AXIAL DEPENDENCE OF NUCLEAR FUEL MANAGEMENT

by

BRUCE ALAN NAPIER

B.S., Kansas State University, 1975

A MASTER'S THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Nuclear Engineering

Kansas State University

Manhattan, Kansas 66506

1977

Approved by:

John O. Mingle

Major Professor

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (CONT.)

LIST OF TABLES

1.0 INTRODUCTION

Nuclear reactor fuel management encompasses those aspects of fuel composition and loading, related to reactor physics, engineering, or economics, that are relevant to optimal fuel utilization within the design limits of the reactor core. Specifically, in-core management, as a subarea of the field, includes determination of the nuclear properties associated with new and partially irradiated fuel and fuel assemblies; specification of fuel loading, discharging patterns, and internal shuffling schemes; implimentation of control procedures during reactor operation which are related and parallel to the objectives specified for fuel use; optimization of fuel and control strategies; and sequencing of decisions for the purpose of fuel procurement.

The interacting requirements of neutronics, materials, control and safety, availability, and economics become quite complex; thus no one simple refueling scheme is possible for all reactors, or even reactor types. However, component parts of the total picture can be studied independently. An adequate representation of the fuel's nuclear properties is available in the current literature[2,3]. It is the areas of fuel shuffling, optimal control, and economical fuel procurement that possibilities of profitable fuel management planning arise.

The objective of fuel management planning is conservation resulting in a reduction in the cost of electrical power generation. Since the costs of the various components of nuclear power generation are interrelated and difficult to predict, a non-cost type of objective, usually directly related to cost, is generally chosen. Maximization of fuel burnup, minimization of

fresh fuel consumption, minimization of local power peaking for both safety and increased thermal efficiency, and maximization of refueling cycle length are all objectives that may be used as bases for fuel management schemes.

Glasstone and Sesonske[4] present a summary of early fuel management schemes. The earliest, and costliest, of these is batch loading. In batch loading the reactor is operated until it becomes subcritical and then the entire core is removed and replaced. Since the neutron flux profile and therefore burnup of fuel is uneven, the fuel near the extremes of the reactor is underused in this method. Improved burnup may be obtained through the use of a zoned core. The core is divided into radial zones of differing enrichment in an attempt to flatten the power density profile. Again, the entire core is replaced at once. Another method of improving burnup is the "in-out" shuffle. Rather than replace the entire core, a fixed fraction of the core is replaced at each refueling, with the new fuel elements being loaded in the center of the reactor and the retained portion of the old fuel being shuffled radially outward. Thus, the new fuel is placed in the position where it will experience maximum neutron irradiation and therefore maximum burnup. This scheme suffers the disadvantage that the maximum power generation is concentrated in the center of the core, causing heat removal problems and often necessitating lowered power levels. The "out-in" shuffling pattern also achieves higher average discharge burnups than the batch method and still avoids the reactivity concentration inherent in the in-out method. For the out-in method the new fuel is placed near the outer edge of the core and the older fuel is shuffled radially inward, from whence it is eventually removed. While the

out-in method is considerably better than the in-out or batch methods, this procedure can still lead to undesirable neutron flux shapes in the reactor because the reactivity difference between the new and old fuel elements in their respective zones can be fairly large. This problem can be reduced by placing elements of high burnup near those of lower burnup during shuffling. This procedure, called "scatter-loading," or "rondelay," is similar to the out-in method, but the inner regions are intermingled in a checkerboard pattern.

It can be seen that none of the above mentioned fuel management programs assures an optimal use of the fuel. Fuel management can be studied with a number of different techniques aimed at various parts of the optimality problem.

Wall and Fenech[5] use a modified dynamic programming algorithm to optimize the refueling of a single- enrichment, three-zone, 1000 MW(e) pressurized water reactor, minimizing fuel cost over the life of the reactor. However, only twenty-eight combinations of replacing and reshuffling the fuel are considered since dynamic programming becomes unwidely if too many variables are used. Wall and Fenech simulate the PWR core response with precalculated performance curves.

Because of the computational difficulties associated with dynamic programming several investigators present attempts to combine it with other techniques. Stover and Sesonske[6] revise the algorithm of Wall and Fenech, reducing the number of cases which must be considered by eliminiating core end states similar to those already tried and determined to be suboptimal. Stover and Sesonske apply this technique to a three-zone boiling water reactor simulated using previously fitted burnup polynomials. Fuel shuffling is again neglected because of the limitations of dynamic programming.

A third type of exhaustive search method is presented by Fagan and Sesonske[7] using a quasi-equilibrium cycle in which the operating conditions of the reference 1000 MW(e) PWR are nearly identical for each cycle. A fixed fraction of the core is replaced by fuel of constant feed enrichment at each refueling. Logical selection rules guide the placement of fuel elements in a trial-and-error approach. To insure optimality, all combinations must be tried, which leads to a large computational effort. The reactor is modeled with simplified polynomial burnup relations.

Accelerated direct search methods are also in use. Naft and Sesonske[8] use variational techniques to minimize the radial power peaking factor in the quasi-equilibrium cycle. A simplified semi-analytical equation is used to calculate the radial power distributions. Martinet and Santamarina[9] describe an improved version of Naft and Sesonske's model. Hoshino[10] presents an heuristic method using weighting factors applied to burnup, power peaking, and effective multiplication factor to determine rules for a one-dimensional four-zone reactor, both with and without fuel shuffling at refueling.

Because linear programming (LP) has advantages in saving computer time and storage over dynamic programming it is used by several investigators. Tabak[11] uses a simplified point reactor model with linear programming to determine the time periods between refueling which either maximized plutonium-239 production or minimized uranium-235 consumption. Fulford and Sesonske[12] use LP to optimize fuel usage for a system of nuclear plants on line together. A number of researchers direct efforts to minimizing fresh fuel consumption[13,14,15]. The model of Suzuki and Kiyose[13] selects new fuel and shuffles the retained fuel with it into a pattern which

minimizes fuel use. The model uses a reactor with five annular regions in which the fuel elements are identified by their exposure. The important point identified by Suzuki and Kiyose is that "the sequential nature of the refueling decision may not be as strong as generally regarded,"[10] that is, a solution generated by stagewise optimization of fuel shuffling compared to a solution globally optimized over all refueling stages compares very closely. This implies that the simpler, less costly, stagewise optimization procedure is very nearly the best available.

The fact that cycle-by-cycle optimization is acceptable is taken advantage of by many[17,18,19,20,21,22]. Brown[18] uses a stagewise linear programming approach to minimize the fresh fuel consumption of a PWR with three radial zones and a constant flux shape. Brown's model contains constraints on energy removal, neutronic properties, and reloading time restrictions. Chen[19] uses LP to generate a shuffling pattern for a core which minimizes the radial power peaking factor. Chen uses the LP constraints to minimize the deviation of the fuel loading pattern from an ideal distribution described by Winterburg[23]. Chen's program has the advantage that the individual fuel element identities are not lost.

Some investigators [17,22,24] combine the various approaches to fuel management in multi-step techniques, using the most appropriate method for the individual parts of the problem. Melice[24] breaks the problem into two parts, first finding the number and enrichment of the new fuel assemblies and then determining the loading pattern. The reloading pattern determination is based on an optimal core k-infinity profile. The optimal profile is approximated through a trial-and-error approach using interactive graphics. This method produces good results but requires a large amount of effort, and the use of interactive graphics may in many cases be inconvenient. Sauar[15]

uses a two-step procedure minimizing the fuel cost. The first step uses LP to determine the number of fuel elements assigned to a few region core and the second step uses the interactive graphics method of Melice to fine tune the fuel element distribution. Mingle[17] also uses a two-step program via perturbation theory. First the cycle cost is minimized in a procedure similar to that development by Brown[18], linear programming with a zoned core and discrete burnup groups. Then perturbation theory is used to examine the effects of interchanging individual fuel assemblies in a search to minimize either the power peaking factor or an overall power maldistribution figure-of-merit.

The most sophisticated multi-step method yet proposed is a six phase technique formulated by Motoda, Herczeg, and Sesonske[22]. In the first phase, an initial guess is made as to the optimal cycle length and control and refueling requirements for one cycle of a three-zone reactor. The guess is then iterated with a small LP program to optimize the result. For phase two, the phase one result is extrapolated to a five-zone core. Steps one and two could be combined with the added penalty of greatly increased computer time requirements. A larger, more accurate LP program is used in phase three to optimize the five-zone core. The fuel loading and shuffling requirements are first approximated in phase four using the minimum integrated k-deviation technique, and then optimized in phase five with a direct search scheme resembling that developed by Naft[8]. Finally, the burnup, actual cycle length, and power profile are calculated in phase six, allowing the procedure to be repeated for the next cycle.

All of the above mentioned fuel management schemes are based on obtaining the desired objectives (flat flux profile, maximum burnup, minimum cost) by either the one-dimensional (radial) or two-dimensional (X-Y) optimal

placement of fuel. A few researchers[25,26,27,28] examine the effects of control rod programming on fuel consumption. Motoda[25] analyses a two-region, single enrichment BWR for in-out, out-in, and parallel refueling schemes for the effects of control rod programming on maximization of fuel burnup. Notably, Motoda finds that maximization of burnup is equivalent to minimization of fuel enrichment, that the rod control programming is strongly dependent on the refueling scheme used, and that the burnup is influenced much more strongly by reshuffling than by rod programming.

Since present light water reactor fuel elements are constructed so that, at least initially, the fuel properties are constant along their length, most of the previously mentioned fuel management programs are limited to two dimensions at most. According to Silvennoinen[1], "practically no concern has yet been given to the optimization of the axial power and burnup distributions" in light water reactors. Work has been done axially for the Canadian CANDU reactors, but since the CANDU is a heavy water moderated reactor with partial length fuel elements that are continuously replaced in a bidirectional fashion, the CANDU results are not directly applicable to PWR's. Some simplified PWR burnup calculations are described by Benedict[29] to demonstrate a two-dimensional code, but the cases described are not truly realistic because they assume a core with continuous out-in refueling in a steady state. No fuel shuffling, axial fuel variation, or optimization is attempted. Although some of the work with control rod programming is directed toward the minimization of axial power peaking, little work exists on the effects of axial fuel variation on different reactor parameters. In this paper the behavior of the fuel characteristics and reactor power profile in the axial direction

of an example PWR is investigated. In addition, the effects of varying the fuel feed enrichment in the axial direction on burnup, power profile, and other important factors are examined.

2.0 THEORY

The study of the in-core section of the fuel cycle requires a familiarity with several branches of nuclear engineering analysis. The optimal use of fuel requires an optimizing technique, such as linear programming, and a method of quantifying the function to be optimized, such as the consumption or burnup of fuel. The burnup, in turn, depends on the local power history of the reactor, which must be either known from operational experience or calculated. A brief discussion of the methods of burnup determination, fuel shuffling optimization, and reactor simulation used in this work is given below.

2.1 Burnup Equations

Nonuniform consumption of fuel in an operating reactor requires the calculation of the burnup separately for each fuel assembly. The value of the atom density, N_i, of each nuclide considered will affect the neutron flux levels in the core. However, the flux levels in turn affect the values of the nuclei density set N_i. Fortunately, the flux varies slowly in a large power reactor and the flux may be considered constant over reasonably long periods of time. Thus the initial flux can be calculated from the initially known values of N_i, and then held constant for some time increment Δt. New values of N_i can be calculated at the end of the time period using the assumed constant flux, the flux levels can be adjusted to correspond to those which would result from the new atom density distribution, and the process repeated for another time increment.

The production and consumption of the various nuclides present during irradiation is modeled with a coupled set of differential equations of the form

$$\frac{dN_i}{dt} = \text{formation rate} - \text{destruction rate} - \text{decay rate} \qquad (1)$$

The equations describing the nuclides modeled in this work are simplified with the assumptions that the neutron flux for each lattice point is known and constant for each irradiation time step, that the radioactive decay of the fission products, except xenon and samarium, and the fuel is negligible, and that only one energy group is required for successful modeling. The derivation of the burnup equations can be found in references 2 and 3. For the example PWR which is modeled in this work the atom densities of the nuclides uranium-235, -236, -238, plutonium-239, -240, -241, and a composite fission-produce group are explicitly followed. The important fission products xenon-135 and samarium-151 are calculated independently of the composite group because of their special effects on reactor operation. For the model used, a time increment for which the fluxes are assumed constant on the order of two to four weeks is believed to be adequate.

2.2 Shuffling Patterns

Several parameters must be considered when new fuel is loaded into a core and the old shuffled: how much fuel, of what enrichment, and in what positions to obtain the best core performance. To reduce the number of variables in this investigation, the standard policy of one-third of the core being replaced at each reloading by fuel of constant feed enrichment is followed. The shuffling is done with the objective of minimizing the local power peaking factor by the method developed by Chen[19].

Chen's procedure is useful in designing a loading pattern for which the fuel distribution is as close as possible to an ideal distribution. The ideal distribution is based on a fuel distribution formula developed

by Winterburg[27] for a bare cylindrical reactor which results in a constant power density distribution during operation. The differences in fuel distribution between a bare reactor and a large water reflected reactor are small. The ideal fuel distribution has the three dimensional form

$$Z(r,z) = \frac{\Sigma_a(r,z)}{\Sigma_1(r,z)} = \frac{\pi k_{eff} R}{4} \left[\sum_{M=1}^{\infty} \sum_{N=1}^{\infty} \frac{1-(-1)^M}{M} \frac{1}{\lambda_N J_1(\lambda_N R)} \right.$$

$$\left. \frac{\eta - \prod_{i=1}^{n} [1 + \alpha_i^2 (\lambda_N^2 + \pi^2 M^2/h^2)]}{\prod_{i=0}^{n} [1 + \alpha_i^2 (\lambda_N^2 + \pi^2 M^2/h^2)]} J_o(\lambda_N r) \cos(\pi M z/h) \right]^{-1} \qquad (2)$$

where

Σ_a is the macroscropic absorption cross section of all fissionable material at radius r and height above center z,

Σ_1 is the macroscopic absorption cross section of all other non-fuel materials at r and z,

k_{eff} is the effective multiplication factor,

R is the effective radius of the core,

η is the average number of neutrons emitted per neutron absorbed in the fissile isotopes,

α_i are the group diffusion lengths,

λ_N is the N-th zero of the equation $J_o(\lambda_N R) = 0$, and

J_o, J_1 are Bessel functions of order zero and one, respectively.

The parameter Z(r,z) represents the quality of the fuel at location (r,z). It is an indication of the power producing ability of the fuel elements. Chen considers this fuel quality factor to be superior to enrichment, infinite multiplication factor, or burnup as a description of the state of the fuel.

The general shape of Equation 2 for a typical pressurized water reactor is given in Figure 1 as a function of radius and height. It is evident that more fissionable material is required near the periphery of the reactor to counterbalance the leakage of thermal neutrons from the core. The relative difference in shape of the radial versus axial curves is explained by the nature of the flux distribution of a uniformly enriched core, which follows a cosine shape axially and a J_o Bessel function radially. The ideal fuel distribution must counteract these tendencies caused by leakage from the reactor. Chen's program uses linear programming to determine the shuffling pattern which minimizes deviation of the fuel element distribution for the ideal radial pattern.

While the Winterburg formula represents an ideal to which to aspire, it is not a practical solution to reactor fueling. If the local power distribution would remain constant for the life of the core, a modified batch loading method based on the ideal distribution could be used. As it is, the effects of fuel burnup decrease the fuel absorption term and increase the non-fuel absorption term in $Z(r,z)$. Thus the ideal distribution cannot be maintained for even a short period. The best that can be hoped for is a good approximation for an appreciable time.

2.3 Computer Programs

The computer simulation of a reactor power history is a complex problem. The primary program used in this study is an original two-group, two-dimensional diffusion code called BURNRZ. It calculates the local power profile and burnup profile for a cross section of a pressurized water reactor in the radial-axial plane. A supplementary code, BURNXY, is used to model the example reactor in the X-Y cross sectional plane.

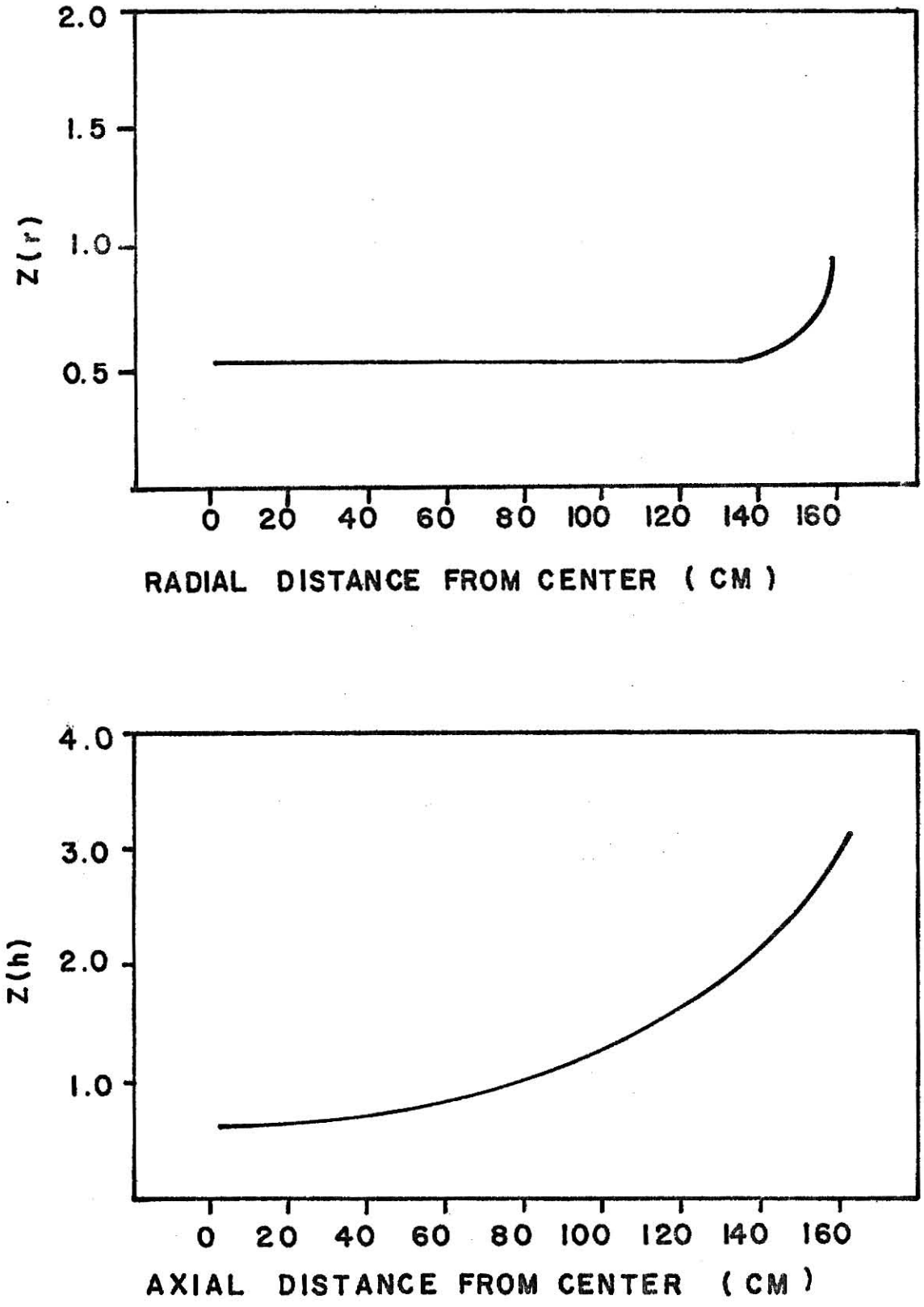


Figure 1. Radial and Axial Behavior of the Ideal Fuel Distribution for Constant Power Density in a Square Cylindrical Unreflected Reactor

The basic flow pattern for both BURNRZ and BURNXY is given in Figure 2. More detailed flow diagrams for the two codes are given in the appendices. The significant difference in the two programs lies in the different geometry of the two systems they solve. BURNXY solves for the power profile of a core with one-quarter symmetry in the X-Y grid where each fuel element is individually represented by a finite-difference mesh point. The quarter core is represented in Figure 3. The unique part of the program BURNRZ is Subroutine XY-TO-R, which converts the information input in the standard X-Y form into an equivalent series of concentric annular rings. The X-Y core pattern is averaged as depicted in Figure 3. The rings are separated by a distance of one fuel element width. Which fuel elements and fractions of fuel elements are present in each annular zone is first determined through an approximate procedure involving straight line segments to represent the arc of each zone boundary through the X-Y grid cells. The resultant concentric rings represent coaxial cylinders in three dimensions. Each of these cylinders is further broken into segments axially, so that the XY-TO-R process results in the configuration shown in Figure 4. The atom densities of the nuclides present in the fuel elements contained within any annular ring are averaged, with the resultant average being used to describe the ring as a whole. The procedure is repeated for each axial division of each cylinder. The net result is a radial-axial, R-Z, distribution of isotopes. Each element of this distribution is used as a mesh point in the flux, power, and burnup calculations.

The neutron flux calculations for BURNXY and BURNRZ are similar. Both are based on numerical solution of the two-energy-group, two-dimensional neutron diffusion equation set up for the appropriate geometry. The solution to the diffusion equation is obtained by use of the central-difference finite

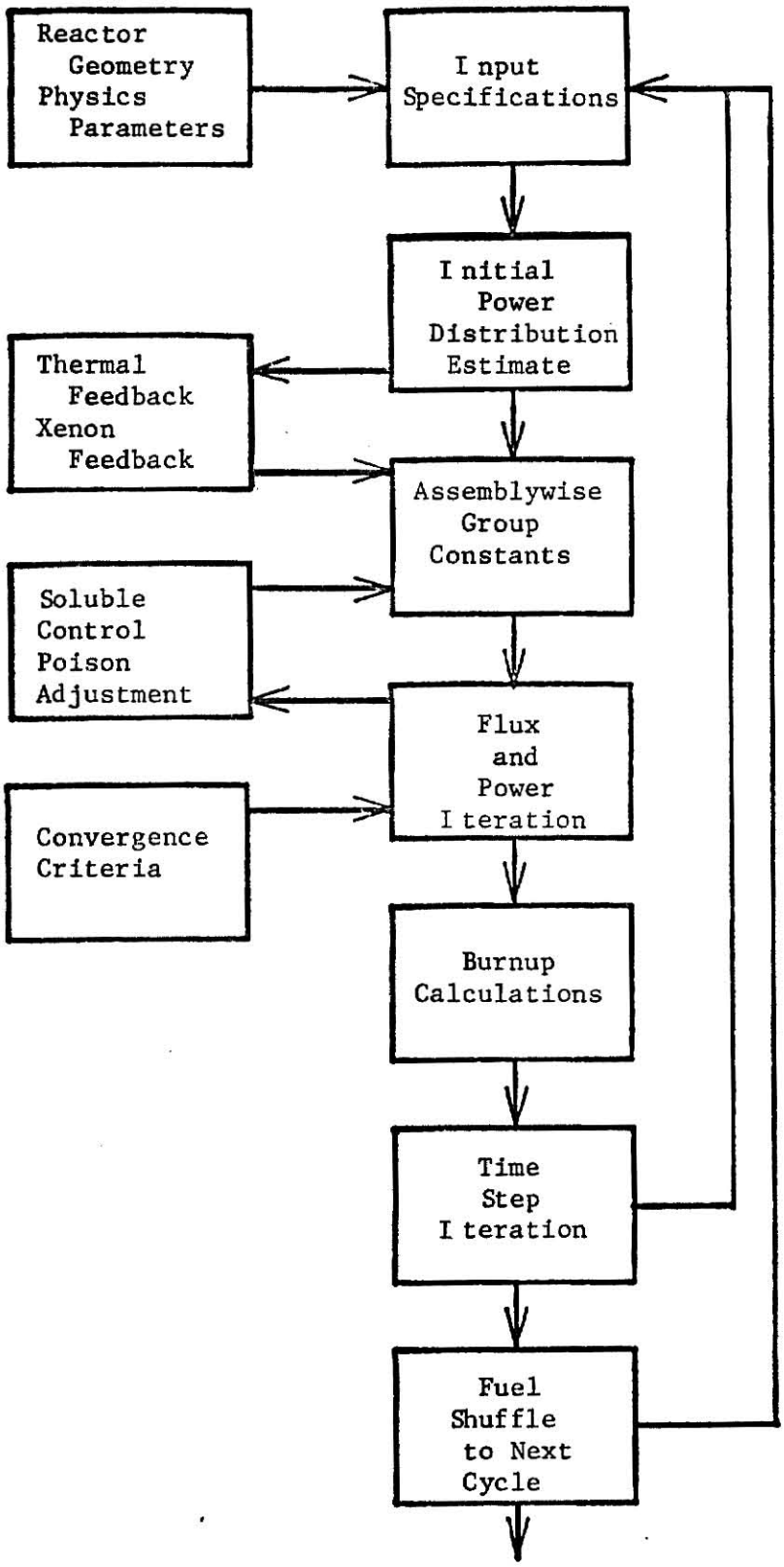


Figure 2. Basic Program Flow for Computer Simulation of Reactor Performance

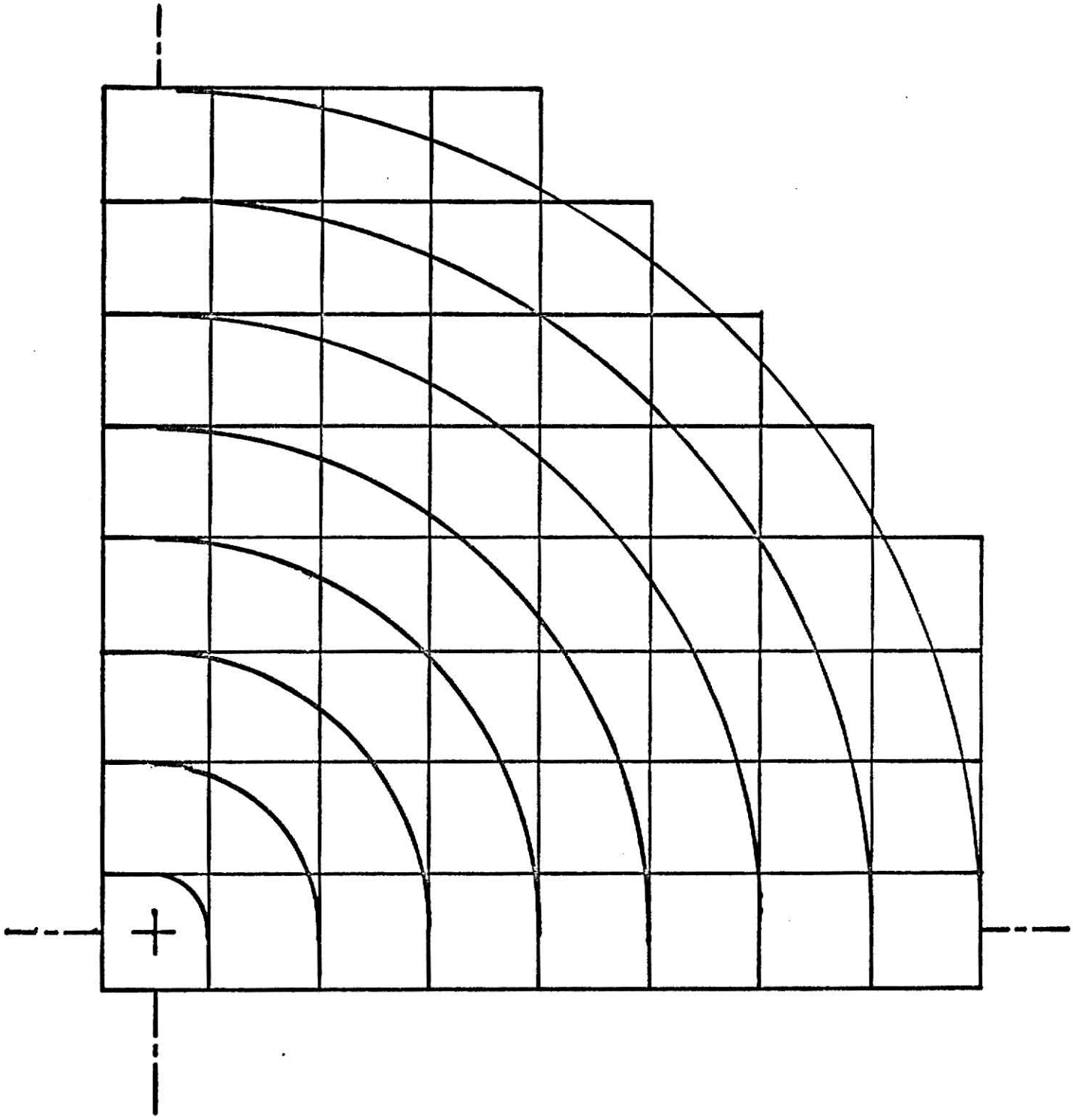


Figure 3. Illustration of Radial Averaging Procedure for Quarter-Symmetric Reactor Core Used in Program BURNRZ

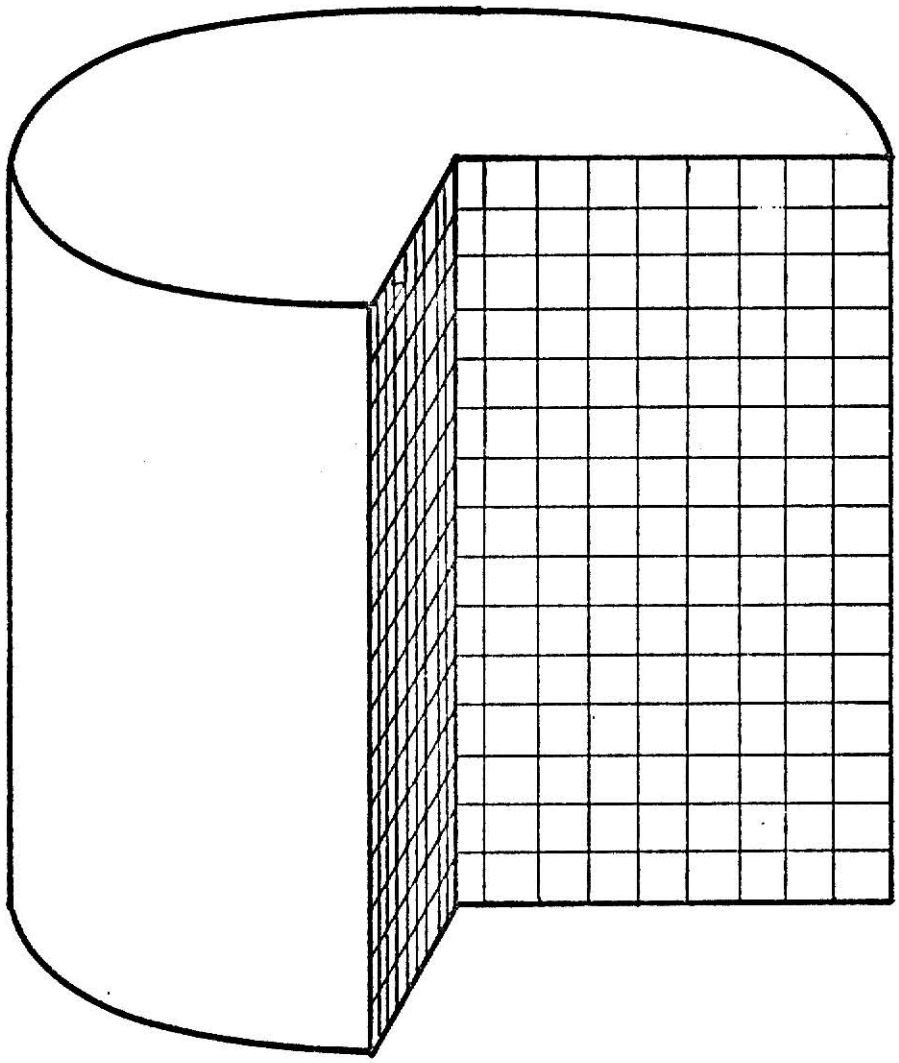


Figure 4. Illustration of Radial-Axial Averaged Fuel Distribution Used in Program BURNRZ

difference approximation applied to each mesh point. A point successive overrelaxation iterative technique is used to determine the solution of the resultant set of coupled linear equations. For the program BURNRZ, the properties of the mesh points are further complicated by the change in the temperature of the upward flowing cooling water. The changes in the moderating properties of the heated coolant/moderator are accounted for in Subroutine LIQHOH and the material properties of each mesh point are suitably corrected. This procedure is of course unnecessary for the code BURNXY, for which the axially averaged water properties are used. The soluble boron concentration required to maintain criticality may be found if desired, and the flux levels modified to compensate for it.

The known fluxes imply known power densities. A useful paramter is the local relative power, defined as

$$P_i = \text{Local Power Density/Core Average Power Density} \qquad (3)$$

The core maximum value of relative power is known as the core power peaking factor. While the power peaking factor is an important parameter in reactor operations, alone it is not particularly descriptive of core power distribution. Mingle[17] defines a core "Figure of Merit," FOM, as

$$\text{FOM} = \left\{ \frac{1}{N} \sum_{i=1}^{N} |P_i - 1| \right\}^{-1} \qquad (4)$$

The Figure of Merit is thus a measure of the deviation of core local power from average. Since the reciprocal of the average deviation is used, a large FOM indicates a flat core power distribution. From an ideal point of view, an infinite Figure of Merit would be desirable.

With the known neutron fluxes the nuclide concentrations for each mesh point are calculated as described in Section 2.1 for a time increment

for which the flux is assumed constant. To save computer time, between major flux calculations the flux is normalized once to compensate for burnup and maintain a constant power output, and atom densities are updated. Every second flux update, then, the new atom densities are used to find new properties for each mesh point. The flux determination and burnup calculations may be repeated as many times as required.

If the BURNRZ simulation is for a cycle other than that corresponding to cycle 1, new fuel, fuel may be shuffled before other calculations are performed. This is done in a procedure that is essentially the reverse of the X-Y to R-Z transformation. The R-Z distribution is returned to an X-Y distribution, the old fuel is subtracted, new added, and the result shuffled in the X-Y mode, and the core is reaveraged to the R-Z mode once again. This is done since all present PWR fuel elements are manufactured as full-core-length units and partial shuffle is physically impossible. Since the individual fuel assembly identity is lost in the initial R-Z averaging the return to X-Y geometry is not exact, and the accuracy of the fuel shuffling is thereby limited. However, it is believed that it is of sufficient accuracy to provide a strong indication of the reactor trends. As a check on the accuracy of the invert-shuffle-convert, the process uses as a comparison the X-Y atom densities of the companion code BURNXY.

The refueling shuffling patterns are derived by the linear programming code LPCHEN, developed by Chen[19] to approximate the ideal fuel distribution described in section 2.4. The code performs the calculations on a model of the core broken into radial and azimuthal zones. There are constraints on radial uniformity, azimuthal uniformity, and local uniformity, as well as a normalization constraint. The input to LPCHEN is from another

of Chen's codes, ZVALUE, which functions identically to BURNXY but which also provides the values of Z(r,z) used in the LP optimization.

Thus four codes are used in this work. The main one is BURNRZ with its companion BURNXY. The shuffling is done with the combination of the somewhat redundant ZVALUE and the LP program LPCHEN.

3.0 THE REFERENCE REACTOR

The example reactor used in this analysis is a typical pressurized water reactor (PWR). It is based on the Westinghouse PWR to be used at Wolf Creek Generating Station near Burlington, Kansas. Pertinent data are taken from the Westinghouse RESAR-3[30].

The reactor is designed to produce 3411 MW thermal and to operate at a nominal pressure of 2250 psia and temperature averaging 590 degrees Fahrenheit. It contains 193 fuel assemblies each composed of a 17 x 17 array of fuel pins. The core is twelve feet high and approximately eleven feet in diameter. The standard cycle one core contains 111 tons of uranium dioxide fuel in three enrichments, 3.1%, 2.6%, and 2.1%. The initial enrichment distribution is as shown in Figure 5. Some other pertinent information is given in Table 1.

Table 1. Reference Reactor Parameters

Average Operating Temperatures (Degrees Fahrenheit)	
Inlet	557.3
Outlet	618.9
Overall average	590.8
Coolant Flow Rate (Pounds/hr)	1.358×10^{9}
Physics Parameters	
Fast Fission Factor	1.0055
Fission to Resonance Non-leakage Probability	0.995
Resonance Escape Probability	0.7088
ν-235	2.07
ν-239	1.87
Fermi Age (cm^2)	36.0
Diffusion Length (cm^2)	21.4
Total Buckling (cm^{-2})	2.77×10^{-4}
Mesh Size (cm)	21.4
Average Plant Capacity Factor	0.75

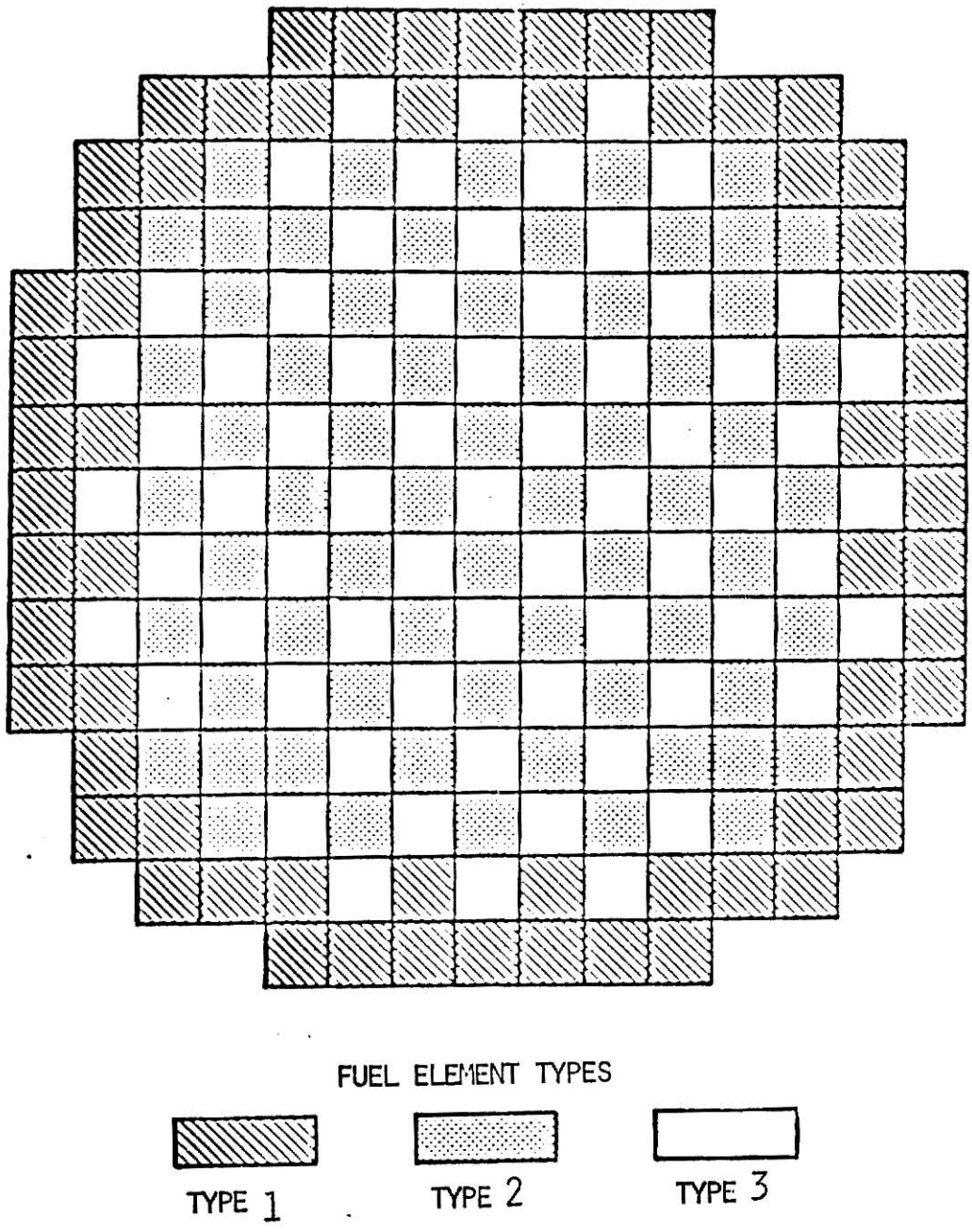


Figure 5. Initial Fuel Element Distribution for Example PWR

4.0 RESULTS

Several cases are investigated. A reference standard: a series of three consecutive burnup cycles of length one year, ten months, and one year, respectively, is calculated for a reference reactor core using three types of fuel elements of axially constant fuel enrichment. This reference case closely follows procedures presently used for PWR's. Three experimental cases, the fuel elements of which have axial enrichment variations, are then presented. Each experimental case is a different attempt to approximate the Winterburg ideal fuel distribution described in Section 2.2. The fuel assembly axial enrichment distributions are summarized in Table 2. The loading patterns for the first cycle of each case examined are the same as shown previously in Figure 5. The new fuel elements inserted in the cores at the beginning of cycles 2 and 3 are always of the type 1 element shown in Table 2 for the particular case. So that fuel burnup among the four separate cases investigated may best be compared, all are calculated for the same burnup period lengths as the reference case.

The code BURNRZ has provisions for the calculation of the equilibrium xenon concentration. The xenon concentration is considered to be directly proportional to the neutron flux levels. Unfortunately, this procedure for determination of xenon concentration is only approximate. The procedure leads to large calculated flux oscillations from top to bottom in the core with time. It is believed that the xenon calculation procedure does not adequately model the xenon oscillations. Therefore, the equilibrium xenon is not calculated in this work. Since the xenon concentration is proportional to the power level, it tends to depress the local power in areas of high reactivity below where it would normally be. The critical concentration of soluble control boron is also not calculated in this work

TABLE 2

Axial Fuel Enrichment Distribution in Fuel Assemblies, Each Number Represents the U-235 Fraction in a 9.6 inch Segment of a Fuel Assembly.

Reference			Case 1		
Type 1	Type 2	Type 3	Type 1	Type 2	Type 3
0.031	0.026	0.021	0.031	0.031	0.031
0.031	0.026	0.021	0.031	0.026	0.021
0.031	0.026	0.021	0.031	0.026	0.021
0.031	0.026	0.021	0.031	0.026	0.021
0.031	0.026	0.021	0.031	0.026	0.021
0.031	0.026	0.021	0.026	0.026	0.021
0.031	0.026	0.021	0.026	0.026	0.021
0.031	0.026	0.021	0.026	0.026	0.021
0.031	0.026	0.021	0.026	0.026	0.021
0.031	0.026	0.021	0.026	0.026	0.021
0.031	0.026	0.021	0.031	0.026	0.021
0.031	0.026	0.021	0.031	0.026	0.021
0.031	0.026	0.021	0.031	0.026	0.021
0.031	0.026	0.021	0.031	0.026	0.021
0.031	0.026	0.021	0.031	0.031	0.031

Case 2			Case 3		
Type 1	Type 2	Type 3	Type 1	Type 2	Type 3
0.032	0.032	0.032	0.032	0.032	0.032
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.026	0.021	0.031	0.026	0.021
0.032	0.032	0.032	0.032	0.032	0.032

since it requires excessive computational effort. The boron tends to increase the power peaking factor, and the xenon to decrease it, so by ignoring both factors it is believed that the ultimate result is not greatly changed. Also, none of the X-Y programs has a xenon correction, so direct comparison is facilitated.

Figures 6, 7, 8, 9, 10, and 11 are example X-Y and R-Z power profiles put in a form for ready comprehension. Figures 12 and 13 are X-Y power profile data. Tables 3, 4, and 5 are X-Y discharge burnup data. Figures 14 and 15 are R-Z power profile data, and Figures 16, 17, 18, 19, 20, 21, and 22 are R-Z burnup data.

4.1 Discussion of Reference Case

The reference case models the performance of pressurized water reactors as they are presently constructed. Three types of fuel elements of constant axial enrichment of 3.1%, 2.6%, and 2.1% are used. For the initial burnup cycle, the power density of the core is highest in the central part of the core at Beginning-of-Life (BOL) and gradually shifts outward over time both radially and axially, until at the core End-of-Life (EOL) the preponderance of the power output comes from the upper and lower outer edges of the core. The power density shift follows from a thermal neutron flux density shift as the inner regions of the core burn. This basic pattern is followed from BOL to EOL of every case studied, but is most pronounced in the reference case. To illustrate the nature of the power production shift, Figure 6 contains example data on BOL and EOL local relative power for the symmetric quarter core in the X-Y plane as it is provided by the program BURNXY. To aid comprehension of these data, Figure 7 presents a graphical representation of the BOL numbers for the

0.922 1.018	0.931 1.037	0.782 0.953	0.587 0.707				
0.958 0.931	1.300 1.356	0.874 0.930	1.082 1.295	0.854 1.031	0.588 0.721		
1.157 1.101	0.993 0.927	1.099 1.120	0.908 0.949	1.017 1.154	1.037 1.278	0.589 0.721	
0.995 0.888	1.160 1.077	0.966 0.903	1.101 1.112	0.909 0.942	1.019 1.154	0.857 1.031	
1.173 1.053	0.998 0.876	1.154 1.067	0.967 0.899	1.104 1.112	0.912 0.950	1.089 1.296	0.583 0.707
1.009 0.865	1.179 1.046	0.999 0.873	1.157 1.067	0.971 0.903	1.107 1.120	0.881 0.930	0.789 0.953
1.188 1.039	1.013 0.862	1.182 1.046	1.003 0.876	1.169 1.077	1.002 0.927	1.312 1.365	0.941 1.038
1.016 0.860	1.190 1.039	1.013 0.865	1.181 1.053	1.003 0.888	1.169 1.101	0.967 0.931	0.931 1.018

0.922	BOL Local Power Density
1.018	EOL Local Power Density

Figure 6. Example Cycle 1 Local Relative Power Density Data from Program BURNXY

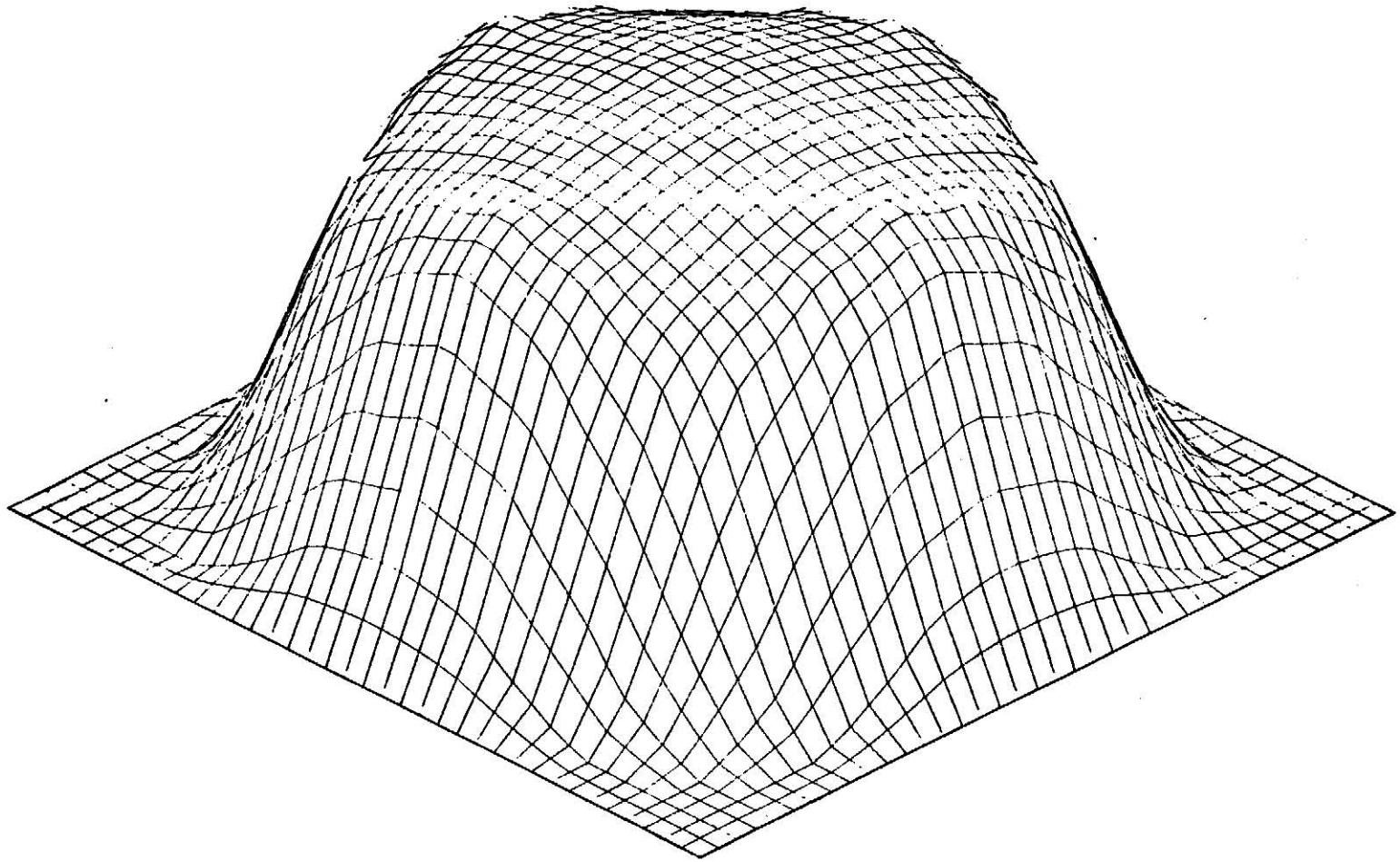


Figure 7. Example Cycle 1 BOL X-Y Local Relative Power Density Profile Map. The Distance Above the Base Plane is Proportional to Power Density.

entire core. The height above the base plane is proportional to the value of relative power at that position. Figure 8 is a similar representation for the EOL case. The shift in power density from the center to the edges of the core is evident. Radially, the cycle 1 reference case power peaking factor, the maximum value of relative power, declines with increased burnup and the position of maximum local power density moves from near the core center line toward the outer edge. The axial peaking factor behaves in a similar manner, but the area of maximum power generation splits into two parts which migrate slowly to the reactor extremes. Figure 9 is similar to Figure 6 but contains reference data for one-half of the reactor R-Z cross sectional plane. Figure 10 is a graphical representation of the BOL local power values. The curves enclose areas in which the local power is equal to or greater than that indicated on the curve. Figure 11 represents the R-Z plane at cycle 1 EOL. The decline in the maximum value of power peaking due to the diffusion of power production to the edges of the core results in a higher reactor Figure of Merit of EOL than at BOL. The total core burnup at EOL is fairly even, concentrated somewhat in the inner two-thirds of the core. This is to be expected, since the neutron leakage is negligible in the inner core regions. It is also desirable since the lower enrichment fuel scattered in the inner regions is generally that which is first discharged at refueling.

The second cycle of the reference case is shorter than the first, since the burnup of the first cycle is so even. This causes the remaining shuffled fuel to be less neutronically potent and the cycle 2 BOL core effective multiplication factor to be lower. The local power peaking factor at BOL is less than that for the cycle 1 BOL, and actually goes lower than at any time in cycle 1 about halfway through the second cycle,

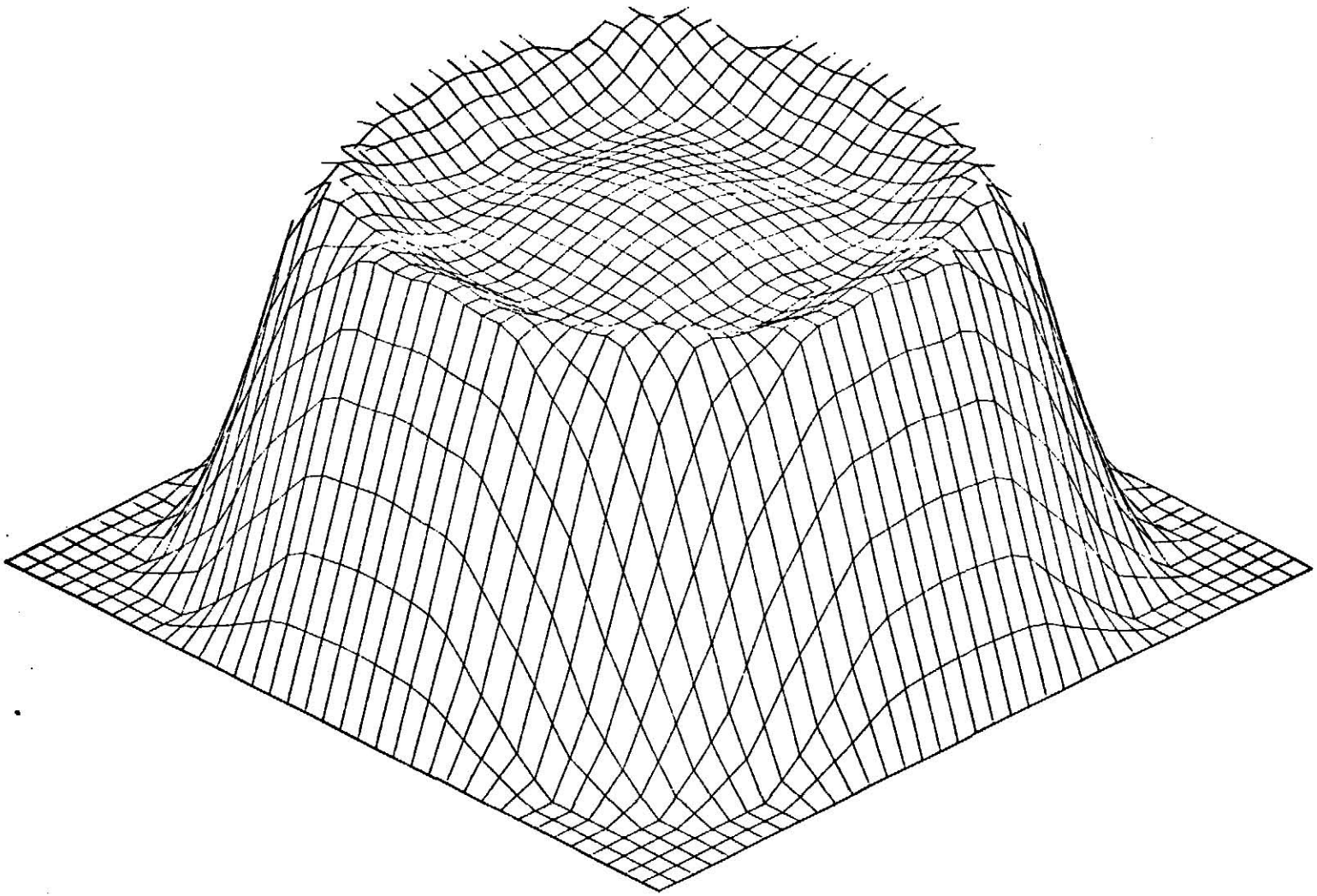


Figure 8. Example Cycle 1 EOL X-Y Local Power Density Profile Map. The Distance Above the Base Plane is Proportional to Power Density.

0.419 0.664	0.472 0.774	0.441 0.738	0.435 0.774	0.407 0.778	0.386 0.806	0.338 0.767	0.270 0.648	0.087 0.175
0.764 0.988	0.863 1.164	0.804 1.103	0.795 1.165	0.744 1.177	0.706 1.240	0.619 1.201	0.496 1.028	0.160 0.254
1.071 1.016	1.213 1.203	1.129 1.137	1.118 1.206	1.048 1.226	0.997 1.313	0.875 1.299	0.702 1.139	0.224 0.267
1.323 0.938	1.507 1.114	1.400 1.051	1.389 1.119	1.303 1.143	1.244 1.239	1.094 1.246	0.880 1.115	0.287 0.254
1.511 0.862	1.732 1.026	1.606 0.967	1.597 1.031	1.500 1.056	1.438 1.153	1.267 1.172	1.022 1.062	0.319 0.238
1.627 0.816	1.879 0.973	1.738 0.915	1.733 0.975	1.630 0.999	1.570 1.095	1.388 1.120	1.123 1.022	0.346 0.226
1.671 0.795	1.945 0.950	1.794 0.891	1.794 0.949	1.691 0.971	1.637 1.065	1.453 1.092	1.180 1.002	0.358 0.219
1.646 0.792	1.930 0.974	1.775 0.886	1.782 0.942	1.683 0.962	1.638 1.055	1.459 1.083	1.190 0.998	0.355 0.216
1.558 0.801	1.841 0.958	1.690 0.895	1.701 0.949	1.610 0.965	1.576 1.058	1.410 1.088	1.155 1.006	0.340 0.215
1.419 0.821	1.688 0.983	1.545 0.915	1.561 0.969	1.481 0.983	1.457 1.076	1.309 1.107	1.078 1.028	0.312 0.217
1.237 0.861	1.481 1.032	1.354 0.958	1.372 1.012	1.304 1.023	1.290 1.117	1.164 1.149	0.938 1.071	0.275 0.223
1.026 0.926	1.234 1.112	1.127 1.029	1.144 1.086	1.090 1.094	1.083 1.190	0.982 1.221	0.815 1.138	0.230 0.234
0.792 1.001	0.958 1.207	0.873 1.115	0.889 1.175	0.848 1.180	0.847 1.278	0.770 1.302	0.642 1.204	0.180 0.246
0.546 0.977	0.662 1.211	0.603 1.117	0.615 1.178	0.588 1.180	0.588 1.273	0.536 1.280	0.448 1.161	0.125 0.241
0.293 0.705	0.356 0.860	0.324 0.794	0.311 0.838	0.316 0.839	0.317 0.901	0.290 0.894	0.242 0.794	0.067 0.175

BOL Local Power Density
EOL Local Power Density

Figure 9. Example Cycle 1 Local Relative Power Density Data from Program BURNRZ

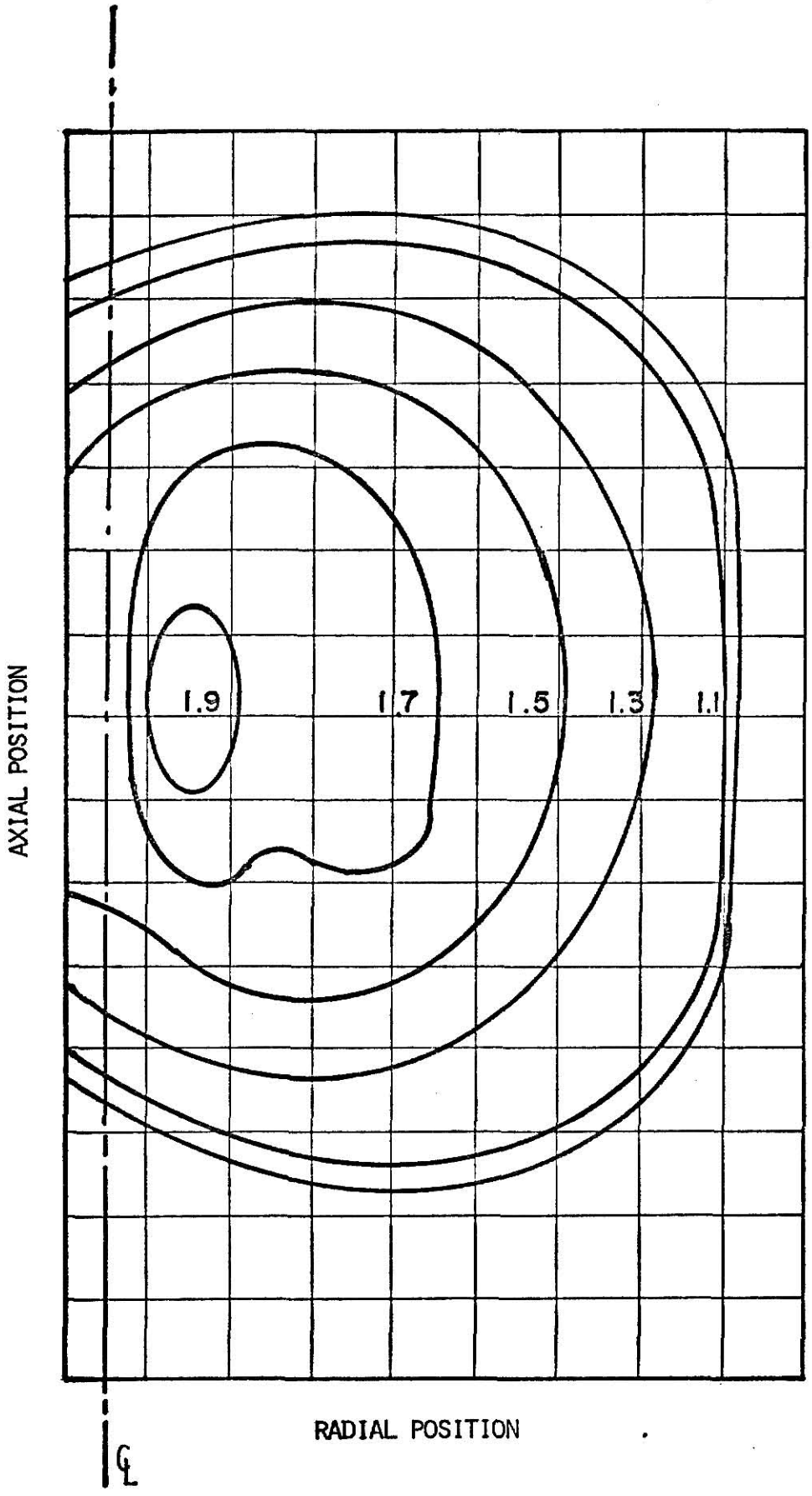


Figure 10. Example Cycle 1 BOL R-Z Local Relative Power Density Profile Map. The Curves Enclose Volumes in which the Relative Power is Greater than Indicated on the Curve.

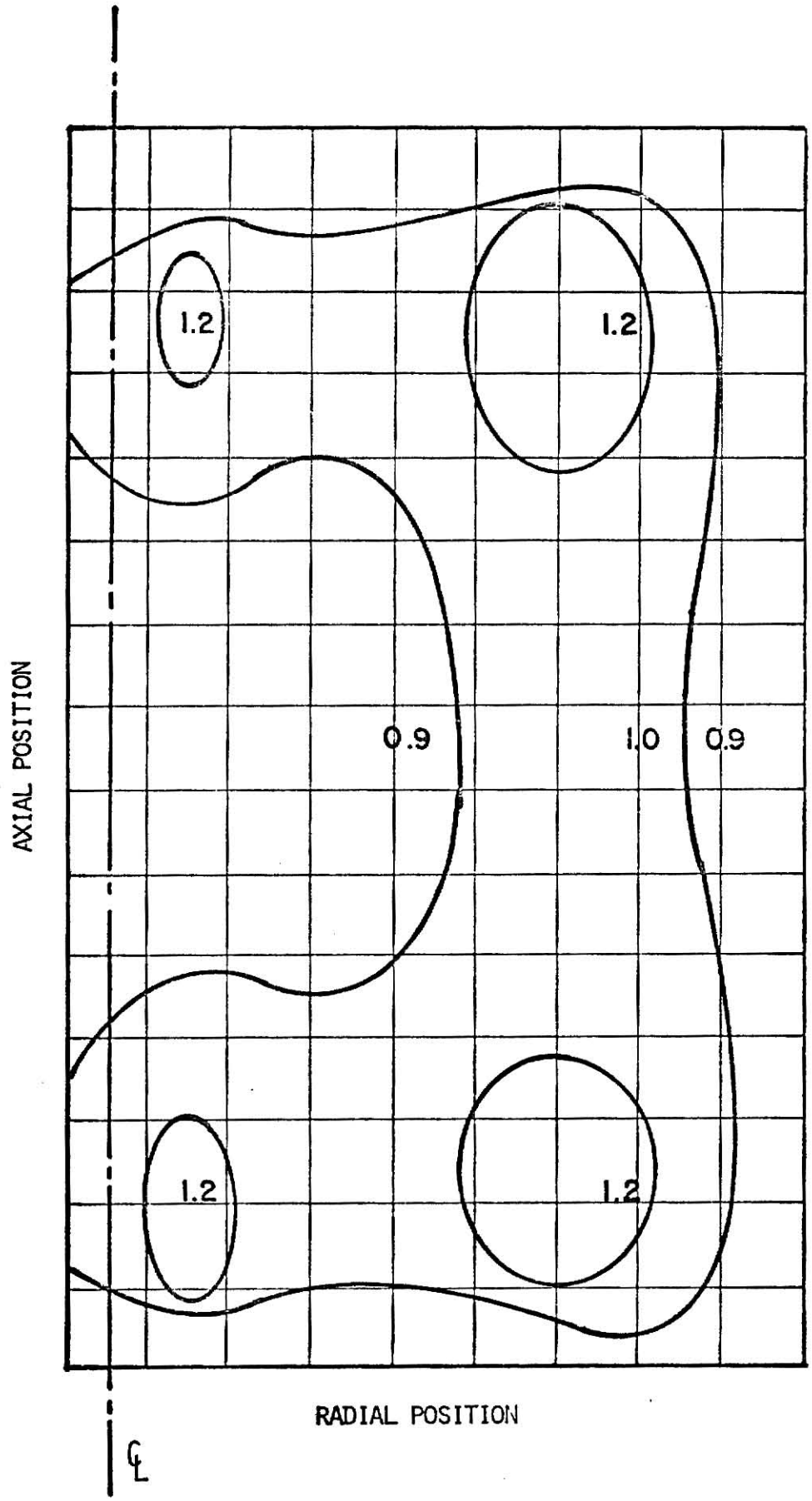


Figure 11. Example Cycle 1 EOL R-Z Local Relative Power Density Profile Map. The Curves Enclose Volumes in which the Relative Power is Greater than Indicated on Curve.

but it then increases again near EOL as the flux maxima move to the reactor core extremes seeking to escape the highly burned up central regions. Radially the power profile begins and ends much higher at the outer edge of the core. This implies that much of the cycle two power production is occurring in the new fuel.

Figures 12 and 14 are depictions of the power peaking factor behavior with time for the X-Y and R-Z conditions, respectively. Figures 13 and 15 display the actions of the core Figures of Merit respectively for X-Y and R-Z cases. On all graphs the open circle represents the data for the reference case.

The third cycle of the reference case lasts as long as does the first. The last of the low enrichment fuel elements are shuffled out between cycles two and three and the whole core is made up of fuel that, at least originally, is of 3.1% enrichment. This fact combines with the short life of cycle two, thus inflicting less burnup on the elements retained from that core, implying that the cycle three core should last longer than that of the second cycle. As a result of this higher reactivity, the BOL flux and power profiles are rather highly peaked towards the center of the core. The power density quickly becomes more uniform as the center burns out and the flux maxima glide out toward the reactor edges. Finally, as this process proceeds, the maxima begin again to increase slightly as the power generation concentrates at the outer edges.

In all three cycles of the reference case the core burnup is quite uneven axially. In cycle one, the center parts of some fuel elements suffer as much as 143% more burnup than the extreme end sections. All the elements are at least twice as burned near the center as near the edges. Cycles two and three are not quite as severe, but still average 80% more and 60% more, respectively, in the center than the edges.

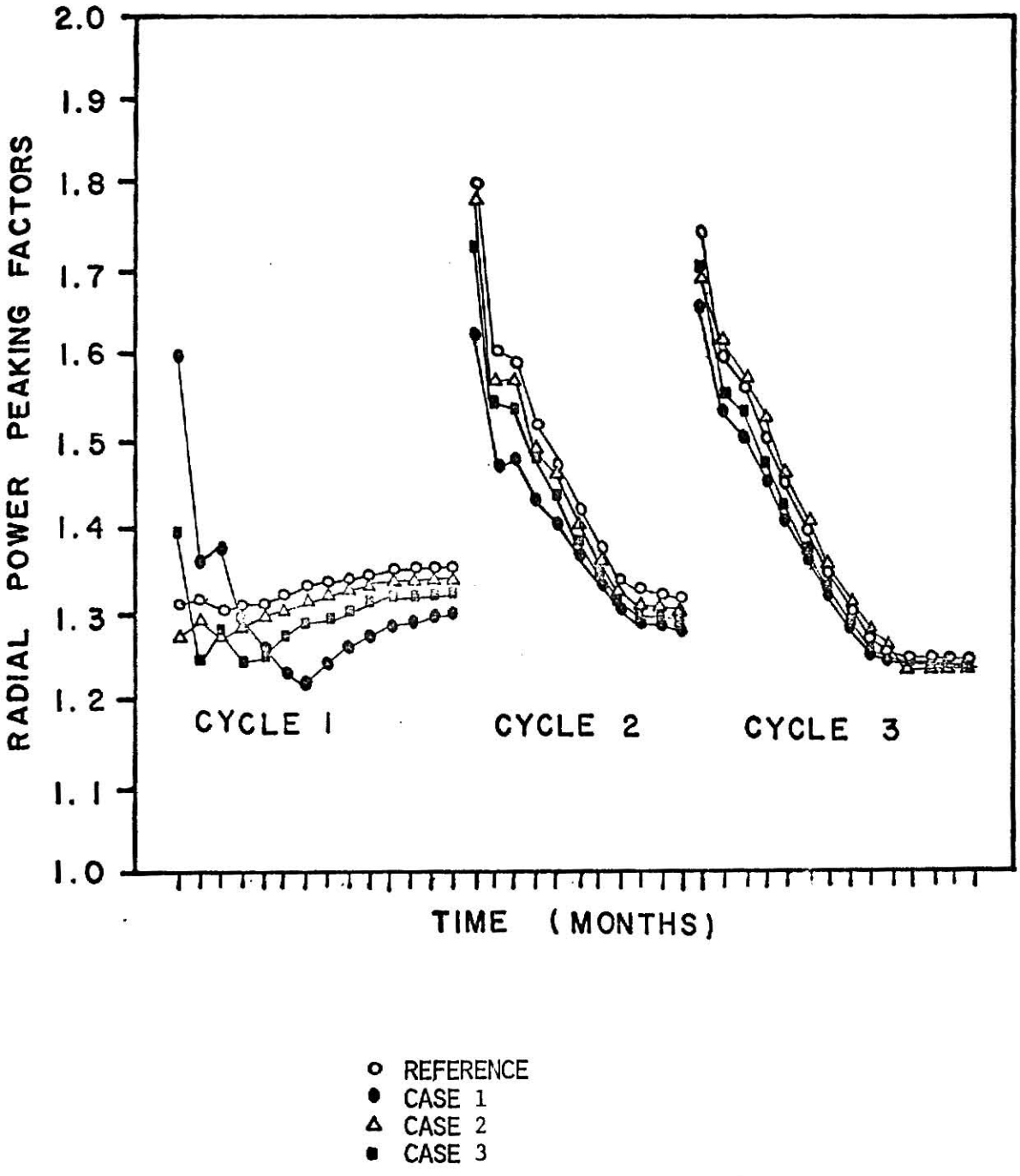


Figure 12. Behavior of Radial Power Peaking Factors with Increased Burnup

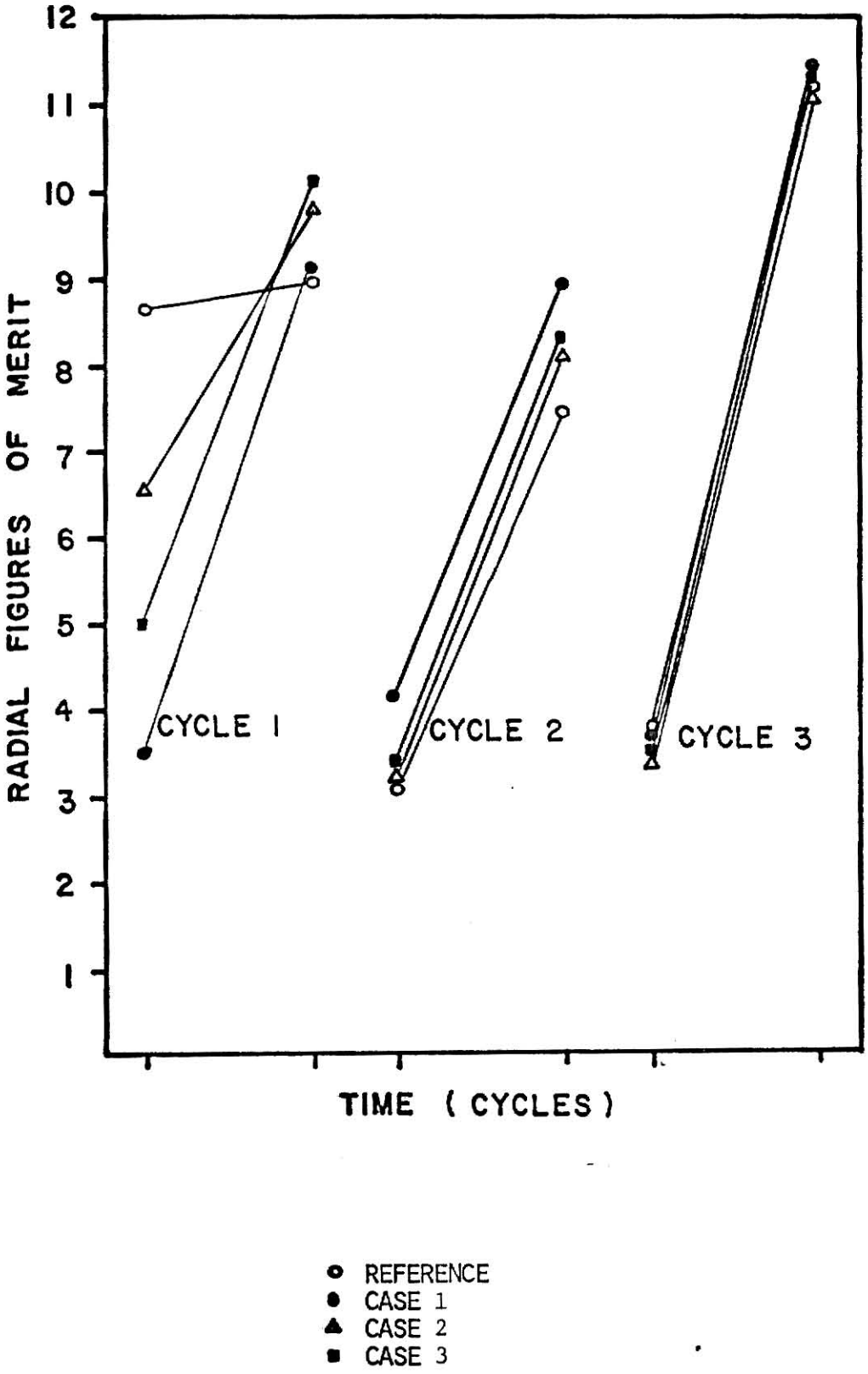


Figure 13. Behavior of Radial Figures of Merit with Increased Burnup

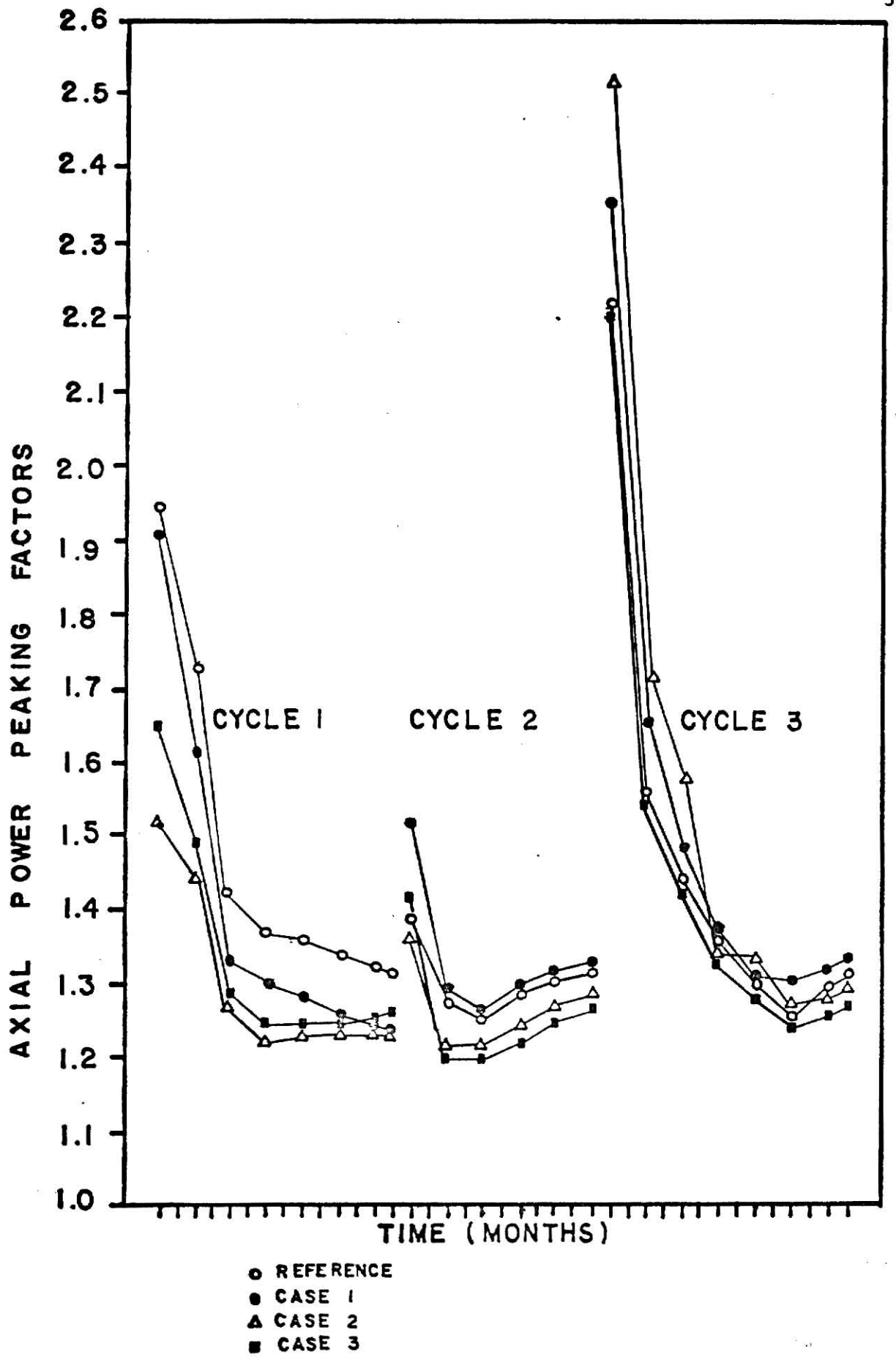


Figure 14. Behavior of Axial Power Peaking Factors with Increased Burnup

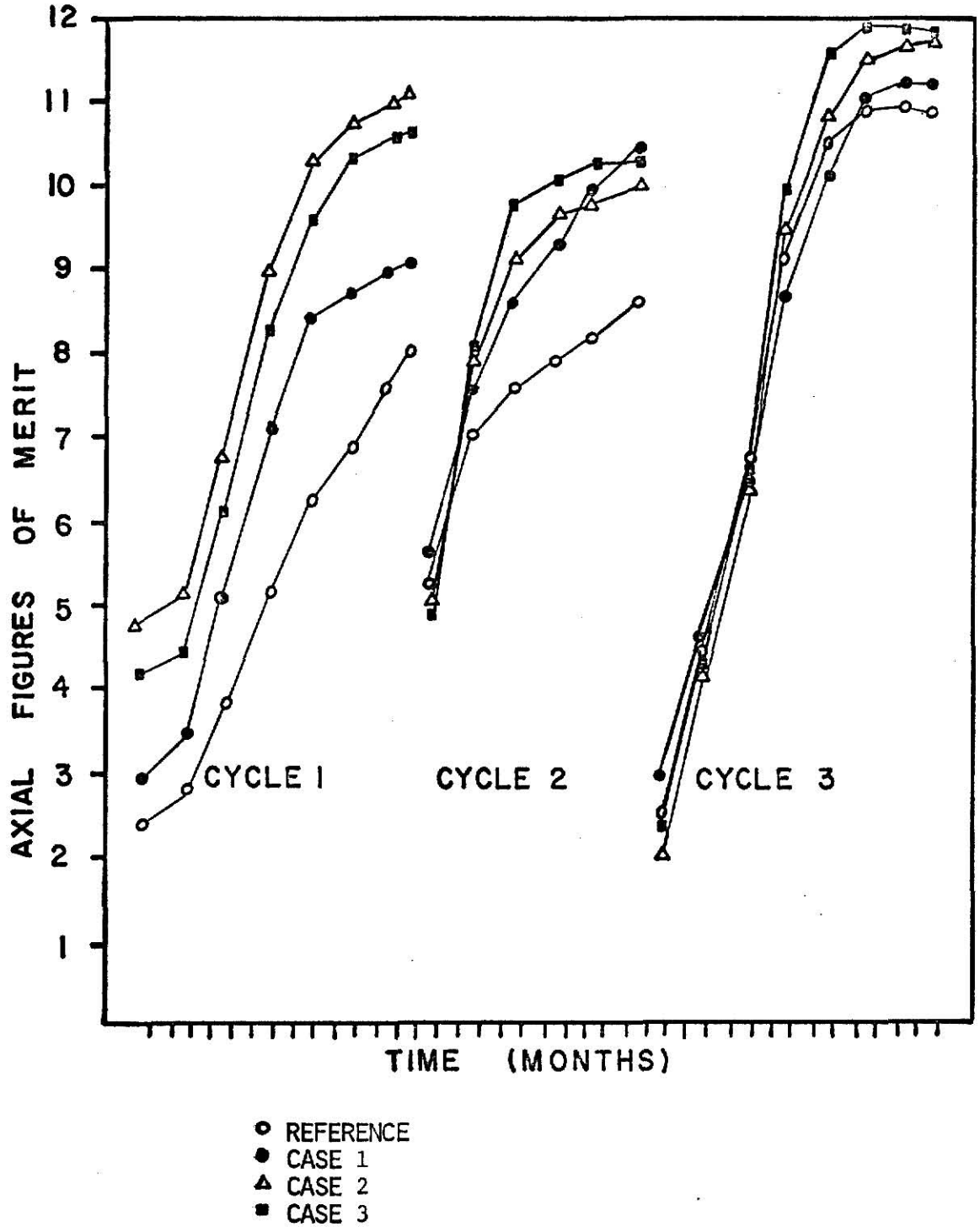


Figure 15. Behavior of Axial Figures of Merit with Increased Burnup

Figures 16, 20, 21 and 22 illustrate this problem. Figure 16 shows the column average burnup for all three cycles of the reference case. Figures 20, 21, and 22 show the percent deviation from the column average, that is, the standard deviation of the fifteen mesh points along the fuel element from the mean value divided by the mean. The higher the coefficient of variation expressed as a percent, the greater the unevenness of the burnup. An ideal case, perfectly even burnup, would have zero deviation. This display of the underutilization of the fuel in the reactor shows the need for some means of evening the fuel burnup throughout the core.

4.2 Case 1 Discussion

The fuel elements used in Case 1 are designed to lower both the axial and radial power peaking factors and increase the burnup of the fuel at the reactor top and bottom edges by more closely approximating the ideal Winterburg fuel distribution. The fuel in the top and bottom segments of all fuel elements in the core is replaced by fuel of 3.1% enrichment, and that in the elements of the outer ring lowered to 2.6% in the middle third of each element. Element compositions are given in Table 2.

The radial behavior of all cycles of Case 1 is very similar to that of the reference case cycles, except at the very start of cycle 1, when the lowered enrichment of the Type 1 fuel elements causes a greater central peaking tendency. The similarity in X-Y power peaking factors can be seen in Figure 12 and the similarity in radial Figures of Merit in Figure 13. That the variations in axial enrichment do not greatly effect the radial power distribution can also be seen in Tables 3, 4, and 5 which given the X-Y discharge burnup for each cycle of each case. The discharged fuel for each case is very near the burnup of the reference case.

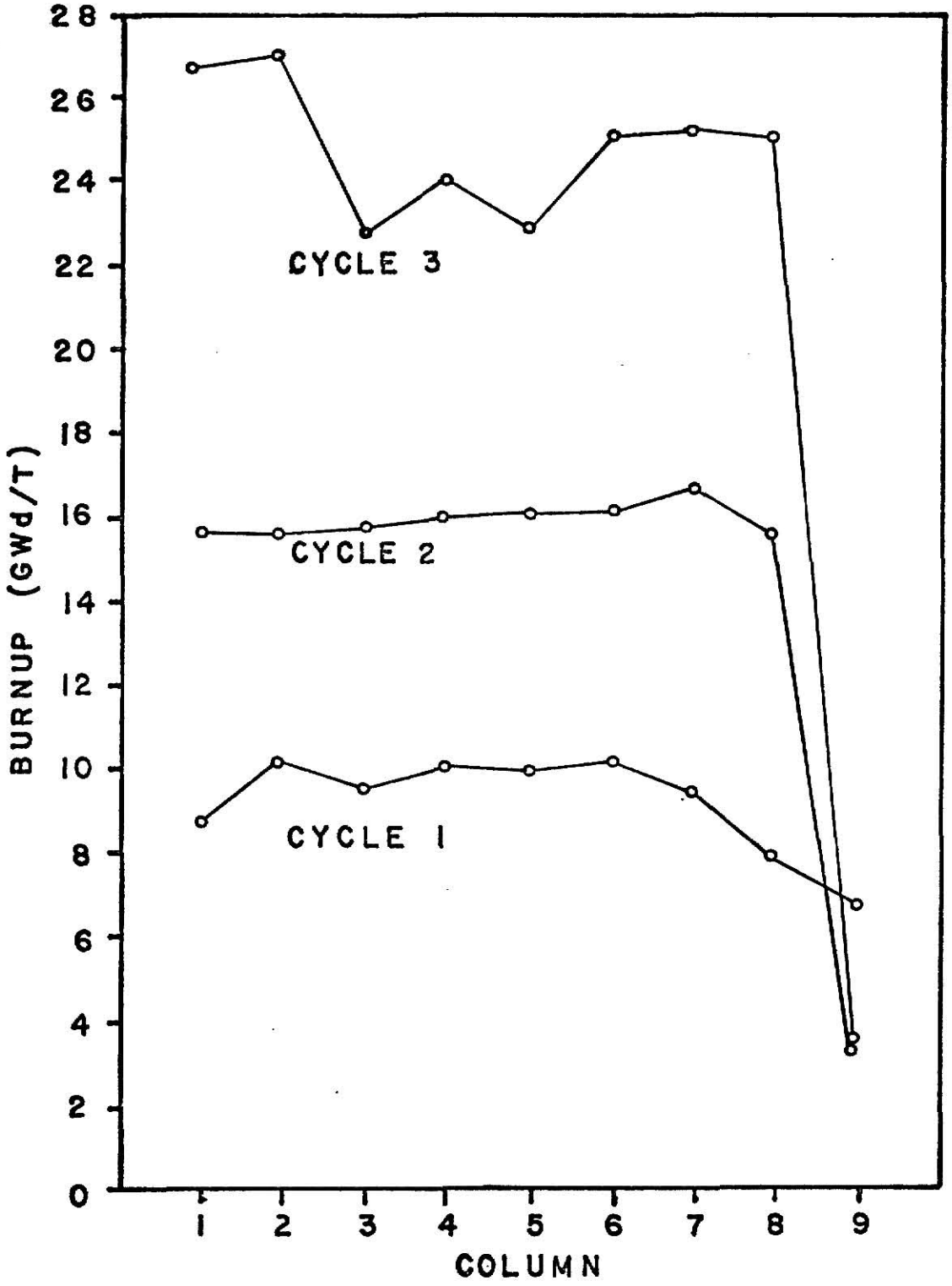


Figure 16. EOL Column Average Burnup for the Reference Case

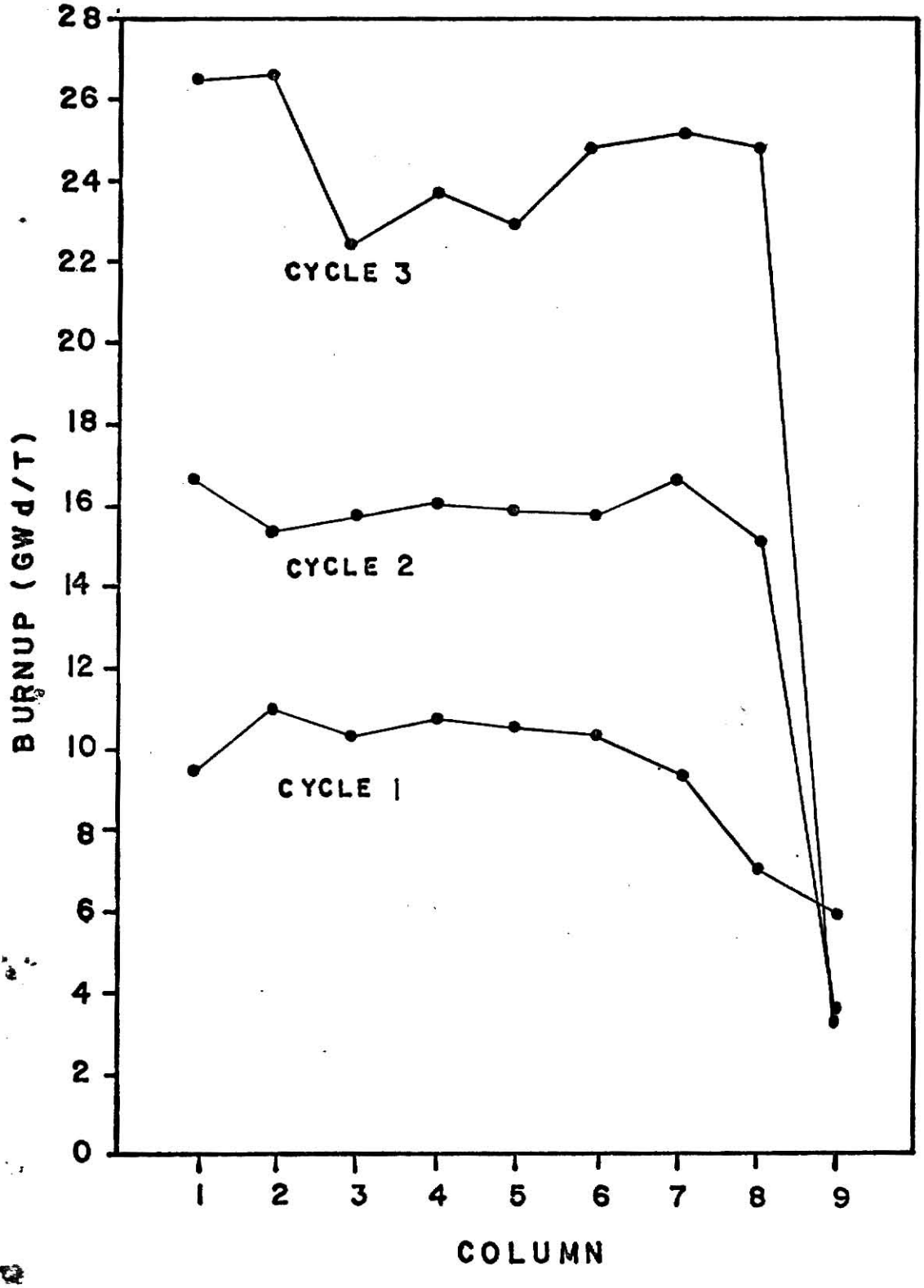


Figure 17. EOL Column Average Burnup for Case 1

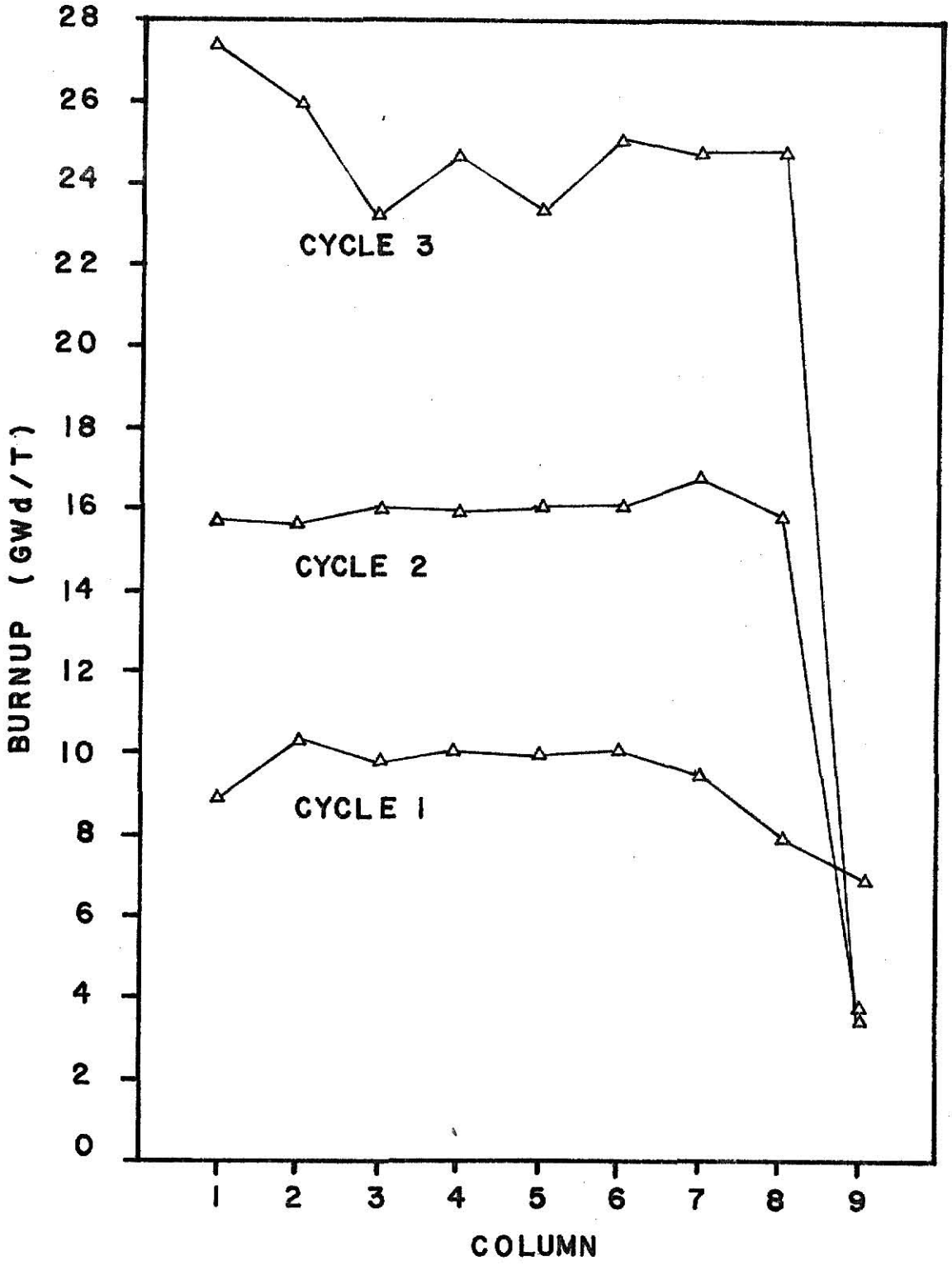


Figure 18. EOL Column Average Burnup for Case 2

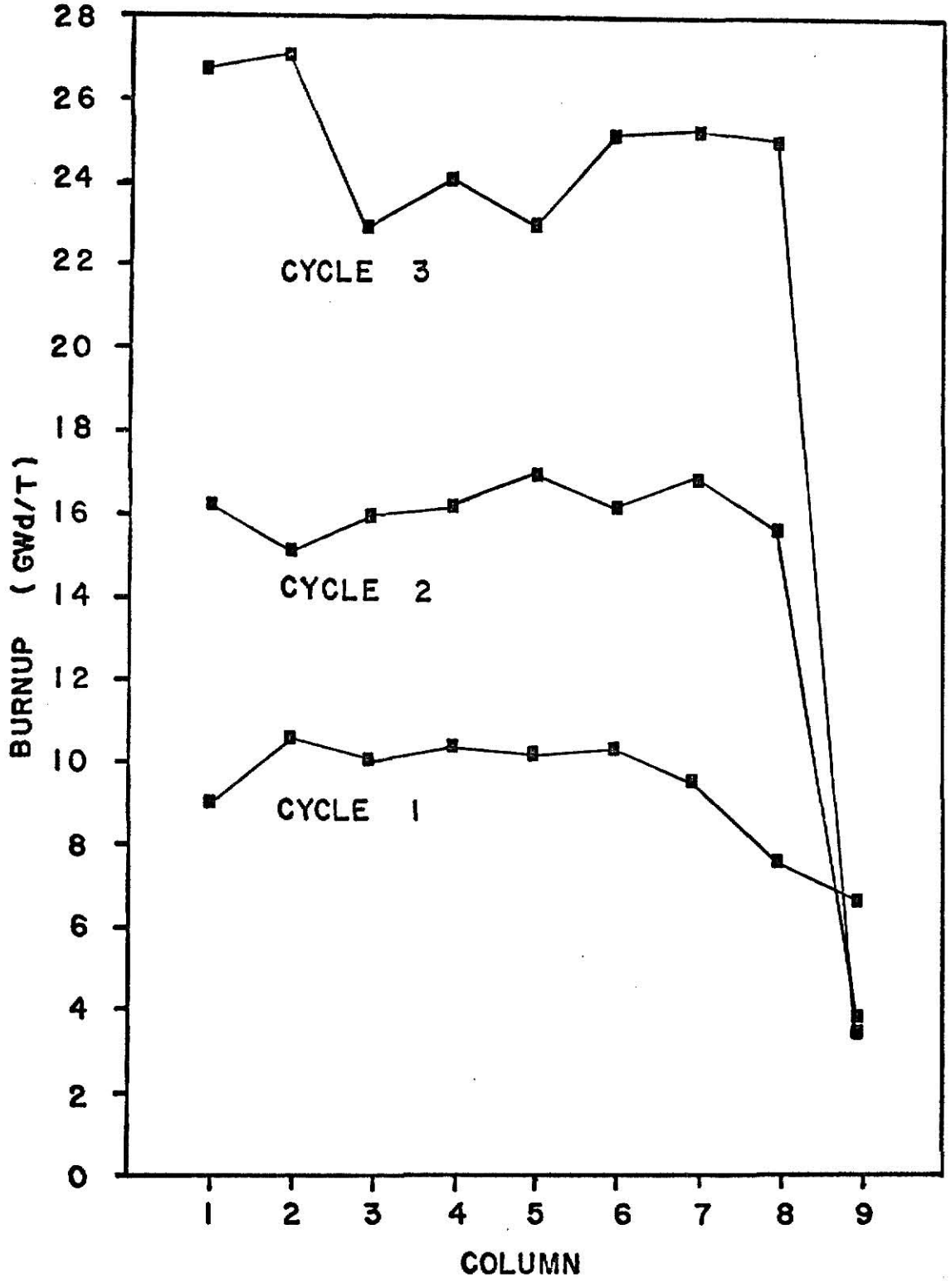


Figure 19. EOL Column Average Burnup for Case 3

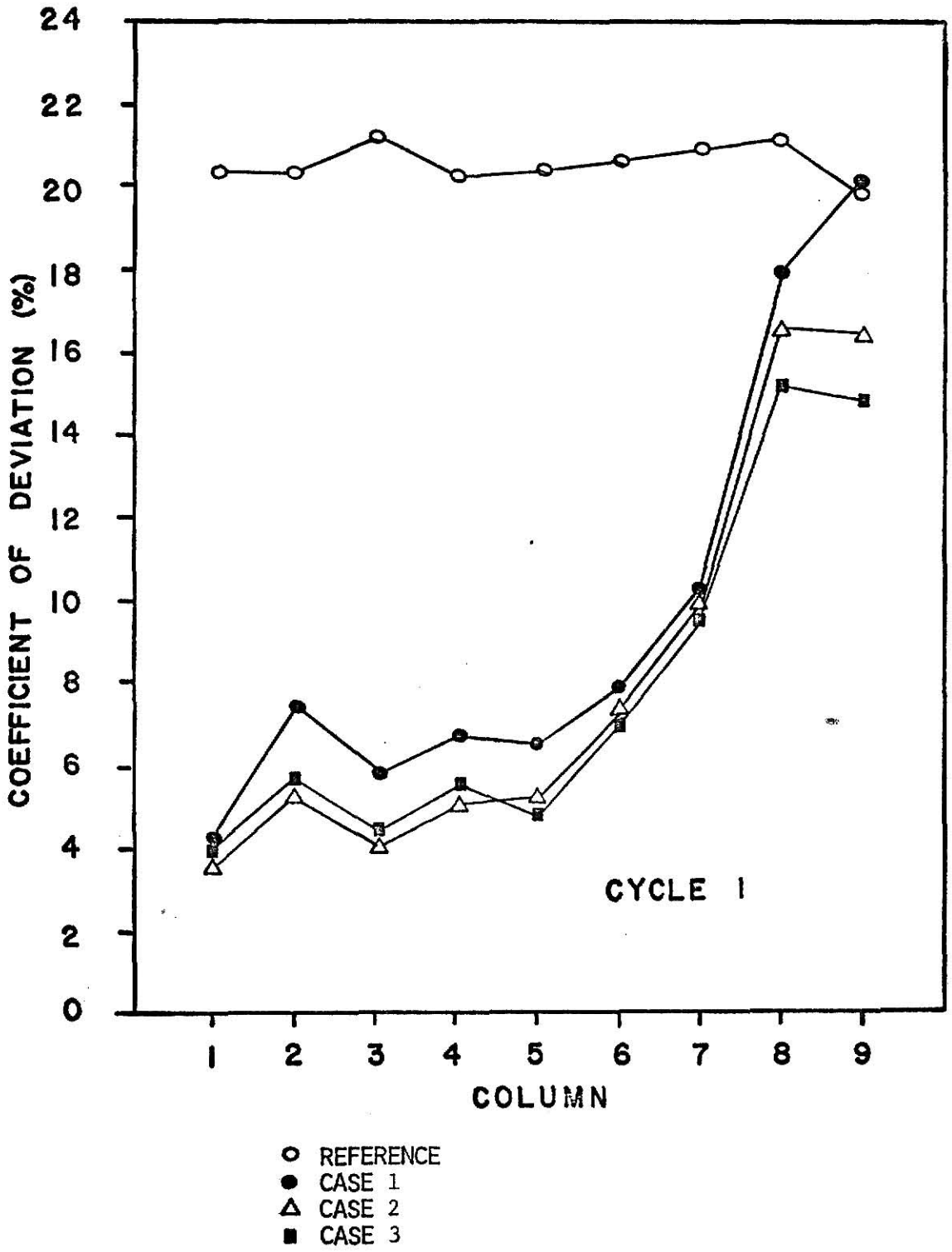


Figure 20. Coefficient of Deviation from Column Average Burnup for Cycle 1 of all Cases Examined

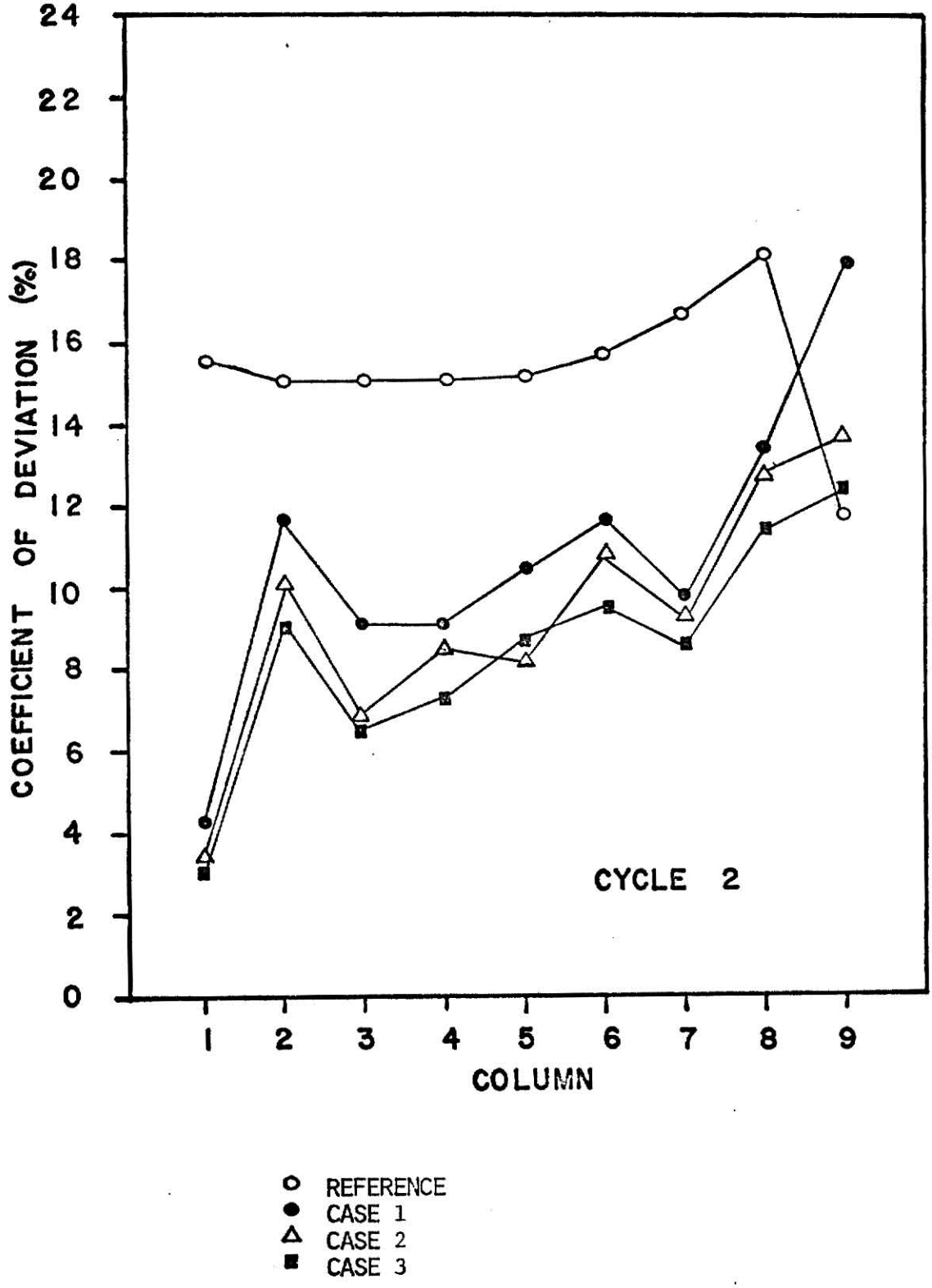


Figure 21. Coefficient of Deviation from Column Average Burnup for Cycle 2 of all Cases Examined

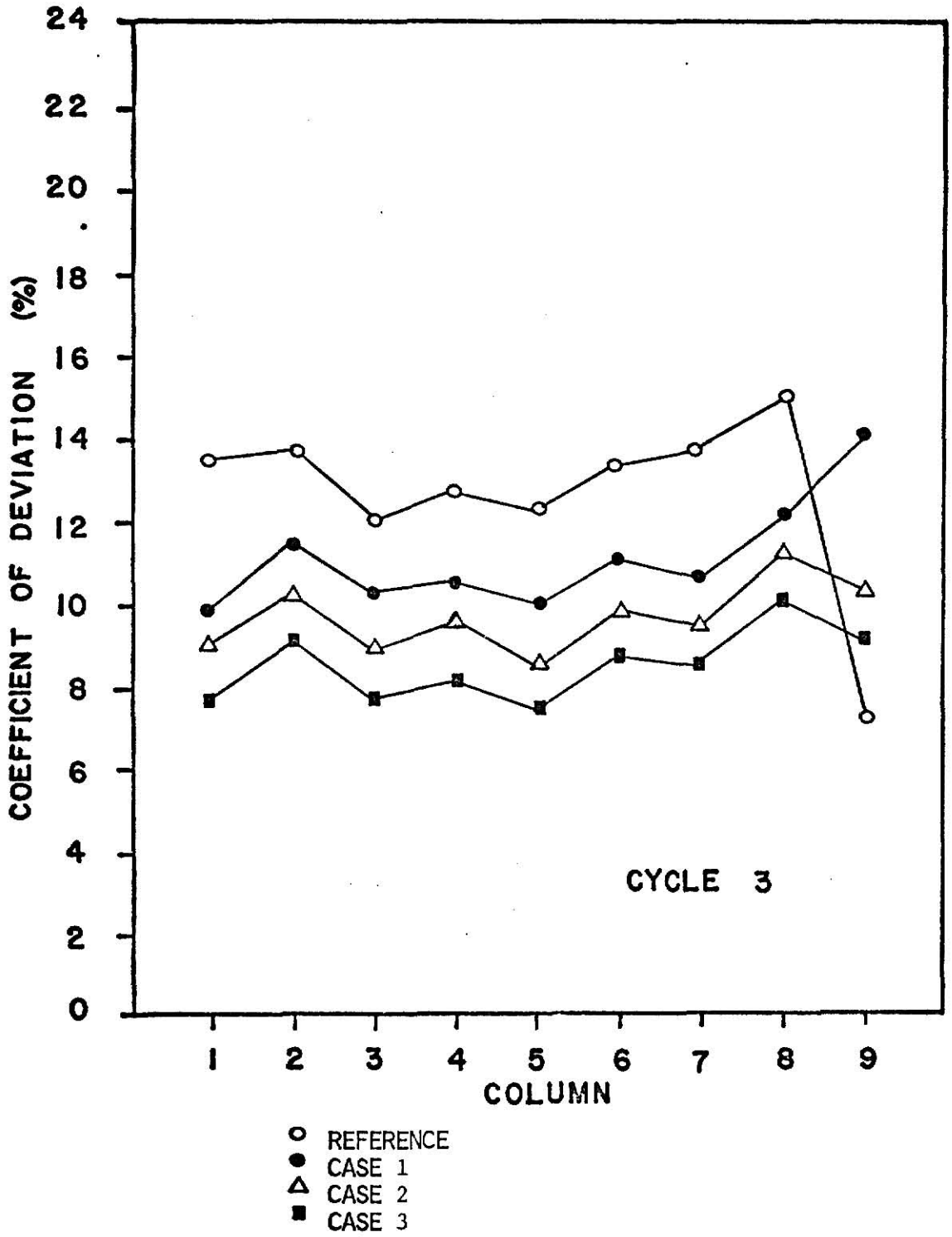


Figure 22. Coefficient of Deviation from Column Average Burnup for Cycle 3 of All Cases Examined

TABLE 3

Burnup of Discharged Fuel Assemblies, X-Y Determination
(MWd/T)

	Cycle 1			
Assembly	Reference	Case 1	Case 2	Case 3
1	8,718	8,767	9,365	8,852
2	9,163	9,258	9,390	9,334
3	9,176	9,630	9,399	9,474
4	9,184	9,642	9,528	9,484
5	9,231	10,084	9,551	9,721
6	9,250	10,109	9,565	9,749
7	9,260	10,123	9,608	9,767
8	9,297	10,132	9,634	9,817
9	9,312	10,149	9,632	9,828
10	9,324	10,202	9,637	9,833
11	9,327	10,216	9,638	9,840
12	9,331	10,216	9,643	9,847
13	9,337	10,229	9,666	9,847
14	9,355	10,237	9,782	9,863
15	9,500	10,245	9,800	9,897
16	9,516	10,246	11,063	9,916
Average	9,268	9,968	9,681	9,692
Std. Dev.	172	415	377	272
Coeff. of Dev.	1.85%	4.16%	3.89%	2.80%

TABLE 4

Burnup of Discharged Fuel Assemblies, X-Y Determination
(MWd/T)

	Cycle 2			
Assembly	Reference	Case 1	Case 2	Case 3
1	13,215	14,043	13,589	13,798
2	13,515	14,880	13,916	14,360
3	16,253	17,611	14,229	16,802
4	16,543	17,974	15,934	17,119
5	16,582	18,006	16,581	17,161
6	16,676	18,057	16,656	17,237
7	16,874	18,127	16,755	17,394
8	16,923	18,151	16,907	17,413
9	17,041	18,191	16,923	17,473
10	17,064	18,224	17,002	17,622
11	17,788	18,443	17,121	17,806
12	18,872	19,678	17,760	19,137
13	18,898	19,720	18,916	19,174
14	20,147	20,987	20,134	20,391
15	20,148	20,990	20,140	20,392
16	20,469	21,302	20,498	20,707
Average	17,313	18,399	17,071	17,746
Std. Dev.	2,024	1,902	2,032	1,877
Coeff. of Dev.	11.69%	10.34%	11.90%	10.58%

TABLE 5

Burnup of Discharged Fuel Assemblies, X-Y Determination
(MWd/T)

		Cycle 3		
Assembly	Reference	Case 1	Case 2	Case 3
1	20,713	20,164	20,820	20,544
2	23,251	22,411	23,163	22,970
3	24,109	23,199	23,755	23,715
4	24,129	23,218	23,995	23,736
5	24,151	23,295	24,837	23,824
6	24,870	24,253	25,623	24,916
7	25,284	24,715	25,637	25,218
8	25,957	24,811	25,811	25,517
9	25,999	25,122	25,869	25,529
10	26,230	25,481	26,067	25,796
11	26,534	25,938	26,380	26,203
12	28,320	27,240	27,968	28,017
13	29,268	27,994	29,272	28,834
14	30,114	28,843	29,437	29,628
15	30,674	29,178	30,153	30,044
16	31,279	29,494	30,493	30,531
Average	26,299	25,335	26,206	25,939
Std. Dev.	2,837	2,572	2,608	2,721
Coeff. of Dev.	10.79%	10.15%	9.95%	10.49%

The general cycle 1 axial behavior of Case 1 follows that of the reference case. The power peaking factors are uniformly about 5 to 10 percent lower and the Figures of Merit are roughly 30 to 40 percent higher as seen in Figures 14 and 15. The standard deviations of the axial fuel burnup are much smaller, as much as 75 percent lower, averaging about 60 percent lower, indicating that the fuel is being consumed much more evenly.

The BOL power profile of cycle 2 begins to show the difficulty inherent in using a multi-enrichment replacement fuel element. The BOL axial power peaking factor is about 10 percent higher than in the reference case. Although the axial peaking factor temporarily dips below the reference case about halfway through the cycle, the EOL axial peaking factor is also higher. This happens because the power generation is primarily coming from the top and bottom thirds of the reactor. The in-shuffling of the initial Type 1 fuel element from cycle 1 has produced a concentration of fuel near the top and bottom, and left the center underenriched.

The third cycle of Case 1 is much like the second, except that at BOL the axial power profile is already split into top and bottom parts due to the enrichment variation. The problem is compounded at EOL. The local power peaks exceed those of the reference case through most of the third core's life, even though overall the core burnup is more even than the reference.

The final burnup profile of Case 1 indicates that, while the deviations from average burnup are lower than the reference case (Figure 22) the elements are now being underused in the center because of the enrichment variation of the replacement elements.

4.3 Discussion of Case 2

Case 2 is again an attempt to achieve the ideal distribution. The results of case 1 indicate that the Type 1 fuel elements in cycle 1 and the replacement elements should be of more uniform enrichment and that the changed top and bottom segments of the other elements should be of higher enrichment. Therefore Case 2 core interior fuel elements have a 3.2% enriched segment top and bottom and the edge elements are entirely of 3.2% enriched fuel.

As in Case 1, the radial behavior of all cycles of Case 2 differs little from the reference case. The power peaking factors, radial Figure of Merit, and average discharge burnup are not significantly changed.

In all three cycles of Case 2 the general behavior of the axial power profile again follows that of the reference case. The peak power is considerably lower at all times except the beginning of the third cycle, when the in-shuffling of the higher enriched, 3.2%, fuel elements causes a temporary overconcentration near the core center. The axial Figures of Merit are higher than in the reference case, and in the first cycle, higher than any other case as well, indicating more even power profiles and therefore axial burnup, except again at the start of cycle 3, when it is slightly lower. The deviations from column average burnup are noticeable lower for all three cycles (Figures 20, 21, and 22) than either the reference case or Case 1, indicating better axial fuel consumption.

4.4 Discussion of Case 3

Case 2, while it has noticeable advantages over Case 1 and the reference case, has a disadvantage in its higher power peaking factor at the beginning of cycle 3. The fuel elements of Case 3 are designed to rectify this problem.

The fuel elements are like those of the reference case but with the enrichments of top and bottom segments of all elements raised to 3.2% as in Case 2.

The general radial and axial behavior of the power and flux distribution follows the general trend dictated by the reference case. The maximum axial power peak is always lower than the reference and the axial Figures of Merit are always higher. The radial averaged behavior is almost identical with the reference. As a result of having such a smooth power profile the Case 3 burnup distributions are also proportionately smoother. The columnwise burn-up deviation from average for Case 3 is approximately equal to that for Case 2 in cycle 1 (Figure 20) but in cycles 2 and 3 is lower in the inner regions of the core, from which the discharged fuel comes, than for any of the other cases (Figures 21 and 22). Thus the fairly simple step of increasing the enrichment of the tips of the fuel elements results in a considerable improvement in fuel use.

5.0 CONCLUSIONS

The axial variations in fuel enrichment considered in this work are relatively small, and thus cause only small deviations from the reference in most factors. Figures 12 and 13 show that there is little difference in the radial power peaking factors or radial Figures of Merit among the four cases. Tables 3, 4, and 5, giving the X-Y discharge burnups, verify the observation that the X-Y results are little changed. The X-Y discharge burnup averages for the three cycles of all four cases follow similar patterns and vary by only a few percent. A most striking verification is given by Figures 16, 17, 18, and 19, the axial column average burnups. The four figures are almost identical, indicating the radial average burnup is unchanged by small variations in the axial enrichment.

While the axial enrichment variations have little effect on the radially averaged behavior of the reactor, they do have a significant impact on axial reactor response. The closer approximation of the Winterburg ideal fuel distribution has an immediate effect on all the pertinent parameters. Figure 14 shows the decrease of axial power peaking factor for the three experimental cases in cycle 1, for Cases 2 and 3 in cycle 2, and for Case 3 in cycle 3. Figure 15 shows the increase in axial Figure of Merit for all cycles of all the experimental cases, indicating a more even core power density profile. Finally, Figures 20, 21, and 22 demonstrate the leveling of the burnup profile along the length of the fuel elements, showing improved fuel usage.

The benefits of small axial variations are plain. Fuel burnup can be increased and power maldistribution decreased. Unfortunately, the fuel

distribution factor Z(r,z) is highly sensitive to fuel enrichment. A deviation in uranium-235 enrichment of only one-tenth of one percent is quite significant. The difficulty in quality control in the manufacture of fuel elements will probably prohibit the implementation of axial variation in the near future. However, the idea is sound and is deserving of further study.

6.0 SUGGESTIONS FOR FURTHER STUDY

A logical extension of this work would be the investigation of a number of burnup cycles beyond the third to determine if an equilibrium cycle is reached, and what the effects of axial variation of fuel enrichment are on the equilibrium should one be reached.

The short length of cycle 2 in all cases indicates that it should be examined. The fuel enrichments or distributions in cycle 1 could be modified so that the shuffle into cycle 2 results in a core with more desirable characteristics.

While the advantages of axial variation of operating parameters are clear it is not obvious that compensation for the added cost in enrichment and manufacture of the axially graded elements is obtained. A study of the costs and benefits would be in order.

Finally, all the enrichment variations studied have been derived from an empirical fit to the ideal distribution. It would be advantageous if the fuel element configuration and shuffling could be devised by a linear programming model like that of Chen's. Conditions on radial and axial conformity to the ideal distribution could be formulated, as well as constraints to insure that fuel elements are shuffled as continuous units. This would ensure a three-dimensional adherence to the ideal distribution and thus give a reasonable base from which to launch further burnup studies.

7.0 LITERATURE CITED

1. SILVENNOINEN, P., Reactor Core Fuel Management, Pergamon Press, New York, 1976.

2. BENEDICT, M. and PIGFORD, T., Nuclear Chemical Engineering, McGraw-Hill Book Company, Inc., New York, 1057.

3. HAWK, M. "A Cash-Flow Study for a Non-Base Loaded Power Reactor," Master's Thesis, Kansas State University, 1971.

4. GLASSTONE, S. and SESONSKE, A., Nuclear Reactor Engineering, Van Nostrand Reinhold Company, 1967.

5. WALL, I., and Fenech, H., "The Application of Dynamic Programming to Fuel Management Optimization," Nuclear Science and Engineering, 22, 1965, 285-297.

6. STOVER, R., and SESONSKE, A., "Optimization of BWR Fuel Management Using an Accelerated Exhaustive Search Algorithm," Journal of Nuclear Energy, 23, 1969, 673-682.

7. FAGAN, J. and SESONSKE, A., "Optimal Fuel Replacement in Reactivity Limited Systems," Journal of Nuclear Energy, 23, 1969.

8. NAFT, B. and SESONSKE, A., "A Direct Search Algorithm for Optimizing Fuel Loading Patterins in PWR'S," Transactions, American Nuclear Society, 13, 1970, 40.

9. MARTINET, M. and SANTAMARINA, A., "PWR Fuel Loading Optimization," Transactions, American Nuclear Society, 20, 1975, 371.

10. HOSHINO, T., "In-Core Fuel Management Optimization by Heuristic Learning Technique," Nuclear Science and Engineering, 49, 1972, 59.

11. TABAK, D., "Optimization of Nuclear Reactor Fuel Recycle via Linear and Quadratic Programming," IEE Transactions on Nuclear Science, NS-15, 1, February 1968, 60-64.

12. FULFORD, P. and SESONSKE, A., "In-Core Fuel Management Optimization," Nuclear News, 13, 7, July 1970, 58-62.

13. SUZUKI, A., and KIYOSE, R., "Application of Linear Programming to Refueling Optimization for Light Water Moderated Reactors," Nuclear Science and Engineering, 46, 1, 1971, 112-130.

14. CLARK, S., "A Linear Programming Reactor Refueling Model," Master's Thesis, Kansas State University, 1971.

15. SAUAR, T., "Application of Linear Programming to In-Core Fuel Management Optimization in Light Water Reactors," Nuclear Science and Engineering, 46, 1971, 274.

16. ORCHARD-HAYS, W., Advanced Linear Programming Techniques, McGraw-Hill Book Company, New York, 1968.

17. MINGLE, J., "In Core Fuel Management via Peturbation Theory," Nuclear Technology, 27, 1975, 248.

18. BROWN, M., MINGLE, J., and ECKHOFF, N., "Key Constraints for Simulation of Nuclear Fuel Management," ICRE Report 73.01.

19. CHEN, Y., MINGLE, J. and ECKHOFF, N., "Optimal Power Profile Fuel Management," Submitted to Annals of Nuclear Science and Engineering.

20. CHITKARA, K., and WEISMAN, J., "An Equilibrium Approach to Optimal In-Core Fuel Management for Pressurized Water Reactors," Transactions, American Nuclear Society, 15, 1972, 665.

21. HOWLAND, H., DORNY, C., and STOVER, R., "Power Flattening and Loading Pattern Searches for Pressurized Water Reactors," Transactions, American Nuclear Society, 16, 1973, 170.

22. MOTODA, H., HERCZEG, J., and SESONSKE, A., "Optimization of Refueling Schedule for Light Water Reactors," Nuclear Technology, 25, March 1975, 477-496.

23. WINTERBURG, F., "Special Nonuniform Fuel Distributions and Cooling Problems in Reactors," Nuclear Data and Reactor Theory, Second United Nations International Conference in the Peaceful Uses of Atomic Energy, Geneva, 16, 1958, 675-686.

24. MELICE, M., "Pressurized Water Reactor Optimal Core Management and Reactivity Profiles," Nuclear Science and Engineering, 37, 3, 1969, 451-477.

25. MOTODA, H., "Optimal Control Rod Programming of Light Water Reactors in Equilibrium Fuel Cycle," Nuclear Science and Engineering, 46, 1971, 88.

26. SUZUKI, A. and KIYOSE, R., "Maximizing the Average Fuel Burnup Over Entire Core: A Poison Management Optimization Problem for Multizone Light Water Reactor Cores," Nuclear Science and Engineering, 44, 1971, 121.

27. MOTODA, H. and KAWAI, T., "A Theory of Control Rod Programming Optimization in Two-Region Reactors," Nuclear Science and Engineering, 39, 1970, 114.

28. FUKUNISHI, MOTODA, and KAWAI, "Optimal Control Rod Programming in an Equilibrium Fuel Cycle," Transactions, American Nuclear Society, 13, 1970, 41.

29. BENEDICT, M., et al., "FUELCYCLE, A New Computer Code for Fuel Cycle Analysis, Part II, Examples of Applications," Nuclear Science and Engineering, 11, 1961, 386.

30. Westinghouse RESAR-3, Summary Description, Reference Safety Analysis Report, Volume 2, Section 4.1, Westinghouse Nuclear Energy Systems.

APPENDIX A: THE CODE BURNRZ

A.1 BURNRZ Functions

The code BURNRZ calculates the core burnup, power and flux profiles, and reactivity changes of a load of PWR fuel from start to end of a single burnup cycle. The processes or governing equations used in the dominant steps of the program are discussed below. The discussion follows the progression of the code flow diagram presented in the next section (A.2).

The code inputs are, first, the microscopic cross sections and nuclear physics constants of the elements used in the fuel and structure. Read in next are some of the basic lattice cell parameters; the diffusion coefficients, Fermi Age, diffusion length, moderator/fuel/cladding volume ratios, disadvantage factors, fast fission factor, and resonance escape probabilities for the fuel assemblies. Finally, the basic core size and geometry variables and reactor power level and operating history are input.

Subroutine XYTOR does the radial averaging. The square grid array of fuel assemblies is broken into an equivalent set of concentric annular rings. The amount of each fuel element present in each ring is found by a straight-line-segment technique, in which the arc of the zone boundary through the X-Y grid cell is represented by a straight line. The area of the cell within the zone is then approximated with various triangle or trapezoid area summations. When the fraction of each fuel element in a zone is known, the atom densities of the nuclides in that element are multiplied by the fraction and added to the zone, resulting ultimately in an overall zone average atom density.

The subroutine LIQHOH is used as a tabular storage of thermodynamic data for the PWR coolant water. The data is read in for water at the operating pressure of the PWR and for a wide range of temperatures. Since

the pressure is constant, a specification of any other property of the coolant defines the state of water. Thus, if the subroutine is called, giving one property, any of several other properties can be returned. The variables considered, besides pressure, are temperature, enthalpy, and density. The returned values are found by interpolation or extrapolation from the tabulated data.

The burnup calculation is based on the atom densities of the fission products present in the fuel element. Expressed in terms of megawatt-days per metric ton of uranium, the burnup is found from the equation

$$\text{B.U.} = \frac{1}{2} \frac{(\text{atoms of fission products/cm}^3)}{(\text{atoms of fuel originally present/cm}^3)} \frac{(\text{MWd})}{(1.052 \text{ g U-235})} \frac{(10^6 \text{g})}{(\text{ton})} \tag{A.1}$$

since to a good approximation of fissioning of 1.052 grams of U-235 is required for the release of one megawatt-day of energy. Use of atoms of fission products allows the contribution of plutonium to be included, with the assumption that the energy release is roughly equivalent for Pu-238 and U-235.

Subroutine CPROP is used to determine the group constants for each mesh point in the calculations. For points outside the reactor or reflector, a vacuum boundary is assumed, with zero properties, which implies that any neutron which gets that far is immediately lost to the system. For the water reflector the properties are set to those input for the moderator, except for those cases for which soluble boron control poison is used, for which the diffusion length is modified to account for the increased absorptions. The regions containing fuel assemblies are treated as homogenized regions in standard unit cell practice. The fuel macroscopic thermal fission cross sections are based on the product of atom densities and

microscopic cross sections. The fast group removal cross sections are based on the number of neutrons per fission and the fast fission factor, as well as the fission cross sections, as

$$\Sigma_R = \left(\sum_{i=1}^{6} \nu_i \Sigma_i\right)(\varepsilon-1)/(\nu_{238}) \tag{A.2}$$

The thermal utilization is found as

$$f = \frac{N_{fuel}\ \sigma_{f\ fuel}}{N_{fuel}\ \sigma_{f\ fuel}+N_{poison}\ \sigma_{a\ poison}+\Sigma_{a\ material}(\phi_m/\phi_f)(V_m/V_f)} \tag{A.3}$$

The diffusion length is approximated as

$$L^2_{fuel} = L^2_{moderator}\ (1-f) \tag{A.4}$$

The infinite multiplication factor is

$$k_\infty = pf\ \Sigma_i(\nu_i\Sigma_i)/\Sigma_i(\Sigma_i) \tag{A.5}$$

Subroutine RZFLUX solves the two-dimensional two-group neutron diffusion equations

$$-\nabla\cdot D_f\nabla\phi_f + \Sigma_R\ \phi_f = \nu\Sigma_f\varepsilon\phi_{th}/k_\infty \tag{A.6}$$

$$-\nabla\cdot D_{th}\nabla\phi_{th} + \Sigma_{tot}\phi_{th} = p\Sigma_R\phi_f \tag{A.7}$$

The radial-axial two-dimensional Laplacian operator gives, with the group notation suppressed and the source term represented by S,

$$-\frac{1}{r}\frac{\partial}{\partial r} Dr \frac{\partial\phi}{\partial r} - \frac{\partial}{\partial z} D \frac{\partial\phi}{\partial z} + \Sigma\phi = S \tag{A.8}$$

This equation is integrated over the radial-axial area around the mesh point as

$$\int_{r-h/2}^{r+h/2} 2\pi r\ dr \int_{z-\ell/2}^{z+\ell/2} dz\ \left[-\frac{1}{r}\frac{\partial}{\partial r} Dr \frac{\partial\phi}{\partial r} - \frac{\partial}{\partial z} D \frac{\partial\phi}{\partial z} + \Sigma\phi\right] = f \tag{A.9}$$

where

$$f = \int_{r-h/2}^{r+h/2} 2\pi r\, dr \int_{z-\ell/2}^{z+\ell/2} dz\, S \tag{A.10}$$

Equation (A.9) integrates to the form

$$-2\pi\ell \left[Dr \left.\frac{\partial\phi}{\partial r}\right|_{r+h/2} - Dr \left.\frac{\partial\phi}{\partial r}\right|_{r-h/2}\right] - \pi r^2 \Big|_{r-h/2}^{r+h/2} \left[D \left.\frac{\partial\phi}{\partial z}\right|_{z+\ell/2} - D \left.\frac{\partial\phi}{\partial z}\right|_{z-\ell/2}\right]$$

$$+ \pi r^2 \Big|_{r-h/2}^{r+h/2} \ell\Sigma\phi = f \tag{A.11}$$

The last term on the left hand side of (A.11) is evaluated as

$$r^2 \Big|_{r-h/2}^{r+h/2} = 2rh \tag{A.12}$$

The derivatives in (A.11) may be approximated, assuming the diffusion coefficient to be constant, with a central difference approximation as

$$Dr \left.\frac{\partial\phi}{\partial r}\right|_{r+h/2} = D(r+h/2)\ [\phi_{r+h} - \phi_r]/h \tag{A.13}$$

Substituting these results into Equation (A.11) and assuming a square box around the mesh point so that $h=\ell$, the resultant equation is

$$-(1 + h/2r)\phi_{r+h,z} - (1 - h/2r)\phi_{r-h,z} - \phi_{r,z+h} - \phi_{r,z-h}$$

$$+ [4 + h^2\Sigma/D]\phi_{r,z} = f/2\pi Dr \tag{A.14}$$

Equation (A.14) is the general form of the diffusion equation solved by the code BURNRZ.

A good derivation of the finite difference equations is given in Bell and Glasstone[4] for X-Y geometry. The procedure is equivalent to the R-Z case with only small variations in the Laplacian operator.

The resultant set of coupled linear equations for the flux density at each mesh point is solved using point simultaneous overrelaxation. In symbolic form the matrix equations can be written as

$$A\Phi = S \tag{A.15}$$

The matrix A can be partitioned into a diagonal matrix D, an upper triangular matrix E, and a lower triangular matrix F, such that

$$A = D - E - F \ . \tag{A.16}$$

Substitution into equation (A.15) gives

$$(D - E - F)\Phi = S \tag{A.17}$$

which may be rewritten as

$$D\Phi = E\Phi + F\Phi + S \ . \tag{A.18}$$

Since D is a diagonal matrix, it has an easily determined inverse, so

$$\Phi^{(m+1)} = D^{-1} \ [E\Phi^{(m+1)} + F\Phi^{(m)} + S] \tag{A.19}$$

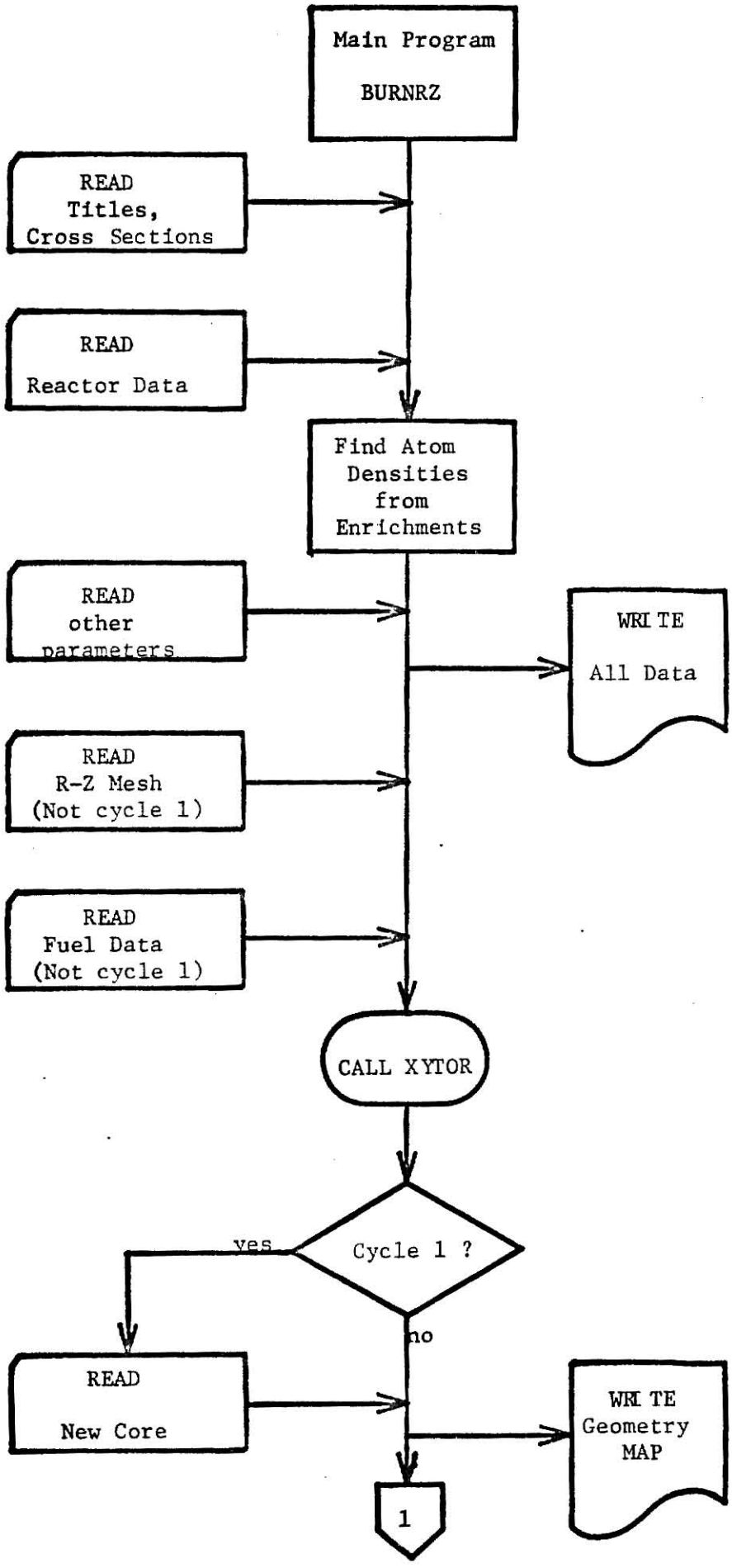
where the superscripts on the solution vector Φ indicate that in the iterative solution the components of Φ for the m+1 iteration are calculated from the values of the m-th iteration and that the new values are used as soon as they are calculated. Equation (A.19) can be written as

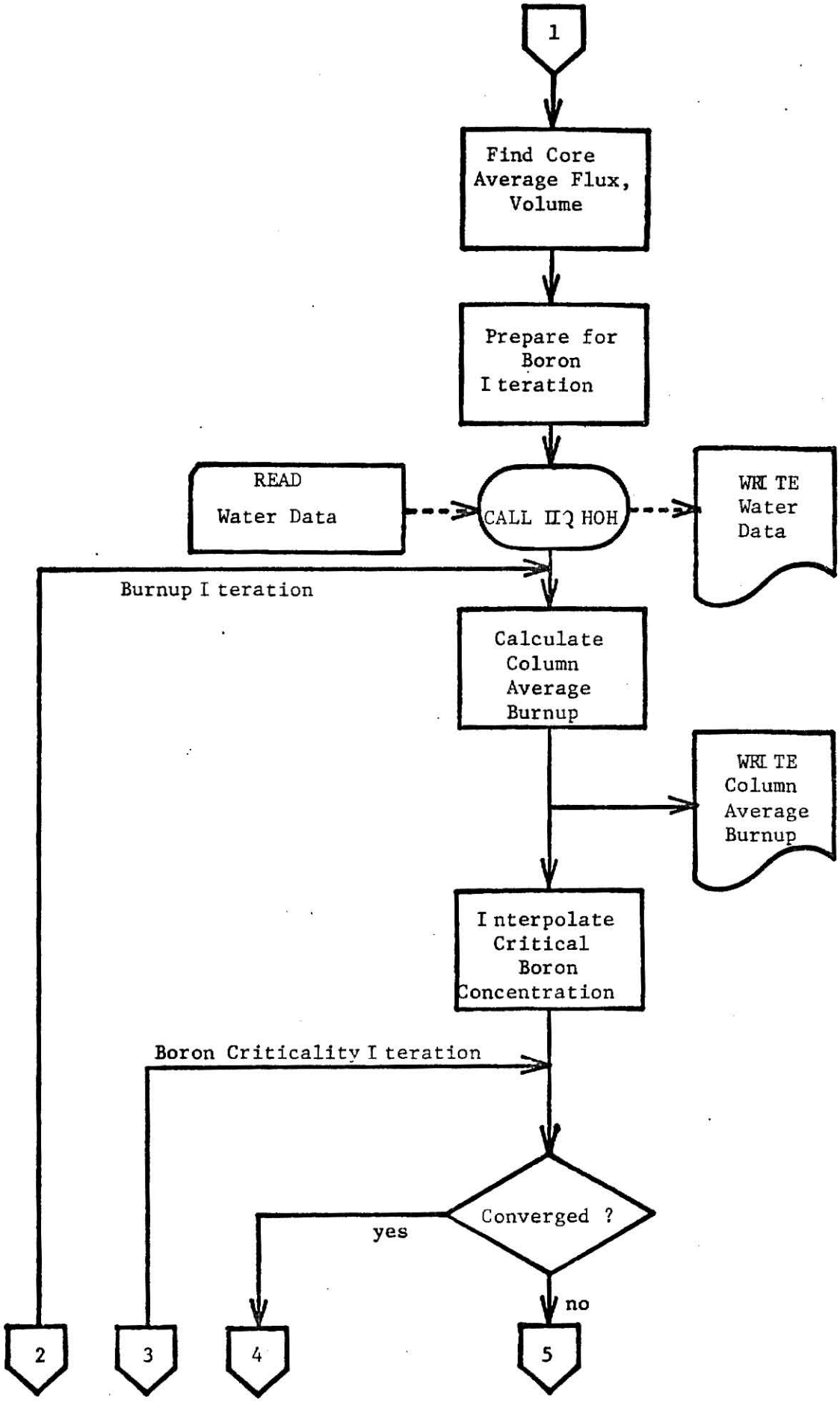
$$\Phi^{(m+1)} = \Phi^{(m)} + [D^{-1}(E\Phi^{(m+1)} + F\Phi^{(m)} + S) - \Phi^{(m)}]\omega \tag{A.20}$$

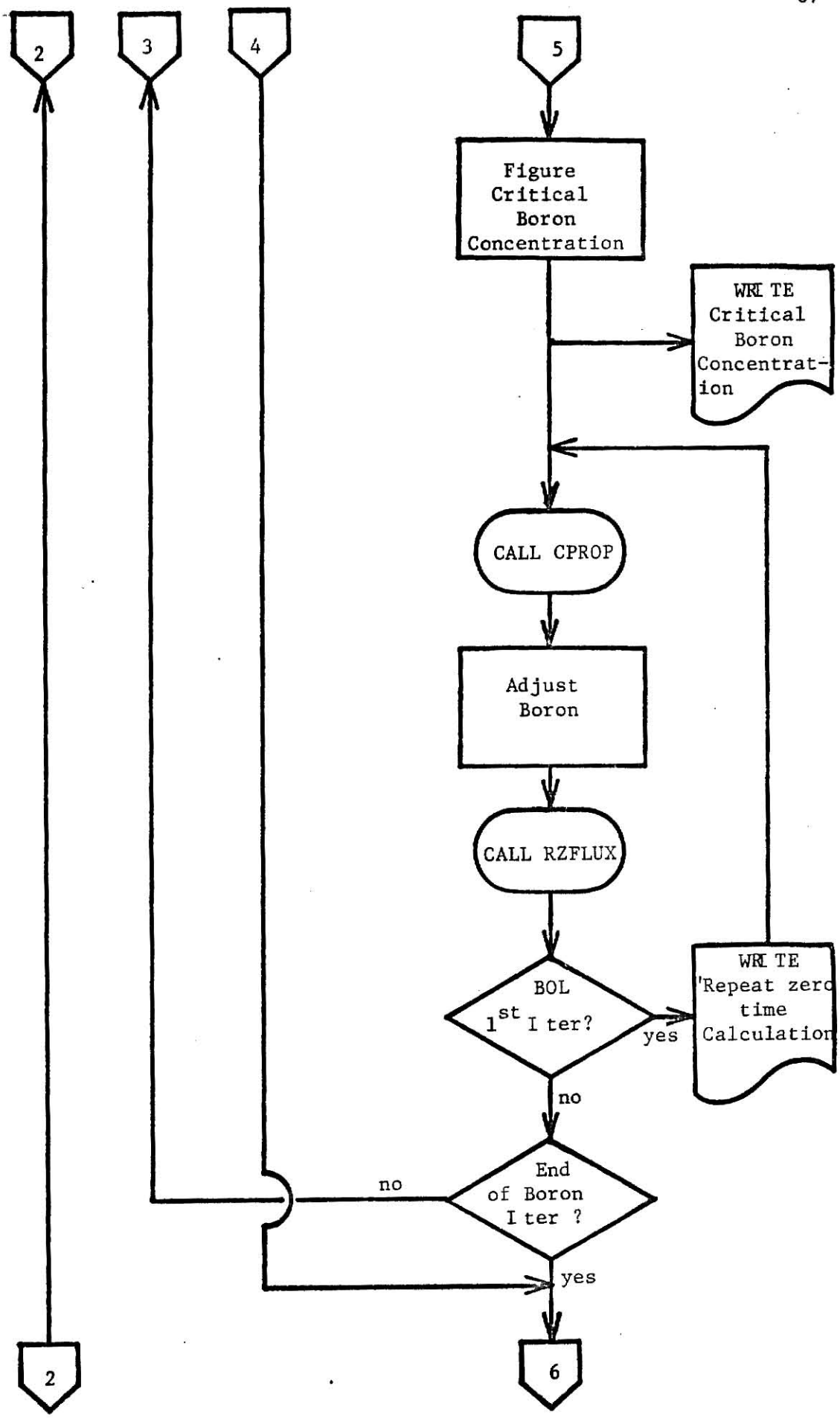
The term in brackets is the correction term added to the m-th iteration. An acceleration parameter ω, usually with a value between one and two, may be added to give the scheme an increased rate of convergence. A good discussion of successive overrelaxation, with methods of chosing the optimum relaxation paramter ω, is given by Varga[31].

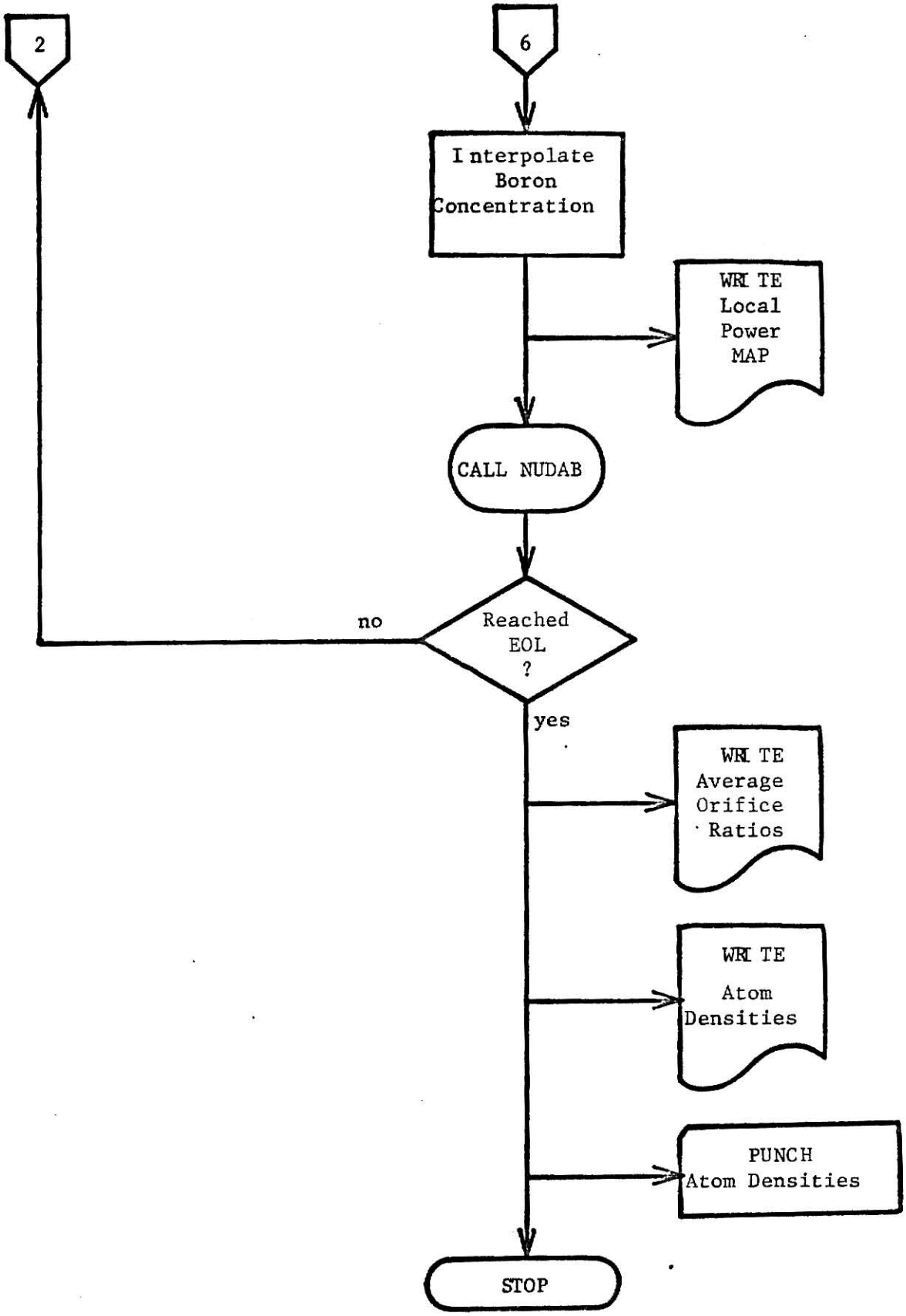
The subroutine NUDAB is used to find the atom densities if the various nuclides followed in the program. The nuclide densities are modeled as a set of coupled differential equations. The derivation of these equations and a complete listing thereof can be found in Hawk[3] or in Chen[19].

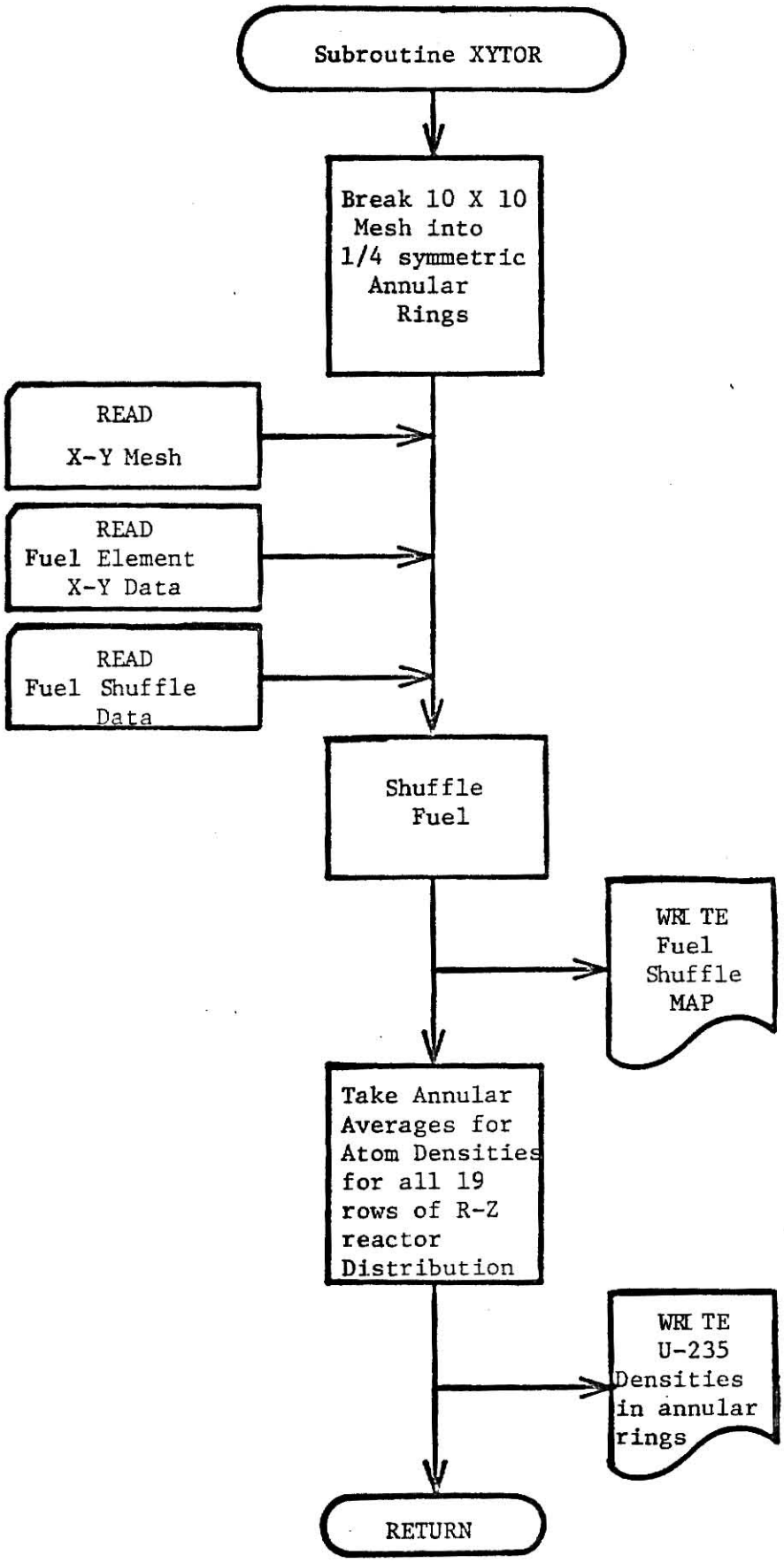
A.2 Flow Diagram of the Code BURNRZ

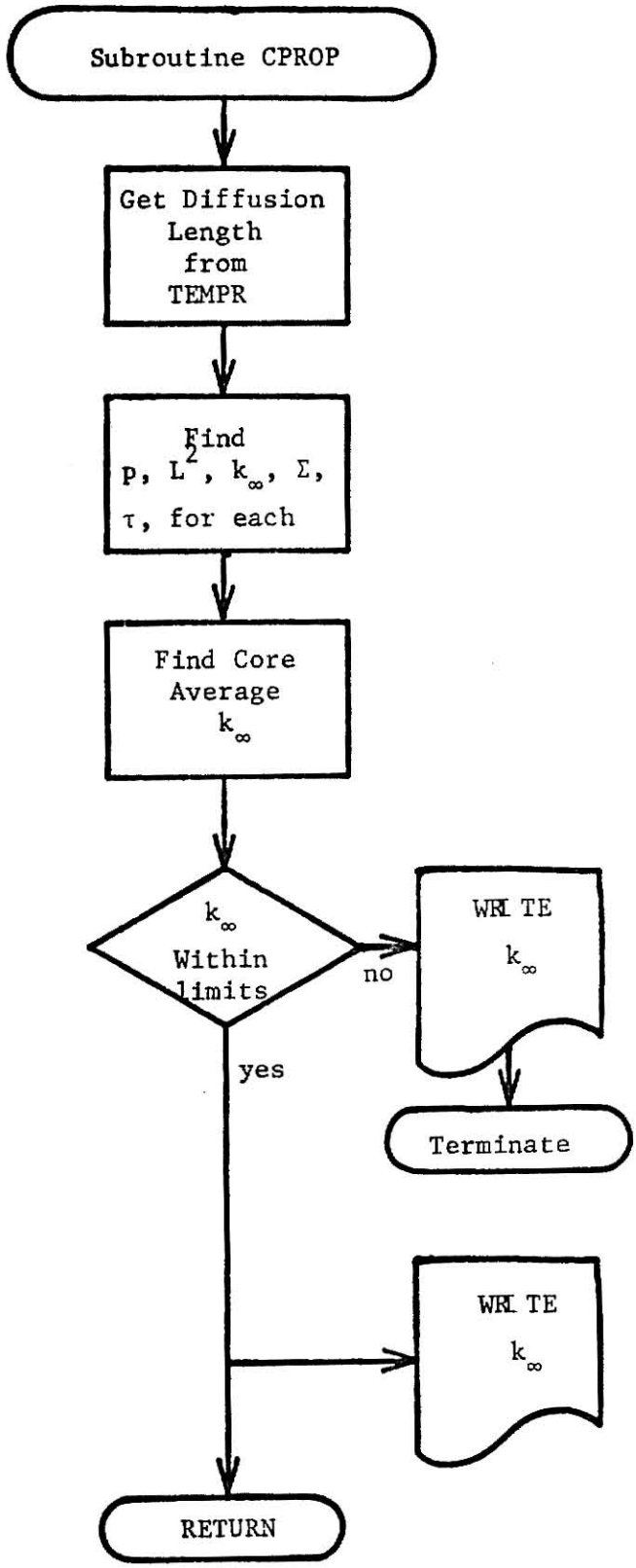

Main Program
BURNRZ
READ
Titles,
Cross Sections
READ
Reactor Data
Find Atom
Densities
from
Enrichments
READ
other
parameters
WRITE
All Data
READ
R-Z Mesh
(Not cycle 1)
READ
Fuel Data
(Not cycle 1)
CALL XYTOR
yes
Cycle 1 ?
no
READ
New Core
WRITE
Geometry
MAP
1

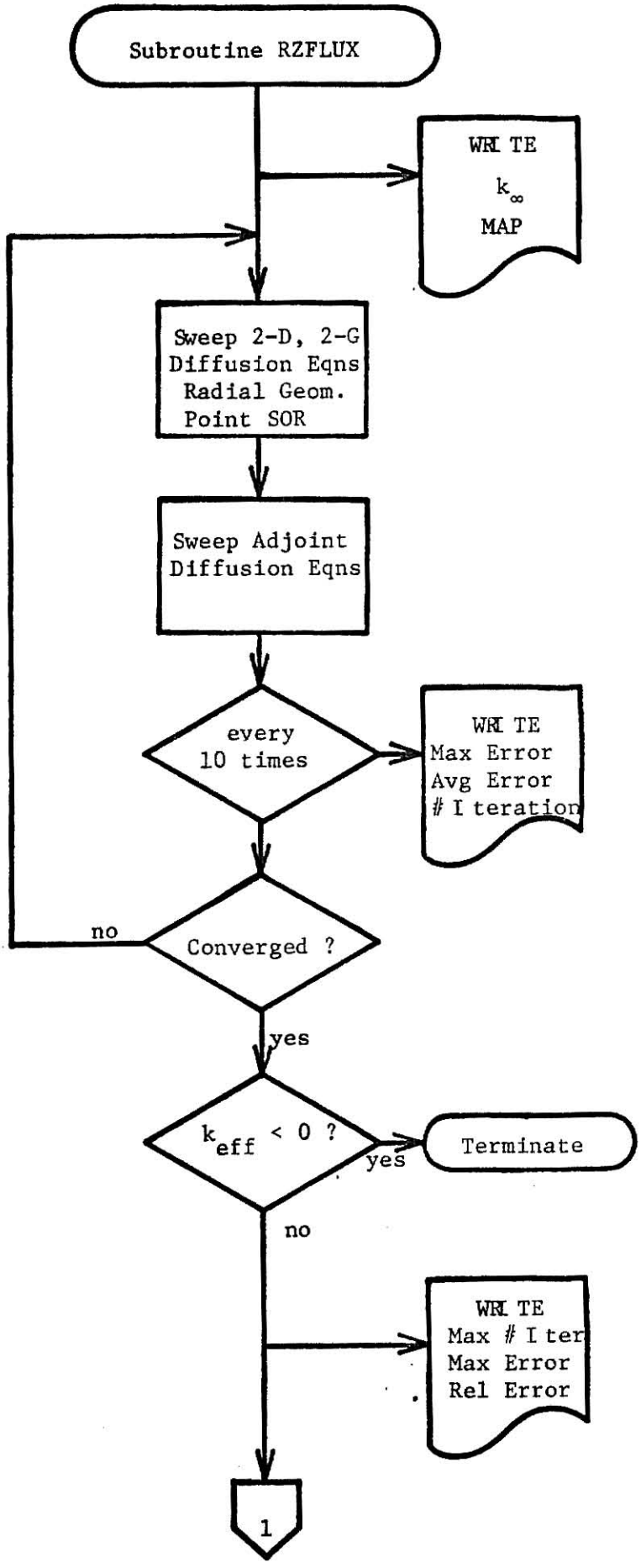

1
Find Core Average Flux, Volume
Prepare for Boron I teration
READ Water Data
CALL HOH
WRITE Water Data
Burnup I teration
Calculate Column Average Burnup
WRITE Column Average Burnup
I nterpolate Critical Boron Concentration
Boron Criticality I teration
Converged ?
yes
no
2
3
4
5

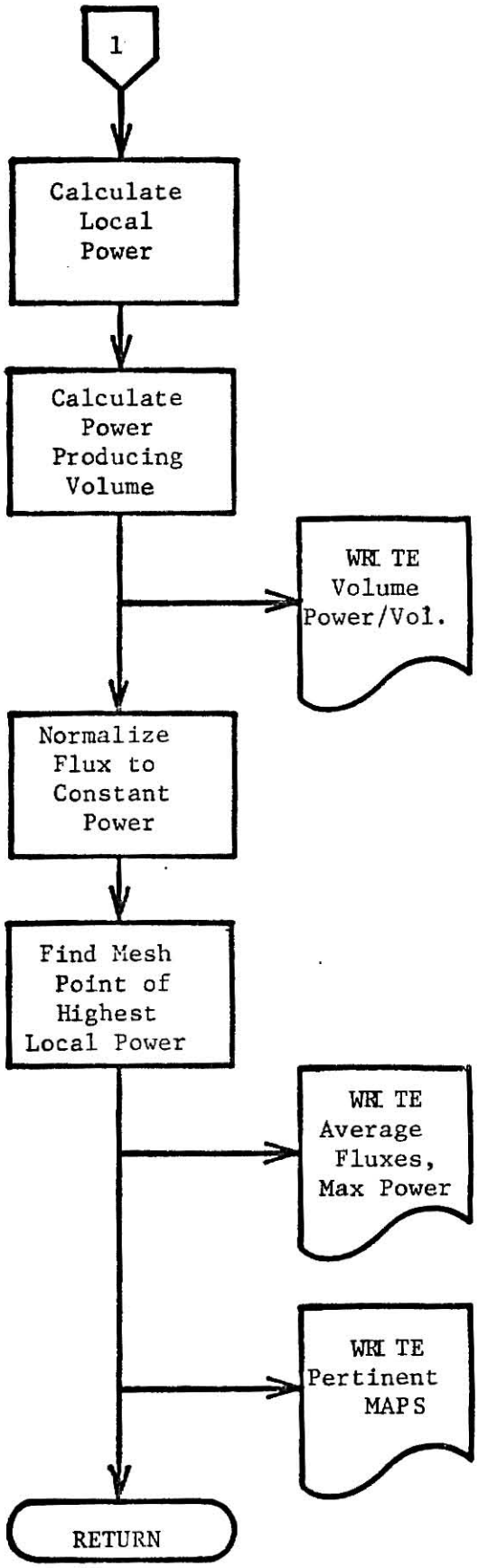

2
3
4
5
Figure Critical Boron Concentration
WRITE Critical Boron Concentration
CALL CPROP
Adjust Boron
CALL RZFLUX
BOL 1st Iter?
yes
WRITE 'Repeat zero time Calculation
no
End of Boron Iter ?
no
yes
2
6

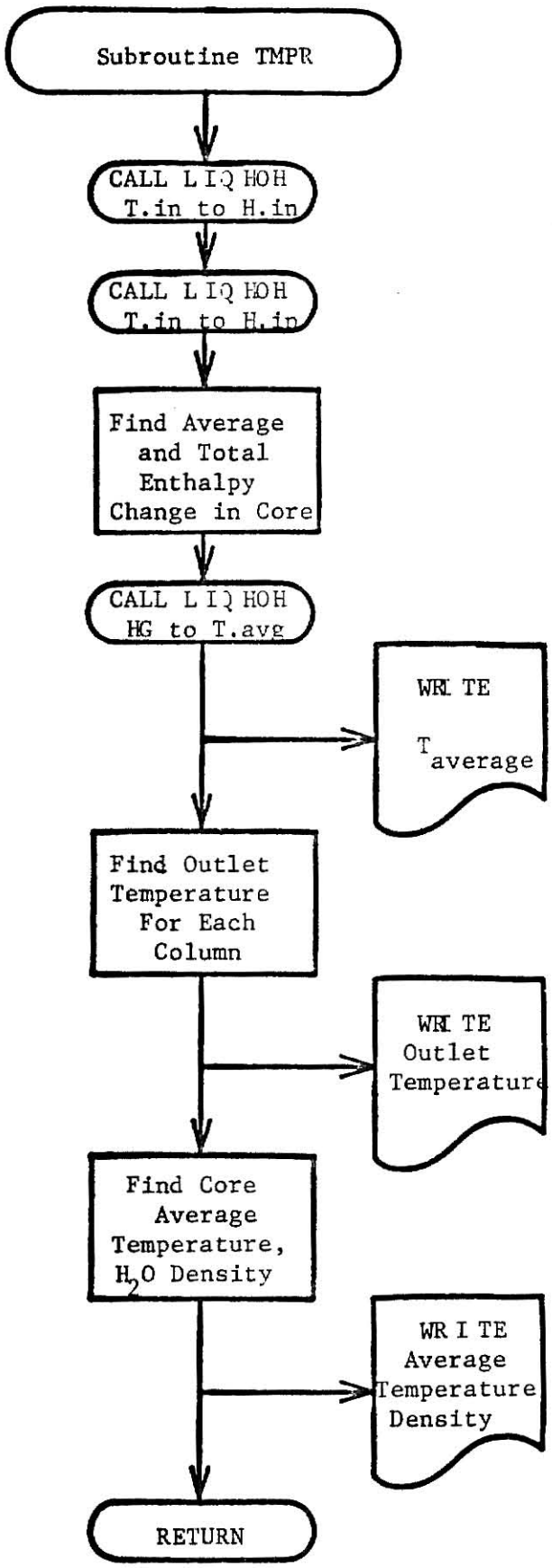

2
6
Interpolate Boron Concentration
WRITE Local Power MAP
CALL NUDAB
Reached EOL ?
no
yes
WRITE Average Orifice Ratios
WRITE Atom Densities
PUNCH Atom Densities
STOP


Subroutine XYTOR
Break 10 X 10 Mesh into 1/4 symmetric Annular Rings
READ X-Y Mesh
READ Fuel Element X-Y Data
READ Fuel Shuffle Data
Shuffle Fuel
WRITE Fuel Shuffle MAP
Take Annular Averages for Atom Densities for all 19 rows of R-Z reactor Distribution
WRITE U-235 Densities in annular rings
RETURN


Subroutine CPROP
Get Diffusion Length from TEMPR
Find p, L^2, k_∞, Σ, τ, for each
Find Core Average k_∞
k_∞ Within limits
no
WRITE k_∞
Terminate
yes
WRITE k_∞
RETURN


Subroutine RZFLUX
WRITE
k_∞
MAP
Sweep 2-D, 2-G
Diffusion Eqns
Radial Geom.
Point SOR
Sweep Adjoint
Diffusion Eqns
every
10 times
WRITE
Max Error
Avg Error
Iteration
no
Converged ?
yes
k_{eff} < 0 ?
yes
Terminate
no
WRITE
Max # Iter
Max Error
Rel Error
1


1
Calculate Local Power
Calculate Power Producing Volume
WRITE Volume Power/Vol.
Normalize Flux to Constant Power
Find Mesh Point of Highest Local Power
WRITE Average Fluxes, Max Power
WRITE Pertinent MAPS
RETURN


Subroutine TMPR
CALL LIQHOH T.in to H.in
CALL LIQHOH T.in to H.in
Find Average and Total Enthalpy Change in Core
CALL LIQHOH HG to T.avg
WRITE T average
Find Outlet Temperature For Each Column
WRITE Outlet Temperature
Find Core Average Temperature, H2O Density
WRITE Average Temperature Density
RETURN

A.3 Computer Listing of the Code BURNRZ

THIS BOOK IS OF POOR LEGIBILITY DUE TO LIGHT PRINTING THROUGH OUT IT'S ENTIRETY.

THIS IS AS RECEIVED FROM THE CUSTOMER.

```
C ***    INDIVIDUAL ASSEMBLY BURNUP PROGRAM                                  ABP00020
C ***    REQUIRES SEPARATE INPUT DECK OF ATOM DENSITIES
C ***    A-Z FORM
      DATA PI/3.14159265/
      DATA L,K,M/5,6,7/,AM/6.02E23/                                          ABP00030
      DATA KPRINT/1/
      DIMENSION SF(10,10), FEN(10)
      DIMENSION MESH(190), MESHN(190), ACMT(20), ENRICH(10,20),
     1F25(10,20), F28(10,20), BPPM(10), SAB(10,190), TPSUM(190),
     2SCB(10,190), SCD(10,10,20)
      COMMON/PARM/EPS,P1,P,ET5,ET9,TPAF(190),VNU5,VNU9,VNU1,VNU8,TAU
      COMMON/CODE/PCTA(190),PCTF(190),WDCP0M(190),CK(190),RE(190),
     1TLSQU(190),TAUF(190),SIGFI1(190),SIGFI2(190),P2AV(190),NCORS,
     2AFLX, XFCRF,DL2,SAC,XNUF,XNUT(190),EPC
      COMMON/CORE2/MRCW,NCCL,NSYM,JMAX,H,D1,D2,HEIGHT,ALPHA,CCRIT,           ABP00100
     1CONCW,FIWATT,VOLMF,VOLLM,V(10), FCFRAC
      COMMON/TP/SAC1,SAC2,DENHOM,XMCOT,TIN,RATOR(10),XRAT,TOUT(30,10),
     1IPURR,TBAR
      COMMON/XSECT/XST(2,9),VNU(9),GAM15,GAMX5,GAM19,ALAMX,SAX,GAMP5,        ABP00120
     1GAMP9,VOELL                                                           ABP00130
      EQUIVALENCE (XST(1,1),SA5),(XST(1,2),SA6),(XST(1,3),SC8),             ABP00140
     1(XST(1,4),SA9),(XST(1,5),SA0),(XST(1,6),SA1),(XST(1,9),SABP),         ABP00150
     2(XST(2,1),SF5),(XST(2,4),SF9),(XST(2,6),SF1)                          ABP00160
     3,(XST(1,7),SPP),(XST(1,8),SAS)                                        ABP00170
    2 FORMAT(24F8F10.4)                                                     ABP00180
 33   FORMAT(I5.5X,G10.0/(8G10.0))                                           ABP00190
    3 FORMAT(8G10.0)                                                        ABP00200
    1 FORMAT(16I5)                                                          ABP00210
 9753 FORMAT(20A4)                                                          ABP00220
      READ(L,9753)ACMT                                                      ABP00230
      READ(L,1)NCYCLE,MODE,IXBURN
C ***    MODE = 0:  NORMAL
C             > 0:  PRINT ATOM DENSITIES
C             < 0:  PUNCH ATOM DENSITIES IN ADDITION
      DO 2588 I=1,2                                                         ABP00250
      DO 2588 J=1,9                                                         ABP00260
 2588 XST(I,J)=0.                                                           ABP00270
      READ(L,2)SA5,SA6,SA9,SA0,SA1,SC5,SC8,SC9,SC0,SF5,SF9,SF1,SPP,SAS,     ABP00280
     1SAX,SABP                                                              ABP00290
      READ(L,3)SASS,SAM                                                     ABP00300
      DO 7712 I=1,9                                                         ABP00310
 7712 VNU(I)=0.                                                             ABP00320
      READ(L,3)EPS,P1,P,ET5,ET9,GAMP5,GAMP9,GAMX5,GAM15,GAM19,ALAMX,VNU5    ABP00330
     1,VNU9,VNU1,VNU8,B2,TAU,DL2,VOLMF,VOLSF,DISMF,DISSF,ALAMP              ABP00340
      VNU(1)=VNU5                                                           ABP00350
      VNU(3)=VNU8                                                           ABP00360
      VNU(4)=VNU9                                                           ABP00370
      VNU(6)=VNU1                                                           ABP00380
      READ(L,3)PCF,PERIOD,FUELD,PELDX,FCFRAC
      READ(L,1)NROW,NCOL,NSYM,MKEY,JMAX                                     ABP00510
      READ(L,1) NRICH
      DO 2829 I=1,NRICH
      READ(L,3) (ENRICH(I,J), J=1,NROW)
      DO 100 J=1,NROW
      DUM=FUELD*AM/(1.+(32./235.)*(1.+3.*ENRICH(I,J)/235.))
      F25(I,J)=ENRICH(I,J)*DUM/235.
      F28(I,J)=DUM*(1.-ENRICH(I,J))/238.
  100 CONTINUE
 2829 CONTINUE
C ***    SOLUTION FORM ONLY OF B-10 ALLOWED                                  ABP00460
```

```
      READ(L,33)NDLP,DENH2O,(BPPM(I),I=1,NDLP)                          ABP00470
      CONS=VOLMF*DISMF*AC*0.1978E-6*DENH2O/10.                          ABP00480
      READ(L,5)XMDOT,TIN
      DO 2205 I=1,NDLP                                                  ABP00490
 2205 BPPM(I)=BPPM(I)*CONS                                              ABP00500
      READ(L,3)P,D1,D2,HIGHT,ALPHA,CCRIT,CONPOW,FIWATT                  ABP00520
      READ(L,1)NRAT
      IF(NRAT.NE.0)READ(L,3)RATOR
C        ABOVE IS EIGHT PER CARD
      WRITE(K,9752)ACMT                                                 ABP00530
      NC=IABS(NCYCLE)                                                   ABP00540
      WRITE(K,1071)NC                                                   ABP00550
      WRITE(K,2216)DENH2O,(BPPM(I),I=1,NDLP)                            ABP00560
      WRITE(K,2217)XMDOT,TIN
 2217 FORMAT(//5X,'CORE FLOW RATE =',E11.4,' #/HR'/5X,
     1'T-INLET =',F6.1,' DEG. F')
      DO 1212 I=1,NRICH
      WRITE(K,1200) I, (ENRICH(I,J), J=1,NROW)
 1200 FORMAT(/,5X,' ELEMENT TYPE  ', I4,'   ENRICHMENTS',/,20F6.3)
 1212 CONTINUE
      WRITE(K,1947)SA5,SA6,SA9,SA0,SA1,SC5,SC8                          ABP00570
      WRITE(K,1048)SC9,SC0,SF6,SF9,SF1,SPP,SAS                          ABP00580
      WRITE(K,1949)SAX,SA3P,SASS,S4M                                    ABP00590
      WRITE(K,1050)EPS,P1,P,ET5,ET9                                     ABP00600
      WRITE(K,1051)GAM25,GAMP9,GAMX5,GAMI5,GAMI9                        ABP00610
      WRITE(K,1052)ALAMX                                                ABP00620
      WRITE(K,1152)ALAMP                                                ABP00630
      WRITE(K,1053)VNU5,VNUP,VNU1,VNU3                                  ABP00640
      WRITE(K,1054)TAU,DL2,B2                                           ABP00650
      WRITE(K,1055)DISMF,DISSF                                          ABP00660
      WRITE(K,1056)VOLMF,VOLSF                                          ABP00670
      WRITE(K,942)NROW,NCOL,NSYM,MKEY,JMAX,H,D1,D2,HIGHT,ALPHA,CCRIT,   ABP00680
     1CONPOW,FIWATT                                                     ABP00690
  942 FORMAT(//5X,'ADDITIONAL PARAMETERS NEEDED FOR FLUX CORE CALCULATIOABP00700
     1N'//5X,'NO. OF ROWS =',I3,5X,'NO. OF COLUMNS =',I3,5X,'SYMMETRY FAABP00710
     2CTOR =',I3//5X,'FLUX PRINTOUT KEY =',I3,5X,'MAXIMUM NO. OF ITERATIABP00720
     3ONS =',I5//5X,'MESH SIZE =',F7.2,5X,'D1 =',F7.2,5X,'D2 =',F7.2,5X,ABP00730
     4'CORE HEIGHT =',F10.2,5X,'OVER-RELAXATION PARAMETER =',F7.2//5X,   ABP00740
     5'CONVERGENCE CRITERIA =',F8.5,5X,'CORE POWER =',E12.4,5X,'FISSIONSABP00750
     6 PER WATT =',E12.4)                                               ABP00760
 1071 FORMAT('-',4X,'*****   BURNUP PERIOD CYCLE NUMBER ',I2,3X,'*****')ABP00770
 2216 FORMAT(//5X,'BURNABLE POISON TABLE DURING CYCLE (PWR CASE)'/10X,   ABP00780
     1'WATER DENSITY =',F9.4/10X,'ATOM DENSITIES BASED UPON FUEL VOLUME'ABP00790
     2//(10X,8E14.4))                                                   ABP00800
 1047 FORMAT('-',8X,'CROSS SECTIONS IN CM**2'/T1,' ',T15,'SA25',T25,'SA2ABP00810
     16',T35,'SA49',T45,'SA40',T55,'SA41',T65,'SC25',T75,'SC28'/' ',10X,ABP00820
     27E10.3/)                                                          ABP00830
 1049 FORMAT(T1,' ',T15,'XE-135',T25,'SA3P'/' ',10X,2E10.3/'0',8X,      ABP00840
     1'CROSS SECTIONS IN CM**-1     STRUCTURE - CLAD =',F5.3,5X,'MODERATABP00850
     2OR =',F5.3)                                                       ABP00860
 1050 FORMAT(T1,'-', T7,'REACTOR PHYSICS PARAMETERS'/'0',8X,'FAST FISSIOABP00870
     1N FACTOR =',F7.4,5X,'FISSION-TO-RESONANCE NONLEAKAGE =',F7.4/' ',3XABP00880
     2X,'RESONANCE ESCAPE =',F7.4,5X,'ETA U-235 =',F7.4,5X,'ETA PU-239 =ABP00890
     3',F7.4)                                                           ABP00900
 1051 FORMAT('0',8X,'FRACTIONAL FISSION PRODUCT YIELDS (GAMMA)'/' ',10X,ABP00910
     1'PM-149 (U-235) =',F7.4,5X,'PM-149 (PU-239) =',F7.4/' ',10X,'XE-13ABP00920
     25 (U-235) =',F7.4/' ',10X,'I-135  (U-235) =',F7.4,5X,'I-135  (PU-2ABP00930
     339) =',F7.4)                                                      ABP00940
 1052 FORMAT('0',8X,'DECAY CONST. FOR XE-135 =',E10.2,'/SEC.')          ABP00950
 1152 FORMAT('0',8X,'DECAY CONST. FOR PM-149 =',E10.2,'/SEC.')          ABP00960
```

```
1053 FORMAT('0',8X,'AVERAGE NUMBER OF NEUTRONS RELEASED PER FISSON IN  ABP00970
     1U-235 =',F7.4/T1,' ',T57,'IN PU-239 =',T68,F7.4/T1,' ',T57,'IN PU-ABP00980
     2241 =',T68,F7.4/T1,' ',T57,'IN  U-238 =',T68,F7.4,T76,'(FAST)')   ABP00990
1054 FORMAT('0',8X,'FERMI AGE =',F5.1,' CM**2',5X,'THERMAL DIFFUSION LEABP01000
     1NGTH IN MODERATOR (**2) =',F5.1,' CM**2'/' ',10X,'BUCKLING =',E11.ABP01010
     23,'/CM**2')                                                       ABP01020
1055 FORMAT('0',8X,'DISADVANTAGE FACTOR IN MODERATOR =',F5.2,5X,'DISADVABP01030
     1ANTAGE FACTOR IN CLAD =',F5.2)                                    ABP01040
1056 FORMAT('-',6X,'PHYSICAL DESCRIPTION OF CORE'/'0',8X,'VOLUME RATIOSABP01050
     1 - MODERATOR/FUEL =',F5.2,5X,'CLAD/FUEL =',F5.2)                  ABP01060
      WRITE(K,8057)FUELD,PCF,PERIOD                                     ABP01070
8057 FORMAT('0',8X,'UO2 DENSITY =',F6.2,' GRAMS/CUBIC CM OF FUEL'/     ABP01080
     1       '0',8X,'THE PLANT CAPACITY FACTOR =',F5.3,' FOR A PERIOD OFABP01090
     2',F6.2,' YEARS')                                                  ABP01100
1048 FORMAT(T1,' ',T15,'SC49',T25,'SC40',T35,'SF25',T45,'SF49',T55,'SF4ABP01110
     11',T64,' SAFP ',T75,'SM-149'/' ',10X,7E10.3/)                     ABP01120
      NCORO=NCOL*NROW                                                   ABP01130
      DO 4000 I=1,NCORO                                                 ABP01140
      CK    (I)=0.                                                      ABP01150
      RE    (I)=0.                                                      ABP01160
      ELSOU (I)=0.                                                      ABP01170
      TAUF  (I)=0.                                                      ABP01180
      SIGFI1(I)=0.                                                      ABP01190
      SIGFI2(I)=0.                                                      ABP01200
      WOCROW(I)=0.
      P2AV  (I)=0.                                                      ABP01210
      XNUT  (I)=0.                                                      ABP01220
      TPAF  (I)=0.                                                      ABP01230
      DO 4000 J=1,10                                                    ABP01240
      SCB(J,I)=0.                                                       ABP01250
4000  SAB(J,I)=0.                                                       ABP01260
      AVE=0.                                                            ABP01270
      READ(L,5001)(MESH(I),I=1,NCORO)                                   ABP01280
5001  FORMAT((80I1))                                                    ABP01300
      DO 6050 I=1,NCORO
      IF (MESH(I) .EQ. 0) GO TO 6050
      IF (MESH(I) .EQ. 2) GO TO 6050
      READ(L,5002)(SAB(J,I),J=1,10)
6050  CONTINUE
      CALL XYTOR(SAB,F25,F28,NROW,NCOL,SR,SDD,NCYCLE)
      DO 6085 I=1,NCORO
      IF(SAB(1,I).NE.0.)GO TO 6086
6085  CONTINUE
      READ(L,5001)(MESH(I),I=1,NCORO)
      KI=0
      WRITE(K,6065)
6065  FORMAT( '1')
      DO 6062 I=1,NROW
      DO 6060 J=1,NCOL
      FEM(J)=0.0
      KI=KI+1
      IF(MESH(KI) .EQ. 0) GO TO 6060
      IF(MESH(KI) .EQ. 2) GO TO 6060
      LLL=I
      IF(NCYCLE .NE. 1) LLL=1
      FEM(J)=100.*SDD(1,J,LLL)/(SDD(1,J,LLL)+SDD(2,J,LLL)+SDD(3,J,LLL))
      DO 6061 JR=1,10
6061  SAB(JR,KI)=SDD(JR,J,LLL)
6060  CONTINUE
      WRITE(K,6063) (FEM(J), J=1,NCOL)
```

```
6063 FORMAT(//,10X,10F7.3)
6062 CONTINUE
     WRITE(K,6064)
6064 FORMAT(   10X,' RADIAL AVG FUEL ENRICHMENT MAP, IN %,
    1 COLUMN 9 HAS BEEN MODIFIED')
6065 DO 770 I=1,NCORD
     TPSUM(I)=MESH(I)
     IF(MESH(I).EQ.1)GO TO 770
     IF (MESH(I) .EQ. 3) GO TO 770
     DO 771 J=1,10
771  SAB(J,I)=0.
770  AVE=AVE+SAB(1,I)
     CALL MAP(NROW,NCOL,TPSUM,1)
6000 VOLT=PI*H*HIGHT                                                  ABP01540
     AFLX=CONPOW*FIWATT/(VOLT*AVE*SFS)                                ABP01550
     V(1)=PI*H*H*H=0.25
     VOLB=2.*PI*H*H
     VOLUM=V(1)
     DO 6110 I=2,NCOL
     V(I)=VOLB*I*H
     IF (I .EQ. 9) V(I)=V(I)*FCFRAC
6110 VOLUM=VOLUM+V(I)
     VOLUM=VOLUM*(NROW-4)
     SAC1=SASS*VOLSF*DISSF
     SAC2=SAM*VOLMF*DISMF
     SAC=SAC1+SAC2
     VCELL=1.+VOLSF+VOLMF                                             ABP01570
     NBN=PERIOD/PFLUX+1.0001                                          ABP01580
     DF1=1./(NBN-1)                                                   ABP01590
     DF2=1./(NDLF-1)                                                  ABP01600
     READ(L,1)N4                                                      ABP01610
     WRITE(K,699)N4                                                   ABP01620
699  FORMAT(//5X,'BORON ITERATIONS =',I2)                             ABP01630
     WRITE(K,87)MRAT
87   FORMAT(//5X,'MRAT =',I2)
     IF(MRAT.NE.0)WRITE(K,86)RATOR
86   FORMAT(//5X,'RATCR VALUES'/(10X,10F12.5))
     CALL LIQHCH(DEN,HW,TT,C)
     CPRE=2.*BPPM(1)
     DO 7020 I=1,NBN                                                  ABP01640
     IBURN=I
     MOB=0.0
     IF(MOD(I,2) .EQ. 0) MOB=2
     IF (I .EQ. NBN) MOB=0
     WRITE(K,8254)
8254 FORMAT('1',5X,'AXIAL BURNUP AVERAGES BY COLUMN POSITION'//T21,'J',
    1T30,'BURNUP'/)
     DO 8255 J=1,NCOL
     BN=0.
     BU=0.
     DO 8256 KJ=1,NROW
     KK=NCOL*(KJ-1)+J
     IF(SAB(10,KK).EQ.0.)GO TO 8256
     BN=BN+1.
     BU=BU+SAB(10,KK)
8256 CONTINUE
     IF(BN.EQ.0.)GO TO 8258
     BU=BU/BN
8258 WRITE(K,8257)J,BU
8257 FORMAT(T20,I2,T30,F7.0)
```

```
8255 CONTINUE
     CPPM=BPPM(1)                                                        ABP01650
     IF(I.EQ.1)GO TO 7037                                                ABP01660
     CPPM=BPPM(NDLF)                                                     ABP01670
     IF(I.EQ.NBN)GO TO 7037                                              ABP01680
     DO 7036 J=1,NDLF                                                    ABP01690
     N=J                                                                 ABP01700
     IF((J-1)*DF2+1.E-6.GE.(I-1)*DF1)GO TO 7039                          ABP01710
7036 CONTINUE                                                            ABP01720
7039 CPPM=BPPM(N)-(BPPM(N)-BPPM(N-1))*((N-1)*DF2-(I-1)*DF1)/DF2          ABP01730
7037 N4J=N4
     IF (MOD .EQ. 2) N4J=1
     DO 700 NJ=1,N4J
     IF(NJ.EQ.1)GO TO 701                                                ABP01750
     IF(ABS(XKEFF-1.).LT.0.0005)GO TO 702
     IF(NJ.GT.3)GO TO 703
     IF((I.GT.1).AND.(NJ.GT.2))GO TO 703
     CCLD=CPPM                                                           ABP01780
     SUM=0.
     DO 800 J=1,NCORO
800  SUM=SUM+RCTA(J)
     CPPM=CPPM+(XKEFF-1.)/(SUM*SABP*2.)
     IF((I.EQ.1).AND.(NJ.EQ.1)CPPM=CPPM*0.95
     SVK=XKEFF                                                           ABP01820
     GO TO 701                                                           ABP01830
703  DC=CPPM-CCLD                                                        ABP01840
     CCLD=CPPM                                                           ABP01850
     CPPM=CPPM-DC*(XKEFF-1.)/(XKEFF-SVK)                                 ABP01860
     SVK=XKEFF                                                           ABP01870
     IF(CPPM.GT.CPRE)CPPM=CPRE*0.95
701  DO 7038 J=1,NCORO                                                   ABP01880
7038 SAB(9,J)=CPPM                                                       ABP01890
     CPM =CPPM/CONB                                                      ABP01900
     WRITE(K,7040)I,CPM                                                  ABP01910
7040 FORMAT(//5X,'AT BURNUP INCREMENT NUMBER ',I2,', THE NATURAL BORON   ABP01920
    1CONCENTRATION IS ',F8.0,' PPM.')                                    ABP01930
     NTQ=0
     IF(I.GT.1)NTQ=1
89   CONTINUE
     CALL CPROP(SAB,MESH,IXEUSE)
     CONB=VOLMF*DISMF*AC*0.1978E-6*DENHCH/10.
     CALL FZFLUX(MOB, MKEY, KPRINT)
CCC  CALCULATE FIGURE OF MERIT, FOM
     AD=0.0
     FOM = 0.0
     DO 888 KJ=1,NCORO
     IF(MESH(KJ) .NE. 1) GO TO 888
     FOM=FOM+ABS(NORPOW(KJ)-1.0)
     AD=AD+1.0
 888 CONTINUE
     FOM=AD/FOM
     WRITE(K,887) FOM
 887 FORMAT(/,5X, ' FIGURE OF MERIT =',E14.6)
     IF(N4.GT.1)GO TO 700
     IF(NTQ.EQ.1)GO TO 700
     NTQ=1
     WRITE(K,88)
88   FORMAT(//5X,'REPEAT ZERO-TIME CALCULATION')
     GO TO 89
700  CONTINUE                                                            ABP01990
```

```
  702  CPPF=CPPM
       IF(I.NE.1)GO TO 710
       IF(N4.EQ.1)GO TO 710                                          ABP02010
       NDI=NDLF-1                                                    ABP02020
       DO 709 J=2,NDI                                                ABP02030
  709  BPPM(J)=BPPM(NDLF)+(BPPM(J)-BPPM(NDLF))*(CPPM-BPPM(NDLF))/    ABP02040
      1(BPPM(1)-BPPM(NDLF))                                          ABP02050
       BPPM(1)=CPPM                                                  ABP02060
  710  CONTINUE                                                      ABP02070
       CALL MAP(NROW,NCOL,WCCPOW,4)
       IF(I.EQ.NBN)GO TO 7022                                        ABP02080
       DO 7021 J=1,NCORC                                             ABP02090
       FLU=0.                                                        ABP02100
       IF(CK(J).EQ.0.)GO TO 7021                                     ABP02110
       FLU=P2AV(J)*PFLUX*3.1536E7*PCF*AFLX                           ABP02120
 7021  TRAP(J)=FLU                                                   ABP02130
 7023  CALL RLTAB(SAB ,NCORC)                                        ABP02150
 7022  CONTINUE                                                      ABP02160
       WRITE(K,7909)
 7909  FORMAT('1',5X,'AVERAGE CYCLE ORFICE RATIOS')
       NC2=NCOL-2
       DO 7910 I=1,NC2
       SUM=0.
       DO 7920 J=1,NBN
 7920  SUM=SUM+POUT(J,I)
 7910  RATOR(I)=(TSAR-TIN)/(SUM/NBN-TIN)
       WRITE(K,7908)(I,RATOR(I),I=1,NC2)
 7908  FORMAT(//(5X,I3,7X,F10.5))
       IF(MODE.EQ.0)STOP
       WRITE(K,7599)                                                 ABP02170
 7599  FORMAT(//5X,'ATOM DENSITIES BY LOCATION')                     ABP02180
       DO 7601 I=1,NCORC                                             ABP02190
       IF (MESH(I) .EQ. 0) GO TO 7601
       IF (MESH(I) .EQ. 2) GO TO 7601
       WRITE(K,7603)I                                                ABP02210
 7603  FORMAT(/5X,'LOCATION =',I3/)                                  ABP02220
       WRITE(K,7602)(SAB(J,I),J=1,10)                                ABP02230
       IF(MODE  .LT.0)WRITE(M,5002)(SAB(J,I),J=1,10)
 7601  CONTINUE                                                      ABP02250
 7602  FORMAT(5E20.8)                                                ABP02260
 5002  FORMAT((5X,5E15.8))
       STOP                                                          ABP02270
       END                                                           ABP02280
       SUBROUTINE XYTOR(SAZ,F25,F28,NROWZ,NCOLZ,SB,SDD,NCYCLE)
       DIMENSION SB(10,10),SC(10,10),SCB(10,100),SAB(10,100),MESH(100),
      1MESHN(100), F25(10,20), F28(10,20),SDD(10,10,20),XYMAP(100),
      2SAZ(10,190)
C         X-Y AVERAGING INTO DELTA-R RINGS.
       DIMENSION F(100,2),JC(10,20)
       DATA LI,LC,LP/5,6,7/
       DATA N,M/10,1/,MXY/0/
       EQUIVALENCE (NCORC,NA)
C          N X N AS QUARTER SYMMETRY
C               M=1, CELL CENTER; M=2, CELL BOUNDARIES
       WRITE(LC,11)N,M
  11   FORMAT(// ,5X,'ORDER =',I3,5X,'TYPE =',I2)
       NN=N*N
       DO 50 I=1,NN
       DO 50 J=1,2
  50   F(I,J)=0.
```

```
      NC=NN-N+1
      F(NC,1)=1.
      F(NC,2)=3.14159265/4.
      Y=0.
      HY=1.
      IF(M.EQ.1)HY=0.5
      DO 200 I=1,N
      IB=N-I+1
      X=0.
      HX=1.
      IF(M.EQ.1)HX=0.5
      DO 100 J=1,N
      KN=(IB-1)*N+J
      IF(F(KN,1).NE.0.)GO TO 99
      PLN=SQRT(X*X+Y*Y)
      R1=1.
      IF(M.EQ.1)R1=0.5
      DO 150 L=1,N
      R2=R1+1.
      IF(R2.GT.PLN)GO TO 151
  150 R1=R1+1.
      GO TO 99
  151 IR=L+1
      PUF=SQRT((X+HX)**2+(Y+HY)**2)
      IF(PUF.LE.R1)GO TO 99
      IF((PLN.GE.R1).AND.(PUF.GT.R2))GO TO 101
      FC=1.
      GO TO 98
  101 PUN=SQRT(X*X+(Y+HY)**2)
      PLF=SQRT((X+HX)**2+Y*Y)
      IF(PUN.GE.R2)GO TO 110
      IF(PLF.LE.R2)GO TO 120
C     CASE B FOLLOWS:
      ALPHA=SQRT(R2*R2-Y*Y)-X
      GAMMA=SQRT(R2*R2-(Y+HY)**2)-X
      FC       =0.5*(ALPHA+GAMMA)/HX
      GO TO 98
  110 IF(PLF.GE.R2)GO TO 111
C     CASE A' FOLLOWS:
      BETA=SQRT(R2*R2-X*X)-Y
      DELTA=SQRT(R2*R2-(X+HX)**2)-Y
      FC       =0.5*(BETA+DELTA)/HY
      GO TO 98
C     CASE A FOLLOWS:
  111 ALPHA=SQRT(R2*R2-Y*Y)-X
      BETA=SQRT(R2*R2-X*X)-Y
      FC       =0.5*ALPHA*BETA/(HX*HY)
      GO TO 98
C     CASE C FOLLOWS:
  120 GAMMA=SQRT(R2*R2-(Y+HY)**2)-X
      DELTA=SQRT(R2*R2-(X+HX)**2)-Y
      FC       =1.-0.5*(HX-GAMMA)*(HY-DELTA)/(HX*HY)
   98 F(KN,1)=IR
      F(KN,2)=FC
   99 HX=1.
      X=X+1.
      IF((Y.EQ.1).AND.(J.EQ.1))X=0.5
  100 CONTINUE
      HY=1.
      Y=Y+1.
```

```
      IF((M.EQ.1).AND.(I.EQ.1))Y=0.5
  200 CONTINUE
      DO 250 I=1,N
      DO 249 J=1,20
  249 JG(I,J)=0
      KN=1
      DO 251 J=1,NN
      IR=F(J,1)+.001
      IF(IR.NE.I)GO TO 251
      JG(I,KN)=J
      KN=KN+1
  251 CONTINUE
  250 CONTINUE
      IF(MXY.EQ.0)GO TO 409
      WRITE(LO,19)
   19 FORMAT(//5X,'ASSEMBLY FRACTIONS IN RADIAL ZONES'//T18,'POSITION',
     1T40,'ZONE',T62,'FRACTION'/)
      DO 410 I=1,NN
      IF(F(I,1).EQ.0.)GO TO 410
      J=F(I,1)+.001
      WRITE(LO,20)I,J,F(I,2)
  410 CONTINUE
   20 FORMAT(T20,I3,T40,I3,T60,F10.6)
  409 CONTINUE
      DO 699 I=1,NCORO
      DO 699 J=1,10
      SCB(J,I)=0.
  699 SAB(J,I)=0.
      READ(LI,5001)(MESH(I),I=1,NCORO)
C ***    FOR CYCLE ONE USE ZEROS FOR FUEL POSITIONS OF NEW FUEL
 5001 FORMAT((80I1))
      DO 5009 I=1,NCORC                                          ABP01340
      IF(MESH(I).NE.1)GO TO 5009                                 ABP01350
      READ(LI,5002)(SCB(J,I),J=1,10)
 5002 FORMAT((5X,5E15.8))                                        ABP01370
 5009 CONTINUE                                                   ABP01380
      READ(LI,8776)(MESHN(I),I=1,NCORO)
 8776 FORMAT((20I4))                                             ABP01400
      MCHK=0
      NZZ=NRCMZ
      IF(NCYCLE .NE. 1) NZZ=1
      DO 8760 L=1,NZZ
      DO 8750 I=1,NCORC                                          ABP01410
      XYMAP(I)=IABS(MESHN(I))
      SAB(3,I)=SCB(8,I)                                          ABP01420
      KI=MESHN(I)                                                ABP01430
      IF(KI)8749,8750,8748                                       ABP01440
 8749 SAB(1,I)=F25(-KI,L)
      SAB(3,I)= F28(-KI,L)
      MESH(I)=1
      GO TO 8750
 8748 DO 8744 J=1,10                                             ABP01480
      IF(J.EQ.8)GO TO 8744                                       ABP01490
      SAB(J,I)=SCB(J,KI)                                         ABP01500
      MCHK=1
 8744 CONTINUE                                                   ABP01510
 8750 CONTINUE                                                   ABP01530
      CALL ARAD(N, MESH,F,SAB,SB,JG)
      DO 2727 LMM=1,10
      DO 2727 LMN=1,10
```

```
      SDD(LMM,LMN,L)=SB(LMM,LMN)
 2727 CONTINUE
 8760 CONTINUE
      CALL MAP(N,N,XYMAP,8)
      WRITE(LO,710)
  710 FORMAT(//5X,'ABOVE MAP IS X-Y CORE POSITIONING FOR FORMING RADIAL
     1ZONES')
      IF(MOHK.EQ.0)RETURN
      NG=-N
      DO 700 I=1,NROWZ
      DO 701 J=1,N
      KI=(I-1)*N+J
      DO 701 KK=1,10
  701 SC(KK,J)=SAZ(KK,KI)
      CALL ARAD(NG,MESH,F,SCB,SC,JG)
      DO 709 IR=1,NCORC
      DO 709 JR=1,10
  709 SAB(JR,IR)=0.
      DO 711 IR=1,NCORO
      SAB(8,IR)=SCB(8,IR)
      KI=MESHN(IR)
      IF(KI)712,711,713
  712 SAB(1,IR)=F25(-KI,I)
      SAB(3,IR)=F28(-KI,I)
      GO TO 711
  713 DO 714 JR=1,10
      IF(JR.EQ.8)GO TO 714
      SAB(JR,IR)=SCB(JR,KI)
  714 CONTINUE
  711 CONTINUE
      CALL ARAD(N,MESH,F,SAB,SC,JG)
      DO 715 J=1,N
      KI=(I-1)*N+J
      DO 715 KK=1,10
  715 SAZ(KK,KI)=SC(KK,J)
  700 CONTINUE
      DO 720 I=1,NCOLZ
      SUM=0.
      SUMN=0.
      DO 721 J=1,NROWZ
      KI=(J-1)*N+I
      IF(SAZ(1,KI).EQ.0.)GO TO 721
      SUM=SUM+SAZ(1,KI)
      SUMN=SUMN+1.
  721 CONTINUE
      IF(SUMN.EQ.0.)SUMN=1.
      SUM=SUM/SUMN
  720 WRITE(LO,722)I,SUM,SB(1,I)
  722 FORMAT(//5X,'U-235 CHECK FOR COLUMN',I3,5X,'R-Z AVG =',E13.6,
     #5X,' X-Y AVG= ',E13.6,/,5X,' X-Y AVG NOT VALID FOR AXIAL VARIATIO
     2NS IN FUEL ENRICHMENT')
      WRITE(LO,730)
  730 FORMAT('1',5X,'U-235 CORE MAP BEFORE CORRECTION, COLUMN 9 IS MOD
     #IFIED WITH WATER'//)
      NR1=NROWZ+1
      DO 740 I=1,NR1
      K1=(I-1)*NCOLZ+1
      K2=I*NCOLZ
  740 WRITE(LO,731)(SAZ(1,J),J=K1,K2)
  731 FORMAT(/5X,10E12.3)
```

```
      RETURN
      END
      SUBROUTINE TMPR(DLT,SXC,MESH,DIN,DEN)
      DATA KKK/6/
      DIMENSION DLT(190),SXC(190),T(10),MESH(190)
      COMMON/CORE/RCTA(190),RCTF(190),WDCPOW(190),CK(190),RE(190),
     1ELSCU(190),TAUF(190),SIGFI1(190),SIGFI2(190),P2AV(190),NCORO,
     2AFLX, XKEFF,DL2,SAC,XNUF,XNUT(190),EPQ
      COMMON/CORE2/NROW,NCOL,NSYM,JMAX,H,D1,D2,HIGHT,ALPHA,CCRIT,
     1CONPOW,FIWATT,VOLMF,VOLUM,V(10), FCFRAC
      COMMON/TP/SAC1,SAC2,DENHOH,XMDOT,TIN,RATOR(10),MRAT,POUT(30,10),
     1IBURN,TBAR
      TOUT=0.
      VA=0.
      VO=0.
      TAVG=0.
      NC1=NCOL-1
      CALL LIQFCH(DIN,HIN,TIN,1)
      CALL LIQHCH(DIN,HIN,TIN,4)
      HAVG=CONPOW=3.413/XMDOT
      HG=HAVG+HIN
      CALL LIQHCH(DG,HG,TBAR,3)
      WRITE(KKK,27)TBAR
27    FORMAT(//5X,'OVERALL AVG. EXIT TEMP. =',F7.1)
      DO 100 I=1,NC1
      T(I)=TIN
      HZ=HIN
      CON=HAVG/(NROW-4)
      IF(MRAT.NE.0)CON=CON*RATOR(I)
      DO 200 J=1,NROW
      JP=NROW-J+1
      K=NCOL*(JP-1)+I
      DLT(K)=DL2
      SXC(K)=SAC
      IF (MESH(K) .EQ. 0) GO TO 200
      IF (MESH(K) .EQ. 2) GO TO 200
      HZ=HZ+CON*WDCPOW(K)
      CALL LIQHCH(DZ,HZ,TZ,2)
      DI=DZ/DIN
      DLT(K)=DL2/DI
      SXC(K)=SAC1+SAC2*DI
      CALL LIQHCH(DZ,HZ,T(I),3)
      TAVG=TAVG+T(I)*V(I)
      VA=VA+V(I)
200   CONTINUE
      POUT(IBURN,I)=T(I)
      IF(I.EQ.NC1)GO TO 100
      TOUT=TOUT+T(I)*V(I)
      VO=VO+V(I)
100   CONTINUE
      WRITE(KKK,25)(T(I),I=1,NC1)
25    FORMAT(//  5X,'OUTLET TEMPERATURES'//(10F11.1))
      TOUT=TOUT/VO
      WRITE(KKK,28)TOUT
28    FORMAT(//5X,'AVERAGE EXIT TEMPERATURE =',F7.1)
      TAVG=TAVG/VA
      CALL LIQHCH(DEN,HA,TAVG,4)
      DENHOH=DEN/62.4
      WRITE(KKK,26)DENHOH,TAVG
26    FORMAT(//5X,'AVERAGE CORE WATER SPECIFIC GRAVITY =',F7.4/
```

```
     15X,'AVERAGE CORE WATER TEMPERATURE =',F7.1)
      WRITE(KKK,600)
  600 FORMAT(///5X,'CALCULATED ORFICE RATIOS (IN TO OUT)'/)
      DO 601 I=1,NC1
      RT=0.
      IF(ABS(T(I)-TIN).LT.0.1  )GO TO 601
      RT=(TBAR-TIN)/(T(I)-TIN)
  601 WRITE(KKK,602)I,RT
  602 FORMAT(10X,I3,5X,F8.4)
      RETURN
      END
      SUBROUTINE RZFLUX(MODE,MKEY,MPROP)
C     ********************  2-D, R-Z, 2 GROUP CORE PROGRAM  *********
      REAL*4 LOCPOW
      DATA KKK/6/                                                        ABP02320
      DIMENSION P1(190),P2(190),OLD(190),P1A(190),P2A(190)
      COMMON/COPE2/NROW,NCOL,NSYM,JMAX,H,D1,D2,HIGHT,ALMI, CCRIT,
     1CONPOW,FIWATT,VOLME,VOLUM,V(10),FCFRAC
      COMMON/CORE/RCTA(190),RCTF(190),LOCPOW(190), C(190), P(190),
     1ELSQU(190), TAU(190),SIGFI1(190),SIGFI2(190), P2U(190),NCORO,
     2P2FAG,XKEFF,DL2,SAC,XNUF,XNUT(190),EPQ
      EQUIVALENCE (LOCPOW(1),OLD(1))                                     ABP02390
C                                                                        ABP02400
C        CODE IS:  C IS K-INF.                                           ABP02410
C                  P IS RESONANCE ESCAPE PROBABILITY.                    ABP02420
C                  ELSQU IS DIFFUSION LENGTH SQUARED.                    ABP02430
C                  TAU IS TAU THERMAL.                                   ABP02440
C                  SIGFI-J IS SIGMA FISSION FOR FAST, J=1, AND THERMAL,  ABP02450
C                  J=2, GROUPS.                                          ABP02460
C        R-Z GEOMETRY WITH DELTA-R = DELTA-Z = H
C                                                                        ABP02470
      TOD=4.0
      IF (MODE .EQ. 2) GO TO 1126
      HS=H*H                                                             ABP02490
      IF(MPROP.NE.0)CALL MAP(NROW,NCOL,C,7)                              ABP02510
      WRITE(KKK,861)                                                     ABP02520
  861 FORMAT(///5X,'FLUX CALCULATION'///)                                ABP02530
      DO 49 I=1,NCORO                                                    ABP02540
      OLD(I)=0.                                                          ABP02550
      SET=1.0                                                            ABP02560
      IF(P(I).EQ.0.)SET=0.                                               ABP02570
      P1(I) =SET                                                         ABP02580
      P2(I) =SET                                                         ABP02590
      P1A(I)=SET                                                         ABP02600
   49 P2A(I)=SET                                                         ABP02610
C ***    NSYM IS SYMMETRY KEY                                            ABP02620
C        NSYM = 0: NO SYMMETRY
C             = 1,2: MAP IS UPPER CORE SO SYMMETRY ABOUT Z=0
C        IF NSYM = 1, SYMMETRY IS THROUGH CELL BOUNDARY
C                = 2, SYMMETRY IS THROUGH CELL CENTER OR MESH POINT
      ZNORM=1.0                                                          ABP02700
      SUM1=0.                                                            ABP02710
      SUM2=0.                                                            ABP02720
      NOO=NSYM+1                                                         ABP02730
      ALPHA=1.0                                                          ABP02740
      XKEFF=1.0                                                          ABP02750
      ERR=0.01                                                           ABP02760
      ALT=1.0                                                            ABP02770
      DO 123 J=1,JMAX                                                    ABP02780
      ANUM=0.                                                            ABP02790
```

```
      ERAV=0.                                                           ABP02800
      NIT=J                                                             ABP02810
      NK=0                                                              ABP02820
      ERMAX=0.                                                          ABP02830
      NA=NCCL+1                                                         ABP02840
      NB=NCORD-NCOL                                                     ABP02850
      IF(NSYM.NE.0)NB=NCORD-1                                           ABP02860
C     SWEEP THE DIFFUSION EQUATIONS                                     ABP02870
      DO 113 I=NA,NB                                                    ABP02880
      IF(P(I).EQ.0.)GO TO 113                                           ABP02890
      PP=D1*HS*P(I)/(D2*TAU(I))                                         ABP02900
      S=1.0/(TCD+HS/TAU(I))                                             ABP02910
      TC=1.0/(TCD+HS/ELSQU(I))                                          ABP02920
      CC=D2*HS*C(I)*ZNORM/(D1*ELSQU(I)*P(I))                            ABP02930
      IP=I+NCCL                                                         ABP02940
      IM=I-NCOL                                                         ABP02950
      IM1=I-1                                                           ABP02960
      IP1=I+1
      GO TO (108,609,608),NQQ
  609 IF(IP.GT.NCORD)IP=I                                               ABP02990
      GO TO 108                                                         ABP03000
  608 IF(IP.GT.NCORD)IP=IM                                              ABP03020
  108 IDR=NCC(I-1,NCOL)
      IF(IDR.EQ.0)GO TO 1080
      CIP1=1.+1./(2*IDR)
      CIM1=1.-1./(2*IDR)
      GO TO 1081
 1080 CIP1=2.
      CIM1=0.
 1081 CONTINUE
      P1(I)=((CC*P2(I)+CIP1*P1(IP1)+CIM1*P1(IM1)+P1(IP)+P1(IM))*S-P1(I))
     1*ALPHA+P1(I)                                                      ABP03040
      P2(I)=((PP*P1(I)+CIP1*P2(IP1)+CIM1*P2(IM1)+P2(IP)+P2(IM))*TC
     1-P2(I))*ALPHA+P2(I)
C         ADJOINT CALCULATIONS FOLLOW.                                  ABP03070
      P1A(I)=((PP*P2A(I)+CIP1*P1A(IP1)+CIM1*P1A(IM1)+P1A(IP)+P1A(IM))*S
     1-P1A(I))*ALPHA+P1A(I)
      P2A(I)=((CC*P1A(I)+CIP1*P2A(IP1)+CIM1*P2A(IM1)+P2A(IP)+P2A(IM))*TC
     1-P2A(I))*ALPHA+P2A(I)
      IF(SIGF2(I).EQ.0.)GO TO 113                                       ABP03120
      ERR=ABS(P2(I)-OLD(I))/P2(I)                                       ABP03130
      IF(ERR.GT.CCRIT)NK=NK+1                                           ABP03140
      ERMAX=AMAX1(ERMAX,ERR)                                            ABP03150
      ANUM=ANUM+1.                                                      ABP03160
      ERAV=ERAV+ERR                                                     ABP03170
      OLD(I)=P2(I)                                                      ABP03180
  113 CONTINUE                                                          ABP03190
      IF(ALPHA.GT.1.)ERM=0.005                                          ABP03200
      IF(J.GE.1)GO TO 115                                               ABP03210
  114 IF(ABS(ERAV/ESV).GT.1.)GO TO 115                                  ABP03220
      ALTN=2.0/(1.+SQRT(1.-(ERAV/ESV)**2))                              ABP03230
      IF(ABS(1.-ALTN/ALT).LT.ERM) ALPHA=AMIN1(ALMI,ALTN)                ABP03240
      ALT=ALTN                                                          ABP03250
  115 ESV=ERAV                                                          ABP03260
      ERAV=ERAV/ANUM                                                    ABP03270
      IF(NK.EQ.0)GO TO 125                                              ABP03280
      IF(MOD(J,60).EQ.0)ALPHA=ALPHA-0.1                                 ABP03290
      ALPHA=AMAX1(ALPHA,1.0)                                            ABP03300
  119 IF(J- 1) 123,120,1120                                             ABP03310
  120 DO 121 I=1,NCORD                                                  ABP03320
```

```
      SUM1=SUM1+P1(I)+P2(I)                                              ABP03330
  121 CONTINUE                                                           ABP03340
      GO TO 123                                                          ABP03350
 1120 DO 1121 I=1,NCORE                                                  ABP03360
      SUM2=SUM2+P1(I)+P2(I)                                              ABP03370
 1121 CONTINUE                                                           ABP03380
      ZNORM=ZNORM*SUM1/SUM2                                              ABP03390
      XKEFF=XKEFF*SUM2/SUM1                                              ABP03400
      SUM1=SUM2                                                          ABP03410
      SUM2=0.                                                            ABP03420
      IF(MOD(J,10).EQ.0)WRITE(KKK,2350)J,ERMAX,ERAV,ALPHA                ABP03430
 2350 FORMAT(5X,'ITERATION NO. =',I3,5X,'MAXIMUM ERROR =',E11.4,5X,      ABP03440
     1'AVERAGE ERROR =',E11.4,5X,'ALPHA =',F9.6)                         ABP03450
      IF(XKEFF.GT.0.)GO TO 123                                           ABP03460
      WRITE(KKK,5805)XKEFF,J                                             ABP03470
 5805 FORMAT(//5X,7HK-EFF =,F11.6,5X,3HJ =,I4,5X,9HTERMINATE  )          ABP03480
      STOP                                                               ABP03490
  123 CONTINUE                                                           ABP03500
  125 WRITE(KKK,127)JMAX,NIT,ERMAX,CCRIT                                 ABP03510
  127 FORMAT(// ,5X,28HMAXIMUM NUMBER OF ITERATIONS ,I4,18H, ITERATIONS  ABP03520
     1USED  ,I4,19H TO RELATIVE ERROR  ,E10.4,21H. ERROR CRITERIA IS  ,  ABP03530
     2E10.4)                                                             ABP03540
C                                                                        ABP03550
C     CALCULATE POWER PRODUCING VOLUME                                   ABP03560
C                                                                        ABP03570
 1126 POWR=0.
      P1FAG=0.
      P2FAG=0.
      VOLUM=0.0                                                          ABP03620
      DO 1110 I=1,NCORE                                                  ABP03630
      LOCPOW(I)=0.
      IF(SIGFI2(I))  1110,1110,1221                                      ABP03640
 1221 IDR=MOD(I,NCOL)
      VOL=V(IDR)
      GO TO (1220,1220,1232),NQQ
 1232 IF(I.GT.NCORE-NCOL)VOL=VOL*0.5
 1220 VOLUM=VOLUM + VOL
      LOCPOW(I)=(SIGFI1(I)*P1(I)+SIGFI2(I)*P2(I))*VOL
      POWR=POWR+LOCPOW(I)
      P1FAG=P1FAG+P1(I)*VOL
      P2FAG=P2FAG+P2(I)*VOL
      P2U(I)=VOL
 1110 CONTINUE                                                           ABP03730
      APD=CONPOW/VOLUM                                                   ABP03740
      WRITE(KKK,1111)VOLUM,APD                                           ABP03760
 1111 FORMAT(//5X,8HVOLUME =E11.4,5X,14HPOWER/VOLUME =E11.4//)           ABP03770
      POWER=POWR/FIWATT
      ENORM = CONPOW/POWER                                               ABP03820
      P1FAG=P1FAG*ENORM/VOLUM
      P2FAG=P2FAG*ENORM/VOLUM
C                                                                        ABP03790
C      FLUX NORMALIZATION TO CONSTANT POWER                              ABP03800
C                                                                        ABP03810
      DO 131 I=1,NCORE                                                   ABP03830
      P1(I) = P1(I)*ENORM                                                ABP03840
      P2(I) = P2(I) * ENORM                                              ABP03850
      P1A(I)=P1A(I)*ENORM                                                ABP03860
      P2A(I)=P2A(I)*ENORM                                                ABP03870
      IF(LOCPOW(I).EQ.0.)GO TO 131
      J=MOD(I,NCOL)
```

```
      LOCPOW(I)=LOCPOW(I)*VOLUM/(POWR*V(J))
  131 CONTINUE                                                          ABP03880
C                                                                       ABP03890
C     CALCULATE POWER AT MESH  POINTS                                   ABP03900
C                                                                       ABP03910
      SMAX=0.                                                           ABP03950
      DO 2160 I=1,NCORC                                                 ABP03960
      IF(LOCPOW(I).LT.SMAX)GO TO 2160                                   ABP04070
      SMAX=LOCPOW(I)                                                    ABP04080
      KPOW=I                                                            ABP04090
 2160 CONTINUE                                                          ABP04100
      J=MOD(KPOW,NCOL)                                                  ABP04150
      JR=KPOW/NCOL                                                      ABP04160
      IF(J.NE.0)JR=JR+1                                                 ABP04170
      IF(J.EQ.0)J=NCOL                                                  ABP04180
      WRITE(KKK,2161)JR,J,SMAX                                          ABP04190
 2161 FORMAT(//5X,4HROW ,I2,9H, COLUMN ,I2,46H, IS POSITION OF MAXIMUM LABP04200
     1OCAL POWER OF VALUE ,F8.5//)                                      ABP04210
      WRITE(KKK,2133)P1FAG,P2FAG                                        ABP04240
 2133 FORMAT(//5X,'AVERAGE FAST FLUX =',E12.5,10X,'AVERAGE THERMAL FLUX ABP04250
     1=',E12.5)                                                         ABP04260
      WRITE(KKK,2805)XKEFF                                              ABP04270
 2805 FORMAT(//2X,'HK-EFF =,F11.6)                                      ABP04280
      IF(MODE.NE.1)GO TO 2990                                           ABP04290
C        MAP KEY PRINTOUT                                               ABP04300
C            MKEY=0, LOCPOW PRINTOUT.                                   ABP04310
C            MKEY=1, P1 PRINTOUT.                                       ABP04320
C            MKEY=2, P2 PRINTOUT.                                       ABP04330
C            MKEY=3, P1 & P2 PRINTOUT.                                  ABP04340
C            MKEY=4, P1 & P2 & LOCPOW PRINTOUT.                         ABP04350
C            MKEY<0, PUNCH P1 & P2.                                     ABP04360
      IF(MKEY.NE.0)GO TO 1179                                           ABP04370
      CALL MAP(NROW,NCOL,LOCPOW,4)                                      ABP04380
      GO TO 2990                                                        ABP04390
 1179 IF (IABS(MKEY).NE.2)CALL MAP(NROW,NCOL,P1,3)                      ABP04400
      IF (IABS(MKEY).GE.2)CALL MAP(NROW,NCOL,P2,2)                      ABP04410
      IF (IABS(MKEY).EQ.4)CALL MAP(NROW,NCOL,LOCPOW,4)                  ABP04420
      IF (IABS(MKEY).NE.2)CALL MAP(NROW,NCOL,P1A,-3)                    ABP04430
      IF (IABS(MKEY).GE.2)CALL MAP(NROW,NCOL,P2A,-2)                    ABP04440
 2990 SUMD=0.                                                           ABP04450
C ***    SYMMETRY CONDITIONS ON MESH DO NOT APPLY TO PERTURBATION       ABP04460
C        COEFFICIENTS SINCE THEORY REQUIRES A FULL CORE WITH VACUUM     ABP04470
C ***    BOUNDARY CONDITIONS.                                           ABP04480
      DO 3001 I=1,NCORC                                                 ABP04490
      P2U(I)=P2(I)/P2FAG                                                ABP04500
      IF(P(I).EQ.0.)GO TO 3001                                          ABP04510
      J=MOD(I,NCOL)
      SUMD=SUMD+(XNUF*SIGFI1(I)*P1A(I)*P1(I)+XNUT(I)*SIGFI2(I)*P1A(I)*
     1 P2(I))*V(J)
 3001 CONTINUE                                                          ABP04540
      DO 3002 I=1,NCORC                                                 ABP04550
      RCTF(I)=0.                                                        ABP04560
      RCTA(I)=0.                                                        ABP04570
      IF(P(I).EQ.0.)GO TO 3002                                          ABP04580
      J=MOD(I,NCOL)
      RCTA(I)=(P2A(I)*P2(I)*V(J)/SUMD)*XKEFF*XKEFF
      RCTF(I)=((EPS-1.)*P1(I)+P2(I))*P1A(I)*XNUT(I)*XKEFF*V(J)/SUMD
 3002 CONTINUE                                                          ABP04610
      IF(MKEY.NE.0)CALL MAP(NROW,NCOL,RCTA,5)                           ABP04620
      IF(MKEY.NE.0)CALL MAP(NROW,NCOL,RCTF,6)                           ABP04630
```

```
85    RETURN                                                                  ABP04640
      END                                                                     ABP04650
      SUBROUTINE NUDAB(SAB,NCORD)
      DIMENSION SAB(10,190)
      COMMON/DABA/SPS,PI,P,ET5,ET9,TPAB(190),VNU5,VNU9,VNU1,VNU8,TAU
      COMMON/XSECT/XST(2,9),VNU(9),GAMI5,GAMX5,GAMI9,ALAMX,SAX,GAMP5,     ABP05570
     1GAMP9,NCELL                                                             ABP05580
      EQUIVALENCE (XST(1,1),SA5),(XST(1,2),SA6),(XST(1,3),SC8),              ABP05590
     1(XST(1,4),SA9),(XST(1,5),SA0),(XST(1,6),SA1),(XST(1,9),SABP),          ABP05600
     2(XST(2,1),SF5),(XST(2,4),SF9),(XST(2,6),SF1)                           ABP05610
      SC5=SA5-SF5                                                             ABP05620
      SC9=SA9-SF9                                                             ABP05630
      SC0=SA0                                                                 ABP05640
      C5PS=SPS*PI*(1.-P)*SA5*ET5                                              ABP05650
      C9P=(SPS*PI*(1.-P)*SA9*ET9)-SA9                                         ABP05660
      C8S=-SC8/C9P                                                            ABP05670
      C6S=1./(SA6+C9P)                                                        ABP05680
      C10S=SC9/SA0                                                            ABP05690
      C12S=SC9/(SA6-SA0)                                                      ABP05700
      C14S=SC0/SA1                                                            ABP05710
      C15S=SC0/(C9P+SA1)                                                      ABP05720
      C16S=SC0/(SA1-SA5)                                                      ABP05730
      C17S=SC0/(SA1-SA0)                                                      ABP05740
      C22S=(-2.*SF1)/SA0                                                      ABP05750
      C23S=(-2.*SF1)/SA1                                                      ABP05760
      DO 200 I=1,NCORD                                                        ABP05770
      IF(TPAB(I).EQ.0.)GO TO 200                                              ABP05780
      T=TPAB(I)                                                               ABP05790
      A5I=SAB(1,I)                                                            ABP05800
      A6I=SAB(2,I)                                                            ABP05810
      A8I=SAB(3,I)                                                            ABP05820
      A9I=SAB(4,I)                                                            ABP05830
      A0I=SAB(5,I)                                                            ABP05840
      A1I=SAB(6,I)                                                            ABP05850
      AFI=SAB(7,I)                                                            ABP05860
      API=SAB(9,I)                                                            ABP05870
      SAB(9,I) =API*EXP(-SA8P*T)                                              ABP05880
      SAB(1,I) =A5I*EXP(-SA5*T)                                               ABP05890
      SAB(2,I) =((A5I*SC5)/(SA6-SA5))*(EXP(-SA5*T)-EXP(-SA6*T))+A6I*EXP(      ABP05900
     1-SA6*T)                                                                 ABP05910
      C5P=C5PS*A5I                                                            ABP05920
      C8=C8S*A8I                                                              ABP05930
      C6=C6S*C5P                                                              ABP05940
      SAB(4,I) =C8*(1.-EXP(C9P*T))+C6*(EXP(C9P*T)-EXP(-SA5*T))+A9I*EXP(CABP05950
     19P*T)                                                                   ABP05960
      C10=C10S*C8                                                             ABP05970
      C11=(SC9*(C6+A9I-C8))/(C9P+SA0)                                         ABP05980
      C12=C12S*C6                                                             ABP05990
      C13=A0I-(C10+C11+C12)                                                   ABP06000
      SAB(5,I) =C10+C11*EXP(C9P*T)+C12*EXP(-SA5*T)+C13*EXP(-SA0*T)            ABP06010
      C14=C14S*C10                                                            ABP06020
      C15=C15S*C11                                                            ABP06030
      C16=C16S*C12                                                            ABP06040
      C17=C17S*C13                                                            ABP06050
      C18=A1I-(C14+C15+C16+C17)                                               ABP06060
      SAB(6,I) =C14+C15*EXP(C9P*T)+C16*EXP(-SA5*T)+C17*EXP(-SA0*T)+C18*EABP06070
     1XP(-SA1*T)                                                              ABP06080
      C22=C22S*C17                                                            ABP06090
      C23=C23S*C18                                                            ABP06100
      C25=1.0-EXP(-SA5*T)                                                     ABP06110
```

```
      F5=2.(*A5I*SF5*C25/SA5                                                    ABP06120
      F9=2.0*SF9*(C8*T-C6*C25/SA5+(C8-C6-A9I)*(1.-EXP(C9P*T))/C9P)              ABP06130
      F1=2.0*SF1*(C14*T+C16*C25/SA5)-C23*(1.-EXP(-SA1*T))     -2.*SF1*C15ABP06140
     1*(1.-EXP(C9P*T))/C9P-C22*(1.-EXP(-SA0*T))                                 ABP04150
      C19=(EPS-1.)/(VNU8-1.)                                                    ABP06160
      F8=C19*(VNU5*F5+VNU9*F9+VNU1*F1)                                          ABP06170
      SA8(7,I) =F8+F5+F9+F1+AFI                                                 ABP06180
      SA8(8,I) =A8I*(1.-SC8*T)-0.5*(VNU5*EPS*P1*(1.-P)*F5+VNU9*EPS*P1*(1ABP06190
     1.-P)*F9+F8)                                                               ABP06200
      IF(AFI.NE.0.)GO TO 160                                                    ABP06210
      IF(A5I+A8I.EQ.0.)GO TO 200                                                ABP06220
      SA8(10,I)=(SA8(7,I)/(2.*(A5I+A8I)))*9.E5                                  ABP06230
      GO TO 200                                                                 ABP06240
  160 SA8(10,I)=SA8(10,I)*SA8(7,I)/AFI                                          ABP06250
  200 CONTINUE                                                                  ABP06260
      RETURN                                                                    ABP06270
      END                                                                       ABP06280
      SUBROUTINE MAP(NUM1,NUM2,VAR,NCA)
      DIMENSION VAR(190),KVAR(190),XAR(190)
      DATA KKK/6/                                                               ABP04680
      ND=IABS(NCA)                                                              ABP04690
      WRITE(KKK,171)                                                            ABP04700
  171 FORMAT(1H1)                                                               ABP04710
  172 FORMAT(1H0,I3,29(I4))                                                     ABP04720
  174 FORMAT (14HJGEOMETRY  MAP)                                                ABP04730
  175 FORMAT (22HOTHERMAL FLUX MAP      ,5HNORM:,E14.6)                         ABP04740
  176 FORMAT (22HOFAST FLUX MAP         ,5HNORM:,E14.6)                         ABP04750
  177 FORMAT(22HOLOCAL POWER MAP        ,5HNORM:,E14.6)                         ABP04760
  178 FORMAT(55HOREACTIVITY PERTURBATION MAP FOR UNIT SIGMA-A2 CHANGE           ABP04770
     1/24HNOTE NORM IS NEGATIVE    ,5X,5HNORM:,E15.6)                           ABP04780
  179 FORMAT(55HOREACTIVITY PERTURBATION MAP FOR UNIT SIGMA-F2 CHANGE           ABP04790
     1/6HNORM: ,E14.6)                                                          ABP04800
 1790 FORMAT(22HOK-INF MAP              ,5HNORM:,E15.6)                         ABP04810
 1791 FORMAT(12HOSHUFFLE MAP)                                                   ABP04820
      IF(NUM2.GT.15)GO TO 20                                                    ABP04830
   10 NUM21= NUM2                                                               ABP04840
      NUM22=0                                                                   ABP04850
      GOTO   30                                                                 ABP04860
   20 NUM21= NUM2/2 +1                                                          ABP04870
      NUM22 =NUM2- NUM21                                                        ABP04880
   30 CONTINUE                                                                  ABP04890
      NB=NUM1*NUM2                                                              ABP04900
      IF(ND.EQ.1)GO TO 2000                                                     ABP04910
      IF(ND.EQ.8)GO TO 2000                                                     ABP04920
      XNORM=1.0                                                                 ABP04930
      BV=0.                                                                     ABP04940
      DO 184 I=1,NB                                                             ABP04950
      BV=AMAX1(BV,VAR(I))                                                       ABP04960
  184 CONTINUE                                                                  ABP04970
      IF(BV.GE.10.)GO TO 188                                                    ABP04980
      IF(BV.GT.1.)GO TO 189                                                     ABP04990
  188 XNORM=10./BV                                                              ABP05000
  189 DO 190 I=1,NB                                                             ABP05010
  190 XAR(I)=VAR(I)*XNORM                                                       ABP05020
 2000 J=1                                                                       ABP05030
      L=NUM21                                                                   ABP05040
      KON=0                                                                     ABP05050
      NUMM=NUM21                                                                ABP05060
  191 DO 193 I=1,NUM1                                                           ABP05070
      IF(ND.EQ.1)GO TO 195                                                      ABP05080
```

```
      IF(ND.EQ.8)GO TO 195                                              ABP05090
      DO 1912 K=J,L                                                     ABP05100
      IF(XAR(K).LT.0.)XAR(K)=0.                                         ABP05110
      IF(XAR(K).NE.0.)XAR(K)=XAR(K)+0.0005                              ABP05120
 1912 CONTINUE                                                          ABP05130
      WRITE(KKK,1913)(XAR(K),K=J,L)                                     ABP05140
 1913 FORMAT(//3X,15F7.3)                                               ABP05150
      GO TO 192                                                         ABP05160
  195 DO 196 K=J,L                                                      ABP05170
  196 KVAR(K)=XAR(K)+0.1                                                ABP05180
      WRITE(KKK,197)(KVAR(K),K=J,L)                                     ABP05190
  197 FORMAT(/3X,15I4)                                                  ABP05200
  192 J=J+NUM2                                                          ABP05210
      L=J+NUMM-1                                                        ABP05220
  193 CONTINUE                                                          ABP05230
      KCN= KCN +1                                                       ABP05240
      IF(NUM22)  2950 ,2950, 2910                                       ABP05250
 2910 IF (KCN-2)   2920, 2950 ,2950                                     ABP05260
 2920 J=NUM21+1                                                         ABP05270
      L=NUM2                                                            ABP05280
      NUMM= NUM22                                                       ABP05290
      WRITE(KKK,171)                                                    ABP05300
      GO TO 191                                                         ABP05310
 2950 CONTINUE                                                          ABP05320
      IF(NDA.LT.0)WRITE(KKK,170)                                        ABP05330
  170 FORMAT(17HOADJOINT SOLUTION)                                      ABP05340
      GO TO (1960,1962,1964,1966,1967,1968,1969,1970),ND                ABP05350
 1960 WRITE(KKK,174)                                                    ABP05360
      RETURN                                                            ABP05370
 1962 WRITE(KKK,175)XNORM                                               ABP05380
      RETURN                                                            ABP05390
 1964 WRITE(KKK,176)XNORM                                               ABP05400
      RETURN                                                            ABP05410
 1966 WRITE(KKK,177)XNORM                                               ABP05420
      RETURN                                                            ABP05430
 1967 XNORM=-XNORM                                                      ABP05440
      WRITE(KKK,178)XNORM                                               ABP05450
      RETURN                                                            ABP05460
 1968 WRITE(KKK,179)XNORM                                               ABP05470
      RETURN                                                            ABP05480
 1969 WRITE(KKK,1790)XNORM                                              ABP05490
      RETURN                                                            ABP05500
 1970 WRITE(KKK,1791)                                                   ABP05510
      RETURN                                                            ABP05520
      END                                                               ABP05530
      SUBROUTINE LIQH2O(D,H,T,MT)
      DATA L,K,N/5,6,11/
      DIMENSION P(3,11),G(3)
      IF(MT.GT.0)GO TO 100
      READ(L,10)G,((P(J,I),J=1,3),I=1,N)
   10 FORMAT(3A4/(3F10.0))
C        J=1, TEMPERATURE; J=2, SPECIFIC VOLUME; J=3, ENTHALPY.
      WRITE(K,11)G,((P(J,I),J=1,3),I=1,N)
   11 FORMAT(//5X,'WATER THERMODYNAMIC DATA'/T8,3A4,T40,'TEMPERATURE',
     1T60,'SPECIFIC VOLUME',T80,'ENTHALPY'//(T43,F5.0,T63,F8.5,T82,F5.1)
     2)
      RETURN
C        MT=1, T-IN, H-OUT
C        MT=2, H-IN, D-OUT
C        MT=3, H-IN, T-OUT
```

```
C        MT=4, T-IN, D-OUT
  100  GO TO (101,102,103,104),MT
  101  J1=1
       J2=3
       PX=T
       GO TO 110
  102  J1=3
       J2=2
       PX=H
       GO TO 110
  103  J1=3
       J2=1
       PX=H
       GO TO 110
  104  J1=1
       J2=2
       PX=T
  110  DO 111 I=1,N
       IF(PX.LE.P(J1,I))GO TO 112
  111  CONTINUE
       WRITE(K,113)PX,P(J1,N),MT
  113  FORMAT(/5X,'EXTRAPOLATION, VALUE =',E11.4,2X,'LAST TABLE VALUE =',
      1E11.4,2X,'TYPE =',I2)
  112  I=I-1
       IF(I.GE.N-1)I=N-2
       IF(I.EQ.0)I=1
       F1=(P(J2,I+1)-P(J2,I))/(P(J1,I+1)-P(J1,I))
       F12=(P(J2,I+2)-P(J2,I+1))/(P(J1,I+2)-P(J1,I+1))
       F2=(F12-F1)/(P(J1,I+2)-P(J1,I))
       A=P(J2,I)+(PX-P(J1,I))*(F1+(PX-P(J1,I+1))*F2)
       GO TO (201,202,203,202),MT
  201  H=A
       GO TO 210
  202  G=1./A
       GO TO 210
  203  T=A
  210  RETURN
       END
       SUBROUTINE ARAD(N, MESH,F,SAB,SR,JG)
       DIMENSION MESH(100),F(100,2),SAB(10,100),SR(10,10),SN(10,10)
       DIMENSION JG(10,20)
       NN=N*N
       IF(N.LT.0)GO TO 800
       DO 698 I=1,N
       DO 698 J=1,10
       SN(J,I)=0.
  698  SR(J,I)=0.
       DO 700 I=1,NN
       IF(MESH(I).NE.1)GO TO 700
       IF(F(I,1).EQ.0.)GO TO 700
       IR=F(I,1)+.001
       DO 701 J=1,10
       IF(SAB(J,I).EQ.0.)GO TO 701
       SR(J,IR)=SR(J,IR)+F(I,2)*SAB(J,I)
       SN(J,IR)=SN(J,IR)+F(I,2)
       IF(IR.EQ.N)GO TO 701
       SR(J,IR+1)=SR(J,IR+1)+(1.-F(I,2))*SAB(J,I)
       SN(J,IR+1)=SN(J,IR+1)+1.-F(I,2)
  701  CONTINUE
  700  CONTINUE
```

```
      DO 702 J=1,10
      DO 702 I=1,N
      IF(SN(J,I).EQ.0.)SN(J,I)=1.
  702 SR(J,I)=SR(J,I)/SN(J,I)
      RETURN
  800 M=-N
      DO 801 I=1,NN
      DO 801 J=1,10
  801 SAB(J,I)=0.
      DO 810 L=1,M
      DO 811 K=1,20
      KJ=JG(L,K)
      IF(KJ.EQ.0)GO TO 811
      DO 812 J=1,10
      SAB(J,KJ)=SAB(J,KJ)+SR(J,L)*F(KJ,2)
      IF(L.NE.M)SAB(J,KJ)=SAB(J,KJ)+SR(J,L+1)*(1.-F(KJ,2))
  812 CONTINUE
  811 CONTINUE
  810 CONTINUE
      RETURN
      END
      SUBROUTINE CPROP(SAB,MESH,IXEUSE)
      DATA KKK/6/                                                        ABP06310
      DIMENSION SAB(10,190),MESH(190),DLT(190),SXC(190)
      COMMON/CORE/RCTA(190),RCTF(190),WDCPOW(190),CK(190),RE(190),
     1ELSQU(190),TAUF(190),SIGFI1(190),SIGFI2(190),PZAV(190),NCORD,
     2AFLX, XKEFF,DL2,SAC,XNUF,XNUT(190),EPQ
      COMMON/XSECT/XST(2,9),VNU(9),GAMI5,GAMX5,GAMI9,ALAMX,SAX,GAMP5,    ABP06350
     1GAMP9,VCELL                                                        ABP06360
      COMMON/PARA/EPS,P1,P,ET5,ET9,TPAF(190),VNU5,VNU9,VNU1,VNU8,TAU
      COMMON/CORE2/NROW,NCOL,NSYM,JMAX,H,D1,D2,HIGHT,ALPHA,CCRIT,        ABP06380
     1CONPOW,FIWATT,VOLMF,VOLUM,V(10), FCFRAC
      CALL       TMPR(DLT,SXC,MESH,DIN,DAVG)
      EPQ=EPS                                                            ABP06400
      XNUF=VNU8                                                          ABP06410
      CAVG=0.                                                            ABP06420
      CNUM=0.                                                            ABP06430
      DO 6001 I=1,NCORD                                                  ABP06440
      IF(MESH(I) .EQ. 0) DLT(I)=0.0
      TAUM=TAU*(DLT(I)/DL2)**2
      XNUT(I)=0.                                                         ABP06450
      L=MESH(I)                                                          ABP06460
C ***    L=1, REGULAR;  L>1, WATER;  L=0, VACUUM.                        ABP06470
      IF(L-1)61,60,59                                                    ABP06480
   59 IF (L .GT. 2) GO TO 60
      CK(I)=0.0
      SIGFI2(I)=0.                                                       ABP06500
      SIGFI1(I)=0.                                                       ABP06510
      RE(I)=1.                                                           ABP06520
      TAUF(I)=TAUM
      ELSQU(I)=DLT(I)
      IF(SAB(9,I).NE.0.)ELSQU(I)=D2/(XST(1,9)*SAB(9,I)/VOLMF+D2/DLT(I))
      GO TO 6001                                                         ABP06560
   61 CK(I)=0.                                                           ABP06570
      SIGFI2(I)=0.                                                       ABP06580
      SIGFI1(I)=0.                                                       ABP06590
      RE(I)=0.                                                           ABP06600
      TAUF(I)=0.                                                         ABP06610
      ELSQU(I)=0.                                                        ABP06620
      GO TO 6001                                                         ABP06630
```

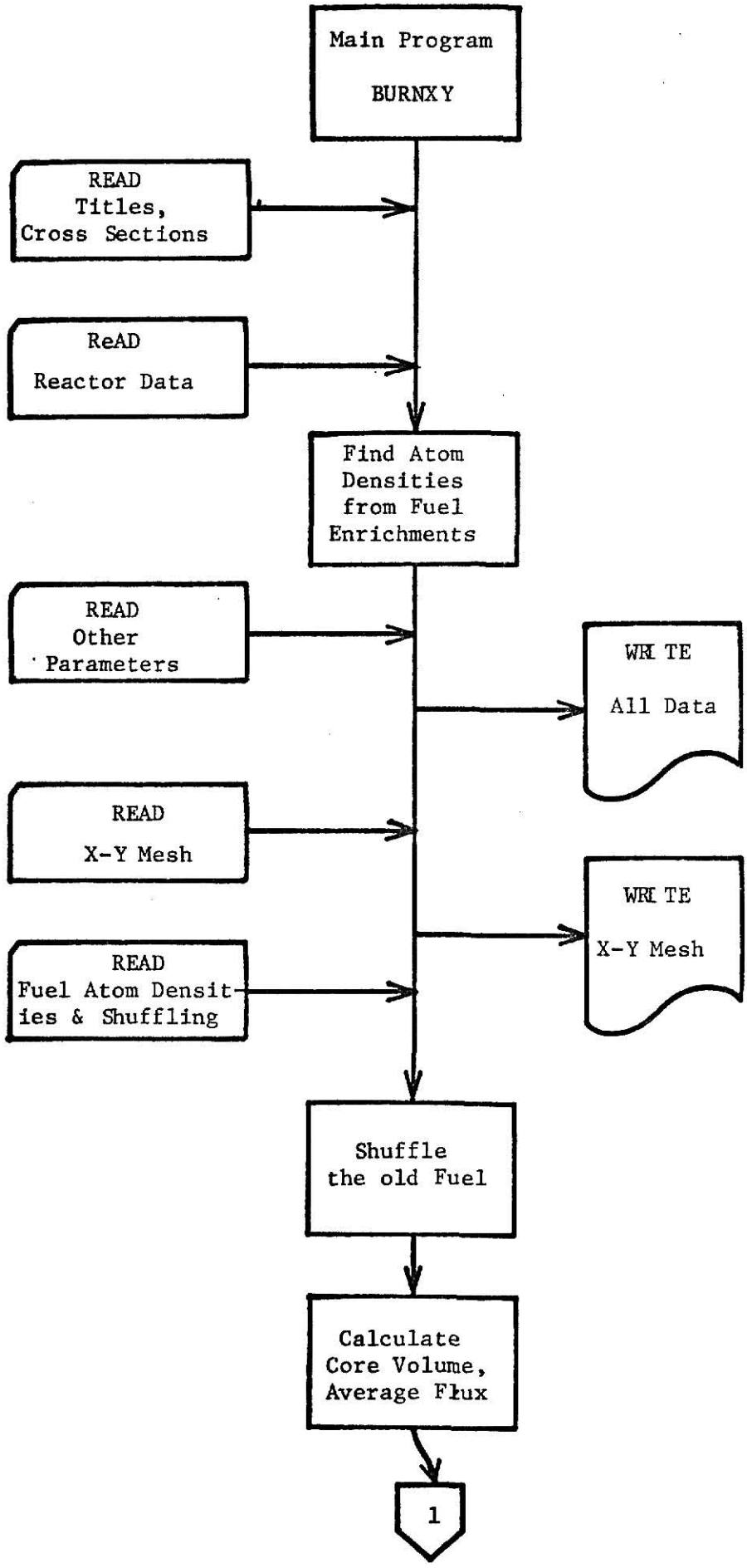
```
 60   ASIGF=0.                                                          ABP06640
      ANUM=0.                                                           ABP06650
      ADEMB=0.                                                          ABP06660
      ADEMB=0.                                                          ABP06670
      FLUX=AFLX*P2AV(I)                                                 ABP06680
      IF(FLUX.EQ.0.)FLUX=SAB(8,I)                                       ABP06690
      DO 6915 K=1,6                                                     ABP06700
      ASIGF=ASIGF+XST(2,K)*SAB(K,I)                                     ABP06710
      ANUM=ANUM+VNU(K)*SAB(K,I)*XST(2,K)                                ABP06720
 6915 ADEMB=ADEMB+SAB(K,I)*XST(1,K)                                     ABP06730
      IF(FLUX .LE. 0.0) SIGAX = 0.0
      IF (FLUX .GT. 0.0)
     1SIGAX=((GAMI5+GAMX5)*XST(2,1)*SAB(1,I)+GAMI9*XST(2,4)*SAB(4,I))*  ABP06740
     2FLUX/(ALAMX/SAX+FLUX)
      SIGAS=GAMP5*XST(2,1)*SAB(1,I)+GAMP9*XST(2,4)*SAB(4,I)             ABP06760
C ***    SIGAS IS EQUILIBRIUM SM-149 MACRO CROSS SECTION FOR ABS.       ABP06770
C ***    SINCE SAB(8,I) NOT NEEDED FOR SM-149, SET IT TO FLUX          ABP06780
      IF(IXELSE .NE. 0) GO TO 22
      SIGAX=0.0
      SIGAS=0.0
   22 CONTINUE
      SAB(8,I)=FLUX                                                     ABP06790
      DO 6916 K=7,9,2                                                   ABP06800
 6916 ADEME=ADEME+XST(1,K)*SAB(K,I)                                     ABP06810
      ADEMB=ADEME+SIGAX+SIGAS                                           ABP06820
      TU=ADEMB/(ADEMB+ADEME+SXC(I))
      ELSQU(I)=DLT(I)*(1.-TU)
      CK(I)=TUNEPS*P*ANUM/ADEMB                                         ABP06850
      SIGFI2(I)=(ASIGF/VCELL)
      SIGFI1(I)=((ANUM/VNU8)*(EPS-1.)/VCELL)
      PC(I)=P                                                           ABP06880
      TAUF(I)=TAUM
      XNUT(I)=ANUM/ASIGF                                                ABP06900
      IF (L .LE. 2) GO TO 6002
      SIGFI1(I)=SIGFI1(I)*FCFRAC
      SIGFI2(I)=SIGFI2(I)*FCFRAC
      PE(I)=1.-FCFRAC+FCFRAC*PE(I)
      CK(I)=CK(I)*FCFRAC
      ELSQU(I)=DLT(I)*(1.-FCFRAC)+FCFRAC*ELSQU(I)
 6002 CAVG=CAVG+CK(I)
      CNUM=CNUM+1.                                                      ABP06920
 6001 CONTINUE                                                          ABP06930
      CAVG=CAVG/CNUM                                                    ABP06940
      IF(CAVG.LT.0.75) GO TO 70
      IF(CAVG.GT.VNU5)GO TO 70                                          ABP06960
      WRITE(KKK,71)CAVG                                                 ABP06970
 71   FORMAT(//5X,'CPROP CALCULATION:  K-AVG =',F8.5)                   ABP06980
      RETURN                                                            ABP06990
 70   WRITE(KKK,72) CAVG                                                ABP07000
 72   FORMAT(//5X,'TERMINATION ON OUT OF BOUNDS FOR K-AVG =',E15.6)     ABP07010
      STOP                                                              ABP07020
      END                                                               ABP07030
```

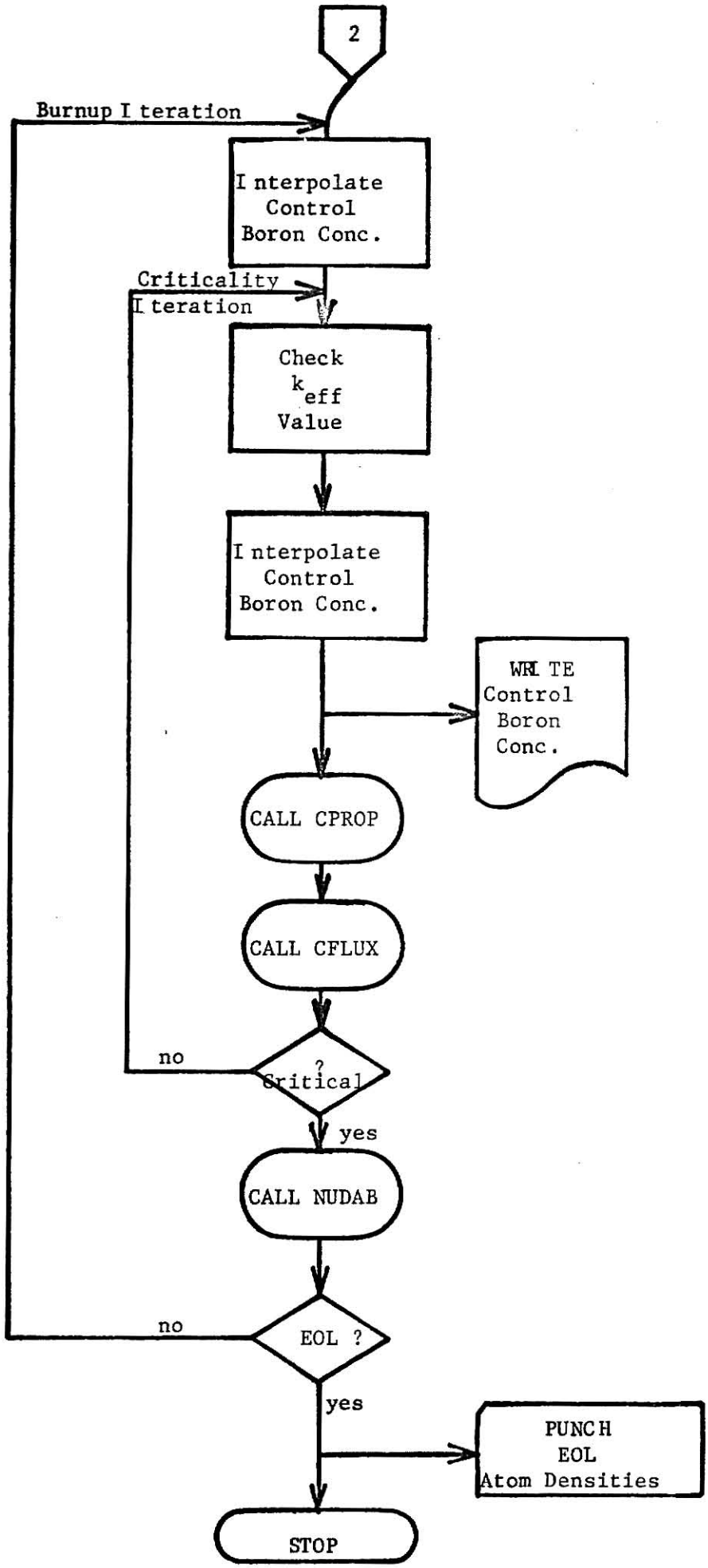
APPENDIX B: THE CODE BURNXY

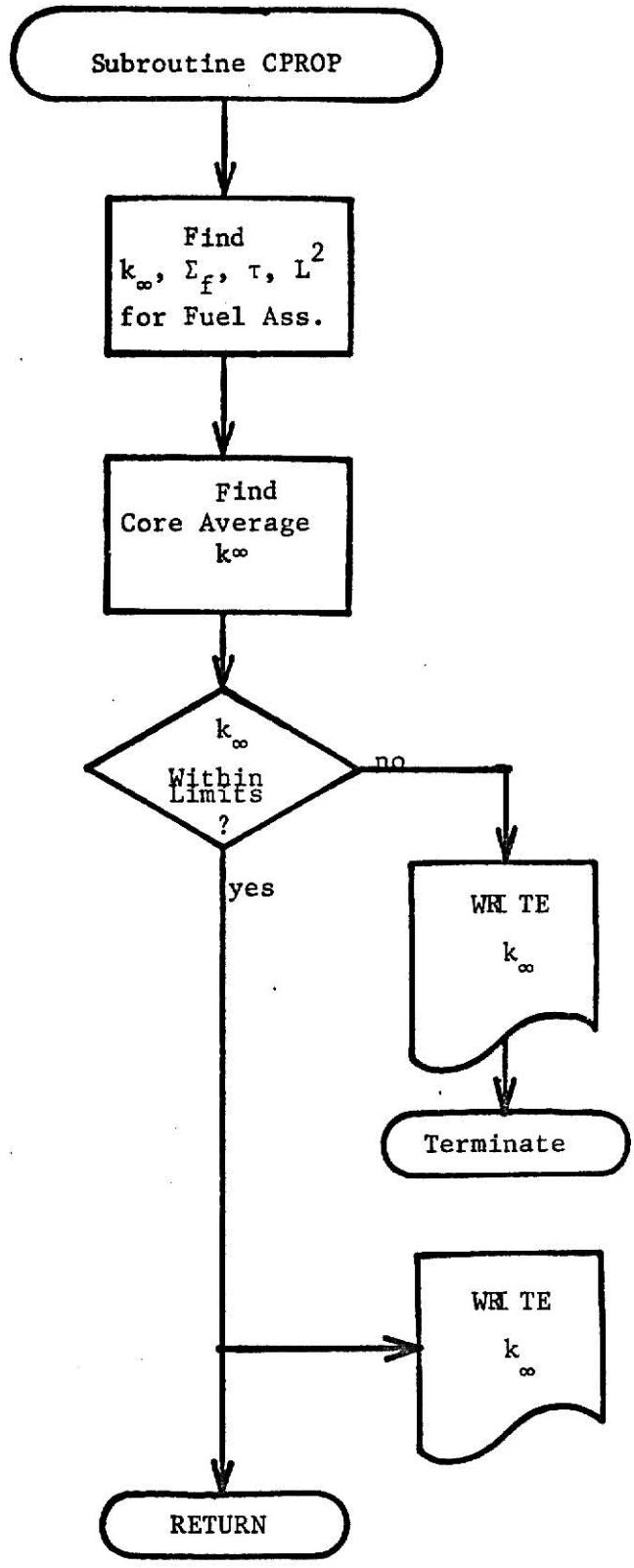
B.1 BURNXY Description

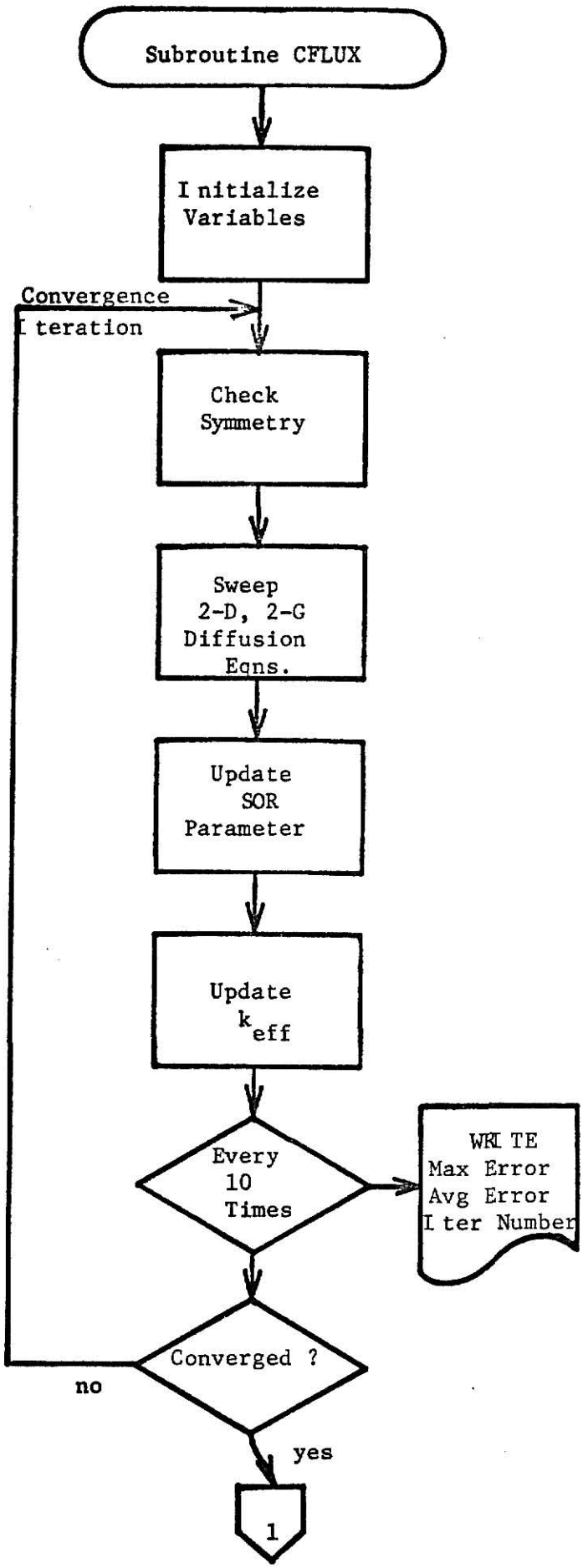
The code BURNXY is similar in function to the previously described code BURNRZ. The codes differ significantly only in the flux calculation routines and in the lack of a subroutine in BURNXY analogous to LIQHOH in BURNRZ. Naturally, the radial averaging routines XYTOR and ARAD are also not present. The BURNXY flux calculation is very similar to that in BURNRZ, but the geometry of the Laplacian operator is X-Y, not R-Z.

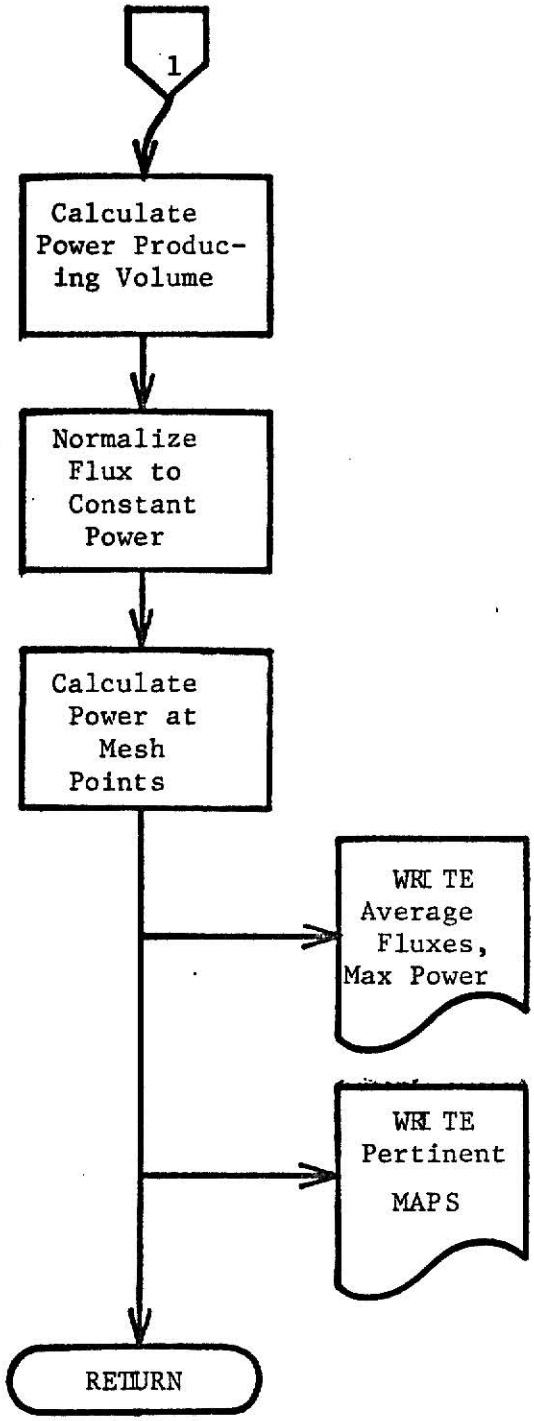
B.2 Flow Diagram of the Code BURNXY


Main Program
BURNXY
READ
Titles,
Cross Sections
ReAD
Reactor Data
Find Atom
Densities
from Fuel
Enrichments
READ
Other
Parameters
WRITE
All Data
READ
X-Y Mesh
WRITE
X-Y Mesh
READ
Fuel Atom Densit-
ies & Shuffling
Shuffle
the old Fuel
Calculate
Core Volume,
Average Flux
1


2
Burnup I teration
I nterpolate Control Boron Conc.
Criticality I teration
Check k eff Value
I nterpolate Control Boron Conc.
WRI TE Control Boron Conc.
CALL CPROP
CALL CFLUX
no
Critical ?
yes
CALL NUDAB
no
EOL ?
yes
PUNCH EOL Atom Densities
STOP


Subroutine CPROP
Find
k_∞, Σ_f, τ, L^2
for Fuel Ass.
Find
Core Average
k^∞
k_∞
Within
Limits
?
no
yes
WRITE
k_∞
Terminate
WRITE
k_∞
RETURN


Subroutine CFLUX
Initialize Variables
Convergence Iteration
Check Symmetry
Sweep 2-D, 2-G Diffusion Eqns.
Update SOR Parameter
Update k_{eff}
Every 10 Times
WRITE Max Error Avg Error Iter Number
Converged ?
no
yes
1


1
Calculate Power Producing Volume
Normalize Flux to Constant Power
Calculate Power at Mesh Points
WRITE Average Fluxes, Max Power
WRITE Pertinent MAPS
RETURN

APPENDIX C: LINEAR PROGRAMMING EQUATIONS

The linear programming equations used in the fuel shuffle model are fully described in Chen[19], and are here reproduced for convenience.

Objective Function: Deviation of Fuel Distribution = $\sum_{j=1}^{59} d_{3j}$

(Minimize)

Constraints:

Radial Uniformity:

$$1 = \frac{1}{A_k} \sum_{i=1}^{N} g_{ik} X_i - D_k \qquad k=1,2,3,\ldots 14$$

Azimuthal Uniformity:

$$1 = \frac{1}{A_j} \sum_{i=1}^{N} f_{ij} X_i - D_j \qquad j=1,2,3,\ldots 12$$

Local Uniformity:

$$1 = \frac{1}{5} (X_i + G_i) - D_i \qquad i=1,2,3,\ldots 33$$

Normalization:

$$M_\ell = \sum_{i=1}^{N} a_{i\ell} X_i \qquad \ell=2,3$$

Deviation:

$$d_{3j} = d_{1j} + d_{2j} \qquad j=1,2,3,\ldots 59$$

where the variables are defined as

N = the total number of positions in the core,

D_j = deviation of fuel distribution from ideal, sector j,

d_{1j}, d_{2j}, d_{3j} = modified deviations for sector j,

A_j = total area in sector j,

g_{ik} = fraction of area if position i that lies in the boundary of ring k,

X_i = state variable,

f_{ig} = fraction of area of position i that lies inside aximuthal area g,

G_i = summation of four state variables immediately adjacent to i,

M_ℓ = total number of positions in zone ℓ, and

$a_{i\ell}$ = 1 if i is in ℓ, 0 otherwise.

AXIAL DEPENDENCE OF NUCLEAR FUEL MANAGEMENT

by

BRUCE ALAN NAPIER

B.S., Kansas State University, 1975

AN ABSTRACT OF A MASTER'S THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Nuclear Engineering

Kansas State University

Manhattan, Kansas 66506

1977

ABSTRACT

This study represents an investigation of the effects of variations of burnup in the axial direction of nuclear reactor fuel on the operation of an example pressurized water reactor (PWR). The effects of axially changing the fuel enrichments to compensate for burnup and leakage are examined.

Flux and power distributions are neither uniform nor constant through the life of a nuclear reactor. Theoretical considerations can suggest a fuel distribution for which the flux or power distribution is instantaneously uniform, but burnup effects quickly distort this uniformity. Therefore the best that can be obtained is a fuel distribution which results in an acceptable power distribution for an adequate period of time.

A typical PWR core is modeled with a two-group, two-dimensional diffusion code and a linear programming fuel shuffling code. The behavior of the fuel burnup, local power peaking factor, and a related parameter, the core "Figure-of-Merit," are examined for a new clean core and two subsequent burnup cycles. In addition the same parameters are examined for three experimental cases for which the virgin fuel enrichments are varied axially. It is found that a small increase in fuel enrichment, for example from 3.1% to 3.2%, at the reactor top, bottom and outside edge results in lower power peaking factors, higher Figures-of-Merit, and thus in more even burnup of fuel. More even burnup indicates better fuel consumption and eventually lower fuel costs.