THREE DIMENSIONAL EFFECTS IN ANALYSIS OF PWR
STEAM LINE BREAK ACCIDENT

by

Chon-Kwo Tsai, Mujid S. Kazimi, and Allan F. Henry

Energy Laboratory Report No. MIT-EL 85-004

March 1985

Energy Laboratory

and

Department of Nuclear Engineering

Massachusetts Institute of Technology
Cambridge, Mass.   02139

THREE DIMENSIONAL EFFECTS IN ANALYSIS OF PWR
STEAM LINE BREAK ACCIDENT

by

Chon-Kwo Tsai, Mujid S. Kazimi, and Allan F. Henry

March 1984

Three Dimensional Effects in Analysis of PWR Steam Line Break Accident

by

Chon-Kwo Tsai, Mujid S. Kazimi, and Allan F. Henry

ABSTRACT

A steam line break accident is one of the possible severe abnormal transients in a pressurized water reactor. It is required to present an analysis of a steam line break accident in the Final Safety Analysis Report (FSAR) of a nuclear power reactor. Only simple one-dimensional core simulations are currently applied to such accident analysis. However, the asymmetric characteristics of a steam break accident may require more detailed local information in order to determine the potential fuel damage resulting from the transient. TITAN, a coupled (neutronics and thermal-hydraulics) code with state-of-the-art neutronics and thermal-hydraulics models, is therefore modified and applied to steam line break accident simulations.

The capabilities that are added to the code for a steam line break analysis include multi-region core inlet temperature forcing function, total inlet coolant flow rate boundary condition, total inlet coolant flow rate transient simulation capability, boron tracking equations, flow/coolant temperature transient plus control rod transient option, and one-dimensional, fully implicit numerical scheme for thermal-hydraulics calculations. The modifications to TI-TAN are tested with a ten-channel PWR model. For inlet coolant temperature transients (one of the transients involved in a steam line break accident) test calculations lead to the conclusion that there is no significant difference between the results of closed- and open-channel calculations until boiling occurs.

A ten-channel model with two partially inserted control rods is employed for the transient simulations. Steady state conditions are obtained first by both open- and closed-channel calculations. Results show that cross flow between channels is insignificant. Thus, the one-dimensional, fully implicit numerical scheme for the thermal-hydraulics equations is useful to speed up the calculations. More than half of the computational effort is saved by using this scheme compared to the semi-implicit numerical scheme.

Two extreme situations relevant to a steam line break accident are investigated: (1) Significant boiling due to severe depressurization when no return to power exists. (2) Return to power with no boiling because of high coolant temperature feedback coefficients. It is concluded that even after boiling occurs, the global parameters, such as total power and assembly power, still show no significant difference between the closed- and open-channel calcualtions. However, the local parameters, such as nodal power, void fraction and MDNBR, reveal differences between the two calculations. Results show that the open-channel calculation predicts lower MDNBR values as compared with the results of closed-channel calculation in a vapor generation process, since the coolant is driven away from of the hot spots. On the other hand, during a vapor condensation process, closed-channel calculations predict lower MDNBR results, since no cross flow is allowed to accelerated the condensation process.

A closed-channel, uniform inlet coolant temperature transient calculation is performed. The results verify the necessity for a three-dimensional calculation of the accident simulations, since no boiling was predicted by the one-dimensional calculation throughout the simulation period.

A high fission cross section coolant temperature coefficient ($\frac{\partial \Sigma_f}{\partial T_c}$) results in rapid return to power while no boiling exists. No significant difference is found between the results of closed- and open-channel calculations in the return to power studies. However, potential damage of the fuel rod due to high power peaking in cold region is found; the fuel center line temperature predicted by TITAN exceeds the grain-growth temperature of UO_2 which is a design limit.

The time step size decoupling of the neutronics and thermal-hydraulics calculations leads to a significant savings in computational effort. However, this option is appropriate only for thermal-hydraulic driven transients, since it has been programmed so that the time step size for neutronics calculation is an integer multiple of the thermal-hydraulics time step size.

Publications of the TITAN Project

Papers

1. D. P. Griggs, M. S. Kazimi and A. F. Henry, "TITAN: An Advanced Three-Dimensional Neutronics/Thermal-Hydraulics Code for LWR Safety Analysis," Proc. ANS Conf. on Advances in Reactor Physics and Core Thermal Hydraulics, Sept. 21-24, 1982, Kiamesha Lake, NY.

2. D. Griggs, C. Tsai, A. Henry and M. Kazimi, "TITAN: An Advanced Three Dimensional Coupled Code," Trans. Am. Nucl. Soc. 46, pp. 1984.

Technical Reports

1. D. P. Griggs, A. F. Henry and M. S. Kazimi, "Development of a Three-Dimensional Two-Fluid Code with Transient Neutronic Feedback for LWR Applications," MIT-EL 81-013, Energy Laboratory, M.I.T., April 1981 (NTIS #PB-82-180 217).

2. D. P. Griggs, M. S. Kazimi, and A. F. Henry, "Advanced Methods Development of LWR Transient Analysis, Final Report: 1981 - 1982," E-Lab Report No. MIT-EL 82-021, May 1982.

3. C. K. Tsai, D. P. Griggs, M. S. Kazimi and A. F. Henry, "Development and Quarter Core PWR Rod Ejection Accident Application of the TITAN Code, Final Report: 1982 - 1983," Energy Laboratory Report No. MIT-EL 83-007, June 1983.

4. C. K. Tsai, M. S. Kazimi and A. F. Henry, "TITAN Code Development for Application to a PWR Steam Line Break Accident," MIT-EL 84-014, July 1984.

5. D. P. Griggs, M. S. Kazimi and A. F. Henry, " TITAN: An Advanced Three Dimensional Coupled Neutronic/Thermal-Hydraulics Code for Light Water Nuclear Reactor Core Analysis," MIT-EL 84-011, June 1984.

6. C. K. Tsai, M. S. Kazimi, and A. F. Henry, "Three Dimensional Effects in Analysis of PWR Steam Line Break Accident," MIT-EL 85-004, March 1985.

iiia

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ix

NOMENCLATURE

A	area
a	constant in Eq. (1.1)
b	constant in Eq. (1.1)
c	heat capacity
D	diameter
e	internal energy
F	momentum exchange term
h	heat transfer coefficient
k	thermal conductivity
y	spatial coordinate
z	spatial coordinate

Greek

α	void fraction
Γ	interphase mass exchange term
γ	eigenvalue
μ	viscosity
ν	total number of neutrons produced by a fission
ρ	density
Σ	macroscopic cross section
σ	microscopic cross section
ϕ	neutron flux
χ	neutron probability density

Subscript

c	coolant
CHF	critical heat flux
f	fuel
f	fission

g neutron group number

gg' scattering from group g' to g

il interphase exchange to liquid

iv interphase exchange to vapor

l liquid

m Wallis quality

$msfb$ minimum stable film boiling

N   neutronics

p pressure

sat saturation condition

TH   thermal-hydraulics

tg total removal cross section at group g

tl turbulent liquid mixing

tv turbulent vapor mixing

u spatial coordinate

v vapor

v spatial coordinate

wl liquid to wall exchange

wv vapor to wall exchange dviqms nom del nom.lis; del nom.dvi; print nom.bit

end

Chapter 1

INTRODUCTION

Nuclear fission power generation is one of the important energy resources nowadays. In the foreseeable future, nuclear fission power will still play an important role in the energy industry. The reasons are the limited resources of oil, the political influence on the oil supply, the green house effect caused by CO_2 which is produced by burning coal, the acid rain caused by using low quality oil, etc. All the above considerations are now sensitive matters to the public. There are several other energy resources which are under developing such as the solar energy, the nuclear fusion energy, and the wind energy. Although all of them have almost unlimited resources, each of them has its own difficulties. Solar energy has the problem of low power density and is strongly affected by weather conditions. Nuclear fusion energy still has the challenge of the plasma confinement and dealing with the extremely high temperature plasma. Wind energy is affected by weather conditions just as solar energy is. Not everyplace has relatively constant, strong wind to give economic energy.

The conclusion from the above discussion is that the energy resource which can best satisfy the constantly increasing energy demand today is nuclear fission energy. Nuclear fission energy has been successfully commercialized in the past few decades. However, there remains an apprehension of potential accidents with nuclear fission technology. The Three Mile Island accident is perhaps the most famous and serious accident in the short history of the nuclear fission power industry. Therefore, to understand more detailed phenomena of nuclear fission reactor transients is an important task for nuclear engineers. The analysis of such transients can provide nuclear engineers and operators with more information to design a safer reactor and can suggest how to operate reactors more safely.

A digital computer code is one option for obtaining detailed information about a reactor transient (normal or abnormal). There are many existing computer codes developed for analyzing various aspects of nuclear reactor behavior. These codes deal with neutronics, thermal-hydraulics, structural mechanics, fuel management, etc. When the codes are categorized by reactor components, there are codes for analyzing reactor cores, steam generators (PWR), pressurizers (PWR), pumps and pipes, reactor vessels, containments, etc. Such codes appear to be the most practical and economic tools for analyzing reactor transients. One example of a transient analysis code is TITAN [1,2,3,4]. This is a coupled (neutronics and thermal-hydraulics) code for analyzing reactor core behavior. It has been successfully applied to a 47-channel PWR quarter core model for a rod ejection accident analysis [3,4].

The goal of the present report is to develop and apply the TITAN code to the analyses of three-dimensional effects associated with a steam line break (SLB) accident.

1.1 The Phenomena of Pressurized Water Reactor Steam Line Break Accidents

The Steam Line Break (SLB) accident is one of the serious transients which might occur in a nuclear reactor. The study of this transient is required in the Final Safety Analysis Report (FSAR) of a nuclear power plant. For light water reactors (LWR), the pressurized water reactor (PWR) and the boiling water reactor (BWR) are two typical designs. Since there is only a primary loop in BWR's, the steam line break accident for this class of LWR's is actually a kind of loss of coolant accident (LOCA). This is quite different from the PWR steam line break accident. In this report only the analysis of the PWR steam line break accident is presented.

A PWR steam line break accident is a very complicated transient. In general, the concerns of the SLB accident are the possible return to power condition and high local power peaking which might damage fuel rods.

Figure 1.1 shows a typical PWR primary and secondary loop. If there is a break somewhere in the steam line pipe, the large pressure difference between the secondary loop (6.9 MPa) and the containment (0.1 MPa) accelerates the secondary loop working fluid out of the break point. The velocity of the fluid therefore is much higher than that during normal condition. The heat transfer coefficient of the steam generator can be determined from the Nusselt number relation:

$$Nu = cRe^a Pr^b \qquad (1.1)$$

where

$$Nu = \frac{hD}{k}$$

$$Re = \frac{\rho v D}{\mu}$$

$$Pr = \frac{\mu c_p}{k}$$

From equation (1.1), we know that as the fluid velocity, v, increases, the heat transfer coefficient, h, also increases. The heat flux is determined by:
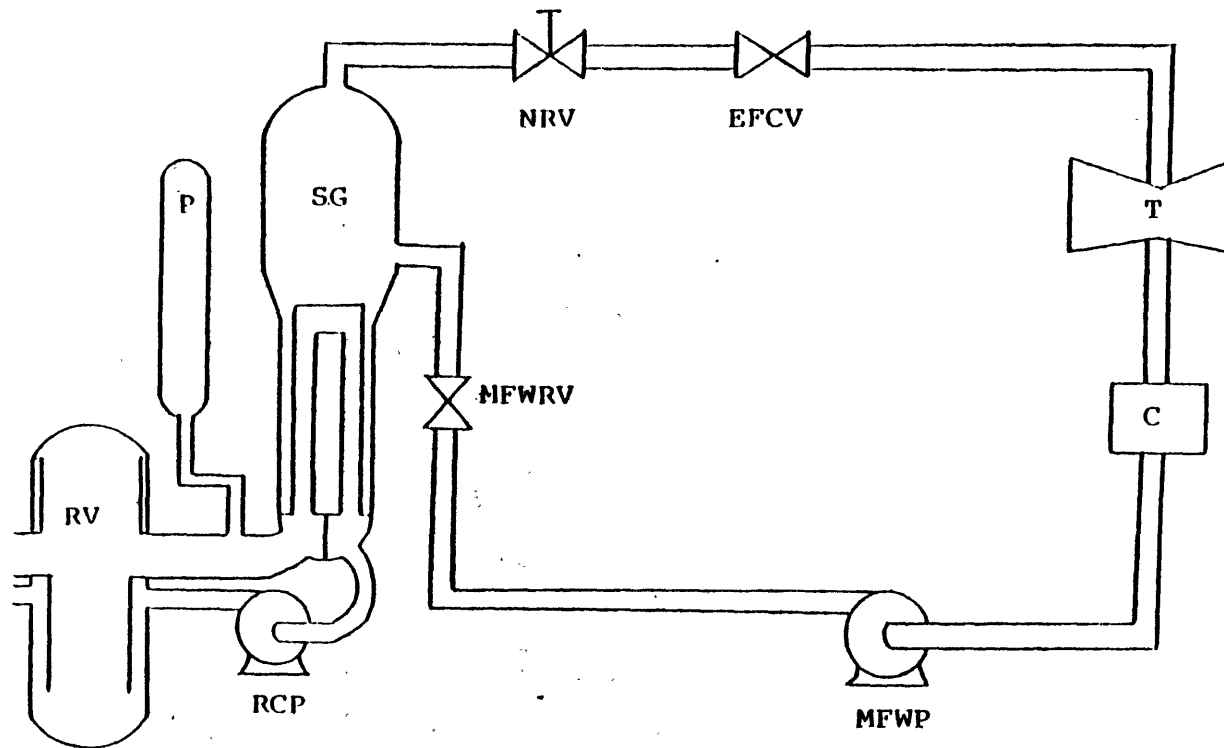
RV:	Reactor Vessel	P:	Pressurizer
SG:	Steam Generator	RCP:	Reactor Coolant Pump
NRC:	Non-Return Check Valve	EFCV:	Excess Flow Check Valve
T:	Turbine	C:	Condensor
MFWP:	Main Feedwater Pump	MFWRV:	Main Feedwater Regulatory Valve

Figure 1.1 A set of typical PWR primary and secondary loops.

$$q'' = h\Delta T \tag{1.2}$$

Therefore, the heat removal from the primary loop is much higher than under normal conditions. This is so called excess heat removal in a steam line break accident.

The thermal-hydraulic consequence of the excess heat removal is that the core inlet coolant temperature drops rapidly. This causes the primary coolant inventory to shrink and hence depressurizes the primary system. The low steam generator pressure results in reactor scram, and the reduction of core power generation enhances the rate of primary coolant inventory shrinkage. This process will continue until the steam blowdown out of the rupture steam pipe is reduced or terminated. This will happen even if no action is taken in response to the transient, since the buildup of the containment pressure by the steam from the rupture pipe decreases the pressure difference between the secondary loop and the containment. Essentially, an equilibrium state will be reached and terminate the blowdown.

The neutronic consequence of the excess heat removal is related to the low coolant temperature during the transient. Throughout most of core life the Fuel Temperature Reactivity Coefficient (Doppler effect) and the Moderator Temperature Reactivity Coefficient are negative. The Moderator Density Reactivity Coefficient is positive. The decrease of the coolant temperature decreases fuel rod temperature and increases the coolant density. All the three reactivity coefficients result in positive reactivity additions to the core. The scram response to the transient adds negative reactivity to the core to compensate the positive reactivity addition resulted from the accident. However, at high burnups the magnitude of the possible positive reactivity addition resulting from the cooldown of the reactor coolant system following an SLB accident is large, and may even offset the negative reactivity addition due to scram. As a result, recriticality and an unintentional power generation could occur. This is the so called return to power of the SLB accident. The restriction set by the NRC for performing an SLB accident analysis is the assumption that the Control Element Assembly (CEA) of highest worth fails and is not inserted into the core. The combination of this restriction and the return to power condition may result in high local power peaking, and fuel rod damage will be possible.

An important characteristic of an SLB accident is that it is an asymmetric problem. A typical PWR has three or four steam generators. Obviously, the coolant temperatures of the intact steam generator loops will be higher than that of the broken loop. Therefore, the core inlet coolant temperature will not be uniformly distributed.

There are many possible combinations of break location, power level, operating mode, and assumed single failure in an SLB accident. The limiting break location at the steam

generator nozzle was analysed in the FSAR for the Main Yankee Nuclear Power Plant [5]. In general, six operating modes can be identified [6]:

1. Refueling
2. Cold Shutdown
3. Hot Shutdown
4. Hot Standby
5. Critical
6. Power Operations

A return to power condition is possible only for modes 4, 5, and 6.

General Design Criteria in Appendix A to 10CFR50 [7] require that a single active component of any safety class system be assumed to fail. Systems which play an important role in the mitigation of an SLB accident containing active components whose performance would be degraded by the failure of a single active component are:

1. ECCS: Emergency Core Cooling System
2. FWIS: Feedwater Isolation System
3. AFWS: Auxiliary Feedwater System
4. MSIS: Main Steam Isolation System

The limiting significant single failure in each system is:

1. ECCS: Failure of the power supply to a High Pressure Safety Injection pump (HPSI). This significantly reduces the rate of boron negative reactivity addition.
2. FWIS: Failure of a Main Feedwater Regulatory Valve (MFWRV) to close on receipt of an isolation signal. This results in a large increase in the feedwater flow to the faulted steam generator.
3. AFWS: Failure of the auxiliary feedwater control valve in the line to faulted steam generator to close after receipt of the Excess Flow Check Valve (EFCV) trip.
4. MSIS: Failure of an EFCV to close is significant only for breaks located downstream of EFCVs, as it allows continuous blowdown of the faulted steam generator. Failure of the passive Non-Return Check Valve (NRV) to close is significant for breaks located upstream of EFCVs.

All calculations presented in this report are based on the features of a limiting SLB transient from full power [6]:

1. Break at the steam generator nozzle
2. Power operation mode
3. Failure of MFWRV to close
4. Electric Main Feedwater Pump (MFWP) continuously operating

5. Reactor Coolant Pump (RCP) trip by the operator at receipt of Safety Injection Actuation Signal (SIAS) plus 30 seconds.

1.2 Approaches to Analyzing Steam Line Break Accidents

A nuclear fission reactor is a highly sophisticated device. In the process of generating power, up to 54 radioactive isotopes are created (based on U. S. NRC, 1975 information). Some of them are radioactive gases such as I^{131}, Xe^{135}, and Kr^{85}. To prevent these radioactive fission products from entering the environment the integrity of fission product barriers should be maintained. For large power reactors, there are normally at least three of these barriers: the reactor fuel elements, the primary system envelope, and one or more containment structures. Should any accident occur, it is necessary to make sure that at least one of the barriers maintains its integrity.

For steam line break accidents, fuel rods have the greatest possibility of being damaged as discussed in Section 1.1. Thus, it is necessary to have detailed information about the core condition during the transient so that possible fuel rod damage may be determined. The results of an SLB accident analysis can assure that the reactor design is able to withstand the postulated accident without leaking out any radioactive isotopes.

Conceptually, one can always perform some experiments to obtain enough knowledge about the SLB accident. However, in reality it is almost impractical to do so, especially for full scale experiments. The amazing development of digital computers makes it possible to perform complex accident analysis with computer programs (computer codes). The physical behavior of cores has been successfully represented in mathematical forms. The thermal-hydraulic behavior can be modeled with the coolant fluid mass, momentum, energy conservation equations, and with the fuel rod heat conduction equation, and the neutronic behavior can be modeled with approximations to the neutron transport equation. With various numerical techniques, the resultant equations can be solved by using a digital computer.

In terms of calculation speed, computer codes are categorized as on-line (real time) and off-line codes. The major requirement of the on-line codes is to perform calculations equal to or even faster than the real time scale of a transient. This is important to operators since the results may be helpful in making decisions in response to the transient. In order to gain the calculation speed, accuracy must be sacrificed by using simplified models in the code. For study and design purposes, codes with models as close as possible to reality are required. This means that accuracy is the major concern. In addition, not only the global information but also the detailed local information of the core should be available with such kinds of

codes.

To study an SLB accident, several characteristics of a computer code are required. For the description of the accident in Section 1.1, both thermal-hydraulic and neutronic transients are involved. Thus a coupled code with a feedback model is necessary.

The asymmetric characteristic of the SLB accident requires three-dimensional modeling for both neutronic and thermal-hydraulic calculations. For a large, loosely coupled reactor core and an asymmetric perturbation of the reactor, the point kinetics model has been shown to be inaccurate [8]. Basically this is because of the constant flux shape assumption throughout a transient in the point kinetics model. In an SLB accident, the flux shape is tilted by scram and unevenly distributed reactivity insertion. One- or two-dimensional neutron diffusion models are not able to provide detailed local information throughout the transient. Thus a three-dimensional neutronics modeling is necessary for this asymmetric transient.

When HPSI starts working in the latter stage of an SLB accident, boron solution is injected into the core. Thus to model the effect of boron on the total neutron absorption cross section, a boron concentration model is required for an SLB transient analysis.

As to the boundary conditions: The total core mass flow rate is usually available rather than the individual flow velocity of each channel. For more accurate simulation, a total core inlet flow rate boundary condition should be applied. The code should also have the capability to accept a non-uniform inlet coolant temperature boundary condition in an SLB accident analysis. In addition, it should be possible for the code to simulate pressure transients, coolant temperature transients and coolant flow transients. Finally, the ability to simulate a thermal-hydraulic transient (i.e., temperature, flow transients) and a control rod transient (scram) should be available in the code.

1.3 Literature Review For Steam Line Break Accident Analysis

1.3.1 Loop Codes

Steam line break accident analysis is required in an FSAR of a nuclear fission power plant. Computer codes used are mostly loop codes, i.e., codes which perform calculations not only for the reactor core but also for steam generators, pressurizers, pumps, and plenums. Examples of such kind of codes are RELAP5 [9], RETRAN [10], and ALMOS [11]. Peeler et. al. performed audit calculations [12] of the limiting and steam line break transient

presented in the CESSAR FSAR [13,14] with the RELAP5/MOD1.5 code. The objective of the comparison was to confirm the acceptability of the C-E methodology for evaluating steam line break events. Figures 1.2 and 1.3 present the noding diagrams of the reactor vessel and the Nuclear Steam Supply System (NSSS). Figure 1.4 shows CESEC-III nodalization for the same problem. Only global results are presented for the reactor core behavior. Notice that two channels with twelve nodes in RELAP5 simulation and two channels with two ndoes in CESEC-III simulation were assigned for the reactor core. Even with these simple nodal schemes for the reactor core, one of the conclusions made by the authors stated: 'Asymmetric thermal-hydraulics plays an important role in determining the reactivity feedback'. To have more detailed and more accurate results about the reactivity feedback, it seems to be necessary to perform detailed calculations for the core with well developed core codes.

Another analysis performed by White et. al. [15] was to evaluate the system response configuration to the rupture of single tube or multiple tubes in one or both once-through steam generators (OTSGs). The RELAP5/MOD1, Cycle 18 code was employed as an analytical tool. The nodalization diagram for this analysis is shown in Figure 1.5. The objective of this paper was to present experience of using the RELAP5 code for SLB transient analysis and suggestions were made by the authors. It was concluded that 'RELAP5 code is a valueable analytical tool for application to steam generator tube rupture calculations for B&W plants. It has been found that the code results are comparable to other analytical tools, ...'. Notice that the nodalization for the reactor core was two channels with six nodes. One still cannot obtain detailed information from such kind of calculations.

Similar analysis for a B&W PWR lowered-loop design with RELAP5/MOD1.5 was performed by Bolander et. al. [16]. These analyses were performed for Oconee plants 1 and 3. The objective of this work was to support the Pressurized Thermal Shock Integration Study (PTS). The results presented in this paper (primary system pressure and loop A hot leg fluid temperature) agreed with the plant data pretty well.

We conclude that loop codes (such as RELAP5) may give good global predictions about the reactor core such as a core outlet pressure transient and a core-intact and broken loop inlet coolant temperature transient, during an SLB transient. However, because of the importance of the asymmetric thermal-hydraulics in determining the reactivity feedback, detailed analysis for the reactor core is required. The global results obtained from these loop codes are however important since they supply the core inlet boundary conditions and transient forcing functions.
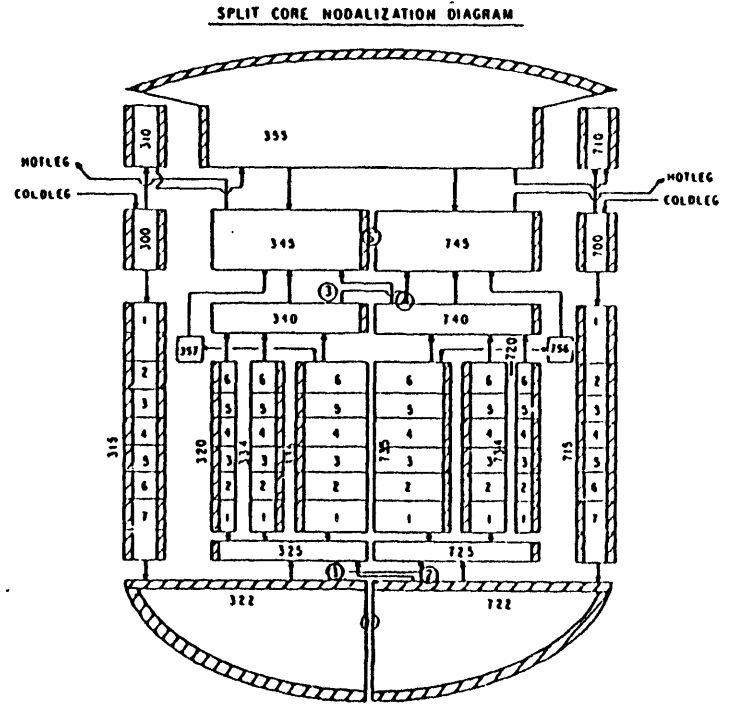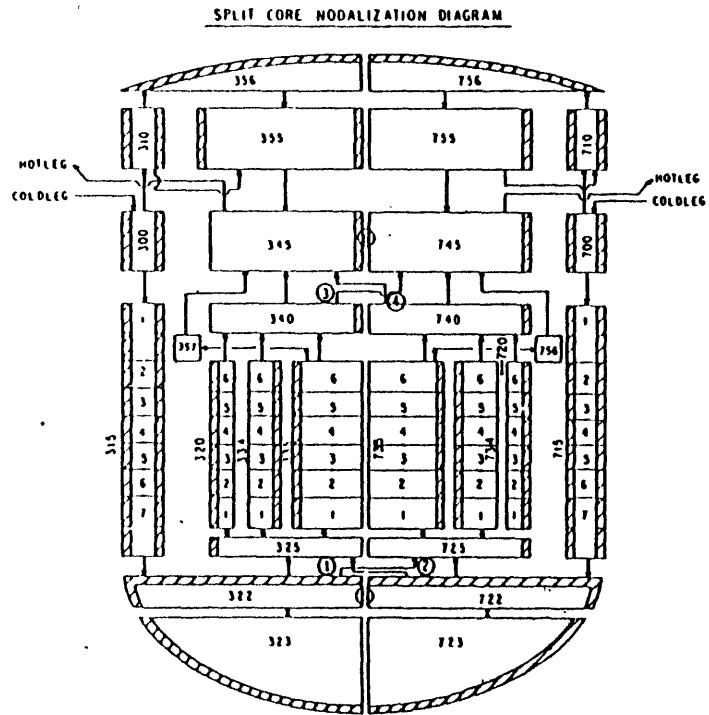
Figure 1.2 Split vessel RELAP5/MOD1.5 noding diagram for CE NSSS plant. [12]
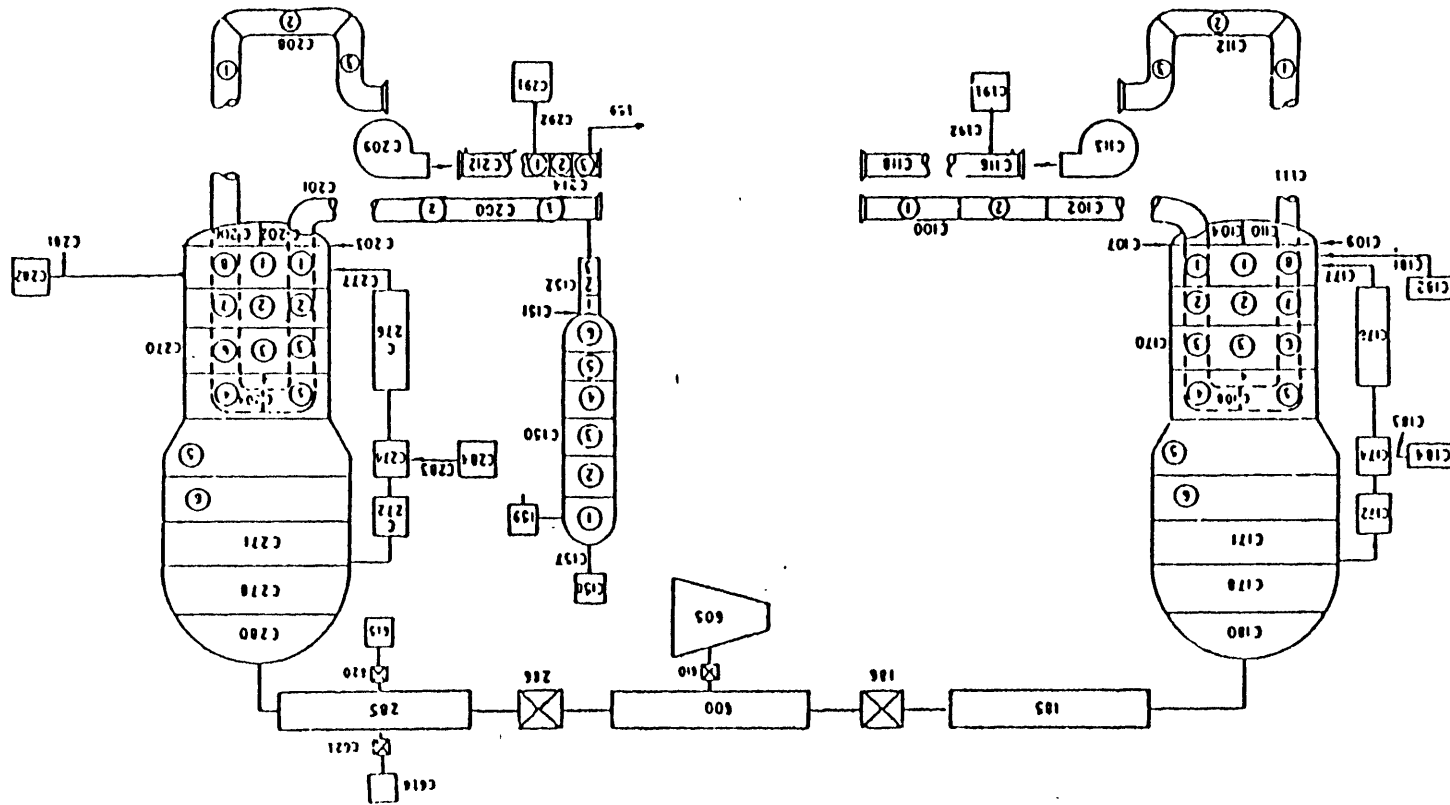
Figure 1.3 Nuclear Steam Supply System (NSSS) RELAP5/MOD1.5 noding diagram for CE NSSS plant. [12]

10
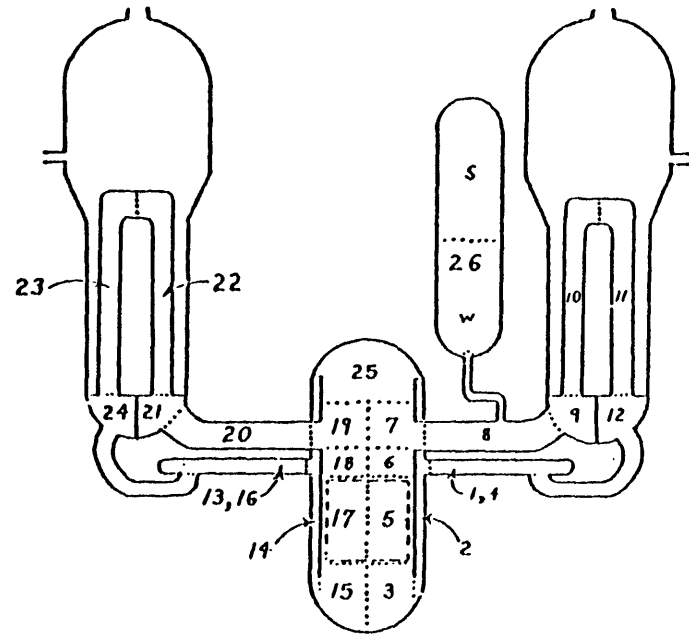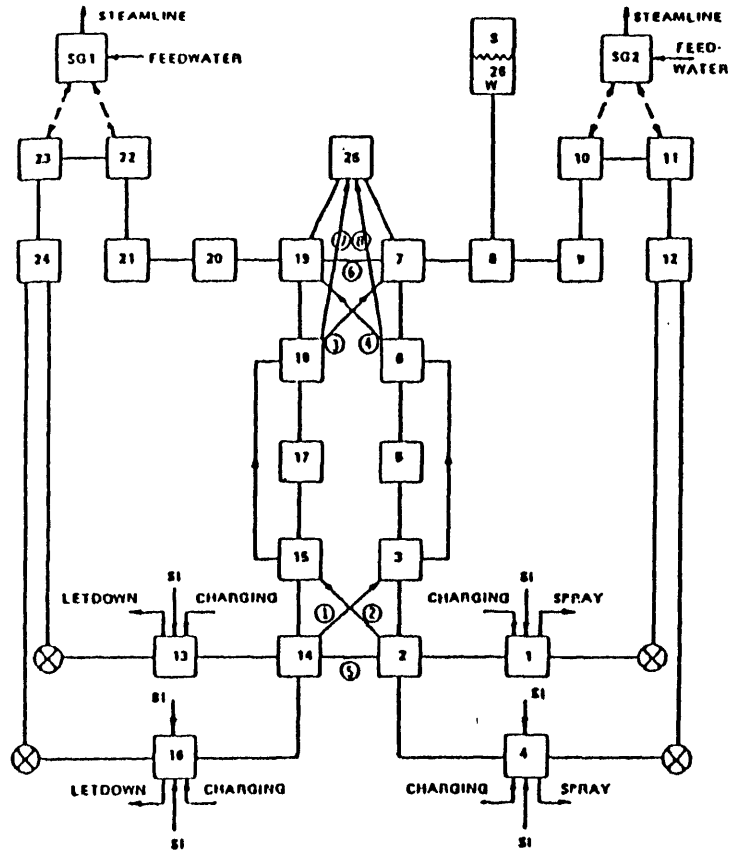
Figure 1.4 CESEC-III nodalization and schematic of CE NSSS plant. [12]
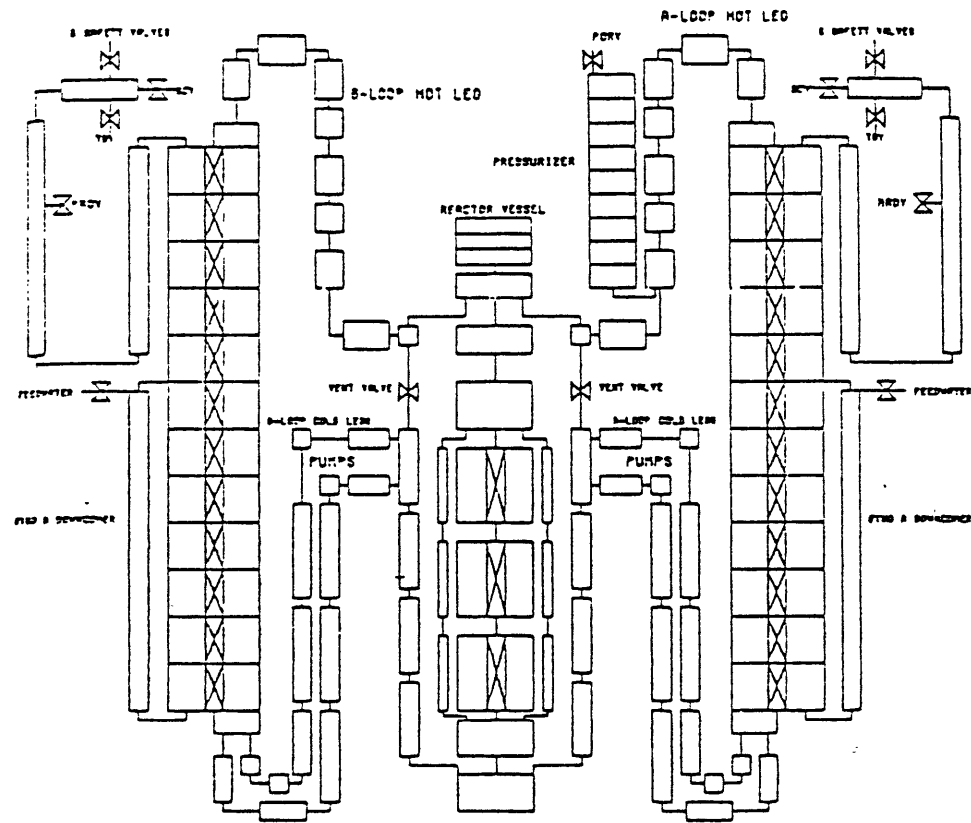
Figure 1.5 RELAP5 nodalization diagram for B&W steam generator tube rupture analysis. [15]

1.3.2 Core Codes With Traditional Safety Analysis Approach

Woodcock [17] studied inter-assembly crossflow effects in PWR cores during a secondary steamline rupture. The objective of his work was to supply improved results of the flow distribution and heat removal characteristics within the core, as well as the feedback effects of the reactor during the cooldown transient of a Departure from Nucleate Boiling (DNB) calculation. Woodcock mentioned in his paper that *'Traditional safety analyses for the subject event are performed in three steps: 1) determine the system transient using one-dimensional neutron kinetics (point kinetics) to determine the core power, 2) perform a neutronics calculation to determine core reactivity and power distribution using a more detailed core model, and 3) perform a DNB evaluation using an open lattice core thermal-hydraulics model'.*

In Woodcock's study, the transient calculation was performed with the LOFTRAN program [18]. The neutronics calculations were performed with a three-dimensional two-group neutron computer program, TURTLE [19]. The thermal hydraulics calculations were performed using the THINC computer code [20]. The LOFTRAN is a loop code with standard point kinetics model and six delayed neutron groups for the neutronics calculation. The coupling between the neutronics calculations (TURTLE) and the thermal-hydraulics calculations (THINC) was described in the paper. THINC calculated a distribution of moderator densities throughout the core with given inlet conditions for a particular transient time and an initial guess for the power level and power distribution. With these fixed densities and target k_{eff}, TURTLE provided a core power level and full dimensional power distribution. A comparison was then made between the power level and distribution assumed in THINC and that produced by TURTLE. An iteration process was performed until convergence criteria were met. The target eigenvalue k_{eff} was determined in neutronics calculations and the steady state core power level which can be supported at the given system conditions was obtained. Actually, the whole calculation was a quasi steady state analysis performed by doing steady state nuclear and thermal-hydraulic calculations at various times.

With respect to the first step of the analysis described above, the discussion in Section 1.2 shows that the results obtained from the point kinetics model are questionable. Also the iterative coupling between the second and third steps with quasi steady state calculation is not a complete transient calculation. However, one important phenomenon was identified from the paper: the effects of inter-assembly cross flow in PWR cores during an SLB transient are important for the prediction of core power distribution.

1.4 Objectives and Organization of the Report

In this chapter, the importance of analyzing a reactor transient was discussed. A steam line break accident is a possible serious abnormal transient which might happen in a nuclear reactor. It is necessary to understand the detailed behavior of the transient so that safer reactor designs and safer reactor operations can be achieved. The objectives of this report are:

1. To develop an existing complete three-dimensional coupled core code, TITAN [1-4,21], such that it is applicable to SLB transient simulations;
2. To investigate the three-dimensional effects on steady state calculations;
3. To investigate the three-dimensional effects on SLB transient simulations;
4. To investigate the boron effects on the core behaviors such as return to power condition and power distribution;
5. To perform sensitivity studies for code cpapbilities.

The organization of this report is as follows. The modifications of the TITAN code to satisfy the requirements of simulating a steam line break accident will be described in Chapter 2. Also, the necessary test calculations relevent to the modifications and a steam line break transient will be presented in Chapter 2. The steady state calculations of two ten-channel PWR models will be discussed in Chapter 3. Based on the conclusions of the steady state results, a fully implicit numerical scheme solving for the thermal-hydraulics equations will be introduced and the steady state results of the scheme will be presented in this chapter. Chapter 4 will include a series of steam line break accident simulations to study the three-dimensional effects of the accident before return to power occurs. The three-dimensional effects of the accident after return to power and boron injection will be presented in Chapter 5. Finally, conclusions and suggestions will be included in Chapter 6.

14

Chapter 2

CODE DEVELOPMENT, TESTING AND INLET
COOLANT TEMPERATURE SENSITIVITY
STUDIES FOR SLB APPLICATIONS

2.1 Introduction

The requirements of a computer code to simulate a steam line break accident have been discussed in Section 1.2. In summary, these requirements are:

1. A thermal-hydraulics and neutronics coupled core code;
2. Both the thermal-hydraulics and neutronics models should be 3-D models;
3. A boron concentration model is required;
4. Total inlet flow rate should be one of the boundary condition options;
5. Should be able to simulate a flow transient;
6. Should be able to simulate a multi-zone (non-uniform) inlet coolant temperature transient;
7. Should be able to simulate a pressure transient;
8. Should be able to simulate a control rod initiated transient (scram) and a thermal-hydraulic transient simultaneously.

The TITAN computer code has been reviewed in Ref. [21]. The capabilities of the current version of TITAN related to simulating a steam line break accident are summarized as follows:

1. It is a neutronics and thermal-hydraulics coupled core code;
2. The code solves three-dimensional, two-group nodal diffusion equations to give three-dimensional core power distribution;
3. The code solves a three-dimensional, two-fluid, ten-equation fluid dynamics model to give detailed three-dimensional information for both boiling and non-boiling situations;
4. The code has the capability of simulating a uniform core inlet coolant temperature as a function of time, i.e., only one forcing function is permitted for all channels;
5. The code has the capability of simulating either a control rod initiated transient or a flow/temperature initiated transient;
6. The code has the capability of simulating a core with distributed inlet flow velocities as boundary condition;

15

7. The code has the capability of simulating a flow transient provided the inlet flow velocity forcing functions are given;

8. The code has the restart capability for steady state calculations.

Comparing the requirements of simulating a SLB accident and the capabilities of the current TITAN code shows that, several code enhancements had to be made before the code could be applied to a steam line break accident simulation. It was necessary to test these code enhancements, and, at the same time, some of the code capabilities which have not been used previously were tested. These code enhancements and tests are described in the following sections.

2.2 Inlet Coolant Temperature Transient Forcing Function

2.2.1 Development

As mentioned in the previous section, TITAN can simulate a uniform core inlet coolant temperature transient. However, to simulate a steam line break accident requires having a non-uniform inlet coolant temperature simulation capability since the accident involves an asymmetric situation for which the inlet coolant temperature will not necessarily be the same for all channels. A modification was necessary to extend the capability of the code so that a more flexible forcing function could be employed for different channels at the core inlet.

By modifying the code input logic and data arrangement, the code was altered so that it now has the capability to simulate up to ten inlet coolant regions with ten different temperature forcing functions for each region. For each region, at most fifty channels can be modeled. Altogether, up to 500 channels can be modeled. This is quite sufficient to provide even a whole core analysis without losing detailed description of the inlet coolant temperature distribution.

2.2.2 Testing

A ten-channel PWR type test problem was studied to test the new modified capabilities. Three regions were assumed in this model. Figure 2.1 shows the ten-channel model. Channels 2, 3, 4, and 7 are the hot region (intact loop). Channels 1 and 6 are the mixing region (region between the intact and broken loops). Channels 5, 8, 9, and 10 are the cold region (broken loop). It was assumed that there are four circulation loops for the core and attached north, south, east, and west. The north and south direction loops are the broken loops. The east and west direction loops are the intack loops. Also, the two broken loops were assumed
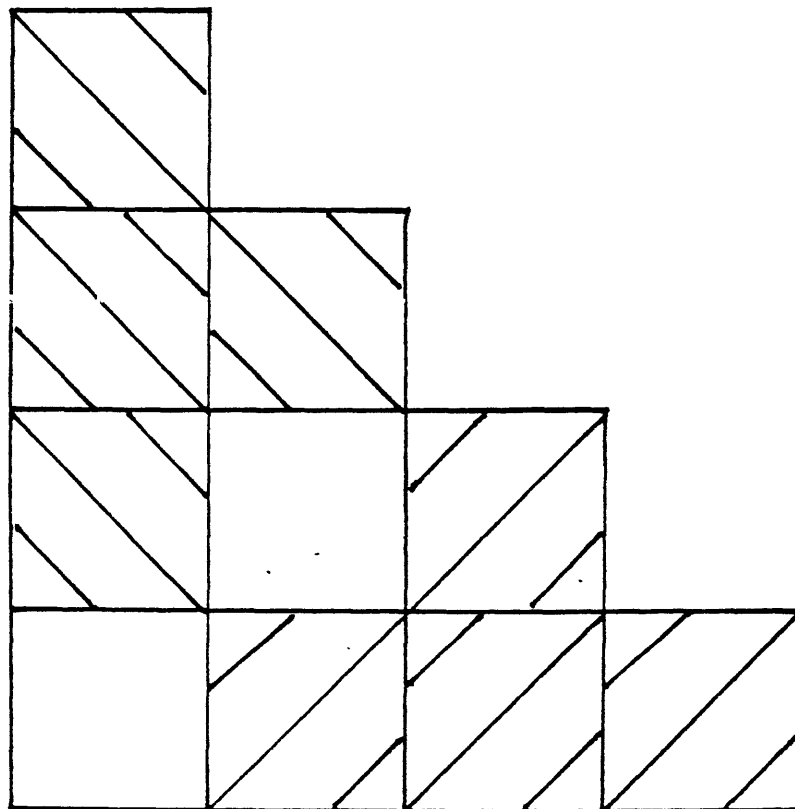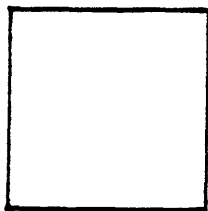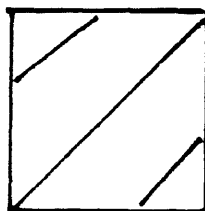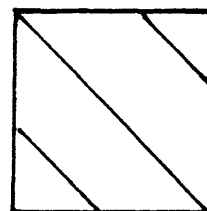
Figure 2.1 Three inlet coolant temperature zones steam line break transient simulation.

to-undergo identical transients. Thus, the four quarters of the core are identical. This is illustrated in Figure 2.2. The nuclear composition layout is shown in Figure 2.3. Channel one has a fully inserted control rod and channels 3 and 8 have 45% partially inserted control rods. These are shown in Figure 2.4. The three regions in the above ten-channel quarter core geometry (Figure 2.1) represent the intact loop region, the mixing region, and the broken region. Each region has its own inlet coolant temperature forcing function, as shown in Figure 2.5. The node dimension and the initial conditions are shown in Figure 2.6.

Two items ought to be checked in the code output. The first item is whether the inlet coolant temperature is changing as described by the given forcing function for each region. From the output of a sample calculation, this has been verified. The second item to be checked is the core power history. Since nothing is changing except the inlet coolant temperature which keeps dropping, positive reactivity is added by the coolant temperature, coolant density coefficients and the fuel Doppler feedback effect as discussed in Ref. [21]. Therefore, a power excusion is expected. This phenomenon is discussed in Ref. [21], Section 3.7.

2.3 Total Inlet Coolant Mass Flow Rate Boundary Condition

A total inlet mass flow rate boundary condition is required for transients where only total inlet flow rates are known. This is the case for most transients, since the detailed inlet flow split is generally not available.

The approach of previous THERMIT work was adopted [22]. The idea is that, for a given total inlet flow rate, the code will calculate the lower plenum pressure, p_l, based on the current guess for pressures inside the domain. An additional equation is solved as part of the usual pressure solution:

$$(\sum_i a'_{1i})p_l - (\sum_i a'_{1i}p_{1i}) = \sum_i b'_{1i} - W_1 \qquad (2.1)$$

where

$$p_l = \text{the lower plenum pressure}$$
$$p_{1i} = i\text{th channel entrance node pressure}$$
$$W_1 = \text{total inlet mass flow rate}$$
$$a'_{1i} \text{ and } b'_{1i} = \text{the coefficients}$$

In terms of pressure corrections, equation (2.1) leads to

North Loop

B   B

M   M

East Loop   I   I   West Loop

I   I

M   M

B   B

South Loop
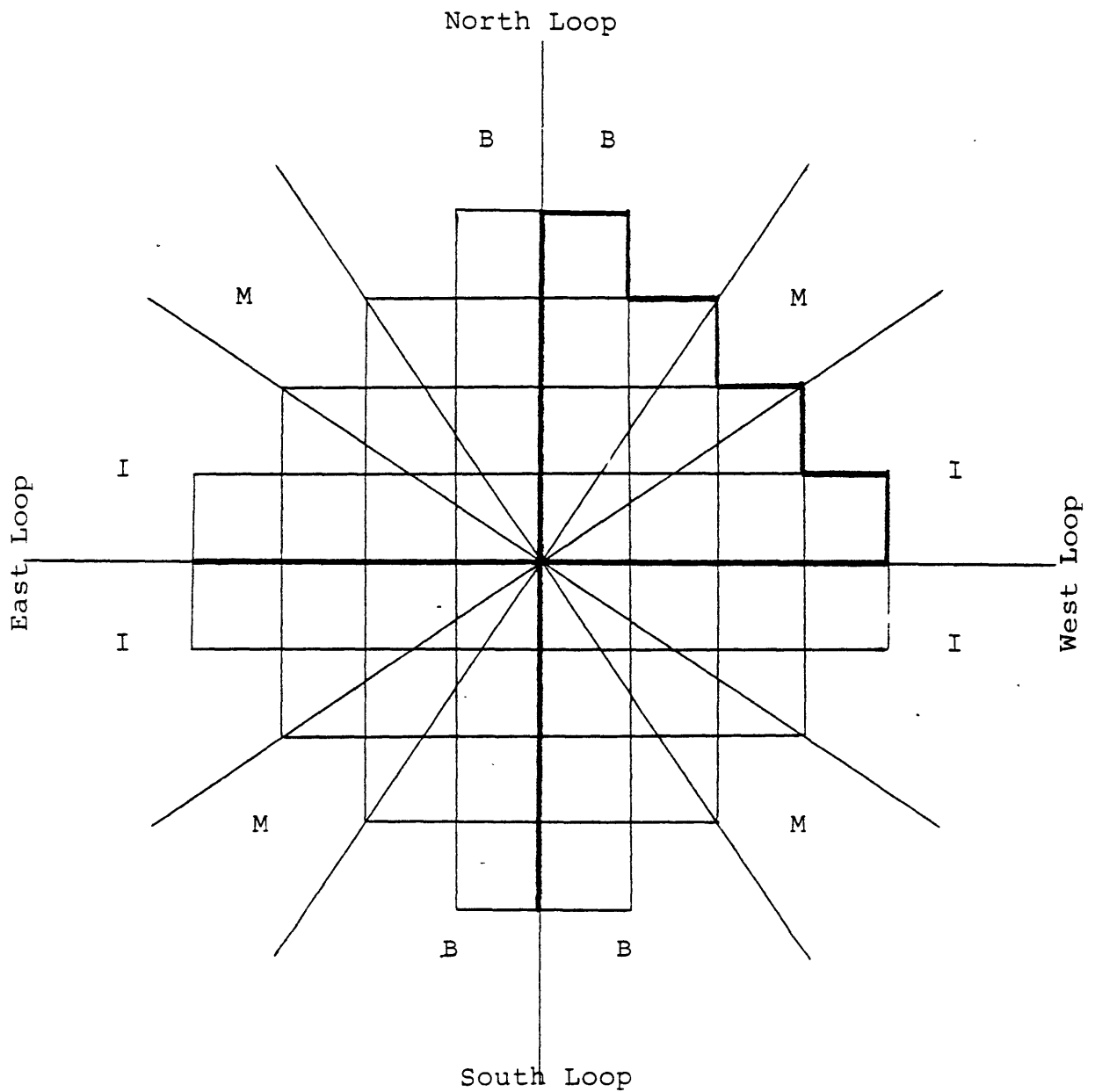
B: Broken Loop Region

M: Mixing Region

I: Intact Loop Region

Figure 2.2 Illustration of the symmetry quarter core
10-channel model in a steam line break accident.
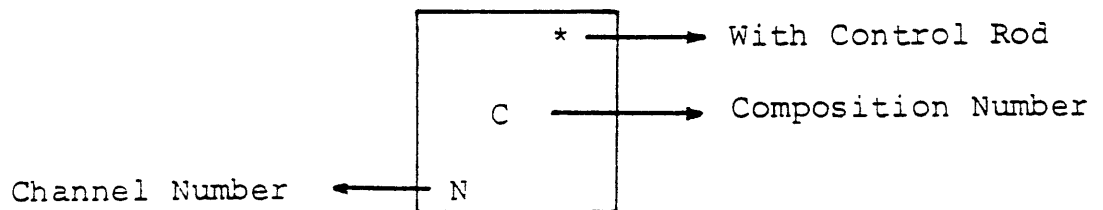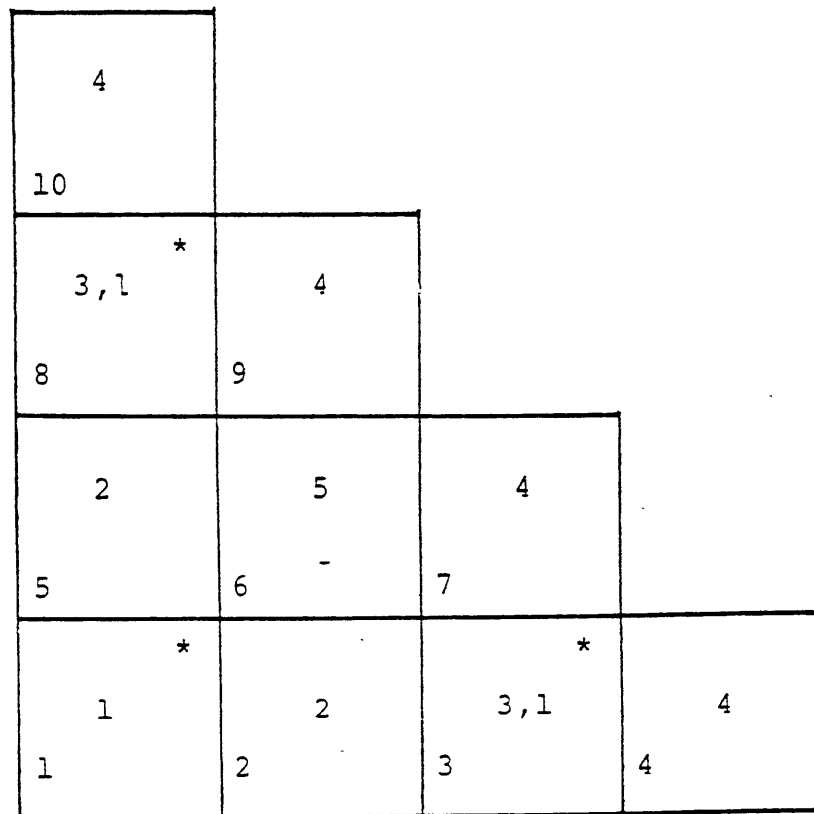
Figure 2.3   Composition layout for 10-channel steam line break transient.
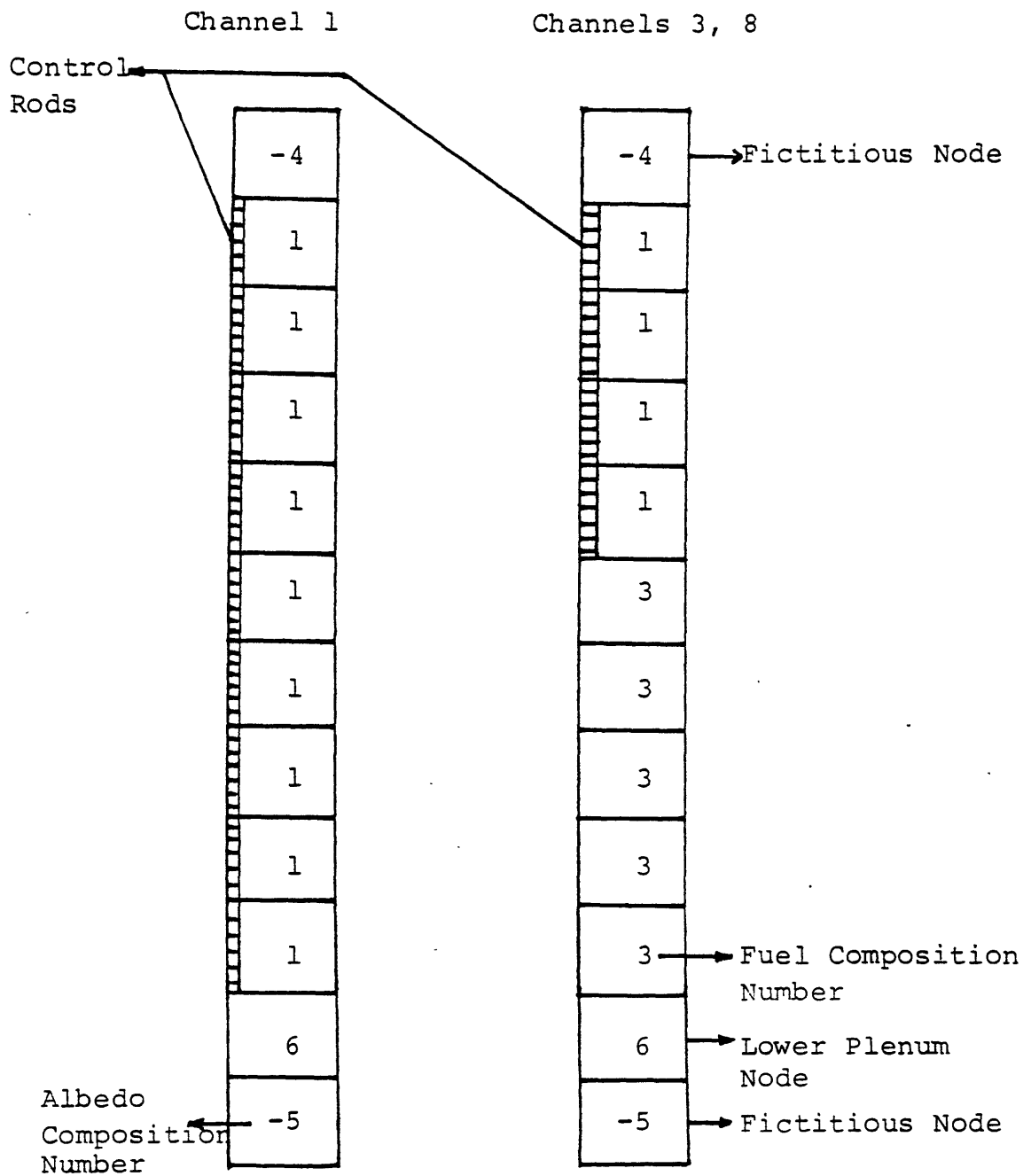
Figure 2.4 Assemblies with control rod inserted.
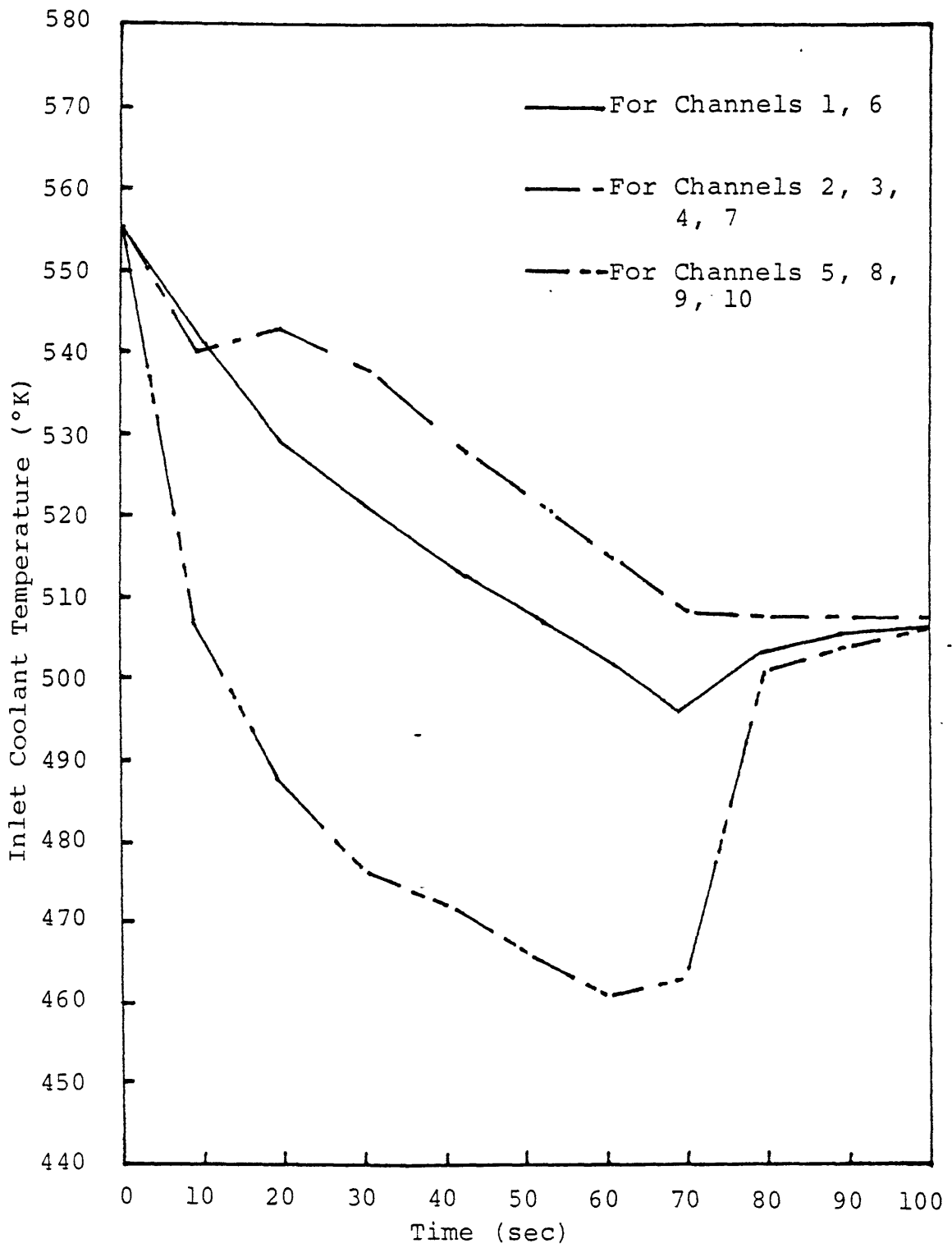
Figure 2.5 Inlet coolant temperature v.s. time.
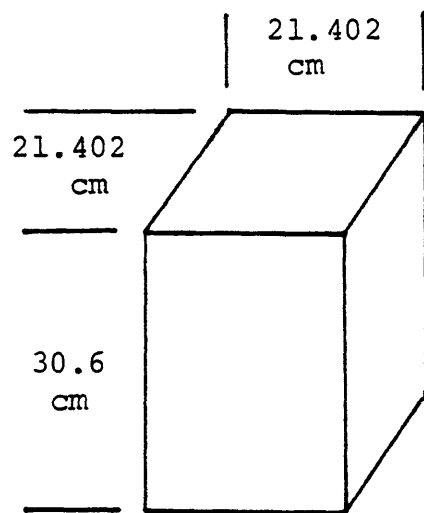
21.402
cm

21.402
cm

30.6
cm

System Pressure: 15.65MPa

Inlet Mass Flow Rate: 651.2 Kg/s

Initial Inlet Temperature: 555°K

Initial Reactor Power: 37.95MW

Figure 2.6 Node dimension and initial conditions

$$\left(\sum_i a'_{1i}\right)\delta p_l - \sum_i a'_{1i}\delta p_{1i} = 0 \qquad (2.2)$$

where

$\delta p_l =$ lower plenum pressure correction

$\delta p_{1i} =$ channel i entrance node pressure correction

The resulting correction, δp_l, is used to update the pressure in the plenum just as the other pressure corrections are used to update the pressures inside the domain.

Once the new lower plenum pressure and the new pressures of entrance nodes of the core are obtained, the corresponding inlet coolant velocities of all channels are calculated. This is important for steam line break accident simulations since the only available information is the total inlet flow rate instead of the inlet coolant velocities of the various channels. If channel velocity boundary conditions at the core inlet are used, inaccuracy will be introduced.

The next modification made to TITAN was to add the capability to simulate total inlet coolant flowrate transients. This is necessary since the inlet flowrate is a function of time during steam line break accidents. The modification was achieved by modifying the code input logic and data arrangement.

By checking a sample calculation output, it was proved that the total inlet flow rate follows the given flow transient forcing function, and a converged calculated total inlet mass flow rate is obtained.

2.4 The Boron Concentration Tracking Model and The Boron Reactivity Feedback Model

In order to take into account the effect on the total absorption cross section in each node of the boron solution injected into the core during a safety injection, it is necessary to know the boron concentration of each node at any time. With the boron concentration of each node, the number of boron-10 atoms per unit volume can be calculated. The macroscopic boron absorption cross section can then be calculated provided the microscopic boron cross section averaged over the energy spectrum is known.

The simplest way to describe boron transport is to solve its mass conservation equation with the following assumptions [23]:

1. Injected boron is transported only in the liquid phase of the coolant.
2. The liquid water (solvent) injected with the boron is neglected.

24

3. Liquid properties are not altered by the presence of boron.

4. Boron travels at the same velocity as the liquid coolant.

5. No axial diffusion (mixing) is considered.

6. Energy transported by boron and inertial effects due to boron are negligible.

The general boron balance equation then takes the form:

$$\frac{\partial[\rho_l(1 - \alpha)C_B]}{\partial t} + \frac{1}{A_f}\nabla \cdot [\rho_l C_B(1 - \alpha)\bar{v}_l A_f] = B \tag{2.3}$$

where

$\rho_l =$ liquid coolant density

$\alpha =$ void fraction

$v_l =$ liquid coolant velocity

$C_B =$ mass fraction of boron-10 in the liquid coolant

$B =$ source of boron

$A_f =$ flow area in the flow direction

To solve equation (2.3), a simple implicit numerical scheme with the donor cell concept used in the THERMIT code (reviewed in Ref. [21]) was employed. The finite difference equation corresponding to the balance equation for node (i, j, k) with nodal volume V at time step $n + 1$ is:

$$\frac{1}{\Delta t}\left[(\rho_l(1 - \alpha)C_B)_{ijk}^{n+1} - (\rho_l(1 - \alpha)C_B)_{ijk}^{n}\right] +$$

$$\frac{1}{V}\{[A_x\rho_l(1 - \alpha)C_B v_{lx}]_{i+1/2jk}^{n+1} - [A_x\rho_l(1 - \alpha)C_B v_{lx}]_{i-1/2jk}^{n+1}\} +$$

$$\frac{1}{V}\{[A_y\rho_l(1 - \alpha)C_B v_{ly}]_{ij+1/2k}^{n+1} - [A_y\rho_l(1 - \alpha)C_B v_{ly}]_{ij-1/2k}^{n+1}\} +$$

$$\frac{1}{V}\{[A_z\rho_l(1 - \alpha)C_B v_{lz}]_{ijk+1/2}^{n+1} - [A_z\rho_l(1 - \alpha)C_B v_{lz}]_{ijk-1/2}^{n+1}\} = B \tag{2.4}$$

Since the boron concentration is assumed to move with the liquid coolant, the above equation can be solved at the end of each time step. Therefore, only one quantity, C_{Bijk}^{n+1}, is unknown in equation (2.4) and then can be found by applying the equation.

The absorption cross section increase because of the addition of boron can be obtained by finding the number of boron-10 atoms per unit volum, N_B, first:

$$N_B = \frac{\gamma(1 - \alpha)\rho_l C_B A_f}{M_B A_{zn}} \times N_A \tag{2.5}$$

25

where

$$A_f = \text{z-direction flow area}$$
$$A_{zn} = \text{z-direction node corss section area}$$
$$M_B = \text{atomic weight of boron-10}$$
$$= 10.8 \times 10^{-3} Kg/mole$$
$$N_A = \text{Avogadro's number}$$
$$= 6.023 \times 10^{23}/mole$$
$$\gamma = \text{natural abundance of boron-10}$$
$$= 19.2\%$$

The macroscopic boron absorption cross section for a given energy group is therefore:

$$\Sigma_a^B = \sigma_a^B \times N_B \tag{2.6}$$

where

$\sigma_a^B = $ boron-10 microscopic cross section averaged over the energy group.

If all constants are substituted in equation (2.6), the equation can be rewritten as:

$$\Sigma_a^B = 1.07 \times 10^{-11} \sigma_a^B \left(\frac{A_f}{A}\right)(1-\alpha)\rho_l C_B \tag{2.7}$$

where

Σ is in the unit of $[\bar{m}^{-1}]$

σ is in the unit of [barn]

ρ_l is in the unit of $[Kg/m^3]$

C_B is in the unit of [ppm]

The application of the boron models is presented in Chapter 5.

2.5 Scram Simulation

As mentioned in Section 1.2 one of the simulation requirements for a steam line break accident is the reactor scram. Before performing a steam line break accident simulation, it is necessary to make sure that the code predicts correct scram results. With the same testing problem described in Section 2.2.2, a scram only transient calculation was performed. The two partially inserted control rods in channels 3 and 8 were allowed to fall into the core starting at 2.5 seconds and ending at 5-seconds after the beginning of the transient.

The expected result of a scram is a reactor power drop because of the large negative

26

reactivity addition to the core. As seen from Figure 2.7, the total core power history decreases after 2.5 seconds. There is no power change during the first 2.5 seconds period because there is no scram during that period. It should be pointed out that in the current version TITAN no fission product decay heat is included. Only fission heating is calculated.

This calculation gives confidence in the code's scram simulation capability.

2.6 Flow/Temperature Transient plus Control Rod Transient Option and It's Testing

A code logic modification was introduced to provide the user an option to simulate events which include both control rod movement and flow/coolant temperature transient. This is necessary for a steam line break accident simulation. The original code could handle either a control rod transient or a flow transient, but not both transients simultaneously.

A test calculation was performed involving a simultaneous inlet coolant temperature transient and scram transient, an open-channel non-uniform inlet coolant temperature distribution calculation. The 10-channel model and initial conditions were the same as those described in Section 2.2.2.

The combined net effect of the inlet coolant temperature transient and the scram transient is illustrated in Figure 2.8. As mentioned in the previous section, the scram starts at 2.5 seconds after the transient begins. The only reactivity (which is positive) introduced into the core is from the inlet coolant temperature drop. In Figure 2.8, a tilt is observed in the scram plus temperature feedback transient power history during the first 2.5 seconds. There are sharp power drops for both scram only and scram plus temperature feedback transients from 2.5 seconds to 5 seconds. This correlates with the scram motion during this period. Afterwards, the difference between the two power histories becomes larger and larger. This demonstrates the effect of the positive reactivity insertion due to the temperature transient. To check the detailed local behavior of the combined transient, Figures 2.9 and 2.10 show the radial assembly power distributions at different times. The scram effect on channels 3 and 8 can be seen clearly from these figures. Recall that channels 3 and 8 contain partially inserted control rods initially and that scram is simulated by inserting the two partially inserted control rods into the core.

Figures 2.11 and 2.12 show the axial nodal power distributions of channels 3 and 8. These figures provide a more detailed local information. The peak power of channel 3 at 60 seconds is about 250 KW while the peak power of channel 8 at the same time is about
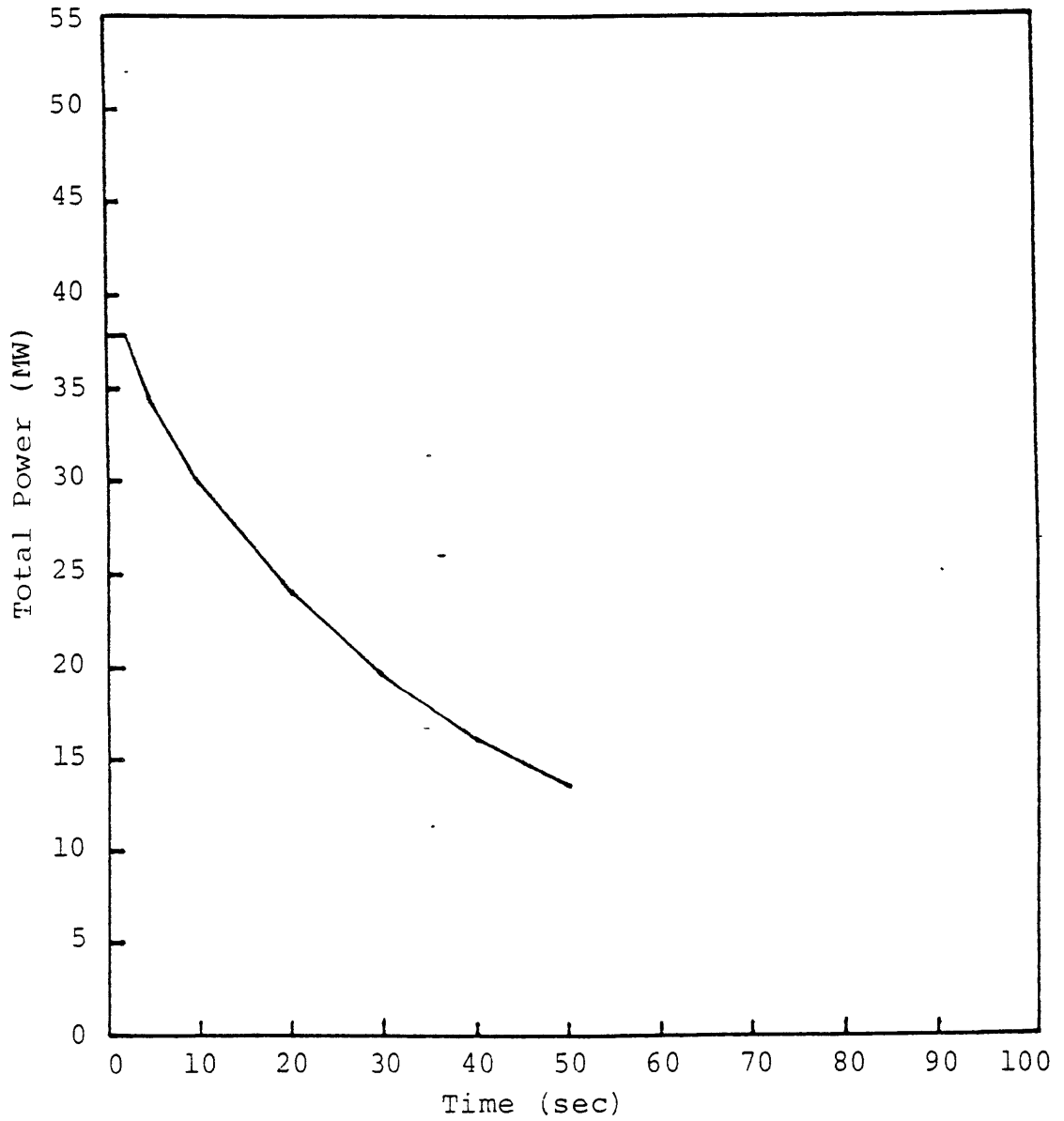
Figure 2 ..7 Power history of 10-channel PWR scram transient.
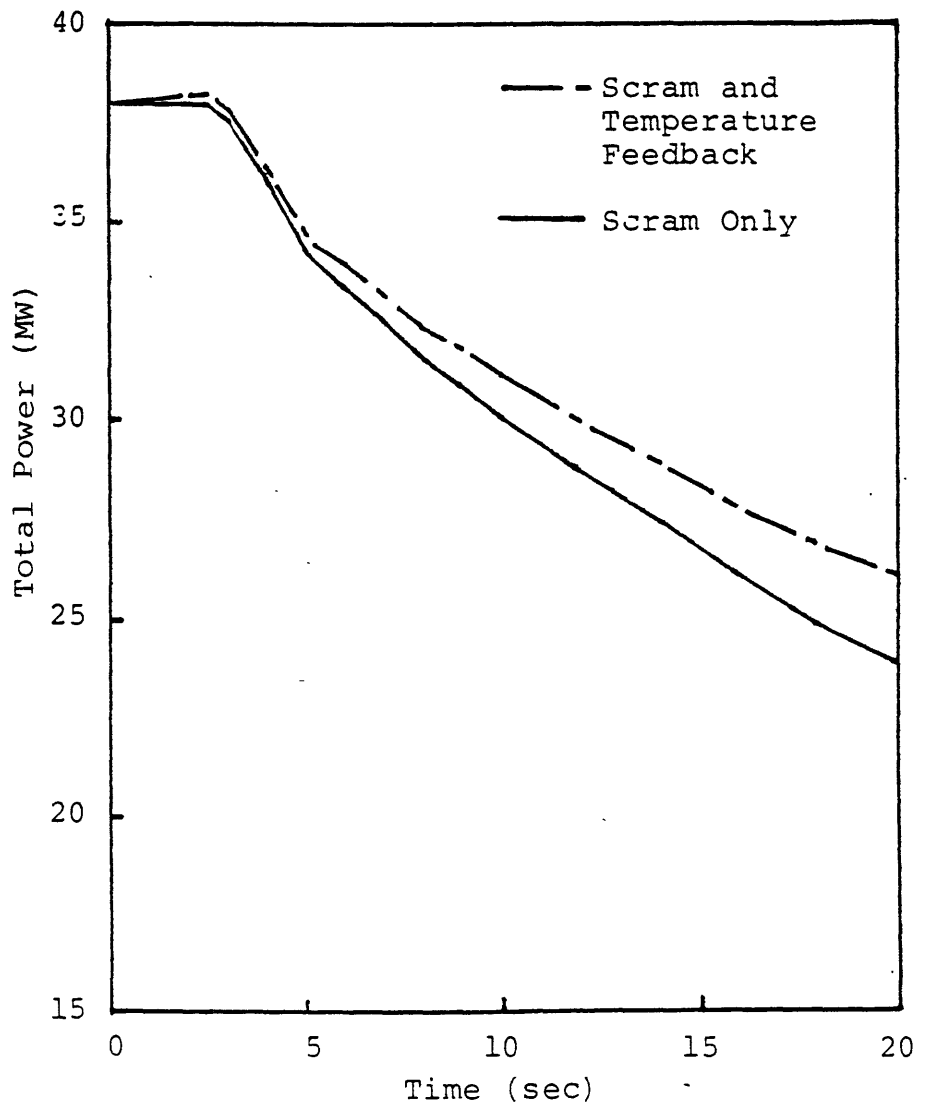
Figure 2. 8 Power histories of scram only and scram
plus temperature feedback transients.
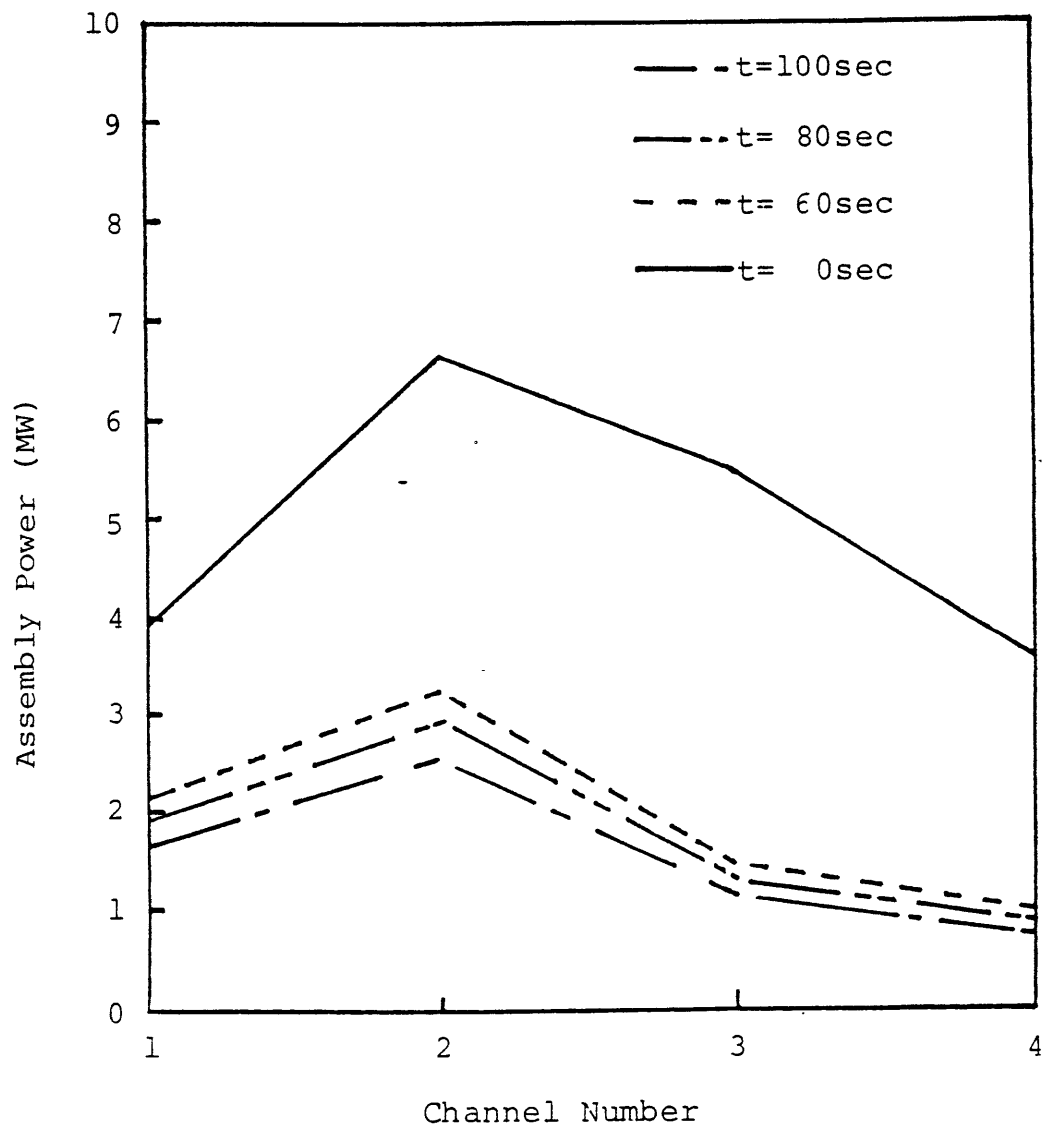
Figure 2.9 Radial assembly power distribution v.s. time
            for channels 1, 2, 3, and 4; scram and
            temperature feedback.
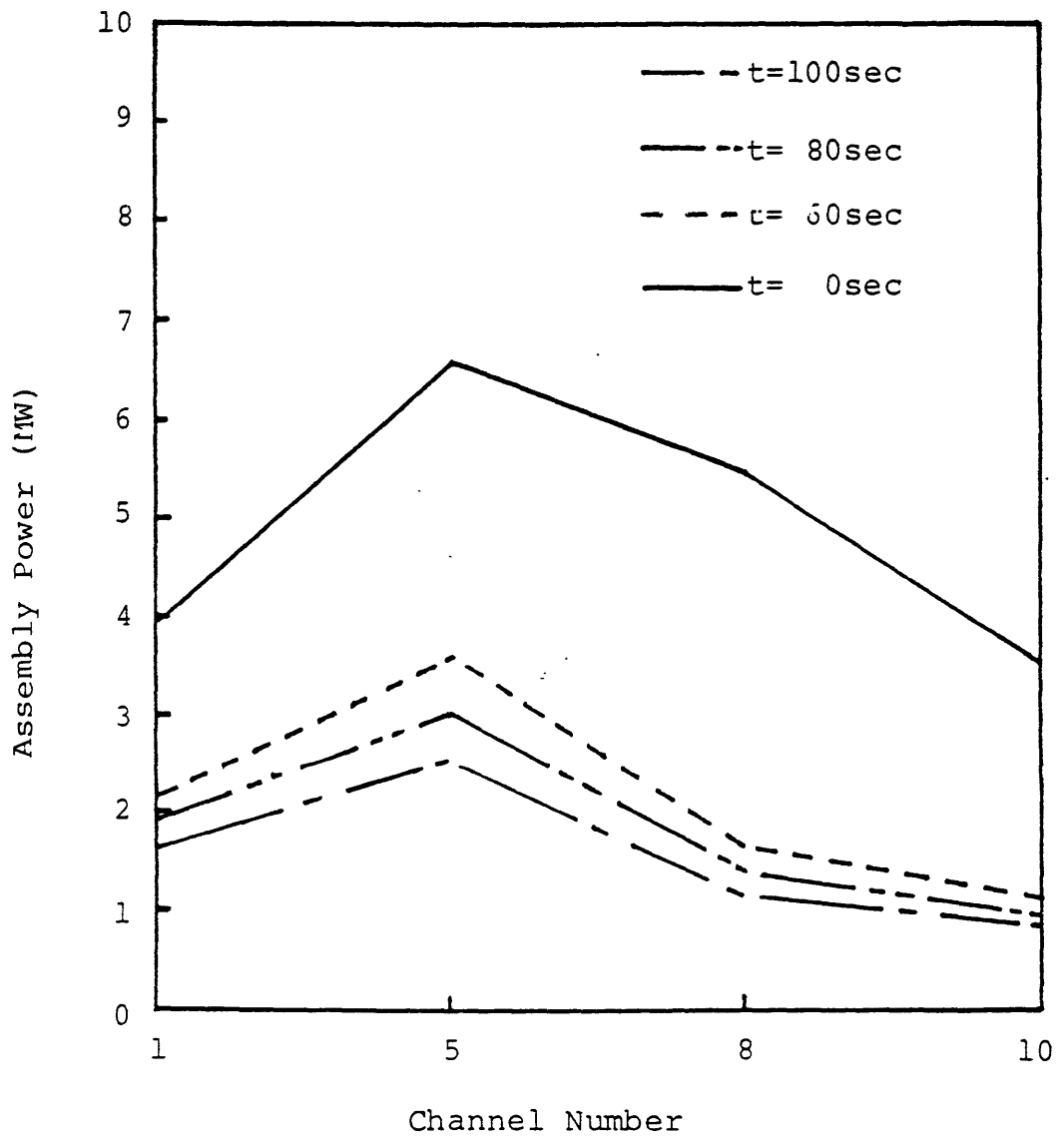
Figure 2.10 Radial assembly power distribution v.s. time
for channels 1, 5, 8, and 10 ; scram and
temperature feedback.
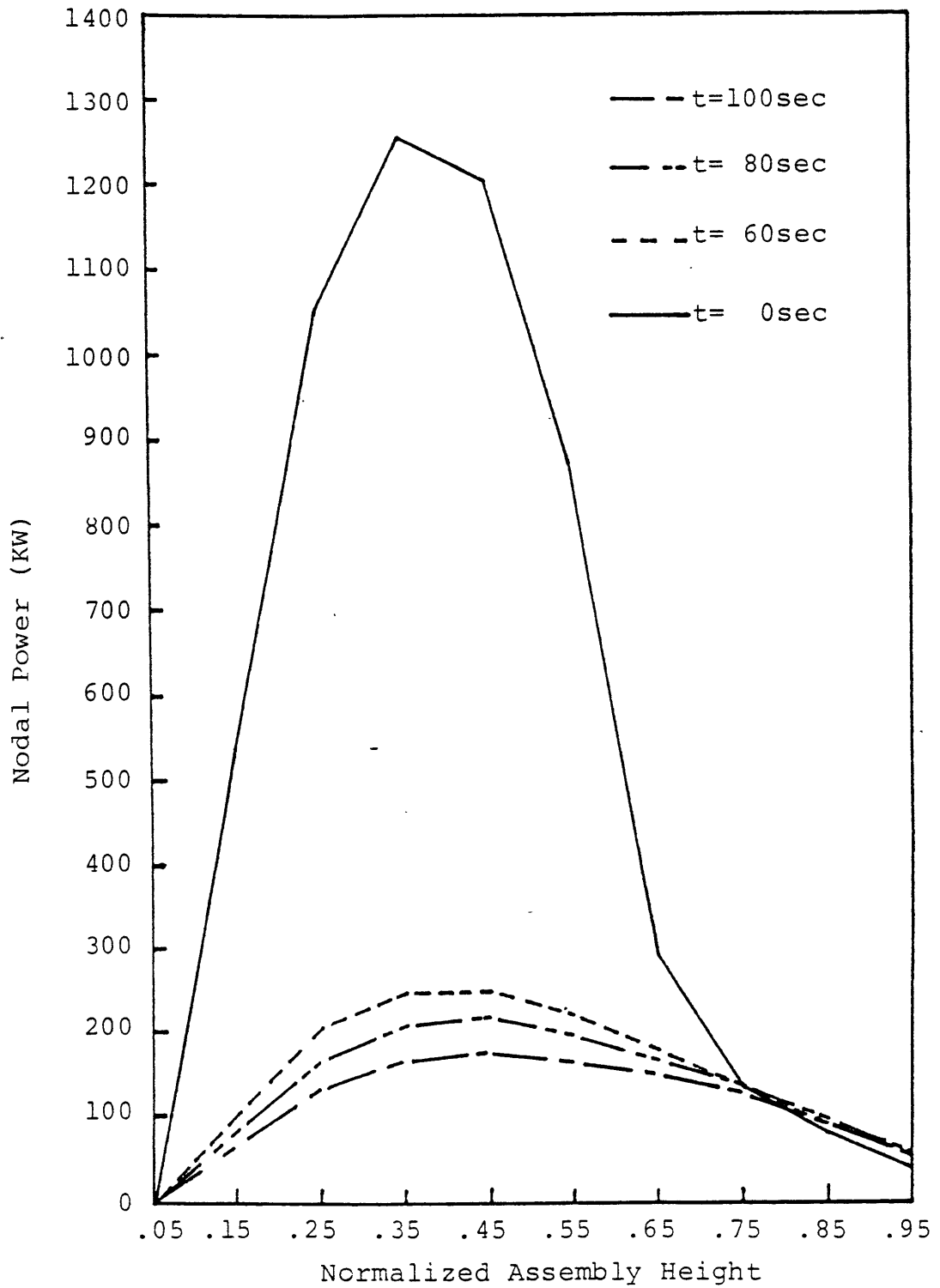
Figure2.11 Axial nodal power distribution for channel 3
at different times;scram and temperature
feedback.
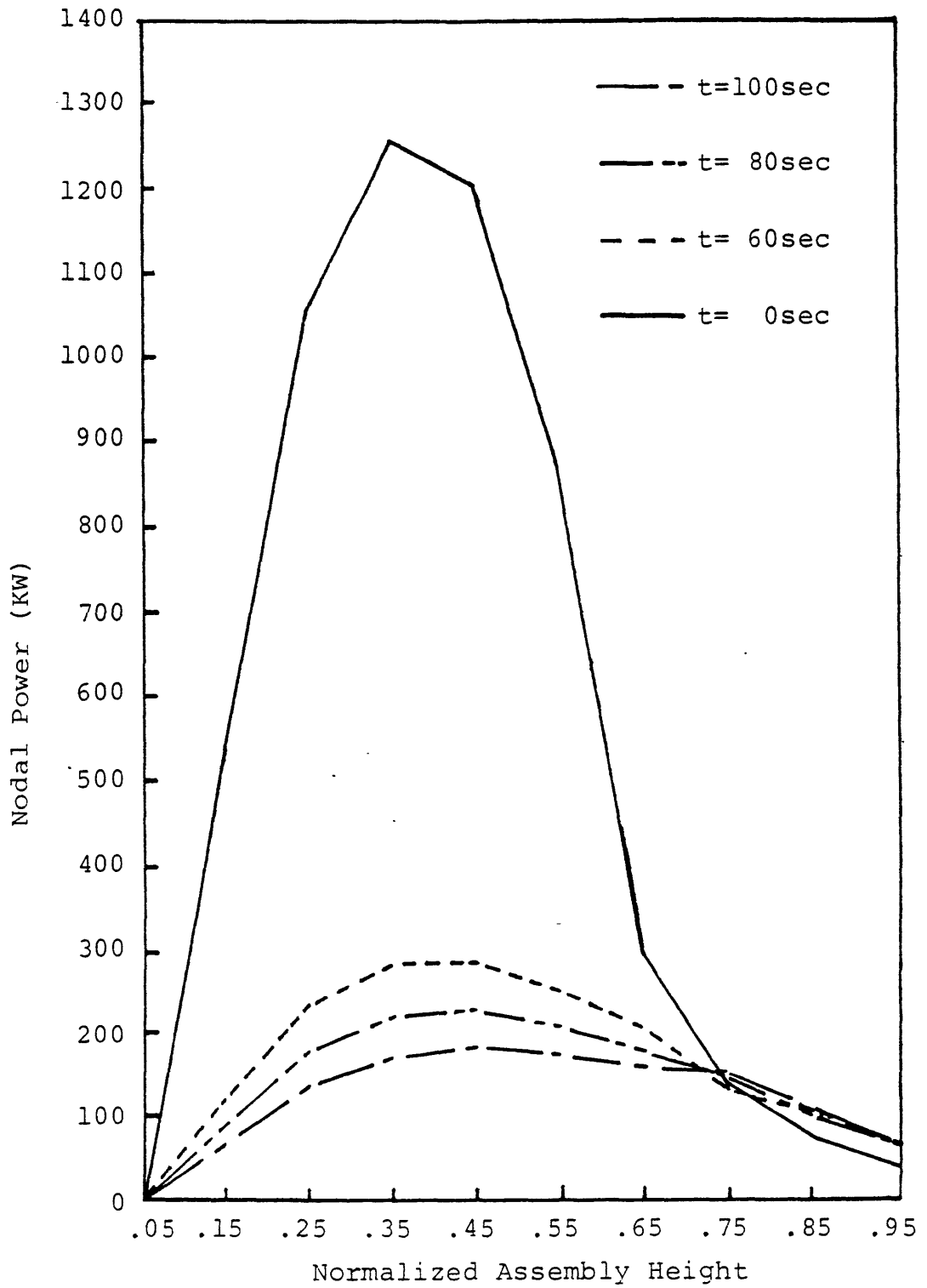
Figure 2.12 Axial nodal power distribution for channel 8 at different time; scram and temperature feedback.

300 KW. This again reflects the different amount of reactivity insertion in 'hot' and 'cold' regions. At 100 seconds, the axial power distributions of the channels are almost the same. This correlates with the inlet coolant temperature forcing functions for the two regions which approach each other after 70 seconds.

All of the above global power histories, radial assembly power distributions and axial nodal power distributions show a power drop within the 100 second simulation period. This demonstrates that with the combination of 10-channel model, nuclear data, and transient functions employed in the above test calculation, the net reactivity insertion was negative. No return to power condition was reached within the simulation period.

2.7 Summary

Necessary modifications of TITAN to satisfy the requirements of a code to simulate a steam line break accident were presented in this chapter. The modifications include:
1. Multi-zone inlet coolant temperature forcing functions capability;
2. Total inlet coolant flow rate boundary condition option;
3. Total inlet coolant flow rate transient simulation capability;
4. Boron concentration tracking model and boron reactiviy feedback model;
5. Flow/coolant temperature transient plus control rod transient option.

In addition, a transient restart capability was provided. With this capability, it is easier to perform lengthy transient calculations, such as ones described in this chapter. Also, simulation conditions can be changed during a transient calculation for various purposes.

Finally the transient calculation time step sizes for the thermal-hydraulics and neutronics parts of the calculations were decoupled. The neutronics time step size can be an integer multiple of the thermal-hydraulics time step size. This permits taking advantage of the fact that because of the fully implicit numerical scheme used in solving the nodal diffusion equations, there is no restriction on the neutronics time step size. For a fast thermal-hydraulic transient with slow neutronic response, this option can save computation time in the neutronics part.

Three test calculations were discussed in this chapter. They are:
1. Inlet coolant temperature transient;
2. Scram transient;
- 3. Inlet coolant temperature transient plus scram transient.
The above tests cover the major phenomena of a steam line break accident.

34

The conclusions drawn from the results of these test calculations are:

1. The code with the new modifications described in this chapter can simulate a complicated transient, such as a steam line break accident.

2. Three-dimensional codes are necessary for a non-uniform inlet coolant temperature distribution transient calculation.

3. Before boiling occurs, there is no significant difference between results of closed- and open-channel calculations.

The TITAN User's Guide is in Appendix A for reference.

Chapter 3

STEADY STATE CALCULATIONS AND
ONE-DIMENSIONAL IMPLICIT
SCHEME IMPLEMENTATION AND TESTING

3.1 Introduction

The TITAN code modifications for a steam line break transient simulation have been made and tested as discussed in the previous chapter. The code is now ready for simulating a steam line break accident. However, before starting any transient calculation, it is necessary for the code to generate converged steady state results.

In this chapter studies of the steady state closed- and open-channel calculations with two 10-channel PWR models are discussed. The conclusion of the studies is that there is no significant difference between results of closed- and open-channel calculations. Thus using a one-dimensional fully implicit numerical scheme for the thermal-hydraulic calculations instead of the semi-implicit numerical scheme currently used by the current version of TITAN may be numerically more efficient. A one-dimensional fully implicit numerical scheme is reviewed and implemented in by TITAN as an option for solving the thermal-hydraulics equations. The steady state results of the one-dimensional numerical scheme are compared with those of the semi-implicit numerical scheme and presented in this chapter.

The ability of TITAN to generate converged and reliable steady state results has been demonstrated by Griggs [3] and Tsai [4]. The calculations performed by Griggs and Tsai covered from two-channel test models to a 47-channel PWR quarter core real model. A comparison [3] between the steady state results of TITAN and MEKIN-B [24] for the 47-channel PWR quarter core model with node size around $20 \times 20 \times 30$ cm^3 shows that the nodal method for neutron diffusion equations used by TITAN give more accurate results than those by MEKIN-B which uses a finite difference numerical scheme. When the node size decreases, results of the finite difference numerical scheme converge to those of the nodal approach. The conclusion is that the nodal approach is more realistic to get accurate results than the finite difference approach for a real core model calculation, since the node size is restricted by the computational effort one can afford.

3.2 Description of Ten-Channel PWR Models

Two similar 10-channel PWR models are employed in the investigations mentioned above. One of the two models has been described in Section 2.2.2 and is designated as Model A hereinafter. In summary, Model A has three core inlet regions, the intact loop (hot) region, the mixing region, and the broken loop (cold) region as shown in Figure 3.1. However, for steady state calculations, the core inlet coolant temperature is uniform and is set to be $555°K$ as one of the core inlet boundary conditions. A fully inserted control rod is in channel 1 and two 45% inserted control rods are in channels 3 and 8, respectively. The nuclear composition layout is also shown in the figure. Figure 3.2 shows the nuclear composition layouts and control rod positions of channels with control rod inserted. The node dimension and initial conditions are shown in Figure 3.3. Model A is used for the test calculations discussed in Chapter 2.

Another 10-channel PWR model used in the investigation is designated as Model B hereinafter. Model B is exactly the same as Model A except that there is no control rod in channel 1. This is used to simulate the case of a control rod stuck outside the core during a steam line break transient.

The neutronic boundary conditions along the core center lines (x-y plane) are zero neutron current because of the symmetric geometry. The neutronic boundary conditions along the zig-zag side of the core are albedo boundary conditions. As to the top and bottom sides of the core the neutronic boundary conditons are also albedo boundary conditions. The nuclear data (reference cross sections, feedback coefficients, and albedo ratios) are taken from a Brookheaven National Laboratory (BNL) report [24], and converted to TITAN format, as described in Ref. [3,4].

Axially, there are 10 nodes for each channel in addition to the fictitious boundary nodes. The first node (the bottom one) has no fuel rods. This is the node used to simulate part of the lower plenum. The flow split is based on the calculated pressure of the node when the total inlet flow rate boundary condition is chosen (refer to Section 2.3). Channel 1 has quarter size of a normal channel. Channels 2, 3, 4, 5, 8, and 10 have half size of a normal channel. With these arrangements, a quarter core is simulated. Note that this model is not a typical PWR since it has a much smaller radius.

The designed steady state operating conditions of the two models are chosen from Ref. [24]:

1. The steady state core power (quarter core) is 37.95 MW;
2. The total inlet coolant flow rate is 651.2 Kg/sec (quarter core);

37
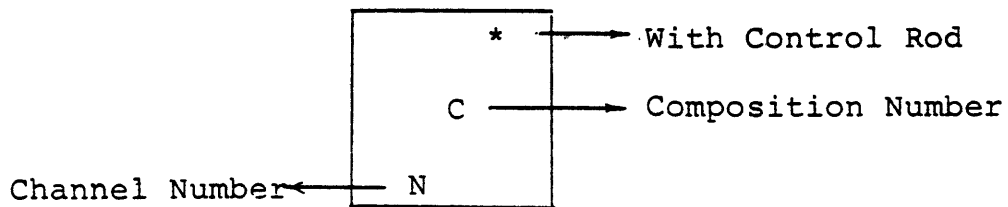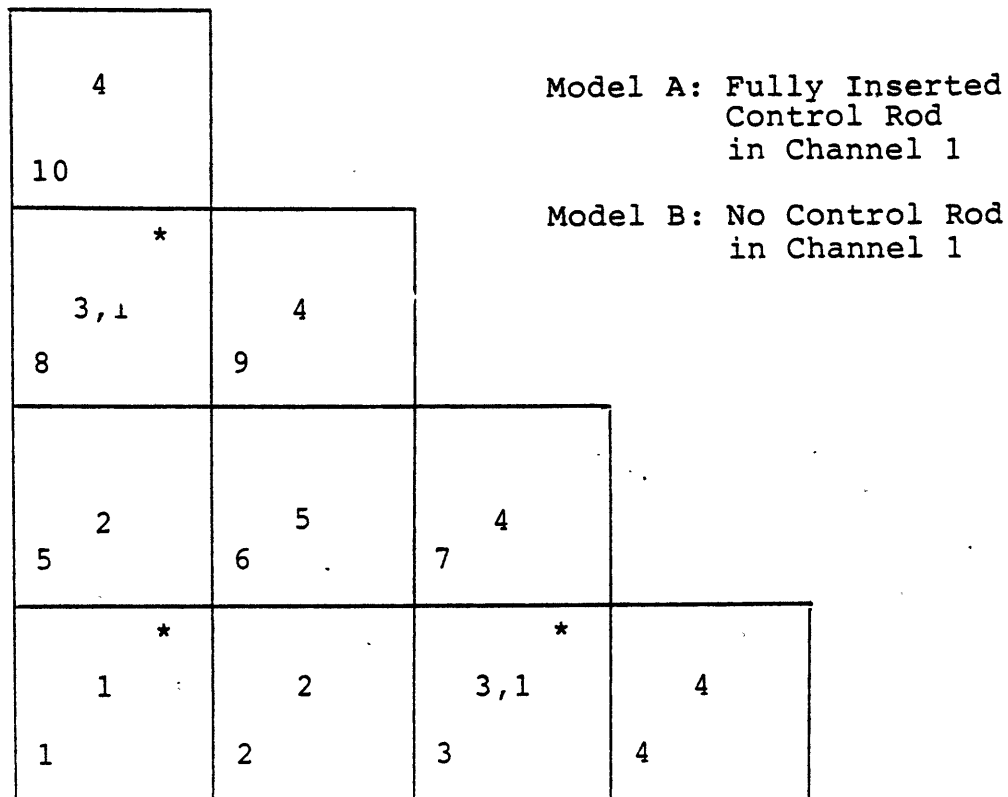
Figure 3.1 Compositon layout for 10-channel steam line
break transient, Models A and B.

Channel 1   Channels 3, 8
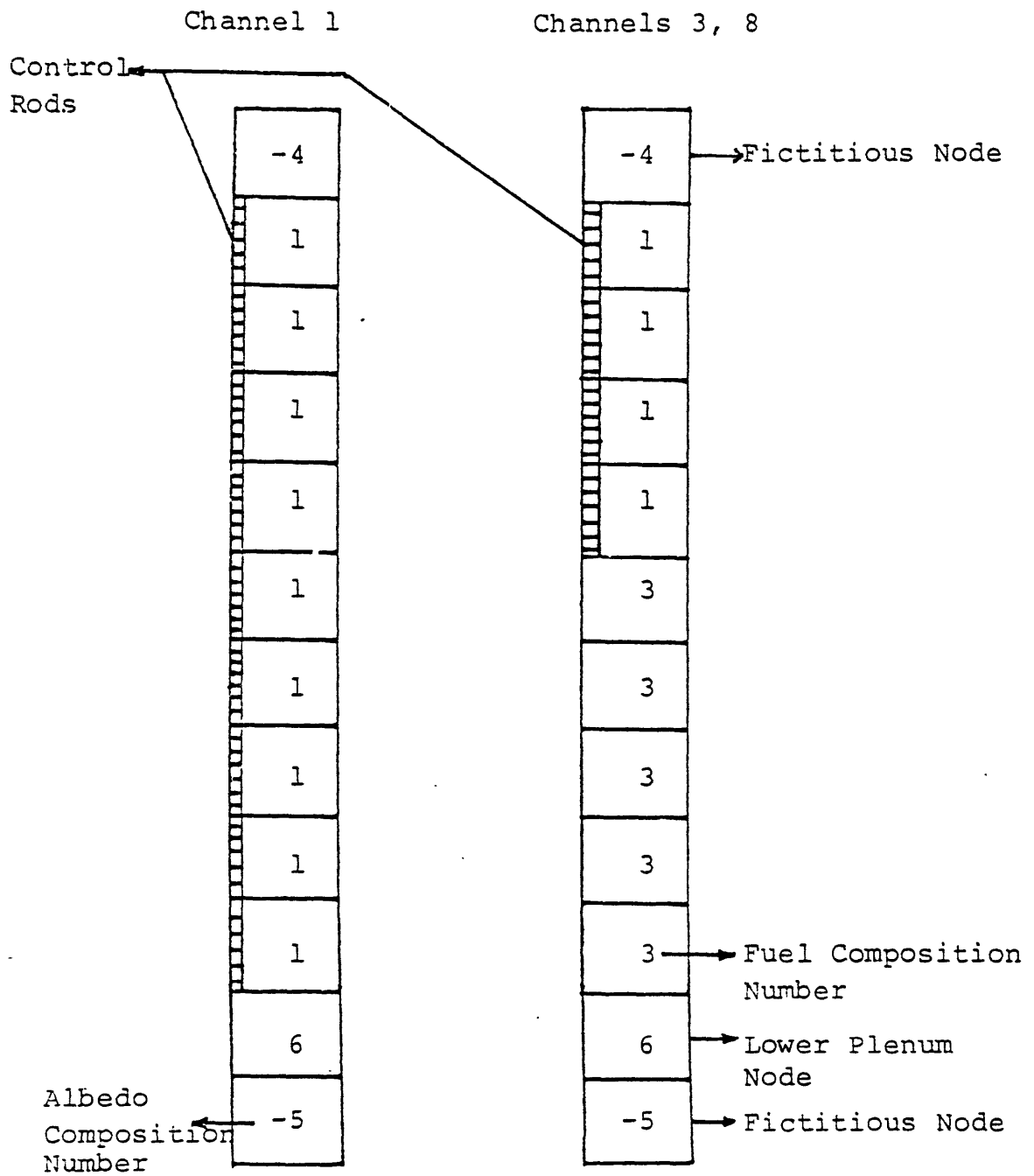
Control Rods

Figure 3.2 Assemblies with control rod inserted.
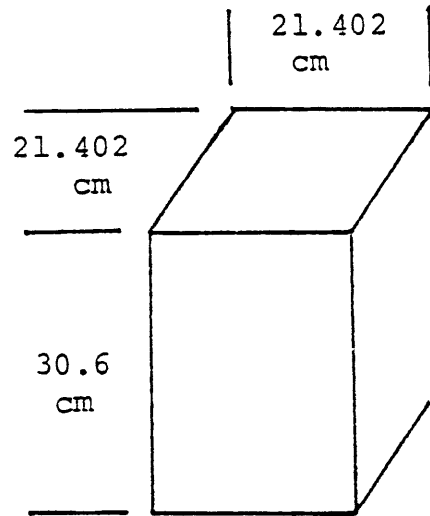
System Pressure: 15.65MPa

Inlet Mass Flow Rate: 651.2 Kg/s

Initial Inlet Temperature: 555°K

Initial Reactor Power: 37.95MW

Figure 3.3 Node dimension and initial conditions

3. The system (exit) pressure is 15.65 MPa;

4. The inlet coolant temperature is $555°K$.

For the thermal-hydraulic calculation, the exit pressure and total inlet coolant flow rate boundary conditions are chosen for top and bottom of the core, respectively. It is assumed that the direct heating in the coolant is 5% of the total power.

For both Models A and B, the convergence criteria are the same:

$$\text{Pressure iteration convergence crit.} = 1.0 \times 10^{-7}$$
$$\text{Newton iteration convergence crit.} = 1.0 \times 10^{-7}$$
$$\text{Eigenvalue convergence crit.} = 1.0 \times 10^{-6}$$
$$\text{Power convergence crit.} = 1.0 \times 10^{-6}$$

3.3 Steady State Results and Discussion

The procedure of performing steady-state calculations is described as follows. At the beginning, all channels are closed and no cross flow among channels is allowed. After several time step calculations, the nuclear cross section average option (see Appendix A for parameter 'ixavg') is used to accelerate the convergence of the results. Once converged closed-channel results are obtained, the calculation is restarted with open-channels since the closed-channel results provide a better initial guess for the open-channel calculation. Finally, converged open-channel results are obtained.

To judge if the steady state result is converged or not, five parameters must be checked. The definitions of the five parameters are:

$$\text{Flow error} = \left(\frac{\text{exit mass flow rate}}{\text{inlet mass flow rate}} \right)^n - 1.0$$

$$\text{Energy error} = \left(\frac{\text{total core enthalpy rise}}{\text{core power}} \right)^n - 1.0$$

$$\text{Win} = \left(\frac{\text{calculated inlet mass flow rate}}{\text{input inlet mass flow rate}} \right)^n - 1.0$$

$$\text{wchk} = \left[\frac{(\text{total core exit mass flow rate})^{n-1}}{(\text{total core exit mass flow rate})^n} \right] - 1.0$$

$$\text{qchk} = \left[\frac{(\text{total core enthalpy rise})^{n-1}}{(\text{total core enthalpy rise})^n} \right] - 1.0$$

where n means at time step n, and $n-1$ means at time step $n-1$.

41

The 'Win' parameter is to be checked when the total inlet flow rate option is chosen. The parameters 'wchk' and 'qchk' are used to ensure that the results are not only converged within a time step (checked by flow and energy errors), but also converged timewise (i.e., real steady state results).

Table 3.1 gives the results of the above five parameters for both Model A and Model B. Closed- and open-channel results are shown in the table for each model. Convergence is achieved with flow errors less than 0.1% and energy errors less than 2%.

Figure 3.4 shows bottom peaking of the axial power distributions of channel 3 for both models. This is expected because of the top inserted control rods. The average power and peak power of channel 3 with Model B is lower than that of channel 3 with Model A. Recall that in Model B, there is no control rod in channel 1. With equal total core powers of the two models, the power of channel 1 with Model B is much higher than that with Model A. Therefore, the power of channel 3 with Model B is lower than that of channel 3 with Model A.

As to the comparison of closed- and open-channel results, no significant difference is found between the axial power distributions as shown in Figure 3.4. The radial assembly power distributions for both models are shown in Figure 3.5. The maximum difference of the distributions between the closed- and open-channel calculations is about 100 W for both models. This is hardly observed in the figure, since the assembly power is higher than 1 MW.

Perhaps the best way to determine if there is a significant difference between closed- and open-channel calculation results is to check the cross flow among channels from the results of open-channel calculations. The cross flow for Model A ranges from 5.6×10^{-3} to 0.7 Kg/sec per node. For Model B, the cross flow ranges from 1.4×10^{-3} to 0.7 Kg/sec per node. The results show that cross flow is small relative to axial flow. Based on the above discussion, it is concluded that there is no significant difference between the steady state results of closed- and open-channel calculations for both models. Therefore, a one-dimensional, multi-channel thermal-hydraulic model should be sufficient for steady state calculations. Furthermore, a fully implicit numerical scheme should be able to speed the calculation compared to the semi-implicit numerical scheme used in the current version of TITAN, since no restrictions would be imposed on the time step size of the fully implicit numerical scheme.

The cpu usages (in a Honeywell 6180 computer with Multics operating system) of the two models are summarized in Table 3.2. The cpu time per time step per node is much higher than the values shown in Table 3.1 of Ref. [21] (for transient test calculations) because steady-state calculations require more iterations than the transient calculations within each

Table 3.1

Steady State Convergence Results of Models A and B

	Model A		Model B	
	Closed-Channel	Open-Channel	Closed-Channel	Open-Channel
Energy Error	-2.1088×10^{-3}	-1.9016×10^{-2}	-1.9183×10^{-3}	-1.9734×10^{-2}
Flow Rate Error	1.1268×10^{-4}	-7.6004×10^{-4}	1.1878×10^{-4}	-7.9465×10^{-4}
Win	4.1357×10^{-6}	-4.9676×10^{-6}	-1.1470×10^{-5}	2.3080×10^{-6}
qchk	3.8178×10^{-5}	-9.9310×10^{-6}	-4.4010×10^{-4}	2.1927×10^{-4}
wchk	-4.1239×10^{-4}	-3.1878×10^{-4}	4.1833×10^{-5}	2.3080×10^{-6}

□ = Model A. Open-Channel
○ = Model A. Closed-Channel
+ = Model B. Open-Channel
× = Model B. Closed-Channel
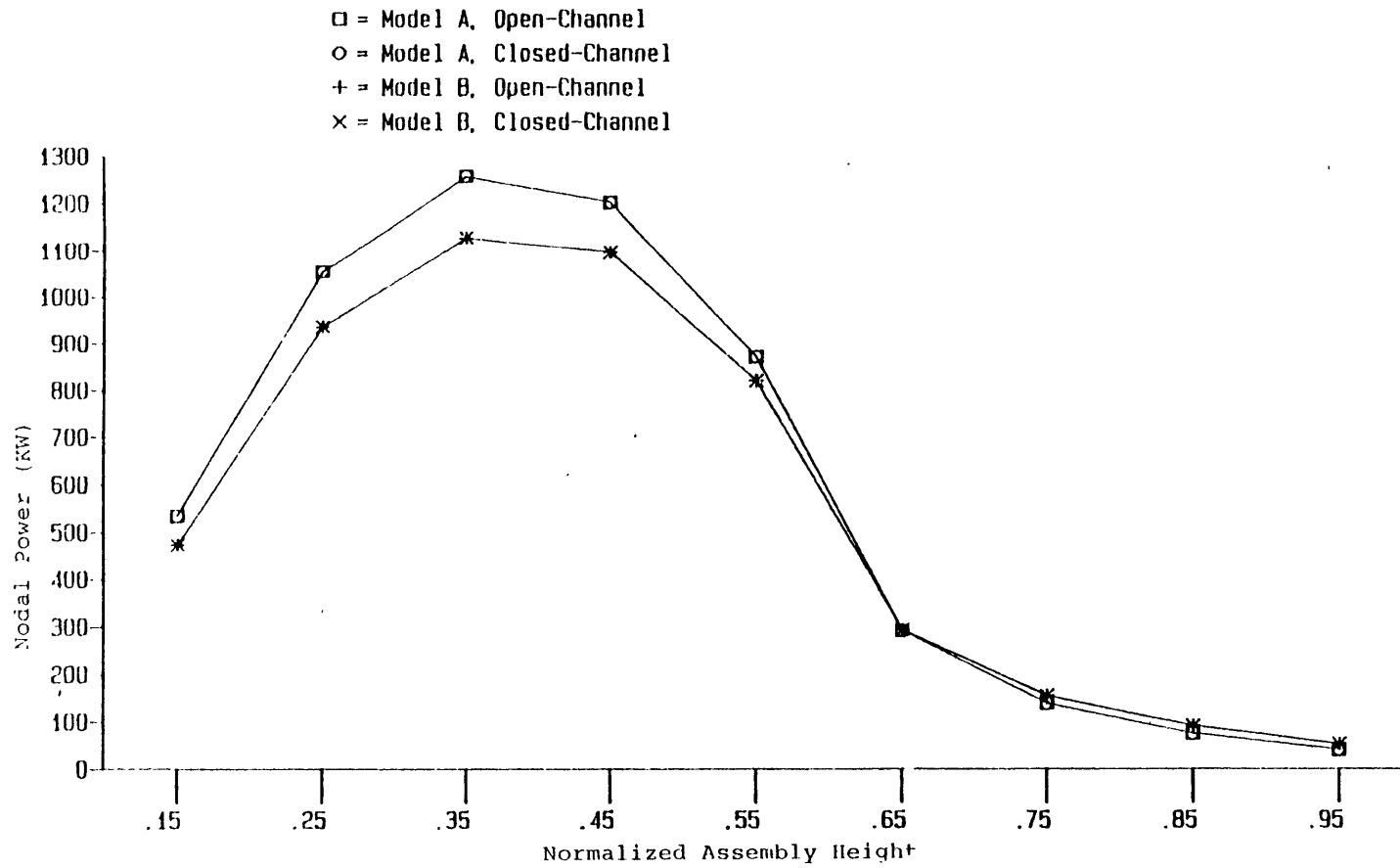
Figure 3.4 Steady state axial power distributions of channel 3, for Models A and B.

44

+ = Model A Closed-Channel
O = Model A Open-Channel
+ = Model B Closed-Channel
□ = Model B Open-Channel
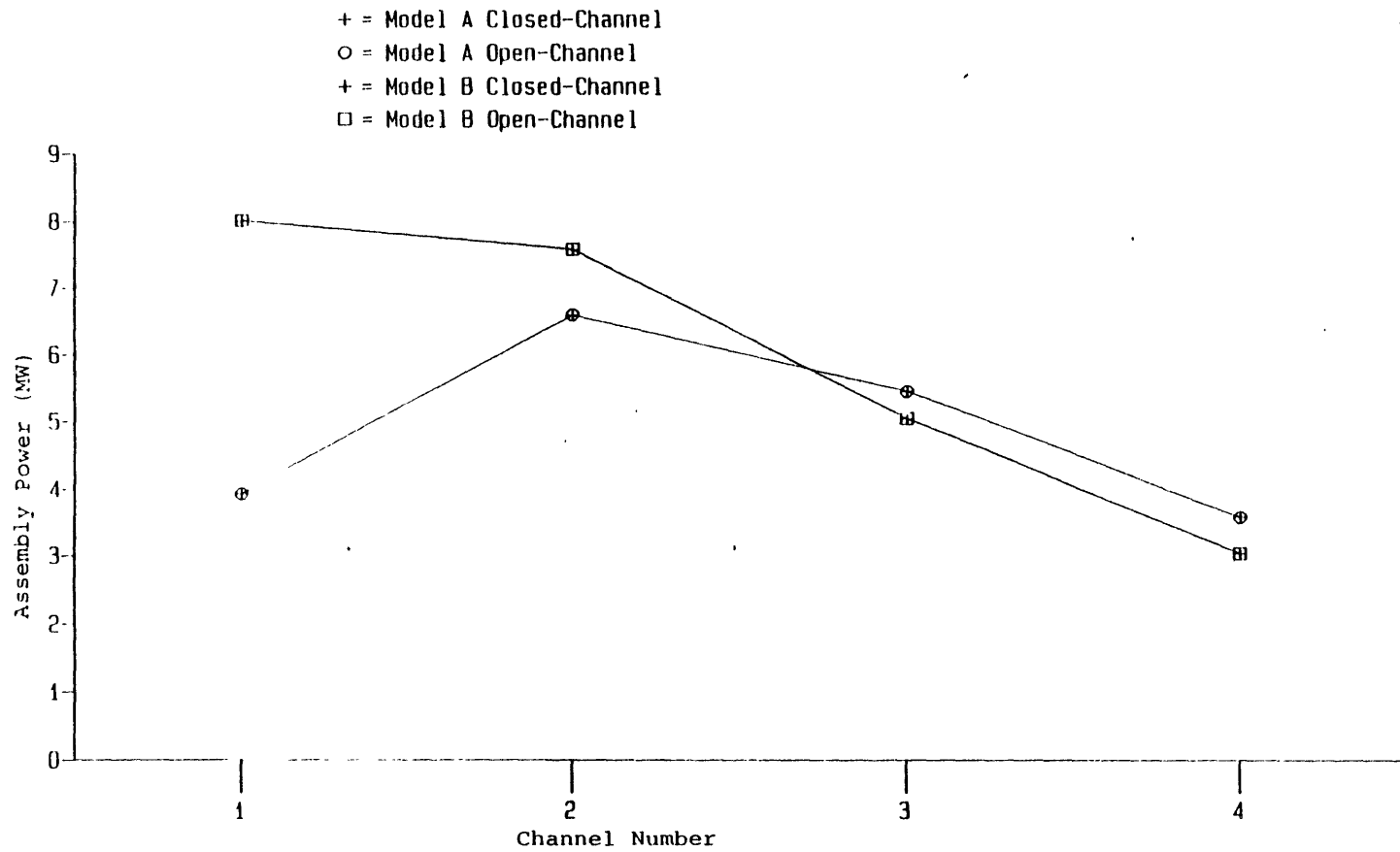
Figure 3.5 Steady state radial assembly power distributions for Models A and B.

Table 3.2

Steady State Computational Usage

Model A	Time Step	CPU Time (sec)	CPU Time/ Time Step/ Node
Closed-Channel	20	527.23	0.2636
Open-Channel	10	246.94	0.2470
Model B			
Closed-Channel	20	647.86	0.324
Open-Channel	10	321.78	0.3218

time step.

An interesting investigation was made to determine the cpu time used for the neutronic calculations in each time step with the help of the parameter 'nfeed' (refer to Appensix A). The function of 'nfeed' can be explained with an example. If nfeed=2, that means there will be 2 thermal-hydraulic calculations per one neutronic calculation. More explicitly, with nfeed=2, the first time step includes both thermal-hydraulic and neutronic calculations. In the second time step, only thermal-hydraulic calculation is performed. In the third time step, both calculations are performed again. To determine the cpu time used for the neutronic calculations within each time step, two calculations were performed. Two time steps were performed in the first calculation with nfeed=1, and the cpu time is for two thermal-hydraulic calculations and two neutronic calculations. Two time steps were performed in the second calculation with nfeed=2 and the cpu time is for two thermal-hydraulic calculations and one neutronic calculation. The difference of the cpu usages of the two calculations is roughly the cpu usage of one neutronic calculation. It was found that with nfeed=1 about 78% of the cpu usage is used for the neutronic calculation in each time step. Therefore, whenever converged and accurate steady-state results can be obtained with 'nfeed' larger than 1, significant cpu time savings will be achieved.

3.4 One-dimensional, Fully Implicit Numerical Scheme and Its Solution Methods

3.4.1 One-Dimensional, Fully Implicit Numerical Scheme

A one-dimensional, fully implicit numerical scheme which was used by Dube [25] was adopted in TITAN as an option for solving thermal-hydraulics equations. The basic thermal-hydraulic fluid dynamic equations are the same as those described in Ref. [21] (equations (2.2) to (2.7)) except the momentum equations are written for z-direction only. Therefore, six scalar equations are left. Altogether, ten unknowns $(\alpha, p, \rho_v, \rho_l, e_v, e_l, T_v, T_l, v_{vz}, v_{lz})$ are in the six scalar equations. The four equations of state (equations (2.14) to (2.17) of Ref. [21]) are required to complete the system of equations for the ten unknowns.

All the ten unknowns are treated fully implicitly in the finite difference equations. All the constitutive terms (the terms on the right hand sides of the conservation equations) are treated fully implicitly except the mass exchange term, Γ, which is evaluated highly implicitly. This is the same as Γ treatment in the current version of TITAN. The meaning of 'highly implicit' is that in the empirical formula of the mass exchange term, some of the variables are treated implicitly while others are treated explicitly. The terms which are evaluated explicitly are the heat transfer coefficients and viscosities. Dube [25] mentioned

47

in his thesis: *'These (heat transfer coefficients and viscosities) may be evaluated implicitly should there be a real requirement, but in practice it has not been found to be important'.* The donor cell concept and the staggered grid technique mentioned in Ref. [21] are adopted in the scheme.

The resultant 1-D finite difference equations are as follows: (Refer to Nomenclature Listing for definitions of symbols)

Vapor Mass

$$\left(\frac{(\alpha\rho_v)^{n+1} - (\alpha\rho_v)^n}{\Delta t}\right)_k + \frac{1}{V}\left\{[A(\alpha\rho_v v_v)^{n+1}]_{k+\frac{1}{2}} - \right.$$

$$\left. [A(\alpha\rho_v v_v)^{n+1}]_{k-\frac{1}{2}}\right\} = \Gamma^{n+\frac{1}{2}} \tag{3.1}$$

Liquid Mass

Same as equation (3.1) with α replaced by $(1 - \alpha)$, subscript v by l and Γ by $-\Gamma$.

Vapor Energy

$$\left(\frac{(\alpha\rho_v e_v)^{n+1} - (\alpha\rho_v e_v)^n}{\Delta t}\right)_k + \frac{1}{V}\left\{[p^{n+1} + (\rho_v e_v)_{k+\frac{1}{2}}^{n+1}]\cdot\right.$$

$$[A(\alpha v_v)^{n+1}]_{k+\frac{1}{2}} - [p^{n+1} + (\rho_v e_v)_{k-\frac{1}{2}}^{n+1} \cdot [A(\alpha v_v)^{n+1}]_{k-\frac{1}{2}}\right\} +$$

$$p^{n+1}\left(\frac{\alpha^{n+1} - \alpha^n}{\Delta t}\right)_k = Q_{wv}^{n+1} + Q_i^{n+1} \tag{3.2}$$

Liquid Energy

Same as equation (3.2) with α replaced by $(1 - \alpha)$, subscript v replaced by l, and Q_i by $-Q_i$.

Vapor Momentum.

$$(\alpha\rho_v)_{k+\frac{1}{2}}^{n+1}\frac{[(v_v^{n+1} - v_v^n]_{k+\frac{1}{2}}}{\Delta t} +$$

$$(\alpha \rho_v v_v)_{k+\frac{1}{2}}^{n+1} (\frac{\Delta v_v}{\Delta z})_{k+\frac{1}{2}}^{n+1} + \alpha_{k+\frac{1}{2}}^{n+1} \frac{(p_{k+1} - p_k)^{n+1}}{\Delta z_{k+\frac{1}{2}}} =$$

$$-(F_{wv})_{k+\frac{1}{2}}^{n+1} - (F_{iv})_{k+\frac{1}{2}}^{n+1} - (\alpha \rho_v)_{k+\frac{1}{2}}^{n+1} g \qquad (3.3)$$

Liquid Momentum

Same as equation (3.3) with α replaced by $(1 - \alpha)$, subscript v replaced by l, and $-F_{iv}$ by F_{iv}.

The superscripts n and $n + 1$ refer to old and new time values, respectively. The superscript $n + \frac{1}{2}$ in the Γ term refers to the highly implicit treatment. The subscripts k and $k + 1$ refer to the axial noding while $k + \frac{1}{2}$ refers to the donor cell concept and staggered grid technique.

Two boundary conditions can be applied at both top and bottom of the core: the pressure and the velocity boundary conditions. The total inlet coolant mass flow rate boundary condition mentioned in Section 2.3 can also be applied to the core inlet in this one-dimensional scheme. The combinations of these options for the top and bottom boundaries are decided by users for various physical conditions.

To solve the above finite difference equations, two solution methods are introduced. They are described in the following sections.

3.4.2 Direct Solution Method

The system of ten scalar equations is non-linear, therefore an exact solution is practically impossible. The first step to approach this problem is to use the four equations of state such that the internal energies (e_v, e_l) and densities (ρ_v, ρ_l) can be represented by pressure (p) and temperatures (T_v, T_l). The remaining equations with six unknowns are linearized by a Newton-Raphson iteration scheme. This is explained as follows:

All the right hand side terms of the six differencing equations are placed to the left hand side. The six equations can be rewritten as:

$$F_1(\alpha, \ p, \ T_v, \ T_l, \ v_v, \ v_l) = 0 \qquad (3.4)$$

$$F_2(\alpha, \ p, \ T_v, \ T_l, \ v_v, \ v_l) = 0 \qquad (3.5)$$

$$F_3(\alpha,\ p,\ T_v,\ T_l,\ v_v,\ v_l) = 0 \tag{3.6}$$

$$F_4(\alpha,\ p,\ T_v,\ T_l,\ v_v,\ v_l) = 0 \tag{3.7}$$

$$F_5(\alpha,\ p,\ T_v,\ T_l,\ v_v,\ v_l) = 0 \tag{3.8}$$

$$F_6(\alpha,\ p,\ T_v,\ T_l,\ v_v,\ v_l) = 0 \tag{3.9}$$

The linearlized form of F_1 for a certain node can be written as:

$$\left(\frac{\partial F_1}{\partial \alpha}\right)^m (\Delta \alpha)^m + \left(\frac{\partial F_1}{\partial p}\right)^m (\Delta p)^m + \left(\frac{\partial F_1}{\partial T_v}\right)^m (\Delta T_v)^m + \left(\frac{\partial F_1}{\partial T_l}\right)^m (\Delta T_l)^m +$$

$$\left(\frac{\partial F_1}{\partial v_v}\right)^m (\Delta v_v)^m + \left(\frac{\partial F_1}{\partial v_l}\right)^m (\Delta v_l)^m = -F_1^m \tag{3.10}$$

where
$$(\Delta \chi)^m = \chi^{m+1} - \chi^m; \quad \chi = \alpha,\ p,\ T_v,\ T_l,\ v_v,\ v_l;$$

m : denotes the old iterate values;

$m + 1$: denotes the new iterate values.

Five similar equations can be written for F_2, F_3, F_4, F_5, and F_6. The matrix form of the linearized equations is:

$$\begin{pmatrix} \left(\frac{\partial F_1}{\partial \alpha}\right)^m & \left(\frac{\partial F_1}{\partial p}\right)^m & \left(\frac{\partial F_1}{\partial T_v}\right)^m & \left(\frac{\partial F_1}{\partial T_l}\right)^m & \left(\frac{\partial F_1}{\partial v_v}\right)^m & \left(\frac{\partial F_1}{\partial v_l}\right)^m \\ \left(\frac{\partial F_2}{\partial \alpha}\right)^m & \left(\frac{\partial F_2}{\partial p}\right)^m & \left(\frac{\partial F_2}{\partial T_v}\right)^m & \left(\frac{\partial F_2}{\partial T_l}\right)^m & \left(\frac{\partial F_2}{\partial v_v}\right)^m & \left(\frac{\partial F_2}{\partial v_l}\right)^m \\ \left(\frac{\partial F_3}{\partial \alpha}\right)^m & \left(\frac{\partial F_3}{\partial p}\right)^m & \left(\frac{\partial F_3}{\partial T_v}\right)^m & \left(\frac{\partial F_3}{\partial T_l}\right)^m & \left(\frac{\partial F_3}{\partial v_v}\right)^m & \left(\frac{\partial F_3}{\partial v_l}\right)^m \\ \left(\frac{\partial F_4}{\partial \alpha}\right)^m & \left(\frac{\partial F_4}{\partial p}\right)^m & \left(\frac{\partial F_4}{\partial T_v}\right)^m & \left(\frac{\partial F_4}{\partial T_l}\right)^m & \left(\frac{\partial F_4}{\partial v_v}\right)^m & \left(\frac{\partial F_4}{\partial v_l}\right)^m \\ \left(\frac{\partial F_5}{\partial \alpha}\right)^m & \left(\frac{\partial F_5}{\partial p}\right)^m & \left(\frac{\partial F_5}{\partial T_v}\right)^m & \left(\frac{\partial F_5}{\partial T_l}\right)^m & \left(\frac{\partial F_5}{\partial v_v}\right)^m & \left(\frac{\partial F_5}{\partial v_l}\right)^m \\ \left(\frac{\partial F_6}{\partial \alpha}\right)^m & \left(\frac{\partial F_6}{\partial p}\right)^m & \left(\frac{\partial F_6}{\partial T_v}\right)^m & \left(\frac{\partial F_6}{\partial T_l}\right)^m & \left(\frac{\partial F_6}{\partial v_v}\right)^m & \left(\frac{\partial F_6}{\partial v_l}\right)^m \end{pmatrix}$$

$$\begin{pmatrix} (\Delta\alpha)^m \\ (\Delta p)^m \\ (\Delta T_v)^m \\ (\Delta T_l)^m \\ (\Delta v_v)^m \\ (\Delta v_l)^m \end{pmatrix} = - \begin{pmatrix} F_1^m \\ F_2^m \\ F_3^m \\ F_4^m \\ F_5^m \\ F_6^m \end{pmatrix} \tag{3.11}$$

The first term on the left hand side of equation (3.11) is the so called Jocobian matrix.

Since the scheme is for one-dimensional problem and by examing equations (3.1) to (3.3), it is found that any given node (except the boundary nodes) is coupled only to its neighbouring two nodes. Therefore, a tri-diagonal supermatrix is formed for N nodes:

$$\begin{pmatrix} [\chi]_1 & [\chi]_2 & & & & \\ [\chi]_1 & [\chi]_2 & [\chi]_3 & & & \\ & [\chi]_2 & [\chi]_3 & [\chi]_4 & & \\ & & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot \\ & & & & \cdot & \cdot & \cdot \\ & & & & & \cdot & \cdot & \cdot \\ & & & & & & [\chi]_{N-1} & [\chi]_N \end{pmatrix} \begin{pmatrix} [Y]_1 \\ [Y]_2 \\ [Y]_3 \\ \cdot \\ \cdot \\ \cdot \\ [Y]_N \end{pmatrix} = \begin{pmatrix} [Z]_1 \\ [Z]_2 \\ [Z]_3 \\ \cdot \\ \cdot \\ \cdot \\ [Z]_N \end{pmatrix} \tag{3.12}$$

where

  $[\chi]_i$ represents the Jocobial matrix of the ith node as shown in equation (3.11);

  $[Y]_i$ represents the six unknown terms of the ith node;

  $[Z]_i$ represents the six residues of the ith node.

Equation (3.12) is solved by factorizing the Jocobian supermatrix into lower and uper triangular supermatrices, and then through a series of matirx manipulations the unknowns, $[Y]$, are found by forward elimination, backward substitution. The iterative process is continued by setting the solved unknowns in this time step as the old values for the next time step and starting the above procedure again until the solved unknowns, $\Delta\chi$, are smaller than some specified convergence criteria.

The shortcoming of the method is the slow convergence characteristics as mentioned by Dube [25]. However, it is still twice as fast as the semi-implicit method for steady state calculations in THERMIT [25].

3.4.3 Marching Method

Another approach to solving the difference equations (eq.'s (3.1) to (3.3)) is the so called marching method. The main difference between the marching method and the direct method described in the previous section is that instead of solving all N nodes at once with

a supermatrix form equation (eq. (3.12)), the marching method solves the equations node by node. This is explained as follows:

Equation (3.11) is obtained first for all nodes with the specified boundary conditions. With the initial guess of the unknowns, the first node is solved by inverting the 6×6 Jocobian matrix and multiplying the inverted matrix by the residue vector for the ten unknowns. Then, an iteration process is involved until $\Delta \chi$ of the first node is less than some specified convergence criteria. The next step is to proceed the same procedure to the next node with the solution of the previous node as its boundary conditions. This process marches upward to the last node of the channel. With the specified boundary conditions of the last node, the process marches downward to the first node again. In this way, the upper plenum boundary information is related to the solution. The upward and downward march processes continue alternatively until the steady state convergence criteria are satisfied.

The advantage of the marching method is that it converges much faster than the direct method. However, the method cannot treat reverse flow conditions within the one-dimensional geometry. This is because the method assumes a priori that flow is in one direction as the procedure 'marches' from one node to the next.

3.5 Steady State Results of the One-Dimensional, Fully Implicit Numerical Scheme

The ten-channel PWR quarter core Model B described in Section 3.2 was employed for the test problems of the new adopted scheme. The input variables of the test problems are the same as the input variables of the steady state calculations described in Section 3.2 and 3.3 except the new numerical scheme was applied.

Two calculations were performed for the same test problem. One employed the direct method while the other employed the marching method. No nuclear cross section average process (ixavg=0, see Section 3.3 and Appendix A) was used for both of the two calculations. Table 3.3 shows the covergence results of the two calculations. It is clear from the table that with 4 time step calcualtions the cpu usages of both methods are very close, however, the marching method has better converged results than those of the direct method. This verifies the conclusion made by Dube [25] that the direct method converges slower than the marching method.

The calculation with the marching method was continued two more time steps, restarting from the results of the third time step of the marching method calculation mentioned above.

Table 3.3

One-Dimensional, Fully Implicit Numerical Scheme Steady State Convergence Results

	Direct Method	Marching Method
Time Steps	4	4
CPU Time (sec)	215.43	220.08
CPU Time/Step/Node	0.5386	0.5502
Energy Error	-0.40715×10^{-2}	-0.2089×10^{-2}
Flow Rate Error	0.54751×10^{-3}	0.79085×10^{-5}
Win	-0.40701×10^{-4}	-0.40678×10^{-4}
qchk	-0.63832×10^{-1}	-0.37631×10^{-3}
wchk	-0.37350×10^{-2}	-0.10170×10^{-4}

The nuclear cross-section-average process was included in the two time step calculations. Table 3.4 gives the convergence results of the original semi-implicit scheme and the fully-implicit, marching method scheme. The results of the fully implicit, marching method scheme are much more converged than those of the semi-implicit scheme. In addition, the cpu usage of the fully implicit, marching method scheme is less than half of that of the semi-implicit scheme.

Another important item to be checked is whether the results calculated by the two methods are consistent with each other. Figure 3.6 shows the radial power distributions calculated by the two methods. The axial nodal power distributions of the two methods are shown in Figure 3.7. It is clear that the two distributions agree with each other. The two figures lead to the conclusion that not only the global results (assembly power distribution) but also the local results (nodal power distribution) calculated by the fully implicit, marching method match the results of the semi-implicit method and the results are reliable.

3.6 Summary

In this chapter the results of steady state calculations of two 10-channel PWR models (A and B) with semi-implicit numerical scheme for the thermal-hydraulics calculations are presented. Both open- and closed-channel calculations were performed. The open-channel calculation was restarted from the results of closed-channel calculation. The bottom peak axial nodal power shape of channel 3 correlates the fact that partially top inserted control rod exists in the channel. The cpu usages of the closed-channel calculations for Models A and B are 68% and 67% of the total cpu usages of the open-channel calculations, respectively. No significant differences between the results of the two calculations for both Models A and B are found. A fully implicit numerical scheme is thus appropriate for the thermal-hydraulics steady state calculations to save cpu usages.

A one-dimensional, fully implicit numerical scheme (similar to Dube's [25]) is adopted by TITAN as an option for solving the thermal-hydraulics equations. The donor cell concept and staggered grid technique are used in the scheme. Two solution methods were used to solve the finite difference equations. The direct method solves the equations with the following procedure:

    1. Linearize the equations with a Newton-Raphson iteration scheme, and construct a 6 × 6 Jacobian matrix for a node.

    2. Construct a tri-diagonal supermatrix for all one-dimensional nodes.

    3. Factorize the super-Jacobian matrix into upper and lower triangular supermatri-

Table 3.4

Convergence Results of Fully Implicit, Marching Method and Semi-Implicit Method

	Fully Implicit	Semi-Implict
Time Steps	5	20
CPU Time (sec)	265.50	647.86
CPU Time/Step/Node	0.531	0.324
Energy Error	-0.24860×10^{-2}	-0.19183×10^{-2}
Flow Rate Error	-0.26127×10^{-5}	0.11878×10^{-3}
Win	0.10541×10^{-6}	-0.11470×10^{-4}
qchk	0.38251×10^{-3}	-0.44010×10^{-3}
wchk	-0.29091×10^{-4}	0.41833×10^{-4}

+ = Semi-Implicit Method

□ = Fully-Implicit, Marching Method
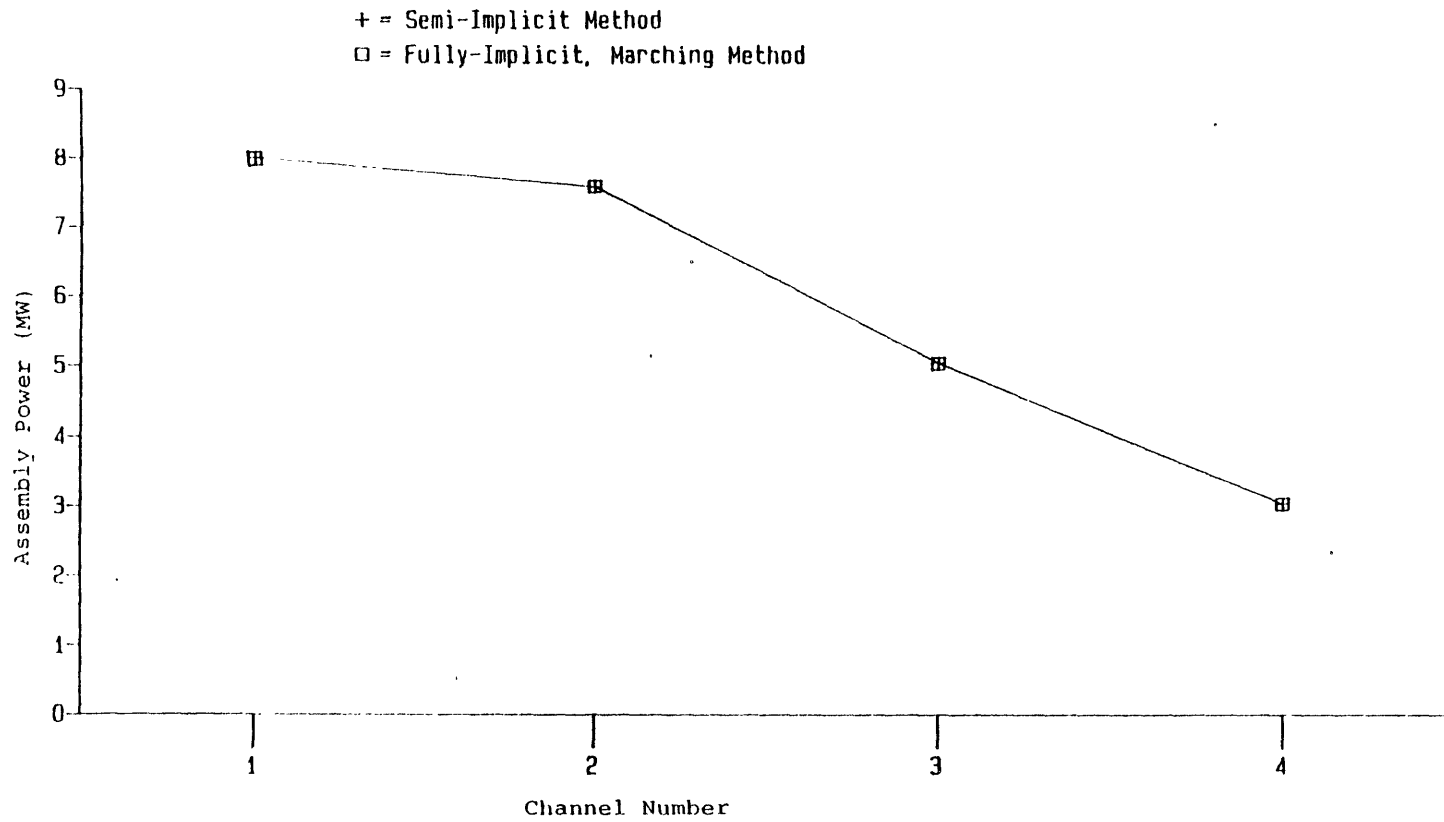
Figure 3.6 Steady state radial assembly power distributions of two numerical schemes.

+ = Semi-Implicit Method

□ = Fully-Implicit. Marching Method
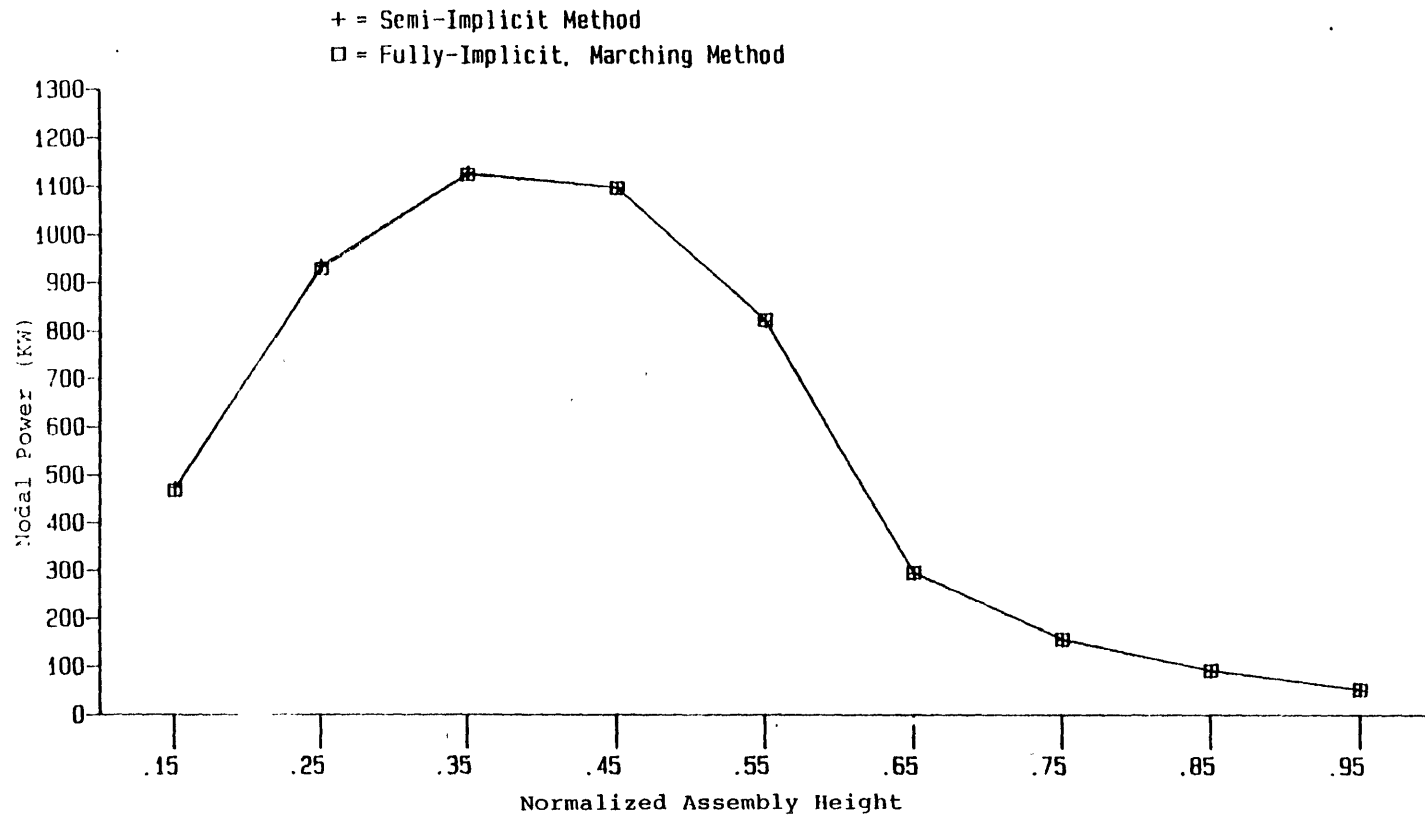
Figure 3.7 Steady state axial nodal power distributions of two numerical schemes.

ces.

4. Use forward elimination, backward substitution method to solve for the unknowns.

5. If the convergence criteria are not reached, the super-Jacobian matrix and the super-residue vector are updated and steps 3 and 4 are repeated. This process continues until the specified convergence criteria are met.

The marching method solves the finite difference equations node by node as follows:

1. The 6×6 Jacobian matrix for a node is constructed as the first step of the direct method.

2. With boundary conditions and initial guess of the variables, the unknowns of the first node are found by inverting the Jacobian matrix and multiplying it by the residue vector. This step is repeated until converged results are obtained.

3. The unknowns of the second node is then solved as described in step 2 with the results of the first node as boundary conditions.

4. The process of step 3 marches to the last node and marches back to the first node with the last node boundary condition included.

5. Step 4 continues until converged results are found.

The results presented in this chapter show that the direct method converging rate is slower than the marching method. The marching method uses less than half of the cpu time needed by the semi-implicit method to obtain much better converged steady state results. The results agree very well with the results of the semi-implicit method for both global and local power distributions.

Chapter 4

TRANSIENT SIMULATIONS; MDNBR STUDIES

4.1 Introduction

In the previous chapters, the TITAN code has been reviewed and the requirements of a code to simulate a steam line break accident have been discussed. Accordingly, the necessary development of the TITAN code has been made. Testing calculations involving the physical phenomena of a steam line break accident have been performed. The results of these testing calculations verify the ability of the code to predict reasonably accurate results of a steam line break accident.

The converged steady state results of two ten-channel PWR quarter core models, A and B, are obtained as described in Chapter 3. The steady state results of Model A are used as the initial conditions of the testing calculations mentioned in Chapter 2. The transient calculations which will be discussed in this chapter are initiated from the steady state results of Model B. As mentioned in Chapter 3, Model B consists of ten PWR channels with channels 3 and 8 including 45% paritally inserted control rods. There is no control rod in channle 1, while Model A. During the period of a transient, the partially inserted control rods of channels 3 and 8 are inserted into the core to simulate a scram action. Channel 1 remains at its original status, i.e., no control rod is inserted during the transients. The arrangement is used to simulate a stuck control rod outside the core in a steam line break accidnet.

The assumptions of two broken steam line loops (north and south loops, refer to Chapter 2) and identical transients of the two loops separate the whole core into four symmetric quarter cores. Model B simulates one of the four quarter cores and three core inlet regions are partitioned to correspond to the broken loop (cold) region, the mixing region, and the intact loop (hot) region. Channels 2, 3, 4, and 7 are in the hot region. Channels 1 and 6 are in the mixing region, and channels 5, 8, 9, and 10 are in the cold region. The arrangement is shown in Figure 4.1.

The objectives of this chapter are to present the effects of three-dimensional calculations on the results after boiling occurs. Notice that one of the conclusions stated in Chapter 3 that there is no significant difference between results of closed- and open-channel calculations was made under the condition that no boiling is observed. It is possible that once boiling occurs corss flow between channels should become significant. Global parameters (such as
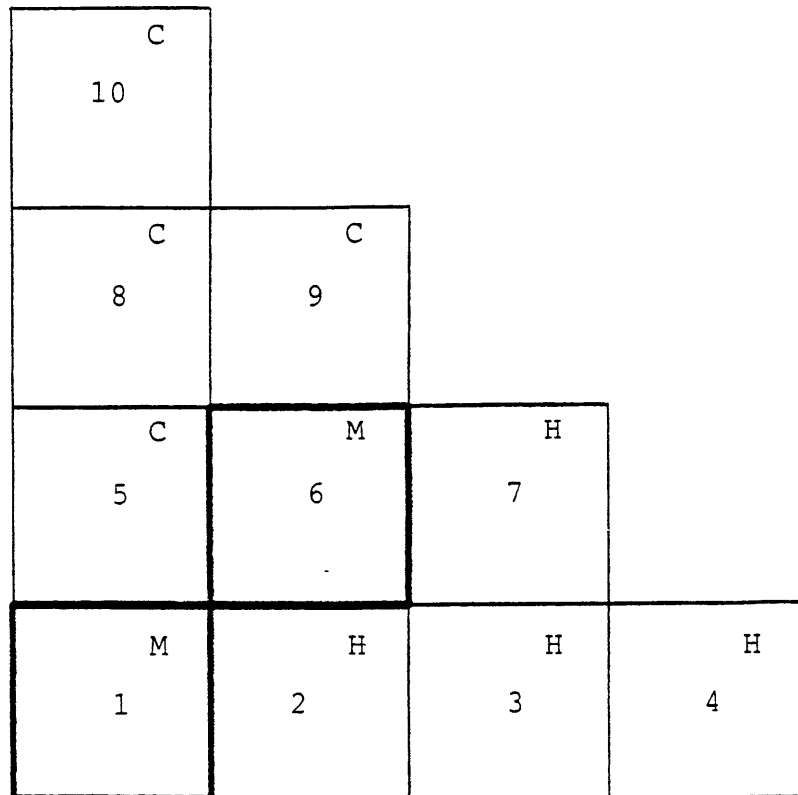
C: Cold Region

M: Mixing Region

H: Hot Region

Figure 4.1   Model B core inlet arrangement.

the total power history) and local parameters (such as nodal power and Minimum Departure from Nucleate Boiling Ratio (MDNBR)) are to be studied. The results of the studies will provide information about the corss flow affecting local parameters, determining whether the fuel rods are in danger of suffering damages or not. There is no return to power in the studies to be described in this chapter. Chapter 5 will discuss the return to power conditions of a steam line break accident in detail.

The transient calculations are initiated from the steady state results of Model B as mentioned earlier. The steam line break accident is simulated by the following simultaneous transients:

1. Inlet coolant temperature transient;

2. Exit pressure transient;

3. Inlet coolant mass flow rate transient;

4. Scram;

5. Boron-10 solution injection.

The boron injection action is not simulated in the studies of this chapter, since there is no return to power condition observed in these calculations. A summary for all transients to be reviewed in this chapter is provided in Section 4.6.

4.2 Simulations with the Original Yankee Atomic Pressure Forcing Function YAT1X

The forcing functions of the four transients describing the SLB accident are taken from a Yankee Atomic report [6], and the calculations are identified as YAT1X (Yankee Atomic Transient case 1X). Figure 4.2 shows the inlet coolant temperature forcing functions of the three core inlet regions. The mixing region forcing function is the average of the functions of the other two regions. After ten seconds of a null transient, the temperatures keep dropping as expected for a steam line break accident. An exit pressure forcing function is applied to all the ten channels, i.e., the upper plenum pressure distribution is uniform. Figure 4.3 shows the normalized pressure forcing function and the initial pressure is 15.65 MPa. The decreasing pressure reflects the shrinkage of the coolant inventory because of the cool down process. The increasing total inlet coolant mass flow rate before 60 seconds as shown in Figure 4.4 reflects the density increase of the coolant and the untripped Reactor Coolant Pump (RCP). At 60 seconds the RCP is tripped and results in a sharp flow rate drop accordingly. The initial total inlet coolant mass flow rate is 651.21 Kg/sec for the quarter core model. The last transient involved in a steam line break-accident studied in this chapter is the scram transient. The action of scram starts at 0.9 seconds after the initiation of the steam line

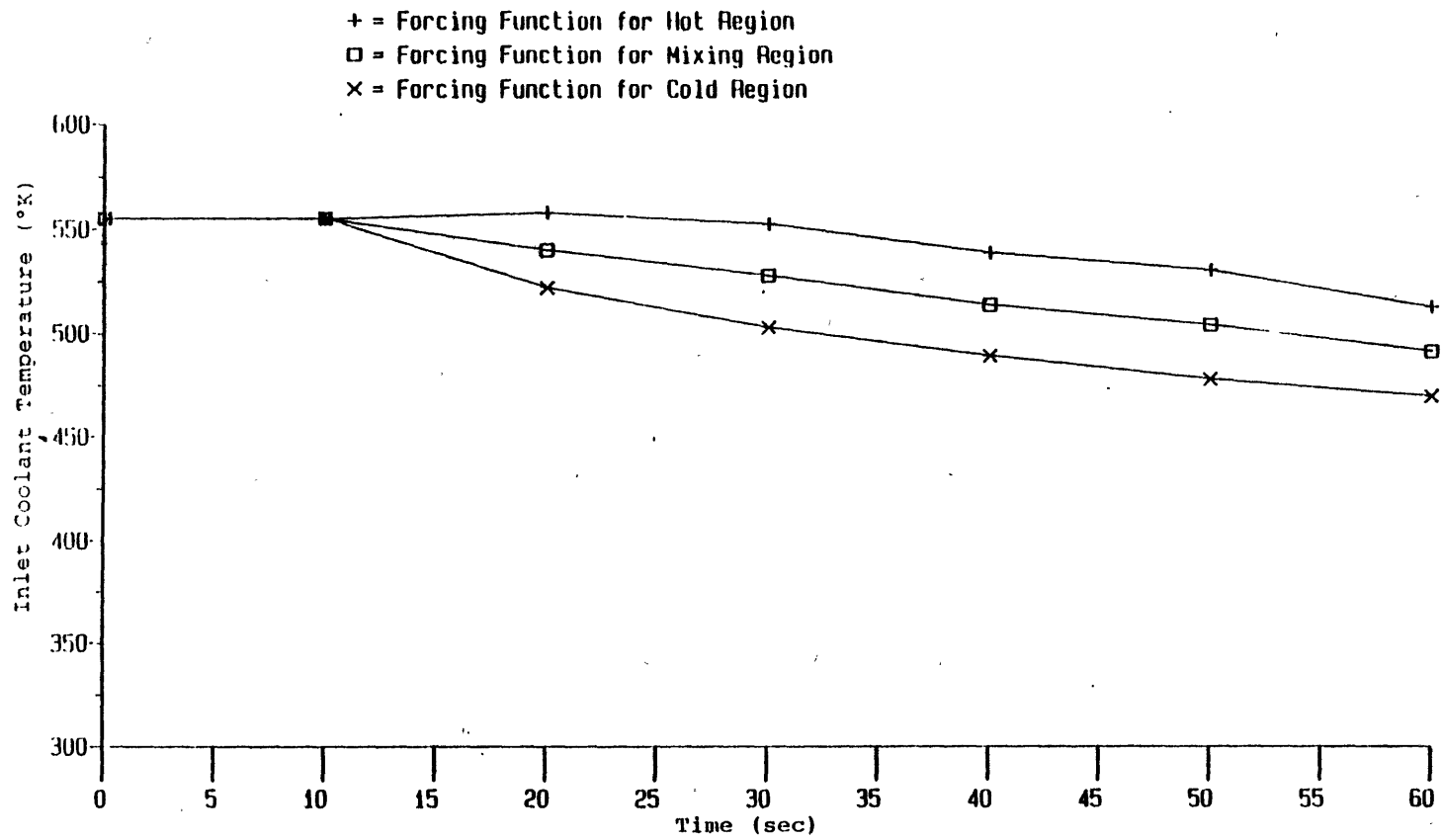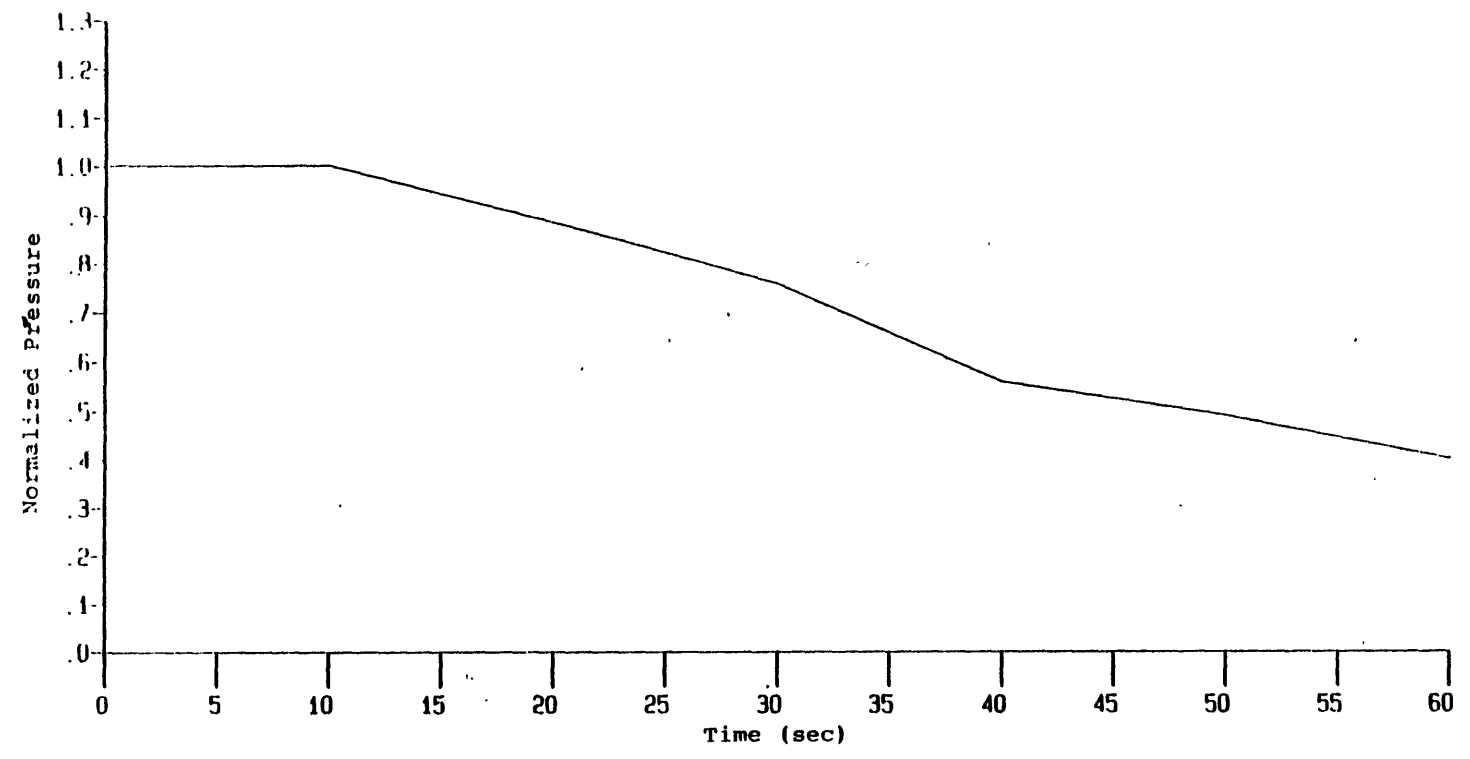Figure 4.2 Inlet coolant temperature forcing functions for cases YAT1X, YAT2X, YAT3X.

Figure 4.3 Pressure forcing function for case YAT1X.
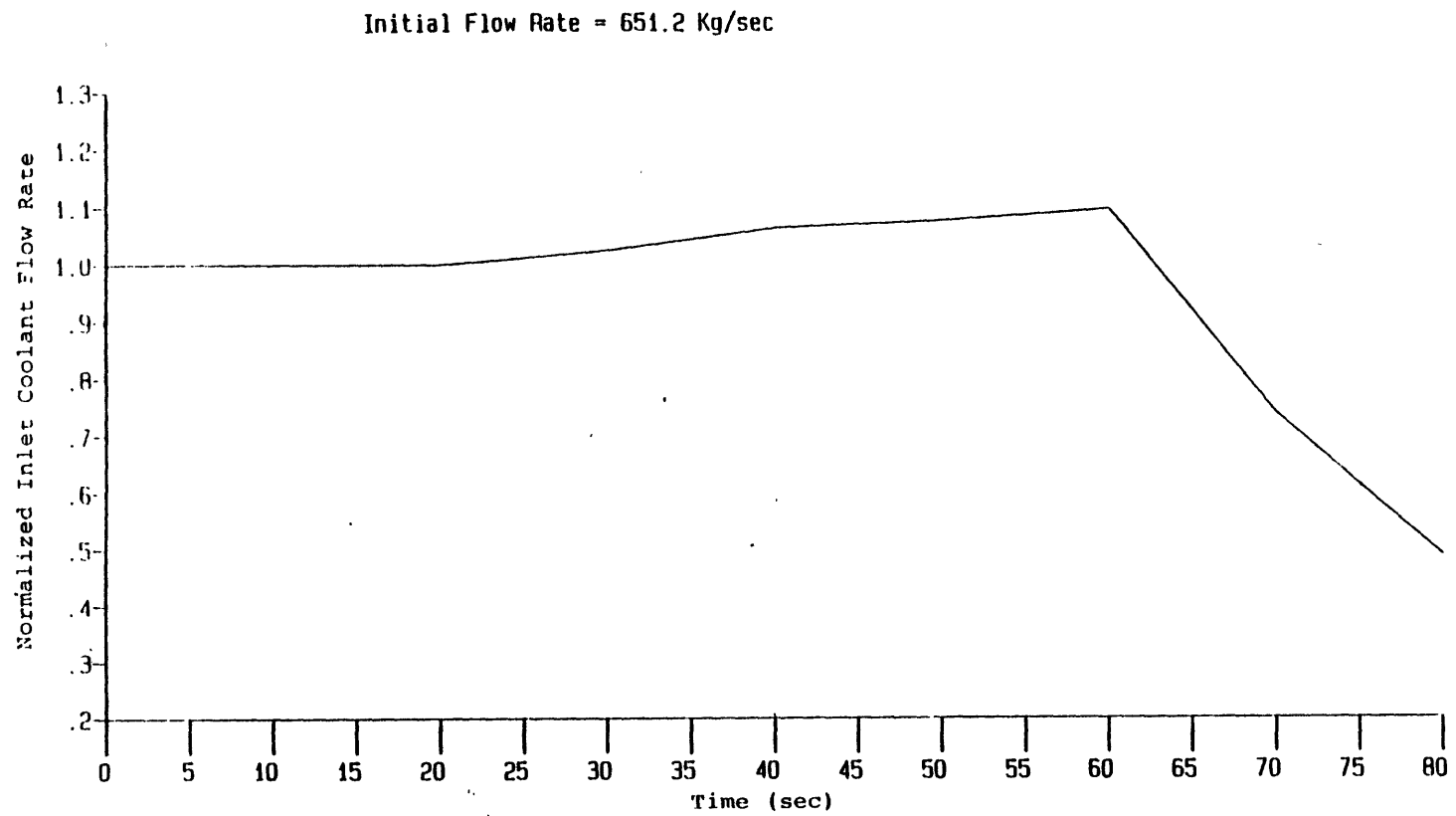
Initial Flow Rate = 651.2 Kg/sec

Figure 4.4 Inlet coolant mass flow rate forcing function for cases YAT1X, YAT2X, YAT3X.

break accident (i.e., at 10.9 seconds which includes the 10 seconds null transient period). After 5 seconds, the scram action is completed. The control rods insertion speeds in both channels 3 and 8 are constant and equal.

The transient is followed up to 50 seconds with a closed-channel calculation, YAT11. No boiling is found from the results of the calculation. This is the combined result of the four forcing transients. The inlet coolant temperature and mass flow rate transients introduce positive reactivity to the core. However, they reduce the possibility of boiling because of the addition of the colder coolant to the core. On the other hand, the pressure transient favors the formation of vapor voids since it is a depressurization transient. But one should remember that the depressurization arises from the coolant inventory shrinkage. The scram transient simply adds negative reactivity to the core and of course works against the formation of vapor voids. Figure 4.5 shows the coolant saturation temperature and the hottest fuel wall temperature (node (2,5)) histories of case YAT11. The result shows that the coolant saturation temperature and fuel wall temperature approach each other. This implies that had the calculation been continued, the fuel rod surface temperature will become higher than the coolant saturation temperature and local boiling may occur. The calculation was not continued since it seemed that, under the prevailing conditions, to reach a boiling state would be a very lengthy calculation. The cpu time for case YAT11 is 8103.04 seconds in a Honeywell 6180 computer. Eight hundred time steps were performed. In order to study the boiling effects, a reduced pressure forcing function is employed to speed the vapor voids formation. This will be presented in the following sections.

Before going into the next section, it is necessary to check the results of case YAT11 since this is the first calculation involving the four forcing transients mentioned in Section 4.1. Figure 4.6 shows the total power history. The slope of the power history within the period 10 to 15 seconds is larger than that of the rest transient. This reflects the scram effect during the 5 seconds period. the smaller and smaller slope of the power history shows the effect of inlet coolant temperature transient to compensate the negative reactivity insertion by scram. Figures 4.7 and 4.8 show a more local view of the results. The radial assembly power distributions at different times in both the hot and cold regions show the scram effect particularly in channels 3 and 8 where the scram action happens. In addition, the assembly powers in Figure 4.8 (cold region) are higher than those in Figure 4.7 (hot region) which is expected due to the coolant temperature feedback. Figure 4.9 shows the local axial nodal power distributions of channels 3 and 8 at 50 seconds. Again, the nodal powers of channel 8 are higher than those of channel 3 as expected.

Since the calculation is still in the single liquid phase heat transfer region, the MDNBR's predicted by YAT11 are much greater than one (99, set by TITAN) throughout the calcu-

+ = Saturation Coolant Temperature   at Node (2, 5)
□ = Fuel Surface Temperature

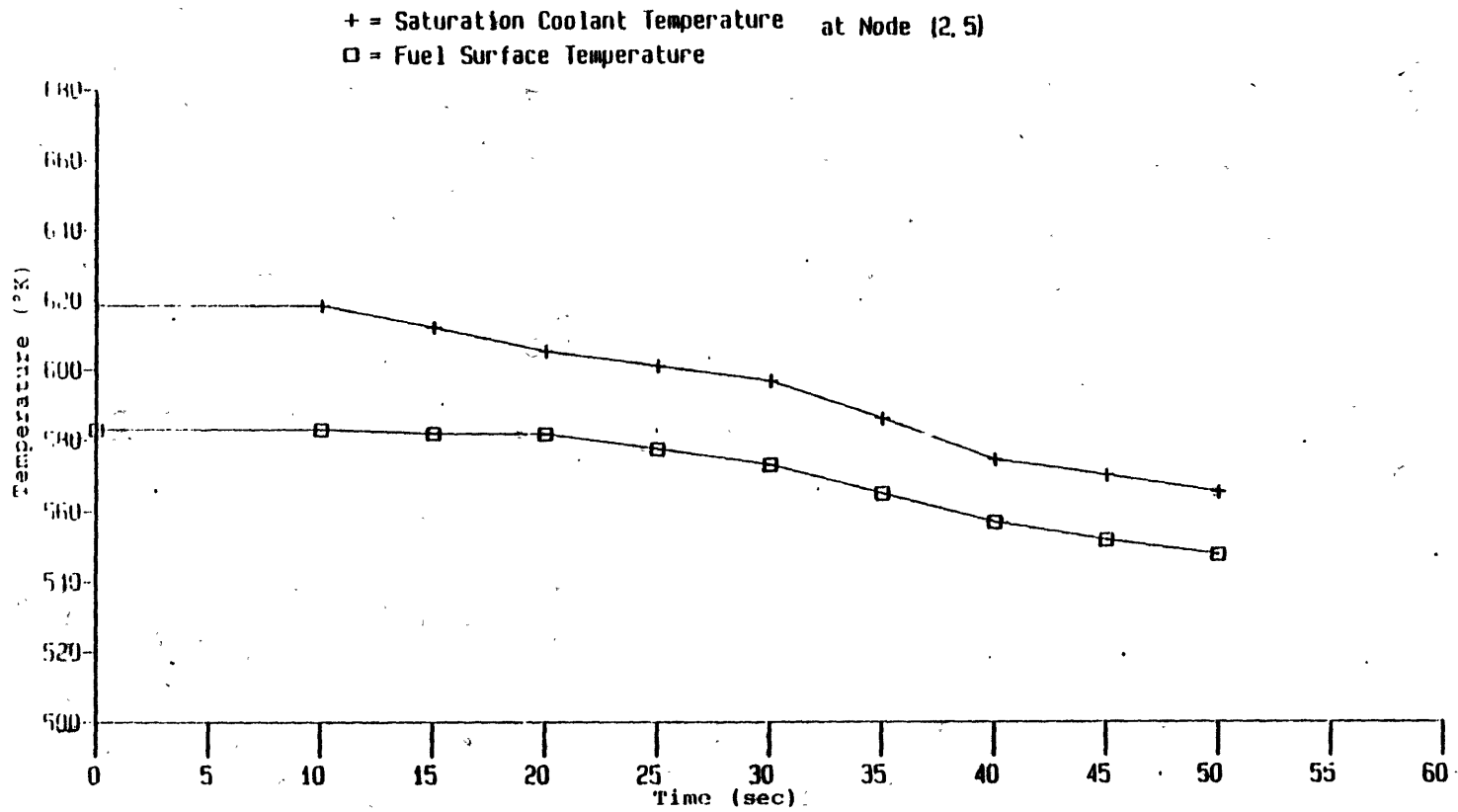Figure 4.5   Saturation coolant and fuel surface temperature histories of node (2,5).
             YAT11.
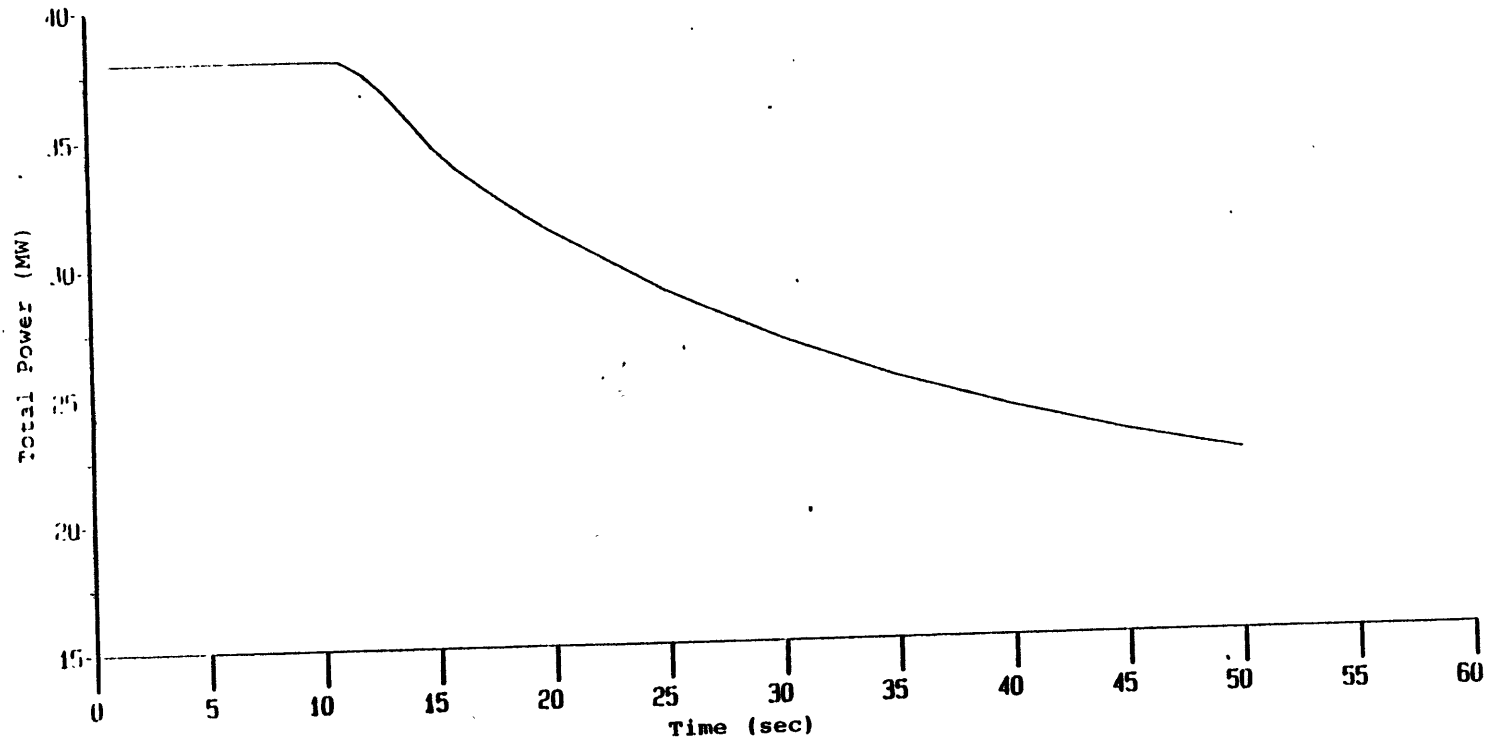
Figure 4.6 Total power history of YAT11.
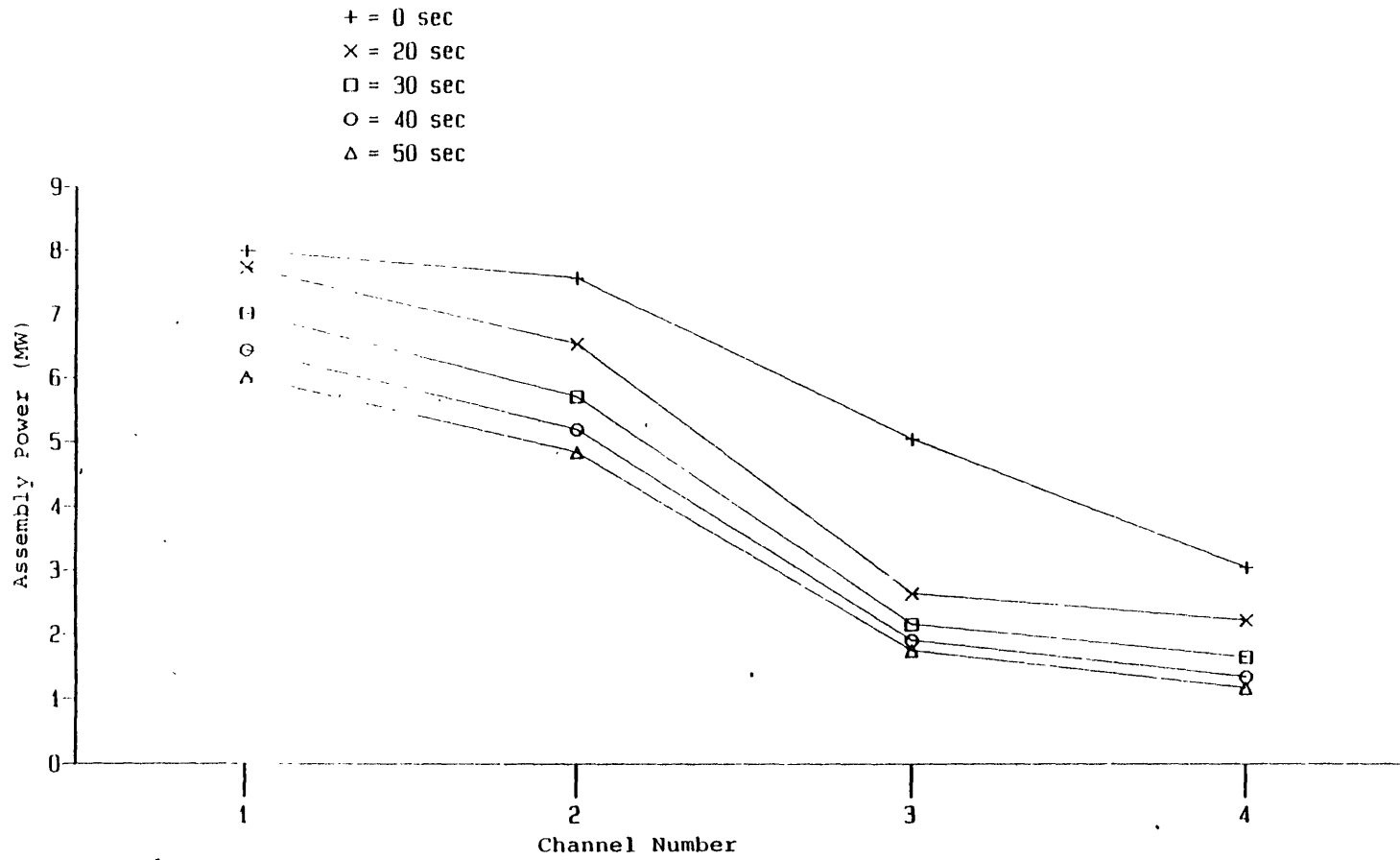
Figure 4.7 Radial power distributions, channels 1, 2, 3, 4; YAT11.

+ = 0 sec
× = 20sec
⊔ = 30 sec
○ = 40 sec
△ = 50 sec
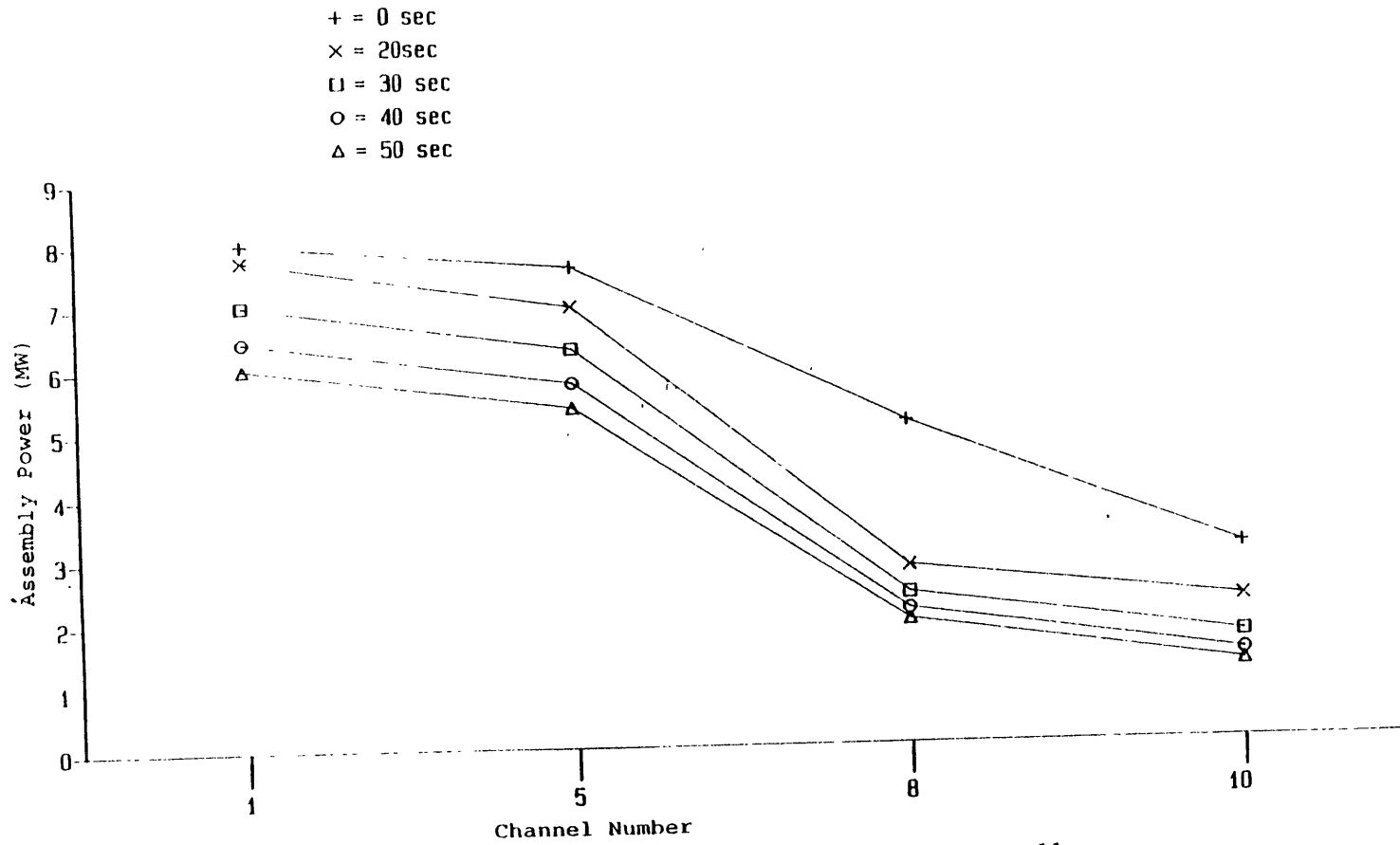
Figure 4.8 Radial power distributions, channels 1, 5, 8, 10; YAT11.
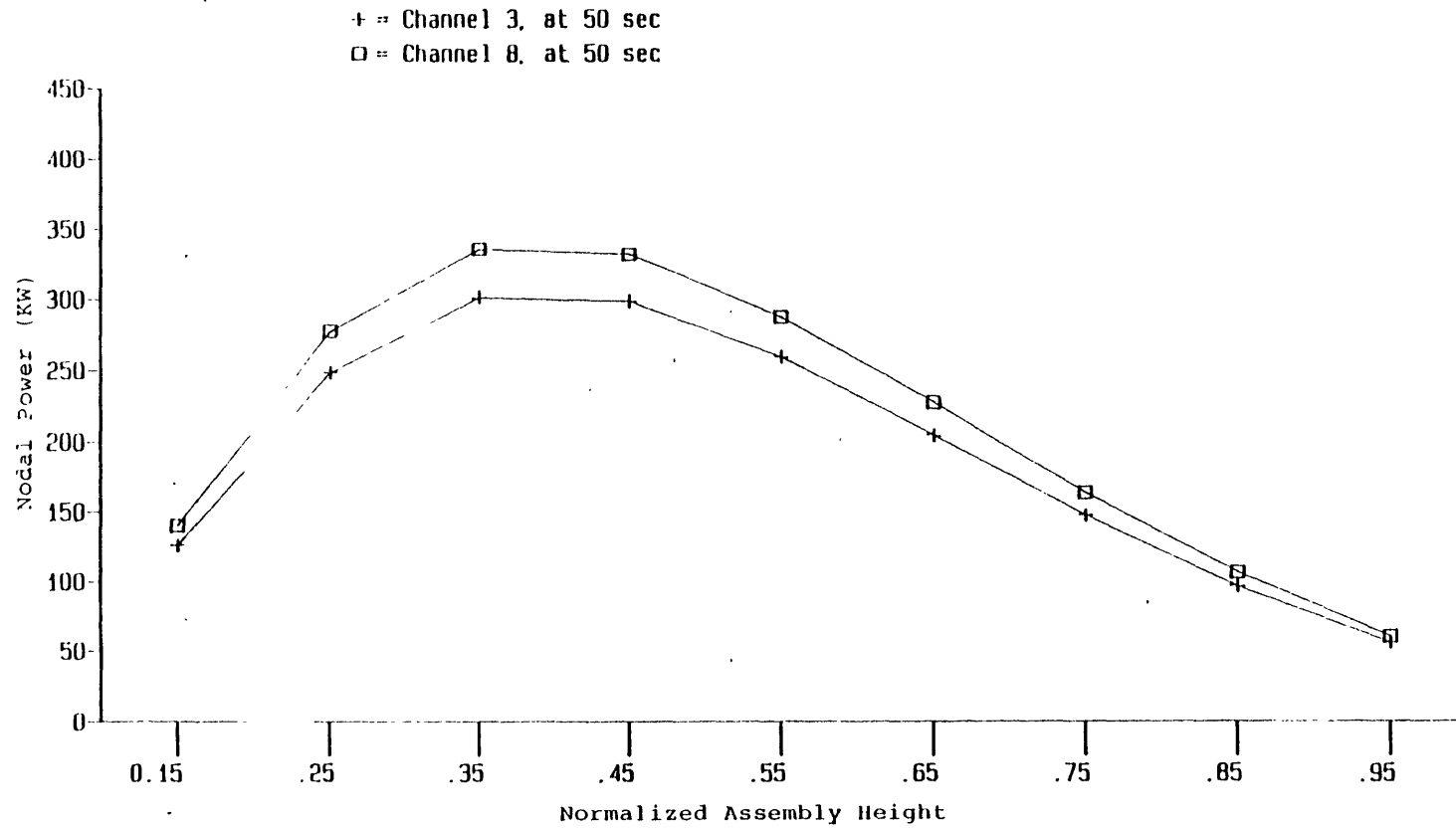
Figure 4.9   Axial nodal power distributions of channels 3 and 8, case YAT11.

lation. However, it is expected that once boiling occurs, significant differences of the local parameters between the closed- and open-channel calculations will appear.

4.3 Simulations with the Reduced Pressure Forcing Function YAT2X

Since three-dimensional thermal-hydraulic effects are of interest in a steam line break transient, particularly once boiling starts, reduced pressure forcing function cases (YAT2X) were performed to accelerate the boiling phenomena. Figure 4.10 shows the pressure forcing function for YAT2X. The initial pressure is still 15.65 MPa. The pressure after 45 seconds is kept constant because of the limitation of the data base of the W-3 CHF correlation (around 5.5 MPa) used in the calculations. All other conditions are the same as those employed for case YAT11 calculation.

Both closed- (YAT21) and open-channel (YAT22) cases were performed. They were restarted from the steady state results of Model B as described before. They were followed up to 50 seconds. Figure 4.11 shows the two total power histories of the two calculations. It is clear from the figure that before 45 seconds almost no difference is found between the two curves while a slight difference is found after 45 seconds. Boiling is found to start at 40.7 seconds and 42.05 seconds for YAT21 and YAT22, respectively. Two points can be drawn from the results:

1. Before boiling occurs, the global parameter (total power) is not strongly affected by closed- or open-channel calculations. This is expected since all the involved transients are symmetric with respect to the x-y plane of the core except the coolant inlet temperature transient. It has been concluded that no significant difference between the two calculations will be induced by the transient before boiling occurs.

2. After boiling occurs, the global parameter (total power) is slightly affected by the open-channel calculation. However, the effect is not significant.

The radial assembly power distributions of the two calculations are shown in Figures 4.12 to 4.15. The first result observed from these figures is that the assembly powers in the cold region (channels 5, 8, 10) are higher than those in the hot region (channels 2, 3, 4) as expected. Also, the scram effects are still shown in channels 3 and 8 for both calculations. In addition, by comparing the results of YAT21 and YAT22 (Figures 4.12 vs. 4.14 and Figures 4.13 vs. 4.15), the difference between the closed- (YAT21) and open-channel (YAT22) is not large, as also concluded from the total power histories. Figures 4.16 and 4.17 show the axial nodal power distributions of channel 2 for the two calculations. Figure 4.16 shows the nodal

Initial Pressure = 15.65 MPa
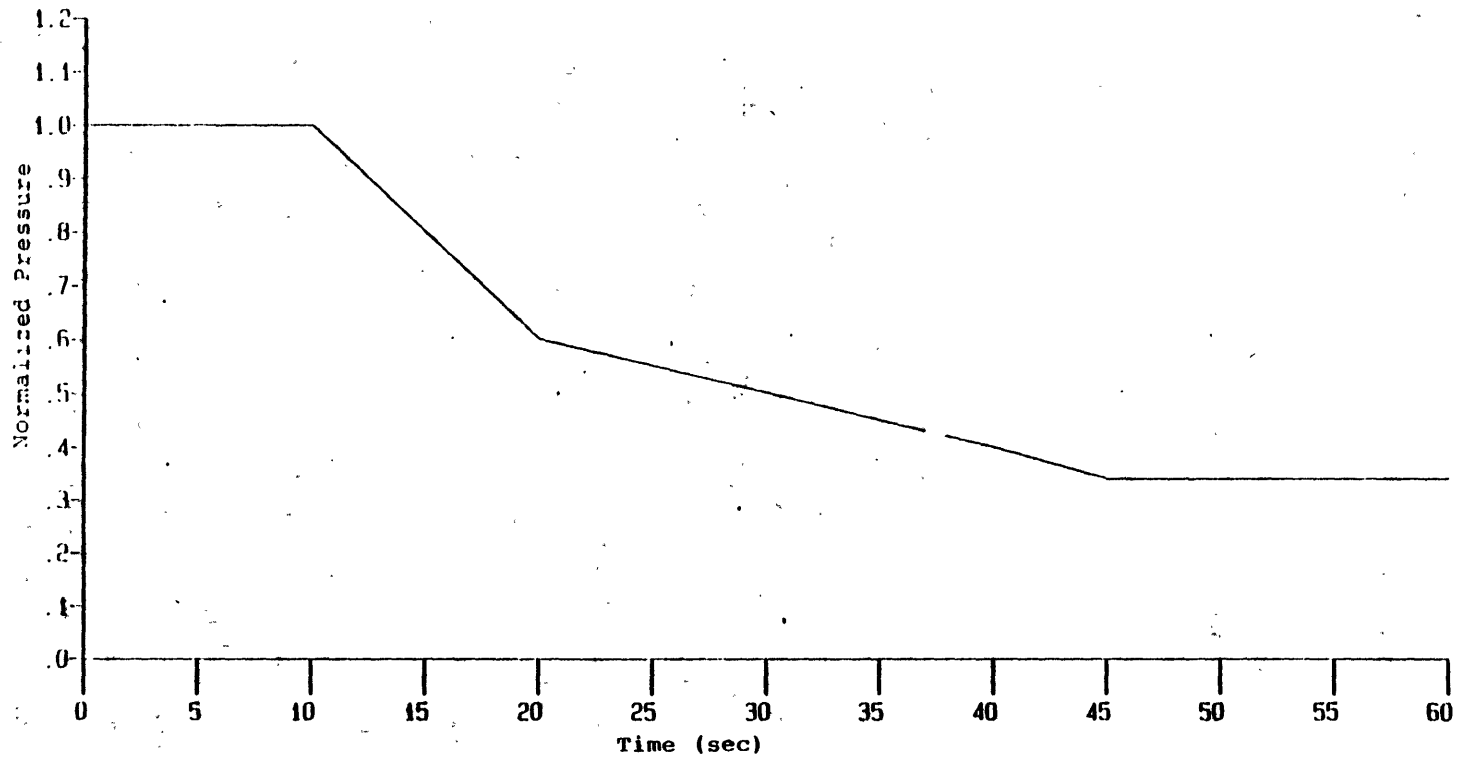
Figure 4.10   Pressure forcing function of YAT2X.
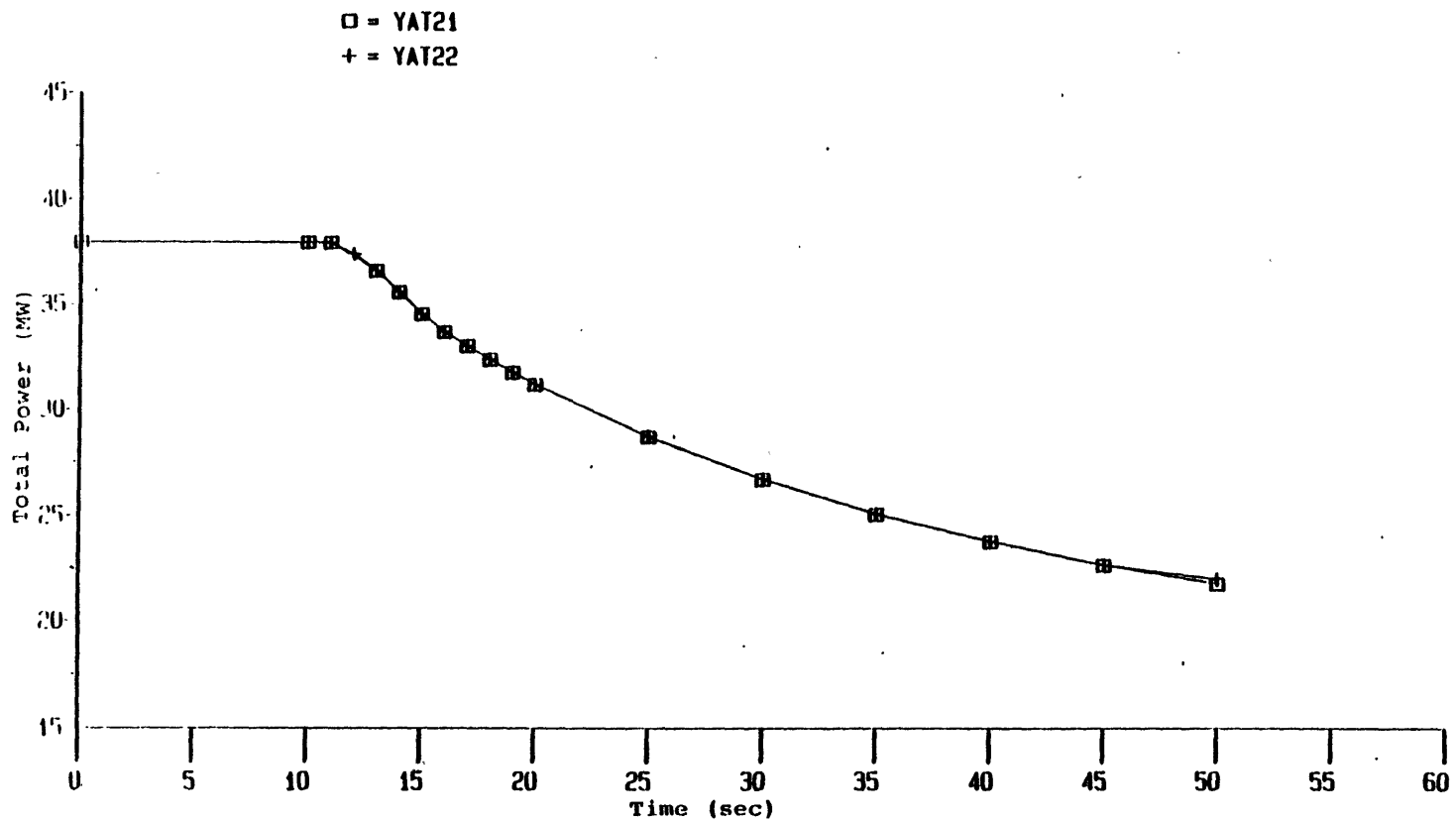
Figure 4.11   Total power histories of YAT21 and YAT22.

73
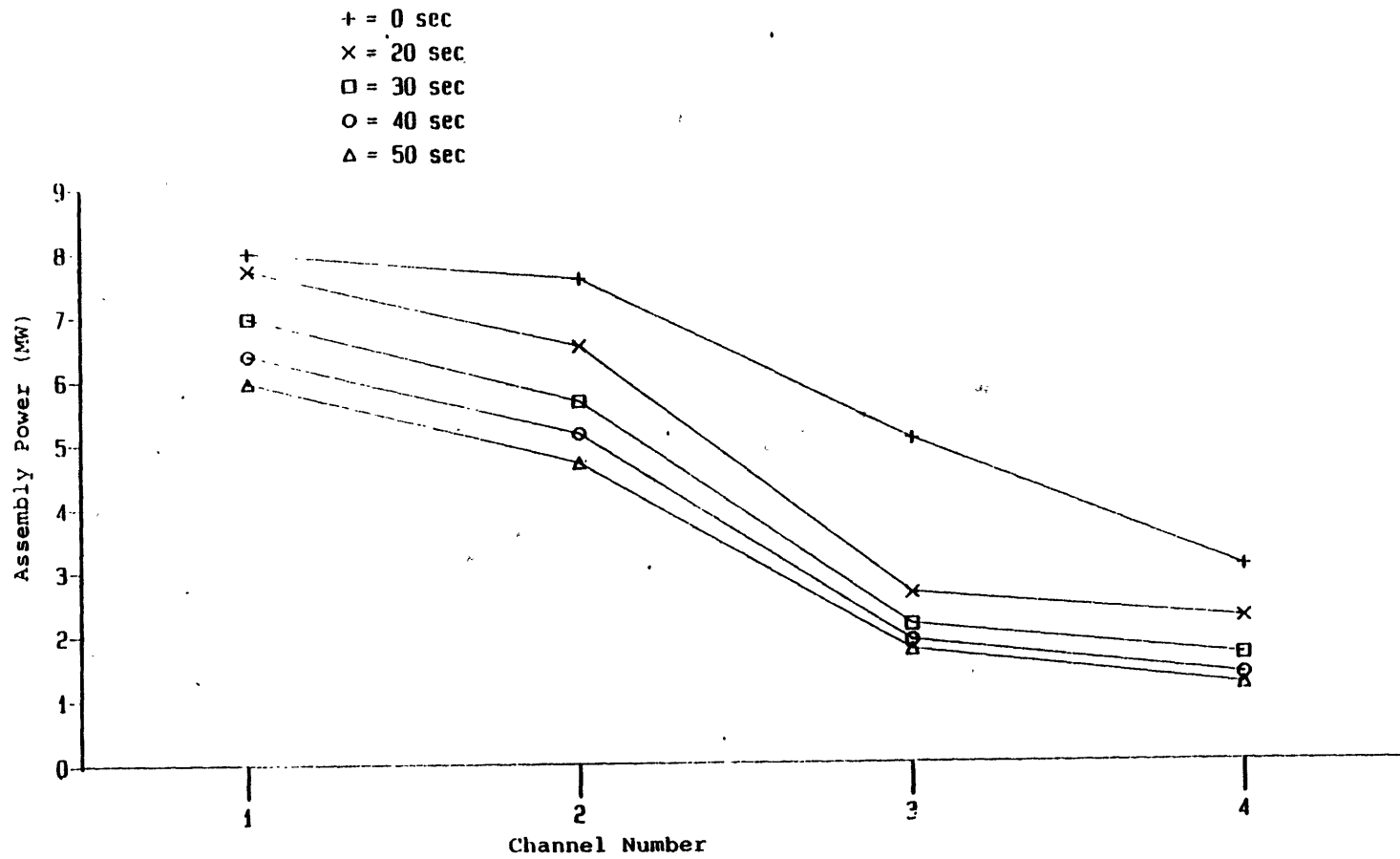
Figure4 .12   Radial assembly power distributions in hot region, YAT21.

+ = 0 sec
× = 20 sec
□ = 30 sec
○ = 40 sec
△ = 50 sec
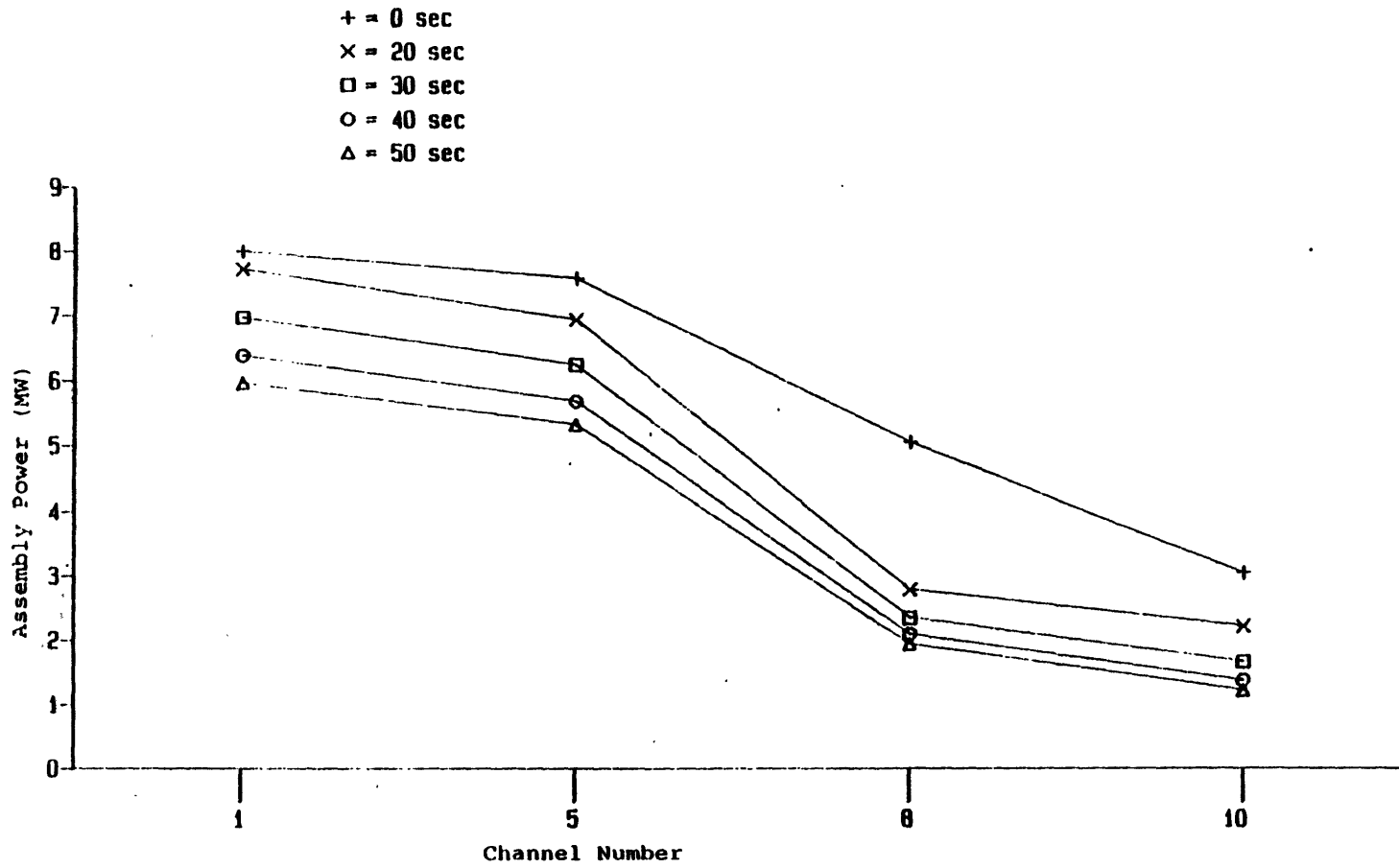
Figure 4.13   Radial assembly power distributions in cold region, YAT21.
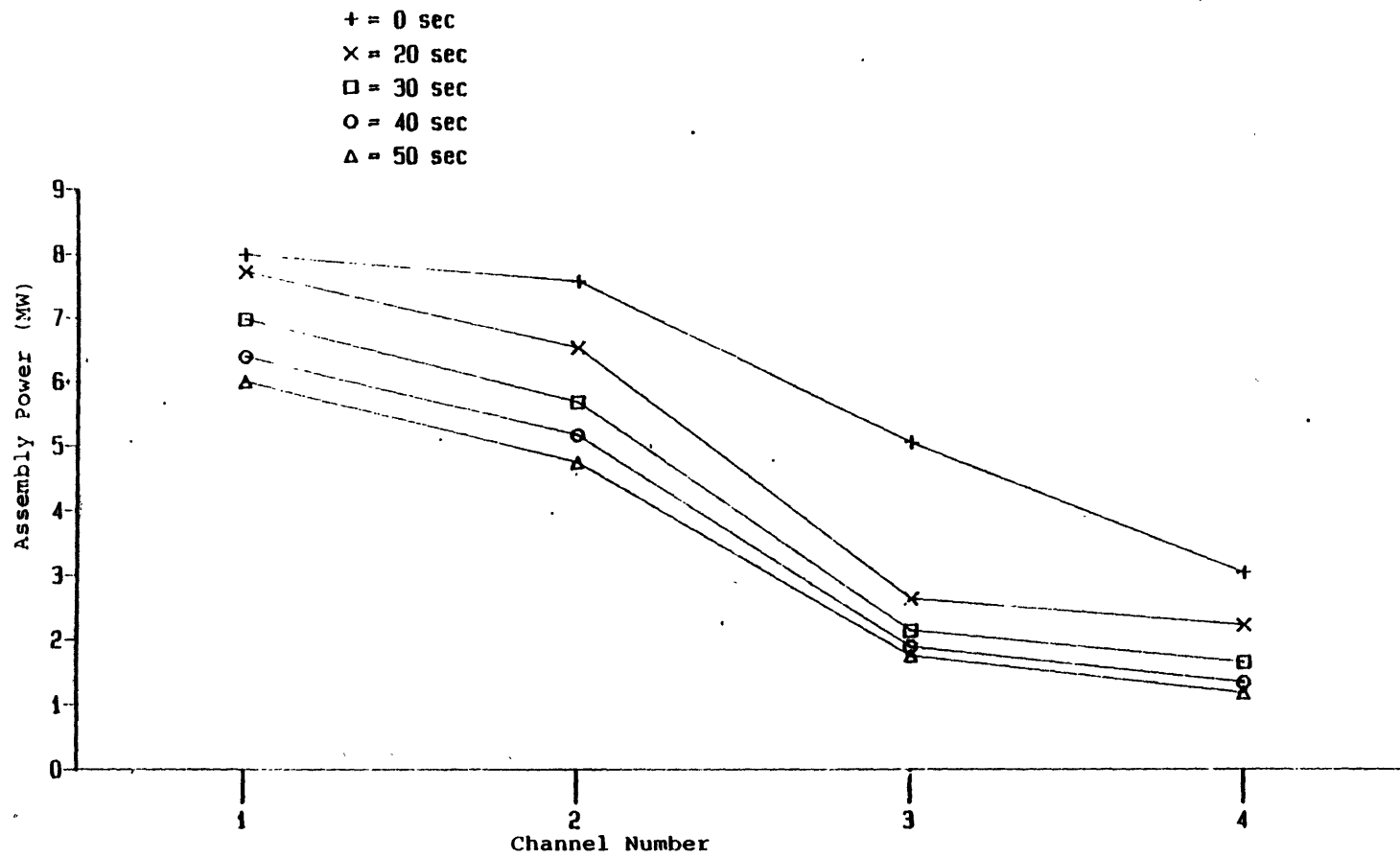
Figure 4.14 Radial assembly power distributions in hot region, YAT22.
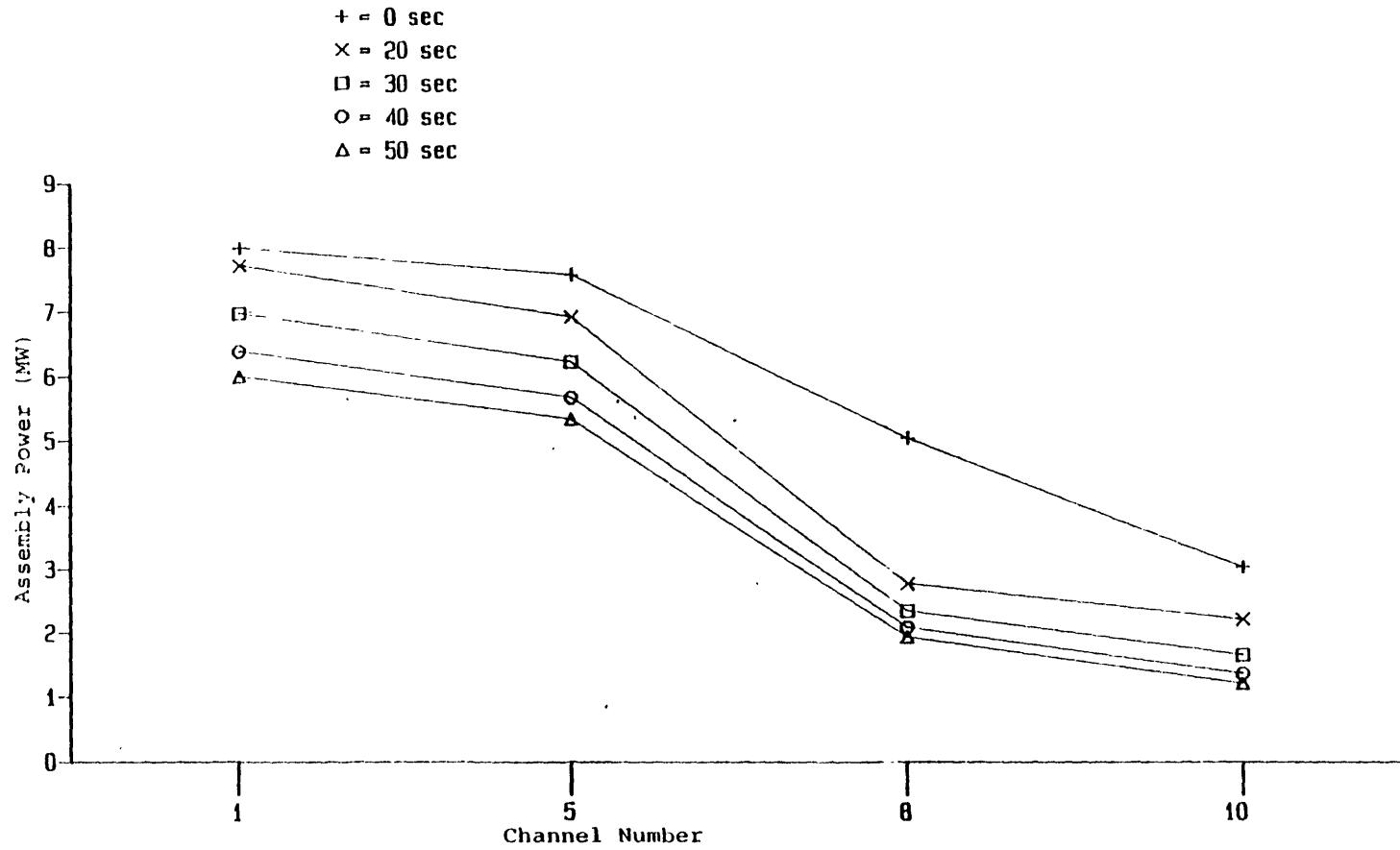
Figure 4.15   Radial assembly power distributions in cold region, YAT22.
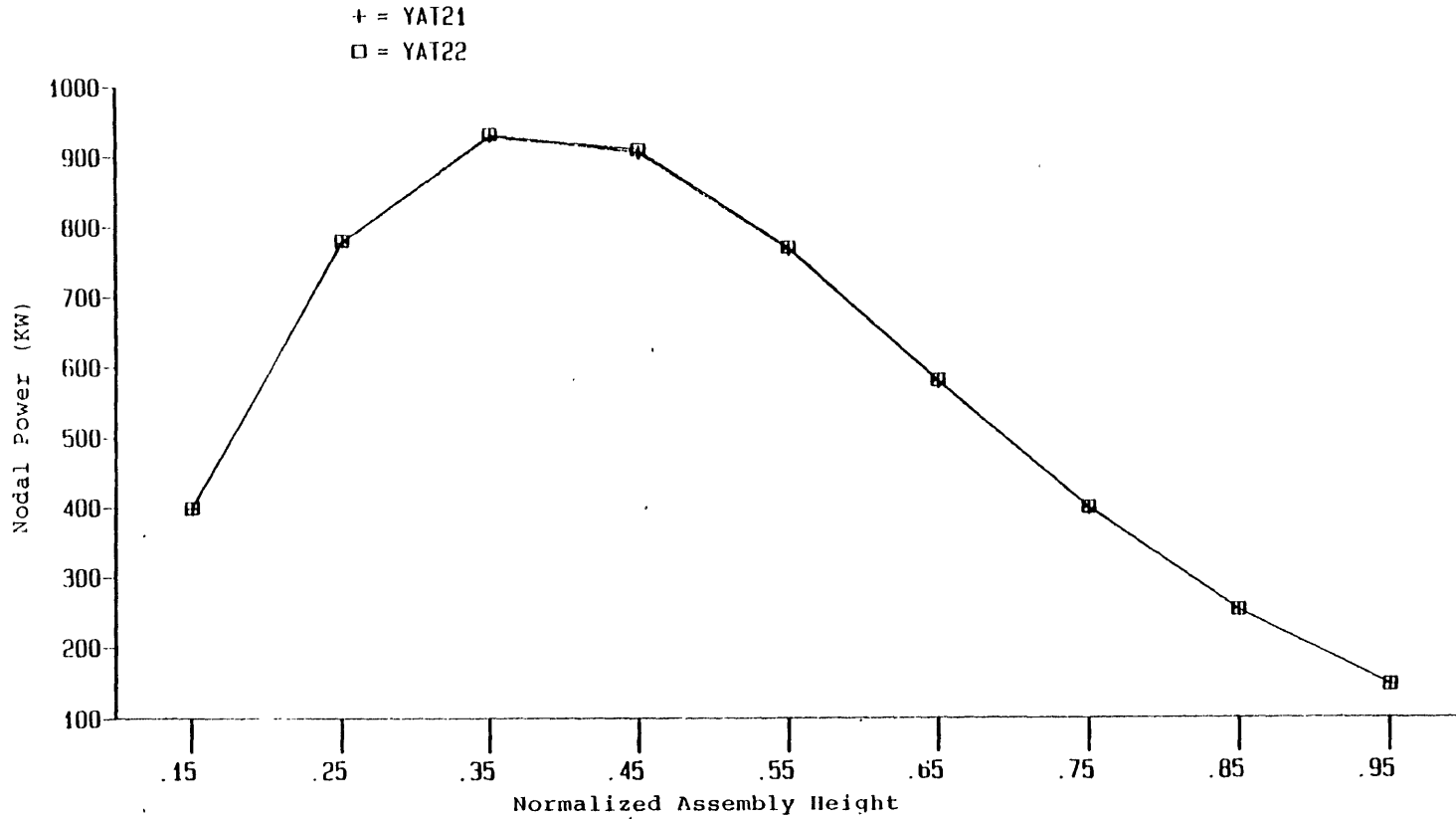
+ = YAT21

□ = YAT22

Figure 4.16   Axial nodal power distributions of channel at 40 sec,
              YAT21 and YAT22.
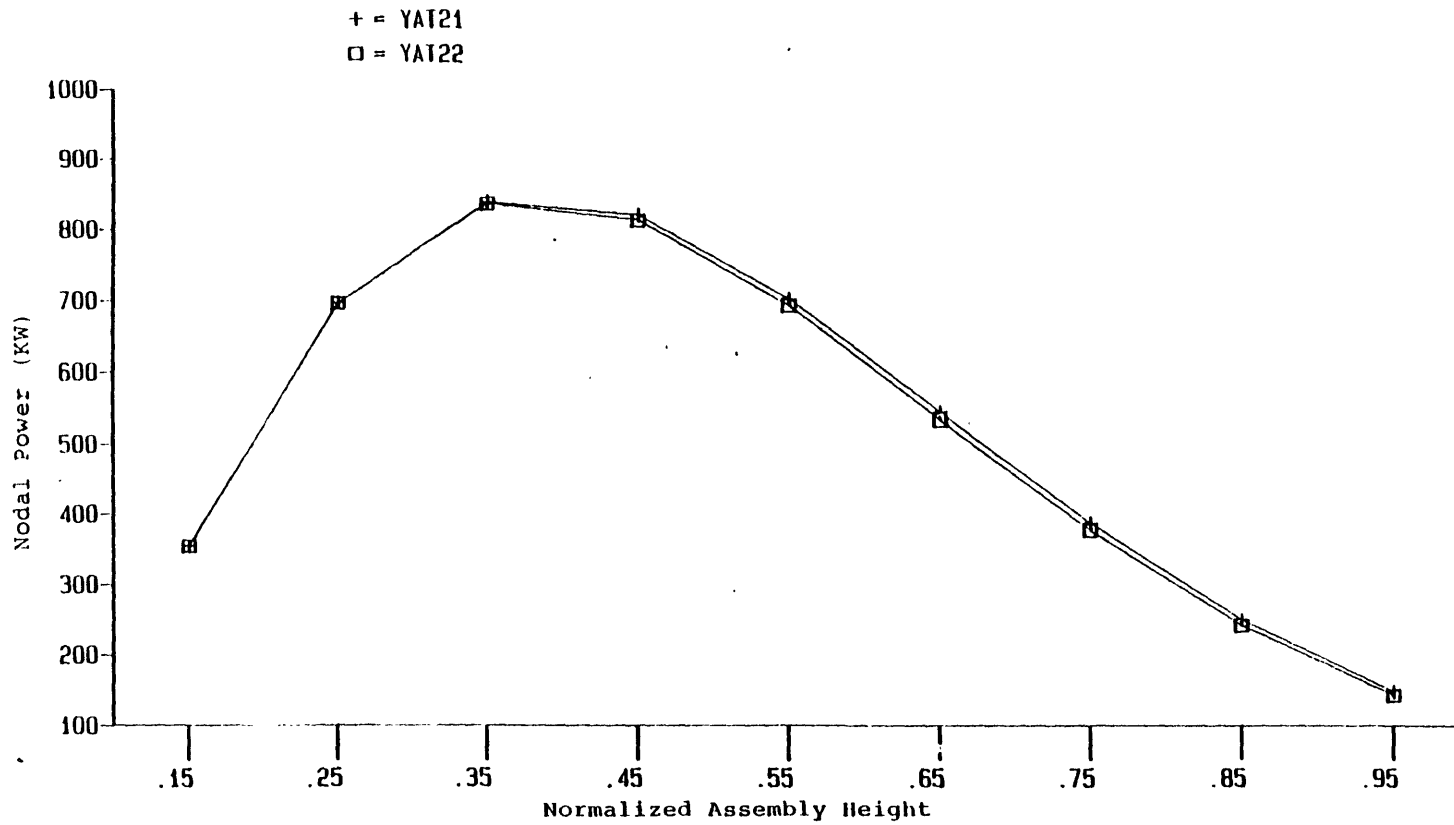
Figure 4.17 Axial nodal power distibutions of channel 2 at 50 sec, YAT21 and YAT22.

powers at 40 seconds which is just before boiling occurs. Clearly the results are very close and verify the conclusion that no significant difference between closed- and open-channel calculations before boiling starts for a steam line break accident. Figure 4.17 shows the nodal powers at the end of the calculations (at 50 seconds). It is clear that the differences between the two distributions are small but larger than those shown in Figure 4.16. At this point, one temporary conclusion is made that after boiling occurs, even though the global parameters (such as total power, assembly powers) do not show significant difference between results of closed- and open-channel calculations, the local parameters (such as the nodal powers) may reveal the difference. To support the conclusion, the axial void fraction distributions of channel 2 of the two calculations are shown in Figure 4.18. Apparently the cross flow allowed by the open-channel calculation (YAT22) provides a much better mixing of the coolants among channels. The void fraction distribution of the open-channel calculation (YAT22) is much lower than that of YAT21 (closed-channel calculation). In addition, the vapor voids fromed at a higher position of YAT22 compared with that of YAT21 (0.45 to 0.35 normalized height of the channel).

In regard to the MDNBR, Figure 4.19 shows the histories of the two calculations. As mentioned before, the boiling starts eariler for YAT21 than YAT22. This correlates with the MDNBR of YAT21 droping from 99 to a low value (around 5) eariler than the MDNBR of YAT22. Again, this is because of the cross flow allowed in case YAT22. Both MDNBR values keep decreasing up to 45 seconds. Recall that the depressurization (see Figure 4.10) stops at 45 seconds and the pressure remains constant afterwards. Therefore, after 45 seconds the void is condensing as the colder and colder coolant keeps coming in from the core inlet. Figure 4.20 shows node (2,5) void fraction histories from the two calculations. It is clear that the void generation stops and condensation begins at 45 seconds. This is also the reason that the MDNBR of both calculations increasing instead of decreasing after 45 seconds.

During the void generation process (before 45 seconds), the open-channel case allows the fluids to be driven out of the node and causes lower MDNBR predictions than the closed-channel case. The liquid cross flow of node (2,5) (where MDNBR occurs) at 40 seconds (before boiling starts) is −0.66 Kg/sec while at 44 seconds (after boiling starts) is −2.76 Kg/sec (negative sign means the liquid is leaving the node). During the void condensation process (after 45 seconds), the MDNBR of YAT22 is higher than that of YAT21. This is again due to the cross flow allowed in YAT22. The open channel allows cold coolant across channels and helps condensing voids, hence driving the MDNBR away from the limitation. The closed-channel case does not have this advantage and leads to conservative MDNBR results. The liquid cross flow at 50 seconds of node (2,5) for YAT22 is −0.39842 Kg/sec which is even lower than that at 40 seconds (−0.66158 Kg/sec).
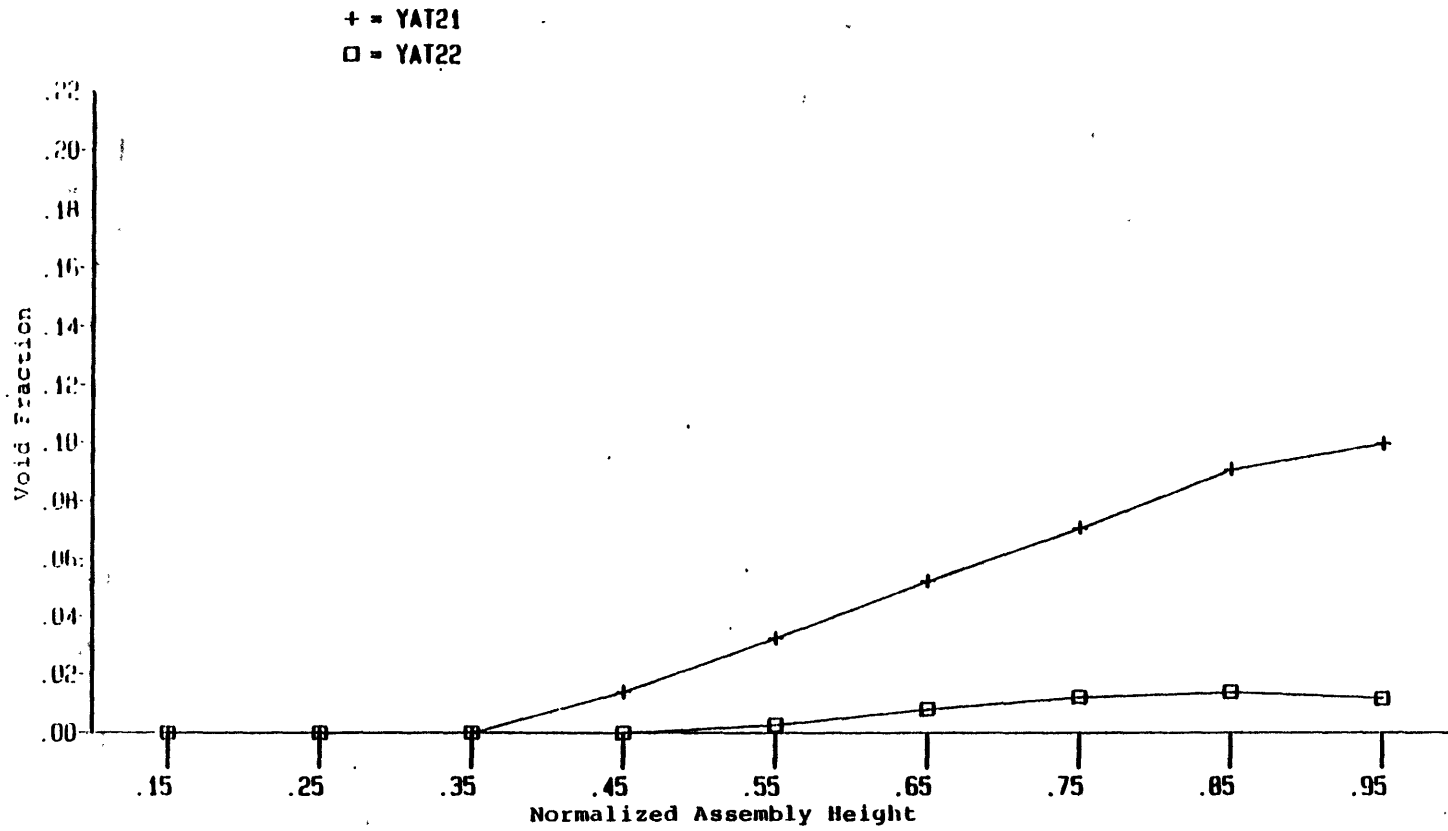
Figure 4.18   Axial void fraction distributions of channel 2 at 50 sec.
YAT21 and YAT22.
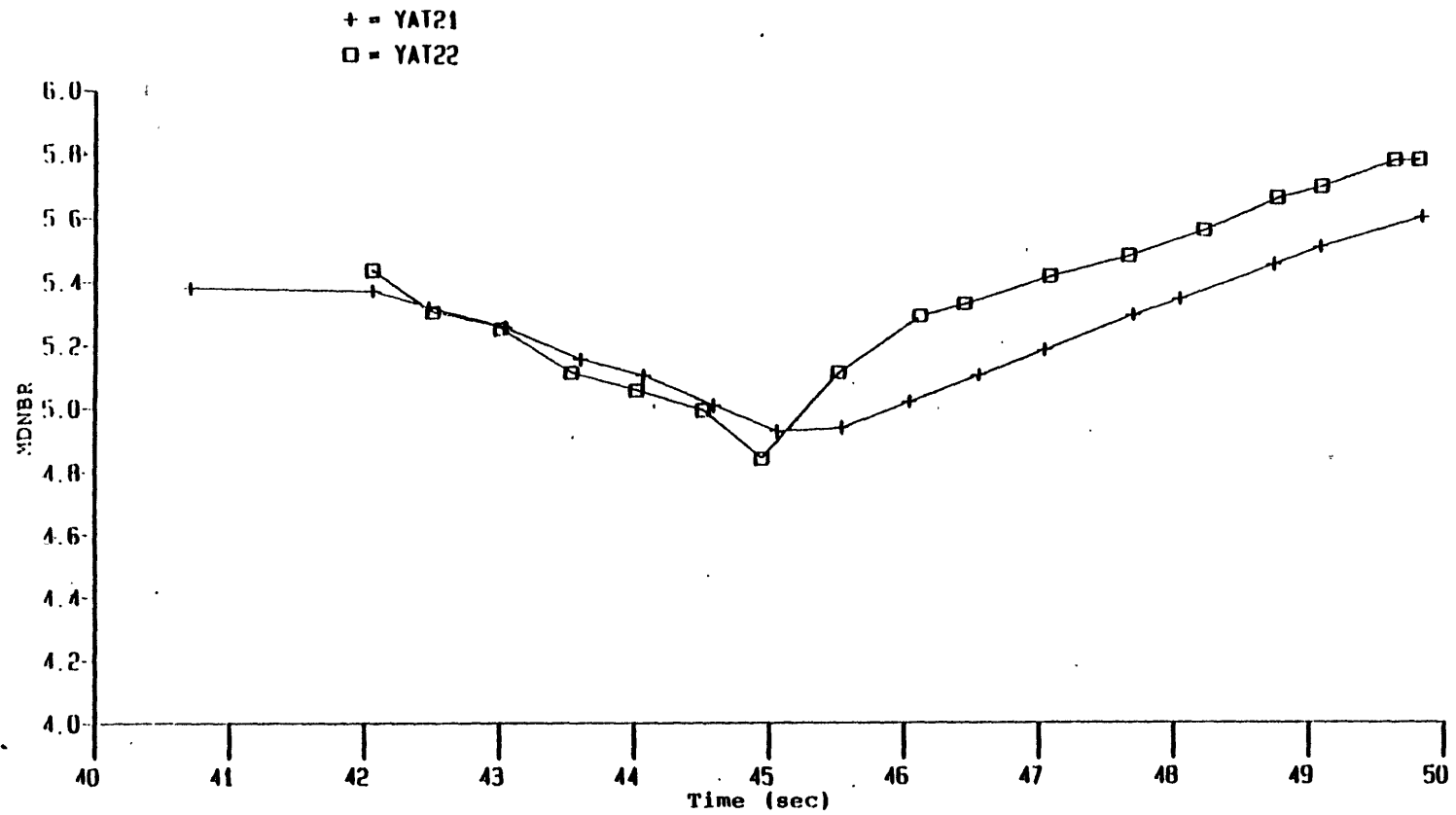
Figure 4 .19   MDNBR histories of YAT21 and YAT22.
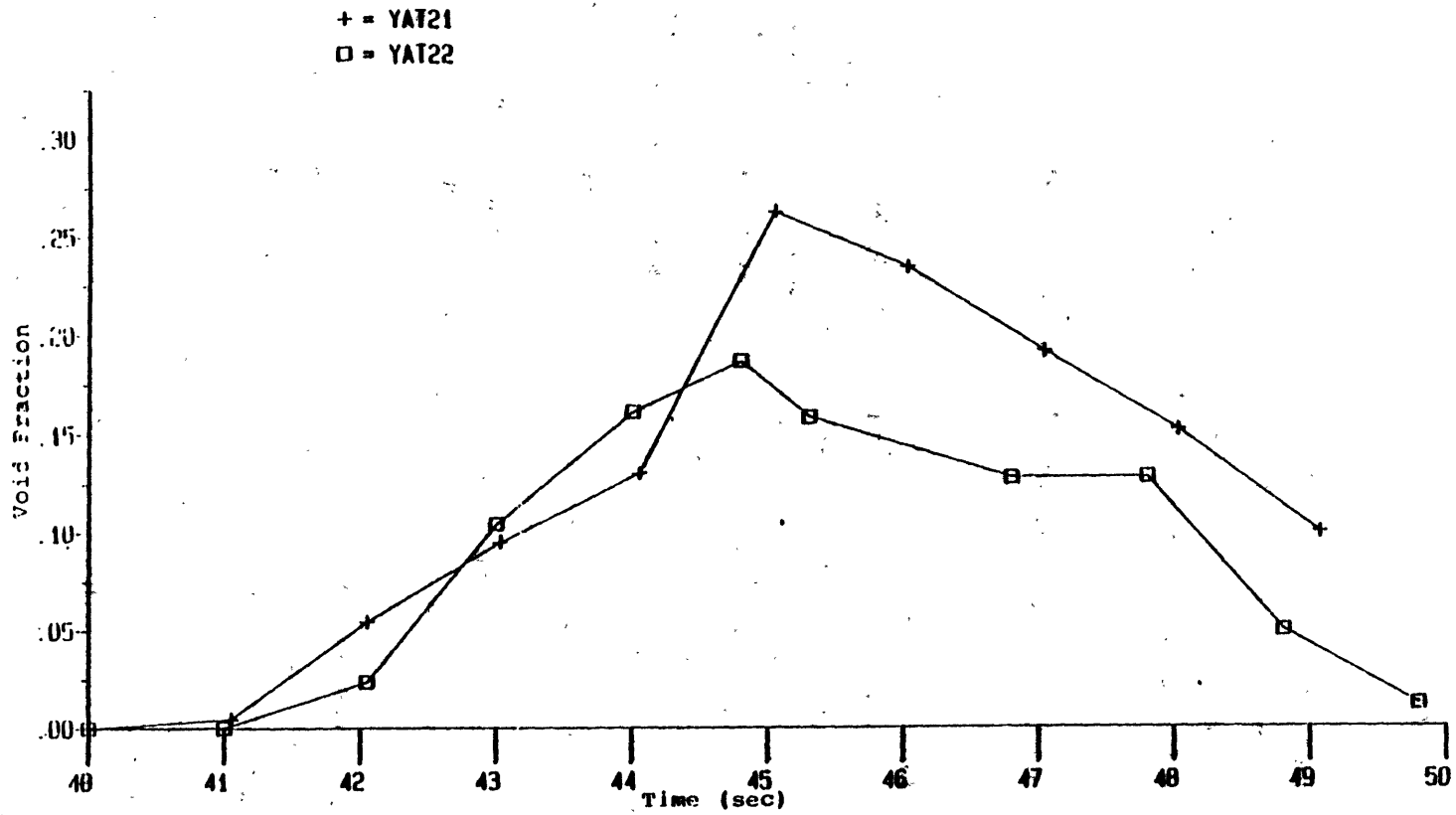
Figure 4.20 Void fraction histories of node (2,10), YAT21 and YAT22.

One remark should be made here: almost all the boiling phenomena mentioned above are subcooled boiling except the last two nodes of channel 2 which reach the saturation temperature at about 35 seconds. All the voids are found in the hot and mixing regions. The bulk coolant temperature is still lower than the saturation tempetrature for almost all the nodes.

The computational time of YAT21 and YAT22 are 10239.22 and 16097.39 cpu seconds in a Honeywell 6180 computer, respectively. Although much more effort is required by the open-channel calculation, it is necessary to do so to obtain detailed local information which can not be obtained from a closed-channel calculation.

4.4 Simulations with Reduced Pressure Forcing Function YAT3X

The pressure forcing function used in cases YAT2X is not realistic for a steam line break accident simulation, since the depressurization will not stop at 5.5 MPa. In order to study three-dimensional effects of a steam line break accident in more detail, the calculations, YAT3X, employs the same pressure forcing function used in YAT2X calculations except the pressure keeps decreasing after 45 seconds as shown in Figure 4.21. All the other three forcing transients: the coolant mass flow rate transient, the inlet coolant temperature transient, and the scram action, are kept the same as those used for cases YAT11 and YAT2X.

Figures 4.22 to 4.25 show comparisons between the results of YAT31 (closed-channel) and YAT32 (open-cahnnel) calculations from the global parameters (total powers, assembly powers) to the local parameters (nodal powers). It can be seen that the total power histories of the two calculations still match each other till the end of the calculations. However, the assembly powers and the nodal powers show relatively more pronounced differences between the two calculations, compared with the results of YAT2X (refer to Figures 4.12 to 4.15, and 4.17). This points out the importance and necessity of a three-dimensional analysis for a steam line break accident to obtain detailed local information when boiling is significant.

The axial void fraction distributions of channel 2 at 50 seconds are shown in Figure 4.26. The open-channel results (YAT32) have lower void fractions than those of the closed-channel (YAT31). The cross flow contributes to a better mixing among the channels leading to such results. The void fraction histories of node (2,10), the exit node of the hottest assembly, of the two calculations are shown in Figure 4.27. The continuous increase of the void fraction in both cases are expected since the pressure is decreasing throughout the calculations.

Although the pressure after 45 seconds is out of the data base of W-3 CHF correlation,
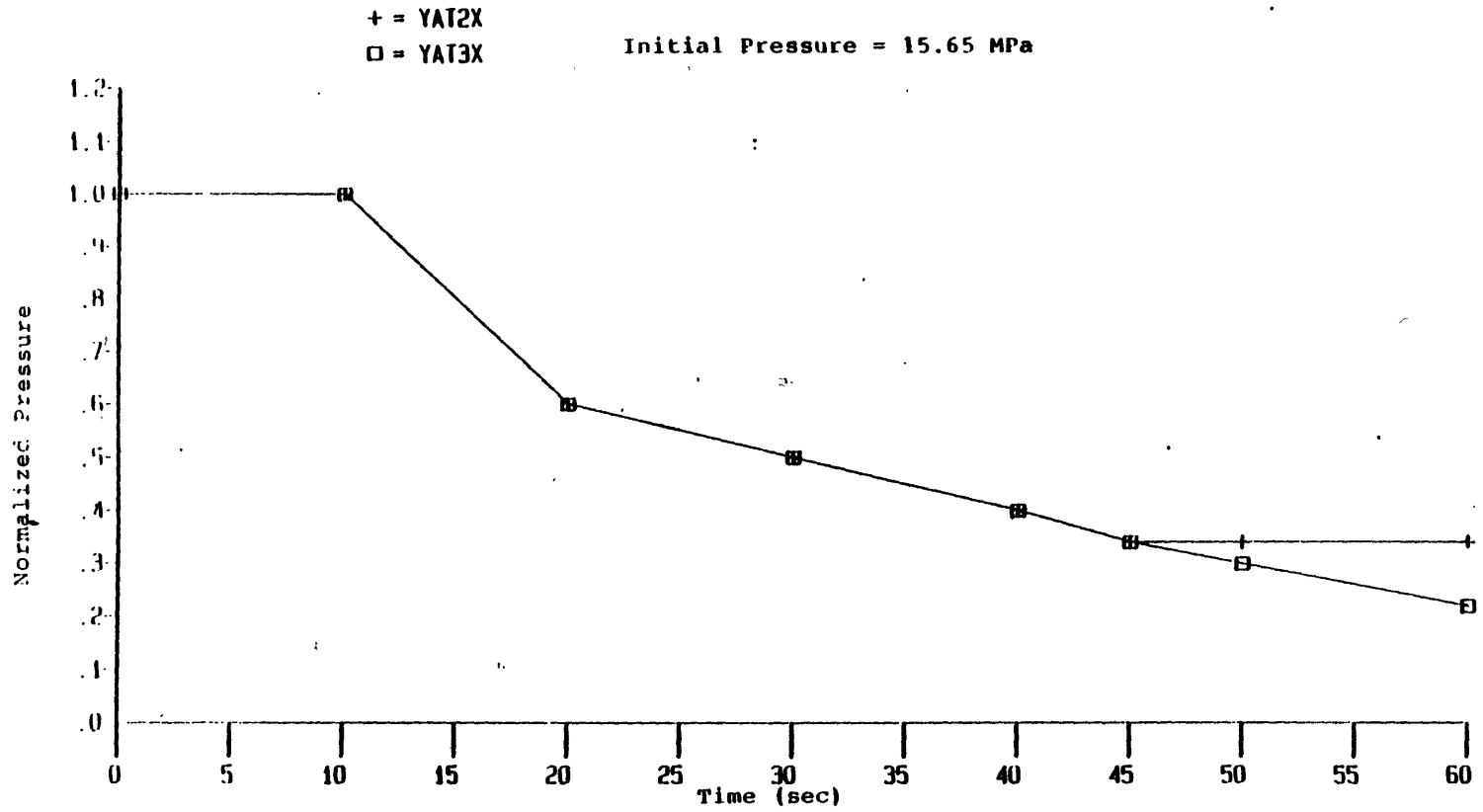
Figure 4.21   Pressure forcing functions of YAT2X and YAT3X.
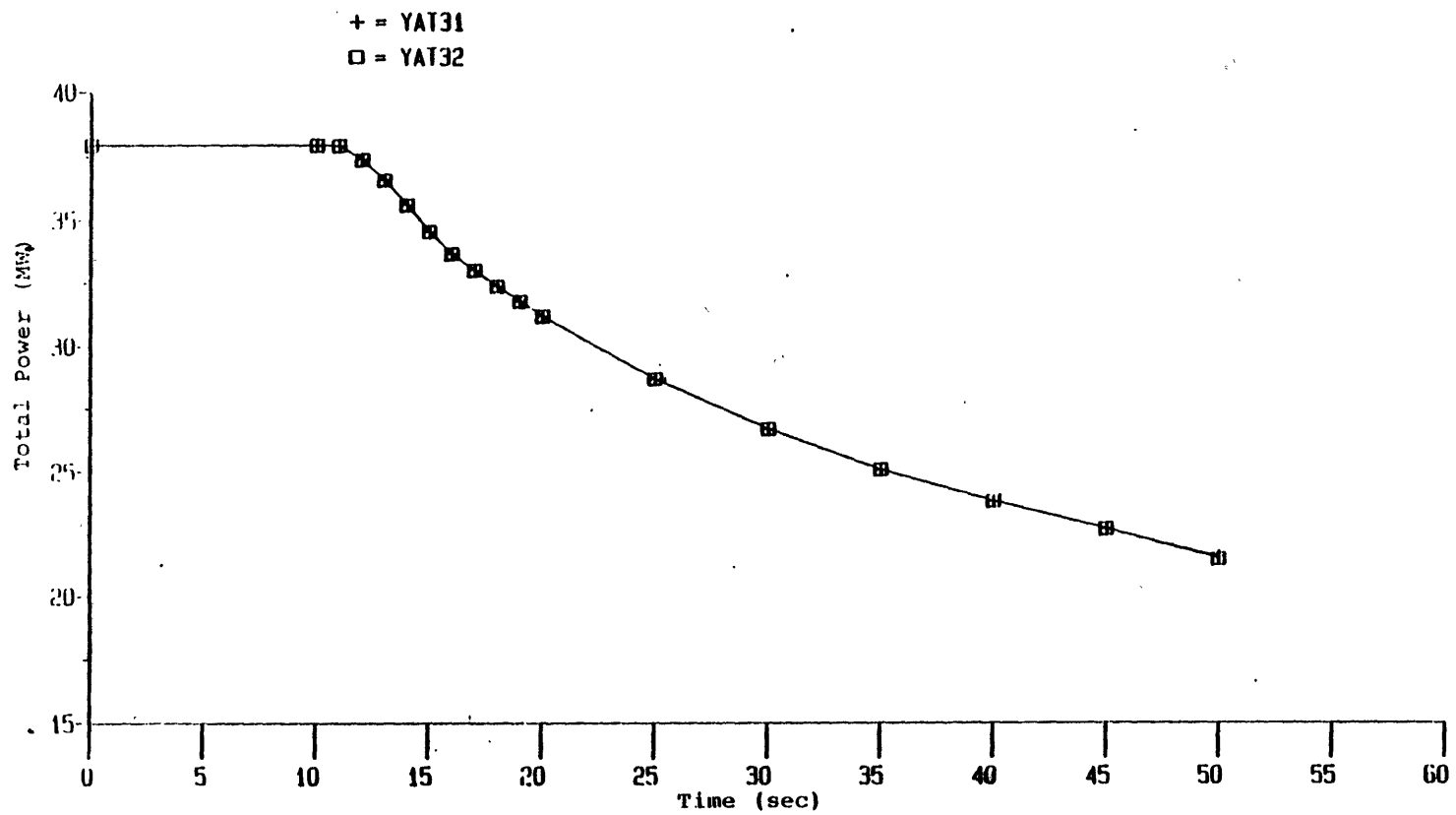
Figure 4.22 Total power histories of YAT31 and YAT32.
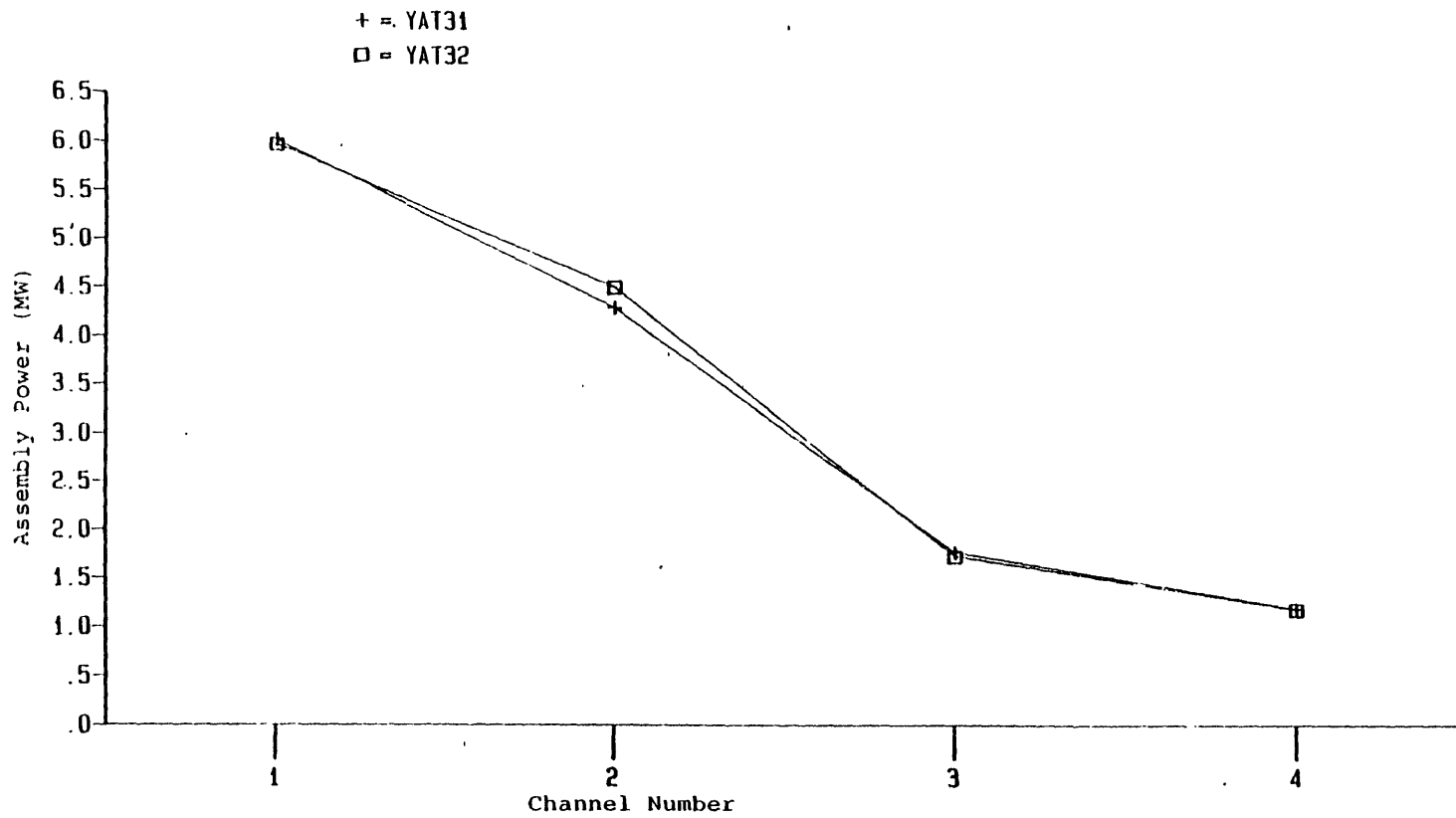
+ =. YAT31

□ = YAT32

Figure 4.23   Radial assembly power distributions in hot region, at 50 sec,
YAT31 and YAT32.

Figure 4.24   Radial assembly power distributions in cold region, at 50 sec
             YAT31 and YAT32.
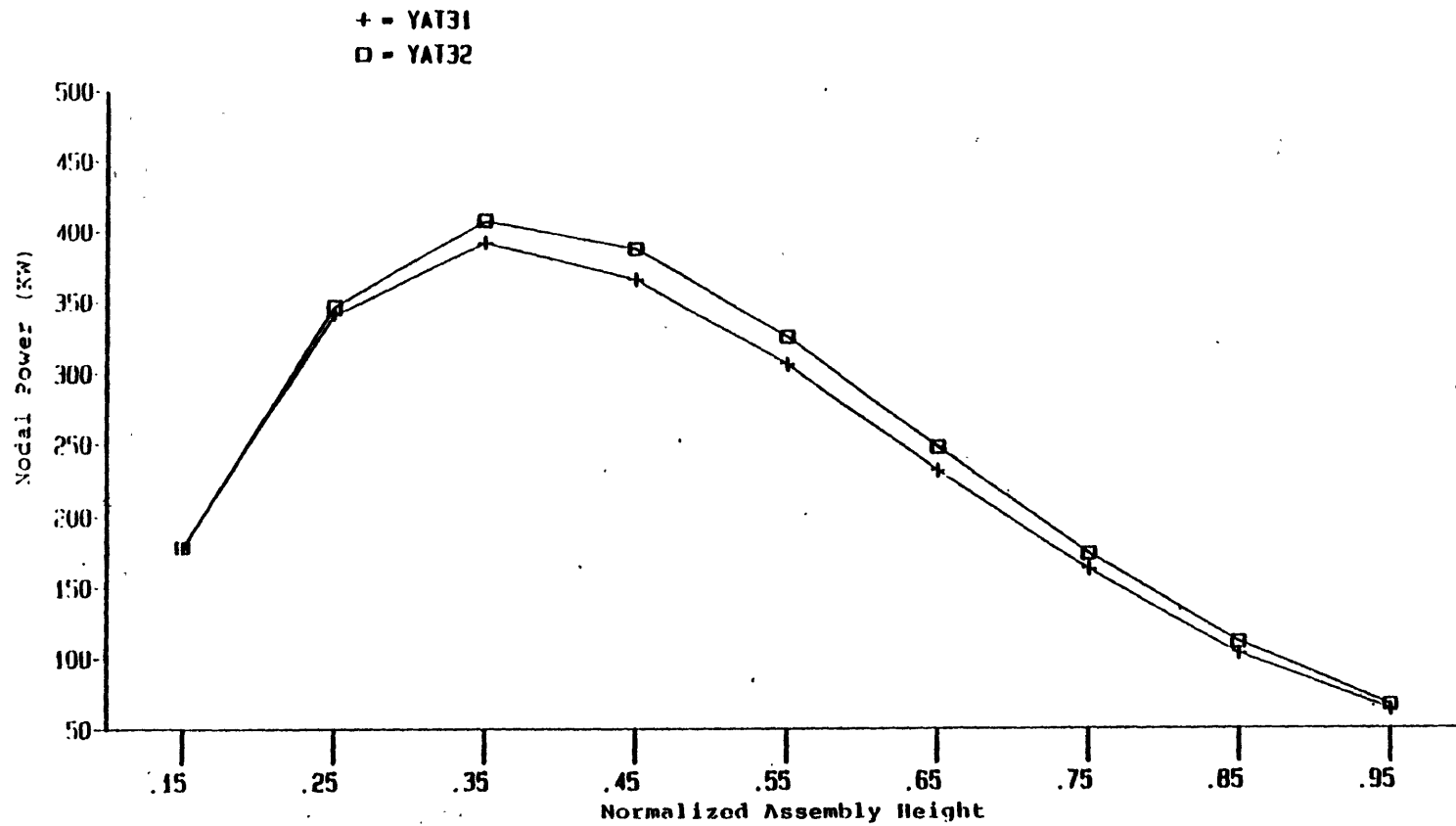
Figure 4.25 Axial nodal power distributions of channel 2 at 50 sec.
          YAT31 and YAT32.
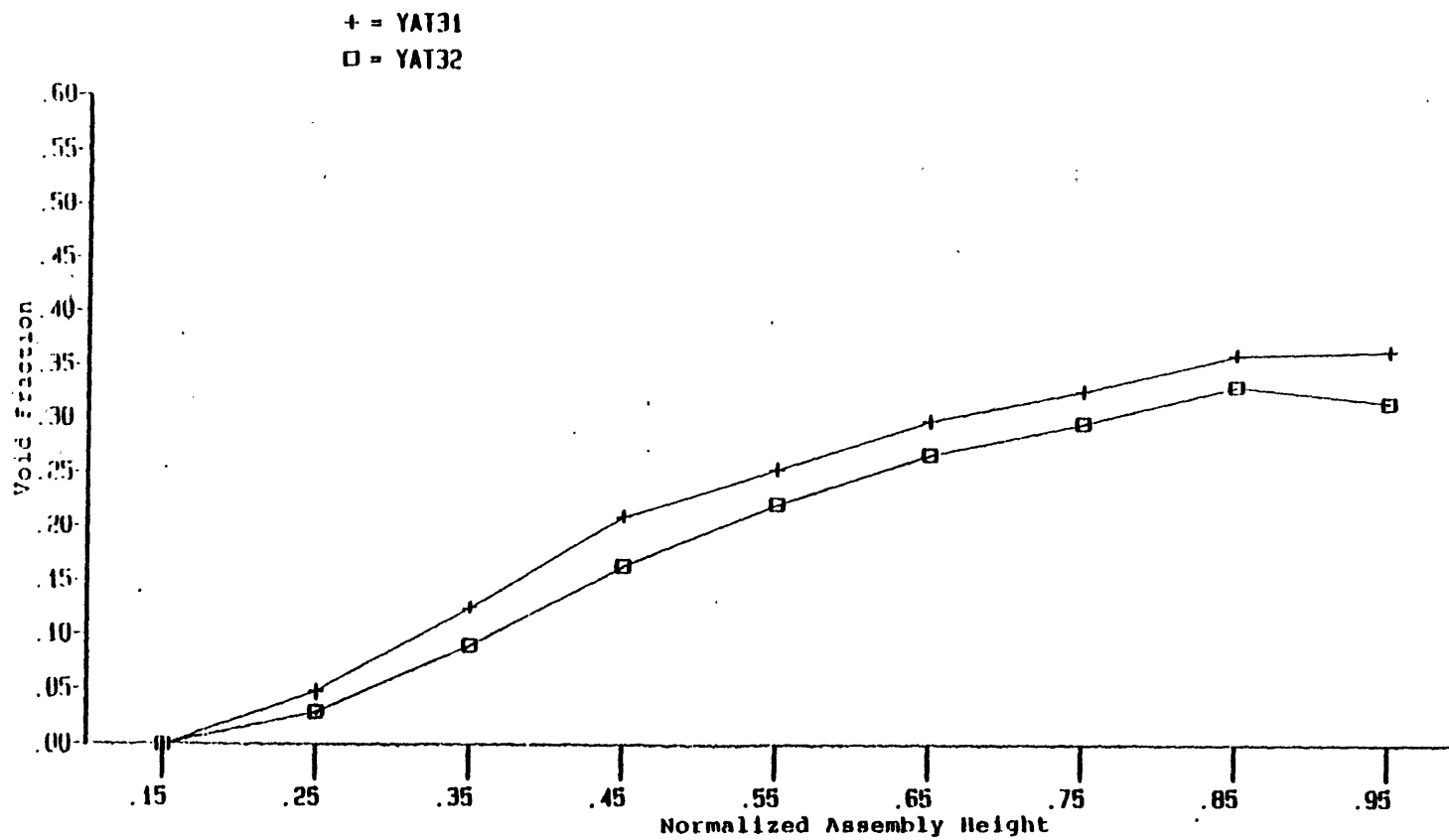
Figure 4.26   Axial void fraction distributions of channel 2 at 50 sec.
YAT31 and YAT32.
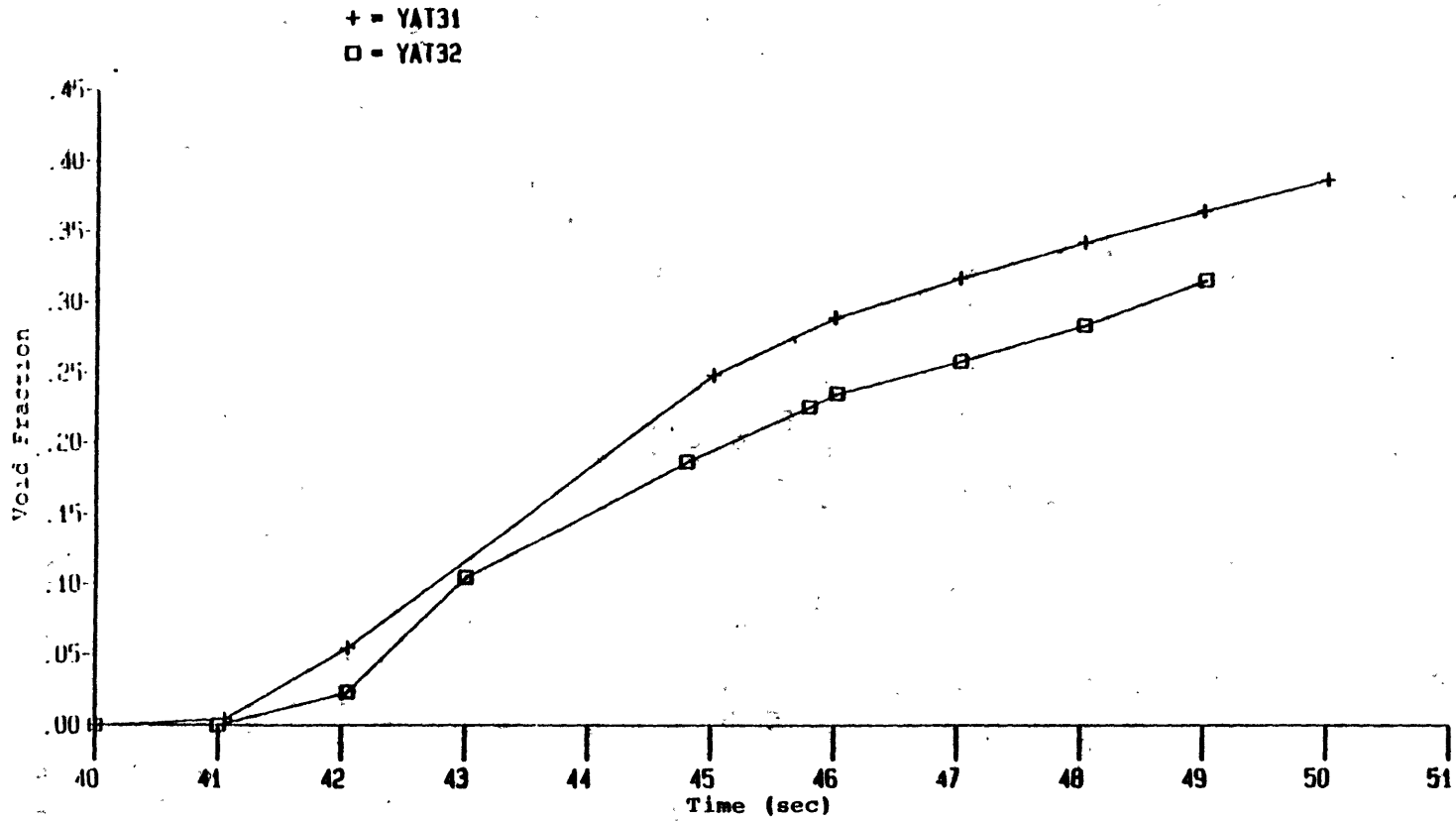
Figure 4.27 Void fraction histories of node (2,10), YAT31 and YAT32.

the MDNBR predicted by the correlation, shown in Figure 4.28, seems to be reasonable. The MDNBR's of both calculations keep decreasing after 45 seconds and the difference between them is getting larger and larger as expected. The open-channel calculation is conservative with regard to MDNBR calculation. The liquid corss flow of node (2,5) at 50 seconds is -5.8549 kg/sec which is about 8.85 times larger than the liquid corss flow at 40 seconds (-0.66158 Kg/sec). The significant amount of the corss flow does affect the local behavior during the transients.

The boiling after 45 seconds is saturation boiling for the last few nodes of the hot and mixing region channels. If depressurization keeps going on, even though the voids result in a negative reactivity addition to the core, a hot spot may form and fuel damage may occur. Actually, the depressurization and the inlet coolant temperature forcing transients should be coupled together. The former adds negative reactivity to the core while the latter adds positive reactivity to the core. The net effect of the two transients, after the scram action is completed, depend on the feedback coefficients. For example, if the coolant temperature feedback coefficient is very high while the density feedback coefficient is low, the net effect may result in return to power. If it is the reverse, once boiling starts the power will have no chance to return back hence no return to power is expected. However, the possibility of fuel surface blanketing by vapor should be carefully considered in such situation. The Doppler feedback phenomenon is a 'self-balance' process. The decreasing fuel rod temperature adds positive reactivity to the core because of the Doppler effect. Therefore, the fuel rod will generate a higher power. Higher power results in a higher fuel rod temperature and hence adds negative reactivity to the core by the same Doppler effect. It is therefore expected that the power transient of a steam line break is mainly dominated by the coolant temperature and coolant density feedback coefficients. Of course, if boiling is not significant, the coolant density feedback effect will not be important for PWR's.

The computational for YAT31 and YAT32 are 11184.59 and 21076.48 cpu seconds in a Honeywell 6180 conputer, respectively. Performing an open-channel (three-dimensional) analysis for a steam line break accident may be needed to have detailed local description based on the discussion above even though much more computational effort is required. Of course, if one is only interested in the total power history of a steam line break accident, the closed-channel analysis will be sufficient and nearly half of the computational effort can be saved.

In addition to the above two simulations, it is of interest to study a uniform inlet coolant temperature distribution case (YAT34). The arrangement leads to a real one-dimensional, multi-channel calculation. All the forcing transients are the same as case YAT32 except that only one inlet coolant temperature forcing function is used for all the ten channels. The
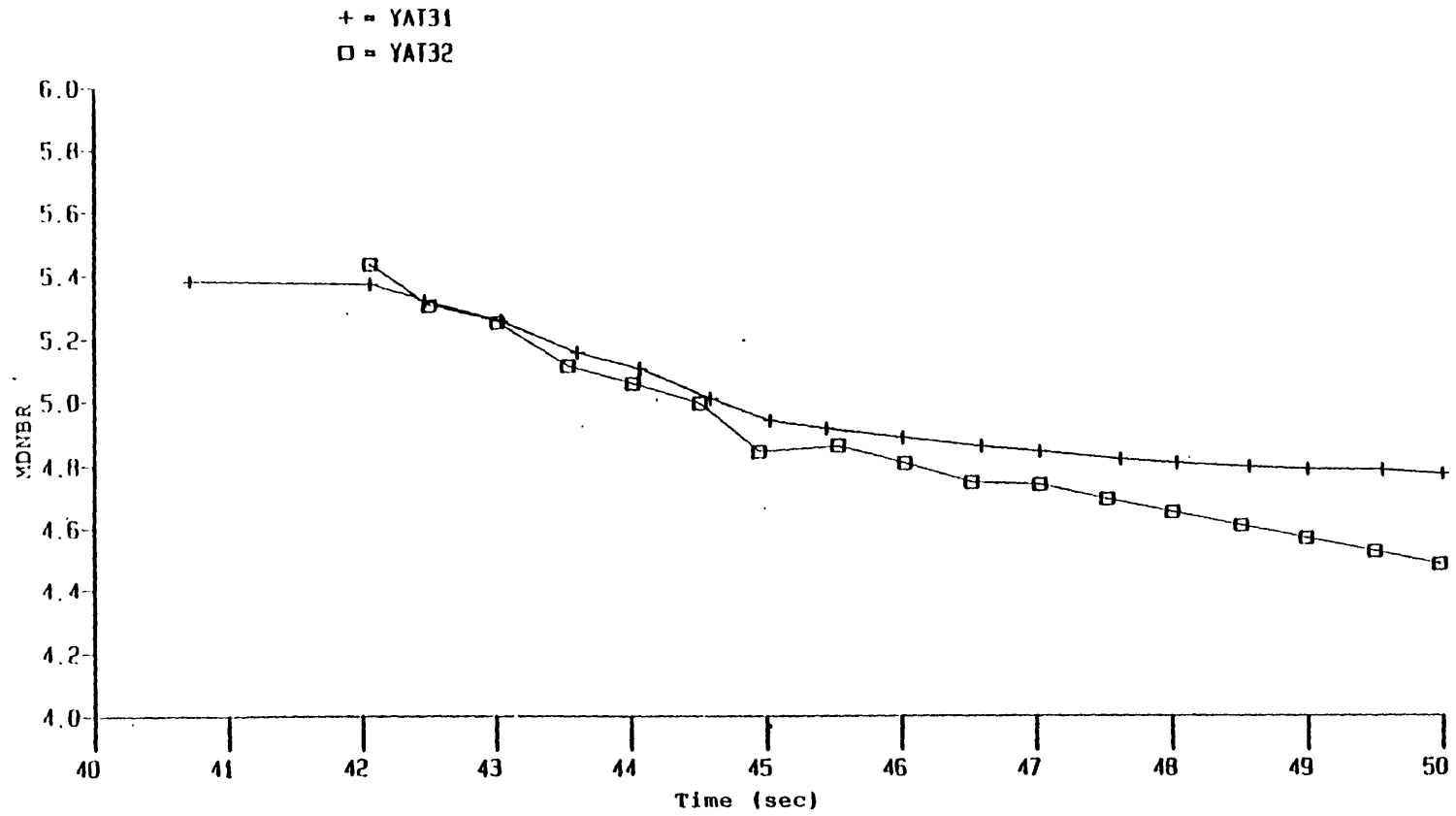
Figure 4.28   MDNBR histories of YAT31 and YAT32.

93

forcing function for the inlet temperature is the mixing region temperature function (refer to Figure 4.2) which is the average of the hot and cold region functions.

Some difference is observed after around 45 seconds in the total power histories between cases YAT32 and YAT34 as shown in Figure 4.29. The assembly powers and the nodal powers, shown in Figures 4.30 to 4.32, reveal that powers of YAT34 are in between powers of hot and cold regions of YAT32 as expected. The MDNBR calculation by YAT34 predicts a value of 99 (i.e., the MDNBR is much greater than five) throughout the calculation as shown in Figure 4.33. This implies that a one-dimensional, multi-channel calculation using average inlet temperature condition can not provide accurate information about local conditions during a steam line break transient.

4.5 Interphase Mass Exchange Model and Time Step Size Sensitivity Studies

4.5.1 Interphase Mass Exhange Model Sensitivity Studies

Three interphase mass exchange options are available in TITAN: the Nigmatulin Model [26], the non-equilibrium model [27], and the suppressed exchange option (i.e., no interphase mass exchange is allowed). All the calculations discussed in this chapter so far use the non-equilibrium model, since subcooled boiling is expected in a steam line break accident. The vaporization and condensation rates of the Nigmatulin Model depend on the bulk temperatures of the liquid and vapor, respectively. For this reason, this is a saturated boiling model and subcooled boiling is not predicted by this model. It is of interest to know what the impact on the global and local solutions by the Nigmatulin Model will be.

The calculation YAT36 is exactly the same as the calculation YAT32 except that the Nigmatulin Model instead of the non-equilibrium model is chosen. It is found that the results (total powers, assembly powers, and nodal powers) of the two calculations agree with each other very well (Figures 4.34 to 4.37). The axial void fraction distributions of channel 2 for the two calculations reveal discrepancy between the models during the transition period from subcooled to saturated conditions (Figure 4.38). Once boiling is in a saturated regime, predictions by both models approach each other. The MDNBR histories predicted by the two models (Figure 4.39) lead to the same conclusion.

From the above discussion, it seems that without considering the transition period of subcooled to saturated conditions both models can be employed for the analysis. However, this is not always true. If the coolant temperature feedback coefficient is very large and a

94
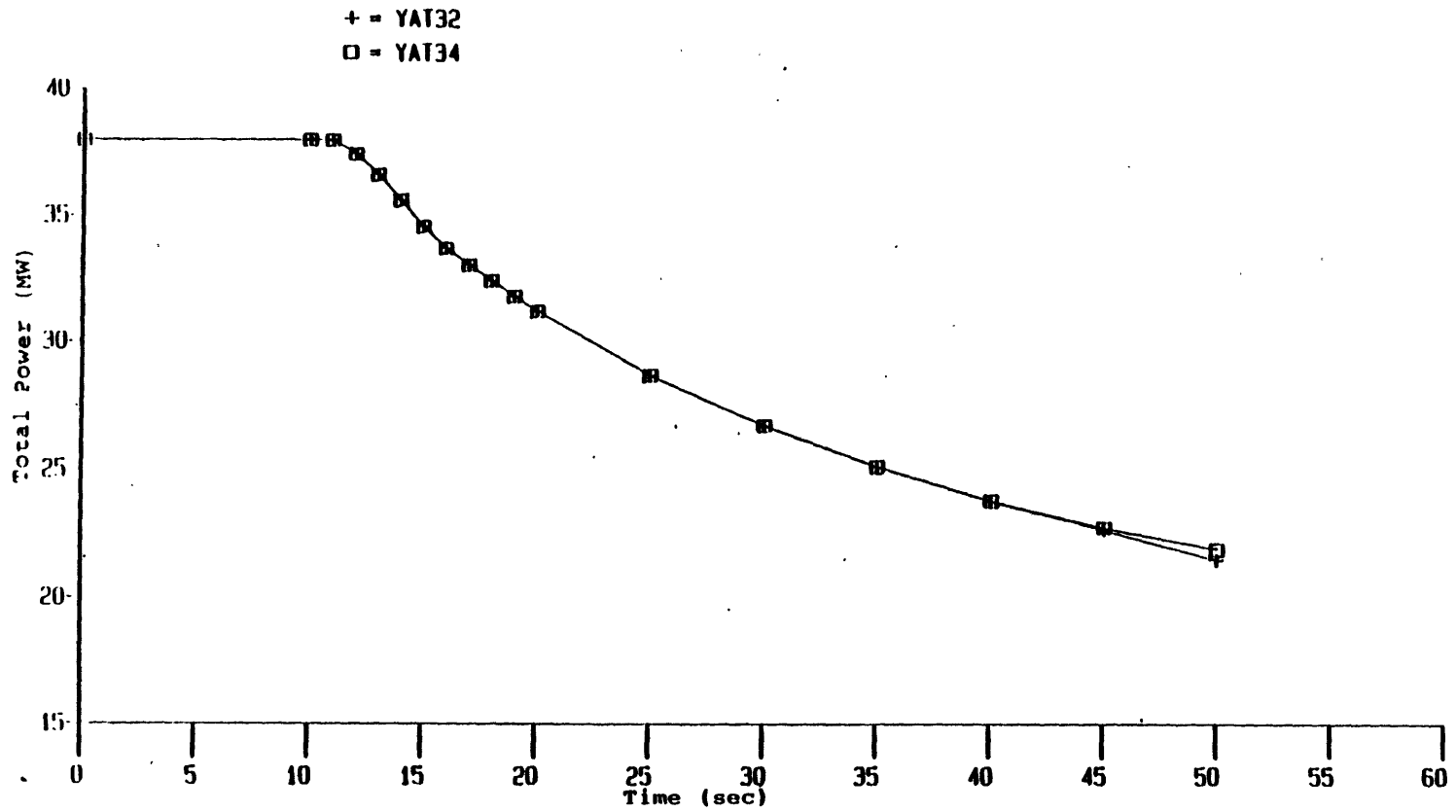
Figure 4.29   Total power histories of YAT32 and YAT34.
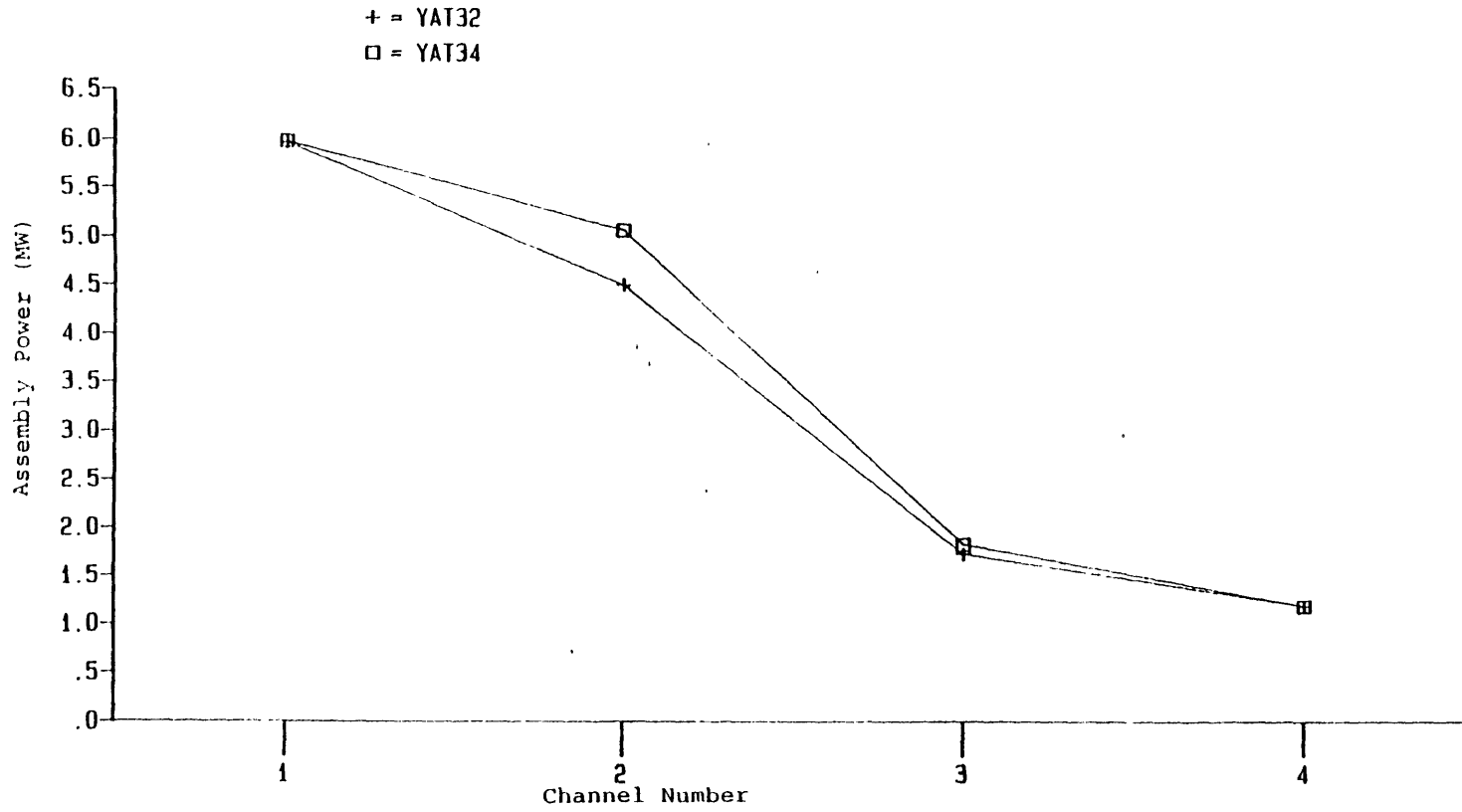
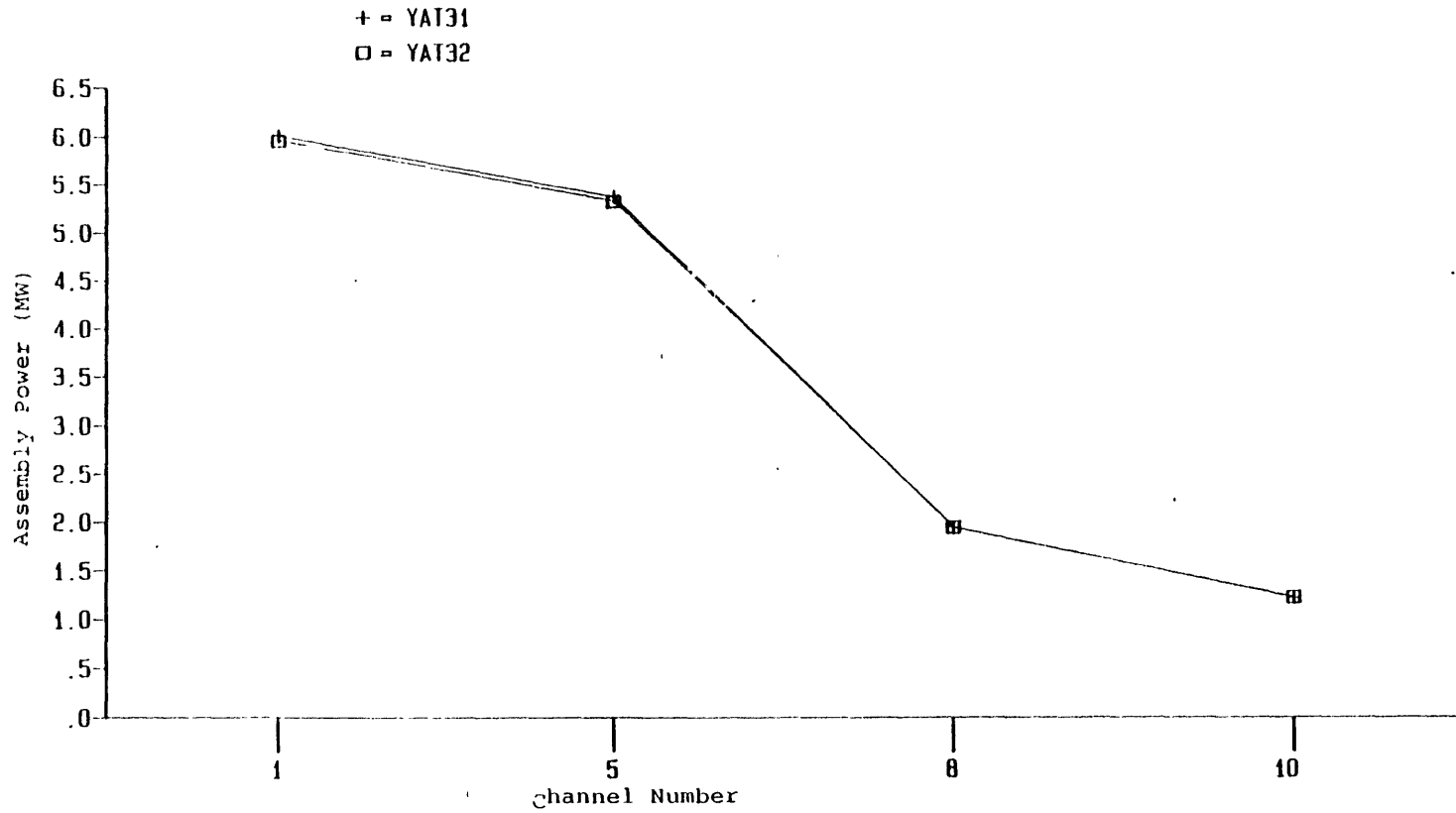Figure 4.30   Radial `assembly power distributions in hot region, at 50 sec, YAT32 and YAT34.

+ = YAT32
□ = YAT34

Figure 4.31 Radial assembly power distibutions in cold region, YAT32 and YAT34.
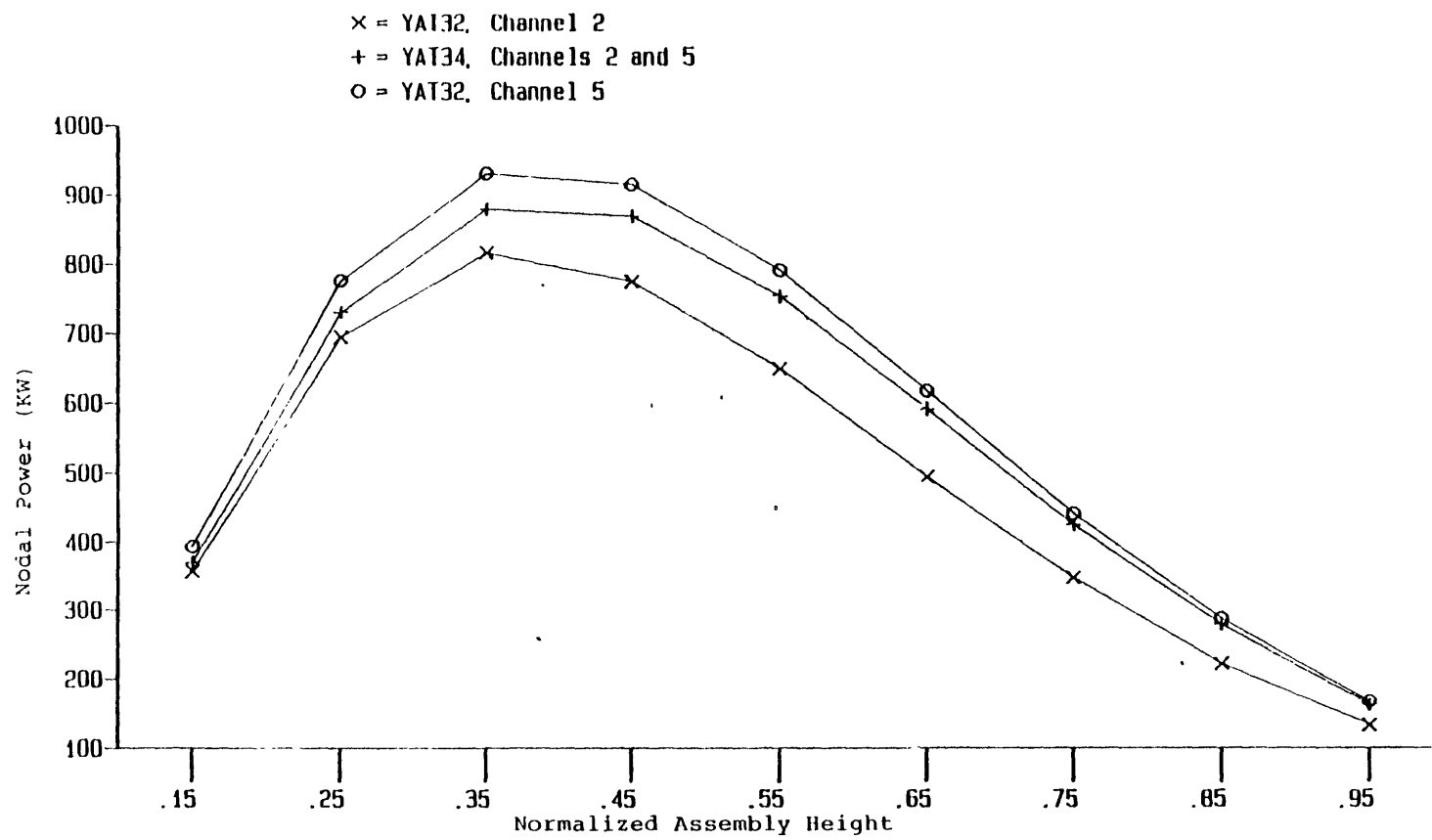
Figure 4.32   Axial nodal power distributions of channels 2 and 5 at 50 sec,
YAT32 and YAT34.
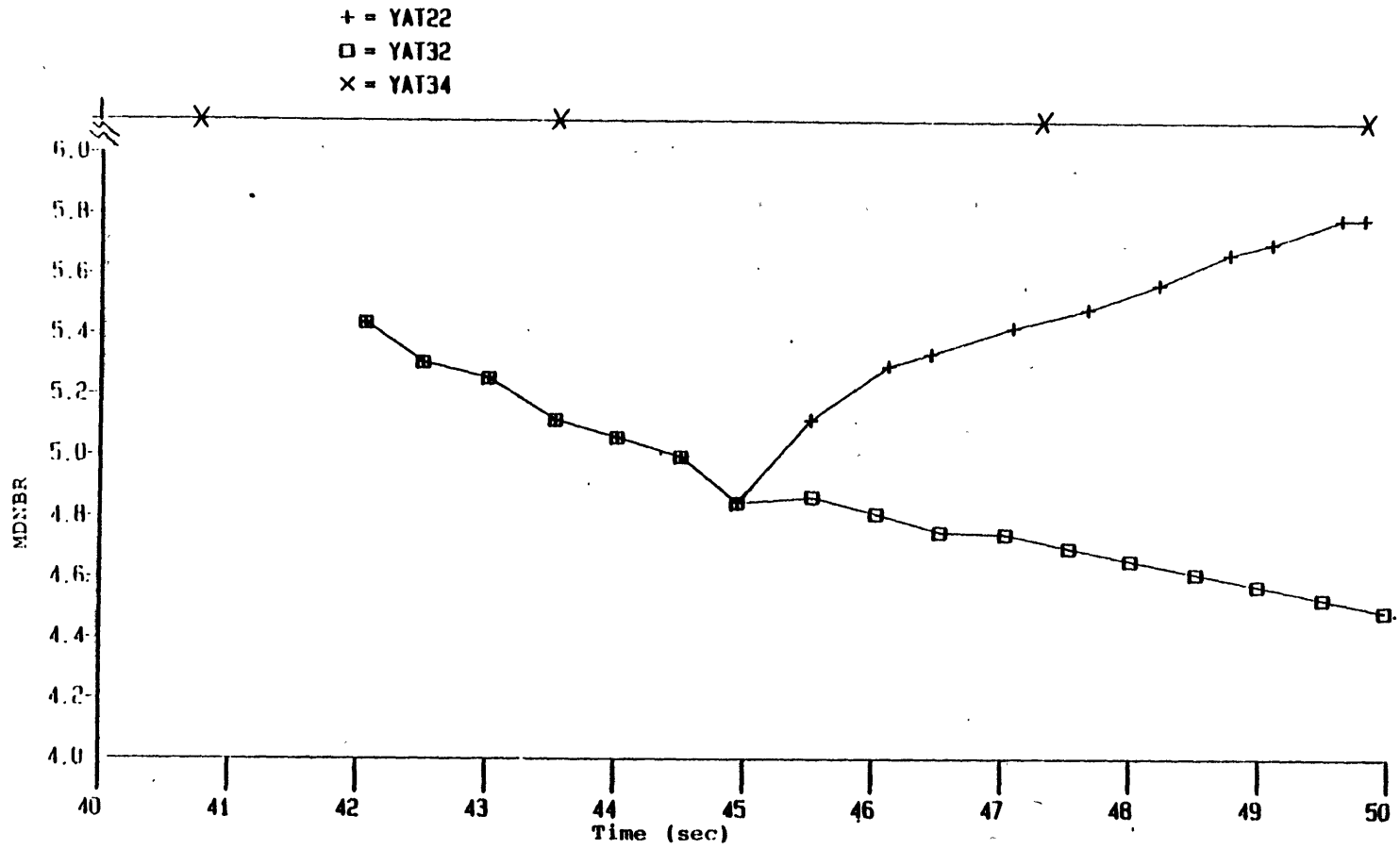
Figure 4.33   MDNBR histories of YAT22, YAT32, and YAT34.
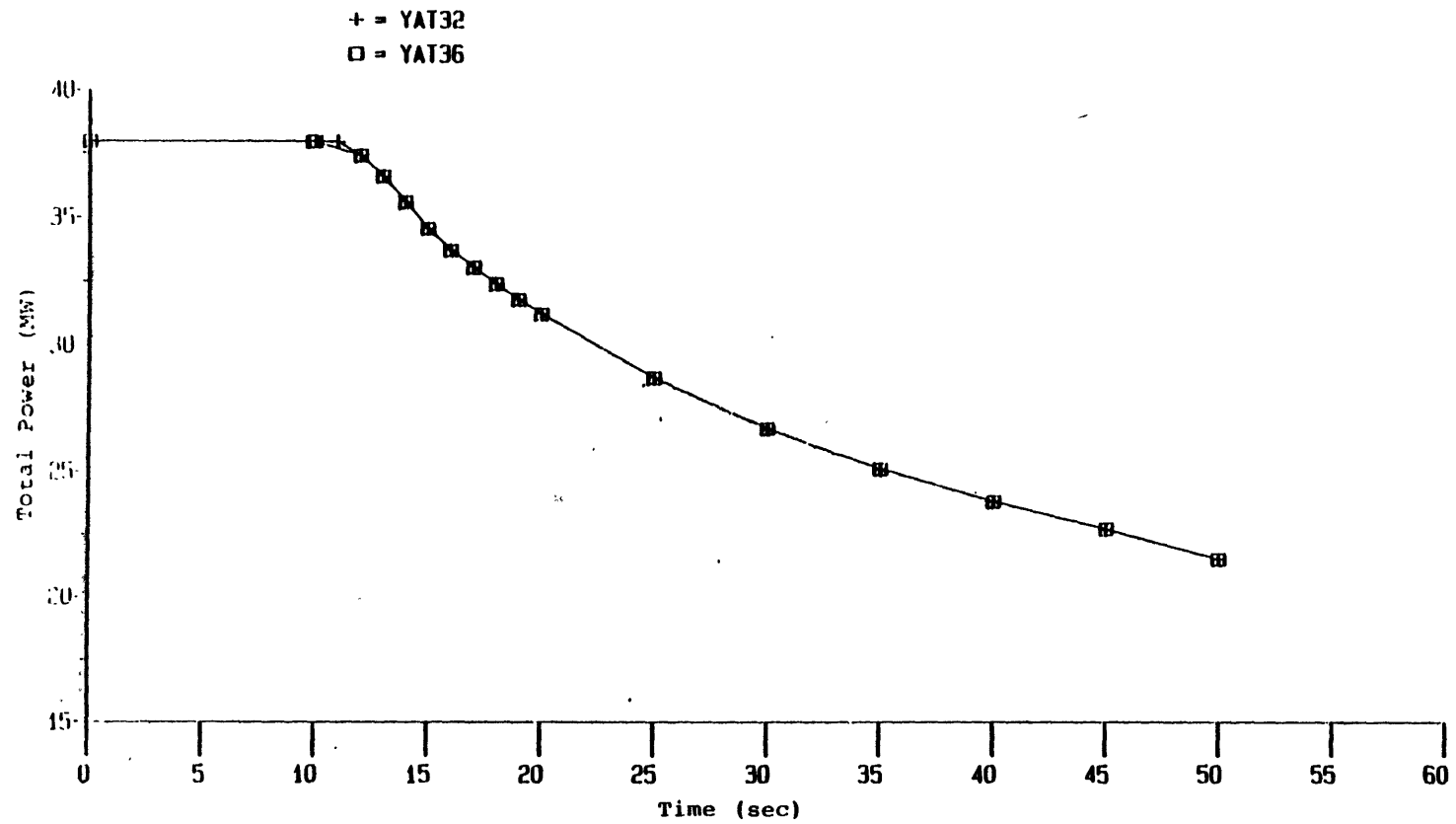
Figure 4.34   Total power histories of YAT32 and YAT36.
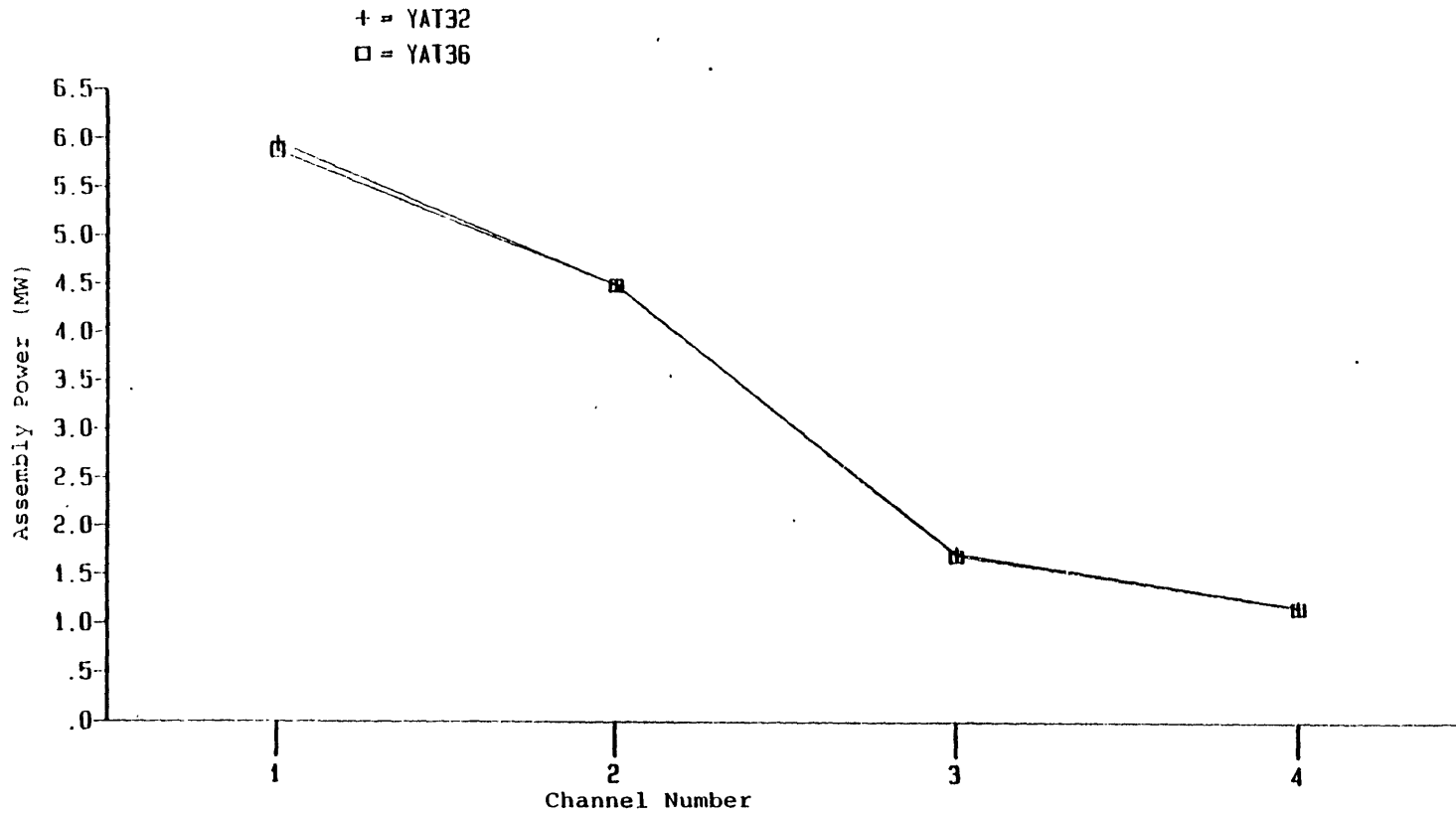
+ = YAT32
□ = YAT36

Figure 4.35   Radial assembly power distributions in hot region, at 50 sec,
YAT32 and YAT36.
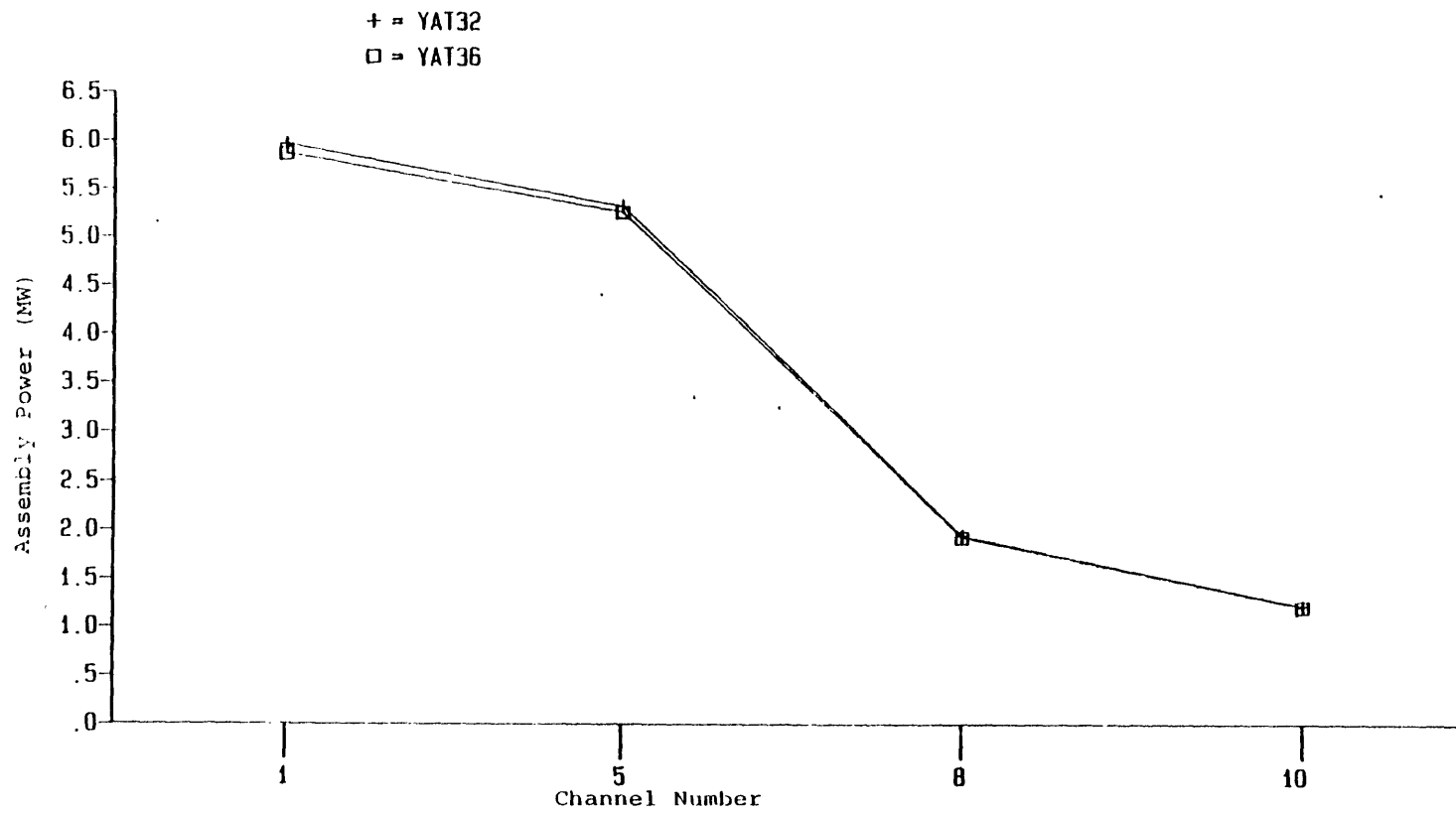
Figure 4 .36   Radial assembly power distributions in cold region, at 50 sec,
YAT32 and YAT36.
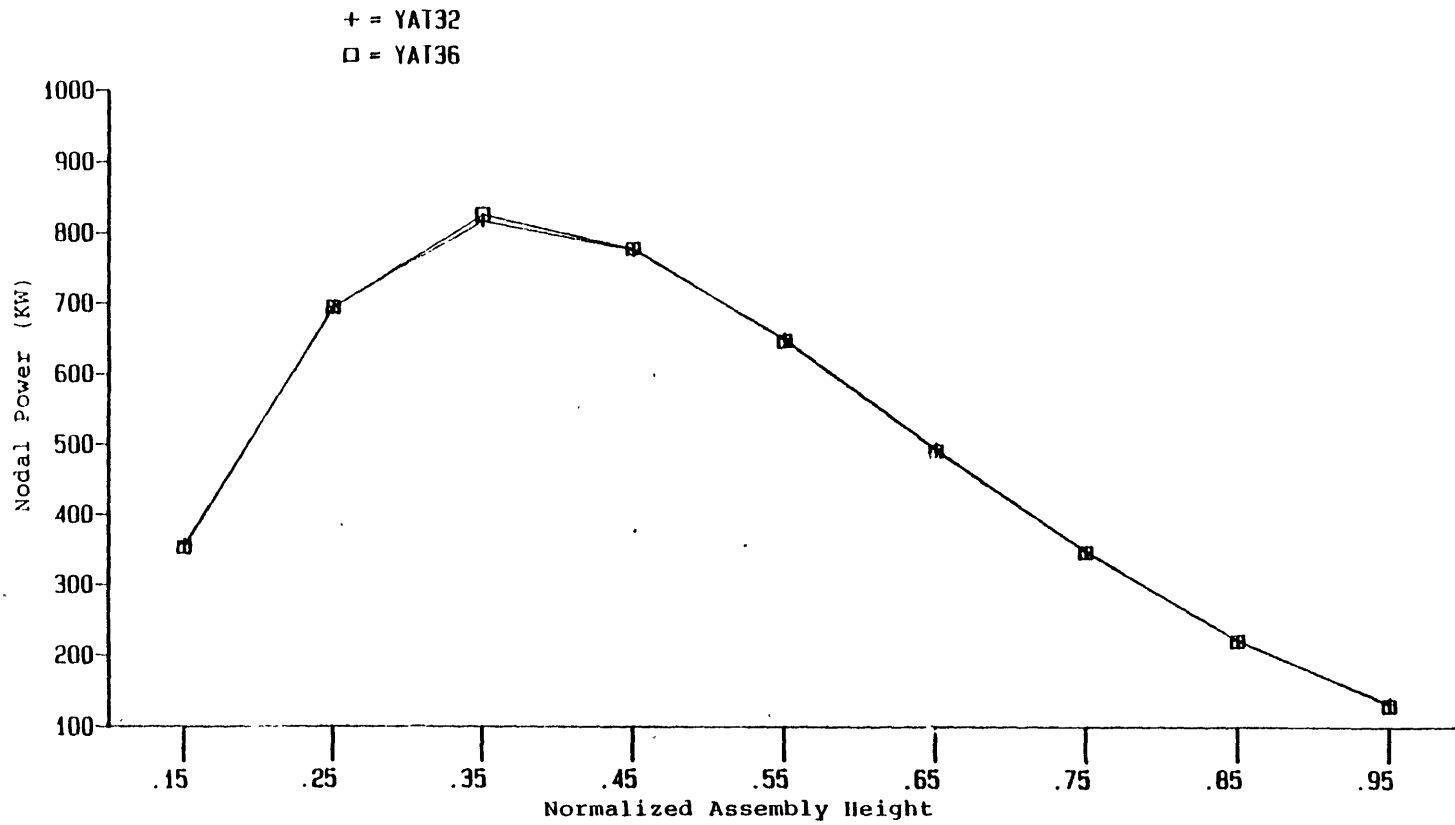
Figure 4.37   Axial nodal power distributions of channel 2 at 50 sec,
YAT32 and YAT36.
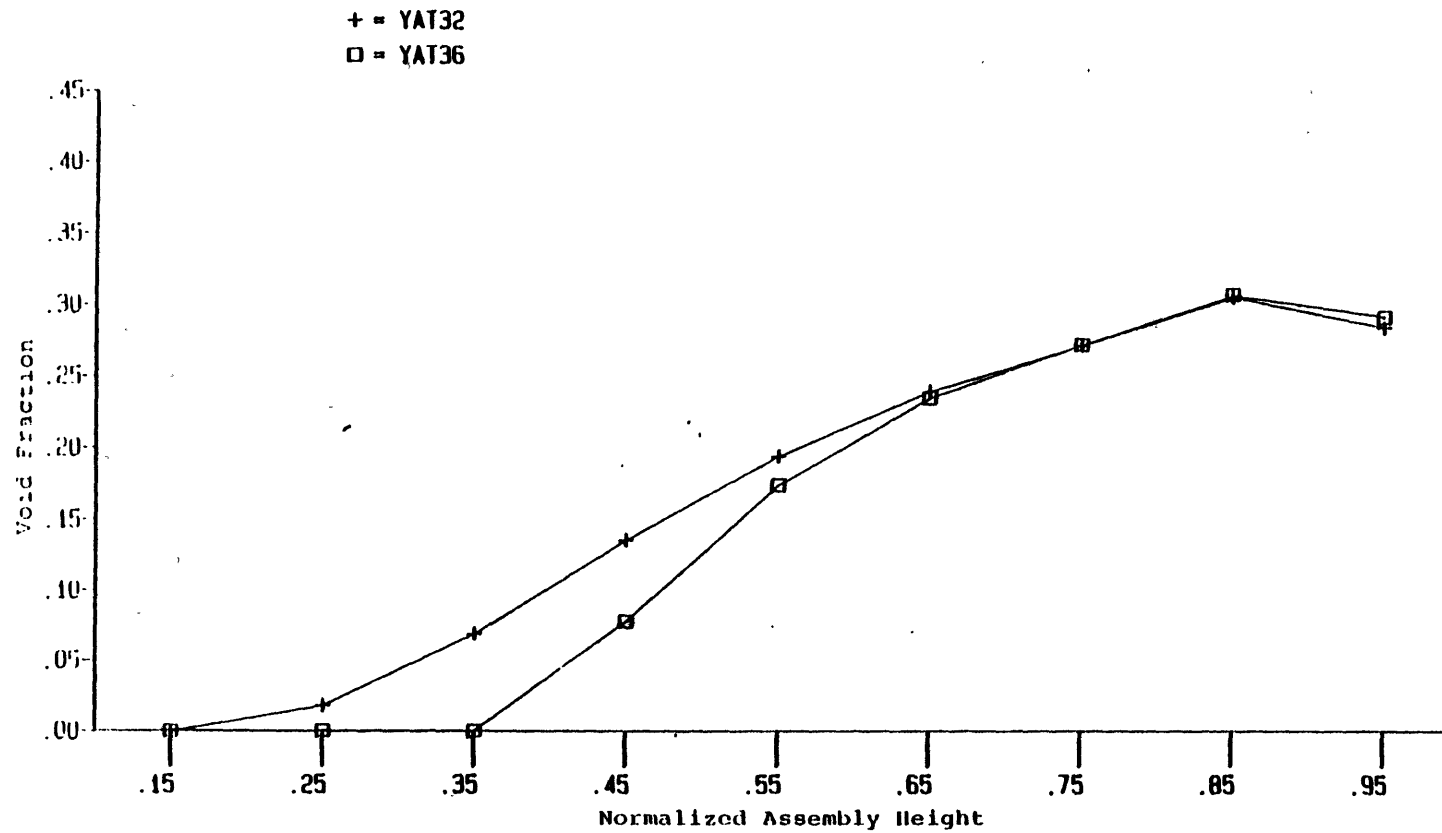
Figure 4.38   Axial void fraction distributions of channel 2 at 48 sec.
YAT32 and YAT36.
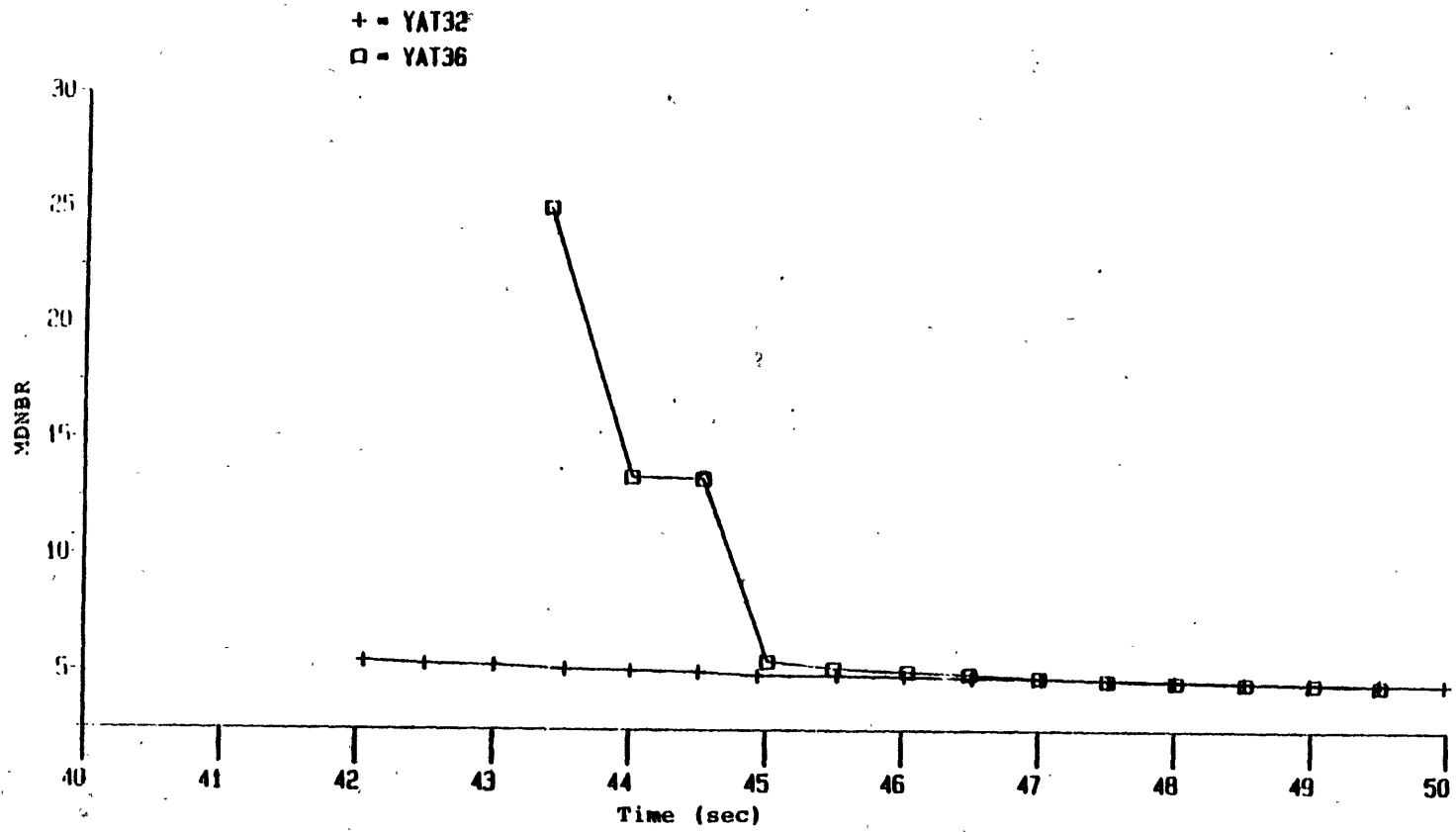
Figure 4.39 MDNBR histories of YAT32 and YAT36.

return to power occurs in a subcooled condition, subcooled boiling may be the only boiling phenomenon. The combination of high peaking power and possible vapor film covering fuel rod surface may result in fuel damage. Nigmatulin model will not be able to predict such a situation. In addition, it seems that the computational effort is not affected too much by employing either model (22893.41 vs. 21076.48 cpu seconds for YAT36 and YAT32, respectively, in a Honeywell 6180 computer). However, it should be noted that the time step sizes selected by the user have an influence on the total cpu time. Therefore, it is suggested that the non-equilibrium mass exchange model be employed for a steam line break accident simulation.

4.5.2 Time Step Size Sensitivity Studies, YAT4X

The time step size for the thermal-hydraulics and neutronics transient calculations has been decoupled as mentioned in Section 2.7. The neutronics time step size can be an integer multiple of the thermal-hydraulics time step size. This is controlled by the parameter 'nfeed' (refer to Appendix A). If 'nfeed' equals 2, the thermal-hydraulics part of the code will perform two time step calculations with time step size Δt. The neutronics part will perform one time step calculation with time step size $2\Delta t$. This procedure is appropriate only when the neutronics transient is not fast. Therefore, it is not appropriate to use this procedure for a scram transient simulation. For a steam line break accident, the transient is dominated by thermal-hydraulics transients after the scram action. The decoupling procedure mentioned above is therefore expected to be applicable in the latter stage of the accident, unless a return to power occurs.

Two calculations (YAT43, YAT44) are performed with 'nfeed' equals 2 and 4, respectively. All other conditions are the same as the case YAT21 (refer to Section 4.3). The two calculations are restarted from a dump file of YAT21 at 42 seconds. The calculations are carried out up to 50 seconds. Actually, this is the period where boiling exists. Also, the accident is dominated by thermal-hydraulics transients in this period. All results (Figures 4.40 to 4.45), including total powers, assembly powers, nodal powers, void fraction distributions, and MDNBR histories show good agreement among the three calculations (YAT21, YAT43, YAT44). As to the computational effort, 1785.74 cpu seconds, 1334.4 cpu seconds and 1178.91 cpu seconds are used by YAT21, YAT43, and YAT44 in the 8 seconds period, respectively. About 25% cpu time is saved with 'nfeed' equals 2 and 34% cpu time is saved with 'nfeed' equals 4.

Based on the above results, it is concluded that 'nfeed' not equal to 1 is a very promising computational effort saving method. Studies on the limitations of the values of 'nfeed' should be performed to miximize the computational effort saving of the procedure in a steam line break accident analysis without compromising accuracy.
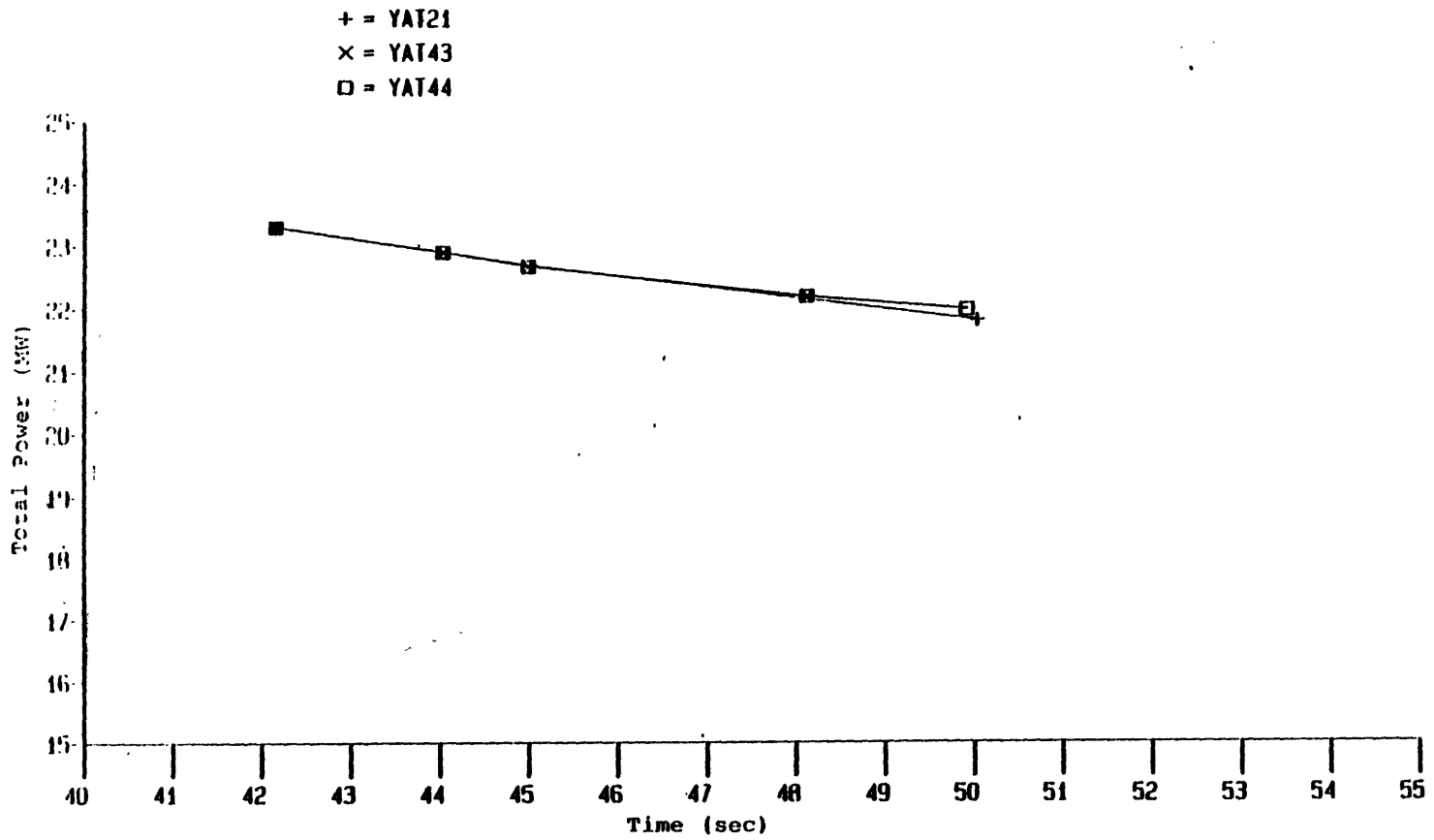
Figure 4.40   Total power histories of YAT21, YAT43, and YAT44.
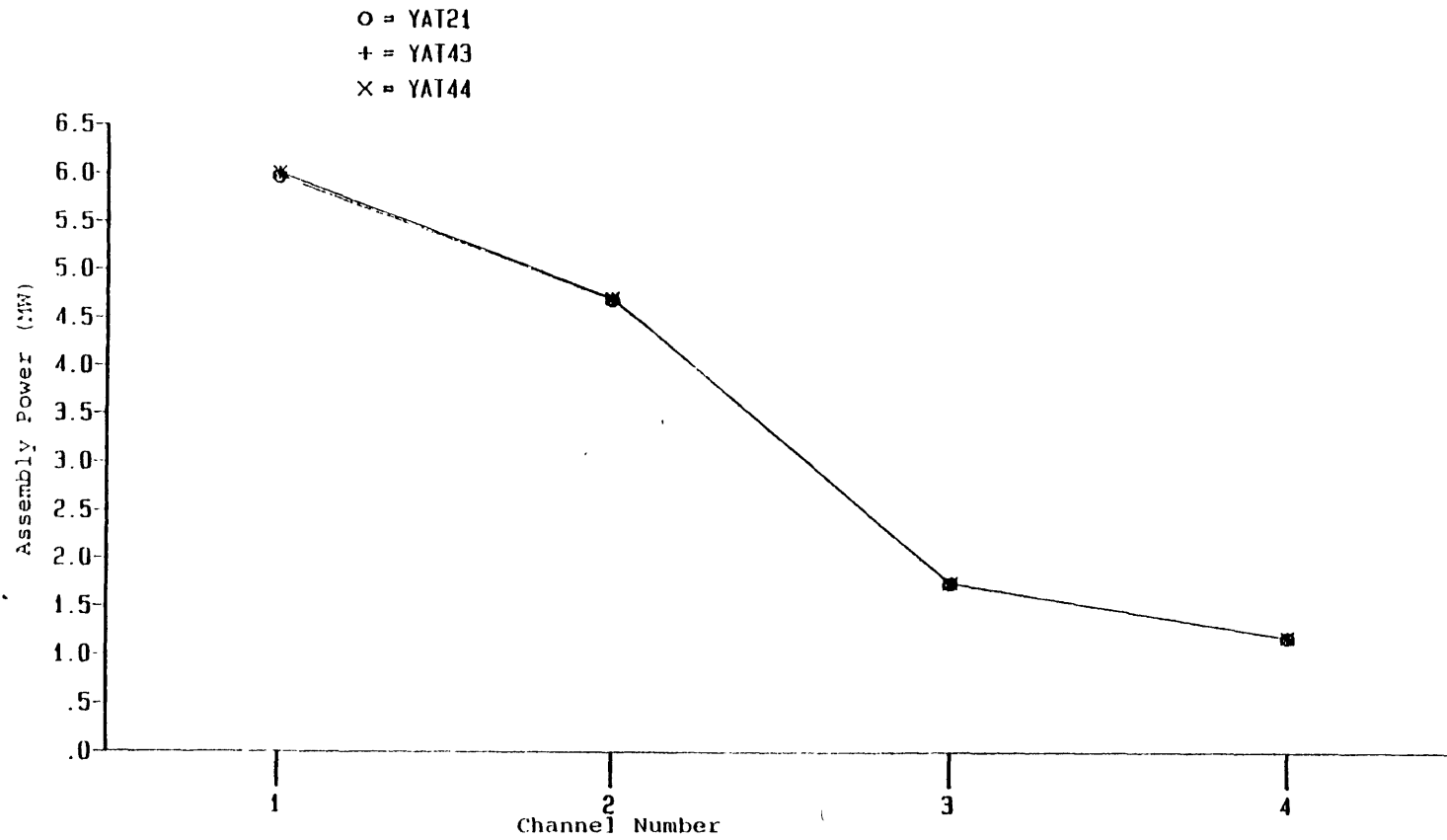
Figure 4.41   Radial assembly power distributions in hot region, at 50 sec,
YAT21, YAT43, and YAT44.
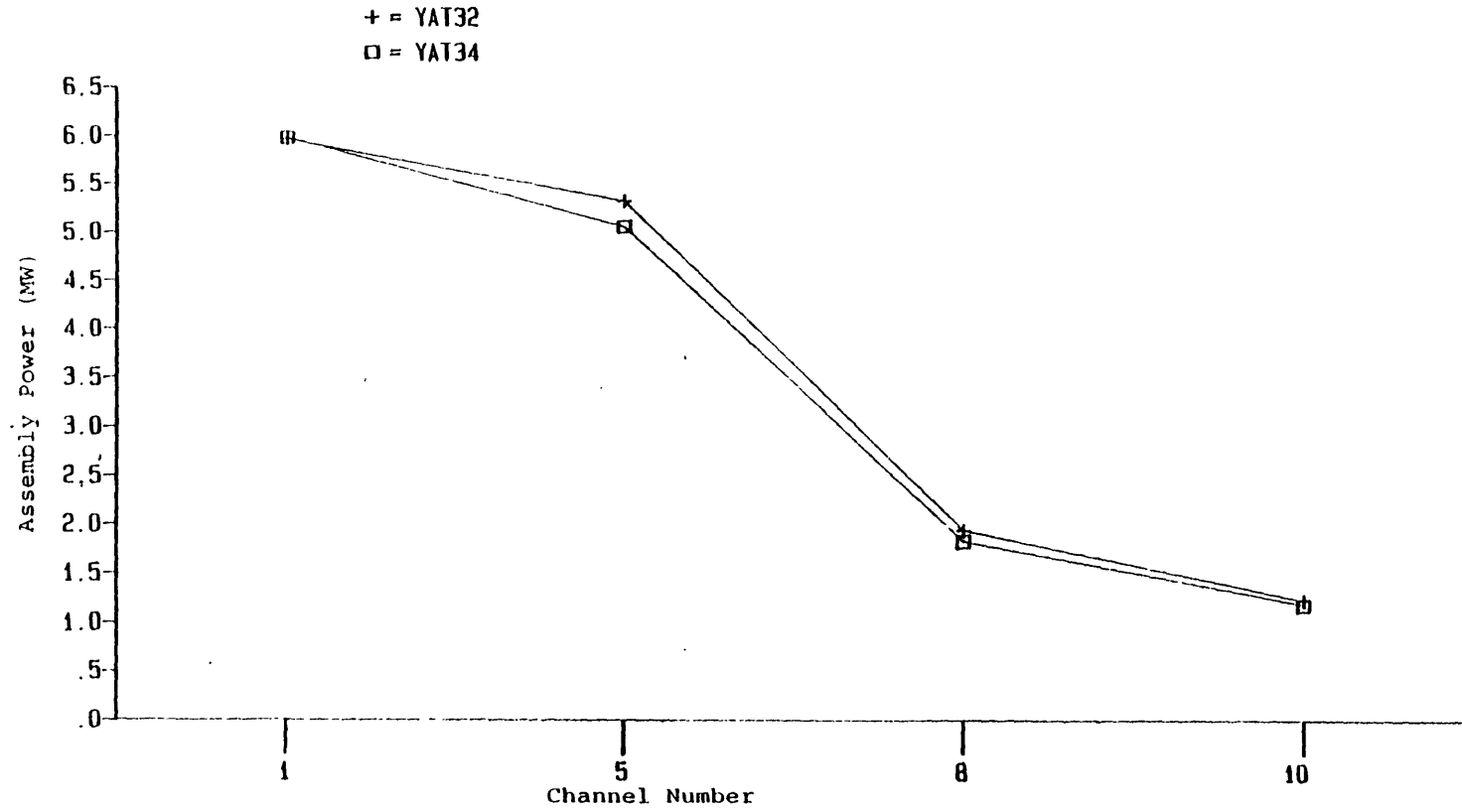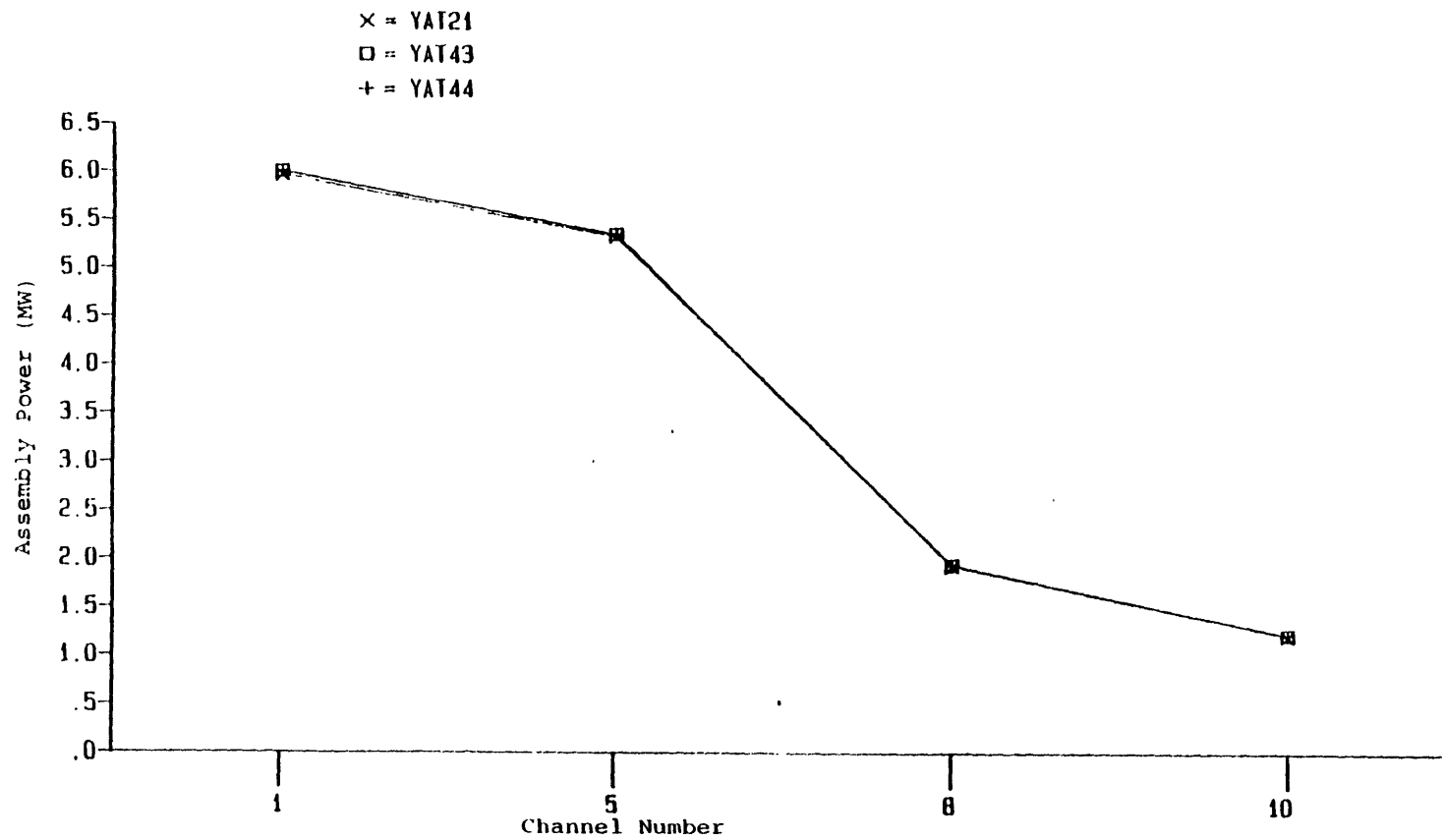
Figure 4.42   Radial assembly power distibutions in cold region, at 50 sec, YAT21, YAT43, and YAT44.
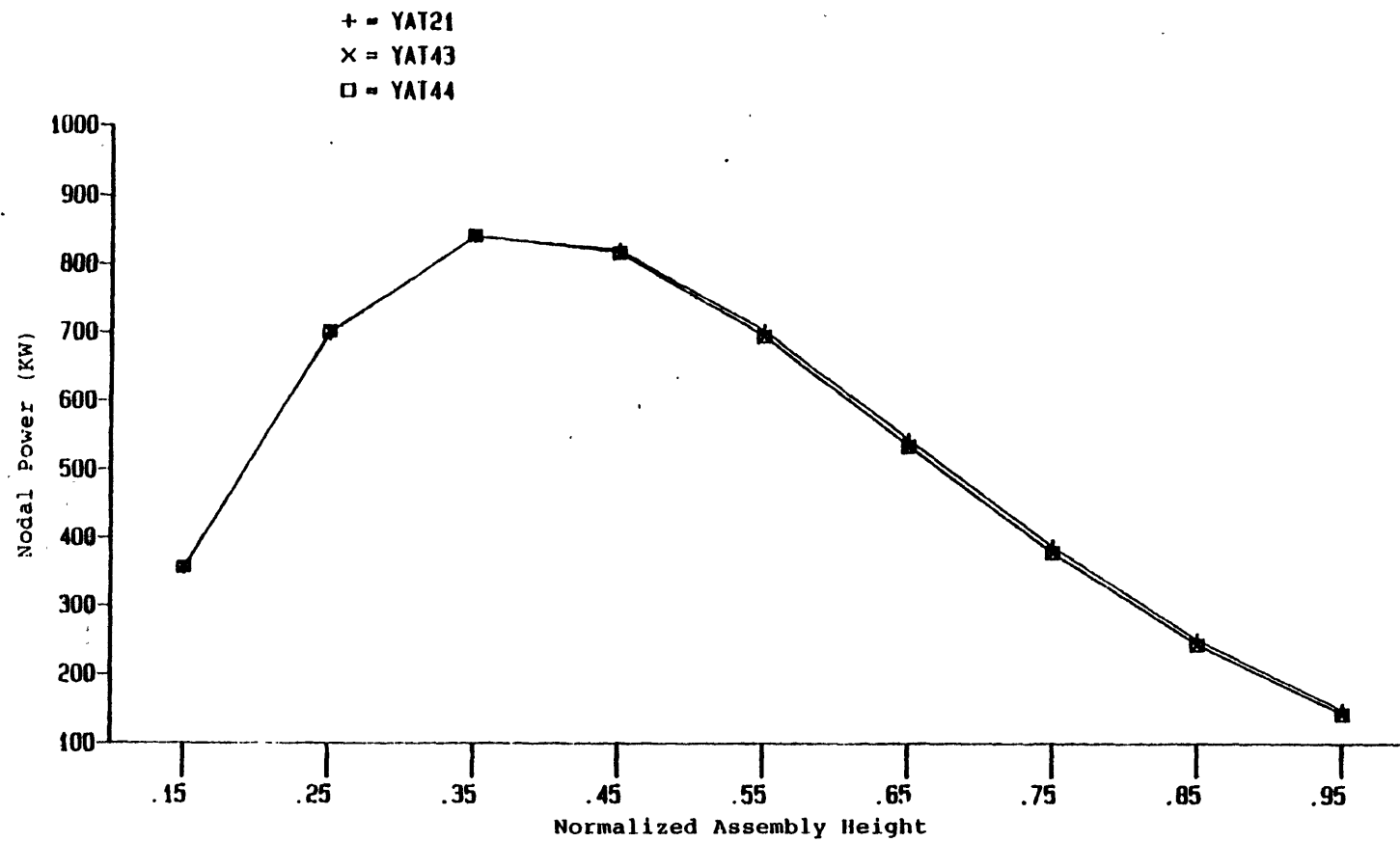
Figure 4.43   Axial nodal power distributions of channel 2 at 50 sec,
YAT32, YAT43, and YAT44.
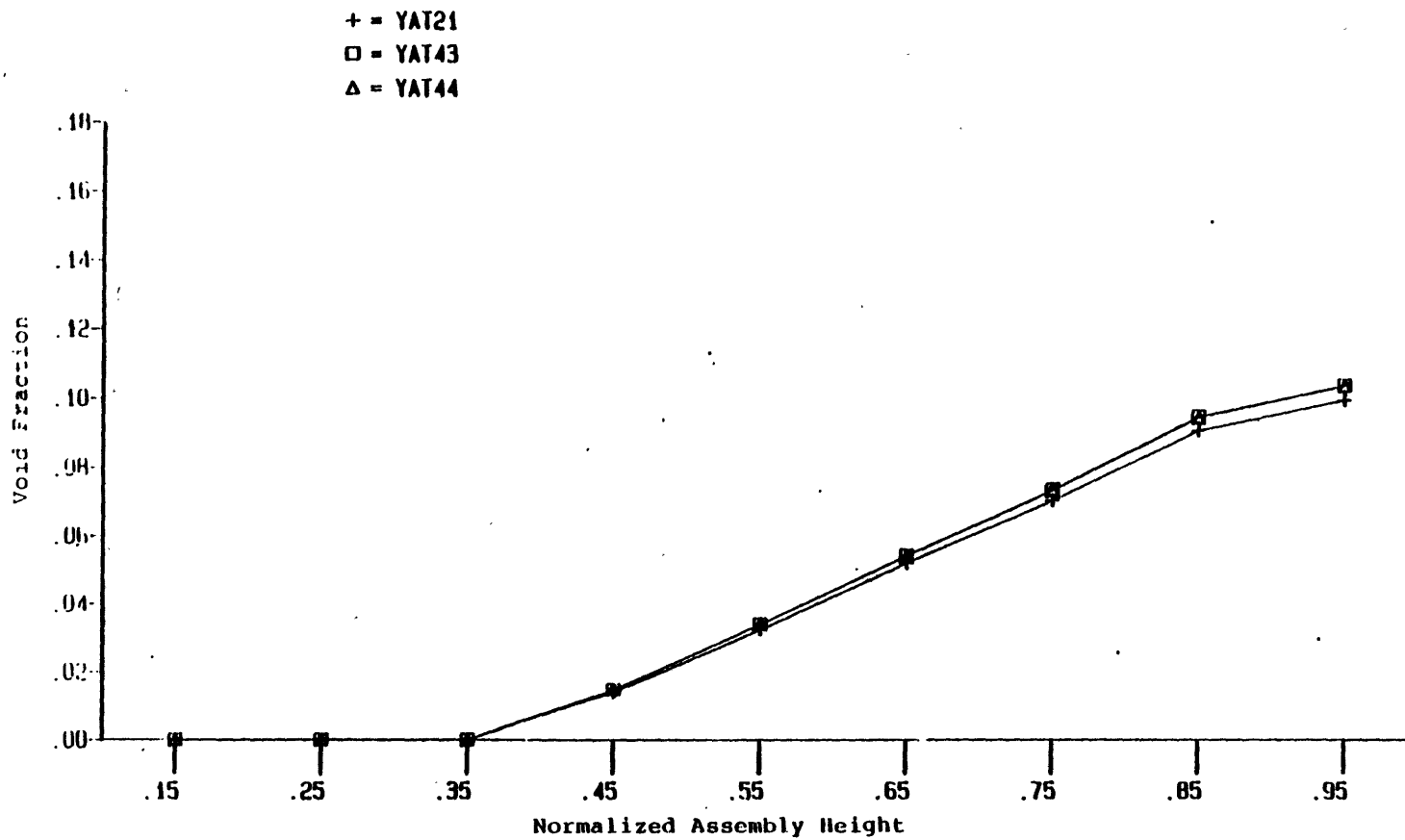
+ = YAT21
□ = YAT43
△ = YAT44

Figure 4.44   Axial void fraction distributions of channel 2 at 50 sec.
YAT21, YAT43, and YAT44.
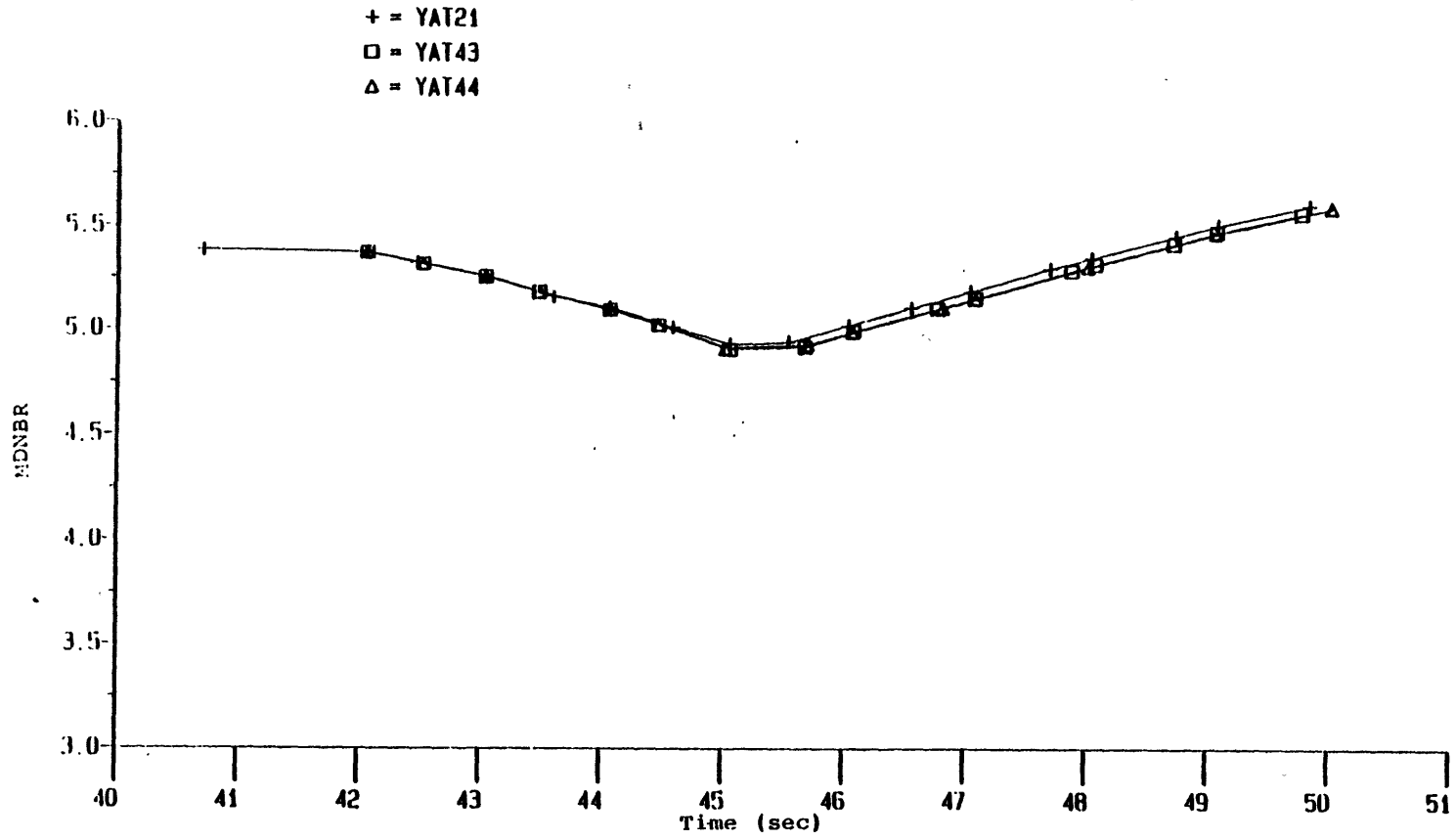
Figure 4.45 MDNBR histories of YAT21, YAT43, and YAT44.

4.6 Summary

The three-dimensional effects of a steam line break accident were studied in this chapter. Seven calculations were performed as summarized in Table 4.1. The conclusions drawn from these calculations are summarized as follows:

1. Before boiling occurs, there is no significant difference in the total power results of 3-D (3D), 1-D non-uniform inlet coolant temperature distribution (1DN), and 1-D, uniform inlet coolant temperature distribution (1DU) calculations.

2. After boiling occurs, the global parameters (total powers) are not significantly affected by using either one of the three approaches mentioned above.

3. Continuously increasing differences between the assembly powers and more particularly the nodal powers (local parameters) of the 3D and 1DN calculations is found after boiling occurs.

4. The assembly and nodal powers calculated by 1DU approach lie in between the hot region and cold region results of the 3D calculation.

5. The void fractions predicted by a 3D calculation are lower than those predicted by a 1DU calculation, and 1DN calculation.

6. The 3D calculations predict more conservative MDNBR than 1DN calculations during a void generation process because of the liquid cross flow is driven out of the hot node.

7. In a void condensation process, the MDNBR predicted by the 1DN calculation is more conservative, since no cold coolant is allowed to cross channel boundaries.

8. The starting point of boiling predicted by a 1DU calculation will be much later than the prediction by a 3D or a 1DN calculaiton.

9. A non-equilibrium interphase mass exchange model is suggested for a steam line break accident analysis.

In addition to the above calculations, two more calculations were performed to study the effect of time step size decoupling for thermal-hydraulics and neutronics calculations as summarized in Table 4.1. The conclusions are summarized as follows:

1. The decoupling logic in the present version of TITAN is good for thermal-hydraulic dominant transients, since the time step size of neutronics calculation is an interger multiple of that of thermal-hydraulics calculation.

2. The results predicted by using 'nfeed' up to 4 agrees with the results using 'nfeed' equals 1 very well for 1DN calculations.

3. Up to 34% cpu time can be saved in a Honeywell 6180 computer by using 'nfeed' equals 4 for 1DN calculations.

Table 4 .1

Summary of Calculations in Chapter 4

Case ID	Scram	Pressure	Flow Rate*	Temp.	nfeed	Γ
YAT11**	Yes	OP	Yes	TZ	1	SB
YAT21**	Yes	RPC	Yes	TZ	1	SB
YAT22	Yes	RPC	Yes	TZ	1	SB
YAT31**	Yes	RP	Yes	TZ	1	SB
YAT32	Yes	RP	Yes	TZ	1	SB
YAT34***	Yes	RP	Yes	OZ	1	SB
YAT36	Yes	RP	Yes	TZ	1	NM
YAT43**	No	RPC	Yes	TZ	2	SB
YAT44**	No	RPC	Yes	TZ	4	SB

OP: Original Pressure Forcing Function, Fig. 4 .3.
RPC: Reduced Pressure Forcing Function with Constant
     Pressure after 45 Seconds, Fig. 4.10.
RP: Reduced Pressure Forcing Function, Fig. 4 .21.
TZ: Three-Zone Inlet Coolant Temperature, Fig. 4.2.
OZ: One- Zone Inlet Coolant Temperature, Fig. 4.2.
SB: Non-Equilibrium Subcooled Boiling Model.
NM: Nigmatulin Model.
*   Refer to Fig. 4.4.
** One-Dimensional, Non-Uniform Inlet Coolant Temperature
     Distribution Calculations.
*** One-Dimensional, Uniform Inlet Coolant Temperature
     Distribution Calculations.

More sensitivity studies are suggested to verify the appropriateness of applying the decoupling time step size to three-dimensional calculations. Also, the limiting value of 'nfeed' without losing accurate predictions should be found to save more computational effort.

Table 4.2 summarizes the computational statistics of the nine calculations discussed in this chapter. The values of cpu per time step per node for 'nfeed' equals 1 are in the same order. The computational effort saving by using 'nfeed' greater than one is clearly seen by the cpu per time step per node results of YAT21, YAT43 and YAT44. The savings come from the less computational effort used in the neutronics calculations of YAT43 and YAT44.

Table 4.2
Summary of Computational Statistics

	Time Steps	cpu Time (sec)*	cpu time/node/step
YAT11	800	8103.04	0.1013
YAT21	802	10239.22	0.1277
YAT22	922	16097.39	0.1746
YAT31	855	11184.59	0.1308
YAT32	929	21076.48	0.2269
YAT34	810	12027.77	0.1485
YAT36	1253	22893.41	0..1827
YAT43	161	1334.40	0.0829
YAT44	161	1178.91	0.0732

* In A Honeywell 6180 Computer.

Chapter 5

TRANSIENT SIMULATIONS; RETURN TO POWER STUDIES

5.1 Introduction

The concern about a steam line break accident involves the possibility of a return to power. This may happen if the positive reactivity insertion introduced by the cooldown process compensates the negative reactivity caused by the scram action. The expected result of a return to power condition is high power peaking and hence high fuel temperature. Fuel rod damage, therefore, may occur in such transients. In general, the design base of fuel rod center limiting temperature for UO_2 is the grain-growth temperature of UO_2 (1400 to 1700 $°C$) [28]. An acceptable reactor design should not only never reach the melting point of UO_2 (2800 $°C$) but should also always be below the grain-growth temperature of UO_2 during any normal or abnormal transient. To avoid the return to power condition, a boron solution injection is combined with the HPSI (High Pressure Safety Injection) action in the transient. Because of the high neutron absorption cross section of boron-10, it is expected that the net positive reactivity insertion by the cooldown process and scram may be compensated by the negative reactivity insertion of boron-10.

It is obvious that one-dimensional computer codes are not suitable for such transient analyses, since no detailed local information (such as power peaking and fuel rod center line temperature) is provided by such codes. A three-dimensional computer code, or at least a one-dimensional, multi-channel computer code (provided no significant cross flow is expected), should be employed for the analysis. In this chapter, the return to power studies of the ten-channel model of TITAN (Model B described in Chapter 3), are presented. Both open-channel (three-dimensional) and closed-channel (one-dimensional, multi-channel) calculations were performed. The objective of these studies is to indentify the three-dimensional effects of such transients without boron solution injection. With the same 10-channel model, boron injection was simulated to study its effect on the return to power condition. Results are reviewed in the following sections. No boiling was observed throughout the studies. The three-dimensional effects of an SLB accident without return to power but with boiling have been discussed in Chapter 4.

5.2 Steady-State Calculations

It has been shown in the previous chapters that with the current arrangement of Model B, there is no return to power condition. To initiate the return to power condition with Model B, it is necessary to rearrange some of the conditions. Several approaches can be taken. For example, the inlet coolant temperature may be reduced and more positive reactivity will be inserted into the core. However, we have already assumed that the SLB accident is at limiting conditions, i.e., the excess heat removal is at the maximum value and the inlet coolant temperature forcing function should have reached the lowest possible value.

Another approach is to change the nulcear properties of the model, such as the feedback coefficients. As mentioned in the previous chapter, the coolant temperature feedback coefficients provide the most positive reactivity insertion. The coolant density feedback coefficients will have little effect until boiling occurs. The fuel temperature feedback coefficients lead to a kind of self-balance situation as discussed in Section 4.4 and have less effect on positive reactivity insertion than the coolant temperature feedback coefficients during a steam line break accident (see Ref. [5]). It was therefore decided to increase the fission cross section coolant temperature feedback coefficient $(\frac{\partial \Sigma_f}{\partial T_c})$ in order to make a return to power condition possible with Model B.

Since the rearrangement is equivalent to changing the fuel material, it is therefore necessary to perform steady state calculations to generate new steady state results. Three calculations were performed in order to decide how best to simulate an earlier return to power condition. Nothing was changed in Model B as described in Chapter 3 except the fission cross section coolant temperature feedback coefficient which was multiplied by factors of 2, 4, and 10 for the three calculations. The designated case identifications are RTPSS2, RTPSS4, and RTPSS10 for multipliers 2, 4, and 10, respectively.

Figure 5.1 shows the radial assembly power distributions of the original Model B result (i.e., multiplier = 1), RTPSS2, RTPSS4, and RTPSS10. It seems that there is no significant difference among them. Channel 2 has the highest power according to all four calculations.

The axial nodal power distributions of channel 2 are shown in Figure 5.2. The effect of the multipliers to the fission cross section coolant temperature feedback coefficient is revealed in this figure. As the multiplier increases from 1 to 10, the power peak shifts towards bottom. The reason for the power peak shift is that the strong coolant temperature feedback effect for case RTPSS10 causes an earlier bottom peak and therefore supresses the power in the upper region because of the constant total power in a steady state condition. In addition, the higher power in the lower region of the core causes the coolant temperature to be higher

118
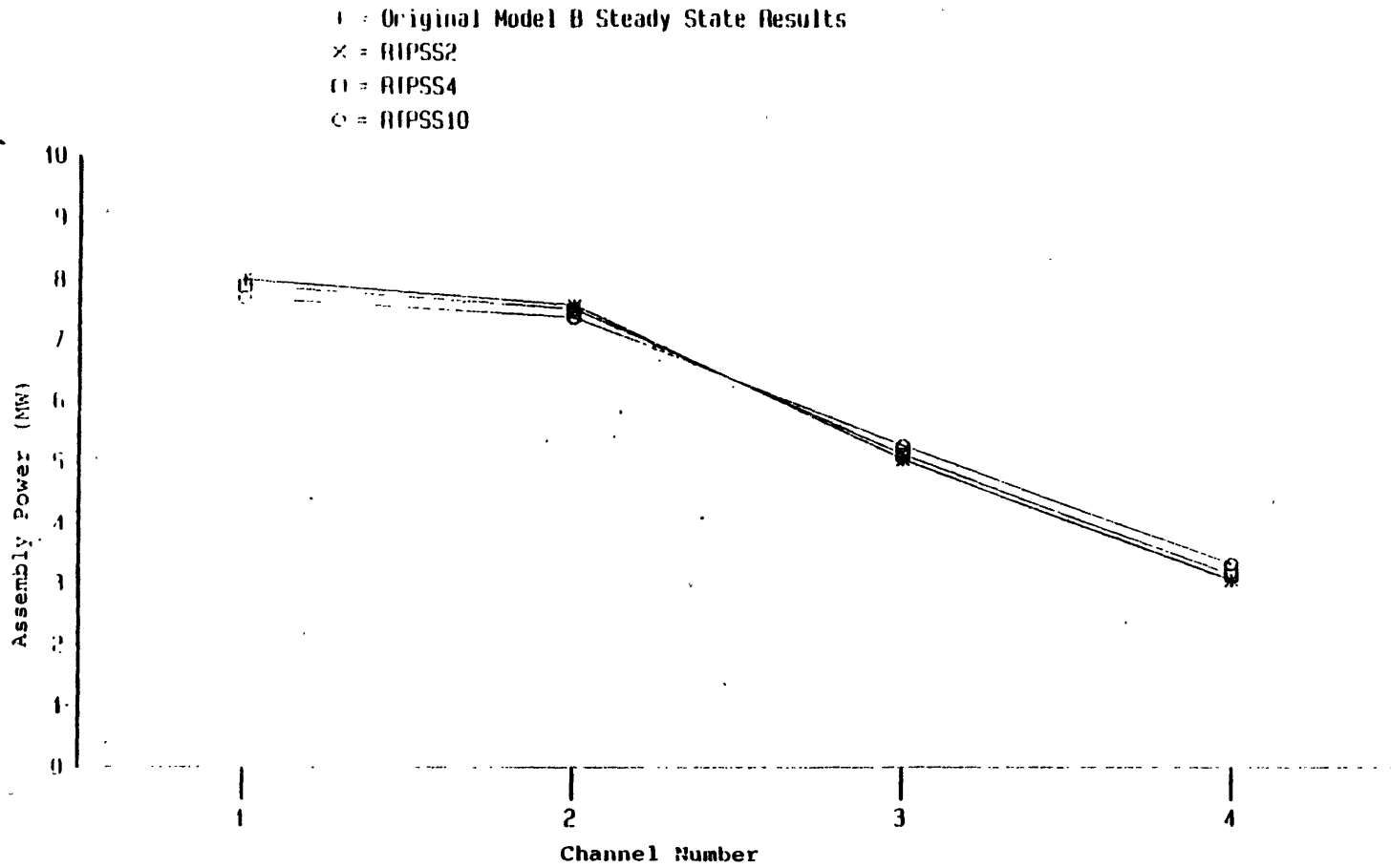
Figure 5.1   Radial assembly power distributions; steady state results of RTPSS2, RTPSS4, RTPSS10, and the original Model B.
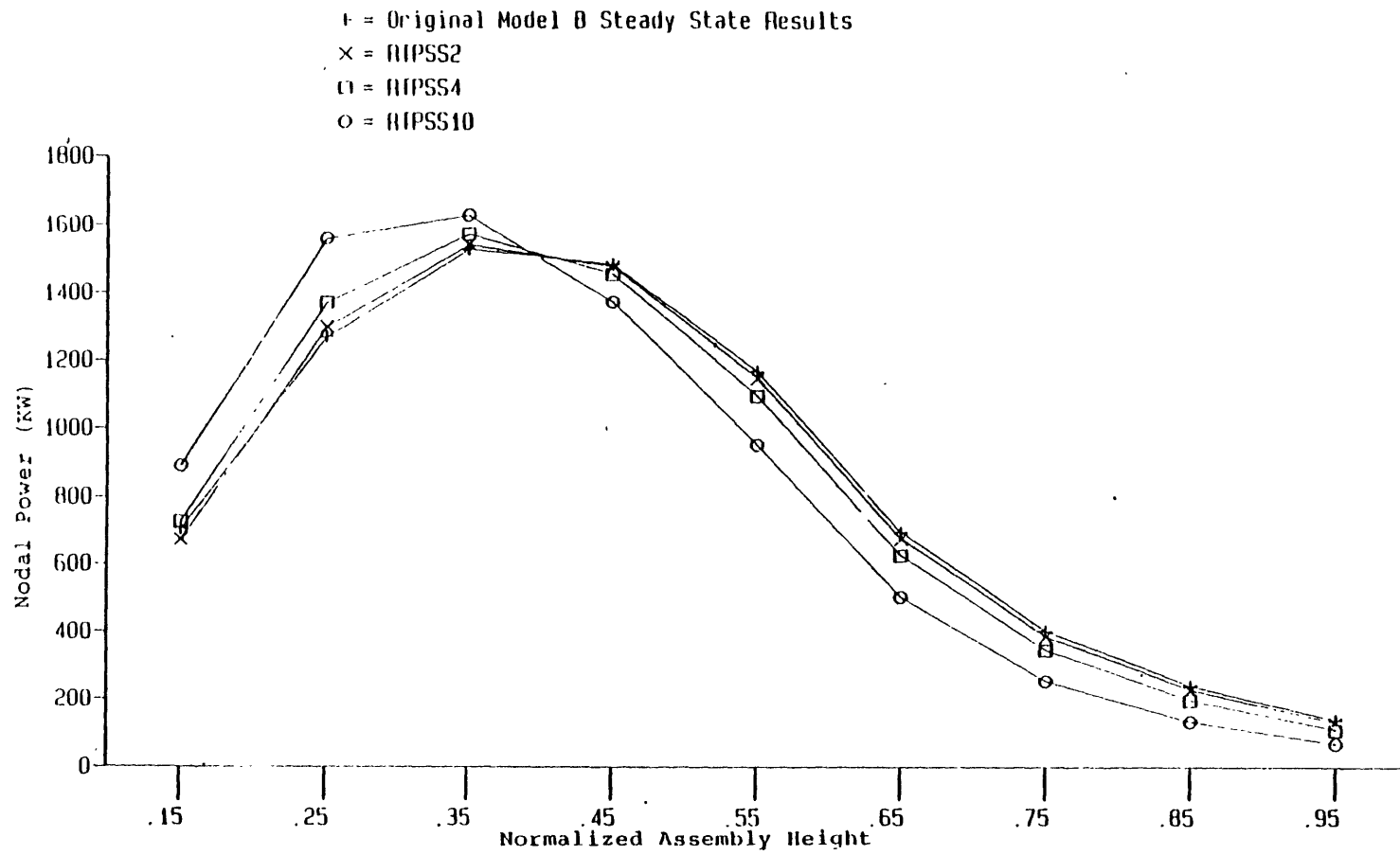
Figure 5.2 Axial nodal power distributions; steady state results of RTPSS2, RPTSS4, RTPSS10, and the original Model B.

in the upper region of the core and hence reduces the power in the upper part of the core.

All the three calculations (RTPSS2, RTPSS4, RTPSS10) were performed by using the one-dimensional fully implicit numerical scheme with the marching solution method as discussed in Chapter 3. The computational statistics and the convergence results are summarized in Table 5.1. From the results, it is shown that the steady state results are tightly converged.

5.3 Return to Power Simulations RTPX

The transient conditions of the return to power simulations were based on the conditions of case YAT11 (see Section 4.2). The first part of the transient calculations is used to determine the fission cross section coolant temperature coefficient multiplier. Three calculations, RTP2, RTP4, and RTP10, were performed restarted from the steady state results of RTPSS2, RTPSS4, and RTPSS10, respectively. After 10 seconds of null transient period, a 10 second transient was followed for each case. Figure 5.3 shows the total power histories of the three calculations. The coolant temperature feedback effect is clearly seen in this figure. As the multiplier increases from 2 to 4, the slope of the power history of RTP4 becomes smaller than that of RTP2. For the case RTP10 (multiplier is 10) the total power starts increasing around 16 seconds. Also, the scram effect is not significant as shown in the figure in the period from 10.9 to 15.9 seconds. To save computational effort, multiplier 10 was chosen for the return to power studies and case RTP10 was continued up to 40 seconds.

RTP10 is a closed-channel calculation. Another calculation, RTP12, restarted from the steady state results of RTPSS10, had exactly the same conditions as RTP10 except cross flow between channels was allowed in this case. That is, RTP12 is an open-channel calculation. The total power histories of the two calculations are shown in Figure 5.4. No boiling was found up to 40 seconds simulation for both calculations. It is clear from Figure 5.4 that there is no singificant difference between the global results of the two calculations. Figures 5.5 and 5.6 show the radial assembly power distributions in the hot and cold regions of RTP10. Channel 5 in the cold region has the highest assembly power and the ratio of channel 5 assembly power to the average assembly power is 1.6238 at 40 seconds. Figures 5.7 and 5.8 show the radial assembly power distributions of case RTP12. Again, channel 5 has the highest assembly power and the ratio of the assembly power to the average assembly power is 1.6093 at 40 seconds. The comparison between the open- (RTP12) and closed-channel (RTP10) results for the radial assembly power distributions is shown in Figure 5.9. The distributions in the cold region at 40 seconds are shown in the figure. No significant difference is found.

121

Table 5.1

Steady State Computational Statistics and
Convergence Results of RTPSS2, RTPSS4, and RTPSS10

	RTPSS2	RTPSS4	RTPSS10
Time Step	5	5	5
CPU Time* (sec)	236.84	265.27	264.82
Energy ** Error	-2.3820×10^{-3}	-2.3843×10^{-3}	-2.3829×10^{-3}
Flow Rate Error**	1.2887×10^{-7}	2.9284×10^{-7}	1.9913×10^{-7}
qchk**	-5.8424×10^{-6}	-1.1620×10^{-5}	-7.5836×10^{-6}
wchk**	-2.2262×10^{-7}	-4.6866×10^{-7}	-3.0463×10^{-7}
Win**	7.0257×10^{-8}	1.9913×10^{-7}	1.0541×10^{-7}

* In a Honeywell 6180 Computer

** Defined in Chapter 3.

Figure 5.3   Total power histories of RTP2, RTP4, and RTP10.
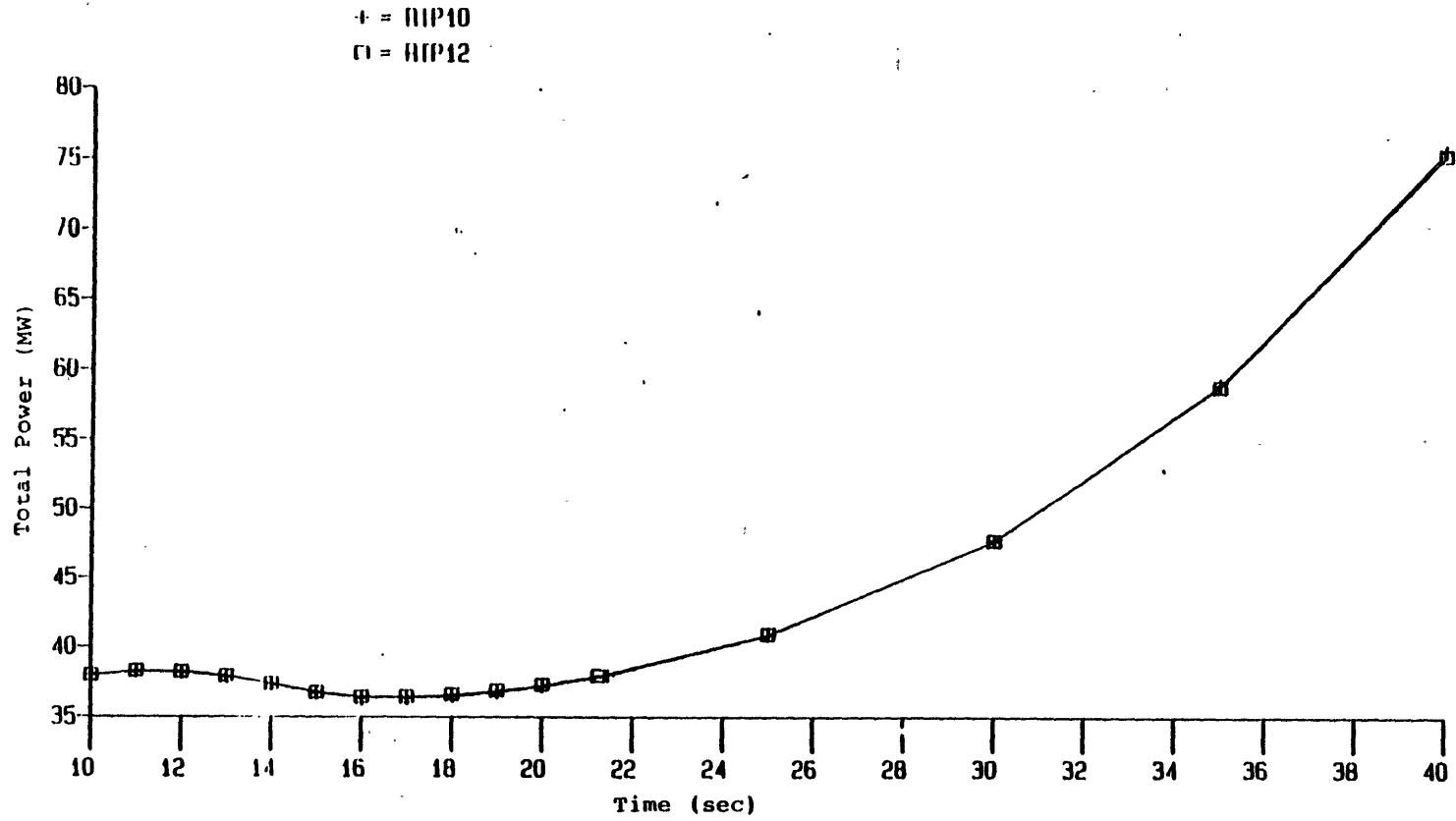
Figure 5.4 Total powwer histories of RTP10 and RTP12.
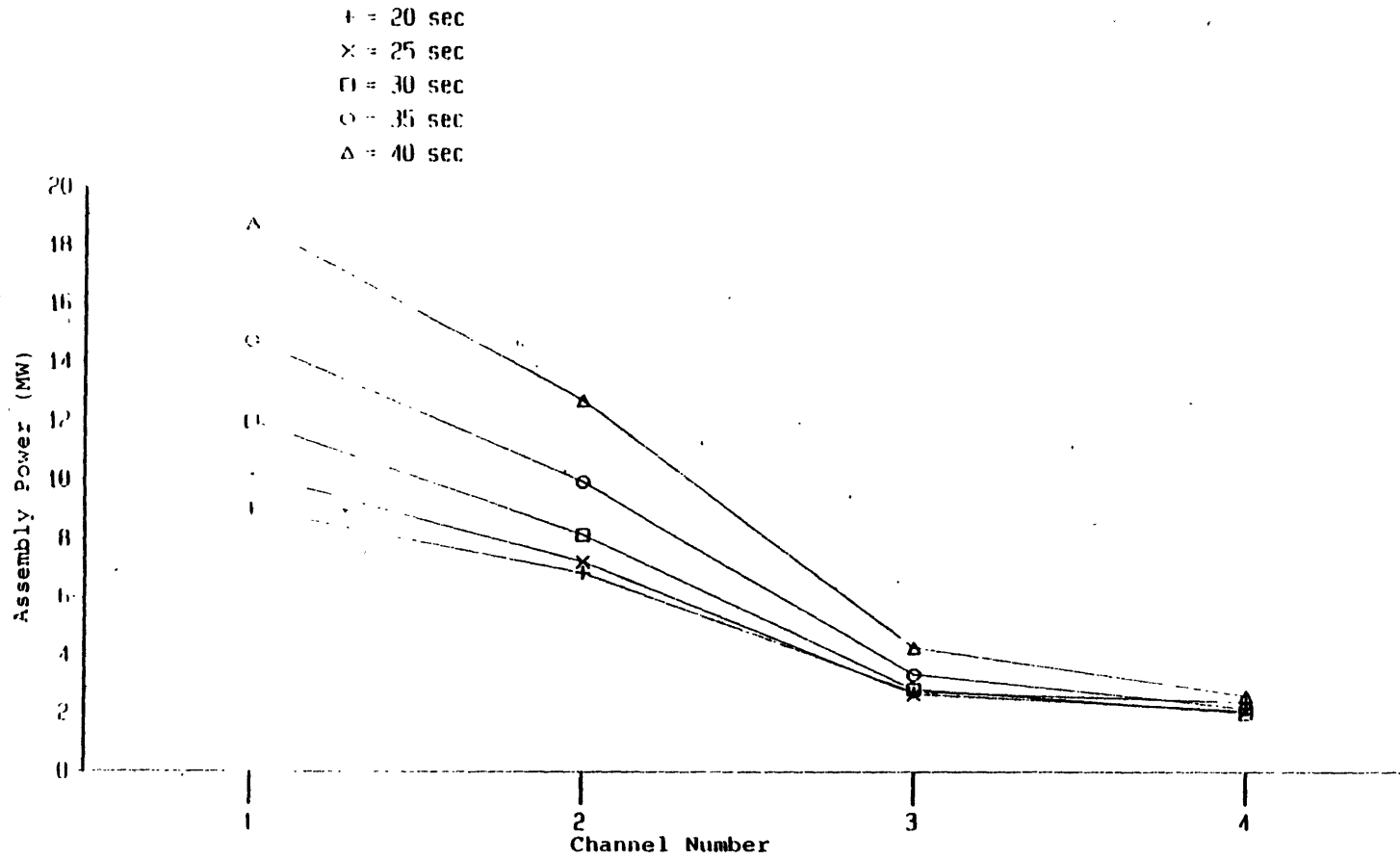
Figure 5.5 Radial assembly power distributions in the hot region, RTP10.
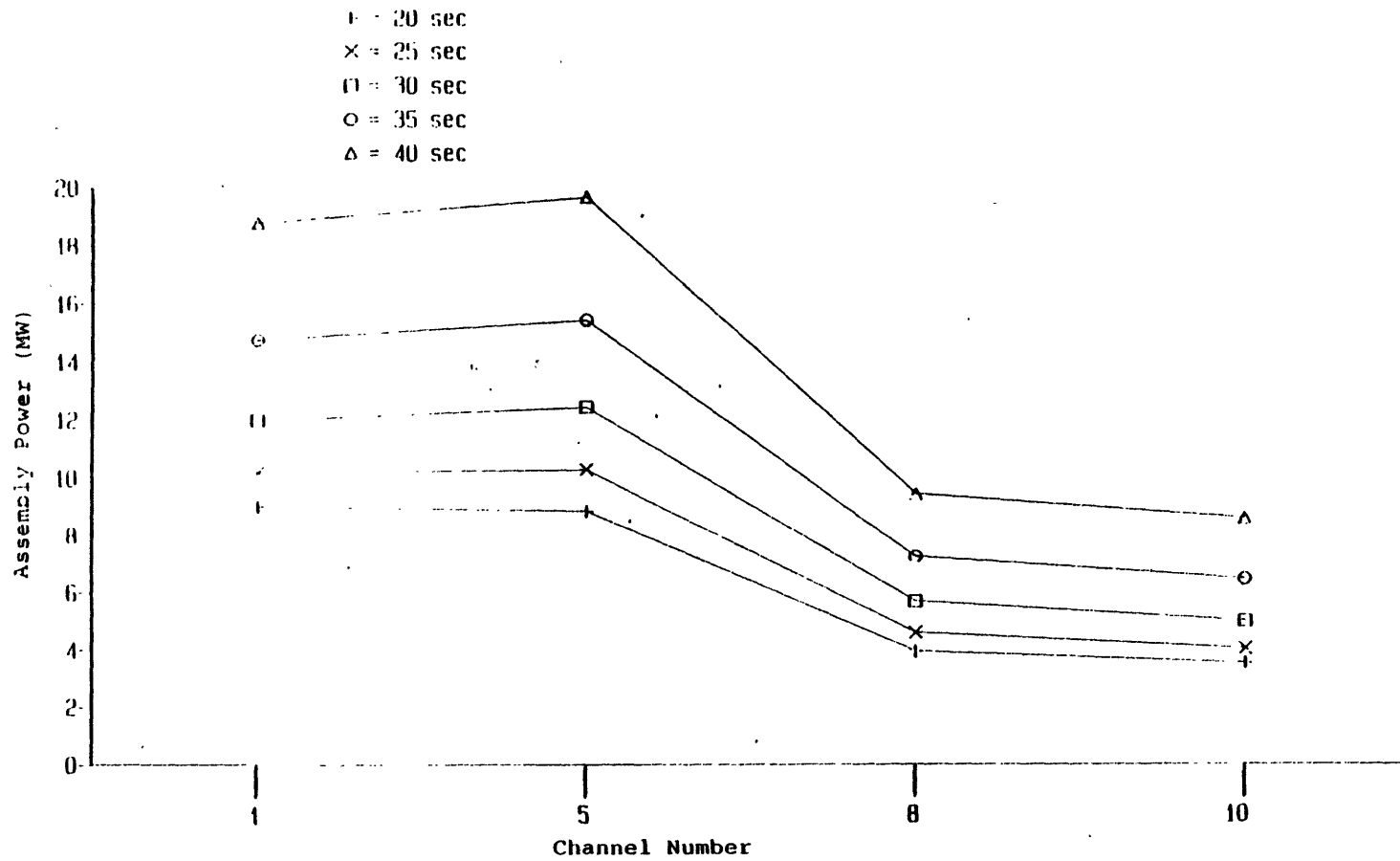
Figure 5.6 Radial assembly power distributions in the cold region, RTP10.

I = 20 sec
× = 25 sec
II = 30 sec
O = 35 sec
∧ = 40 sec
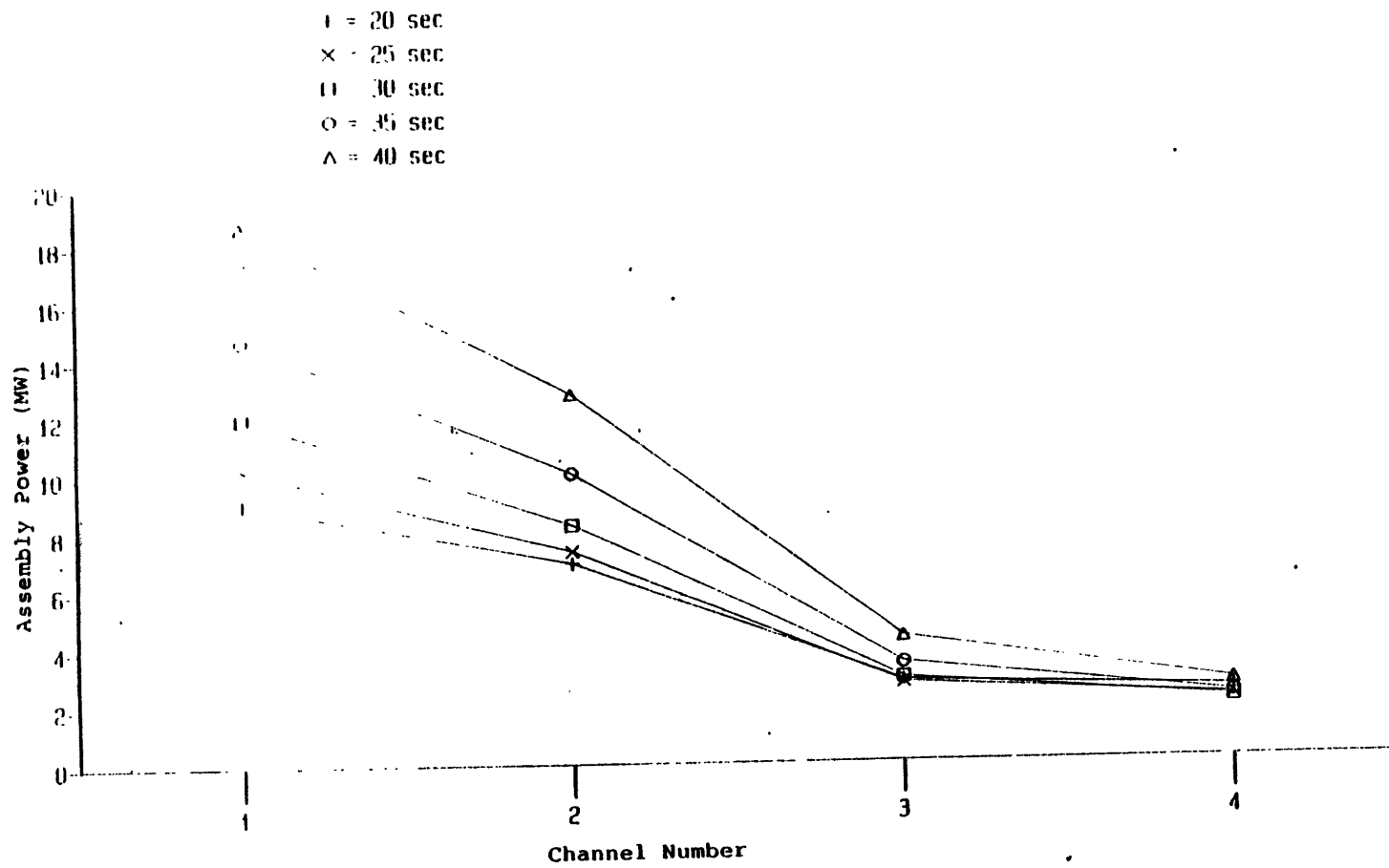
Figure 5.7 Radial assembly power distributions in the hot region, RTP12.
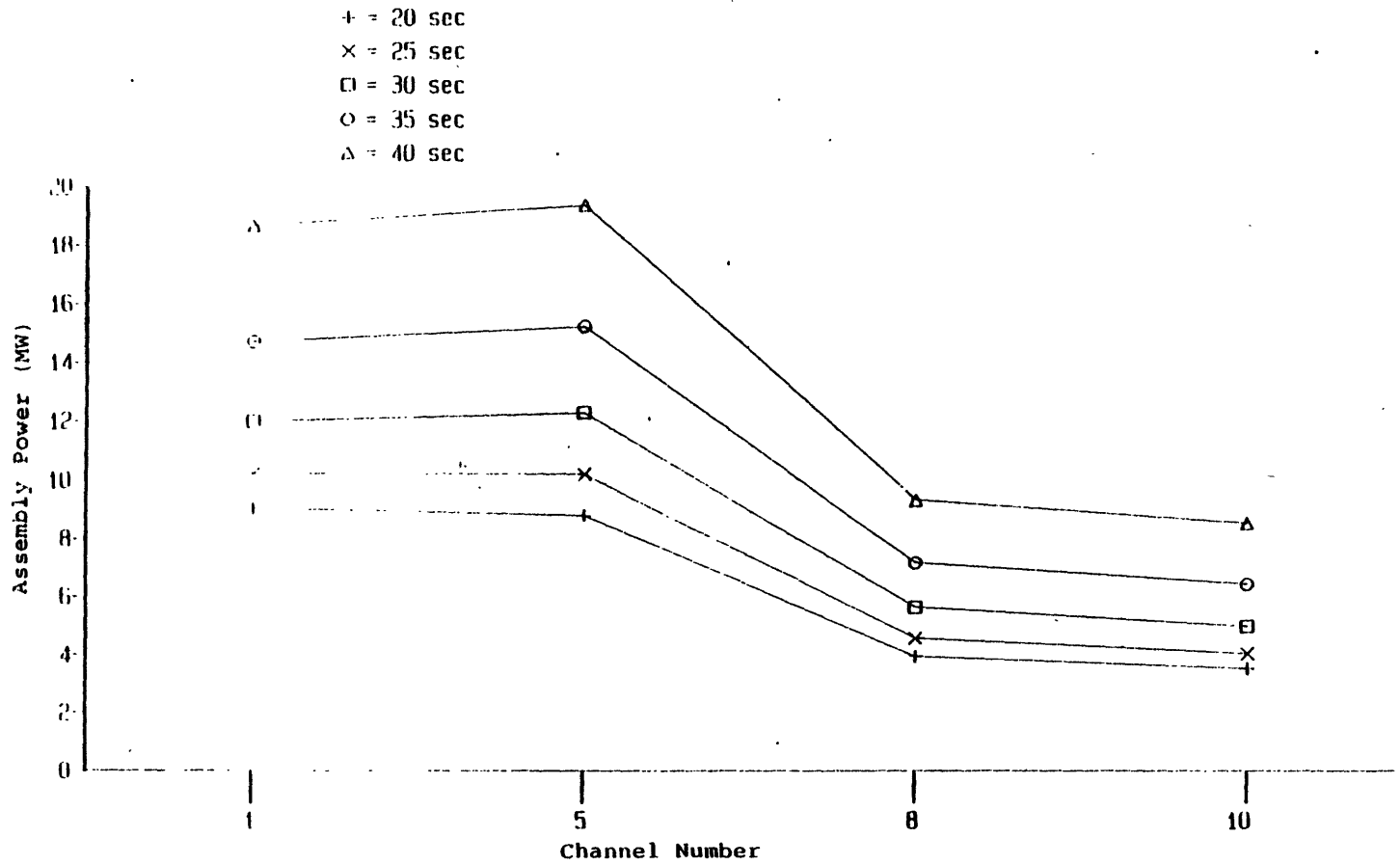
Figure 5.8 Radial assembly power distributions in the cold region, RTP12.

128
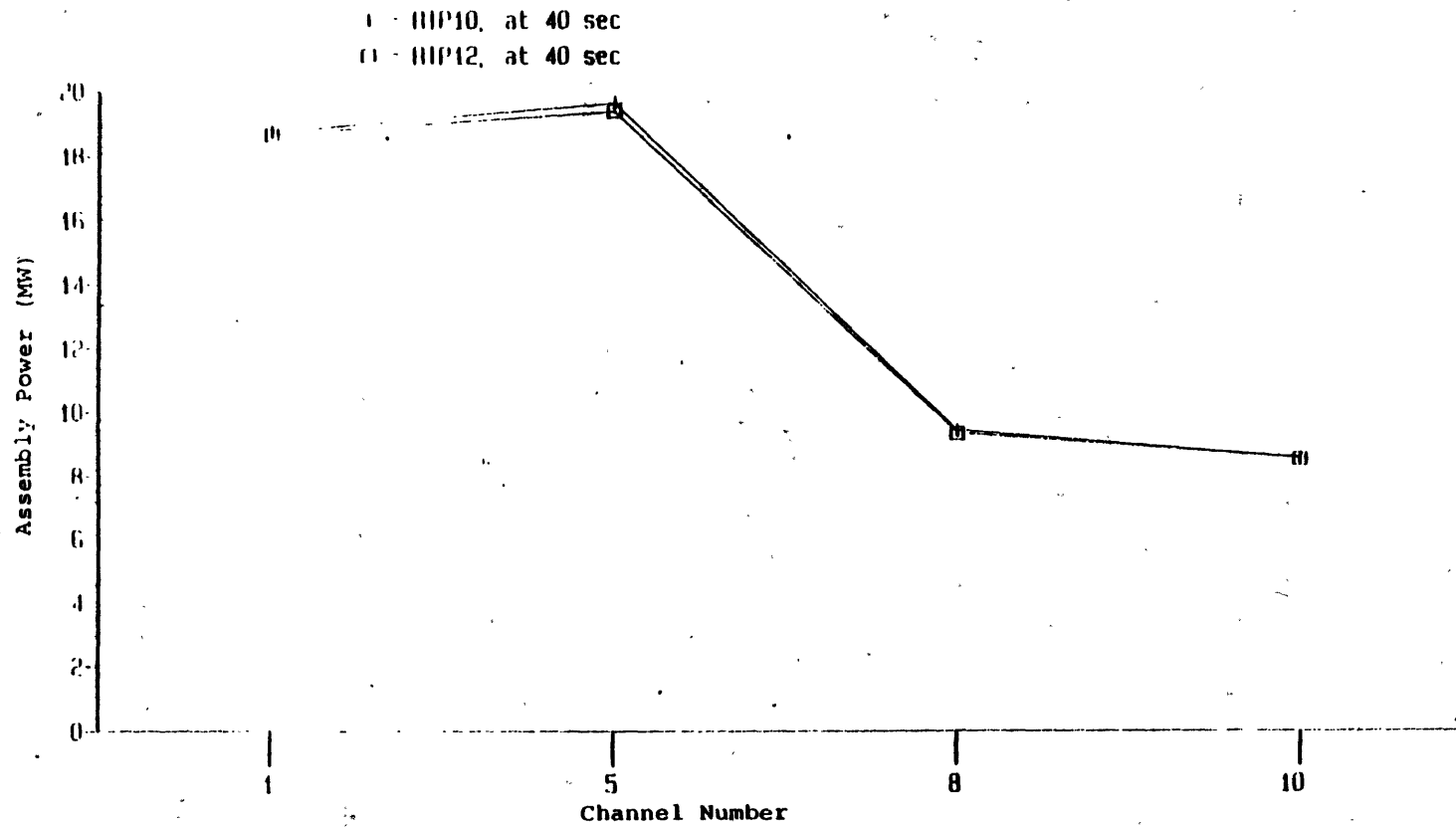
Figure 5.9   Radial assembly power distributions in the hot region, RTP10 and RTP12.

Figures 5.10 and 5.11 show the axial nodal power distributions of channel 5 for RTP10 and RTP12 and Figure 5.12 shows the comparision between them at 40 seconds. It is concluded from these results that the difference between the results of open- and closed-channel calculations is insignificant for both global and local parameters. This is consistent with the conclusion in Chapter 4 that there is no significant difference between the results of open- and closed-channel calculations before boiling occurs.

With respect to safety consideration for fuel rods, the return to power condition introduces a high fuel center line temperature problem, especially for those high power peaking channels. Figures 5.13 and 5.14 show the axial temperature distributions of channel 5 for RTP10 and RTP12. Again, the difference between the two results is insignificant as shown in Figure 5.15. Both of them indicate the peak center line temperature occurs about a quarter of the way along the fuel rod. At 30 seconds, the peak temperature is about 1800 $^\circ K$ which is above the lower limit of the grain-growth temperature of UO_2 (1673 $^\circ K$). At 40 seconds, the peak fuel center line temperature is higher than the upper limit of the grain-growth temperature of UO_2 (1973 $^\circ K$). This leads to the conclusion that the possible return to power condition of a steam line break accident (with high value of fission cross section coolant temperature feedback coefficient) may damage the fuel rod in the center of the fuel rod while the MDNBR is still well above the limitation. The most probable damage location will be in the cold region. It must be remembered however, that it was necessary to multiply the nominal value of the fission cross section coolant temperature coefficient by a factor of ten in order to achieve this effect.

The computational effort were 6277.93 and 7493.85 cpu seconds in a Honeywell 6180 computer for RTP10 and RTP12, respectively. About 16% cpu time can be saved by using closed-channel calculations.

5.4 Boron Solution Injection Simulations

One important response of the reactor safety system to the return to power condition is the boron solution injection. The boron-10 solution is injected into the core through an HPSI system. The HPSI pump starts when the SIAS (Safety Injection Actuation Signal) is on in response to the low RCS (Reactor Coolant System) pressure ($<$ 1622$psia$) [6]. The high neutron absorption cross section of boron-10 reduces the positive reactivity insertion due to the cooldown process during an SLB accident.

A boron concertration tracking model was incorporated in TITAN as described in Section 2.4. The boron-10 mass conservation equation is solved at the end of each time step to
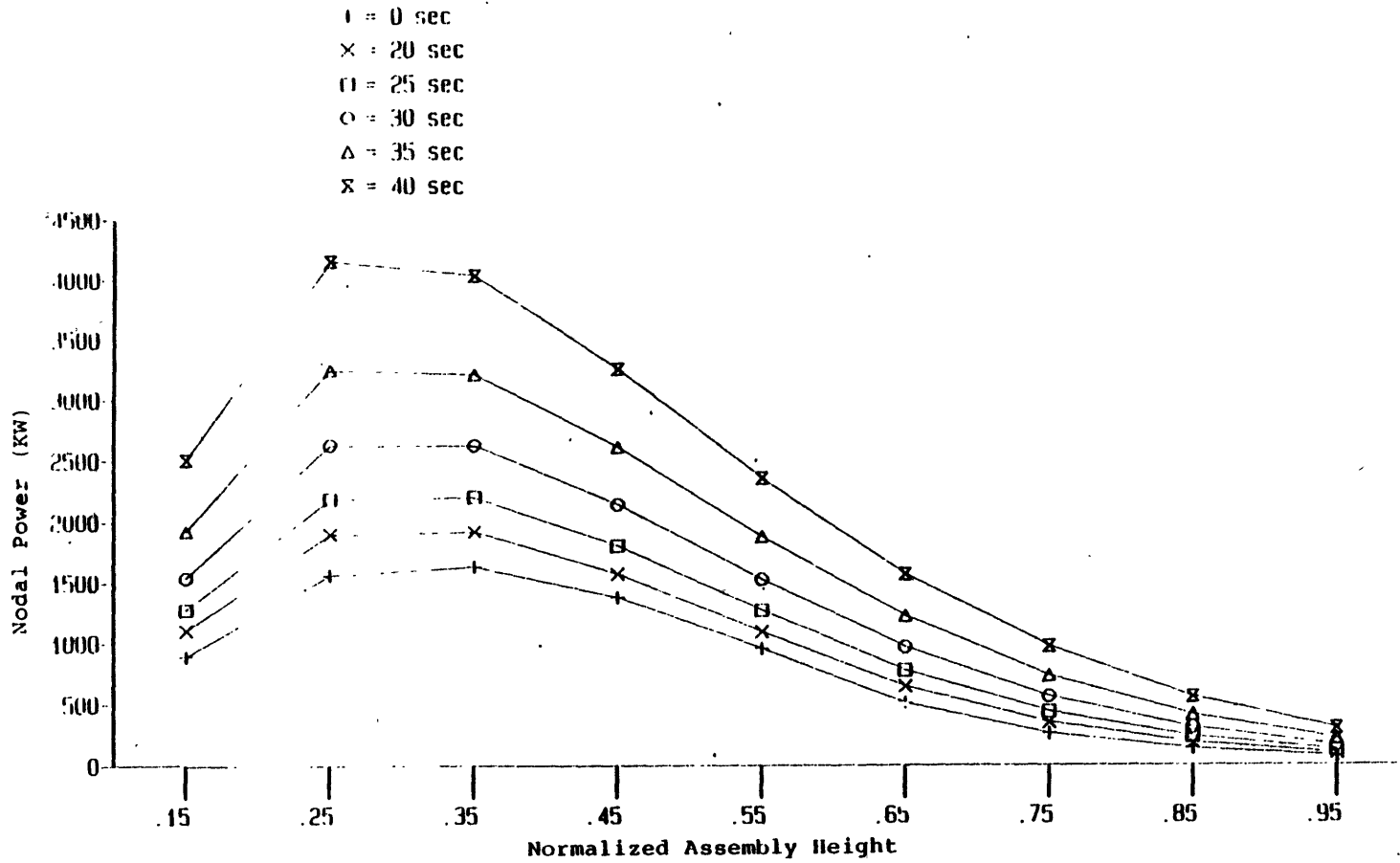
Figure 5.10 Axial nodal power distributions of channel 5, RTP10.
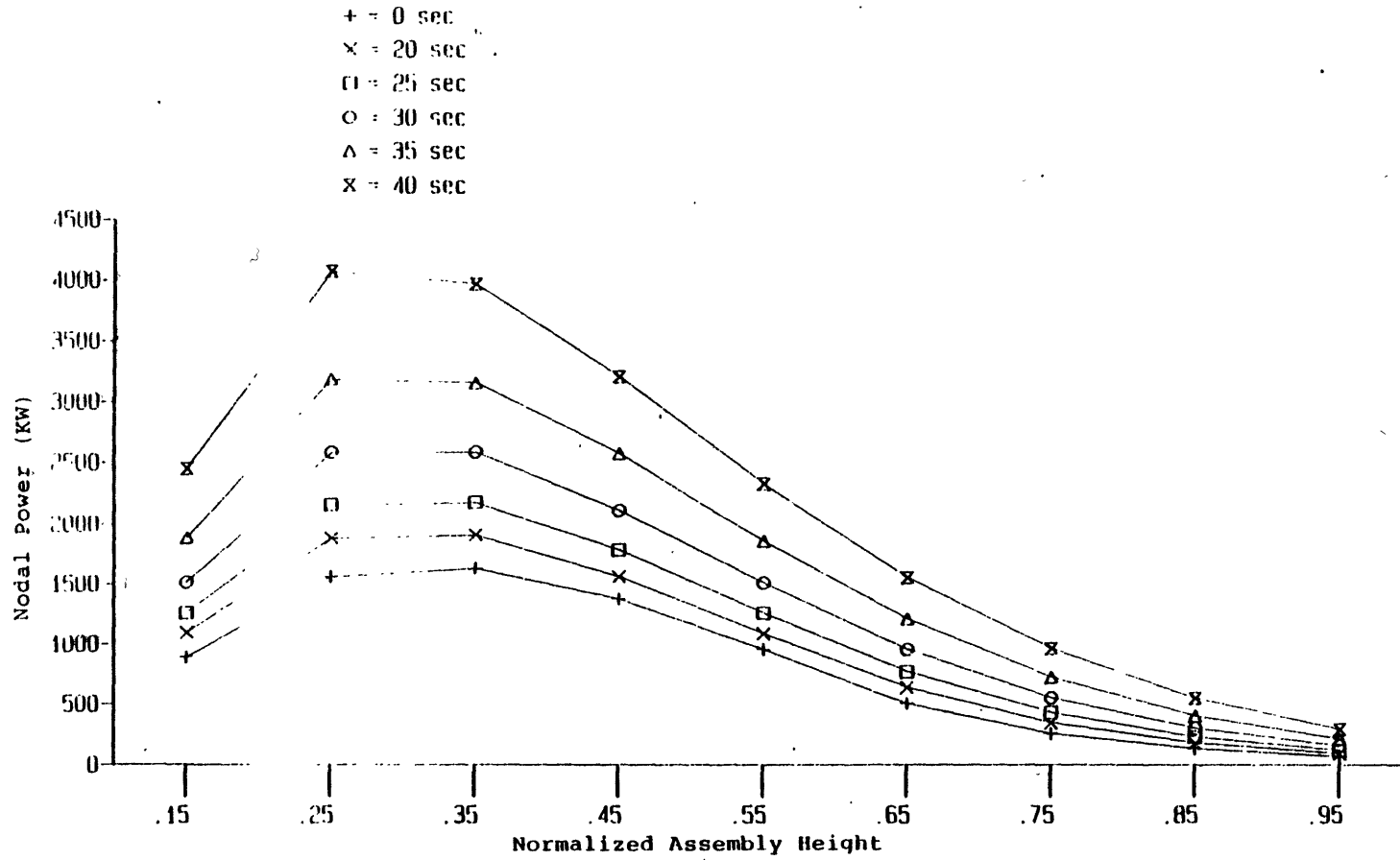
Figure 5.11 Axial nodal power distributions of channel 5, RTP12.

+ = RTP10, at 40 sec
⊓ = RTP12, at 40 sec
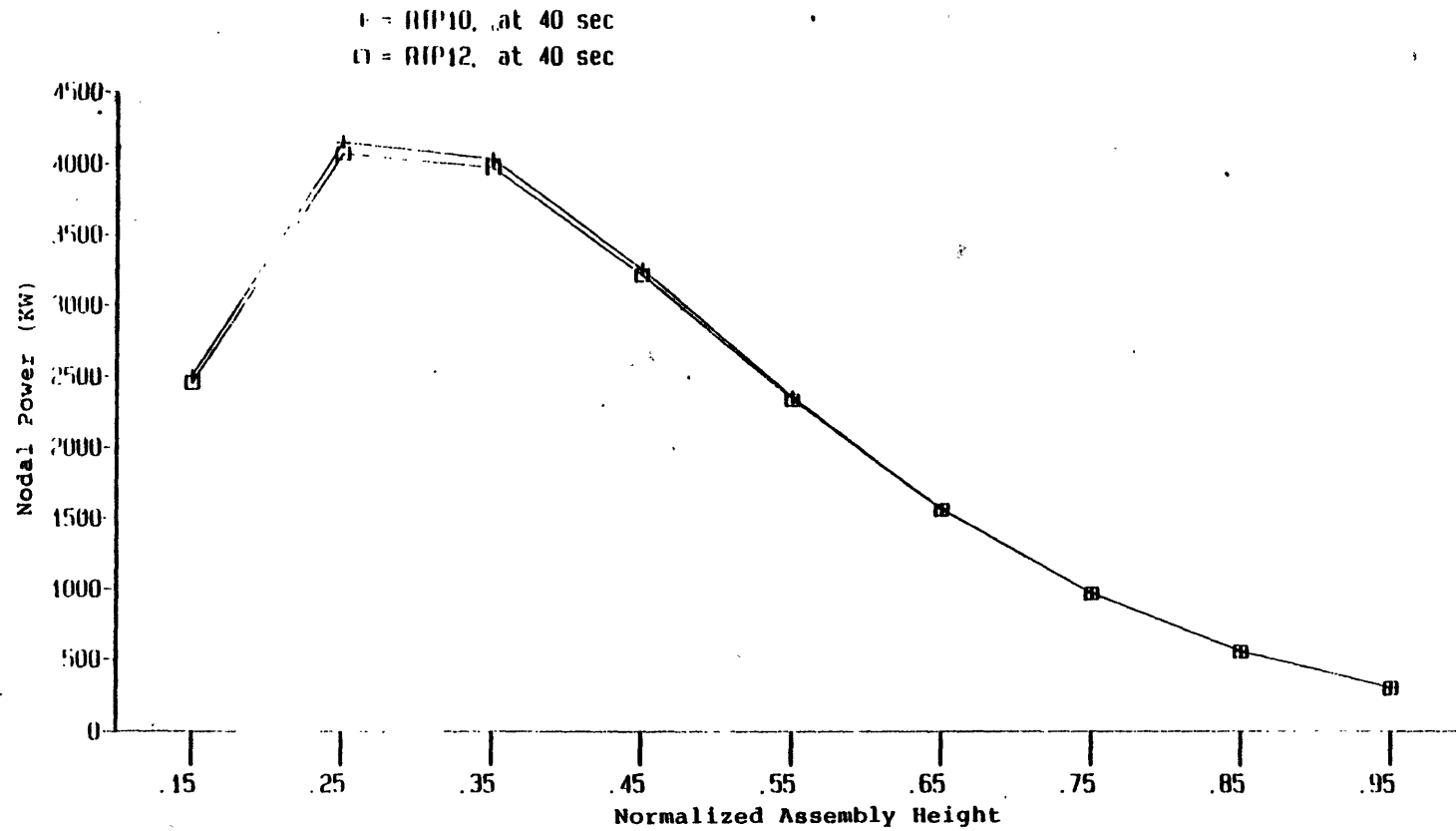
Figure 5.12 Axial nodal power distributions of channel 5, RTP10 and RTP12.
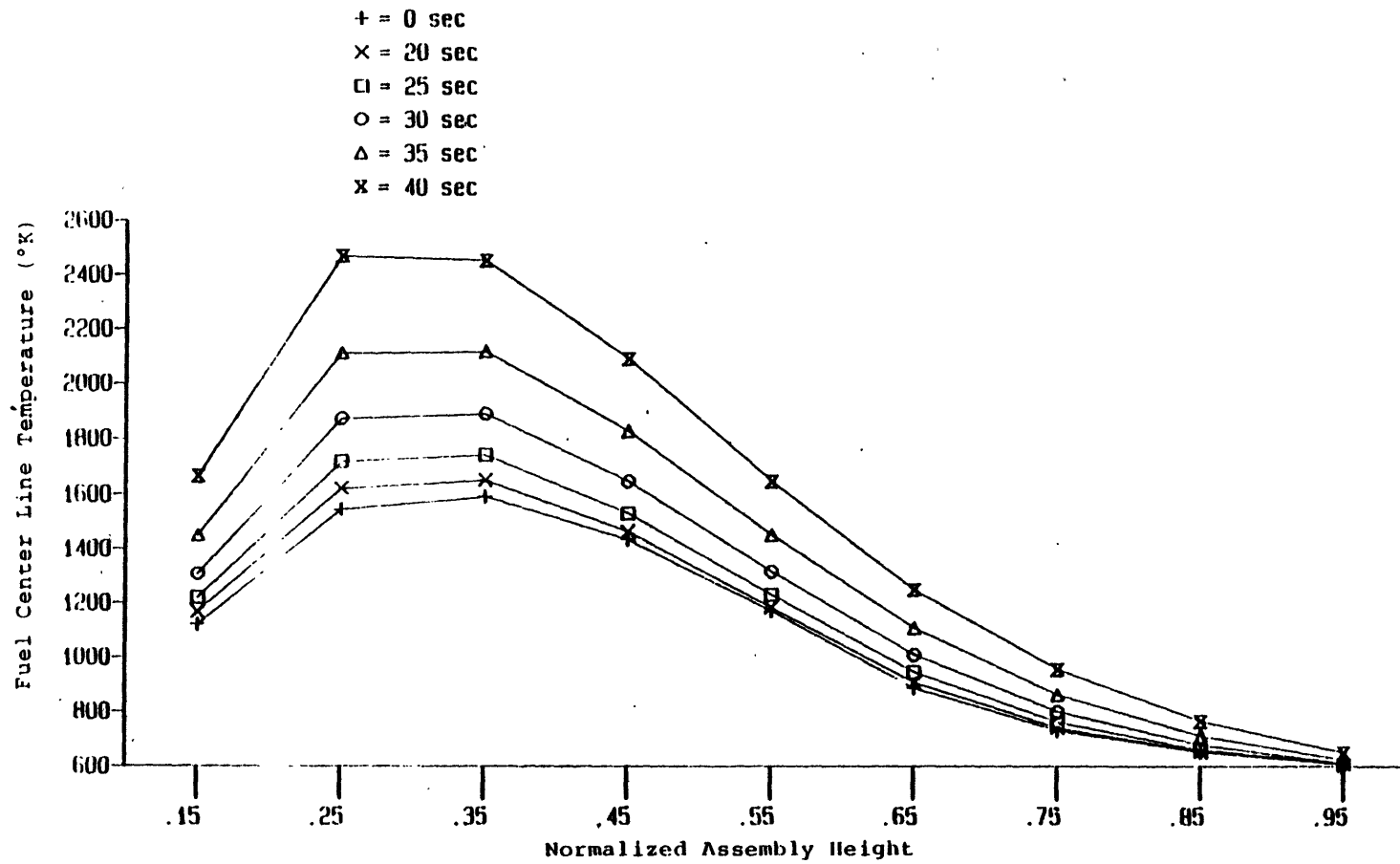
Figure 5.13 Axial fuel center line temperature distributions of channel 5, RTP10.
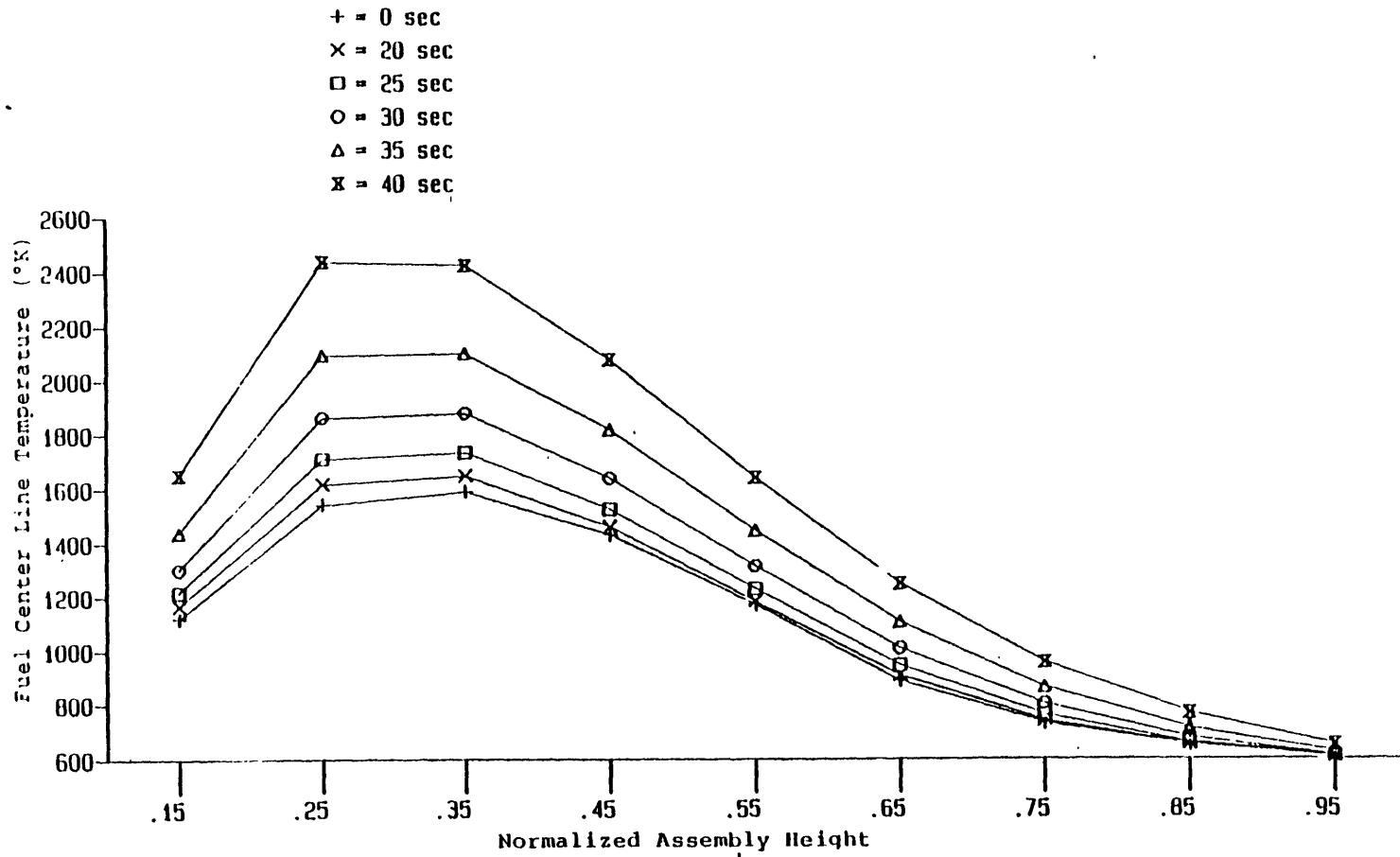
Figure 5.14 Axial fuel center line temperature distributions of channel 5, RTP12.
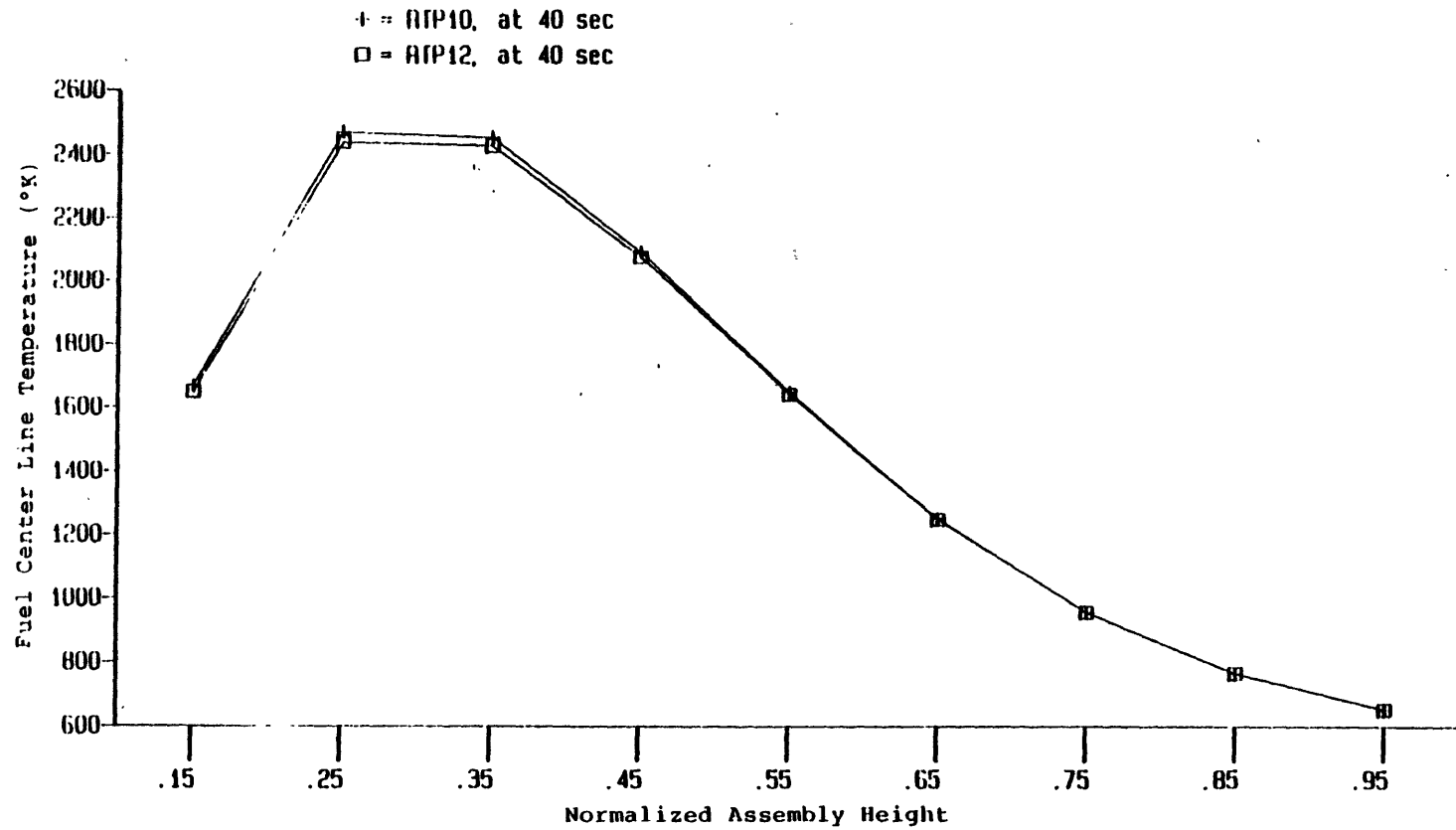
Figure 5.15 Axial fuel center line temperature of channel 5, RTP10 and RTP12.

obtain the boron-10 concentration of each node. This is valid because of the assumption that boron travels at the same velocity as the liquid coolant. After the nodal boron concentration is found, the macroscopic boron absorption cross section for a given energy group of the node can be obtained. The macroscopic absorption cross section is then added to the total absorption cross section at the current time step and the net absorption cross section is used for the next time step calculation.

Four calculations were performed. These included two closed-channel calculations and two open-channel calculations. The closed-channel calculations with 1720 *ppm* and 17200 *ppm* boron solutions are designated as RTP15 and RTP16, respectively. For open-channel calculations with the two boron solutions are designated as RTP17 and RTP18. All these calculations were restarted from the results of RTP10 at 20 seconds. It is obvious that 17200 *ppm* boron solution should have the stronger effect on the power history. This is shown in Figures 5.16 and 5.17. Figure 5.16 shows the power histories of the closed-channel calculations: no boron injection- RTP10; 1720 *ppm* boron injection- RTP15, and 17200 *ppm* boron injection- RTP16. Recall that RTP10 is the calculation with fission cross section coolant temperature feedback coefficient multiplied by a factor of 10. The strong positive reactivity added by this feedback coefficient continues the power excursion even with the injected boron solution. However, the effect of the boron solution is clearly seen by comparing the results of RTP15 and RTP16 (closed-channel calculations), or RTP17 and RTP18 (open-channel calculations). With 10 times higher boron concentration, both RTP16 and RTP18 calculations predict lower power histories than RTP15 and RTP17. The results verifies that the boron concentration tracking model is predicting the correct trend in power history.

Figure 5.18 shows the comparision between the results of closed- (RTP16) and open-channel (RTP18) calculations. The total power predictied by RTP16 is lower than that by RTP18. The coolant in the high power channels (such as channels 2 and 5) is driven out of the channel through the corss flow allowed in RTP18 (about -0.8 Kg/sec per node in channel 2 at 28 seconds). The boron-10 is then also driven out of the high power region and hence reduces the worth of boron-10. This is also shown in channel 2 axial nodal power distributions of the two calculations as shown in Figure 5.19. If boiling exists the cross flow will be much higher than with no boiling (refer to Chapter 4), and hence the difference induced by boron injection between the results of closed- and open-channel calculations may be significan.
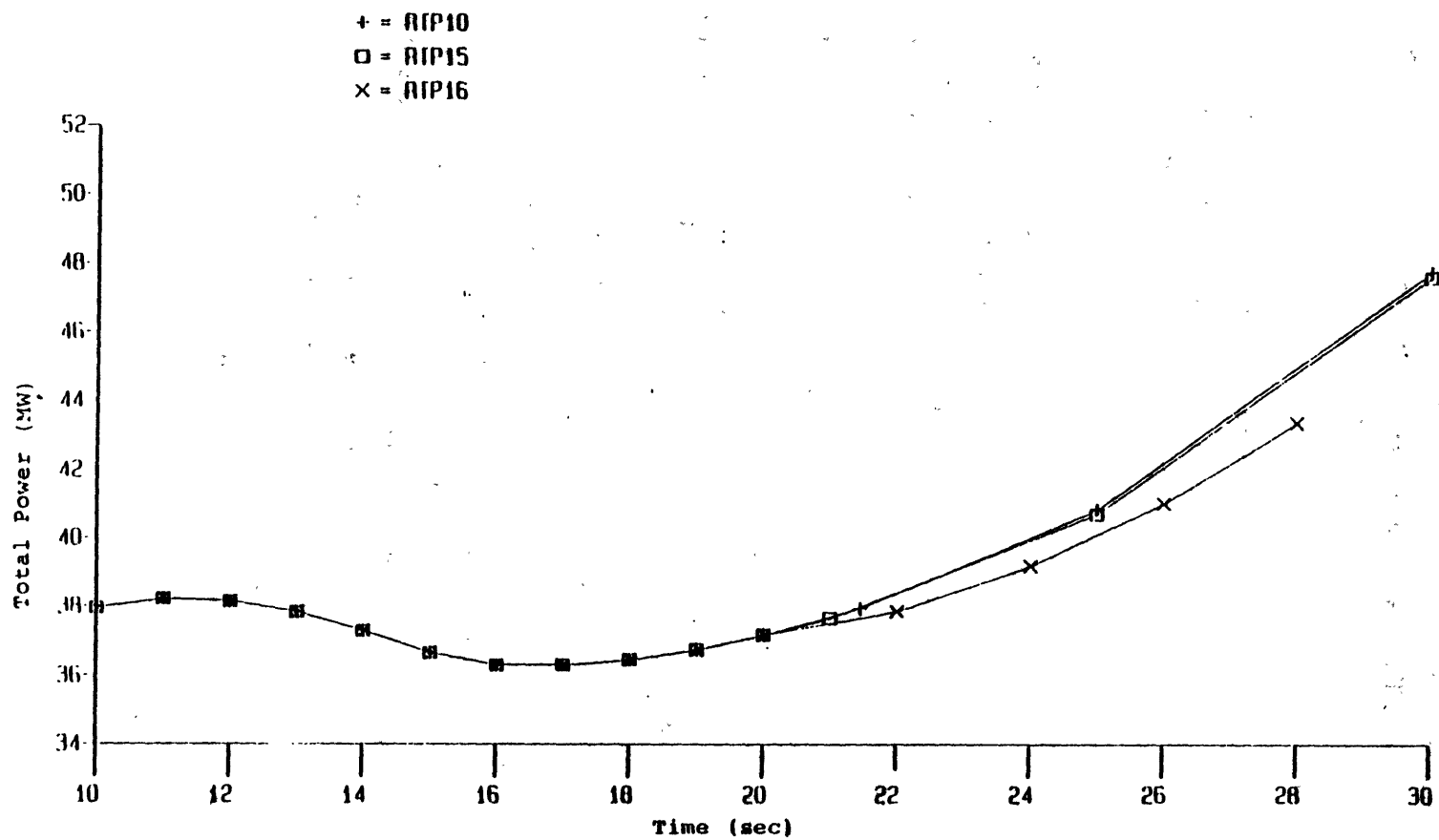
+ = RTP10
□ = RTP15
× = RTP16

Figure 5.16 Total power histories of RTP10, RTP15, and RTP16.
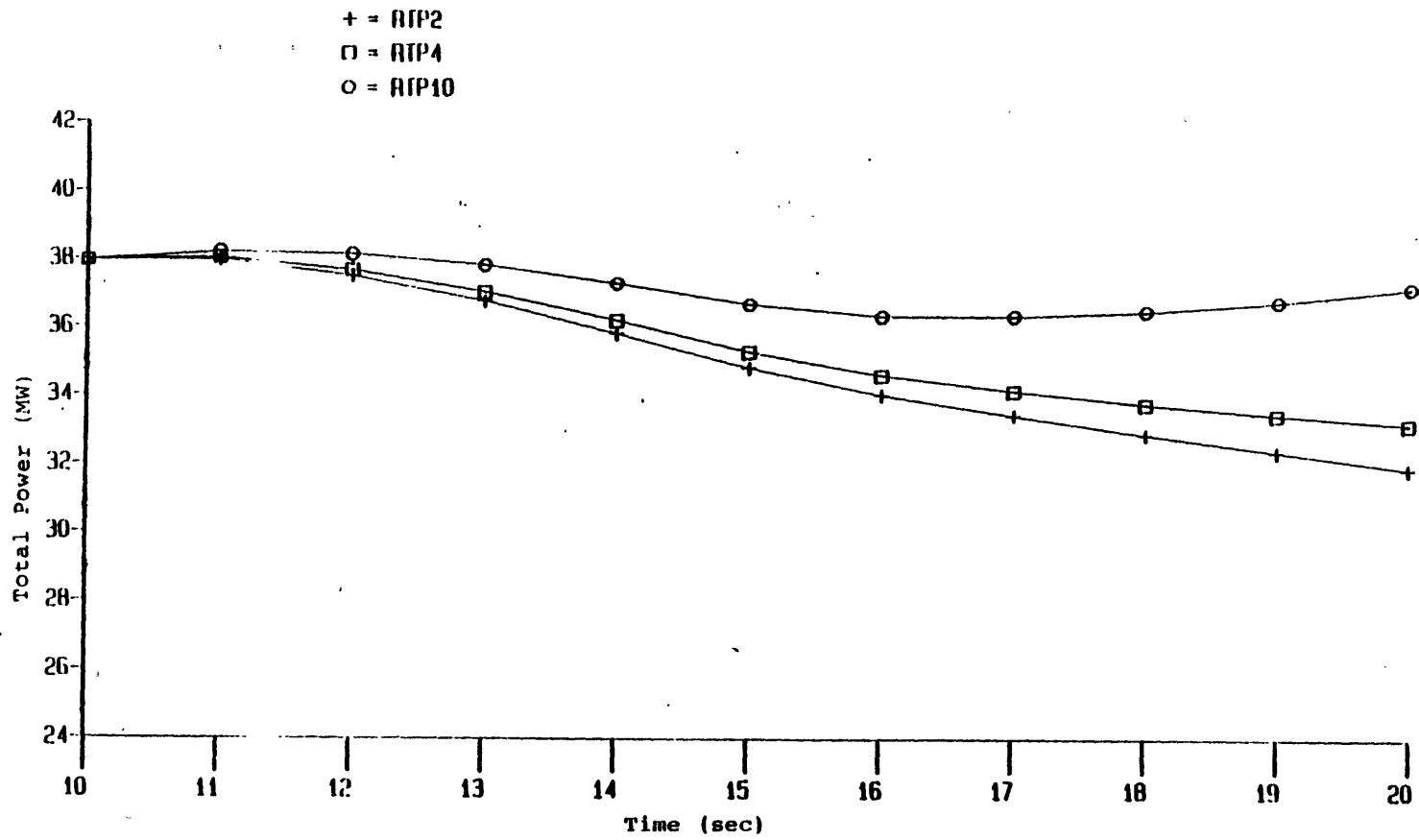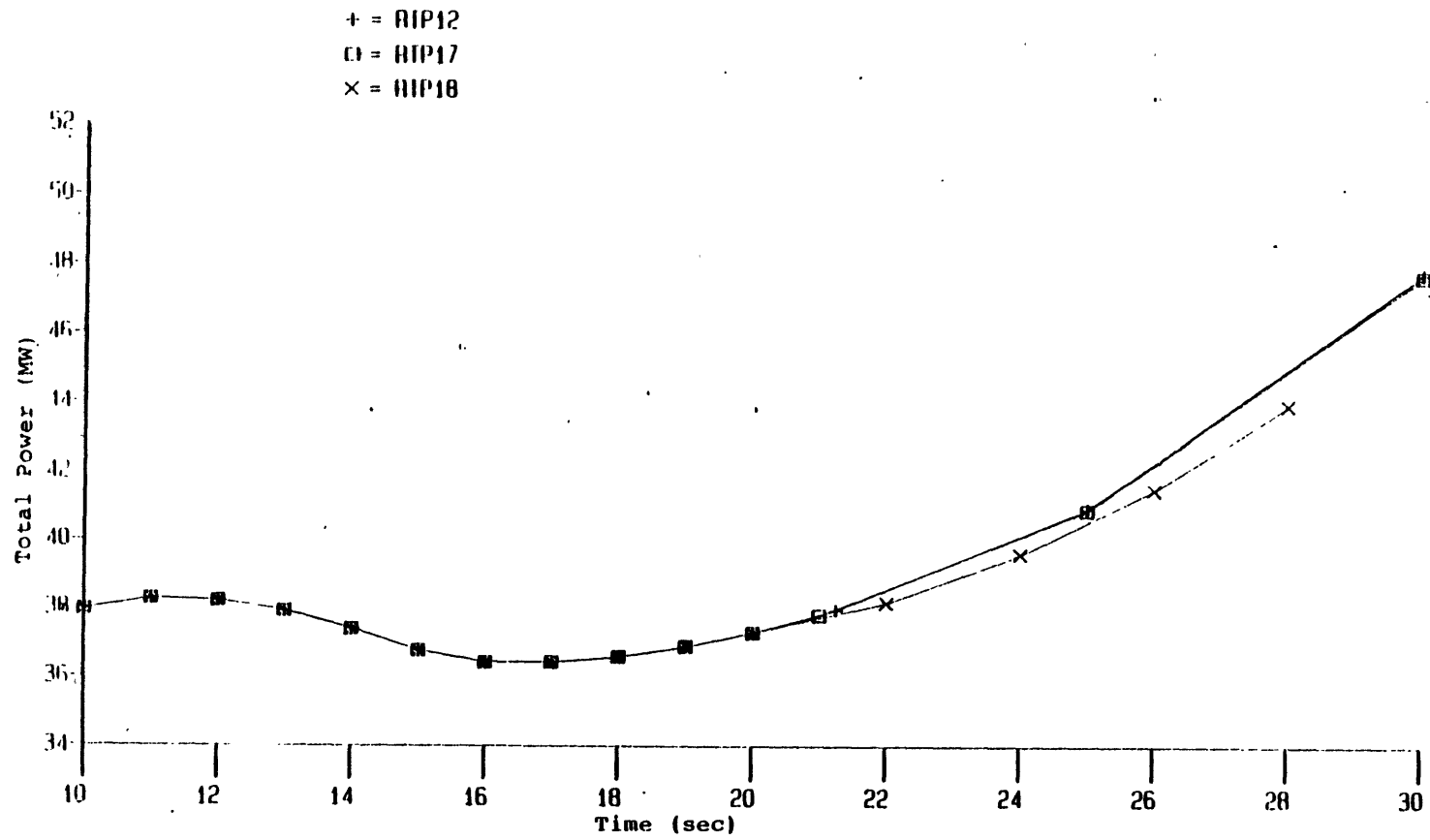
Figure 5.17 Total power histories of RTP12, RTP17, and RTP18.

Figure 5.18 Total power histories of RTP16 and RTP18.
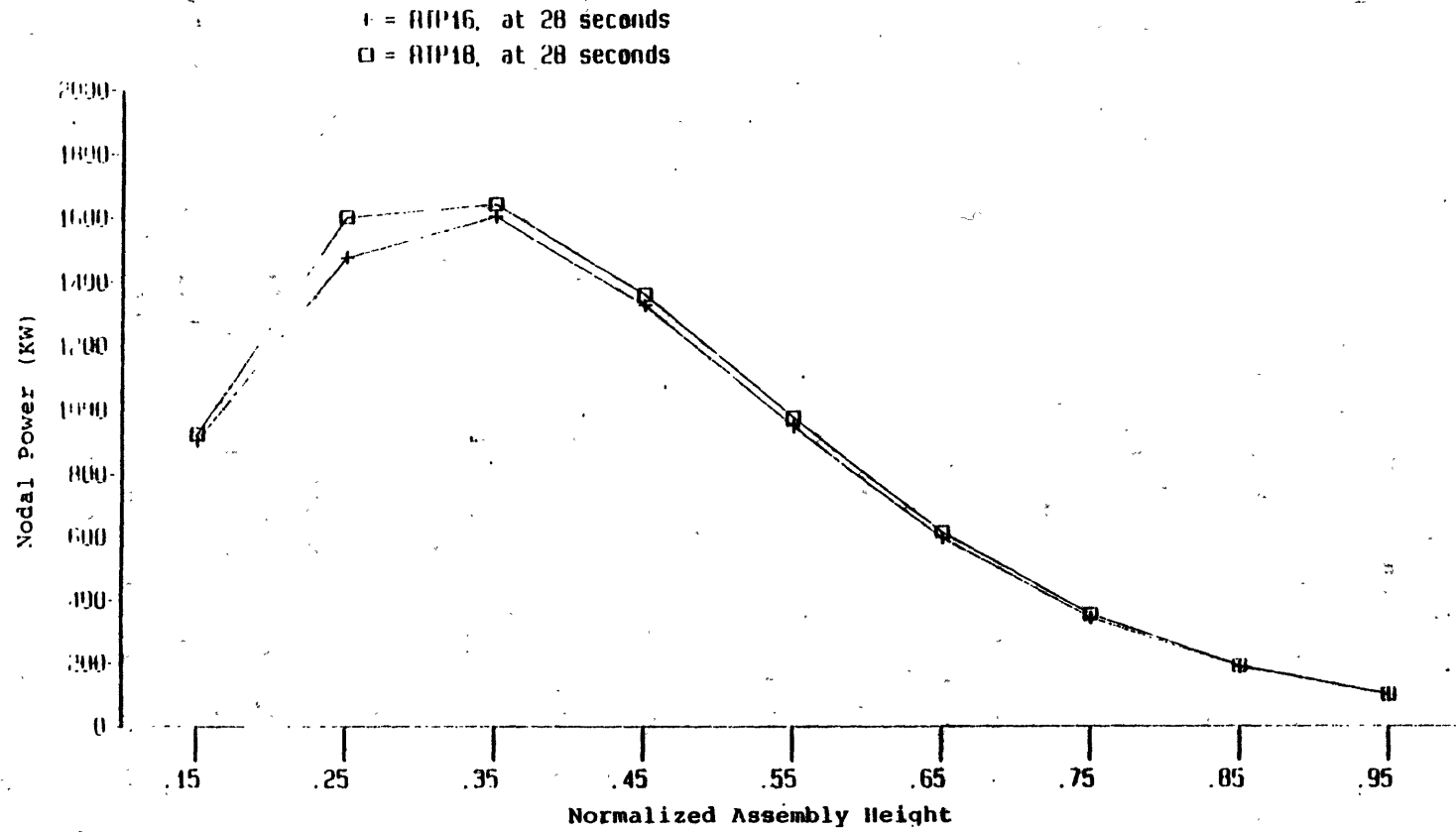
Figure 5.19   Axial nodal power distributions of channel 2, RTP16 and RTP18.

5.5 Summary

The steady state calculations with different coolant temperature feedback coefficients were performed to provide the starting points for return to power calculations. Althogh the global parameters (radial assembly power distributions) do not show singificant difference among the results (original coefficients, multiplied by 2, by 4, and by 10), the local axial nodal power distributions show that the peak power shifts to the bottom as the multiplier to the coefficient increases.

The return to power calculations were performed with a factor of 10 multiplied into the original fission cross section coolant temperature feedback coefficient. The return to power occurs around 22 seconds for both open- (RTP12) and closed- (RTP10) channel calculations and the power excursion continues. Both calculations were run to 40 seconds, and no boiling was found. No significant difference between the results of the two calculations for the single liquid phase regime was found. This is expected from the conclusion of the previous chapter that there is no significant difference between the open- and closed-channel calculations until boiling occurs. About 16% cpu time can be saved by using the closed-channel calculation in a Honeywell 6180 computer.

The fuel rod center line temperature predicted by TITAN shows that the hottest fuel rod is in the cold region and the center line temperature exceeds the design limit (grain growth temperature of UO_2, 1400 to 1700 $°C$) after 30 seconds. This leads to the conclusion that with high values of coolant temperature feedback coeffecients, MDNBR may not be a major concern but the high power rods in the cold region may suffer damage because of the high center line temperature. This is different from the results in Chapter 4 where the return to power never occurs, but the MDNBR in the hot region may result in fuel clad damage if the transient is continued.

Boron injection during the return to power studies was simulated with two boron concentration values, 1720 and 17200 *ppm*. The predictions by the boron concentration tracking model show the power reduction when boron injection is included in the calculations. The cross flow driven out of the high power channels leads to a higher power prediction for the open-channel calculation (RTP18) than for the closed-channel calculation (RTP16). This is because the boron worth is lower in the low power channels.

The calculations discussed in this chapter are summarized in Table 5.2.

Table 5.2

Summary of Calculations in Chapter 5

Steady State Calculations	
	Fission Cross Section Coolant Temperature Feedback Coefficient Multiplier
RTPSS2	2
RTPSS4	4
RTPSS10	10

Transient Calculations					
	Closed-Channel	Open-Channel	Boron Model	Boron 1720	Concen. ppm 17200
RTP10	X				
RTP12		X			
RTP15	X		X	X	
RTP16	X		X		X
RTP17		X	X	X	
RTP18		X	X		X

Chapter 6

SUMMARY AND RECOMMENDATIONS

6.1 Summary

A steam line break accident is one of the serious abnormal transients that might happen in a nuclear reactor. The large pressure difference between the secondary loop and the containment at the broken steam line accelerates the working fluid blow down. A much higher heat transfer rate between the primary loop and the secondary loop in the broken loop steam generator therefore results. The core inlet coolant temperature of the broken loop drops accordingly. The negative coolant, fuel temperature and positive coolant density feedback coefficients respond to the cooldown process by inserting positive reactivity into the core. A power excursion therefore results. To compensate for the positive reactivity, scram is an necessary safety response to the accident. However, there is concern that after the scram action has been taken, the continuous cooldown process may offset the negative reactivity insertion of the control rod and result in a net positive reactivity addition to the core. This is a hypothetical case that may be postulated if the main control rod was not inserted into the core. A boron solution safety injection along with the High Pressure Safety Injection (HPSI) system must then be included to compensate for the possible net positive reactivity addition.

The asymmetric characteristics of a steam line break accident (non-uniformly distributed inlet coolant temperature) rules out the possibility of applying a one-dimensional computer code for such transients. In addition, from the design point of view, it is important to define the detailed local information during the transient. The three-dimensional effects of a steam line break accident are therefore of interest.

Since both neutronics and thermal-hydraulics are involved in a steam line break accident, a coupled (neutronics and thermal-hydraulics) code must be used for the analysis. The TITAN code can be employed for such analyses. In this code the neutronics (original by QUANDRY) and thermal-hydraulics (original by THERMIT) models are coupled through a feedback model in the code. Both the neutronics and thermal-hydraulics models are three-dimensional. In addition, the thermal-hydraulics involves describing the liquid and vapor as separate fluids which results in a ten-equation model. The nodal solution method in the neutronics calculation has the advantage of using a large mesh size without losing accuracy. Several modifications had to be made before the original version of TITAN could be applied to a steam line break

accident. These included the following additions:

1. Multi-zone inlet coolant temperature forcing functions capability;
2. Total inlet coolant flow rate boundary condition option;
3. Total inlet coolant flow rate transient simulation capability;
4. Boron concentration tracking model and boron reactivity feedback model;
5. Flow/coolant temperature transient plus control rod transient option.

All the above modifications were made and tested. In addition, a transient restart capability was provided so that simulation conditions could be changed during a transient calculation for various purposes. The time step size of the neutronics and thermal-hydraulics calculations were decoupled so that the neutronics time step size could be an interger multiple of the thermal-hydraulics time step size. This permits taking advantage of the fact that fully implicit numerical scheme is used for solving the neutronics equations.

Four inlet coolant temperature transient sensitivity studies were performed [21]. These included:

1. An open-channel, uniform inlet coolant temperature case;
2. A closed-channel, uniform inlet coolant temperature case;
3. An open-channel, non-uniform inlet coolant temperature case;
4. A closed-channel, non-uniform inlet coolant temperature case.

The conclusions of the above studies are:

1. Three-dimensional codes are necessary to simulate a non-uniform inlet coolant temperature distribution transient.
2. Before boiling occurs, there is no singificant difference between results of closed- and open-channel calculations.

Before performing a transient calculation, a steady state condition must be obtained. Two PWR 10-channel models simulating quarter core geometry were used for the study. Model A had three control rods. One control rod was fully inserted in the center channel while two other rods were 45% inserted in channels 3 and 8 (see Figure 2.3). Model B was the same as Model A except no control rod was inserted in the center channel. Model A was used for the test calculations mentioned above and Model B was used to simulate a stuck control rod (required by NRC regulation for a steam line break accident simulation). Both steady state results of the two models showed that there is no significant difference between the closed- and open-channel calculations. A one-dimensional, fully-implicit numerical scheme was therefore incorporated to TITAN as an option for faster steady state calculations. Less than half of the cpu usage is required by the one-dimensional, fully-implicit numerical scheme with marching solution method than that in the semi-implicit scheme used by the original version of TITAN.

Three-dimensional effects concerning MDNBR in a steam line break accident were first studied. None of the studies involved a return to power condition. The conclusions from these studies are as follows:

1. Before boiling occurs, there is no significant difference between the total power results of a 3-D (3D), a 1-D non-uniform inlet coolant temperature distribution (1DN), and a 1-D, uniform inlet coolant temperature distribution (1DU) calculations.

2. After boiling occurs, the global parameters (total powers) are somewhat affected, but not significantly, by using either one of the three approaches mentioned above.

3. Continuously increasing differences between the assembly powers – and more particularly the nodal powers – predicted by the 3D and 1DN calculations were found after boiling occurs.

4. The assembly and nodal powers calculated by the 1DU approach lie in between the hot region and cold region results of the 3D calculation.

5. The void fractions predicted by a 3D calculation are lower than those predicted by a 1DU calculation, and a 1DN calculation.

6. The 3D calculations predict more conservative MDNBR than the 1DN calculations during a void generation process since the liquid cross flow is driven out of the hot node.

7. In a void condensation process, the MDNBR predicted by the 1DN calculation is more conservative, since no cold coolant is allowed to cross channel boundaries.

8. The starting point of boiling predicted by a 1DU calculation will be much later than that predicted by a 3D or a 1DN calculaiton.

9. A non-equilibrium interphase mass exchange model is better suited for a steam line break accident analysis.

To reach a return to power condition, Model B was modified by multiplying the fission cross section coolant temperatrue coefficient $\frac{\partial \Sigma_f}{\partial T_c}$ by a factor of 10. This is equivalent to changing the core material. A steady state calculation is therefore necessary to provide a starting point for the transient calculation. The new steady state results show that the axial power peak shifts towards the bottom as the multiplier to the feedback coefficient increases. This is expected, since the stronger coolant temperature feedback effect causes an earlier bottom peak.

Four return to power calculations predicted high power peaking in the cold inlet coolant region and the fuel center line temperature exceeded the design limits (grain-growth temperature of UO_2). No boiling was found in either the hot or cold regions, hence the MDNBR is well above the limitation. It is concluded that should a return to power occurs, the fuel may suffer damage because of the high power peaking in the cold region. If there is no return to

power, fuel clad may be in danger of damage in the hot region because of the possible low MDNBR. However, one should note that a steam line break accident is a very complicated transient. The above conclusions are valid for the two postulated extreme conditions.

The difference between the results of closed- and open-channel return to power transients is insignificant. This is consistent with the conclusion drawn from MDNBR studies: there is no significant difference between the results of closed- and open-channel calculations before boiling occurs.

The boron injection simulation in the return to power studies was performed by using two concentration values: 1720 and 17200 *ppm*. The predictions by the boron concentration tracking model show the expected power reduction when boron injection is involved. The cross flow driven out of the high power channels leads to a higher power prediction for the open-channel calculation than that for the closed-channel calculation. This is because the boron worth is lower in the low power channels.

Two additional calculations were performed to study the effect of time step size on decoupling for thermal-hydraulics and neutronics calculations. The conclusions of the studies are:

1. The decoupling logic in the present version of TITAN is good for thermal-hydraulic dominant transients, since the time step size of neutronics calculation is an interger multiple of that of thermal-hydraulics calculation.

2. The results predicted by using $\Delta t_N = 4\Delta t_{TH}$ agrees with the results using $\Delta t_N = \Delta t_{TH}$ very well for 1DN calculations.

3. Up to 34% cpu time can be saved in a Honeywell 6180 computer by using the decoupling relationship.

6.2 Recommendations

Mesh size decoupling between the neutronics and thermal-hydraulics calculations should be made an option in TITAN. Because of time and money restrictions, there was no real size quarter core model employed for the studies. One way to approach the real model is to lump the thermal-hydraulic nodes to reduce the size of the model and thus save major computational effort of the calculations. The neutronics nodes should not be lumped because of the sharp variation of the power shape around the control rod region. It is the opinion of the author that until this work is done, a real core model simulation will be impractical with TITAN.

Another recommendation for furture work is to perform a steam line break accident

147

simulation without enhancing the boiling or the feedback coefficient. Two extreme cases were presented in this dissertation. However, it is of interest to know the three-dimensional effects under normal conditions.

A steam line break accident with low core flow rate (corresponding to a reactor coolant pump trip) should be studied. Significant boiling is expected under such conditions. The three-dimensional effects of the transient are therefore of interest to know.

REFERENCES

1. D. P. Griggs, M. S. Kazimi, and A. F. Henry, "TITAN: An Advanced 3-D Neutronics/Thermal Hydraulics Code for Light Water Reactor Safety Analysis," NUREG/ CP-0034, Vol. 2, Proc. of Topical Mtg. on Advances in Reactor Physics and Core Thermal Hydraulics, Kiamesha Lake, NY, pp. 766-780 (Sept. 1982).

2. D. P. Griggs, M. S. Kazimi, and A. F. Henry, "Advanced Methods Development for LWR Transient Analysis, Final Report: 1981-1982," MIT-EL-82-021 (1982).

3. D. P. Griggs, M. S. Kazimi, and A. F. Henry. "TITAN: An Advanced Three Dimensional Coupled Neutronic/Thermal-Hydraulics Code for Light Water Nuclear Reactor Core Analysis," MIT-EL-84-001 (1984).

4. C. K. Tsai, D. P. Griggs, M. S. Kazimi, and A. F. Henry, "Development and Quarter Core PWR Rod Ejection Accident Application of the TITAN Code, Final Report: 1982-1983," MIT-EL-83-007 (1983).

5. Main Yankee Final Safety Analysis Report.

6. Yankee Atomic Cycle 6 MSLB Analysis, Attachment A.

7. Code of Federal Regulations, Section 10, Part 50, Appendix A, "General Design Criteria for Nuclear Power Plants".

8. J. Yasinsky, "Notes on Nuclear Reactor Kinetics," WAPD-TM-960, Bettis Atomic Power Laboratory (1970).

9. V. H. Ransom, et. al., "RELAP5/MOD1 Code Manual Vols. 1 and 2," NUREG/ CR-1826 (EGG-2070 Rev. 2 Draft), September 1982.

10. J. McFadden et. al., "RETRAN-02 - A Program for Transient Thermal-Hydraulic Analysis of Complex Fluid Flow Systems," EPRI-NP-1850, Electric Power Research Institute (1981).

11. W. Frisch, S. Langenbuch, P. Peternell, "The Significance of Fast Moderator Feedback Effects in a BWR During Severe Pressure Transients," Nuclear Science and Engineering, 64, 843-848 (1977).

12. G. B. Peeler, D. L. Caraher, and J. Guttmann, "Calculation of the Limiting CESSAR Steam Line Break Transients," Anticipated and Abnormal Plant Transients in LWR, Vol. 1, edt. P. L. Lassahn, D. Majumdar, and G. F. Brockett, Plenum Press, New York and London, pp. 539-553 (1984).

13. CESSAR FSAR, C-E Power Systems.

14. CESSAR FSAR, Amendment No. 7, March 31, 1982, C-E Power Systems.

15. J. R. White, R. T. Jensen, V. Chexal, and J. Lang, "Evaluation of RELAP5 MOD1 for Analysis of Steam Generator Tube Rupture Transients in OTSG Plants," Anticipated and Abnormal Plant Transients in LWR, Vol. 1, edt. P. L. Lassahn, D. Majumdar, and G. F. Brockett, Plenum Press, New York and London, pp. 565-574 (1984).

16. M. A. Bolander, C. D. Fletcher, D. M. Ogden, B. D. Stitt and M. E. Waterman, "RELAP5 Analyses of Overcooling Transients in A Pressurized Water Reactor," Anticipated and

Abnormal Plant Transients in LWR, Vol. 1, edt. P. L. Lassahn D. Majumdar, and G. F. Brockett, Plenum Press, New York and London, pp. 603-615 (1984).

17. J. Woodçock, "Inter-Assembly Crossflow Effects in PWR Cores During A Secondary Steamline Rupture," ASME 81-WA-HT-61.

18. T. W. T. Burnett, et. al., "LOFTRAN Code Description," WCAP-7907, June 1972.

19. S. Altomare and R. F. Barry, "The TURTLE 24.0 Diffusion Depletion Code," WCAP-7758, 1971.

20. H. Chelemer et. al., "THINC IV, An Improved Program for Thermal Hydraulic Analysis of Rod Bundle Cores," WCAP-7956, June 1973.

21. C. K. Tsai, "Three Dimensional Effects in Analysis of PWR Steam Line Break Accident," Ph.D. Thesis in the Dept. of Nuclear, M. I. T. Feb. 1985.

22. A. T. Schor and N. E. Todreas, "A Four-Equation Two Phase Flow Model for Sodium Boiling Simulation of LMFBR Fuel Assemblies," MIT-EL-82-038 (1982).

23. W. Wulff, H. S. Cheng, D. J. Diamond and M. Khatib-Rahbar, "A Description and Assessment of RAMONA-3B MOD.0 Cycle 4: A Computer Code with Three-Dimensional Neutron Kinetics for BWR System Transients," NUREG/CR-3664, BNL-NUREG-51746 Jan. 1984.

24. H. S. Cheng, A. L. Aronson, J. W. Herczeg, and D. J. Diamond, "The Use of MEKIN-B for Light Water Reactor Transient Calculations," BNL-NUREG-23785.

25. D. A. Dube, "Development of A Fully Implicit Two-Fluid, Thermal-hydraulic Model for Boiling Water Reactor Transient Analysis," Ph.D. Thesis at M. I. T. Dept. of Nuclear Engineering, 1980.

26. W. C. Rivard and M. D. Torrey, "Numerical Calculation of Flashing from Long Pipes Using a Two-Field Model," Los Alamos Scientific Lab. LA-6140-MS, Nov. 1975.

27. P. Saha, B. S. Shiralkar, and G. E. Dix, "A Post-Dryout Heat Transfer Model Based on Actual Vapor Generation Rate in Dispersed Droplet Regime," ASME, 77-HT-80.

28. D. H. Gurinsky, "Nuclear Fuels," The Technology of Nuclear Reactor Safety, Vol. 2, Reactor Materials and Engineering, edited by J. J. Thompson and J. G. Beckerly, The MIT Press, 1973.

Appendix A
TITAN USER'S GUIDE

1 Introduction

In this Appendix, the TITAN code input information is given. The basic structure of the input format is based on the formats of THERMIT code and the QUANDRY code.

The description of the required input variables is presented in the following sections.

2 Detailed Input Description

2.1 Introduction

Three types of input formats are used in the code. The first is that associated with the standard FORTRAN READ statement. Both format-free and fixed format type variables are used. The format-free input is referred to as *-format consistent with IBM FORTRAN. All integer and real non-array variables are input via the format-free option. Only the title card is input in fixed character format.

The second type of input format is that associated with the standard FORTRAN Namelist option. This option is part of the restart feature and allows the user to change selectively the value of any of a variety of variables. The details of the Namelist option can be found in FORTRAN reference manuals and only an example will be given here. If the variable 'iflash' is to be changed from 1 to 2 during a restart, then the input statement would be

$$\$restart\ iflash = 2\$$$

This statement would set iflash equal to 2 while not affecting any other variable. Of course, if other variables are to be changed, they also can be included in the Namelist statement. As indicated above, this type of input format is only used for the restart option.

The third type of input format is that associated with the input processor found in subroutine 'nips'. This subroutine is used to read the array data. The input processor

151

permits relatively easy input of the values for the arrays. The key to this processor is that blocks of data may be repeatedly read. To achieve this result, a special type of format is used. Input fields are separated by blanks (no commas are allowed) with repeated fields inside parentheses proceeded by an integer multiplier. The end of a card group is marked with a dollar sign ($). An example serves to illustrate the use of this format. Suppose the array P(6,4) (6 levels, 4 channels) must be read in. There are 24 total values which are required. If these values are all the same (e.g., 6.9MPa), then the input would be

$$24(6.9e6)\$P$$

(Everything after the $-sign is ignored so that comments can be placed here). If the four channels all have the same distribution, but not axially uniform, then the input would be

$$4(6.9e6\ 6.85e6\ 6.8e6\ 6.75e6\ 6.7e6\ 6.65e6)\$P$$

It should be noted that the values for the variables can be given in any format, but will be interpreted according to the variable type. Up to 10 levels of parentheses nesting are permitted. Also no blank may appear between a left parentheses and the integer proceeding it. With this type of format the array data can be specified with a minimum amount of input.

2.2 General Problem Information, Real and Integer Constants

The first group of input contains the general information of the case we are dealing with. There are thirteen cards in this group. The variables in each card are described in the following sections. Note here, all inputs for TITAN are in the free format except for the title information card (card two), which is in the A format.

2.2.1 Card One
  (i) Variable: ntc
  (ii) Meaning: Two meanings

  - The number of title cards.

  - An input flag indicating whether the job is a restart or a new problem.
  (iii) Description:

- ntc>0, a new problem is started and ntc is the actual number of title cards to be read in card 2.

- ntc=0 the execution is terminated.

- ntc=−2, the job is restarted from a previously created dump file. This is a steady state restart. (See 3.2.1)

- ntc=−3, the job is restarted from a previously created dump file. This is a transient restart. (See 3.2.2)

2.2.2 Card Two

   (i) Variable: Title information

  (ii) Description: The number of cards which are read in is equal to ntc. On each card 80 characters of information may be given.

2.2.3 Card Three

   (i) Variables: nc, nr, nz, ncf, ncc, nopt, noppt

  (ii) Description:

  nc= Number of channels.

  nr= Number of rows.

  nz= Number of axial nodes.

  ncf= Number of nodes in the fuel.

  ncc= Number of nodes in the clad.

 nopt= Number of channels to be printed out.

noppt= Indicator of print out (0/1) (print all channels/optional print out)

  (iii) Example: Refer to Figures A.1 and A.2.

        nc=6

        nr=3

        nz=4

        ncf=4

        ncc=2
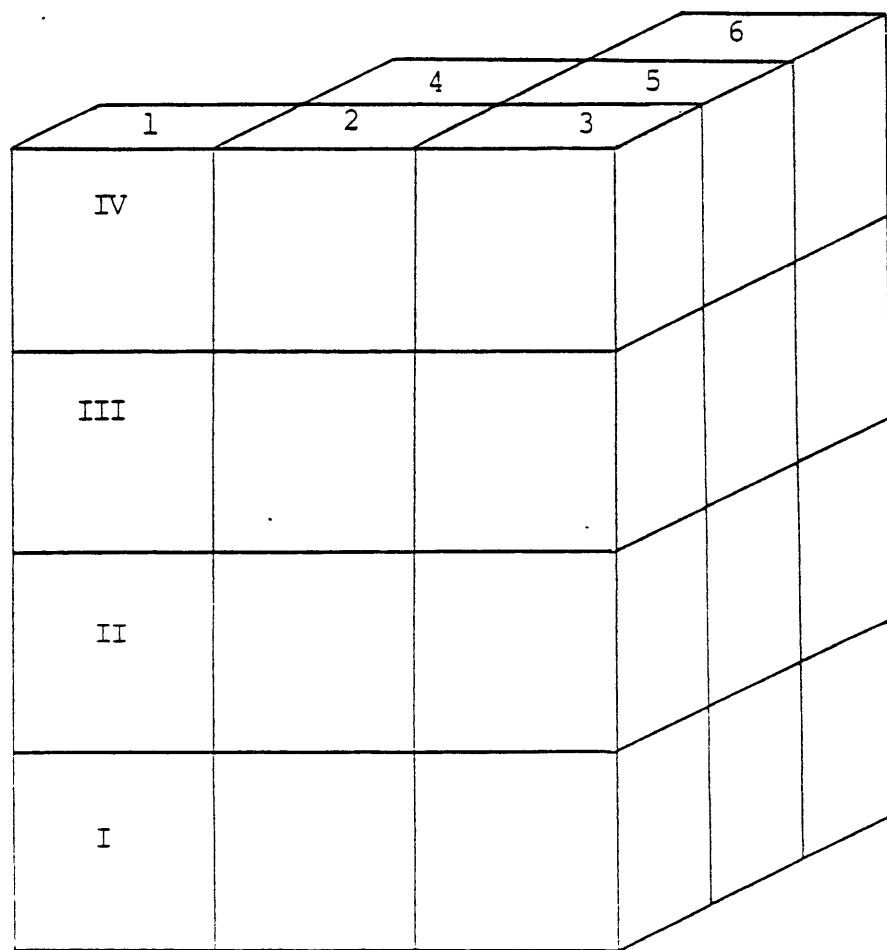
Figure A.1 Six-channel example.
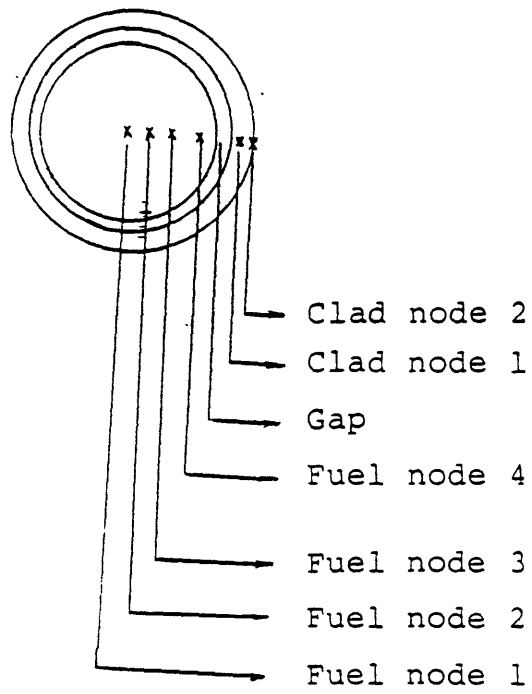
Figure A.2   Fuel and clad nodes example.

noppt=0, print all six channels.
nopt=6

If noppt=1 and nopt=2, two of the six channels will be printed out. The two channels will be determined by the array input 'ncopt(nopt)' (Card 14).

2.2.4 Card Four

(i) Variable: itb, ibb, iflash, ifintr, iht, iss, iqss, ichf, iwft, ivec, itam, imixm, imixe, iafm, itfm, igfm, grav, hdt, velx, inton, idpmax

(ii) Description: These variables are related to the thermohydraulics model. The interger variables act as indicators for the options which the user may select.

itb= Top boundary condition indicator (0/1) (pressure/velocity)

ibb= Bottom boundary condition indicator (0/1/2) (pressure/velocity/total inlet flow rate)

iflash= Interfacial mass exchange model (0/1/2) (Nigmatulin Model/Suppressed, i.e., Γ =0/Nonequilibrium Boiling Model)

ifinter= Interfacial momentum exchange model (0/1) (MIT/LASL)

iht= Heat transfer indicator (0/1/2/3) (No heat transfer/Constant gap conductance, temperature independent fuel, cladding conductance/ Constant gap conductance, temperature dependent fuel, cladding conductance/ Temperature dependent gap, fuel, cladding conductance)

iss= Heat transfer calculation type (0/1/2) (Transient/ Steady-state/ Steady-state with critical heat flux check suppressed)

iqss= Steady-state heat flux indicator (0/1) (Heat flux is held constant and no fuel temperatures are calculated/Heat flux is not held constant and the fuel temperature is calculated)

ichf= Critical heat flux indicator (1/2/3/4/5/6) (Biasi and CHF-Void correlations/ W-3/ CISE/ Barnett/Bowring/ Hench-Levy)

iwft= Transverse friction model indicator (0/1) (No friction/Gunter-Shaw correlation)

156

ivec= Transverse velocity indicator (0/1) (Actual transverse velocity is used/ The magnitude of the velocity vector is used)

itam= Fluid dynamics indicator (0/1) (No transverse flow allowed, i.e., closed-channel calculation/Normal, i.e., open-channel calculation)

imixm= Momentum turbulent mixing indicator (0) (No mixing is allowed in this version of TITAN)

imixe= Energy turbulent mixing indicator (0) (No mixing is allowed in this version of TITAN)

iafm= Axial friction model indicator (0/1) (Default/User supplied, see card six)

itfm= Transverse friction model indicator (0/1) (Default/User supplied, see card six)

igfm= Grid friction model indicator (0/1) (Default/User supplied, see card six)

grav= Gravitational constant (Usually: $-9.81 m/s^2$)

hdt= Hydraulic diameter in transverse direction (m) =(4× Free volume)/Rod surface area

velx= Velocity multiplier for transverse friction, normally should be set equal to the ratio of the maximum to average transverse flow area

inton= Numerical scheme indicator (0/1/2/3) (Semi implicit/ 1-D fully implicit with march method/ 1-D fully implicit with direct method/ 1-D fully implicit with 3 time steps of direct method then march method)

idpmax= Maximum number of outer iterations for correct flow split (typically 2 for steady state; 4 for transients)

2.2.5 Card Five

This card is required only if ibb=2.
   (i) Variable: winlet
   (ii) Description:
          winlet=The total inlet flow rate $[Kg/s]$.

2.2.6 Card Six

    (i) Variables:

           If iafm=1, need $a0$, rex, a, b

           If itfm=1, need $a0$, ret, a, b

           If igfm=1, need a, b

    (ii) Description: The friction coefficient is defined as

        (a) Laminar flow

$$f = \frac{a_0}{Re}; \quad Re < \frac{Re_x}{Re_t} \tag{A.1}$$

        (b) Turbulent flow:

$$f = a \times Re^b; \quad Re > \frac{Re_x}{Re_t} \tag{A.2}$$

Therefore, we need $a0$, rex, ret, a, b as input variables. The default values are in Table A.1.

2.2.7 Card Seven

    (i) Variables: idump, nitmax, iitmax, epsn, epsi, marmax, epsmar, epso

    (ii) Description: This card contains the thermohydraulic iteration control and dump indicator variables.

idump= Dump file request indicator (0/1) (No/Yes)

nitmax= Maximum number of Newton iterations

iitmax= Maximum number of inner iterations

epsn= Newton iteration convergence criterion

epsi= Inner iteration convergence criterion, i.e., the pressure iteration convergence criterion

marmax= Maximum number of marching iterations (typically 4 for steady state; 6 to 10 for transients)

epsmar =Convergence criterion for marching iteration (typically 1.e-6 for steady state; 1.e-3 for transients)

epso= Convergence criterion for outer iteration (0.01 or so)

Table A.1

Friction Model Default Constants

	a0	rex/ret	a	b
Axial Friction Model	64	1502.11	0.184	-0.2
Transverse Friction Model	180	202.5	1.92	-0.145
Grid Friction Model	-	-	3.0	-0.1

A relative error check on the pressure is used in this code, so that the iteration proceeds until the condition

$$max \left| \frac{p^m - p^{m-1}}{p^m} \right| < eps \qquad (A.3)$$

is met, where m designates either the Newton or inner iteration and where eps is either epsn or epsi. The maximum is taken over all mesh cells, but only the pressure is checked for convergence during the Newton iteration. In no case, however, will the total iteration count be allowed to exceed the limits specified by nitmax and iitmax. If these limits are reached and nitmax> 0, iteration ceases, and code operation continues as if the iteration had converged. If nitmax<0, then the calculation stops when the limits are reached.

It is important to remember that all variables in the calculation of thermohydraulic part of this code are derived from the pressure solution, therefore, if the pressure solution is not converged tightly enough, errors in pressure solution may be amplified as other variables are computed from it. The user is therefore cautioned to be certain that the convergence criteria are sufficiently small by repeating the calculation with smaller values for those quantities, if possible. One must also remember, however, that on a finite precision machine there is a lower limit to these quantities, below which roundoff errors will prevent convergence.

2.2.8 Card Eight

    (i) Variables: q0, qcf, t0, omg, radr, thc, thg, hgap, ftd, fpuo2, fpress, cpr, expr, grgh, pgas, (gmix(k),k=1,4), burn

    (ii) Description: This card is required only if iht is not equal to 0. (See card four).

q0= Initial total power (W); If q0<0, then q0 is set to be the current power.

qcf= Fraction of power which is generated from the coolant.

t0= Delay time (s) = 0.0 for this version of TITAN.

omg= Inverse reactor period $(1/s)$ = 0.0 for this version of TITAN.

radr= Outer fuel rod radius (m)

thc= Clad thickness (m)

thg= Gap width (m)

160

hgap= Gap heat transfer coefficient $[W/(m^2 \cdot {}^\circ K)]$. Suggested value $= 5.678e3$

ftd= Fraction of theoretical density of fuel

fpuo2= Fraction of PUO_2 in fuel

fpress= Fuel pressure on clad for gap conductance model $(Pa = N/m^2)$

cpr= Coefficient for the above pressure

expr= Exponent for the above pressure

grgh= Gap roughness (m). If zero is given as input, a default value of $4.4e - 6\ m$ is assumed.

pgas= Gap gas pressure (Pa)

gmix(1)= Helium fraction in gap gas

gmix(2)= Argon fraction in gap gas

gmix(3)= Krypton fraction in gap gas

gmix(4)= Xenon fraction in gap gas

burn= Fuel average burnup (MWD/MTU). This variable is used in the cracked-pellet model, which accounts for partial contact of fuel against clad.

The variables fpress, cpr, expr are used if the gap conductance is to be supplemented by a term CP_f^M to represent the effect of a closed gap (grgh>thg) with fuel pressing against clad with pressure P_f. The term may be suppressed by giving cpr=0, fpress=0 and expr=1.

2.2.9 Card Nine

(i) Variables: idiag, irstrt, jprinta, jprintb, jprintc, jprintd, jprinte, jprint, ibpont, icore, itran, ndpg, ithfbk, ixenon, ecf, ibc

(ii) Description

idiag= Diagonal symmetry, lower right to upper left in the reactor plane (0/1) (No/Yes)

irstrt= Leakage Approximation/Point Kinetics extrapolation

irstrt	Leakage Approximation	Point Kinetics Extrapolation
0	Quadratic	Yes
1	Flat	Yes
2	Quadratic	No
3	Flat	No

Suggested value=0 for most problems.

jprinta= Print flag for total power(0/1) (No/Yes)

jprintb= Print flag for normalized nodal power(0/1) (No/Yes)

jprintc= Print flag for nertron fluxes(0/1) (No/Yes)

jprintd= Print flag for nertron leakages(0/1) (No/Yes)

jprinte= Print flag for normalized assembly power(0/1) (No/Yes)

jprint= Print flag (<3/3/4/5/6) (No print for expansion coefficient, steady state matrices, albedo oriented map and reactor oriented map/ expansion coefficient/ steady state matrices / albedo oriented map/ albedo oriented and reactor oriented maps)

ibpont= BPOINTER print flag=0 (Not used in this version of TITAN)

icore= container array size in CDC words=0 (Not used in this version of TITAN)

itran= Transient problem (0/1/2/3/4) (Static/ Initiate transient with control rod, Cusping applied/ Initiate Transient with flow rate/ Initiate transient with inlet temperature/ Same as 1 but without Cusping effect)

ndpg= Number of delayed neutron groups (≤ 6)

ithfbk= Type of thermohydraulic feedback (0/1/2/3) (None/ Cross sections are linear functions of fuel, moderator temperatures and moderator density/ Quadratic feedback model is used/ In addition to option 2, feedback coefficients of perturbed portion are considered)

162

ixenon= Equilibrium xenon model indicator (0/1) (No/Yes)

ecf= Energy conversion factor in W-sec per fission, default= 3.204e − 11

ibc= Indicator of boron concentration model (0/1/2) (not used/ 1-D model/ 3-D model)

2.2.10 Card Ten

(i) Variables: minout, maxout, minflx, maxflx, ninner, noutpu. This card describes the neutronic iteration specifications.

(ii) Description:

minout= Minimum number of outer iterations

maxout= Maximum number of outer iterations, if maxout=0, default number, 100, is used.

minflx= Minimum number of flux iterations

maxflx= Maximum number of flux iterations, if maxflx=0, default number, 3, is used.

ninner= Number of inner iterations, if ninner=0, default number, 1, is used. Default number is recommended.

noutpu= Number of outer iterations per matrix update, if noutpu=0, default number, 5, is used.

2.2.11 Card Eleven

(i) Variables: guessk, guark, shiftk

(ii) Description: This card describes the input eigenvalue and shift factors.

guessk= Initial eigenvalue guess, if guessk=0, default number, 1.0, is used.

guark= Shift factor (generally 0.04 to 0.05)

shiftk= Initial eigenvalue shift, if shiftk=0, default number, 1.5, is used. Default number is recommended.

2.2.12 Card Twelve

(i) Variables: epsk, epsphi, errorr

(ii) Description: This card describes the neutronic convergence criteria.

epsk= Eigenvalue convergence criteria, default value is $1.e-6$.

epsphi= Node power convergence criteria, default value is $1.e-4$.

errorr= Error reduction on flux iterations, default value is 0.3.

2.2.13 Card Thirteen

(i) Variables: nx, nunqpl, nalb, ncomp, nedtx, nedty, nedtz

(ii) Description: This card describes the problem size.

nx= Number nodes in x-direction, for example, in Figure A.1, nx is 3.

nunqpl= Number of unique planes, that is, number of different composition map planes. This variable counts the bottom and top fictitious cells too.

nalb= Number of albedo sets.

ncomp= Number of compositions, i.e., unique cross-section sets.

nedtx= Number of edit bounds, x-direction.

nedty= Number of edit bounds, y-direction.

nedtz= Number of edit bounds, z-direction.

(iii) Example: See Figure A.3.

nx=2
nunqpl=4
nalb=2, see Figure A.6 too.
ncomp=4
nedtx=2
nedty=1
nedtz=3

Note here, nedtx, nedty, nedtz are not necessary to be 2, 1, 3 respectively. This will be explained more clearly in Section 2.3.6.12.

2.3 Array Data

Plane #

Fictitious Cell

Albedo
B.C.

5   -1   -1

4   1   2

3   3   4

2   1   2   Compo-
sition
#

1   0   0

Zero Flux B.C.
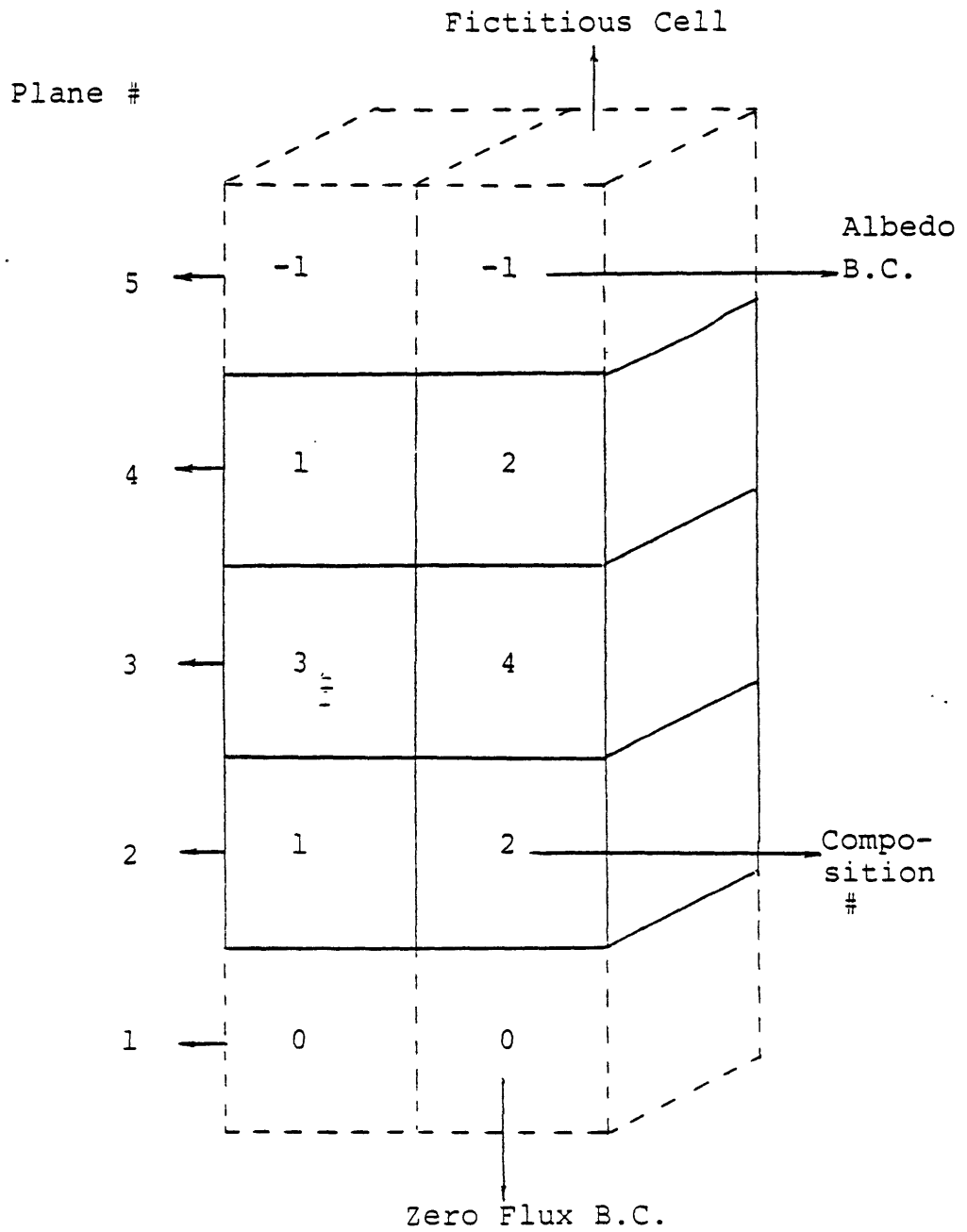
Figure A.3   Card twelve example.

The second group of input is that related to the array data. These data are read in via the 'NIPS' input subroutine. The array data is divided into six general sections: geometrical data, friction model, initial and boundary conditions, heat transfer model, transient forcing function, and the neutronic data. Each of these is discussed below.

2.3.1 Geometrical Data

The mesh is basically a regular, orthogonal, x-y-z grid, but boundaries in the x-y plane may be irregular and mesh spacings in all three dimensions can vary with location. The node numbering schemes of THERMIT and QUANDRY are different. This was discussed in the report.

The numbering scheme assigns the index 1 to the left-most cell in the bottom row of cells, incrementing the index from left to right and bottom to top as indicated in Figure A.4. The convention for the positive direction for the x and y axes is also indicated in this figure. This convention must be remembered in interpreting the signs of the velocities printed out by the code.

The geometrical data required is described in the following sections.

2.3.1.1 Card Fourteen
    (i) Variables: ncopt(nopt)
    (ii) Description: This card is required only if nopt>0 and noppt=1

ncopt= The channel numbers whose information will be printed out.
    (iii) Example: See Figure A.4.

Case I: Print out all 12 channels–
        nopt=12
        noppt=0 No ncopt(nopt) is required.

Case II: No thermal-hydraulic print out–
        nopt=0
        noppt=1 No ncopt(nopt) is required.

Case III: Channels 2 and 3 are printed out–
        nopt=2
        noppt=1
        ncopt(1)=2
        ncopt(2)=3
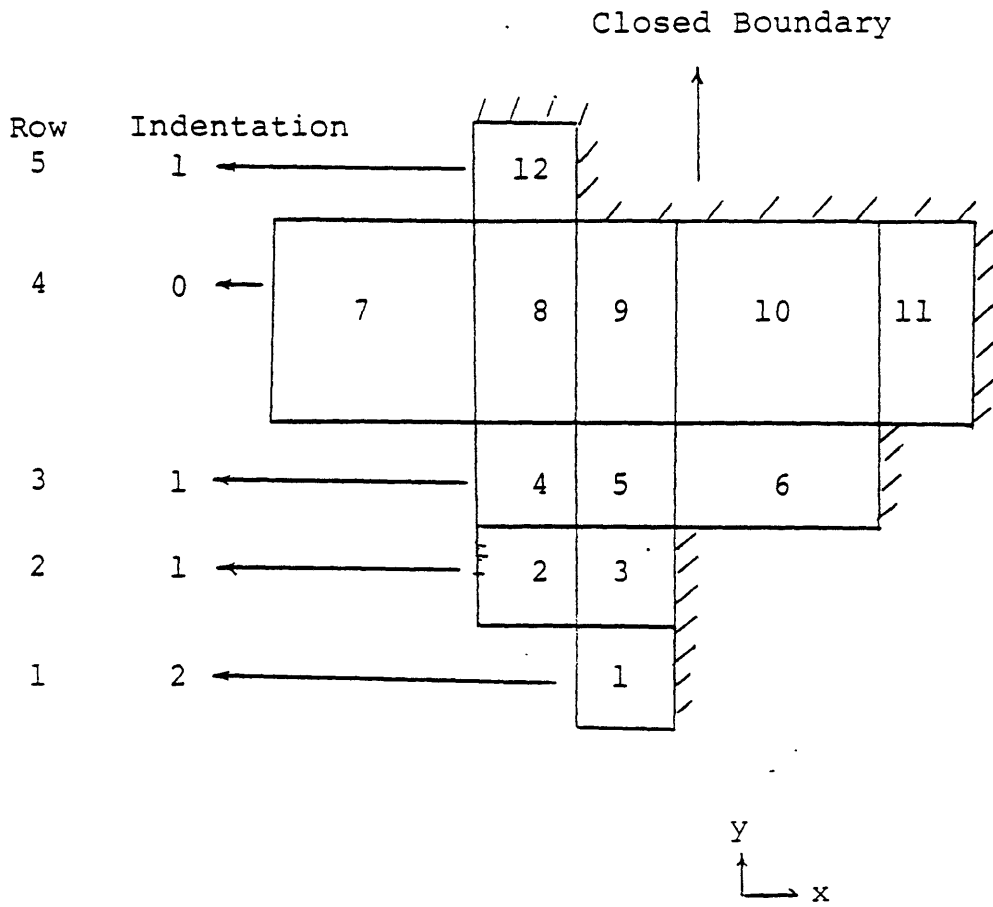
Figure A.4 A possible arrangement of channels in the horizontal plane, showing the index conventions.

2.3.1.2 Card Fifteen

(i) Variables: ncr(nr)

(ii) Description:

ncr= The number of cells in each row. nr is the dimension of ncr. Recall that nr is the number of rows. No gaps are allowed in a row of cells.

(iii) Example: See Figure A.4.

ncr(1)=1

ncr(2)=2

ncr(3)=3

ncr(4)=5

ncr(5)=1

2.3.1.3 Card Sixteen

(i) Variables: indent(nr)

(ii) Description:

indent= Identation for each row. The identation is specified from a left boundary which is a fixed position defined by the leftmost node (channel). This can be explained in the following example.

(iii) Example: See Figure A.4.

ident(1)=2

ident(2)=1

ident(3)=1

ident(4)=0

ident(5)=1

2.3.1.4 Card Seventeen

(i) Variables: arx(nz,nc)

(ii) Description:

arx= Mesh cell areas in the x-direction (m^2). arx for each mesh cell is the area on the left side of the cell. If we set arx at one mesh cell to be equal to zero, that means no flow is allowed to cross that boundary. Also note here, right most cells have been automatically set to be zero, see Figure A.4. The dimension (nz,nc) means that we have to specify the areas from bottom to top for each channel. The flow area should be the volume average areas between the two appropriate cells.

(iii) Example: See Figure A.1. The order of arx is

$$
\begin{pmatrix}
arx(1,1) & arx(2,1) & arx(3,1) & arx(4,1) \\
arx(1,2) & arx(2,2) & arx(3,2) & arx(4,2) \\
arx(1,3) & arx(2,3) & arx(3,3) & arx(4,3) \\
& \cdot & & \cdot \\
& \cdot & & \cdot \\
& \cdot & & \cdot \\
arx(1,6) & \cdots\cdots & \cdots\cdots & arx(4,6)
\end{pmatrix}
$$

The value of arx(3,5), for example, should be

$$
arx(3,5) = \frac{vol(3,4) + vol(3,5)}{hx(3,4) + hx(3,5)} \tag{A.4}
$$

where *vol* and *hx* are defined in cards 20 and 21 respectively.

2.3.1.5 Card Eighteen

    (i) Variables: ary(nz,nc)

    (ii) Description:

ary= Mesh cell areas in the y-direction (m^2). ary for each mesh cell is the area on the lower side of the cell in the x-y plane. All other things about ary are the same as those of arx.

    (iii) Example: See Figure A.1. The value of ary(3,5), for example, should be

$$
ary(3,5) = \frac{vol(3,3) + vol(3,5)}{hy(3,3) + hy(3,5)} \tag{A.5}
$$

where *hy* is defined in card 22.

2.3.1.6 Card Ninteen

    (i) Variable: arz(nz+1,nc)

    (ii) Description:

arz= Mesh cell flow area in the z-direction (m^2). Here, for each channel i, the quantities arz(j,i) for j=1,2,3... nz+1 represents the areas beginning with the bottom face of the first nonfictitious cell in the channel and ending with the top face of the final non-fictitious cell.

    (iii) Example: See Figure A.5. In Figure A.5, we can see the positions of arz. Note, the definition of arz is the flow area in the z-direction, therefore, we have to exclude the area occupied by the fuel rods and all other structure within that cell when we calculate arz.

2.3.1.7 Card Twenty

    (i) Variables: vol(nz,nc)
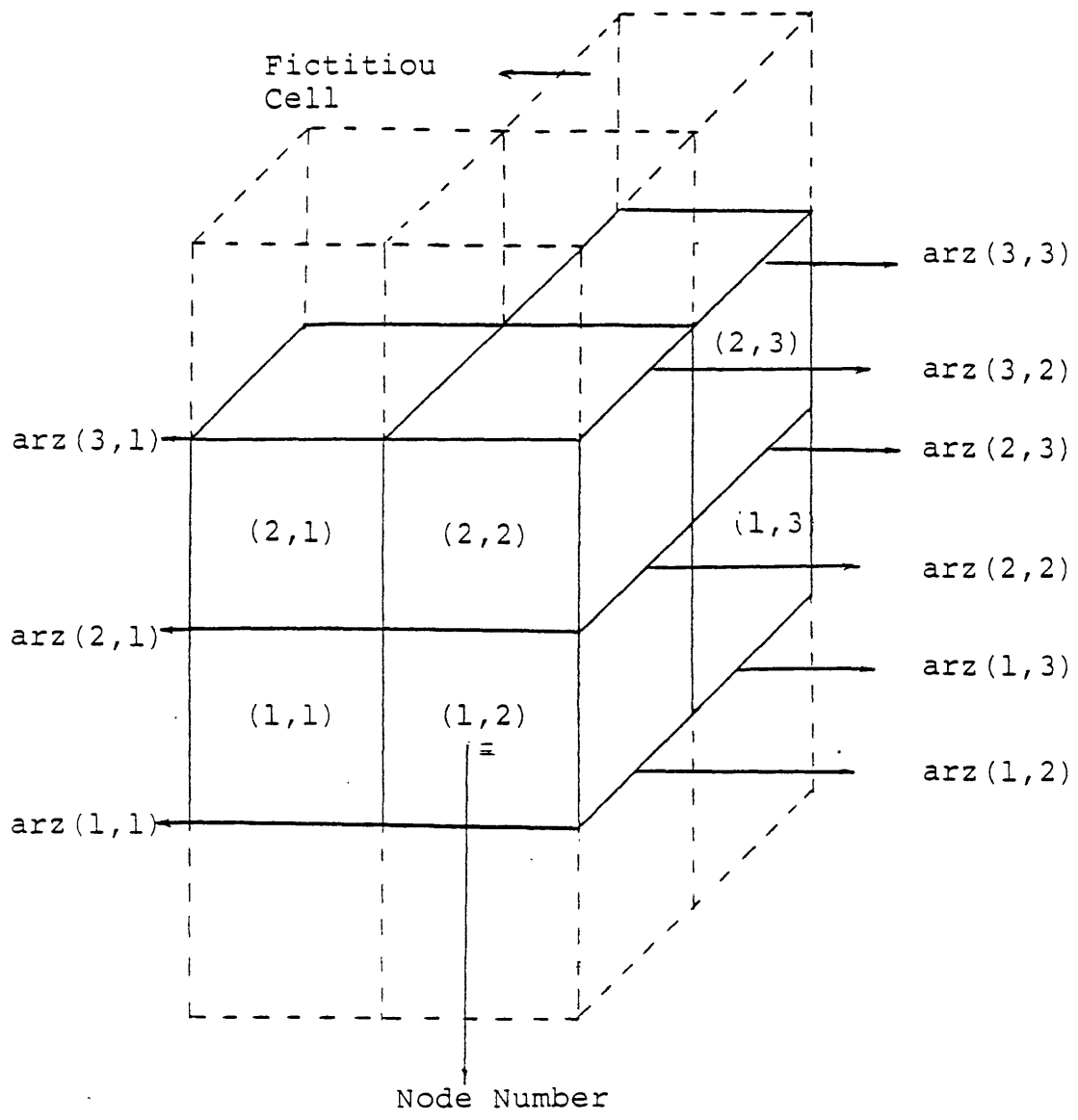
Figure A.5   Example of arz and fictitiou cells.

(ii) Description:

vol= Mesh cell free volumes (m^3). The meaning of free volume is that the mesh cell volume excluding the volume of fuel rods and all other structure volume within that cell.

2.3.1.8 Card Twenty-One
(i) Variables: hx(nx)
(ii) Description:

hx= Mesh spacing in the x-direction (m). Since all hx must be the same for a given column, therefore, we need to specify only nx values of hx.

2.3.1.9 Card Twenty-Two
(i) Variables: hy(nr)
(ii) Description:

hy= Mesh spacing in the y-direction (m). Since all hy must be the same for a given row, therefore, we need to specify only nr values of hy.

2.3.1.10 Card Twenty-Three
(i) Variables: dz(nz+2)
(ii) Description:

.dz= Mesh spacing in the z-direction (m). nz+2 means we have to specify the z-direction mesh spacing for both bottom and top fictitious cells. For fictitious cells, see Figure A.5.

2.3.1.11 Card Twenty-Four
(i) Variables: hdz(nz+2,nc)
(ii) Description:

hdz= Axial hydraulic diameter or each channel (m). This is defined as

$$hdz = \frac{4 \times arz}{P_{wet}} \qquad (A.6)$$

where P_{wet} = Wetted perimeter within the cell.

2.3.1.12 Card Twenty-Five
(i) Variables: sij(4,nc)
(ii) Description:

sij= Gap interconnections for each channel $[m]$; =0.0 for this version of TITAN.

2.3.2 Friction Model Data, Card Twenty-Six
   (i) Variables: iwfz(nz+1)
   (ii) Description:

iwfz= Indicator for axial friction Model. Axial friction and form loss are specified by the array iwfz(nz+1), with one value associated with each axial velocity level. Form loss is attributed to a given axial velocity level if the spacer grid lies anywhere between the two neighboring pressure points. The indicator iwfz is made up of a tens digit and a unit digit, whose meanings are as follows:
$$- \text{tens digit}= \quad 0 \text{ axial friction only}$$
                1 axial friction+form loss
                2 as 1 + funnel effect
$$- \text{unit digit}= \quad 0 \text{ no friction}$$
               1 Martinelli multiplier
               2 Martinelli-Nelson multiplier
                 with mass flow effect
               3 Levy multiplier
               4 Rough tube correlation
                 with Levy multiplier

If iwfz=10, that means that we chose form loss without axial friction.

2.3.3 Initial and Boundary Conditions

The initial conditions are required for both transient and steady-state calculations. For steady-state calculations, the initial condition is simply a guess, the final solution is independent of this guess. But, because of the characteristics of the boiling curve, if the initial rod temperatures are in the stable film boiling regime, the final steady-state solution may yield rod temperatures in this regime, whereas a starting guess of a lower rod temperature may yield a final steady-state solution with rod temperatures in the nucleate boiling heat transfer regime.

2.3.3.1 Card Twenty-Seven
   (i) Variables: p(nz+2,nc)
   (ii) Description:

p= Initial pressures (Pa). $nz + 2$ means we have to give the initial guess to the

bottom and top fictitious mesh cells too. If pressure boundary condition is chosen, the boundary velocities are determined by solving momentum equations at the boundary.

2.3.3.2 Card Twenty-Eight
    (i) Variabels: alp(nz+2,nc)
    (ii) Description:

alp= Initial vapor volume fraction. Same as p, we have to give the initial guess to the bottom and top fictitious mesh cells too. Note, if there is only single vapor phase exists, use alp=0.9999 instead of 1.

2.3.3.3 Card Twenty-Nine
    (i) Variables: tv(nz+2,nc)
    (ii) Description:

tv= Initial vapor temperature ($^\circ K$). The initial liquid temperature is set equal to tv.

2.3.3.4 Card Thirty
    (i) Variables: vvz(nz+1,nc)
    (ii) Description:

vvz= Initial vapor axial velocity (m/s). The initial liquid axial velocity is set equal to vvz. All the initial transverse vapor and liquid velocities are set to be zero. The (nz+1,nc) array was explained in section 2.3.1.6. If velocity boundary condition is chosen, there is no momentum equation is solved at the boundary, and the boundary velocities are set to be the input values.

2.3.4 Heat Transfer Model Input Data

This part of data is required if iht>0.

2.3.4.1 Card Thirty-One
    (i) Variables: icr(nc)
    (ii) Description:

icr= Adjacent channel number for a given rod. For this version of TITAN, icr(nc)=nc.
    (iii) Example: For the channel arrangement in Figure A.5,

icr(1)= 1

icr(2)= 2

icr(3)= 3

2.3.4.2 Card Thirty-Two
  (i) Variables: hdh(nz+2,nc)
  (ii) Description:

hdh= Equivalent heated diameter for given channel

$$hdh = \frac{4 \times Flow\ area}{Heated\ Perimeter} \qquad (A.7)$$

2.3.4.3 Card Thirty-Three
  (i) Variables: tw(nz,nc)
  (ii) Description:

tw= Initial wall surface temperature $(^\circ K)$.

2.3.4.4 Card Thirty-Four
  (i) Variables: qr(ncf+1+ncc)
  (ii) Description:

qr= Fuel pin radial power shape.

2.3.4.5 Card Thirty-Five
  (i) Variables: rn(nc)
  (ii) Description:

rn= Number of fuel rods in each channel.

2.3.4.6 Card Thirty-Six
  (i) Variables: fracp(nc)
  (ii) Description:

fracp= Fraction of heated perimeter facing adjacent channel. In this version of TITAN, we cannot use it for subchannel analysis, therefore, we set fracp=1 for all channels.

2.3.5 Transient Forcing Functions

The transient forcing functions are used to change the boundary conditions as a function

of time so that reactor transients may be simulated. The code linearly interpolates between given multipliers. If at any time which is less than the first entry then a multiplier of 1.0 is used. If at any time which is greater than the last entry, the last factor in the table is used. All of these forcing function tables can be changed or updated in the 'tffdata' restart namelist (See 3.2.1.3).

2.3.5.1 Card Thirty-Seven

This card includes four transient forcing function indicators.
- (i) Variables: nb, nt, ntemp, nq
- (ii) Description:

nb= Number of entries in bottom boundary condition forcing function table ($\leq$ 30).

nt= Number of entries in top boundary condition forcing function table ($\leq$ 30).

ntemp= Number of entries in inlet temperature forcing function table ($\leq$ 30).

nq= Number of entries in reactor power forcing function table. Since the reactor power is generated by the code itself, nq=0 for this version of TITAN.

nitr= Number of inlet temperature regions ($\leq$ 10).

2.3.5.2 Card Thirty-Eight

This card is required only if nb>0.
- (i) Variables: botfac(i),yb(i); i=1,nb
- (ii) Description:

botfac= Bottom boundary condition multiplier. The multipliers are for pressure, velocity and total inlet flow rate according to ibb=0, 1, 2, respectively.

yb= Time corresponding to multiplier

These variables should be read in as pairs, i.e., botfac(1), yb(1), botgad(2), yb(2), ... botfac(nb), yb(nb).
- (iii) Example: If the bottom boundary condition option is velocity, and we have a flow decay transient, the multiplier is 1, 0.5, 0.2 corresponding to time 0 sec, 1 sec and 2 sec, respectively, then, the input should be;
1.,0.,0.5,1.,0.2,2.

2.3.5.3 Card Thirty-Nine

This card is required only if nt>0.
    (i) Variables: topfac(i),yt(i); i=1,nt
    (ii) Description:

topfac= Top boundary condition multiplier. The multipliers are for pressure, velocity according to ibb=0, 1, respectively.

yt= Time corresponding to multiplier

These variables should be read in as pairs i.e., topfac(1), yt(1), topfac(2), yt(2), ..., topfac(nt), yt(nt)

2.3.5.4 Card Forty

This card is required only if ntemp>0.
    (i) Variables: tinfac(i,j), ytemp(i,j); i=1,ntemp, j=1,nitr
    (ii) Description:

tinfac= Inlet temperature multiplier

ytemp= Time corresponding to multiplier

These variables should be read in as pairs, i.e., tinfac(1,1), ytemp(1,1), tinfac(2,1), ytemp(2,1), .., tinfac(ntemp,1), ytemp(ntemp,1), tinfac(1,2), ytemp(1,2), .., tinfac(ntemp,2), ytemp(ntemp,2), .., tinfac(ntemp,nitr), ytemp(ntemp,nitr)

2.3.5.5 Card Forty-One

This card is required only if ntemp>0.
    (i) Variables: nctr(j); j=1,nitr
    (ii) Description:

nctr(j)= Number of channels in region j; (≤ 50)

2.3.5.6 Card Forty-Two

This card is required only if ntemp>0.
    (i) Variables: ncit(i,j); i=1,nctr, j=1,nitr
    (ii) Description:

ncit(i,j)= The ith channel number in region j.

2.3.6 The Nuclear Data

In this part of input data, necessary nuclear information is read.

2.3.6.1 Card Forty-Three

    (i) Variables: nplane(nunqpl)

    (ii) Description:

nplane= Number of planes in this unique plane.

    (iii) Example: See Figure A.3.

        nplane(1)= 1 – the 1st bottom fictitious plane

        nplane(2)= 2 – the 2nd and 4th planes

        nplane(3)= 1 – the 3rd plane

        nplane(4)= 1 – the 5th top fictitious plane

2.3.6.2 Card Forty-Four

    (i) Variables: iasn(nz+2)

    (ii) Description:

iasn= Plane numbers belong to a certain unique plane

    (iii) Example: See Figure A.3.

        iasn(1)= 1 – nplane(1)

        iasn(2)= 2 – nplane(2)

        iasn(3)= 4 – nplane(2)

        iasn(4)= 3 – nplane(3)

        iasn(5)= 5 – nplane(4)

2.3.6.3 Card Forty-Five

    (i) Variables: irow(nx+2,nr+2,nunqpl)

    (ii) Description:

irow= Composition assignment for each unique plane. This input data must be input from left to right, bottom to top.

    (iii) Example: See Figures A.3 and A.6.

    irow(1,1,1)= 0

    irow(2,1,1)= 0

    irow(4,1,1)= 0

    irow(1,2,1)= 0

            1st plane

```
      0   0   0   0
      0   0   0   0          1
      0   0   0   0

     -2  -1  -1  -1
     -2   1   2  -1          2
     -2  -1  -1  -1

     -2  -1  -1  -1
     -2   3   4  -1          3
     -2  -1  -1  -1

     -2  -1  -1  -1
     -2   1   2  -1          4
     -2  -1  -1  -1

     -1  -1  -1  -1
     -1  -1  -1  -1          5
     -1  -1  -1  -1


          ≛      Plane #
```

Figure A.6   Cross section view of Figure B.3,
             showing the nuclear composition
             assignment.

irow(4,2,1)= 0
irow(1,3,1)= 0

.

.

.

irow(4,3,1)= 0

irow(1,1,2)= −2
irow(2,1,2)= −1
irow(3,1,2)= −1
irow(4,1,2)= −1
irow(1,2,2)= −2
irow(2,2,2)= 1   2nd and 4th planes
irow(3,2,2)= 2
irow(4,2,2)= −1
irow(1,3,2)= −2
irow(2,3,2)= −1
irow(3,3,2)= −1
irow(4,3,2)= −1

irow(1,1,3)= −2
irow(2,1,3)= −1
irow(3,1,3)= −1
irow(4,1,3)= −1
irow(1,2,3)= −2
irow(2,2,3)= 3   3rd plane
irow(3,2,3)= 4
irow(4,2,3)= −1
irow(1,3,3)= −2
irow(2,3,3)= −1
irow(3,3,3)= −1
irow(4,3,3)= −1

irow(1,1,4)= −1

5th plane

irow(4,2,3)= −1

2.3.6.4 Card Forty-Six

(i) Variables: xsrf(14,ncomp)

(ii) Description:

xsrf= The reference cross section of each composition material. For each composition, we need 14 different cross sections corresponding to two-group approximation. They are described as follows.

xsrf(1,j)= Group 1 diffusion coefficient in x-direction, D_{1x}

xsrf(2,j)= Group 1 total cross section, $\Sigma_1 = \Sigma_a + \Sigma_{21}$.

xsrf(3,j)= Group 1 scattering cross section Σ_{21}

xsrf(4,j)= Group 1 $\nu \times$ fission cross section $\nu\Sigma_{f1}$

xsrf(5,j)= Group 1 fission cross section Σ_{f1}

xsrf(6,j)= Group 2 diffusion coefficient in x-direction, D_{2x}

xsrf(7,j)= Group 2 total cross section $\Sigma_2 = \Sigma_a$

xsrf(8,j)= Group 2 scattering cross section $\Sigma_{12} = 0$

xsrf(9,j)= Group 2 $\nu \times$ fission cross section $\nu\Sigma_{f2}$

xsrf(10,j)= Group 2 fission cross section Σ_{f2}

xsrf(11,j)= Group 1 diffusion coefficient in y-direction, D_{1y}

xsrf(12,j)= Group 2 diffusion coefficient in y-direction, D_{2y}

xsrf(13,j)= Group 1 diffusion coefficient in z-direction, D_{1z}

xsrf(14,j)= Group 2 diffusion coefficient in z-direction, D_{2z}

where j is the jth composition; j=1,ncomp. All the above variables have the same dimension $[cm^{-1}]$.

2.3.6.5 Card Forty-Seven

This is required only if ibc is not equal 0.
  - (i) Variables: bbs(nc)
  - (ii) Description:

bbs= Pure boron mass flow rate at the core inlet of each channel $[Kg/sec]$.

2.3.6.6 Card Forty-Eight

This card is required only if ixenon=1.
  - (i) Variables: xesig(ncomp)
  - (ii) Description:

xesig= Xenon cross section of each composition $[b]$.

2.3.6.7 Card Forty-Nine

This card is required only if ixenon=1.
  - (i) Variables: yield(ncomp)
  - (ii) Description:

yield= Xenon yield. This yield includes γ_{Xe} and γ_I i.e.,

$$\gamma_{sum} = \gamma_{Xe} + \gamma_I \qquad (A.8)$$

This is due to the fact that xenon is created by the decay of I and the direct fission process.

2.3.6.8 Card Fifty

This card is required only if ixenon=1.
  - (i) Variables: xelam(ncomp)
  - (ii) Description:

xelam= Xenon decay constant of each composition.

2.3.6.9 Card Fifty-One

This card is required only if nalb>0.
  - (i) Variables: alb(15,nalb)
  - (ii) Description:

alb= Albedoes and expansion factors. Altogether we need 15 variables, five for each direction (x,y,z). The five basic variables are defined as follows.

$$\begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \cdot \begin{pmatrix} J_1 \\ J_2 \end{pmatrix} \qquad (A.9)$$

where ϕ_1, ϕ_2 are the group 1 and 2 neutron fluxes, and J_1, J_2 are the group 1 and 2 neutron currents. a_1, a_2, a_3, and a_4 are the first four variables. The fifth one is so called ALRATIO, which is defined as the ratio of the transverse leakage in the last node of the reactor to the next nonexistant node. A value of -1.0 is usually a good value.

In Summary:

> alb(1,j)= x-directed a_1
> alb(2,j)= x-directed a_2
> alb(3,j)= x-directed a_3
> alb(4,j)= x-directed a_4
> alb(5,j)= x-directed ALRATIO
> alb(6,j)= y-directed a_1
> alb(7,j)= y-directed a_2
> alb(8,j)= y-directed a_3
> alb(9,j)= y-directed a_4
> alb(10,j)= y-directed ALRATIO
> alb(11,j)= z-directed a_1
> alb(12,j)= z-directed a_2
> alb(13,j)= z-directed a_3
> alb(14,j)= z-directed a_4
> alb(15,j)= z-directed ALRATIO

where j=1,nalb.

2.3.6.10 Card Fifty-Two

(i) Variables: iedx(nedtx)

(ii) Description:

iedx= Last node number in each edit segment; x-direction.

(iii) Example: See 2.3.6.12.

2.3.6.11 Card Fifty-Three

(i) Variables: iedy(nedty)

(ii) Description:

iedy= Last node number in each edit segment, y-direction.

(iii) Example: See 2.3.6.12.

2.3.6.12 Card Fifty-Four

    (i) Variables: iedz(nedtz)

    (ii) Description:

iedz= Last node number in each edit segment, z-direction.

    (iii) Example: See Figure A.3 and refer to 2.2.13. The example given in 2.2.13 set nedtx=2, nedty=1 and nedtz=3. This means that we want to edit the power for each node. Therefore, iedx, iedy, iedz should be:

    idex(1)= 1

    iedx(2)= 2

    iedy(1)= 1

    iedz(1)= 1

    iedz(2)= 2

    iedz(3)= 3

Now, if we want to edit the power for the two channels together, then

    nedtx= 1

    nedty= 1

    nedtz= 3

and

    iedx(1)= 2

    iedy(1)= 1

    iedz(1)= 1

    iedz(2)= 2

    iedz(3)= 3

2.3.6.13 Card Fifty-Five

This card reads the cross section feedback coefficients with repect to the moderator density, ρ_m.

    (i) Variables: aa(10,ncomp) $[cm^{-1}/(gm/cm^3)]$

    (ii) Description:

aa= Partial of cross sections w.r.t. ρ_m.

    For each composition we need 10 aa's.

aa(1,j)=

$$\frac{\partial D_1}{\partial \rho_m}; \ if \ ithfbk = 1, \ then$$

aa(1,j)=

$$\frac{\partial D_{-1}}{\partial \rho_m}$$

183

aa(2,j)=
$$\frac{\partial \Sigma_{c1}}{\partial \rho_m} = \frac{\partial \Sigma_1}{\partial \rho_m} - \frac{\partial \rho_m}{\partial \rho_m}$$

aa(3,j)=
$$\frac{\partial \Sigma_{21}}{\partial \rho_m}$$

aa(4,j)=
$$\frac{\partial \nu \Sigma_{f1}}{\partial \rho_m}$$

aa(5,j)=
$$\frac{\partial \Sigma_{f1}}{\partial \rho_m}$$

aa(6,j)=
$$\frac{\partial D_2}{\partial \rho_m}; \ if \ ithfbk = 1, \ then$$

aa(6,j)=
$$\frac{\partial D_2^{-1}}{\partial \rho_m}$$

aa(7,j)=
$$\frac{\partial \Sigma_{c2}}{\partial \rho_m} = \frac{\partial \Sigma_2}{\partial \rho_m} - \frac{\partial \Sigma_{f2}}{\partial \rho_m}$$

aa(8,j)=
$$\frac{\partial \Sigma_{12}}{\partial \rho_m} = 0$$

aa(9,j)=
$$\frac{\partial \nu \Sigma_{f2}}{\partial \rho_m}$$

aa(10,j)=
$$\frac{\partial \Sigma_{f2}}{\partial \rho_m}$$

where j=1,ncomp.

2.3.6.14 Card Fifty-Six

This card reads the cross section feedback coefficients with respect to the coolant temperature.

    (i) Variables: bb(10,ncomp) $[cm^{-1}/^\circ K]$

    (ii) Description:

      bb= Partial of cross sections w.r.t. T_c. The definitions of bb's are the same as aa's except ρ_m now is T_c.

184

2.3.6.15 Card Fifty-Seven

This card reads the cross section feedback coefficients with respect to the square root of the fuel temperature. If ithfbk=1, the coefficients are dependent of the fuel temperature with square root.

(i) Variables: cc(10,ncomp) $[cm^{-1}/\sqrt{°K}]$ or $[cm^{-1}/°K]$

(ii) Description:

cc= Partial of cross sections w.r.t $\sqrt{T_f}$ or T_f. The definitions of cc's are the same as aa's except ρ_m now is $\sqrt{T_f}$ if ithfbk is not equal to 1. Otherwise, ρ_m now is T_f.

2.3.6.16 Card Fifty-Eight

This card is required only if ithfbk is not equal to 1.

(i) Variables: dd(10,ncomp)

(ii) Description:

dd= Partial of unperturbed cross sections w.r.t. ρ_m^2 .The definitions of dd's are the same as aa's except $\partial/\partial\rho_m$ now is $\partial^2/\partial\rho_m^2$.

In summary:

Cross Section	ithfbk	aa	bb	cc	dd
D	$\neq 1$	$\frac{\partial D}{\partial \rho_m}$	$\frac{\partial D}{\partial T_c}$	$\frac{\partial D}{\partial \sqrt{T_f}}$	$\frac{\partial^2 D}{\partial \rho_m^2}$
D	$= 1$	$\frac{\partial(1/D)}{\partial \rho_m}$	$\frac{\partial(1/D)}{\partial T_c}$	$\frac{\partial(1/D)}{\partial T_f}$	
Σ	$\neq 1$	$\frac{\partial \Sigma}{\partial \rho_m}$	$\frac{\partial \Sigma}{\partial T_c}$	$\frac{\partial \Sigma}{\partial \sqrt{T_f}}$	$\frac{\partial^2 \Sigma}{\partial \rho_m^2}$
Σ	$= 1$	$\frac{\partial \Sigma}{\partial \rho_m}$	$\frac{\partial \Sigma}{\partial T_c}$	$\frac{\partial \Sigma}{\partial T_f}$	

2.4 Neutronic Boundary Conditions, Card Fifty-Nine

This card reads the neutronic boundary conditions. Define the notations as follows.

0= Zero Flux

1= Zero Current

2= Albedo

(i) Variables: ibcxl, ibcxu, ibcyl, ibcyu, ibczl, ibczu

(ii) Description:

ibcxl= Boundary condition at x-direction lower side

ibcxu= Boundary condition at x-direction upper side

ibcyl= Boundary condition at y-direction lower side

ibcyu= Boundary condition at y-direction upper side

ibczl= Boundary condition at z-direction lower side

ibczu= Boundary condition at z-direction upper side
  (iii) Example: See Figure A.7.
ibcxl= 1 (left side)
ibcxu= 2 (right side)
ibcyl= 0 (front side)
ibcyu= 2 (back side)
ibczl= 1 (bottom side)
ibczu= 0 (top side)

2.5 Thermal-Hydraulic Refference Data

2.5.1 Card Sixty
  (i) Variables: cfuel, cmod, rofuel, flozro, hzero, u, ah, vfracm, tempin, ratiom, pressr, drhdtm
  (ii) Description: The variables required in this card are needed as the initial guess for the simple heat transfer model used in the code.

cfuel= Specific heat of fuel $[Erg/(gm°K)]$

cmod= Specific heat of coolant $[Erg/(gm°K)]$

rofuel= Density of fuel $[gm/cm^3]$

flozro= Initial mass slow rate through the core $[gm/sec]$

hzero= Film coefficient at initial slow rate $[Erg/(cm^2 - °Ksec)]$, obtained from $hD/k = 0.023Re^{0.8}Pr^{0.4}$

u= Conductivity/conduction length of fuel clad $[Erg/(cm^2 - °Ksec)]$

186

Back Side

Top Side

Right Side

Left
Side

Front
Side

Bottom Side

z

y

x

Zero Flux: Top and Front Sides
Zero Current: Bottom and Left Sides
Albedo: Back and Right Sides
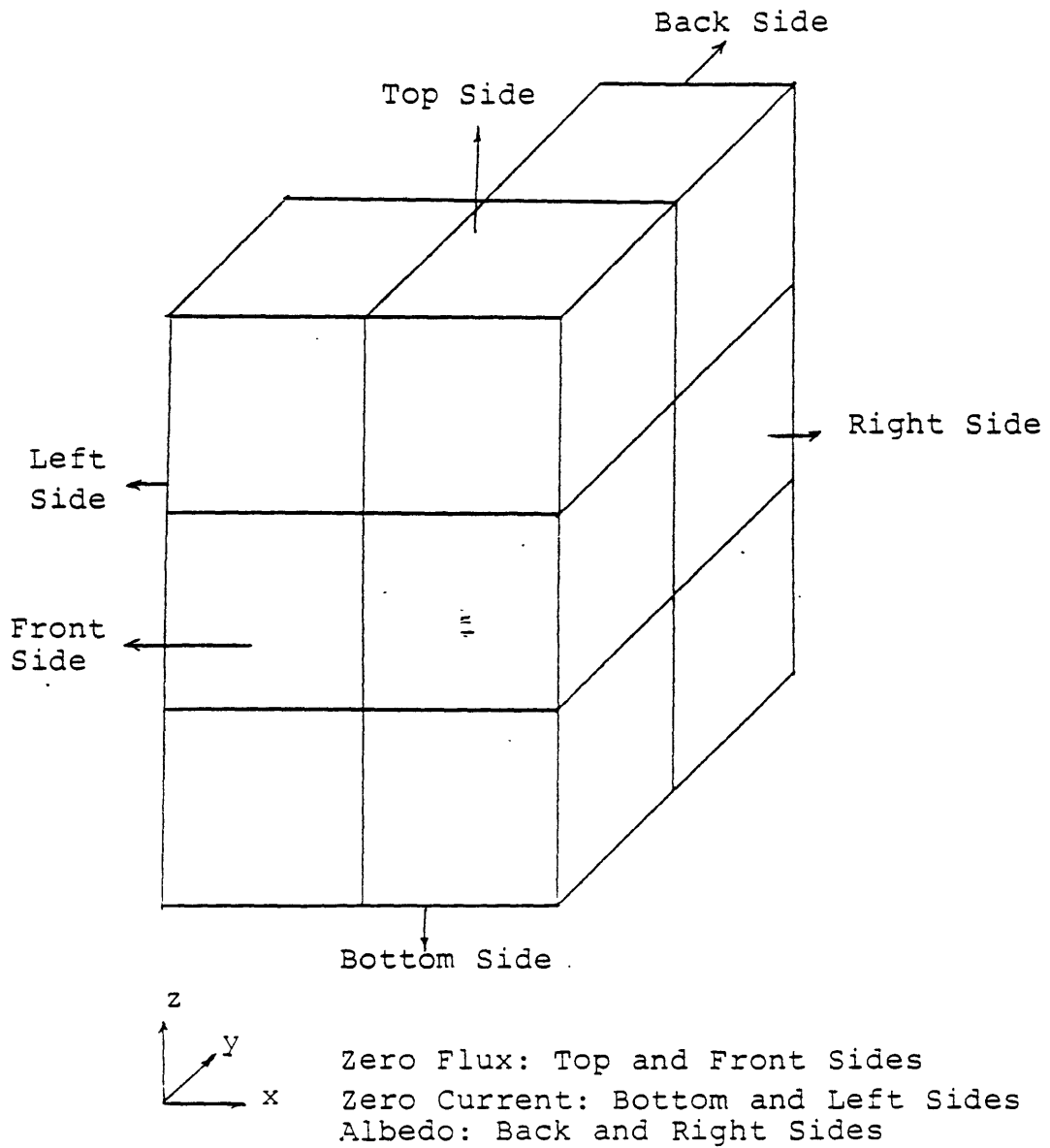
Figure A.7   Example of neutronic boundary condition.

ah= Surface area of clad/Volume of coolant $[cm^{-1}]$

vfracm= Volume fraction of coolant, i.e.,

$$\frac{V_{coolant}}{V_{coolant} + V_{fuel}}$$

tempin= Inlet temperature of the coolant $[^\circ K]$

ratiom= Fraction of fission energy released into the coolant

pressr= Coolant pressure $[Pa]$

drhdtm= Patial of density*enthalpy w.r.t. coolant temperature $[Erg/(cm^3 - ^\circ C)]$, i.e.,

$$\frac{\partial(\rho h)_c}{\partial T_c}$$

2.5.2 Card Sixty-One
(i) Variables: tfref, tmref, denref, ixavg
(ii) Description:

tref= Reference fuel temperature $[^\circ K]$

tmref= Reference coolant temperature $[^\circ K]$

denref= Reference coolant density $[gm/cm^3]$

ixavg= Cross section average option (0/1) (No/Yes)

2.6 Time Step Control, Card Sixty-Two

For each time zone, eight variables are required.
(i) Variables: tend, dtmin, dtmax, dtsp, dtlp, clm, iredmx, nfeed
(ii) Description:

tend= End of time zone

dtmin= Minimum time step size allowed in time zone

dtmax= Maximum time step size allowed in time zone

dtsp= Time interval for short prints

dtlp= Time interval for long prints

clm= Multiplier for convective time step limit

iredmx= Maximum allowed number of time strp reduction

nfeed= Number of the thermohydraulic calculations per neutronic calculation

The time step sizes for both neutronic part and thermohydraulic part are the same. To determine the time step sizes in a time zone, at the beginning of a time step, the z-direction vapor velocities and axial mesh spacings are used to compute the convective time step limit; the z-direction liquid velocities and transverse velocities for vapor and liquid are ignored, under the assumption that the true convective limit will normally be determined by axial vapor velocities. There are, of course, situations in which this is not the case. We next multiply the convective limit by the parameter clm and call the result $\overline{\Delta t}$. The time step size actually used by the code is then set to the following value:

$$\Delta t = min(dtmax, \overline{\Delta t}) \qquad (A.10)$$

When the user sets $dtmax = dtmin$ the code bypasses the calculations of the convective limit and sets $\Delta t = dtmin$.

Printing occurs at selected time steps as determined by the parameters dtsp and dtlp. These parameters are used to determine the times at which a print is desired. If t_0 represents the time at the beginning of the time zone, then prints should occur at the time $t_0 + k \times dtlp$ for k=1,2,... In fact these times may not correspond to time step boundaries, so the code attemps to print at the time steps nearest the above times. Computation continues in the above manner until the time exceeds "tend" or until the time equals "tend" within a tolerance of 1.e-7 sec. At this point, new values of the above eight quantities are input, defining a new time zone. As many time zones as desired can be used in any one problem. The code will continue the computation as long as a positive value is input for tend. The value tend=0 is always taken to signify the end of the problem, and at this point the code attempts to read data for another problem from the input data file. If tend<0, it means that the code

189

requests a restart. The code will then prompt the user to enter data and will request a new time zone card. For more detailed discussion, see the next section, problem restart.

If at any time step the pressure problem diverges (e.g. negative void fraction), then the code automatically reduces the time step size by a factor of 10 and tries to converge using this smaller time step size. If with this smaller size the code still does not converge, the time step is again reduced. This procedure continues until $\Delta t < dtmin$ or until the number of reductions is greater than iredmx at which point execution is terminated. Of course, if with the smaller time step the code converges, then the calculations continue and the time step size is gradually increased.

For steady-state calculation, the user can request the code to perform "nfeed" times thermal-hydraulics calculations per one time neutronics calculation. This will save computation time since the variation of the thermohydraulic behavior is not as sensitive as that of the neutronic behavior. Here we say one time neutronic calculation means that with the thermohydraulic boundary conditions from the previous time step, the neutronic part of the calculation will iterate until the critical condition obtained at this time step. While one time thermohydraulic calculation means that the pressure solution meets the convergerce criteria but not necessary obtain the steady-state situation.

3 Problem Restart

Several restart options are described in this section. The restart option makes use of external devices to dump and read-in the common blocks. The dump file is automatically created at the end of a run when idump=1.

3.1 Continue Running A Problem

This option is not a real restart option. As described in 2.6, every time the code finishing on time zone, it reads the next time control card. If tend=0, the job is finished, the common blocks are then dumped into dump file. If tend¡0, this means you want the code to continue the same problem by supplying the following two cards through the terminal directly.

3.1.1 Card One

This card is required only if the problem is a transient restart problem, i.e., itrans is not equal to 0 (see 3.2.2.7).

    (i) Variables: itd, nupdat, nedit, nprint, theta1, theta2, error

    (ii) Description: All these variables will be described in detail in 3.2.2.19. Since this card is required only if the problem is a restart transient problem, all the above variables have been defined in the previous time zone, we want to keep this information except itd, we have to set it to be 2. This is an indicator tells the code that now we want to continue the restart transient problem. Therefore, the input for itd should be 2.

3.1.2 Card Two

This card is required whenever you use this option.

    (i) Variables: tend, dtmin, dtmax, dtsp, dtlp, clm, iredmx, nfeed

    (ii) Description: All the above variables have been described in 2.6.

3.2 Restart A Problem

If we have a dump file obtained from the previous calculation, we wnat to restart this problem, two types of restart options are available depend on the value of ntc.

3.2.1 Steady-State restart

This is the case that ntc$=-2$ and ithfbk$=2$. No neutronic transient data for perturbed cross sections are read in. With ntc$=-2$, the job restarts from a previously created dump file.

A number of variables may be changed at a restart option. This is accomplished through use of the FORTRAN namelist input feature. Two namelists are available, the 'restart' and the 'tfdata'.

3.2.1.1 Card One

    (i) Variable: ntc

    (ii) Description:

ntc$= -2$ for this option.

3.2.1.2 Card Two

This card contains the 1st namelist, 'restart', variables.

    (i) Variables: nitmax, iitmax, epsn, epsi, iflash, itb, ibb, hdt, grav, iht, iss, iwft, ivec, idump, itam, ichf, iqss, imixm, imixe, ithfbk, ixavg, idiag, jprinta, jprintb, jprintc, jprintd, jprinte, jprint, irstrt, nitr, ntemp, inton, marmax, epsmar, idpmas, epso, ibc, nbcont

    (ii) Description: All the above variables have been defined except nbcont. We can choose the variables we want to change during the restart calculation, let the input look like

    $restart f1=x1,f2=x2,...$

    where fi is the name of the variable we want to change and xi is the new assigned value.

nbcont= indicator of starting a boron injection (1/2) (beginning of boron injection /otherwise)

    (iii) Example: $restart iht=2$

3.2.1.3 Card Three

This card contains the 2nd namelist, 'tffdata', variables.

    (i) Variables: nb, nt, nq, botfac(i), topfac(i), gfac(i), yb(i), yt(i), yq(i)

    (ii) Description: All the above variables have been defined. We can choose the variables we want to change during the restart calculation, let the input look like

    $tffdata f1=x1,f2=x2,...$

    where fi is the name of the variable we want to change and xi is a single new assinged value or a set of new assigned values.

    (iii) Example:

    $tffdata nb=2,botfac(1)=1,0.7,yb(1)=0,0.2$

3.2.1.4 Card Four

This card is required only if ntemp>0.

    (i) Variables: tinfac, ytemp

    (ii) Description: See 2.3.5.4.

3.2.1.5 Card Five

This card is required only if ntemp>0.

    (i) Variables: nctr

    (ii) Description: See 2.3.5.5.

3.2.1.6 Card Six

This card is required only if ntemp>0.
- (i) Variable: ncit
- (ii) Description: See 2.3.5.6.

3.2.1.7 Card Seven

This is the time control card. Please refer to 2.6.

3.2.2 Transient Restart

This is the case that ntc=−3. A set of neutronic transient data should be read in, which takes care of the perturbed cross sections feedback coefficients. If you set ntc=−3 and ithfbk=2, then the perturbed cross sections feedback coefficients are neglected.

3.2.2.1 Card One
- (i) Variable: ntc
- (ii) Description:

ntc= −3 for this option.

3.2.2.2 Card Two

This card contains the 1st namelist, 'restart', variables. It is exactly the same as described in 3.2.1.2.

3.2.2.3 Card Three

This card contains the 2nd namelist,'tffdata', variables. See 3.2.1.3.

3.2.2.4 Card Four

This card is the same as the card described in 3.2.1.4.

3.2.2.5 Card Five

This card is the same as the card described in 3.2.1.5.

3.2.2.6 Card Six

This card is the same as the card described in 3.2.1.6.

3.2.2.7 Card Seven

This card contains transient indicators.

(i) Variables: itrans, ndpg, nodalt, epsk, epsphi, errorr

(ii) Description:

itrans= Transient type indicator

  = 0: Null transient

  = 1: Control rod transient with Cusping correction

  = 2: Flow transient or temperature transient

  = 3: Only thermohydraulic part calculation is performed

  = 4: Control rod transient without Cusping correction

  = 5: Combined condition 2 and 1

  = 6: Combined condition 2 and 4

ndpg= Number of delayed neutron groups

nodalt= Number of nodes in which rod will move

epsk= See 2.2.12

epsphi= See 2.2.12

errorr= See 2.2.12

3.2.2.8 Card Eight

(i) Variables: v(2,ncomp)

(ii) Description:

v= Group neutron velocity

v(1,j)= Group one neutron velocity $[cm/sec]$

v(2,j)= Group two neutron velocity $[cm/sec]$

3.2.2.9 Card Nine

(i) Variables: beta(ndpg)

(ii) Description:

beta(j)= delayed neutron fraction for group j; j=1,ndpg.

3.2.2.10 Card Ten

(i) Variables: lmda(ndpg)

(ii) Description:

lmda(j)= Precursor delay constant for group j [$1/sec$]; j=1,ndpg.

3.2.2.11 Card Eleven

This card is required only if nodalt is not equal to 0.

(i) Variables: ni(nodalt)

(ii) Description:

ni= The x-direction index of node to be perturbed.

(iii) Example: See Figure A.8.

From this figure, we can see that we have 5 nodes that are going to be perturbed. Therefore,

    nodalt= 5

      ni(1)= 1

      ni(2)= 1

      ni(3)= 1

      ni(4)= 1

      ni(5)= 1

3.2.2.12 Card Twelve

This card is required only if nodalt is not equal to 0.

(i) Variables: nj(nodalt)

(ii) Description:

nj= The y-direction index of node to be perturbed.

(iii) Example: See Figure A.8.

      nj(1)= 1

      nj(2)= 1

      nj(3)= 1

Control Rod

Ch. 1   Ch. 2

z
y
x

* is the node perturbed

Figure A.8 Example of ni,nj,nk.
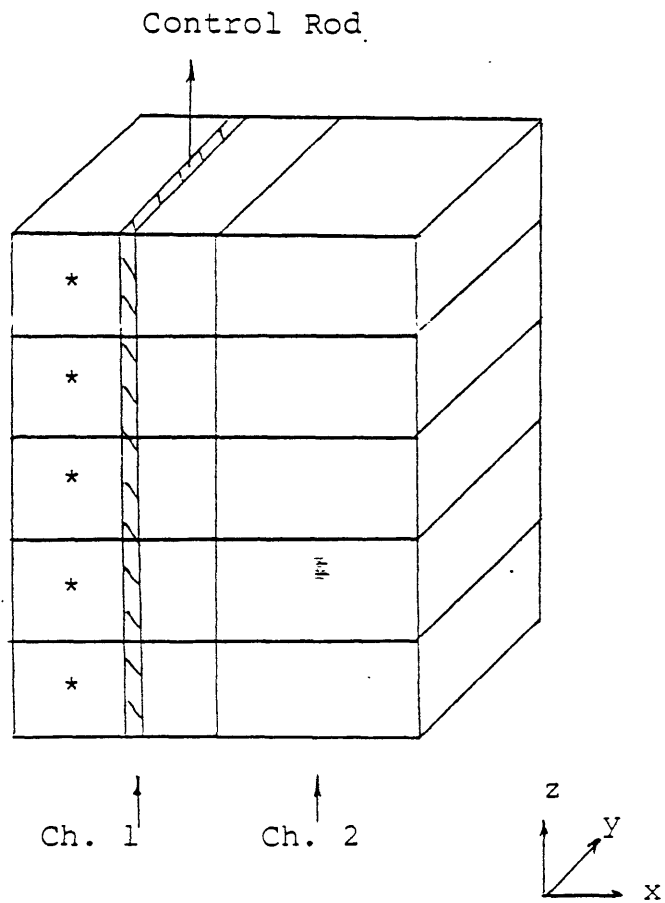
nj(4)= 1

nj(5)= 1

3.2.2.13 Card Thirteen

This card is required only if nodalt is not equal to 0.

 (i) Variables: nk(nodalt)

 (ii) Description:

nk= The z-direction index of node to be perturbed.

 (iii) Example: See Figure A.8.

   nk(1)= 1

   nk(2)= 2

   nk(3)= 3

   nk(4)= 4

   nk(5)= 5

 (iv) Composition Perturbation:

There is another way to use ni, nj, nk, and nodalt to model the perturbed portion of the core. This is so called the "composition perturbation", i.e., once the perturbation is initiated, the nodes which have the same composition (identified by "nk") are perturbed.

If we set:

   ni= 0

   nj= 0,

the code will automatically go to the "composition perturbation" option. Now,

   nk= The compositions to be perturbed

  ndoalt= Number of the compositions to be perturbed

For example, let

   ndoalt= 3

   nk(1)= 2

   nk(2)= 3

   nk(3)= 5,

then, the code will know that the nodes with composition 2 are going to be perturbed at the time tstart(1) (See 3.2.2.14). The perturbation ceases at the time tend(1) (See 3.2.2.15). The same procedure is applied to compositions 3 and 5.

3.2.2.14 Card Fourteen

This card is required only if nodalt is not equal to 0.

(i) Variables: tstart(nodalt)

(ii) Description:

tstart= Time that node perturbation commences [*sec*].

(iii) Example: See Figure A.8.

Now, suppose we have the control rod ejection accident, the control rod in channel will be ejected from the node (1,1,1) all the way out of that channel within 0.1 sec. Therefore, the time that node perturbation commences should be

tstart(1)= 0.

tstart(2)= 0.02

tstart(3)= 0.04

tstart(4)= 0.06

tstart(5)= 0.08

Note, the value 0.02 is an averaged value obtained from 0.1 sec. divided by 5 nodes.

3.2.2.15 Card Fifteen

This card is required only if nodalt is not equal to 0.

(i) Variables: tendq(nodalt)

(ii) Description:

tendq= Time that node perturbation ceases [*sec*].

(iii) Example: See Figure A.8.

tendq(1)= 0.02

tendq(2)= 0.04

tendq(3)= 0.06

tendq(4)= 0.08

tendq(5)= 1.0

3.2.2.16 Card Sixteen

This card is required only if nodalt is not equal to 0.

(i) Variables: dlxs(14,nodalt)

(ii) Description:

dlxs= Actual perturbed cross sections

dlxs(1,j)= Perturbed x-direction group 1 diffusion constant

$$= \Delta D_{1x}$$

dlxs(2,j)= Perturbed group 1 total cross section

$$= \Delta \Sigma_1$$

dlxs(3,j)= Perturbed group 1 scattering cross section

$$= \Delta \Sigma_{21}$$

dlxs(4,j)= Perturbed group 1 $\nu \times$ fission cross section

$$= \Delta \nu \Sigma_{f1}$$

dlxs(5,j)= Perturbed group 1 fission cross section

$$= \Delta \Sigma_{f1}$$

dlxs(6,j)= Perturbed x-direction group 2 diffusion constant

$$= \Delta D_{2x}$$

dlxs(7,j)= Perturbed group 2 total cross section

$$= \Delta \Sigma_2$$

dlxs(8,j)= Perturbed group 2 scattering cross section

$$= \Delta \Sigma_{12} = 0$$

dlxs(9,j)= Perturbed group 2 $\nu \times$ fission cross section

$$= \Delta \nu \Sigma_{f2}$$

dlxs(10,j)= Perturbed group 2 fission cross section

$$= \Delta\Sigma_{f2}$$

dlxs(11,j)= Perturbed y-direction group 1 diffusion constant

$$= \Delta D_{1y}$$

dlxs(12,j)= Perturbed y-direction group 2 diffusion constant

$$= \Delta D_{2y}$$

dlxs(13,j)= Perturbed z-direction group 1 diffusion constant

$$= \Delta D_{1z}$$

dlxs(14,j)= Perturbed z-direction group 2 diffusion constant

$$= \Delta D_{2z}$$

where $\Delta=$ (Cross section after perturbed) - (Cross section before perturbed)

3.2.2.17 Card Seventeen

This card is required only if nodalt is not equal to 0 and ithfbk=3.
  (i) Variables: ee(10,ncomp)
  (ii) Description:

ee= Partial of the perturbed cross sections w.r.t. ρ_m in the perturbed nodes. The definitions of ee's are the same as aa's described in 2.3.6.13 except that ee's are for the perturbed part of the nodes only.

3.2.2.18 Card Eighteen

This card is required only if nodalt is not equal to 0 and ithfbk=3.
  (i) Variables: ff(10,ncomp)
  (ii) Description:

ff= Partial of the perturbed cross sections w.r.t. ρ_m^2 in the perturbed nodes. The definition of ff's are the same as dd's described in 2.3.6.16, except that ff's are for the perturbed part of the nodes only.

3.2.2.19 Card Nineteen

(i) Variables: itd, nupdat, nedit, nprint, theta1, theta2, error

(ii) Description:

itd= Number of time domains. Suppose we have a steady-state dump file which has been created by a steady-state calculation. We want to perform a 2 seconds transient calculation by performing two one second calculations. Two steps should be followed. 1: restart the steady-state dump file with itd=1 for the 1st second calculation. A new dump file is created after this calculation. 2: restart the dump file created in step 1 with itd=2 for the 2nd second calculation.

nupdat= Steps for updating the matrix

nedit= Times for editing the neutronic data. For example, if nedit=2, the code will print the neutronic data once every 2 time steps.

nprint= Print flag for point kinetics omegas (0/1) (No/Yes)

theta1= Flux theta (0.0< to ≤1.0) (Recommend 1.0)

theta2= Delayed theta (0.0< to ≤1.0) (Recommend 1.0)

error= Convergence criteria (Recommend 1.e-3,1.e-4)

3.2.2.20 Card Twenty

This is the time control card. See 2.6.

Steady State Input for Model B

```
1
SLB -PWR 10-Channel Problem-
10 4 10 4 1 10 0
0 2 2 0 1 1 1 2 0 0 0 0 0 0 0 0 -9.81 1.5e-2 1. 0 1
6.512e2
1 4 50 1.e-7 1.e-7 4 1.e-5 1.e-5
3.795e7 .05 0 0 5.36e-3 0.617e-3 0.08255e-3 5.678e3 1.
0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.
1 0 1 0 0 0 1 0 0 0 0 1 3 1 3.2042e-11 0
1 100 1 5 1 3
1.0 .03 1.5
1.e-6 1.e-4 .3
4 5 5 6 4 4 10
4 3 2 1 $ncr
4(0) $indent
10(0.) 3.275e-2 9(1.864e-2) 3.275e-2 9(1.916e-2)
3.275e-2 9(1.864e-2)
10(0.) 6.549e-2 9(3.833e-2) 6.549e-2 9(3.833e-2)
10(0.) 6.549e-2 9(3.833e-2)
10(0.) $arx
4(10(0.))
3.275e-2 9(1.864e-2) 6.549e-2 9(3.833e-2) 6.549e-2 9(3.833e-2)
3.275e-2 9(1.916e-2) 6.549e-2 9(3.833e-2)
3.275e-2 9(1.864e-2) $ary
1.145e-2 10(.6449e-2) 2.29e-2 10(1.311e-2) 2.29e-2 10(1.37e-2)
2.29e-2 10(1.311e-2)
2.29e-2 10(1.311e-2) 4.58e-2 10(2.739e-2) 4.58e-2 10(2.739e-2)
2.29e-2 10(1.37e-2) 4.58e-2 10(2.739e-2)
2.29e-2 10(1.311e-2) $arz
3.504e-3 9(1.937e-3) 7.008e-3 9(4.012e-3) 7.008e-3
9(4.191e-3) 7.008e-3 9(4.012e-3)
7.008e-3 9(4.012e-3) 1.402e-2 9(8.383e-3) 1.402e-2 9(8.383e-3)
7.008e-3 9(4.191e-3) 1.402e-2 9(8.383e-3)
7.008e-3 9(4.012e-3) $vol
10.701e-2 3(21.402e-2) $hx
10.701e-2 3(21.402e-2) $hy
12(30.6e-2) $dz
2.14e-1 11(1.42e-2) 2.14e-1 11(1.48e-2) 2.14e-1 11(1.64e-2)
2.14e-1 11(1.48e-2)
2.14e-1 11(1.48e-2) 2.14e-1 11(1.64e-2) 2.14e-1 11(1.64e-2) 2.14e-1 11(1.64e-2) 2.14e-1 11(1.64e-2)
```

2.14e-1 11(1.48e-2) $hdz
40(0.) $sij
12 01 3(12) 01 3(12) 01 12 $ifwz
10(6(15.65e6) 6(15.5133e6)) $p
120(0.) $alp
10(6(554.) 6(554.66666)) $tv
10(2.5 10(4.138)) $vvz
1 2 3 4 5 6 7 8 9 10 $icr
1.e9 11(15.6e-3) 1.e9 11(15.72e-3) 1.e9 11(16.4e-3) 1.e9 11(15.72e-3)
1.e9 11(15.72e-3) 1.e9 11(16.4e-3) 1.e9 11(16.4e-3)
1.e9 11(16.4e-3) 1.e9 11(16.4e-3)
1.e9 11(15.72e-3) $hdh
10(5(615) 5(620)) $tw
4(1.) 2(0.) $qr
51 4(102) 204 204 102 204 102 $rn
10(1.) $fracp
0 0 0 0 1
1 1 5 4 1 $nplan
1 2 3 4 5 6 7 8 9 10 11 12 $iasn
36(-5)
5(0) -3
0 4(6) -3
0 3(6) -2 -2
0 2(6) 3(-2)
0 6 4(-2)
6(-1)
5(0) -3
0 3 2 3 4 -3
0 2 5 4 2(-2)
0 3 4 3(-2)
0 4 4(-2)
6(-1)
5(0) -3
0 3 2 1 4 -3
0 2 5 4 2(-2)
0 1 4 3(-2)
0 4 4(-2)
6(-1)
36(-4) $irow
1.41049e-0 2.90569e-2 1.64668e-2 4.88185e-3 1.89220e-3
3.89470e-1 9.76471e-2 0 8.64417e-2 3.56800e-2
1.41049e-0 3.89470e-1 1.41049e-0 3.89470e-1
1.38522 2.66655e-2 1.73163e-2 5.80896e-3 2.25200e-3
3.90350e-1 8.92592e-2 0 1.08556e-1 4.48020e-2
1.38522 3.90350e-1 1.38522 3.90350e-1
1.30194 2.69555e-2 1.84013e-2 4.96895e-3 1.92595e-3
3.65436e-1 6.59341e-2 0 8.36471e-2 3.45221e-2
1.30194 3.65436e-1 1.30194 3.65436e-1

1.30711 2.64006e-2 1.73245e-2 6.45220e-3 2.50085e-3
3.71557e-1 8.45212e-2 0 1.25514e-1 5.18011e-2
1.30711 3.71557e-1 1.30711 3.71557e-1
1.37954 2.64986e-2 1.68400e-2 6.52310e-3 2.52830e-3
3.92575e-1 9.84291e-2 0 1.27768e-1 5.27310e-2
1.37954 3.92575e-1 1.37954 3.92575e-1
1.30194 2.69555e-2 1.84013e-2 0 0
3.65436e-1 6.59341e-2 0 0 0
1.30194 3.65436e-1 1.30194 3.65436e-1 $xsref
5(2.714e-18) 0. $xesig
5(.059) 0. $yield
5(2.1e-5) 0. $xelam
5(0.)
2.5641 0 3.2967 1.4286 -1
5(0.)
2.5641 0 3.2967 1.4286 -1
2.5641 0 3.2967 1.4286 -1
5(0.)
2.5641 0 3.2967 1.4286 -1
10(0.)
10(0.)
7.6923 0 9.6154 12.5 -1
10(0.)
7.6923 0 0.9737 12.6582 -1 $alb
1 2 3 4 $ieditx
1 2 3 4 $iedity
1 2 3 4 5 6 7 8 9 10 $ieditz
-1.08573 3.06642e-2 2.59790e-2 1.91031e-3 7.40430e-4
-8.56196e-1 2.74602e-2 0 2.41238e-2 9.95617e-3
-1.30138 3.02963e-2 2.69450e-2 1.76269e-3 6.83213e-4
-9.14690e-1 3.63623e-2 0 2.95809e-2 1.22084e-2
-1.25486 3.11506e-2 2.81636e-2 1.51682e-3 5.87915e-4
-8.87887e-1 3.14391e-2 0 1.85289e-2 7.64709e-3
-1.33066 3.03082e-2 2.71127e-2 1.87574e-3 7.27031e-4
-9.03244e-1 3.78988e-2 0 3.21228e-2 1.32574e-2
-1.31592 3.00060e-2 2.64943e-2 1.97196e-3 7.64326e-4
-9.13192e-1 3.97420e-2 0 3.62959e-2 1.49797e-2
-1.25486 3.11506e-2 2.81636e-2 0 0
8.87887e-1 3.14391e-2 0 0 0 $aa
6.66666e-7 1.04747e-6 9.85333e-7 4.50666e-8 1.74677e-8
8.25734e-5 -2.96487e-5 0 -4.39454e-5 -1.81368e-5
5.20000e-6 9.59733e-7 8.71999e-7 6.12000e-8 2.37209e-8
7.10800e-5 -3.31965e-5 0 -5.12534e-5 -2.11529e-5
6.66666e-7 1.04747e-6 9.85333e-7 4.50666e-8 1.74677e-8
8.25734e-5 -2.96487e-5 0 -4.39454e-5 -1.81368e-5
5.33333e-7 9.23467e-7 8.51999e-7 7.07999e-8 2.72619e-8
7.03600e-5 -3.37351e-5 0 -5.78533e-5 -2.38766e-5
4.26667e-6 9.04534e-7 8.14667e-7 6.99334e-8 2.71060e-8

6.56933e-5 -3.47479e-5 0 -5.67599e-5 -2.41684e-5
6.66666e-7 1.04747e-6 9.85333e-7 0 0
8.25734e-5 -2.96487e-5 0 0 0 $bb
3.20939e-4 2.16235e-5 -1.55548e-5 -3.31630e-7 -1.30477e-7
8.27309e-5 -3.23714e-5 0 -8.43284e-5 -3.48034e-5
3.48040e-4 2.02769e-5 -1.55905e-5 -2.47479e-7 -9.59221e-8
6.81104e-5 -3.08322e-5 0 -9.31099e-5 -3.84120e-5
3.20939e-4 2.16235e-5 -1.55548e-5 -3.31630e-7 -1.30477e-7
8.27309e-5 -3.23714e-5 0 -8.43284e-5 -3.48034e-5
2.95977e-4 2.08946e-5 -1.50512e-5 5.34899e-8 2.07325e-8
5.64852e-5 -3.16388e-5 0 -1.01488e-4 -4.10853e-5
3.33064e-4 2.02641e-5 -1.54193e-5 -9.98477e-9 -3.87007e-9
5.73411e-5 -3.02096e-5 0 -1.00632e-4 -4.15321e-5
3.20939e-4 2.16235e-5 -1.55548e-5 0 0
8.27309e-5 -3.23714e-5 0 0 0 $cc
60(0.) $dd
1 2 1 2 2 2
3.3488e6 6.2e7 10.2518 6.512e5 2.94e7 1.124e6 2.6515 0.5721 555.0
.05 15.46e6 1.93e7
942.34 574.85 7.23188e-1 0
.15 .00001 .05 .05 .05 1. 5 1
-1 0 0 0 0 0 0 0
0.
0.

Steady State Restart Input

```
-2
$restart ixavg=1 iwft=0 itam=0$
$tffdata$
.25 .00001 .05 .05 .05 1. 5 1
0.,,,,,,,,
0.
```

Transient Restart

```
-3
$restart nitmax=10 iitmax=200 epsi=1.e-5 epsn=1.e-4
ichf=2 iss=0 iwft=1 itam=1
ntemp=10 nitr=3 ibc=0 ithfbk=2 ixavg=0 idiag=0 iflash=1$
$tffdata nt=8
topfac(1)=1.000,0.6,0.5,0.4,0.34,0.3,0.22,02
yt(1)=0.,10.,20.,30.,35.,40.,50.,60.
nb=8
botfac(1)=1.000,1.000,1.025,1.06,1.065,1.01,0.738,0.73
yb(1)=0.,10.,20.,30.,35.,40.,50.,60. $
1. 0. 1.005 10. 0.995 20. 0.9703 30. 0.9554 40. 0.9237 50.
0.9108 60. 0.9009 70. 0.8939 80. 0.8861 90.
1. 0. 0.9728 10. 0.9505 20. 0.9257 30. 0.9084 40. 0.8851 50.
0.8752 60. 0.8573 70. 0.8430 80. 0.8291 90.
1. 0. 0.9405 10. 0.9059 20. 0.8811 30. 0.8613 40. 0.8464 50.
0.8395 60. 0.8137 70. 0.7920 80. 0.7721 90.
4 2 4
2 3 4 7 1 6 5 8 9 10
6 1 10 1.e-7 1.e-6 .2
6(1.e8 434782.6) $v
7.38e-1 $beta
.4353 $ lmda
5(1) 5(3) $ni
5(3) 5(1) $nj
2(6 5 4 3 2) $nk
2(0.9 1.9 2.9 3.9 4.9) $tstart
2(1.9 2.9 3.9 4.9 5.9) $tendq
10(1.0855e-1 2.194e-3 -1.9345e-3 -8.71e-5 -3.375e-5
2.4034e-2 3.1713e-2 0. 2.7946e-3 1.1579e-3
1.0855e-1 2.4034e-2 1.0855e-1 2.4034e-2) $dlxs
1 3 100 0 1. 1. 1.e-7
10. .00001 .05 1. 1. 1. 5 1
2 3 100 0 1. 1. 1.e-4
0.,,,,,,,,
0.
```