Purdue University Purdue e-Pubs

Department of Electrical and Computer Engineering Technical Reports Department of Electrical and Computer Engineering

6-1-1987

Optimal Burnable Absorber Assignment for PWR Core Reload Design

Timothy R. Niggel *Purdue University*

Follow this and additional works at: https://docs.lib.purdue.edu/ecetr

Niggel, Timothy R., "Optimal Burnable Absorber Assignment for PWR Core Reload Design" (1987). *Department of Electrical and Computer Engineering Technical Reports*. Paper 569. https://docs.lib.purdue.edu/ecetr/569

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.



Optimal Burnable Absorber Assignment for PWR Core Reload Design

Timothy R. Niggel

TR-EE 87-25 June 1987

School of Electrical Engineering Purdue University West Lafayette, Indiana 47907

CONTENTS

| | LIST OF FIGURES | 11 |
|------|--|-----|
| | LIST OF TABLES | iii |
| | ABSTRACT | iv |
| | 에 가장하는 것 같은 것이 있는 것은 것이 같은 것 같은 것이 있는 것이다. 이 가지 않는 것이 있는 것이 있다. 같은 것은 사람이 있는 것은 것이 같은 것이 있는 것이 같은 것이 같은 것이 같이 있는 것이 같이 있다. 것이 같은 것이 있는 것이 있 | |
| 1. | INTRODUCTION | 1 |
| | 1.1 Existing Method | 1 |
| | 1.2 Objective of Current Work | 6 |
| 2. | BP ASSIGNMENT METHODOLOGY | 7 |
| - | 2.1 Introduction | 7 |
| | 2.2 BOC Linear Programming. | 12 |
| | 2.3 Hi-Lo Algorithm | 20 |
| | 2.4 Power Reconstruction and Case Pruning | 24 |
| | 2.5 Case Depletion and Optimal Selection | 28 |
| | | |
| 3. | APPLICATION TO CORE RELOAD | 30 |
| | 3.1 Introduction | 30 |
| | 3.2 Optimization Results | 34 |
| j. i | | |
| 4. | CONCLUSIONS AND RECOMMENDATIONS. | 46 |
| | LIST OF REFERENCES | 50 |
| | APPENDIX A: SOURCE LISTING | 52 |
| | APPENDIX B: ADDITIONAL INPUT REQUIREMENTS | 69 |
| | APPENDIX C: SAMPLE INPUT LISTING | 71 |
| | APPENDIX D: SAMPLE OUTPUT | 79 |
| | | |

LIST OF FIGURES

| Figure 1. | Optimization Procedure for Existing Reload Design Method | 2 |
|------------|---|----|
| Figure 2. | Improved Burnable Absorber Assignment Algorithm | 8 |
| Figure 3. | Simple Two Region Optimization Problem | 16 |
| Figure 4. | Number of BP Combinations Formed by Hi-Lo Algorithm | 23 |
| Figure 5. | BOC Power Distribution Comparison - Cycle 9, Case 2 | 26 |
| Figure 6. | Reference Loading Pattern for Cycle 9 | 31 |
| Figure 7. | Reference Loading Pattern for Cycle 10 | 32 |
| Figure 8. | Optimal Fuel Loading Pattern for Cycle 9 | 36 |
| Figure 9. | Optimal Fuel Loading Pattern for Cycle 10 | 37 |
| Figure 10. | Optimal Design Pattern for Cycle 9 | 38 |
| Figure 11. | Optimal Design Pattern for Cycle 10 | 39 |

LIST OF TABLES

and the second

والمجريج من بعريدهم م

经济税额 法保险费 计学 医内部 变化的 医闭合的

| TABLE 1. | Optimal BP Lo | ading Re | sults fro | m Existin | ng Metho | d4 |
|----------|---------------|----------|-----------|-----------|----------|--------------|
| TABLE 2. | Comparison of | Loading | Patterns | for Cyc | 10 9 | 41 |
| TABLE 3. | Comparison of | Loading | Patterns | for Cyc | le 10 | |
| TABLE 4. | Maximum Power | Peaking | - Cycle | 9 | | ••••••••• 44 |
| TABLE 5. | Maximum Power | Peaking | - Cycle | 10 | | 45 |

ABSTRACT

A new method has been developed to assign burnable poison loadings in the optimization of Pressurized Water Reactor core reload design. The method utilizes successive linear programming to determine the desired burnable poison loading. The optimum loading is selected after the evaluation of all candidate loadings close to the desired loading. The design method was implemented as a sub-program in the nodal core analysis code SIMULATE. The technique was applied to re-design Commonwealth Edison's Zion Unit-1 cycles 9 and 10. Significant improvements were achieved in cycle length, number of BP rods required, and power peaking. The present work completely automates the core reload design problem, significantly decreasing the time and effort required of the designer.

al par le de la construction de la La construction de la construction d

1. INTRODUCTION

1.1 Existing Method

A method for the optimization of core reload design has been developed and reported by previous researchers.¹ An algorithm based on this method was implemented as a sub-program within the LWR nodal core analysis computer program, SIMULATE-E.² The procedure was essentially divided into two separate optimization processes. The first is to determine the fuel loading pattern that yields the longest cycle. Secondly, the burnable poison (BP) loading is determined to control the core power peaking of the optimal pattern. The complete optimization procedure logic is shown in Figure 1. The optimization problem is made separable in this manner through the use of the Haling depletion.³ By using this constant power depletion, the best loading pattern can be obtained totally independent of the control strategy.

ta parte de Maria de Co

自己的现在分词 医子宫上的 法法的法律法律法律

and the same the second and the second se

The method used in the fuel loading optimization is a direct search technique which examines all possible two assembly exchanges from a user-input base loading pattern. Assembly exchanges are performed which yield an increase in the cycle length while still meeting peaking constraints. This procedure is repeated until the cycle length can no longer be increased by fuel shuffling. At this point, the direct search has been completed and the optimal loading pattern has been identified. The final loading pattern, and thus the 'optimum' cycle length, are



Figure 1. Optimization Procedure for Existing Reload Design Method

strongly influenced by the initial guess loading pattern because the search is not exhaustive, but only a subset of the complete factorial problem.

After the optimal fuel loading has been selected, it remains only to determine the required burnable poison power control. This task is accomplished through a linear programming solution to the second optimization problem. In the direct search algorithm, the objective function was the maximization of the cycle burnup. In this routine, however, the objective was to minimize the difference between the actual power distribution and some 'target' power distribution. For the control to be truly optimum, it would be necessary to have an optimum power shape as a target. This, however, is in itself a fairly significant problem and is not addressed in this work. Instead, the Haling power distribution from the previously determined optimal fuel loading pattern is utilized.

The solution method employs a successive linear programming technique in which the power distribution is represented as a linear function of the burnable absorber loading. This procedure is performed at each burnup step, giving an optimal BP trace throughout the depletion. Typical results of this procedure can be seen in Table 1. In the table, the assembly position is given in terms of the row and column indices of the assembly in the southeast octant of the reactor core (with position 1,1 being the core center). For the current work, as in most core design

- 3 -

TABLE 1. Optimal BP Loading Results from Existing Method

| Cycle | cle Assembly Position | | | | | | | | | |
|--------------------|-----------------------|-----|-------|-----------------|-----|-----|-------|-------|-------|--|
| Burnup (GWD/MT) | (2,2) (4,1) | | (4,3) | (5,4) (6,5) (6, | | | (7,2) | (7,3) | (7,4) | |
| 0.0 | 2.2 | 6.0 | 5.4 | 6.3 | 3.5 | 1.1 | 6.2 | 5.2 | 2.1 | |
| 2.0 | 2.9 | 6.4 | 6.1 | 6.7 | 5.4 | 0.7 | 6.2 | 6.2 | 0.6 | |
| 4.0 | 2.5 | 6.5 | 6.5 | 7.4 | 6.1 | 1.5 | 5.6 | 7.2 | 0.0 | |
| 6.0 | 3.3 | 7.4 | 7.7 | 8.5 | 6.7 | 2.2 | 7.0 | 7.5 | 0.8 | |
| 8.0 | 5.8 | 7.9 | 8.7 | 9.3 | 7.1 | 3.3 | 7.6 | 7.3 | 4.2 | |
| 10.0 | 4.9 | 4.8 | 6.2 | 6.9 | 6.7 | 4.4 | 6.4 | 6.6 | 4.2 | |
| 12.0 | 0.0 | 0.0 | 0.0 | 0.1 | 4.1 | 5.2 | 3.0 | 3.7 | 3.1 | |

work, octant symmetry of the reactor core is assumed.

The major shortcoming of this present method is that the procedure is not completely automated. In fact, as it is now, the code requires a great deal of user interaction in the design process. This requires that the user possess a significant amount of insight into the core design problem. From data such as that in Table 1, the designer must select a BP distribution using available designs. Current design practice permits only multiples of four BP's per assembly. And, of course, this loading is held constant over the entire duration of the cycle. Thus the engineer is faced with the non-trivial task of selecting a BP loading which (subject to the above criteria) best fits the time varying non-realizable BP distributions returned by the code.

Even after this has been accomplished, the designer's work is far from over. To validate the core design loading, the engineer must perform a series of depletion calculations. If, at any point in the depletion, the core power peaking limits are violated, the BP loading must be adjusted and the procedure repeated. This step in the design process, which is basically a trial and error procedure, is by far the most time consuming and laborious. Finally, even when this manual iteration is complete and a BP loading which controls power peaking has been found, it still remains to be seen whether or not it is the 'optimal' BP distribution.

1.2 Objective of Current Work

Basically, the objective of the current work is to completely automate the burnable absorber assignment process. This would make the overall optimization procedure both faster and much easier for the core design engineer. Instead of the code returning a different, non realizable number of BP's for each burnup step, the updated code will return a single BP loading that <u>is</u> physically realizable. In addition, rather than simply accepting the first BP distribution that meets the core peaking constraints, the improved method selects the loading that gives the cycle of greatest length. Finally, the new code is totally automated, requiring no user interaction and a minimum of additional input.

and the second secon

1. B. S. 28

2. BP ASSIGNMENT METHODOLOGY

Read Service and Service

2.1 Introduction

The existing BP assignment method requires the designer to perform a large portion of the design work manually and, therefore, achieves a final reload design which is less than optimal. There are basically two separate processes currently being done manually. The first is a selection of an initial BP loading from the linear programming results. The second is the depletion calculation and subsequent alterations to the initial loading due to power peaking violations. A methodology for automating these two problems and achieving a more nearly optimal design will be discussed in the following chapter. The complete logic flow diagram for the improved BP assignment method is given in Figure 2.

- 7 -

a state of the second state of the

化氯化合物化合物剂

Previously, the core designer was required to use personal intuition to select the initial BP distribution from the non-integer linear programming results. One method used was averaging the BP values returned at the various burnup steps and using the closest available number of BP rods. Another method involved selecting the available BP loading closest to the desired BP loading at the point in the cycle requiring the greatest total number of BP rods. Typically, this occurred at the middle of the cycle, at a cycle burnup of approximately 8 GWD/MT. The basis for the current method is that the BP loading is best chosen from the beginning of the cycle (BOC) results.



n stand an an standard a stand an stand Standard Standard an stand an s

Figure 2. Improved Burnable Absorber Assignment Algorithm



| 1.1 | |
|---------------------------|--------------------------------------|
| н. н. 1919 - П. 1919 | · relative power in node 1 |
| • | |
| 1. s. | . tellet most to seen y |
| | - barmable person rod imerement |
| а . В . с. 1. с | |
| | |
| 19 ^{11 5.00} . | - sodal power peaking limit |
| • | - total sember of feel assemblies |
| | of essenbling with BP rod |
| | |
| BP _k (a |) = BP incding is note to 18 |
| | |
| 10703 | - total sentor of tisdidate Br |
| | 10061040 100000000 |
| 2 ⁶ (n) | - reseastructed power is ande i |
| · | ATCH The construction - |
| P | a relative power in ande i vith |
| 1. T | optimel Br loseins |
| 2. (n) | · relative power in node i with |
| | EP combination # |
| PPROL | I - reconstructed power publics |
| | talerees . |
| PPHIL. | TT . STADLATE power peaking telecase |
| | |

Figure 2. Continued

This can not be proven rigorously because of the complicated physics of the problem, but rather the motivation for this assumption can be seen through the following 'heuristic' rules:

- 1. If the assembly power is too high throughout the cycle (thus the BOC power, P_o , is high), then the BP loading should be increased.
- 2. If the power is too low throughout the cycle (P_o is too low), the BP loading should be decreased.
 - 3. If the power is too low at BOC (P_0 low) and thus too high at the middle of the cycle (MOC), the BP loading should be decreased
- 4. If the power is too high at BOC (P_0 high) and too low at MOC, the BP loading should be increased.

From the above implied correlation between the BOC power and the control requirements throughout the cycle, $(P_0 \text{ high } -)$ increase, P_0 low -> decrease), it appears that the BP assignment can effectively be performed at the beginning of the cycle. Thus, the linear programming procedure need only be executed at BOC, with the actual BP loading determined from these results.

At this point in the procedure, another major difference arises between the old and new methods. Instead of choosing a single initial guess BP loading, the improved code generates all possible BP distributions close to the optimal BP loading using a Hi-Lo algorithm. This algorithm will be discussed in more detail in following sections. Each of these loadings is then depleted to end-of-cycle. If, at any point in the depletion, power peaking constraints are violated, the case is terminated and the next case is depleted. Following the depletion of all cases, the BP loading yielding the greatest cycle length is chosen as the optimum.

2.2 BOC Linear Programming

The objective for the optimal BP loading search is the minimization of the absolute difference between the actual power distribution and the target power distribution. As mentioned previously, an 'optimum' power shape has yet to be determined. In the absence of an optimum shape, the Haling power shape from the optimal fuel loading is used as the target. This target power does possess some inherent advantages. Haling first proposed³ that maintaining a constant power shape throughout the cycle would yield the minimum power peaking for a given fuel loading. Maintaining a constant power distribution throughout the cycle is referred to as the Haling depletion.

The key to the solution of the BP loading optimization problem is the accurate prediction of the nodal relative powers, P_i . If the power distributions were calculated by solving the nodal diffusion equation (which is the standard procedure), the required computation time would be prohibitive. In order to accelerate the solution procedure, a first order perturbation approximation is made of the nodal relative power. This permits the representation of the power as a linear function of the BP loading. This approximation is given below:

 $P_{i} = P_{i}^{0} + \sum_{k=1}^{K} \frac{\partial P_{i}}{\partial BP_{k}} \Delta BP_{k}$

P.,

P⁰i

K

= relative power in node i = initial power in node i with base BP loading $\frac{\partial P_{i}}{\partial BP_{i}}$ = first order approximation of the change in the

power in node i due to addition of one BP rod in position k

- 13 -

∆BP . = difference between the searched BP loading and the base BP loading and

= total number of BP positions in the core.

The first derivative, $\frac{\partial P_i}{\partial BP_i}$, is determined numerically by perturbing the loading and performing standard nodal power calculations. This BP linear approximation of the nodal relative power is actually a very good one. The core-averaged relative difference between the actual power and the estimated power is less than a percent. The accuracy of the first order approximation will be discussed in further detail in the following

sections.

The objective function for the linear programming problem can be written

 $f = \min \sum_{i=1}^{N} | P_i - P_i^{\dagger} |$

where,

8 8

 P_i = actual power in node i P = target power for node i N = total number of nodes (assemblies)in the core.

The problem is subject to the following constraints:

$$P_i \leq P_{1im}$$
 $i = 1, N$

 $BP_{k} \ge 0 \qquad k = 1, K$

where,

P_{lim} = core nodal power peaking limit

 BP_{k} = number of rods in BP position k.

The above three relationship are a statement of the optimization problem for the burnable absorber assignment. However, the objective function requires modification for solution by linear programming methods. The transformation of the three equations into the 'standard LP form' is as follows:

By making use of the following substitution of an independent variable, d_i

$$\mathbf{d}_{\mathbf{i}} = \left| \mathbf{P}_{\mathbf{i}} - \mathbf{P}_{\mathbf{i}}^{*} \right|,$$

the objective function can be re-written as

gián saite a fossit a

$$= \min \sum_{i=1}^{N} d_{i}.$$

Since it is standard practice to write an optimization problem in terms of the <u>maximization</u> of some quantity, the objective function for this problem can easily be transformed as follows:

$$f = max (-\sum_{i=1}^{N} d_i).$$

- 15 -

Since P_i was previously defined as

$$P_{i} = P_{i}^{o} + \sum_{k=1}^{K} \frac{\partial P_{i}}{\partial BP_{k}} \Delta BP_{k}$$

the independent variable d_i can be re-written by

$$d_{i} = | P_{i}^{0} + \sum_{k=1}^{K} \frac{\partial P_{i}}{\partial BP_{k}} \Delta BP_{k} - P_{i}^{*} |.$$

Re-arranging gives

$$d_{i} = | (P_{i}^{O} - P_{i}^{*}) + \frac{K}{\Sigma} \frac{\partial P_{i}}{\partial BP_{k}} \Delta BP_{k} |$$

The absolute value in the objective function is not acceptable for LP solution because it is not a 'linear' function. The objective function needs to be re-formulated using additional inequality constraints. This re-formulation can be shown through the following simple example.

Suppose that the objective of a two region optimization problem is the minimization of the variable Y, subject to the constraint that Y = |X|. This is shown graphically in Figure 3.



Figure 3. Simple Two Region Optimization Problem

In the figure, two functions are plotted; those being Y = X and Y = -X. The function Y = |X| can be represented as Y = X, for $X \ge 0$, and Y = -X, for $X \le 0$. Even though these are equality constraints, this is still unsatisfactory for LP formulation. For a linear programming solution procedure, it is necessary to have the objective function (and constraints) in terms of relationships that are valid for <u>all</u> ranges of possible values. The shaded area of the figure is represented by the inequality constraints $Y \ge X$ and $Y \ge -X$. These two inequalities can actually be represented by the single inequality $Y \ge |X|$. Now this is not exactly the desired function, since it corresponds to the entire region above the curve Y = |X|. However, since the objective function is to minimize Y, the solution procedure would converge to a point <u>on</u> the curve. Thus, the two inequality constraints $Y \ge X$ and $Y \ge -X$ can be used to replace Y = |X| in the LP formulation.

- 17 -

Returning now to the BP optimization problem, d_i can now be re-written as follows:

$$d_{i} \geq [(P_{i}^{o} - P_{i}^{*}) + \sum_{k=1}^{K} \frac{\partial P_{i}}{\partial BP_{k}} \Delta BP_{k}]$$

and,

$$d_{i} \geq -[(P_{i}^{o} - P_{i}^{*}) + \sum_{k=1}^{K} \frac{\partial P_{i}}{\partial BP_{k}} \Delta BP_{k}].$$

The peaking constraint can be re-written making use of the definition of

the nodal power, P.

$$\mathbf{P}_{i}^{O} + \sum_{k=1}^{K} \frac{\partial \mathbf{P}_{i}}{\partial \mathbf{BP}_{k}} \Delta \mathbf{BP}_{k} \leq \mathbf{P}_{1 \text{ im}}.$$

Finally, by replacing the search variable ΔBP_k with

$$\Delta BP_{k} = BP_{k}^{n} - BP_{k}^{n-1}$$

the system of equations can be re-written in their final form (after some re-arranging) as

$$f = \max \left(\begin{array}{c} N \\ -\overline{\Sigma} \\ i=1 \end{array} \right)$$

 $\frac{K}{\sum} \frac{\partial P_{i}}{\partial BP_{k}} BP_{k}^{n} - d_{i}^{n} \leq (P_{i}^{\bullet} - P_{i}^{0}) + \sum_{k=1}^{K} \frac{\partial P_{i}}{\partial BP_{k}} BP_{k}^{n-1} \quad \text{for } i = 1, N$

 $-\frac{K}{\Sigma}\frac{\partial P_{i}}{\partial BP_{k}}BP_{k}^{n}-d_{i}^{n}\leq -(P_{i}^{*}-P_{i}^{0})-\frac{K}{\Sigma}\frac{\partial P_{i}}{\partial BP_{k}}BP_{k}^{n-1} \text{ for } i=1,N$ k=1

$$\frac{K}{\sum_{k=1}^{n} BP_{k}^{n}} \leq (P_{1im} - P_{i}^{0}) + \sum_{k=1}^{K} \frac{\partial P_{i}}{\partial BP_{k}} BP_{k}^{n-1} \qquad \text{for } i = 1, N$$

 $BP_k \ge 0 \qquad \text{for } k = 1, K.$

The linear programming technique used to solve the above problem is the revised simplex method.⁴ This method is currently employed by all commercial LP computer codes. The particular code used in this work is

the IMSL library subroutine ZX3LP.

2.3 Hi-Lo Algorithm

The solution of the preceding mathematical programming problem prescribes the desired BP loading. However, this desired BP loading is not physically realizable and thus not of great value to the designer. A Hi-Lo algorithm is utilized to determine a practical BP loading closest to the optimal BP loading.

The first task performed by the Hi-Lo algorithm is the determination of the high and low values for the number of BP rods to be placed in each poisonable assembly. Since current design allows BP's to be used in multiples of four only, this corresponds to finding the two multiples of four that bracket the desired number of BP's. For example, if the desired number of BP's is 7.2 for a certain position, the high and low values would be four and eight, respectively.

The next step is the formation of all possible combinations of these high and low values at each BP position. Since this is simply a binary decision (high or low) at each BP position, the total number of these combinations would be 2^{K} , where K is the total number of BP positions. The identification of all possible combinations is facilitated by the above fact. To generate these combinations, all one has to do is convert the 2^{K} numbers from 0 to 2^{K} -1 from decimal to binary. This results in a K digit number consisting of nothing but 0's and 1's. In

- 20 -

this representation, a 1 in the i'th position in the binary number corresponds to the placement of the high number of BP's in the i'th core BP position. Similarly, a 0 corresponds to the low number of BP's. Given below is an example of the transformation from decimal to binary to BP loading representation.

- 21 -

| Decima | 1 Bin | Binary | | BP Loading | | | | | | | |
|--------|-------|--------|----|------------|----|-------|------|----|---------|-------|--|
| 100 | 110 | 0100 | Hi | Hi | Lo | Lo | Hi I | Lo | Lo | | |
| e 전화 (| | | | | | 1, 11 | den. | | e siner | te de | |

The number of BP combinations formed is expanded when the desired number of BP's for any position is very close to an available loading. For example, if the desired number of BP's is 7.9 for a particular position, then loadings of 4,8 and 12 BP's in that position are investigated. These additional combinations are formed utilizing a variation of the binary transformation routine described above. Exactly how close the desired number of BP's and the available number of BP's must be is specified by the user. The tolerance used for all of this work and recommended for any subsequent analysis is 0.5 (one half of one BP rod). This tolerance may, however, be adjusted to values ranging from 0 to 2 in order to increase or decrease the number of BP loadings examined. In the instance where no acceptable BP loading is identified, the tolerance should definitely be relaxed. To the other extreme, if an unwieldy number of BP combinations are created, the tolerance may be tightened. The total number of BP combinations is given by

 $N = \sum_{1=0}^{n} \begin{bmatrix} n \\ 1 \end{bmatrix} 2^{K-1}$

where,

- $\begin{bmatrix} n \\ 1 \end{bmatrix} = number of combinations of n items taken 1 at a time$
- K = total number of BP positions in the core

n = number of positions in the core where 'extra' BP loadings are considered.

Utilizing basic statistical theory for the determination of $\begin{bmatrix} n\\1 \end{bmatrix}$, the above equation can be re-written as

$$N = \sum_{1=0}^{n} \frac{n!}{1!(n-1)!} 2^{K-1}.$$

If there are no 'extra' positions (n=0), the previous relationship reduces to 2^{K} , which is the base number of combinations discussed earlier in the section. On the other hand, if every BP position is an 'extra' position (n=K), then the number of combinations swells to 3^{K} . A graphical representation of this relationship is given by Figure 4. The upper limit shown in the figure is simply due to the declared array size in the coding itself. Realistically, this upper limit would probably never be approached due to the relatively high cost of depleting such a large number of cases.





2.4 Power Reconstruction and Case Pruning

The first order perturbation approximation discussed in earlier sections was fundamental to the solution of the optimization problem using a linear programming technique. In the solution procedure, a matrix of linear sensitivity coefficients, $\frac{\partial P_i}{\partial BP_k}$, are calculated numerically. This sensitivity matrix is used to approximate the modal power distribution, as the BP loading is iteratively modified. As an additional benefit this matrix may also be used to estimate the power distribution resulting from each of the candidate BP loadings identified by the Hi-Lo algorithm. Similarly to the relationship given in section 2.2, the nodal power is given by

$$P_{i}^{r}(m) = P_{i}' + \sum_{k=1}^{K} \frac{\partial P_{i}}{\partial BP_{k}} (BP_{k}(m) - BP_{k}')$$

where, $P_i^r(m) = reconstructed power in node i with BP combination m$ $<math>P_i^r = relative power in node i with optimal BP loading$ $BP_k^r = optimal BP loading for node k$ $BP_k(m) = BP loading in node k in combination m$ K = total number of BP positions in the core.

Thus, without a time consuming normal power calculation, a relatively accurate representation of the nodal power distribution may be obtained. A comparison of the reconstructed power and the SIMULATE calculated

- 24 -

power for a particular BP loading is given in Figure 5. Of particular note is the relatively small average absolute difference of 0.9%.

In order to minimize computation time and thus cost, an excessive number of nodal power calculations should be avoided. This can be accomplished by eliminating infeasible cases before the normal power calculations are performed. The current method incorporates one such pruning procedure in its calculational scheme. The pruning parameter currently used is the nodal power peaking. If the maximum power peaking exceeds a preset limit, the case is discarded and the following case is examined.

The preset peaking limit is based on two multiplicative factors: a base nodal power peaking limit, PP_{lim} , and a power peaking tolerance multiplier, PPMULT1. The user input base limit, PP_{lim} , is derived from thermal hydraulic safety considerations and previous core operational data. For this work, a fairly typical value of 1.33 was implemented. The peaking tolerance multiplier is applied to relax the peaking limitations on the reconstructed power distributions. The motivation for this relaxation is the error introduced by the linear approximation made in the power reconstruction. Based on power distribution comparisons for several different burnable poison loadings such as the one given in Figure 5, the overall average percent difference is found to be approximately 0.5% with a standard deviation of about 0.5%. From basic statistical theory, it is 99.7% certain that all data points are

| 0.953 0.982 +3.0 | | | | x.xxx x.xxx +x.x +0.9 | <pre><- Reference Power <- SIMULATE Power <- % Difference <- AVG % Diff</pre> |
|------------------------|------------------------|------------------------|------------------------|--------------------------------|---|
| 1.110 1.139 +2.6 | 1.249 1.278 +2.3 | | | | |
| 1.020 1.040 +2.0 | 1.227 1.249 +1.8 | 1:034 1.045 +1.1 | | | |
| 1.217 1.228 +0.9 | 1.338 1.349 +0.8 | 1.174 1.173 -0.1 | 1.143 1.140 -0.3 | | |
| 0.868 0.872 +0.5 | 1.276 1.279 +0.2 | 1.181 1.178 -0.3 | 1.285 1.278 -0.5 | 1.129 1.118 -0.9 | |
| 0.754 0.752 -0.3 | 1.077 1.074 -0.3 | 1.322 1.311 -0.8 | 1.089 1.081 -0.7 | 1.255 1.242 -1.0 | 1.006 0.994 -1.2 |
| 0.815 0.810 -0.6 | 1.194 1.187 -0.6 | 1.264 1.257 -0.6 | 1.179 1.179 +0.0 | 0.858 0.856 -0.2 | 0.345 0.341 -1.2 |
| 0.301 0.299 -0.7 | 0.387 0.384 -0.8 | 0.422 0.419 -0.7 | 0.311 0.309 -0.6 | | |

Figure 5. BOC Power Distribution Comparison Cycle 9 - Case 2

within range of $\bar{x} + 3\sigma_{x}$. Thus, all reconstructed nodal powers should be within 0.5 + 3(0.5) or 2% of the actual nodal power calculated by SIMULATE. The peak powers ($P_{i} \ge 1.3$) that are of most interest here were predicted slightly more accurately, with all reconstructed peak powers expected to be within 1% of the SIMULATE powers.

It follows, then, that a multiplier of 1.02 would effectively account for the uncertainty in the reconstructed power distributions. This multiplier is also user-input, and may be altered if necessary, but this value is recommended for any subsequent work.

This pruning procedure provides a significant reduction in the number of cases which must be evaluated using the normal power calculations. Equally important, however, is the fact that while many cases are discarded, no potentially successful cases are discarded. The relatively loose tolerance insures that all BP loadings that are even close to being acceptable are passed on to the next step in the selection process.

2.5 Case Depletion and Optimal Selection

The final step in the core design process is the depletion of all the cases that passed the pruning test described in the previous section. This is by far the most time consuming portion of the code, involving numerous power distribution calculations for each depletion. The optimum BP loading is selected after all cases have been depleted to EOC.

For a particular loading pattern to be acceptable, it must meet power peaking constraints throughout the cycle. Thus, in the core design process, the core power peaking must be checked at each step in the depletion. In addition, the burnup steps must be small enough to insure that power peaking violations do not occur between steps. This checking process is included in the code's depletion procedure. If, at any point in the depletion, the power peaking limits are exceeded, the depletion is terminated and depletion of the succeeding case is begun.

Just as in the power reconstruction pruning, the peaking constraint in the core depletion also includes a power peaking tolerance in the form of a multiplier. In this case, the multiplier, PPMULT2, accounts for the error in the SIMULATE power distribution calculations as compared to the actual power distribution. The quantitative evaluation of this error is beyond the scope of this work. For this reason and basic conservatism, a multiplier of 1.00 is used here and recommended in all subsequent work. As it applies here, conservatism means that while some potentially successful cases are discarded, the cases that <u>are</u> deemed acceptable by the code are much more likely to satisfy the requirements for loading into the reactor.

After all candidate BP loadings have been depleted, it remains only to select the optimum loading. This selection is based on the maximization of the cycle length. Thus, of those cases that deplete to EOC, the one yielding the longest cycle is selected as the optimum.

3. APPLICATION TO CORE RELOAD

3.1 Introduction

The improved core reload design method was employed for a re-design of Commonwealth Edison's Zion Unit-1 cycles 9 and 10. The actual loading patterns for both cycles are given in Figures 6 and 7, respectively. Note: The fuel type descriptor given for each assembly is simply a variable used internally by the program to differentiate between assemblies of different design and/or fuel enrichments. The re-design is done utilizing the original fuel assemblies (i.e. same enrichments) ; the difference between the new and the old designs is the loading <u>pattern</u> and the BP loading. Thus, the result of the optimization is the improvement in the cycle length for a given fuel loading. Conversely, the code could also be used to give a desired cycle length using a lower reload enrichment.

The nodal code SIMULATE has been benchmarked for both Zion-1 cycles 9 and 10, and has been shown to yield sufficiently accurate results for design work of this type. All current work has been performed utilizing the core model formulated through benchmarking procedure from previous work.¹ In addition, all core calculations are performed assuming that equilibrium Xenon and Samarium concentrations are present in the core. This approximation is valid at all burnup steps except BOC, at which point these two fission products have not yet reached their saturation

- 30 -

医血压 化合金 化合金合金

이 같은 감독권 것이 있을까?
| | | | | · <u></u> | | |
|-------------------------|--------------------------|--------------------------|---------------------|-----------------|--------------------------------|----------------------|
| 1 2.90 | | | | × ×.×× ×× | <- Fuel <- Enric <- # of | Type Chment BP |
| 20.620 | | | | xx.xxx | <- BOC E | Burnup |
| 4 2.70 12.960 | 5 2.80 20 0.000 | | | | | |
| 3 2.80 | 6 3.20 | 2.70 | | | | |
| 25.880 | 11.410 | 13.100 | | | | |
| 5 2.80 8 0.000 | 4 2.70 12.240 | 5 2.80 12 0.000 | 2 3.20 23.200 | | | |
| 3 2.80 | 6 3.20 | 3.20 | 5 2.80 4 | 2 3.20 | | |
| 25.890 | 7.910 | 23.770 | 0.000 | 21.220 | | |
| 3 2.80 | 3 2.80 | 6 3.20 16 | 3.20 | 2.70 | 5 2.80 | |
| 24.940 | 24.040 | 0.000 | 22.750 | 12.770 | 0.000 | |
| 4 2.70 | 6 3.20 8 | 6 3.20 | 6 3.20 8 | 6 3.20 | 2 3.20 | |
| 9.810 | 0.000 | 8.270 | 0.000 | 0.000 | 31.420 | |
| 1 2.90 | 6 3.20 | 4 3.20 | 3 2.80 | | | |
| 20.620 | 9.410 | 0.000 | 22.630 | | | |

Figure 6. Reference Loading Pattern for Cycle 9

31 -

| 3 2.90 27.530 | | | × × . x x × x × x . x x x | <- Fuel <- Enri <- # of <- BOC | Type chment BP Burnup |
|--|--------------------------|--------------------------|------------------------------------|---|--------------------------------|
| 6 4 2.80 3.20 8 27.200 0.000 | | | | | |
| 2 4 2.80 3.20 13.980 14.320 | 2 2.80 14.670 | | | | |
| 4 1 3.20 2.70 12 0.000 25.060 | 4 3.20 12 0.000 | 1 2.70 24.600 | | | |
| 2 2.80 3.20 10.400 13.060 | 1 2.70 23.160 | 4 3.20 12 0.000 | 1 2.70 25.100 | | |
| 4 2 3.20 2.80 14.990 14.950 | 4 3.20 10.110 | 4 3.20 22.080 | 4 3.20 8 0.000 | 5 3.60 4 0.000 | |
| 4 5 3.20 3.60 16 14.990 0.000 | 4 3.20 21.570 | 5 3.40 8 0.000 | 4 3.20 17.960 | 4 3.20 25.290 | |
| 1 4 2.70 3.20 20.730 10.200 | 5 3.40 0.000 | 2 2.80 15.200 | | | |

Figure 7. Reference Loading Pattern for Cycle 10

32 -

level. However, at BOC, the reactor is just beginning its ascent to full power. By a cycle burnup of 150 MWD/MT, the reactor has reached full power and the fission product inventories have attained their equilibrium values. In addition, at this small core burnup, the burnable poison concentration has not changed appreciably from its BOC value. Thus, performing the core design calculations at BOC while assuming equilibrium Xenon and Samarium is equivalent to designing the core reload at a cycle burnup of 150 MWD/MT. Since it is at just this burnup step that almost all core benchmarking is performed, this seems to be an acceptable approximation. The only possible drawback with this method is the lack of a representation of core behavior at the actual beginning-of-cycle conditions. Power peaking at the actual BOC should not pose a problem, however, since the reactor is operating at such a low power level.

The optimization process consists of two separate stages: the fuel loading search and the burnable poison search. Since only the latter has been altered in this work, only the BP assignment results will be discussed in detail. The results of the fuel loading search will be given for completeness. A more detailed analysis of the fuel loading optimization can be found in reference 1.

- 33 -

3.2 Optimization Results

In order to achieve the longest cycle for a given fuel loading, it is necessary to move as much of the fresh fuel as possible to the core interior. This both decreases neutron leakage by placing less reactive fuel on the core periphery and increases the worth of the fresh fuel by placing those assemblies in areas of 'high neutron importance'. The limiting factor in how much fresh fuel can be moved inboard is the imposed power peaking limit. This peaking can be partially controlled through the use of a proper burnable poison loading.

Figures 6 and 7 illustrate the loading patterns that were actually used in cycles 9 and 10, respectively. In both cycles, a fairly large percentage of the fresh fuel has been placed in the 56 core peripheral positions. In the cycle 9 loading, 28 of the 68 fresh fuel assemblies are on the periphery, while cycle 10 has 20 out of 60. In addition, the cycle 9 loading has 8 once-burned assemblies on the periphery and cycle 10 has 24. Clearly, there is a large amount of highly reactive fuel on the core periphery. Thus, there seems to be a great deal of room for improvement in the achievable cycle length for both cycles.

This presumption is supported by the results from the direct search procedure. By moving much of the reactive fuel away from the periphery, the shuffling procedure obtained an improvement in cycle length of over 1000 MWD/MT. The improved fuel loading patterns are given is Figures 8 and 9. In striking contrast to the actual loadings, the improved cycle 9 loading has 36 <u>twice</u>-burned fuel assemblies on the periphery and cycle 10 now has 44. This yields improvements in both cycle length and pressure vessel neutron fluence. The latter consideration has recently received a considerable amount of attention.

The next step in the optimal design process is the determination of a BP loading capable of controlling the power peaking in the new fuel loadings. Since so much highly reactive fuel was moved inboard, power peaking control becomes much more difficult. The direct search method did, however, apply a power peaking limit to the fuel loading optimization, so a feasible control strategy should exist. In fact, in the the cycle 9 BP search, 31 different loadings were found to be acceptable. Cycle 10, on the other hand, yielded only 2 acceptable BP loadings. The reason for this apparent discrepancy is that the cycle 9 optimal fuel loading had 10 poisonable positions in the octant while cycle 10 had only 9. Thus, many more candidate BP loadings were identified for cycle 9 than for cycle 10 (2304 compared to 1152). The optimal BP loading was selected from these acceptable loadings for both cycles. The optimal BP loadings determined by the code for cycles 9 and 10 are shown in Figures 10 and 11, respectively.

- 35 -

| 1 2.90 | | | | x x x | <- Fuel <- Enric | Type hment |
|-----------|-----------|-----------|-----------|-----------|---------------------|--|
| 20.620 | | | | xx.xxx | <- BOC B | urnup |
| 4 2.70 | 5 2.90 | | | | | |
| 9.810 | 0.000 | | | | | |
| 1 2.90 | а.20 | 3.20 | | | | |
| 20.620 | 11.410 | 23.200 | | | | n 10 - Sili Silisan Silis Silisan Silisan Silisan Silisan |
| 5 2.80 | 6 3.20 | 5 2.80 | 4 2.70 | | | |
| 0.000 | 9.410 | 0.000 | 13.100 | | | |
| 3 2.80 | 6 3.20 | 4 2.70 | 6 3.20 | 2 3.20 | | |
| 25.890 | 7.910 | 12.770 | 8.270 | 21.220 | | |
| 3 2.80 | 4 2.70 | ь 3.20 | 3.20 2 | 3.20 | 5.80 | |
| 24.940 | 12.240 | 0.000 | 22.750 | 0.000 | 0.000 | |
| 2.70 | 3.20 | 5 2.80 | 6 3.20 | 3.20 | 3.20 | |
| 12.960 | 0.000 | 0.000 | 0.000 | 0.000 | 31.420 | |
| 3 2.50 | 3 2.80 | 3.20 | 3 2.80 | | | |
| 25.880 | 22.630 | 23.770 | 24.040 | | | |

Figure 8. Optimal Fuel Loading Pattern for Cycle 9

- 36

| 3 2.90 | | • | | × x.xx | <- Fuel Type <- Enrichment |
|--------------|--------------|-----------|-----------|-----------|---------------------------------------|
| 27.530 | | | | xx.xxx | <- BOC Burnup |
| 2.80 5.80 | 5 3.60 | | | • | |
| 13.980 | 0.000 | | | | |
| 1 2.70 | 5.80 5.80 | 5.80 5 | | | |
| 20.730 | 15.200 | 14.670 | | - | |
| 4 3.20 | 4 3.20 | 4 3.20 | 1 2.70 | | |
| 0.000 | 17.960 | 0.000 | 24.600 | | • • • • • • • • • • • • • • • • • • • |
| 2 2.80 | 4 3.20 | 4 3.20 | 3.20 | 1 2.70 | |
| 10.400 | 13.060 | 14.320 | 0.000 | 24.600 | |
| 4 3.20 | 4 3.20 | 2.80 | 4 3.20 | 5 3.60 | 4 3.20 |
| 14.990 | 10.110 | 14.950 | 10.200 | 0.000 | 0.000 |
| 4 3.20 | 5 3.60 | 5 3.60 | 4 3.20 | 4 3.20 | 2.70 |
| 14.990 | 0.000 | 0.000 | 0.000 | 21.570 | 25.060 |
| 2.80 | 3.20 | 4 3.20 | 1 2.70 | | |
| 27.200 | 22.080 | 25.290 | 23.160 | | |

Figure 9. Optimal Fuel Loading Pattern for Cycle 10

- 37 -

| | | | · · · · · | | | |
|--------------------------|--------------------------|--------------------------|---------------------|-------------------------|--------------------|----------------|
| 1 2.90 | | | | X X.XX | <- Fuel <- Enri | Type chment |
| 20.620 | | | | xx.xxx | <- # 07 <- BOC | Burnup |
| 4 2.70 9.810 | 5 2.80 12 0.000 | | | | | |
| 1 2.90 | 6 3.20 | 2 05.Е | | | | |
| 20.620 | 11.410 | 23.200 | | | | |
| 5 2.80 12 0.000 | 6 3.20 9.410 | 5 2.80 20 0.000 | 4 2.70 13.100 | | | |
| 3 2.80 | 3.20 | 4 2.70 | 6 3.20 | 2 3.20 | | |
| | /.910 | 12.770 | 8.270 | 21.220 | | |
| 3 2.90 24.940 | 4 2.70 12.240 | 5.20 12 0.000 | 2 3.20 22.750 | 6 3.20 8 0.000 | 5 2.80 0.000 | |
| 4 2.70 | 6 3.20 | 5 2.80 4 | 6 3.20 | 6 3.20 | 3.20 | |
| 12.960 | 0.000 | 0.000 | 0.000 | 0.000 | 31.420 | |
| 3 2.80 | 3 2.80 | 2 3.20 | 3 2.80 | | | |
| 25.880 | 22.630 | 23.770 | 24.040 | | | |

Figure 10. Optimal Design Pattern for Cycle 9

- 38 -

| 3 2.90 27.530 | | | | × ×.×× ×× (x.××× | <pre><- Fuel Type <- Enrichment <- # of BP <- BOC Burnup</pre> |
|-------------------------|-------------------------|-------------------------|-------------------------|---------------------------|--|
| 2 2.80 13.980 | 5 3.60 4 0.000 | | | | |
| 1 2.70 20.730 | 2 2.80 15.200 | 2 2.80 14.670 | | | |
| 4 3.20 4 0.000 | 4 3.20 17.960 | 4 3.20 8 0.000 | 1 2.70 24.600 | | |
| 2 2.80 10.400 | 4 3.20 13.060 | 4 3.20 14.320 | 4 3.20 8 0.000 | 1 2.70 24.600 | |
| 4 3.20 14.990 | 4 3.20 10.110 | 2 2.80 14.950 | 4 3.20 10.200 | 5 3.60 4 0.000 | 4 3.20 0.000 |
| 4 3.20 14.990 | 5 3.60 8 0.000 | 5 3.60 4 0.000 | 4 3.20 0.000 | 4 3.20 21.570 | 1 2.70 25.060 |
| 6 2.80 27.200 | 4 3.20 22.080 | 4 3.20 25.290 | 1 2.70 23.160 | | |

Figure 11. Optimal Design Pattern for Cycle 10

- 39 -

Comparisons of three core loading alternatives: the reference (actual) design, the previous 'optimal' design, and the improved design are given in Tables 2 and 3. The superiority of both of the optimal designs as compared to the reference case is apparent. The length of cycle 9 was improved by over 1000 MWD/MT while cycle 10 saw an increase of over 1500 MWD/MT. Also, the number of BP rods necessary for peaking control was decreased. This decrease was quite drastic in the cycle 10 design, where the number of BP's was more than cut in half. The focus of this work, however, is the improvement of the optimization procedure, which is evident by comparison of the new and old 'optimal' results.

The new optimal design of cycle 9 gave a slight improvement in both the achievable cycle length and the number of BP's required as compared to the old optimal design. The cycle was lengthened by about 15 MWD/MT while using 32 fewer BP rods. This improvement also coincided with a small reduction in the cycle maximum power peaking. In fact, the old optimal design would not have been considered acceptable by the present design method due to peaking violations.

At first it may appear that the old optimal design for cycle 10 is superior to the newer design in that it results in an identical cycle length while using slightly fewer BP rods. However, while the old design does give an equal cycle length, the power peaking is significantly higher than the new optimal design.

40 -

TABLE 2. Comparison of Loading Patterns for Cycle 9

| | Reference | OLD | NEW |
|-----------------------|-----------|--------|--------|
| Cycle Length (GWD/MT) | 11.944 | 12.964 | 12.980 |
| Total Number of BP's | 496 | 480 | 448 |
| Max. Power Peaking | 1.3146 | 1.3342 | 1.3253 |

Loading Pattern

TABLE 3. Comparison of Loading Patterns for Cycle 10

| | Loading Pattern | | | | |
|-----------------------|-----------------|--------|--------|--|--|
| | Reference | OLD | NEW | | |
| Cycle Length (GWD/MT) | 11.638 | 13.241 | 13.241 | | |
| Total Number of BP's | 544 | 256 | 288 | | |
| Max. Power Peaking | 1.3420 | 1.3457 | 1.3295 | | |

A more detailed comparison of the power peaking for both cycles is given in Tables 4 and 5. The cycle 9 results show that the only peaking violation of the old optimal design occurs at the beginning-of-cycle. As was mentioned earlier, peaking is not an important consideration at BOC. This, coupled with the fact that the power peaking does not greatly exceed the limit, would lead to the conclusion that the old optimal design would, in fact, be acceptable. Even so, the peaking in the new design is superior to both the old optimal design and even the reference design.

From the cycle 10 results, it appears that the old optimal design exceeded power peaking limits for nearly half of the cycle. Clearly, this is an unacceptable design. Even the reference design violates the peaking limit of 1.33, although this violation occurs at BOC (similar to the old optimal design of cycle 9). Just as in the cycle 9 design, the new optimal design for cycle 10 is far superior to the old optimal design, with a lower power peaking factor at nearly every burnup step.

- 43 -

TABLE 4. Maximum Power Peaking - Cycle 9

.

1 Jacob

| Cycle | Loa | ding Patter | 0 . |
|--------------------|-----------|-------------|------------|
| Barnap (GWD/MT) | Reference | OLD | NEW |
| 0.0 | 1.314 | 1.334 | 1.309 |
| 1.0 | 1.315 | 1.309 | 1.291 |
| 2.0 | 1.309 | 1.299 | 1.284 |
| 3.0 | 1.300 | 1.294 | 1.297 |
| 4.0 | 1.283 | 1.296 | 1.301 |
| 5.0 | 1.283 | 1.300 | 1.309 |
| 6.0 | 1.293 | 1.321 | 1.317 |
| 8.0 | 1.284 | 1.318 | 1.325 |
| 10.0 | 1.296 | 1.305 | 1.314 |

and the state

TABLE 5. Maximum Power Peaking - Cycle 10

| Cycle | Loading Pattern | | | | | |
|--------------------|-----------------|----------------|----------------|--|--|--|
| Barnap (GWD/MT) | Reference | OLD | NEW | | | |
| 0.0 | 1.342 | 1.346 | 1.330 | | | |
| 1.0 | 1.297 | 1.341 | 1.323 | | | |
| 2.0 | 1.273 | 1.336 | 1.325 | | | |
| 3.0 | 1.256 | 1.332 | 1.324 | | | |
| 4.0 | 1.249 | 1.325 | 1.319 | | | |
| 5.0 | 1.251 | 1.320 | 1.313 | | | |
| 6.0 | 1.254 | 1.316 | 1.315 | | | |
| 8.0 10.0 | 1.258 1.257 | 1.310 1.300 | 1.312 1.305 | | | |

4. CONCLUSIONS AND RECOMMENDATIONS

Conclusions

With the completion of the present work, there now exists a completely automated method for optimizing the burnable poison loading for a core reload design pattern. It should be stressed that the method is capable of performing <u>reload</u> design only. The code is not designed to be used for startup cycle design. This is due to the fact that in the startup core, all of the assemblies are 'fresh' and available for BP loading. The code is simply not capable of handling such a large number of poisonable assemblies.

The improved method will save the core designer a great deal of time and effort. Perhaps even more significantly, the design work can now be performed by individuals not having a tremendous amount of insight or experience in the core reload design problem. This makes the code ideal for utilities desiring to become vendor-independent in the designing of their own reload cores.

The major benefit of the improved BP assignment method is not in lengthening the cycle, for it is becoming more and more evident that the BP loading has very little effect on the achievable cycle length. Its

- 46 -

greatest worth is in determining a burnable poison loading that makes possible the safe operation of the optimal fuel loading pattern. Also, the improved method investigates a far greater number of alternative loading strategies and therefore increases the likelihood that the safest and most economical scenario is chosen.

Recommendations

It has been shown in previous work¹ that the control strategy has but a minor effect on the achievable cycle length. This theory is further supported by the current work. The difference between the longest and the shortest cycles that were deemed as acceptable by the code is less than 40 MWD/MT for both cycles 9 and 10. This would seem to lead to the conclusion that the selection of the optimal BP loading on the basis of cycle length is not such a viable alternative. Other possible alternatives for the objective function include: the minimization of the total number of BP rods and the minimization of the power peaking. The minimization of the total number of BP rods in the core would obviously lower fuel cycle costs by reducing the cost of fuel fabrication. This would also lead to an lessening of the reactivity penalty due to BP residue. The minimization of the core power peaking gives no explicit benefit in terms of cycle length or fuel costs. It does, however, give rise to a larger margin of safety, which could ultimately outweigh either of the above benefits.

The entire foundation of the burnable absorber assignment method is based on the attempt to achieve some desired power distribution. Theoretically, this target power shape is the optimum power shape. Since this optimum is currently unknown, a Haling power distribution was used as a target in the present method. Unfortunately, however, this is most probably <u>not</u> the optimum power shape. The method would be improved if the true optimum power distribution could be determined. This task is currently being undertaken here at Purdue.

The current method provides an adequate tool for optimized reload design utilizing discrete burnable absorber rods. This is by far the most common means of power peaking control in today's reactors. However, the inherent advantages of other types of absorbers, such as gadolinia and IFBA may soon lead to the replacement of the discrete absorber rod. Therefore, it would be advisable to examine the possibility of incorporating the use of these other types of absorbers into the core reload design method.

LIST OF REFERENCES

 Kim, Y. IJ., 'Optimization of Core Reload Design for Low-Leakage Fuel Management in Pressurized Water Reactors,' Ph. D. Thesis, Purdue University, 1986.

- 50 -

- 2. Ver Planck, D., 'SIMULATE-E: A Nodal Core Analysis Program for LWR's; Computer Code Users's Manual,' EPRI-NP-2792-CCM, 1983.
- Haling, R., 'Operating Strategy for Maintaining An Optimum Power Distribution Throughout Life,' Proc. ANS Topl. Mtg.; Nuclear Performance Power Reactor Cores, TID-7672, 1964.
- Reklaitis, G. V. et. al., 'Engineering Optimization: Methods and Applications,' John Wiley and Sons, New York, 1983.

APPENDICES

APPENDIX A: SOURCE LISTING

The following partial source listing gives all of the major routines that were either created or altered in this work. Excluded are routines

1. A. 4

with minor or cosmetic changes.

| 1SUBROUTINE PA4CHARACTER*81DIMENSION JOB1LOGICAL DEPL(9DIMENSION IBP0TEMP=0.01ITEMP=0.01ITEMP=0.01ITEMP=0.02NBPERM=13DEPL(1)=.TRUE4MOPTIM=A1(46)5IMABWR=MODES(6IDESUG=07IFIRST=.TRUE.8CALL GSSNDMI9CALL TIMER (2)9CALL TIMER (2)9CALL TIMER (2)9CALL TIMER (2)9CALL TIMER (2)9CALL PAGES (2)9ISTEP=09FIRST=1.010FORMAT (~45X)10FORMAT (~45X)111010FORMAT (~45X)11C11DETERMINE THING12INES=LINES-1141010FORMAT (~45X)11C11DETERMINE THING12C141010FORMAT (~45X)14IC10FORMAT (~45X)14IC10FORMAT (~45X)14IC10FORMAT (~45X)15IF (MODEB(5))16C17IF (MODEB(5))18NC=019C10S211ITERS=012ISCHAL NEXTPT14IC15IF (MODEB(1 | RTB PAR ADATE, BDATE, IHTIME, JHTIME F20 NAM(2) BLO SICNTRL, IFIRST, LEUELU, LEUELC OPL S0000, KPFRST OPL (S0000, 20), IRGH(20), JCGL(20), EXP(8,8), CGNC(6,8,8) OPL OPL OPL 2) PAR HPARTB,2) PAR NM PAR SED(1) PAR IN PAR PAR | RTOCK USSUSSERENT RTOCK USSUSSERENT RECREATE |
|--|--|--|
| 4 CHARACIER®S 1 DIMENSION JOB 1 LOGICAL IVOID 8 LOGICAL DEPL(9 DIMENSION IBP 0 TEMP=0.0 1 ITEMP=0 2 NBPERM=1 3 DEPL(1)=.TRUE 4 MOPTIM=AI(46) 5 IMABUR=MODE3(6 IDEBUG=0 7 IFIRST=.TRUE. 8 CALL GSSNDMI 9 CALL TIMER (5) 10 LAUSED(2)=LAL 10 CALL TIMER (5) 11 IEOL=0 12 DEP=DE 13 DEP=DE 14 IEOL=0 15 IBRN=IBURN 16 DEP=DE 17 FIRST=1.0 18 CALL PAGES (1) 10 FORMAT (~45X 11 IO FORMAT (~45X 12 IVOID=.FALSE 13 LF (MODE3(5) 14 IO FORMAT (~45X 15 IF (MODE3(5) 16 IF (MODE3(5) | HURLEY BURTERST, LEVELU, LEVELC SO000), KPFRST (S0000,20), IRGW(20), JCGL(20), EXP(8,8), CONC(6,8,8) OPU OPU (S0000,20), IRGW(20), JCGL(20), EXP(8,8), CONC(6,8,8) OPU OPU (PAR OPU (PAR PAR PAR PAR PAR PAR PAR PAR | OKUSUSUSUSERRUCERRERERERERERERERERERERERERERERERER |
| LOGICAL IVOID B LOGICAL DEPL(DIMENSION IBP TEMP=0.0 I ITEMP=0 S NBPERM=1 DEPL(1)=.TRUE MOPTIM=A1(46) I IDEBUG=0 I IDEBUG=0 I IFIRST=.TRUE. B CALL OSSNDML C ALL TIMER (S CALL JIMER (S CALL TIMER (S CALL TIMER (S CALL TIMER (S) C INITIALIZATIO C INES=LINES I INES=1.0 S IBRN=IBURN I IOFORMAT (~45X C DETERMINE THI C INITES=LINES=0 I INITES=LINES=1 I IOIDE.FALSE I IOIDE.FALSE I IOIDE.FALSE I IOIDE.S I IF (MODEB(S). C THE HALING I C INEED FIN C ITHE END OF L C INEED S I INITES=0 S I ISRCH=0 S I ISRCH=0 S I ISRCH=0 S I IF (MODEB(S) C PUT ONLY THE C INEED = NSTA C ONTROL ITER S I CONTROL ITED S I C C PUT ONLY THE C ALL NEXTPT C CALL NEXTPT C CALL NEXTPT D C C INITERS=1 C CONTROL ITED OC INITERS=1 C CONTROL ITED OC I IDSC=0 S I IDSC | <pre>,ICHTRL, IF IRST, LEVELU, LEVELU S0000), KPFRST (S0000,20), IROW(20), JCOL(20), EXP(8,8), CONC(6,8,8) OPU OPU 2) #F('LEN=',15, 'MSG=','** IN PARTB **') SHPARTB,2) #FED(1) N PAR PAR PAR PAR PAR PAR PAR PAR PAR PA</pre> | USUSUSE BEELERERERERERERERERERERERERERERERERERER |
| 9 DIMENSION IBP 9 DIMENSION IBP 0 TEMP=0.0 1 ITEMP=0.0 1 ITEMP=0.0 2 NBPERM=1 3 DEPL(1)=.TRUE 4 MOPTIM=A1(46) 5 IMABWR=MODE3(6 IDEBUG=0 7 IFIRST=.TRUE. 8 CALL GSSNDMI 9 CALL TIMER (5) 10 LAUSED(2)=LAL 11 IEOL=0 12 DEP=DE 13 TIMXE=0.0 14 IEOL=0 15 IBRN=IBURN 16 IEOL=0 17 FIRST=1.0 18 CALL PAGES (2) 10 FORMAT (/45X 10 FORMAT (/45X 10 FORMAT (/45X 11 IO 10 FORMAT (/45X 11 IO 12 DETERMINE TH 14 IO 10 FORMAT (/45X <td>(S0000,20), IROW(20), JCOL(20), EXP(8,8), CONC(5,8,8). OPU OPU OPU OPU OPU OPU OPU OPU</td> <td>USUUUUSRRRCRRRRRRRRRRRRRRRRRRRRRRRRRRRR</td> | (S0000,20), IROW(20), JCOL(20), EXP(8,8), CONC(5,8,8). OPU OPU OPU OPU OPU OPU OPU OPU | USUUUUSRRRCRRRRRRRRRRRRRRRRRRRRRRRRRRRR |
| 0 TEMP=0.0 1 ITEMP=0.0 2 NBPERM=1 3 DEPL(1)=.TRUE 4 MOPTIM=A1(46) 5 IMABUR=MODES(6 IDEBUG=0 7 IFIRST=.TRUE. 8 CALL GSSNDMID 9 CALL TIMER (2)=LAL 10 LAUSED(2)=LAL 11 IEOL=0 12 DEP=DE 13 TIMXE=0.0 14 NEC=0 15 IBRN=IBURN 16 INES=LINES-1 17 IF (MODEB(5). 18 IF (MODEB(5). 19 C 10 FORMAT (-45X 11 IF (MODEB(5). 14 IC 10 FORMAT (-45X) 11 I | OPL 2) PAR PA | USUSERRCTERRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR |
| 2 NBPERM=1 3 DEPL(1)=.TRUE 4 MOPTIM=A1(46) 5 IMABUR=MODE3(6 IDEBUG=0 7 IFIRST=.TRUE. 8 CALL GSSNDME 9 CALL TIMER (5) 10 LAUSED(2)=LAL 11 IEOL=0 12 DEP=DE 13 TIMXE=0.0 14 IEOL=0 15 DEP=DE 16 IERN=IBURN 17 FIRST=1.0 18 CALL PAGES (1) 10 FORMAT (/45X 11 IO FORMAT (/45X 12 DETERMINE TH 14 IO FORMAT (/45X) 15 IF (MODE3(5)) 16 IF (MODE3(5)) 17 IF (MODE3(5)) 18 NC=0 19 NU=0 10 SE 10 THE HALING I 14 IC THE END OF L 14 IC THE END OF L 15 | ID ID 2) PAR 10) PAR 3 PAR 10) PAR 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) PAR | UURRESERERERERERERERERERERERERERERERERER |
| A MOPTIM=A1(46) 4 MOPTIM=A1(46) 5 IMABUR=MODE3(6 IDEBUG=0 7 IFIRST=.TRUE. 9 CALL GSSNDMI 9 CALL TIMER (5) 90 LAUSED(2)=LAL 91 IEOL=0 92 CALL TIMER (5) 93 CALL TIMER (5) 94 NEC=0 95 IBEN=IBURN 96 ISTEP=0 97 FIRST=1.0 98 LINES=LINES-1 99 LINES=LINES-1 90 HRITE (ITAPO 91 10 FORMAT (~45X 93 LINES=LINES-1 94 NETE 95 CALL PAGES (1) 95 IF (MODEB(5)) 95 IF (MODEB(5)) 95 IF (MODEB(5) 95 ISC 957 CALL NEXTPT 958 IF (MODEB(1) 957 CALL NEXTPT 958 CALL NEXTPT </td <td>2) PAR PAR PAR PAR PAR PAR PAR PAR</td> <td>USTBEERRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR</td> | 2) PAR PAR PAR PAR PAR PAR PAR PAR | USTBEERRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR |
| 5 IMABUR=MUDES(6 IDEBUG=0 7 IFIRST=.TRUE. 8 CALL JIMER (5) 9 CALL TIMER (5) 10 IOP=DE 10 FIRST=1.0 33 CALL PAGES (1) 10 FIRST=1.0 33 CALL PAGES (1) 10 FIRMAT (*45X) 11 10 10 FIRMAT (*45X) 11 TO 12 DETERMINE THINE 13 CALL PAGES (1) 14 IO 10 FORMAT (*45X) 11 INTERS=1.0 12 INTERS=1.0 13 IF (MODEB(5)) 14 IC 15 IF (MODEB(5)) 16 ITERS=0 17 IF (MODEB(1)) </td <td>2) PAR PAR PAR PAR PAR PAR PAR PAR</td> <td>RRURRERERERERERERERERERERERERERERERERER</td> | 2) PAR PAR PAR PAR PAR PAR PAR PAR | RRURRERERERERERERERERERERERERERERERERER |
| 7 IFIRST=.TRUE. 8 CALL OSSNDMI 9 CALL TIMER (S) 9 DEP=DE 10 DEP=DE 11 IEOL=0 12 DEP=DE 13 TIMXE=0.0 14 IEOL=0 15 IBRN=IBURN 16 ISTEP=0 17 FIRST=1.0 18 CALL PAGES (C) 19 LINES=LINES- 10 FORMAT (>45X 11 IO FORMAT (>45X 12 INES=LINES- 141 10 FORMAT (>45X 15 INES=LINES- 16 DETERMINE THIC 17 IF (MODEB(S). 18 IF (MODEB(S). 14 IC THE HALING I 15 IF (MODEB(S). 16 | IF('LEN=', 15, 'MSG=', '** IN PARTB **') SMC SHPARTB,2) PAR ISED(1) PAR IN PAR PAR PAR | CRTBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB |
| B CALL IMER (S) G CALL IMER (S) O LAUSED(2)=LAL C INITIALIZATIO C INITIALIZATIO C INITIALIZATIO C INITIALIZATIO C INITIALIZATIO C IEOL=0 BEP=DE IEOL3 C IEOL=0 BEP=DE IEOL3 C IEOL=0 C DATES C DETERMINE TH | HPARTB, 2) ISED(1) IN PAR PAR PAR PAR PAR PAR PAR PAR | KRTRERERERERERERERERERERERERERERERERERER |
| C LAUSED(2)=LAL C INITIALIZATIC C INITIALIZATIC C INITIALIZATIC C IEOL=0 B2 DEP=DE C IEOL=0 B3 TIMXE=0.0 B4 NEC=0 B5 IBRN=IBURN B6 ISTEP=0 B7 FIRST=1.0 B8 CALL PAGES (1) B9 LINES=LINES=1 A0 WRITE (ITAPO A1 10 FORMAT (>45X C DETERMINE THIC C THE HALING I C THE HALING I C THE HALING I C THE HALING I SS | IN PAR IN PAR PAR PAR .0) IVOID=.TRUE. PAR .0) IVOID | RTTB B RRTB RRTTB B RRTTB B B B B B B B |
| C INITIALIZATIO C INITIALIZATIO C IEQL=0 IE | IN PAR PAR PAR .0) IVOID=.TRUE. PAR .0) IVOID=.TRUE. PAR .0) IVOID=.TRUE. PAR .0) ICNTRL=.TRUE. PAR <td>RTB RTB RTB RTB RTB RTB RTB RTB RTB RTB</td> | RTB RTB RTB RTB RTB RTB RTB RTB RTB RTB |
| IEQL=0 IEQL=0 IEQ | 10) 3 7,10) 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) PA PA PA PA PA PA PA PA PA PA | RRTBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB |
| Image Image <thimage< th=""> Image <thi< td=""><td>10) 3 7 10) 3 7 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) PA PA PA PA PA PA PA PA PA PA</td><td>RTB BRARTB BRARTB BRARTB BRARTB BB BRARTB BB BB BB BB BB BB BB BB BB BB BB BB B</td></thi<></thimage<> | 10) 3 7 10) 3 7 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) PA PA PA PA PA PA PA PA PA PA | RTB BRARTB BRARTB BRARTB BRARTB BB BRARTB BB BB BB BB BB BB BB BB BB BB BB BB B |
| 34 NEC=0 35 IBRN=IBURN 36 ISTEP=0 37 FIRST=1.0 38 CALL PAGES () 39 LINES=LINES=1 40 WRITE (ITAPO 41 10 FORMAT (| 10) PAI 3 PAI 7,10) PAI , 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) PA PAI PAI GE.3) ICNTRL=.TRUE. PAI IFE EXPOSURE SEARCH. PAI <t< td=""><td>RTB RTB RTB RTB RTB RTB RTB RTB RTB RTB</td></t<> | RTB RTB RTB RTB RTB RTB RTB RTB RTB RTB |
| 36 ISTEP=0 37 FIRST=1.0 38 CALL PAGES () 39 LINES=LINES- 40 WRITE (ITAPO 41 10 FORMAT (./45X C DETERMINE THI C DETERMINE THI 42 IUOID=.FALSE 43 IF (NUMAX.GT 44 ICNTRL=.FALSE 45 IF (MODEB(5). 46 IF (MODEB(5). 47 IF (MODEB(5). 48 NC=0 49 NU=0 50 C 52 ITERS=0 53 ISRCH=0 154 LPMEM = NPAR 155 LNEED =NMACRO 156 IF (MODEB(1)) 157 CALL NEXTPT 158 LNEED = NSTF 159 CALL NEXTPT 150 | 10) 3 7,10) , 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) PA E ITERATION LEVELS INVOLVED IN THS CASE. PA .0) IVOID=.TRUE. E. GE.3) ICNTRL=.TRUE. .GT.5) ICNTRL=.TRUE. .GT.5) ICNTRL=.FALSE. TERATION USES THE CONTROL ITERATION LEVEL DURING PA PA PA PA PA PA PA PA PA PA | RATE RATE RATE RATE RATE RATE RATE RATE |
| 37 FIRSTEIL0 38 CALL PAGES () 39 LINES=LINES=() 39 LINES=LINES=() 40 HRITE (ITAPO) 41 10 FORMAT (>45X) 42 IUOID=.FALSE 43 IF (NUMAX.GT) 44 ICNTRL=.FALSE 45 IF (MODEB(5)). 46 IF (MODEB(5)). 47 IF (MODEB(5)). 48 NC=0 50 NS=0 51 XL=1.0 52 ITERS=0 53 ISRCH=0 54 LPMEM = NPAR 55 LNEED=NMACRO 56 IF (MODEB(1)) 57 CALL NEXTPT 58 LNEED = NSTF 595 CALL NEXTPT 561 LPH=LPMEM 575 CALL NEXTPT 586 IF (CONTROL ITEL 595 CALL NEXTPT 595 CALL NEXTPT 596 ICONTROL ITEL 597 CONTROL ITEL 598 IDSC=0 5 | 10) PAI 3 PAI 7,10) PAI 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) PA 6 ITERATION LEVELS INVOLVED IN THS CASE. PA 0) IVOID=.TRUE. PA 6 .0) IVOID=.TRUE. PA .0) ICNTRL=.TRUE. PA .0) ICNTRL=.FALSE. PA .0.1 .0) ICNTRI =.TRUE. .0.1 .0) .0) .0) | ARTB ARTB ARTB ARTB ARTB ARTB ARTB ARTB |
| 29 LINES=LINES-1 40 WRITE (ITAPO 41 10 FORMAT (/45X C DETERMINE THI C IVOID=.FALSE 43 IF (NUMAX.GT 44 ICNTRL=.FALSE 45 IF (MODEB(5)). 46 IF (MODEB(5)). 47 IF (MODEB(5)). 48 NC=0 49 NU=0 50 NS=0 51 XL=1.0 52 ITERS=0 53 ISRCH=0 54 LPMEM = NPAR 55 LNEED=NMACRO 56 IF (MODEB(1)) 57 CALL NEXTPT 58 LNEED = NSTF 59 CALL NEXTPT 58 LNEED = NSTF 59 CALL NEXTPT 50 NITERS=1 506 LPMEMEM 507 CONTROL ITEL 598 CONTROL ITEL 599 CALL NEXTPT 5061 LPMEPMEM 5062 IOSC=0 5063 IOSC=0 <td>3 PA T,10) 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) PA • 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) PA • 100 INTERTION LEVELS INVOLVED IN THS CASE. PA • 0) IVOID=.TRUE. PA • 0, ICNTRL=.TRUE. PA • 0, ICNTRL=.FALSE. PA • 1FE EXPOSURE SEARCH. PA • 0, ICNTRI=.TRUE. PA</td> <td>ARTB ARTB ARTB ARTB ARTB ARTB ARTB B B ARTB B B B B B B B B B B B B B B B B B B</td> | 3 PA T,10) 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) PA • 44HBEGIN COUPLED NUCLEAR - HYDRAULIC ITERATIONS) PA • 100 INTERTION LEVELS INVOLVED IN THS CASE. PA • 0) IVOID=.TRUE. PA • 0, ICNTRL=.TRUE. PA • 0, ICNTRL=.FALSE. PA • 1FE EXPOSURE SEARCH. PA • 0, ICNTRI=.TRUE. PA | ARTB ARTB ARTB ARTB ARTB ARTB ARTB B B ARTB B B B B B B B B B B B B B B B B B B |
| 4110FORMAT (/45XCDETERMINE THICCUOID=.FALSE43IF (NUMAX.GT44ICNTRL=.FALSE45IF (MODEB(S).46IF (MODEB(S).47IF (MODEB(S).48NC=049NU=050NS=051XL=1.052ITRCH=053ISRCH=054LPMEM = NPAR55LNEED=NMACRO56IF (MODEB(I)57CALL NEXTPT58LNEED = NSTF595CALL NEXTPT561LPH=LPMEM575CONTINUE586IF (MODES(I)597CALL NEXTPT598CALL NEXTPT598LNEED = NSTF595CALL NEXTPT595CALL NEXTPT596IDSC=0597CONTINUE598IDSC=0599CALL NEXTPT590CONTINUE591CONTINUE592IDSC=0593IDSC=059430595CONTROL ITEL595CONTROL ITEL595IDSC=0596SIC=XKEFF597LEUELC=.FAL598IDSC=0599SIC=3590IBPAS=0591IDSC=1592IDSC=1593IDSC=1594IDSC=1595IDSC=1595IDSC=1595IDSC=1596IDSC=1 <td>, 44HBEGIN COUPLED NUCLEAR - HYDRAULIC TIERATIONS PA E ITERATION LEVELS INVOLVED IN THS CASE. PA .0) IVOID=.TRUE. PA .6. PA .7. ICNTRL=.TRUE. .7. PA .7. ICNTRL=.TRUE. .7. PA .7. ICNTRL=.FALSE. .7. TERATION USES THE CONTROL ITERATION LEVEL DURING .7. PA .7. PA</td> <td>ARTB ARTB ARTB ARTB ARTB ARTB ARTB ARTB</td> | , 44HBEGIN COUPLED NUCLEAR - HYDRAULIC TIERATIONS PA E ITERATION LEVELS INVOLVED IN THS CASE. PA .0) IVOID=.TRUE. PA .6. PA .7. ICNTRL=.TRUE. .7. PA .7. ICNTRL=.TRUE. .7. PA .7. ICNTRL=.FALSE. .7. TERATION USES THE CONTROL ITERATION LEVEL DURING .7. PA | ARTB ARTB ARTB ARTB ARTB ARTB ARTB ARTB |
| C DETERMINE TH C DETERMINE TH C DETERMINE TH C DUID=.FALSE 43 IF (NUMAX.GT 44 ICNTRL=.FALSE 45 IF (MODEB(S). 46 IF (MODEB(S). 46 IF (MODEB(S)) 47 IF (MODEB(S) 48 NC=0 49 NU=0 50 NS=0 51 XL=1.0 52 ITERS=0 53 ISRCH=0 54 LPMEM = NPAR 48 NC=0 55 LNEED=NMACRC 54 LPMEM = NPAR 55 LNEED=NMACRC 54 LPMEM = NPAR 55 LNEED=NMACRC 54 LPMEM = NTAR 55 LNEED=NMACRC 54 LPMEM = NTAR 55 LNEED=NMACRC 56 IF (MODEB(I) 57 CALL NEXTPT 58 LNEED = NSTA 59 CALL NEXTPT 58 LNEED = NSTA 59 CALL NEXTPT 58 LNEED = NSTA 59 CALL NEXTPT 59 CALL NEXTPT 50 NITERS=1 C CONTROL ITEL 50 NGS IDSC=0 50 XLC=XKEFF 50 NGS KSTEP=1 50 NC=NC+1 | E ITERATION LEVELS INVOLVED IN THS CASE. PA PA PA PA PA PA PA PA PA PA | ARTB ARTB ARTB ARTB ARTB ARTB ARTB ARTB |
| C 42 43 44 43 44 44 45 45 45 45 46 46 46 47 47 47 47 47 47 47 47 47 47 | .0) IVOID=.TRUE. PA E. PA GE.3) ICNTRL=.TRUE. PA .GT.5) ICNTRL=.FALSE. PA TERATION USES THE CONTROL ITERATION LEVEL DURING PA IFE EXPOSURE SEARCH. PA CONDERDED(C) CT.0) ICNTRI =.TRUE. PA | ARTB ARTB ARTB ARTB ARTB ARTB ARTB ARTB |
| 43 IF (NUMAX.GT 44 ICNTRL=.FALS 45 IF (MODEB(5). 46 IF (MODEB(5). 47 IF (MODEB(5). 48 NC=0 49 NU=0 50 NS=0 51 XL=1.0 52 IFECTOR 53 ISRCH=0 54 LPMEM = NPAR 955 LNEED=NMACRO 956 IF (MODEB(1) 957 CALL NEXTPT 958 LNEED = NSTF 959 CALL NEXTPT 959 CALL NEXTPT 960 CALL NEXTPT 961 LPHEPMEM 962 21 963 NITERS=1 964 30 965 IOSC=0 965 IOSC=0 966 XLC#XKEFF 967 LEUELC=.FAL 968 IBPAS=0 969 KSTEP=1 963 KSTEP=1 | .0) IVOID=.TRUE. E. GE.3) ICNTRL=.TRUE. .GT.5) ICNTRL=.FALSE. TERATION USES THE CONTROL ITERATION LEVEL DURING IFE EXPOSURE SEARCH. FOR OND MODERCH. | ARTB ARTB ARTB ARTB ARTB |
| 44 ICHAL-FRES 45 IF (MODEB(5). 46 IF (MODEB(5). 47 IF (MODEB(5). 48 NC=0 49 NU=0 50 NS=0 51 XL=1.0 52 IFERS=0 53 ISRCH=0 155 LPMEM = NPAR 155 LNEED=NMACRO 155 LNEED = NSTA 155 LNEED = NSTA 155 CALL NEXTPT 158 LNEED = NSTA 159 CALL NEXTPT 150 LOGE 21 159 CONTINUE 150 SEC=0 < | GE.3) ICNTRL=.TRUE. .GT.5) ICNTRL=.FALSE. TERATION USES THE CONTROL ITERATION LEVEL DURING IFE EXPOSURE SEARCH. IFE EXPOSURE SEARCH. CONTRUESTION OF AN ICNTRUE. | ARTB ARTB ARTB ARTB |
| 46 IF (MODEB(5)) C THE HALING I C THE END OF L 47 IF (MODEB(5)) 48 NC=0 49 NU=0 50 NS=0 51 XL=1.0 52 ITERS=0 53 ISRCH=0 54 LPMEM = NPAR 55 LNEED=NMACRO 956 IF (MODEB(1)) 957 CALL NEXTPT 958 LNEED = NSTF 959 CALL NEXTPT 958 LANEID = NSTF 959 CALL NEXTPT 958 LOBED = NSTF 959 CALL NEXTPT 960 CALL NEXTPT 961 LPM=LPMEM 962 21 963 NITERS=1 964 30 965 IOSC=0 965 IDSC=0 965 IEVELC=.FAL 968 KSTEP=1 969 KSTEP=1 970 NC=NC+1 | TERATION USES THE CONTROL ITERATION LEVEL DURING PA | ARTB |
| C THE END OF L 47 IF (MODEB(5) 48 NC=0 49 NU=0 50 NS=0 51 XL=1.0 52 ITERS=0 53 ISRCH=0 54 LPMEM = NPAR 55 LNEED=NMACRO 56 IF (MODEB(1) 57 CALL NEXTPT 58 LNEED = NSTF 59 CALL NEXTPT 58 LNEED = NSTF 59 CALL NEXTPT 59 CALL NEXTPT 50 NITERS=1 C CONTINUE 50 NITERS=1 C CONTINUE 50 NITERS=1 C CONTINUE 50 NITERS=1 50 CALL SETF 50 CALL NEXTPT 50 C | IFE EXPOSURE SEARCH. | ADTR |
| 47 IF INDED(3) 48 NC=0 49 NU=0 50 NS=0 51 XL=1.0 52 ITERS=0 53 ISRCH=0 54 LPMEM = NPAR 55 LNEED=NMACRO 56 IF 57 CALL 58 LNEED = NSTF 59 CALL 561 LPMEMEM 57 CALL 58 LNEED = NSTF 59 CALL 59 CALL 561 LPMEMEM 562 CONTINUE 57 CALL 58 LNEED = NSTF 59 CALL 59 CALL 5061 LPMEM 562 IOST 563 IDSC=0 564 30 5065 IDSC=0 5066 XLC=XKEFF 5067 LEUELC=.FAL 508 IBPAS=0 5063 KSTEP=1 5070 <t< td=""><td>- 눈 (눈, 날랐던, 김나이는 것, 눈이가 눈이 깨끗이 가는 것 가까 흔들고</td><td></td></t<> | - 눈 (눈, 날랐던, 김나이는 것, 눈이가 눈이 깨끗이 가는 것 가까 흔들고 | |
| 49 NU=0 50 NS=0 51 XL=1.0 52 ITERS=0 53 ISRCH=0 54 LPMEM = NPAR 955 LNEED=NMACRO 956 IF (MODEB(1)) 957 CALL NEXTPT 958 LNEED = NSTF 959 CALL NEXTPT 960 CALL NEXTPT 961 LPH=LPMEM 962 21 CONTINUE 963 NITERS=1 964 30 CONTINUE 965 IOSC=0 966 XLC=XXEFF 967 LEUELC=.FAL 968 IBPAS=0 963 KSTEP=1 964 063 965 IOSC=0 966 XLC=XXEFF 967 LEUELC=.FAL 968 IBPAS=0 969 KSTEP=1 970 NC=NC+1 | | ARTB |
| 51 XL=1.0 52 ITERS=0 53 ISRCH=0 154 LPMEM = NPAR 155 LNEED=NMACRO 156 IF (MODEB(1)) 157 CALL NEXTPT 158 LNEED = NSTF 159 CALL NEXTPT 156 IF (MODEB(1)) 157 CALL NEXTPT 158 LNEED = NSTF 159 CALL NEXTPT 158 LNEED = NSTF 156 CONTINUE 1562 21 1563 NITERS=1 1564 30 1565 IDSC=0 1566 IDSC=0 1567 LEUELC=.FAL 168 IBPAS=0 069 KSTEP=1 070 NC=NC+1 <td></td> <td>ARTE</td> | | ARTE |
| 52 1+EX3=0 53 ISRCH=0 154 LPMEM = NPAR 155 LNEED=NMACRO 156 IF (MODEB(1)) 157 CALL NEXTPT 158 LNEED = NSTF 159 CALL NEXTPT 156 LPW=LPMEM 157 CALL NEXTPT 158 LNEED = NSTF 159 CALL NEXTPT 156 LPH=LPMEM 156 CONTINUE 156 IOST 157 CALL NEXTPT 158 CALL NEXTPT 159 CALL NEXTPT 156 CONTINUE 156 IOST 156 IOST 157 CONTROL ITEL 158 LOST 159 CONTROL ITEL 156 IOSC=0 155 IOST 156 IDSC=0 157 LEVELC=.FAL 158 IBPAS=0 159 KSTEP=1 150 NC=NC+1 | | ARTB |
| 154 LPMEM = NPAR 155 LNEED=NMACRC 156 IF (MODEB(1)) 157 CALL NEXTPT 158 LNEED = NSTF 159 CALL NEXTPT 158 LNEED = NSTF 159 CALL NEXTPT 158 LNEED = NSTF 159 CALL NEXTPT 150 CALL NEXTPT 150 CALL NEXTPT 158 LNEED = NSTF 159 CALL NEXTPT 150 CALL NEXTPT 150 CALL NEXTPT 150 CALL NEXTPT 159 CALL NEXTPT 150 CALL NEXTPT 151 CONTINUE 152 CONTROL ITE 154 30 CONTINUE 1555 IDSC=0 1565 IDSC=0 1565 IDSC=0 1566 LEVELC=.FAL 1667 LEVELC=.FAL | n en | ARTB |
| IF (MODEB(1)) IF (METTER) | RTA PrNDIM3D | ARTB |
| 57 CALL NEXTPT 58 LNEED = NSTF 59 CALL NEXTPT 60 CALL NEXTPT 61 LPH=LPMEM 62 21 CONTINUE 163 NITERS=1 164 30 CONTROL ITE 165 IOSC=0 166 XLC#XKEFF 167 LEUELC=.FAL 168 IBPAS=0 169 KSTEP=1 1070 NC=NC+1 | .EQ.0) LNEED=NDIM3D SCINCE TO POWER CONVERSION FACTOR ON I/O FOR 1 GROUP | ARTB |
| 158 LNEED = NSIF 159 CALL NEXTPT 160 CALL NEXTPT 161 LPH=LPMEM 162 21 163 NITERS=1 164 30 165 IDSC=0 166 XLC=XKEFF 167 LEUELC=.FAL 168 IBPAS=0 063 KSTEP=1 070 NC=NC+1 | (LPMEM, 10, LNEED, 20HCROSS SECTION) | ARTB |
| D60 CALL NEXTPT D61 LPW=LPMEM D62 21 CONTINUE D63 NITERS=1 D64 30 CONTINUE D65 IOSC=0 D66 XLC=XKEFF D67 LEUELC=.FAL D68 IBPAS=0 D69 KSTEP=1 D70 NC=NC+1 | (LPMEM, 18, LNEED, 15HSTATE VARIABLES) | PARTE |
| CONTINUE 062 21 CONTINUE 063 NITERS=1 064 30 CONTROL ITE 065 IDSC=0 066 XLC=XKEFF 067 LEUELC=.FAL 068 IBPAS=0 063 KSTEP=1 070 NC=NC+1 | (LPMEM, 24, NUIMJU, 2001 HEKNHE LENKNOL | ARTB |
| JB3 ITEKS=1 C CONTROL ITE 064 30 CONTINUE 065 IDSC=0 066 XLC=XKEFF 067 LEUELC=.FAL 068 IBPAS=0 069 KSTEP=1 070 NC=NC+1 | | XXZ |
| 064 30 CONTINUE 065 IDSC=0 066 XLC=XKEFF 067 LEUELC=.FAL 068 IBPAS=0 069 KSTEP=1 070 NC=NC+1 | RATION STARTS HERE | PARTB |
| 066 XLC#XKEFF 067 LEUELC=.FAL 068 IBPAS=0 069 KSTEP=1 070 NC=NC+1 | | PARTE |
| 068 IBPAS=0 069 KSTEP=1 070 NC=NC+1 | se de la construcción de la constru SE construcción de la construcción d | BRTE |
| 069 KSTEP=1 070 NC=NC+1 | X | XXX2 |
| | n en en en en el contra en | PARTB OPUS |
| 071 KPFRST=.TRU | E. T. 2) GO TO 40 | ÖPUS |
| 073 CALL POWDI | S (DEP, ICNTRL, IDEBUG, IOPTEX, IOSC, ISRCH, IVOID, | XXX5 |
| + LEVELC, LEV + NFLOH, NPOL | | OPUS |
| 1074 DO 60 I=1, I | ER, NSOURC, NUOID, XPOLST, XT, IFIRST, KPFRST) | OPUS |
| 1075 LHLL KCHUX 1076 DO 60 J=11. | ER, NSOURC, NUCID, XPOLST, XT, IFIRST, KPFRST) MAX CONC, LP12, ITAPH, I, ND, JD, KD, LPXTR(12)) | OPUS |
| 1077 EXP(I,J)=CC | ER, NSOURC, NUOID, XPOLST, XT, IFIRST, KPFRST) MAX CONC, LP12, ITAPH, I, ND, JD, KD, LPXTR(12)) MX(I) | OBH IS |
| 079 CALL POISF | ER, NSOURC, NUOID, XPOLST, XT, IFIRST, KPFRST) MAX CONC, LP12, ITAPH, I, ND, JD, KD, LPXTR(12)) JMX(I) DNC(NC4, J, 1) | 2222 2222 |

| | | 이 없습니다. 그는 것이 아파 이 것이 같은 것이 같이 있는 것이 같이 있는 것이 없는 것이 없는 것이 없다. 것이 없는 것이 없이 않이 | | |
|-----------|------------------|--|--------------|---|
| | | A NUMER NUMBER NUMBER NUMBER ST. XT. IFIRST | OPUS | 36 |
| | - 1 1 - | NFLOW, NPOWER, NSUURE, NOUTH, A CLOST AND A STORE (| JPUS XXX2 | 12 |
| 00080 | | IF(IDEBUG.GT.0) GO TO 660 | XXXZ | 13 |
| 00081 | | IF(IBPAS.EQ.O) THEM | XXX2 XXX2 | 14 |
| 00082 | | NC≈0 | XXXZ | 16 |
| 00084 | 1.1 | | | 17 18 |
| 00085 | | ELSE FINDIF | XXX2 | 19 |
| 00087 | 40 | CONTINUE | OPUS | 38 |
| 00088 | | NODEP=0 NODEP1=0 | 0PUS | 40 |
| 000083 | 1.1 | IF (MOPTIM .EQ. 2) THEN | OPUS | 41 |
| 00091 | =0 | WRITE(6,50) | OPUS | 43 |
| 00095 | 20 | *COMBINATIONS: , /, 3X, 61(1H*)) | OPUS | 44 |
| 00093 | | ENDIF | OPUS OPUS | 45 |
| 00094 | ÷., | PTEMP=0.0 | OPUS | 47 |
| 00096 | | XTEMP=0.0 | OPUS | 48 |
| 00097 | | JPEAK=0 | OPUS | 50 |
| 00099 | | IF(.NOT. DEPL(KPOIS)) THEN | OPUS | 51 |
| 00100 | | NODEP=NUULF+1 | OPUS OPUS | 53 |
| 00102 | 1 | ENDIF | OPUS | 54 |
| ан. Ал | Ç | INITIAL TRATION | OPUS | 56 |
| | Č | | OPUS | 57 |
| 00103 | | IEOL=0 | OPUS | 59 |
| 00104 | | TIMXE=0.0 | OPUS | 60 |
| 00106 | | NEC=0 | OPUS | 61 |
| 00107 | | ISTEP=U FIRST=1.0 | OPUS | 63 |
| 00109 | | IBRN=IBURM | OPUS | 64 |
| | C · | DETERMINE THE ITERATION LEVELS INVOLUED IN THE CASE. | OPUS | 66 |
| | | | OPUS | 67 |
| 00110 | | IVOID=.FALSE. | OPUS | 63 |
| 00111 | | ICNTRL=,FALSE. | CPUS | 70 |
| 00113 | | IF(MODEB(5).GE.3) ICNTRL=. IKUL. | OPUS | 71 |
| 00114 | r | THE HALING ITERATION USES THE CONTROL ITERATION LEVEL DURING | OPUS | 73 |
| | Č. | THE END OF LIFE EXPOSURE SEARCH. | OPUS | 74 |
| 00115 | | NC=0 | OPUS | 76 |
| 00117 | • | NU=0 | OPUS | 77 |
| 00118 | | NS#U XI=1.0 | OPUS | 79 |
| 00120 | | ITERS=0 | OPUS | 80 |
| 00121 | | ISRCH=U UPMFM = NPARTA | OPUS | 82 |
| 00122 | Ų · | LNEED=NMACRO+NDIM3D | OPUS | 83 |
| 00124 | | IF (MODEB(1), EG. 0) LINEED-INDITION FACTOR ON I/O FOR 1 GROUP | OPUS OPUS | 85 |
| 00125 | Ŀ | CALL NEXTPT (LPMEM, 10, LNEED, 20HCROSS SECTION | OPUS | 86 |
| 00126 | | NEED = NSTATE + NDINGU | OPUS | 88 |
| 00127 | (; | CALL NEXTPT (LPMEM, 24, NDIM3D, 20HTHERMAL LEAKAGE | OPUS | 89 |
| 00129 | 1 te | PH=LPMEM | OPUS | 90 |
| 00130 | | XP0=0.0 | OPUS | 92 |
| 00132 | 2 | A1(1)=0.0 | OPUS | 93 |
| 00133 | 3 | X1=0.0 XPOLST=0.0 | OPUS | 95 |
| 00135 | 5 | NTOTBP=0 | OPUS | 96 |
| 00136 | 5 | 1 | OPUS | 97 98 |
| 00137 | 3 | | OPUS | 99 |
| 00139 |) | IF(J.EG. 1) THEN | OPUS | 100 |
| 00140 |] | TEAT CONTRACTOR AND A CONT | OPUS | 102 |
| 00143 | 5 | ELSE and the second | OPUS | 103 |
| 0014 | 3 | NYY=4 | OPUS | 104 |
| 0014 | 5 | ELSEIF(I .EQ. J) THEN | OPUS | 106 |
| 0014 | 5 | n an an Arthur an Art Arthur an Arthur an Ar | OPUS | 107 |
| 0014 | f 1 a 1 a | | | 1. A. |

| | | 일을 수 있는 것이 아니는 것은 것은 것을 위한 것이 없는 것 않이 | | |
|---------------|---------------|--|----------------|------------|
| 00149 | | | neus | 108 |
| 00149 | | ENDIF | OPUS | 109 |
| 00150 | | NBP(I,J)=IBP(KPOIS,K)*10+(NBP(I,J)/1000)*1000 NTOTPP-NTOTPP+NYY+IBP(KPOIS,K) | OPUS | 110 |
| 00152 | 1 | CONTINUE | OPUS | 112 |
| 00153 | 20 | | OPUS | 113 |
| 00134 | C S | CONTROL ITERATION STARTS HERE | OPUS | 115 |
| 00155 | | | OPUS | 115 |
| 00156 | | LEVELC=.FALSE. | OPUS | 118 |
| 00158 | | NC=NC+1 | OPUS | 119 |
| 00159 | | • EVELCALEVELUALPAXLS, LPBU, NCOEFS, NCROSS, NEDITS, NEDTS, | OPUS | 121 |
| 00150 | | • NEXPOS, NFLOH, NPOWER, NSOURC, NUOID, XPOLST, XT, IFIRST, KPFRST) | OPUS | 122 |
| 00161 | | PPLIM#A1(39) | OPUS | 124 |
| 00162 | 470 | PPMUL T2=A2(52) | OPUS | 125 |
| 00163 | | NODER1=NODEP1+1 | OPUS | 127 |
| 00165 | | KPFRST#.FALSE. | OPUS | 128 |
| 00167 | | | OPUS | 130 |
| 00168 | n na Na sa | IF (SMAX .GT. PTEMP) THEN | OPUS OPUS | 131 |
| 00169 | 1, 16 s. | IPEAK=MAXSI | OPUS | 133 |
| 00171 | | JPEAK=MAXSJ | OPUS | 134 |
| 00172 | | | OPUS | 136 |
| 00174 | ~ | | OPUS PARTE | 137 |
| | C | END BUR LIMITS CALCULATION | PARTB | 504 |
| | C. | 이가 사실을 받았다. 그는 것은 것은 것은 것은 것은 것은 것은 것은 것은 것이 있었다. 이가 것은 것은 것은 것은 것은 것은 것은 것은 것은 것을 가지 않는 것은 것을 가지 않는 것은 것은 것은 가 같은 것은 | PARTE | 505 |
| | Č | EDIT SEVERAL AXIAL AVERAGE DISTRIBUTIONS | PARTB | 507 |
| 00175 | C | TE (MORTIM IT 2) THEN | OPUS | 138 |
| 00175 | | CALL EDT1 (S, MZ, MZ, MZ, MZ) | PARTB | 509 |
| 00177 | | IF(MODEB(5).EQ.1.OR.MODEB(5).EQ.2) CALL HTBAL(2) | PARTE PARTE | 510 511 |
| 00179 | | WRITE (ITAPOT, 300) | PARTB | 512 |
| 00180 | 300 | WRITE (ITAPN, 300) FORMAT (/54X, 27HAUERAGE AXIAL DISTRIBUTIONS, //, | PARTE | 514 |
| 00101 | | 1 11X, 4HNODE, 3X, SHPOWER, 8X, SHWATER, 7X, 8H SOURCE , 7X, 7HTHERMAL, 7X, | PARTE | 515 |
| | | 2 7HCONTROL, 11X, 1HK, 1UX, 6HBYPASSZ 3 35X, ZHDENSITY, 21X, 7HLEAKAGE, 20X, 8HINFINITY, 9X, 4HUQIDZ | PARTB | 517 |
| · · · · · · · | | 4 25X, 1HP, 13X, 1HU, 13X, 1HS, 13X, 1HT, 12X, 2HCT, 13X, 1HK, 12X, 2HBU/) | PARTB | 518 |
| 00182 | | CALL AXIALP (A(LPAXLS),KD,7,22H(12X,12,7F14.4,12X,12),ITAPO) | PARTE | 520 |
| 00184 | - | ENDIF | OPUS | 139 |
| 1 | C C | APPLY THE CONTROL SEARCH CALLED FOR BY MODEB(5) | PARTE | 522 |
| 00105 | Ĉ | | PARTB | 523 524 |
| 00185 | | IF (.NOT.ICNTRL) GO TO 390 | PARTE | 525 |
| 00187 | | IF (NC.GE.NCMAX) GO TO 390 | PARTE | 525 |
| 00189 | | IF (LEVELC) GO TO 390 server test of the set | PARTB | 528 |
| 00190 | 310 | | PARTE | 529 |
| 00192 | | GO TO (340,350,370),M | PARTB | 531 |
| 00193 | 340 | CALL NHWORD (NAM, 8HSEQUENCE, 8) | PARTE | 113 533 |
| 00195 | 350 | CALL NHUCRD(NAM, 8HPATTERN ,8) | SMCY1 | 114 |
| 00196 | 360 | ANOTWT=NOTWT | PARTE SMEY1 | 535 |
| 00198 | | CALL SEARCH(SR(4), ANOTHT, NAM, NAM(13), XKEFF, XLMBDA, ISRCH) | SMCY1 | 116 |
| 00199 | | NOTWT=ANOTWT | PARTE | 538 |
| 00201 | | CALL_RODMUU (A(LPR), A(LPNPER), A(LPARAY), A(LPNTCH), ID, JD, IRMX, LIMRA | PARTS | 540 |
| 00303 | | 1Y,LIMPAS) | PARTS | 541 542 |
| 00202 | 370 | n Guine (1980) and a state of the state of the CONTINUE | PARTE | 543 |
| 00004 | C | HALING SEARCH (SEE SIGCAL/SIGDAT) | PARTE | 544 525 |
| 00204 | | CALL NHWORD(NAM; SHHALING D; 8) | SMCY1 | 117 |
| 00206 | | CALL NHHORD (NAM (13), SHEXPOSURE, 8) | SMCY1 | 118 |
| 00208 | 380 | CONTINUE | PARTE | 549 |
| 00209 | | GO TO 30 | PARTE | 550 |

| | | | PARTB | 551 |
|---|---|--|-------|------------|
| 00210 | 390 | | PARTE | 552 |
| UUCII | C | A CHEL TERATION | PARTB | 554 |
| | C | END OF CONTROL LEVEL TIERHITUN | PARTB | 555 |
| 00313 | C | I PSTAT=NFLOW | PARIB | 557 |
| 00212 | | LPMEN = LPSTAT + NSTATE+JD+KD | PARTB | 558 |
| 00214 | 2 S | LNEED = LIST+NDAT+LIMNFT | PARTB | 559 |
| 00215 | 11 A. | | PARTB | 560 |
| 00215 | | CALL FINDET (LEMEM, LEKTAB, 17, LNEED) | PARTE | 562 |
| 00218 | | LPSIG = LPMEM | PARTE | 563 |
| 00219 | | LPCON = LPSIG + NMACRO+JB*KB | PARTB | 564 |
| 00220 | | LPCNLD = LPCUN + ND+JD+KD | PARTB | 565 |
| 00222 | | LPTKN = LPHF + KD+LIMNFT | PARTS | 567 |
| 00223 | | LPTKN2 = LPTKN + KD+LIMNFT | PARTE | 568 |
| 00224 | | NEXPOS = LPTKN2 + KU+LINOFT | PARTB | 569 |
| 00225 | C | | PARIS | 571 |
| | Č. | FUEL DEPLETION CONTROL | PARTB | 572 |
| | C S | | PARTB | 573 |
| en de ales | ç | USAGE OF IEOL AND IBURN | PARTB | 575 |
| | č | INPUT O LT IEOL LT 39 TO INITIATE SEARCH FOR EOL EXPUSURE BY | PARTB | 576 |
| | C - | EDLEXP SETS IEDL=99 WHEN IT SEES EUPPL UN HEAT STELL | PARTB | 577 |
| | | TRUESSA WHEN AFINAL SOURCE CALCULATION IS TO BE MADE. | PARTB | 578 578 |
| | Č. | IBURN=1 IF NO MORE SOURCE CALCULATIONS ARE TO BE MADE | PARID | 580 |
| | č | | PARTB | 581 |
| 00225 | | IF (DE.EQ.0.0) IEUL=999 | PARTB | 582 |
| 00227 | • | TE (DE. LT. 0. 0. AND. XPO.LE. XPOMAX) IEOL=999 | PARTB | 583 |
| 00228 | | IF (IEOL.EQ.999) IBRN=1 | PHRIB | 585 |
| 00230 | i en ser | IF (MODEB(5).NE.6) GO TO 400 | PARTB | 586 |
| • 00231 | | [BRN=] | PARTB | 587 |
| 00235 | 400 | CONTINUE | PARTB | 588 |
| 00234 | | DEP=DE | PARTB | 590 |
| 00235 | • • | IF (IEOL.EQ.999) GO TU 600 | OPUS | 140 |
| 00236 | . •] | COLL PACES (MAXLIN) | PARTB | 591 |
| 00238 | | ENDIF | PARTE | 592 |
| 00239 | | u ¹ LINES=4 a tradición de la contractor de | PARTB | 593 |
| 00240 | 1. 1. s. | IOPILX=1 | PARTB | 594 |
| 00241 | · · · | MD12=ISTEP/2 | PARIS | 233 |
| 00243 | | IF (MODEB(12).NE.0.AND.MD12+2.NE.ISTEP) GO TO 490 | PARTB | 597 |
| 00244 | | EOL=MODEB(10) | PARTB | 598 |
| 00245 | | TF (TFOLLE.0) GD TO 470 | PARTB | 599 |
| 00248 | C | | PHRIB | 601 |
| | C | FOLLOWING LOGIC CONTROLS THE END OF LIFE EXPOSURE SEARCH | PARTB | 605 |
| | <u> </u> | FOR XKEFF =XLIBUH | PARTB | 603 |
| 1. J. | č | ASSIGN CONTROL SEARCH PARAMETER | PARIB | 605 |
| 00247 | | MM=MODEB(5) | PARTE | 606 |
| 00248 | | CO TO (420.430.440.450.450.460) MM | PARTB | 607 |
| 00243 | 410 | 10 (420) 400 Hor Cor Cor Cor Cor Cor Cor Cor Cor Cor C | PARID | 600 |
| 00251 | | CALL EOLEXP (XKEFF, DE, XPO, XLMBDA, DEP, DUMY, LEUL, FIRST) | PARTB | 610 |
| 00252 | | GO TO 470 CONTENT (VEFF. DF. XPO. XL MRDA, DEP, POI, IEOL, FIRST) | PARTB | 611 |
| 00253 | 420 | | PARTB | 612 |
| 00255 | 430 | CONTINUE | PARIB | 614 |
| 00256 | | GO TO 470 | PARTB | 615 |
| 00257 | -440 | CONTINUE | PARTB | 616 |
| 00258 | 450 | SRCH=NOTWT | PARTB | 612 |
| 00250 | | CALL EOLEXP (XKEFF, DE, XPO, XLMBDA, DEP, SRCH, IEOL, FIRST) | PARTB | 619 |
| 00251 | | NOTWT=SRCH | PARTB | 620 |
| 00252 | 400 | CONTINUE | PARTB | 621 |
| 00264 | 470 | CONTINUE | PAPTR | 623 |
| 00265 | | IF (DEP.LT.O.O.AND.XPO+DEP.LT.XPOMAX) DEP=XPUMAX-XPO | PARTB | 624 |
| 00255 | ~ | | PARTB | 625 |
| | Ľ | SIGCAL DOES THE FUEL DEPLETION WHEN IOPTEX=1 | PARTB | 626 |
| | ć | | PARTB | 628 |
| 00267 | harana tarta. Ara | XPO≠A1(1)+DEP | | |
| · · | | | | |
| | | | | |

| | | 그는 사람들에 가지 않는 것이 있는 것이 아니는 것이 가지 않는 것이 가지 않는 것이 없는 것이 없다. | |
|-------|--------|--|---------|
| | | | |
| | | | |
| | et e s | | |
| 268 | | IF (MODEB(12).GT.O.AND.MD12+2.EQ.ISTEP) GU TU 480 | PARTB |
| 269 | 480 | .GO TO 510 TOPTFX=3 | PARTE |
| 271 | -00 | G0 T0 510 | PARTB |
| 272 | 490 | CONTINUE UPTTE (ITAPOT, 500) | PARTB |
| 274 | 500 | FORMAT (/10X, 87HTHIS IS THE FIRST STEP IN THE STEP AVERAGE DEPLETI | PARIB |
| | | 10N PROCEDURE CONTROLLED BY MODEB(12).) | PARTB |
| 1275 | 510 | CALL SIGCAL (A(LPSTAT), A(LPDAT), A(LPCON), A(LPCNLD), A(LPSIG), | PARTE |
| | | 1 A(LPNFT), A(LPTKN), A(LPTKN2), A(LPWF), A(LPKTAB), A(LPXK), | PARTE |
| | | 2 A(LPYY), A(LPCNLU), 10, JU, KU, NU, NU, NU, HIHKKU, HISTARC, CITAR FROM CONCERNMENTED AND A STATE AN | OPUS |
| 277 | | A1(1)=XPO | PARIS |
| 278 | a - | IF (IOPTEX.NE.3) GO TO 530 | PARTE |
| 279 | | LINES=LINESTE URITE (ITAPOT, 520) EBAR | PARTB |
| 281 | | WRITE (ITAPN, 520) EBAR | PARTB |
| 585 | 520 | FORMAT (/10X, STHTHIS IS THE REDEPLETION TIMESTEP CONTROLLED BY HOS | PARTE |
| 283 | | | PARTE |
| 284 | 530 | CONTINUE | OPUS |
| 285 | 1.1 | IF(MOPTIM .LT. 2) INLN UNTIF (ITAPAT.540) FRAR | PARTB |
| 287 | | WRITE (ITAPN, 540) EBAR | PARTE |
| 238 | 540 | FORMAT (10X, 63HTHE AVERAGE EXPOSURE OF THE FUEL IN CORE HAS BEEN | PARTB |
| 1000 | | IINCREASED TU + 10.4) | OPUS |
| 1583 | 550 | CONTINUE | PARTB |
| 231 | 555 | IF(MOPTIM .LT. 2) THEN | PARTB |
| 292 | • • | CALL PAGES (2) | PARTB |
| 1293 | | WRITE (ITAPN, 560) DEP | PARTB |
| 295 | 560 | FORMAT (/10X, 42HTHE (ADJUSTED) EXPOSURE INTERVAL USED WAS .F10.4) | OPUS |
| 1296 | · . · | ENDIF | PARTB |
| 1297 | | 11 (HUDEB(5).20.2.0K.HUDEB(5).20.47 55 15 514 | PARTB |
| 0299 | 570 | CONTINUE | PHRID |
| 0300 | | CALL PAGES (1) | PARTB |
| 0301 | | IF(MUDEB(12).61.0.HND.NDIC*C.EG.ISIEF/ GG TO STO TIMXE1=TIMXE | PARTB |
| 0302 | | TIMXE=TIMXE+XT/3600. | PARTE |
| 0304 | | XETIME=XETIME+XT/3600. | PARTB |
| 0305 | 57 | 5 CUNTINUE UPITE (ITAPOT.580) ISTEP.TIMXE1,TIMXE,XETIME | PARTB |
| 0305 | · * . | WRITE (ITAPN, 580) ISTEP, TIMXE1, TIMXE, XETIME | PARTE |
| 0308 | 580 | FORMAT (10X, 10HTIME STEP, 14, 10H BEGAN AT , F10.4, 20H HUUKS HOU ENL | FTN200F |
| | | 1ED AT ,F10.4,7H HUUKS.,/10X, 32HTUTHE THE SHOE HITTHEILATION , | PARTB |
| 0309 | 590 | CONTINUE | PARTB |
| 0310 | 600 | CONTINUE | PARTB |
| | C | ENTT ANTAL AUFRAGE EXPOSURE, VOID HISTORY AND CONCENTRATIONS | PARTE |
| | Č. | | PARTE |
| 0311 | | IF (MOPTIM .LT. 2) THEN | PARTB |
| 0312 | | CALL ULEAK(HULFHALS);(***)(***) | PARTB |
| 0314 | | 10 620 [=1, IMAX | PARTB |
| 0315 | | CALL READX (A(LPCON), LP12, ITAPH, I, NU, JU, KU, LPX (LC)) | PARTB |
| 0316 | | IF(ISET1.EQ.0) | PARTB |
| , | | ICALL AXIALC(A(LPCON), A(LPYY), ID, JD, KD, ND, A(LPAXLS), NCS, 2, 7, 1) | PARTB |
| 0318 | | TE (ISET2.EQ.0) | PARTB |
| 10319 | | NCSM=NCS-1 | PARTE |
| 00320 | | IF (NC5M.LT.1) GO TO 620 | PARTB |
| 00321 | | DO SIO NZ=1, NCSM | PARTB |
| 00322 | 610 | CONTINUE | PARTB |
| 00324 | 620 | CONTINUE | PARTR |
| 00325 | | CALL PAGES(15+KMAX) | PARTE |
| 00326 | | WRITE (ITAPN.630) | PARTB |
| 00327 | F | 30 FORMAT (/54X, 27HAVERAGE AXIAL DISTRIBUTIONS, //, | PARTB |
| | | 1 11X, 4HNODE, 5X, BHEXPOSURE, 9X, 4HUOID, 8X, 7HCONTROL, 5X, | PARTB |
| | | 2 SIH CONCENTRALIUNS CONCENTRALIUNS | PARTB |
| | | | PARTB |
| | | | T OADTP |
| 00329 | | CALL AXIALP(A(LPAXLS), KD, 7, 29H(12X, 12, 3F14, 4, 4E14, 4, 12X, 12), ITAPO | PARTE |

| | • | | | • |
|-------------------------|------------------------|--|--|---------------|
| | | ANTAL SCALL BOYLES KD. 7. 204(124, 12. 3F14, 4. 4F14, 4. 12X, 12), ITAPN |) PARTE 704 | |
| 00330 | C | GET 3D EDITS OF CONCENTRATION FILE ARRAYS, E, U, I, XE, PM, SM | PARTE 705 PARTE 706 | |
| 00332 | | CALL FINDET(LPMEM, LPSRC, 11, NDIM3D) LPE=LPMEM | PARTE 707 PARTE 708 PARTE 709 | |
| 00334 | | $\begin{array}{llllllllllllllllllllllllllllllllllll$ | PARTE 710 PARTE 711 | |
| 00336 | т. 1911 г. | LPTKN = LPCON + ND*JD*KD LPTKN = LPCON + ND*JD*KD LPTKN2 = LPTKN + KD*LIMNFT | PARTE 712 PARTE 713 | |
| 00339 00340 | | LPWF = LPTKN2 + KD+LIMNFT LPMAP1 = LPWF + KD+LIMNFT | PARTE 714 PARTE 715 PARTE 716 | ء بر بر |
| 00341 0 0342 | | LPMAP2=LPMAP1+MAXU(JD+KD,576) NEDTS=LPMAP2+KD+16 Coll clstze (NEDTS-1015ET(2),17HTARGET FILE EDITS) | PARTE 717 PARTE 718 | |
| 00343 | | CALL FZONES(A(LPTKN), A(LPTKN2), A(LPWF), ID, JD, KD, LIMNFT) | PARTE 719 PARTE 720 PARTE 721 | |
| 00345 | | CALL EDITB3 (A(LPSRC),A(LPE),A(LPV),A(LPXE),A(LPUN),A(LPNFT),A(L NFID),A(LPMAP1),A(LPMAP2),A(LPBATF),A(LPYY),ID,JD,KD,ND, | PARTE 722 PARTE 723 | |
| 00346 | • | ENDIF IF (DE.GT.0.0.AND.XPO.GE.0.99999*XPOMAX) IEOL=999 | 0PUS 148 PARTB 724 | · · · |
| 00348 | C . | IF (DE.LT.O.O.AND.XPO.LE.1.00001+XPOMAX) IEOL=999 | PARTE 725 PARTE 725 PARTE 727 | · · |
| 00349 | C | TE (TERN.ED.O.AND.IEOL.GE.99) GO TO 640 | PARTE 728 PARTE 729 | ÷ |
| 00350 | | IF (IEOL.EQ.999) GO TO 650 GO TO 20 | PARTE 730 PARTE 731 PARTE 732 | · |
| 00352 | 640 | IEOL=999 IBRN=1 CD TO 20 | PARTE 733 PARTE 734 | |
| 00355 | 650 | IEOL=999 IF (MODEB(15).EQ.0) GO TO 660 | PARTE 735 PARTE 736 PARTE 737 | |
| 00357 00358 | | A1(4)=PTH A1(5)=WT ⊙tc1o>=PT | PARTE 738 PARTE 739 | |
| 00359 00360 00361 | • • | A1(11)=NOTWT IF(MODEB(5).NE.6) GO TO 660 | PARTE 740 PARTE 741 | |
| | 000 | CONTINUATION MODE FOR COMPLETED HALING DEPLETIONS | PARTE 743 PARTE 744 | |
| 0 0362 | L , | XPO=XPOMAX A1(1)=XPO | PARTB 745 PARTB 746 PARTB 747 | |
| 00364 | 660 C | CONTINUE | 0PUS 149 0PUS 150 | |
| 00365 | Č | IF(MOPTIM .EQ. 2) THEN | 0PUS 151 0PUS 152 0PUS 153 | |
| 00366 | 664 | WRITE(6,664) KPOIS FORMAT(10(/),1X,5HCASE ,IS,10H COMPLETED,/,1X,20(1H=)) URITE(6,667) (IBP(KPOIS,1),1=1,KBPOS) | 0PUS 154 0PUS 155 | |
| 00369 | 667 | FORMAT(/,10X,11HBP LOADING ,2014) WRITE(6,668) NTOTBP | 0PUS 156 0PUS 157 0PUS 158 | |
| 00371 00372 00373 | 668 654 | FORMAT(10X,25HTOTAL NUMBER OF BP RODS ,14) WRITE(6,654) PTEMP,XTEMP,IPEAK,JPEAK FORMAT(/,1X,21HMAX POWER PEAKING OF ,F7.5,4H AT ,F6.3, | 0PUS 159 0PUS 160 | |
| 00374 | | +21H GHD/MT AT POSITION (, 11, 1H, , 11, 1H)) WRITE(6, 655) XPO COMMATCHE (SHOULD E LENGTH OCHIEVED | 0PUS 162 0PUS 163 | |
| 00375 00376 00377 | 622 | IF (XPO .GT. TEMP) THEN | 0PUS 164 0PUS 165 | |
| 00378 00379 | | ITEMP=KPOIS ENDIF | 0PUS 167 0PUS 167 0PUS 168 | |
| 00380 | C | KPFRST=.FALSE. | 0PUS 169 PARTE 748 | |
| 00000 | C C | SET SEARCH EIGENVALUE FOR THE NEXT (SERIES OF)CASES | PARTE 750 PARTE 751 | |
| 00382 00383 00384 | el l'angla. Na Sara | A1(19)=XKEFF XLMBDA=A1(19) | PARTE 752 PARTE 753 | |
| 00385 | 665 | MODEB(21)=0 5 CONTINUE | PARTE 755 PARTE 756 | |
| 00387 00388 00389 | | NORIGN=LHA-IBIAS CALL PAGES(18) | PARTE 757 PARTE 758 | |
| 00390 00391 | | IF(MOPTIM .EQ. 2) THEN DO 675 K=1.KBPOS | 0PUS 170 0PUS 171 0PUS 172 | |
| 00395 | | I≖IKUWKKJ | | |
| | | | an an Araban San Araban An Araban Ang a | |

| | | | UPUS | 11.2 |
|-------|-----------|--|--------|-------|
| 00393 | | | OPUS | 174 |
| 00394 | | NBP(I,J)=IBP(ITEMP,K)=10+(NBP(I,J)/1000)=1000 | OPUS | 175 |
| 00335 | 675 | CONTINUE | 1PUS | 176 |
| 00333 | . 913 | NDD1-NBPEPM-NODEP | | 177 |
| 00336 | · · · · · | | UFUS | 170 |
| 00397 | | HEP2=HOP1-HODEL NTREEM NERT, NERT, TEMP, TTEMP, EXP, JMX) | UPUS | 110 |
| 00398 | · · · · | CALL UPRESCHEP HEPERHINE TYTELEP IEL TELE | OPUS | 1/3 |
| 00399 | | ENDIF | PARTB | 759 |
| 00400 | | WRITE (ITAPOT, 670) LWA, NORIGA, MPARTH, ME LOW, NOUTH HEROSOF LEVEL | PARTE | 760 |
| 00400 | | I NSDURC. NPOHER, NEDITS, NEXPOS, NEDITS | 21 19N | 180 |
| | 070 | FORMAT(141,10%,25HMEMORY BOUNDARIES IN PARTE, | DARTR | 762 |
| 00401 | 670 | A HOW TO SOU INSTRUCTIONS (LUA) | PHK D | 702 |
| | | HY TUX, TO, SOL THE TON OF CONTAINER ARRAY | PHRID | 704 |
| | | B/10X, 18, 30H URIGIN OF COMMINDE | PARTS | (64 |
| · | | 1/10X, IS, 30H INPUT UNIT BOUNDARY | PARTE | 755 |
| | 1 | 2/10X, IB, 30H FLOW ITERATION BOUNDHRY | PARTE | 766 |
| | | 3//10X. IS. 30H UOID CALCULATION | PARTE | 767 |
| | | ALION TR. 30H CROSS SECTION CALCULATION | POPTR | 768 |
| | | ALLOY TR. 30H CREETCIENTS CALCULATION | CADTO | 769 |
| | | | THRID | 770 |
| | 1.1 | SZIUX, 18, 30H BOURDE THERMISON | PHRID | |
| | te gitte | 5/10X, 18, 30H PURER CHELOLAN ICH | PARTS | (() |
| | 1.1 | 7/10X, 18, 30H EUIT SELLIUN I | PARTE | 172 |
| | | 8/10X, 18, 30H EXPOSURE CALCULATION | PARTE | - 773 |
| | | 9/10X, IB, 30H EDIT SECTION 2 | PARTE | 774 |
| 00402 | | CALL TIMER (SHPARTB, 3) | PARTB | . 775 |
| 00402 | | DETLINN | PORTR | 776 |
| 00403 | · | | | |
| 00404 | | e a Guill a de la constante de la const | | |

| • | SUBROUTINE POISPUT (POW, XBR, KSTEP, IV, JU, NITERS, IBPAS, | POISPUT |
|--------------|---|-------------------------------|
| | LPAXLS, LPBU, NCOEFS, NCROSS, NEDITS, NEDTS, NEXPOS, NFLOH, NPOWER, NSOURC, NUOID, XPOLST, XT, IFIRST, KBPOS, NBPERM, IBP, DEPL, IROH, JCOL) | POISPUT POISPUT POISPUT |
| 5 1 | COMMON/BLOK1/IBLOK1(1), ID, JD, KD, IX1, JX1, KX1, IX2, JX2, KX2, ISET1, | BLOK1 F200CPU |
| | DIMENSION JOBNAM(2) | BLOK4 |
| 5 | DIMENSION POW(IU, JU), XBR(IV, JU) | POISPUT |
| 5 | DIMENSION IROW(20), JCOL(20), NBPBAS(15, 15), PUERIO(15, 15, 20), NUMBP(20), RHS(120), ACY(120, 50), T(100), DET(2) | POISPUT |
| | <pre># ,PBASE(15,15),COST(50),PSOL(120),DSOL(120),RW(15000), # IW(400),DUM(120),BPLIM(20),IBP(50000,20)</pre> | POISPUT |
| | LOGICAL IUDID, ICNTRL, IFIRST, LEVELU, LEVELC, DEPL (50000), KPFRST | POISPUT |
| | AT FIRST STEP OF DEPLETION. FIND FRESH FUEL POSITIONS IN WHICH | POISPUT |
| ġ | BP REQUIREMENTS WILL BE SEARCHED FOR | POISPUT |
| | JBPSOL=IROUND(A1(47)) | POISPUT |
| | BPDEL=A2(50) | POISPUT |
| · . | PPMULT2=A2(52) | POISPUT |
| | UBPEDIT=IROUND(A1(48)) | POISPUT |
| | IF(JBPEDIT.EG.U) NSTEP=1 K=1 | POISPUT |
| | NODES=0 DO 20 I=1, IMAX | POISPUT |
|] | JL=JMN(I) JR≈JMX(I) | POISPUT |
| | DD 20 J=JL,JR INBP=NBP(I,J)-(NBP(I,J)/10000)+10000 | POISPUT |
| | IF(INBP.LE.O) GO TO 30 IROW(K)=I | POISPUT |
| | JCOL(K)=J K=K+1 | POISPUT |
| | | POISPUT |
| į | 0 CONTINUE | POISPUT |
| <u>י</u> ני | AT FACH DUITED STEP, KEEP BASE POUED DISTRIBUTION IN ORDER TO GET | POISPUT |
| | FIRST ORDER DERIVATIVE, D(POW(I,J))/D(BPNUM) | POISPUT |
| | DO 10 I=1, IMAX | POISPUT |
| 3 | | POISPUT |
| 5. | DU IU J=JL,JR PBASE(I,J)=POW(I,J) | POISPUT |
| | O CONTINUE | POISPUT |
| 3 | KLCUN=U DO 180 KPP=1,KBPOS | POISPUT |
| | J=JCOL(KPP) | POISPUT |
| ŝ | BPLIM(KPP)=TTNBP/10. | POISPUT |
| + 5 | IF(LDIFF.GT.0) KLCON=ABS(NUMBP(KPP)) | POISPUT |
| 5) - (| 80 CUNIINUE | POISPUT |
| | COMPUTE THE LEAST SQUARE DIFFERENCE, (XBR(I,J)-PBASE(1,J))**2 | POISPUT |
| 7° 3 . | TUTAL=0. TOTAL1=0.0 | POISPUT |
| ∃ . | IF(JBPSOL.EQ.0) THEN DO 120 KK=1,KBPOS | POISPUT |
| 1 | 1=IROW(KK) J=JCOL(XK) | POISPUT |
| 3 | TOTAL=TOTAL+ABS(XBR(I,J)-PBASE(I,J))/XBR(I,J) L20 CONTINUE | POISPUT |
| 5 | NTOTL=KBPOS | POISPUT |
| 7 | JL=JMN(I) JR=JMX(I) | POISPUT |
| i e | | POISPUT |

| 2 | 65 | CONTINUE AVGDIFF=100+TOTAL1/NODES | POISPU |
|--------|---|--|------------------|
| 3 L | | ELSE DO 85 T=1+IMAX | POISPU |
| 5 | | | POISPU |
| | | JR=Jmx(1) D0 85 J=JL,JR | POISPU POISPU |
| 5 1 | 85 | TOTAL=TOTAL+ABS(XBR(I,J)-PBASE(I,J))/XBR(I,J) CONTINUE | POISPU |
| | | NTOTL=NODES | POISPUI |
| | C C TES | TCONVERGENCE | POISPUT |
| i i | C | | POISPUT |
| | 3000 | HRITE(6,3000) OBSQ | POISPUT |
| | | PERCENT) | POISPUT |
| | | HRITE(6,3060) AUGDIFF | POISPUT |
| | 3060 | ENDIF FORMAT(//,3X,'=== CORE AVERAGE PERCENT DIFFERENCE =',F10.5, | POISPUT |
| • | * | IBPAS=0 | POISPUT |
| | Ç | | POISPUT |
| | • | IF(NITERS.EG.1) GO TO 130 | POISPUT |
| | | (ABS(OBSG-OLDSG) .LE. 0.1 .AND. KLCON .LT. 5)) THEN | POISPUT |
| | | 18PAS=1 GO. TO 999 | POISPUT |
| : | 130 | ENDIF OLDSQ=0BSQ | POISPUT |
| | C C GET | SENSITIUITY COFFETCIENT HERE. INCREMENT OF RONIM IS 1 | POISPUT |
| | č | HOLTE(C. 4545) | POISPUT |
| • | 4545 | FORMAT(//, 120(1H+),//) | POISPUT |
| | 4550 | FORMAT(///,2X,101(1H=),/,2X,'(',12,')=',' OUTER ITERATION STEP' | POISPUT |
| • | i interi 🕈 Lini interiori Lini interiori | , /, 2X, 101(1H=), /) IF(OBSQ.LE.3.0) THEN | POISPUT |
| | | IF(NITERS.EQ.1) GO TO 600 HRITE(6,3050) | POISPUT |
| | 3050 | FORMAT(//,3X,55(1H-),/,3X,' NO MORE EVALUATION OF POWER ', | POISPUT |
| | | GO TO 900 | POISPUT |
| | 600 | | POISPUT |
| e t | | MDELBP=2 IF(JBPEDIT.EQ.0) THEN | POISPUT |
| | | IEDIT(2)=1 IEDIT(3)=1 | POISPUT |
| | | IEDIT(5)=1 | POISPUT |
| | 4560 | HRITE(6,4560) | POISPUT |
| | | DO 50 K=1,KBPOS | POISPUT |
| | | JL=JMN(I) | POISPUT |
| | | JR=JRX(1) DO 40 J=JL, JR | POISPUT |
| | 40 | NBP(I,J)=NBPBAS(I,J) IP=IROW(K) | POISPUT |
| | | UP#UCOL(K) WRITE(6,4570) K,IP,UP | POISPUT |
| ÷., | 4570 | FORMAT(//,3X,'(KSTEP)=',13,' ++ POWER SENSITIVITY ARRAY DUE TO', | POISPUT |
| • | | NEP(IP, JP)=NEP(IP, JP)+10+MDELEP | POISPUT |
| | | KPFRST=.TRUE. | PUISPUT |
| | 가 가장하다. 이 가장하는 것이 가장하는 것이 같이 | CALL POHDIS(DEP, ICNTRL, IDEBUG, IOPTEX, IOSC, ISRCH, IVOID, LEVELC, LEVELU, LPAXLS, LPBU, NCDEFS, NCROSS, NEDITS, NEDTS, NEXPOS. | POISPUT |
| | | NFLOW, NPOWER, NSOURC, NUOID, XPOLST, XT, IFIRST, KPFRST) | POISPUT |
| | | CALL READX(POW, LP11, ITAPU, 1, IU, JU, KD, LPXTR(11)) | POISPUT |
| | | JU-JU IIIA JL=JMN(II) | PUISPUT |
| | | ארב⊐טרא(111) 10 61 JJ=JL, JR | POISPUT |

| | PROPERTY AND (BOUCTT, AND PROSECTT, AND)/MDELBP | POISPUT 166 |
|-----------------------|---|----------------------------|
| 00135 | PUERIO(11, JJ, K)=(FUM(11, JJ) / BHOE(11, JO)/ | POISPUT 167 |
| 00136 | WRITE(6,1000) (PDERIU(II,JH,K),JH=1,JR) | PUISPUI 168 POISPUT 169 |
| 00138 | 1000 FORMAT(3X, 15F8.5) | POISPUT 170 |
| 00139 | 60 CONTINUE | POISPUT 171 |
| 00140 | 50 LUNIINUL 900 CONTINIE | POISPUT 172 |
| 00141 | | POISPUI 173 POISPUT 174 |
| | C SOLVE LINEAR EQUATION, AC(I, J)+BPNUM(J)=RHS(J) WHERE | POISPUT 175 |
| | C I=J=1,KBPOS, RHS(J)=XBR-PBASE | POISPUT 176 |
| | C HERE SUBROUTINE LEGIF IN INSL LIBRART IS COLD | POISPUT 177 |
| | | POISPUT 178 |
| | C APPLY LINEAR PROGRAMMING TECHNIQUE HERE, | POISPUT 180 |
| | $\mathbf{C} = MAX(-SUM(D(J)))$ | POISPUT 181 |
| | | POISPUT 182 |
| | | POISPUI 183 |
| | C AC(I,J)*X(J) .LE. RHS(J) | POISPUT 185 |
| | | POISPUT 186 |
| | C WHERE I=1, (Switches) | POISPUT 187 |
| (a,b) = (a,b) = (a,b) | | PUISPUI 188 |
| | C (X(J), J=1, KBPOS+NODES)=(NUMBP(1)NUMBP(KEPOS).D(1)D(HODES)) | POISPUT 190 |
| | C THE THE THE IN THE I TREAPY IS LITTLITED | POISPUT 191 |
| | C SUBROUTINE 2XJLP IN INSL LIDKAKT IS OTTELED | POISPUT 192 |
| | | POISPUT 193 |
| 00142 | IF(JBPSOL.EQ.0) GO TO 700 | |
| 00143 | N=KBPOS+NODES | POISPUT 196 |
| 00144 | NODE2=2+NODES | POISPUT 197 |
| 00145 | | POISPUT 198 |
| 00147 | WRITE(6,5000) (BPLIM(L),L=1,KBPOS) | PUISPUI 133 |
| 00148 | 5000 FORMAT(//,4X,'BPLIM(K) =',10F8.3) | POISPUT 201 |
| · | C DO TO TO NODES | POISPUT 202 |
| 00149 | 7C DUM(1)=0.0 | POISPUT 203 |
| 00150 | (8 KY=1 | POISPUI 204 |
| 00152 | DO 77 I=1, IMAX | |
| 00153 | utina <mark>ul=uMN(I)</mark> | POISPUT 207 |
| 00154 | JR=JNX(1) no 29 l= 1 . 19 | POISPUT 208 |
| 00133 | D0 79 K=1,KBPOS | POISPUT 203 |
| 00157 | 79 DUM(KY)=DUM(KY)+PDERIU(I,J,K)+BPLIM(K) | POISPUT 211 |
| 00158 | 78 KY=KY+1 | POISPUT 212 |
| 00159 | KX=1 | POISPUT 213 |
| 00161 | DO 70 I=1, IMAX | PUISPUI 214 POISPUI 215 |
| 00162 | JL=JMN(I) | POISPUT 216 |
| 00163 | | POISPUT 217 |
| 00164 | | POISPUT 218 |
| 00166 | ACY(KX,K)=PDERIU(I, J,K)/XBR(I, J) | POISPUT 220 |
| 00167 | ACY(KX+NODES,K)=-PDERIU(I,J,K)/XBR(1,J) | POISPUT 221 |
| 00168 | $71 \qquad AUY(KX+HUUE2;K)=PUERIV(1;J;K) \\ AUY(KY,KY+KRPOS)=-1.0$ | POISPUT 222 |
| 00153 | ACY(KX+NODES,KX+KBPOS)=-1.0 | POISPUT 223 |
| 00171 | RHS(KX)=(XBR(I,J)-PBASE(I,J)+DUM(KX))/XBR(I,J) | POISPUT 225 |
| 00172 | | POISPUT 226 |
| 00173 | COST(KX+KBPOS)=-1.0 | POISPUT 227 |
| 00174 | KX=KX+1 | POISPUT 228 |
| 00176 | 70 CONTINUE | |
| | | POISPUT 231 |
| 00177 | $\begin{bmatrix} 0 & 75 \\ -5 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ -5 \end{bmatrix} = \begin{bmatrix} 1 \\ $ | POISPUT 232 |
| 00178 | 75 LU31(J)-0.0 | POISPUT 233 |
| 00179 | WRITE(6, 4440) | PUISPUI 234 |
| 00180 | 4440 FORMAT(///, 3X, 65(1H=), /, 3X, '??? LINEAR PROGRAMMING SULVER: | POISPUT 236 |
| | + ,' SUBROUTINE ZAGEM IS CALLED TTT', JA, BOCINETIN | POISPUT 237 |
| 00181 | COLL TYSI PLACY, 120, RHS, COST, N. NODES, M2, S. PSOL, DSOL, RH, | POISPUT 238 |
| 00185 | s IW, IER) | POISPUT 239 |
| 00183 | TIME2=SECOND() | POISPUT 240 |
| 00184 | TOTT=TIME2-TIME1 | |
| 00185 | DO 500 I=1;KBMUS | POISPUT 243 |
| 00186 | SOBO=S/NODES+100 | POISPUT 244 |
| 00188 | GO TO 800 | POISPUT 245 |
| | | |
| | | |
| | | |

| | 이는 것에 속했는 것 이 것 같아요. 이 것 같아요. 이 것 이 가지 않는 것은 것을 못 하는 것이 있다. 것 같아요. 이 것 같아요. | |
|--------------|--|---------|
| | | |
| 9 | 700 CONTINUE | POISPUT |
| 0 | | POISPUT |
| 2 | | POISPUT |
| 3 4 | JJ=JLUL(KK) RHS(KK)=XBR(II,JJ)-PBASE(II,JJ) | POISPUT |
| 56 | DO 90 K=1,KBPOS ACY(KK,K)=PDERIV(II,JJ,K) | POISPUT |
| 7 | 90 CONTINUE TE(KK EQ.KBPOS) GO TO 95 | POISPUT |
| 3 | | POISPUT |
| 1 | S5 CONTINUE | POISPUT |
| 2 | WRITE(6, 4449) 4449 FORMAT(///, 3X, 65(1H=), /, 3X, '??? LINEAR EQUATION SOLVER:* | POISPUT |
| | SUBROUTINE LEGIF IS CALLED ???',/,3X,65(1H=),/) TIME1=SECOND() | POISPUT |
| 5 | CALL LEGIF (ACY, 120, KBPOS, 1, RHS, 120, 1, 0, T, IER) | POISPUT |
| 5 | TOTT=TIME2-TIME1 | POISPUT |
| 3 | 100 NUMBP(KI)=RHS(KI)+10 | POISPUT |
| 0 1 | 800 CONTINUE WRITE(6,4011) TOTT | POISPUT |
| Ş | 4011 FORMAT(/, 3X, '(CPU TIME CONSUMED IN EITHER LP OR LINEAR ', | POISPUT |
| 3 | IF(JBPSQL.NE.0) THEN | POISPUT |
| 4 5 | 4000 FORMAT(//,3X,'OBJECTIVE FUNCTION =',F10.5,' PERCENT',/) | POISPUT |
| 6 · · | ENDIF WRITE(6,2000) (NUMBP(KM),KM=1,KBPOS) | POISPUT |
| 8 | 2000 FORMAT(/4X, ' (INCREMENT OF BP DISTIBUTION)+10', /, 4X, 1515) | POISPUT |
| ÷ | C ASSIGN SEARCHED BP DISTRIBUTION AT THIS PLACE | POISPUT |
| 9 | C. 160 KL=1,KBPOS | POISPUT |
| Ō | | POISPUT |
| ż. | LNBP=NBPBAS(I,J)-(NBPBAS(I,J)/1000)*1000 MTPO-LNBP+NUMBP(KL) | POISPUT |
| 4 | IF(MTBP.LE.O) NUMBP(KL)=-LNBP | POISPUT |
| 5 | 160 CONTINUE | POISPUT |
| 7 | WRITE(6,4120) 4120 FORMAT(///,3X,'*** CONSTRUCTED POWER USING LINEAR', | POISPUT |
| | * APPROXIMATION IN SEARCHED BP DISTRIBUTION ***' ///) | POISPUT |
| 9 | 1. μ. | POISPUT |
| 11 | JR=JMX(I) D0 410 J=JL, JR | POISPUT |
| 3 | PPBAS=PBASE(I,J) DO 420 K=1.KBPOS | POISPUT |
| 5 | 420 PPBAS=PPBAS+PDERIU(I, J, K) *NUMBP(K)/10 | POISPUT |
| 17 | 410 CONTINUE | POISPUT |
| 38 39 - 1 | 3020 FORMAT(3X, 8F8.3) | POISPUT |
| •0 •1 | 400 CONTINUE WRITE(6,3030) | POISPUT |
| 15 | 3030 FORMAT(///,3X,65(1H-),/,3X,' NORMAL POWER CALCULATION USING', | POISPUT |
| 43 | KSTEP=KSTEP+1 | POISPUT |
| 44 45 | NITERS=N1(2KS+1) S99 CONTINUE | POISPUT |
| 46 47 | IF(JBPEDIT.EQ.0) THEN TFDIT(2)=5 | POISPUT |
| 48 | iEDIT(3)=5 | POISPUT |
| +3 50 | | POISPUT |
| 51 52 | IF (IBFAS .EQ. 1) THEN CALL INTBP(IROW, JCOL, NBPBAS, PBASE, PDERIU, IMAX, JMN, JMX, KBPOS, | POISPUT |
| 57 | XBR, IV, JV, NBPERM, IBP, DEPL, PPLIM, BPDEL, PPMULT1) | POISPUT |
| 54 | | POISPUT |
| 35 56 | n and the second s | POISPUT |
| 57 | RETURN | POISPUT |

| | THE REPORT OF THE PARTY OF THE THE THE THE THE | INTRO | ' a |
|--|--|--|--|
| 00001 | SUBROUTINE INTEP(IROW, JCCL, NEPSAS, PSASE, PBERIO, INHA, SUBROUTINE | INTRP | 3 |
| | * KBPUS.XBR, IV. JU. NBPERN, IBF, DEFL, FFLIN, BFDELS, THEL. I | INTEP | 4 |
| 00002 | | INTEP | 5 |
| 00003 | HINEMSIUN INFERTALL IPOURON, ICH. (20), NEPBAS(15,15), | INTEP | 6 |
| | HINT (24), HMV(24), PBOSE(15, 15), PDERIU(15, 15, 20), | INTBP | 7 |
| | | INTEP | - 8 |
| | BEAULTS, SRP(TU, U), TRPEX(50000), IBP(50000, 20) | INTBP | 9 |
| | | INTEP | 10 |
| 00004 | L DAPAMETER(INTMAX=6) | INTBP | 11. |
| 00004 | DATA (NTISTR(1), 1=1.7) /0.4.8.12.16.20.24/ | INTBP | 12 |
| 00003 | | INTEP | 13 |
| 1990 - 1990 - 1990 - 1990 - 1990 - 1990 - 1990 - 1990 - 1990 - 1990 - 1990 - 1990 - 1990 - 1990 - 1990 - 1990 - | TETERMINE THE HIGH AND LOW INTEGRAL NUMBER OF BP'S | IN RE | 14 |
| | C CORRESPONDING TO THE POISPUT RESULTS | INIB | 10 |
| ÷ | | | 17 |
| 00006 | DO 100 K=1.KBPOS | INTER | 19 |
| 00007 | I=IROH(K) | INTRO | 19 |
| 00008 | J=JCOL(K) | INTER | 20 |
| 00009 | RNBP(K)=FLOAT(NBPBAS(I,J) - (NBPBAS(I,J)/1000)+1000)/10.0 | INTRO | 21 |
| 00010 | IF(RNBP(K) .GT. NDISBP(INTMAX)) THEN | INTRP | 22 |
| 00011 | WRITE(6,150) I.J | INTRP | 23 |
| 00012 | 150 FORMAT(/, 3X, BP LIMIT EXCEEDED AT PUSITION (, 12, , 12, / | INTEP | 24 |
| 00013 | NHIBP(K)=NDISBP(INTHAX) | INTEP | 25 |
| 00014 | NLOBP(K)=NDISBP(IN(MAX-I) | INTEP | 26 |
| 00015 | | INTBP | . 27 |
| 00016 | $U = 200 E^{-2}$ ITTIMES TECHNOLIN I ANTERPRIA ANTE PARE(X) (F. NITSBP(L-1)) THEN | INTEP | 28 |
| 00017 | IF (RIDF(K) | INTBP | - 29 |
| 00018 | | INTEP | 30 |
| 00019 | | INTEP | 31 |
| 00020 | | INTBP | 32 |
| 00051 | EUU CUTTICE MAY=NTISBP(INTMAX) | INTBP | 33 |
| 00022 | TELLINHTEP(K)-RNEP(K)).LE. BPDEL .AND. NHIBP(K) .NE. NMAX) THEN | INTBP | 34 |
| 00023 | NXRP(K)=NHIBP(K)+4 | INTBP | |
| 00025 | FLSEIF((RNBP(K)-NLOBP(K)).LE. BPDEL .AND. NLOBP(K) .NE. 0)THEN | INTEP | 36 |
| 00025 | NXBP(K)=NLOBP(K)-4 | INTEP | 30 |
| 00027 | in the second | INIBP | 20 |
| 00028 | | TUI RM. | . 33 . |
| | | TAITDO . | 40 |
| 00029 | · 法教育部署ENDIF · · · · · · · · · · · · · · · · · · · | INTEP | 40 |
| 00029 | 100 CONTINUE | INTBP INTBP | 40 41 42 |
| 00029 | ENDIF 100 CONTINUE C | INTBP INTBP INTBP | 40 41 42 43 |
| 00029 00030 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS | INTBP INTBP INTBP INTBP INTBP | 40 41 42 43 44 |
| 00029 00030 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C | INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 42 43 44 45 |
| 00029 00030 00031 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS | INTBP INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 42 43 44 45 46 |
| 00029 00030 00031 00032 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) | INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 42 43 44 45 46 47 |
| 00029 00030 00031 00032 00033 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1, KBPOS TOTAL (L) | INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 42 43 44 45 46 47 48 |
| 00029 00030 00031 00032 00033 00034 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K CONTINUE | INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 42 43 44 45 46 47 48 49 |
| 00029 00030 00031 00032 00033 00034 00035 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUSE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES | INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 42 43 44 45 46 47 48 49 50 |
| 00029 00030 00031 00032 00033 00034 00035 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMY=0 | INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 42 43 44 45 46 47 48 49 50 51 |
| 00029 00030 00031 00032 00033 00034 00035 00036 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS | INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 42 43 44 45 46 47 48 49 51 51 52 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF(NXBP(K),GE,0) THEM | INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 423 445 466 478 490 551 552 551 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF(NXBP(K).GE. 0) THEN NUMX=NUMX+1 | INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 423 445 467 489 551 553 45 553 45 |
| 00029 00030 00031 00032 00033 00034 00035 00035 00036 00037 00038 00039 00040 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF(NXBP(K).GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K | INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP | 40 41 42 44 45 46 47 49 51 23 45 55 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00039 00040 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF(NXBP(K) .GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF | INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 42 44 44 44 44 44 55 55 55 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00035 00036 00037 00038 00039 00040 00041 00042 | ENDIF 100 CONTINUE C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF(NXBP(K) .GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE | INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP INTBP | 40 41 42 44 44 44 44 44 55 55 55 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00035 00036 00037 00038 00039 00043 | ENDIF 100 CONTINUE C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF(NXBP(K) .GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX | INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP INTEP | 40 41 42 44 44 44 44 44 55 55 55 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 | ENDIF 100 CONTINUE C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF(NXBP(K) .GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX, NBPX, IBPX0) | INTERP | 40 41 42 44 45 45 55 55 55 55 55 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00035 00038 00037 00038 00039 00040 00041 00042 00043 00044 | ENDIF 100 CONTINUE C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF(NXBP(K) .GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX,NBPX, IBPX0) NUPERM=0 | INTERP | 40 41 42 44 45 45 45 45 55 55 55 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 00045 00046 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KEPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF(NXBP(K) .GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX,NBPX, IBPX0) NUPERM=0 DO 220 I=2,NBPX | INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP INTERP | 40 4423 44567890123456789012 555555555555555555555555555555555555 |
| 00029 00030 00031 00032 00033 00034 00035 00035 00035 00038 00039 00041 00042 00044 00044 00045 00046 00047 | ENDIF 100 CONTINUE C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS.NBPERM.IBPERM) DO 205 K=1.K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1.KBPOS IF(NXBP(K).GE. 0) THEN NUMX=NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX,NBPX.IBPX0) NUPERM=0 DO 220 I=2.NBPX DO 225 II=1.KBPOS | INTERP | 40 44 44 44 44 44 44 44 44 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00035 00035 00038 00039 00040 00041 00042 00044 00045 00044 00045 00046 00047 00048 | ENDIF 100 CONTINUE C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINCS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF(NXBP(K).GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX,NBPX, IBPX0) NUPERM=0 DO 220 I=2,NBPX DO 225 II=1,KBPOS IPOINT(I,II)=II | INTERP | 40 44 44 44 44 44 44 44 44 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 00045 00044 00045 00046 00047 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1, K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1, KBPOS IF(NXBP(K).GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX, NBPX, IBPX0) NUPERM=0 DO 225 II=1, KBPOS IPOINT(I, II)=II 225 CONTINUE | INTERP | 40 44 44 44 44 44 44 44 44 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00040 00041 00042 00043 00044 00045 00044 00045 00047 00048 00049 00050 | ENDIF 100 CONTINUE C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS,NBPERM,IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF(NXBP(K).GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX,NBPX,IBPX0) NUPERM=0 DO 220 I=2,NBPX DO 225 II=1,KBPOS IPOINT(I,II)=II 225 CONTINUE NUM=IBPX0(I) | INTEP | 40 44 44 44 44 44 44 44 44 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 00045 00044 00045 00045 00045 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTECRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1.K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1.KBPOS IF(NXBP(K) .GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX, NBPX, IBPX0) NUPERM=0 DO 220 I=2.NBPX DO 225 II=1.KBPOS IPOINT(I.II)=II 225 CONTINUE NUM=IBPX0(I) NSUM=0 DO 230 L=1.NUMX | INTEP | 40 44 44 44 44 44 44 44 44 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 00045 00044 00045 00048 00049 00050 00051 00052 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1.K3POS IPOINT(1.K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1.KBPOS IF(MXBP(K).GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX.NBPX.IBPX0) NUPERM=0 DO 220 I=2.NBPX DO 225 II=1.KBPOS IPOINT(I.II)=II 225 CONTINUE NUM=1BPX0(I) NSUM=0 DO 220 J=1.NUMX | INTBP | 40 41 42 34 45 67 89 01 23 45 67 89 01 23 45 67 8 90 12 34 56 7 89 01 23 45 67 8 90 12 34 56 7 89 01 23 45 66 7 89 01 23 45 66 7 89 01 23 45 66 7 89 01 23 45 66 7 89 01 23 45 89 01 23 45 89 01 20 10 20 10 20 10 20 20 20 20 20 20 20 20 20 20 20 20 20 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 00045 00044 00045 00048 00049 00050 00051 00053 00054 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C ALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1.K3POS IPOINT(1.K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1.KBPOS IF(NXBP(K) .GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX, NBPX, IBPX0) NUPERM=0 DO 220 I=2.NBPX DD 225 II=1.KBPOS IPOINT(I.II)=II 225 CONTINUE NUMEIBPX0(I) NSUM=0 DO 230 J=1.NUMX J=NUMX-J LLI=MUM/10*+JJ | INTERP | 40 41 42 34 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 89 01 23 45 89 01 23 45 89 01 23 45 89 01 23 45 89 01 23 45 89 01 23 45 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 89 01 23 80 80 80 80 80 80 80 80 80 80 80 80 80 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00043 00044 00045 00044 00045 00044 00045 00046 00047 00048 00049 00050 00051 00052 00053 00054 00055 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS,NBPERM,IBPERM) DO 205 K=1.K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1.KBPOS IF(INXBP(K).GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX.NBPX.IBPX0) NUPERM=0 DO 220 I=2.NBPX DD 220 I=2.NBPX DD 220 I=1.KBPOS IPOINT(1,II)=II 225 CONTINUE NUM=IBPX0(I) NSUM=0 DO 230 J=1.NUMX JJ=NUMX-J JJJ=NUMX-J JJJ=NUMX-JJ NUM=NUM-JJJ+10**JJ | INTERP | 40 41 42 34 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 89 01 20 10 10 10 10 10 10 10 10 10 10 10 10 10 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00043 00044 00045 00044 00045 00044 00045 00046 00047 00048 00049 00049 00050 00051 00052 00055 00055 00055 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS,NBPERM,IBPERM) DO 205 K=1.K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1.KBPOS IF(NXBP(K).GE. 0) THEN NUMX=NUMX+1 LOCX(NUHX)=K ENDIF 210 CONTINUE MBPX=2**NUMX CALL COMBO(NUMX,NBPX,IBPX0) NUPERM=0 DO 220 I=2,NBPX DO 225 II=1.KBPOS IPOINT(I,II)=II 225 CONTINUE NUM=1BPX0(I) NSUM=0 DO 230 J=1.NUMX JJ=NUMX-J JJJ=NUMX-J JJJ=NUMX-J JJJ=NUMX-J JJJ=NUMAJJ+10**JJ NUM=NUM-JJJ+10**JJ | INTERP | 40 44 44 44 44 44 44 44 44 44 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 00045 00044 00045 00046 00047 00048 00049 00048 00049 00050 00051 00052 00053 00055 00055 00055 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1,KBPOS IF (HXBP(K) .GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX, NBPX, IBPX0) NUPERM=0 DO 220 I=2,NBPX DO 225 II=1,KBPOS IPOINT(I,II)=II 225 CONTINUE NUM=IBPX0(I) NSUM=0 DO 230 J=1.NUMX JJ=NUMX-J JJ=NUMX-J JJ=NUMX-J IF(JJJ .EQ. 1) THEN NSUM=NSUM+1 | INTERP | 40 44 44 44 44 44 44 44 44 44 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00041 00042 00043 00044 00045 00044 00045 00046 00047 00048 00049 00051 00051 00051 00055 00055 00055 00055 00055 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM. IBPERM) DO 205 K=1, KBPOS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMM=0 DO 210 K=1, KBPOS IF(NXBP(K).GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX, NBPX, IBPX0) NUPERM=0 DO 220 I=2, NBPX DO 225 II=1.KBPOS IPOINT(I,I)=II 225 CONTINUE NUM=1BPX0(1) NSUM=0 DO 220 J=1.NUMX JJ=NUMX-J JJ=NUMX-J JJ=NUMX-J IF(JJJ.E3.1) THEN NSUM=NSUM+1 IPTEM=IPOINT(I,NSUM) | INTERP | 40 44 44 44 44 44 44 44 44 44 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 00045 00044 00045 00046 00047 00048 00049 00051 00052 00053 00054 00055 00056 00057 00058 00059 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1.K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1.KBPOS IF(NXBP(K).CE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX,NBPX, IBPX0) NUPERM=0 DO 220 I=2.NBPX DO 225 II=1.KBPOS IPOINT(1.I)=II 225 CONTINUE NUM=IBPX0(1) NSUM=0 DO 220 J=1.NUMX JJ=NUMX-J JJ=NUMX-J JJ=NUMX-J IF(JJ).C3.1) THEN NSUM=NSUM+1 IPTEM=IPOINT(I,NSUM) JPOS=LOCX(J) | INTERP IN | 40 44 43 44 55 55 55 55 55 55 55 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 00045 00044 00045 00044 00045 00048 00049 00051 00052 00053 00054 00055 00055 00055 00055 00055 | ENDIF 100 CONTINUE C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS,NBPERM.IBPERM) DO 205 K=1.K3POS IFOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1.KBPOS IF(NXBP(K).GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX,NBPX.IBPX0) NUPERM=0 DO 220 I=2.NBPX DO 225 II=1.KBPOS IPOINT(1.II)=II 225 CONTINUE NUM=IBPX0(1) NSUM=0 DO 230 J=1.NUMX JJ=NUMX-J IPOINT(1.NSUM) IPOINT(1.NSUM) IPOINT(1.NSUM) IPOINT(1.NSUM) | INTERP | 40 44 42 34 45 46 78 90 12 34 55 55 55 55 55 56 66 66 66 66 66 66 66 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 00045 00044 00045 00045 00055 00055 00055 00055 00055 00055 00055 00055 00055 00055 00055 00055 | <pre>ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINCS C CALL COMBO(XBPOS,NBPERM,IBPERM) DO 205 K=1.X3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1.KBPOS IF(NXBP(K).GE. 0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX,NBPX,IBPX0) NUPERM=0 DO 220 I=2,NBPX DO 225 II=1.KBPOS IPOINT(1,I)=II 225 CONTINUE NUM=IBPX0(1) NSUM=0 DO 230 J=1.NUMX JJ=NUMX-J JJJ=NUMX-J JJJ=NUMX-J IF(JJJ.E3.1) THEN NSUM=NSUM+1 IPTEM=IPOINT(I,NSUM) JPOS=LOCX(J) IPOINT(I,INSUM)=IPOINT(I,JPOS) IPOINT(I,JPOS)=IPTEM</pre> | INTERP IN | 40 41 42 34 45 67 89 01 23 45 67 89 01 23 45 67 77 77 77 77 77 77 77 77 77 77 77 77 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00043 00044 00044 00045 00044 00045 00044 00045 00045 00051 00052 00055 00056 00057 00058 00059 00061 00061 00062 | ENDIF 100 CONTINUE C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS,NBPERM.IBPERM) DO 205 K=1.KBPOS IPOINT(1.K)=K 205 CONTINUE C INCLUDE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1.KBPOS IF(NXBP(K).GE.0) THEN NUMX=MUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX.NBPX.IBPX0) NUPERM=0 DO 220 I=2.NBPX DO 220 I=2.NBPX DO 220 I=2.NBPX DO 220 I=2.NBPX DO 220 J=1.NUMX JJ=NUMY10*JJ NUM=IBPX0(1) NSUM=0 DD 230 J=1.NUMX JJ=NUMY10*JJ NUM=NUM-JJJ*10**JJ IF(JJJ.EQ.1) THEN NSUM=NUM-JJJ*10**JJ IF(JJJ.EQ.1) THEN NSUM=NUM-JJJ*10**JJ IF(JJJ.EQ.1) THEN NSUM=NUM-JJJ*10**JJ IF(JJJ.EQ.1) THEN NSUM=NUM-JJJ*10**JJ IPOINT(I,NSUM)=IPOINT(I,JPOS) IPOINT(I,JPOS)=IPTEM ENDIF | INTERP IN | 40 44 42 34 45 67 89 01 23 45 67 89 01 23 45 67 89 01 23 45 67 77 77 77 77 77 77 77 77 77 77 77 77 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 00045 00044 00045 00044 00045 00045 00053 00055 | ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS,NBPERM.IBPERM) D0 205 K=1,K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUBE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 D0 210 K=1,KBPOS IF(NXBP(K).GE.0) THEN NUMX=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE MBPX=2**NUMX CALL COMBO(NUMX,NBPX.IBPX0) NUPERM=0 D0 220 I=2,NBPX D0 225 II=1,KBPOS IPOINT(I,II)=II 225 CONTINUE NUM=IBPX0(1) NSUM=0 D0 230 J=1,NUMX JJ=NUMX/J*JJ NUM=NUM-JJ=10**JJ NUM=NUM-JJ=10**JJ NUM=NUM-JJ=10**JJ NUM=NUM-JJ=10**JJ NUM=NUM-JJ=10**JJ NUM=NUM-JJ=10**JJ IF(JJJ.EQ.1) THEN NSUM=NSUM+1 IPTEM=IPOINT(I,NSUM) JPOS=LOCX(J) IPOINT(I,NSUM)=IPOINT(I,JPOS) IPOINT(I,NSUM)=IPOINT(I,JPOS) IPOINT(I,NSUM)=IPTEM ENDIF 230 CONTINUE | INTERP IN | 40 41 42 34 45 67 89 55 55 55 55 55 55 55 55 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00040 00041 00042 00043 00044 00045 00044 00045 00044 00045 00046 00045 00051 00052 00055 | <pre>ENDIF 100 CONTINUE C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS, NBPERM, IBPERM) DO 205 K=1.K3POS IPOINT(1,K)=K 205 CONTINUE C INCLUDE CONDINATIONS WITH 'EXTRA' BP POSSIBILITIES NUMX=0 DO 210 K=1.KBPOS IF(NXBP(K).GE. 0) THEN NUMX=NUMX+1 LOCK(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL CONBO(NUMX, NBPX, IBPX0) NUMPERM=0 DO 220 I=2.NBPX DO 225 II=1.KBPOS IPOINT(1,II)=II 225 CONTINUE NUM=IBPX0(I) NSUM=0 DO 220 J=1.NUMX JJ=NUMX-J JJ=NUMX-J JJ=NUMX-J IF(JJJ.EQ. 1) THEN NSUM=NSUM+1 IF(JJJ.EQ. 1) THEN NSUM=NSUM+1 IPOINT(I,NSUM)=IPOINT(I,JPOS) IPOINT(I,JPOS)=IPTEM ENDIF 230 CONTINUE NUMEX-KBPOS-NSUM</pre> | INTERP IN | 40 44 42 34 45 67 89 01 23 45 55 55 55 55 55 55 55 55 55 55 55 55 |
| 00029 00030 00031 00032 00033 00034 00035 00036 00037 00038 00039 00040 00041 00042 00043 00044 00045 00044 00045 00046 00047 00048 00049 00050 00051 00052 00055 | <pre>ENDIF 100 CONTINUE C C PRODUCE ALL POSSIBLE COMBINATIONS OF INTEGRAL BP LOADINGS C NBPERM=2**KBPOS CALL COMBO(KBPOS,NBPERM.IBPERM) DO 205 K=1.K3POS IPDINT(1,K)=K 205 CONTINUE C INCLUBE COMBINATIONS WITH 'EXTRA' BP POSSIBILITIES NUM%=0 DO 210 K=1.KBPOS IF(NXBP(K).GE. 0) THEN NUM%=NUMX+1 LOCX(NUMX)=K ENDIF 210 CONTINUE NBPX=2**NUMX CALL COMBO(NUMX.NBPX.IBPX0) NUMERM=0 DO 220 I=2,NBPX DO 225 II=1.KBPOS IPOINT(I.II)=II 225 CONTINUE NUMERD DO 220 J=1.NUMX JJ=NUMX-J JJ=NUMX-J JJ=NUMX-J JJ=NUMX-I IF(JJJ.G2.1) THEN NSUM=0 DO 220 J=1.NUMX JJ=NUMX(I.NSUM) POS=LOCX(J) IPOINT(I.NSUM)=IPOINT(I.JPOS) IPOINT(I.JPOS)=IPTEM ENDIF 230 CONTINUE NUMEXERDOS-NSUM NBPEX=2**NUMEX</pre> | INTERP | 40 44 43 44 55 55 55 55 55 55 55 55 55 55 55 55 |

| 00067 | DO 250 M=1, NBPEX | | 82 |
|-------|--|-------|-------|
| 00068 | IBPERM(NBPERM+NUPERM)=INT((2.0/9.0)+10++NSUM)+10++(KBPOS-NSUM) | INTBP | 84 |
| 00070 | 1 +IBPEX(M) | INTEP | 85 |
| 00070 | IPOINT(NBPERM+NUPERM,K)=IPOINT(I,K) | INTEP | 87 |
| 00072 | 255 CONTINUE | INTEP | 88 |
| 00073 | 250 CUNIINUE 220 CONTINUE | INTEP | 90 |
| 00075 | DO 245 M=1, NBPERM | INTBP | 91 |
| 00076 | DO 245 K=1.KBPOS | | 30 |
| 00077 | 245 CONTINUE | INTEP | 94 |
| 00079 | NBPERM=NBPERM+NUPERM | INTEP | 95 |
| 00080 | 270 CONTINUE | INTEP | 97 |
| 00081 | DO 300 M=1,NBPERM | INTBP | 98 |
| 00085 | DEPL(M)=.TRUE. | INTEP | 100 |
| | C DETERMINE THE ACTUAL BP LOADING FROM THE BP IDENTIFIER 'IBPERM' | INTBP | 101 |
| 00083 | | INTBP | 102 |
| 00084 | DO 350 KK= <bpos-1,0,-1< td=""><td>INTBP</td><td>104</td></bpos-1,0,-1<> | INTBP | 104 |
| 00085 | K=IPOINT(M,KBPOS-KK) | INTEP | 105 |
| 00085 | NOLDBP=NOLDBP-IBPID+10++KK | INTBP | 107 |
| 00088 | IF(IBPID .EQ. 0) THEN | INTEP | 108 |
| 00089 | IBP(M,K)=MLUBP(K) FISEIF(IBPID_EQ. 1) THEN | INTEP | 110 |
| 00091 | IBP(M.K)=NHIBP(K) | INTBP | 111 |
| 26000 | ELSEIF(IBPID .EQ. 2) THEN IBP(M.K)=NXBP(K) | INTEP | 113 |
| 00094 | ENDIF | INTEP | 114 |
| 00095 | 350 CONTINUE | INTEP | 115 |
| | C CONSTRUCT POWER DISTRIBUTION USING LINEAR APPROXIMATION | INTBP | - 117 |
| | C COMPUTE THE DIFFERENCE, (XBR(I,J)-PBASE(I,J)) | INTER | 118 |
| 00096 | C TOTAL =0 | INTEP | 120 |
| 00097 | DO 400 I=1, IMAX | INTEP | 12 |
| 00098 | | INTEP | 123 |
| 00100 | | INTBP | 124 |
| 00101 | PPBAS=PBASE(I,J) | INTEP | 12: |
| 00102 | DU 420 K=1,KBPUS PPBAS=PPBAS+PDERIU(I.J.K)*(IBP(M.K)-RNBP(K)) | INTBP | 12 |
| 00104 | 420 CONTINUE | INTEP | 129 |
| 00105 | PPOW(I,J)=PPBAS | INTEP | 13 |
| 00107 | 410 CONTINUE | INTBP | 13 |
| 00108 | 400 CONTINUE | INTEP | 13 |
| 00109 | RETURN | INTBP | 134 |
| 00111 | | INTBP | 13 |
| | | | |

| 00001 | | SUBPOUTINE OPRES (NRP. NEPERM. NEP1. NEP2, TEMP, ITEMP, EXP, JMX) | OPRES 2 |
|-------|---------|---|----------------------|
| 00002 | | DIMENSION NBP(15,15), EN(8), NOBP(8), EXP(8,8), JMX(34) | OPRES 3 |
| 00003 | | CALL FACE | UPRES 4 |
| 00004 | | HRITE(6,50) | OPRES 6 |
| 00005 | 50 | FORMAT(1H1,15H UPUS KESULIS //,13(10*// | OPRES 7 |
| 00005 | 100 | FORMAT(///. 5X. 45H TOTAL NUMBER OF INTEGRAL BP COMBINATIONS | OPRES 8 |
| 00001 | 100 | *15, /, 5X, 45H NUMBER OF DEPLETABLE BP COMBINATIONS , 15, | OPRES 9 |
| | | */, 5X, 45H NUMBER OF ACCEPTABLE BP COMBINATIONS , 15, ///) | 00055 11 |
| 80000 | . * ~ ~ | WRITE(6, 150) ITEMP, TEMP | OPRES 12 |
| 00009 | 150 | | OPRES 13 |
| 00010 | | WRITE(6,200) | OPRES 14 |
| 00011 | 200 | FORMAT(/, 25X, 16(1H*), /, 25X, 16H* ENRICHMENT *, /, 25X, | 09955 15 |
| 00017 | | *16H* BP LOADING *,/,25X,16H* BOC EXPUSURE *,/,25X,16(17*)) | OPRES 17 |
| 00012 | | 10 350 1=1,6 10 350 J=1.JMX(T) | OPRES 18 |
| 00014 | | NOBP(J)=MOD(NBP(I,J),1000)/10 | OPRES 19 |
| 00015 | | EN(J)=(MOD(NBP(I,J),1000000)/10000)/100.0 | UPRES 20 |
| 00015 | 350 | CONTINUE CD TO (DEE 200 DEE 270 275 280 DEE 290) I | OPRES 22 |
| 00017 | -255 | U (U (332,360,363,370,373,360,363,360,1 HOITE(C.352) | OPRES 23 |
| 00019 | 356 | FORMAT(1X,10(1H*)) | OPRES 24 |
| 00020 | | GO TO 395 | OPRES 25 |
| 00021 | 360 | WRITE(6,361) | OPRES 27 |
| 00022 | 361 | | OPRES 28 |
| 00024 | 365 | WRITE(6,366) | OPRES 29 |
| 00025 | 366 | FORMAT(1X,28(1H*)) | OPRES 30 |
| 00025 | | GO TO 395 | |
| 00027 | 370 | WRITE(6,371) | OPRES 33 |
| 00028 | SUT | GO TO 395 | OPRES 34 |
| 00030 | 375 | WRITE(6,376) | OPRES 35 |
| 00031 | 376 | FORMAT(1X,46(1H*)) | 02055 30 02055 37 |
| 00032 | 280 | G0 T0 395 | OPRES 38 |
| 00033 | 381 | FORMAT(1X,55(1H+)) | OPRES 39 |
| 00035 | | GO TO 395 | OPRES 40 |
| 00036 | 385 | WRITE(6,386) | OPRES 42 |
| 00037 | 386 | FUKMAI(IX, 35(17*)) | OPRES 43 |
| 00039 | 390 | WRITE(6,391) | OPRES 44 |
| 00040 | 391 | FORMAT(1X, 55(1H+)) | OPRE5 45 |
| 00041 | 395 | CONTINUE | 000000 47 |
| 00042 | 400 | WRIE(6,400) (EN(J),J=1,JNX(I)) | OPRES 48 |
| 00043 | 400 | GO TO (405,410,415,420,425,430,435,440) I | OPRES 49 |
| 00045 | 405 | WRITE(6,406) | OPRES 50 |
| 00046 | 406 | FORMAT(1X, 1H+, 8X, 1H+) | OPRES 51 |
| 00047 | | GO TO 445 | OPRES 53 |
| 00048 | 410 | HK1(E(6)411) FORMAT(1Y, 1He, 2(8X, 1He)) | OPRES 54 |
| 00050 | | GO TO 445 | OPRES 55 |
| 00051 | 415 | WRITE(6,416) | OPRES SE |
| 00052 | 416 | FORMAT(1X, 1H+, 3(8X, 1H+)) | |
| 00053 | 420 | URITE(S.421) | OPRES 55 |
| 00055 | 421 | FORMAT(1X, 1H*, 4(8X, 1H*)) | OPRES 60 |
| 00055 | | GO TO 445 | OPPES 62 |
| 00057 | 425 | WRITE(5,425) SORMAT(1Y,1H=,5(RY,1H=)) | OPRES 63 |
| 00058 | 425 | CO TO 445 | OPRES 64 |
| 00000 | 430 | WRITE(6,431) | OPRES 65 |
| 00061 | 431 | EORMAT(1X,1H+,6(8X,1H+)) | UPKLS bi |
| 23000 | | GO TO 445 | OPRES A |
| 00063 | 435 | WKIELU0:430/ SORMAT(1X,1H+,S(8X,1H+)) | OPRES 65 |
| 00065 | | G0 T0 445 | OPRES 70 |
| 00066 | 440 | URITE(6,441) | OPRES 7 |
| 00067 | 441 | FURMAT(1X, 1H+, 4(8X, 1H+)) | UPRES 70 |
| 83000 | 445 | | OPRES 7 |
| 00053 | | IF(NOBP(J)), LE. 0) GO TO 450 | OPRES 7 |
| 00071 | | GO TO (500, 550, 600, 650, 700, 750) J | OPRES 71 |
| 00072 | 500 | WRITE(6,525) NOBP(J) | OPRES 7 |
| 00073 | 525 | FORMAT(1H+, 1X, 17) | OPRES 7 |
| 00074 | 550 | URITE(6.575) NORP(1) | OPRES 8 |
| 00076 | 575 | FORMAT(1H+, 10X, I7) | OPRES 8 |
| | | Maxim dia 1990 and an ann | | | and the plant of the | (小学) 전문을 다 나는 것이다. |
|--|---|--|--|---|---|--------------------|
| | | | | and the second second | | |
| | | No. A start of the second second | | | a de la trata de la composición de la c | |
| | | | · · · | | | |
| | ч., ¹ ., | | | | | |
| | | | | | | |
| | | | | | | |
| 00077 | | CD TD 450 | | | | |
| 00078 | 500 | UPITE(6,625) NOBO(| 15 | | | OPRES 82 |
| 00079 | 625 | FORMAT(144,194,17) | | | | OPRES 83 |
| 00080 | | | | | | UPRES 84 |
| 00081 | 650 | UPTTE(6.675) NORD(| 6 - C. | | | OPRES 85 |
| 00082 | 675 | EURMAT(1H+.28%.17) | J) | | | UPRES 85 |
| 00083 | | GO TO 450 | | | | UPRES 87 |
| 00084 | 700 | URITE (6.725) NORPE | n | | | UFKE3 88 |
| 00085 | 725 | FORMAT(1H+.37X.17) | J, | | | 08855 03 |
| 00086 | | GO TO 450 | | 1. State 1. | | |
| 00087 | 750 | HRITE(S.775) NORPO | 15 | | | 08055 31 |
| 00088 | 775 | EDRMAT(1H+.46X.17) | | | | UFRE3 32 |
| 00089 | 450 | CONTINUE | | | | 08856 94 |
| 00090 | | WRITE(6.800) (EXP(| T. D. HET. MY(T)) | | | 00050 95 |
| 00091 | 800 | FORMAT(1X+1H++8(F7 | 3.24 +)) | | | |
| 28000 | 300 | CONTINUE | | | the provide states of the second | |
| 00093 | | HRITE(6,850) | 1 | and the second | | OPPES 98 |
| 00094 | 850 | FORMAT(1X, 37(1H+)) | | | | OPRES 99 |
| 00095 | | RETURN | at the second | | | OPRES 100 |
| 00035 | 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - | END | | | | OPRES 101 |
| | | | | | | 승규는 가장 가장 가지? |
| 8 C. S. S. S. | | | | | | |
| | 이 사람 | | | | Constraints of the second | |
| | | | | | en de la secola de l | |
| | - 1 · · · · · | | | | | |
| | | | | | A second sec second second sec | |
| e, i | | | | | | |
| 1997 - A.S. | | | | | | |
| | | | | | | |
| | 1.1.1.1.1.1 | | | | | |
| | | | | | | |
| 1 | | | | a de la companya de l | | |
| 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1 | | | and the second second | | | |
| | | | | | and the second | |
| | | and the second | | | and the second | |

| 00001 00002 00003 00004 00005 00005 00005 00005 00008 00009 00010 00011 00012 00013 00014 | 60 50 | SUBROUTINE COMBO(X, Y, A) INTEGER X, Y, A(Y) DO SO M=1, Y NUMBER=M-1 IX=0 DO GO KK=1, X K=X-KK IDELX=NUMBER/2**K IX=IX+IDELX*10**K NUMBER=NUMBER-IDELX*2**K CONTINUE A(M)=IX CONTINUE RETURN |
|---|----------|---|
| 00015 | | END |
| | | |

| COMBO | 5 |
|-------|------|
| COMBO | 3 |
| DEMO: | . 4 |
| | 2 |
| | 7 |
| | ់ត្ត |
| OMBO | ğ |
| COMBO | 10 |
| OMBO | 11 |
| COMBO | 12 |
| COMBO | 13 |
| COMBO | 14 |
| COMBO | .15 |
| COMBO | 16 |

APPINDIX B: ADDITIONAL INPUT REQUIREMENTS

A. lard Type 1

| | J(39) | PPLIM | Nodal power peaking limit | |
|---|---------|-----------|---|--------------|
| | J(46) | MOPTIM | Control of optimization flow: =1 - direct search only =2 - BP search only =3 - combined (currently = | mavailable) |
| | .1 (47) | JBPSOL | BP search method. =0 fresh fuel search only =1 linear programming | |
| | 1(48) | JBPEDIT | Output edit control =0 normal operation =1 for debugging purposes | |
| Ł | lard Ty | pe 2 | | |
| | 12(50) | BPDEL. | Burnable poison tolerance (0.5 is recommended) | |
| • | 12(51) | PPMULT1 | Reconstructed power peaking (1.02 is recommended) | zolerance |
| | 12(52) | PPMULT2 | SIMULATE power peaking tole== (1.00 is recommended) | Ince |
| • | Card Ty | rpe 17 | | |
| | (BR(i, | j) | Target power distribution | |
| • | Card T | ype 32 | | |
| | NBP(i, | j) | Burnable poison identifier of the form ABBBCDDD. where, | |
| | | | A = fuel type $(1 - 15x)^{-1}$ 2 - 15x1^{-1} | sta, ofa) |

69

 $\begin{array}{rcl} & 2 & -15x1 \\ & 2 & -15x1 \\ \end{array} & \text{of a} \end{array}$ BBB= enrichment (e.g. 2. w/o - 270) C = BP type (1 - glass, 1 - WABA) DDD= number of BP's (e.g. 12 BP - 120) In the highly unlikely situation that none of the candidate BP loadings successfully depletes to end-of-cycle, several courses of action are available. Corrective actions should be pursued in the following order:

- 1. increasing the BP tolerance, BPDEL, to create more candidate BP loadings to be evaluated
- 2. increasing the reconstructed power peaking tolerance, PPMULT1, to allow more cases to be passed on to the normal power calculation / depletion
- 3. increasing the SIMULATE power peaking tolerance, PPMULT2, to allow a higher power peaking to be acceptable for a final design pattern

It is much more likely that the opposite situation would occur, that too many cases reach the depletion stage such that the computation costs become excessive. In this case, the above actions should be reversed (i.e. instead of increasing tolerances, decrease them). However, the same order still applies to these alternatives.

APPENDIX C: SAMPLE INPUT LISTING

15411, BBL, CY, L10000. RESOURCE (JCAT=S2) ATTACH, SIMTRN. SIMTRN. 8 8 ZION-1 CYCLE 9 SIMULATION (2-D MODEL, FALRE OPTION) 47 23 2 2 2 I ****************** n ZION-1 CYCLE-S 2-D SIMULATION, MODERATOR FEEDBACK+ FLARE OPTION USED WITH INODE/1 ASSEM 0 ٠ INCLUDED, BUILT-IN BPMODEL USED, SO EVERY XS IS FOR 0 BP FUEL TYPE MAR. 17, 1986 0 a FUEL TYPE FUEL SHUFFLING USING DIRECT SEARCH METHOD n -Ó ***** 1 0.0 11.5 11.5 12.01 16.875 178.6 2250. 0 S1 0.0 S3 82.17 S2 .981 .981 1.0 21.608 1 -1 -1 0 0 3 12.0 10.804 S1 -1 t 12.01 16.875 99.56 n 1 0 1 1 10 11 13.563 2250. n .73634 1.33 3 4 .5 1 1 20 0 1 0 5 1.27 3 2 n 1 0 2 .0005 4 .00005 12 2 .00005 4 .0001 0 0 0 1 0 1.0 0. 1.0 0.6 0 1.0 4 1.2 .6 52 .35 1.0 0 2 18 1 00004 6 15 0 0 0 0 0 20.0 SE 0.455 S2 3 1 30.48 S13 S9 6.241E+18 2.679 52 4 0.0 3 .00633 S2 .211E-4 .01255 S2 .385E-5 0.0 S2 0.0 2 30.48 513 S9 6.241E+18 2.672 S2 4 0.0 .06405 .00625 S2 .211E-4 .01255 S2 .385E-5 0.0 S2 0.0 -06414 S2 .288E-4 2 S6 0.455 S2 3 3 2 -52 .06405 .288E-4 0.455 S2 .288E-4 3 3 S6 3 30.48 S13 59 6.241E+18 2.721 52 4 0.0 3 .00670 52 .211E-4 .01 4 30.48 513 59 6.241E+18 .06420 S2 3 .01277 S2 .385E-5 0.0 S2 0.0 3 4 SE .455 52 3 4 2.4428E+00 1.4184E-02 -1.3229E-04 4 0.0 3 4 6.3547E-02 1.2444E-05 2.4107E-07 .288E-4 3 4 2.7119E-03 2.2301E-04 -2.6877E-06 .211E-4 3 4 1.1340E-02 6.2331E-05 -4.5584E-07 .385E-5 0.0 S2 0.0 5 56 .455 52 5 30.48 513 59 6.241E+18 5 2.4421E+00 1.3728E-02 -1.2446E-04 4 0.0 3 3 3 6.3550E-02 1.1127E-05 2.57622-07 .2882-4 3 5 2.1710E-04 -2.5757E-06 .211E-4 2.6939E-03 1.13362-02 3 5 6.0015E-05 -4.2172E-07 .385E-5 0.0 S2 0.0 3 6 S6 .455 S2 6 30.48 S13 S9 6.241E+18 3 6 2.4396E+00 1.2124E-02 -9.8852E-05 4 0.0 3 6.3561E-02 6 6.8125E-06 3.0254E-07 .288E-4 1.9567E-04 -2.1907E-06 .211E-4 3 6 2.6363E-03 3 1.1321E-02 5.1940E-05 -2.9976E-07 .385E-5 0.0 S2 0.0 6 4 1 з 3 4 4 2 ۵ R 4 3 6 6 6 3 4 3 5 4 5 2 6 6 6 4 4 5 4 5 2 5 5 6 3 5 4 3 6 2 5 2 4 6 4 6 3 3 6 2 4 5 6 ۵ 4 6 6 6 6 6 4 8 1 6 8 3 5 : 1 1 11001203 0 1.5000E-01 1.0000E+00 2.0000E+00 4.0000E+00 5 0 1.5000E-01 1.0000E+00 2.0000E+00 4.0000E+00 6.0000E+00 8.0000E+00 1.0000E+01 1.2000E+01 1.4000E+01 1.6000E+01 1.8000E+01 2.0000E+01 2.4000E+01 2.8000E+01 3.2000E+01 3.6000E+01 4.0000E+01 4.5000E+01 5.0000E+01 2.3504E-01 2.353E-01 2.3584E-01 2.3561E-01 2.3534E-01 2.3499E-01 2.3467E-01 2.3443E-01 2.3396E-01 2.3346E-01 2.3297E-01 2.3253E-01 2.3217E-01 2.3222E-01 2.3322E-01 2.3408E-01 2.3489E-01 2.3563E-01 2.3640E-01 2.3704E-01 5 1 1 5 1 1 1 1 5 1 1 1 1 5 1 5 2 1 0 0 1 20 3 0 1.5000E-01 1.0000E+00 2.0000E+00 4.0000E+00 5 1 5.0000E+00 8.0000E+00 1.0000E+01 1.2000E+01 1.4000E+01 1.5000E+01 1.8000E+01 2.0000E+01 2.4000E+01 2.8000E+01 1 1 3.2000E+01 3.6000E+01 4.0000E+01 4.5000E+01 5.0000E+01 1 8.9364E-03 8.9471E-03 8.9775E-03 9.0403E-03 9.1868E-03 5 2 9.3689E-03 9.5361E-03 9.6798E-03 9.8195E-03 9.9477E-03 1.0064E-02 1.0172E-02 1.0272E-02 1.0454E-02 1.0619E-02 5 1 1.0769E-02 1.0909E-02 1.1041E-02 1.1189E-02 1.1326E-02 5 2 1 5 3 31001203 0 1.5000E-01 1.0000E+00 2.0000E+00 4.0000E+00 з 5 6.0000E+00 8.0000E+00 1.0000E+01 1.2000E+01 1.4000E+01 3 1 1.6000E+01 1.8000E+01 2.0000E+01 2.4000E+01 2.8000E+01 5 3 1 5 3.2000E+01 3.6000E+01 4.0000E+01 4.5000E+01 5.0000E+01 1.7108E-02 1.7157E-02 1.7138E-02 1.7117E-02 1.7035E-02 5 1 1.6924E-02 1.6822E-02 1.6724E-02 1.6682E-02 1.6661E-02 1 1.6650E-02 1.6645E-02 1.6642E-02 1.6655E-02 1.6677E-02 1 1.6710E-02 1.6745E-02 1.6781E-02 1.6819E-02 1.6849E-02 3 5 3 5

5 8 6 1.3776E-01 1.3571E-01 1.3351E-01 1.2874E-01 1.2370E-01 1.1861E-01 1.1361E-01 1.0885E-01 1.0375E-01 9.9153E-02 5 8 6 91001203 ğ 5 6 0 1.5000E-01 1.0000E+00 2.0000E+00 4.0000E+00 5 q 6 5.0000E+00 8.0000E+00 1.0000E+01 1.2000E+01 1.4000E+01 5 9 6. 1.6000E+01 1.8000E+01 2.0000E+01 2.4000E+01 3.2000E+01 3.6000E+01 4.0000E+01 4.5000E+01 ŝ 2.8000E+01 9 6 ŝ ġ 6 5.0000E+01 1.1007E+01 1.0985E+01 1.1075E+01 1.1148E+01 1.1222E+01 ŝ 9 6 1.0742E+01 5 9 1.1214E+01 1.1151E+01 1.1048E+01 1.0906E+01 6 1.0562E+01 1.0371E+01 1.0172E+01 9.7534E+00 9.3264E+00 8.9043E+00 8.4972E+00 8.1147E+00 7.7077E+00 7.3458E+00 5 9 6 5 9 6 7.22 1 0 0 1 20 3 (XE2 3.20 0 BP 510 Ĝ 1.0000E+00 510 ۵ 1.0000E-01 Ē 2.0000E+00 4.0000E+00 510 6.0000E+00 8.0000E+00 1.0000E+01 1.2000E+01 1.4000E+01 Ĝ 1.5000E+01 1.8000E+01 2.4000E+01 510 2.0000E+01 2.8000E+01 6 4.8000E+01 3.2000E+01 4.0000E+01 510 6 3.6000E+01 4.4000E+01 1.4680E+06 510 6 1.4554E+06 1.4460E+06 1.4562E+06 1.4599E+06 1.4785E+06 1.4911E+06 1.4928E+06 510 6 1.5011E+06 1.4825E+06 1.52562+06 1.4999E+06 1.5132E+06 1.5167E+06 1.5367E+06 510 6 1.5758E+06 (SM2) 1.5552E+06 7.24 1 0 0 SIO 6 1.56622+06 1.5852E+06 1.5929E+06 511 3.20 0 BP 1 6 20 3 0 1.0000E-01 1.0000E+00 511 6 2.0000E+00 4.0000E+00 8.0000E+00 1.0000E+01 1.2000E+01 1.4000E+01 511 6 6.0000E+00 2.8000E+01 1.6000E+01 511 6 1.8000E+01 2.0000E+01 2.4000E+01 511 6 3.2000E+01 3.6000E+01 4:0000E+01 4.4000E+01 4.8000E+01 4.2382E+04 4.3413E+04 4.2901E+04 4.3958E+04 4.3166E+04 4.3721E+04 511 8 4.2574E+04 4.2981E+04 4.3719E+04 511 4.3471E+04 6 4.3889E+04 4.4787E+04 511 ŝ 4.4219E+04 4.4530E+04 4.4291E+04 4.5254E+04 4.5527E+04 4.5770E+04 4.6008E+04 4.5204E+04 511 6 7 8 1 0 0 1 2 (BORON2) 3.20 512 6 512 1.4544E-05 6 0 0 1.5523E-08 -1.5057E-10 0 (DOP1) 310023 3.20 513 6 2 513 513 00000000 6 -4.1533E-05 1.3450E-06 -8.1371E-08 1.3755E-09 6 2 3 1 0 0 1 3 (DOP2) 3.20 514 6 514 0000 3.1933E-04 -9.7971E-06 6.3531E-07 -1.1467E-08 6 0 3 0 (DOP3) 3.20 515 1 0 0 6 5 -2.7780E-04 515 8.4521E-06 -5.5394E-07 6 1.0092E-08 2.0440E-04 7.3374E-03 516 517 240 740 0 0 -1.5051E-04 3.20 1 n (DEL AB1) 6 3.20 -5.4028E-03 6 0 0 1 0 (DEL AB2) -1.1836E-01 3.20 518 1.6074E-01 6 1 4 0 0 0 1 ٥ (DEL TR1) 519 -7.0851E-01 3.20 4 0 0 0 1 0 9.6220E-01 (DEL TR2) 6 6 3.20 520 6 3 4 0 0 0 1 0 -2.0335E-02 2.76202-02 (DEL REMU) 3.20 2.8985E-07 (BORON1) 521 8 2 8 00010.0 0.0 2.22 0 0 0 0 0 0.0 1.0340E+02 3.20 522 8 (XE1) 3.20 6 2.24 0 0 0 0 0 0.0 7.9930E+01 (SMI) 523 -.20 RS. 0.0 7 Ż 0.0 RS -.15 3 RS. 0.0 R6 R5 777 4 0.0 5 -.15 0.0 -.15 67 **R5** 0.0 77 R3 0.0 -.15 15 Ŕ -.20 .15 .15 1F7 2F7 188 289 208 168 1H7 269 2H8 10 1Å7 107 1 D9 127 208 309 2E8 3E7 4E9 188 1C7 10 2 3F9 4F7 308 10 з 368 3H9 10 4 109 **2D8** 309 407 4G9 4H7 5 1E7 SE8 3E7 4E9 4F7 5F8 5G9 10 SE7 2F7 2G9 3F9 3G8 1F7 6F9 6 5F8 10 666 10 7 1**G**8 4G9 5G9 -6G6 2H8 3H9 10 8 1H7 4H7 20.620 10 1 1 2 12.960 10 1 10 3 25.880 10 1 4 a 25.890 10 1 5 10 1 67 24.940 10 9.870 1 10 1 8 20.520 10 งงงงง 1 12.960 23 10 n 10 11.410 12.240 4 10 10 5 ş 24.040 10 6 2 10 7 0 10 8 SUS 9.410 52.880 10 1 10 2 11.410

| 13.1 23.7 8.2 12.2 23.2 22.7 22.6 7.9 23.7 21.2 23.7 21.2 22.7 23.7 21.2 23.7 21.2 23.7 21.2 23.7 21.2 23.7 21.2 23.7 21.2 23.7 21.2 23.7 21.2 23.7 21.2 22.5 31.42 20.52 31.42 20.52 .0 .0 .1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 | | | 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 | 202 1 1 1 1 1 1 1 1 1 1 1 1 1 | 215 52 0000 0000 0000 0000 0000 0000 000 | 0 21 2280 2000 2280 2000 2320 2000 2320 2000 | 1 2080 2000 2120 0000 2040 0000 2080 0000 |
|--|---|--|---|--|---|---|---|
| | 13.1 23.7 8.2 12.2 23.2 22.7 23.7 21.27 24.9 21.27 21.27 23.7 21.27 24.9 21.27 31.42 9.8 31.42 9.8 31.42 9.8 9.41 0.20.62 9.41 22.55 9.41 23.7 11.1 11.1 11.1 11.1 12.55 000000 0000000 0000000 0000000 0000000 0000000 0000000 0000000 0000000 0000000 0000000 0000000 0000000 0000000 0000000 0000000 0000000 0000000 000 | $\begin{array}{c} 13.100\\ 0\\ 23.770\\ 0\\ 8.270\\ 0\\ 12.240\\ 0\\ 23.200\\ 23.200\\ 23.200\\ 23.200\\ 23.200\\ 23.770\\ 0\\ 22.630\\ 25.890\\ 7.910\\ 23.770\\ 0\\ 23.770\\ 0\\ 24.940\\ 24.040\\ 24.040\\ 24.040\\ 24.040\\ 24.040\\ 24.040\\ 24.040\\ 24.040\\ 3.810\\ 3.810\\ 0\\ 22.750\\ 12.770\\ 0\\ 31.420\\ 3.810\\ 0\\ 31.420\\ 3.810\\ 0\\ 31.420\\ 3.810\\ 0\\ 31.420\\ 3.810\\ 0\\ 31.420\\ 3.810\\ 0\\ 31.420\\ 3.810\\ 0\\ 31.420\\ 3.810\\ 0\\ 0\\ 31.420\\ 3.810\\ 0\\ 0\\ 31.420\\ 3.810\\ 0\\ 0\\ 0\\ 31.420\\ 3.810\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0$ | $\begin{array}{c} 13.100 \\ 0 \\ 23.770 \\ 0 \\ 8.270 \\ 0 \\ 12.240 \\ 0 \\ 23.200 \\ 23.200 \\ 23.200 \\ 23.200 \\ 23.200 \\ 23.750 \\ 22.750 \\ 12.770 \\ 0 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 31.420 \\ 9.810 \\ 0 \\ 22.750 \\ 12.770 \\ 0 \\ 31.420 \\ 9.810 \\ 0 \\ 22.530 \\ 0 \\ 0 \\ 31.420 \\ 9.810 \\ 0 \\ 22.530 \\ 0 \\ 0 \\ 31.420 \\ 9.810 \\ 0 \\ 22.530 \\ 0 \\ 0 \\ 0 \\ 31.420 \\ 9.810 \\ 0 \\ 22.530 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\$ | $\begin{array}{cccccccccccccccccccccccccccccccccccc$ | $\begin{array}{c} 13.100 \\ 0 \\ 23.770 \\ 0 \\ 0 \\ 8.270 \\ 0 \\ 12.240 \\ 0 \\ 23.200 \\ 23.200 \\ 23.200 \\ 23.200 \\ 23.200 \\ 23.200 \\ 22.750 \\ 22.630 \\ 25.830 \\ 7.910 \\ 23.770 \\ 0 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 20.520 \\ 9.810 \\ 0 \\ 31.420 \\ 9.810 \\ 0 \\ 31.420 \\ 9.810 \\ 0 \\ 31.420 \\ 9.810 \\ 0 \\ 22.630 \\ 0 \\ 0 \\ 31.420 \\ 9.810 \\ 0 \\ 22.630 \\ 0 \\ 0 \\ 31.420 \\ 9.810 \\ 0 \\ 22.630 \\ 0 \\ 0 \\ 31.420 \\ 20.520 \\ 9.410 \\ 0 \\ 22.630 \\ 0 \\ 0 \\ 0 \\ 0 \\ 31.420 \\ 20.520 \\ 9.410 \\ 0 \\ 0 \\ 31.420 \\ 20.520 \\ 9.410 \\ 0 \\ 0 \\ 31.420 \\ 20.520 \\ 9.410 \\ 0 \\ 0 \\ 22.630 \\ 0 \\ 0 \\ 31.420 \\ 20.520 \\ 9.410 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\$ | $\begin{array}{c} 13.100 \\ 0 \\ 23.770 \\ 0 \\ 0 \\ 8.270 \\ 0 \\ 23.200 \\ 23.200 \\ 23.200 \\ 23.200 \\ 23.200 \\ 23.200 \\ 22.750 \\ 0 \\ 22.750 \\ 12.770 \\ 0 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 24.940 \\ 25.830 \\ 0 \\ 31.420 \\ 9.810 \\ 0 \\ 0 \\ 31.420 \\ 9.810 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\$ | $\begin{array}{cccccccccccccccccccccccccccccccccccc$ |

n чm mm ოოო **m**mm ოოო លលល

4440 NNN00

(U N 00

.

JLAST

| 15411, BBL, CY, L100 RESOURCE (JCAT=S3) ATTACH, ZZSYM. ATTACH, SIMTRN. SIMTPN | 100. |
|---|--|
| 8 8 1 6 1 2 2 2 2 2 2 I === ZION-1 | 1 47 23 5 2 2 2 2 2 2 2 2 2 2 CYCLE 9 SIMULATION (2-D MODEL, FLARE OPTION) |
| | ZION-1 CYCLE-9 2-D SIMULATION, MODERATOR FEEDBACK INCLUDED, FLARE OPTION USED WITH INODE/1 ASSEM BUILT-IN BPMODEL USED, SO EVERY XS IS FOR 0 BP FUEL TYPE DEC.14,1986 BP SEARCH - PATTERN 2B |
| 0 1. 1. 1. | 0.0 18.0 2.0 12.01 16.875 178.6 2250. 0 S1 1000. S3 82.17 S2 .381 .381 1.0 21.608 10.804 |
| | S1 -1 -1 0 0 3 12.01 16.875 59.56 0 0 1 1 13.563 0 225073634 1.33 3 4 2 5 6 0 2 1 0 5 1.33 0 3 1 0 20 0 2 .0005 4 .00005 12 |
| 2 | .00005 4 .0001 1 0 0 0 0 1.0 1.0 0.6 0 1.0 4 1.2 .6 52 .35 1.0 0 5 7 18 1 0 0 0 0 4 6 15 0 0 0 0 20.0 |
| 2 3 3 1 3 2 3 2 3 2 | S6 0.455 S2 1 30.48 S13 S9 6.241E+18 2.679 S2 4 0.0 .06404 S2 .288E-4 .00633 S2 .211E-4 .01255 S2 .385E-5 0.0 S2 0.0 S6 0.455 S2 2 30.48 S13 S9 6.241E+18 2.672 S2 4 0.0 .06405 S6 0.455 S2 2 30.48 S13 S9 6.241E+18 2.672 S2 4 0.0 .06405 S2 .288E-4 .00626 S2 .211E-4 .01255 S2 .385E-5 0.0 S2 0.0 |
| 3 3 3 3 3 4 3 4 | S6 0.455 S2 3 30.48 S13 S9 6.241E+18 2.721 52 4 0.0 .06420 S2 .288E-4 .00670 S2 .211E-4 .01277 S2 .385E-5 0.0 S2 0.0 S6 .455 S2 4 30.48 S13 S9 6.241E+18 2 4428E+00 1.4184E-02 -1.3229E-04 4 0.0 |
| 3 4 3 4 3 4 | 6.3547E-02 1.2444E-05 2.4107E-07 .288E-4 2.7119E-03 2.2301E-04 -2.6877E-06 .211E-4 1.1340E-02 6.2331E-05 -4.5984E-07 .385E-5 0.0 S2 0.0 |
| 3 5 3 5 3 5 3 5 | 56.455 52 5 30.46 313 33 5.2462-04 4 0.0 2.4421E+00 1.3728E-02 -1.2446E-04 4 0.0 5.3550E-02 1.1127E-05 2.5762E-07 .288E-4 2.5939E-03 2.1710E-04 -2.5757E-06 .211E-4 2.5939E-03 2.1710E-04 -2.5757E-06 .211E-4 |
| 3 25 3 6 3 6 3 6 | 1.1336E-02 6.0015E-05 -4.2172E-07 .385E-5 0.0 5E 0.0 56 455 52 6 30.48 513 59 6.241E+18 2.4396E+00 1.2124E-02 -9.8852E-05 4 0.0 6.3561E-02 6.8125E-06 3.0254E-07 .288E-4 |
| 3 6 3 6 4 1 | 2.5363E-03 1.9567E-04 -2.1907E-06 .211E-4 1.1321E-02 5.1940E-05 -2.9976E-07 .385E-5 0.0 S2 0.0 |
| 4 3 4 4 4 5 | 1 6 2 5 6 5 4 3 6 4 6 2 7 4 6 2 |
| 4 6 4 7 4 8 5 1 1 | 4 6 5 6 6 2 3 3 2 3 1 1 0 0 1 20 3 |
| 5 1 1 5 1 1 5 1 1 5 1 1 | 6.0000E+00 8.0000E+01 1.0000E+01 1.2000E+01 1.4000E+01 1.5000E+01 1.8000E+01 2.0000E+01 2.4000E+01 2.8000E+01 3.2000E+01 3.6000E+01 4.0000E+01 4.5000E+01 5.0000E+01 3.2000E+01 3.6000E+01 4.0000E+01 4.5000E+01 3.2574E=01 |
| 5 1 1 5 1 1 5 1 1 5 1 1 5 1 1 | 2.3604E-01 2.3593E-01 2.3584E-01 2.3561E-01 2.3346E-01 2.3499E-01 2.3467E-01 2.3443E-01 2.3396E-01 2.3346E-01 2.3297E-01 2.3253E-01 2.3217E-01 2.3222E-01 2.3322E-01 2.3408E-01 2.3489E-01 2.3563E-01 2.3640E-01 2.3704E-01 |
| 521 521 521 521 | 2 1 0 0 1 20 3 0 1.5000E-01 1.0000E+00 2.0000E+00 4.0000E+00 5.0000E+00 8.0000E+00 1.0000E+01 1.2000E+01 1.4000E+01 1.6000E+01 1.8000E+01 2.0000E+01 2.4000E+01 2.8000E+01 |
| 5 2 1 5 2 1 5 2 1 5 2 1 5 2 1 | 3.2000E+01 3.6000E+01 4.0000E+01 4.5000E+01 5.0000E+01 8.9364E-03 8.9471E-03 8.9775E-03 9.0403E-03 9.1868E-03 9.3689E-03 9.5361E-03 9.6798E-03 9.8195E-03 9.9477E-03 1 1.0064E-02 1.0172E-02 1.0272E-02 1.0454E-02 1.0619E-02 |
| 5 2 1 5 3 1 5 3 1 | 1 1.0769E-02 1.090SE-02 1.1041E-02 1.1189E-02 1.1326E-02 1 3 1 0 0 1 20 3 1 0 1.5000E-01 1.0000E+00 2.0000E+00 4.0000E+00 1 6.0000E+00 8.0000E+00 1.0000E+01 1.2000E+01 1.4000E+01 |
| 53 53 53 53 | 1 1.6000E+01 1.8000E+01 2.0000E+01 2.4000E+01 2.8000E+01 1 3.2000E+01 3.6000E+01 4.0000E+01 4.5000E+01 5.0000E+01 1 1.7108E-02 1.7157E-02 1.7138E-02 1.7117E-02 1.7035E-02 1 1.6924E-02 1.6822E-02 1.6724E-02 1.6682E-02 1.6661E-02 |
| | |

ω 3.20 μ ω L ω ພາຍອອກອອ N N N N Ŋ 000000000 00000000 Ö 0 벽 벽 00 H (i) - (f) **NHHHHWHO** ญ **ω ⊷ ຫ** N. A ▲ 4288E 3776E 1861E LOH-HNOO 100000000000 - οuř 0.00 à oω -----A N NOOOOOOO 00 ö N N N N N N 21-02 20-162 $\mathbf{n} \mapsto \mathbf{n} \mathbf{n}$ 099999 B -3 NNOH លហ័ល ទ័ហ័ 000000''0 Ö 00-0 555555 88949 00001 8895600001 **t 3**35555 0 00 Ò 7700 7700 7700 7700 7700 7700 N044 0 ōο ω ώ H JON -44 ø 00 0 U A U O 00000 0000 $\omega \rightarrow \omega$ 3 (10P2) 9 (10P2) 9 (10P2) 9 (10P2) 8 (450E -1 (10P3) 8 (450E -1 (100P3) 8 (450E -1 (100P3) 9 (100P3) 1 (100P3 οò 00 00 0111 1 1.1 · ... (1) جو سو سو سو سو سو ແບບແບບບວວວ 5 1 3588-103 516-100 516-100 516-100 516-100 516-100 516-100 516-1 öF ō m. the second second UN, USS CE ٠. 101728 11 ខ្លួយ សូល ខ្លួយសូល ខ្លួយសូល 1.0000E 1.4502E 1.5255E 1.5255E 1.5255E 1.5255E 1.5255E 252 ġ 1.5394E-07 1.00 14 2.0440E-04 13 7.3374E-03 11 1.6074E-01 11 9.620E-01 12 2.7620E-02 12 2.7620E-02 1385E-07 (BORON) 140E+02 130E+01 0000E <u>ت</u>ر 371 m ġ 5 5 ***** 9 **....** õ ō m 1 0 8 A NI - NI 0000 ò òò شو منو عبو عنو • FO 0000----0 - 8 S) 5057E 2000E 4111 3531E-07 4 4 N - N - - A N - N -**អ្**អ្នក អ្ន 20000E 40000E 51675E 3755E 10 +++++ 33 6666 ż $\sim \sim$ 60 -- -- -- --• SHI M 7.3458E . ò ġ 1 4203E ABL) ت ت 4 - U 4 -467E-08 ÷ ٠ 40000E 40000E 46800E 5367E **** too 5 õ \frown ----2222222222 ò

| 041 | | 0 | and the second | 10 - C | | |
|---------------------------------------|---------------------------------|--------------|---|-----------|---|--|
| 042 | 9.4 | 10 | at in the state | 19 | | |
| 043 | | 0 | | | | |
| 044 | 13.1 | 00 | | · | · . | and the states |
| 0 4 5 | 2.6 | 50 | | | | |
| 046 | CE •(| 0 | 111 | | | |
| 047 | 24.0 | 40 | and the second second | | | 1997 - N. L. |
| 0 5 1 | 25.8 | 90 | 1. A | | | |
| 052 | 7.9 | 10 | | | | |
| 0 5 3 | 12.7 | 70 | | | | |
| 054 | 8.2 | 270 | | | | |
| 0 5 5 | 51.5 | 20 | | | · · · | |
| 55 | | 0 | · . · | | | |
| n c t | 24.9 | 140 | 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - | | · · · · | |
| 62 | 12.2 | 40 | | | | |
| 5 6 3 | 이 가 가 가 주말했다. 이 가 가 다 아이지 않는 | 0 | | | | |
|) 6 4 | 22.7 | 750 | | | | |
| 65 | | 0 | | | | an da ser an |
| 66 | | 0 | t ti | | | |
| 6 7 | 13 4 | 120 | | | | 1 |
| 7 3 | 2 (A) 15 • • | 1 | and the second | · · · · | 1 d. | |
| 7 3 | | . Õ | · | | 1 | |
| 7 4 | | 0 | * . | | | |
| 75 | 4 - S. S. A. | 0 | and a second second | 이 고향 것을 | | |
| 76 | 31.4 | 420 | | | an an an a | |
| 81 | 25.1 | 880 | 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - | | • | |
| 8 2 | 22.1 | 530 | | | | · · · . |
| 83 | 23. 24 | ((V 040 | | | e - 1 - 1 | and the second |
| 184 | - 0001 | 53 .008 | | | 1. A. | |
| | 0.59582 | -0.49396 - | -0.20215 | | | |
| 1 | 1.0 0. | 0 | | | 이 아이는 것을 가지? | |
| $\mathbf{i} \in \mathbf{\bar{1}}^{+}$ | 1 1 1 | 111 | 1 1 | | | |
| 2 | 1 1 1 | 1 1 1 | L I | | | |
| 3 | 1 1 1 | | L L • • | | | |
| 4 | | | 1 1 1 | | | |
| 2 | | 1 1 1 | 1 | | | |
| 7 | 1 1 1 | ī ī ī | | | | $(1,2,2,1) \in \mathbb{R}^{n}$ |
| 8 | īīī | 1 | | | · · · | |
| ī | 0.983 | | | | | |
| 2 | 1.115 | 1.296 | | | | 18 - 19 - 19 - 19 - 19 - 19 - 19 - 19 - |
| 3 | 1.032 | 1.175 1.08 | | | · · · | |
| 4 | 1.261 | 1.254 1.29 | J 1.14J | 081 | 1.5 | |
| 5 | 0.899 | 1.1/2 1.14 | 3 1.173 J 6 1.067 J | .241 1.02 | 5 | |
| 6 | 0.806 | 1 133 1 19 | 8 1,124 (| .921 0.41 | 6 | |
| | 0.297 | 0.461 0.48 | 6 0.364 | | | |
| . • | 15515 | 5 1 R60 | • • • • • • • • | | at a series | e de la composición de |
| a n | ົດ ດັຣຣິ (| 1 51 1 53 | 1 57 0 | | | a she ta p |
| 1 | 1 1 3 | 0 1 0 1 | S2 1 1 S2 | 0 1 1 | 56 U | n de la seconda de la second |
| 2 1 | | 00 | | | | and a second |
| 2 2 | 227000 | 00 22802080 | 33300000 | | | |
| 2 3 | 553000 | 00 23200000 | 23500000 | 22700000 | | general de la composición de l |
| 4 | 558050 | 80 53500000 | | 23200000 | 53500000 | · · · · |
| 5 | 558000 | 00 22200000 | 23202080 | 23200000 | 23202080 | 55805080 |
| : 5 | 228000 | 00 23202080 | 22802080 | 23202080 | 23202080 | 53500000 |
| | 222000 | 00 22800000 | 23200000 | 22800000 | • | |
| | 660000 | | | | | |
| JJ | 1. A | | | | | |

LAST

APPENDIX D: SAMPLE OUTPUT



OPUS RESULTS

TOTAL NUMBER OF INTEGRAL BP COMBINATIONS NUMBER OF DEPLETABLE BP COMBINATIONS NUMBER OF ACCEPTABLE BP COMBINATIONS

1152 39 2

فيمادينا الكلال

| CASE 37 | 3 GIVES L | ONGEST CY | CLE AT 13 | .240 GWD/ | MT |
|---------------|-----------------------|--|--|--|------------|
| FUEL LOA | DING: | · · · · | | | |
| | | i en | | | |
| | | .*** * E | ********* NRICHMENT | **** | |
| | | * B * B | P LOADING OC EXPOSU | * RE * | a da da es |
| ******** | ÷ | *** | ******** | **** | |
| * 2.90 | * | | | | |
| + 27.530 | ₩ ₩8734 | _ | | | |
| * 2.80 | * 3.60 | ₩ 4. ₩ 4. | | | |
| • 13.980 | * 0.000 | * | | | |
| * 2.70 | * 2.80 | * 2.80 | ₩ | | |
| * * 20.730 | • 15.200 | • • 14.670 | ₩ 5.5 ° | | |
| ********* | ******** | ******** | ************************************** | The second s | • ÷ |
| * 4 i | * • 17 950 - | 8 | + 26 100 A | | |
| ******** | ******** | | ******** | ******** | |
| * | 3.20 | - 3.2U | * 3.20 | • 2.70 | ₩ ₩ |
| * 10.400 / | * 13.060 (******* | * 14.320 · | * 0.000 ******** | * 24.600 | * |
| * 3.20 * | 3.20 | 2.80 | • 3.20 | 3,60 | 3.20 * |
| • 14.990 | • 10.110 | • 14.950 | 10.200 | 0.000 | 0.000 + |
| • 3.20 | 3.60 | 3.60 | 3.20 | 3.20 | 2.70 * |
| 14.990 | 0.000 | 0.000 | 0.000 | 21.570 | 25.060 * |
| 2.30 | • 3.20 | • 3.20 | • 2.70 · | ********** * | ***** |
| 27.200 | 22.080 | 25.290 | 23.160 | | |