Progress Report - AT(30-1)2073

Fuel Cycle Code FUELMOVE III

J. Sovka and M. Benedict

September 1964

MIT-2073-3
(MITNE-58)

Progress Report on
Work Done under Contract AT(30-1)2073
with U.S. Atomic Energy Commission

FUEL CYCLE CODE, "FUELMOVE III"

by

J.A. Sovka and M. Benedict

Department of Nuclear Engineering
Massachusetts Institute of Technology

September, 1964

This work was done in part at the M.I.T. Computation Center,
Cambridge, Massachusetts

FUEL CYCLES CODE, "FUELMOVE III"

by

J.A. Sovka and M. Benedict

ABSTRACT

Further modifications to the fuel cycle code FUELMOVE are described which were made in an attempt to obtain results for reflected reactors operated under batch, outin, and bidirectional fueling schemes. Numerical methods used to obtain solutions to the condensed two-group diffusion equation are presented. Results indicated that the method for obtaining solutions for the thermal flux distribution in reflected reactors using this condensed two-group formulation appears to be inadequate in certain cases in which the reactor is treated explicitly as a separate region. A recommendation is made for one additional evaluation of this technique with a further recommendation that subsequent studies of the fuel cycle behavior of reflected reactors be made using the full two-group diffusion formulation.

TABLE OF CONTENTS

FUEL CYCLES CODE, "FUELMOVE III"

I. INTRODUCTION

The MIT Fuel Cycles Project has had available and made use of a Fortran computer code, called FUELMOVE, developed mainly by N.B. McLeod, which is described in detail in NYO-9715 (Ref. 1). The code is a two dimensional, two-group fuel depletion code capable of studying the effect of fuel and poison management on nuclear power plants fueled with U-235, U-238 and their irradiation products.

FUELMOVE was written as two separate codes, FUEL and MOVE I. In the FUEL code, the homogenized reactor unit cell properties are evaluated as a function of flux-time. The properties at specified flux-times are then put on punched cards and/or magnetic tape for subsequent use by the MOVE I code. The MOVE I code represents fuel by its flux-time transfer. It evaluates flux and power density distributions, control poison requirements, the criticality factor and average core properties throughout fuel lifetime, and when fuel is discharged, it obtains the nuclide concentrations, fuel burnup, fuel cycle cost and total energy cost.

The MOVE I code is capable of treating cylindrical reactors with azimuthal symmetry, whose reflector can be represented by a reflector savings, and allows for 150 regions, 15 axial by 10 radial. Up to five radial zones of arbitrary dimensions can be used and up to five different fuel types can be specified at any one time, one per radial zone.

Most power reactors contain a reflector and in certain fueling schemes, such as one using a soluble poison in the moderator and reflector for reactivity control, the reflector savings approximation is not adequate to describe the effect of the reflector upon criticality and flux distribution. Therefore, work was initiated in September, 1962, to modify MOVE I so that the reflector region could be treated explicitly. Flux distributions could then be calculated throughout the reactor, including reflector, thereby avoiding the need for the reflector savings approximation. The modified code, called MOVE II, was tested and compared with results obtained by MOVE I for the bidirectionally fueled CANDU reactor. The calculational changes were described in NYO-9717 (Ref. 2) along with results for the steady-state bidirectionally fueled and batch loaded cases.

However, it was found that the form of the condensed two group diffusion equation used in MOVE I and II led to an instability in the solutions for the thermal flux distributions in certain cases, giving non-physical flux solutions. No satisfactory results could be obtained for a reactor following the OUT IN fueling pattern if the reflector was treated explicitly as a separate region. The highly distorted flux distributions at high exit fuel burnups, with a large peak occurring at the core-reflector boundary, introduced a positive feedback effect upon the coefficients of the difference diffusion equation; the solution to which is obtained by the "extrapolated Liebmann" iterative method. Thus errors remaining in the flux distribution from previous iterations would increase in subsequent iterations due to this feedback until non-real flux distributions, including negative fluxes, began to occur. It was apparent that the existing method of solving the condensed, two-group diffusion equation in its differential difference form was inadequate to treat most cases of interest of reflected reactors. Therefore, a decision was made to rewrite the sections of the code whose functions are to calculate the flux distribution, criticality, power density and control poison requirements utilizing the integral form of the diffusion equation in the difference form.

At the same time, the following improvements to the MOVE I and MOVE II calculational methods were included.

(1) The reactor criticality factor of MOVE I and II is based on a thermal flux and volume weighted reactor average of a local criticality factor calculated at each mesh point. In two group theory, it is necessary to weight the diffusion

coefficients and cross-sections with the fast and thermal adjoint fluxes in order to obtain the correct criticality relationship. Since the adjoint fluxes are not available, the thermal flux is used for weighting thus introducing a possibly incorrect criticality factor. With the integral form of the diffusion equation, the correct neutron balance is obtained directly with the solution for the flux distribution, and the criticality factor is correctly obtained, consistent with the physics model assumed.

(2) The diffusion coefficients calculated as input for MOVE I and II are intended for use with a lattice-cell-averaged flux. However, the fission and absorption cross-sections, and other reactor physics parameters calculated and used by FUELMOVE I and II, require the use of a fuel average flux. Thus, the diffusion coefficients are not consistent with the model. In the new code, the flux distributions and neutron balance are calculated using a cell-averaged flux, since the diffusion equation really holds only for homogenized regions, in which the actual cell composition has been taken into account by means of suitable weighting with the cell "fine structure" flux. The use of the cell average flux also allows one to use the correct boundary conditions of continuity of thermal flux and current at the interfaces between two dissimilar regions such as the

core and the reflector. Indeed, the "fuel" flux is undefined for the reflector region;

(3) MOVE I and II are unable to meet the boundary conditions of continuity of thermal neutron current at interfaces where the diffusion coefficients are different on the two sides. This condition is taken into account directly by the use of the integral formulation provided the flux mesh points are chosen to fall on the boundary.

(4) Axial mesh spacings in MOVE I and II are required to be constant. In the new code, variable axial as well as radial mesh spacings will be possible.

(5) Up to 15 radial by 10 axial mesh points are allowed with up to 6 different radial zones.

(6) The use of soluble poison in the moderator and reflector is again a possible method of reactivity control as developed for MOVE II.

The following is a detailed description of the underlying theory and calculational methods used in the modified code, called FUELMOVE III, and includes results obtained with the code. Because of unsatisfactory behavior of the code for reflected reactors, recommendations are made to incorporate the regular two-group equations into subsequent fuel cycle codes.

II. THE MOVE III CODE

A. General

The objectives and procedures of MOVE III are essentially the same as those of MOVE I as described in the introductory sectionI. MOVE III evaluates the effect of fuel and poison management on the fuel burnup, flux and power distributions and nuclide concentrations in the fuel. It uses reactor physics parameters of the fuel, as characterized by flux-time, which are calculated by the FUEL Code. The MOVE III code was written for two dimensional (r,z) 2 group analysis of cylindrical reactors with azimuthal and axial symmetry; and allows the specification of fuel properties in a maximum of 150 regions, 15 radial by 10 axial, for one-half of the core. Up to six radial zones, each with different fuel properties, or alternately, reflector properties, can be used with an arbitrary number of radial mesh points per zone and, within certain limits, an arbitrary radial mesh spacing. Radial reflectors can be treated either explicitly, as a separate radial zone, or by means of the reflector savings approximation, while axial reflectors can only be treated by the latter technique.

B. Methods of Reactivity Control

1. Poison Control

The method of controlling the reactivity of a specific reactor is closely tied in with the type of fuel management. For those fueling schemes which require additional reactivity control, poisoning of the reactor and/or reflector is achieved either by means of absorbers or soluble poison, with an equivalent cell-homogenized absorption cross-section. It is assumed that the control poison does not affect the neutron spectrum and thereby the fuel physics properties which have been calculated by the FUEL code. The following methods of poison control of reactivity are possible in the MOVE III code.

1) Uniform poison removal, in which the spatial distribution of poison has a specified relative shape. Its magnitude is varied for reactivity control. This method could be used to approximate the use of control rods.

2) Uniform soluble poison removal in the core moderator and reflector.

3) A constant fixed poison with arbitrary shape used for power density shaping.

2. Fuel Management

It is planned to include the following possible fuel management schemes in MOVE III:

a) Batch Irradiation. The reactor is charged with a fresh load of fuel and controlled during irradiation by means of one of the poison management schemes above. The fuel is discharged when all poison has been removed and the reactor can no longer remain critical in that operating condition.

b) Steady-State Bidirectional Fueling. Short fuel elements are charged continuously at one end of a channel, moved steadily along the channel and discharged at the opposite end. The fuel in adjacent channels moves in opposite directions. The fueling rate is adjusted so that the reactor is just critical without the use of control poison.

c) Discontinuous Outin. The reactor core is divided into a number of radial zones of equal volume. At the end of a cycle, fuel is discharged from the center zone; all other fuel zones are moved one zone inward and fresh fuel is loaded into the outer zone. This operation can be performed with or without axial inversion in which fuel is divided in the middle and each half turned end for end and returned to the reactor.

C. The Neutron Diffusion Equation and its Numerical Solution

1. The Condensed Two-Troup Equation

The two-neutron group, reactor physics model as outlined in NYO-9715 results in the following equations. The fast flux behaves according to Eqn. (1.1).

$$-D_f\,\mathrm{div}(\mathrm{grad}\;\phi_f(r,z)) + \Sigma_R\,\phi_f(r,z) = \frac{\epsilon}{1-\epsilon\eta(1-p)}\left[\nu\Sigma^f\phi_t(r,z) + \eta(1-p)D_f\,\mathrm{div}(\mathrm{grad}\;\phi_f(r,z))\right] \qquad (1.1)$$

which upon simplifying becomes (1.2)

$$\frac{-D_f}{1-\epsilon\eta(1-p)}\,\mathrm{div}(\mathrm{grad}\;\phi_f(r,z)) + \Sigma_R\,\phi_f(r,z) = \frac{\epsilon\nu\Sigma^f}{1-\epsilon\eta(1-p)}\;\phi_t(r,z) \qquad (1.2)$$

while the thermal flux follows Eqn. (1.3)

$$-D_t\,\mathrm{div}(\mathrm{grad}\;\phi_t(r,z)) + (\Sigma_t+\Sigma_w)\,\phi_t(r,z) = p\,\Sigma_R\,\phi_f(r,z) \qquad (1.3)$$

where

$\phi_f(r,z)$ = fast neutron flux at (r,z)

$\phi_t(r,z)$ = thermal neutron flux

D_f , D_t = fast, thermal diffusion coefficients

Σ_R = fast removal cross-section from the fast group to the thermal group

ϵ = fast fission factor

ν = number of neutrons emitted per fission

Σ^f = thermal fission macroscopic cross-section

$\eta(1-p)$ = resonance fission contribution

Σ_t = thermal absorption macroscopic cross-section

Σ_w = the poison control contribution to the thermal absorption cross-section

p = resonance escape probability

Solving for $\phi_f(r,z)$ in (1.3) we obtain

$$\phi_f(r,z) = \frac{1}{\Sigma_R}\left[\frac{-D_t \operatorname{div}(\operatorname{grad} \phi_t(r,z)) + (\Sigma_t+\Sigma_w)\phi_t(r,z)}{p}\right] \tag{1.4}$$

Now, let

$$X(r,z) = \left[-D_t \operatorname{div}(\operatorname{grad} \phi_t(r,z)) + (\Sigma_t+\Sigma_w)\phi_t(r,z)\right] \tag{1.5}$$

Substituting expressions (1.4) and (1.5) into (1.2) gives

$$-\frac{p\ D_f}{\Sigma_R[1-\epsilon\eta(1-p)]}\ \text{div}(\text{grad}\,\frac{X(r,z)}{p}) - D_t\,\text{div}(\text{grad}\,\phi_t(r,z) + (\Sigma_t+\Sigma_w)\phi_t(r,z) = \frac{p\epsilon\nu\Sigma^f}{1-\epsilon\eta(1-p)}\ \phi_t(r,z) \tag{1.6}$$

If the resonance escape probability, p, does not vary greatly with position, one can cancel the p's in the first term of (1.6) with small error. D_f/Σ_R can be replaced by the Fermi age, τ, and Eqn. (1.6) can be simplified to

$$-\tau\alpha\ \text{div}(\text{grad}\ X(r,z)) - D_t\,\text{div}(\text{grad}\,\phi_t(r,z)) + \Sigma^T\phi_t(r,z) = \beta\ \phi_t(r,z) \tag{1.7}$$

where

$$\alpha = \frac{1}{1-\epsilon\eta(1-p)} \tag{1.8}$$

$$\beta = \frac{p\epsilon\nu\Sigma^f}{1-\epsilon\eta(1-p)} \tag{1.9}$$

$$\Sigma^T = \Sigma_t + \Sigma_w \tag{1.10}$$

Eqn. (1.7) is then the condensed two-group equation with which it is required to solve for the thermal flux, $\phi_t(r,z)$, having been given (or in the case of Σ_w, assumed) values for the core parameters. The reactor may be divided into a number of regions such that D_t, τ, α, Σ^T and β are constant within each region. The following conditions must also be satisfied:

a) The thermal flux, $\phi_t(r,z)$ is continuous in the reactor and the neutron current, $-D_t\frac{\partial\phi_t(r,z)}{\partial n}$ is continuous across interfaces between regions.

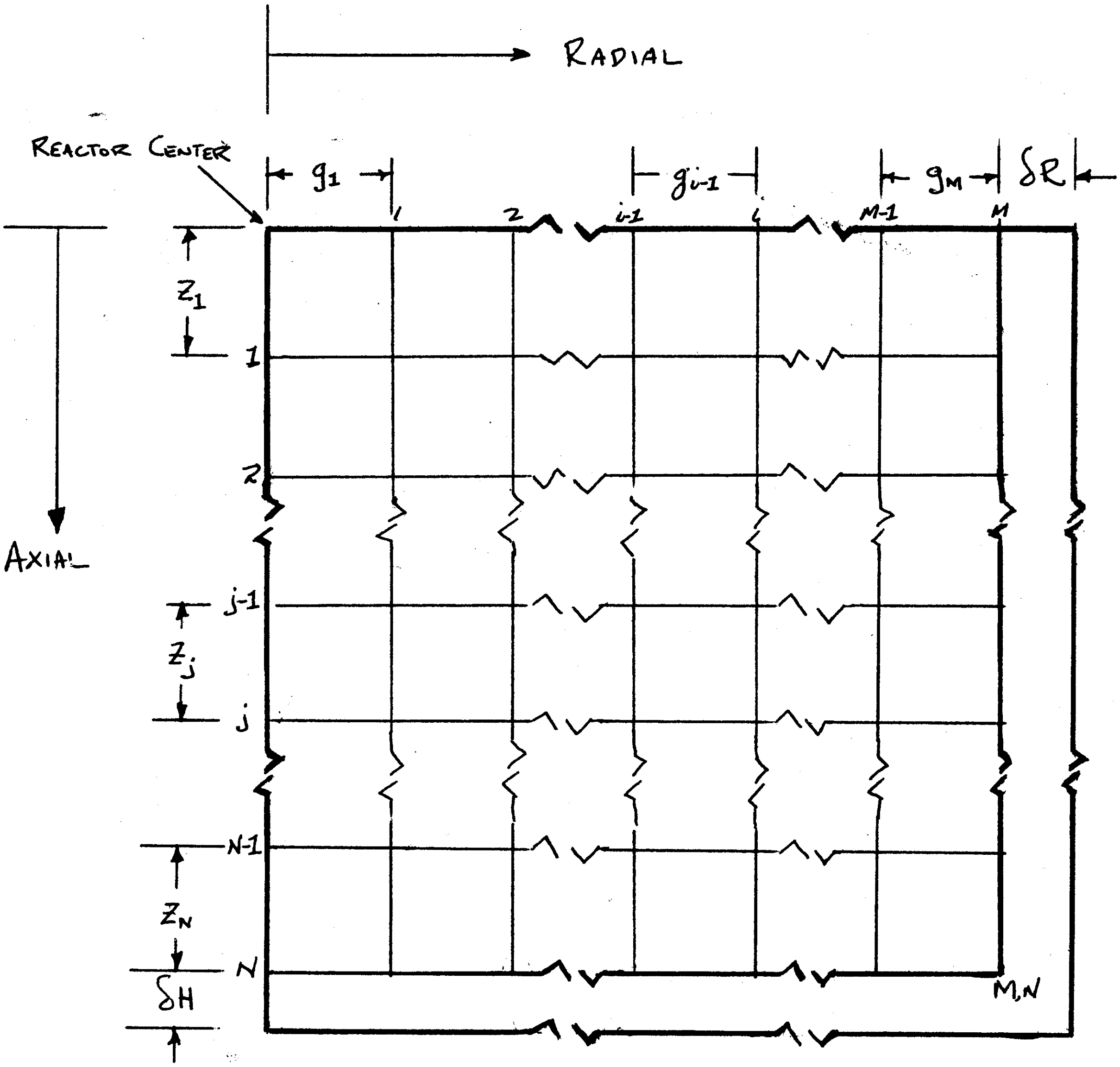


FIGURE 1 - MESH REPRESENTATION OF REACTOR CORE QUADRANT

b) On the centre-line axis $\frac{\partial \phi(0,z)}{\partial r} = 0$

and on the reactor mid-plane $\frac{\partial \phi(r,0)}{\partial z} = 0$

while on the external boundary of the reactor, the extrapolation distance, λ or logarithmic derivative will be specified $\frac{\partial \phi}{\partial n} / \phi = -\lambda$.

With the homogeneous boundary condition (b), the problem stated then defines an eigenvalue problem and we seek to find solutions of the thermal flux in (1.7) by adjusting the control poison cross-section Σ_w or the other fuel parameters by manipulating the flux-time. For the complicated reactor designs to be studied, only approximate solutions to this problem can be found by the use of numerical methods. The following will describe these methods as used in MOVE III to solve this problem numerically.

2. Derivation of the Difference Diffusion Equation

In order to proceed to the numerical solution, the mathematical derivations as described by Hageman (Ref. 3) will be adapted to the diffusion equation (1.7) above. We first impose a non-uniform mesh of horizontal and vertical lines on the reactor such that all internal interfaces and external boundaries lie exactly on mesh lines. The intersections of the horixontal and vertical lines define the mesh points at which the solution for the thermal flux $\phi(r,z)$, is sought (Fig. 1).

Consider an arbitrary interior mesh point (i,j) in the (r,z) plane as shown in Figure 2. Each mesh point will have a volume associated with it which is shown further subdivided into

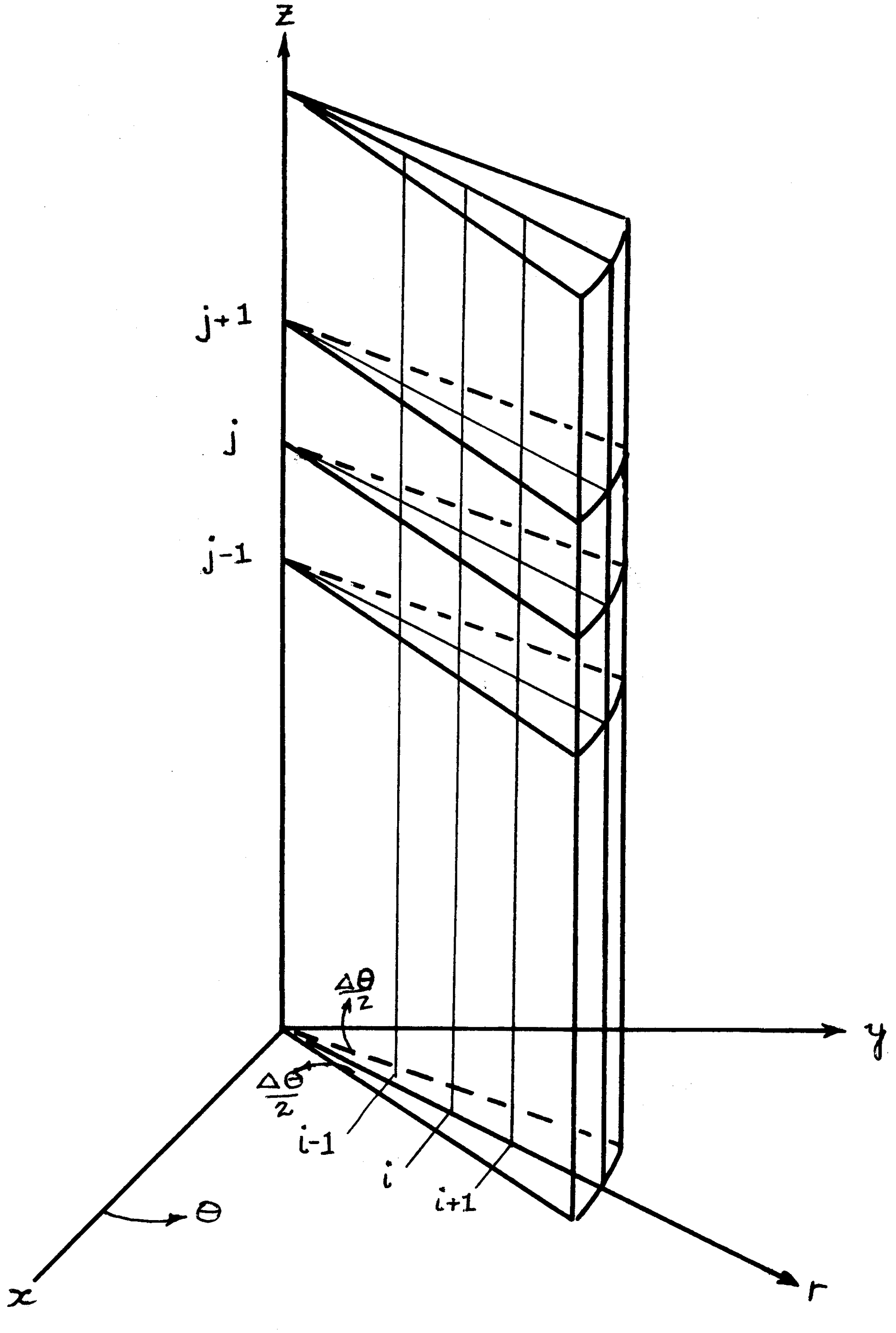


FIGURE 2 - TWO DIMENSIONAL VIEW OF CYLINDRICAL CORE

four smaller volumes, V_ℓ in Figure 3. For each of the mesh volumes, V_ℓ , surrounding the point (i,j) the "condensed" diffusion equation (1.6) may be written

$$-(\tau\alpha)_\ell \operatorname{div}(\operatorname{grad} X(r,z)) - D_\ell \operatorname{div}(\operatorname{grad} \phi(r,z)) + \Sigma_\ell^T \phi(r,z) = \beta_\ell \phi(r,z) \qquad (2.1\ )$$

Integrating (2.1) over each of the mesh volumes V_ℓ

$$\left\{ -(\tau\alpha)_\ell \int_{V_\ell} \operatorname{div}(\operatorname{grad} X(r,z))dV - D_\ell \int_{V_\ell} \operatorname{div}(\operatorname{grad} \phi(r,z))dV + \Sigma_\ell^T \int_{V_\ell} \phi(r,z)dV = \beta_\ell \int_{V_\ell} \phi(r,z)dV \right\}_{\ell=1}^{\ell=4} \qquad (2.2)$$

By the divergence theorem, the first two terms of (2.2) can be reduced to surface integrals of $\overline{\frac{\partial X}{\partial n}}$ and $\overline{\frac{\partial \phi}{\partial n}}$ respectively, over the six surfaces enclosing V_ℓ. $\overline{\frac{\partial \phi}{\partial n}}$ represents the derivative of ϕ in the direction of the outward normal to the surface. Hence

$$\overline{\frac{\partial \phi}{\partial n}} = \pm \frac{\partial \phi}{\partial n}$$

Since neither ϕ nor X (which is effectively the fast flux ϕ_f) are functions of Θ , then $\frac{\partial X}{\partial n}$ and $\frac{\partial \phi}{\partial n}$ are both zero over the two vertical plane surfaces which enclose V_ℓ. Writing Eqn. (2.2) for Volume 2

$$-(\tau\alpha)_2 \left\{ \int_{\sigma_1} \overline{\frac{\partial X}{\partial n}} d\sigma + \int_{\sigma_2} \overline{\frac{\partial X}{\partial n}} d\sigma + \int_{\sigma_9} \overline{\frac{\partial X}{\partial n}} d\sigma + \int_{\sigma_{10}} \overline{\frac{\partial X}{\partial n}} d\sigma \right\}$$
$$- D_2 \left\{ \int_{\sigma_1} \overline{\frac{\partial \phi}{\partial n}} d\sigma + \int_{\sigma_2} \overline{\frac{\partial \phi}{\partial n}} d\sigma + \int_{\sigma_9} \overline{\frac{\partial \phi}{\partial n}} d\sigma + \int_{\sigma_{10}} \overline{\frac{\partial \phi}{\partial n}} d\sigma \right\}$$
$$+ \Sigma^T \int_{V_2} \phi \, d\bar{V} = \beta_2 \int_{V_2} \phi \, d\bar{V} \qquad (2.3)$$

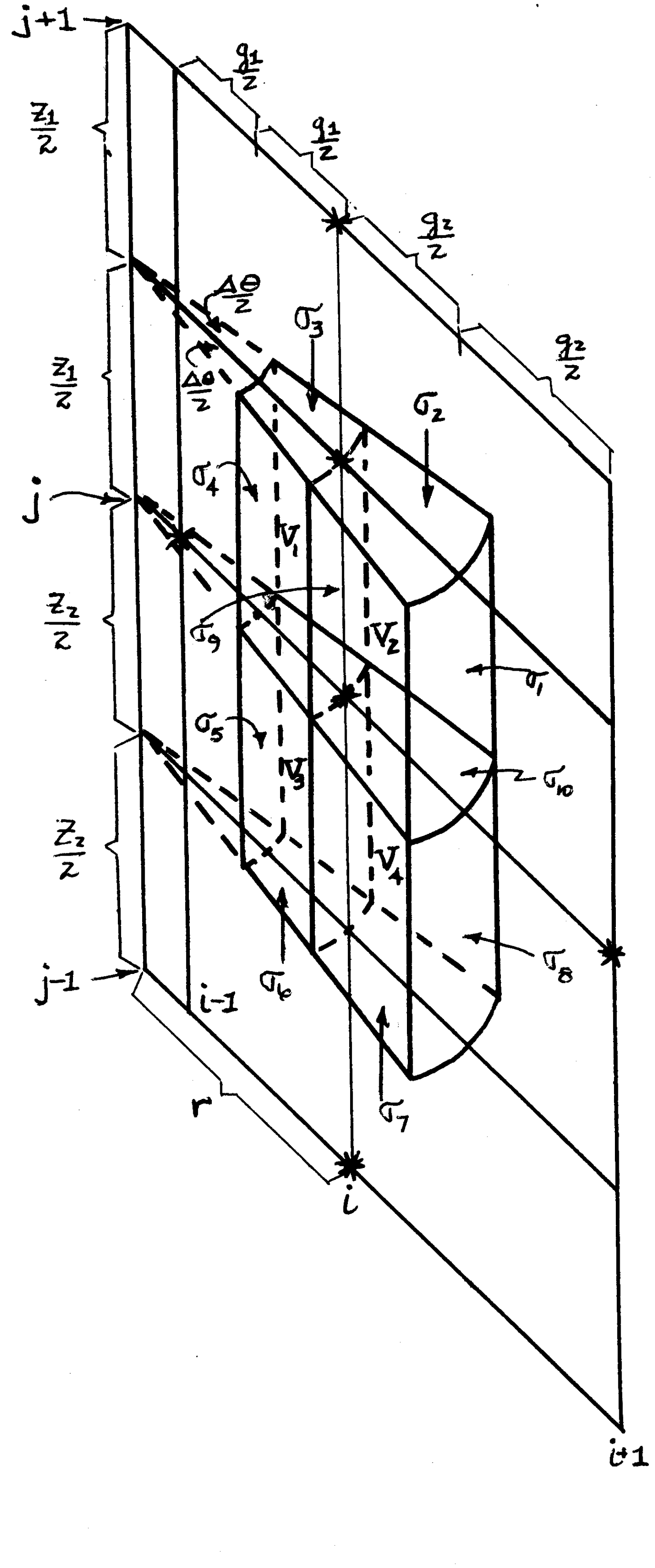


FIGURE 3 – VOLUME ASSOCIATED WITH MESH POINT (i,j)

Equations similar to (2.3) may be written for V_l , l = 1, 3 and 4.

Since the neutron currents $-D_f \frac{\partial \chi}{\partial n}$ and $-D_t \frac{\partial \phi}{\partial n}$ are assumed to be continuous across interfaces, the surface integrals over the common surfaces cancel when the four expressions of (2.2) are added. Hence, summing (2.2) over the four volume elements, we obtain Equation (2.4)

$$
\begin{aligned}
&-(\tau\alpha)_2\left[\int_{\sigma_1}\frac{\partial \chi}{\partial n}\,d\sigma + \int_{\sigma_2}\frac{\partial \chi}{\partial n}\,d\sigma\right] - (\tau\alpha)_1\left[\int_{\sigma_3}\frac{\partial \chi}{\partial n}\,d\sigma + \int_{\sigma_4}\frac{\partial \chi}{\partial n}\,d\sigma\right]\\
&-(\tau\alpha)_3\left[\int_{\sigma_5}\frac{\partial \chi}{\partial n}\,d\sigma + \int_{\sigma_6}\frac{\partial \chi}{\partial n}\,d\sigma\right] - (\tau\alpha)_4\left[\int_{\sigma_7}\frac{\partial \chi}{\partial n}\,d\sigma + \int_{\sigma_8}\frac{\partial \chi}{\partial n}\,d\sigma\right]\\
&-D_2\left[\int_{\sigma_1}\frac{\partial \phi}{\partial n}\,d\sigma + \int_{\sigma_2}\frac{\partial \phi}{\partial n}\,d\sigma\right] - D_1\left[\int_{\sigma_3}\frac{\partial \phi}{\partial n}\,d\sigma + \int_{\sigma_4}\frac{\partial \phi}{\partial n}\,d\sigma\right]\\
&-D_3\left[\int_{\sigma_5}\frac{\partial \phi}{\partial n}\,d\sigma + \int_{\sigma_6}\frac{\partial \phi}{\partial n}\,d\sigma\right] - D_4\left[\int_{\sigma_7}\frac{\partial \phi}{\partial n}\,d\sigma + \int_{\sigma_8}\frac{\partial \phi}{\partial n}\,d\sigma\right]\\
&\qquad + \sum_{l=1}^{4}\left[\Sigma_l^T \int_{V_l}\phi\,d\bar{V}\right. = \sum_{l=1}^{4}\beta_l\int_{V_l}\phi\,dV
\end{aligned}
\tag{2.4}
$$

In order to obtain the finite-difference equations at mesh point (i,j) it is necessary to make numerical approximations to the integrals of (2.4). Integrals such as

$\int_{\sigma_1} \frac{\partial \phi}{\partial n} d\sigma$ are approximated by

$$\int_{\sigma_1} \frac{\partial \phi}{\partial n} d\sigma \approx \left[\frac{\phi_{i+1,j} - \phi_{i,j}}{g_2}\right] \int_{\sigma_1} d\sigma \tag{2.5}$$

Now

$$\int_{\sigma_1} d\sigma = \frac{z_1}{2} \times S_{\sigma_1} = \frac{z_1}{2}\left(r_i + \frac{g_2}{2}\right)\Delta\theta \tag{2.6}$$

Thus

$$\int_{\sigma_1} \frac{\partial \phi}{\partial n} d\sigma = \left[\frac{\phi_{i+1,j} - \phi_{i,j}}{g_2}\right] \cdot \frac{z_1}{2}\left(r_i + g_2/2\right)\Delta\theta \tag{2.7}$$

Likewise,

$$\int_{\sigma_1} \frac{\partial X}{\partial n} d\sigma = \left[\frac{X_{i+1,j} - X_{i,j}}{g_2}\right] \cdot \frac{z_1}{2}\left(r_i + g_2/2\right)\Delta\theta \tag{2.8}$$

In a similar way for surface σ_2,

$$\int_{\sigma_2} \frac{\partial \phi}{\partial n} d\sigma \approx -\left[\frac{\phi_{i,j} - \phi_{i,j-1}}{z_1}\right] \cdot \int_{\sigma_2} d\sigma \tag{2.9}$$

where

$$\begin{aligned}\int_{\sigma_2} d\sigma &= \left[\left(r_i + g_2/2\right)^2 - r_i^2\right]\frac{\Delta\theta}{2} \\ &= \frac{g_2}{2}\left(r_i + g_2/4\right)\Delta\theta\end{aligned} \tag{2.10}$$

So

$$\int_{\sigma_2} \frac{\partial \phi}{\partial n} d\sigma = -\left[\frac{\phi_{i,j} - \phi_{i,j-1}}{z_1}\right] \frac{g_2}{2}\left(r_i + g_2/4\right)\Delta\theta \tag{2.11}$$

Analogous expressions are derived for the remaining surfaces.

The volume integrals, such as $\int_{V_2} \phi(r,z)\,dV$, are approximated by

$$\int_{V_2} \phi(r,z)\,dV \approx \phi_{i,j}\left[\frac{z_1 g_2 (r_i + g_2/4)}{4}\right]\Delta\theta \qquad (2.12)$$

Using the above approximations, the condensed two-group finite-difference diffusion equation at a mesh point (i,j) may be written as

$$\delta_1 X(i,j-1) + \delta_2 X(i-1,j) + \delta_3 X(i,j+1) + \delta_4 X(i+1,j) + \delta_5 X(i,j)$$
$$+\gamma_1 \phi(i,j-1) + \gamma_2 \phi(i-1,j) + \gamma_3 \phi(i,j+1) + \gamma_4 \phi(i+1,j) + \gamma_0 \phi(i,j)$$
$$= \gamma_5 \phi(i,j) \qquad (2.13)$$

where

$$\delta_1 = \frac{-(\tau\alpha)_1 g_1 (r_i - g_1/4) - (\tau\alpha)_2 g_2 (r_i + g_2/4)}{2 z_1} \qquad (2.14)$$

$$\delta_2 = -\left(\frac{r_i}{2g_1} - \frac{1}{4}\right)\left[(\tau\alpha)_1 z_1 + (\tau\alpha)_3 z_2\right] \qquad (2.15)$$

$$\delta_3 = \frac{-(\tau\alpha)_3 g_1 (r_i - g_1/4) - (\tau\alpha)_4 g_2 (r_i + g_2/4)}{2 z_3} \qquad (2.16)$$

$$\delta_4 = -\left(\frac{r_i}{2g_2} + \frac{1}{4}\right)\left[(\tau\alpha)_2 z_1 + (\tau\alpha)_4 z_2\right] \qquad (2.17)$$

$$\delta_5 = -(\delta_1 + \delta_2 + \delta_3 + \delta_4) \qquad (2.18)$$

$$\gamma_1 = \frac{-D_1 g_1(r_i - g_1/4) - D_2 g_2(r_i + g_2/4)}{2 z_1} \qquad (2.19)$$

$$\gamma_2 = -\left(\frac{r_i}{2g_1} - \frac{1}{4}\right)\left[D_1 z_1 + D_3 z_2\right] \qquad (2.20)$$

$$\gamma_3 = \frac{-D_3 g_1(r_i - g_1/4) - D_4 g_2(r_i + g_2/4)}{2 z_3} \qquad (2.21)$$

$$\gamma_4 = -\left(\frac{r_i}{2g_2} + \frac{1}{4}\right)\left[D_2 z_1 + D_4 z_2\right] \qquad (2.22)$$

$$\gamma_5 = \frac{1}{4}\Big[\beta_1 g_1 z_1(r_i - g_1/4) + \beta_2 g_2 z_1(r_i + g_2/4) + \beta_3 g_1 z_2(r_i - g_1/4) + \beta_4 g_2 z_2(r_i + g_2/4)\Big] \qquad (2.23)$$

The coefficient γ_0 is further subdivided into the following

$$\gamma_0 = \gamma_{oa} + \gamma_{ob} + \gamma_{oc} + \gamma_{ofp} \qquad (2.24)$$

where in turn

$$\gamma_{oa} = -\left[\gamma_1 + \gamma_2 + \gamma_3 + \gamma_4\right] \qquad (2.25)$$

$$\gamma_{ob} = \frac{1}{4}\Big[\sigma^{n}_{w,1}\, g_1 z_1 (r_i - g_1/4) + \sigma^{n}_{w,2}\, g_2 z_1 (r_i + g_2/4) + \sigma^{n}_{w,3}\, g_1 z_2 (r_i - g_1/4) + \sigma^{n}_{w,4}\, g_2 z_2 (r_i + g_2/4)\Big] \qquad (2.26)$$

$$\gamma_{oc} = \frac{1}{4}\Big[\Sigma^{cell}_{1}\, g_1 z_1 (r_i - g_1/4) + \Sigma^{cell}_{2}\, g_2 z_1 (r_i + g_2/4) + \Sigma^{cell}_{3}\, g_1 z_2 (r_i - g_1/4) + \Sigma^{cell}_{4}\, g_2 z_2 (r_i + g_2/4)\Big] \qquad (2.27)$$

$$\gamma_{ofp} = \frac{1}{4}\Big[\Sigma^{FP}_{1}\, g_1 z_1 (r_i - g_1/4) + \Sigma^{FP}_{2}\, g_2 z_1 (r_i + g_2/4) + \Sigma^{cell}_{3}\, g_1 z_2 (r_i - g_1/4) + \Sigma^{cell}_{4}\, g_2 z_2 (r_i + g_2/4)\Big] \qquad (2.28)$$

The various absorption coefficients used in Equns. (2.26) to (2.28) are

σ^{n}_{w} = the weighting factor for multiplying the adjustable poison cross-section which can be used to control reactivity. The factor ean be varied between 0 and 1.0 throughout all possible 150 regions, thereby approximating the effect of spatially lumped absorbers such as control rods. (2.28a)

Σ^{cell}_{ℓ} = total homogenized macroscopic absorption cross-section of the reactor cell, including the unpoisoned cell absorption (2.28b) cross-section, the xenon homogenized cross-section, and any fixed absorber built into the fuel.

If the mesh point (i,j) lies on a segment of the boundary where $\frac{\partial\phi}{\partial n} = 0$, then the constants D_ℓ, $(\tau\alpha)_\ell$, β_ℓ and the absorption cross-sections for those regions which are outside of V_ℓ are set equal to zero.

For mesh points (i,j) lying on the external boundaries of the reactor, the logarithmic boundary condition is applied, i.e.

$$\left(\frac{\partial\phi}{\partial n}\right)_R = -\frac{\phi_R}{\delta_R} \qquad (2.29)$$

where ϕ_R = value of flux on the outer boundary

δ_R = extrapolation distance beyond the outer boundary at which the flux is assumed zero.

The expression for the normal derivative of the flux then is inserted into the surface integral terms of equation (2.3). Since the mesh point falls on the outer boundaries, then surfaces bounding the reactor are the ones at which $\left(\frac{\partial\phi}{\partial n}\right)$ of Eqn. (2.29) is to be applied. Thus, for example, at the outer radius, one of the relevant surface integrals would be $\int_{\sigma_9} \frac{\partial\phi}{\partial n} d\sigma$ and would be approximated by

$$\int_{\sigma_9} \frac{\partial\phi}{\partial n} d\sigma \approx -\frac{\phi(i,j)_R}{\delta_R} \cdot \frac{g_1}{2} \cdot r_i \, \Delta\theta \qquad (2.30)$$

Similarly, at the axial boundary, for example

$$\int_{\sigma_{ax}} \frac{\partial \phi}{\partial n} d\sigma \approx - \frac{\Phi(i,j)}{\delta_H} H \quad \frac{g_1}{2} \left(r_i - g_1/4 \right) \Delta \theta \tag{2.31}$$

the spatial coefficients of (2.30) and (2.31) are then added to the coefficients γ_{oq} for the boundary mesh points.

The quantity X(i,j) at each mesh point is obtained by first integrating Eqn. (1.5) over the volume element V_ℓ , i.e.

$$\int_{V_\ell} X(r,z) dV = - \int_{V_\ell} D \operatorname{div}(\operatorname{grad} \phi(r,z)) dV + \sum^T \int_{V_\ell} \phi(r,z) dV \tag{2.32}$$

X(r,z) is assumed constant throughout V_ℓ , thus

$$X(r,z) = \frac{1}{V_\ell} \left[-D \int_{V_\ell} \operatorname{div}(\operatorname{grad} \Phi(r,z)) dV + \sum^T \int_{V_\ell} \Phi(r,z) dV \right] \tag{2.33}$$

Here again the volume integral of the first term is transformed to a surface integral by the divergence theorem and

$$X(r,z) = \frac{1}{V_\ell} \left[-D \int_{\sigma_\ell} \frac{\partial \phi}{\partial n} d\sigma + \sum^T \int_{V_\ell} \Phi(r,z) dV \right] \tag{2.34}$$

The integrals are then approximated as given above, so that in finite difference form

$$X(i,j) = \frac{1}{V(i,j)} \left[\gamma_1 \phi(i,j-1) + \gamma_2 \phi(i-1,j) + \gamma_3 (i,j+1) + \gamma_4 \phi(i+1,j) + \gamma_o \phi(i,j) \right] \tag{2.35}$$

Note that this derivation assumes that X(r,z) and $\frac{\partial X(r,z)}{\partial r}$ are continuous across region boundaries which effectively requires the fast flux to obey the same boundary conditions. In order to stabilize the operation of the code and prevent oscillations in flux magnitudes or poison estimates, a damping factor FLDAMP is used which enables one to choose a specified fraction of the present value of X(i,j) as well as a fraction of the previous value in the following way.

$$X(i,j) = \text{FLDAMP} \cdot X(i,j)^{New} + (1-\text{FLDAMP}) \cdot X(i,j)^{old}$$

3. The Spatial Flux Distribution Solution

Numerical values for the fluxes at the mesh points, Ø(i,j), are obtained with the "extrapolated Leibmann" iterative method. Equation (2.12) is first rearranged in the following way

$$\phi(i,j)\cdot\left[-\gamma_0+\gamma_5\right] - \delta_5 X(i,j) = \delta_1 X(i,j-1) + \delta_2 X(i-1,j) + \delta_3 X(i,j+1) + \delta_4 X(i+1,j) + \gamma_1\phi(i,j-1) + \gamma_2\phi(i-1,j) + \gamma_3\phi(i,j+1) + \gamma_4\phi(i+1,j) \qquad (3.1)$$

The X(i,j) are previously calculated by the use of Eqn. (2.34) using fluxes Ø(i,j) from the preceding converged solution (or, if it is the first time through the calculations, fluxes calculated from an assumed flux shape). They are then transformed into quantities Q(i,j) where

$$Q(i,j) = \frac{X(i,j)}{\phi_P(i,j)} \qquad (3.2)$$

where $\phi_P(i,j)$ = fluxes at mesh point (i,j) from the preceding converged solution.

Eqn. (3.1) is then rewritten as

$$\phi(i,j)\cdot\left[\gamma_6 - \delta_5 Q(i,j)\right] = \phi(i,j-1)\left[\gamma_1 + \delta_1 Q(i,j-1)\right] + \phi(i-1,j)\left[\gamma_2 + \delta_2 Q(i-1,j)\right] + \phi(i,j+1)\left[\gamma_3 + \delta_3 Q(i,j+1)\right] + \phi(i+1,j)\left[\gamma_4 + \delta_4 Q(i+1,j)\right] \qquad (3.3)$$

where $\gamma_6 = \gamma_5 - \gamma_0$ (3.4)

The flux at a mesh point for the ($\nu+1$) iteration is then calculated by the algorithm (3.5)

$$\phi^{\nu+1}(i,j) = F\left\{\phi^{\nu+1}(i,j-1)\left[\gamma_1 + \delta_1 Q(i,j-1)\right] + \phi^{\nu+1}(i-1,j)\left[\gamma_2 + \delta_2 Q(i-1,j)\right] + \phi^{\nu}(i,j+1)\left[\gamma_3 + \delta_3 Q(i,j+1)\right] + \phi^{\nu}(i+1,j)\left[\gamma_4 + \delta_4 Q(i+1,j)\right]\right\} \div \left\{\gamma_6 - \delta_5 Q(i,j)\right\} + (1-F)\cdot\phi^{\nu}(i,j) \qquad (3.5)$$

where F = extrapolated Leibmann parameter with a value between 1 and 2.

The iteration proceeds until the error criterion, given by equation (3.6) is satisfied.

$$\frac{\phi^{\nu+1}(i,j) - \phi^{\nu}(i,j)}{\phi^{\nu+1}(i,j)} \leq \epsilon \qquad (3.6)$$

where ϵ is a smallpredetermined number, usually about 0.001.

4. The Neutron Balance

The converged fluxes $\phi(i,j)$ from the spatial solution are then used to calculate a new neutron balance so that the reactivity of the reactor can be determined. Firstly, new values of X(i,j) are calculated using Eqn. (2.34). Then the thermal and fast neutron leakages from the volume elements V_ℓ , associated with each mesh point as defined by Eqns. (4.1) and (4.2) respectively, are calculated.

C36(i,j) = Thermal neutron leakage = $\gamma_1 \phi(i,j-1) + \gamma_2 \phi(i-1,j)$

$$+\gamma_3 \phi(i,j+1) + \gamma_4 \phi(i+1,j) + \gamma_{0a} \phi(i,j) \qquad (4.1)$$

FASK (i,j) = Fast neutron leakage = $\delta_1 X(i,j-1) + \delta_2 X(i-1,j)$

$$+\delta_3 X(i,j+1) + \delta_4 X(i+1,j) + \delta_5 X(i,j) \qquad (4.2)$$

The total thermal leakage from the reactor is obtained by summing the leakage from each individual volume element for all elements, i.e.

TOTTL = total thermal neutron leakage = $\sum_{i=1}^{IRL} \sum_{j=1}^{JZL}$ C36(i,j)

(4.3)

Similarly, the total fast leakage is

$$\text{TOTFL} = \sum_{i=1}^{IRL} \sum_{j=1}^{JZL} \text{FASK}(i,j) \qquad (4.4)$$

The thermal neutron absorption in the reactor is just

$$\text{TOTABS} = \sum_{i=1}^{IRL} \sum_{j=1}^{JZL} \phi(i,j) \cdot \left[\gamma_{0a} + \gamma_{0b} + \gamma_{fp}\right] \qquad (4.5)$$

The absorption excluding the contribution due to poison and absorbers is

$$\text{ABSNP} = \sum_{i=1}^{IRL} \sum_{j=1}^{JZL} \phi(i,j)\, \gamma_{oc}(i,j) \qquad (4.6)$$

Production of thermal neutrons is calculated using the coefficient γ_5 from Eqn. (2.22) so that

$$\text{TOTPR} = \text{Total neutron production} = \sum_{i=1}^{IRL} \sum_{j=1}^{JZL} \phi(i,j)\, \gamma_5(i,j) \qquad (4.7)$$

The neutron multiplication factor, C, is then defined as the total production of neutrons divided by total loss of neutrons due to thermal and fast leakage and absorption. Thus

$$C = \frac{\text{TOTPR}}{\text{TOTTL} + \text{TOTFL} + \text{TOTABS}} \qquad (4.8)$$

5. Adjustment of Criticality

If the multiplication factor C, defined in (4.8) is not unity, and therefore the reactor is either supercritical or subcritical, it is necessary to adjust the reactor properties until C = 1. The means by which this is done depends upon the fuel management scheme being studied.

For a steady state bidirectional fueled reactor, the fuel charge rate is adjusted, the reactor properties re-evaluated, the flux distribution recalaulated and the neutron balance obtained until $C = 1.0 \pm \epsilon$ where ϵ is a small number, e.g. 0.0001.

For the batch irradiation or the outin fueling pattern, adjustments in control poison, Σ_W, are made to keep the reactor critical. An initial estimate of Σ_W can be obtained from an initial guess for the relative flux distribution and by making a neutron balance for the reactor. For example, in uniform poison removal, the relative poison magnitude σ_W^n defined by Eqn. (2.28a) is specified. it is then necessary to evaluate the normalization constant Σ_{W1} , so that the absolute magnitude, Σ_W , can be obtained. This is done, as mentioned above, by making a neutron balance for the reactor using the following equation

$$\Sigma_{W1} = \frac{\sum\limits^{i,j} \left\{ \Phi(i,j)\left[\gamma_5 - \gamma_{oc} - \gamma_{ofp}\right] - C36(i,j) - FASK(i,j) \right\}}{\sum\limits^{i,j} \gamma_{ob}\,\Phi(i,j)} \tag{5.1}$$

The spatial coefficients used in the spatial flux distribution iteration are recalculated with the new value of

$$\Sigma_W = \Sigma_{W1} \cdot \sigma_W^n(i,j) \tag{5.2}$$

The new poisoned multiplication factor is calculated, compared to 1.0 and if still not within ϵ of unity, another poison estimate is made. In order to damp out oscillations in this outer iteration loop on Σ_W , a damping factor, DAMPIN, is employed, using the previous value of and the latest value with the use of (5.3)

$$\Sigma_{W1} = \text{DAMPIN} \cdot \Sigma_{W1}^{New} + (1 - \text{DAMPIN})\, \Sigma_{W1}^{old} \tag{5.3}$$

DAMPIN is an input number and usually has a value between 0.5 and 1.

Similarly, if soluble poison is used in the core moderator and reflector to control reactivity, the poison concentration in parts per million, PPM is calculated by (5.4)

$$PPM = \frac{\sum^{i,j} \left\{ \phi(i,j)\left[\gamma_5 - \gamma_c - \gamma_{ofp}\right] - C36(i,j) - FASK(i,j) \right\}}{\sum^{i,j} TDFMOD(i) \cdot \phi(i,j) \cdot \gamma_{ob}(i,j) \cdot FACMOD} \tag{5.4}$$

where FACMOD = $(0.6025 \times 10^{-6})\ \rho_{mod}\, \sigma_{pois} / A_{pois}$ (5.5)

and TDFMOD(i) = thermal disadvantage factor for the moderator

$$= \frac{\overline{\Phi}_{mod}\, V_{mod}}{\sum_{cell} \phi_i V_i} \tag{5.6}$$

and where, in turn ρ_{mod} = moderator density, g/ec

σ_{pois} = poison thermal microscopic absorption eross section

A_{pois} - atomic weight of poison

$\overline{\Phi}_{mod.}$ = average moderator flux

V_{mod} = volume of moderator in unit cell

$\sum_{cell} \phi_i V_i$ = sum of flux times volume of all components in unit cell

Here again PPM is multiplied by a damping factor DAMPIN, so that

$$PPM = DAMPIN \cdot PPM^{New} + (1-DAMPIN) \cdot PPM^{Old} \tag{5.7}$$

6. Homogenized Cell Cross-Sections

As pointed out in the Introduction, the diffusion coefficients calculated as input to the MOVE code are intended for use with an average cell flux, whereas the fission and absorption cross-sections calculated by FUEL as a function of flux-time, require the use of an average <u>fuel</u> flux. In order to make the diffusion coefficients and the cross-sections consistent, cross-sections in MOVE 3 are converted so that the neutron reaction rates when calculated using the average <u>cell</u> flux are the same as those obtained by using the average fuel flux. The absorption cross-section to be multiplied by the fuel flux, $\overline{\Sigma}_a$, is defined in NYO-9715 as

$$\overline{\Sigma}_a = \Sigma_{f1} V_{f1} + \Sigma^{\circ}_{mod} (1-V_{f1}) \psi_{mod} \qquad (6.1)$$

where Σ_{f1} = unhomogenized fuel cross-section

V_{f1} = volume fraction of fuel in the unit cell

Σ°_{mod} = unhomogenized cross-section of non-fuel materials

$\psi_{mod} = \phi_{mod}/\phi_{f1}$ = ratio of non-fuel flux to fuel flux.

However, the cell homogenized absorption cross-section to be used with the average cell flux is

$$\Sigma_a' = \frac{\Sigma_{f1} V_{f1} \phi_{f1} + \Sigma_{mod} (1-V_{f1}) \phi_{mod}}{V_{f1} \phi_{f1} + (1-V_{f1}) \phi_{mod}} \qquad (6.2)$$

Dividing the right side of (6.2) by ϕ_{f1} gives

$$\Sigma_a' = \frac{\Sigma_{fl} V_{fl} + \Sigma_{mod}(1-V_{fl})\psi_{mod}}{V_{fl} + (1-V_{fl})\psi_{mod}} \qquad (6.3)$$

which from (6.1) becomes

$$\Sigma_a' = \frac{\overline{\Sigma}_a}{V_{fl} + (1-V_{fl})\psi_{mod}} \qquad (6.4)$$

Thus, macroscopic absorption and fission cross-sections must be divided by the factor $V_{fl} + (1-V_{fl})\psi_{mod}$ in order that the diffusion coefficients be used correctly.

III. RESULTS

The first debugging and initial test runs of MOVE III were made with a different form of the spatial flux solution from that described in Section II, p. 20. The fast leakage terms, instead of entering the five point difference formula were used directly as the quantity FASK (i,j) as defined by Eqn. (4.2). The extrapolated Liebmann iterative method then was carried out on Eqn. (B.1)

$$\begin{aligned}\phi^{\nu+1}(i,j) = F\Big\{ & \phi^{\nu+1}(i,j-1)\gamma_1 + \phi^{\nu+1}(i-1,j)\gamma_2 + \phi^{\nu}(i,j+1)\gamma_3 \\ & + \phi^{\nu}(i+1,j)\gamma_4 + \mathrm{FASK}(i,j)\Big\} \div \gamma_6 \\ & + (1-F)\,\phi^{\nu}(i,j)\end{aligned} \qquad (B.1)$$

The fast leakage term, FASK(i,j) was calculated using fluxes from the previously converged solution and used the poison cross-section estimated prior to the flux calculation. A damping factor, FLDAMP, was used to try to eliminate any oscillations in the fast leakage such that the value of FASK(i,j) used consisted of a fraction of the newly calculated value of the leakage and the remaining fraction of the previously calculated leakage, i.e.

$$\mathrm{FASK}(i,j) = \mathrm{FLDAMP}\cdot\mathrm{FASK}(i,j)^{New} + (1-\mathrm{FLDAMP})\cdot\mathrm{FASK}(i,j)^{old} \qquad (B.2)$$

Figure 4 shows the thermal flux distribution obtained with MOVE III for the CANDU, natural uranium, bidirectionally fueled reactor compared with the thermal flux calculated by a regular two-group solution by AECL. (Ref.4) The burnup predicted for steady state by MOVE III is about 3.4% lower than the AECL figure. The flux obtained by the code differs from the AECL results by up to 13% higher in the core, at midway between the inner ,"zero-radial-buckling" zone and the core-reflector boundary. On the other hand, the reflector zone thermal flux calculated by MOVE III is more than 13% lower than the AECL values. The major discrepancy between the two results is the lack of thermal flux "bump" in the reflector just outside the core in the MOVE III distribution. The reflector therefore appears less effective in returning thermal neutrons to the core. This in turn lowers the reactivity of the reactor and hence the steady state discharge fuel burnup predicted by the code is less than the regular two group value. This underestimate of

FIGURE 4

RADIAL THERMAL FLUX DISTRIBUTION FOR NAT. U, BIDIRECTIONALLY-FUELED CANDU REACTOR

CALCULATED ACCORDING TO EQUATION B.1

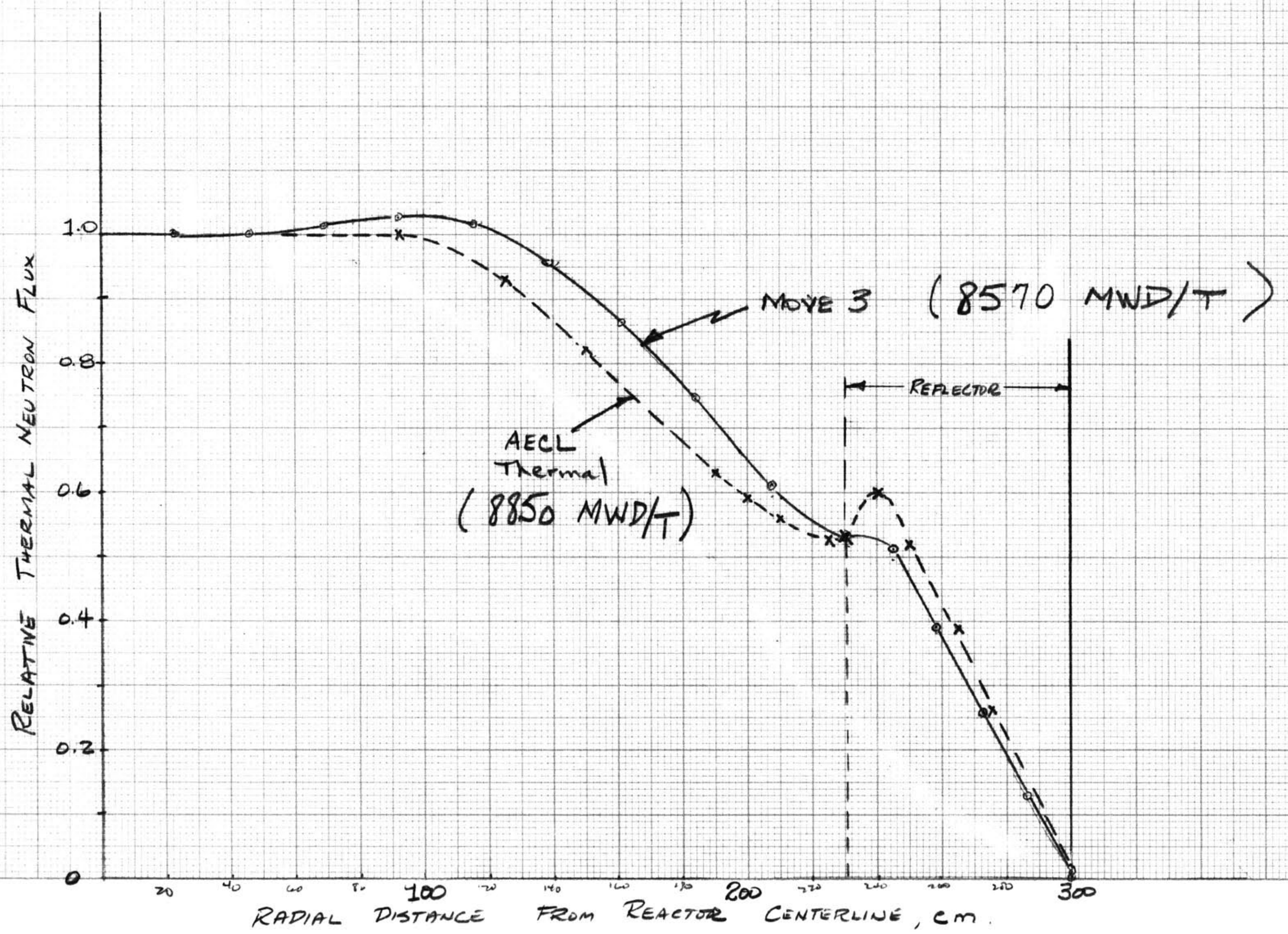
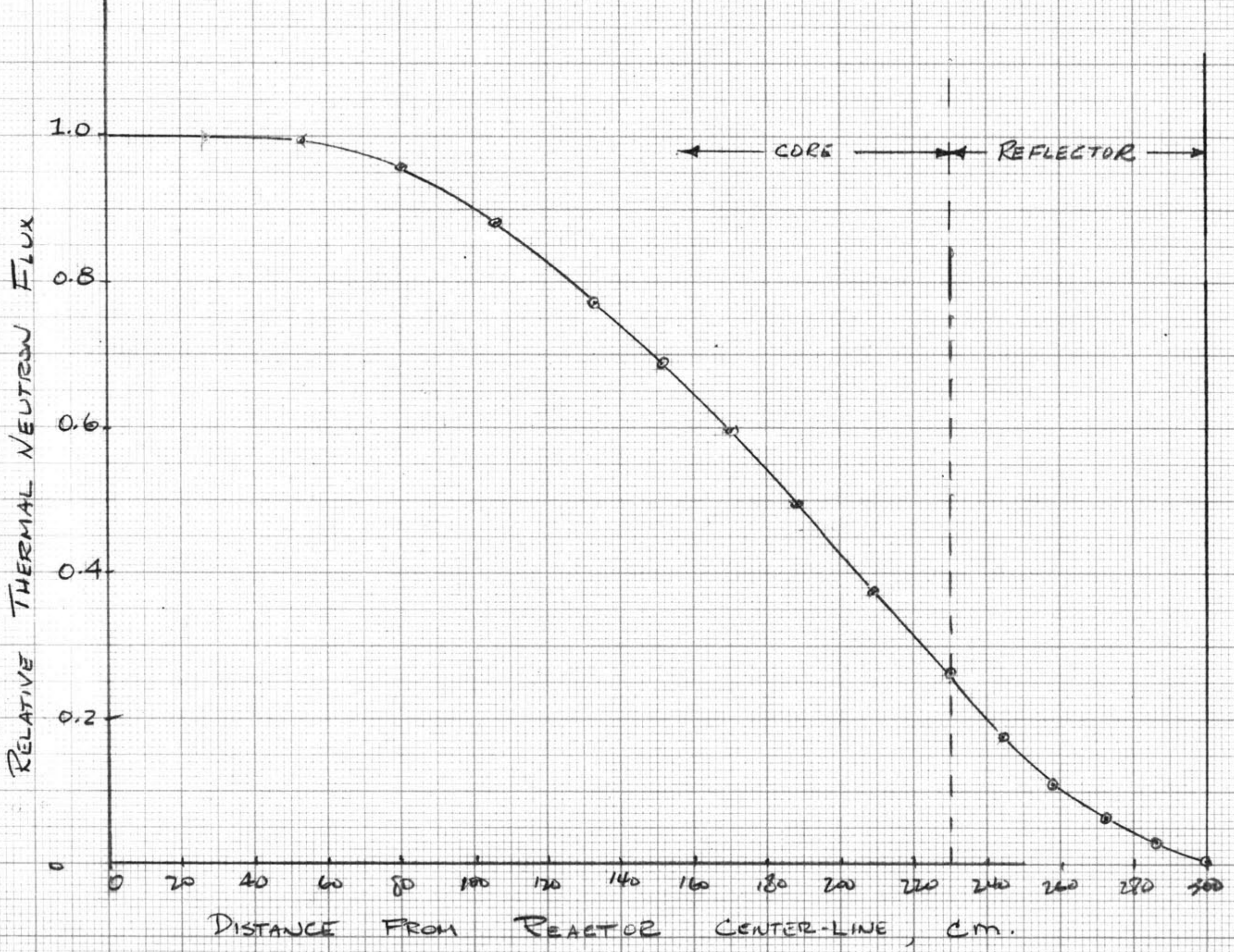


FIGURE 5

RADIAL THERMAL FLUX DISTRIBUTION
FOR BATCH LOADED, NAT. U, CANDU REACTOR.
CALCULATED ACCORDING TO EQUATION B.1
USING SOLUBLE POISON IN MODERATOR
FOR CONTROL.



the reflector effectiveness would probably be even greater for a more highly enriched core fuel.

Results were also obtained for batch and discontinuous outin fueling patterns for the reflected CANDU reactor using soluble poison for reactivity control. Burnup predictions did not disagree greatly with those of MOVE I using the reflector savings model. However, the flux distributions calculated were unsatisfactory since neither a "bump" nor inflection occurred in the reflector thermal flux. Figure 5 shows the CANDU thermal flux at the beginning of batch irradiation for $C_{Reactor} = 1.00000$ after 21 outer iterations. The error criterion for the criticality factor C was kept small (e.g. $\epsilon = 10^{-5}$) thus many outer iterations were required on the poison cross-section to converge within this criterion.

It appeared that since the poison estimate cross-section, Σ_w, is part of the fast leakage term FASK(i,j), and since both Σ_w and FASK(i,j) were adjusted after each outer iteration, then the converged flux distribution obtained after many outer iterations was not entirely based on the physical properties of the core and reflector. Instead, the distribution became a function, in some complicated way, of the many previously adjusted values of FASK(i,j) and Σ_w such that the results appeared similar to one that would be obtained from a one-group calculation.

It is possible that reasonably correct flux distributions might be obtained by carrying out the so-called inner iterations on not only the thermal flux, ϕ, but also on the fast leakage term FASK. With this method, using a specific poison estimate and old values of FASK, converged solutions would be obtained for the thermal flux, ϕ. Then using these values of ϕ, new values of FASK would be calculated. Rather than now calculating a new neutron balance and thereby a new poison estimate as was done in MOVE III, the calculations would instead return to the spatial distribution subroutine. Iterations for the thermal flux would then be carried out using new values of FASK but still the same value of poison Σ_W. These inner iterations would be repeated until both ϕ and FASK for that particular Σ_W had converged to within a small quantity of the exact or equilibrium values. Now a neutron balance would be calculated and the criticality factor, C, compared to unity. If C was not within ϵ' of unity, a compensating adjustment to the poison cross-section would be made and the inner iterations upon ϕ and FASK would be repeated. It appears that in this way, the inability of MOVE III to obtain a thermal flux "bump" for poisoned reactors might be eliminated.

However, this method was not tried and instead it was decided to reprogram the spatial flux solution subroutine, SPACFX, using the theory outlined in Section A.3 above. In this way the fast flux leakage term was incorporated into the five point difference formulation and became a more direct part of the inner iterations. However, the results obtained with this latter approach were even worse than those calculated with the previous method. Now, for the CANDU reactor with reflector, no satisfactory results for flux distribution or fuel burnup were obtained, either for bidirectionaly fueled or discontinuous OUTIN. The cause of this inability to converge to a physically realistic solution is in the formulation of the fast flux factor, Q(i,j) defined by Eqn. (3.2) used in SPACFX, where

$$Q(i,j) = \frac{X(i,j)}{\phi(i,j)}$$

Due to the program logic, it is necessary that in the first outer iteration, the X(i,j) must be calculated by Eqn. (2.35) using initial guesses to the flux distribution and the poison cross-section Σ_W. Thus, they may be considerably in error especially in the reflector. Thus the converged solution to the flux distribution, which uses the X(i,j) as coefficients, may also be in error. For mesh points in the reflector near the outer boundary, the fluxes become very small and sometimes negative. Since negative fluxes are not physically allowable they are arbitrarily set equal to zero at the end of the SPACFX subroutine. Thus the next time control reaches SPACFX, the Q(i,j) become very large for those points which have very small fluxes. The Q(i,j) for points with zero fluxes become zero. This in turn, during the course of the inner iteration, increases the thermal flux at these and neighboring points to unrealistically high values and the unstable situation has begun. Successive outer iterations now make the solution even worse until the program stops upon reaching a s ituation that prevents it from calculating some of the required parameters.

Varying the damping factor, for , DAMPIN, or that for X(i,j) the fast flux factor, FLDAMP, does not appreciably change the results. Non-physical flux distributions appear first in the reflector and then spread throughout the whole reactor. It has thus been decided to abandon this approach.

IV. CONCLUSIONS

The method for obtaining the thermal flux distribution using the condensed two group equation appears to be inadequate, at least in its form described above, to obtain solutions for reflected reactors, in which the reflector is treated explicitly as a separate region. Attempts to stabilize the behavior of the iterations by the use of various damping factors has not resulted in any appreciable improvement. However, stable and correct solutions might be possible if the inner iterations were carried out on both the thermal fluxes and in the fast leakage terms.

V. RECOMMENDATIONS

It is recommended that one final attempt be made to try to utilize the advantages of the condensed two-group formulation of the diffusion equation by reprogramming MOVE III to carry out inner iterations oh both the thermal fluxes and the fast leakage terms, FASK. The neutron balance and subsequent adjustments to the poison cross-section, if any, would then be made using self-consistent, converged values of ϕ and FASK.

However, it is suspected that this approach may give reasonably correct solutions only for certaih cases and that the instability observed previously may reoccur. Therefore it is further recommended that for further studies of reflected reactors, the flux distributions and criticality be obtained with the use of the full two group formulation thereby removing the inherent instability observed with the condensed two-group equation, Flux solutions will be obtained for the fast and thermal flux in the inner iteration, while the critical eigenvalue either as control poison or as ν, the number of neutrons per fission, will be obtained in the outer iteration.

REFERENCES

1. N.B. McLeod, M. Benedict, et. al., "The Effect of Fuel and Poison Management on Nuclear Power Systems". USAEC Report No. NYO-9715, September, 1961.

2. J.A. Sovka and M. Benedict, "Modifications to Fuel Cycle Code "FUELMOVE". USAEC Report No. NYO-9717, April, 1963.

3. L.A. Hageman, "Numerical Methods and Techniques used in the Two-Dimensional Neutron-Diffusion Program PDQ-5". USAEC Report WAPD-TM-364, February, 1963.

4. D.G. Hurst and W.J. Henderson, "The Effect of Flux Flattening on the Economics of Heavy Water Moderated Reactors", AECL-949, December, 1959.