,RC)-GDEL                                     SOPT1078
      LAUX=LOC(3,NR,RC,0)                                               SOPT1079
      KC(LAUX)=10000                                                    SOPT1080
```

```
      KU(LAUX)=100000
      LAUX=LAUX+1
      KC(LAUX)=-10000
      KL(LAUX)=-100000
   30 CONTINUE
      L=LOC(5,0,0,0)
C     TYPE 5
      KU(L)=LARGE
      KU(L+1)=LARGE
      KU(L+2)=LARGE
      DO 60 NR=1,NRCRS
      DO 60 NP=1,NPERS
      L=LOC(4,NR,0,NP)
      MON=KL(L)
      MOX=KU(L)
      IF(MOX.LE.0) GO TO 60
      MAV=(CAVG(NP)+BETAP(NP,NR))/ALPHA(NP,NR)+0.5
      MDX=(MOX-MAV)/(JFRWRD-1)+1
      MDN=(MAV-MON)/JBKWRD+1
      L=LOC(7,NR,0,NP)-1
C     TYPE 7
      KU(L+1)=MAV
      KL(L+1)=KU(L+1)
      DO 40 J=2,JFRWRD
      KC(L+J)=(J-1)**4
   40 KU(L+J)=MDX
      L=L+JFRWRD
      IF(JBKWRD.LE.0) GO TO 60
      DO 50 J=1,JBKWRD
      KC(L+J)=-J**4
   50 KL(L+J)=-MDN
   60 CONTINUE
      CALL ONLY$$
      RETURN
  900 FORMAT('0',T6,F12.8,' = RDFACT, NUCL.GEN.ROUND-OFF CORRECTION FACT
     $OR')
```

```
      END
      SUBROUTINE SETUPT
C     SETS UP INPUT TAPE FOR O-O-K
C     SYSOPT VERSION 12-16-72
      IMPLICIT INTEGER(C,G)
      REAL*8 RCFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
     $,GESFRS,ECUPLM(18),CORCTL,OPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
      LOGICAL NPM,OPRCOR
      LOGICAL MIDCYC
      INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX
      COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),
     $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),ICNO(15),GWHOLD(15),MWD(15)
     $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)
     $,DYHOLD(15),TOY(18,15)
      COMMON/KC/KC(1)/KU/KU(1)/KL/KL(1)
      LOGICAL*1 AL(26)/'A','B','C','D','E','F','G','H','I','J','K','L',
     $'M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z'/,AR
      INTEGER RC
      REWIND NPIN
      WRITE(NPIN,931)SGTITL
      L=0
C     TYPE 1
      DO 101 NR=1,NRCRS
      CLIM=CYCRMX(NR)
      AR=AL(NR)
      DO 101 RC=1,CLIM
      L=L+1
  101 WRITE(NPIN,901) AR,RC,AR,RC,KC(L),KU(L),KL(L)
C     TYPE 2
      DO 102 NR=1,NRCRS
      CLIM=CYCRMX(NR)
```

```
      AR=AL(NR)
      DO 102 RC=1,CLIM
      L=L+1
  102 WRITE(NPIN,902) AR,RC,KC(L),KU(L),KL(L)
C     TYPE 3
      DO 103 NR=1,NRCRS
      CLIM=CYCRMX(NR)
      AR=AL(NR)
      DO 103 RC=1,CLIM
      DO 103 I=1,IAUXM
      L=L+1
  103 WRITE(NPIN,902) AR,RC,KC(L),KU(L),KL(L)
C     TYPE 4
      DO 114 NR=1,NRCRS
      CLIM=CYCRMX(NR)
      AR=AL(NR)
      DO 104 RC=1,CLIM
      CYC=CYCNUM(RC,NR)
      ILO=CYCRNG(1,CYC)
      IHI=CYCRNG(2,CYC)
      DO 104 NP=ILO,IHI
      L=L+1
  104 WRITE(NPIN,904) AR,RC,AR,NP,KC(L),KU(L),KL(L)
      L=L+1
  114 WRITE(NPIN,914) AR,CLIM,KC(L),KU(L),KL(L)
C     TYPE 5
      L=L+1
      LP2=L+2
      WRITE(NPIN,905) (KC(N),KU(N),KL(N),N=L,LP2)
      L=LP2
C     TYPE 6
      WRITE(NPIN,906) (N,KC(L+N),KU(L+N),KL(L+N),N=1,NPERS)
      L=L+NPERS
C     TYPE 7
      JTOTAL=JFRWRD+JBKWRD
      DO 107 NP=1,NPERS
```

```
      DO 1C7 NR=1,NRCRS
      AR=AL(NR)
      DO 107 J=1,JTOTAL
      L=L+1
  107 WRITE(NPIN,907) AR,NP,NP,KC(L),KU(L),KL(L)
      CUMARC=NCYCT*(IAUX+1)+NPERS*((JFRWRD+JBKWRD+1)*NRCRS+1)+NRCRS+3
      IF(L.NE.CUMARC) CALL OPERR('SETUPT',4)
      WRITE(NPIN,932)
      J=2*MXITER
      WRITE(NPIN,933) (I,SGTITL,I=2,J)
      END FILE NPIN
      REWIND NPIN
      RETURN
  901 FORMAT(6X,'R',A1,'C',I2,'A'  ,'R',A1,'C',I2,'  ',T21,3I10)
  902 FORMAT(6X,'NUKFUL' ,'R',A1,'C',I2,'A',          T21,3I10)
  904 FORMAT(6X,'R',A1,'C',I2,' '  ,'R',A1,'P',I3,    T21,3I10)
  905 FORMAT(6X,'DEMAND' ,'DUMMY ',                   T21,3I10/
     $       6X,'HLDOVR' ,'DUMMY ',                   T21,3I10/
     $       6X,'DUMMY ' ,'NUKFUL',                   T21,3I10)
  906 FORMAT(6X,2X,'P',I3 ,'DEMAND',                  T21,3I10)
  907 FORMAT(6X,'R',A1,'P',I3 ,2X,'P',I3,             T21,3I10)
  908 FORMAT(6X,2X,'P',I3 ,'R',A1,'P',I3,             T21,3I10)
  914 FORMAT(6X,'R',A1,'C',I2,' ' ,'HLDOVR',          T21,3I10)
  931 FORMAT('READY'/'TAPE'/' 1',10A7/'ARCS')
  932 FORMAT('END'/'OUTPUT PRINTER'/'COMPUTE'/'PAUSE')
  933 FORMAT('SAVE'/I2,10A7/'OUTPUT PRINTER'/'COMPUTE'/'PAUSE')
      END
      SUBROUTINE CONVRG(OPTRCH,$LAST)
C     SUPERVISES CONVERGENCE BETWEEN O-O-K AND IN-CORE MODEL
C     SYSOPT VERSION 12-16-72
      IMPLICIT INTEGER(C,G)
      REAL*8 RDFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
     $PARCCP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
```

```
$,GESFRS,ECUPLM(18),CORDTL,OPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK     SOPT1225
 INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP                            SOPT1226
 LOGICAL NPM,OPRCOR                                                 SOPT1227
 LOGICAL MIDCYC                                                     SOPT1228
 INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX                               SOPT1229
 COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),        SOPT1230
$CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),IDNO(15),GWHOLD(15),MWD(15) SOPT1231
$,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)   SOPT1232
$,DYHOLD(15),TOY(18,15)                                             SOPT1233
 COMMON/OOKCOM/KIX,KOX,KQ1X,KQ2X,KQ3X,KQ4X,KQ5X                     SOPT1234
 COMMON/KC/KC(1)/KU/KU(1)/KL/KL(1)                                  SOPT1235
 COMMON/KX/KX(1)                                                    SOPT1236
 INTEGER*2 LSTIM(270)                                               SOPT1237
 REAL*8 $,$LAST,$NUCL(100),RTC,$IMPLS,$IMP,$CRIT                    SOPT1238
 LOGICAL CNVGD,OPTRCH                                               SOPT1239
 INTEGER NECBAL(18)/18*1/                                           SOPT1240
 IF($LAST.GT.0.0D0) GO TO 5                                         SOPT1241
 KIX=NPIN                                                           SOPT1242
 KOX=NPOT                                                           SOPT1243
 KQ1X=7                                                             SOPT1244
 KQ2X=NPOT                                                          SOPT1245
 KQ3X=NPIN                                                          SOPT1246
 KQ4X=MXARCS                                                        SOPT1247
 KQ5X=MXNODS                                                        SOPT1248
 ITERTO=0                                                           SOPT1249
 MESHNO=0                                                           SOPT1250
 GMESH=-1                                                           SOPT1251
 GWHCNV=-1                                                          SOPT1252
 $CRIT=1.E3*TH$CON                                                  SOPT1253
 CALL ERASE(LSTIM,NCYCT/2)                                          SOPT1254
5 $LAST=1.D50                                                       SOPT1255
 $IMPLS=$LAST                                                       SOPT1256
 OPTRCH=.FALSE.                                                     SOPT1257
10 CALL OOKMAN                                                      SOPT1258
 CALL ARCPRT(0)                                                     SOPT1259
 CALL CALSHP                                                        SOPT1260
```

```
      ITERTO=ITERTO+1                                                   SOPT1261
      ITER=MOD(ITERTO-1,100)+1                                          SOPT1262
      $NUCL(ITER)=1.D50                                                 SOPT1263
      CNVGD=.TRUE.                                                      SOPT1264
      DO 20 C=1,NCYCT                                                   SOPT1265
      IF(IABS(KX(C)-LSTIM(C)).GT.GWHCNV) CNVGD=.FALSE.                  SOPT1266
   20 LSTIM(C)=KX(C)                                                    SOPT1267
      NARCTP=PARCAL                                                     SOPT1268
      IF(.NOT.CNVGD.AND.GMESH.GT.0) GO TO 50                            SOPT1269
   25 NARCTP=PARCON                                                     SOPT1270
      IF(MESHNO.LT.NMESH) GO TO 40                                      SOPT1271
      CPTRCH=.TRUE.                                                     SOPT1272
      WRITE(WT,902)                                                     SOPT1273
   30 $NUCL(ITER)=-$NUCL(ITER)                                          SOPT1274
      WRITE(WT,901) (I,$NUCL(I),I=1,ITER)                               SOPT1275
      CALL ARCPRT(PARCON)                                               SOPT1276
      RETURN                                                            SOPT1277
   40 MESHNO=MESHNO+1                                                   SOPT1278
      GMESH=MESH(MESHNO)                                                SOPT1279
      GWHCNV=(PCONVG+0.001)*GMESH*0.01                                  SOPT1280
      $LAST=1.D50                                                       SOPT1281
      $IMPLS=$LAST                                                      SOPT1282
   50 CALL ARCPRT(NARCTP)                                               SOPT1283
      $=0.0D0                                                           SOPT1284
C     ERASE OLD MARGINAL COSTS                                          SOPT1285
      LFRS=LOC(2,1,1,1)                                                 SOPT1286
      NZERO=IAUX*NCYCT                                                  SOPT1287
      CALL ERASE(KC(LFRS),NZERO,KU(LFRS),NZERO,KL(LFRS),NZERO)          SOPT1288
      DO 60 NR=1,NRCRS                                                  SOPT1289
      CALL SETELE(NR,GMESH)                                             SOPT1290
      IDNUM=IDNO(NR)                                                    SOPT1291
      NCYCIN=CYCRMX(NR)                                                 SOPT1292
      IF(.NOT.MIDCYC(NR)) NCYCIN=NCYCIN+1                               SOPT1293
      NCYCXS=CYCXS(NR)                                                  SOPT1294
      ECHDCV=GWHOLD(NR)                                                 SOPT1295
      CALL INCCRE(IDNUM,NCYCIN,NCYCXS,NCYCIN+NCYCXS,TSY(1,NR),TEY(1,NR),SOPT1296
```

```
     $NECBAL,ELAME,MXESX2,ECHDCV,RTC,PVR,YBS,ECUPLM,TOY(1,NR))
      $=$+RTC
   60 CALL NEWMRG(NR,GMESH)
      IF(PVR.NE.PVRATE) CALL OPERR('CONVRG',10)
      IF(YBS.NE.YBASE) $=$*PVPER$(YBASE,YBS)
      $NUCL(ITER)=$*1.D3*RDFACT
      WRITE(WT,900) ITER,$NUCL(ITER)
      $IMP=$LAST-$NUCL(ITER)
      IF(($IMP.GT.$CRIT.OR.$IMPLS.GT.$CRIT).AND.$IMP.GT.0.0D0) GO TO 70
      CALL OPERR('CONVRG',9)
      $LAST=$NUCL(ITER)+0.01D0
      $IMPLS=$IMP
      GO TO 25
   70 $LAST=$NUCL(ITER)
      $IMPLS=$IMP
      IF(ITERTC.LT.MXITER) GO TO 10
      CALL OPERR('CONVRG',7)
C     FAKE 'IF' AND 'RETURN' TO AVOID COMPILATION WARNING MESSAGE
      IF(.TRUE.) GO TO 30
      RETURN
  900 FORMAT('0SYSTEM NUCLEAR COST AT ',I3,' TH ITERATION =',-3PF15.3,
     $' THOUS. P.V.DOLLARS')
  901 FORMAT(' SYSTEM NUCLEAR COST AT ',I3,' TH ITERATION =',-3PF15.3,
     $' THOUS. P.V.DOLLARS')
  902 FORMAT('1'/'0',T20,'* * * * *  TRUE OPTIMUM REACHED FOR GIVEN ARC
     $CONSTRAINTS  * * * * *'/'0'/)
      END
      SUBROUTINE CALSHP
C     CALCULATES SHAPE PARAMETERS FOR EACH PERIOD
C     SYSOPT VERSION 12-16-72
      IMPLICIT INTEGER(C,G)
      REAL*8 RDFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
```

```
$,GESFRS,ECUPLM(18),CORDTL,CPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK          SOPT1333
 INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP                                  SOPT1334
 LOGICAL NPM,OPRCOR                                                       SOPT1335
 LOGICAL MIDCYC                                                           SOPT1336
 INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX                                     SOPT1337
 COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),              SOPT1338
$CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),IDNO(15),GWHOLD(15),MWD(15)       SOPT1339
$,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)         SOPT1340
$,DYHOLD(15),TOY(18,15)                                                   SOPT1341
 COMMON/KX/KX(1)                                                          SOPT1342
 COMMON/PDPERM/S(100,15),ALPHA(100,15),BETAP(100,15),FINVAR(100)          SOPT1343
 INTEGER*2 S                                                              SOPT1344
 COMMON/SHPINF/SLNCRT(100),SLNWSR(100),ITRSHP,PDWSBD(100)                 SOPT1345
 LOGICAL PDWSBD                                                           SOPT1346
 REAL LR                                                                  SOPT1347
 REAL*8 SKL,SKL2                                                          SOPT1348
 L=LOC(4,1,1,1)-1                                                         SOPT1349
 DO 40 NP=1,NPERS                                                         SOPT1350
 L=L+1                                                                    SOPT1351
 LOK=L-NPERSP                                                             SOPT1352
 SKL=0.0                                                                  SOPT1353
 SKL2=0.0                                                                 SOPT1354
 MWDTOT=0                                                                 SOPT1355
 DO 20 NR=1,NRCRS                                                         SOPT1356
 LOK=LOK+NPERSP                                                           SOPT1357
 IF(S(NP,NR).NE.2) GO TO 20                                               SOPT1358
 KR=MWD(NR)                                                               SOPT1359
 MWDTOT=MWDTOT+KR                                                         SOPT1360
 LR=KX(LOK)*ALPHA(NP,NR)-BETAP(NP,NR)                                     SOPT1361
 SKL=SKL+KR*LR                                                            SOPT1362
 SKL2=SKL2+KR*LR*LR                                                       SOPT1363
20 CONTINUE                                                               SOPT1364
 SLNWSR(NP)=SKL2/MWDTOT-(SKL/MWDTOT)**2                                   SOPT1365
40 SLNCRT(NP)=FINVAR(NP)-SLNWSR(NP)-REJLVL                                SOPT1366
 RETURN                                                                   SOPT1367
 END                                                                      SOPT1368
```

```
      SUBROUTINE ARCPRT(ITYPE)                                          SOPT1369
C     PRINTS ARCS THROUGH TYPE ITYPE                                    SOPT1370
C     SYSOPT VERSION 12-16-72                                           SOPT1371
      IMPLICIT INTEGER(C,G)                                             SOPT1372
      REAL*8 RDFACT,SGTITL                                              SOPT1373
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART, SOPT1374
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,    SOPT1375
     $MXNPER,MXPCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,   SOPT1376
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER      SOPT1377
     $,GESFRS,ECUPLM(18),CCRCTL,CPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK   SOPT1378
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP                           SOPT1379
      LOGICAL NPM,OPRCOR                                                SOPT1380
      COMMON/SHPINF/SLNCRT(100),SLNWSR(100),ITRSHP,PCWSBD(100)          SOPT1381
      LOGICAL PCWSBD                                                    SOPT1382
      LOGICAL COK                                                       SOPT1383
      DIMENSION DUM1(33),DUM10(33,10),DUM2(17)                          SOPT1384
      EQUIVALENCE (DUM1(1),DUM10(1),DUM2(1))                            SOPT1385
      REAL*8 $BARC$/'ARCS    '/,$RCSB$/'CS ARE 0'/,DUM2                 SOPT1386
      REAL*8 $COST$/'    COST'/                                         SOPT1387
      REWIND NPOT                                                       SOPT1388
      IF(ITYPE.LE.0) RETURN                                             SOPT1389
      COK=.FALSE.                                                       SOPT1390
      WRITE(WT,900) ITER,NFM,IDSTRG,SGTITL                              SOPT1391
   10 READ(NPOT,903) DUM2                                               SOPT1392
      WRITE(WT,903) DUM2                                                SOPT1393
      IF(DUM2(1).EQ.$BARC$.AND.DUM2(3).EQ.$COST$) GO TO 20              SOPT1394
      IF(DUM2(2).EQ.$RCSB$) COK=.TRUE.                                  SOPT1395
      GO TO 10                                                          SOPT1396
   20 READ(NPOT,901) DUM1                                               SOPT1397
      WRITE(WT,901) DUM1                                                SOPT1398
      LLST=LOC(ITYPE+1,1,1,1)-1                                         SOPT1399
      IF(OOK) LLST=LOC(9,0,0,0)-1                                       SOPT1400
      NPRNT=LLST                                                        SOPT1401
      NEXT=1                                                            SOPT1402
      IF(LLST.LT.LOC(6,0,0,1)) GO TO 28                                 SOPT1403
      LTEMP=LLST                                                        SOPT1404
```

```
      LLST=LOC(6,0,0,1)-1                                               SOPT1405
      NPRNT=LLST                                                        SOPT1406
      NEXT=0                                                            SOPT1407
      GO TO 28                                                          SOPT1408
   22 WRITE(WT,904)                                                     SOPT1409
      DO 26 NP=1,NPERS                                                  SOPT1410
      READ(NPOT,901) DUM1                                               SOPT1411
   26 WRITE(WT,905) (DUM1(I),I=1,27),SLNCRT(NP),SLNWSR(NP)              SOPT1412
      NPRNT=LTEMP-(LLST+NPERS)                                          SOPT1413
      LLST=LTEMP                                                        SOPT1414
      NEXT=1                                                            SOPT1415
   28 N10=NPRNT/10                                                      SOPT1416
      N1=NPRNT-N10*10                                                   SOPT1417
      IF(N1.LT.1) GO TO 40                                              SOPT1418
      DO 30 N=1,N1                                                      SOPT1419
      READ(NPOT,901) DUM1                                               SOPT1420
   30 WRITE(WT,901) DUM1                                                SOPT1421
   40 IF(N10.LT.1) GO TO 60                                             SOPT1422
      DO 50 N=1,N10                                                     SOPT1423
      READ(NPOT,901) DUM10                                              SOPT1424
   50 WRITE(WT,901) DUM10                                               SOPT1425
   60 IF(NEXT.EQ.0) GO TO 22                                            SOPT1426
      LLSTX=LOC(9,0,0,0)-1                                              SOPT1427
      NSKIP=LLSTX-LLST                                                  SOPT1428
      IF(NSKIP.GT.0) READ(NPOT,902) (X,I=1,NSKIP)                       SOPT1429
   70 READ(NPOT,901,END=80) DUM1                                        SOPT1430
      WRITE(WT,901) DUM1                                                SOPT1431
      GO TO 70                                                          SOPT1432
   80 IF(OOK) CALL OPERR('ARCPRT',11)                                   SOPT1433
      REWIND NPOT                                                       SOPT1434
      RETURN                                                            SOPT1435
  900 FORMAT('1'/'0ITER =',I4,5X,'NPM+IDSTRG =',L2,I7,5X,               SOPT1436
     $'STRATEGY TITLE : ',1H',10A7,1H')                                 SOPT1437
  901 FORMAT(1X,33A4)                                                   SOPT1438
  902 FORMAT(A4)                                                        SOPT1439
  903 FORMAT(1X,16A8,A4)                                                SOPT1440
```

```
  904 FORMAT('+',T114,'SLNCRT',5X,'SLNWSR')                                SOPT1441
  905 FORMAT(1X,27A4,F11.6,F12.6)                                          SOPT1442
      END                                                                  SOPT1443
      SUBROUTINE SETELE(NR,GMESH)                                          SOPT1444
C     SETS UP NEW ELAME FOR INPUT TO INCORE                                SOPT1445
C     SYSOPT VERSION 12-16-72                                              SOPT1446
      IMPLICIT INTEGER(C,G)                                                SOPT1447
      REAL*8 RDFACT,SGTITL                                                 SOPT1448
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,    SOPT1449
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,       SOPT1450
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPOT,RD,WT,PARCAL,PARCON,      SOPT1451
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER         SOPT1452
     $,GESFRS,ECUPLM(18),CORDTL,OPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK      SOPT1453
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP                              SOPT1454
      LOGICAL NPM,OPRCOR                                                   SOPT1455
      LOGICAL MIDCYC                                                       SOPT1456
      INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX                                 SOPT1457
      COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),          SOPT1458
     $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),IDNO(15),GWHOLD(15),MWD(15)   SOPT1459
     $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)     SOPT1460
     $,DYHOLD(15),TOY(18,15)                                               SOPT1461
      COMMON/KC/KC(1)/KU/KU(1)/KL/KL(1)                                    SOPT1462
      COMMON/KX/KX(1)                                                      SOPT1463
      DATA FAKE/0.03/                                                      SOPT1464
      INTEGER RC                                                           SOPT1465
      CALL ERASE(ELAME,MXESX2*MXRCYC)                                      SOPT1466
      IC=0                                                                 SOPT1467
      IF(MIDCYC(NR)) GO TO 10                                              SOPT1468
      IC=1                                                                 SOPT1469
      ELAME(1,1)=FAKE                                                      SOPT1470
   10 CLIM=CYCRMX(NR)                                                      SOPT1471
      DO 20 RC=1,CLIM                                                      SOPT1472
      CYC=CYCNUM(RC,NR)                                                    SOPT1473
      GBAL=KX(CYC)                                                         SOPT1474
      MIN =KL(CYC)                                                         SOPT1475
      MAX =KU(CYC)                                                         SOPT1476
```

```
      IGMIN=GBAL/GMESH-IAUXM/2
      ILO=MAXO(MIN/GMESH,IGMIN,1)
      IHI=MINO((MAX-1)/GMESH+1,IGMIN+IAUXM)
      IC=IC+1
      ELAME(1,IC)=GBAL
      DO 20 I=ILO,IHI
   20 ELAME(2*(I-ILO)+3,IC)=I*GMESH
      NCYCXS=CYCXS(NR)
      IF(NCYCXS.LT.1) GO TO 40
      DO 30 I=1,NCYCXS
      IC=IC+1
   30 ELAME(1,IC)=GWHXS(I,NR)
   40 RETURN
      END
      SUBROUTINE NEWMRG(NR,GMESH)
C     ALTERS NETWORK ARCS OF TYPE 2 & 3 FOR NEW SET OF MARGINAL COSTS
C     SYSOPT VERSION 12-16-72
      IMPLICIT INTEGER(C,G)
      REAL*8 RDFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
     $,GESFRS,ECUPLM(18),CORDTL,CPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
      LOGICAL NPM,OPRCOR
      LOGICAL MIDCYC
      INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX
      COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),
     $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),IDNO(15),GWHOLD(15),MWD(15)
     $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)
     $,DYHOLD(15),TOY(18,15)
      COMMON/KC/KC(1)/KU/KU(1)/KL/KL(1)
      INTEGER RC
      IC=1
      IF(MIDCYC(NR)) IC=0
```

```
      CLIM=CYCRMX(NR)
      DO 60 RC=1,CLIM
      IC=IC+1
      L=LCC(2,NR,RC,0)
C     TYPE 2   BASE POINT
      GBAL=ELAME(1,IC)
      KU(L)=GBAL
      KL(L)=KU(L)
      KCC=-10000
C     TYPE 3   INCREMENTS
      L=LCC(3,NR,RC,0)-1
      LIM=ECUPLM(IC)
      IF(LIM.LE.0) LIM=1000000
      IF(GBAL.GT.LIM) CALL OPERR('NEWMRG',13)
      NARC=-1
      DO 10 I=3,MXESX2,2
      G=ELAME(I,IC)
      IF(G.LE.0) GO TO 30
      NARC=NARC+1
      IF(LIM.LE.G) GO TO 20
   10 CONTINUE
      GO TO 30
   20 ELAME(I,IC)=LIM
   30 DO 60 I=1,NARC
      ILAM=I+I+2
      LI=L+I
      KC(LI)=1000.*ELAME(ILAM,IC)+0.5
      IF(KC(LI).LT.KCC) CALL OPERR('NEWMRG',5)
      KCC=KC(LI)
      GLO=ELAME(ILAM-1,IC)
      GUP=ELAME(ILAM+1,IC)
      GDEL=GUP-GLO
      IF(GBAL.GT.GUP) GO TO 50
      IF(GBAL.LT.GLO) GO TO 40
      KU(LI)=GUP-GBAL
      KL(LI)=GLO-GBAL
```

```
      GO TO 60
   40 KU(LI)=GDEL
      GO TO 60
   50 KL(LI)=-GDEL
   60 CONTINUE
      RETURN
      END
      FUNCTION PVPER$(T,TBASE)
C     CALCULATE PRESENT VALUE AT TIME T OF 1$ AT TIME TBASE
C     SYSOPT VERSION 12-16-72
      REAL*8 PVPER$,LN1PX
      PVPER$=DEXP(-LN1PX*(T-TBASE))
      RETURN
      ENTRY PVINIT(PVRATE)
C     PRE-CALCULATE LOG OF (1+X) IN UNITS OF INVERSE YEARS
      LN1PX=DLOG(1.D0+PVRATE)
      PVINIT=LN1PX
      RETURN
      END
      SUBROUTINE CHKSHP(SHPSCK)
C     CHECKS SHAPE CRITERIA TO EVALUATE FEASIBILITY
C     SYSOPT VERSION 12-16-72
      IMPLICIT INTEGER(C,G)
      REAL*8 RDFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
     $,GESFRS,ECUPLM(18),CORDTL,OPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
      LOGICAL NPM,OPRCOR
      COMMON/KC/KC(1)/KU/KU(1)/KL/KL(1)
      COMMON/KX/KX(1)
      COMMON/SHPINF/SLNCRT(100),SLNWSR(100),ITRSHP,PCWSBD(100)
      LOGICAL PCWSBD
      DIMENSION DELTAL(100)
```

```
      LOGICAL SHPSOK,PCDOK                                              SOPT1585
      INTEGER KSTHLD(20)                                                SOPT1586
      PCDOK=PCDELA.GT.1.                                                SOPT1587
      LFRS=LOC(3,1,1,1)                                                 SOPT1588
      LJFRS=LOC(7,1,1,1)                                                SOPT1589
      NZERO=IAUXM*NCYCT                                                 SOPT1590
      CALL ERASE(KC(LFRS),NZERO,KU(LFRS),NZERO,KL(LFRS),NZERO)          SOPT1591
C     SET UP ARCS TO ATTEMPT MINIMIZING SHAPE CRITERIA                  SOPT1592
      LAUX=LFRS-IAUXM                                                   SOPT1593
      DO 10 NC=1,NCYCT                                                  SOPT1594
      LAUX=LAUX+IAUXM                                                   SOPT1595
      KL(NCYCT+NC)=KX(NC)                                               SOPT1596
      KU(NCYCT+NC)=KX(NC)                                               SOPT1597
      KC(LAUX)=10000                                                    SOPT1598
      KU(LAUX)=100000                                                   SOPT1599
      KC(LAUX+1)=-10000                                                 SOPT1600
   10 KL(LAUX+1)=-100000                                                SOPT1601
      LAUX=LJFRS-1                                                      SOPT1602
      DO 15 NP=1,NPERS                                                  SOPT1603
      DO 15 NR=1,NRCRS                                                  SOPT1604
      DO 15 J=1,JFRPBK                                                  SOPT1605
      LAUX=LAUX+1                                                       SOPT1606
   15 KC(LAUX)=KSTHLD(J)                                                SOPT1607
      ITRSHP=ITRSHP+1                                                   SOPT1608
      WRITE(WT,905) ITRSHP                                              SOPT1609
      CALL OCKMAN                                                       SOPT1610
      CALL ARCPRT(0)                                                    SOPT1611
      CALL CALSHP                                                       SOPT1612
      SHPSOK=.TRUE.                                                     SOPT1613
      DO 20 NP=1,NPERS                                                  SOPT1614
      DELTAL(NP)=1.E50                                                  SOPT1615
      IF(SLNCRT(NP).GE.0.0) GO TO 20                                    SOPT1616
      SHPSOK=.FALSE.                                                    SOPT1617
      PDWSBD(NP)=.TRUE.                                                 SOPT1618
      DELTAL(NP)=SQRT(-SLNCRT(NP))                                      SOPT1619
      IF(PCDOK) CALL SQUEEZ(NP,PCDELA*DELTAL(NP)*0.01)                  SOPT1620
```

```
   20 CONTINUE                                                          SOPT1621
      IF(PARCOP.GE.4) CALL ARCPRT(PARCOP)                               SOPT1622
      WRITE(WT,906) ITRSHP                                              SOPT1623
      WRITE(WT,910) (SLNCRT(N),N=1,NPERS)                               SOPT1624
      IF(SHPSOK) GO TO 40                                               SOPT1625
      WRITE(WT,920) (DELTAL(N),N=1,NPERS)                               SOPT1626
      WRITE(WT,930)                                                     SOPT1627
      IF(PCDCK) GO TO 40                                                SOPT1628
      WRITE(WT,940) PCDELA                                              SOPT1629
      GO TO 40                                                          SOPT1630
      ENTRY ONLY$$                                                      SOPT1631
      ITRSHP=0                                                          SOPT1632
      LJFRS=LOC(7,1,1,1)                                                SOPT1633
      LAUX=LJFRS-1                                                      SOPT1634
      DO 30 J=1,JFRPBK                                                  SOPT1635
   30 KSTHLD(J)=KC(LAUX+J)                                              SOPT1636
      DO 35 NP=1,NPERS                                                  SOPT1637
   35 PDWSBD(NP)=.FALSE.                                                SOPT1638
   40 NZERO=JFRPBK*NRCRS*NPERS                                          SOPT1639
      CALL ERASE(KC(LJFRS),NZERO)                                       SOPT1640
      RETURN                                                            SOPT1641
  905 FORMAT('1'/'2',T20,'* * * * *  ENTERING SHAPE ITERATION NUMBER',  SOPT1642
     $  I4,'   * * * * *')                                              SOPT1643
  906 FORMAT('1'/'2',T20,'* * * * *  RESULTS FOR SHAPE ITERATION NUMBER'SOPT1644
     $ ,I4,'   * * * * *')                                              SOPT1645
  910 FORMAT('0'/'0 SLNCRT(NP), NP=1,NPERS :'/(10F10.6))                SOPT1646
  920 FORMAT('0'/'0 DELTAL(NP), NP=1,NPERS :'/(10F10.6))                SOPT1647
  930 FORMAT('0'/'0',T20,'SHAPE CRITERION REQUIRES ANOTHER OUTER ITERATISOPT1648
     $ON')                                                              SOPT1649
  940 FORMAT('0',10X,'EXCEPT THAT PCDELA =', F7.2,' PERCENT PREVENTS REQSOPT1650
     $UIRED IMPROVEMENT')                                               SOPT1651
      END                                                               SOPT1652
      SUBROUTINE SQUEEZ(NP,DEL)                                         SOPT1653
C     SQUEEZES PERIOD CAPACITY FACTOR RANGE BY DEL ON BOTH ENDS         SOPT1654
C     SYSOPT VERSION 12-16-72                                           SOPT1655
      IMPLICIT INTEGER(C,G)                                             SOPT1656
```

```
      REAL*8 RCFACT,SGTITL
      CCMMCN/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
     $,GESFRS,ECUPLM(18),CORCTL,OPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
      LCGICAL NPM,OPRCOR
      CCMMCN/KC/KC(1)/KU/KU(1)/KL/KL(1)
      COMMON/PDPERM/S(100,15),ALPHA(100,15),BETAP(100,15),FINVAR(100)
      INTEGER*2 S
      CCMMCN/SHPINF/SLNCRT(100),SLNWSR(100),ITRSHP,PCWSBD(100)
      LCGICAL PCWSBD
      REAL LMAX,LMIN
      DO 60 NR=1,NRCRS
      A=ALPHA(NP,NR)
      B=BETAP(NP,NR)
      LJFRS=LOC(7,NR,0,NP)
      IF(KU(LJFRS).LE.0) GO TO 60
      LAUX=LJFRS-1
      KFMIN=0
      KFMAX=0
      DO 20 J=1,JFRPBK
      LAUX=LAUX+1
      KFMIN=KFMIN+KL(LAUX)
   20 KFMAX=KFMAX+KU(LAUX)
      LINIT=LOC(4,NR,0,NP)
      KIMIN=KL(LINIT)
      KIMAX=KU(LINIT)
      KAMIN=MAX0(KFMIN,KIMIN)
      KAMAX=MIN0(KFMAX,KIMAX)
      LMAX=A*KAMAX-B
      LMIN=A*KAMIN-B
      DEL=AMIN1(DEL,(LMAX-LMIN)/3.)
      LMAX=LMAX-DEL
      LMIN=LMIN+DEL
```

```
C       PLACE NEW CONSTRAINTS ON ARCS
        MOX=(LMAX+B)/A
        MON=(LMIN+B)/A+0.5
        MXD=KFMAX-MOX
        MND=MON-KFMIN
        LFRS=LOC(7,NR,0,NP)
        LAUX=LFRS
        JF=1
   30   JF=JF+1
        IF(JF.GT.JFRWRD) GO TO 40
        LAUX=LAUX+1
        MW=KL(LAUX)
        IF(MXD.LT.MW) GO TO 35
        KU(LAUX)=0
        MXD=MXD-MW
        GO TO 30
   35   KU(LAUX)=MW-MXD
   40   LAUX=LFRS+JFRPBK
        JB=JBKWRD+1
   50   JB=JB-1
        IF(JB.LT.1) GO TO 60
        LAUX=LAUX-1
        MW=-KL(LAUX)
        IF(MND.LT.MW) GO TO 55
        KL(LAUX)=0
        MND=MND-MW
        GO TO 50
   55   KL(LAUX)=-(MW-MND)
   60   CONTINUE
        RETURN
        END
        SUBROUTINE EDTSHP(SHPSOK)
C       EDITS SHAPE INFO. AND PRINTS FINAL ALTERED ENERGY LIMITS
C       SYSOPT VERSION 12-16-72
        IMPLICIT INTEGER(C,G)
        REAL*8 RDFACT,SGTITL
```

```
      COMMON/OPTLIM/RCFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,   SOPT1729
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,      SOPT1730
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,    SOPT1731
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER       SOPT1732
     $,GESFRS,ECUPLM(18),CCRDTL,CPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK    SOPT1733
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP                            SOPT1734
      LOGICAL NPM,CPRCOR                                                 SOPT1735
      COMMON/KC/KC(1)/KU/KU(1)/KL/KL(1)                                  SOPT1736
      COMMON/SHPINF/SLNCRT(100),SLNWSR(100),ITRSHP,PDWSBD(100)           SOPT1737
      LOGICAL PDWSBD                                                     SOPT1738
      LOGICAL SHPSOK                                                     SOPT1739
      DATA STAR/'* * '/,$NOT/'NOT '/                                     SOPT1740
      WORD=STAR                                                          SOPT1741
      IF(.NOT.SHPSOK) WORD=$NOT                                          SOPT1742
      KEY=0                                                              SOPT1743
      IF(ITRSHP.LE.1) KEY=2                                              SOPT1744
      WRITE(WT,910) KEY,ITRSHP                                           SOPT1745
      IF(KEY.EQ.2)  WRITE(WT,920)                                        SOPT1746
      WRITE(WT,911) WORD                                                 SOPT1747
      IF(KEY.EQ.2) RETURN                                                SOPT1748
      WRITE(WT,930)                                                      SOPT1749
      DO 80 NP=1,NPERS                                                   SOPT1750
      IF(.NOT.PDWSBD(NP)) GO TO 80                                       SOPT1751
      WRITE(WT,900)                                                      SOPT1752
      DO 60 NR=1,NRCRS                                                   SOPT1753
      LJFRS=LOC(7,NR,0,NP)                                               SOPT1754
      LAUX=LJFRS-1                                                       SOPT1755
      KFMIN=0                                                            SOPT1756
      KFMAX=0                                                            SOPT1757
      DO 20 J=1,JFRPBK                                                   SOPT1758
      LAUX=LAUX+1                                                        SOPT1759
      KFMIN=KFMIN+KL(LAUX)                                               SOPT1760
   20 KFMAX=KFMAX+KU(LAUX)                                               SOPT1761
      LINIT=LOC(4,NR,0,NP)                                               SOPT1762
      KIMIN=KL(LINIT)                                                    SOPT1763
      KIMAX=KU(LINIT)                                                    SOPT1764
```

```
      KFMIN=MAXO(KFMIN,KIMIN)                                           SOPT1765
      KFMAX=MINO(KFMAX,KIMAX)                                           SOPT1766
      KFDEL=KFMAX-KFMIN                                                 SOPT1767
      KIDEL=KIMAX-KIMIN                                                 SOPT1768
      PCDEL=KFDEL*100./(KIDEL+1.E-20)                                   SOPT1769
      IF(KU(LJFRS).LE.0) PCDEL=0.0                                      SOPT1770
   60 WRITE(WT,940)NP,NR,KIMAX,KIMIN,KIDEL,KFMAX,KFMIN,KFDEL,PCDEL      SOPT1771
   80 CONTINUE                                                          SOPT1772
      RETURN                                                            SOPT1773
  900 FORMAT('0')                                                       SOPT1774
  910 FORMAT('1'/I1,T20,'* * * * * ',I4,'   SHAPE ITERATIONS WERE REQUIR SOPT1775
     $ED  * * * * *')                                                   SOPT1776
  911 FORMAT('C',T20,'* * * * * ',A4,'  ALL FINAL SHAPES MET SHAPE CRITE SOPT1777
     $RION  * * * * *')                                                 SOPT1778
  920 FORMAT('0',T40,'THEREFORE, NO PERIODS WERE ALTERED * * * * *')    SOPT1779
  930 FORMAT('0'/'0',10X,'ALTERED PERIOD ENERGY RANGE LIMITS  ( G W H E SOPT1780
     $)'/'OPERIOD  REACTOR  INIT.MAX  INIT.MIN  INIT.DEL    FINL.MAX  F SOPT1781
     $INL.MIN  FINL.DEL   % INIT.DEL'/)                                 SOPT1782
  940 FORMAT(I5,I8,3I10,3X,3I10,F12.1)                                  SOPT1783
      END                                                               SOPT1784
      SUBROUTINE OPTMUM(OPTRCH,$NKPRD)                                  SOPT1785
C     SUPERVISES PRINTING OF OPTIMUM SOLUTION                           SOPT1786
C     SYSOPT VERSION 12-16-72                                           SOPT1787
      IMPLICIT INTEGER(C,G)                                             SOPT1788
      REAL*8 RDFACT,SGTITL                                              SOPT1789
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART, SOPT1790
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,    SOPT1791
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,   SOPT1792
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER      SOPT1793
     $,GESFRS,ECUPLM(18),CORDTL,CPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK   SOPT1794
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP                           SOPT1795
      LOGICAL NPM,OPRCOR                                                SOPT1796
      LOGICAL MIDCYC                                                    SOPT1797
      INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX                              SOPT1798
      COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),       SOPT1799
     $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),ICNO(15),GWHOLD(15),MWD(15) SOPT1800
```

```
      $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)
      $,DYHOLD(15),TOY(18,15)
       COMMON/FINALS/S4,SA4,SP4,SL4,SP8
       REAL*8 S4(13),SA4(13),SP4(13),SL4(13),SP8(13)
       COMMON/PRINTS/RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS,KRD,KWT
       LOGICAL RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS
       INTEGER NECBAL(18)/18*1/
       REAL*8 $NKPRD,$DEL,WORD,$FORC$/'FORCED'/,$TRUE$/' TRUE'/,$,RTC
       LOGICAL OPTRCH,STORE(6),USE(6)
       EQUIVALENCE (USE(1),RELCST)
       DO 10 I=1,6
       STORE(I)=USE(I)
    10 USE(I)=OPRCOR(I)
       IF(CORDTL.LE.0) GO TO 30
       IF(OPRCOR(3).OR.OPRCOR(4).OR.OPRCOR(5).OR.OPRCOR(6)) GO TO 20
       GO TO 30
    20 GHUGE=10**6
       $=0.0D0
       DO 28 NR=1,NRCRS
       CALL SETELE(NR,GHUGE)
       DO 26 I=1,MXRCYC
    26 ELAME(3,I)=0.0
       IDNUM=IDNC(NR)
       NCYCIN=CYCRMX(NR)
       IF(.NOT.MIDCYC(NR)) NCYCIN=NCYCIN+1
       NCYCXS=CYCXS(NR)
       ECHDOV=GWHOLD(NR)
       CALL INCCRE(IDNUM,NCYCIN,NCYCXS,NCYCIN+NCYCXS,TSY(1,NR),TEY(1,NR),
      $NECBAL,ELAME,MXESX2,ECHDOV,RTC,PVR,YBS,ECUPLM,TOY(1,NR))
    28 $=$+RTC
       $NKPRD=$*1.D3*RDFACT
    30 DO 40 I=1,6
    40 USE(I)=STORE(I)
       IF($NKPRD.GE.1.D20) GO TO 20
       $DEL=$NKPRD-SP8(3)
       SP8(2)=SP8(2)+$DEL
```

```
      SP8(3)=SP8(3)+$DEL
      SP8(8)=SP8(8)+$DEL
      SP8(9)=SP8(9)+$DEL
      SP8(12)=SP8(12)+$DEL
      WORD=$FORC$
      IF(OPTRCH) WORD=$TRUE$
      WRITE(WT,904) NPM,IDSTRG,SGTITL
      WRITE(WT,901) WORD,$NKPRD
      WRITE(WT,907)
      WRITE(WT,908) (S4(I),I=2,13),(SA4(I),I=2,13),(SP4(I),I=2,13),
     $(SL4(I),I=2,13)
      WRITE(WT,909) YBASE
      WRITE(WT,910) (SP8(I),I=2,12)
      WRITE(WT,902) WORD,SP8(12)
      RETURN
  901 FORMAT('0'/'0',T20,'AT ',A6,' OPTIMUM, $NKPRD = ',-3PF15.3,
     $' THOUS. P.V. DOLLARS'/'0')
  902 FORMAT('0'/'0',T20,'AT ',A6,' OPTIMUM, TOTAL SYSTEM COST = ',
     $-3PF15.3,' THOUS. P.V. DOLLARS'/'0')
  904 FORMAT('1'/'0'/,10X,'NPM+IDSTRG =',L2,I7,5X,
     $'STRATEGY TITLE : ',1H',10A7,1H')
  907 FORMAT('0          --------- M E G A W A T T S ---------  FRACT.',
     $4X,'------------------ GEGAWATT-HOURS ELECTRIC ------------------'
     $       /' PERIOD   MWINST  MWONLN  MWPEAK  MWMRGN  MWSPIN   PLOFL',
     $4X,'EXPDEM     EXPGEN     XNKGEN     XNNGEN     EXPEMR     UNSRVD'
     $)
  908 FORMAT('OTOTAL :',5F8.0,F8.4,6F11.2/
     $       'OAVG.  :',5F8.0,F8.4,6F11.2/
     $       'OPVTOTL:',5F8.0,F8.4,6F11.2/
     $       'OLVAVG.:',5F8.0,F8.4,6F11.2/)
  909 FORMAT('0'/'0',T20,'ALL COSTS IN THOUSANDS OF DOLLARS PRESENT VALU
     $ED TO YBASE =',F9.4,' YEARS'/
     $     ' PERIOD      PRCD$      $NKPRD      $NNPRD      SUSD$      $NKS
     $US      $NNSUS      $SBTOT      $NKTOT      $NNTOT      EMRP$      TOTA
     $L$')
  910 FORMAT('OPVTOTL:',-3P11F11.2)
```

```
      END
      FUNCTION LOC(ITYPE,R,C,P)
C     CALCULATES LOC AS POINTER TO DESIRED ARC
C     SYSOPT VERSION 12-16-72
      IMPLICIT INTEGER(C,G)
      REAL*8 RDFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
     $,GESFRS,ECUPLM(18),CORDTL,OPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
      LOGICAL NPM,OPRCOR
      LOGICAL MIDCYC
      INTEGER*2 CYCNUM,CYCRNG,CYCXS,CYCRMX
      COMMON/RCRDAT/DYDWN(3,15),DYUP(3,15),GWHXS(3,15),CYCXS(15),
     $CYCRMX(15),CYCNUM(18,15),CYCRNG(2,270),ICNO(15),GWHOLD(15),MWD(15)
     $,TSY(18,15),TEY(18,15),INSTAT(15),MWMIN(15),MWMAX(15),MIDCYC(15)
     $,DYHOLD(15),TOY(18,15)
      INTEGER R,C,P
      GO TO (1,2,3,4,5,6,7,8,9,10),ITYPE
    1 LOC=CYCNUM(C,R)
      RETURN
    2 LOC=LOC1X+CYCNUM(C,R)
      RETURN
    3 LOC=LOC2X+(CYCNUM(C,R)-1)*IAUXM+1
      RETURN
    4 IF(P.GT.NPERS) GO TO 44
      LOC=LOC3X+(R-1)*NPERSP+P
      RETURN
   44 LOC=LOC3X+R*NPERSP
      RETURN
    5 LOC=LOC4X+1
      RETURN
    6 LOC=LOC5X+P
      RETURN
```

```
    7 LOC=LOC6X+((P-1)*NRCRS+R-1)*JTOTAL+1
      RETURN
    8 CONTINUE
    9 LOC=LOC9X
      RETURN
C     INITIALIZATION
   10 JTOTAL=JFRWRD+JBKWRD
      LOC1X=NCYCT
      LOC2X=LOC1X+NCYCT
      LOC3X=LOC2X+IAUXM*NCYCT
      LOC4X=LOC3X+NPERSP*NRCRS
      LOC5X=LOC4X+3
      LOC6X=LOC5X+NPERS
      LOC7X=LOC6X+JTOTAL*NRCRS*NPERS
      LOC9X=LOC7X+1
      LOC=0
      RETURN
      END
      SUBROUTINE  OPERR(SUBR,JERR)
C     WRITES OUT ALL ERROR MESSAGES FOR SYSOPT
C     SYSOPT VERSION 12-16-72
      IMPLICIT INTEGER(C,G)
      REAL*8 RDFACT,SGTITL
      COMMON/OPTLIM/RDFACT,SGTITL(10),ELAME(40,18),PVRATE,YBASE,YSTART,
     $IAUX,IAUXM,NRCRS,NCYCT,NPERS,NPERSP,NPERIN,ITER,MXESX2,MXRCYC,
     $MXNPER,MXRCRS,MXNODS,MXARCS,SIOT,NPIN,NPCT,RD,WT,PARCAL,PARCON,
     $PARCOP,PCONVG,NPM,IDSTRG,JFRWRD,JBKWRD,NMESH,MESH(15),MXITER
     $,GESFRS,ECUPLM(18),CORDTL,OPRCOR(6),REJLVL,PCDELA,TH$CON,JFRPBK
      INTEGER SIOT,RD,WT,PARCAL,PARCON,PARCOP
      LOGICAL NPM,OPRCOR
      INTEGER ERRCOD
      REAL*8 SUBR,$QUIT$
      DATA NPRINT/0/,$QUIT$/' QUIT'/,ERRCOD/0/,MAXERR/16777216/
C     MAXERR=16**6
      REAL*8 CC1(11)/'CDFMIN AND CDFMAX DATA ARE INCONSISTENT IN SOME SE
     $NSE                                       '/
```

```
      REAL*8 C02(11)/'MWDTOT.NE.LVLMAX-LVLMIN    .OR.    MW.GE.PEMIN          SOPT1945
     $                                             '/                         SOPT1946
      REAL*8 C03(11)/'REACTOR OR STRATEGY ID''S DO NOT AGREE                  SOPT1947
     $                                             '/                         SOPT1948
      REAL*8 C04(11)/'NUMBER OF ARCS INPUT TO O-O-K AND ARC EQ. DO NOT A      SOPT1949
     $GREE                                         '/                         SOPT1950
      REAL*8 C05(11)/'MARGINAL COST CURVE NOT MONOTONICALLY DECREASING        SOPT1951
     $                                             '/                         SOPT1952
      REAL*8 C06(11)/'IMPROPER INPUT SEQUENCE &/OR CARD; INPUT OPTIONS O      SOPT1953
     $UTSIDE CURRENT LIMITS                        '/                         SOPT1954
      REAL*8 C07(11)/'MXITER REACHED WITHOUT COMPLETE CONVERGENCE             SOPT1955
     $                                             '/                         SOPT1956
      REAL*8 C09(11)/'$NUCL NOT CONVERGING RAPIDLY TO MINIMUM ;      ASS      SOPT1957
     $UME COST HAS CONVERGED FOR THIS GMESH '/                                SOPT1958
      REAL*8 C10(11)/'INCORE AND SYSOPT USING DIFFERENT P.V.RATES             SOPT1959
     $                                             '/                         SOPT1960
      REAL*8 C11(11)/'O-O-K NETWORK SOLUTION IS TRULY OUT-OF-KILTER           SOPT1961
     $                                             '/                         SOPT1962
      REAL*8 C12(11)/'PREMATURE END TO SYSINT DATA ; SOME PERIODS NOT RE      SOPT1963
     $AD IN                                        '/                         SOPT1964
      REAL*8 C13(11)/' CYCLE ENERGY GREATER THAN ITS UPPER LIMIT              SOPT1965
     $                                             '/                         SOPT1966
      IERR=JERR                                                               SOPT1967
  100 ERRCOD=MOD(ERRCOD,MAXERR)                                               SOPT1968
      ERRCOD=16*ERRCOD+IERR                                                   SOPT1969
      NPRINT=NPRINT+1                                                         SOPT1970
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13),IERR                              SOPT1971
    1 WRITE(WT,900) SUBR,ERRCOD,C01,NPRINT                                    SOPT1972
      GO TO 1000                                                              SOPT1973
    2 WRITE(WT,900) SUBR,ERRCOD,C02,NPRINT                                    SOPT1974
      GO TO 1000                                                              SOPT1975
    3 WRITE(WT,900) SUBR,ERRCOD,C03,NPRINT                                    SOPT1976
      GO TO 1000                                                              SOPT1977
    4 WRITE(WT,900) SUBR,ERRCOD,C04,NPRINT                                    SOPT1978
      GO TO 1000                                                              SOPT1979
    5 WRITE(WT,900) SUBR,ERRCOD,C05,NPRINT                                    SOPT1980
```

```
      RETURN
    6 WRITE(WT,900) SUBR,ERRCOD,C06,NPRINT
      GO TO 1000
    7 WRITE(WT,900) SUBR,ERRCOD,C07,NPRINT
      RETURN
    8 WRITE(WT,908) SUBR,ERRCOD,NPRINT,NPRINT
      CALL ICERRS('OPERR ',8)
      STOP
    9 WRITE(WT,900) SUBR,ERRCOD,C09,NPRINT
      RETURN
   10 WRITE(WT,900) SUBR,ERRCOD,C10,NPRINT
      GO TO 1000
   11 WRITE(WT,900) SUBR,ERRCOD,C11,NPRINT
      GO TO 1000
   12 WRITE(WT,900) SUBR,ERRCOD,C12,NPRINT
      GO TO 1000
   13 WRITE(WT,900) SUBR,ERRCOD,C13,NPRINT
      RETURN
 1000 NPRINT=NPRINT+1
      WRITE(WT,999) NPRINT
      SUBR=$QUIT$
      IERR=8
      GO TO 100
  900 FORMAT(/' ',130('-')/,' | SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $11A8,T131,'|',/,' ',130('-'),I2)
  908 FORMAT(/' ',130('-')/,' | SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'RDCPTN ENCOUNTERED STOP CARD,  OPERR CALLED ONCE TOO OFTEN OR O',
     $'THER FATAL ERROR',  T131,'|'/' | DURING THIS ENTIRE RUN,  OPERR',
     $' PRINTED A TOTAL OF ',I3,' ERROR MESSAGES JUST LIKE (AND ',
     $'INCLUDING) THIS ONE',
     $T131,'|',/,' ',130('-'),I2)
  999 FORMAT(/' ',130('-')/,' | PREVIOUS ERROR SEVERE ENOUGH TO',
     $' INVALIDATE FURTHER COMPUTATIONS. THEREFORE,',
     $' TERMINATING EXECUTION.',
     $T131,'|',/,' ',130('-'),I2)
       END
```

```
      SUBRCUTINE CMPTIM(LV,ENT)
C     PRINTS TIME OF INTRA-SUBROUTINE TRANSFERS OR CATE&TIME
C     SYSOPT VERSION 03-06-72
C     "TIMING" IS AN M.I.T. INTERNAL SUBROUTINE THAT RETURNS THE CPU TIME
C     IN HUNDRECTHS OF SECONDS.
C     "WHEN" IS AN M.I.T. INTERNAL SUBROUTINE THAT RETURNS THE DATE AND
C     TIME IN THE FOLLOWING 5A4 FORMAT:  MM/DD/YY HR*MI*SS.FF
      CIMENSION A(5)
      DOUBLE PRECISION LV,ENT
      INTEGER  TNOW,TSTART,TREL
      INTEGER WT
      CALL TIMING(TNOW)
      TREL=TNOW-TSTART
      IF(TREL.LT.0)   TREL=TREL+8640000
      TI=TREL/100.
      WRITE(WT,10)LV,ENT,TI
      RETURN
      ENTRY STRTIM(WT)
      CALL TIMING(TSTART)
      CALL WHEN(A)
      WRITE(WT,20) A
      RETURN
   10 FORMAT(/,T103,29('*'),/,T103,'* LV.   ',A6,T131,'*',/,
     $T103,'* ENT. ',A6,' @',F7.2,' SEC. *',/,T103,29('*'),/)
   20 FORMAT(/T103,29('*')/T103,'*     DATE =     ',2A4,T131,'*'/
     $T103,'*     TIME = ',  3A4,T131,'*'/T103,29('*')/)
      END
********************************************************************* 00000000
*                                                                   * 00000010
*                 ASSEMBLER LANGUAGE SUBRCUTINE ERASE               * 00000011
*                       WRITTEN BY JOHN W. KIDSON                   * 00000012
*                     MIT DEPARTMENT OF METEOROLOGY                 * 00000014
*                                                                   * 00000016
*        TO SET ELEMENTS OF REAL OR INTEGER ARRAYS TO ZERO. A1,A2,... * 00000020
*        ARE ARRAY NAMES AND N1,N2,... ARE INTEGER VALUES OR        * 00000030
*        EXPRESSICNS GIVING THE ARRAY SIZES.                        * 00000040
```

```
**       I.E. - CALL ERASE(C,26*31,N,7*31,E,254)                      ** 00000050 SOPT2053
*                                                                      * 00000060 SOPT2054
*********************************************************************** 00000070 SOPT2055
ERASE    START 0                                                         00000080 SOPT2056
         SAVE  (14,12),,*                                                00000090 SOPT2057
         BALR  12,0                                                      00000100 SOPT2058
         USING *,12                                                      00000110 SOPT2059
         SR    0,0                                                       00000120 SOPT2060
         SR    2,2            PARAMETER LIST INDEX=0                     00000130 SOPT2061
         L     6,=F'4'                                                   00000140 SOPT2062
E1       L     3,0(2,1)       LOAD 3 WITH ARRAY ADDRESS                  00000150 SOPT2063
         L     4,4(2,1)       LOAD 4 WITH ADDRESS OF ARRAY LENGTH        00000160 SOPT2064
         L     7,0(4)         LOAD 7 WITH ARRAY LENGTH-1 TIMES 4         00000170 SOPT2065
         SLA   7,2                                                       00000180 SOPT2066
         SR    7,6                                                       00000190 SOPT2067
         SR    5,5                                                       00000200 SOPT2068
E2       ST    0,0(5,3)       STORE ZERO                                 00000210 SOPT2069
         BXLE  5,6,E2                                                    00000220 SOPT2070
         LTR   4,4            TEST FOR LAST ARGUMENT IN LIST             00000230 SOPT2071
         BM    RETN                                                      00000240 SOPT2072
         A     2,=F'8'                                                   00000250 SOPT2073
         B     E1             PICK UP NEXT ARGUMENT PAIR                 00000260 SOPT2074
RETN     RETURN (14,12),T                                                00000270 SOPT2075
         END                                                             00000280 SOPT2076
*********************************************************************** 00000290 SOPT2077
```

```
C      OUT-OF-KILTER MAIN PROGRAM
C      ONLY DIMENSION STATEMENTS IN THIS PROGRAM NEED BE CHANGED TO
C      ALTER MAXIMUM ARCS OR MAXIMUM NODES  ALLOWABLE
C      IF A = MAXIMUM ARCS  AND  N = MAXIMUM NODES ,
C      KL,KC,KU,KX, AND JI ARE DIMENSIONED BY 'A'
C      NL(N),NN(2*N),NP(N),IJ(MAX(N,A-2*(N+1))),IL(N+1),JL(N+1)
       DIMENSION KL(2000),KC(2000),KU(2000),KX(2000),NL(1000)
       DIMENSION NN(2000),NP(1000),IJ(1000),IL(1001),JL(1001),JI(2000)
       DIMENSION LC(9),KA(18,2),KQ(9)
       COMMON /KL/KL/KC/KC/KU/KU/KX/KX/NL/NL/NN/NN/NP/NP/IJ/IJ/IL/IL
       COMMON /JL/JL/JI/JI
       COMMON          /M/M/N/N/LER/LER/KAT/KAT/KOR/KOR/KTER/KTER
       COMMON /MINE/MINE/LC/LC/KA/KA/IFIN/IFIN/KI/KI/KO/KO/KQ/KQ/K/K
C     SYSTEM INPUT DEVICE
       KI=5
C     SYSTEM OUTPUT DEVICE
       KO=6
C     CARD PUNCH
       KQ(1)=7
C     RESERVED OUTPUT TAPE
       KQ(2)=3
C     RESERVED INPUT TAPE
       KQ(3)=2
C     MAXIMUM ARCS
       KQ(4)=2000
C     MAXIMUM NODES
       KQ(5)=1000
       KQ(6)=0
       KQ(9)=0
       IFIN=32767
       CALL MAINE
       STOP
       END
       SUBROUTINE MAINE
       DIMENSION LC(9),KA(18,2),KQ(9)
       COMMON/KL/KL(1)/KC/KC(1)/KU/KU(1)/KX/KX(1)/NL/NL(1)
```

```
      COMMON/NN/NN(1)/NP/NP(1)/IJ/IJ(1)/IL/IL(1)/JL/JL(1)/JI/JI(1)
      COMMON            /M/M/N/N/LER/LER/KAT/KAT/KOR/KOR/KTER/KTER
      COMMON /MINE/MINE/LC/LC/KA/KA/IFIN/IFIN/KI/KI/KO/KO/KQ/KQ/K/K
      DIMENSION KE(101)
      CALL STRTIM(KO)
  100 CALL CMPTIM('      ','DATA IN')
      CALL PREDAT(KS)
      IF(KS.EQ.-1) RETURN
      IF(KS.NE.0) GO TO 1
      CALL ARCASY(L)
      CALL MAKEJL
      IF(LER.GE.KQ(4)) GO TO 88
      LER=LER*KQ(8)
      IF(L.EQ.0) GO TO 1
      CALL NODASY
    1 CALL READER
      CALL TRANSL
      IF(LER.NE.0) GO TO 88
      CALL CMPTIM('DATAIN','ALGOR.')
      I=0
      KUP=1
      KE(101)=0
      DO 26 K=1,N
      IF(K.LT.KUP) GO TO 38
      I=I+1
      KUP=LDECR(IL(I+1))
   38 CALL KILTER(I)
      IF(LER.EQ.0) GO TO 26
      IF(LER.NE.107) GO TO 24
      KE(101)=KE(101)+1
      KF=KE(101)
      IF(KE(101).GT.100) GO TO 26
      KE(KF)=K
   26 CONTINUE
C    COMPLETED CHECKING ALL ARCS
      LER=0
```

```
   99 CALL CMPTIM('ALGOR.','OUTPUT')
      CALL OUTPUT(KE)
      CALL CMPTIM('OUTPUT','      ')
C     CYCLE BACK FOR ANOTHER RUN
      GO TO 100
   24 WRITE(KO,54)
      LL=LADDR(IJ(K))
      WRITE(KO,55) NN(2*I-1),NN(2*I),NN(2*LL-1),NN(2*LL)
      GO TO 99
   88 WRITE(KO,56)
      STOP
      ENTRY OOKMAN
C     ENTRY TO OOK FROM OTHER CODES (WHICH HAVE ALREADY CALLED STRTIM)
      COMMON/OOKCOM/KIX,KOX,KQ1X,KQ2X,KQ3X,KQ4X,KQ5X
      KI=KIX
      KO=KOX
      KQ(1)=KQ1X
      KQ(2)=KQ2X
      KQ(3)=KQ3X
      KQ(4)=KQ4X
      KQ(5)=KQ5X
      KQ(6)=0
      KQ(9)=0
      IFIN=32767
      IF(.TRUE.) GO TO 100
      STOP
   51 FORMAT(A4)
   54 FORMAT(24H0OVERFLOW IN NODE PRICES)
   55 FORMAT(23H0RUN TERMINATED AT ARC ,4A4)
   56 FORMAT(37H0RUN TERMINATED DUE TO ERRORS IN DATA)
      END
C*************************************************************************
      SUBROUTINE PREDAT(KS)
      DIMENSION LC(9),KA(18,2),KQ(9)
      COMMON/KL/KL(1)/KC/KC(1)/KU/KU(1)/KX/KX(1)/NL/NL(1)
      COMMON/NN/NN(1)/NP/NP(1)/IJ/IJ(1)/IL/IL(1)/JL/JL(1)/JI/JI(1)
```

```
      COMMON              /M/M/N/N/LER/LER/KAT/KAT/KOR/KOR/KTER/KTER
      COMMON /MINE/MINE/LC/LC/KA/KA/IFIN/IFIN/KI/KI/KO/KO/KQ/KQ/K/K
       INTEGER PAUSE,SAVE,READY,CARDS,TAPE,SKIP,TRANSP,ARCS,END
      DATA PAUSE,SAVE,READY/4HPAUS,4HSAVE,4HREAD/
      DATA CARDS,TAPE,SKIP,TRANSP/4HCARD,4HTAPE,4HSKIP,4HTRAN/
      DATA ARCS/4HARCS/
      DATA END/4HEND /
      WRITE(KO,93)
      CALL ERASE(LC,9)
      KOR=KQ(3)
      KQ(7)=0
      KS=0
   21 READ(KI,90) (KA(I,1),I=1,18)
      WRITE(KO,91) (KA(I,1),I=1,18)
      IF(KA(1,1).EQ.PAUSE) GO TO 180
      IF(KA(1,1).EQ.SAVE)  GO TO 50
      IF(KA(1,1).EQ.READY) GO TO 100
      GO TO 21
C          END JOB
  180 IF(KQ(6).EQ.0) GO TO 182
      K2=KQ(2)
      WRITE(KO,98)
      END FILE K2
      GO TO 183
  182 WRITE(KO,99)
  183 KS=-1
      RETURN
C          SAVE
   50 KS=1
    8 RETURN
C          READY
  100 IEA=KQ(4)
      IEN=KQ(5)
      IEJL=MAX0(IEN+1,IEA-2*IEN-1)
      CALL ERASE(KL,IEA,KC,IEA,KU,IEA,KX,IEA,IJ,IEA,JI,IEA,
     1NL,IEN,NN,2*IEN,NP,IEN,IL,IEN+1,JL,IEJL)
```

```
      M=0                                                  0KF01099 00K 0145
      N=0                                                  0KF01100 00K 0146
      LER=0                                                0KF01101 00K 0147
      KQ(8)=1                                              0KF01110 00K 0148
    3 READ(KI,90) (KA(I,1),I=1,18)                         0KF01120 00K 0149
      WRITE(KO,91) (KA(I,1),I=1,18)                        0KF01130 00K 0150
      IF(KA(1,1).EQ.CARDS) GO TO 1                         0KF01140 00K 0151
      IF(KA(1,1).EQ.TAPE) GO TO 6                          0KF01150 00K 0152
      IF(KA(1,1).EQ.SKIP) GO TO 6                          0KF01160 00K 0153
      IF(KA(1,1).EQ.TRANSP) GO TO 14                       0KF01170 00K 0154
      GO TO 3                                              0KF01180 00K 0155
   14 KQ(8)=0                                              0KF01190 00K 0156
      GO TO 3                                              0KF01200 00K 0157
    6 IF(KQ(9).NE.0) GO TO 7                               0KF01210 00K 0158
      KQ(9)=1                                              0KF01220 00K 0159
      REWIND KCR                                           0KF01230 00K 0160
    7 IF(KA(1,1).EQ.TAPE) GO TO 4                          0KF01240 00K 0161
      GO TO 13                                             0KF01250 00K 0162
    1 KOR=KI                                               0KF01260 00K 0163
    4 READ(KOR,90) (KA(I,1),I=1,18)                        0KF01270 00K 0164
      WRITE(KO,91) (KA(I,1),I=1,18)                        0KF01280 00K 0165
      IF(KA(1,1).EQ.ARCS) GO TO 8                          0KF01290 00K 0166
C           TITLE                                          0KF01300 00K 0167
      DO 10 I=1,18                                         0KF01310 00K 0168
   10 KA(I,2)=KA(I,1)                                      0KF01320 00K 0169
      GO TO 4                                              0KF01330 00K 0170
   13 READ(KOR,92) KA(1,1)                                 0KF01340 00K 0171
      IF(KA(1,1).EQ.END) GO TO 3                           0KF01350 00K 0172
      GO TO 13                                             0KF01360 00K 0173
   90 FORMAT(18A4)                                         0KF01370 00K 0174
   91 FORMAT(1H018A4)                                      0KF01380 00K 0175
   92 FORMAT(A4)                                           0KF01390 00K 0176
   93 FORMAT(1H1)                                                   00K 0177
   98 FORMAT(31HORESERVED TAPE HAS BEEN WRITTEN///1H0)     0KF01400 00K 0178
   99 FORMAT(34HONO RESERVED TAPE HAS BEEN WRITTEN)        0KF01410 00K 0179
      END                                                  0KF01420 00K 0180
                                                                PAGE   5
```

```
C***************************************************************************************** 00K 0181
      SUBROUTINE ARCASY(LL)                                              0KF01460 00K 0182
      DIMENSION LC(9),KA(18,2),KQ(9)                                     0KF01490 00K 0183
      CIMENSION KE(2),KF(2),KD(2)                                        0KF01500 00K 0184
      COMMON/KL/KL(1)/KC/KC(1)/KU/KU(1)/KX/KX(1)/NL/NL(1)                         00K 0185
      COMMON/NN/NN(1)/NP/NP(1)/IJ/IJ(1)/IL/IL(1)/JL/JL(1)/JI/JI(1)                00K 0186
      CCMMON             /M/M/N/N/LER/LER/KAT/KAT/KOR/KOR/KTER/KTER               00K 0187
      COMMON /MINE/MINE/LC/LC/KA/KA/IFIN/IFIN/KI/KI/KO/KO/KQ/KQ/K/K      0KF01530 00K 0188
       INTEGER END,NODES,BLANK                                           0KF01540 00K 0189
      DATA END,NODES,BLANK/4HEND ,4HNODE,4H    /                         0KF01550 00K 0190
      LER=0                                                              0KF01560 00K 0191
      KF(1)=0                                                            0KF01570 00K 0192
      KF(2)=0                                                            0KF01580 00K 0193
      M=0                                                                0KF01590 00K 0194
      N=1                                                                0KF01600 00K 0195
    6 READ(KOR,90) KD(1),KD(2),KE(1),KE(2),IJ(2*N-1),IJ(2*N),(KA(I,1),I=0KF01610 00K 0196
     11,4)                                                               0KF01620 00K 0197
      IF(KD(1).EQ.END) GC TO 1                                           0KF01630 00K 0198
      IF(KD(1).EQ.NODES) GO TO 2                                         0KF01640 00K 0199
      IF(KD(1).EQ.BLANK) GO TO 3                                         0KF01650 00K 0200
      GO TO 4                                                            0KF01660 00K 0201
C            NO NCDES TO DO                                              0KF01670 00K 0202
    1 LL=0                                                               0KF01680 00K 0203
      GO TO 5                                                            0KF01690 00K 0204
C            NODES TC DO                                                 0KF01700 00K 0205
    2 LL=2                                                               0KF01710 00K 0206
      WRITE(KO,94) KD(1),KD(2)                                           0KF01720 00K 0207
    5 N=N-1                                                              0KF01730 00K 0208
   99 IF(LER.EQ.0) GO TO 101                                             0KF01740 00K 0209
      KQ(8)=2                                                            0KF01750 00K 0210
  101 RETURN                                                             0KF01760 00K 0211
C            ARC TO FILE                                                 0KF01770 00K 0212
    3 IF(KE(1).EQ.BLANK.AND.KE(2).EQ.BLANK) GO TO 6                      0KF01780 00K 0213
      KC(N)=KA(1,1)                                                      0KF01790 00K 0214
      KU(N)=KA(2,1)                                                      0KF01800 00K 0215
      KL(N)=KA(3,1)                                                      0KF01810 00K 0216
```

PAGE 6

```
      KX(N)=KA(4,1)
      IF(KE(1).EQ.KF(1).AND.KE(2).EQ.KF(2)) GO TO 9
      IF(NODE(KE(1),KE(2)).EQ.M+1) GO TO 11
      WRITE(KO,91) KE(1),KE(2),IJ(2*N-1),IJ(2*N)
      GO TO 12
   11 KF(1)=KE(1)
      KF(2)=KE(2)
      IF(M.GT.KQ(5)) GO TO 23
      M=M+1
      NN(2*M-1)=KE(1)
      NN(2*M)=KE(2)
      NL(M)=N
    9 IF(N.GT.KQ(4)) GO TO 20
      N=N+1
      GO TO 6
    4 WRITE(KO,92) N
   12 WRITE(KO,93) KD(1),KD(2),KE(1),KE(2),IJ(2*N-1),IJ(2*N),(KA(I,1),I=
     11,4)
      LER=LER+1
      GO TO 6
   20 WRITE(KO,89)
   25 LER=10000
      GO TO 99
   23 WRITE(KO,88)
      GO TO 25
   88 FORMAT(27H0TOO MANY NODES IN THIS RUN)
   89 FORMAT(26H0TOO MANY ARCS IN THIS RUN)
   90 FORMAT(3(A4,A2),2X,4I10)
   91 FORMAT(36H0SOURCE NODES ARE NOT ADJACENT, ARC 4A4)
   92 FORMAT(36H0CARD PUNCHING ERROR IN ARC CARD NO.,I6)
   93 FORMAT(1H03(A4,A2),2X,4I10)
   94 FORMAT(1H0A4,A2)
      END
C*****************************************************************************
      SUBROUTINE MAKEJL
      DIMENSION LC(9),KA(18,2),KQ(9)
```

```
      COMMON/KL/KL(1)/KC/KC(1)/KU/KU(1)/KX/KX(1)/NL/NL(1)                            00K 0253
      COMMON/NN/NN(1)/NP/NP(1)/IJ/ IJ(1)/IL/IL(1)/JL/JL(1)/JI/JI(1)                  00K 0254
      COMMON              /M/M/N/N/LER/LER/KAT/KAT/KOR/KOR/KTER/KTER                 00K 0255
      COMMON /MINE/MINE/LC/LC/KA/KA/IFIN/IFIN/KI/KI/KO/KO/KQ/KQ/K/K         0KF02240 00K 0256
C          NUMBERS TO IJ LIST                                               0KF02250 00K 0257
      I=1                                                                   0KF02260 00K 0258
      DO 1 L=1,N                                                            0KF02270 00K 0259
    3 K=NODENO(IJ(2*L-1),IJ(2*L))                                           0KF02280 00K 0260
      IF(K.LE.M) GO TO 6                                                    0KF02290 00K 0261
      IF(M.GE.KQ(5)) GO TO 9                                                0KF02300 00K 0262
      M=M+1                                                                 0KF02310 00K 0263
      NN(2*M-1)=IJ(2*L-1)                                                   0KF02320 00K 0264
      NN(2*M)=IJ(2*L)                                                       0KF02330 00K 0265
      IJ(L)=K                                                                        00K 0266
      NL(M)=N+1                                                             0KF02350 00K 0267
      LER=LER+1                                                             0KF02360 00K 0268
   19 IF(NL(I+1).GT.L) GO TO 18                                             0KF02370 00K 0269
      I=I+1                                                                 0KF02380 00K 0270
      IF(I.LT.M) GO TO 19                                                   0KF02390 00K 0271
   18 WRITE(KO,90) NN(2*I-1),NN(2*I),NN(2*M-1),NN(2*M)                      0KF02400 00K 0272
      GO TO 1                                                               0KF02410 00K 0273
    6 IJ(L)=K                                                                        00K 0274
    1 CONTINUE                                                              0KF02430 00K 0275
C          FIX IL LIST                                                      0KF02460 00K 0276
      DO 8 I=1,M                                                            0KF02470 00K 0277
      CALL PLACE(NL(I),IL(I))                                               0KF02480 00K 0278
    8 NL(I)=0                                                               0KF02490 00K 0279
      CALL PLACE(N+1,IL(M+1))                                               0KF02500 00K 0280
C          COUNT J-S                                                        0KF02510 00K 0281
      DO 10 J=1,N                                                           0KF02520 00K 0282
      I=LADDR(IJ(J))                                                        0KF02530 00K 0283
   10 NL(I)=NL(I)+1                                                         0KF02540 00K 0284
C          FORM JL LIST                                                     0KF02550 00K 0285
      KK=1                                                                  0KF02560 00K 0286
      CALL PLACE(KK,JL(1))                                                  0KF02570 00K 0287
      DO 20 I=1,M                                                           0KF02580 00K 0288
```

```
      IF(NL(I).NE.0) GO TO 23                                           0KF02590 00K 0289
      WRITE(KO,91) NN(2*I-1),NN(2*I)                                    0KF02600 00K 0290
      LER=1                                                             0KF02610 00K 0291
   23 KK=KK+NL(I)                                                       0KF02620 00K 0292
      NL(I)=LDECR(JL(I))                                                         00K 0293
   20 CALL PLACE(KK,JL(I+1))                                            0KF02630 00K 0294
      NL(M+1)=LDECR(JL(M+1))                                                     00K 0295
C         START OF JL LIST SEGMENT MOVED TO MAKEJL FROM TRANSL                   00K 0296
C    COMPUTE JI LISTS                                                   0KF04350 00K 0297
      I=0                                                               0KF04360 00K 0298
      LUP=1                                                             0KF04370 00K 0299
      DO 22 L=1,N                                                       0KF04380 00K 0300
      IF(L.LT.LUP) GO TO 25                                             0KF04390 00K 0301
      I=I+1                                                             0KF04400 00K 0302
      LUP=LDECR(IL(I+1))                                                0KF04410 00K 0303
   25 K=LADDR(IJ(L))                                                    0KF04420 00K 0304
      J=NL(K)                                                           0KF04430 00K 0305
      JI(J)=I                                                           0KF04440 00K 0306
      CALL PLACE(L,JI(J))                                               0KF04450 00K 0307
   22 NL(K)=NL(K)+1                                                     0KF04460 00K 0308
C           END OF JL LIST SEGMENT MOVED TO MAKEJL FROM TRANSL                   00K 0309
  100 RETURN                                                            0KF02640 00K 0310
    9 LER=100000                                                        0KF02650 00K 0311
      WRITE(KO,92)                                                      0KF02660 00K 0312
      GO TO 100                                                         0KF02670 00K 0313
   90 FORMAT(5HOARC ,4A4,18H IS A DEAD END ARC)                         0KF02680 00K 0314
   91 FORMAT(21HONO ARC ENDS AT NODE ,2A4)                              0KF02690 00K 0315
   92 FORMAT(27HOTOO MANY NODES IN THIS RUN)                            0KF02700 00K 0316
      END                                                               0KF02710 00K 0317
C*********************************************************************** 00K 0318
      SUBROUTINE NODASY                                                 0KF02750 00K 0319
      DIMENSION LC(9),KA(18,2),KQ(9)                                    0KF00780 00K 0320
      DIMENSION KE(2),KD(2)                                             0KF02790 00K 0321
      COMMON/KL/KL(1)/KC/KC(1)/KU/KU(1)/KX/KX(1)/NL/NL(1)                        00K 0322
      COMMON/NN/NN(1)/NP/NP(1)/IJ/IJ(1)/IL/IL(1)/JL/JL(1)/JI/JI(1)               00K 0323
      COMMON           /M/M/N/N/LER/LER/KAT/KAT/KOR/KOR/KTER/KTER                00K 0324
```

```
      COMMON /MINE/MINE/LC/LC/KA/KA/IFIN/IFIN/KI/KI/KO/KC/KQ/KQ/K/K      OKF02820 OOK 0325
       INTEGER END,BLANK                                                OKF02830 OOK 0326
      DATA END,BLANK/4HEND ,4H    /                                     OKF02840 OOK 0327
      I=0                                                               OKF02850 OOK 0328
    3 I=I+1                                                             OKF02860 OOK 0329
      READ(KOR,90) KD(1),KD(2),KE(1),KE(2),KA(1,1)                      OKF02870 OOK 0330
      IF(KD(1).EQ.END) GO TO 99                                         OKF02880 OOK 0331
      IF(KD(1).NE.BLANK) GO TO 2                                        OKF02890 OOK 0332
      IF(KE(1).EQ.BLANK) GO TO 3                                        OKF02900 OOK 0333
      K=NODENO(KE(1),KE(2))                                             OKF02910 OOK 0334
      IF(K.GT.M) GO TO 6                                                OKF02920 OOK 0335
      NP(K)=KA(1,1)                                                     OKF02930 OOK 0336
      GO TO 3                                                           OKF02940 OOK 0337
    6 WRITE(KO,91) I,KE(1),KE(2)                                        OKF02950 OOK 0338
   10 LER=LER+1                                                         OKF02960 OOK 0339
      GO TO 3                                                           OKF02970 OOK 0340
    2 WRITE (KO,92) I,KD(1),KD(2),KE(1),KE(2),KA(1,1)                   OKF02980 OOK 0341
      GO TO 10                                                          OKF02990 OOK 0342
   99 RETURN                                                            OKF03000 OOK 0343
   90 FORMAT(2(A4,A2),8X,I10)                                           OKF03010 OOK 0344
   91 FORMAT(5HOCARD I6,6H NODE A4,A2,12H NOT IN ARCS)                  OKF03020 OOK 0345
   92 FORMAT(37HOCARD PUNCHING ERROR IN NODE CARD NO.I6/1H 2(A4,A2),8X,IOKF03030 OOK 0346
     110)                                                               OKF03040 OOK 0347
      END                                                               OKF03050 OOK 0348
C*********************************************************************** OOK 0349
      SUBROUTINE READER                                                 OKF03090 OOK 0350
      DIMENSION LC(9),KA(18,2),KQ(9)                                    OKF03120 OOK 0351
      COMMON/KL/KL(1)/KC/KC(1)/KU/KU(1)/KX/KX(1)/NL/NL(1)                        OOK 0352
      COMMON/NN/NN(1)/NP/NP(1)/IJ/IJ(1)/IL/IL(1)/JL/JL(1)/JI/JI(1)               OOK 0353
      COMMON            /M/M/N/N/LER/LER/KAT/KAT/KOR/KOR/KTER/KTER               OOK 0354
      COMMON /MINE/MINE/LC/LC/KA/KA/IFIN/IFIN/KI/KI/KO/KC/KQ/KQ/K/K      OKF03150 OOK 0355
       INTEGER TAPE1,PUNCH1,NODES1,PRINT1                               OKF03160 OOK 0356
       INTEGER ALTER,OUTPUT,COMPUT,TAPE,PUNCH,NODES,PRINT               OKF03170 OOK 0357
      DATA TAPE1,PUNCH1,NODES1,PRINT1/4HAPE ,4HUNCH,4HODES,4HRINT/      OKF03180 OOK 0358
      DATA ALTER,OUTPUT,COMPUT/4HALTE,4HOUTP,4HCOMP/                    OKF03190 OOK 0359
      DATA TAPE,PUNCH,NODES,PRINT/4H TAP,4H PUN,4H NOD,4H PRI/          OKF03200 OOK 0360
                                                                           PAGE  10
```

```
    5 READ(KI,95) (KA(I,1),I=1,18)                    OKF03210 OOK 0361
      IF(KA(1,1).EQ.ALTER) GO TO 18                   OKF03220 OOK 0362
      IF(KA(1,1).EQ.OUTPUT) GO TO 119                 OKF03230 OOK 0363
      IF(KA(1,1).EQ.COMPUT) GO TO 18                  OKF03240 OOK 0364
      WRITE(KO,96) (KA(I,1),I=1,18)                   OKF03250 OOK 0365
      DO 15 I=1,18                                    OKF03260 OOK 0366
   15 KA(I,2)=KA(I,1)                                 OKF03270 OOK 0367
      GO TO 5                                         OKF03280 OOK 0368
   18 WRITE(KO,97)                                    OKF03290 OOK 0369
      WRITE(KO,96) (KA(I,1),I=1,18)                   OKF03300 OOK 0370
      LER=1                                           OKF03310 OOK 0371
      IF(KA(1,1).EQ.COMPUT) GO TO 111                 OKF03320 OOK 0372
   20 READ(KI,90) (KA(I,1),I=1,11)                    OKF03330 OOK 0373
      IF(KA(1,1).EQ.ALTER) GO TO 140                  OKF03340 OOK 0374
      IF(KA(1,1).EQ.OUTPUT) GO TO 121                 OKF03350 OOK 0375
      IF(KA(1,1).EQ.COMPUT) GO TO 111                 OKF03360 OOK 0376
      GO TO 200                                       OKF03370 OOK 0377
C          COMPUTE                                    OKF03380 OOK 0378
  111 WRITE(KO,93)  N,M,KQ(4),KQ(5)                   OKF03390 OOK 0379
  999 RETURN                                          OKF03400 OOK 0380
C          SET OUTPUT CONTROL                         OKF03410 OOK 0381
  119 WRITE(KO,96) (KA(I,1),I=1,18)                   OKF03420 OOK 0382
      L=5                                             OKF03430 OOK 0383
      IF(KA(3,1).EQ.TAPE1) L=1                        OKF03440 OOK 0384
      IF(KA(3,1).EQ.PUNCH1) L=2                       OKF03450 OOK 0385
      IF(KA(3,1).EQ.NODES1) L=3                       OKF03460 OOK 0386
      IF(KA(3,1).EQ.PRINT1) L=4                       OKF03470 OOK 0387
      GO TO 80                                        OKF03480 OOK 0388
  121 WRITE(KO,88) (KA(I,1),I=1,6)                    OKF03490 OOK 0389
  120 L=5                                             OKF03500 OOK 0390
      IF(KA(3,1).EQ.TAPE) L=1                         OKF03510 OOK 0391
      IF(KA(3,1).EQ.PUNCH) L=2                        OKF03520 OOK 0392
      IF(KA(3,1).EQ.NODES) L=3                        OKF03530 OOK 0393
      IF(KA(3,1).EQ.PRINT) L=4                        OKF03540 OOK 0394
   80 IF(L-4) 81,86,200                               OKF03550 OOK 0395
   81 LC(L)=1                                         OKF03560 OOK 0396
```

```
      GO TO 20                                                          0KF03570 00K 0397
   86 KQ(7)=1                                                           0KF03580 00K 0398
      GO TO 20                                                          0KF03590 00K 0399
C           ALTER                                                       0KF03600 00K 0400
  140 IF(KA(7,1).GT.0) GO TO 142                                        0KF03610 00K 0401
      KA(7,1)=1                                                         0KF03620 00K 0402
  142 WRITE(KO,91) (KA(I,1),I=1,11)                                     0KF03630 00K 0403
      N1=NODENO(KA(3,1),KA(4,1))                                        0KF03640 00K 0404
      N2=NODENO(KA(5,1),KA(6,1))                                        0KF03650 00K 0405
      IF(N1.GT.M) GO TO 144                                             0KF03660 00K 0406
      IF(N2.LE.M) GO TO 145                                             0KF03670 00K 0407
  144 WRITE(KO,92)                                                      0KF03680 00K 0408
      LER=1                                                             0KF03690 00K 0409
      GO TO 20                                                          0KF03700 00K 0410
  145 L1=LDECR(IL(N1))                                                  0KF03710 00K 0411
      L2=LDECR(IL(N1+1))-1                                              0KF03720 00K 0412
      IF(L2.LT.L1) GO TO 144                                            0KF03730 00K 0413
      DO 147 LL=L1,L2                                                   0KF03740 00K 0414
      IF(LADDR(IJ(LL)).NE.N2) GO TO 147                                 0KF03750 00K 0415
      KA(7,1)=KA(7,1)-1                                                 0KF03760 00K 0416
      IF(KA(7,1).EQ.0) GO TO 149                                        0KF03770 00K 0417
  147 CONTINUE                                                          0KF03780 00K 0418
      GO TO 144                                                         0KF03790 00K 0419
  149 KC(LL)=KA(8,1)                                                    0KF03800 00K 0420
      KU(LL)=KA(9,1)                                                    0KF03810 00K 0421
      KL(LL)=KA(10,1)                                                   0KF03820 00K 0422
      KX(LL)=KX(LL)+KA(11,1)                                            0KF03830 00K 0423
      GO TO 20                                                          0KF03840 00K 0424
C           CARD PUNCHING ERROR                                         0KF03850 00K 0425
  200 LER=1                                                             0KF03860 00K 0426
      WRITE(KO,87) (KA(I,1),I=1,6)                                      0KF03870 00K 0427
      GO TO 20                                                          0KF03880 00K 0428
   87 FORMAT(23H ILLEGAL CONTROL CARD (3(A4,A2),1H))                    0KF03890 00K 0429
   88 FORMAT(1H03(A4,A2))                                               0KF03900 00K 0430
   90 FORMAT(3(A4,A2),I2,4I10)                                          0KF03910 00K 0431
   91 FORMAT(1H0,3(A4,A2),I2,4I10)                                      0KF03920 00K 0432
```

```
   92 FORMAT(47HOTHE ARC ON THE ABOVE ALTER CARD IS NOT IN CORE)      OKF03930 OOK 0433
   93 FORMAT(12HONO.OF ARCS=I5,2X,13H NO.OF NODES=,I5,6X,'(MAXIMUMS FOR        OOK 0434
     $THIS VERSION : ',I5,' ARCS AND ',I5,' NODES)')                           OOK 0435
   95 FORMAT(18A4)                                                    OKF03950 OOK 0436
   96 FORMAT(1H018A4)                                                 OKF03960 OOK 0437
   97 FORMAT(47HOOUTPUT CONTROL CARD MISSING OR OUT OF SEQUENCE)      OKF03970 OOK 0438
      END                                                             OKF03980 OOK 0439
C*************************************************************************** OOK 0440
      SUBROUTINE TRANSL                                               OKF04020 OOK 0441
      DIMENSION LC(9),KA(18,2),KQ(9)                                  OKF04050 OOK 0442
      COMMON/KL/KL(1)/KC/KC(1)/KU/KU(1)/KX/KX(1)/NL/NL(1)                      OOK 0443
      COMMON/NN/NN(1)/NP/NP(1)/IJ/IJ(1)/IL/IL(1)/JL/JL(1)/JI/JI(1)             OOK 0444
      COMMON              /M/M/N/N/LER/LER/KAT/KAT/KOR/KOR/KTER/KTER           OOK 0445
      COMMON /MINE/MINE/LC/LC/KA/KA/IFIN/IFIN/KI/KI/KO/KO/KQ/KQ/K/K   OKF04080 OOK 0446
C     CLEAR NL STORAGE                                                OKF04090 OOK 0447
      CALL ERASE(NL,M)                                                         OOK 0448
C     CALCULATE CIRCULATION AND C-BAR                                 OKF04120 OOK 0449
      I=0                                                             OKF04130 OOK 0450
      LUP=1                                                           OKF04140 OOK 0451
      DO 2 L=1,N                                                      OKF04150 OOK 0452
      IF(L.LT.LUP) GO TO 13                                           OKF04160 OOK 0453
      I=I+1                                                           OKF04170 OOK 0454
      LUP=LDECR(IL(I+1))                                              OKF04180 OOK 0455
   13 LU=LADDR(IJ(L))                                                 OKF04190 OOK 0456
      NL(I)=NL(I)-KX(L)                                               OKF04200 OOK 0457
      NL(LU)=NL(LU)+KX(L)                                             OKF04210 OOK 0458
    2 KC(L)=KC(L)+NP(I)-NP(LU)                                        OKF04220 OOK 0459
C   CIRCULATION MESSAGE FOR NON-ZERO CIRCULATION AND MOVE JL LIST     OKF04230 OOK 0460
C     CLEAR NL STORAGE                                                OKF04470 OOK 0461
      DO 5 I=1,M                                                      OKF04240 OOK 0462
      IF(NL(I).EQ.0) GO TO 5                                          OKF04250 OOK 0463
      WRITE(KO,90) NN(2*I-1),NN(2*I),NL(I)                            OKF04260 OOK 0464
      NL(I)=0                                                                  OOK 0465
    5 CONTINUE                                                                 OOK 0466
C   COMPUTE EXCESS OF X AND UPPER BOUND OVER LOWER BOUND              OKF04280 OOK 0467
      DO 1 J=1,N                                                      OKF04290 OOK 0468
```

```
      KU(J)=KU(J)-KL(J)                                                 OKF04300 00K 0469
      IF(KU(J).GE.0) GO TO 1                                            OKF04310 00K 0470
      WRITE(KO,51) J                                                    OKF04320 00K 0471
      LER=LER+1                                                         OKF04330 00K 0472
    1 KX(J)=KX(J)-KL(J)                                                 OKF04340 00K 0473
C        JL LIST SEGMENT MOVED FROM TRANSL TO MAKEJL                             00K 0474
      RETURN                                                            OKF04500 00K 0475
   51 FORMAT(4H0ARC,I6,42H HAS LOWER BOUND GREATER THAN UPPER BOUND.)   OKF04510 00K 0476
   90 FORMAT(6H0NODE 2A4,28H NON-CONSERVATIVE, NET FLOW=I12)            OKF04520 00K 0477
      END                                                               OKF04530 00K 0478
C************************************************************************ 00K 0479
      SUBROUTINE KILTER(I)                                              OKF04570 00K 0480
      DIMENSION LC(9),KA(18,2),KQ(9)                                    OKF04600 00K 0481
      COMMON/KL/KL(1)/KC/KC(1)/KU/KU(1)/KX/KX(1)/NL/NL(1)                        00K 0482
      COMMON/NN/NN(1)/NP/NP(1)/IJ/IJ(1)/IL/IL(1)/JL/JL(1)/JI/JI(1)               00K 0483
      COMMON              /M/M/N/N/LER/LER/KAT/KAT/KOR/KOR/KTER/KTER             00K 0484
      COMMON /MINE/MINE/LC/LC/KA/KA/IFIN/IFIN/KI/KI/KO/KO/KQ/KQ/K/K     OKF04630 00K 0485
      IF(LDECR(IJ(1)).EQ.0) GO TO 70                                    OKF04640 00K 0486
      CALL PLACE(0,IJ(1))                                               OKF04650 00K 0487
      CALL ERASE(NL,M)                                                           00K 0488
   70 LER=0                                                             OKF04680 00K 0489
    5 IF(KC(K)) 10,20,30                                                OKF04690 00K 0490
   10 IF(KX(K)-KU(K)) 13,40,14                                          OKF04700 00K 0491
   13 MINE=-KX(K)+KU(K)                                                 OKF04710 00K 0492
      GO TO 50                                                          OKF04720 00K 0493
   14 MINE=KX(K)-KU(K)                                                  OKF04730 00K 0494
      GO TO 60                                                          OKF04740 00K 0495
   20 IF(KX(K).LT.0) GO TO 13                                           OKF04750 00K 0496
      IF(KX(K)-KU(K)) 40,40,35                                          OKF04760 00K 0497
   30 IF(KX(K)) 32,40,35                                                OKF04770 00K 0498
   32 MINE=-KX(K)                                                       OKF04780 00K 0499
      GO TO 50                                                          OKF04790 00K 0500
   35 MINE=KX(K)                                                        OKF04800 00K 0501
      GO TO 60                                                          OKF04810 00K 0502
   50 KOR=LADDR(IJ(K))                                                  OKF04820 00K 0503
      KAT=0                                                             OKF04830 00K 0504
```

```
      KTER=I                                                            0KF04840
      GO TO 65                                                          0KF04850
   60 KOR=I                                                             0KF04860
      KAT=10                                                            0KF04870
      KTER=LADDR(IJ(K))                                                 0KF04880
   65 CALL LABELN(KBR)                                                  0KF04890
      IF(KBR.EQ.0) GO TO 68                                             0KF04900
      CALL BREAKT                                                       0KF04910
      GO TO 5                                                           0KF04920
   68 CALL UPNOPR                                                       0KF04930
      IF(LER.EQ.0) GO TO 5                                              0KF04940
   40 RETURN                                                            0KF04950
      END                                                               0KF04960
C***********************************************************************
      SUBROUTINE OUTPUT(KZ)                                             0KF05000
      DIMENSION LC(9),KA(18,2),KQ(9)                                    0KF05030
      DIMENSION KZ(101)                                                 0KF05040
      COMMON/KL/KL(1)/KC/KC(1)/KU/KU(1)/KX/KX(1)/NL/NL(1)
      COMMON/NN/NN(1)/NP/NP(1)/IJ/IJ(1)/IL/IL(1)/JL/JL(1)/JI/JI(1)
      COMMON            /M/M/N/N/LER/LER/KAT/KAT/KOR/KOR/KTER/KTER
      COMMON /MINE/MINE/LC/LC/KA/KA/IFIN/IFIN/KI/KI/KO/KO/KQ/KQ/K/K     0KF05070
      DATA KILT,BLANK,IEN/1HK,1H ,1HN/                                  0KF05080
      INTEGER OUT(9)
      LOGICAL OUTTAP,OUTPRT,OUTPCH
      DOUBLE PRECISION KCUM
      KQ1=KQ(1)                                                         0KF05090
      KCUM=0                                                            0KF05100
      IF(KZ(101).NE.0) GO TO 10                                         0KF05110
      IF(LER.NE.0) GO TO 30                                             0KF05120
      MZ=KILT                                                           0KF05130
      GO TO 100                                                         0KF05140
   10 IF(LER.NE.0) GO TO 18                                             0KF05150
      WRITE(KO,99) KZ(101)                                              0KF05160
   18 KZ(101)=MIN0(KZ(101),100)                                         0KF05170
   30 MZ=BLANK                                                          0KF05180
  100 K2=KQ(2)                                                          0KF05190
```

```
      IF(LC(2).EQ.0) GO TO 12                                             OKF05200 OOK 0541
      WRITE(KQ1,90) (KA(I,2),I=1,18)                                      OKF05210 OOK 0542
      WRITE(KO,89)                                                        OKF05220 OOK 0543
   12 IF(LC(1).EQ.0) GO TO 41                                             OKF05230 OOK 0544
      IF(KQ(6).NE.0) GO TO 24                                             OKF05240 OOK 0545
      KQ(6)=1                                                             OKF05250 OOK 0546
      REWIND K2                                                           OKF05260 OOK 0547
   24 WRITE(K2,90) (KA(I,2),I=1,18)                                       OKF05270 OOK 0548
      WRITE(KO,88)                                                        OKF05280 OOK 0549
   41 IF(KQ(7).EQ.0) GO TO 7                                              OKF05290 OOK 0550
      WRITE(KO,91) (KA(I,2),I=1,18)                                       OKF05300 OOK 0551
    7 L=1                                                                 OKF05310 OOK 0552
      LL=1                                                                OKF05320 OOK 0553
      CUTTAP=LC(1).NE.0                                                            OOK 0554
      CUTPRT=KQ(7).NE.0                                                            OOK 0555
      CUTPCH=LC(2).NE.0                                                            OOK 0556
      DO 3 I=1,M                                                          OKF05330 OOK 0557
      LUP=LDECR(IL(I+1))                                                  OKF05340 OOK 0558
  302 IF(L.GE.LUP) GO TO 3                                                OKF05350 OOK 0559
      LU=LADDR(IJ(L))                                                     OKF05360 OOK 0560
      LLC=KC(L)                                                           OKF05370 OOK 0561
      KC(L)=KC(L)-NP(I)+NP(LU)                                            OKF05380 OOK 0562
      KX(L)=KX(L)+KL(L)                                                   OKF05390 OOK 0563
      KU(L)=KU(L)+KL(L)                                                   OKF05400 OOK 0564
      LZ=KX(L)*KC(L)                                                      OKF05410 OOK 0565
      KCUM=KCUM+LZ                                                        OKF05420 OOK 0566
      MX=MZ                                                               OKF05430 OOK 0567
      II=I+I                                                                       OOK 0568
      LU2=LU+LU                                                                    OOK 0569
      CUT(1)=NN(II-1)                                                              OOK 0570
      CUT(2)=NN(II)                                                                OOK 0571
      CUT(3)=NN(LU2-1)                                                             OOK 0572
      CUT(4)=NN(LU2)                                                               OOK 0573
      OUT(5)=KC(L)                                                                 OOK 0574
      CUT(6)=KU(L)                                                                 OOK 0575
      CUT(7)=KL(L)                                                                 OOK 0576
```

```
      OUT(8)=KX(L)
      OUT(9)=LZ
      IF(KZ(101).LE.0) GO TO 16
      IF(KZ(LL).NE.L) GO TO 16
      KZ(101)=KZ(101)-1
      LL=LL+1
      MX=IEN
   16 IF(OUTTAP) WRITE(K2,93) OUT,MX
      IF(OUTPRT) WRITE(KC,94) OUT,NP(I),NP(LU),LLC,MX
      IF(OUTPCH) WRITE(KQ1,93) OUT
  333 L=L+1
      GO TO 302
    3 CONTINUE
      IF(LC(3).EQ.0) GO TO 15
      IF(LC(1)+LC(2).EQ.0) GO TO 27
      IF(LC(1).EQ.0) GO TO 203
      WRITE(K2,96)
  203 IF(LC(2).EQ.0) GO TO 115
      WRITE(KQ1,96)
  115 DO 200 I=1,M
      IF(LC(1).EQ.0) GO TO 85
      WRITE(K2,95) NN(2*I-1),NN(2*I),NP(I)
   85 IF(LC(2).EQ.0) GO TO 200
      WRITE(KQ1,95) NN(2*I-1),NN(2*I),NP(I)
  200 CONTINUE
   15 IF(LC(1).EQ.0) GO TO 27
      WRITE(K2,97)
   27 IF(KQ(7).EQ.0) GO TO 57
      WRITE(KO,98)
      WRITE(KO,999) KCUM
   57 IF(LC(2).EQ.0) GO TO 77
      WRITE(KQ1,97)
   77 WRITE(KO,92)  (LC(I),I=5,8)
      RETURN
   88 FORMAT(24H0THIS RUN OUTPUT TO TAPE )
   89 FORMAT(25H0THIS RUN OUTPUT TO PUNCH)
```

```
   90 FORMAT(18A4/4HARCS22X,4HCOST5X,5HUPPER5X,5HLCWER9X,1HX8X,4HFLOW)  0KF05840 00K 0613
   91 FORMAT(1H118A4/5H ARCS16X,4HCOST6X,5HUPPER6X,5HLOWER10X,1HX8X,    0KF05850 00K 0614
     1 4HFLOW9X,3HPI19X,3HPI28X,4HCBAR/1X)                              0KF05860 00K 0615
   92 FORMAT(18HONO OF BREAKTHRUS=I12,22H, NO CF NCNBREAKTHRUS=I12,18H, 0KF05870 00K 0616
     1NO OF X CHANGES=I12,/42H NO OF NODES FROM WHICH LABELING WAS DONE=0KF05880 00K 0617
     2I12)                                                              0KF05890 00K 0618
   93 FORMAT(6X,2(A4,A2),2X,4I10,I12,1X,A1)                             0KF05900 00K 0619
   94 FORMAT(2(1X,A4,A2),4(1X,I10),4I12,1X,A1)                          0KF05910 00K 0620
   95 FORMAT(6X,A4,A2,6X,I12)                                           0KF05920 00K 0621
   96 FORMAT(6HNODES )                                                  0KF05930 00K 0622
   97 FORMAT(3HEND)                                                     0KF05940 00K 0623
   98 FORMAT(4H0END)                                                    0KF05950 00K 0624
   99 FORMAT(1H0I5,23H ARCS ARE CUT OF KILTER)                          0KF05960 00K 0625
  999 FORMAT(29H0TOTAL SYSTEM CONTRIBUTION =  F20.0)                             00K 0626
      ENC                                                               0KF05980 00K 0627
C*************************************************************************** 00K 0628
      SUBROUTINE CMPTIM(LV,ENT)                                                  00K 0629
C     FRINTS TIME OF INTRA-SUBROUTINE TRANSFERS OR DATE&TIME                     00K 0630
C     "TIMING" IS AN M.I.T. INTERNAL SUBROUTINE THAT RETURNS THE CPU TIME        00K 0631
C     IN HUNDREDTHS OF SECCNDS.                                                  00K 0632
C     "WHEN" IS AN M.I.T. INTERNAL SUBROUTINE THAT RETURNS THE DATE AND          00K 0633
C     TIME IN THE FOLLOWING 5A4 FORMAT:  MM/DD/YY HR*MI*SS.FF                    00K 0634
      DIMENSION A(5)                                                             00K 0635
      DOUBLE PRECISION LV,ENT                                                    00K 0636
      INTEGER  TNOW,TSTART,TREL                                                  00K 0637
      INTEGER WT                                                                 00K 0638
      CALL TIMING(TNOW)                                                          00K 0639
      TREL=TNOW-TSTART                                                           00K 0640
      IF(TREL.LT.0)   TREL=TREL+8640000                                          00K 0641
      TI=TREL/100.                                                               00K 0642
      WRITE(WT,10)LV,ENT,TI                                                      00K 0643
      RETURN                                                                     00K 0644
      ENTRY STRTIM(WT)                                                           00K 0645
      CALL TIMING(TSTART)                                                        00K 0646
      CALL WHEN(A)                                                               00K 0647
      WRITE(WT,20) A                                                             00K 0648
```

```
      RETURN
   10 FORMAT(/,T103,29('*'),/,T103,'* LV.   ',A6,T131,'*',/,
     $T103,'* ENT. ',A6,' @',F7.2,' SEC. *',/,T103,29('*'),/)
   20 FORMAT(/T103,29('*')/T103,'*    DATE =    ',2A4,T131,'*'/
     $T103,'*    TIME = ',  3A4,T131,'*'/T103,29('*')/)
      END
//STEP EXEC ASMC,PARM.C='LOAD,DECK'
//C.SYSIN DD   *
ASSEM1 START 0
       ENTRY LABELN,BREAKT,UFNCPR,NODENO
       SPACE 5
LABELN SAVE  (14,12),,*             SUBROUTINE LABELN(KBR)
       BALR  12,0
       USING *,12                        (R12 IS BASE FOR THIS PROGRAM)
       LA    11,SAVER
       ST    13,4(0,11)
       ST    11,8(0,13)
       L     11,IJAD
       S     11,FOUR                     (R11 HAS ADDRESS OF IJ-4)
       L     10,NLAD
       S     10,FOUR                     (R10 HAS ADDRESS OF NL-4)
       L     13,KCAD
       S     13,FOUR                     (R13 HAS ADDRESS OF KC-4)
       L     14,KXAD
       S     14,FOUR                     (R14 HAS ADDRESS OF KX-4)
       L     15,KUAD
       S     15,FOUR                     (R15 HAS ADDRESS OF KU-4)
       L     1,0(0,1)
       SR    2,2
       ST    2,0(0,1)               KBR=0
       ST    1,SAVER
       L     1,JIAD
       S     1,FOUR                      (R1 HAS ADDRESS OF JI-4)
       L     2,KORAD
       L     2,0(0,2)
       ST    2,I                    I=KOR
```

```
         SLL    2,2                    (R2 HAS I*4)                      OKF06300 OOK 0685
         L      3,EIGHT            NU=2                                  OKF06310 OOK 0686
         L      4,NUP                  (R3 HAS NU*4)                     OKF06320 OOK 0687
         SR     7,7                    (R4 HAS NUP*4)                    OKF06330 OOK 0688
         CH     7,4(0,11)          IF(LDECR(IJ(1)).NE.0) GO TO 14        OKF06340 OOK 0689
         BNE    L14                                                      OKF06350 OOK 0690
         L      7,IFINAD                                                 OKF06360 OOK 0691
         L      7,0(0,7)                                                 OKF06370 OOK 0692
         STH    7,2(2,10)          NL(I)=IFIN                            OKF06380 OOK 0693
         L      7,I                                                      OKF06390 OOK 0694
         STH    7,4(0,11)          CALL PLACE(I,IJ(1))                   OKF06400 OOK 0695
         L      4,EIGHT            NUP=2                                 OKF06410 OOK 0696
L14      L      9,ILAD                                                   OKF06420 OOK 0697
         S      9,FOUR                 (R9 HAS ADDRESS OF IL-4)          OKF06430 OOK 0698
         LH     5,4(2,9)        14 L2=LDECR(IL(I+1))-1                   OKF06440 OOK 0699
         BCTR   5,0                                                      OKF06450 OOK 0700
         SLL    5,2                    (R5 HAS L2*4)                     OKF06460 OOK 0701
         LH     6,0(2,9)               (R6 HAS L*4)                      OKF06470 OOK 0702
         SLL    6,2                L=LDECR(IL(I))                        OKF06480 OOK 0703
L16      CR     5,6             16 IF(L2.LT.L) GO TO 28                  OKF06490 OOK 0704
         BL     L28                                                      OKF06500 OOK 0705
         LH     8,2(6,11)          J=LADDR(IJ(L))                        OKF06510 OOK 0706
         SLL    8,2                    (R8 HAS J*4)                      OKF06520 OOK 0707
         SR     7,7                                                      OKF06530 OOK 0708
         C      7,0(8,10)          IF(NL(J).NE.0) GO TO 27               OKF06540 OOK 0709
         BNE    L27                                                      OKF06550 OOK 0710
         C      7,0(6,13)          IF(KC(L).GT.0) GO TO 21               OKF06560 OOK 0711
         BL     L21                                                      OKF06570 OOK 0712
         L      7,0(6,14)                                                OKF06580 OOK 0713
         C      7,0(6,15)          IF(KX(L)-KU(L)) 22,27,27              OKF06590 OOK 0714
         BL     L22                                                      OKF06600 OOK 0715
         B      L27                                                      OKF06610 OOK 0716
L21      SR     7,7                                                      OKF06620 OOK 0717
         C      7,0(6,14)       21 IF(KX(L).GE.0) GO TO 27               OKF06630 OOK 0718
         BNH    L27                                                      OKF06640 OOK 0719
L22      L      7,I                                                      OKF06650 OOK 0720
```

```
         ST     7,0(8,10)          22 NL(J)=I                                    OKF06660 OOK 0721
         SRL    6,2                                                              OKF06670 OOK 0722
         STH    6,0(8,10)             CALL PLACE(L,NL(J))                        OKF06680 OOK 0723
         SLL    6,2                                                              OKF06690 OOK 0724
         SRL    8,2                                                              OKF06700 OOK 0725
         STH    8,0(4,11)             CALL PLACE(J,IJ(NUP))                      OKF06710 OOK 0726
         A      4,FOUR                NUP=NUP+1                                  OKF06720 OOK 0727
         L      7,KTERAD                                                         OKF06730 OOK 0728
         L      7,0(0,7)                                                         OKF06740 OOK 0729
         CR     7,8                   IF(J.EQ.KTER) GO TO 47                     OKF06750 OOK 0730
         BE     L47                                                              OKF06760 OOK 0731
L27      A      6,FOUR             27 L=L+1                                      OKF06770 OOK 0732
         B      L16                   GO TO 16                                   OKF06780 OOK 0733
L28      L      9,JLAD                                                           OKF06790 OOK 0734
         S      9,FOUR                     (R9 HAS ADDRESS OF JL-4)              OKF06800 OOK 0735
         LH     5,4(2,9)           28 L2=LDECR(JL(I+1))-1                        OKF06810 OOK 0736
         BCTR   5,0                                                              OKF06820 OOK 0737
         SLL    5,2                                                              OKF06830 OOK 0738
         LH     6,0(2,9)              L=LDECR(JL(I))                             OKF06840 OOK 0739
         SLL    6,2                                                              OKF06850 OOK 0740
L30      CR     5,6                30 IF(L2.LT.L) GO TO 43                       OKF06860 OOK 0741
         BL     L43                                                              OKF06870 OOK 0742
         LH     8,2(6,1)              J=LADDR(JI(L))                             OKF06880 OOK 0743
         SLL    8,2                                                              OKF06890 OOK 0744
         SR     7,7                                                              OKF06900 OOK 0745
         C      7,0(8,10)             IF(NL(J)).NE.0) GO TO 42                   OKF06910 OOK 0746
         BNE    L42                                                              OKF06920 OOK 0747
         LH     9,0(6,1)              KR=LDECR(JI(L))                            OKF06930 OOK 0748
         SLL    9,2                        (R9 HAS KR*4)                         OKF06940 OOK 0749
         C      7,0(9,13)             IF(KC(KR).GE.0) GO TO 36                   OKF06950 OOK 0750
         BNH    L36                                                              OKF06960 OOK 0751
         L      7,0(9,14)                                                        OKF06970 OOK 0752
         C      7,0(9,15)             IF(KX(KR)-KU(KR)) 42,42,37                 OKF06980 OOK 0753
         BH     L37                                                              OKF06990 OOK 0754
         B      L42                                                              OKF07000 OOK 0755
L36      SR     7,7                                                              OKF07010 OOK 0756
```

```
         C      7,0(9,14)           36 IF(KX(KR).LE.0) GO TO 42          OKF07020 OOK 0757
         BNL    L42                                                      OKF07030 OOK 0758
L37      L      7,I                                                      OKF07040 OOK 0759
         LCR    7,7                 37 NL(J)=-I                          OKF07050 OOK 0760
         ST     7,0(8,10)                                                OKF07060 OOK 0761
         SRL    9,2                                                      OKF07070 OOK 0762
         STH    9,0(8,10)              CALL PLACE(KR,NL(J))              OKF07080 OOK 0763
         SRL    8,2                                                      OKF07090 OOK 0764
         STH    8,0(4,11)              CALL PLACE(J,IJ(NUP))             OKF07100 OOK 0765
         A      4,FOUR                 NUP=NUP+1                         OKF07110 OOK 0766
         L      7,KTERAD                                                 OKF07120 OOK 0767
         L      7,0(0,7)                                                 OKF07130 OOK 0768
         CR     7,8                    IF( J.EQ.KTER) GO TO 47           OKF07140 OOK 0769
         BE     L47                                                      OKF07150 OOK 0770
L42      A      6,FOUR              42 L=L+1                             OKF07160 OOK 0771
         B      L30                    GO TO 30                          OKF07170 OOK 0772
L43      CR     3,4                 43 IF(NU.GE.NUP) GO TO 48            OKF07180 OOK 0773
         BNL    L48                                                      OKF07190 OOK 0774
         LH     2,0(3,11)              I=LDECR(IJ(NU))                   OKF07200 OOK 0775
         A      3,FOUR                 NU=NU+1                           OKF07210 OOK 0776
         ST     2,I                                                      OKF07220 OOK 0777
         SLL    2,2                                                      OKF07230 OOK 0778
         B      L14                    GO TO 14                          OKF07240 OOK 0779
L47      L      1,SAVER                                                  OKF07250 OOK 0780
         L      7,ONE                                                    OKF07260 OOK 0781
         ST     7,0(0,1)            47 KBR=1                             OKF07270 OOK 0782
L48      SRL    3,2                                                      OKF07280 OOK 0783
         BCTR   3,0                 48 NU=NU-1                           OKF07290 OOK 0784
         L      7,LCAD                                                   OKF07300 OOK 0785
         A      3,28(0,7)              LC(8)=LC(8)+NU                    OKF07310 OOK 0786
         ST     3,28(0,7)                                                OKF07320 OOK 0787
         ST     4,NUP                                                    OKF07330 OOK 0788
         L      13,SAVER+4             RETURN                            OKF07340 OOK 0789
         RETURN (14,12),T              END                               OKF07350 OOK 0790
         EJECT                                                           OKF07360 OOK 0791
BREAKT   SAVE   (14,12),,*             SUBROUTINE BREAKT                 OKF07370 OOK 0792
```

```
         BALR  12,0                                                      0KF07380 00K 0793
         USING *,12                    (R12 IS BASE FOR THIS PROGRAM)    0KF07390 00K 0794
         LA    11,SAVER                                                  0KF07400 00K 0795
         ST    13,4(0,11)                                                0KF07410 00K 0796
         ST    11,8(0,13)                                                0KF07420 00K 0797
         L     10,NLAD                                                   0KF07430 00K 0798
         S     10,FOUR                 (R10 HAS ADDRESS OF NL-4)         0KF07440 00K 0799
         L     11,IJAD                                                   0KF07450 00K 0800
         S     11,FOUR                 (R11 HAS ADDRESS OF IJ-4)         0KF07460 00K 0801
         L     13,KCAD                                                   0KF07470 00K 0802
         S     13,FOUR                 (R13 HAS ADDRESS OF KC-4)         0KF07480 00K 0803
         L     14,KXAD                                                   0KF07490 00K 0804
         S     14,FOUR                 (R14 HAS ADDRESS OF KX-4)         0KF07500 00K 0805
         L     15,KUAD                                                   0KF07510 00K 0806
         S     15,FOUR                 (R15 HAS ADDRESS OF KU-4)         0KF07520 00K 0807
         L     8,MINEAD                                                  0KF07530 00K 0808
         L     8,0(0,8)                (R8 HAS MINE)                     0KF07540 00K 0809
         L     7,LCAD                                                    0KF07550 00K 0810
         L     9,16(0,7)                                                 0KF07560 00K 0811
         A     9,ONE               LC(5)=LC(5)+1                         0KF07570 00K 0812
         ST    9,16(0,7)                                                 0KF07580 00K 0813
         L     1,LERAD                                                   0KF07590 00K 0814
         SR    7,7                 LER=0                                 0KF07600 00K 0815
         ST    7,0(0,1)                                                  0KF07610 00K 0816
         L     4,KTERAD                                                  0KF07620 00K 0817
         L     4,0(0,4)            KT=KTER                               0KF07630 00K 0818
         SLL   4,2                     (R4 HAS KT*4)                     0KF07640 00K 0819
         L     2,FOUR                  (R2 HAS 4)                        0KF07650 00K 0820
         LR    1,2                     (R1 HAS J*4)                      0KF07660 00K 0821
         L     3,MAD                   (R3 HAS M*4)                      0KF07670 00K 0822
         L     3,0(0,3)            DO 29 J=1,M                           0KF07680 00K 0823
         SLL   3,2                                                       0KF07690 00K 0824
B11      LH    5,2(4,10)           KP=LADDR(NL(KT))                      0KF07700 00K 0825
         LH    6,0(4,10)           KK=LDECR(NL(KT))                      0KF07710 00K 0826
         SLL   6,2                     (R5 HAS KP)                       0KF07720 00K 0827
         L     7,IFINAD                (R6 HAS KK*4)                     0KF07730 00K 0828
                                                                             PAGE  23
```

```
          LH     7,2(0,7)                                                 OKF07740 OOK 0829
          CR     5,7                IF(KP.EQ.IF IN) GC TO 31              OKF07750 OOK 0830
          BE     B31                                                      OKF07760 OOK 0831
          LTR    5,5                IF(KP.GT.0) GO TO 23                  OKF07770 OOK 0832
          BP     B23                                                      OKF07780 OOK 0833
          SR     7,7                                                      OKF07790 OOK 0834
          C      7,0(6,13)          IF(KC(KK).GE.0) GO TO 19              OKF07800 OOK 0835
          BNH    B19                                                      OKF07810 OOK 0836
          L      7,0(6,14)                                                OKF07820 OOK 0837
          S      7,0(6,15)                                                OKF07830 OOK 0838
          CR     8,7                MINE=MINO(MINE,KX(KK)-KU(KK))         OKF07840 OOK 0839
          BNH    B21                                                      OKF07850 OOK 0840
          LR     8,7                                                      OKF07860 OOK 0841
          B      B21                GO TO 21                              OKF07870 OOK 0842
B19       L      7,0(6,14)                                                OKF07880 OOK 0843
          CR     8,7             19 MINE=MINO(MINE,KX(KK))                OKF07890 OOK 0844
          BNH    B21                                                      OKF07900 OOK 0845
          LR     8,7                                                      OKF07910 OOK 0846
B21       SRL    6,2                                                      OKF07920 OOK 0847
          LCR    6,6             21 KK=-KK                                OKF07930 OOK 0848
          STH    6,0(1,11)          CALL PLACE(KK,IJ(J))                  OKF07940 OOK 0849
          B      B29                GO TO 29                              OKF07950 OOK 0850
B23       SR     7,7                                                      OKF07960 OOK 0851
          C      7,0(6,13)       23 IF(KC(KK).GT.0) GO TO 26              OKF07970 OOK 0852
          BL     B26                                                      OKF07980 OOK 0853
          L      7,0(6,15)                                                OKF07990 OOK 0854
          S      7,0(6,14)                                                OKF08000 OOK 0855
          CR     8,7                MINE=MINO(MINE,KU(KK)-KX(KK))         OKF08010 OOK 0856
          BNH    B28                                                      OKF08020 OOK 0857
          LR     8,7                                                      OKF08030 OOK 0858
          B      B28                GO TO 28                              OKF08040 OOK 0859
B26       S      7,0(6,14)                                                OKF08050 OOK 0860
          CR     8,7             26 MINE=MINO(MINE,-KX(KK))               OKF08060 OOK 0861
          BNH    B28                                                      OKF08070 OOK 0862
          LR     8,7                                                      OKF08080 OOK 0863
B28       SRL    6,2                                                      OKF08090 OOK 0864
                                                                             PAGE  24
```

```
         STH   6,0(1,11)       28 CALL PLACE(KK,IJ(J))          OKF08100 OOK 0865
B29      LPR   4,5             29 KT=IABS(KP)                   OKF08110 OOK 0866
         SLL   4,2                                              OKF08120 OOK 0867
         BXLE  1,2,B11                                          OKF08130 OOK 0868
B31      L     9,KAD                                            OKF08140 OOK 0869
         L     9,0(0,9)                                         OKF08150 OOK 0870
         SLL   9,2                     (R9 HAS K*4)             OKF08160 OOK 0871
         L     7,0(9,14)                                        OKF08170 OOK 0872
         L     5,KATAD                                          OKF08180 OOK 0873
         L     5,0(0,5)                (R5 HAS KAT)             OKF08190 OOK 0874
         S     5,FOUR          31 IF(KAT.GT.4) GO TO 34         OKF08200 OOK 0875
         BP    B34                                              OKF08210 OOK 0876
         AR    7,8                                              OKF08220 OOK 0877
         ST    7,0(9,14)          KX(K)=KX(K)+MINE              OKF08230 OOK 0878
         B     B35                GO TO 35                      OKF08240 OOK 0879
B34      SR    7,8                                              OKF08250 OOK 0880
         ST    7,0(9,14)       34 KX(K)=KX(K)-MINE              OKF08260 OOK 0881
B35      S     1,FOUR                                           OKF08270 OOK 0882
         LR    3,1             35 LC(7)=LC(7)+J                 OKF08280 OOK 0883
         SRL   1,2                     (R3 HAS JJ*4)            OKF08290 OOK 0884
         L     7,LCAD                                           OKF08300 OOK 0885
         L     9,24(0,7)                                        OKF08310 OOK 0886
         AR    9,1                JJ=J-1                        OKF08320 OOK 0887
         A     9,ONE                                            OKF08330 OOK 0888
         ST    9,24(0,7)                                        OKF08340 OOK 0889
         LR    1,2                DO 43 J=1,JJ                  OKF08350 OOK 0890
B36      SR    6,6                                              OKF08360 OOK 0891
         AH    6,0(1,11)          KK=LDECR(IJ(J))               OKF08370 OOK 0892
         BP    B42                IF(KK.GT.0) GO TO 42          OKF08380 OOK 0893
         LCR   6,6                KK=-KK                        OKF08390 OOK 0894
         SLL   6,2                                              OKF08400 OOK 0895
         L     7,0(6,14)                                        OKF08410 OOK 0896
         SR    7,8                KX(KK)=KX(KK)-MINE            OKF08420 OOK 0897
         ST    7,0(6,14)                                        OKF08430 OOK 0898
         B     B43                GO TO 43                      OKF08440 OOK 0899
B42      SLL   6,2                                              OKF08450 OOK 0900
```

PAGE 25

```
        L       7,0(6,14)                                              0KF08460 00K 0901
        AR      7,8              42 KX(KK)=KX(KK)+MINE                 0KF08470 00K 0902
        ST      7,0(6,14)                                              0KF08480 00K 0903
B43     BXLE    1,2,B36          43 CONTINUE                           0KF08490 00K 0904
        LR      1,2                                                    0KF08500 00K 0905
        L       3,MAD                                                  0KF08510 00K 0906
        L       3,0(0,3)            DO 45 J=1,M                        0KF08520 00K 0907
        SLL     3,2                                                    0KF08530 00K 0908
        SR      7,7                                                    0KF08540 00K 0909
B44     ST      7,0(1,10)        45 NL(J)=0                            0KF08550 00K 0910
        BXLE    1,2,B44                                                0KF08560 00K 0911
        STH     7,4(0,11)           CALL PLACE(0,IJ(1))                0KF08570 00K 0912
        L       13,SAVER+4          RETURN                             0KF08580 00K 0913
        RETURN  (14,12),T           END                                0KF08590 00K 0914
        EJECT                                                          0KF08600 00K 0915
UPNOPR  SAVE    (14,12),,*          SUBROUTINE UPNOPR                  0KF08610 00K 0916
        BALR    12,0                                                   0KF08620 00K 0917
        USING   *,12                     (R12 IS BASE FOR THIS PROGRAM) 0KF08630 00K 0918
        LA      11,SAVER                                               0KF08640 00K 0919
        ST      13,4(0,11)                                             0KF08650 00K 0920
        ST      11,8(0,13)                                             0KF08660 00K 0921
        L       9,ILAD                                                 0KF08670 00K 0922
        S       9,FOUR                   (R9 HAS ADDRESS OF IL-4)      0KF08680 00K 0923
        L       11,IJAD                                                0KF08690 00K 0924
        S       11,FOUR                  (R11 HAS ADDRESS OF IJ-4)     0KF08700 00K 0925
        L       10,NLAD                                                0KF08710 00K 0926
        S       10,FOUR                  (R10 HAS ADDRESS OF NL-4)     0KF08720 00K 0927
        L       14,KXAD                                                0KF08730 00K 0928
        S       14,FOUR                  (R14 HAS ADDRESS OF KX-4)     0KF08740 00K 0929
        L       15,KUAD                                                0KF08750 00K 0930
        S       15,FOUR                  (R15 HAS ADDRESS OF KU-4)     0KF08760 00K 0931
        L       13,KCAD                                                0KF08770 00K 0932
        S       13,FOUR                  (R13 HAS ADDRESS OF KC-4)     0KF08780 00K 0933
        L       7,LCAD                                                 0KF08790 00K 0934
        L       6,20(0,7)                                              0KF08800 00K 0935
        A       6,ONE               LC(6)=LC(6)+1                      0KF08810 00K 0936
```

```
         ST     6,20(0,7)                                                   OKF08820 OOK 0937
         L      8,IFINAD                    (R8 HAS NDELTA)                 OKF08830 OOK 0938
         L      8,0(0,8)              NDELTA=IFIN                           OKF08840 OOK 0939
         SR     4,4                   I=0                                   OKF08850 OOK 0940
         SR     5,5                   KUP=0                                 OKF08860 OOK 0941
         L      2,FOUR                      (R4 HAS I*4)                    OKF08870 OOK 0942
         LR     1,2                         (R5 HAS KUP*4)                  OKF08880 OOK 0943
         L      3,NAD                       (R3 HAS N*4)                    OKF08890 OOK 0944
         L      3,0(0,3)              DO 24 L=1,N                           OKF08900 OOK 0945
         SLL    3,2                         (R2 HAS 4)                      OKF08910 OOK 0946
U12      CR     1,5                   IF(L.LE.KUP) GO TO 16                 OKF08920 OOK 0947
         BNH    U16                         (R6 HAS J*4)                    OKF08930 OOK 0948
         A      4,FOUR                I=I+1                                 OKF08940 OOK 0949
         LH     5,4(4,9)                    (R1 HAS L*4)                    OKF08950 OOK 0950
         BCTR   5,0                   KUP=LDECR(IL(I+1))-1                  OKF08960 OOK 0951
         SLL    5,2                                                         OKF08970 OOK 0952
U16      LH     6,2(1,11)          16 J=LADDR(IJ(L))                        OKF08980 OOK 0953
         SLL    6,2                                                         OKF08990 OOK 0954
         SR     7,7                                                         OKF09000 OOK 0955
         C      7,0(4,10)             IF(NL(I).EQ.0) GO TO 20               OKF09010 OOK 0956
         BE     U20                                                         OKF09020 OOK 0957
         C      7,0(6,10)             IF(NL(J).NE.0) GO TO 24               OKF09030 OOK 0958
         BNE    U24                                                         OKF09040 OOK 0959
         L      7,0(1,14)                                                   OKF09050 OOK 0960
         C      7,0(1,15)             IF(KX(L)-KU(L)) 22,24,24              OKF09060 OOK 0961
         BL     U22                                                         OKF09070 OOK 0962
         B      U24                                                         OKF09080 OOK 0963
U20      C      7,0(6,10)          20 IF(NL(J).EQ.0) GO TO 24               OKF09090 OOK 0964
         BE     U24                                                         OKF09100 OOK 0965
         C      7,0(1,14)             IF(KX(L).LE.0) GO TO 24               OKF09110 OOK 0966
         BNL    U24                                                         OKF09120 OOK 0967
U22      L      7,0(1,13)                                                   OKF09130 OOK 0968
         LPR    7,7                22 LL=IABS(KC(L))                        OKF09140 OOK 0969
         CR     8,7                                                         OKF09150 OOK 0970
         BNH    U24                   NDELTA=MINO(LL,NDELTA)                OKF09160 OOK 0971
         LR     8,7                                                         OKF09170 OOK 0972
```

```
U24      BXLE   1,2,U12                 24 CONTINUE                                  0KF09180 00K 0973
         L      7,IFINAD                                                             0KF09190 00K 0974
         L      7,0(0,7)                                                             0KF09200 00K 0975
         CR     8,7                         IF(NDELTA.LT.IFIN) GO TO 31              0KF09210 00K 0976
         BL     U31                                                                  0KF09220 00K 0977
         L      5,KAD                                                                0KF09230 00K 0978
         L      5,0(0,5)                                                             0KF09240 00K 0979
         SLL    5,2                                 (R5 HAS K*4)                     0KF09250 00K 0980
         SR     7,7                                                                  0KF09260 00K 0981
         A      7,0(5,14)                   IF(KX(K).EQ.0) GO TO 28                  0KF09270 00K 0982
         BZ     U28                                                                  0KF09280 00K 0983
         C      7,0(5,15)                   IF(KX(K).NE.KU(K)) GO TO 51              0KF09290 00K 0984
         BNE    U51                                                                  0KF09300 00K 0985
U28      L      8,0(5,13)                                                            0KF09310 00K 0986
         LPR    8,8                     28 NDELTA=IABS(KC(K))                        0KF09320 00K 0987
U31      L      15,NPAD                                                              0KF09330 00K 0988
         S      15,FOUR                             (R15 HAS ADDRESS OF NP-4)        0KF09340 00K 0989
         LR     1,2                                 (R3 HAS M*4)                     0KF09350 00K 0990
         L      3,MAD                   31 DO 47 I=1,M                               0KF09360 00K 0991
         L      3,0(0,3)                            (R1 HAS I*4)                     0KF09370 00K 0992
         SLL    3,2                                                                  0KF09380 00K 0993
U32      LH     4,4(1,9)                                                             0KF09390 00K 0994
         BCTR   4,0                         L2=LDECR(IL(I+1))-1                      0KF09400 00K 0995
         SLL    4,2                                 (R4 HAS L2*4)                    0KF09410 00K 0996
         LH     5,0(1,9)                    L=LDECR(IL(I))                           0KF09420 00K 0997
         SLL    5,2                                 (R5 HAS L*4)                     0KF09430 00K 0998
         SR     7,7                                                                  0KF09440 00K 0999
         C      7,0(1,10)                   IF(NL(I).NE.0) GO TO 41                  0KF09450 00K 1000
         BNE    U41                                                                  0KF09460 00K 1001
         L      7,0(1,15)                                                            0KF09470 00K 1002
         AR     7,8                         NP(I)=NP(I)+NDELTA                       0KF09480 00K 1003
         ST     7,0(1,15)                                                            0KF09490 00K 1004
         C      7,=F'100000000'             IF(NP(I).GT.100000000) GO TO 49          0KF09500 00K 1005
         BH     U49                                                                  0KF09510 00K 1006
U36      CR     4,5                     36 IF(L2.LT.L) GO TO 47                      0KF09520 00K 1007
         BL     U47                                                                  0KF09530 00K 1008
```

```
         LH    6,2(5,11)            J=LADDR(IJ(L))                         OKF09540 OOK 1009
         SLL   6,2                                                         OKF09550 OOK 1010
         SR    7,7                                                         OKF09560 OOK 1011
         C     7,0(6,10)            IF(NL(J).EQ.0) GO TO 40                OKF09570 OOK 1012
         BE    U40                                                         OKF09580 OOK 1013
         L     7,0(5,13)                                                   OKF09590 OOK 1014
         AR    7,8                  KC(L)=KC(L)+NDELTA                     OKF09600 OOK 1015
         ST    7,0(5,13)                                                   OKF09610 OOK 1016
U40      A     5,FOUR            40 L=L+1                                  OKF09620 OOK 1017
         B     U36                  GO TO 36                               OKF09630 OOK 1018
U41      CR    4,5               41 IF(L2.LT.L) GO TO 47                   OKF09640 OOK 1019
         BL    U47                                                         OKF09650 OOK 1020
         LH    6,2(5,11)            J=LADDR(IJ(L))                         OKF09660 OOK 1021
         SLL   6,2                                                         OKF09670 OOK 1022
         SR    7,7                                                         OKF09680 OOK 1023
         C     7,0(6,10)            IF(NL(J).NE.0) GO TO 46                OKF09690 OOK 1024
         BNE   U46                                                         OKF09700 OOK 1025
         L     7,0(5,13)                                                   OKF09710 OOK 1026
         SR    7,8                  KC(L)=KC(L)-NDELTA                     OKF09720 OOK 1027
         ST    7,0(5,13)                                                   OKF09730 OOK 1028
U46      A     5,FOUR            46 L=L+1                                  OKF09740 OOK 1029
         B     U41                  GO TO 41                               OKF09750 OOK 1030
U47      BXLE  1,2,U32           47 CONTINUE                               OKF09760 OOK 1031
         B     U50                  GO TO 50                               OKF09770 OOK 1032
U49      L     7,LERAD                                                     OKF09780 OOK 1033
         L     8,=F'404'         49 LER=404                                OKF09790 OOK 1034
         ST    8,0(0,7)                                                    OKF09800 OOK 1035
U50      L     13,SAVER+4                                                  OKF09810 OOK 1036
         RETURN (14,12),T        50 RETURN                                 OKF09820 OOK 1037
U51      L     7,LERAD                                                     OKF09830 OOK 1038
         L     8,=F'107'         51 LER=107                                OKF09840 OOK 1039
         ST    8,0(0,7)             GO TO 50                               OKF09850 OOK 1040
         B     U50                  END                                    OKF09860 OOK 1041
         EJECT                                                             OKF09870 OOK 1042
NODENO   SAVE  (14,12),,*           FUNCTION NODENO(IN1,IN2)               OKF09880 OOK 1043
         BALR  12,0                                                        OKF09890 OOK 1044
                                                                                PAGE  29
```

```
         USING *,12                         (R12 IS BASE FOR THIS PROGRAM)  OKF09900 00K 1045
         LA    11,SAVER                                                     OKF09910 00K 1046
         ST    13,4(0,11)                                                   OKF09920 00K 1047
         ST    11,8(0,13)                                                   OKF09930 00K 1048
         L     10,NNAD                                                      OKF09940 00K 1049
         S     10,FOUR                      (R10 HAS ADDRESS OF NN-4)       OKF09950 00K 1050
         L     7,0(0,1)                                                     OKF09960 00K 1051
         L     7,0(0,7)                     (R7 HAS IN1)                    OKF09970 00K 1052
         L     8,4(0,1)                                                     OKF09980 00K 1053
         L     8,0(0,8)                     (R8 HAS IN2)                    OKF09990 00K 1054
         L     2,EIGHT                      (R2 HAS 8)                      OKF10000 00K 1055
         LR    1,2                                                          OKF10010 00K 1056
         L     3,MAD                                                        OKF10020 00K 1057
         L     3,0(0,3)                DO 9 K=1,M                           OKF10030 00K 1058
         SLL   3,3                          (R3 HAS M*8)                    OKF10040 00K 1059
         L     4,FOUR                       (R1 HAS K*8)                    OKF10050 00K 1060
N3       C     7,0(4,10)                    (R4 HAS 4)                      OKF10060 00K 1061
         BNE   N9   IF(IN1.EQ.NN(2*K-1).AND.IN2.EQ.NN(2*K)) GO TO 12        OKF10070 00K 1062
         C     8,0(1,10)                                                    OKF10080 00K 1063
         BE    N12                                                          OKF10090 00K 1064
N9       A     4,EIGHT                                                      OKF10100 00K 1065
         BXLE  1,2,N3                9 CONTINUE                             OKF10110 00K 1066
         SRL   3,3                                                          OKF10120 00K 1067
         A     3,ONE                   NODENO=M+1                           OKF10130 00K 1068
         ST    3,20(0,13)                                                   OKF10140 00K 1069
         B     N13                     GO TO 13                             OKF10150 00K 1070
N12      SRL   1,3                  12 NODENO=K                             OKF10160 00K 1071
         ST    1,20(0,13)           13 RETURN                               OKF10170 00K 1072
N13      RETURN (14,12),T              END                                  OKF10180 00K 1073
         SPACE 2                                                            OKF10190 00K 1074
FOUR     DC    F'4'                                                         OKF10200 00K 1075
EIGHT    DC    F'8'                                                         OKF10210 00K 1076
ONE      DC    F'1'                                                         OKF10220 00K 1077
I        DS    1F                                                           OKF10230 00K 1078
NUP      DS    1F                                                           OKF10240 00K 1079
SAVER    DS    18F                                                          OKF10250 00K 1080
```

```
         SPACE 2                                          OKF10260
KLAD     DC     V(KL)                                     OKF10270
KCAD     DC     V(KC)                                     OKF10280
KUAD     DC     V(KU)                                     OKF10290
KXAD     DC     V(KX)                                     OKF10300
NLAD     DC     V(NL)                                     OKF10310
NNAD     DC     V(NN)                                     OKF10320
NPAD     DC     V(NP)                                     OKF10330
IJAD     DC     V(IJ)                                     OKF10340
ILAD     DC     V(IL)                                     OKF10350
JLAD     DC     V(JL)                                     OKF10360
JIAD     DC     V(JI)                                     OKF10370
MAD      DC     V(M)                                      OKF10380
NAD      DC     V(N)                                      OKF10390
LERAD    DC     V(LER)                                    OKF10400
KATAD    DC     V(KAT)                                    OKF10410
KORAD    DC     V(KOR)                                    OKF10420
KTERAD   DC     V(KTER)                                   OKF10430
MINEAD   DC     V(MINE)                                   OKF10440
LCAD     DC     V(LC)                                     OKF10450
KAAD     DC     V(KA)                                     OKF10460
IFINAD   DC     V(IFIN)                                   OKF10470
KIAD     DC     V(KI)                                     OKF10480
KOAD     DC     V(KO)                                     OKF10490
KQAD     DC     V(KQ)                                     OKF10500
KAD      DC     V(K)                                      OKF10510
         END                                              OKF10520
/*                                                        OKF10530
//STEP EXEC ASMC,PARM.C='LOAD,DECK'
//C.SYSIN DD    *                                         OKF06010
ASSEM2   START  0                                         OKF10560
         ENTRY  PLACE,LADDR,LDECR                         OKF10570
         SPACE  2                                         OKF10580
PLACE    SAVE   (14,12),,*                                OKF10590
         BALR   12,0                                      OKF10600
         USING  *,12                                      OKF10610
```

```
         LA     11,SAVER                                                        OKF10620 OOK
         ST     13,4(11)                                                        OKF10630 OOK
         ST     11,8(13)                                                        OKF10640 OOK
         L      6,0(1)                   CALL PLACE(A,B)                        OKF10650 OOK
         L      7,4(1)                   PLACES THE RIGHTMOST 16 BITS OF A      OKF10660 OOK
         MVC    0(2,7),2(6)              IN THE LEFTMOST 16 BITS OF B           OKF10670 OOK
         RETURN (14,12),T                                                       OKF10680 OOK
         SPACE 2                                                                OKF10690 OOK
LADDR    SAVE   (14,12),,*                                                      OKF10700 OOK
         BALR   12,0                                                            OKF10710 OOK
         USING  *,12                                                            OKF10720 OOK
         LA     11,SAVER                                                        OKF10730 OOK
         ST     13,4(11)                                                        OKF10740 OOK
         ST     11,8(13)                                                        OKF10750 OOK
         L      6,0(1)                   THE FUNCTION LADDR(A) RETURNS THE      OKF10760 OOK
         LH     6,2(6)                   RIGHTMOST 16 BITS OF A AS A 32-BITS    OKF10770 OOK
         ST     6,20(13)                 FORTRAN INTEGER.                       OKF10780 OOK
         RETURN (14,12),T                                                       OKF10790 OOK
         SPACE 2                                                                OKF10800 OOK
LDECR    SAVE   (14,12),,*                                                      OKF10810 OOK
         BALR   12,0                                                            OKF10820 OOK
         USING  *,12                                                            OKF10830 OOK
         LA     11,SAVER                                                        OKF10840 OOK
         ST     13,4(11)                                                        OKF10850 OOK
         ST     11,8(13)                                                        OKF10860 OOK
         L      6,0(1)                   THE FUNCTION LDECR(A) RETURNS THE      OKF10870 OOK
         LH     6,0(6)                   LEFTMOST 16 BITS OF A AS A 32-BITS     OKF10880 OOK
         ST     6,20(13)                 FORTRAN INTEGER                        OKF10890 OOK
         RETURN (14,12),T                                                       OKF10900 OOK
         SPACE 2                                                                OKF10910 OOK
SAVER    DS     18F                                                             OKF10920 OOK
         END                                                                    OKF10930 OOK
/*                                                                              OKF10940 OOK
************************************************************************ 00000000 OOK
*                                                                      * 00000010 OOK
*                ASSEMBLER LANGUAGE SUBROUTINE ERASE                   * 00000011 OOK
```

```
*                    WRITTEN BY JOHN W. KIDSON                        * 00000012 00K 1153
*                  MIT DEPARTMENT OF METEOROLOGY                      * 00000014 00K 1154
*                                                                     * 00000016 00K 1155
*      TO SET ELEMENTS OF REAL OR INTEGER ARRAYS TO ZERO. A1,A2,...  * 00000020 00K 1156
*      ARE ARRAY NAMES AND N1,N2,... ARE INTEGER VALUES OR            * 00000030 00K 1157
*      EXPRESSIONS GIVING THE ARRAY SIZES.                            * 00000040 00K 1158
**     I.E. - CALL ERASE(C,26*31,N,7*31,E,254)                       ** 00000050 00K 1159
*                                                                     * 00000060 00K 1160
*********************************************************************** 00000070 00K 1161
ERASE    START 0                                                        00000080 00K 1162
         SAVE  (14,12),,*                                               00000090 00K 1163
         BALR  12,0                                                     00000100 00K 1164
         USING *,12                                                     00000110 00K 1165
         SR    0,0                                                      00000120 00K 1166
         SR    2,2              PARAMETER LIST INDEX=0                  00000130 00K 1167
         L     6,=F'4'                                                  00000140 00K 1168
E1       L     3,0(2,1)         LOAD 3 WITH ARRAY ADDRESS               00000150 00K 1169
         L     4,4(2,1)         LOAD 4 WITH ADDRESS OF ARRAY LENGTH     00000160 00K 1170
         L     7,0(4)           LOAD 7 WITH ARRAY LENGTH-1 TIMES 4      00000170 00K 1171
         SLA   7,2                                                      00000180 00K 1172
         SR    7,6                                                      00000190 00K 1173
         SR    5,5                                                      00000200 00K 1174
E2       ST    0,0(5,3)         STORE ZERO                              00000210 00K 1175
         BXLE  5,6,E2                                                   00000220 00K 1176
         LTR   4,4              TEST FOR LAST ARGUMENT IN LIST          00000230 00K 1177
         BM    RETN                                                     00000240 00K 1178
         A     2,=F'8'                                                  00000250 00K 1179
         B     E1               PICK UP NEXT ARGUMENT PAIR              00000260 00K 1180
RETN     RETURN (14,12),T                                               00000270 00K 1181
         END                                                            00000280 00K 1182
*********************************************************************** 00000290 00K 1183
```

```
C**********************************************************************
C*                                                                    *
C*     Q K C O R E   :   A QUICK IN-CORE MODEL WITH COSTING INCLUDED  *
C*                        WRITTEN BY PAUL F. DEATON                   *
C*                  M.I.T. DOCTORAL THESIS,   MARCH  1973             *
C*                                                                    *
C**********************************************************************
C      QKCORE MAIN PROGRAM
C      QKCORE VERSION 12-15-72
       REAL*8 RTC
       COMMON/FXDDAT/MXZONE,MXCYTO,MXRCRS,MXRCRK,MXFULK,IRCRS,IRCRK,IFULK
      $,NRCRS,NRCRK,NFULK,EFF,XF,XW,TXRATE,PVRATE,TBASE,DTPRE,DTPST,
      $CTY2F6,CCRATE,FCOR,FFAB,FSAR,FCRE,NCYCIN,NCYCXS,NCYCTO,NZONE,NZP,
      $ZONEKG,ECHDOV,EFFAV,MWS
       COMMON/PRINTS/RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS,RD,WT
       LOGICAL RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS
       INTEGER RD,WT
       DIMENSION ELAME(50,20),NECBAL(20),TE(20),TS(20),CATITL(20)
       DIMENSION ECUPLM(20),TO(20)
       DATA $NEWB$,$CASE$,$STOP$/'NEW ','CASE','STOP'/
       MXESX2=50
       PRINT 900
    10 CALL ICNPUT
    20 READ (RD,920) CATITL
       WRITE(WT,921) CATITL
       IF(CATITL(1).EQ.$NEWB$) GO TO 10
       IF(CATITL(1).EQ.$STOP$) CALL ICERRS('QKCORE',8)
       IF(CATITL(1).NE.$CASE$) CALL ICERRS('QKCORE',3)
       READ (RD,920) CATITL
       WRITE(WT,922) CATITL
       READ (RD,923) NCYCIN,NCYCXS,IDNUM,ECHDOV
       WRITE(WT,924) NCYCIN,NCYCXS,IDNUM,ECHDOV
       NCYCTO=NCYCIN+NCYCXS
       CALL ERASE(ELAME,MXESX2*MXCYTO,TS,MXCYTO,TE,MXCYTO,NECBAL,MXCYTO)
       MXNES=0
       DO 30 IDUM=1,NCYCTO
```

```
      READ(RD,925) I,NECBAL(I),TS(I),TE(I),NES,TO(I),(ELAME(2*N-1,I),
     $N=1,NES)
   30 MXNES=MAX0(MXNES,NES)
      MXE2=(MXESX2+1)/2
      IF(MXNES.GT.MXE2) CALL ICERRS('QKCORE',10)
      WRITE(WT,908)
      WRITE(WT,903)           (I,I=1,NCYCTO)
      WRITE(WT,904) (     TS(I),I=1,NCYCTO)
      WRITE(WT,910) (     TO(I),I=1,NCYCTO)
      WRITE(WT,905) (     TE(I),I=1,NCYCTO)
      WRITE(WT,906) (NECBAL(I),I=1,NCYCTO)
      WRITE(WT,909)
      DO 40 N=1,MXNES
      M=2*N-1
   40 WRITE(WT,907) (ELAME(M,I),I=1,NCYCTO)
      INCCST=.TRUE.
      CALL INCCRE(IDNUM,NCYCIN,NCYCXS,NCYCTO,TS,TE,NECBAL,ELAME,MXESX2,
     $ECHDOV,RTC,PVRAT,BASETM,ECUPLM,TO)
      WRITE(WT,926) RTC,BASETM,PVRAT
      WRITE(WT,927) CATITL
      IF(INCCST) GO TO 20
      STOP
  900 FORMAT(T31,72('*')/T31,'*',T102,'*'/T31,'*',T37,'Q K C O R E   :
     $  A QUICK IN-CORE MODEL WITH COSTING INCLUDED ',T102,'*'/
     $T31,'*',T64,'WRITTEN BY PAUL F. DEATON',T102,'*'/
     $T31,'*',T58,'M.I.T. DOCTORAL THESIS,    MARCH  1973 ',T102,'*'/
     $T31,'*',T102,'*'/T31,72('*')//
     $T56,'VERSION  12-15-72')
  903 FORMAT('0 CYCLE',14(I6,3X)/(12X,12(I6,3X)))
  904 FORMAT('0TSTART',14F9.4/(12X,12F9.4))
  905 FORMAT('   TEND ',14F9.4/(12X,12F9.4))
  906 FORMAT(' NECBAL',14(I6,3X)/(12X,12(I6,3X)))
  907 FORMAT('        ',14F9.2/(12X,12F9.2))
  908 FORMAT('0'/'0           CASE INPUT DATA :')
  909 FORMAT('0',T7 ,'TABLE OF EC''S TO BE INVESTIGATED :')
  910 FORMAT('   TOPR ',14F9.4/(12X,12F9.4))
```

```
  920 FORMAT(20A4)
  921 FORMAT('0  QKCORE READ CARD :',2H ',20A4,1H')
  922 FORMAT('1',T15,'QKCORE CASE TITLE :',2H ',20A4,1H')
  923 FORMAT(3I10,F10.2)
  924 FORMAT('0',T7,'NCYCIN    NCYCXS    IDNUM   ECHDOV'/3I10,F10.2)
  925 FORMAT(2I10,2F10.4,I10,F10.4/(8F10.4))
  926 FORMAT('0'/'0'/'0','INCORE RETURNED THE FOLLOWING VALUES TO QKCORE
     $ :'/           '0    REACTOR TOTAL COST =',-3PF15.6,' MILLION DOLLA
     $RS PRESENT VALUED TO YEAR ',0PF8.4,' AT THE RATE OF ',2PF6.3,' PER
     $ CENT PER YEAR'/'0'/'0'/)
  927 FORMAT('0'/'0'/'0 END OF QKCORE CASE TITLE :',2H ',20A4,1H')
      END
      SUBROUTINE INCORE(IDNUM,NCYCIN,NCYCXS,NCYCTO,TS,TE,NECBAL,ELAME,
     $MXESX2,ECHDOV,PVRTC,PVRAT,BASETM,ECUPLM,TC)
C     MAIN SUBROUTINE OF IN-CORE FUEL SIMULATOR
C     QKCORE VERSION 12-15-72
C*********** DEFINITIONS OF IMPORTANT VARIABLES **********************
C     ACCYC  = AVERAGE CYCLE COST AT IT'S MID-PT. ($/MWHE)
C     ACEOCD = AVERAGE COST OF BATCH DISCHARGED AT END OF CYCLE ($/MWHE)
C     B      = BURNUP (MWD/KG)
C     BALCST = PRINT DETAILED COST TABLES FOR BALANCED EC'S ?
C     BASETM = BASE TIME FOR PRESENT VALUING (YEARS)
C     BATCST = TOTAL BATCH COST (10**3 $)
C     BSRT   = ZONE BURNUPS OF FUELS AT START OF SIMULATION (MWD/KG)
C     C      = UNIT  BATCH COST ($/KG)
C     CCRATE = CARRYING CHARGE RATE (FRACTION)
C     DECRIT = FIRST CYCLE ENERGY AVAILABLE BEFORE BARELY CRITICAL (GWHE)
C     DESTCH = UPPER LIMIT ON STRETCHOUT ENERGY (GWHE)
C     DTC    = ON-LINE CYCLE LENGTH (YEARS)
C     DTPRE  = EFFECTIVE DELAY TIME FOR PRE-REACTOR PAYMENTS (YEARS)
C     DTPST  = EFFECTIVE DELAY TIME FOR POST-REACTOR RECEIPTS (YEARS)
C     DTY2F6 = EFFECTIVE DELAY TIME FROM YELLOWCAKE TO UF6 (YEARS)
C     EC     = ELECTRICAL ENERGY PRODUCED IN THE CYCLE (GWHE)
C     ECHDOV = GWHE HELD OVER FOR PROD. BEYOND HORIZON IN SPLIT CYCLE
C     ECUPLM = UPPER LIMIT ON CYCLE PRODUCTION  (GWHE)
C     EFF    = EFFINC
```

```
C       EFFAV  = EFFNET                                                    QKCR0109
C       EFFINC = REACTOR INCREMENTAL EFFICIENCY (FRACTION)                 QKCR0110
C       EFFNET = REACTOR NET THERMAL EFFICIENCY (FRACTION)                 QKCR0111
C       ELAME  = SANDWICHED MATRIX OF EC'S, LAMBDAS AND EC'S (GWHE,$/MWHE) QKCR0112
C       EPF    = ENRICHMENT AS-FABRICATED (W/O U-235)                      QKCR0113
C       EPFFX  = FIXED ENRICHMENTS OF INITIAL CYCLES (W/O U-235)           QKCR0114
C       EPFSRT = AS-FAB. ENRICHMENT OF INITIALLY PRESENT FUELS (W/O U-235) QKCR0115
C       ERRCOD = ACCUMULATED ERROR CODE                                    QKCR0116
C       FABINV = UN-DEPREC. FAB. INVENTORY FOR STARTING FUELS ($/KG-FAB)   QKCR0117
C       FCOR   = YIELD IN CONVERSION STEP OF FUEL CYCLE (FRACTION)         QKCR0118
C       FCRE   = YIELD IN RECYCLE CONVERSION STEP OF FUEL CYCLE (FRACTION) QKCR0119
C       FFAB   = YIELD IN FABRICATION STEP OF FUEL CYCLE (FRACTION)        QKCR0120
C       FSAR   = YIELD IN SHIP.&REPROC. STEP OF FUEL CYCLE (FRACTION)      QKCR0121
C       FULCON = SETS OF EMPIRICAL FUEL CONSTANTS                          QKCR0122
C       IDNO   = REACTOR I.D. NUMBER                                       QKCR0123
C       IDNUM  = I.D. NUMBER OF REACTOR TO BE SIMULATED                    QKCR0124
C       IFULK  = FUEL CONSTANTS INDEX                                      QKCR0125
C       IFULKA = POINTER TO SET OF FUEL CONSTANTS TO BE USED               QKCR0126
C       INCCST = PRINT INCREMENTAL COST TABLE ?                            QKCR0127
C       IRCRK  = REACTOR CONSTANTS INDEX                                   QKCR0128
C       IRCRKA = POINTER TO SET OF REACTOR CONSTANTS TO BE USED            QKCR0129
C       IRCRS  = REACTOR INDEX                                             QKCR0130
C       MODIRR = MODE OF IRRADIATION                                       QKCR0131
C       MWCAP  = REACTOR RATED CAPACITY (MWE)                              QKCR0132
C       MXCYTO = MAXIMUM ALLOWED VALUE OF NCYCTO                           QKCR0133
C       MXESX2 = FIRST DIMENSION OF ELAME = MAX.NO. EC'S * 2               QKCR0134
C       MXFULK = MAXIMUM NUMBER OF ALLOWABLE SETS OF FUEL CONSTANTS        QKCR0135
C       MXRCRK = MAXIMUM NUMBER OF ALLOWABLE SETS OF REACTOR CONSTANTS     QKCR0136
C       MXRCRS = MAXIMUM NUMBER OF ALLOWABLE SETS OF REACTOR SPECS.        QKCR0137
C       MXZONE = MAXIMUM NUMBER OF ZONES                                   QKCR0138
C       NAME   = REACTOR NAME                                              QKCR0139
C       NBLCST = PRINT DETAILED COST TABLE FOR UNBALANCED EC'S ?           QKCR0140
C       NCYCFX = NUMBER OF INITIAL CYCLES WITH ENRICHMENT FIXED            QKCR0141
C       NCYCIN = NUMBER OF CYCLES INVOLVED IN HORIZON                      QKCR0142
C       NCYCTO = TOTAL NUMBER OF CYCLES = NCYCIN + NCYCXS                  QKCR0143
C       NCYCXS = NUMBER OF EXCESS CYCLES BEYOND HORIZON                    QKCR0144
```

```
C       NECBAL = POSITION OF ECBAL WITHIN A COLUMN OF EC'S OF ELAME             QKCR0145
C       NFULK  = NUMBER OF SETS OF FUEL CONSTANTS READ IN                       QKCR0146
C       NOESX2 = (NUMBER OF EC'S)*2 IN EACH CYCLE OF THE SIMULATION             QKCR0147
C       NOZONE = NUMBER OF ZONES IN FUEL MANAGEMENT SCHEME                      QKCR0148
C       NRCRK  = NUMBER OF SETS OF REACTOR CONSTANTS READ IN                    QKCR0149
C       NRCRS  = NUMBER OF SETS OF REACTOR SPECS.    READ IN                    QKCR0150
C       NZONE  = NOZONE                                                         QKCR0151
C       NZP    = NOZONE + 1                                                     QKCR0152
C       PBATCS = PRINT DETAILED COST FOR ALL BATCHES ?                          QKCR0153
C       PIRDAT = PRINT DATA FOR EACH IRRADIATION CYCLE ?                        QKCR0154
C       POWFRC = ZONE POWER-SHARING FRACTIONS OF STARTING CYCLE                 QKCR0155
C       PVFACT = PRESENT VALUE OF 1$ AT MID-PT. OF CYCLE                        QKCR0156
C       PVRAT  = PRESENT VALUE RATE (FRACTION PER YEAR)                         QKCR0157
C       PVRATE = PVRAT                                                          QKCR0158
C       PVRTC  = PRESENT VALUE OF REACTOR TOTAL COST (10**3 $)                  QKCR0159
C       PVTCYC = PRESENT VALUE OF CYCLE COST (10**3 $)                          QKCR0160
C       RCRCON = SETS OF EMPIRICAL REACTOR CONSTANTS                            QKCR0161
C       RD     = UNIT NUMBER OF COMPUTER INPUT READING DEVICE                   QKCR0162
C       RELCST = PRINT RELATIVE COST TABLE ?                                    QKCR0163
C       SRCINV = UN-DEPREC. SRC. INVENTORY OF STARTING FUELS ($/KG-FAB)         QKCR0164
C       TBASE  = BASE TIME FOR PRESENT VALUING (YEARS)                          QKCR0165
C       TCCYC  = TOTAL CYCLE COST AT IT'S MID-PT. (10**3 $)                     QKCR0166
C       TCEOCD = TOTAL COST OF BATCH DISCHARGED AT END OF CYCLE (10**3 $)       QKCR0167
C       TE     = ENDING   CYCLE DATES (YEARS)                                   QKCR0168
C       TMID   = MID-POINT OF CYCLE (YEARS)                                     QKCR0169
C       TO     = CYCLE OPERATING TIME (YEARS)                                   QKCR0170
C       TREFUL = REFUELING DATE (YEARS)                                         QKCR0171
C       TS     = STARTING CYCLE DATES (YEARS)                                   QKCR0172
C       TXRATE = INCOME TAX RATE (FRACTION)                                     QKCR0173
C       UNTCOR = UNIT CONVERSION COST ($/KG-U CONV)                             QKCR0174
C       UNTCRE = UNIT RECYCLE CONVERSION COST ($/KG-U CONV)                     QKCR0175
C       UNTFAB = UNIT FABRICATION COST ($/KG-FAB)                               QKCR0176
C       UNTPUV = UNIT PLUTONIUM VALUE ($/GM-FIS.PU)                             QKCR0177
C       UNTSAR = UNIT SHIP.&REPROC. COST ($/KG-SAR)                             QKCR0178
C       UNTSWU = UNIT SEPARATIVE WORK COST ($/KG-SWU)                           QKCR0179
C       UNTYEL = UNIT YELLOWCAKE COST ($/LB-U308)                               QKCR0180
```

```
C      WT      = UNIT NUMBER CF COMPUTER OUTPUT WRITING DEVICE
C      XF      = ENRICHMENT OF YELLOWCAKE (WT.FR. U-235)
C      XW      = ENRICHMENT OF DIFFUSION PLANT TAILS (WT.FR. U-235)
C      ZEROHT = TOTAL HEAT REQT. FOR ZERO POWER DURING TO (GWHTH)
C      ZONEKG = ZONKG
C      ZCNKG  = MASS RELOADED AT EACH REFUELING (KILOGRAMS)
C*********** END OF DEFINITIONS ***************************************
       REAL*8 PVRTC
       CIMENSION G(1000)
       DIMENSION TS(NCYCTO),TE(NCYCTO),NECBAL(NCYCTO),TO(NCYCTO),
      $ELAME(MXESX2,NCYCTO),ECUPLM(NCYCTO)
       COMMCN/ARDATA/IDNO(15),NAME(15),MWCAP(15),EFFNET(15),IRCRKA(15),
      $IFULKA(15),NOZONE(15),ZONKG(15),DECRIT(15),DESTCH(15),NCYCFX(15),
      $EPFFX(20,15),EPFSRT(10,15),BSRT(10,15),FABINV(10,15),SRCINV(10,15)
      $,POWFRC(10,15),RCRCCN(18,15),FULCON(48,5),EFFINC(15)
       CIMENSION NOESX2(20)
       COMMON/FXDDAT/MXZONE,MXCYTO,MXRCRS,MXRCRK,MXFULK,IRCRS,IRCRK,IFULK
      $,NRCRS,NRCRK,NFULK,EFF,XF,XW,TXRATE,PVRATE,TBASE,DTPRE,DTPST,
      $CTY2F6,CCRATE,FCOR,FFAB,FSAR,FCRE,DUMMY1,DUMMY2,DUMMY3,NZONE,NZP,
      $ZONEKG,DUMMY4,EFFAV,MWS
       COMMON/PRINTS/RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS,RD,WT
       LOGICAL RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS
       INTEGER RD,WT
       INTEGER CUMMY1,DUMMY2,DUMMY3
       DUMMY1=NCYCIN
       CUMMY2=NCYCXS
       DUMMY3=NCYCTO
       CUMMY4=ECHDOV
       GO TC 5
C
C
       ENTRY ICNPUT
C      ENTRY POINT TO INCORE AS SIGNAL TO PREPARE FOR SIMULATION BY
C      READING PERTINENT INPUT CARDS
C
C      SET VERSION MAXIMUMS
```

```
      MXLAST=1000
      MXZONE=10
      MXCYTO=20
      MXRCRS=15
      MXRCRK=15
      MXFULK=5
      RD=5
      WT=6
      CALL REDCOR
      RETURN
C
C
    5 IF(NCYCIN+NCYCXS.NE.NCYCTO) CALL ICERRS('INCORE',6)
      NCYCTO=NCYCIN+NCYCXS
      IF(NCYCTO.GT.MXCYTO) CALL ICERRS('INCORE',5)
      PVRAT=PVRATE
      BASETM=TBASE
      DO 10 I=1,NRCRS
      IF(IDNO(I).EQ.IDNUM) GO TO 20
   10 CONTINUE
      CALL ICERRS('INCORE',7)
   20 IRCRS=I
      NZONE=NOZONE(IRCRS)
      NZP=NZONE+1
      ZONEKG=ZONKG(IRCRS)
      IRCRK=IRCRKA(IRCRS)
      IFULK=IFULKA(IRCRS)
      EFF=EFFINC(IRCRS)
      EFFAV=EFFNET(IRCRS)
      MWS=MWCAP(IRCRS)
      ECRIT1=DECRIT(IRCRS)
      STCHLM=DESTCH(IRCRS)
C     SETUP POINTERS AND INITIALIZE SOME SUBROUTINES
      NCYCTP=NCYCTO+1
      LTREFU=1
      LTMID =LTREFU+NCYCTP
```

```
      LDTC  =LTMID +NCYCTP                                              QKCR0253
      LMODIR=LDTC  +NCYCTP                                              QKCR0254
      LUNTYE=LMODIR+NCYCTP                                              QKCR0255
      LUNTCO=LUNTYE+NCYCTP                                              QKCR0256
      LUNTSW=LUNTCO+NCYCTP                                              QKCR0257
      LUNTFA=LUNTSW+NCYCTP                                              QKCR0258
      LUNTSA=LUNTFA+NCYCTP                                              QKCR0259
      LUNTCR=LUNTSA+NCYCTP                                              QKCR0260
      LUNTPU=LUNTCR+NCYCTP                                              QKCR0261
      LPVFAC=LUNTPU+NCYCTP                                              QKCR0262
      LEC   =LPVFAC+NCYCTP                                              QKCR0263
      LPVTCY=LEC   +NCYCTP                                              QKCR0264
      LTCCYC=LPVTCY+NCYCTP*2                                            QKCR0265
      LACCYC=LTCCYC+NCYCTP*2                                            QKCR0266
      LTCEOC=LACCYC+NCYCTP                                              QKCR0267
      LACEOC=LTCEOC+NCYCTP                                              QKCR0268
      LZEROH=LACEOC+NCYCTP                                              QKCR0269
      LEPF  =LZEROH+NCYCTP                                              QKCR0270
      LB    =LEPF  +NZP*NCYCTP                                          QKCR0271
      LBATCS=LB    +NZP*NCYCTP                                          QKCR0272
      LA    =LBATCS+NZP*NCYCTP                                          QKCR0273
      LBC   =LA    +NZP                                                 QKCR0274
      LDBC  =LBC   +NZP                                                 QKCR0275
      LDT   =LDBC  +NZP                                                 QKCR0276
      LKGU  =LDT   +NZP                                                 QKCR0277
      LEPNOW=LKGU  +NZP                                                 QKCR0278
      LUVALU=LEPNOW+NZP                                                 QKCR0279
      LGMP  =LUVALU+NZP                                                 QKCR0280
      LIUF6 =LGMP  +NZP                                                 QKCR0281
      LIFAB =LIUF6 +NZP                                                 QKCR0282
      LISRC =LIFAB +NZP                                                 QKCR0283
      LIPUV =LISRC +NZP                                                 QKCR0284
      LITOT =LIPUV +NZP                                                 QKCR0285
      LTCST =LITOT +NZP                                                 QKCR0286
      LACST =LTCST +NZP                                                 QKCR0287
      LNEXT =LACST +NZP                                                 QKCR0288
```

```
      LLAST=LNEXT-1
C     LLAST=21*NCYCTP+3*NZP*NCYCTP+15*NZP
      IF(LLAST.GT.MXLAST) WRITE(WT,900) LLAST,MXLAST
      IF(LLAST.GT.MXLAST) CALL ICERRS('INCORE',4)
      DUMMY=EMPRCL(FULCCN(1,IFULK),RCRCON(1,IRCRK))
      CALL INIT3(FABINV(1,IRCRS),SRCINV(1,IRCRS),
     $G(LEPF  ),G(LDTC  ),G(LB    ),G(LUNTYE),G(LUNTCO),G(LUNTSW),
     $G(LUNTFA),G(LUNTSA),G(LUNTCR),G(LUNTPU),G(LTCECC),G(LACEOC),
     $G(LA    ),G(LBC   ),G(LDBC  ),G(LDT   ),G(LKGU  ),G(LEPNOW),
     $G(LUVALU),G(LGMP  ),G(LIUF6 ),G(LIFAB ),G(LISRC ),G(LIPUV ),
     $G(LITOT ),G(LTCST ),G(LACST ))
      MX2EUS=0
      DO 45 N=1,NCYCTO
      DO 30 I=1,MXESX2,2
      IF(ELAME(I,N).EQ.0.0) GO TO 40
   30 CONTINUE
      I=MXESX2/2*2+1
   40 NOESX2(N)=I-1
   45 MX2EUS=MAX0(MX2EUS,NOESX2(N))
      CALL FULSIM(MXESX2,NOESX2,ELAME,NECBAL,EPFSRT(1,IRCRS),
     $EPFFX(1,IRCRS),BSRT(1,IRCRS),POWFRC(1,IRCRS),TS,TE,
     $G(LDTC  ),G(LMCDIR),G(LUNTYE),G(LUNTCO),G(LUNTSW),G(LUNTFA),
     $G(LUNTSA),G(LUNTCR),G(LUNTPU),G(LPVFAC),G(LEC   ),G(LPVTCY),
     $G(LTCCYC),G(LACCYC),G(LTCEOC),G(LACEOC),G(LEPF  ),G(LB    ),
     $G(LBATCS),G(LTCST ),G(LTREFU),G(LTMID ),ECRIT1,PVRTC,ECUPLM,
     $STCHLM,IDNUM,TO,G(LZEROH))
      IF(MX2EUS.LT.4) GO TO 110
C     PRINT ELAME TABLE IN UNITS OF $ &/OR $/MWHE
      L=LEC-1
      IF(.NOT.RELCST) GO TO 80
      WRITE(WT,901)
      WRITE(WT,919) IRCRS,IDNUM
      WRITE(WT,902) PVRTC,(G(L+J),J=1,NCYCIN)
      WRITE(WT,918) (ECUPLM(J),J=1,NCYCIN)
      WRITE(WT,903) (J,J=1,NCYCIN)
      WRITE(WT,904) (ELAME(1,J),J=1,NCYCIN)
```

```
      WRITE(WT,905) (ELAME(2,J),J=1,NCYCIN)
      WRITE(WT,906) (ELAME(3,J),J=1,NCYCIN)
      WRITE(WT,907) (ELAME(4,J),J=1,NCYCIN)
      DO 50 K=6,MX2EUS,2
      WRITE(WT,906) (ELAME(K-1,J),J=1,NCYCIN)
   50 WRITE(WT,907) (ELAME(K   ,J),J=1,NCYCIN)
   80 DO 100 N=1,NCYCIN
      NE2=NOESX2(N)
      DO 90 I=4,NE2,2
   90 ELAME(I-2,N)=(ELAME(I-2,N)-ELAME(I,N))/(ELAME(I-3,N)-ELAME(I-1,N)
     $      + 1.E-20)
  100 ELAME(NE2,N)=1.E20
      IF(.NOT.INCCST) GO TO 110
      WRITE(WT,911)
      WRITE(WT,919) IRCRS,IDNUM
      WRITE(WT,902) PVRTC,(G(L+J),J=1,NCYCIN)
      WRITE(WT,918) (ECUPLM(J),J=1,NCYCIN)
      WRITE(WT,903) (J,J=1,NCYCIN)
      WRITE(WT,904) (ELAME(1,J),J=1,NCYCIN)
      WRITE(WT,915) (ELAME(2,J),J=1,NCYCIN)
      WRITE(WT,916) (ELAME(3,J),J=1,NCYCIN)
      WRITE(WT,917) (ELAME(4,J),J=1,NCYCIN)
      DO 60 K=6,MX2EUS,2
      WRITE(WT,907) (ELAME(K-1,J),J=1,NCYCIN)
   60 WRITE(WT,917) (ELAME(K   ,J),J=1,NCYCIN)
  110 CONTINUE
      RETURN
  900 FORMAT('1'/'  THIS ITERATION USES',I5,' LOCATIONS IN G ARRAY',
     $' COMPARED TO THE',I5,' AVAILABLE' /'0')
  901 FORMAT('1',T25,'* * * * * REACTOR TOTAL COSTS RELATIVE TO R.T.C.'
     $,' FOR ECBAL (1000 P.V.$) * * * * *')
  902 FORMAT('0',T10,'REACTOR TOTAL COST FOR BALANCED EC''S (ECBAL)  =',
     $F12.3,'  10**3P.V.$'/'0 ECBAL',14F9.1/(12X,12F9.1))
  903 FORMAT('0 CYCLE',14(I6,3X)/(12X,12(I6,3X)))
  904 FORMAT('0  EC   ',14F9.2/(12X,12F9.2))
  905 FORMAT(' DELRTC',14F9.2/(12X,12F9.2))
```

```
  906 FORMAT('0 ETC. ',14F9.2/(12X,12F9.2))
  907 FORMAT('       ',14F9.2/(12X,12F9.2))
  911 FORMAT('1',T25,'* * * * * INCREMENTAL REACTOR TOTAL COST',
     $'  (P.V.$/MWHE) * * * * * ')
  915 FORMAT(' INCCST',14F9.4/(12X,12F9.4))
  916 FORMAT('  ETC. ',14F9.2/(12X,12F9.2))
  917 FORMAT('       ',14F9.4/(12X,12F9.4))
  918 FORMAT('DECUPLM',14F9.1/(12X,12F9.1))
  919 FORMAT('+INDEX=',I3,' IDNO=',I5)
      END
      SUBROUTINE REDCOR
C     READ INPUT DATA FOR INCORE
C     QKCORE VERSION 12-15-72
      COMMON/ARDATA/IDNO(15),NAME(15),MWCAP(15),EFFNET(15),IRCRKA(15),
     $IFULKA(15),NOZONE(15),ZONKG(15),DECRIT(15),DESTCH(15),NCYCFX(15),
     $EPFFX(20,15),EPFSRT(10,15),BSRT(10,15),FABINV(10,15),SRCINV(10,15)
     $,POWFRC(10,15),RCRCON(18,15),FULCON(48,5),EFFINC(15)
      COMMON/FXDDAT/MXZONE,MXCYTC,MXRCRS,MXRCRK,MXFULK,IRCRS,IRCRK,IFULK
     $,NRCRS,NRCRK,NFULK,EFF,XF,XW,TXRATE,PVRATE,TBASE,DTPRE,DTPST,
     $DTY2F6,CCRATE,FCOR,FFAB,FSAR,FCRE,NCYCIN,NCYCXS,NCYCTO,NZONE,NZP,
     $ZONEKG,ECHDOV,EFFAV,MWS
      COMMON/PRINTS/RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS,RD,WT
      LOGICAL RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS
      INTEGER RD,WT
      COMPLEX*16 HD(7)/' $/LB U308  ',' $/KG U CONV',' $/KG SWU   ',
     $' $/KG FAB   ',' $/KG SH&REP',' $/KG U CONV',' $/GM FIS.PU'/
      DATA $INCO$,$ENDB$/'INCO','END '/
      DIMENSION RCRKTL(20,15),FULKTL(20,5)
      DIMENSION X(20),ECTITL(20),A0(7),A1(7),A2(7),XX(20)
      READ(RD,903) XX
      WRITE(WT,931) XX
      IF(XX(1).NE.$INCO$) CALL ICERRS('REDCOR',3)
      READ (RD,901) NUECON,NURCRS,NURCRK,NUFULK,RELCST,INCCST,BALCST,
     $NBLCST,PIRDAT,PBATCS
      WRITE(WT,902) NUECON,NURCRS,NURCRK,NUFULK,RELCST,INCCST,BALCST,
     $NBLCST,PIRDAT,PBATCS
```

```
      IF(NURCRS.GT.MXRCRS.OR.NURCRK.GT.MXRCRK.OR.NUFULK.GT.MXFULK)
     $  CALL ICERRS('REDCOR',5)
      IF(NUECON.LE.0) GO TO 20
C     READ ECONOMIC DATA
      READ(RD,903) ECTITL
      READ(RD,908) XF,XW,TXRATE,PVRATE,TBASE,DTPRE,DTPST,DTY2F6
      X(8)=DTPRE*365.
      X(9)=DTPST*365.
      X(10)=DTY2F6*365.
      CCRATE=PVRATE/(1.-TXRATE)
      DO 10 I=1,7
      READ(RD,908) A0(I),A1(I),A2(I),F
      IF(I.EQ.2)  FCOR=F
      IF(I.EQ.4)  FFAB=F
      IF(I.EQ.5)  FSAR=F
      IF(I.EQ.6)  FCRE=F
   10 CONTINUE
      X(11)=100.*XF
C     INITIALIZE & SET POINTERS WHERE POSSIBLE
      DUMMY=PVINIT(PVRATE)
      CALL INIT2(A0,A1,A2,DTPRE,DTPST,TBASE,X)
      DUMMY=SETUVL(DTY2F6,FCOR,XF,XW)
      X(12)=UF6VAL(X(11),X(1),X(2),X(3))
   20 WRITE(WT,905) ECTITL
      WRITE(WT,907) XF,XW,TXRATE,PVRATE,TBASE,DTPRE,DTPST,DTY2F6,
     $CCRATE,X(8),X(9),X(10)
      WRITE(WT,909) FCOR,FFAB,FSAR,FCRE,(A0(I),A1(I),A2(I),X(I),
     $FD(I),I=1,7),X(12)
      IF(NURCRS.LE.0) GO TO 40
C     READ REACTOR PHYSICAL INFO.
      NRCRS=NURCRS
      CALL ERASE(EPFFX,20*15)
      DO 30 I=1,NRCRS
      READ(RD,910) IDNO(I),NAME(I),MWCAP(I),IRCRKA(I),IFULKA(I),
     $NOZONE(I),ZONKG(I),EFFNET(I),DECRIT(I),DESTCH(I),EFFINC(I)
      IF(EFFINC(I).LT.0.2) EFFINC(I)=EFFNET(I)
```

```
      READ(RD,911) N,(EPFFX(1+J,I),J=1,N)
      NCYCFX(I)=N
      N=NOZONE(I)
      IF(N.GT.MXZONE) CALL ICERRS('REDCOR',5)
   30 READ(RD,912) (EPFSRT(J,I),BSRT(J,I),FABINV(J,I),SRCINV(J,I),
     $POWFRC(J,I),J=1,N)
   40 WRITE(WT,913) NRCRS
      DO 60 I=1,NRCRS
      WRITE(WT,914)I,IDNO(I),NAME(I),MWCAP(I),IRCRKA(I),IFULKA(I),
     $NOZONE(I),ZONKG(I),EFFNET(I),DECRIT(I),DESTCH(I),EFFINC(I)
      N=NCYCFX(I)
      WRITE(WT,915)N,(EPFFX(1+J,I),J=1,N)
      N=NOZONE(I)
      WRITE(WT,916)(J,EPFSRT(J,I),BSRT(J,I),FABINV(J,I),SRCINV(J,I),
     $POWFRC(J,I),J=1,N)
      SUM=0.0
      DO 50 J=1,N
   50 SUM=SUM+POWFRC(J,I)
      IF(ABS(SUM-1.).GT.1.E-5) CALL ICERRS('REDCOR',9)
   60 CONTINUE
      IF(NURCRK.LE.0) GO TO 70
C     READ REACTOR EMPIRICAL CONSTANTS
      NRCRK=NURCRK
      READ (RD,917)   ((RCRKTL(K,I),K=1,20),(RCRCON(J,I),J=1,18),
     $I=1,NRCRK)
   70 WRITE(WT,918) (I,(RCRKTL(K,I),K=1,20),(RCRCON(J,I),J=1,18),
     $I=1,NRCRK)
      IF(NUFULK.LE.0) GO TO 80
C     READ FUEL EMPIRICAL CONSTANTS
      NFULK=NUFULK
      READ (RD,919)   ((FULKTL(K,I),K=1,20),(FULCON(J,I),J=1,48),
     $I=1,NFULK)
   80 WRITE(WT,920) (I,(FULKTL(K,I),K=1,20),(FULCON(J,I),J=1,48),
     $I=1,NFULK)
      READ(RD,903) XX
      WRITE(WT,932) XX
```

```
      IF(XX(1).NE.$ENDB$) CALL ICERRS('REDCOR',3)                        QKCR0469
      RETURN                                                             QKCR0470
  901 FORMAT(4I5,6L1)                                                    QKCR0471
  902 FORMAT(' * * * * *  INCORE HAS BEEN ENTERED THRU ICNPUT TO READ',  QKCR0472
     $' CORE INPUT DATA * * * * *'/'0'/'0      NUECCN     NURCRS     NURCRK QKCR0473
     $    NUFULK     RELCST     INCCST     BALCST     NBLCST     PIRDAT',   QKCR0474
     $'     PBATCS'/4I10,9L10)                                           QKCR0475
  903 FORMAT(20A4)                                                       QKCR0476
  905 FORMAT('0'/'0',T35,'* * * * * ECONOMIC  DATA * * * * *'//          QKCR0477
     $T10,1H',20A4,1H')                                                  QKCR0478
  907 FORMAT('0       XF         XW    TXRATE    PVRATE     TBASE',      QKCR0479
     $'     DTPRE     DTPST    DTY2F6'/8F10.5,'  YEARS'/T22,'CCRATE =',  QKCR0480
     $F10.5,T51,3F10.2,'  DAYS')                                         QKCR0481
  908 FORMAT(8F10.3)                                                     QKCR0482
  909 FORMAT('0 REPROCESSING YIELDS:',T51,'UNIT COST ESCALATION COEFFS:' QKCR0483
     $,'  COST = A0 + A1*TPAY + A2*TPAY**2'/T6,'FCOR      FFAB      FSAR   QKCR0484
     $      FCRE',T67,'A0         A1         A2         COST @ TREFUL=TBASE QKCR0485
     $'/4F10.4,7(T61,F10.3,F10.4,F10.5,F15.3,A8,A4/),                     QKCR0486
     $'0COST OF NAT. UF6 AT ',                                           QKCR0487
     $'TREFUL=TBASE (I.E., TPAY=TREFUL-DTPRE) :',F10.3,' $/KG U AS UF6')  QKCR0488
  910 FORMAT(I5,1X,A4,4I5,5F10.2)                                        QKCR0489
  911 FORMAT(I2,F8.3,7F10.3/(8F10.3))                                    QKCR0490
  912 FORMAT(5F10.3)                                                     QKCR0491
  913  FORMAT('1'     /'0',T20,'* * * * *  REACTOR ENGINEERING DATA FOR', QKCR0492
     $' THE',I3,' REACTORS * * * * *'/)                                  QKCR0493
  914 FORMAT('0'/'0 REACTOR DATA FOR IRCRS =',I3/T7,'IDNO      NAME',    QKCR0494
     $T27,'MWCAP      IRCRK      IFULK    NOZONE ZONEKG    EFFNET',      QKCR0495
     $'    DECRIT    DESTCH    EFFINC'/I10,A10,4I10,F10.2,F10.5,2F10.2,  QKCR0496
     $F10.5)                                                             QKCR0497
  915 FORMAT('0  NCYCFX =',I3,'  EPFFX =',(1X,12(F7.4,',')))             QKCR0498
  916 FORMAT('0  CONDITION OF CORE WHEN SIMULATION COMMENCES AT CYCLE 1' QKCR0499
     $,' :'/T8,'ZONE      EPF         B      FABINV    SRCINV    POWFRC'/ QKCR0500
     $(I10,F10.4,F10.1,2F10.2,F10.4))                                    QKCR0501
  917 FORMAT((20A4,3(/6E12.6)))                                          QKCR0502
  918 FORMAT('1'     /'0',T30,'* * * * * REACTOR  EMPIRICAL  DATA  ',    QKCR0503
     $'* * * * *'/'0'/('0    TYPE ',I3,10X,20A4/3(1P6E15.6/)))           QKCR0504
```

```
  919 FORMAT((20A4,8(/6E12.6)))
  920 FORMAT('1'    /'0',T30,'* * * * * FUEL  EMPIRICAL DATA  * * * * *'
     $/'0'/('0   TYPE ',I3,10X,20A4/8(1P6E15.6/)))
  931 FORMAT('1  FIRST INCORE DATA CARD :',2H ',20A4,1H')
  932 FORMAT('   LAST  INCORE DATA CARD :',2H ',20A4,1H')
      END
      SUBROUTINE FULSIM(MXESX2,NOESX2,ELAME,NECBAL,EPFSRT,EPFFX,BSRT,
     $POWFRC,TS,TE,DTC,MODIRR,UNTYEL,UNTCOR,UNTSWU,UNTFAB,UNTSAR,UNTCRE,
     $UNTPUV,PVFACT,EC,PVTCYC,TCCYC,ACCYC,TCEOCD,ACEOCD,EPF,B,BATCST,C,
     $TREFUL,TMID,ECE1A1,BALRTC,ECUPLM,STCHLM,IDNUM,TO,ZEROHT)
C     PERFORMS FUEL IRRAD. SIMUL. FOR ALL SETS OF E'S
C     QKCORE VERSION 12-15-72
      COMMON/FXDDAT/MXZONE,MXCYTO,MXRCRS,MXRCRK,MXFULK,IRCRS,IRCRK,IFULK
     $,NRCRS,NRCRK,NFULK,EFF,XF,XW,TXRATE,PVRATE,TBASE,DTPRE,DTPST,
     $CTY2F6,CCRATE,FCOR,FFAB,FSAR,FCRE,NCYCIN,NCYCXS,NCYCTO,NZONE,NZP,
     $ZONEKG,ECHDOV,EFFAV,MWS
      COMMON/PRINTS/RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS,RD,WT
      LOGICAL RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS
      INTEGER RD,WT
      DIMENSION NOESX2(NCYCTO),ELAME(MXESX2,NCYCTO),NECBAL(NCYCTO),
     $EPFSRT(NZONE),EPFFX(NCYCTO),BSRT(NZONE),POWFRC(NZONE),TS(NCYCTO),
     $TE(NCYCTO),DTC(NCYCTO),MODIRR(NCYCTO),UNTYEL(NCYCTO),UNTCOR(NCYCTO
     $),UNTSWU(NCYCTO),UNTFAB(NCYCTO),UNTSAR(NCYCTO),UNTCRE(NCYCTO),
     $UNTPUV(NCYCTO),PVFACT(NCYCTO),EC(NCYCTO),TCCYC(NCYCTO),
     $ACCYC(NCYCTO),TCEOCD(NCYCTO),ACEOCD(NCYCTO),EPF(NZP,NCYCTO),
     $B(NZP,NCYCTO),BATCST(NZP,NCYCTO),PVTCYC(NCYCTO),C(NZP)
      DIMENSION TO(NCYCTO),ZEROHT(NCYCTO)
      DIMENSION TREFUL(NCYCTO),TMID(NCYCTO),ECUPLM(NCYCTO)
      REAL*8 TCCYC,PVTCYC,SUM,RTC,BALRTC
      INTEGER CYC,FRSCYC,FRSBAT,BAT
      ZTON=ZONEKG*0.001
      BALRTC=0.0D0
      CALL CONSTS(NCYCTO,TS,TE,UNTYEL,UNTCOR,UNTSWU,UNTFAB,UNTSAR,UNTCRE
     $,UNTPUV,DTC,PVFACT,TBASE,TREFUL,TMID)
      FRSCYC=1
      LSTCYC=NCYCTO
```

```
      FRSBAT=1                                                          QKCR0541
      LSTBAT=NCYCIN+MINO(NZCNE,NCYCXS)                                  QKCR0542
      CALL ERASE(ECUPLM,NCYCTO)                                         QKCR0543
      ECUPLM(1)=ECE1A1+STCHLM                                           QKCR0544
      DO 10 CYC=1,NCYCTO                                                QKCR0545
      ZEROHT(CYC)=MWS*(1./EFFAV-1./EFF)*TO(CYC)*8.760                   QKCR0546
      IF(EPFFX(CYC)) 2,1,3                                              QKCR0547
    1 MODIRR(CYC)=1                                                     QKCR0548
      GO TO 10                                                          QKCR0549
    2 MODIRR(CYC)=2                                                     QKCR0550
      GO TO 10                                                          QKCR0551
    3 MODIRR(CYC)=3                                                     QKCR0552
   10 EC(CYC)=ELAME(2*NECBAL(CYC)-1,CYC)                                QKCR0553
      MODIRR(1)=0                                                       QKCR0554
      IF(NCYCXS.EQ.0)   GO TO 20                                        QKCR0555
      $=1.E20                                                           QKCR0556
      NCP=NCYCIN+1                                                      QKCR0557
      DO 15 I=NCP,NCYCTO                                                QKCR0558
      PVTCYC(I)=$                                                       QKCR0559
      TCCYC(I)=$                                                        QKCR0560
      ACCYC(I)=$                                                        QKCR0561
      DO 15 N=1,NZP                                                     QKCR0562
   15 BATCST(N,I)=$                                                     QKCR0563
   20 DO 50 CYC=FRSCYC,LSTCYC                                           QKCR0564
      IF(PIRDAT.AND.CYC.EQ.FRSCYC) WRITE(WT,901) IRCRS,IDNUM            QKCR0565
      MODE=MODIRR(CYC)                                                  QKCR0566
      ECESPC=EC(CYC)                                                    QKCR0567
      ECTSPC=ECESPC/EFF+ZEROHT(CYC)                                     QKCR0568
      EPFSPC=ABS(EPFFX(CYC))                                            QKCR0569
      IF(PIRDAT) WRITE(WT,900) CYC,ECTSPC,ECESPC                        QKCR0570
      IF(CYC.EQ.1) GO TO 30                                             QKCR0571
      CALL NXTIRR(MODE,ECTSPC,EPFSPC,ZONEKG,NZCNE,EPF(1,CYC),B(1,CYC),  QKCR0572
     $B(2,CYC+1),PIRDAT,WT,ECTCRT)                                      QKCR0573
      IF (BALRTC.EQ.0.0D0.AND.MODE.NE.1.AND.ECUPLM(CYC).EQ.0.0)         QKCR0574
     $      ECUPLM(CYC)=EFF*(ECTCRT-ZEROHT(CYC))+STCHLM                 QKCR0575
      IF(MODE.EQ.2) EC(CYC)=EFF*(ECTSPC-ZEROHT(CYC))                    QKCR0576
```

```
22 E(1,CYC+1)=0.0
   DO 25 I=1,NZONE
25 EPF(I+1,CYC+1)=EPF(I,CYC)
   GO TO 50
30 DO 40 I=1,NZONE
   EPF(I,1)=EPFSRT(I)
40 B(I,1)=BSRT(I)
   EPF(NZP,1)=1.E20
   B(NZP,1)=1.E20
   CALL FRSIRR(MODE,ECTSPC,ZONEKG,ECE1A1/EFF+ZEROHT(CYC),NZONE,
  $EPF(1,CYC),B(1,CYC),B(2,CYC+1),PIRDAT,WT,POWFRC)
   GO TO 22
50 CONTINUE
   IF(PBATCS) WRITE(WT,902) IRCRS,IDNUM
   DO 70 BAT=FRSBAT,LSTBAT
   NIRRAD=MIN0(NZONE,BAT)
   CALL CSTBAT(BAT,NIRRAD)
   NIP=NIRRAD+1
   M1=MAX0(0,NZONE-BAT)
   M2=MAX0(BAT-NZONE,0)
   DO 60 I=1,NIP
60 BATCST(I+M1,I+M2)=C(I)*ZTON
70 TCEOCD(BAT)=TCEOCD(BAT)*ZTON
   BATCST(NZP,1)=0.0
   DO 85 CYC=1,NCYCIN
   SUM=0.0D0
   DO 80 I=1,NZP
80 SUM=SUM+BATCST(I,CYC)
   TCCYC(CYC)=SUM
85 ACCYC(CYC)=SUM/EC(CYC)
   IF(FRSBAT.LE.NCYCIN.AND.LSTBAT.GE.NCYCIN)
  $    TCCYC(NCYCIN)=TCCYC(NCYCIN)*(1.-ECHDOV/EC(NCYCIN))
   RTC=0.0D0
   DO 90 CYC=1,NCYCIN
   PVTCYC(CYC)=TCCYC(CYC)*PVFACT(CYC)
90 RTC=RTC+PVTCYC(CYC)
```

```
      IF(BALRTC.GT.0.0D0) GO TO 100
      IF(.NOT.BALCST.AND..NOT.NBLCST) GO TO 150
      CALL PRTTOP(NZP,NCYCTO,WT,TS,TE,DTC,MODIRR,UNTYEL,UNTCOR,UNTSWU,
     $UNTFAB,UNTSAR,UNTCRE,UNTPUV,PVFACT,EC,PVTCYC,TCCYC,ACCYC,TCEOCD,
     $ACEOCD,EPF,B,BATCST,TREFUL,TMID,NCYCIN,ECHDOV,IRCRS,IDNUM)
      GO TO 110
  100 IF(.NOT.NBLCST) GO TO 200
  110 CALL PRTBTM(RTC)
      IF(BALRTC.GT.0.0D0) GO TO 200
  150 BALRTC=RTC
      DO 180 N=1,NCYCIN
      NCYC=NCYCIN-N+1
      ECBAL=EC(NCYC)
      IF(MODIRR(NCYC).EQ.2) GO TO 180
      NE2=NOESX2(NCYC)
      FRSCYC=NCYC
      LSTCYC=NCYCTO
      FRSBAT=FRSCYC
      LSTBAT=NCYCIN+MIN0(NZONE,NCYCXS)
      DO 170 J=1,NE2,2
      EC(NCYC)=ELAME(J,NCYC)
      RTC=BALRTC
      IF(EC(NCYC).EQ.ECBAL) GO TO 160
      GO TO 190
  160 ELAME(J+1,NCYC)=RTC-BALRTC
  170 CONTINUE
  180 EC(NCYC)=ECBAL
      RETURN
  190 GO TO 20
  200 GO TO 160
  900 FORMAT('0'/'OCYCLE ',I2, 9X,'ECTSPC =',F10.2,' GWHTH',10X,
     $'ECESPC =',F10.2,' GWHE ')
  901 FORMAT('1'/'OCYCLE IRRADIATION DATA FOR ',I3,' TH REACTOR  (IDNO =
     $',I5,')  :'/)
  902 FORMAT('1'/'OBATCH COSTS FOR '           ,I3,' TH REACTOR  (IDNO =
     $',I5,')  :'/)
```

```
      END
      SUBROUTINE CONSTS(NCYCTO,TS,TE,UNTYEL,UNTCOR,UNTSWU,UNTFAB,
     $UNTSAR,UNTCRE,UNTPUV,DTC,PVFACT,TBASE,TREFUL,TMID)
C     CALCULATE CONSTANT DATA FOR THIS ITERATION THRU INCORE
C     QKCORE VERSION 3-04-72
      DIMENSION TS(NCYCTO),TE(NCYCTO),COST(7)
      DIMENSION DTC(NCYCTO),PVFACT(NCYCTO),UNTYEL(NCYCTO),UNTCOR(NCYCTO)
     $,UNTSWU(NCYCTO),UNTFAB(NCYCTO),UNTSAR(NCYCTO),UNTCRE(NCYCTO),
     $UNTPUV(NCYCTO),TREFUL(NCYCTO),TMID(NCYCTO)
      REAL*8 PVPER$
      TEMP=TS(NCYCTO+1)
      TS(NCYCTO+1)=TE(NCYCTO)+TS(NCYCTO)-TE(NCYCTO-1)
      I=1
      TSRT=TS(1)
  100 CALL UNTCOS(TSRT,COST)
      UNTYEL(I)=COST(1)
      UNTCOR(I)=COST(2)
      UNTSWU(I)=COST(3)
      UNTFAB(I)=COST(4)
      UNTSAR(I)=COST(5)
      UNTCRE(I)=COST(6)
      UNTPUV(I)=COST(7)
      TSRTNX=0.5*(TE(I)+TS(I+1))
      DTC(I)=TSRTNX-TSRT
      TMD=TSRT+0.5*DTC(I)
      PVFACT(I)=PVPER$(TMD,TBASE)
      TREFUL(I)=TSRT
      TMID(I)=TMD
      TSRT=TSRTNX
      I=I+1
      IF(I.LE.NCYCTO) GO TO 100
      TS(NCYCTO+1)=TEMP
      RETURN
      END
      SUBROUTINE NXTIRR(MODE,ECSPC,EPFSPC,ZONEKG,NZONE,EPF,BGIN,BFNL,
     $      PRINT,NPRNTR,ECTCRT)
```

```
C     PERFORMS SIMULATION OF NEXT IRRADIATION                           QKCR0685
C     QKCORE VERSION 3-04-72                                            QKCR0686
C     ALL EC'S IN UNITS OF GWHTH FROM THE ENTIRE REACTOR                QKCR0687
C     MODE = 0  FIRST CYCLE WHICH IS ALREADY UNDERGOING IRRADIATION     QKCR0688
C               THEREFORE ONLY FRSIRR CAN BE CALLED                     QKCR0689
C          = 1  EC SPECIFIED; EPFNEW TO BE DETERMINED                   QKCR0690
C          = 2  EPFNEW SPECIFIED; EC TO BE DETERMINED                   QKCR0691
C          = 3  EC & EPFNEW SPECIFIED (STRETCHOUT OR EARLY REFUELING)   QKCR0692
      IMPLICIT REAL (K)                                                 QKCR0693
      LOGICAL PRINT                                                     QKCR0694
      DIMENSION EPF(NZONE),BGIN(NZONE),BFNL(NZONE)                      QKCR0695
      DIMENSION K8(10),SIGA(10),DB(10),F(10)                            QKCR0696
      COMMON/IRRDAT/K8INR,ECOUT,ECRIT,UTIL,EC$24Z,K8,SIGA,DB,F          QKCR0697
      DATA EPFMIN,EPFMAX/1.5,5.0/                                       QKCR0698
      IF(MODE.EQ.0) CALL ICERRS('NXTIRR',12)                            QKCR0699
      K8INR=0.0                                                         QKCR0700
      UTIL=1.0                                                          QKCR0701
      FSUM=0.0                                                          QKCR0702
      IF(NZONE.EQ.1) GO TO 30                                           QKCR0703
      TEMP=0.0                                                          QKCR0704
      DO 20 N=2,NZONE                                                   QKCR0705
      E=EPF(N)                                                          QKCR0706
      B=BGIN(N)                                                         QKCR0707
      K8(N)=FK8(E,B)                                                    QKCR0708
      SIGA(N)=FSIGA(E)                                                  QKCR0709
      F(N)=SIGA(N)*K8(N)                                                QKCR0710
      FSUM=FSUM+F(N)                                                    QKCR0711
   20 TEMP=TEMP+K8(N)                                                   QKCR0712
      K8INR=TEMP/(NZONE-1)                                              QKCR0713
   30 IF(MODE.GT.1) GO TO 80                                            QKCR0714
      K81=FK8NEW(ECSPC,K8INR)                                           QKCR0715
      K8(1)=K81                                                         QKCR0716
      EPF1=FEPF(K81)                                                    QKCR0717
      IF(EPF1.GT.EPFMAX.OR.EPF1.LT.EPFMIN) GO TO 100                    QKCR0718
      EPF(1)=EPF1                                                       QKCR0719
      ECOUT=ECSPC                                                       QKCR0720
```

```
      ECRIT=ECSPC                                                       QKCR0721
   40 EPF1=EPF(1)                                                       QKCR0722
      PHI=FPHI(EPF1,K8INR)                                              QKCR0723
      SIGA(1)=FSIGA(EPF1)                                               QKCR0724
      F(1)=SIGA(1)*K8(1)*PHI                                            QKCR0725
      TEMP=1./(FSUM+F(1))                                               QKCR0726
   50 EC$24Z=ECOUT/(24.*ZONEKG*0.001)                                   QKCR0727
      DO 70 N=1,NZONE                                                   QKCR0728
      F(N)=F(N)*TEMP                                                    QKCR0729
      DB(N)=F(N)*EC$24Z                                                 QKCR0730
   70 BFNL(N)=BGIN(N)+DB(N)                                             QKCR0731
      ECTCRT=ECRIT                                                      QKCR0732
      IF(.NOT.PRINT) RETURN                                             QKCR0733
      NZ=NZONE                                                          QKCR0734
      WRITE(NPRNTR,900) MODE,ECSPC,EPFSPC,ZONEKG,K8INR,EC$24Z,ECOUT,ECRI QKCR0735
     $T,UTIL,(N,EPF(N),BGIN(N),DB(N),BFNL(N),K8(N),SIGA(N),F(N),N=1,NZ)  QKCR0736
      RETURN                                                            QKCR0737
   80 EPF(1)=EPFSPC                                                     QKCR0738
      E=EPF(1)                                                          QKCR0739
      B=BGIN(1)                                                         QKCR0740
      K81=FK8(E,B)                                                      QKCR0741
      K8(1)=K81                                                         QKCR0742
      ECRIT=FECOUT(K81,K8INR)                                           QKCR0743
      IF(MODE.GT.2) GO TO 85                                            QKCR0744
      ECOUT=ECRIT                                                       QKCR0745
      ECSPC=ECOUT                                                       QKCR0746
      GO TO 40                                                          QKCR0747
   85 ECOUT=ECSPC                                                       QKCR0748
      UTIL=ECOUT/ECRIT                                                  QKCR0749
C     CHECK FOR WARNING OF TOO MUCH STRETCHOUT                          QKCR0750
      IF(UTIL.GT.1.25) CALL ICERRS('NXTIRR',1)                          QKCR0751
C     CHECK FOR WARNING OF VERY LITTLE IRRADIATION                      QKCR0752
      IF(UTIL.LT.0.75) CALL ICERRS('NXTIRR',2)                          QKCR0753
      GO TO 40                                                          QKCR0754
C     COMPLETE FIRST CYCLE IRRADIATION                                  QKCR0755
      ENTRY FRSIRR(MODE,ECSPC,ZONEKG,EKRIT,NZONE,EPF,BGIN,BFNL,PRINT,   QKCR0756
```

```
     $NPRNTR,POWFRC)                                                     QKCR0757
      DIMENSION POWFRC(NZONE)                                            QKCR0758
      ECRIT=EKRIT                                                        QKCR0759
      EPFSPC=0.0                                                         QKCR0760
      K8INR=0.0                                                          QKCR0761
      ECOUT=ECSPC                                                        QKCR0762
      UTIL=ECOUT/ECRIT                                                   QKCR0763
      TEMP=1.0                                                           QKCR0764
      DO 90 N=1,NZONE                                                    QKCR0765
      F(N)=POWFRC(N)                                                     QKCR0766
      K8(N)=0.0                                                          QKCR0767
   90 SIGA(N)=0.0                                                        QKCR0768
      GO TO 50                                                           QKCR0769
  100 MODE=3                                                             QKCR0770
      EPFSPC=EPFMIN                                                      QKCR0771
      IF(EPF1.GT.EPFMAX) EPFSPC=EPFMAX                                   QKCR0772
      CALL ICERRS('NXTIRR',11)                                           QKCR0773
      GO TO 80                                                           QKCR0774
  900 FORMAT(                         '0MODE =',I2,10X,'ECSPC =',F10.2,' GWHTH QKCR0775
     $',10X,'EPFSPC =',F10.5,10X,'ZONEKG =',F10.1/'0  K8INR =',F10.6,5X, QKCR0776
     $'EC$24Z =',F10.4,5X,'ECOUT =',F10.2,5X,'ECRIT =',F10.2,5X,'UTIL =' QKCR0777
     $,F10.6/'0 N     EPF         BGIN         DB       BFNL        K8', QKCR0778
     $6X,'  SIGA        F'/(I3,F10.6,3F10.4,3F10.6))                     QKCR0779
      END                                                                QKCR0780
      SUBROUTINE CSTBAT(LSTIRR,NIRRAD)                                   QKCR0781
C     CALCULATE COST OF BATCH DISCHARGED AT END OF LSTIRR AND WHICH WAS  QKCR0782
C     IRRADIATED NIRRAD TIMES WITHIN THE SIMULATION                      QKCR0783
C     QKCORE VERSION 12-15-72                                            QKCR0784
      IMPLICIT REAL (K)                                                  QKCR0785
      COMMON/FXDDAT/MXZONE,MXCYTO,MXRCRS,MXRCRK,MXFULK,IRCRS,IRCRK,IFULK QKCR0786
     $,NRCRS,NRCRK,NFULK,EFF,XF,XW,TXRATE,PVRATE,TBASE,DTPRE,DTPST,      QKCR0787
     $DTY2F6,CCRATE,FCOR,FFAB,FSAR,FCRE,NCYCIN,NCYCXS,NCYCTO,NZONE,NZP,  QKCR0788
     $ZONEKG,ECHDOV,EFFAV,MWS                                            QKCR0789
      COMMON/PRINTS/RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS,RD,WT      QKCR0790
      LOGICAL RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS                  QKCR0791
      INTEGER RD,WT                                                      QKCR0792
```

```
      INTEGER FRSIRR                                                    QKCR0793
      REAL IUF6,IFAB,ISRC,IPUV,ITOT                                     QKCR0794
      REAL*8 PVPER$                                                     QKCR0795
      LOGICAL NEWFUL                                                    QKCR0796
      GO TO 5                                                           QKCR0797
      ENTRY INIT3(FABINV,SRCINV,EPF,DTC,B,UNTYEL,UNTCOR,UNTSWU,UNTFAB,  QKCR0798
     $UNTSAR,UNTCRE,UNTPUV,TCEOCD,ACEOCD,A,BC,DBC,DT,KGU,EPNOW,UVALUE,  QKCR0799
     $GMP,IUF6,IFAB,ISRC,IPUV,ITOT,C,AC)                                QKCR0800
      DIMENSION FABINV(NZONE),SRCINV(NZONE)                             QKCR0801
      DIMENSION EPF(NZP,NCYCTO),DTC(NCYCTO),B(NZP,NCYCTO),UNTYEL(NCYCTO)QKCR0802
     $,UNTCOR(NCYCTO),UNTSWU(NCYCTO),UNTFAB(NCYCTO),UNTSAR(NCYCTO),     QKCR0803
     $UNTCRE(NCYCTO),UNTPUV(NCYCTO),TCEOCD(NCYCTO),ACEOCD(NCYCTO)       QKCR0804
      DIMENSION A(NZP,15),BC(NZP),DBC(NZP),DT(NZP),KGU(NZP),            QKCR0805
     $EPNOW(NZP),UVALUE(NZP),GMP(NZP),IUF6(NZP),IFAB(NZP),ISRC(NZP),    QKCR0806
     $IPUV(NZP),ITOT(NZP),C(NZP),AC(NZP)                                QKCR0807
      CCPRE=DTPRE*CCRATE                                                QKCR0808
      CCPST=DTPST*CCRATE                                                QKCR0809
      FABLOS=(1.-FFAB)/FFAB                                             QKCR0810
      SARLOS=1.-FSAR                                                    QKCR0811
      CRELOS=FSAR*(1.-FCRE)                                             QKCR0812
      RETURN                                                            QKCR0813
    5 CALL ERASE(A,15*NZP)                                              QKCR0814
      NI=NIRRAD                                                         QKCR0815
      NIP=NIRRAD+1                                                      QKCR0816
      NIM=NIRRAD-1                                                      QKCR0817
      NEWFUL=.TRUE.                                                     QKCR0818
      FRSIRR=LSTIRR-NZONE+1                                             QKCR0819
      IF(FRSIRR.GT.1) GO TO 10                                          QKCR0820
      FRSIRR=1                                                          QKCR0821
      NEWFUL=.FALSE.                                                    QKCR0822
   10 EPFAB=EPF(NZONE,LSTIRR)                                           QKCR0823
      JCYCL=FRSIRR-1                                                    QKCR0824
      JZONE=NZONE-NI                                                    QKCR0825
      DO 20 I=1,NI                                                      QKCR0826
      A(I,1)=I                                                          QKCR0827
      JCYCL=JCYCL+1                                                     QKCR0828
```

```
      JZONE=JZONE+1                                                     QKCR0829
      DT(I)=DTC(JCYCL)                                                  QKCR0830
      DBC(I)=B(JZONE+1,JCYCL+1)-B(JZONE,JCYCL)                          QKCR0831
   20 BC(I)=B(JZONE,JCYCL)                                              QKCR0832
      BC(NIP)=BC(NI)+DBC(NI)                                            QKCR0833
      DBC(NIP)=0.0                                                      QKCR0834
      DT (NIP)=0.0                                                      QKCR0835
      DO 30 I=1,NIP                                                     QKCR0836
      BURN=BC(I)                                                        QKCR0837
      KGU(I)=FKGUR(EPFAB,BURN)                                          QKCR0838
      EPNOW(I)=FEPB(EPFAB,BURN)                                         QKCR0839
      GMP(I)=FKGPU(EPFAB,BURN)*1000.                                    QKCR0840
      UVALUE(I)=UF6VAL(EPNOW(I),UNTYEL(FRSIRR),UNTCCR(FRSIRR),UNTSWU(FRS QKCR0841
     $IRR))                                                             QKCR0842
      IUF6(I)=UVALUE(I)*KGU(I)                                          QKCR0843
   30 IPUV(I)=UNTPUV(LSTIRR)*GMP(I)                                     QKCR0844
      IFAB(1)=UNTFAB(FRSIRR)+FABLOS*IUF6(1)                             QKCR0845
      ISRC(1)=0.0                                                       QKCR0846
      IF(NEWFUL) GO TO 40                                               QKCR0847
      JZONE=NZONE-NI+1                                                  QKCR0848
      IFAB(1)=FABINV(JZONE)                                             QKCR0849
      ISRC(1)=SRCINV(JZONE)                                             QKCR0850
   40 ISRC(NIP)=UNTSAR(LSTIRR)*(KGU(NIP)+0.001*GMP(NIP))                QKCR0851
     $   + SARLOS*(IUF6(NIP)+IPUV(NIP))                                 QKCR0852
     $   + UNTCRE(LSTIRR)*KGU(NIP)*FSAR+CRELOS*IUF6(NIP)                QKCR0853
      DISRC=ISRC(NIP)-ISRC(1)                                           QKCR0854
      OVDB=1./(BC(NIP)-BC(1))                                           QKCR0855
      DO 50 I=1,NIP                                                     QKCR0856
      F=(BC(I)-BC(1))*OVDB                                              QKCR0857
      IFAB(I)=IFAB(1)*(1.-F)                                            QKCR0858
      ISRC(I)=ISRC(1)+DISRC*F                                           QKCR0859
   50 ITOT(I)=IUF6(I)+IFAB(I)-ISRC(I)+IPUV(I)                           QKCR0860
      DO 60 I=1,NI                                                      QKCR0861
   60 C(I)=ITOT(I)-ITOT(I+1)+(ITOT(I)+ITOT(I+1))*0.5*DT(I)*CCRATE       QKCR0862
      IF(LSTIRR.GT.NIRRAD) C(1)=C(1)+ITOT(1)*CCPRE                      QKCR0863
      C(NIP)=ITOT(NIP)*CCPST                                            QKCR0864
```

```
      TWOTPV=0.0                                                        QKCR0865
      N=1                                                               QKCR0866
   70 IF(N.EQ.NI) GO TO 80                                              QKCR0867
      N=N+1                                                             QKCR0868
      TWOTPV=TWOTPV+DT((N+1)/2)                                         QKCR0869
      GO TO 70                                                          QKCR0870
   80 TCBAT=0.0                                                         QKCR0871
      PVBRN=0.0                                                         QKCR0872
      DO 90 I=1,NI                                                      QKCR0873
      PVPER=PVPER$(-0.5*TWOTPV,0.0)                                     QKCR0874
      TCBAT=TCBAT+C(I)*PVPER                                            QKCR0875
      PVBRN=PVBRN+DBC(I)*PVPER                                          QKCR0876
      AC(I)=C(I)/(24.*EFFAV*DBC(I))                                     QKCR0877
   90 TWOTPV=TWOTPV-DT(I)-DT(I+1)                                       QKCR0878
      TCBAT=TCBAT+C(NIP)*PVPER$(-0.5*TWOTPV,0.0)                        QKCR0879
      AC(NIP)=1.E20                                                     QKCR0880
      PVELEC=PVBRN*24.*EFFAV                                            QKCR0881
      ACEOCD(LSTIRR)=TCBAT/PVELEC                                       QKCR0882
      TCEOCD(LSTIRR)=TCBAT                                              QKCR0883
      LST=LSTIRR                                                        QKCR0884
      IF(PBATCS) WRITE(WT,900) LSTIRR,NIRRAD,TCBAT,PVELEC,ACEOCD(LST),  QKCR0885
     $((A(I,J),J=1,15),I=1,NIP)                                         QKCR0886
      RETURN                                                            QKCR0887
  900 FORMAT('0'/10X,'COST OF BATCH DISCH. AT END OF CYCLE',I3,         QKCR0888
     $' WHICH WAS IRRADIATED FOR',I3,' CYCLES OF THE SIMULATION :'/     QKCR0889
     $' TOTAL COST OF DISCHARGED BATCH (P.V. AT MID-PT. OF MIDDLE',     QKCR0890
     $' IRRAD.)    =',F8.2,' $/KGFAB'/  ' AVERAGE COST FOR THE',F8.2,    QKCR0891
     $' MWHE/KGFAB (ALSO P.V.)',T70 ,'=',F8.4,' $/MWHE'/                QKCR0892
     $'  I     BC       DBC       DT      KGUR    ENRICH  UF6VAL  GMSPU',QKCR0893
     $7X,'UF6     FAB      SRC      PUV    TOTINV   COST    AVGCST'/    QKCR0894
     $'    MWD/KGFAB MWD/KGFAB  YRS   KG/KGFAB  W/O235  $/KGUF6 GM/KGFAB'QKCR0895
     $,T67,'----- DOLLARS  PER  KILOGRAM  FABRICATED -----  $/MWHE'/    QKCR0896
     $(F4.0,F8.4,F9.4,F8.4,F9.6,F8.4,F8.2,F7.3,3X,6F8.2,F8.4))          QKCR0897
      END                                                               QKCR0898
      SUBROUTINE PRTTOP(NZP,NCYCTO,WT,TS,TE,DTC,MODIRR,UNTYEL,UNTCOR,   QKCR0899
     $UNTSWU,UNTFAB,UNTSAR,UNTCRE,UNTPUV,PVFACT,EC,PVTCYC,TCCYC,ACCYC,  QKCR0900
```

```
     $TCEOCD,ACEOCD,EPF,B,BATCST,TREFUL,TMID,NCYCIN,ECHDCV,IRCRS,IDNUM)       QKCR0901
C      PRINT TOP OF FULSIM RESULT TABLE                                        QKCR0902
C      QKCORE VERSION 3-04-72                                                  QKCR0903
      REAL*8 TCCYC,PVTCYC,RTC                                                 QKCR0904
      DIMENSION TS(NCYCTO),                                                   QKCR0905
     $TE(NCYCTO),DTC(NCYCTO),MODIRR(NCYCTO),UNTYEL(NCYCTO),UNTCOR(NCYCTO     QKCR0906
     $),UNTSWU(NCYCTO),UNTFAB(NCYCTO),UNTSAR(NCYCTO),UNTCRE(NCYCTO),          QKCR0907
     $UNTPUV(NCYCTO),PVFACT(NCYCTO),EC(NCYCTO),TCCYC(NCYCTO),                 QKCR0908
     $ACCYC(NCYCTO),TCEOCD(NCYCTO),ACEOCD(NCYCTO),EPF(NZP,NCYCTO),           QKCR0909
     $B(NZP,NCYCTO),BATCST(NZP,NCYCTO),PVTCYC(NCYCTO),TREFUL(NCYCTO),        QKCR0910
     $TMID(NCYCTO)                                                            QKCR0911
      COMPLEX*16 HD(60),BLANK,E1,NP1,B1,$1                                    QKCR0912
      INTEGER WT,FRS                                                          QKCR0913
      DATA HD/'     CYCLE',' TIRSRT  YRS',' TIREND  YRS',' DTREF.  YRS',     QKCR0914
     $'     MODIRR',' UNTYEL $/LBY',' UNTCOR $/KGC',' UNTSWU $/KGS',         QKCR0915
     $' UNTFAB $/KGF',' UNTSAR $/KGS',' UNTCRE $/KGC',' UNTPUV $/GMP',       QKCR0916
     $' PVFACT @TMID','OEC GWHE',' PVTCYC  K$',' TCCYC   K$',' ACCYC  $/     QKCR0917
     $MWH',' TCEOCD  K$',' ACEOCD $/MWH','OTREFUL  YRS',' TMID    YRS'/      QKCR0918
      DATA BLANK,E1,B1,$1,NP1/' ',' EPF(1)',' EGIN(1)','  BATCST(1)',         QKCR0919
     $'  (N+1)'/                                                              QKCR0920
      FRS=22                                                                  QKCR0921
      LST=FRS+3*NZP-1                                                         QKCR0922
      NZ=NZP-1                                                                QKCR0923
      DO 10 I=FRS,LST                                                         QKCR0924
   10 HD(I)=BLANK                                                             QKCR0925
      HD(FRS)=E1                                                              QKCR0926
      HD(FRS+NZ )=NP1                                                         QKCR0927
      HD(FRS+NZP)=B1                                                          QKCR0928
      HD(LST-NZP)=NP1                                                         QKCR0929
      HD(LST-NZ )=$1                                                          QKCR0930
      HD(LST)=NP1                                                             QKCR0931
      WRITE(WT,930) IRCRS,IDNUM                                               QKCR0932
      WRITE(WT,901)  HD( 1),(I,I=1,NCYCTO)                                    QKCR0933
      WRITE(WT,914)  HD(20),TREFUL                                            QKCR0934
      WRITE(WT,914)  HD( 4),DTC                                               QKCR0935
      WRITE(WT,914)  HD( 2),TS                                                QKCR0936
```

```
      WRITE(WT,914)  HD( 3),TE                                          QKCR0937
      WRITE(WT,901)  HD( 5),MCDIRR                                      QKCR0938
      WRITE(WT,914)  HD(21),TMID                                        QKCR0939
      WRITE(WT,915)  HD(13),PVFACT                                      QKCR0940
      WRITE(WT,902)                                                     QKCR0941
      WRITE(WT,912)  HD( 6),UNTYEL                                      QKCR0942
      WRITE(WT,912)  HD( 7),UNTCCR                                      QKCR0943
      WRITE(WT,912)  HD( 8),UNTSWU                                      QKCR0944
      WRITE(WT,912)  HD( 9),UNTFAB                                      QKCR0945
      WRITE(WT,912)  HD(10),UNTSAR                                      QKCR0946
      WRITE(WT,912)  HD(11),UNTCRE                                      QKCR0947
      WRITE(WT,912)  HD(12),UNTPUV                                      QKCR0948
      RETURN                                                            QKCR0949
      ENTRY PRTBTM(RTC)                                                 QKCR0950
C     PRINT BOTTOM OF FULSIM RESULT TABLE                               QKCR0951
      WRITE(WT,931)  IRCRS,IDNUM,RTC,NCYCIN,ECHDOV                      QKCR0952
      WRITE(WT,901)  HD( 1),(I,I=1,NCYCTO)                              QKCR0953
      WRITE(WT,912)  HD(14),EC                                          QKCR0954
      WRITE(WT,912)  HD(15),PVTCYC                                      QKCR0955
      WRITE(WT,912)  HD(16),TCCYC                                       QKCR0956
      WRITE(WT,914)  HD(17),ACCYC                                       QKCR0957
      WRITE(WT,912)  HD(18),TCECCD                                      QKCR0958
      WRITE(WT,914)  HD(19),ACECCD                                      QKCR0959
      IX=FRS-1                                                          QKCR0960
      WRITE(WT,900)                                                     QKCR0961
      DO 20 M=1,NZP                                                     QKCR0962
   20 WRITE(WT,914)  HD(M+IX),   (EPF(M,I),I=1,NCYCTO)                  QKCR0963
      IX=IX+NZP                                                         QKCR0964
      WRITE(WT,900)                                                     QKCR0965
      DO 30 M=1,NZP                                                     QKCR0966
   30 WRITE(WT,914)  HD(M+IX),     (B(M,I),I=1,NCYCTO)                  QKCR0967
      IX=IX+NZP                                                         QKCR0968
      WRITE(WT,900)                                                     QKCR0969
      DO 40 M=1,NZP                                                     QKCR0970
   40 WRITE(WT,912)  HD(M+IX),(BATCST(M,I),I=1,NCYCTO)                  QKCR0971
      RETURN                                                            QKCR0972
```

```
  900 FORMAT('0')
  901 FORMAT(A8,A5,12(I7,3X)/(30X,10(I7,3X)))
  902 FORMAT('    UNIT COSTS CA. TREFUL')
  912 FORMAT(A8,A5,12F10.2/(30X,10F10.2))
  913 FORMAT(A8,A5,12F10.3/(30X,10F10.3))
  914 FORMAT(A8,A5,12F10.4/(30X,10F10.4))
  915 FORMAT(A8,A5,12F10.5/(30X,10F10.5))
  930 FORMAT('1'/'0'/'0'/T35,'INDEX = ',I3,10X,'IDNO =',I5/
     $               '0',T20,' * * * * *  FULSIM RESULT TABLE FOR BALANCE
     $C SET OF EC''S  * * * * *'/'0'/'0')
  931 FORMAT('1'     /'0'/T35,'INDEX = ',I3,10X,'IDNO =',I5/
     $           '0* * *  REACTOR TOTAL COST TC HORIZON OF INTEREST :',
     $F12.3,' (10**3 DOLLARS P.V. TO TBASE) * * * '/   ' ( HORIZON IS IN
     $ CYCLE ',I2,' WITH ',F10.2,' GWHE HELDOVER FOR POST-HORIZON PRODUC
     $TION IN THAT CYCLE )'/)
      END
      FUNCTION EMPRCL(F,R)
C     INITIALIZE EMPIRICAL EQUATIONS
C     QKCORE VERSION 3-04-72
      IMPLICIT REAL(K)
      DIMENSION F(100),R(25)
C     EVALUATE QUADRATIC Q=C0 + C1*X + C2*X**2
      Q(C0,C1,C2,X)=(C2*X+C1)*X+C0
C     UNIT FUEL SIMULATION EQUATIONS
C     SETUP INVERSION OF K8NEW TO GET EPFNEW
      EMPRCL=0.0
      IF(F(3).EQ.0.0) GO TO 10
      CEF1=-0.5*F(2)/F(3)
      CEF2=(F(2)**2-4.*F(3)*F(1))/(4.*F(3)**2)
      CEF3=1./F(3)
      CEF4=0.0
      K8MAX=Q(F(1),F(2),F(3),CEF1)-1.E-5
      RETURN
   10 CEF1=-F(1)/F(2)
      CEF2=0.0
      CEF3=0.0
```

```
      CEF4=1/F(2)
      K8MAX=100.
      RETURN
C***************************
      ENTRY FK8(EPF,B)
      FK8=Q(Q(F(1),F(2),F(3),EPF),Q(F(4),F(5),F(6),EPF),
     $Q(F(7),F(8),F(9),EPF),B)
      RETURN
C***************************
      ENTRY FKGUR(EPF,B)
      FKGUR=Q(Q(F(10),F(11),F(12),EPF),Q(F(13),F(14),F(15),EPF),
     $Q(F(16),F(17),F(18),EPF),B)
      RETURN
C***************************
      ENTRY FEPB(EPF,B)
      DUM=Q(Q(F(19),F(20),F(21),EPF),Q(F(22),F(23),F(24),EPF),
     $Q(F(25),F(26),F(27),EPF),B)
      FEPB=EPF*EXP(-B*DUM)
      RETURN
C***************************
      ENTRY FKGPU(EPF,B)
      DUM=Q(Q(F(28),F(29),F(30),EPF),Q(F(31),F(32),F(33),EPF),
     $Q(F(34),F(35),F(36),EPF),B)
      ULAM=Q(F(37),F(38),F(39),EPF)
      PLAM=Q(F(40),F(41),F(42),EPF)
      FKGPU=DUM*(EXP(-B*ULAM)-EXP(-B*PLAM))
      RETURN
C***************************
      ENTRY FSIGA(EPF)
      FSIGA=F(43)+F(44)*EPF
      RETURN
C***************************
      ENTRY FEPF(K8NEW)
      FEPF=100.
      IF(K8NEW.GT.K8MAX) RETURN
      FEPF=CEF1-SQRT(CEF2+CEF3*K8NEW)+CEF4*K8NEW
```

```
      RETURN
C*****************************
C     REACTOR IRRADIATION SIMULATION EQUATIONS
      ENTRY FK8NEW(EC,K8INR)
      DK=K8INR-1.
      FK8NEW=1.+Q(R(1),R(2),R(3),EC)+Q(0.0,R(4)+R(6)*EC,R(5),DK)
      RETURN
C*****************************
      ENTRY FPHI(EPF,K8INR)
      DK=K8INR-1.
      FPHI=1./(1.+EPF*Q(R(8),R(9),R(10),EPF)+Q(R(7),R(11),R(12),DK))
      RETURN
C*****************************
      ENTRY FECOUT(K8NEW,K8INR)
C     REWRITE K8NEW AS AA*EC**2 +BB*EC+CC=0 AND SOLVE FOR EC
      DK=K8INR-1.
      AA=R(3)
      BB=R(2)+R(6)*DK
      CC=Q(1.+R(1)-K8NEW,R(4),R(5),DK)
      IF(AA.EQ.0.0) GO TO 20
      FECOUT=BB*(SQRT(1.-4.*AA*CC/BB**2)-1.)/(AA+AA)
      RETURN
   20 FECOUT=-CC/BB
      RETURN
      END
      SUBROUTINE UNTCOS(TREFUL,COST)
C     CALCULATE ESCALATED UNIT COSTS
C     QKCORE VERSION 3-04-72
      DIMENSION COST(7),A0(7),A1(7),A2(7),B0(7),B1(7),B2(7)
      GO TO 10
      ENTRY INIT2(B0,B1,B2,DTPRE,DTPST,TREFUL,COST)
C     INITIALIZE POINTERS AND DATA
      DO 5 I=1,7
      A0(I)=B0(I)
      A1(I)=B1(I)
    5 A2(I)=B2(I)
```

```
   10 TPRE=TREFUL-DTPRE
      TPST=TREFUL+DTPST
      DO 20 I=1,4
   20 COST(I)=(TPRE*A2(I)+A1(I))*TPRE+A0(I)
      DO 30 I=5,7
   30 COST(I)=(TPST*A2(I)+A1(I))*TPST+A0(I)
      RETURN
      END
      FUNCTION UF6VAL(/EP/,/UNTYEL/,/UNTCOR/,/UNTSWU/)
C     CALCULATES VALUE OF ENRICHED URANIUM AS $/KG UF6
C     QKCORE VERSION 3-04-72
      REAL*8 PVPER$
      PHI(X)=(X+X-1.)*ALOG(X/(1.-X))
      SOVP(X)=PHI(X)+A+B*X
      FOVP(XP)=(XP-XW)*OVDX
      CF=C1*UNTYEL+UNTCOR
      XP=0.01*EP
      UF6VAL=CF*FOVP(XP)+UNTSWU*SOVP(XP)
      RETURN
      ENTRY SETUVL(DTY2F6,FCOR,XF,XW)
C     SETUP URAN. VALUE EQUATION
      C1=2.59985*PVPER$(-DTY2F6,0.0)/FCOR
      OVDX=1./(XF-XW)
      PHIXF=PHI(XF)
      PHIXW=PHI(XW)
      A=(-XF*PHIXW + XW*PHIXF)*OVDX
      B=(PHIXW-PHIXF)*OVDX
      SETUVL=0.0
      RETURN
      END
      FUNCTION PVPER$(T,TBASE)
C     CALCULATE PRESENT VALUE AT TIME T OF 1$ AT TIME TBASE
C     QKCORE VERSION 3-04-72
      REAL*8 PVPER$,LN1PX
      PVPER$=DEXP(-LN1PX*(T-TBASE))
      RETURN
```

```
      ENTRY PVINIT(PVRATE)
C     PRE-CALCULATE LOG OF (1+X) IN UNITS OF INVERSE YEARS
      LN1PX=DLOG(1.D0+PVRATE)
      PVINIT=LN1PX
      RETURN
      END
      SUBROUTINE ICERRS(SUBR,JERR)
C     WRITES OUT ALL ERROR MESSAGES FOR INCORE
C     QKCORE VERSION 3-04-72
      COMMON/PRINTS/RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS,RD,WT
      LOGICAL RELCST,INCCST,BALCST,NBLCST,PIRDAT,PBATCS
      INTEGER RD,WT
      INTEGER ERRCOD
      REAL*8 SUBR,$QUIT$
      DATA NPRINT/0/,$QUIT$/' QUIT'/,ERRCOD/0/,MAXERR/16777216/
C     MAXERR=16**6
      IERR=JERR
  100 ERRCOD=MOD(ERRCOD,MAXERR)
      ERRCOD=16*ERRCOD+IERR
      NPRINT=NPRINT+1
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12),IERR
    1 WRITE(WT,901) SUBR,ERRCOD,NPRINT
      RETURN
    2 WRITE(WT,902) SUBR,ERRCOD,NPRINT
      RETURN
    3 WRITE(WT,903) SUBR,ERRCOD,NPRINT
      GO TO 1000
    4 WRITE(WT,904) SUBR,ERRCOD,NPRINT
      GO TO 1000
    5 WRITE(WT,905) SUBR,ERRCOD,NPRINT
      GO TO 1000
    6 WRITE(WT,906) SUBR,ERRCOD,NPRINT
      RETURN
    7 WRITE(WT,907) SUBR,ERRCOD,NPRINT
      GO TO 1000
    8 WRITE(WT,908) SUBR,ERRCOD,NPRINT,NPRINT
```

```
      STOP
    9 WRITE(WT,909) SUBR,ERRCOD,NPRINT
      RETURN
   10 WRITE(WT,910) SUBR,ERRCOD,NPRINT
      GO TO 1000
   11 WRITE(WT,911) SUBR,ERRCOD,NPRINT
      RETURN
   12 WRITE(WT,912) SUBR,ERRCOD,NPRINT
      GO TO 1000
 1000 NPRINT=NPRINT+1
      WRITE(WT,999) NPRINT
      SUBR=$QUIT$
      IERR=8
      GO TO 100
  901 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'    UTIL.GT. 1.25  VERY LONG STRETCHOUT         ',
     $T131,'*',/,' ',130('*'),I2)
  902 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'     UTIL.LT.0.75  VERY EARLY REFUELING         ',
     $T131,'*',/,' ',130('*'),I2)
  903 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'INPUT DECK HAS IMPROPER SEQUENCE &/OR CARD     ',
     $T131,'*',/,' ',130('*'),I2)
  904 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'ARRAY G IN THIS VERSION IS TOO SMALL FOR THIS',
     $' PROBLEM                                      ',
     $T131,'*',/,' ',130('*'),I2)
  905 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'TOO MANY ZONES, REACTORS, OR SETS OF REACTOR ',
     $'&/OR FUEL CONSTANTS FOR THIS VERSION          ',
     $T131,'*',/,' ',130('*'),I2)
  906 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'     WARNING :  NCYCTC WAS NOT EQUAL TC NCYCIN',
     $'+NCYCXS WHEN INCORE ENTERED                   ',
     $T131,'*',/,' ',130('*'),I2)
  907 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
```

```
     $'REACTOR FOR CASE IDNUM NOT READ IN BY ICNPUT  ',
     $T131,'*',/,' ',130('*'),I2)
 908 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'QKCCRE ENCOUNTERED STCP CARD, ICERRS CALLED CNCE TOO OFTEN OR O',
     $'THER FATAL ERROR.', T131,'*'/' * DURING THIS ENTIRE RUN, ICERRS',
     $' PRINTEC A TOTAL CF ',I3,' ERROR MESSAGES JUST LIKE (AND ',
     $'INCLUDING) THIS ONE',
     $T131,'*',/,' ',130('*'),I2)
 909 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'SUMMATION OF POWFRC DIFFERS FROM 1.0 BY MORE ',
     $'THAN 10**-5                                ',
     $T131,'*',/,' ',130('*'),I2)
 910 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'ELAME TABLE IS TOC LARGE FOR THIS VERSICN.   ',
     $T131,'*',/,' ',130('*'),I2)
 911 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'FEED ENRICHMENT AS DETERMINED IN NXTIRR UNDER',
     $' MODE 1, OUTSIDE PRESCRIBED LIMITS ',
     $T131,'*',/,' ',130('*'),I2)
 912 FORMAT(/' ',130('*')/,' * SUBR. ',A6,' HAS ERRCOD = ',Z8,' : ',
     $'NXTIRR CALLED WHEN MODE=0 (SHOULD CALL FRSIRR',
     $')                                         ',
     $T131,'*',/,' ',130('*'),I2)
 999 FORMAT(/' ',130('*')/,' * PREVIOUS ERROR SEVERE ENOUGH TO',
     $' INVALICATE FURTHER COMPUTATIONS. THEREFORE,',
     $' TERMINATING EXECUTICN.',
     $T131,'*',/,' ',130('*'),I2)
      END
************************************************************************ 00000000
*                                                                      * 00000010
*                   ASSEMBLER LANGUAGE SUBROUTINE ERASE                * 00000011
*                        WRITTEN BY JOHN W. KICSON                     * 00000012
*                      MIT DEPARTMENT OF METEOROLOGY                   * 00000014
*                                                                      * 00000016
*      TO SET ELEMENTS OF REAL OR INTEGER ARRAYS TC ZERO. A1,A2,...   * 00000020
*      ARE ARRAY NAMES AND N1,N2,... ARE INTEGER VALUES OR            * 00000030
```

```
*        EXPRESSIONS GIVING THE ARRAY SIZES.                                  *
**       I.E. - CALL ERASE(C,26*31,N,7*31,E,254)                             **
*                                                                             *
*******************************************************************************
ERASE    START 0
         SAVE  (14,12),,*
         BALR  12,0
         USING *,12
         SR    0,0
         SR    2,2              PARAMETER LIST INDEX=0
         L     6,=F'4'
E1       L     3,0(2,1)         LOAD 3 WITH ARRAY ADDRESS
         L     4,4(2,1)         LOAD 4 WITH ADDRESS OF ARRAY LENGTH
         L     7,0(4)           LOAD 7 WITH ARRAY LENGTH-1 TIMES 4
         SLA   7,2
         SR    7,6
         SR    5,5
E2       ST    0,0(5,3)         STORE ZERO
         BXLE  5,6,E2
         LTR   4,4              TEST FOR LAST ARGUMENT IN LIST
         BM    RETN
         A     2,=F'8'
         B     E1               PICK UP NEXT ARGUMENT PAIR
RETN     RETURN (14,12),T
         END
*******************************************************************************
```