DRAFT (12/1/09)

PARCS v3.0
U.S. NRC Core Neutronics Simulator

THEORY MANUAL

T. Downar

Y. Xu V. Seker

Department of Nuclear Engineering and Radiological Sciences

University of Michigan

Ann Arbor, MI 48109

N. Hudson

RES / U.S. NRC

Rockville, Md

TABLE OF CONTENTS

1. INTRODUCTION

The solution of a spatial kinetics problem encompasses a wide range of physical phenomena such as neutron transport, and the generation and decay of neutron precursors. Numerous computational methods have been developed and implemented in PARCS to model each phenomenon properly and have been formulated such that they ensure solution accuracy as well as provide high computational efficiency. This manual provides the theoretical basis for the major computational methods implemented in PARCS.

Spatial kinetics calculations involve the solution of the eigenvalue problem and the time-dependent neutron transport equations. The theoretical basis for both these problems is presented in Chapter 2. The first step in the solution of either process is to discretize the balance equations in both time and space. For the temporal discretization, the theta-method with exponential transformation is employed in PARCS along with a second-order analytic precursor integration technique. The temporal discretization scheme allows sufficiently large time step sizes even in severe transients involving super-prompt critical reactivity insertion. For spatial discretization, the efficient nonlinear nodal method is employed in which the coarse mesh finite difference (CMFD) problems and the local two-node problems are repetitively solved during the course of the nonlinear iteration. It has been well-documented that the nonlinear nodal methods are more efficient than the conventional response matrix formulation because of lower memory requirements and the efficient linear system solvers available for the CMFD problems.[1],[2] It is particularly advantageous in the transient calculation because the two-node calculation need not be performed at every time step, leading to a very efficient transient calculation. The temporal and spatial differencing of the spatial kinetics equation results in a fixed source type of the problem at every time step. The solution of a transient fixed source problem (TFSP) consists of the simultaneous solutions of the CMFD and two-node problems. The CMFD problem involves a linear system with a block penta-diagonal matrix in three-dimensional problems. In PARCS, the solution of the linear system is obtained using a Krylov subspace method because it is more efficient and robust than the classical iterative methods and also because it is easier to achieve a coarse grain parallelism. The spatial discretization methods and the Krylov CMFD solver are both presented in Chapter 3.

The two-node problems are solved to correct for the discretization error in the nodal interface current resulting from the finite difference approximation in a coarse mesh structure. They can be solved using any one of a number of so-called advanced nodal diffusion methods. In PARCS, the nodal expansion method (NEM)[3] and the analytic nodal method (ANM)[4] can both be used to obtain the two-node solution. Because the NEM can provide a more robust and faster solution than ANM, it has been preferred in many other reactor physics codes even though for some applications it can be less accurate. ANM is used as the primary nodal solver in PARCS because of the improvements which were used to produce a robust solution regardless of the nodal condition. Robustness was insured by providing a hybrid ANM/NEM scheme[5] in which "near critical" two-node problems are solved with the NEM while the ANM is used in the rest of the two-node problems. The two-node ANM and the hybrid two node method is described in Chapter 4.

Also described in Chapter 4 is the pin power reconstruction method used in PARCS. Fuel pin power information is essential for the detailed safety assessment of a core loading pattern. In order to obtain the pin power distribution from the PARCS nodal solution, a modulation method is

used that involves reconstruction of pin powers from the nodal flux solution for the homogenized fuel assembly nodes and heterogeneous form functions from the lattice calculations[5],[6],[7]. A surface current constrained two-dimensional analytic function expansion method and two-group form functions are employed used in PARCS for the reconstruction.

For some applications such as the analysis of cores partially loaded with MOX fuel, the accuracy achievable with diffusion theory may not be adequate. Sevearl P_N transport methods were considered for PARCS and a multigroup Simplified P_3 (SP_3) kernel[8] was chosen for implementation in PARCS for both the fine mesh (pin by pin) finite difference and nodal NEM discretizations. The theoretical basis for the SP_3 static and time-dependent kernel is presented in Chapter 6.

PARCS was also extended to handle non-orthogonal geometries for applications such as the VVER. A hexagonal multigroup nodal diffusion kernel was implemented in PARCS based on the Triangular Polynomial Expansion Method (TPEN)[9]. The basis for TPEN, as well as the acceleration technique used to improve the computational efficiency, is presented in Chapter 7.

Although the primary spatial kinetics methods are described in Chapters 2 through 7, there are other methods which need to be introduced to complete the calculation which are presented in Chapters 8 and 9. For example a control rod cusping correction method was implemented to minimize the error when control rods are partially inserted into a coase computational node. The cusping correction is performed by solving a three-node problem for the intranodal flux using the fine mesh finite difference method. Various other calculations are also discussed in Chapter 9, to include the methods used to compute the adjoint flux, which is necessary for the reactivity edits during transient calculations. Additionally, the decay heat and Xenon/Samarium treatment are discussed in Chapter 9, along with a steady-state critical boron concentration search.

While 3D kinetics is the preferred means of achieving high-fidelity neutronics simulation, it is sometimes necessary to complement 3D kinetics in several practical situations with a 1-D kinetics solution. These include: 1) when significant reductions in computing time are important, 2) when detailed neutronics data is not available for 3D kinetics calculations (e.g. input decks of codes such as TRAC-B), and 3) when a convenient means of performing sensitivity analyses on safety parameters is required (e.g. fuel enthalpy deposit vs. ejected rod worth). The 1-D kinetics was implemented in PARCS using an innovative current conservation factor (CCF) method which guarantees that the same axial neutron currents as the 3D reference values are obtained in the 1D solution. The 1-D kinetics methods are also described in Chapter 9. Finally, in Chapter 8 the methods introduced in PARCS to perform fuel depletion analysis are presented. Currently, only a macroscopic depletion scheme has been implemented in PARCS which utilizes burnup dependent cross sections provided in the appropriate format by the program GENPMAXS. Chapter 8 also provides a description of the multi-cycle depletion methodology implemented in PARCS.

All the computational methods introduced above were integrated into PARCS in a modular form. To perform a given task, these computational modules must be properly coordinated. Since the global performance of the code is largely dependent on the coordination scheme as well as on the individual methods themselves, efforts were made to formulate an efficient coordination scheme for each type of task. The calculational logic control for both steady-state and the transient calculations is described in Chapter 10. Several auxialiary code functions such as the control

rod scram logic, restart capability, as well as the neutronic to TH mapping, are also described in Chapter 10.

2. BASIC NEUTRONICS PROBLEMS

Two basic types of neutronics problems are solved in PARCS, the eigenvalue problem and the fixed source problem. The eigenvalue problem is solved during the steady-state initialization prior to a transient, as well as during fuel depletion analysis. The most common fixed source application of PARCS is to the transient fixed source for the spatial kinetics problem. This chapter reviews the essential details of these two types of problems.

2.1 The Eigenvalue Problem

For a computational node m in Cartesian geometry, the time-dependent behavior of the neutron flux is governed by the following nodal balance equations, given in terms of the node-average flux (ϕ_g^m), precursor density (C_k^m), the surface average net current ($J_{gw}^{m\pm}$), and other standard notations:

$$\frac{1}{v_g^m}\frac{d\phi_g^m}{dt} = \frac{1}{k_{eff}}\chi_{pg}\sum_{g=1}^{G}\nu_{pg}\Sigma_{fg}^m\phi_g^m + \chi_{dg}\sum_{k=1}^{K}\lambda_k C_k^m + \sum_{g'=1}^{G}\Sigma_{g'g}^m\phi_{g'}^m - \sum_{u=x,y,z}\frac{1}{h_u^m}(J_{gu}^{m+} - J_{gu}^{m-}) - \Sigma_{tg}^m\phi_g^m \quad (2.1)$$

and

$$\frac{dC_k^m}{dt} = \frac{1}{k_{eff}}\sum_{g=1}^{G}\nu_{dgk}\Sigma_{fg}^m\phi_g^m - \lambda_k C_k^m \quad . \quad (2.2)$$

Here the plus and minus superscripts of J represent the positive and negative side surfaces of node m in the u-direction, the subscripts p and d stand for prompt and delayed neutrons, and G and K are the numbers of neutron energy and delayed neutron precursor groups, respectively. For the steady-state problem, the prompt and delayed neutron sources are combined and the time dependent eigenvalue k_{eff} is introduced to handle the transients that start from a noncritical initial state. The eigenvalue is determined during the initial steady-state calculation and remains constant during the transient calculation. All the transient calculations require an initial steady-state calculation to initialize the core conditions. Although it is possible to initialize the core condition by solving a steady-state fixed source problem, most transient analyses are based on an initial condition achieved by an eigenvalue calculation because the external source is meaningful only for low power physics test conditions.

In the steady-state calculation, the time derivative terms appearing in the LHS of Eq. (2.1) and Eq. (2.2) become zero and there is no need for making a distinction between prompt and delayed neutrons. The discretized neutron balance equations derived for a transient time step can be converted into steady-state equations by taking the time step size and/or the neutron velocity to be infinite. The linear system for the steady-state can then be converted into the usual form of an eigenvalue problem by moving the fission terms to the RHS in Eq. (2.1):

$$M\phi = \lambda F\phi \equiv \frac{1}{k_{eff}}F\phi \quad (2.3)$$

where the migration matrix M consists of all the nonfission terms in Eq. (2.1) while the fission matrix F consists of the fission terms. The eigenvalue k_{eff} needs to be determined for a nontrivial solution of Eq. (2.3).

The eigenvalue problem given by Eq. (2.3) can be solved by the fission source iteration method which involves the solution of the following source problem:

$$M\phi^{n+1} = s^n \equiv \frac{1}{k_{eff}^n} F\phi^n \tag{2.4}$$

To accelerate the fission source iteration, the Chebyshev polynomial method is used in many codes[10]. However, the Wielandt eigenvalue shift method[11] is employed in PARCS because the Krylov CMFD solver developed for the solution of the transient fixed source problem (to be discussed in Chapter 3) can then be used with only minor corrections for the eigenvalue calculation.

The Wielandt shift method involves solving the following equation which is obtained by subtracting a fission source from both sides of Eq. (2.3):

$$\left(M - \frac{1}{k_s} F\right)\phi = \left(\frac{1}{k_{eff}} - \frac{1}{k_s}\right) F\phi \tag{2.5}$$

The fission source iteration for this case can then be cast into:

$$\left(M - \frac{1}{k_s^n} F\right)\phi^{n+1} = \tilde{s}^n = \left(\frac{1}{k_{eff}^n} - \frac{1}{k_s^n}\right) F\phi^n \tag{2.6}$$

where

$$k_s^n = k_{eff}^n + \delta k \tag{2.7}$$

This can be rewritten as:

$$k_A^n \psi^{n+1} = FA^{-1}\psi^n \tag{2.8}$$

where

$$A \equiv M - \frac{1}{k_s^n} F \quad , \quad \psi^n \equiv F\phi^n \tag{2.9}$$

and

$$\frac{1}{k_A^n} \equiv \frac{1}{k_{eff}^n} - \frac{1}{k_s^n} \tag{2.10}$$

Note that k_A is the eigenvalue of $F^{-1}A$ for a fixed value of k_s^n.

The iteration scheme given by Eq. (2.8) corresponds to the power method and the eigenvalue is updated by the following relation:

$$k_A^{n+1} = k_A^n \frac{\langle \psi^{n+1}, \psi^{n+1} \rangle}{\langle \psi^{n+1}, \psi^n \rangle} \tag{2.11}$$

This expression can be used to derive an expression for the $(n+1)$-th iterate of the eigenvalue. First,

$$\frac{1}{k_A^{n+1}} \equiv \frac{1}{k_{eff}^{n+1}} - \frac{1}{k_s^n} = \frac{\gamma}{k_A^n} \tag{2.12}$$

where

$$\gamma = \frac{\langle \psi^{n+1}, \psi^{n} \rangle}{\langle \psi^{n+1}, \psi^{n+1} \rangle} \tag{2.13}$$

The new estimate of k_{eff} is then obtained as:

$$k_{eff}^{n+1} = \left[\frac{\gamma}{k_A^n} + \frac{1}{k_s^n} \right]^{-1} = \left[\frac{\gamma}{k_{eff}^n} + \frac{1-\gamma}{k_s^n} \right]^{-1} \tag{2.14}$$

The effectiveness of the Wielandt shift method depends on the choice of the eigenvalue shift, δk in Eq. (2.7). In general, fewer iterations are necessary to converge the (outer) iteration scheme given in Eq. (2.6) as δk becomes smaller. On the other hand, the source problem, Eq. (2.6), becomes less diagonally dominant with a smaller δk so that the number of (inner) iterations to solve Eq. (2.6) increases. Therefore, there is no guarantee that a smaller δk is always better and there is an optimum value of δk depending on the problem. Empirically, it was found that δk=0.04 performs well for a wide range of LWR problems.

2.2 The Transient Fixed Source Problem

To solve the time dependent form of the nodal balance equation (2.1) for Light Water Reactor applications, several approximations are normally made to include the use of two energy groups and the following simplifications:

(1) $\chi_{p1} = \chi_{d1} = 1.0$, $\chi_{p2} = \chi_{d2} = 0.0$

(2) no dependence of the delayed neutron precursor yields on neutron energy,

(3) $\nu_{dgk}\Sigma_{fg}^m = \beta_k^m \nu\Sigma_{fg}^m$, $\nu_{pg}\Sigma_{fg}^m = (1-\beta^m)\nu\Sigma_{fg}^m$ where $\beta^m \equiv \sum_{k=1}^{K} \beta_k^m$, and

(4) no upscattering ($\Sigma_{21}^m = 0$).

The two-group kinetics equations can then be obtained from Eq. (2.1) as follows:

$$\frac{1}{v_g^m}\frac{d\phi_g^m}{dt} = R_g^m \equiv \begin{cases} (1-\beta^m)\psi^m + S_d^m - L_1^m - \Sigma_{r1}^m \phi_1^m, & g = 1 \\ \Sigma_{12}^m \phi_1^m - L_2^m - \Sigma_{r2}^m \phi_2^m, & g = 2 \end{cases} \tag{2.15}$$

and

$$\frac{dC_k^m}{dt} = \beta_k^m \psi^m - \lambda_k C_k^m \tag{2.16}$$

where the total fission source term ψ, delayed neutron source S_d, and group leakage L_g are defined as:

$$\psi^m \equiv \frac{1}{k_{eff}} \sum_{g=1}^{2} \nu\Sigma_{fg}^m \phi_g^m, \qquad S_d^m \equiv \sum_{k=1}^{K} \lambda_k C_k^m \tag{2.17}$$

$$L_g^m \equiv \sum_{u=x,\,y,\,z} L_{gu}^m \quad , \qquad L_{gu}^m \equiv \frac{1}{h_u^m}(J_{gu}^{m+} - J_{gu}^{m-}) \tag{2.18}$$

A transient fixed source problem (TFSP) is formulated from the two-group kinetics equations by applying temporal discretization methods involving the theta method and a second order precursor integration technique and also by applying a spatial discretization method involving a coarse mesh finite difference (CMFD) formulation and a two-node nodal method. The resulting transient fixed source problem will contain only node average fluxes as the unknowns. The theta time discretization method is described here, and the spatial discretization method is described in the following Chapter.

One of the fundamental methods to discretize Eq. (2.15) in time is the theta method which results in the following equation at a time point *n* with a time step size of $\Delta t_n = t_n - t_{n-1}$:

$$\frac{\phi_g^{m,n} - \phi_g^{m,n-1}}{v_g^m \Delta t_n} = \theta R_g^{m,n} + (1-\theta) R_g^{m,n-1} \Rightarrow \frac{\phi_g^{m,n}}{\theta v_g^m \Delta t_n} - R_g^{m,n} = \frac{\phi_g^{m,n-1}}{\theta v_g^m \Delta t_n} + \bar{\Theta} R_g^{m,n-1} \tag{2.19}$$

where $\bar{\Theta} \equiv \frac{1}{\theta} - 1$ for $0 < \theta \le 1$. It is assumed here that all terms at time point (TP) *n-1* are known when solving for the terms at TP *n*.

The well-known Crank-Nicholsen scheme realized with θ=0.5 is second-order accurate and permits sufficiently large time step sizes in most transient calculations. The accuracy of the theta method, however, can be enhanced in the events where an exponential variation is anticipated, such as the hot-zero-power (HZP) control rod ejection transient in a PWR, by applying the theta method after an exponential transform:

$$\phi_g^m(t) = e^{\alpha_g^{m,n} t} \tilde{\phi}_g^m(t) \qquad for \qquad t \in [t_{n-1}, t_n] \tag{2.20}$$

Note here that the transformed flux ($\tilde{\phi}_g^m(t)$) can be a much more slowly varying function of time than the original flux because it is possible to make the exponential function carry the rapidly varying component by a proper choice of the inverse period ($\alpha_g^{m,n}$).

With the exponential transform, Eq. (2.15) is converted to

$$\frac{1}{v_g^m} \frac{d\tilde{\phi}_g^m}{dt} = \left(R_g^m - \frac{\alpha_g^{m,n}}{v_g^m} \phi_g^m \right) e^{-\alpha_g^{m,n} t} \quad . \tag{2.21}$$

Applying the theta method to Eq. (2.21) yields

$$\left(\frac{1}{\theta v_g^m \Delta t_n} + \frac{\alpha_g^{m,n}}{v_g^m} \right) \phi_g^{m,n} - R_g^{m,n} = \left[\left(\frac{1}{\theta v_g^m \Delta t_n} - \frac{\bar{\Theta} \alpha_g^{m,n}}{v_g^m} \right) \phi_g^{m,n-1} + \bar{\Theta} R_g^{m,n-1} \right] e^{\alpha_g^{m,n} \Delta t_n} \tag{2.22}$$

instead of Eq. (2.19).

The $R_1^{m,n}$ term shown in Eq. (2.15) contains the delayed neutron source at TP *n*, $S_d^{m,n}$, which is coupled to the precursor balance equation. Instead of discretizing the precursor balance equation as well, it is possible to obtain an analytic solution for $S_d^{m,n}$ in terms of the total fission source

at TP n, ψ_n^m, if the time variation of the fission source over the time interval is assumed to be linear or quadratic in time. In the present formulation, a quadratic form is chosen instead of the common linear form because the flux variation associated with θ=0.5 in the theta method is also second order.

Based on the assumption of quadratic fission source variation, the following formula can be derived for the precursor density at TP n in terms of time dependent parameters defined as: $\gamma \equiv \frac{\Delta t_n}{\Delta t_{n-1}}$, $\kappa_k = e^{-\lambda_k \Delta t_n}$, and $\bar{\kappa}_k = 1 - \kappa_k$:

$$C_k^n = \kappa_k C_k^{n-1} + \frac{\beta_k}{\lambda_k}(\Omega_k^{n-2}\psi_{n-2} + \Omega_k^{n-1}\psi_{n-1} + \Omega_k^n \psi_n) \tag{2.23}$$

where

$$\Omega_k^{n-2} = \frac{1}{\lambda_k \Delta t_{n-1}(\gamma+1)}\left(\frac{2\bar{\kappa}_k}{\lambda_k \Delta t_{n-1}} - \gamma(\kappa_k + 1)\right) \quad , \tag{2.24}$$

$$\Omega_k^{n-1} = \frac{1}{\lambda_k \Delta t_{n-1}}\left(\kappa_k + 1 + \frac{\bar{\kappa}_k}{\gamma}\left(1 - \frac{2}{\lambda_k \Delta t_{n-1}}\right)\right) - \kappa_k \quad , \quad and$$

$$\Omega_k^n = 1 - \frac{2}{(\gamma+1)\lambda_k \Delta t_{n-1}} + \frac{\bar{\kappa}_k}{\gamma(\gamma+1)\lambda_k \Delta t_{n-1}}\left(\frac{2}{\lambda_k \Delta t_{n-1}} - 1\right)$$

Here and below the node index m is omitted momentarily for brevity. The delayed neutron source at TP n is then obtained as:

$$S_d^n = \sum_{k=1}^{K} \lambda_k C_k^n = \sum_{k=1}^{K} \kappa_k \lambda_k C_k^{n-1} + \sum_{l=n-2}^{n-1} \sum_{k=1}^{K} \beta_k \Omega_k^l \psi_l + \sum_{k=1}^{K} \beta_k \Omega_k^n \psi_n \equiv \tilde{S}_d^{n-1} + \omega_n \psi_n \tag{2.25}$$

where

$$\tilde{S}_d^{n-1} = \sum_{k=1}^{K} \kappa_k \lambda_k C_k^{n-1} + \sum_{l=n-2}^{n-1} \sum_{k=1}^{K} \beta_k \Omega_k^l \psi_l \tag{2.26}$$

and

$$\omega_n = \sum_{k=1}^{K} \beta_k \Omega_k^n \tag{2.27}$$

Note that Ω_k^n becomes 0 and 1 as Δt_n approaches to 0 and infinity, respectively, and hence ω_n is in the range between 0 and ß and is an increasing function on Δt_n.

By using Eq. (2.25), $S_d^{m,n}$ is now eliminated from $R_1^{m,n}$ so that:

$$R_1^{m,n} = \beta_p^{m,n} \psi_n^m + \tilde{S}_d^{m,n-1} - L_1^{m,n} - \Sigma_{r1}^{m,n} \phi_1^{m,n} \quad . \tag{2.28}$$

where the effective prompt neutron fraction $\beta_p^{m,n}$ is defined as:

$$\beta_p^{m,n} = 1 - \beta^m + \omega_n^m \tag{2.29}$$

Collecting terms at TP *n* on the LHS and terms at TP *n-1* on the RHS, and also substituting ψ_n^m with corresponding flux terms after inserting Eq. (2.28) into Eq. (2.22), the following fixed source problem is obtained:

$$(\Sigma_{r1}^{m,n} - \beta_p^{m,n}\lambda\nu\Sigma_{f1}^{m,n})\phi_1^{m,n} - \beta_p^{m,n}\lambda\nu\Sigma_{f2}^{m,n}\phi_2^{m,n} + L_1^{m,n} = S_1^{m,n} + \tilde{S}_d^{m,n-1} \tag{2.30}$$
$$-\Sigma_{12}^{m,n}\phi_1^{m,n} + \Sigma_{r2}^{m,n}\phi_2^{m,n} + L_2^{m,n} = S_2^{m,n}$$

where

$$\tilde{\Sigma}_{rg}^{m,n} \equiv \frac{1}{\theta v_g^m \Delta t_n} + \frac{\alpha_g^{m,n}}{v_g^m} + \Sigma_{rg}^{m,n} \tag{2.31}$$

$$S_g^{m,n} \equiv \left[\left(\frac{1}{\theta v_g^m \Delta t_n} - \frac{\bar{\Theta}\alpha_g^{m,n}}{v_g^m}\right)\phi_g^{m,n-1} + \bar{\Theta}R_g^{m,n}\right]e^{\alpha_g^{m,n}\Delta t_n} \tag{2.32}$$

and $\lambda \equiv \frac{1}{k_{eff}}$.

Note here that different degrees of sophistication can be achieved in the above equation by choosing different values of θ and $\alpha_g^{m,n}$. For instance, the ordinary theta method can be realized by choosing $\alpha_g^{m,n} = 0$ and the fully implicit method is realized by choosing θ=1 ($\bar{\Theta} = 0$). In the next section, a spatial discretization method is introduced that allows the representation of the leakage terms at TP *n*, $L_g^{m,n}$, in terms of node average fluxes at TP *n*. The time index n will be omitted for brevity in the following section.

3. NUMERICAL SOLUTION METHODS

3.1 Coarse Mesh Finite Difference Method (CMFD)

This chapter will describe the spatial discretization methods used to solve the eigenvalue and transient fixed source problems developed in Chapter 2. The balance equation for each node is coupled to that of the neighboring nodes through the leakage terms. The nodal coupling is resolved by using the nonlinear nodal method in which the interface current between any two nodes is represented in terms of the node average fluxes of the two facing nodes by the following relation:

$$J_{gu}^{m\pm} = \mp\tilde{D}_{gu}^{m\pm}(\phi_g^{m\pm l_u} - \phi_g^m) - \hat{D}_{gu}^{m\pm}(\phi_g^{m\pm l_u} + \phi_g^m) \tag{3.1}$$

where $m+l_u$ is the node index of the neighbor of node m in the positive u direction while $m-l_u$ is for the negative u-direction. Here $\tilde{D}_{gu}^{m\pm}$ is the base nodal coupling coefficient because it represents the first order estimate of the nodal coupling based on the finite difference approximation which is given as:

$$\tilde{D}_{gu}^{m\pm} = \frac{2D_g^{m\pm l_u}D_g^m}{D_g^{m\pm l_u}\Delta u_m + D_g^m\Delta u_{m\pm l_u}} \quad . \tag{3.2}$$

On the other hand, $\hat{D}_{gu}^{m\pm}$ is the corrective nodal coupling coefficient which represents a correction term. This correction term forces the interface current obtained by Eq. (3.1) to be the same as that obtained by a higher order nodal coupling method applied in the two-node problem. The determination of the corrective nodal coupling coefficients will be discussed in Chapter 4. For now, it is assumed that $\hat{D}_{gu}^{m\pm}$'s are known.

By using Eq. (3.1), the leakage terms in the transient fixed source problem, Eq. (2.18), become:

$$L_{gu}^m = a_{gu}^{m-}\phi_g^{m-l_u} + a_{gu}^m\phi_g^m + a_{gu}^{m+}\phi_g^{m+l_u} \tag{3.3}$$

where

$$a_{gu}^{m-} \equiv -\frac{1}{h_u^m}(\tilde{D}_{gu}^{m-} - \hat{D}_{gu}^{m-}), \quad a_{gu}^m \equiv \frac{1}{h_u^m}(\tilde{D}_{gu}^{m+} + \tilde{D}_{gu}^{m-} - \hat{D}_{gu}^{m+} + \hat{D}_{gu}^{m-}), \quad a_{gu}^{m+} \equiv -\frac{1}{h_u^m}(\tilde{D}_{gu}^{m+} - \hat{D}_{gu}^{m+}) \quad . \tag{3.4}$$

Inserting Eq. (3.3) into Eq. (2.30) yields a fixed source problem that contains only node average fluxes as unknowns. The two group fixed source problem can then be represented compactly using matrix notation as:

$$A_n\phi_n = s_n \tag{3.5}$$

where

$$A_n \equiv col(\alpha_n^1, \ldots, \alpha_n^M) \in \Re^{2M,2M} \quad , (\phi_n, s_n) \in \Re^{2M} \tag{3.6}$$

$$\phi_n \equiv col\left(\begin{bmatrix}\phi_1^{1,n}\\ \phi_2^{1,n}\end{bmatrix}, \ldots, \begin{bmatrix}\phi_1^{M,n}\\ \phi_2^{M,n}\end{bmatrix}\right) \quad , s_n \equiv col\left(\begin{bmatrix}S_1^{1,n}+\tilde{S}_d^{1,n-1}\\ S_2^{1,n}\end{bmatrix}, \ldots, \begin{bmatrix}S_1^{M,n}+\tilde{S}_d^{M,n-1}\\ S_2^{1,n}\end{bmatrix}\right) \tag{3.7}$$

$$\alpha_n^m \equiv row(\ldots a_z^{m-,n} \ldots a_y^{m-,n} \ldots a_x^{m-,n}, d^{m,n}, a_x^{m+,n} \ldots a_y^{m+,n} \ldots a_z^{m+,n} \ldots) \in \Re^{1,2M} \tag{3.8}$$

$$a_u^{m\pm,n} \equiv \begin{bmatrix} a_{1u}^{m\pm,n} & 0 \\ 0 & a_{2u}^{m\pm,n}\end{bmatrix}, \qquad d_n^m \equiv \begin{bmatrix} d_1^{m,n} - \beta_p^m \lambda \nu \Sigma_{f1}^{m,n} & (-\beta_p^m \lambda \nu \Sigma_{f2}^{m,n}) \\ -\Sigma_{12}^{m,n} & d_2^{m,n}\end{bmatrix} \in \Re^{2,2} \tag{3.9}$$

and

$$d_g^{m,n} \equiv \Sigma_{rg}^{m,n} + \frac{1}{\theta v_g^m \Delta t_n} + \frac{\alpha_g^{m,n}}{v_g^m} + \sum_{u=x,y,z} a_{gu}^{m\pm,n} \tag{3.10}$$

with M being the total number of computational nodes.

The linear system represented by Eq. (3.5) through Eq. (3.10) is referred to as a CMFD transient fixed source problem and can be solved by any iterative linear system solution method. In PARCS, a Krylov subspace method, BiCGSTAB[12]and GMRES, preconditioned with the BILU3D preconditioner is employed to solve the CMFD problem. In the following section, the Krylov CMFD solution method will be described in detail. It is then followed in Chapter 4 by the derivation of a two-node kernel which is used to determine the corrective nodal coupling coefficients used in Eq. (3.1).

3.2 Solution of the Linear Systems

Krylov subspace methods constitute a class of the most up-to-date linear system solution methods. In these methods, a linear system $Ax=b$ is solved by building the best approximate solution in a subspace of dimension k which is defined as:

$$K_k(r_0) = \{r_0, Ar_0, A^2 r_0, \ldots, A^{k-1} r_0\} \quad , \quad k \le n \tag{3.11}$$

where r_0 is the initial residual vector defined by $r_0 = b - Ax_0$, n is the dimension of the coefficient matrix, A, and k is the iteration index.

There are various algorithms in the Krylov subspace methods which differ in the manner of building the approximate solution in each subspace. The most basic Krylov solver is the conjugate gradient (CG) method[13] developed in 1950's. Since the CG method is applicable to only symmetric positive definite (SPD) linear systems, there have been many variations of this method developed during the last several decades to extend it to nonsymmetric linear systems. One of the extensions of CG for nonsymmetric systems is the Bi-Conjugate Gradient Stabilized (BiCG-STAB) algorithm. This algorithm is one of two linear solver options used in PARCS because of its proven computational efficiency for solving the CMFD steady-state and transient fixed source problems. A somewhat less efficient Krylov subsapce method, the Generalized Minimum Residual Method (GMRES), has also been implemented in PARCS as an alternative because of its superior stability (e.g. guaranteed monotonic decreasing residual). .In the following section, the

BiCGSTAB and GMRES algorithms will be described and their computational requirements are analyzed.

In general, a Krylov subspace method requires *preconditioning* to achieve the most efficient performance. Preconditioning transforms the original linear system into a new one such that the Krylov subspace method applied to the new system produces a better sequence of approximate solutions, requiring a fewer number of iterations for convergence. When applied to a preconditioned system, the Krylov subspace method involves solutions of a preconditioner equation in the form of $Pz=r$ where P is the preconditioner, as will be shown in Section 3.2.1. Because of the need to solve the preconditioner equation, the computational work required at each iteration increases in a preconditioned Krylov subspace algorithm. If the benefit due to the reduction in number of iterations is larger than the additional work associated with preconditioning, the total computational work can be reduced. With proper preconditioning, it is possible to significantly reduce the total computational work of a Krylov subspace method so that it performs much better than the classical methods such as SOR and CCSI. In Section 3.2.3, an efficient preconditioner is constructed for the solution of three-dimensional CMFD problems such that it takes advantage of the diagonal dominance associated with the use of a coarse spatial mesh structure. The solution of the preconditioner equation is then described in Section 3.2.6.

3.2.1 Preconditioned BiCGSTAB Algorithm

A form of the preconditioned BiCGSTAB is reproduced below from Reference [2].

$r_0=b-Ax_0$ for an initial guess x_0
$\rho_0=\alpha=\omega_0=1$
$v_0=p_0=0$
for i=1,2,3...
  $\rho_i=(r_0,r_{i-1})$
  $\beta=(\rho_i/\rho_{i-1})*(\alpha/\omega_{i-1})$
  $p_i=r_{i-1}+\beta(p_{i-1}-\omega_{i-1}v_{i-1})$
  Solve $\boldsymbol{Py=p_i}$ for y
  $v_i=Ay$
  $\alpha=\rho_i/(r_0,v_i)$
  $s=r_{i-1}-\alpha v_i$
  Solve $\boldsymbol{Pz=s}$ for z
  $t=Az$
  $\omega_i=(t,s)/(t,t)$
  $x_i=x_{i-1}+\alpha y+\omega_i z$
  if x_i is converged, then quit.
  $r_i=s-\omega_i t$
end

In the above algorithm, the variables represented by Greek characters are scalar whereas the lowercase and uppercase characters are for the vector and matrix, respectively. The bold part is for the solution of the preconditioner equation. Note that this algorithm requires 4 vector inner products, 2 matrix-vector products, 2 solutions of preconditioner equation, and 4 vector updates.

3.2.2 Preconditioned GMRES Algorithm

The algorithm of the preconditioned GMRES method used in PARCS is as follows:

$r = b\text{-}Ax_0$ for an initial guess x_0
$\varphi = \| r\|_2$
for $i = 1,2,3\ldots$
  $v_i = r/\varphi$
  $r = A\boldsymbol{P}^{-1} v_i$
  for $j = 1, \ldots i$
    $\eta_{j,i} = (r, v_j)$
    $r = r - \eta_{j,i} v_j$
  end
  $\varphi = \eta_{i+1,i} = \| r\|_2$
  $H = [\eta_{j,i}]_{i+1 \times i}$
  compute $\rho = \min\|\beta e_1 - Hy\|_2$
  exit when $\rho <$ tolerance
end
compute y the minizer of $\|\beta e_1 - Hy\|_2$
$x_0 = x_0 + \boldsymbol{P}^{-1}[v_1 \ldots v_i]y$

In the above algorithm, the variables represented by Greek characters are scalar quantities whereas the lowercase and uppercase characters are for the vector and matrix quantities, respectively. The bold P is for the preconditioner. Note that this algorithm requires one matrix-vector product per step, but the number of vector inner products increases each iteration.

3.2.3 Construction of a Preconditioner for CMFD Problems

The preconditioner can be viewed as an approximation of the original matrix and it should be constructed such that solving the preconditioner equation, $Pz=r$, is much easier than solving the original linear system, $Ax=b$ since the preconditioner equation needs to be solved in every iteration step. In general, a preconditioned Krylov algorithm converges faster as the preconditioner better approximates the original matrix, or in other words, as the condition number of $P^{-1}A$ becomes smaller. Therefore an effective preconditioner is to be constructed such that it approximates the original matrix as closely as possible, yet the computational work to solve the preconditioner equation is as small as possible, so that the total computational work for convergence is minimized. One of the methods for constructing an efficient preconditioner is the incomplete LU factorization.[14] The blockwise incomplete LU (BILU) preconditioner has been widely used in two-dimensional problems. In the following subsections, the BILU factorization is extended to a three-dimensional problem involving neutron diffusion in a coarse mesh structure.

3.2.4 Block Tridiagonal Structure

The CMFD transient fixed source problem represented by Eq. (2.25) through Eq. (2.30) has a three level block tridiagonal structure in a three-dimensional problem employing a plane-wise ordering. It can be rewritten as (omitting the time point index n):

$$A^{(3)}\phi = s \tag{3.12}$$

where the submatrix $A^{(d)}$ representing the coefficient matrix in d-dimensions is recursively defined as:

$$A^{(d)} \equiv \begin{bmatrix} A_1^{(d-1)} & U_1^{(d)} & & & & \\ L_2^{(d)} & A_2^{(d-1)} & U_2^{(d)} & & & \\ & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & \\ & & & L_{n-1}^{(d)} & A_{n-1}^{(d-1)} & U_{n-1}^{(d)} \\ & & & & L_n^{(d)} & A_n^{(d-1)} \end{bmatrix} \tag{3.13}$$

with n being the number of planes (K), rows (J), or columns (I) depending on the value of d. Note that the matrix $A^{(d)}$ is a block tridiagonal matrix while $L^{(d)}$ and $U^{(d)}$ are diagonal matrices consisting of the coupling coefficients, a_u^{m-} and a_u^{m+} defined in Eq. (2.24), respectively. At the lowest dimension, the base submatrix $A_m^{(0)} = d^m$ where d^m is the 2x2 matrix defined by Eq. (2.29).

3.2.5 BILU3D Preconditioner

Consider a block tridiagonal matrix $A^{(d)}$ (the superscript d omitted in the following for brevity) as defined in Eq. (3.13). It is possible to factorize the matrix completely by employing the following recursion relation:

$$\begin{aligned} \Delta_1 &= A_1 \\ \Delta_m &= A_m - L_m \Delta_{m-1}^{-1} U_{m-1} \quad , \quad m = 2..n \end{aligned} \tag{3.14}$$

to obtain the LU factor:

$$A = (L+\Delta)\Delta^{-1}(U+\Delta) = (L+\Delta)(\Delta^{-1}U+I) \tag{3.15}$$

where L and U are the strictly lower and upper triangular parts of A, and Δ is a block diagonal matrix consisting of Δ_m's. The problem associated with the complete factorization, however, is that it is difficult to find the inverse of Δ_m since Δ_m becomes a full matrix. In the following, two approximations are considered to avoid this problem.

Symmetric Gauss-Seidel Factorization

The easiest way to simplify the factorization process is to neglect completely the $L_m \Delta_{m-1}^{-1} U_{m-1}$ term in Eq. (3.14) so that the matrix Δ becomes the block diagonal matrix D which consists of the submatrices A_m's. In that case, the LU factor, denoted by P, becomes a symmetric Gauss-Seidel (SGS) LU factor and can be represented by the sum of the original matrix A and a remainder matrix R as follows:

$$\begin{aligned} P &= (L+D)D^{-1}(U+D) \\ &= L+D+U+LD^{-1}U = A+R \quad . \end{aligned} \tag{3.16}$$

This factorization can be a good approximation only when the norm of the remainder matrix $LD^{-1}U$ is small compared to that of the original matrix A. In a coarse mesh formulation of a typical reactor problem, this condition is satisfied for $A=A^{(3)}$, in which case the axial coupling effect represented by L and U is weaker than the radial coupling effect represented by D. This follows from the fact that the axial mesh size is normally chosen to be larger than the radial mesh size because the reactor is less heterogeneous in the axial than in the radial direction. The larger axial mesh makes the L and U small compared to D, and thus $LD^{-1}U$ becomes much smaller than A. Therefore it is reasonable to take the symmetric Gauss-Seidel LU factor, P, as the preconditioner which approximates $A^{(3)}$.

Rewriting Eq. (3.16) in terms of the submatrices, the following LU factor is obtained as the global preconditioner:

$$A^{(3)} \approx P = (L^{(3)} + D^{(3)})(I + D^{(3)^{-1}} U^{(3)}) \equiv \hat{L}^{(3)} \hat{U}^{(3)} \tag{3.17}$$

where

$$\hat{L}^{(3)} \equiv \begin{bmatrix} A_1^{(2)} & & & \\ L_2^{(3)} & A_1^{(2)} & & \\ & \ddots & \ddots & \\ & & L_K^{(3)} & A_K^{(2)} \end{bmatrix} \quad \text{and} \quad \hat{U}^{(3)} \equiv \begin{bmatrix} I & A_1^{(2)^{-1}} U_1^{(3)} & & \\ & I & \ddots & \\ & & \ddots & A_{K-1}^{(2)^{-1}} U_{K-1}^{(3)} \\ & & & I \end{bmatrix} \quad . \tag{3.18}$$

The preconditioner equation involving these LU factors can be solved by forward and backward substitution. In the substitution processes it is necessary to solve smaller linear systems involving $A_k^{(2)}$'s which are the coefficient matrices for the planes. Because these matrices are block penta-diagonal matrices which are not easily solved and because the preconditioner equation is to be solved repetitively, it is more appropriate to factorize the $A_k^{(2)}$ matrices.

Blockwise Incomplete LU factorization

The factorization of $A^{(2)}$ (plane index k omitted) by the recurrence relation shown in Eq. (3.14) involves submatrices which have the dimension of two times the number of nodes in the x-direction. To prevent these submatrices from becoming full, some of the elements of Δ_{j-1}^{-1} (the general index m was replaced by the row index j) are neglected such that Δ_j has the same

sparse structure as A_j. By denoting the matrix obtained by neglecting some elements of Δ_{j-1}^{-1} by $\Omega(\Delta_{j-1}^{-1})$, the recursion relation is modified as:

$$\Delta_j = A_j - L_j \Omega(\Delta_{j-1}^{-1}) U_{j-1} \quad , \qquad j = 2..J \tag{3.19}$$

The concern now is how to obtain $\Omega(\Delta_{j-1}^{-1})$. This requires finding the elements of the inverse at the locations where A_j has nonzero elements. Noting that the A_j is a smaller block tridiagonal matrix for a one dimensional row, we need to find the three main diagonal blocks of Δ_{j-1}^{-1}. To find these blocks efficiently without calculating the other unnecessary elements of the inverse, the idea of so called "approximate block inverse method"[11] is used which utilizes the LU factor of a matrix to find some specific entries of the inverse of the matrix.

The LU factorization for Δ_{j-1} can be done completely since it only involves inverses of 2x2 matrices which are easily obtainable. Suppose that the LU factor is available for Δ_{j-1} in the following form in which Γ is a block diagonal matrix consisting of 2x2 matrices, Γ_i's, and H and F are the strictly lower and upper triangular matrices of Δ_{j-1}, respectively:

$$\Delta_{j-1} = (H+\Gamma)\Gamma^{-1}(F+\Gamma) = \begin{bmatrix} \Gamma_1 & & \\ H_2 & \ddots & \\ & \ddots & \\ & H_I & \Gamma_I \end{bmatrix} \begin{bmatrix} \Gamma_1^{-1} & & \\ & \ddots & \\ & & \Gamma_1^{-1} \end{bmatrix} \begin{bmatrix} \Gamma_1 & F_1 & \\ & \ddots & \\ & & F_I \\ & & \Gamma_I \end{bmatrix} . \tag{3.20}$$

By manipulating Eq. (3.20), the following relations are obtained for Δ_{j-1}^{-1} which is denoted by E for brevity:

$$\begin{aligned} \Delta_{j-1}^{-1} &= (H+\Gamma)^{-1} - \Gamma^{-1} F E \\ &= (F+\Gamma)^{-1} - E H \Gamma^{-1} \end{aligned} \tag{3.21}$$

The two terms in the first relation in Eq. (3.21) appear in terms of submatrices as the following equations:

$$(H+\Gamma)^{-1} = \begin{bmatrix} \Gamma_1^{-1} & & & \\ \ddots & \ddots & & \\ \ddots & \ddots & \ddots & \\ \ddots & \ddots & \ddots & \Gamma_1^{-1} \end{bmatrix} \tag{3.22}$$

and

$$\Gamma^{-1} F E = \begin{bmatrix} \Gamma_1^{-1} F_1 E_{21} & \Gamma_1^{-1} F_1 E_{22} & \cdots & \cdots \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \Gamma_{I-1}^{-1} F_{I-1} E_{II-1} & \Gamma_{I-1}^{-1} F_{I-1} E_{II} \\ 0 & 0 & \cdots & 0 \end{bmatrix} . \tag{3.23}$$

Note that the first term as shown in Eq. (3.22) contains no upper triangular blocks. Using Eq. (3.21) through Eq. (3.23), the following recursion relation is derived for the diagonal and upper diagonal blocks of E:

$$\begin{aligned} E_{II} &= \Gamma_I^{-1} \\ E_{i-1\,i} &= -\Gamma_{i-1}^{-1} F_{i-1} E_{ii} \\ E_{i-1\,i} &= \Gamma_{i-1}^{-1}(1 - F_{i-1} E_{ii-1}) \qquad i = I..2 \end{aligned} \quad . \tag{3.24}$$

The lower diagonal blocks $E_{i\ i-1}$ appearing in the above recursion relation can be similarly obtained as follows by using the second relation for E in Eq. (3.21):

$$E_{ii-1} = -E_{ii} H_i \Gamma_{i-1}^{-1} \quad . \tag{3.25}$$

By using the submatrices generated by these recursion relations, $\Omega(\Delta_{j-1}^{-1})$ is now formed as a block tridiagonal matrix, and the blockwise incomplete factorization of $A_k^{(2)}$ can be completed. $A_k^{(2)}$ is now approximated by its BILU factor $P_k^{(2)}$ as:

$$P_k^{(2)} \equiv (L_k^{(2)} + \Delta_k)(I + \Delta_k^{-1} U_k^{(2)}) \approx \hat{L}_k^{(2)} \hat{U}_k^{(2)} = A_k^{(2)} \quad . \tag{3.26}$$

By replacing $A_k^{(2)}$ by $P_k^{(2)}$ in Eq. (3.18) with Eq. (3.26), a preconditioner applicable to a three-dimensional problem is obtained as:

$$P = \hat{L}\hat{U} = \begin{bmatrix} P_1^{(2)} & & & \\ L_2^{(3)} & P_2^{(2)} & & \\ & \ddots & \ddots & \\ & & L_K^{(3)} & P_K^{(2)} \end{bmatrix} \begin{bmatrix} I & P_1^{(2)-1} U_1^{(3)} & & \\ & I & \ddots & \\ & & \ddots & P_{K-1}^{(2)-1} U_{K-1}^{(3)} \\ & & & I \end{bmatrix} \quad . \tag{3.27}$$

This is referred to as the BILU3D preconditioner. In the following section, the solution process of a linear system involving the BILU3D preconditioner is described.

3.2.6 Solution of Preconditioner Equation

The linear system involving the BILU3D preconditioner given in Eq. (3.27) can be solved by three levels of forward and backward substitutions, one for each dimension. At the highest level, the sweep for the forward and backward substitution proceeds in the axial direction. The forward substitution at the highest level requires the solution of the following equations:

$$P_k^{(2)} y_k = s_k - L_k^{(3)} y_{k-1} \quad , \qquad k = 2..K \tag{3.28}$$

while the backward substitution requires the solution of the following equations:

$$P_k^{(2)} \zeta_k = U_k^{(3)} z_{k+1} \quad , \qquad k = K\text{–}1..1 \tag{3.29}$$

where $z_k = y_k - \zeta_k$.

Since $P_k^{(2)}$ involves the BILU factor as given in Eq. (3.26), solving Eq. (3.28) and Eq. (3.29) requires the second level of the forward and backward substitutions which involve the solution of the following equation on each radial plane *k* (shown only for the forward substitution):

$$\Lambda_{j,k} y_{j,k} = \hat{s}_{j,k} - L^{(2)}_{j,k} y_{j-1,k} \quad , \qquad j = 2..J \tag{3.30}$$

where

$$\hat{s}_{j,k} = s_{j,k} - L^{(3)}_{j,k} y_{j,k-1} \quad . \tag{3.31}$$

This equation requires the last level of the forward and backward substitutions which involve the solution of the following equations:

$$\Gamma_{i,j,k} y_{i,j,k} = \tilde{s}_{i,j,k} - H_{i,j,k} y_{i-1,j,k} \quad , \qquad i = 2..I \tag{3.32}$$

where

$$\tilde{s}_{i,j,k} = \hat{s}_{i,j,k} - L^{(2)}_{i,j,k} y_{i,j-1,k} \quad . \tag{3.33}$$

The methods described in this section have been been implemented in PARCS and have been shown to be efficient and stable for a wide range of practical applications.

4. NODAL DIFFUSION METHODS

In the previous two chapters, the corrective nodal coupling coefficient (CNCC), $\hat{D}_{gu}^{m\pm}$, was assumed to be known. The CNCC, however, should be determined during the course of the non-linear iteration which involves the CMFD and the two-node calculations. The two-node calculation is performed for every nodal interface appearing in the problem domain. The goal of a two-node calculation is to determine the nodal interface current based on the node average flux distribution available from the previous CMFD calculation. The CNCC is then determined such that the CMFD interface current obtained by the relation given in Eq. (3.1) is the same as the nodal interface current obtained from the two-node calculation.

In this Chapter, the two-node solution method is derived based on the analytic nodal method (ANM). As shown below, the two-node problem is a one-dimensional problem for which the analytic solution is readily obtainable. The one-dimensional diffusion equation is obtained through the transverse-integration procedure which results in the transverse leakage source on the right hand side of the equation. In the following derivation, it is assumed that the transverse leakage source is known from the previous CMFD calculation. In addition, the node average fluxes are given for both nodes and the resulting two-node solution should produce the same node average fluxes. The following derivation of the two-node ANM kernel will be first given for an eigenvalue problem and then will be extended to the transient problems. The derivations will be limited to the *x*-direction since a similar derivation can be applied to the *y*- and *z*-directions.

4.1 Two Group Analytic Nodal Method

By integrating the three-dimensional steady-state neutron diffusion equation over the transverse plane, the following transverse-integrated equation is obtained:

$$-D_g \frac{d^2\phi_g(x)}{dx^2} + \Sigma_{rg}\phi_g(x) - Q_g(x) = -L_g(x) \tag{4.1}$$

where the source Q and the transverse leakage L terms are defined as:

$$Q_g(x) = \begin{cases} \lambda \sum_{g'=1}^{2} \nu\Sigma_{fg'}\phi_{g'}(x) & , \quad g = 1 \\ \Sigma_{fg'}\phi_{g'}(x) & , \quad g = 2 \end{cases} \tag{4.2}$$

and

$$L_g(x) = \frac{1}{h_y}(J_{gy}^{+}(x) - J_{gy}^{-}(x)) + \frac{1}{h_z}(J_{gz}^{+}(x) - J_{gz}^{-}(x)) \tag{4.3}$$

with $\lambda = \frac{1}{k_{eff}}$.

The common approximation introduced in all the transverse-integrated nodal methods is to assume a quadratic spatial variation of the transverse leakage. This approximation is also

employed in this derivation and the following quadratic form is used for the transverse leakage term:

$$L_g(x) = \bar{L}_g + b_{g1} f_1(\xi) + b_{g2} f_2(\xi) \tag{4.4}$$

where $\bar{L}_g$ is the node average transverse leakage and

$$\xi = \frac{x}{h_x} \quad , \qquad f_1(\xi) = \xi \quad , \qquad f_2(\xi) = 3\xi^2 - \frac{1}{4} \tag{4.5}$$

Note that the polynomial basis functions have the following properties:

$$f_p(\xi)\,d\xi = 0 \quad , \qquad f_1\left(\pm\frac{1}{2}\right) = \pm\frac{1}{2} \quad , \qquad f_2\left(\pm\frac{1}{2}\right) = \frac{1}{2} \quad , \qquad f'_1\left(\pm\frac{1}{2}\right) = 1 \quad , \qquad f'_2\left(\pm\frac{1}{2}\right) = \tag{4.6}$$

The $\bar{L}_g$ and b_{gp} coefficients can be calculated for a given node average flux distribution which is assumed to be known. In addition to the transverse leakage information and the node average fluxes, the reciprocal of the eigenvalue, λ, is also assumed to be known from the previous CMFD calculation. In the following section, the analytic solution of Eq. (4.1) is derived for the two-node problem. The analytic solution will then be used to find the interface current.

4.1.1 The Analytic Solution

The analytic solution of the second order ordinary differential equation given in Eq. (4.1) consists of a homogeneous and a particular solution. In the following two subsections, the homogeneous and particular solution will be derived separately for one of the two nodes, and then will be combined in the next subsection such that the combined solution satisfies the given constraints in both nodes. In addition to the constraint on the node average flux, the continuity of flux and current at the interface will be used as the constraints.

Homogeneous Solution

The homogeneous solution is obtained by solving the homogeneous form of the differential equation, Eq. (4.1), which can be written as:

$$\begin{bmatrix} -D_1\frac{d^2}{dx^2} + (\Sigma_{r1} - \lambda\nu\Sigma_{f1}) & -\lambda\nu\Sigma_{f2} \\ -\Sigma_{12} & -D_2\frac{d^2}{dx^2} + \Sigma_{r2} \end{bmatrix} \begin{bmatrix} \phi_1^H(x) \\ \phi_2^H(x) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad . \tag{4.7}$$

The solution process begins by finding the eigenvalues of Eq. (4.7). Let

$$\frac{d^2\phi_g^H(x)}{dx^2} = -B^2\phi_g^H(x) \quad . \tag{4.8}$$

This results in a homogeneous linear system:

$$A(B^2)\phi^H(x) = 0 \tag{4.9}$$

where $\boldsymbol{A}$ is a 2x2 matrix defined as a function of B^2 as:

$$A(B^2) = \begin{bmatrix} D_1 B^2 + (\Sigma_{r1} - \lambda\nu\Sigma_{f1}) & -\lambda\nu\Sigma_{f2} \\ -\Sigma_{12} & D_2 B^2 + \Sigma_{r2} \end{bmatrix} \tag{4.10}$$

and $\phi^H(x)$ is a two element vector consisting of the homogeneous solutions for each group; $\phi^H(x) = (\phi_1^H(x), \phi_2^H(x))^T$.

For a nontrivial solution the determinant of the coefficient matrix must be zero, i.e.,

$$(D_1 B^2 + \Sigma_{r1} - \lambda\nu\Sigma_{f1})(D_2 B^2 + \Sigma_{r2}) - \lambda\nu\Sigma_{f2}\Sigma_{12} = 0 \tag{4.11}$$

This can be rearranged as:

$$(B^2)^2 + \left(\frac{\Sigma_{r2}}{D_2} + \frac{\Sigma_{r1}}{D_1} - \frac{\lambda\nu\Sigma_{f1}}{D_1}\right) B^2 + (1 - \lambda k_\infty)\frac{\Sigma_{r1}}{D_1}\frac{\Sigma_{r2}}{D_2} = 0 \tag{4.12}$$

where

$$k_\infty = \frac{\nu\Sigma_{f1}}{\Sigma_{r1}} + \frac{\Sigma_{12}}{\Sigma_{r1}}\frac{\nu\Sigma_{f2}}{\Sigma_{r2}} \quad . \tag{4.13}$$

In terms of the following abbreviations,

$$b \equiv \frac{1}{2}(\Sigma_{r1} - \lambda\nu\Sigma_{f1} + \Sigma_{r2}) \tag{4.14}$$

and

$$c \equiv (1 - \lambda k_\infty)\hat{\Sigma}_{r1}\hat{\Sigma}_{r2} = \left(1 - \frac{k_\infty}{k_{eff}}\right)\hat{\Sigma}_{r1}\hat{\Sigma}_{r2} \tag{4.15}$$

where

$$\hat{\Sigma}_{r1} = \frac{\Sigma_{r1}}{D_1} \quad , \quad \hat{\Sigma}_{r2} = \frac{\Sigma_{r2}}{D_2} \quad , \quad \hat{\nu\Sigma}_{f1} = \frac{\nu\Sigma_{f1}}{D_1} \tag{4.16}$$

the two roots of Eq. (4.12) are then obtained as:

$$B^2 \equiv -b \pm \sqrt{b^2 - c} \quad . \tag{4.17}$$

Note that the term inside the square root in Eq. (4.17) is always positive since

$$\begin{aligned} 4(b^2 - c) &= (\Sigma_{r1} - \lambda\nu\Sigma_{f1} + \Sigma_{r2})^2 - 4(\Sigma_{r1}\Sigma_{r2} - \lambda\nu\Sigma_{f1}\Sigma_{r2} - \lambda\nu\Sigma_{f2}\Sigma_{r2}) \\ &= (\hat{\Sigma}_{r1} - \lambda\hat{\nu\Sigma}_{f1} + \hat{\Sigma}_{r2})^2 + 4\lambda\nu\Sigma_{f2}\Sigma_{r2} > 0 \quad , \end{aligned} \tag{4.18}$$

which implies that the roots, B^2, are real numbers. Also, in a thermal reactor in which the following condition is satisfied because of the small value of the fast fission cross section:

$$\frac{D_1}{D_2}\Sigma_{r2} + \Sigma_{r1} \gg \lambda\nu\Sigma_{f1} \quad , \tag{4.19}$$

b is always positive.

Now consider the sign of B^2. Since $b>0$, the two roots of Eq. (4.17) can be written separately as:

$$B_0^2 \equiv -b\left(1-\sqrt{1-\frac{c}{b^2}}\right) \tag{4.20}$$

and

$$B_1^2 \equiv -b\left(1+\sqrt{1-\frac{c}{b^2}}\right) \tag{4.21}$$

The second root, B_1^2, which will be referred to as the first harmonic buckling, is a negative real number while the first root, B_0^2, which will be referred to as the fundamental buckling, is an indefinite real number whose sign depends on the sign of c, which is determined by $1-\lambda k_\infty$. Namely,

$$B_0^2 \begin{cases} >0 & if\ \lambda k_\infty > 1 \quad or \quad k_\infty > k_{eff} \\ =0 & if\ \lambda k_\infty = 1 \quad or \quad k_\infty = k_{eff} \\ <0 & if\ \lambda k_\infty < 1 \quad or \quad k_\infty < k_{eff} \end{cases} \tag{4.22}$$

These two roots are the eigenvalues of the one-dimensional two-group neutron diffusion equations, Eq. (4.1). The fundamental eigenvalue determines the asymptotic flux shape being realized away from the boundaries of a node while the first harmonic eigenvalue governs the boundary effects near the boundary.

There are two eigenfunctions for each eigenvalue, which are the solutions of Eq. (4.8) for the corresponding eigenvalue. They are *sin / cos* or *sinh / cosh* functions depending on the sign of B^2. They can also be *x / 1* if B^2=0. These eigenfunctions form the basis of the functional space for the homogeneous solution.

By introducing the following abbreviations:

$$\kappa \equiv \sqrt{\left|B_0^2\right|} \qquad \text{and} \qquad \mu \equiv \sqrt{\left|B_1^2\right|} \tag{4.23}$$

the homogeneous solution can be represented as follows in terms of the basis which differs depending on the magnitude of k_∞:

$$\phi_g^H(x) \in \begin{cases} \{\sin(\kappa x), \cos(\kappa x), \sinh(\mu x), \cosh(\mu x)\} & , \quad k_\infty > k_{eff} \\ \{x, 1, \sinh(\mu x), \cosh(\mu x)\} & , \quad k_\infty = k_{eff} \\ \{\sinh(\kappa x), \cosh(\kappa x), \sinh(\mu x), \cosh(\mu x)\} & , \quad k_\infty < k_{eff} \end{cases} \tag{4.24}$$

The basis functions above can be represented concisely in terms of the two generic functions defined below:

$$sn(m, \lambda k_\infty, x) \quad \text{or} \quad cn(m, \lambda k_\infty, x) \equiv \begin{cases} \sin(x) \text{ or } \cos(x) & , \quad if\ m=0 \quad and \quad \lambda k_\infty > 1 \\ x \text{ or } 1 & , \quad if\ m=0 \quad and \quad \lambda k_\infty = 1 \\ \sinh(x) \text{ or } \cosh(x) & , \quad if\ m=1 \quad and \quad \lambda k_\infty < 1 \end{cases} \tag{4.25}$$

Here the first argument, *m*, signifies the mode of buckling, not the node index.

The homogeneous solution is then obtained as a linear combination of the basis functions

$$\phi_g^H(x) = a_{g1}\, sn(0, \lambda k_\infty, \kappa x) + a_{g2}\, cn(0, \lambda k_\infty, \kappa x) + a_{g3}\, sn(1, \lambda k_\infty, \mu x) + a_{g4}\, cn(1, \lambda k_\infty, \mu x) \tag{4.26}$$

The first two arguments of the eigenfunctions will be omitted in the following because it is possible to make a distinction between different eigenfunctions by κ or $\mu\mu$ appearing in the third argument.

Inserting the homogeneous solution into Eq. (4.7) yields

$$[A(B_0^2)]\begin{bmatrix} a_{11}\, sn(\kappa x) + a_{12}\, cn(\kappa x) \\ a_{21}\, sn(\kappa x) + a_{22}\, cn(\kappa x) \end{bmatrix} + [A(B_1^2)]\begin{bmatrix} a_{13}\, sn(\mu x) + a_{14}\, cn(\mu x) \\ a_{23}\, sn(\mu x) + a_{24}\, cn(\mu x) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{4.27}$$

For this equation to hold for all values of x, each of the four terms, each containing a different function, must be zero. By applying this condition to the equation for the thermal flux, the following relations are obtained in terms of the fast-to-thermal flux ratios:

$$r \equiv \frac{a_{11}}{a_{21}} = \frac{a_{12}}{a_{22}} = \frac{D_2 B_0^2 + \Sigma_{r2}}{\Sigma_{12}} \tag{4.28}$$

and

$$s \equiv \frac{a_{13}}{a_{23}} = \frac{a_{14}}{a_{24}} = \frac{D_2 B_1^2 + \Sigma_{r2}}{\Sigma_{12}} \tag{4.29}$$

In terms of the fast-to-thermal flux ratios, the homogeneous solution can then be compactly represented as:

$$\begin{bmatrix} \phi_1^H(x) \\ \phi_2^H(x) \end{bmatrix} = \begin{bmatrix} r & s \\ 1 & 1 \end{bmatrix}\begin{bmatrix} a_{21}\, sn(\kappa x) + a_{22}\, cn(\kappa x) \\ a_{23}\, sn(\mu x) + a_{24}\, cn(\mu x) \end{bmatrix} \tag{4.30}$$

This equation contains four unknown coefficients per node which will be determined later by imposing constraints on node average fluxes and the continuity of flux and current at the interface of the two-node problem. Before that, however, the particular solution will be determined.

Particular Solution

As discussed earlier, the RHS of Eq. (4.1) is a quadratic polynomial. Noting that there is a second order derivative in the LHS of Eq. (4.1) and the highest order term on the RHS is quadratic, it is possible to set the particular solution as a quartic polynomial. Namely, let

$$\phi_g^P(x) = c_{g0} + \sum_{p=1}^{4} c_{gp} f_p(\xi) \tag{4.31}$$

where

$$f_3(\xi) = \xi^3 - \frac{1}{4}\xi \tag{4.32}$$
$$f_4(\xi) = \xi^4 - \frac{3}{10}\xi^2 + \frac{1}{80} \quad .$$

Note that these polynomial basis functions are the same ones used in the nodal expansion method (NEM). The nonvanishing second derivatives of the basis functions are as follows:

$$f''_2(\xi) = 6 \quad , \qquad f''_3(\xi) = 6\xi = 6 f_1(\xi) \quad , \qquad f''_4(\xi) = 12\xi^2 - \frac{3}{5} = 4 f_2(\xi) + \frac{2}{5} \tag{4.33}$$

Inserting Eq. (4.31) into Eq. (4.1) yields

$$-\frac{1}{h_x^2}\begin{bmatrix} D_1 g_1(\xi) \\ D_2 g_2(\xi) \end{bmatrix} + \begin{bmatrix} \Sigma_{r1} - \lambda\nu\Sigma_{f1} & -\lambda\nu\Sigma_{f2} \\ -\Sigma_{12} & \Sigma_{r2} \end{bmatrix} \begin{vmatrix} c_{10} + \sum_{p=1}^{4} c_{1p} f_p(\xi) \\ c_{20} + \sum_{p=1}^{4} c_{2p} f_p(\xi) \end{vmatrix} = -\begin{bmatrix} \bar{L}_1 + b_{11} f_1(\xi) + b_{12} f_2(\xi) \\ \bar{L}_2 + b_{21} f_1(\xi) + b_{22} f_2(\xi) \end{bmatrix} \tag{4.34}$$

where

$$g_g(\xi) \equiv 6 c_{g2} + \frac{2}{5} c_{g4} + 6 c_{g3} f_1(\xi) + 4 c_{g4} f_2(\xi) \tag{4.35}$$

For the equality to hold, each of the five different order terms should vanish. First, the terms containing the third and fourth basis function should satisfy:

$$\begin{bmatrix} \Sigma_{r1} - \lambda\nu\Sigma_{f1} & -\lambda\nu\Sigma_{f2} \\ -\Sigma_{12} & \Sigma_{r2} \end{bmatrix} \begin{bmatrix} c_{1p} \\ c_{2p} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad , \quad p=3, 4 \tag{4.36}$$

A nontrivial solution of Eq. (4.36) exists only when the determinant of the coefficient matrix is zero. This occurs only when $\lambda k_\infty = 1$ (or $k_\infty = k_{eff}$). Otherwise ($k_\infty \neq k_{eff}$), the third and fourth order expansion coefficients must vanish yielding quadratic particular solutions.

In the following derivation, only the usual case of $k_\infty \neq k_{eff}$ will be considered because the other case is rarely encountered in practice. But it should be noted here that the particular solution becomes quartic in the case of $k_\infty = k_{eff}$, which is referred to as a *critical* node case and a separate derivation is necessary. The alternative to this approach is to use the Hybrid ANM/NEM method as described in the next section.

In the case of $k_\infty \neq k_{eff}$, the three expansion coefficients of the quadratic particular solution are uniquely determined by solving the following equations:

$$\begin{bmatrix} \Sigma_{r1} - \lambda\nu\Sigma_{f1} & -\lambda\nu\Sigma_{f2} \\ -\Sigma_{12} & \Sigma_{r2} \end{bmatrix} \begin{bmatrix} c_{1p} \\ c_{2p} \end{bmatrix} = \begin{bmatrix} -b_{1p} \\ -b_{2p} \end{bmatrix} \quad , \quad p=0, 1, 2 \tag{4.37}$$

where b_{g1} and b_{g2} are the first and second order coefficients of the transverse leakage in Eq. (4.4) and b_{g0} is defined as:

$$b_{g0} = \bar{L}_g - \frac{6}{h_x^2} D_g c_{g2} \quad . \tag{4.38}$$

Note that the coefficient matrix in Eq. (4.37) is bold $\boldsymbol{A(0)}$, defined in Eq. (4.10), that corresponds to zero buckling. Defining bold $\boldsymbol{A}_\infty \equiv \boldsymbol{A}(0)$ signifies no net leakage. Then the solution of Eq. (4.37) is obtained as:

$$c_p = -A_\infty^{-1} b_p \tag{4.39}$$

where $c_p \equiv [c_{1p}, c_{2p}]^T$ and $b_p \equiv [b_{1p}, b_{2p}]^T$.

Eq. (4.39) completely determines the particular solution at each node and only the transverse leakage information is necessary to determine the particular solution. However, the final solution consisting of the homogeneous as well as heterogeneous solutions is not yet fully specified because the four coefficients in the homogeneous solution are undetermined. In the next subsection, the final solution is obtained by providing equations for the unknown coefficients.

Final Solution

The general solution of Eq. (4.1) is obtained by summing the homogenous and the particular solutions given in Eq. (4.30) and Eq. (4.31), respectively. In the case of $k_\infty \neq k_{eft}$, it is then represented by:

$$\begin{bmatrix} \phi_1(x) \\ \phi_2(x) \end{bmatrix} = \begin{bmatrix} \phi_1^H(x) + \phi_1^P(x) \\ \phi_2^H(x) + \phi_2^P(x) \end{bmatrix} = \begin{bmatrix} r & s \\ 1 & 1 \end{bmatrix} \begin{bmatrix} a_{21} sn(\kappa x) + a_{22} cn(\kappa x) \\ a_{23} sn(\mu x) + a_{24} cn(\mu x) \end{bmatrix} + \begin{bmatrix} c_{10} + c_{11} f_1(\xi) + c_{12} f_2(\xi) \\ c_{20} + c_{21} f_1(\xi) + c_{22} f_2(\xi) \end{bmatrix} \tag{4.40}$$

This solution is valid for each of the two nodes and there are four coefficients to be determined for each node, yielding a total of eight unknowns for the two-node problem. These eight unknowns can be uniquely determined by applying the following eight constraints: four node average flux constraints (2 groups times× 2 nodes), two flux continuity (2 groups ×times 1 interface), and two current continuity (2 groups ×times 1 interface). In the following each constraint is applied separately

Constraint on Node Average Flux

The node average flux constraint reads:

$$\bar{\phi}_g^q = \frac{1}{h_x^q} \int_{-\frac{h_x^q}{2}}^{\frac{h_x^q}{2}} \phi_g^q(x)\, dx = \int_{-\frac{1}{2}}^{\frac{1}{2}} \phi_g^q(h_x^q \xi)\, d\xi \quad , \qquad q \in \{r, l\} \tag{4.41}$$

where $\bar{\phi}_g^q$ is given for each node from the previous CMFD calculation. The node designator, *r* and *l*, are for the right and left side node of the interface, respectively.

The evaluation of the flux integral requires only the following integral since the other integrals vanish:

$$I_c(\alpha) \equiv \int_{-\frac{1}{2}}^{\frac{1}{2}} cn(\alpha\xi)\, d\xi \quad , \qquad \alpha \in \{\kappa h_x, \mu h_x\} \tag{4.42}$$

$$= \frac{2}{\alpha} sn\left(\frac{\alpha}{2}\right)$$

The node average flux constraint is then simplified as (omitting the superscript q which designates the node):

$$\begin{bmatrix} \bar{\phi}_1 \\ \bar{\phi}_2 \end{bmatrix} = \begin{bmatrix} r & s \\ 1 & 1 \end{bmatrix} \begin{bmatrix} a_{22} \dfrac{2}{\kappa h_x} sn\left(\dfrac{\kappa h_x}{2}\right) \\ a_{24} \dfrac{2}{\mu h_x} sn\left(\dfrac{\mu h_x}{2}\right) \end{bmatrix} + \begin{bmatrix} c_{10} \\ c_{20} \end{bmatrix} \tag{4.43}$$

By defining the following parameters for brevity:

$$\kappa_{1/2} \equiv \kappa \frac{h_x}{2} \quad , \qquad \mu_{1/2} \equiv \mu \frac{h_x}{2} \quad , \tag{4.44}$$

the two even expansion coefficients of each node can be determined by solving the following equation for a_{22} and a_{24}:

$$\begin{bmatrix} r\dfrac{sn(\kappa_{1/2})}{\kappa_{1/2}} & s\dfrac{sn(\mu_{1/2})}{\mu_{1/2}} \\ \dfrac{sn(\kappa_{1/2})}{\kappa_{1/2}} & \dfrac{sn(\mu_{1/2})}{\mu_{1/2}} \end{bmatrix} \begin{bmatrix} a_{22} \\ a_{24} \end{bmatrix} = \begin{bmatrix} \bar{\phi}_1 - c_{10} \\ \bar{\phi}_2 - c_{20} \end{bmatrix} \tag{4.45}$$

This 2x2 linear system is solved for each node and the solution determines the even function coefficients. Note that no information about the neighboring node is necessary to determine the even function coefficients for the node of interest. The remaining four odd functions coefficients of the two-node problem are determined by imposing the continuity constraints at the interface.

Flux and Current Continuity

Including the discontinuity factor, the flux continuity condition is represented as:

$$\xi_g^l \phi_g^l\left(\frac{h_x^l}{2}\right) = \xi_g^r \phi_g^r\left(-\frac{h_x^r}{2}\right) \tag{4.46}$$

where ξ_g^q is the discontinuity factor, while the current continuity condition reads

$$-D_g^l \frac{d\phi_g^l(x)}{dx}\bigg|_{\frac{h_x^l}{2}} = -D_g^r \frac{d\phi_g^r(x)}{dx}\bigg|_{-\frac{h_x^r}{2}} \tag{4.47}$$

Inserting Eq. (4.40) into Eq. (4.46) yields:

$$\begin{bmatrix} \xi_1^l r_l & \xi_1^l s_l \\ \xi_2^l & \xi_2^l \end{bmatrix} \begin{bmatrix} a_{21}^l sn(\kappa_{1/2}^l) + a_{22}^l cn(\kappa_{1/2}^l) \\ a_{23}^l sn(\mu_{1/2}^l) + a_{24}^l cn(\mu_{1/2}^l) \end{bmatrix} + \begin{bmatrix} \xi_1^l \phi_1^{Pl} \\ \xi_2^l \phi_2^{Pl} \end{bmatrix} = \begin{bmatrix} \xi_1^r r_r & \xi_1^r s_r \\ \xi_2^r & \xi_2^r \end{bmatrix} \begin{bmatrix} -a_{21}^r sn(\kappa_{1/2}^r) + a_{22}^r cn(\kappa_{1/2}^r) \\ -a_{23}^r sn(\mu_{1/2}^r) + a_{24}^r cn(\mu_{1/2}^r) \end{bmatrix} + \begin{bmatrix} \xi_1^r \phi_1^{Pr} \\ \xi_2^r \phi_2^{Pr} \end{bmatrix} \tag{4.48}$$

where the particular solution at the interface is defined as:

$$\phi_g^{Pq} \equiv c_{g0}^l + (-1)^i \frac{c_{g1}^l}{2} + \frac{c_{g2}^l}{2} \tag{4.49}$$

with the exponent *i* being defined as:

$$i = \begin{cases} 0 & , \quad q = l \\ 1 & , \quad q = r \end{cases} \tag{4.50}$$

The current continuity condition requires the derivatives of each basis functions. The derivatives of the generic functions defined in Eq. (4.25) with respect to *x* are obtained as follows:

$$sn'(m, \lambda k_\infty, x) = cn'(m, \lambda k_\infty, x) \tag{4.51}$$

and

$$cn'(m, \lambda k_\infty, x) \equiv \begin{cases} -sn(m, \lambda k_\infty, x) & , \quad if \ m = 0 \quad and \quad \lambda k_\infty > 1 \\ 0 & , \quad if \ m = 0 \quad and \quad \lambda k_\infty = 1 \\ sn(m, \lambda k_\infty, x) & , \quad if \ m = 1 \quad and \quad \lambda k_\infty < 1 \end{cases} \tag{4.52}$$

Inserting Eq. (4.40) into Eq. (4.47) and evaluating the derivatives yields

$$-\begin{bmatrix} D_1^l r_l & D_1^l s_l \\ D_2^l & D_2^l \end{bmatrix} \begin{bmatrix} a_{21}^l \kappa_l cn(\kappa_{1/2}^l) \mp a_{22}^l \kappa_l sn(\kappa_{1/2}^l) \\ a_{23}^l \mu_l cn(\mu_{1/2}^l) + a_{24}^l \mu_l sn(\mu_{1/2}^l) \end{bmatrix} + \begin{bmatrix} J_1^{Pl} \\ J_2^{Pl} \end{bmatrix} = \begin{bmatrix} D_1^r r_r & D_1^r s_r \\ D_2^r & D_2^r \end{bmatrix} \begin{bmatrix} a_{21}^r \kappa_r cn(\kappa_{1/2}^r) \pm a_{22}^r \kappa_r sn(\kappa_{1/2}^r) \\ a_{23}^r \mu_r cn(\mu_{1/2}^r) - a_{24}^r \mu_r sn(\mu_{1/2}^r) \end{bmatrix} + \begin{bmatrix} J_1^{Pr} \\ J_2^{Pr} \end{bmatrix} \tag{4.53}$$

where the current due to the particular solution is defined as:

$$J_g^{Pq} \equiv -\frac{D_g^q}{h_x^q} (c_{g1}^q + (-1)^i 3 c_{g2}^q) \tag{4.54}$$

with the same exponent *i* as the one defined in Eq. (4.50). The two signs in front of a_{22}^q are for making distinction depending on the relative size of k_∞ to k_{eft}. The upper sign is for $k_\infty > k_{eft}$ while the lower is for $k_\infty < k_{eft}$.

Eq. (4.48) and Eq. (4.53) are coupled together and form a 4x4 linear system. To represent the 4x4 linear system compactly, the following generic 2x2 matrices are defined as a function of f:

$$\Phi_q(f) = \begin{bmatrix} r_q\xi_1^q & s_q\xi_1^q \\ \xi_2^q & \xi_2^q \end{bmatrix} \begin{bmatrix} f(\kappa_{1/2}^q) & 0 \\ 0 & f(\mu_{1/2}^q) \end{bmatrix} \tag{4.55}$$

and

$$J_q(f) = \begin{bmatrix} r_q D_1^q & s_q D_1^q \\ D_2^q & D_2^q \end{bmatrix} \begin{bmatrix} \kappa_q f(\kappa_{1/2}^q) & 0 \\ 0 & \mu_q f(\mu_{1/2}^q) \end{bmatrix} \tag{4.56}$$

Also, two-element vectors consisting of the set of odd and even coefficients are defined as:

$$\boldsymbol{a}_o^q \equiv [a_{21}^q,\ a_{23}^q]^T \quad \text{and} \quad \boldsymbol{a}_e^q \equiv [a_{22}^q,\ a_{24}^q]^T \tag{4.57}$$

and two-element vectors for contributions from the particular solution are defined as:

$$\phi_q^P \equiv [\xi_1^q \phi_1^{Pq},\ \xi_2^q \phi_2^{Pq}]^T \quad \text{and} \quad \boldsymbol{J}_q^P \equiv [J_1^{Pq},\ J_2^{Pq}]^T \quad . \tag{4.58}$$

Then Eq. (4.48) and Eq. (4.53) can be combined to yield

$$\begin{bmatrix} \Phi_l(sn) & \Phi_r(sn) \\ -J_l(sn') & J_r(sn') \end{bmatrix} \begin{bmatrix} \boldsymbol{a}_o^l \\ \boldsymbol{a}_o^r \end{bmatrix} = \begin{bmatrix} -\Phi_l(cn) & \Phi_r(cn) \\ J_l(cn') & J_r(cn') \end{bmatrix} \begin{bmatrix} \boldsymbol{a}_e^l \\ \boldsymbol{a}_e^r \end{bmatrix} + \begin{bmatrix} \phi_r^P - \phi_l^P \\ \boldsymbol{J}_r^P - \boldsymbol{J}_l^P \end{bmatrix} \quad . \tag{4.59}$$

The 4x4 linear system of Eq. (4.59) can be solved directly by Gaussian elimination for the odd function coefficients of the two nodes.

With the even and odd coefficients determined by solving Eq. (4.45) and Eq. (4.59), respectively, the eight unknown coefficients in the two-node problem are now fully determined and the fast and thermal fluxes given in Eq. (4.40) are known. Substituting these fluxes into the right or left side of Eq. (4.47), the net nodal current at the interface of the two nodes (J_g^{nodal}) can be calculated. Using Eq. (2.21), the CNCC is then determined as follows in terms of the nodal interface current and the node average fluxes:

$$\hat{D}_g = -\frac{J_g^{nodal} + \tilde{D}_g(\phi_g^r - \phi_g^l)}{\phi_g^r + \phi_g^l} \tag{4.60}$$

The CNCC can then be used in the subsequent CMFD calculation.

Incorporation of Boundary Conditions

At the external boundaries of the problem domain, the CNCC can be determined by solving one-node problems. In this section, two kinds of boundary conditions are considered: zero flux and zero incoming current. Note that nodal calculation is not necessary for the zero current boundary condition because the nodal current becomes zero in this case.

Zero Flux

The condition of vanishing flux at the boundary is represented as:

$$\begin{bmatrix} \phi_1\left(\pm\frac{h_x}{2}\right) \\ \phi_2\left(\pm\frac{h_x}{2}\right) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} r & s \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \pm a_{21} sn(\kappa_{1/2}) + a_{22} cn(\kappa_{1/2}) \\ \pm a_{23} sn(\mu_{1/2}) + a_{24} cn(\mu_{1/2}) \end{bmatrix} + \begin{bmatrix} \phi_1^P \\ \phi_2^P \end{bmatrix} \tag{4.61}$$

Given the even coefficients and the particular solutions which can be determined by Eq. (4.45) and Eq. (4.39), the two odd coefficients can be solved for using Eq. (4.61).

Zero Incoming Current

From the P_1 approximation, the partial currents are obtained as:

$$J_{g\pm}(x) = \frac{1}{4}\phi_g(x) \pm \frac{1}{2}J_g(x) \tag{4.62}$$

The condition of zero incoming current then becomes

$$J_{g\mp}\left(\pm\frac{h_x}{2}\right) = 0 \quad \Rightarrow \quad \phi_g\left(\pm\frac{h_x}{2}\right) = \pm 2 J_g\left(\pm\frac{h_x}{2}\right) \tag{4.63}$$

At the right boundary, Eq. (4.43) becomes

$$\begin{bmatrix} r & s \\ 1 & 1 \end{bmatrix} \begin{bmatrix} a_{21} sn(\kappa_{1/2}) + a_{22} cn(\kappa_{1/2}) \\ a_{23} sn(\mu_{1/2}) + a_{24} cn(\mu_{1/2}) \end{bmatrix} + \begin{bmatrix} \phi_{1+}^P \\ \phi_{2+}^P \end{bmatrix} = -2 \begin{bmatrix} D_1 r & D_1 s \\ D_2 & D_2 \end{bmatrix} \begin{bmatrix} a_{21}\kappa cn(\kappa_{1/2}) \mp a_{22}\kappa sn(\kappa_{1/2}) \\ a_{23}\mu cn(\mu_{1/2}) - a_{24}\mu sn(\mu_{1/2}) \end{bmatrix} + 2 \begin{bmatrix} J_{1+}^P \\ J_{2+}^P \end{bmatrix} \tag{4.64}$$

while at the left boundary it becomes

$$\begin{bmatrix} r & s \\ 1 & 1 \end{bmatrix} \begin{bmatrix} -a_{21} sn(\kappa_{1/2}) + a_{22} cn(\kappa_{1/2}) \\ -a_{23} sn(\mu_{1/2}) + a_{24} cn(\mu_{1/2}) \end{bmatrix} + \begin{bmatrix} \phi_{1-}^P \\ \phi_{2-}^P \end{bmatrix} = 2 \begin{bmatrix} D_1 r & D_1 s \\ D_2 & D_2 \end{bmatrix} \begin{bmatrix} a_{21}\kappa cn(\kappa_{1/2}) \mp a_{22}\kappa sn(\kappa_{1/2}) \\ a_{23}\mu cn(\mu_{1/2}) - a_{24}\mu sn(\mu_{1/2}) \end{bmatrix} - 2 \begin{bmatrix} J_{1-}^P \\ J_{2-}^P \end{bmatrix} \tag{4.65}$$

Here the ± sign now comes from the k_∞ condition. Eq. (4.64) or Eq. (4.65) can be solved for the two odd function coefficients.

Transient Two-Node Problem

The transverse-integrated one-dimensional neutron diffusion equation corresponding to the transient fixed source problem given in Eq. (2.30) can be obtained as follows, considering the spatial variation of all the variables within a node:

$$\frac{dJ_g^{m,n}(x)}{dx} + \tilde{\Sigma}_{rg}^{m,n}\phi_g^{m,n}(x) - \tilde{Q}_g^{m,n}(x) = \tilde{S}_g^{m,n}(x) - L_g^{m,n}(x) \tag{4.66}$$

where the production term and source term are given as:

$$\tilde{Q}_g^{m,n}(x) \equiv \begin{cases} \beta_p^{m,n}\lambda \sum_{g'=1}^{2} \nu\Sigma_{fg'}^{m,n}\phi_{g'}^{m,n}(x) & , \quad g=1 \\ \Sigma_{12}^{m,n}\phi_1^{m,n}(x) & , \quad g=2 \end{cases} , \tag{4.67}$$

and

$$\tilde{S}_g^{m,n}(x) \equiv \begin{cases} S_1^{m,n}(x) + \tilde{S}_d^{m,n-1}(x) & , \quad g=1 \\ S_2^{m,n}(x) & , \quad g=2 \end{cases} \tag{4.68}$$

respectively. Note here that the fission source term, $Q_1^{m,n}(x)$, contains only the prompt fission source.

In solving a transient two-node problem involving Eq. (4.66), a computational difficulty arises regarding the source terms appearing in Eq. (4.66). This is because the spatial variation of the source terms, $S_g^{m,n}(x)$, $\tilde{S}_d^{m,n-1}(x)$, should be known when solving the two-node problem. This requires saving all the coefficients describing the intra-nodal variation of all the variables such as flux and precursor density, which requires a considerable amount of memory space for practical reactor problems. In addition, a separate transient two-node method, which is different from the steady-state one, has to be applied because of the difference in the functional form of the RHS.

The problem associated with the source terms was addressed by Engrand *et. al*. and was resolved by an innovative iterative strategy[15]. In their approach, the terms that do not appear in the steady state equation are moved to the RHS and the entire RHS term including the transverse leakage is approximated by a single quadratic polynomial. In the work here, the same approach is used in the transient two-node calculations, which leads to the following 1-D neutron balance equation:

$$\frac{dJ_g^{m,n}(x)}{dx} + \Sigma_{rg}^{m,n}\phi_g^{m,n}(x) - Q_g^{m,n}(x) = \hat{S}_g^{m,n}(x) \tag{4.69}$$

where $Q_g^{m,n}(x)$ is the total fission source as defined in Eq. (4.2) and the effective source, $\hat{S}_g^{m,n}(x)$, is defined as:

$$\hat{S}_g^{m,n}(x) \equiv \begin{cases} S_1^{m,n}(x) - L_1^{m,n}(x) - \left(\frac{1}{\theta v_1^m \Delta t_n} + \frac{\alpha_1^{m,n}}{v_1^m}\right)\phi_1^{m,n}(x) \\ \quad + \tilde{S}_d^{m,n-1}(x) + (\beta^m - \omega_m^n)\lambda \sum_{g'=1}^{2} \nu\Sigma_{fg'}^{m,n}\phi_{g'}^{m,n}(x) & , \quad g=1 \\ S_2^{m,n}(x) - L_1^{m,n}(x) - \left(\frac{1}{\theta v_2^m \Delta t_n} + \frac{\alpha_2^{m,n}}{v_2^m}\right)\phi_2^{m,n}(x) & , \quad g=2 \end{cases} \tag{4.70}$$

The quadratic polynomial that approximates the effective source term can be obtained in the same manner that is applied to the transverse leakage in the eigenvalue calculation. Namely, the quadratic polynomial of a node is obtained by preserving three node average values of the RHS terms of Eq. (4.70) using the most recent iterates of the node average fluxes, etc. Once the polynomial coefficients are determined, the exactly same two-node (or one-node) ANM solution method as the eigenvalue calculation can be applied by just regarding the effective source term as the transverse leakage appearing in Eq. (4.1).

Although the same two-node kernel for the eigenvalue calculation can be applied in the transient calculation, the nonlinear iteration strategy which coordinates the solutions of the CMFD and two-node problems may be different in the two calculations. This is primarily because the CNCC does not change significantly during most of the transient calculation once they are initially determined in the steady-state calculation. The nonlinear iteration strategy for the steady-state and the transient calculation will be described in Chapter 8.

4.1.2 Stabilization Techniques for ANM

As described above and noted in Reference [16], unless a linear basis function is explicitly provided for the homogeneous solution, the Analytic Nodal Method can be unstable in cases when the node k_∞ is close to unity. An alternate approach is used in PARCS in which the Nodal Expansion Method (NEM) is applied when a user specified criteria is satisfied:

$$\delta = \left| \frac{k_\infty}{k_{keff}} - 1 \right| < \varepsilon$$

The derivation of the Nodal Expansion Method (NEM) is provided in many references (e.g. Turinsky et al. in Reference [10]). In addition to constraints similar to those of the ANM kernel, two moment balance equations are applied to determine the polynomial expansion coefficients. Since the NEM kernel does not involve a particular solution, it does not have the numerical instability of the ANM and is numerically stable under all conditions.

The implementation of a hybrid ANM/NEM method is conceptually very simple. The user specified criterion is used to identify near-critical nodes and activate the NEM solution. If the value of ε is greater than unity in the hybrid scheme, the solution of all two-node problems will be based on NEM. On the other hand, if the value is zero, the solution be based purely on ANM. For intermediate values, both the NEM and the ANM are used in different portions of the core. A detailed analysis of various values of the criteria is presented in Reference [16].

4.2 Nodal Expansion Method (NEM)

The nodal Expansion Method (NEM) for the solution of the steady state multi-group diffusion equations in Cartesian geometry is described in this sections. This section describes the standard NEM formulation for the solution of the three-dimensional, Cartesian geometry, multi-group, eigenvalue neutron diffusion equation[26,27]. The principal characteristics of the polynomial nodal method are its quartic

expansions of the one-dimensional transverse-integrated flux and quadratic leakage model for the transverse leakage.

Consider the general form of the steady-state multi-group neutron diffusion equation, written in standard form and with the group constants (i.e., properly weighted cross-sections and discontinuity factors) already available from a lattice physics calculation for g = 1, 2, ..., G. It is given by

$$\vec{\nabla} \cdot D_g \vec{\nabla} \phi_g + \Sigma_{tg} \phi_g = \sum_{g'=1}^{G} \Sigma_{sgg'} \phi_{g'} + \frac{\chi_g}{k} \sum_{g'=1}^{G} \nu_{g'} \Sigma_{fg'} \phi_{g'} \tag{4.71}$$

where the dependence of each quantity on the spatial coordinate $\vec{r}$ has been suppressed, and

D_g = diffusion coefficient (cm)

ϕ_g = neutron flux ($cm^{-2}sec^{-1}$)

Σ_{tg} = total macroscopic cross section (cm^{-1})

$\Sigma_{sgg'}$ = group-to-group scattering cross section (cm^{-1})

χ_g = fission neutrons yield

k = multiplication factor (i.e., critical eigenvalue)

ν_g = average number of neutrons created per fission

Σ_{fg} = macroscopic fission cross section (cm^{-1}).

As with most modern nodal methods, we begin by integrating the multi-group neutron diffusion equation over a material-centered spatial node which has homogenized properties. For Cartesian geometry we rewrite Equation (4.71) for the arbitrary spatial node l as

$$D_g^l \frac{\partial^2}{\partial x^2} \phi_g^l(\vec{r}) - D_g^l \frac{\partial^2}{\partial y^2} \phi_g^l(\vec{r}) - D_g^l \frac{\partial^2}{\partial z^2} \phi_g^l(\vec{r}) + A_g^l \phi_g^l(\vec{r}) = Q_g^l(\vec{r}) \tag{4.72}$$

,where $g \in (1, G)$

$$(\vec{r}) \equiv (x, y, z) \in V^l = \Delta x \Delta y \Delta z \equiv \text{Volume of node } l \tag{4.73}$$

$$A_g^l = \Sigma_{tg}^l - \Sigma_{sgg}^l - \frac{\chi_g^l}{k} \nu_g \Sigma_{fg}^l \tag{4.74}$$

$$Q_g^l(\vec{r}) = \sum_{g' \neq g}^{G} Q_{gg'}^l \phi_{g'}^l(\vec{r}) = \sum_{g' \neq g}^{G} \Sigma_{sgg'}^l \phi_{g'}^l(\vec{r}) + \frac{\chi_g^l}{k} \sum_{g' \neq g}^{G} \nu_{g'} \Sigma_{fg'}^l \phi_{g'}^l(\vec{r}) \tag{4.75}$$

For simplicity, in cases where redundant equations exist in all three directions, the illustrating equations will be only given in the x-direction. Using Fick's Law, which in the x-direction can be expressed as,

$$j_{gx}^l(\vec{r}) = - D_g^l \frac{\partial}{\partial x} \phi_g^l(\vec{r}) \tag{4.76}$$

where, $j_{gx}^l(\vec{r})$ is the x-component of the net neutron current which allows Equation (4.72) to be rewritten as

$$\frac{\partial}{\partial x} j_{gx}^l(\vec{r}) + \frac{\partial}{\partial y} j_{gy}^l(\vec{r}) + \frac{\partial}{\partial z} \quad j_{gz}^l(\vec{r}) + A_g^l \phi_g^l(\vec{r}) = Q_g^l(\vec{r}) \tag{4.77}$$

Integration of Equation (4.77) over the volume of node l generates a local neutron balance equation in terms of the face-averaged net currents and the node volume average flux. It is given by

$$\frac{1}{\Delta x^l}(\bar{L}_{gx}^l) + \frac{1}{\Delta y^l}(\bar{L}_{gy}^l) + \frac{1}{\Delta z^l}(\bar{L}_{gz}^l) + A_g^l \bar{\phi}_g^l = \bar{Q}_g^l \tag{4.78}$$

where, assuming node l is centered around the coordinate's origin, the volume integrated quantities are defined as

$$\bar{\phi}_g^l = \frac{1}{V^l} \int_{-\frac{\Delta x^l}{2}}^{\frac{\Delta x^l}{2}} \int_{-\frac{\Delta y^l}{2}}^{\frac{\Delta y^l}{2}} \int_{-\frac{\Delta z^l}{2}}^{\frac{\Delta z^l}{2}} \phi_g^l(\vec{r})\, dx dy dz \equiv \text{Node volume average flux} \tag{4.79}$$

$$\bar{Q}_g^l = \frac{1}{V^l} \int_{-\frac{\Delta x^l}{2}}^{\frac{\Delta x^l}{2}} \int_{-\frac{\Delta y^l}{2}}^{\frac{\Delta y^l}{2}} \int_{-\frac{\Delta z^l}{2}}^{\frac{\Delta z^l}{2}} Q_g^l(\vec{r})\, dx dy dz \equiv \text{Node volume average source} \tag{4.80}$$

$$\frac{1}{\Delta x^l} \bar{L}_{gx}^l = \frac{1}{\Delta x^l}(\bar{J}_{gx+}^l - \bar{J}_{gx-}^l) = \frac{1}{V^l} \int_{-\frac{\Delta x^l}{2}}^{\frac{\Delta x^l}{2}} \int_{-\frac{\Delta y^l}{2}}^{\frac{\Delta y^l}{2}} \int_{-\frac{\Delta z^l}{2}}^{\frac{\Delta z^l}{2}} \frac{\partial}{\partial x} j_{gx}^l(\vec{r})\, dx dy dz \tag{4.81}$$

where $\bar{J}^{l}_{gx\pm}$ is the average x-directed net current on node faces $\pm\frac{\Delta x^{l}}{2}$.

Equation. (4.78) is known as the nodal balance equation. Now for the neutron diffusion equation written in this form, in order to obtain the spatial neutron flux distribution, one must devise some relationship between the node average flux and the face-averaged net (surface) currents. It is the equations used to compute the surface currents in Equation (4.78) which distinguish one nodal formulation from another. In NEM, the widely used method of transverse-integration is used, where the three-dimensional diffusion equation is integrated over the two directions transverse to each axis. This generates three one-dimensional equations, one for each direction in Cartesian coordinates, of the following form:

$$\frac{d}{dx}\bar{j}^{l}_{gx}(x) + A^{l}_{g}\bar{\phi}^{l}_{gx}(x) = \bar{Q}^{l}_{gx}(x) - \frac{1}{\Delta y^{l}}\bar{L}^{l}_{gy}(x) - \frac{1}{\Delta z^{l}}\bar{L}^{l}_{gz}(x) \tag{4.82}$$

where $\bar{L}^{l}_{gy}(x)$ is the average y-direction transverse leakage given by

$$\bar{L}^{l}_{gy}(x) = \frac{1}{\Delta z^{l}}\int_{-\frac{\Delta z^{l}}{2}}^{\frac{\Delta z^{l}}{2}}\int_{-\frac{\Delta y^{l}}{2}}^{\frac{\Delta y^{l}}{2}}\frac{\partial}{\partial y}j^{l}_{gy}(\vec{r})\,dy\,dz \tag{4.83}$$

and $\bar{L}^{l}_{gz}(x)$ is the average z-direction transverse leakage given by

$$\bar{L}^{l}_{gz}(x) = \frac{1}{\Delta y^{l}}\int_{-\frac{\Delta y^{l}}{2}}^{\frac{\Delta y^{l}}{2}}\int_{-\frac{\Delta z^{l}}{2}}^{\frac{\Delta z^{l}}{2}}\frac{\partial}{\partial z}j^{l}_{gz}(\vec{r})\,dz\,dy$$

(4.84)

In NEM, the one-dimensional averaged flux that appears in Equation (4.82) , is expanded as a general polynomial given by

$$\bar{\phi}^{l}_{gx}(x) = \bar{\phi}^{l}_{g} + \sum_{n=1}^{N} a^{l}_{gxn} \quad f_n(x) \tag{4.85}$$

where $\bar{\phi}^{l}_{g}$ is the node average flux, implying for Equation (4.85) to be true that $f_n(x)$ must be chosen such that the basis functions satisfy

$$\int_{-\frac{\Delta x^{l}}{2}}^{\frac{\Delta x^{l}}{2}} f_n(x)\,dx = 0 \quad \text{for} \quad n = 1, 2, \ldots, \quad N \tag{4.86}$$

Note that for quartic NEM, the method used in PARCS, the summation extends to N = 4. The first four basis functions in NEM can be expressed as follows[26]

$$\overbrace{\left(\frac{x}{\Delta x^{l}}\right)}^{f_1} \qquad \overbrace{3\left(\frac{x}{\Delta x^{l}}\right)^2 - \frac{1}{4}}^{f_2} \qquad \overbrace{\left(\frac{x}{\Delta x^{l}}\right)^3 - \frac{1}{4}\left(\frac{x}{\Delta x^{l}}\right)}^{f_3} \qquad \overbrace{\left(\frac{x}{\Delta x^{l}}\right)^4 - \frac{3}{10}\left(\frac{x}{\Delta x^{l}}\right)^2 + \frac{1}{80}}^{f_4}$$

which can be shown to also satisfy

$$f_n\left(\pm\frac{\Delta x^{l}}{2}\right) = 0 \qquad \text{for} \qquad n = 3, 4. \tag{4.87}$$

At this point it is appropriate to consider the elementary concept of accounting for the total number of equations and that of unknowns. For a three-dimensional Cartesian geometry, the node average and N expansion coefficients in each direction appear per node per energy group, implying a total of 3N+1 equations are required. The nodal balance equation, Equation (4.78), provides one equation, where now Equations (4.76) and (4.85) are used to eliminate face-averaged net currents from this equation. Surface current and flux continuity provide 6 more equations per node per energy group. So for N = 2, there would be an equal number of equations and unknowns without any further development. However, for N = 4, two additional unknowns are introduced for each direction per node per energy group. This is addressed by

using a weighted residual scheme[26] applied to Equation (4.82), which in essence provides the additional equations (referred to as the moment equations) needed,

$$< \omega_n(x), \frac{d}{dx} \bar{J}^l_{gx}(x) > + A^l_g \bar{\phi}^l_{gxn} = \bar{Q}^l_{gxn} - \frac{1}{\Delta y^l} \bar{L}^l_{gyxn} - \frac{1}{\Delta z^l} \bar{L}^l_{gzxn} \tag{4.88}$$

where the two weighting functions for n = 1,2 are chosen to be the same as the basic functions, namely $\omega_n(x) = f_n(x)$, as those used in the one-dimensional flux expansion[1]. Here, the first and second (actually linear combination of zeroth and second) moments of the flux, source, and leakage for each group g are defined by

$$\overbrace{\bar{\phi}^l_{gxn}}^{< \omega_n(x), \bar{\phi}^l_{gx}(x) >} \quad \overbrace{\bar{Q}^l_{gxn}}^{< \omega_n(x), \bar{Q}^l_{gx}(x) >} \quad \overbrace{\bar{L}^l_{gyxn}}^{< \omega_n(x), \bar{L}^l_{gy}(x) >} \quad \overbrace{\bar{L}^l_{gzxn}}^{< \omega_n(x), \bar{L}^l_{gz}(x) >} \tag{4.89}$$

The first term in Equation (4.89) is evaluated by using Equations (4.76) and (4.85) and the definition of the expansion coefficients, and completing the integration (i.e. inner product) analytically.

One last point which needs to be addressed before Equation can be solved is the transverse leakage terms appearing on the right hand side. Their spatial dependency is unknown, so their "shape" must be approximated. The most popular approximation in NEM is the quadratic transverse leakage approximation. For example, the x-direction spatial dependence of the y-direction transverse leakage is approximated by

$$\bar{L}^l_{gy}(x) \cong \overline{L}^l_{gy} + \rho^l_{gy1} f_1(x) + \rho^l_{gy2} f_2(x) \tag{4.90}$$

where $\bar{L}^l_{gy}$ is the average y-directed leakage in node l, and the coefficients ρ^l_{gy1} and ρ^l_{gy2} can be expressed in terms of average y-directed leakages of the two nearest-neighbor nodes along the x-direction (i.e. nodes l-1 and l+1) so as to preserve the node average leakages of these three nodes. The quadratic expansion coefficients can be shown to be given by

$$\begin{aligned} \rho^l_{gy1} = {} & g^l(\Delta x^l)[(\bar{L}^{l+1}_g - \bar{L}^l_g)(\Delta x^l + 2\Delta x^{l-1})(\Delta x^l + \Delta x^{l-1}) \\ & + (\bar{L}^l_{gy} - \bar{L}^{l-1}_{gy})(\Delta x^l + 2\Delta x^{l+1})(\Delta x^l + \Delta x^{l+1})] \end{aligned} \tag{4.91}$$

$$\rho^l_{gy2} = g^l(\Delta x^l)^2[(\bar{L}^{l+1}_{gy} - \bar{L}^l_{gy})(\Delta x^l + \Delta x^{l-1}) + (\bar{L}^{l-1}_{gy} - \bar{L}^l_{gy})(\Delta x^l + \Delta x^{l+1} \tag{4.92}$$

1. This constitutes a moments weighting scheme; if one uses $\omega_n(x) = f_{n+2}(x)$ for n = 1,2 it is known as Galerkin weighting. Numerical experiments favor moments weighting.

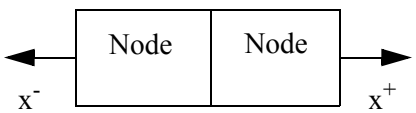
where

$$g^l = [(\Delta x^l + \Delta x^{l+1})(\Delta x^l + \Delta x^{l-1})(\Delta x^{l-1} + \Delta x^l + \Delta x^{l+1}). \quad (4.93)$$

The most common manner of solving the matrix system associated with NEM is the response-matrix formulation. To minimize computer run time and memory requirements, and to facilitate the capability to solve either the NEM or Finite Difference Method (FDM) formulation, the non-linear iterative strategy is employed in PARCS. This technique was developed by Smith[28,29,30] and successfully implemented into the Studsvik QPANDA and SIMULATE code packages. The documentation available on this technique is scarce, but it turns out to be rather simplistic and almost trivial to implement in a FDM code which utilizes the box-scheme (i.e. material-centered).

The basic idea is applicable to the standard FDM solution algorithm of the multi-group diffusion equation. Solving the FDM based equation utilizing an outer-inner iterative strategy, every ΔN_0 outer iterations (where ΔN_0 is somewhat arbitrary but can be optimized) the so-called "two-node problem" calculation (a spatially-decoupled NEM calculation spanning two adjoining nodes) is performed for every interface (for all nodes and in all directions) to provide an improved estimate of the net surface current at that particular interface. Subsequently, the NEM estimated net surface currents are used to update (i.e. change) the original FDM diffusion coupling coefficients. Outer iterations of the FDM based equation are then continued utilizing the updated FDM coupling coefficients for ΔN_0 outer iterations. The entire process is then repeated. This procedure of updating the FDM couplings is a convergent technique which progressively forces the FDM equation to yield the higher-order NEM predicted values of the net surface currents while satisfying the nodal balance Equation (4.78), thus yielding the NEM results for the node-average flux and fundamental mode eigenvalue. The advantages of this technique come in many forms; the storage requirements are minimal because the two-node problem arrays are re-usable (disposable) at each interface, the rate of convergence is nearly comparable to that of the base FDM algorithm being used, the number of iteratively determined unknowns is reduced by a factor of 6 (node flux vs. partial surface current), and the simplicity of the algorithm and ease of implementation, compared to any other nodal technique, is far superior.

The two-node problem produces an 8G x 8G linear system of equations which can be constructed by applying the standard NEM relations to two adjoining nodes. For simplicity, consider two arbitrary adjoining nodes in the x-direction. Denote these notes as l and l+1:



Substitution of the one-dimensional expansion, Equation , into Fick's law yields expressions for the average x-direction net surface currents at the left(-) and right(+) interfaces of node l gives

$$\bar{J}^l_{gx\pm} \equiv \frac{-D^l_g}{\Delta x^l}\left[a^l_{gx1} \pm 3a^l_{gx2} + \frac{1}{2}a^l_{gx3} \pm \frac{1}{5}a^l_{gx4}\right] \quad . \quad (4.94)$$

Now, assume the node average flux, criticality constant, and all transverse direction terms are known from a previous iteration; then, the total number of unknowns associated with the x-direction two node problem is 8G, which corresponds to the 4 expansion coefficients/group/node (x) G groups (x) two nodes. The 8G constraint equations are obtained as follows. We begin with the substitution of Equation into the nodal balance equation for node l, to yield the zeroth moment constraints (G equations/node) given by

$$\frac{-D_g^l}{\Delta x^l}\left[6a_{gx2}^l+\frac{2}{5}a_{gx4}^l\right]=-\frac{1}{\Delta y^l}\bar{L}_{gy}^l-\frac{1}{\Delta z^l}\bar{L}_{gz}^l-A_g^l\bar{\phi}_g^l+\sum_{g'\neq g}^{G}Q_{gg'}^l\bar{\phi}_{g'}^l \quad . \tag{4.95}$$

A similar substitution into the moment-weighted equation, Equation , yields the first and second moment constraints (2G equations/node) given by

$$\left[\frac{60}{\Delta x^l}\frac{D_g^l}{\Delta x^l}+A_g^l\right]a_{gx3}^l-\sum_{g'\neq g}^{G}Q_{gg'}^l a_{g'x3}^l-10A_g^l a_{gx1}^l+10\sum_{g'\neq g}^{G}Q_{gg'}^l a_{g'x1}^l=10\left(\frac{1}{\Delta y^l}\rho_{gy1}^l+\frac{1}{\Delta z^l}\rho_{gz1}^l\right) \tag{4.96}$$

$$\left[\frac{140}{\Delta x^l}\frac{D_g^l}{\Delta x^l}+A_g^l\right]a_{gx4}^l-\sum_{g'\neq g}^{G}Q_{gg'}^l a_{g'x4}^l-35A_g^l a_{gx2}^l+35\sum_{g'\neq g}^{G}Q_{gg'}^l a_{g'x2}^l=35\left(\frac{1}{\Delta y^l}\rho_{gy2}^l+\frac{1}{\Delta z^l}\rho_{gz2}^l\right) \quad . \tag{4.97}$$

Similar equations can be written for node l + 1, producing a total of 6G equations. The continuity of net surface current constraints at the interface (G equations) are obtained by using Equation at the adjoining interface of the two nodes. The result is

$$\frac{-D_g^l}{\Delta x^l}\left[a_{gx1}^l+3a_{gx2}^l+\frac{a_{gx3}^l}{2}+\frac{a_{gx4}^l}{5}\right]=\frac{-D_g^l}{\Delta x^{l+1}}\left[a_{gx1}^{l+1}-3a_{gx2}^{l+1}+\frac{a_{gx3}^{l+1}}{2}-\frac{a_g^l}{5}\right. \tag{4.98}$$

Last, the continuity (or discontinuity) of surface-averaged flux constraints (G equations) are obtained by equating the surface-averaged fluxes of the two adjoining nodes by using Equation . The result is

$$d_{gx+}^l\left[\phi_g^l+\frac{a_{gx1}^l}{2}+\frac{a_{gx2}^l}{2}\right]=d_{gx-}^{l+1}\left[\phi_g^{l+1}-\frac{a_{gx1}^{l+1}}{2}+\frac{a_{gx2}^{l+1}}{2}\right] \tag{4.99}$$

where $d_{gx\pm}^l$ and $d_{gx\pm}^{l+1}$ are the Discontinuity Factors (DFs) obtained from lattice physics calculations. Do note that continuity conditions are never imposed on the outside surfaces of the two-node problem, since the two-node problem is deliberately formulated to be spatially decoupled. Continuity is assured in the formulation of the FDM based equations.

Equations (4.99) through (4.100) constitute the 8G system of equations needed to be solved at each interface. This matrix system, after taking advantage of its reducability and by noting that the even-moment expansion coefficients don't change whether the node is on the left or right of a two-node problem, can be reduced to smaller systems which can be solved quite efficiently[30]. Table (4.1) illustrates this more efficient arrangement of unknowns for the case of G = 2.

Table 4.1 Non Zero entries in the 16 by 16 two-node NEM Problem.

Eqn	Grp	Nod	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
0th Moment	1	l	x		x													
0th Moment	2	l		x		x												
2nd Moment	1	l	x	x	x	x												
2nd Moment	2	l	x	x	x	x												
0th Moment	1	l+1					x		x									
0th Moment	2	l+1						x		x								
2nd Moment	1	l+1					x	x	x	x								
2nd Moment	2	l+1					x	x	x	x								
1st Moment	1	l									x	x	x	x				
1st Moment	2	l									x	x	x	x				
1st Moment	1	l+1													x	x	x	x
1st Moment	2	l+1													x	x	x	x
Cur Con	1		x		x		x		x		x		x		x		x	
Cur Con	2			x		x		x		x		x		x		x		x
Fix Dis	1		x				x				x				x			
Fix Dis	2			x				x				x				x		

Unknown	Node	Group	Exp. Coef.*
a	l	1	2
b	l	2	2
c	l	1	4

Unknown	Node	Group	Exp. Coef.*
d	l	2	4
e	l+1	1	2
f	l+1	2	2
g	l+1	1	4
h	l+1	2	4
i	l	1	1
j	l	2	1
k	l	1	3
l	l	2	3
m	l+1	1	1
n	l+1	2	1
o	l+1	1	3
p	l+1	2	3

* Refers to order of polynomial that transverse integrated flux expansion coefficient is associated with.

In PARCS, the two-node problems are solved by utilizing the analytic solution to the 8G x 8G matrix system. This was accomplished by employing symbolic manipulator software to produce the FORTRAN code segment used in PARCS. This approach is computationally more efficient than utilizing a direct matrix solver (e.g., LU decomposition); however, it limits the values of G to those directly programmed for. Also note that on boundaries special treatments of the two-node problems are required. Depending upon the specified boundary condition (BC), one-node problems may originate (e.g., zero flux BC), or on interior axis geometry unfolding may be required to create a two-node problem (e.g., cyclic BC).

Solutions of the two-node problems provide NEM evaluated values of the currents on all surfaces for specified values of the node average fluxes (recall they were assumed known in solving the two-node problems). To correct the FDM based expression for the surface current, the following approach is utilized. The coupling coefficient update to the FDM equation can be implemented by simply expressing the FDM net surface current at the *x*+ face of node l as follows:

$$\bar{J}^{l,FDM}_{gx+} = -\frac{D^{l,FDM}_{gx+}}{\frac{\Delta x^{l}+\Delta x^{l+1}}{2}}(\bar{\phi}^{l+1}_{g}-\bar{\phi}^{l}_{g})-\frac{\tilde{D}^{l,NEM}_{gx+}}{\frac{\Delta x^{l}+\Delta x^{l+1}}{2}}(\bar{\phi}^{l+1}_{g}+\bar{\phi}^{l}_{g}) \quad . \tag{4.100}$$

The first term on the RHS is the normal FDM approximation for a box scheme, where $D^{l,FDM}_{gx+}$ is the actual FDM diffusion coupling coefficient between nodes l and l+1. It is given by

$$D_{gx+}^{l,\,FDM} = \frac{D_g^l D_g^{l+1}(\Delta x^l + \Delta x^{l+1})}{D_g^l \Delta x^l + D_g^{l+1}\Delta x^{l+1}}. \tag{4.101}$$

The second term on the RHS represents the nonlinear NEM correction applied to the FDM scheme. The (+) sign between the flux values in the second term of Equation (4.87) is purposely there to improve the convergence behavior of the nonlinear iterative method. Note that if $\tilde{D}_{gx+}^{l,\,NEM}$ is zero, which it initially is in PARCS's implementation, then Equation corresponds to the standard FDM definition of the net surface current. This is the basis for the FDM option within PARCS, where now two-node problem solves and coupling coefficients updates are never completed. The value of $\tilde{D}_{gx+}^{l,\,NEM}$ is determined by setting Equation equal to the NEM two-node predicted surface current value, using the associated node average flux values in Equation and solving for this quantity.

Summarizing, to apply a NEM update after ΔN_0 outer iterations of the FDM routine, one solves the two-node problem at a given interface, then (with the expansion coefficients known for that interface) one calculates the NEM estimate of the net surface current using Equation . Finally, one equates this result to Equation , and solves for the value of $\tilde{D}_{gx+}^{l,\,NEM}$ which will be used in the subsequent set of FDM iterations.

5. PIN POWER RECONSTRUCTION AND DETECTOR RESPONSE

5.1 Pin Power Reconstruction

Fuel pin power information is important for the safety assessment of a core fuel loading since it is required in the determination of the peak linear heat generation rate and the minimum Departure from Nucleate Boiling Ratio (DNBR). In most of the modern neutronics codes such as PARCS which employ advanced nodal methods, the pin power information is obtained through a pin power reconstruction or "dehomogenization" process. The terms reconstruction and dehomogenization arise from the fact that the neutron flux calculation performed in a nodal code is based on homogenized nodal cross sections so that the resulting intranodal flux distribution can not reflect any local heterogeneity within an assembly. A reconstruction process is thus required to incorporate the actual heterogeneous structure of each assembly into the smoothly-varying intranodal flux distribution.

The pin power reconstruction process involves a fundamental assumption; that is, detailed pin-by-pin distributions within an assembly can be estimated by the product of a global intranodal distribution and a local heterogeneous form function. The form function accounts for assembly heterogeneities caused by water holes, burnable absorber pins, enrichment zonings, etc. and it is generated for each fuel assembly type by a lattice physics code at the same time when the homogenized nodal cross sections are generated. The assumption of separability of the global intranodal flux and the local form function is commonly adopted in various pin power reconstruction methods that have been extensively researched in the past two decades.[6],[7] The assumption has been shown to be valid such that the reconstruction methods are regularly used in best estimate reactor safety analysis of commercial nuclear reactors. In the following section, the development of the intranodal flux calculation method is presented.

5.1.1 PARCS Pin Power Reconstruction Method

In most advanced nodal methods, the available intranodal flux distributions are restricted to one-dimensional flux shapes in each of the three spatial directions. This limitation arises from the commonly employed transverse integration procedure used to derive the nodal coupling equations. Unfortunately, the multiplication of the three one-dimensional flux distribution for a node does not yield an accurate multidimensional intranodal flux distribution because the flux distribution is not separable in space. Instead, a unique method is required to determine the multidimensional intranodal flux distribution once the nodal solution is obtained for a given core condition. Because of rather smooth axial variations of the flux, however, it is normally assumed that the radial and axial dependences of the intranodal flux are separable so that all the current pin power reconstruction methods deal primarily with a two-dimensional problem.

One of the pin power reconstruction methods which is widely accepted assumes that the radial dependence of the pin power can be described by a two-dimensional polynomial expansion of flux, augmented by a set of exponential functions applied only to the thermal energy group. In the original work by Koebke, the number of basis functions used in the expansion was 21 per energy group including low order cross terms such as x^2y^2 and fourth order terms in each direction. It was later justified by Rempe that the number of expansion terms could be reduced to 13. In

order to determine the 13 expansion coefficients, the same number of constraints should be supplied. Some of the constraints such as node average flux, surface average fluxes and currents are directly available from the nodal solution, but they constitute at most only 9 constraints per node and per group for a rectangular node. The other 4 more constraints required for a unique expansion are specified by the flux values at the 4 corners. Since the corner point fluxes are not nodal unknowns that are solved for during the nodal solution process, additional approximations were made such that the two-dimensional flux distribution around a corner is approximated by a polynomial lower in order than what is used to represent the entire flux inside a node.

Another approach in pin power reconstruction is to use analytic functions, each of which is the solution of the two-dimensional neutron diffusion equation, as the basis function of the expansion. In this "analytic" approach, the determination of the intranodal flux turns into a Dirichlet problem in which a two-dimensional partial differential equation is solved exactly with the function values specified at the boundary. Boer and Finnemann claimed first that continuous function values can be constructed at each boundary from the nodal solution by using so called Method of Successive Smoothing (MSS) so that the Dirichlet problem can be solved with any number of boundary conditions. However, they concluded that the solution accuracy does not gain much by using more than 8 boundary conditions, which are 4 surface average and 4 corner point fluxes. Therefore, in the analytic approach, 8 analytic expansion functions are used per node. In contrast to the polynomial approach in which there is no coupling between the flux expansions of the two energy groups, the analytic approach involves coupling between groups so that the intranodal flux distribution for each group is represented by 16 analytic expansion functions consisting of trigonometric and exponential functions. Because of the sophistication involved, the analytic approach has superior accuracy, as demonstrated by Boer and Finnemann in Reference [7].

Because of the superiority of the analytic approach, the intranodal flux calculation method of PARCS was based on the analytic approach. However, the boundary conditions were chosen differently. Namely, the *surface average current* were used *instead of the surface average flux*. This choice was dictated primarily by the fact that the nodal neutron balance, which was satisfied during the nodal solution process, is not ensured with the resulting 2D intranodal flux if the surface average flux is used as the boundary condition. In other words, the surface average current obtained by taking the derivative of the 2D flux distribution and then integrating over a (perpendicular) surface is not the same as the surface average current obtained during the nodal solution process; which means that the intranodal distribution is not perfectly consistent with the nodal solution. In contrast, the nodal neutron balance of the resulting 2D flux distribution is always guaranteed if the surface average currents are used as the boundary condition during the reconstruction process. The choice of surface current is also supported by the nonlinear nodal solution scheme of PARCS in which the surface average net current is easily obtained from the converged coarse mesh finite different (CMFD) solution, while it requires additional two-node calculations to determine the surface average fluxes. Because of the choice of the surface current, as well as the corner flux as the boundary conditions, the problem now turns to a mixed boundary value problem in which the Dirichlet and Neumann boundary conditions coexist. The analytic 2-dimensional flux expansion is presented in the following section for a given set of 4 surface currents and 4 corner point fluxes.

For the estimation of the corner point fluxes, Boer and Finnemann used the MSS approach. The MSS approach is, however, based on the assumption of linear flux variation around a corner and the resulting expression for the corner flux involves only 4 node average and 4 surface aver-

age fluxes around the corner point. The MSS approach for determining corner point fluxes is computationally efficient, but would lead to nonnegligible errors in the estimated corner flux due to the low order approximation, particularly in the regions where the flux variation is severe, such as at the corner point of a UO2/MOX checkerboard. Therefore, an improved corner flux evaluation method is used in PARCS that employs the so called corner point balance (CPB) or "no net source at corner point" as an additional physical constraint. Given a two-dimensional flux expansion involving already *known* surface currents and *unknown* corner point fluxes, it is possible to write a CPB equation for each corner. The set of resulting CPB equations constitute a linear system with the corner fluxes being the unknowns. The linear system needs to be solved simultaneously. The derivation of the CPB equation, the matrix structure, and the solution process is presented in Section 5.1.2.1. It is followed by the description of the whole solution sequence to determine the homogenous intranodal flux distribution in Section 5.1.2.2.

Conventionally, the form function to be used to obtain the pin-wise power was the one group power form function, which combines the power generation rates of both thermal and fast neutrons at each fuel pin. The one group power form function causes negligible errors in ordinary fuel loading in which the fast fission rate is negligible. In the MOX-fueled core, however, the contribution from the fast fission is nontrivial because of the spectrum hardening. In order to accommodate a wider range of applications including MOX fuels, two-group power form functions will be used in PARCS. The definition of the two-group power form functions and the guidelines to generate them are given in the following section.

5.1.2 Two-Dimensional Analytic Function Expansion for Homogeneous Intranodal Flux

Suppose that the two-group nodal neutron balance equation, given by the following for each node:

$$\sum_{u=x,y,z} (J_u^r - J_u^l) + A\bar{\Phi} = s \tag{5.1}$$

where

$$A = \begin{bmatrix} \Sigma_{r1} - \lambda\nu\Sigma_{f1} & -\lambda\nu\Sigma_{f2} \\ -\Sigma_{12} & \Sigma_{r2} \end{bmatrix} \tag{5.2}$$

and node-averaged fluxes ($\bar{\Phi}$) and surface-averaged currents (J_u^s) are two-element vectors, has been solved during the nodal solution process so that node average fluxes and surface average currents have been determined. The two-dimensional intranodal flux distribution is then to be determined such that it is consistent with the solution of Eq. (5.1) and it also satisfies the two-dimensional neutron diffusion equation for the node. The two-dimensional problem for a node is a boundary value problem in which the boundary conditions are specified at four surfaces. As the boundary condition, the most obvious choice would be the surface average currents that were determined during the nodal calculation. In addition to the surface average current, more boundary conditions are needed for the accurate prediction of the intranodal flux distribution. In the following, the solution of a two-dimensional neutron diffusion equation for a node satisfying a finite number of boundary conditions is presented.

The two-dimensional neutron diffusion equation needed to be solved to determine the intranodal flux distribution is derived by integrating the three-dimensional diffusion equation along the axial direction from the lower to the upper boundary of each plane. The integration yields:

$$-\boldsymbol{D}\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)\Phi(x,y)+\boldsymbol{A}\Phi(x,y) = \boldsymbol{s}(x,y)-\boldsymbol{L}_z(x,y) \tag{5.3}$$

where $\boldsymbol{s}(x,y)$ and $\boldsymbol{L}_z(x,y)$ are vectors consisting of two group-wise elements for the average transient fixed source and the transverse leakage in the z-direction, and $\boldsymbol{D}$ is the diagonal diffusion coefficient matrix.

Eq. (5.3) can be solved analytically given the boundary conditions specified as functions at the four boundaries. The analytic solution consists of both homogeneous and particular parts. The particular solution would represent the effect of the transient fixed source and the axial leakage on the intranodal flux distribution. In general, the spatial dependence of the source appearing on the right hand side (RHS) would be independent of that of the flux. However, the spatial dependence of the RHS source terms can be approximated by the spatial dependence of flux itself based on the assumption of separability of axial and radial dependencies which results from the fact that the radial shape is determined primarily by radial and not radial heterogeneity, and also based on the fact that the radial distribution of the transient fixed source would closely follow the shape of the flux itself. Specifically, the following assumption is introduced for the subsequent derivation:

$$\boldsymbol{s}(x,y)-\boldsymbol{L}_z(x,y) \cong -\underline{\Sigma}_{ps}\Phi(x,y) \tag{5.4}$$

where the pseudo source cross section is defined as:

$$\Sigma_{ps}^{g} \equiv \frac{\bar{L}_z^g - \bar{S}_g}{\bar{\phi}_g} \quad . \tag{5.5}$$

This approximation will be acceptable as long as the relative magnitude of the RHS source terms are small. The assumption of Eq. (5.5) converts Eq. (5.3) to a homogeneous equation. By defining the effective removal cross section as:

$$\Sigma_{rg} \equiv \Sigma_{rg} + \Sigma_{ps}^{g} \tag{5.6}$$

Eq. (5.3) is converted to a form encountered in a pure two-dimensional boundary value problem in which no independent source appears on the RHS. In the following, it is assumed that the cross section matrix given in Eq. (5.2) is defined with the effective removal cross section.

5.1.2.1 Analytic Function Expansion for 2D Boundary Value Problem

The homogeneous solution to Eq. (5.3) can be obtained by the method of separation of variables that yields:

$$\frac{\partial^2\Phi}{\partial x^2}+\frac{\partial^2\Phi}{\partial y^2} = -(B_x^2+B_y^2)\Phi = -B^2\Phi \tag{5.7}$$

with $B_x = B\cos\theta_k$ and $B_y = B\sin\theta_k$ for any value of θ_k. The eigenvalue B^2 is determined from the condition for a nontrivial solution that reads:

$$\det(\boldsymbol{A}+\boldsymbol{D}B^2) = 0 \tag{5.8}$$

Eq. (5.8) has two roots for B^2 and will be denoted by B_0^2 and B_1^2. Note that the two eigenvalues as well as the fast thermal flux ratios r and s in Eq. (5.11) are the same as those calculated for the analytic nodal method (ANM) calculations in PARCS.

In terms of the two eigenvalues and the arbitrary angle θ_k, the homogenous solution is then obtained as an infinite series:

$$\Phi(x,y) = \begin{bmatrix}\phi_1(x,y)\\ \phi_2(x,y)\end{bmatrix} = \begin{bmatrix} r & s\\ 1 & 1\end{bmatrix}\begin{bmatrix}\psi_0(x,y)\\ \psi_1(x,y)\end{bmatrix} = S\Psi(x,y) \tag{5.9}$$

where the modal flux function is represented as:

$$\psi_m(x,y) = \sum_{k=1}^{\infty} \langle a_k^{(m)} \exp(iB_m(x\cos\theta_k + y\sin\theta_k)) + b_k^{(m)} \exp(-iB_m(x\cos\theta_k + y\sin\theta_k))\rangle , \tag{5.10}$$

and the modal fast-to-thermal flux ratios and the similarity transformation matrix are defined as:

$$r = \frac{\Sigma_{r2} + D_2 B_0^2}{\Sigma_{12}} , \quad s = \frac{\Sigma_{r2} + D_2 B_1^2}{\Sigma_{12}} , \quad S \equiv \begin{bmatrix} r & s\\ 1 & 1\end{bmatrix} . \tag{5.11}$$

As in the Analytic Function Expansion Method (AFEN)[17], a practical solution is formulated by truncating the series with four values of θ_k, which are 0, $\frac{\pi}{4}$, $\frac{\pi}{2}$, and $\frac{3\pi}{4}$. Eq. (5.10) can then be recast such that it involves trigonometric and/or hyperbolic functions as:

$$\begin{aligned}\psi_m(x,y) = {} & a_1^{(m)} sn(B_m x) + a_2^{(m)} cn(\tilde{B}_m x) + a_3^{(m)} sn(B_m y) + a_4^{(m)} cn(\tilde{B}_m y)\\ & + a_5^{(m)} sn(\frac{B_m}{\sqrt{2}}x) cn(\frac{\tilde{B}_m}{\sqrt{2}}y) + a_6^{(m)} sn(\frac{B_m}{\sqrt{2}}x) sn(\frac{B_m}{\sqrt{2}}y)\\ & + a_7^{(m)} cn(\frac{\tilde{B}_m}{\sqrt{2}}x) sn(\frac{B_m}{\sqrt{2}}y) + a_8^{(m)} cn(\frac{B_m}{\sqrt{2}}x) cn(\frac{B_m}{\sqrt{2}}y)\end{aligned} \tag{5.12}$$

where

$$sn(B_m x) = \frac{\sin(B_m x)}{B_m} \text{ or } \frac{\sinh(B_m x)}{B_m}$$

$$cn(\tilde{B}_m x) = \begin{cases}\cos(B_m x) \text{ or } \cosh(B_m x) \text{ if } \mathrm{B_m} \geq B_{m,\min}\\ \cos(B_{m,\min} x) \text{ or } \cosh(B_{m,\min} x) \text{ if } \mathrm{B_m} < B_{m,\min}\end{cases}$$

$B_{m,\min}$ is the lower bound for $\mathrm{B_m}$

Note that there are 8 coefficients to be determined in Eq. (5.12). Since there are two modes (the fundamental mode for m=0, and the harmonic mode for m=1), the total number of unknown coefficients then becomes 16. These 16 coefficients can be determined if and only if 16 boundary con-

ditions are specified. Considering two energy groups, the number of boundary conditions needed per group is thus 8.

As discussed earlier, there are four surfaced-averaged currents available as the boundary conditions. The additional four conditions can be specified at the corners of the node. The corner flux values correspond to the choice of $\frac{\pi}{4}$ and $\frac{3\pi}{4}$ as the value of θ_k. If the fluxes at the corners are available, they can be used as the additional boundary conditions so that all the coefficients in Eq. (5.12) can be determined uniquely. The problem is, however, that the corner point fluxes are not determined during the reactor nodal solution process. The commonly used method is a corner point flux balance condition and is described in detail in the following section.

As an alternative to using the surface-averaged currents, it is plausible to use the surface-average fluxes such as is done in Boer and Finnemann's work. There is, however, a problem associated with this alternative. There is no guarantee that the surface average currents and the node average flux calculated based on the flux expansion given in Eq. (5.12) will be the same as those that were already determined during the nodal solution process. This could result in an intranodal flux distribution which is inconsistent with the nodal solution. To avoid this problem, there would have to be a constant term added to the expansion of Eq. (5.12) which would artificially adjust the magnitude of the nodal flux. In contrast, the surface current approach preserves the surface currents in the intranodal expansion and hence the node average flux is also guaranteed to be conserved in the resulting intranodal flux by virtue of the nodal balance given in Eq. (5.1) and also by the attributes of the analytic solution. This is the primary reason why the surface average currents are chosen as the boundary conditions in the development here.

5.1.2.2 Determination of Expansion Coefficients

Suppose that the corner point fluxes are known, then it is possible to find the 8 expansion coefficients of Eq. (5.12) in terms of two sets of 4 node average currents and 4 corner fluxes. This subsection derives the explicit expression for each coefficient. The starting point of the derivation is to convert the specified flux and current values to the modal fluxes and their derivatives. From the relation given in Eq. (5.9), the modal fluxes is represented in terms of fluxes as:

$$\Psi = \mathcal{S}^{-1}\Phi = \frac{1}{r-s}\begin{bmatrix} 1 & -s \\ -1 & r \end{bmatrix}\begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} = \begin{bmatrix} \psi_0 \\ \psi_1 \end{bmatrix} \quad . \tag{5.13}$$

Eq. (5.13) can be used to determine the modal fluxes at each of the four corner points. On the other hand, the following relation holds for the derivative of the modal flux:

$$\nabla_u \Psi = -\mathcal{S}^{-1}\mathcal{D}^{-1}\bar{J}_u \quad . \tag{5.14}$$

Once the modal fluxes at the corners and the average modal flux derivatives along a surface are determined by Eq. (5.13) and Eq. (5.14), in which the coupling between energy groups is incorporated via the similarity transform matrix $\boldsymbol{S}$, the coefficients for the modal flux expansion for each mode can now be determined independently of the other mode and the same derivation can be applied to both modes. In the following derivation, therefore, the coefficients are determined for only one mode and the index for indicating the mode (*m=0 or 1*) is omitted for brevity.

Figure 5.1 shows the coordinate system and the boundary conditions for a rectangular node for which the modal flux expansion coefficients are determined. By inserting the proper coordinate values of x and y into Eq. (5.12), the following relations are obtained for the corner fluxes:

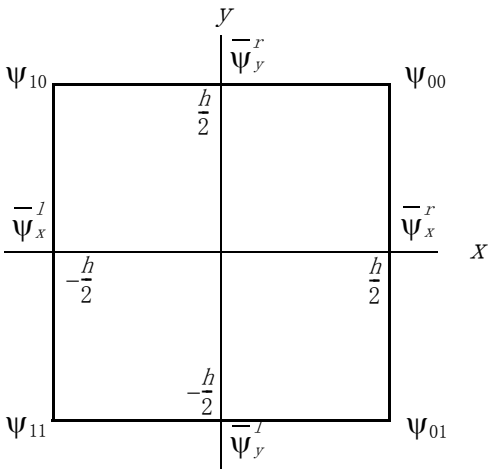


Figure 5.1: Coordinate and Boundary Conditions for Intranodal Expansion

$$\begin{aligned}
\Psi_{00} &= Sa_1 + \underset{\sim}{C}a_2 + Sa_3 + \underset{\sim}{C}a_4 + \tilde{\underset{\sim}{X}}_c a_5 + \tilde{S}_2 a_6 + \tilde{\underset{\sim}{X}}_c a_7 + \tilde{C}_2 a_8 \\
\Psi_{10} &= -Sa_1 + \underset{\sim}{C}a_2 + Sa_3 + \underset{\sim}{C}a_4 - \tilde{\underset{\sim}{X}}_c a_5 - \tilde{S}_2 a_6 + \tilde{\underset{\sim}{X}}_c a_7 + \tilde{C}_2 a_8 \\
\Psi_{11} &= -Sa_1 + \underset{\sim}{C}a_2 - Sa_3 + \underset{\sim}{C}a_4 - \tilde{\underset{\sim}{X}}_c a_5 + \tilde{S}_2 a_6 - \tilde{\underset{\sim}{X}}_c a_7 + \tilde{C}_2 a_8 \\
\Psi_{01} &= Sa_1 + \underset{\sim}{C}a_2 - Sa_3 + \underset{\sim}{C}a_4 + \tilde{\underset{\sim}{X}}_c a_5 - \tilde{S}_2 a_6 - \tilde{\underset{\sim}{X}}_c a_7 + \tilde{C}_2 a_8
\end{aligned} \tag{5.15}$$

where the node dependent trigonometric or hyperbolic function parameters are defined as:

$$\begin{aligned}
&S = SN(\frac{\kappa h}{2}),\ \mathrm{C}=\mathrm{CN}(\frac{\kappa h}{2}),\ \tilde{S} = SN(\frac{\kappa h}{2\sqrt{2}}),\ \tilde{\mathrm{C}}=\mathrm{CN}(\frac{\kappa h}{2\sqrt{2}}), \\
&\tilde{\mathrm{S}}_2 = \tilde{S} * \tilde{S},\ \tilde{\mathrm{C}}_2 = \tilde{\mathrm{C}} * \tilde{\mathrm{C}},\ \tilde{\underset{\sim}{\mathrm{X}}}_{\mathrm{c}} = \tilde{S} * \tilde{\underset{\sim}{C}},\ \underset{\sim}{\mathrm{C}}=\mathrm{CN}(\frac{\underset{\sim}{\kappa} h}{2}), \tilde{\underset{\sim}{C}} = \mathrm{CN}(\frac{\underset{\sim}{\kappa} h}{2\sqrt{2}})
\end{aligned} \tag{5.16}$$

The relations for the surface average derivatives of the modal flux are somewhat more involved. They can be obtained after differentiation along a direction and then integration along the perpendicular direction. For example, the x-directional derivative averaged over the right hand side surface of the node is defined by:

$$\bar{\psi}_x^r = \left.\overline{\frac{\partial \psi}{\partial x}}\right|_r = \frac{1}{h}\int_{-\frac{h}{2}}^{\frac{h}{2}} \left.\frac{\partial \psi}{\partial x}\right|_{x=\frac{h}{2}} dy \quad . \tag{5.17}$$

Eq. (5.17) can be evaluated for the modal flux given in Eq. (5.12). Similar definitions for the other three surface average derivatives yields the following four relations for the derivatives:

$$\begin{aligned}
\bar{\Psi}_x^r &= Ca_1 + \Omega\tilde{\kappa}^2 Sa_2 + 2\tilde{\tilde{X}}_{sh}a_5 + \Omega\kappa^2\tilde{S}_{2h}a_8 \\
\bar{\Psi}_x^l &= Ca_1 - \Omega\tilde{\kappa}^2 Sa_2 + 2\tilde{\tilde{X}}_{sh}a_5 - \Omega\kappa^2\tilde{S}_{2h}a_8 \\
\bar{\Psi}_y^r &= Ca_3 + \Omega\tilde{\kappa}^2 Sa_4 + 2\tilde{\tilde{X}}_{sh}a_7 + \Omega\kappa^2\tilde{S}_{2h}a_8 \\
\bar{\Psi}_y^l &= Ca_3 - \Omega\tilde{\kappa}^2 Sa_4 + 2\tilde{\tilde{X}}_{sh}a_7 - \Omega\kappa^2\tilde{S}_{2h}a_8
\end{aligned} \tag{5.18}$$

where the Ω factor is 1.0, except when the relations are applied to the fundamental mode (*m=0*) and the nodal k_∞ is greater than the core k_{eff},

$$\tilde{\kappa} = \kappa_{\min,}\ \tilde{\tilde{X}}_{sh} = \frac{\tilde{\tilde{S}} * \tilde{C}}{h},\ \tilde{\tilde{S}} = \mathrm{SN}\left(\frac{\tilde{\kappa}h}{2\sqrt{2}}\right), \tilde{C} = \mathrm{CN}\left(\frac{\kappa h}{2\sqrt{2}}\right), \text{ and } \tilde{S}_{2h} = \frac{\tilde{S}_2}{h}$$

Eq. (5.15) and Eq. (5.18) can be solved simultaneously for the unknown a_i s. The solution is given as follows:

$$\begin{aligned}
a_6 &= \frac{\Psi_2}{\tilde{\tilde{S}}_2} \\
a_1 &= \frac{\Psi_1 - \beta\Psi_x^S}{S - \beta C};\ a_5 = \frac{\Psi_1 - Sa_1}{\tilde{\tilde{X}}_c} \\
a_{3=} &\frac{\Psi_3 - \beta\Psi_y^S}{S - \beta C};\ a_7 = \frac{\Psi_3 - Sa_3}{\tilde{\tilde{X}}_c} \\
a_8 &= \frac{hS\Psi_4 - \frac{h\tilde{\tilde{C}}}{\tilde{\kappa}^2}(\Psi_x^D + \Psi_y^D)}{hS\tilde{C}_2 - 2\tilde{\tilde{C}}\tilde{S}_2\gamma} \\
a_2 &= \frac{\frac{h}{\tilde{\kappa}^2}\Psi_x^D - a_8\gamma\tilde{S}_2}{hS};\ a_4 = \frac{\frac{h}{\tilde{\kappa}^2}\Psi_y^D - a_8\gamma\tilde{S}_2}{hS}
\end{aligned} \tag{5.19}$$

where ,

$$\beta = \frac{h\tilde{\tilde{X}}_c}{2\tilde{\tilde{X}}_s}, \gamma = \left(\frac{\kappa}{\tilde{\kappa}}\right)^2$$

$$\begin{bmatrix}\psi_1\\ \psi_2\\ \psi_3\\ \psi_4\end{bmatrix} = \frac{1}{4}\begin{bmatrix}\psi_{00}-\psi_{10}-\psi_{11}+\psi_{01}\\ \psi_{00}-\psi_{10}+\psi_{11}-\psi_{01}\\ \psi_{00}+\psi_{10}-\psi_{11}-\psi_{01}\\ \psi_{00}+\psi_{10}+\psi_{11}+\psi_{01}\end{bmatrix}. \quad (5.20)$$

and

$$\begin{bmatrix}\Psi_x^S\\ \Psi_x^D\\ \Psi_y^S\\ \Psi_y^D\end{bmatrix} = \frac{1}{2}\begin{bmatrix}\bar{\Psi}_x^r+\bar{\Psi}_x^l\\ \Omega(\bar{\Psi}_x^r-\bar{\Psi}_x^l)\\ \bar{\Psi}_y^r+\bar{\Psi}_y^l\\ \Omega(\bar{\Psi}_y^r-\bar{\Psi}_y^l)\end{bmatrix} \quad . (5.21)$$

5.1.2.3 Considerations for the Corner Point Discontinuity

One of the important homogenization factors generated during the single assembly calculation is the Assembly Discontinuity Factors (ADF). They are required in nodal calculations to incorporate the fact that what is continuous in reality is the heterogeneous flux NOT the homogeneous flux. The use of ADFs significantly reduces the error due to homogenization which is unavoidable if they are not used. A similar concept can be applied to the pin power reconstruction process in which case the corner point discontinuity factor (CDF) is required. The CDF's are calculated during the single assembly calculation as the ratio of the assembly corner flux to the average flux. When the CDF is applied during the homogeneous intranodal flux calculation, the corner flux continuity should be imposed on the products of the actual corner flux of a node and the CDF of the node. Suppose that the product is defined as a heterogeneous corner point flux, then there are four different actual corner fluxes (one for each node surrounding the corner obtained by diving the imaginary flux by CDF).

Let Γ be a diagonal 2x2 matrix whose components are CDF's for the two energy groups and Φ be the heterogeneous corner flux. Then Eq. (5.13) should be modified as follows:

$$\Psi = \mathcal{S}^{-1}\Gamma^{-1}\Phi \quad (5.22)$$

In this manner, the corner point flux which is determined first is the heterogeneous corner point flux.

5.1.3 Corner Point Flux Evaluation

In order to determine the corner point fluxes, a new neutron balance equation is formulated at each of the four corners of the node. This is the corner point balance (CPB) condition which imposes the physical constraint that there be no accumulation of neutrons within an infinitesimally small volume. Suppose a small square box surrounding a corner point and the length of one side is *h*. The neutron balance for the box can be written as:

$$(J_x^{ne} - J_x^{nw} + J_x^{se} - J_x^{sw} + J_y^{ne} - J_y^{se} + J_y^{nw} - J_y^{sw})\frac{h}{2} + A\phi h^2 = Sh^2 \quad (5.23)$$

where J_u^d is the *u*-directional current at the surface of the box belonging to the node located in the *d*-direction. Total leakage obtained from Eq. (5.23) approaches to zero as the volume of the box becomes zero, as shown below:

$$L = \lim_{h\to 0}\left[\left(J_x^{ne}+J_y^{ne}\right)-\left(J_x^{nw}-J_y^{nw}\right)-\left(J_x^{sw}+J_y^{sw}\right)+\left(J_x^{se}-J_y^{se}\right)\right] = \lim_{h\to 0}(S-A\phi)2h = 0 \qquad (5.24)$$

The CPB condition given as Eq. (5.24) should be valid at every corner point. In order to complete the CPB equation for a corner, the directional currents must be calculated at all of the four nodes surrounding the corner. In the following, the contribution of a southwest node which corresponds to the third parenthesis term in Eq. (5.24) is derived. From this result, the contribution from the other three nodes can be inferred.

5.1.3.1 Derivation of CPB Coupling Relations

Consider the northeast corner of a node. The x and y directional derivatives of the modal flux at this corner are obtained as follows:

$$\begin{aligned}\left.\frac{\partial\psi}{\partial x}\right|_{00} &= a_1 C + a_2\Omega\tilde{\kappa}^2\tilde{S} + a_5\tilde{C}_2 + a_6\tilde{X} + a_7\frac{\Omega\tilde{\kappa}^2\tilde{S}_2}{2} + a_8\frac{\Omega\kappa^2\tilde{X}}{2}\\ \left.\frac{\partial\psi}{\partial y}\right|_{00} &= a_3 C + a_4\Omega\tilde{\kappa}^2\tilde{S} + a_5\frac{\Omega\tilde{\kappa}^2\tilde{S}_2}{2} + a_6\tilde{X} + a_7\tilde{C}_2 + a_8\frac{\Omega\kappa^2\tilde{X}}{2}\end{aligned} \qquad (5.25)$$

where $\tilde{\kappa} = \frac{\kappa}{\sqrt{2}}$. The addition of the two derivatives in Eq. (5.25) would yield the four contributions to the CPB equation for the corner from the present node. By using Eq. (5.19), the one fourth contribution to the CPB equation can be expressed as follows in terms of the surface average derivatives and the modal corner fluxes belonging to the present node:

$$\left.\frac{\partial\psi}{\partial x}\right|_{00} + \left.\frac{\partial\psi}{\partial y}\right|_{00} = c_{1c}\psi_{00} + c_{nc}\psi_{10} + c_{nc}\psi_{01} + c_{dc}\psi_{11} + \psi_j \qquad (5.26)$$

where the corner coupling coefficients and the current contribution terms are defined as:

$$\begin{aligned} c_{1c} &= \tau_5+\tau_6+\tau_7-\tau_8-\tau_9 && \text{for local corner,}\\ c_{nc} &= -\tau_5+\tau_7-\tau_8 && \text{for neighboring corner,}\\ c_{dc} &= \tau_5+\tau_6+\tau_7-\tau_8+\tau_9 && \text{for diagonal corner,}\\ \psi_j &= (\zeta_5+\zeta_6)(\psi_x^r+\psi_y^r)+(\zeta_5-\zeta_6)(\psi_x^l+\psi_y^l) && \text{for surface current source.}\end{aligned} \qquad (5.27)$$

with the intermediate parameters used above defined as:

$$\tau_1 = \frac{1}{S-\beta C}, \tau_2 = \frac{1}{hS\tilde{C}_2 - 2\underset{\sim}{C}\tilde{S}_2\gamma}, \tau_3 = \frac{1}{\tilde{X}_s}$$
$$\tau_4 = \tilde{C}_2 + \frac{\kappa^2}{2}\Omega\tilde{S}_2, \tau_5 = \frac{\tilde{C}}{2\tilde{S}}, \tau_6 = \frac{1}{2}C\tau_1 \tag{5.28}$$
$$\tau_7 = \frac{1}{2}\Omega\alpha\kappa S\tilde{X}\tau_2, \tau_8 = \frac{1}{2}\Omega\kappa^2\underset{\sim}{S}\tilde{S}_2\tau_2, \tau_9 = \frac{1}{4}hC\tau_1\tau_3\tau_4$$
$$where\ \alpha = \frac{\kappa h}{2}$$

and

$$\zeta_1 = \frac{1}{2}\alpha C\tau_1, \zeta_2 = \frac{1}{2}h\underset{\sim}{S}\tilde{C}_2\tau_2, \zeta_3 = \frac{\alpha\gamma}{\kappa}\tilde{X}\underset{\sim}{C}\tau_2, \tag{5.29}$$
$$\zeta_4 = \frac{1}{4}hS\tau_1\tau_3\tau_4, \zeta_5 = \zeta_4 - \zeta_1, \zeta_6 = \zeta_2 - \zeta_3$$

Note that the intermediate parameters need to be calculated separately for each mode.

Eq. (5.26) can not be used directly for setting the CPB equation since it involves only the modal fluxes. The actual current should be calculated considering the relation between the modal flux and the actual flux. For this, Eq. (5.26) needs to be transformed into a matrix equation first, that involves both fundamental and harmonic modal fluxes. Then multiplying the diffusion coefficient matrix and the similarity transform matrix to the matrix form of Eq. (5.26) yields:

$$\begin{aligned} J_x^{00} + J_y^{00} &= -DS\left(\left.\frac{\partial\psi}{\partial x}\right|_{00} + \left.\frac{\partial\psi}{\partial y}\right|_{00}\right) \\ &= -DS(c_{1c}\Psi_{00} + c_{nc}\Psi_{10} + c_{nc}\Psi_{01} + c_{dc}\Psi_{11} + \Psi_j) \\ &= -\tilde{c}_{1c}\Phi_{00} - \tilde{c}_{nc}\Phi_{10} - \tilde{c}_{nc}\Phi_{01} - \tilde{c}_{dc}\Phi_{11} + J_s \end{aligned} \tag{5.30}$$

where

$$c_i = \begin{bmatrix} c_i^0 & 0 \\ 0 & c_i^1 \end{bmatrix} \text{ and } \tilde{c}_i = DSc_iS^{-1} \text{ for } i \in \{1c, nc, dc\} \tag{5.31}$$

and

$$J_S = -DS\Psi_j \quad . \tag{5.32}$$

Note that if CDF are used then the corner flux coupling coefficients in Eq. (5.31) should be changed to:

$$\tilde{c}_i = DSc_iS^{-1}\Gamma^{-1} \tag{5.33}$$

because of Eq. (5.22).

Eq. (5.30) now completely determines the contribution of the southwest node of a corner to the CPB equation of the corner. Here, the surface current source term J_S is known from the surface currents of all the four nodes while the corner fluxes at the four corners are the unknowns to be solved for. The coupling matrices, $\tilde{c}_i$, are 2x2 matrices which are determined solely by the node properties.

5.1.3.2 CPB Linear System

The partial CPB equation of Eq. (5.30) for a node can be written for all the other three nodes. Summing up all the four partial contributions then yields a CPB equation for a corner. The CPB equation involves the corner fluxes at the eight nearest corner points as well as at the flux at the point for which the CPB equation applies. In terms of the variables defined in Figure 5.2,

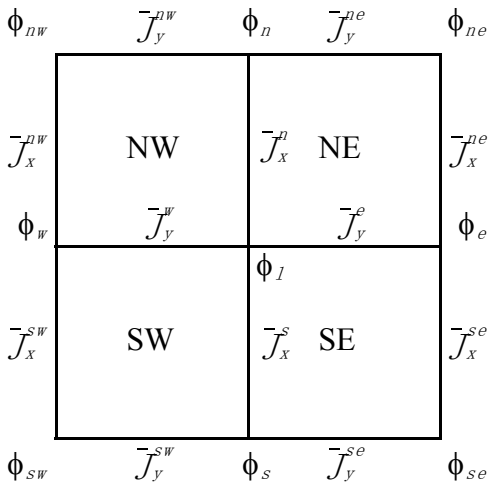


Figure 5.2: Variable Definition for Corner Flux Calculation

the complete CPB equation for a corner is obtained as:

$$\begin{aligned}&\tilde{c}_{dc}^{ne}\Phi_{ne}+\tilde{c}_{dc}^{nw}\Phi_{nw}+\tilde{c}_{dc}^{sw}\Phi_{sw}+\tilde{c}_{dc}^{se}\Phi_{se}+(\tilde{c}_{nc}^{ne}+\tilde{c}_{nc}^{nw})\Phi_n+(\tilde{c}_{nc}^{nw}+\tilde{c}_{nc}^{sw})\Phi_w\\&(\tilde{c}_{nc}^{sw}+\tilde{c}_{nc}^{se})\Phi_s+(\tilde{c}_{nc}^{se}+\tilde{c}_{nc}^{ne})\Phi_e+(\tilde{c}_{1c}^{ne}+\tilde{c}_{1c}^{nw}+\tilde{c}_{1c}^{sw}+\tilde{c}_{1c}^{se})\Phi_1=J_s^{ne}+J_s^{nw}+J_s^{sw}+J_s^{se}\end{aligned} \tag{5.34}$$

The CPB equation of Eq. (5.34) can be set up for all the corners and the set of all CPB equations forms a linear system that needs to be solved simultaneously. The CPB linear system is a nine stripe block diagonal system and would be very diagonally dominant because the CPB equation at a corner is mostly dominated by the flux at that corner rather than by the neighboring ones. Thus it can be solved with a conventional Gauss-Seidel iterative scheme, without the necessity of invoking more sophisticated method.

5.1.4 Solution Sequence for Homogeneous Flux Distribution

The process of determining the homogeneous intranodal flux distribution proceeds as follows:

a. Determine the node-wise AFEN parameters such as *r*, *s*,

$$\frac{\sin(\frac{B_0 h}{2})}{B_0}, \cos(\frac{B_0 h}{2\sqrt{2}}) \text{ and } \cosh(\frac{B_0 h}{2\sqrt{2}})$$

b. Calculate the linear system coefficients for the CPB linear system

c. Perform the Gauss-Seidel iteration to determine the corner point fluxes

d. Determine all the 8 expansion coefficients in Eq. (5.12) by using the four corner point fluxes and the four surface average currents

e. Calculate the volume-averaged intranodal flux at each pin cell.

5.1.5 Definition of Two-Group Form Functions

The form functions are needed for dealing with the local heterogeneity in material composition within a fuel assembly. They are generated along with the homogenized group constant during the single assembly calculation. The form functions are defined groupwise as:

$$f_g(x,y) = \frac{\kappa\Sigma_{fg}(x,y)\phi_g(x,y)}{\kappa\bar{\Sigma}_{fg}\bar{\phi}_g} \tag{5.35}$$

By using the form function and the homogeneous intranodal flux determined at a pin position, the pin power is now obtained as:

$$p(x,y) = \sum_{g=1}^{G} \kappa\bar{\Sigma}_{fg}\phi_g(x,y) f_g(x,y) \tag{5.36}$$

Note here that the customary assumption is being made that the form function in an actual core condition does not differ much from the form functions generated for a single assembly geometry employing the reflective boundary conditions.

5.2 Detector Response

The detector response can be written as

$$R_I(x,y,z) = \sum_{l \in I} v_{I,l} \sum_{g=1}^{G} \bar{\Sigma}^{d}_{l,g}\phi_{lg}(x,y) f_{lg}(x,y) \tag{5.37}$$

where l represents the nodes adjacent to detector I, $\phi_{lg}(x,y)$ is the gth group homogenous flux at location (x,y) in node l which can be obtained using the same method as the pin power reconstruction technique described in the previous sectio, and $f_{lg}(x,y)$ is the form function

which is determed from the fuel assembly lattice calucation as the ratio of the fluxes from the heterogenous and homogeous models.

$$f_{1g}(x,y) = \frac{\phi_{1g}^{Het}(x,y)}{\phi_{1g}^{Hom}(x,y)} \tag{5.38}$$

The lattice calculations are normally performed using a single fuel assembly with reflective boundary conditions in which case the flux for the homogenous model is the same as the assembly average flux. In this case the form function can be defined as:

$$f_{1g}(x,y) = \frac{\phi_{1g}^{Het}(x,y)}{\bar{\phi}_{1g}^{Het}} \tag{5.39}$$

where $\bar{\phi}_{1g}^{Het}$ is the assembly average flux.

$\bar{\Sigma}_{1,g}^{d}$ in equation (5.37) is the detector cross section which can be obtained from the assembly calculation:

$$\bar{\Sigma}_{1,g}^{d} = \frac{1}{\phi_{1g}^{Het}(x,y)A_{I,1}} \int_{A_{I,1}} \int \phi_{1g}^{Het}(x',y')\Sigma_{1,g}dx'\,dy' \tag{5.40}$$

where $A_{I,1}$ is the area of fuel assembly 1 which is occupied by detector I, and $\phi_{1g}^{Het}(x,y)$ is the flux at the center of the detector.

BWR detectors are located at the corners of fuel assemblies. The form function for a detector in a BWR is the same as the corner point discontiniuty factor. The heterogenous corner point flux are available from the pinpower reconstruction algorithm:

$$\phi_{1g}^{c,Het}(x,y) = \phi_{1g}(x,y)\, f_{1g}(x,y) \tag{5.41}$$

Therefore equation (5.37) can be simplified for BWR detectors as:thm:

$$R_I(x,y,z) = \sum_{1 \in I} V_{I,1} \sum_{g=1}^{G} \bar{\Sigma}_{1,g}^{d} \phi_{1g}^{c,Het}(x,y) \tag{5.42}$$

6. NEUTRON TRANSPORT METHODS

For some applications, the accuracy achievable with two group nodal diffusion methods is not adequate and therefore a multigroup transport option was added to PARCS. In this chapter, the P_N method is briefly introduced; and then the detailed derivation is presented for the multigroup SP_3 kernel[8] which was implemented in PARCS.

6.1 P_N Transport Methods

The steady-state Boltzmann transport equation without an external source can be written:

$$\Omega \cdot \nabla \psi(r,\Omega,E) + \Sigma_t(r,E)\psi(r,\Omega,E) = \int d\Omega' \int dE' \Sigma_s(r,\Omega' \to \Omega, E' \to E)\psi(r,\Omega',E') + \frac{1}{4\pi} S_f(r,E), \tag{6.1}$$

where

$$S_f(r,E) = \chi(E)\int dE' \nu\Sigma_f(r,E')\phi(r,E') \quad \phi(r,E) = \int d\Omega' \psi(r,\Omega',E)$$

, .

There are several methods for solving the Boltzmann equation for neutron transport and in the PARCS code the spherical harmonics method (P_N) is used. The angular flux and the differential scattering cross sections are expanded in Legendre polynomials:

$$\psi(r,\Omega,E) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \phi_n^m(r,E) Y_n^m(\Omega) \quad , \tag{6.2}$$

where

$$Y_n^m(\Omega) = Y_n^m(\theta,\varphi) = \left[\frac{(2n+1)(n-m)!}{(n+m)!}\right]^{1/2} P_n^m(\mu)\exp(im\varphi) \quad ,$$

where

$P_n^m(\mu) =$ associated Legendre polynomials, $\mu = \cos\theta$.

Using the orthogonal property of the spherical harmonics, the coefficients of Eq. (6.2) are determined as:

$$\phi_n^m(r,E) = \int d\Omega Y_n^m(\Omega)\psi(r,\Omega,E) \quad . \tag{6.3}$$

Since the angular flux is independent of the azimuthal angle in a one-dimensional geometry, the angular flux can be expressed as:

$$\Omega \cdot \nabla \psi(r,\Omega,E) \to \mu \frac{\partial}{\partial x}\psi(r,\Omega,E) \quad , \tag{6.4}$$

Therefore, the form of the P_N method can be easily introduced using a one-dimensional form written as follows:

$$\mu \frac{\partial \psi(x,\mu,E)}{\partial x} + \Sigma_t(x,E)\psi(x,\mu,E) = \int d\Omega' \int dE'\, \Sigma_s(x,\Omega'\cdot\Omega,E' \to E)\psi(x,\mu',E') + \frac{1}{2}S_f(x,E), \tag{6.5}$$

$$S_f(x,E) = \chi(E)\int dE'\, \nu\Sigma_f(x,E')\phi(x,E')$$

where ,

$$\Omega'\cdot\Omega = \mu_0 = \cos\theta_0 = \cos\theta\cos\theta' - \sin\theta\sin\theta'\cos(\varphi-\varphi') .$$

And the following coefficients are determined by using the orthogonal property of Legendre polynomials:

$$\phi_n(x,E) = \int d\mu \psi(x,\mu,E)P_n(\mu) \quad , \tag{6.6}$$

$$\Sigma_{sn}(x,E' \to E) = 2\pi \int d\mu_0 \Sigma_s(x,\mu_0,E' \to E)P_n(\mu_0) \quad , \tag{6.7}$$

$$P_n(\mu_0) = P_n(\mu)P_n(\mu') + 2\sum_{m=1}^{n}\frac{(n-m)!}{(n+m)!}P_n^m(\mu)P_n^m(\mu')\cos(m(\varphi-\varphi'))$$

where .

Therefore, Eq.(6.5) is rewritten using moments and Legendre polynomials as:

$$\mu \frac{\partial}{\partial x}\sum_{n=0}^{N}\frac{(2n+1)}{2}P_n(\mu)\phi_n(x,E) + \Sigma_t(x,E)\sum_{n=0}^{N}\frac{(2n+1)}{2}P_n(\mu)\phi_n(x,E) = \int dE' \sum_{n=0}^{N}\frac{(2n+1)}{2}\Sigma_{sn}(x,E' \to E)P_n(\mu)\phi_n(x,E') + \frac{1}{2}S_f(x,E). \tag{6.8}$$

Using the recursion relationship of the Legendre polynomials:

$$(2n+1)\mu P_n(\mu) = (n+1)P_{n+1}(\mu) + nP_{n-1}(\mu) \quad , \tag{6.9}$$

One can arrive at (N+1) sets of coupled partial differential equations for the expansion coefficients $\phi_n(x,E)$:

$$\frac{(n+1)}{(2n+1)}\frac{\partial\phi_{n+1}(x,E)}{\partial x}+\frac{n}{(2n+1)}\frac{\partial\phi_{n-1}(x,E)}{\partial x}+\Sigma_t(x,E)\phi_n(x,E)$$
$$=\int dE'\Sigma_{sn}(x,E'\to E)\phi_n(x,E')+\frac{\delta_{0n}}{2}S_f(x,E). \tag{6.10}$$

For convenience in deriving multi-group one-dimensional P_N equations, the energy ranges are divided into G discrete intervals. Eq. (6.10) can be rewritten with these discrete groups:

$$\frac{(n+1)}{(2n+1)}\frac{d\phi_{n+1g}(x)}{dx}+\frac{n}{(2n+1)}\frac{d\phi_{n-1g}(x)}{dx}+\Sigma_{tng}(x)\phi_{ng}(x)$$
$$=\sum_{g'}\Sigma_{sngg'}(x)\phi_{ng'}(x)+\frac{\delta_{0n}}{2}S_{fg}(x), \tag{6.11}$$

where $\phi_{ng}(x)=\int_g\phi_n(x,E)dE$, $\Sigma_{tng}(x)=\dfrac{\int_g\Sigma_t(x,E)\phi_n(x,E)dE}{\phi_{ng}(x)}$,

$\Sigma_{sngg'}(x)=\dfrac{\int_g dE\int_{g'}\Sigma_{sn}(x,E'\to E)\phi_n(x,E')dE'}{\phi_{ng'}(x)}$, $\nu\Sigma_{fg}(x)=\dfrac{\int_g\nu\Sigma_f(x,E)\phi(x,E)dE}{\phi_g(x)}$.

While all cross sections in the first P_N equation are flux weighted, an exact calculation of total and scattering cross sections of the higher P_N equations requires that the flux moments, $\phi_n(x,E)$ be knwon for $n>0$. Instead of introducing different total cross sections in all moment equations, Σ_{t0g}, which is weighted by the flux, is used in the left hand side of the higher moment equations. Modification of diagonal elements of scattering cross sections is then required:

$$\Sigma'_{sngg}(x)=\Sigma_{sngg}(x)-(\Sigma_{tng}(x)-\Sigma_{t0g}(x)). \tag{6.12}$$

Then, Eq. (6.11) can be written as:

$$\frac{(n+1)}{(2n+1)}\frac{d\phi_{n+1g}(x)}{dx}+\frac{n}{(2n+1)}\frac{d\phi_{n-1g}(x)}{dx}+\Sigma_{rng}(x)\phi_{ng}(x)$$
$$=\sum_{g'\neq g}\Sigma_{sngg'}(x)\phi_{ng'}(x)+\frac{\delta_{0n}}{2}S_{fg}(x). \tag{6.13}$$

where $\Sigma_{rng}(x)=\Sigma_{t0g}(x)-\Sigma'_{sngg}(x)$.

A reflective boundary condition requires that all odd moments of the flux vanish:

$$\phi_n(x_s,E)=0,\quad for\ odd\ n, \tag{6.14}$$

where x_s is x at a surface.

The exact interface condition of continuity of angular flux cannot be satisfied exactly by the flux approximation for infinite N. Therefore, the first N+1 Legendre moments of this relation are required to be satisfied:

$$\phi_n(x_s^-,E)=\phi_n(x_s^+,E),\ n=0,1,2,...,N, \tag{6.15}$$

where x_s^+ and x_s^- are x from right or left side of a surface s, respectively.

6.2 SP_N Equations

The original approach of the SP_N approximation is a simple generalization of the one-dimensional P_N equations to the multidimensional P_N equations. The result is (N+1)-coupled SP_N equations in the three-dimensional geometry. The three-dimensional P_1 equations can first be constructed from the one-dimensional P_1 equations as follows:

1) Replace the operator $\partial/\partial x$ in the one-dimensional $n=0$ equation with the divergence operator ∇,
2) Replace the operator $\partial/\partial x$ in the one-dimensional $n=1$ equation with the gradient operator ∇,
3) Consider the zeroth-order Legendre moment of the angular flux ϕ_0 as a scalar,
4) Consider the first-order Legendre moment of the angular flux ϕ_1 as a vector.

For the SP_N equations, the relations of P_1 between one-dimensional and multi-dimensional geometries are extrapolated to P_N when N > 1.

1) Replace the operator $\partial / \partial x$ in the one-dimensional equation for even n with the divergence operator ∇,
2) Replace the operator $\partial / \partial x$ in the one-dimensional equation for odd n with the gradient operator ∇,
3) Consider the even-order Legendre moments of the angular flux as scalars,
4) Consider the odd-order Legendre moments of the angular flux as vectors.

With the procedure above, the final form is obtained as:

$$\begin{aligned} &\frac{(n+1)}{(2n+1)}\nabla\cdot\phi_{n+1}+\frac{n}{(2n+1)}\nabla\cdot\phi_{n-1}+\Sigma_t\phi_n=s_n, \quad \text{n = even,} \\ &\frac{(n+1)}{(2n+1)}\nabla\phi_{n+1}+\frac{n}{(2n+1)}\nabla\phi_{n-1}+\Sigma_t\phi_n=s_n, \quad \text{n = odd.} \end{aligned} \tag{6.16}$$

The general matrix form of the SP_N equations is shown in Eq. (6.17). Their boundary conditions can also be obtained from those for the one-dimensional P_N equations. The advantages of the SP_N equations are that inspite of some approximations in their formulation, they have significantly improved computational efficiency compared to the S_N method or the original P_N method, and they maintain the rotational invariance of the P_N equations which is not present in the S_N equations.

$$\begin{bmatrix} \Sigma_{r0} & \nabla\cdot & & & & & & & \\ \frac{1}{3}\nabla & \Sigma_{t1} & \frac{2}{3}\nabla & & & & & & \\ & \frac{2}{5}\nabla\cdot & \Sigma_{t2} & \frac{3}{5}\nabla\cdot & & & & & \\ & & \frac{3}{7}\nabla & \Sigma_{t3} & \frac{4}{7}\nabla & & & & \\ & & & \frac{4}{9}\nabla\cdot & \Sigma_{t4} & \frac{5}{9}\nabla\cdot & & & \\ & & & & \frac{5}{11}\nabla & \Sigma_{t5} & \frac{6}{11}\nabla & & \\ & & & & & . & . & & \\ & & & & & & . & . & \\ & & & & & & . & . & \\ & & & & & & & \frac{N}{2N+1}\nabla & \Sigma_{tN} \end{bmatrix} \begin{bmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \\ . \\ . \\ . \\ \phi_N \end{bmatrix} = \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ . \\ . \\ . \\ s_N \end{bmatrix}. \quad (6.17)$$

Note that N is normally odd, and accordingly N-1 becomes even. The number of first-order differential equations is reduced by one-half in second-order differential equations with the assumption of $s_i = 0 \ (i \geq 1)$. For simplicity, Eq. (6.17) can be reduced to (N+1)/2 number of second-order differential equations as shown in Eq. (6.18).

It should be noted that the general SP_N equations are asymptotic corrections to the P_1 theory. For planar geometry problems, these corrections exactly reduce to the P_N equations. In practice, the SP_N equations are most accurate for problems that are reasonably close to being diffusive in nature or for problems that have strong transport regions in which the solution behaves nearly one-dimensionally and has weak tangential derivatives at material interfaces. However, for problems that have strong multidimensional transport effects, such as voids, streaming regions, or geometically complex spatial inhomogeneous, the SP_N solutions are less accurate[8].

$$\begin{bmatrix} b_0 & c_0 & & & & & & & & \\ a_2 & b_2 & c_2 & & & & & & & \\ & a_4 & b_4 & c_4 & & & & & & \\ & & \cdot & \cdot & \cdot & & & & & \\ & & & \cdot & \cdot & \cdot & & & & \\ & & & & a_n & b_n & c_n & & & \\ & & & & & \cdot & \cdot & \cdot & & \\ & & & & & & \cdot & \cdot & \cdot & \\ & & & & & & & \cdot & \cdot & \cdot \\ & & & & & & & & a_{N-1} & b_{N-1} \end{bmatrix} \begin{bmatrix} \phi_0 \\ \phi_2 \\ \phi_4 \\ \cdot \\ \cdot \\ \phi_n \\ \cdot \\ \cdot \\ \cdot \\ \phi_{N-1} \end{bmatrix} = \begin{bmatrix} s_0 \\ 0 \\ 0 \\ \cdot \\ \cdot \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{bmatrix}, \tag{6.18}$$

where $b_0 = -D_1\nabla^2 + \Sigma_r$, $c_0 = -2D_1\nabla^2$,

$a_2 = -\frac{2}{5}D_1\nabla^2$, $b_2 = -\left(\frac{4}{5}D_1 + \frac{3}{5}D_3\right)\nabla^2 + \Sigma_{t2}$, $c_2 = -\frac{4}{5}D_3\nabla^2$,

$a_n = -\frac{n}{2n-1}D_{n-1}\nabla^2$, $b_n = -\left(\frac{n^2}{(2n+1)(n-1)}D_{n-1} + \frac{n+1}{2n+1}D_{n+1}\right)\nabla^2 + \Sigma_{tn}$,

$c_n = -\frac{n+2}{2n+1}D_{n+1}\nabla^2$,

$D_m \equiv \frac{m}{(2m+1)\Sigma_{tm}}$, $m = odd\ \ number$, $n = even\ number$.

6.3 Implementation of the SP_3 Method

In PARCS, the SP_N order was truncated for N > 3. The governing equations of the multidimensional SP_3 equation can be written as:

$$\begin{aligned} \nabla\cdot\phi_{1g} + \Sigma_{rg}\phi_{0g} &= s_{0g}, \\ \frac{2}{3}\nabla\phi_{2g} + \frac{1}{3}\nabla\phi_{0g} + \Sigma_{trg}\phi_{1g} &= 0, \\ \frac{3}{5}\nabla\cdot\phi_{3g} + \frac{2}{5}\nabla\cdot\phi_{1g} + \Sigma_{tg}\phi_{2g} &= 0, \\ \frac{3}{7}\nabla\phi_{2g} + \Sigma_{tg}\phi_{3g} &= 0, \end{aligned} \tag{6.19}$$

where

$$s_{0g} = \sum_{g'} \Sigma_{sg'g} \phi_{0g'} + \frac{\chi_g}{k_{eff}} \sum_{g'} \nu\Sigma_{fg'} \phi_{0g'}$$

.

The most common methods used for a direct solution for the SP_N equations involve the elimination of odd-order angular moments to yield a set of coupled diffusion-like equations as shown below in a matrix-equation form for a given group:

$$\begin{bmatrix} -D_1\nabla^2 + \Sigma_r & -2D_1\nabla^2 \\ -\frac{2}{5}D_1\nabla^2 & -\left(\frac{3}{5}D_3 + \frac{4}{5}D_1\right)\nabla^2 + \Sigma_t \end{bmatrix} \begin{bmatrix} \phi_0 \\ \phi_2 \end{bmatrix} = S_0 \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \tag{6.20}$$

where

$$D_1 \equiv \frac{1}{3\Sigma_{tr}}, \quad D_3 \equiv \frac{3}{7\Sigma_t}.$$

These are simply two coupled diffusion equations that can readily be solved with standard diffusion computer codes with appropriately defined diffusion coefficients. For SP_3 boundary conditions, the Marshark boundary condition can be used in the same way as in P_1. For example, the equations for a right boundary become:

$$j_l^+ = \int_0^1 d\mu\, P_l(\mu)\psi(x_r,\mu), \quad j_l^- = \int_0^{-1} d\mu\, P_l(\mu)\psi(x_r,\mu), \tag{6.21}$$

where

x_r = right surface, $\psi(x,\mu) = \sum_{l=0}^{3} \frac{(2l+1)}{2} P_l(\mu)\phi_l(x)$,

$P_0(\mu) = 1$, $P_1(\mu) = \mu$, $P_2(\mu) = \frac{1}{2}(3\mu^2 - 1)$, $P_3(\mu) = \frac{1}{2}(5\mu^3 - 3\mu)$.

Applying P_1 and P_3 to Eq. (6.21) leads to the relationship between partial currents, surface fluxes and net currents:

$$j_1^{\pm} = \frac{1}{4}\phi_{0s} \pm \frac{1}{2}J_1 + \frac{5}{16}\phi_{2s}, \quad j_3^{\pm} = \frac{5}{16}\phi_{2s} \pm \frac{1}{2}J_3 - \frac{1}{16}\phi_{0s}, \tag{6.22}$$

which have the additional contribution of the second moment to the boundary conditions of the P_1 equation.

The time-dependent SP_3 equations are basically the same as Eq. (6.19) except for the additional time derivative terms in each equation:

$$\begin{aligned}
&\frac{1}{\upsilon}\frac{\partial \phi_{0g}}{\partial t}+\nabla\cdot\phi_{1g}+\Sigma_{rg}\phi_{0g}=s_{0tg}\\
&\frac{1}{\upsilon}\frac{\partial \phi_{1g}}{\partial t}+\frac{2}{3}\nabla\phi_{2g}+\frac{1}{3}\nabla\phi_{0g}+\Sigma_{trg}\phi_{1g}=0\\
&\frac{1}{\upsilon}\frac{\partial \phi_{2g}}{\partial t}+\frac{3}{5}\nabla\cdot\phi_{3g}+\frac{2}{5}\nabla\cdot\phi_{1g}+\Sigma_{tg}\phi_{2g}=0\\
&\frac{1}{\upsilon}\frac{\partial \phi_{3g}}{\partial t}+\frac{3}{7}\nabla\phi_{2g}+\Sigma_{tg}\phi_{3g}=0\\
&\frac{dC_k}{dt}=-\lambda_k C_k+\frac{\beta}{k_{eff}}\sum_{g'}\nu\Sigma_{fg'}\phi_{0g'}
\end{aligned}, \tag{6.23}$$

where

$$\begin{aligned}
&s_{0tg}=\sum_{g'}\Sigma_{sg'g}\phi_{0g'}+\frac{\chi_g}{k_{eff}}(1-\beta)\sum_{g'}\nu\Sigma_{fg'}\phi_{0g'}+\chi_{dg}\sum_{k'}\lambda_k C_k\\
&\frac{dC_k}{dt}=-\lambda_k C_k+\frac{\beta}{k_{eff}}\sum_{g'}\nu\Sigma_{fg'}\phi_{0g'}
\end{aligned}.$$

Similar to the method used to simplify the steady-state equation, the moment equations can be combined to yield the two time-dependent 2nd-order differential equations:

$$\begin{aligned}
&-\frac{1}{\upsilon\Sigma_{trg}}\frac{\partial}{\partial t}\left(\nabla\cdot\phi_{1g}\right)+\frac{1}{\upsilon}\frac{\partial \phi_{0g}}{\partial t}-\nabla\cdot\left(2D_{0g}\nabla\phi_{2g}+D_{0g}\nabla\phi_{0g}\right)+\Sigma_{rg}\phi_{0g}=s_{0tg}\\
&-\frac{3}{5}\frac{1}{\upsilon\Sigma_{tg}}\frac{\partial}{\partial t}\left(\nabla\cdot\phi_{3g}\right)-\frac{2}{5}\frac{1}{\upsilon\Sigma_{trg}}\frac{\partial}{\partial t}\left(\nabla\cdot\phi_{1g}\right)+\frac{1}{\upsilon}\frac{\partial \phi_{2g}}{\partial t}\\
&\qquad-\nabla\cdot\left(\frac{3}{5}D_2\nabla\phi_{2g}+\frac{4}{5}D_{0g}\nabla\phi_{2g}+\frac{2}{5}D_{0g}\nabla\phi_{0g}\right)+\Sigma_{tg}\phi_{2g}=0.
\end{aligned}, \tag{6.24}$$

As indicated in Eq. (6.24), time derivatives of divergences of the 1st- and 3rd-moments, which are the change of leakages of 0th- and 2nd-moments with time, are additional terms to be considered, compared to the conventional time-dependent diffusion equation.

In the PARCS code, a slightly modified form of the equations are used. The time derivative term is combined $\left(\partial/\upsilon\partial t\right)^{n+1}$ with Σ^{*}_{sg}, and the term s_{0tg} on the right hand side is also modified with the time derivative term $\left(\partial/\upsilon\partial t\right)^{n}$. A fully implicit time integration method is used, and with the time indices, Eq. (6.23) can then be written as:

$$\nabla\cdot\phi_{1g}^{n+1}+\Sigma^{*}_{rg}\phi_{0g}^{n+1}=s_{0tg}^{n+1}+q_{0g}^{n}, \tag{6.25}$$

$$\frac{2}{3}\nabla\phi_{2g}^{n+1}+\frac{1}{3}\nabla\phi_{0g}^{n+1}+\Sigma_{trg}^{*}\phi_{1g}^{n+1}=q_{1g}^{n},$$

$$\frac{3}{5}\nabla\cdot\phi_{3g}^{n+1}+\frac{2}{5}\nabla\cdot\phi_{1g}^{n+1}+\Sigma_{tg}^{*}\phi_{2g}^{n+1}=q_{2g}^{n},$$

$$\frac{3}{7}\nabla\phi_{2g}^{n+1}+\Sigma_{tg}^{*}\phi_{3g}^{n+1}=q_{3g}^{n},$$

where

$$\Sigma_{sg}^{*}=\Sigma_{sg}+\frac{1}{\upsilon\Delta t},\quad q_{ig}^{n}=\frac{1}{\upsilon}\frac{\phi_{ig}^{n}}{\Delta t},$$

$$s_{0tg}^{n+1}=\frac{1}{\upsilon_g\Delta t}\phi_{0g}^{n}+\sum_{g'}\Sigma_{sg'g}\phi_{0g'}^{n+1}+\frac{\chi_g}{k_{eff}}(1-\beta)\sum_{g'}\nu\Sigma_{fg'}\phi_{0g'}^{n+1}+\chi_{dg}\sum_{k'}\lambda_k C_k^{n+1}.$$

Eq. (6.25) can be reduced to the two second-order differential equations similar to the steady-state equations:

$$\begin{aligned}
&-\nabla\cdot\left(\frac{2}{3\Sigma_{trg}^{*}}\nabla\phi_{2g}^{n+1}+\frac{1}{3\Sigma_{trg}^{*}}\nabla\phi_{0g}^{n+1}-\frac{1}{\Sigma_{trg}^{*}}q_{1g}^{n}\right)+\Sigma_{rg}^{*}\phi_{0g}^{n+1}=s_{0tg}^{n+1}+q_{0g}^{n},\\
&-\frac{3}{5}\nabla\cdot\left(\frac{3}{7\Sigma_{tg}^{*}}\nabla\phi_{2g}^{n+1}-\frac{1}{\Sigma_{tg}^{*}}q_{3g}^{n}\right)-\frac{2}{5}\nabla\cdot\left(\frac{2}{3\Sigma_{trg}^{*}}\nabla\phi_{2g}^{n+1}+\frac{1}{3\Sigma_{trg}^{*}}\nabla\phi_{0g}^{n+1}-\frac{1}{\Sigma_{trg}^{*}}q_{1g}^{n}\right)\\
&\qquad\qquad+\Sigma_{tg}^{*}\phi_{0g}^{n+1}=q_{2g}^{n}.
\end{aligned}\qquad(6.26)$$

The equations above can be expressed in a simple matrix equation form without a group index for comparison with the steady-state form Eq. (6.20):

$$\begin{bmatrix}-D_1^{*}\nabla^2+\Sigma_r^{*} & -2D_1^{*}\nabla^2\\ -\frac{2}{5}D_1^{*}\nabla^2 & -\left(\frac{3}{5}D_3^{*}+\frac{4}{5}D_1^{*}\right)\nabla^2+\Sigma_t^{*}\end{bmatrix}\begin{bmatrix}\phi_0^{n+1}\\ \phi_2^{n+1}\end{bmatrix}=\begin{bmatrix}q_0^{n}-3D_1^{*}\nabla\cdot q_1^{n}+s_{0t}^{n+1}\\ q_2^{n}-\frac{6}{5}D_1^{*}\nabla\cdot q_1^{n}-\frac{7}{5}D_3^{*}\nabla\cdot q_3^{n}\end{bmatrix},\qquad(6.27)$$

where $D_1^{*}\equiv\frac{1}{3\Sigma_{tr}^{*}}$, $D_3^{*}\equiv\frac{3}{7\Sigma_t^{*}}$, $\Sigma_\alpha^{*}=\Sigma_\alpha+\frac{1}{\upsilon\Delta t}$, $q_i^{n}=\frac{1}{\upsilon}\frac{\phi_i^{n}}{\Delta t}$, $n=$ time index.

The terms, $\nabla\cdot q_{ig}^{n}$ ($i=1,3$), are additionally considered compared to the time-dependent diffusion equation, which correspond to the terms, $\partial\left(\nabla\cdot\phi_{ig}\right)/\partial t$ ($i=1,3$), of Eq. (6.23). Since the terms $\nabla\cdot q_{ig}^{n}$ are the change in leakages with time, they can be neglected since their contribution is relatively small compared to the change of the 0th- and 2nd-moments with time.

6.4 Multigroup Nodal Expansion Method For SP_3

The SP_3 equations have been implemented in PARCS using well established fine-mesh finite difference (FMFD) methods. This method was convenient for treating heterogeneous conditions within the fuel assembly (FA). However, when the heterogeous effect is not dominant, the FMFD solution can be inefficient in terms of accuracy and time. Therefore the SP_3 equations were also implemented in PARCS using more computationally efficient adavanced nodal method. The multigroup nodal method for SP_3 is discussed in this section.

The nodal method has been widely used and has become the standard for core neutronics calculation since the early 1980s. There are several types of nodal methods with different basis functions: nodal expansion method (NEM), analytic nodal method (ANM), analytic function exansion nodal method (AFEN), nodal green function method (NGFM), etc. Among them, the NEM provides the most straightforward and stable numerical scheme, however is somewhat less less accurate than the others under some conditions. In this work, the NEM is utilized for the multigroup SP_3 equation since it is easy to implement for mutigroup applications and simple to apply to more than one flux moment.

In order to apply the nodal method to the SP_3 equation, Eq. (6.20) is rearranged to:

$$\begin{bmatrix} -D_1\nabla^2+\Sigma_r & -2\Sigma_r \\ -\frac{2}{3}\Sigma_r & -D_3\nabla^2+\Sigma_{rt} \end{bmatrix} \begin{bmatrix} \Phi_0 \\ \phi_2 \end{bmatrix} = S_0 \begin{bmatrix} 1 \\ -\frac{2}{3} \end{bmatrix}, \tag{6.28}$$

where

$$\Phi_0 \equiv \phi_0 + 2\phi_2, \quad \Sigma_{rt} \equiv \frac{5}{3}\Sigma_t + \frac{4}{3}\Sigma_r, \quad J_1 = -D_1\nabla\Phi_0, \quad J_3 = -D_3\nabla\phi_2 .$$

For numerical stability and computational efficiency, Eq.(6.28) is more convenient for applying the NEM compared to Eq. (6.20). The Marshak boundary conditions of Eq. 6.22) also change corresponding to the moment definitions in Eq. (6.28):

$$j_1^{\pm} = \frac{1}{4}\Phi_{0s} \pm \frac{1}{2}J_1 - \frac{3}{16}\phi_{2s}, \quad j_3^{\pm} = \frac{7}{16}\phi_{2s} \pm \frac{1}{2}J_3 - \frac{1}{16}\Phi_{0s} . \tag{6.29}$$

Based on the boundary conditions above and the following relationship between net and partial moments:

$$J_1 = -D_1\nabla\Phi_0 = j_1^+ - j_1^-, \quad J_3 = -D_3\nabla\phi_2 = j_3^+ - j_3^- . \tag{6.30}$$

Eqs. (6.29) and (6.30) require that partial currents and surface fluxes at interfaces satisfy the following relations:

$$\Phi_{0s} = \frac{8}{25}\left[7\left(j_1^+ + j_1^-\right) + 3\left(j_3^+ + j_3^-\right)\right], \quad \phi_{2s} = \frac{8}{25}\left[\left(j_1^+ + j_1^-\right) + 4\left(j_3^+ + j_3^-\right)\right]. \tag{6.31}$$

The one-dimensional equation integrated over transverse directions within a given group can be expressed as:

$$\begin{bmatrix} -D_{1u}\frac{d^2}{du^2} + \Sigma_r & -2\Sigma_r \\ -\frac{2}{3}\Sigma_r & -D_{3u}\frac{d^2}{du^2} + \Sigma_{rt} \end{bmatrix} \begin{bmatrix} \Phi_{0u} \\ \phi_{2u} \end{bmatrix} = S_0 \begin{bmatrix} 1 \\ -\frac{2}{3} \end{bmatrix} - \begin{bmatrix} L_{1u} \\ L_{3u} \end{bmatrix}, \quad (u = x, y, z), \tag{6.32}$$

where L_{iu} is the transverse leakage to the u-direction in i moment. As in the conventional NEM, the one-dimensional moments can be approximated in 4th-order polynomials:

$$\Phi_{0u}(u) = \sum_{i=0}^{4} a_i h_i(u), \quad \phi_{2u}(u) = \sum_{i=0}^{4} z_i h_i(u), \quad (u = x, y, z), \tag{6.33}$$

where $h_0(u) = 1$, $h_1(u) = 2u - 1$, $h_2(u) = 6u(1-u) - 1$,

$h_3(u) = 6u(1-u)(2u-1)$, $h_3(u) = 6u(1-u)(5u^2 - 5u + 1)$, $0 < u < 1$.

Combining Eqs. (6.29), (6.30), (6.31) and (6.32) yields the partial moment response equation as follows:

$$\begin{aligned} J_1^{out} &= c_{\bar{\Phi}_0}\bar{\Phi}_0 + c_J J_1^{in} + c_A A + c_{\phi_{2s}}\phi_{2s}, \\ J_3^{out} &= g_{\bar{\phi}_2}\bar{\phi}_2 + g_J J_3^{in} + g_A A + g_{\Phi_{0s}}\Phi_{0s}, \end{aligned} \tag{6.34}$$

where $j_{1l}^{out} = c_1\bar{\Phi}_0 + c_2 j_{1l}^{in} + c_3 j_{1r}^{in} + c_1 a_4 - c_4 a_3 - c_5\phi_{2l} - c_6\phi_{2r}$,

$j_{1r}^{out} = c_1\bar{\Phi}_0 + c_3 j_{1l}^{in} + c_2 j_{1r}^{in} + c_1 a_4 + c_4 a_3 - c_6\phi_{2l} - c_5\phi_{2r}$,

$j_{3l}^{out} = g_1\bar{\phi}_2 + g_2 j_{3l}^{in} + g_3 j_{3r}^{in} + g_1 a_4 - g_4 a_3 - g_5\Phi_{0l} - g_6\Phi_{0r}$,

$j_{3r}^{out} = g_1\bar{\phi}_2 + g_3 j_{3l}^{in} + g_2 j_{3r}^{in} + g_1 a_4 + g_4 a_3 - g_6\Phi_{0l} - g_5\Phi_{0r}$,

$a_1 = \frac{1}{2}\left(\Phi_{0r} - \Phi_{0l}\right) = j_{1r}^{out} + j_{1r}^{in} - j_{1l}^{out} - j_{1l}^{in} + \frac{3}{8}\left(\phi_{2r} - \phi_{2l}\right)$,

$a_2 = \bar{\Phi}_0 - \frac{1}{2}\left(\Phi_{0r} + \Phi_{0l}\right) = \bar{\phi} - \left(j_{1r}^{out} + j_{1r}^{in} + j_{1l}^{out} + j_{1l}^{in} + \frac{3}{8}\left(\phi_{2r} + \phi_{2l}\right)\right)$,

$$z_1 = \frac{1}{2}\left(\phi_{2r} - \phi_{2l}\right) = \frac{4}{7}\left(j_{3r}^{out} + j_{3r}^{in} - j_{3l}^{out} - j_{3l}^{in}\right) + \left(\Phi_{0r} - \Phi_{0l}\right),$$

$$z_2 = \bar{\phi}_2 - \frac{1}{2}\left(\phi_{2r} + \phi_{2l}\right) = \bar{\phi}_2 - \left(\frac{4}{7}\left(j_{3r}^{out} + j_{3r}^{in} + j_{3l}^{out} + j_{3l}^{in}\right) + \left(\Phi_{0r} + \Phi_{0l}\right)\right),$$

$$c_1 = \frac{6D_1(1+4D_1)}{1+16D_1+48D_1^2}, \quad c_2 = \frac{1-48D_1^2}{1+16D_1+48D_1^2}, \quad c_3 = \frac{-8D_1}{1+16D_1+48D_1^2},$$

$$c_4 = \frac{6D_1(1+12D_1)}{1+16D_1+48D_1^2}, \quad c_5 = \frac{3D_1(1+6D_1)}{1+16D_1+48D_1^2}, \quad c_6 = \frac{3D_1}{2(1+16D_1+48D_1^2)},$$

$$g_1 = \frac{42D_3}{7+48D_3}, \quad g_2 = \frac{49-768D_1^2}{49+448D_1+768D_1^2}, \quad g_3 = \frac{224D_3}{49+448D_1+768D_1^2},$$

$$g_4 = \frac{42D_3}{7+16D_3}, \quad g_5 = \frac{4D_3(7+24D_1)}{49+448D_1+768D_1^2}, \quad g_6 = \frac{14D_3}{49+448D_1+768D_1^2}.$$

Since the surface moments appear on the right hand sides of the partial moment response equations in Eq. (6.34), it is not easy to formulate and solve them. For simplicity, the following relationships were developed here:

$$\tilde{j}_1^{\pm} = \frac{1}{4}\Phi_{0s} \pm \frac{1}{2}J_1, \quad \tilde{j}_3^{\pm} = \frac{7}{16}\phi_{2s} \pm \frac{1}{2}J_3. \tag{6.35}$$

These are truncated forms for the Marshak boundary conditions and therefore are not conventional partial moments but quantities satisfying the following conditions.

$$J_1 = -D_1\nabla\Phi_0 = \tilde{j}_1^{+} - \tilde{j}_1^{-}, \quad J_3 = -D_3\nabla\phi_2 = \tilde{j}_3^{+} - \tilde{j}_3^{-}, \tag{6.36}$$

and

$$\Phi_{0s} = 2\left(\tilde{j}_1^{+} + \tilde{j}_1^{-}\right), \quad \phi_{2s} = \frac{8}{7}\left(\tilde{j}_3^{+} + \tilde{j}_3^{-}\right). \tag{6.37}$$

Eq. (6.37) is similar to Eq. (6.31) and somewhat simpler to apply. With the variables defined above, reflective boundary conditions and incoming current boundary conditions need to be discussed. Reflective boundary conditions are the same as usual:

$$\tilde{j}_1^{+} = \tilde{j}_1^{-}, \quad \tilde{j}_3^{+} = \tilde{j}_3^{-}. \tag{6.38}$$

However, incoming current boundary conditions become different compared to the original ones

since $\tilde{j}_1^{\pm}$ and $\tilde{j}_3^{\pm}$ are not real partial moments. Inserting Eqs. (6.36) and (6.37), using Eqs. (6.29) and (6.35) and $j_1^{in} = j_3^{in} = 0$ yields:

$$\tilde{j}_1^{+} = \frac{3}{109}\left(\tilde{j}_1^{-} + 8\tilde{j}_3^{-}\right), \quad \tilde{j}_3^{+} = \frac{3}{109}\left(14\tilde{j}_1^{-} + 3\tilde{j}_3^{-}\right). \tag{6.39}$$

The relations between artificial partial currents, net currents, and surface fluxes thus have the same form as in the NEM formulation of the diffusion equation with some additional consideration for the incoming boundary conditions as shown in Eq. (6.39). The final form of the partial moment response equation using $\tilde{j}^{\pm}$ then becomes the same as the conventional NEM formulation:

$$\begin{aligned} \tilde{J}_1^{out} &= c_{\bar{\Phi}_0}\bar{\Phi}_0 + c_J\tilde{J}_1^{in} + c_A A, \\ \tilde{J}_3^{out} &= g_{\bar{\phi}_2}\bar{\phi}_2 + g_J\tilde{J}_3^{in} + g_A A, \end{aligned} \tag{6.40}$$

where

$$\begin{aligned} \tilde{j}_{1l}^{out} &= c_1\bar{\Phi}_0 + c_2\tilde{j}_{1l}^{in} + c_3\tilde{j}_{1r}^{in} + c_1a_4 - c_4a_3, \\ \tilde{j}_{1r}^{out} &= c_1\bar{\Phi}_0 + c_3\tilde{j}_{1l}^{in} + c_2\tilde{j}_{1r}^{in} + c_1a_4 + c_4a_3, \\ \tilde{j}_{3l}^{out} &= g_1\bar{\phi}_2 + g_2\tilde{j}_{3l}^{in} + g_3\tilde{j}_{3r}^{in} + g_1a_4 - g_4a_3, \\ \tilde{j}_{3r}^{out} &= g_1\bar{\phi}_2 + g_3\tilde{j}_{3l}^{in} + g_2\tilde{j}_{3r}^{in} + g_1a_4 + g_4a_3, \\ a_1 &= \frac{1}{2}\left(\Phi_{0r} - \Phi_{0l}\right) = \tilde{j}_{1r}^{out} + \tilde{j}_{1r}^{in} - \tilde{j}_{1l}^{out} - \tilde{j}_{1l}^{in}, \\ a_2 &= \bar{\Phi}_0 - \frac{1}{2}\left(\Phi_{0r} + \Phi_{0l}\right) = \bar{\Phi}_0 - \left(\tilde{j}_{1r}^{out} + \tilde{j}_{1r}^{in} + \tilde{j}_{1l}^{out} + \tilde{j}_{1l}^{in}\right), \\ z_1 &= \frac{1}{2}\left(\phi_{2r} - \phi_{2l}\right) = \frac{4}{7}\left(\tilde{j}_{3r}^{out} + \tilde{j}_{3r}^{in} - \tilde{j}_{3l}^{out} - \tilde{j}_{3l}^{in}\right), \\ z_2 &= \bar{\phi}_2 - \frac{1}{2}\left(\phi_{2r} + \phi_{2l}\right) = \bar{\phi}_2 - \left(\frac{4}{7}\left(\tilde{j}_{3r}^{out} + \tilde{j}_{3r}^{in} + \tilde{j}_{3l}^{out} + \tilde{j}_{3l}^{in}\right)\right). \end{aligned}$$

The definitions of the coefficients c_i and g_i are the same as in Eq. (6.34). However, the definitions of the low-order coefficients, a_1 and a_2, are changed in Eq. (6.40), which do not include surface moments on the right hand side.

One of the most important considerations in the transverse averaged one-dimensional approach is the approximation of the transverse leakage shapes. In PARCS, the transverse leakage shape is approximated in a second-order polynomial for both 0th and 2nd moments as:

$$L_{0u}(u)=\sum_{i=0}^{2}b_i h_i(u), \quad L_{2u}(u)=\sum_{i=0}^{2}p_i h_i(u), \tag{6.41}$$

where b_i and p_i are expansion coefficients of the transverse leakage.

The second-order polynomial approximation for the transverse leakage of the 0th moment has been well established for applying the NEM to the diffusion equation. However, a quadratic approximation is not adequate for the 2nd moment since the 2nd moment shape is dramatically changing near interfaces of very different materials. However, a parabolic approximation, which is a rough approximation for the 2nd moment, is used since the effect of the 2nd moments is relatively small in magnitude compared to the 0th moments and refining more meshes relieves dramatic change of the 2nd moment.

The higher-order coefficients, a_3 and a_4, are determined by solving two more moment equations with the weighted residual method for Eq. (6.32) using $h_1(u)$ and $h_2(u)$, which are shown in Eq. (6.33), as weighting functions. The resulting equations then become:

$$\begin{aligned}
(60\frac{D_u}{\Delta_u}+\Sigma_r)a_3 &= -\frac{5}{3}\Sigma_r a_1 + S_{03} - \frac{5}{3}b_1 + 2\Sigma_r\left(\frac{5}{3}z_1+z_3\right),\\
(140\frac{D_u}{\Delta_u}+\Sigma_r)a_4 &= \frac{7}{3}\Sigma_r a_2 + S_{04} + \frac{7}{3}b_2 + 2\Sigma_r\left(-\frac{7}{3}z_2+z_4\right),\\
(60\frac{D_u}{\Delta_u}+\Sigma_{rt})z_3 &= -\frac{5}{3}\Sigma_{rt} z_1 + S_{23} - \frac{5}{3}p_1 + 2\Sigma_r\left(\frac{5}{3}a_1+a_3\right),\\
(140\frac{D_u}{\Delta_u}+\Sigma_{rt})z_4 &= \frac{7}{3}\Sigma_{rt} z_3 + S_{24} + \frac{7}{3}p_2 + 2\Sigma_r\left(-\frac{7}{3}a_2+a_4\right),
\end{aligned} \tag{6.42}$$

where $S_{03}=\sum_{g'}\left(\left(\frac{5}{3}a_1+a_3-2\left(\frac{5}{3}z_1+z_3\right)\right)\Sigma_{sg'g}+\frac{\chi_g}{k_{eff}}\sum_{g'}\left(\frac{5}{3}a_1+a_3-2\left(\frac{5}{3}z_1+z_3\right)\right)\nu\Sigma_{fg'}\right)$,

$S_{04}=\sum_{g'}\left(\left(-\frac{7}{3}a_2+a_4+2\left(\frac{7}{3}z_2-z_4\right)\right)\Sigma_{sg'g}+\frac{\chi_g}{k_{eff}}\sum_{g'}\left(-\frac{7}{3}a_2+a_4+2\left(\frac{7}{3}z_2-z_4\right)\right)\nu\Sigma_{fg'}\right)$,

$S_{23}=-\frac{2}{3}S_{03}$, $S_{24}=-\frac{2}{3}S_{04}$.

By defining new variables, $\widetilde{j}_1^{\pm}$ and $\widetilde{j}_3^{\pm}$ in Eq. (6.35), all equations became simple and very similar to the conventional NEM equations for the diffusion equation. This means that the existing routines can be utilized for solving the multigroup SP_3 NEM with only minor modifications.

Eqs. (6.43) and (6.44) show moment matrix equations in which 2-D or 3-D one-dimensional equations are solved at the same time, satisfying the nodal neutron balance in a node. The 0^{th} and 2^{nd} moments are not solved simultaneously, but determined sequentially for computational efficiency. In other words, a moment is used as a source of the other moment determined in the previous iteration step. As seen in Eq. (6.43) and (6.44), there are in fact 13 unknowns for a three-dimensional problem, which include currents and higher order coefficients of the nodal expansion polynomials of each direction and node-average moments. The elements of the matrix of the left hand side of the equation are only property dependent, but those of the right hand side are composed of the fission source, partial moments which are boundary conditions, low-order coefficients of the nodal expansion polynomials of the other moments and transverse leakage coefficients.

$$\begin{bmatrix} 1 & 0 & c_4 & -c_1 & -c_1 \\ 0 & 1 & -c_4 & -c_1 & -c_1 \\ -\frac{5}{3}\Sigma_r & \frac{5}{3}\Sigma_r & \left(60\frac{D_{1u}}{\Delta_u}+\Sigma_r\right) & 0 & 0 \\ \frac{7}{3}\Sigma_r & \frac{7}{3}\Sigma_r & 0 & \left(140\frac{D_{1u}}{\Delta_u}+\Sigma_r\right) & -\frac{7}{3}\Sigma_r \\ \frac{1}{\Delta_u} & \frac{1}{\Delta_u} & 0 & 0 & \Sigma_r \end{bmatrix} \begin{bmatrix} \widetilde{j}_{1l}^{out} \\ \widetilde{j}_{1r}^{out} \\ a_3 \\ a_4 \\ \overline{\Phi}_0 \end{bmatrix} = \begin{bmatrix} c_2\widetilde{j}_{1l}^{in}+c_3\widetilde{j}_{1r}^{in} \\ c_3\widetilde{j}_{1l}^{in}+c_2\widetilde{j}_{1r}^{in} \\ -\frac{5}{3}\Sigma_r\left(\widetilde{j}_{1r}^{in}-\widetilde{j}_{1l}^{in}\right)+S_{03}-\frac{5}{3}D_1b_1+2\Sigma_r\left(\frac{5}{3}z_1+z_3\right) \\ -\frac{7}{3}\Sigma_r\left(\widetilde{j}_{1r}^{in}+\widetilde{j}_{1l}^{in}\right)+S_{04}+\frac{7}{3}D_1b_2+2\Sigma_r\left(-\frac{7}{3}z_2+z_4\right) \\ \left(\widetilde{j}_{1l}^{in}+\widetilde{j}_{1r}^{in}\right)/\Delta_u+S_0+2\Sigma_r\overline{\phi}_2 \end{bmatrix}, \tag{6.43}$$

$$\begin{bmatrix} 1 & 0 & z_4 & -z_1 & -z_1 \\ 0 & 1 & -z_4 & -z_1 & -z_1 \\ -\frac{5}{3}\Sigma_{rt} & \frac{5}{3}\Sigma_{rt} & \left(60\frac{D_{3u}}{\Delta_u}+\Sigma_{rt}\right) & 0 & 0 \\ \frac{7}{3}\Sigma_{rt} & \frac{7}{3}\Sigma_{rt} & 0 & \left(140\frac{D_{3u}}{\Delta_u}+\Sigma_{rt}\right) & -\frac{7}{3}\Sigma_{rt} \\ \frac{1}{\Delta_u} & \frac{1}{\Delta_u} & 0 & 0 & \Sigma_{rt} \end{bmatrix} \begin{bmatrix} \widetilde{j}_{3l}^{out} \\ \widetilde{j}_{3r}^{out} \\ z_3 \\ z_4 \\ \overline{\phi}_2 \end{bmatrix} =$$

$$\begin{bmatrix} z_2\widetilde{j}_{3l}^{in}+z_3\widetilde{j}_{3r}^{in} \\ z_3\widetilde{j}_{3l}^{in}+z_2\widetilde{j}_{3r}^{in} \\ -\frac{20}{21}\Sigma_{rt}\left(\widetilde{j}_{3r}^{in}-\widetilde{j}_{3l}^{in}\right)+S_{23}-\frac{5}{3}D_3p_1+\frac{2}{3}\Sigma_r\left(\frac{5}{3}a_1+a_3\right) \\ -\frac{4}{3}\Sigma_{rt}\left(\widetilde{j}_{3r}^{in}+\widetilde{j}_{3l}^{in}\right)+S_{24}+\frac{7}{3}D_3p_2+\frac{2}{3}\Sigma_r\left(-\frac{7}{3}a_2+a_4\right) \\ \left(\widetilde{j}_{3l}^{in}+\widetilde{j}_{3r}^{in}\right)/\Delta_u+\frac{2}{3}\left(\Sigma_r\overline{\Phi}_0-S_0\right) \end{bmatrix}. \tag{6.44}$$

Note that these equations should consist of 13×13 matrices and 13×1 vectors for the 3-D problem by repeating components corresponding to $\left(\widetilde{j}_{1l}^{out},\widetilde{j}_{1r}^{out},a_3,a_4\right)$ and $\left(\widetilde{j}_{3l}^{out},\widetilde{j}_{3r}^{out},z_3,z_4\right)$ for the other directions, respectively.

The time dependent SP3 Equations can be rearranged as below for multigroup nodal expansion method :

$$\begin{bmatrix} -D_1\nabla^2+\Sigma_r^* & -2\Sigma_r^* \\ -\frac{2}{3}\Sigma_r^* & -D_3\nabla^2+\Sigma_{rt}^{**} \end{bmatrix} \begin{bmatrix} \Phi_0^{n+1} \\ \phi_2^{n+1} \end{bmatrix} = \left(p_0^n+s_{0t}^{n+1}\right)\begin{bmatrix} 1 \\ -\frac{2}{3} \end{bmatrix}+p_2^n\begin{bmatrix} 0 \\ \frac{5}{3} \end{bmatrix} \tag{6.45}$$

where

$$\Phi_0\equiv\phi_0+2\phi_2,\quad \Sigma_{rt}^{**}\equiv\frac{5}{3}\Sigma_t^*+\frac{4}{3}\Sigma_r^*,\quad J_1=-D_1\nabla\Phi_0,\quad J_3=-D_3\nabla\phi_2,$$

$$s_{0tg}=\sum_{g'}\Sigma_{sg'g}\phi_{0g'}+\frac{\chi_g}{k_{eff}}(1-\beta)\sum_{g'}\nu\Sigma_{fg'}\phi_{0g'}+\chi_{dg}\sum_{k'}\lambda_kC_k,$$

$$D_1\equiv\frac{1}{3\Sigma_{tr}},\quad D_3\equiv\frac{3}{7\Sigma_t},\quad \Sigma_\alpha^*=\Sigma_\alpha+\frac{1}{\theta\upsilon\Delta t},\quad p_{ig}^n=q_{ig}^n+\overline{\Theta}r_{ig}^n,\quad q_{ig}^n=\frac{\phi_{ig}^n}{\theta\upsilon\Delta t},$$

n=time index,

$$r_{0g}^{n} = \left(s_{0tg} - \nabla \cdot \phi_{1g} - \Sigma_{rg}\phi_{0g}\right)^{n},$$

$$r_{2g}^{n} = \left(-\frac{3}{5}\nabla \cdot \phi_{3g} - \frac{2}{5}\nabla \cdot \phi_{1g} - \Sigma_{tg}\phi_{2g} \right)^{n}$$

$$\phi_1^{n+1} = -\frac{1}{3\Sigma_{tr}}\nabla\Phi_0^{n+1} \quad \text{and} \quad \phi_3^{n+1} = -\frac{3}{7\Sigma_t}\nabla\phi_2^{n+1}$$

Eq. (6.45) is consistent with the steady state equation given in Eq. (6.28) and therefore it can be solved using the steady state solver with minor modifications in the balance equations; e.g., transient fixed source terms.

7. HEXAGONAL NODAL METHODS

In order to treat reactors with hexagonal assemblies (e.g. VVER) a hexagonal nodal method was implemented in PARCS using the Triangle-based Polynomial Expansion Nodal (TPEN) method.[18] The theoretical basis for the method will be presented in this chapter, together with a CMFD acceleration scheme which utilizes an innovative dynamic condensation method.

The TPEN method solves two transverse-integrated neutron diffusion equations for a hex-octahedron node. One is the radial equation defined for a hexagon and the other is the axial equation defined for the z-direction. The radial problem is solved by splitting the hexagon into six triangles and then by employing a polynomial expansion of flux within each triangle. Given the incoming current boundary conditions, the solutions of the two separate transverse-integrated equations yield two sets of node average group fluxes rather than one. This is not a problem in the two-group problem since only the outgoing currents need to be taken from the one-node solution. In the multigroup problem, however, a unique set of group fluxes is desired because the node average group fluxes determine the nodal spectra to be used in the group constant generation for the two-group CMFD problem. In this regard, a simultaneous solution of the two transverse equations is derived in this section. For an efficient solution of general multigroup problems, the fission source terms are moved to the right hand side to avoid the direct inversion of the group block. The following section first presents the TPEN equations and then the solution method used for the TPEN equations.

7.1 Multi-group TPEN Method

The transverse-integrated neutron diffusion equations defined for the hex-octahedron domain shown in Figure 7.1 consist of six radial and one axial equations. With index *m* denoting the *m*-th triangle, the balance equations can be written as:

$$-D_g\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)\phi_g^{Rm}(x,y)+\Sigma_{rg}\phi_g^{Rm}(x,y)-\sum_{g'<g}\Sigma_{g'g}\phi_{g'}^{Rm}(x,y)$$
$$=Q_g^{Rm}(x,y)-L_g^{Zm}(x,y), \quad m=1..6$$

$$-D_g\frac{\partial^2}{\partial z^2}\phi_g^{Z}(z)+\Sigma_{rg}\phi_g^{Z}(z)-\sum_{g'<g}\Sigma_{g'g}\phi_{g'}^{Z}(z)=Q_g^{Z}(z)-L_g^{R}(z)$$

$$Q_g^{U}(u)=\frac{\chi_g}{k_{eff}}\sum_{g'}\nu\Sigma_{fg'}\phi_{g'}^{R,old}(u)+\sum_{g'>g}\Sigma_{g'g}\phi_{g'}^{R,old}(u)+S_g^{TFS}(u), \quad u=(x,y)\text{ or }z,$$

$$L_g^{Zm}(x,y)=\frac{1}{h_Z}\left(J_{gT}^{Zm}(x,y)-J_{gB}^{Zm}(x,y)\right), \text{ and } \quad L_g^{R}(z)=\frac{2\sqrt{3}}{9h_R}\sum_{u=1}^{3}\left(J_{g(u+3)}^{x}(z)-J_{gu}^{x}(z)\right). \tag{7.1}$$

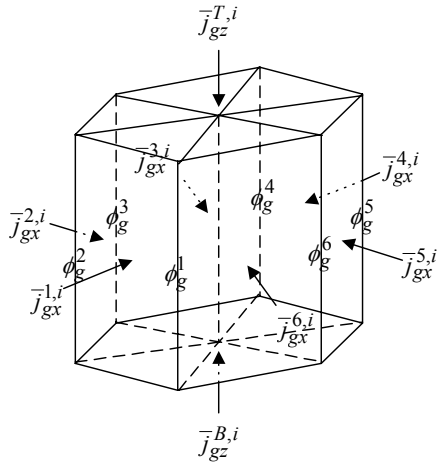
Here the source term Q contains the fission and upscattering sources, and in addition, the transient fixed source (TFS) appearing in the transient formulation. The source distribution is assumed to be known from the previous iterate of the flux distribution. The transverse-leakage sources represented in Equation (7.1) have spatial dependence which is determined in the TPEN approach by the node average transverse leakages of the neighboring nodes as well as the node of interest itself. In the original TPEN formulation, all the transverse leakages are considered known and thus the two balance equations are independent and can be solved separately. In this derivation, however, the transverse-leakage of the node of interest is treated unknown so that the radial and axial equations are now coupled through the transverse leakages.

The radial equation in Eq. (7.1) above is solved by representing the intranodal flux within the triangle by a nine-term, two-dimensional polynomial of the following form:

$$\phi_g^{Rm}(x,y) = c_{g0}^m + a_{gx}^m x + a_{gy}^m y + b_{gx}^m x^2 + b_{gu}^m u^2 + b_{gp}^m p^2 + c_{gx}^m x^3 + c_{gu}^m u^3 + c_{gp}^m p^3 \tag{7.2}$$

where

$$u = -\frac{1}{2}x - \frac{\sqrt{3}}{2}y \quad \text{and} \quad p = -\frac{1}{2}x + \frac{\sqrt{3}}{2}y \ .$$



* *A local coordinate is set for each triangle and the outward normal direction at each hexagon surface is taken as the positive x-direction of the local coordinate.*

Figure 7.1: Geometry and Boundary Conditions and Coordinates for TPEN

The nine coefficients of the third-order polynomial are related with the following nine unknowns: 3 surface average fluxes, 3 corner point fluxes, node average flux, first order *x*-and *y*-moments of the flux. To determine the nine unknowns, nine constraint conditions are imposed: 3 surface average current continuity conditions, 3 corner point leakage balance (CPB) conditions,

nodal balance condition, first-order *x*-moment balance condition, and first order *y*-moment balance condition. On the other hand, the axial equation is solved by the nodal expansion method (NEM) that involves five unknowns: first- and second-order *z*-moments as well as the two surface average fluxes and node average flux.

Because the TPEN method requires two directional first-order moments, the corresponding moments of the source terms appearing in Eq. (7.1) need to be specified. These moments carry the spatial dependence of the source. In the TPEN, the source shape is approximated by a seven-term, two-dimensional polynomial and the seven coefficients are determined by preserving the node average values of the source at the six surrounding nodes and the node of interest. For the one hexagon node problem, there are six sets of unknowns, one for each triangle. In order to determine uniquely all the coefficients of the six sets, six hexagon-corner point fluxes as well as six surface average incoming currents need to be specified per group. The hexagon-corner point fluxes are determined before solving the one-node problems by imposing the CPB condition to each hexagon-corner. In order to obtain the corner point leakage, the intranodal flux distribution within the *hexagon* is approximated by a thirteen-term polynomial whose coefficients are expressed with 6 surface average fluxes, 6 corner point fluxes, and the node average flux of the hexagon. During the CPB calculation, the node average and surface average fluxes are assumed to be known, and only the corner point fluxes are taken as the unknowns so that the CPB conditions yield a linear system for corner point fluxes at each plane. The linear system is solved using the Gauss-Seidel method.

Among various constraint conditions used in TPEN, only the nodal balance and moment equations contain terms originating from the RHS terms of Eqs. (7.1). Hence the node average and directional moments of the sources including the transverse-leakages should be obtained. For the solution of the radial equation, the axial leakage's node average value (0-th order moment) and the *x*- and *y*-directional first order moments are required. Each of these moments (including the 0-th order moment for the node average) consists of two components: one from the contribution of axial leakage of the hexagon of interest node and the other from the six neighboring hexagons. Namely,

$$\overline{L}_g^{Zm} = \overline{L}_{g,neig}^{Zm} + \frac{1}{h_Z} \sum_{s=T,B} \left(\overline{J}_{gs}^{Zo} - \overline{J}_{gs}^{Zi} \right) \quad \text{and} \tag{7.3}$$

$$\widetilde{L}_{gu}^{Zm} = \widetilde{L}_{gu,neig}^{Zm} + \frac{\mu_{gu}^{Rm}}{h_Z} \sum_{s=T,B} \left(\overline{J}_{gs}^{Zo} - \overline{J}_{gs}^{Zi} \right), \quad u=x,y \tag{7.4}$$

where the partial currents replace the net current appearing in Eq. (7.1). Note that $\mu_{gy,neig}^{Zm} = 0$. The contribution from the neighboring hexagons induces a gradient of the axial leakage within the hexagon, and thus the contribution from the neighbors must be different for each triangle so that the first term above retains the triangle index *m*. The contribution of the hexagon of interest, however, is constant over the hexagon and thus no dependence on the triangle location appears in the second term. Since the incoming currents are given as the boundary conditions, the actual

unknown in the axial leakage moment given by Eqs. (7.3) and (7.4) is the surface average outgoing partial current at the top and bottom of the hex-octahedron node. These unknown terms are moved to the left hand side of the nodal balance and the moment equations.

The moments of the radial leakage can be similarly represented for the axial equation. Since moments are required up to the second-order in the NEM used for the axial solution, three moments are defined as follows:

$$\bar{L}_g^R = \frac{2\sqrt{3}}{9h_R}\sum_{m=1}^{6}\left(\bar{J}_{gm}^{xo} - \bar{J}_{gm}^{xi}\right) \quad \text{and} \tag{7.5}$$

$$\tilde{L}_{gq}^R = \tilde{L}_{gq,neig}^R + \mu_q^Z \bar{L}_g^R \ , \ q=1,2. \tag{7.6}$$

With the representations of the radial leakage moments given by Eqs. (7.5) and (7.6), the hexagon surface outgoing partial currents appear in the axial nodal balance equation, 1-st and 2-nd order axial moment equations. These radial partial current terms are moved to the left hand side because they are unknowns as well.

7.2 Solution of the TPEN Equations

For the hex-octahedron shown in Figure 7.1, 36 unknowns are defined per group for the coupled TPEN. They are 6 hexagon surface outgoing partial currents, 6 inner surface average fluxes, 6 triangle average fluxes, 6 *x*-moments, 6 *y*-moments, 1 center point flux, 2 axial partial currents (top and bottom), 1 first-order *z*-moment, and 1 second-order *z*-moment, and finally 1 hexagon average flux. The constraint conditions required to determine these unknowns are as follows: 6 triangle nodal balance conditions, 6 triangle *x*-moment equations, 6 triangle *y*-moment equations, 6 inner surface current continuity conditions, 6 radial incoming current boundary conditions, 1 center point CPB condition, 2 axial incoming current boundary conditions, 1 first-order *z*-moment equation, 1 second-order *z*-moment equation, and 1 hexagon average flux constraint which states the average of the 6 triangle average fluxes is the same as the hexagon average flux. The linear system consisting of these constraints and unknowns can be represented by Eq. (6.7) for a given group where **A, X, Y, S** and $\mathbf{C^R}$ are 6x6 matrices while $\mathbf{C^Z}$ is a 2x2 matrix. Other matrices are all 1x1. The dimensions of the unknown vectors are defined accordingly.

The 36x36 linear system can be reduced algebraically to one equation. The order of elimination of the unknowns to reduce the linear system is as follows: $\bar{\boldsymbol{\varphi}}_{\mathbf{H}}$, $\bar{\boldsymbol{\varphi}}$, $\tilde{\boldsymbol{\varphi}}_{\mathbf{x}}$, $\tilde{\boldsymbol{\varphi}}_{\mathbf{y}}$, $\bar{\mathbf{J}}_{\mathbf{o}}^{\mathbf{R}}$, $\bar{\boldsymbol{\varphi}}_{\mathbf{s}}$, $\bar{\boldsymbol{\varphi}}_{\mathbf{p}}$, $\bar{\mathbf{J}}_{\mathbf{o}}^{\mathbf{Z}}$, $\tilde{\boldsymbol{\varphi}}_{\mathbf{z}}^{\mathbf{S}}$, and $\tilde{\boldsymbol{\varphi}}_{\mathbf{z}}^{\mathbf{F}}$. With this sequence of elimination, the first unknown to be determined is the first-order *z*-moment and the back substitution performed in the reverse order determines finally the hexagon average flux.

$$\begin{bmatrix} \mathbf{A}_1 & \mathbf{0} & \mathbf{0} & \mathbf{A}_2 & \mathbf{A}_3 & \mathbf{A}_4 & \mathbf{A}_5 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{X}_1 & \mathbf{0} & \mathbf{X}_1 & \mathbf{X}_2 & \mathbf{X}_3 & \mathbf{X}_4 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Y}_1 & \mathbf{Y}_2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{S}_1 & \mathbf{S}_2 & \mathbf{S}_3 & \mathbf{S}_4 & \mathbf{0} & \mathbf{S}_5 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{C}_1^{\mathbf{R}} & \mathbf{C}_2^{\mathbf{R}} & \mathbf{0} & \mathbf{0} & \mathbf{C}_3^{\mathbf{R}} & \mathbf{C}_4^{\mathbf{R}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_1 & \mathbf{0} & \mathbf{0} & \mathbf{P}_2 & \mathbf{P}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{C}_2^{\mathbf{Z}} & \mathbf{C}_3^{\mathbf{Z}} & \mathbf{C}_4^{\mathbf{Z}} & \mathbf{C}_5^{\mathbf{Z}} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{Z}_1^{\mathbf{F}} & \mathbf{0} & \mathbf{Z}_2^{\mathbf{F}} & \mathbf{Z}_3^{\mathbf{F}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{Z}_2^{\mathbf{S}} & \mathbf{0} & \mathbf{Z}_3^{\mathbf{S}} & \mathbf{0} & \mathbf{Z}_4^{\mathbf{S}} & \mathbf{Z}_5^{\mathbf{S}} \\ \mathbf{I}_6 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{6} \end{bmatrix} \begin{bmatrix} \overline{\boldsymbol{\varphi}} \\ \widetilde{\boldsymbol{\varphi}}_{\mathbf{x}} \\ \widetilde{\boldsymbol{\varphi}}_{\mathbf{y}} \\ \overline{\boldsymbol{\varphi}}_{\mathbf{s}} \\ \overline{\mathbf{J}}_{\mathbf{o}}^{\mathbf{R}} \\ \boldsymbol{\varphi}_{\mathbf{p}} \\ \overline{\mathbf{J}}_{\mathbf{o}}^{\mathbf{Z}} \\ \widetilde{\boldsymbol{\varphi}}_{\mathbf{z}}^{\mathbf{F}} \\ \widetilde{\boldsymbol{\varphi}}_{\mathbf{z}}^{\mathbf{S}} \\ \overline{\boldsymbol{\varphi}}_{\mathbf{H}} \end{bmatrix} = \begin{bmatrix} \overline{\mathbf{S}} \\ \widetilde{\mathbf{S}}_{\mathbf{x}} \\ \widetilde{\mathbf{S}}_{\mathbf{y}} \\ \overline{\mathbf{S}}_{\mathbf{s}} \\ \overline{\mathbf{S}}_{\mathbf{o}}^{\mathbf{R}} \\ \mathbf{S}_{\mathbf{p}} \\ \overline{\mathbf{S}}_{\mathbf{o}}^{\mathbf{Z}} \\ \widetilde{\mathbf{S}}_{\mathbf{z}}^{\mathbf{F}} \\ \widetilde{\mathbf{S}}_{\mathbf{z}}^{\mathbf{S}} \\ \mathbf{0} \end{bmatrix}. \tag{7.7}$$

The solution of the multigroup one-node problem provides multigroup node-average fluxes as well as multigroup interface currents. The multigroup fluxes define the nodal spectrum used in the dynamic condensation of the multigroup cross sections which is performed by spectrum weighting as follows:

$$\hat{\Sigma}_{\alpha G} = \sum_{g \in G} \varphi_g \Sigma_{\alpha g}, \quad G=1 \text{ or } 2 \tag{7.8}$$

where

$$\varphi_g = \frac{\phi_g}{\hat{\phi}_G} \quad \text{and} \quad \hat{\phi}_G = \sum_{g \in G} \phi_g .$$

On the other hand, the multigroup interface current is used to determine the two-group corrective nodal coupling coefficient by the following equation:

$$\hat{D}_G = -\frac{\widetilde{D}_G \left(\hat{\phi}_G^l - \hat{\phi}_G^r \right) + \hat{J}_G}{\hat{\phi}_G^l + \hat{\phi}_G^r} \tag{7.9}$$

where $\widetilde{D}_G$ is the finite difference based nodal coupling coefficient and

$$\hat{J}_G = \sum_{g \in G} J_g .$$

Once the corrective nodal coupling is determined, the two-group interface current can be represented as a linear combination of the two-group node average fluxes of the left and right nodes of the interface as follows:

$$J_G^{CMFD} = -\tilde{D}_G\left(\phi_G^{r,CMFD} - \phi_G^{l,CMFD}\right) - \hat{D}_G\left(\phi_G^{r,CMFD} + \phi_G^{l,CMFD}\right). \tag{7.10}$$

With this nodal coupling relation and the two-group constants, the CMFD linear system then can be formulated. The solution of the CMFD linear system yields a two-group flux distribution. At convergence of the iteration between the multigroup and two group solutions, the two-group nodal reaction rates preserve the corresponding multigroup reaction rates.

Once a two-group CMFD solution is obtained, the boundary conditions for the multigroup one-node problems have to be established by prolongation of the two-group information into the multigroup one. The prolongation can be achieved simply by using the multigroup shapes obtained in the previous one-node calculation. The multigroup shape for the flux would just be the spectrum used for group constant condensation. Additionally, the multigroup shape for the outgoing partial current is required to specify the multigroup incoming current conditions as well as the transverse leakages. The multigroup outgoing currents are obtained as follows by using the outgoing current spectra:

$$J_g^{o,n+1} = \varsigma_g^n \hat{J}_G^{o,n+1} \tag{7.11}$$

where

$$\varsigma_g^n = \frac{J_g^{o,n}}{\hat{J}_G^{o,n}} \quad \text{and} \quad \hat{J}_G^{o,n} = \sum_{g \in G} J_g^{o,n}.$$

The multigroup flux of the hexagonal node is obtained similarly using the spectra and then used to update the multigroup moments of the triangular nodes.

The multigroup one-node TPEN kernel has been implemented in the PARCS code using a nonlinear iteration scheme in order to control the alternate calculations of the two-group CMFD and multigroup one-node TPEN calculations. The T-H feedback on cross sections is performed in the multigroup structure and the multigroup cross sections are condensed into two groups using the nodal spectra determined by the multigroup TPEN kernel. A surface flux correction factor is used in much the same manner as D-hat in the cartesian CMFD solution to obtain the surface flux from the node average fluxes. The two-group surface flux and the net current determine the two-group partial current to be used in the multigroup prolongation of the outgoing partial current for TPEN. In the TPEN calculation, several sweeps (N_s) of the one-node problem is performed to achieve sufficient convergence of the multigroup solution. Further details about the TPEN implementation in PARCS, as well as the results of validation for a series of benchmark problems is provided in Reference [18].

8. FUEL CYCLE ANALYSIS

8.1 Fuel Depletion Analysis

In order to complete the fuel cycle analysis capability in PARCS, the depletion capability was then added to PARCS in order to time advance each fuel node and to simulate the actual core burnup conditions during the fuel cycle. The depletion capability is available in PARCS beginning with version 2.5. During each "burnup step" the power in each fuel node is used to time advance the burnup of the node, and the assembly cross sections derivatives and other neutronics parameters are computed at the new burnup using the PMAXS file for the corresponding fuel composition. Alternately, a given core "burnup" distribution can be specified by the user and the depletion module in PARCS will generate the fuel assembly neutronics data at the specified burnup. Thus the code provides the functionality to meet the two primary fuel cycle problems of the analyst, to deplete a core to find a typical "equilibrium" condition or to accept the burnup distribution for a specific core and to find the corresponding cross sections and steady-state condition. The current state of GENPMAXS/PMAXS/PARCS cross section functionality is described in the GENPMAXS manual, and an overview of the depletion capability is provided in section 7.8 of this Chapter.

The burnup distribution is calculated using the fluxes provided by PARCS as follows:

$$\Delta B_i = \Delta B_c \frac{P_i}{G_i} / \frac{P_c}{G_c} \tag{8.1}$$

where: i : ith depletion region, one region is one Z-direction node of a assembly, ΔB_i : burnup increase of ith region,

ΔB_c : Core average burnup increment in one step, specified in DEPLETOR input,

G_i : the heavy metal loading in ith region,

G_c: total heavy metal loading in the core($=\sum G_i$),

P_i: power in ith region,

P_c: Total power in core ($=\sum P_i$).

The total Heavy metal loading and the Power, G_i, P_i, respectively, can be calculated as following:

$$G_i = \rho_i \sum_{j \in i} V_j \tag{8.2}$$

$$P_i = \sum_{j \in i} V_j \left[\sum_g \left(\Phi_{g,j} \times \kappa\Sigma_{f\,g,j} \right) \right] \tag{8.3}$$

where: j:jth neutronic node in PARCS,

g: gth energy group,

Vj: volume of jth node, given by PARCS,

ρ_i : heavy metal density in ith region, provided in PMAXS,

$\Phi, \kappa\Sigma_f$: Fluxes and fission energy XS, given by PARCS.

History variables, such as control rod history (HCR), are defined as burnup weighted quantities. Using HCR as an example, it is calculated as follows:

$$HCR(B_p + \Delta B) = \frac{\int_0^{B_p + \Delta B} \alpha(B) dB}{B_p + \Delta B} = \frac{HCR(B_p) \times B_p + \alpha \Delta B}{B_p + \Delta B} \tag{8.4}$$

where:

Bp: burnup at beginning of this depletion step,

ΔB : burnup increment,

a: rodded fraction during this depletion step.

The PARCS PMAXS cross section file contains pre-tabulated cross sections which depend on up to 8 independent variables. These independent variables can be split into two groups: instantaneous variables, such as fraction of control rod insertion, moderator density, soluble boron concentration, fuel temperature, and moderator temperature; and history variables, such as burnup and two history variables from the following 5 history variables:

1) control rod history (HCR)

2) moderator density history(HMD)

3) soluble boron history(HSB)

4) fuel temperature history(HTF)

5) moderator temperature history(HTM)

The cross sections for a given region with a specified history and instantaneous state are generated by a two stage process. At the first stage, a cross section set is determined in the PARCS DEPLETION module using the PMAXS cross section file with the instantaneous values of all dependent variables for a specified history state of a particular region. At the second stage, using the cross section set generated by the DEPLETION module, a cross section is produced for the current instantaneous states using the following equation:

$$\Sigma(\vec{\alpha}, Dm, Sb, Tf, Tm) = \Sigma^r(Dm^r, Sb^r, Tf^r, Tm^r) + \sum_i \alpha_i \Delta\Sigma_{CR}{}^i(Dm^r, Sb^r, Tf^r, Tm^r)$$
$$+ \Delta Dm \left.\frac{\partial\Sigma}{\partial Dm}\right|_{(\vec{\alpha}, Dm^m, Sb^r, Tf^r, Tm^r)} + \Delta Sb \left.\frac{\partial\Sigma}{\partial Sb}\right|_{(\vec{\alpha}, Dm\ , Sb^m, Tf^r, Tm^r)}$$
$$+ \Delta Tf \left.\frac{\partial\Sigma}{\partial Tf}\right|_{(\vec{\alpha}, Dm\ , Sb\ , Tf^m, Tm^r)} + \Delta Tm \left.\frac{\partial\Sigma}{\partial Tm}\right|_{(\vec{\alpha}, Dm\ , Sb\ , Tf\ , Tm^m)} \tag{8.5}$$

where a reference cross section Σ^r (first term on the right hand side) is modified with contributions from the control rod insertion, $\Delta\Sigma^i$, moderator density, Δ Dm, the boron concentration, Δ Sb, the fuel temperature, Δ Tf, and the moderator temperature, Δ Tm.

For the control rod cross section contribution (second term on the right hand side), the partial cross section is provided simply as a perturbation of the cross section ai $\Delta\Sigma^i$. The weighting fraction, a, indicates the insertion fraction for each node and is provided for each control rod type, i. In the equation above $\vec{\alpha}$ is a vector with contents of each ai, since there are multiple control types in some BWR models (e.g. Ringhalls). If the a is flux weighted then the control rod effect is non-linear. If flux weighting is not available or not used, then the control rod effect is linear. Most all standard codes used in the industry (e.g. CASMO/SIMULATE[23]) employ linear weighting.

The contributions from the other independent variables (terms three, four, five, and six on the right hand side) are determined using the product of a partial cross section, $\partial\Sigma/\partial x$, and the amount of the perturbation for each independent variable:

$$\Delta Dm = Dm - Dm^r \quad Dm^m = (Dm + Dm^r)/2 = Dm^r + \Delta Dm/2 \tag{8.6}$$

$$\Delta Sb = Sb - Sb^r \quad Sb^m = (Sb + Sb^r)/2 = Sb^r + \Delta Sb/2 \tag{8.7}$$

$$\Delta Tf = Tf - Tf^r \quad Tf^m = (Tf + Tf^r)/2 = Tf^r + \Delta Tf/2 \tag{8.8}$$

$$\Delta Tm = Tm - Tm^r \quad Tm^m = (Tm + Tm^r)/2 = Tm^r + \Delta Tm/2 \tag{8.9}$$

The superscript, m, denotes the midpoint of two data points in the pre-tabulated cross section data. This provides a second order accurate estimate of the cross section. The partial cross sections are obtained by piecewise interpolation of the pre-tabulated data using a ìtree structureî. Additionally, PARCS uses region-wise XS and derivatives at a specified history state which are determined using a multi-dimensional linear interpolation, which is described in detail in the GENPMAXS manual.[21]

8.2 Multi-cycle Analysis

In order to perform multi-cycle, shuffling function is neccesary in addition to depletion capbility. The user can choose to either repeat one cycle or put a series of different cycles in specific order. A new functionality has been developed to perform fuel handling and disposition between cycles. This functionality enables the user to shuffle the fuel assemblies individually or together as batches, based on the shuffling map provided by the user. The node-wise depletion data is read as a matrix, then moves or discharges the assemblies following the shuffling map. In the batch mode, the code averages the axial distribution of the desired data over the assemblies within one batch as defined in the shuffle map before moving them.

Besides the node-wise burnup distribution, the code also reads matrices with the history data, including the control rod position, coolant density, fuel temperature, coolant density and boron concentration. There are two maps, location map and shuffle map, involved in this process. The location map, appears as a LOCATION subcard, and defines the radial fuel assembly configuration with the index of each assembly at the End of Cycle (EOC). It is similar to the existing RAD_CONF card in the GEOMHEX/GEOM block. The shuffle map (SHUFF_MAP subcard) provides the position of each assembly at the Beginning of Cycle (BOC). For each specific assembly, the code will read the index in the shuffle map and search for its old location by matching the index. If the index in the shuffle map is a negative number, it indicates the fuel is fresh. The LOCATION and SHUFF_MAP subcards work together to determine one shuffling method. The user can define the index of one method and the shuffling mode (individual or batch) in SHUFF_DEF card. The shuffling logic can be described as follows:

In the individual shuffling mode, assume the burnup in *k-th* node of the *j-th* assembly is $B_{j,k}$, and the reloading map ReloMAP created by location and shuffle map gives the index of the new location of the *j-th* assembly is $j=ReloMAP(i)$, then

$$B'_{i,k}=B_{j,k}$$

where $B'_{i,k}$ is the burnup in *k-th* node of the *i-th* assembly after shuffling. Similarly, if the history data in *k-th* node of the *j-th* assembly is $H_{j,k}$, the history data in *k-th* node of the *i-th* assembly after fuel shuffling becomes:

$$H'_{i,k}=H_{j,k}$$

If the *i-th* location in ReloMAP is 0, i.e., $j=0$, the new burnup distribution is $B'_{i,k}=0$ and history data $H'_{i,k}=0$.

In the batch shuffling mode, the code will radially average the burnup first and assign it to all the assemblies within the same batch. Assume the burnup in *k-th* node of the *j-th* assembly is $B_{j,k}$, and *j-th* assemblies is in batch M, then,

$$\bar{B}_{M,k} = \sum_{j \in M} w_{j,k} B_{j,k}$$

where $w_{j,k} = \dfrac{V_{j,k}}{\sum_{i \in M} V_{i,k}}$ and $V_{j,k}$ is volume of *k-th* node of the *j-th* assembly.

For all assemblies in batch M, the burnup after shuffling will be:

$$B'_{j,k} = \begin{cases} 0, M = 1 \\ \bar{B}_{M-1,k}, M > 1 \end{cases}, j \in M$$

Similarly, the history data is treated the same way as the burnup data:

$$\bar{H}_{M,k} = \sum_{j \in M} w_{j,k} H_{j,k}$$

and

$$H'_{j,k} = \begin{cases} 0, M = 1 \\ \bar{H}_{M-1,k}, M > 1 \end{cases}, j \in M$$

The nested iterations for determining the equilibrium core configuration is also implemented. A multi-cycle depletion table is created to configure defined cycles and fuel shuffling methods. The user can choose to terminate the iteration by setting up either the maximum number of iterations or the convergence criterion. If the iteration ends properly, PARCS will print out a depletion summary for each cycle.

In the multi-cycle depletion arrangement table, one cycle definition and one shuffling method are combined as a depletion unit called CYCLE_IND. The user can put one unit after another or just let the last depletion unit repeat. The CONV_EC card is added to the PARCS input representing the convergence criterion and maximum number of iterations for the termination of equilibrium cycle search. The default values for these are 50 MWG/TU and 20 (cycles), respectively.

Under some circumstances (e.g., depletion with long cycle length), it is necessary to determine the critical condition of the reactor. For PWRs, an adjustment is commonly made on the boron

concentrationand an algorithm is implemented in PARCS to determine the zero boron concentration. The logic behind this algorithm is straightforward and shown below.

Assume in PWR depletion, the boron concentration (ppm) changes the sign from the positive to negative in two successive depletion steps. Two node-wise burnup values ($B^{(i)}_{j,k}$, the subscripts j and k represent the node location, as introduced before) are available for two different values of boron concentration (ppm_i). The estimate of the burnup corresponding to zero boron concentration can then be obtained by the linear interpolation as

$$B_{j,k} = B^{(1)}_{j,k} - \frac{ppm_1}{ppm_1 - ppm_2}\left(B^{(2)}_{j,k} - B^{(1)}_{j,k}\right)$$

Similarly, other depletion parameters are also interpolated in the same way. For example, the interpolated node-wise power,

$$P_{j,k} = P^{(1)}_{j,k} - \frac{ppm_1}{ppm_1 - ppm_2}\left(P^{(2)}_{j,k} - P^{(1)}_{j,k}\right)$$

and the modified depletion length (days) of the last step

$$dDay_{j,k} = dDay^{(1)}_{j,k} - \frac{ppm_1}{ppm_1 - ppm_2}\left(dDay^{(2)}_{j,k} - dDay^{(1)}_{j,k}\right)$$

The interpolation process above is performed as soon as the negative boron concentration is detected, when the current depletion cycle is terminated as well. Several desired parameters (such as core average burnup, the maximum node-wise burnup, or the power peaking factor) in are then calculated based on the interpolation and printed in the output edit. PARCS will advance to the next cycle (if any) after the output is generated.

9. OTHER COMPUTATIONAL METHODS

In the previous sections the major computational methods required to solve a transient fixed source problem have been presented. Although the computational modules which incorporate these solution methods perform most of the work, there are other computational methods necessary to complete the eigenvalue and spatial kinetics calculations. In this section, several additional computational methods are discussed which are essential to complete the description of the computational methods implemented in PARCS.

9.1 Critical Boron Concentration Search

For several applications (e.g. fuel depletion analysis) it is necessary to determine the critical condition of the reactor. For PWRs, an adjustment is commonly made on the boron concentration and an algorithm was implemented in PARCS to determine the critical boron concentration (CBC). The logic behind the CBC search is very straightforward. Suppose that two eigenvalues ($k_{eff}^{(i)}$) are available for two different values of boron concentration (ppm_i). The estimate of the CBC corresponding to $k_{eff} = 1.0$ can then be obtained by the linear interpolation as:

$$ppm = ppm_2 + \frac{1 - k_{eff}^{(2)}}{k_{eff}^{(1)} - k_{eff}^{(2)}}(ppm_1 - ppm_2) \tag{9.1}$$

In oder to minimize the computational costs of the search and in PARCS, a partially converged solution obtained during the outer iteration is used to estimate the next iterate on the boron concentration. Specifically, only a few outer iterations are performed with a fixed value of boron concentration and the resulting eigenvalue is then taken as the eigenvalue estimate for the specified boron concentration. The new estimate of the CBC is then obtained according to the above equation using the two most recent pairs of *ppm* and k_{eff}. In the PARCS nonlinear iteration scheme, a periodical nodal update is performed even in the normal eigenvalue calculation in order to incorporate the changes in the nodal group constants due to thermal feedback and also to update the nodal coupling coefficients. Therefore, it is natural to synchronize the boron concentration update with other nodal updates in the case of the CBC search. The resulting CBC search logic in PARCS thus updates the boron concentration whenever there is a nodal update whether it is due to the T/H calculation or due to the two-node nodal calculation. Thus, the CBC search will not be performed if both the nodal calculation and the thermal-hydraulic feedback are turned off.

9.2 Cross Section Formalism

The fundamental fuel cycle problems performed in LWR analysis are to model the actual burnup condition of the reactor or to find a typical “equilibrium cycle” condition of the core in order to perform transient safety analysis. Both of these problems require more than just using a specified set of few group cross sections in PARCS to perform the kinetics problems. In order to solve with reasonable accuracy the steady-state eigenvalue problem for a core, it is necessary for

the burnup dependent few group cross sections to be parameterized as a function of at least five state variables:

α, Tf, Tm, Dm, Sb

control rod insertion, fuel temperature, moderator temperature, moderator density, and soluble boron concentration, respectively, as well as control rod and moderator density "history" variables.

The original cross section model that was incorporated into PARCS version 1.00 did not provide for burnup dependence and modeled the dependence of each state variable as linear, with the exception of the moderator density which was quadratic:

$$\Sigma(\alpha,Tf,Tm,Dm,Sb)=\Sigma^{r}+\alpha\Delta\Sigma^{cr}+\frac{\partial\Sigma}{\partial\sqrt{Tf}}\Delta\sqrt{Tf}+\frac{\partial\Sigma}{\partial Tm}\Delta Tm+\frac{\partial\Sigma}{\partial Dm}\Delta Dm+\frac{\partial\Sigma}{\partial Sb}\Delta Sb+\frac{\partial^2\Sigma}{\partial Dm^2}(\Delta Dm)^2 \tag{9.2}$$

where *Sb* is the boron concentration in *ppm* and the effective Doppler temperature is defined as a weighted average as:

$$\bar{T}_F = \omega\, T_F^{CL} + (1-\omega)\, T_F^{PS} \quad , \tag{9.3}$$

with the superscripts *CL* and *PS* designating centerline and pellet surface, respectively. After the T-H effect is incorporated, the cross sections are modified to incorporate the control rod effects as the following:

$$\Sigma(\xi) = \Sigma_{UR} + \alpha\xi\Delta\Sigma_c \tag{9.4}$$

where ξ is the nodal volume fraction of control rod, α is the flux weighting factor that accounts for the local flux depression in the control rod region, and $\Delta\Sigma_c$ is the cross section change due to the control rod when it is fully inserted into the node and is given as a composition dependent input. The flux weighting factor, α, used in Eq. (9.4) can be specified as a function of ξ. In this case, a generic function should be available prior to the program execution. Since the function relating α with ξ is not readily available, an approach to compute α during the solution process was implemented in PARCS. The control rod cusping correction method which provides α as well as some other correction factors to account for a partially inserted control rod in a node is described in Section 7.4.

This cross section functionality was adequate for performing various spatial kinetics benchmark problems such as the NEACRP control rod ejection cases in which the derivatives were provided as part of the benchmark specifications. The principal drawbacks of this model were that it was not possible to describe alternate burnup states and that the derivatives were accurate only for a limited range of conditions in the vicinity of the state in which they were computed. Because there was no burnup dependence of the cross sections, it was also not possible to simulate the burnup history effect.

The cross section capability in PARCS was therefore generalized to handle burnup dependent cross sections and to process a full range of branch cases necessary to model all anticipated steady-state and transient core thermal-hydraulic conditions. The code GENPMAXS was devel-

oped to process cross sections generated from lattice physics codes (e.g. TRITON[19], HELIOS[20]) and to write the PMAXS cross section file that could be read by PARCS for any burnup point in the fuel cycle. A separate user manual is provided for GENPMAXS.[21]

The new cross section model in PARCS included dependence on the same five state variables as in the previous model and used the same linear dependence. However, in the new model the derivatives are not fixed but are computed at the appropriate burnup point of the node by finite differencing over the range of thermal-hydraulic conditions computed by TRACE for each fuel node. In addition to the dependence on the five instantaneous state variables, the capability was added to treat "history" effects in which the cross section derivative depends on the previous spectral conditions of the fuel assembly. For example, if a fuel assembly has had a control rod inserted for a large part of the fuel cycle or if it has been operating at a high void fraction, then the neutron spectrum will have been "hardened" resulting in different fuel isotopics (e.g. more plutonium production) than the core average conditions. Several previous studies[22] by BWR fuel vendors and utilities have shown that it is important to explicitly model this effect. The functionality was added to PARCS to treat two history variables (e.g. control rod or void history) which are explicitly tracked for each fuel node during the fuel cycle.

In order to accurately model operating Light Water Reactors, the functionality was also added to GENPMAXS to track several additional neutronics parameters generated by the lattice physics code. The parameters generated by the lattice code that are processed in GENPMAXS and included in the PMAXS file are:

Assembly Discontinuity Factors

Corner Point Discontinuity Factors

Local Power Peaking Factors

Power Form Functions

Group-wise Form Function

Detector Information

Xe/Sm cross sections

Beta of Delayed Neutrons

Lambda of Delayed Neutrons

Spectrum of Delayed Neutrons

Decay Heat Data

Each of these parameters can be optionally treated with the same burnup and thermal-hydraulic dependencies as the principal cross sections, thereby providing the PARCS user with the option to match or exceed the capability of the fuel vendors or utilities.

9.3 Rod Cusping Correction

When the control rod is partially inserted into a node, the intranodal flux distribution is largely distorted due to the presence of the strong thermal absorber. In such cases, the volume weighting scheme to obtain the homogenized nodal cross section can lead to a significant error in the core calculation. When the intranodal flux variation is not properly incorporated, so called "rod cusping" effects are observed when the insertion depth of a control rod varies. The typical rod cusping effect occurs in eigenvalue calculations in such a way that the core k_{eff} varies in a cusp (or wavy) shape as the control rod insertion depth changes. The rod cusping effect is also observed in core power variation during a transient that involves a slow control rod motion.

To correct for the rod cusping effect, the flux redistribution should be properly reflected in the homogenized nodal cross section. In addition, the correctional nodal coupling coefficients (CNCC) which are normally obtained from a two-node calculation should be determined such that the intranodal cross section variation can be considered during the determination process. In PARCS, the intranodal flux distribution in a partially rodded node is obtained by solving a three-node problem by the fine mesh finite difference scheme. The flux weighting factor and the interface currents used to determine the CNCC can be readily obtained from the intranodal flux solution.

Consider an axial three-node domain as shown in Figure 9.1. The middle node is partially rodded and it is adjacent to a fully rodded and an unrodded node. For the three-node problem, a transverse-integrated neutron balance equation can be obtained in the same way as the two-node nodal problem. Since the transverse-integrated equation is one-dimensional and there are only a few partially rodded nodes in a reactor, it is feasible to employ a fine mesh finite difference scheme to solve the transverse-integrated equation, instead of employing the analytic nodal method for the three-node problem. For the finite-differenced one-dimensional problem, the coefficient matrix becomes primarily tridiagonal and the Gauss elimination can be applied efficiently to solve the linear system.

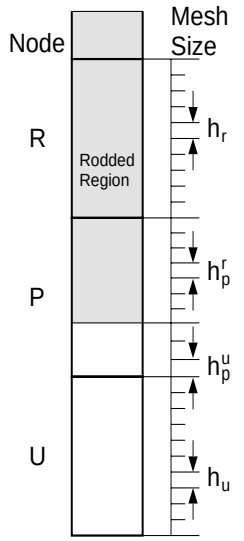


Figure 9.1: Geometry of Three-Node Problem

To solve a one-dimensional second-order differential equation, two boundary or constraint conditions must be specified. In a three-node problem, the boundary conditions may be specified at the two boundaries in terms of the neutron current. Another possibility is to use the node average fluxes of the rodded and unrodded nodes as the two constraints. Both the current and the node average flux are available from the previous CMFD calculation when the nonlinear nodal method is employed. In PARCS, the node average flux constraint is used because it assures the fine mesh solution is consistent with the coarse mesh solution as far as the node average fluxes are concerned. In the following subsections, the solution process of the three-node problem is presented.

9.3.1 Discretization in Fine Mesh Structure

The first step in the solution of the three-node problem is to define the fine mesh structure. For this, each coarse mesh is divided into N_{fine} fine meshes so that the total is $3 \times N_{fine} \equiv N_{total}$ meshes in the problem domain. In order to make the mesh boundary coincide with the material boundary within the partially rodded node, two different mesh sizes are used in the middle node while an equal mesh size is used in the other two nodes. The mesh structure is illustrated in Figure 9.1.

Once the fine mesh structure is determined, the next step is to construct the fine mesh transverse-leakage source which appears on the RHS of Eq. (4.1). Since it is described by a quadratic polynomial, the mesh average values of the transverse-leakage can be obtained easily through an integration process.

The last step in discretization is to employ the finite difference approximation to the current at the mesh boundaries and to write the mesh balance equation in terms of mesh-average fluxes for the N_{fine} - 2 interior meshes, excluding the two boundary meshes. Two additional equations come from the constraint of the node average flux in the upper and lower coarse nodes (Nodes *R* and *U* in Figure 9.1). Then a linear system consisting of N_{fine} equations and unknowns is constructed. The structure of the coefficient matrix is shown in Figure 9.2 for a two-group three-node problem when N_{fine}=10.

In Figure 9.2, the two constraint conditions are represented by the top and bottom rows which have N_{fine} 2x2 elements, respectively. Excluding these two rows, the matrix is a block tridiagonal matrix consisting of 2x2 blocks.

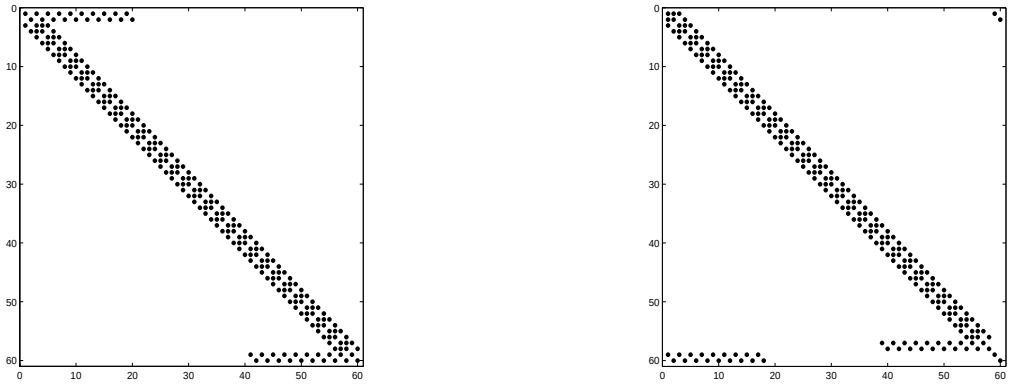


Figure 9.2: Matrix Structure of the Three-Node Problem; Left-Original, Right-Reordered

9.3.2 Solution of Three-Node Problem

The linear system involving the matrix whose structure is shown in Figure 9.2 can be solved by direct elimination. The elimination process, however, will cause nontrivial fill-ins in the upper triangular parts. The fill-in can be significantly reduced by a simple reordering. The right hand side part of Figure 9.2 shows the new structure of the matrix obtained by moving the first mesh to the bottom. With the new structure, the direct elimination can be done efficiently.

The solution of the linear system is the intranodal flux distribution. The flux weighting factor in the middle node can then be obtained as:

$$\alpha_g = \frac{\sum_{i=N_{fine}+N_p^\mu+1}^{2N_{fine}} \phi_g^i h_i}{\sum_{i=N_{fine}+1}^{2N_{fine}} \phi_g^i h_i} \tag{9.5}$$

where N_p^μ is the number of meshes in the unrodded region of the partially rodded node.

The intranodal flux distribution is also used to determine the currents at the upper and lower interfaces of the middle node. The interface currents can then be used in Eq. (4.60) to determine the CNCC. Since the CNCC is determined by the three-node solution, there is no need to perform the two-node ANM calculations for those interfaces. The solution of the three-node problem is

performed every time when a nodal update is performed, resulting in the new flux weighting factors and the CNCC to be used in the subsequent CMFD calculations.

9.4 Adjoint Calculation and Reactivity Edits

The adjoint solution of the initial eigenvalue problem is necessary to compute the dynamic reactivity during the transient calculation. Since the converged solution of the forward problem includes the converged CNCC used in the CMFD coefficient matrix, the adjoint problem within the nonlinear nodal method can be easily formulated by taking the transpose of the final CMFD coefficient matrix after the forward solution is obtained. The adjoint CMFD system can then be solved without invoking any nodal or T-H update using the same solution method as for the forward problem. The solution of the adjoint CMFD linear system is performed by using the same Krylov method and the same Wielandt shift method as the forward problem.

At any time point during a transient, the dynamic reactivity is defined as:

$$\rho = \frac{\langle \phi_0^*, A\phi \rangle}{\langle \phi_0^*, F\phi \rangle} \tag{9.6}$$

where A is the net production operator which is defined as $A = F - M$. Note that this definition gives $\rho_0 = 0$ if the steady-state values of the operators and the flux vector are used.

The changes in the core condition are expressed as perturbations to the net production operator:

$$A = A_0 + \Delta A \tag{9.7}$$

where ΔA represents the total change in operator A from every feedback component. This delta operator consists of several components including the external perturbations such as rod motion and boron changes as well as the T/H feedback effects. In terms of the components, ΔA is defined as:

$$\Delta A = \Delta A_{CR} + \Delta A_{PPM} + \Delta A_{TDOPL} + \Delta A_{TMOD} + \Delta A_{DENS} + \Delta A_{XESM} + \Delta A_{NL} \tag{9.8}$$

where the subscripts are defined as follows:

CR:	Control Rod Component
PPM:	Boron Concentration Component
DOPL:	Doppler Temperature Component
TMOD:	Moderator Temperature Component
DENS:	Moderator Density Component
XESM:	Xenon/Samarium Component
NL:	Nodal Leakage Component

Note that the NL component refers to effect of the higher-order neutron leakage resulting from the non-linear nodal calculation. This term has a non-linear dependence in each of the other terms, and thus the simplest way to treat this component is to separate it out. However, for most transients, this component will largely be affected by the change in control rod positions, and thus can reasonably be ‘lumped’ with the CR reactivity component.

The reactivity components can now be defined as:

$$\rho_X = \frac{\langle \phi_0^*, A_X \phi \rangle}{\langle \phi_0^*, F\phi \rangle} - \frac{\langle \phi_0^*, A_0 \phi \rangle}{\langle \phi_0^*, F\phi \rangle} \quad , \qquad X \in \{ CR, PPM, TDOPL, TMOD, DENS, XESM, NL \} \tag{9.9}$$

Note that this definition gives:

$$\rho_{TOT} = \rho_{CR} + \rho_{PPM} + \rho_{TDOPL} + \rho_{TMOD} + \rho_{DENS} + \rho_{XESM} + \rho_{NL} + \rho_{NULL} \tag{9.10}$$

where ρ_{NULL} comes from Eq. (9.7) and is defined as:

$$\rho_{NULL} = \frac{\langle \phi_0^*, A_0 \phi \rangle}{\langle \phi_0^*, F\phi \rangle} \tag{9.11}$$

By definition, this term should be zero. However, due to numerical inaccuracies in the calculation of the adjoint flux, this term is usually non-zero (although the magnitude is almost negligible), and is thus rigorously included in the reactivity formulation.

9.5 Decay Heat

When a reactor is scrammed, the power does not immediately drop to zero, but falls rapidly before following a negative period determined by the longest-lived delayed neutron group. After a reactor is shut down, substantial amounts of heat continue to be released through the radioactive decay of both fission products and transuranics in the fuel rods. The amount of heat released depends on the fission product concentrations and therefore the operating history of the reactor. Rigorous computation of decay heat release over time can be carried out by solving a series of coupled differential equations for the hundreds of fission products and their daughter nuclides. This computation is simplified by fitting a measured decay heat curve to a series of decay heat groups; analogous to delayed neutron groups.[24]

The total volumetric heat density, $q_T(\vec{r}, t)$, with decay heat contributions is given by

$$q_T(\vec{r}, t) = (1 - \alpha_T) \sum_{g=1}^{G} \kappa_g \Sigma_{fg}(\vec{r}, t) \phi_g(\vec{r}, t) + \sum_{i=1}^{I} \zeta_i D_i(\vec{r}, t) \tag{9.12}$$

where

$D_i(\vec{r}, t)$ = concentration of decay heat precursors in decay heat group i [J/cm^3]

ζ_i = decay constant of decay heat group i [sec^{-1}]

$\alpha_T = \sum_{i=1}^{I} \alpha_i$ = total fraction of the fission energy appearing as decay heat, where I is the total number of decay heat groups

α_i = fraction of total fission energy appearing as decay heat for decay heat grp. i

The following differential equation represents the concentration of the decay heat precursors, $D(\vec{r}, t)$.

$$\frac{\partial}{\partial t} D_i(\vec{r}, t) = \alpha_i \sum_{g=1}^{G} \kappa_g \Sigma_{fg}(\vec{r}, t) \phi_g(\vec{r}, t) - \zeta_i D_i(\vec{r}, t) \tag{9.13}$$

An expression for the decay heat precursor concentrations can be developed by integrating Eq. (9.13) over the time interval, $\Delta t = t_{n+1} - t_n$. This operation results in the following equation

$$D_i(\vec{r}, t_{n+1}) = D_i(\vec{r}, t_n) e^{-\zeta_i \Delta t} + \alpha_i \int_{t_n}^{t_{n+1}} \left(\sum_{g=1}^{G} \kappa_g \Sigma_{fg}(\vec{r}, t') \phi_g(\vec{r}, t') e^{-\zeta_i (t_{n+1} - t')} \right) dt' \tag{9.14}$$

A functional form for the time-dependent fission source density must be developed to solve the integral in Eq. (9.14). Assuming the fission source density is constant at the past time step value over the time interval $t' \subset [t_n, t_{n+1}]$, the fission source density becomes

$$\sum_{g=1}^{G} \kappa_g \Sigma_{fg}(\vec{r}, t') \phi_g(\vec{r}, t') = \sum_{g=1}^{G} \kappa_g \Sigma_{fg}(\vec{r}, t_n) \phi_g(\vec{r}, t_n) \tag{9.15}$$

Incorporating this approximation into Eq. (9.14) and simplifying the equation gives the desired expression

$$D_i(\vec{r}, t_{n+1}) = D_i(\vec{r}, t_n) e^{-\zeta_i \Delta t} + \frac{\alpha_i}{\zeta_i} [1 - e^{-\zeta_i \Delta t}] \sum_{g=1}^{G} \kappa_g \Sigma_{fg}(\vec{r}, t_n) \phi_g(\vec{r}, t_n) \tag{9.16}$$

It is assumed that in the steady-state condition, the longest-lived decay heat precursor group is in equilibrium. The steady-state concentration is calculated by setting the time-dependent derivative in Eq. (9.13) to zero and solving for the precursor concentration:

$$D_{i_\infty} = \frac{\alpha_i}{\zeta_i} \sum_{g=1}^{G} \kappa_g \Sigma_{fg} \phi_g \tag{9.17}$$

Eq. (9.17) is thus used to determine the initial conditions required for the transient solution of the decay heat.

9.6 Xenon/Samarium Treatment

In order to treat global and local power shifts resulting from Xenon/Samarium imbalances during power maneuvering, it is necessary to keep track of the concentrations of these fission products and their global/local reactivity effects. For this reason, the treatment of Xenon/Samarium has been implemented and is described for both the equilibrium and transient cases. First, the time-dependent depletion of the fission products Iodine, Xenon, Promethium, and Samarium, which is used for updating the number densities and thus the absorption cross sections, is described by the following differential equations:

$$\frac{d}{dt}N_I^l(t) = \gamma_I^l \sum_{g=1}^{G} \Sigma_{fg}^l(t)\phi_g^l(t) - \lambda_I^l N_I^l(t) \tag{9.18}$$

(9.19)

$$\frac{d}{dt}N_{Xe}^l(t) = \lambda_I^l N_I^l(t) + \gamma_{Xe}^l \sum_{g=1}^{G} \Sigma_{fg}^l(t)\phi_g^l(t) - \lambda_{Xe}^l N_{Xe}^l(t) - \sum_{g=1}^{G} \sigma_{Xe,ag}^l(t)\phi_g^l(t)N_{Xe}^l(t) \tag{9.20}$$

for the I^{135}-Xe^{135} chain, and

$$\frac{d}{dt}N_{Pm}^l(t) = \gamma_{Pm}^l \sum_{g=1}^{G} \Sigma_{fg}^l(t)\phi_g^l(t) - \lambda_{Pm}^l N_{Pm}^l(t) \tag{9.21}$$

$$\frac{d}{dt}N_{Sm}^l(t) = \lambda_{Pm}^l N_{Pm}^l(t) - \sum_{g=1}^{G} \sigma_{Sm,ag}^l(t)\phi_g^l(t)N_{Sm}^l(t) \tag{9.22}$$

for the Pm^{149}-Sm^{149} chain, where

$N_i^l(t)$ = nuclei number density of isotope *i*

$\sigma_{i,ag}^l(t)$ = group-wise microscopic absorption cross section of isotope *i*

γ_i^l = effective yield (atoms/fission) of isotope *i,* and

λ_i^l = decay constant of isotope *i*

The steady-state number densities can be obtained from Eq. (9.18) through Eq. (9.22) as:

$$N_{I,\infty}^{l} = \frac{\gamma_I^l \sum_{g=1}^{G} \Sigma_{fg}^l \phi_g^l}{\lambda_I^l} \tag{9.23}$$

$$N_{Xe,\infty}^{l} = \frac{\lambda_I^l N_{I,\infty}^l + \gamma_{Xe}^l \sum_{g=1}^{G} \Sigma_{fg}^l \phi_g^l}{\lambda_{Xe}^l + \sum_{g=1}^{G} \sigma_{Xe,ag}^l \phi_g^l} \tag{9.24}$$

for the I^{135}-Xe^{135} chain, and

$$N_{Pm,\infty}^{l} = \frac{\gamma_{Pm}^l \sum_{g=1}^{G} \Sigma_{fg}^l \phi_g^l}{\lambda_{Pm}^l} \tag{9.25}$$

$$N_{Sm,\infty}^{l} = \frac{\lambda_{Pm}^l N_{Pm,\infty}^l}{\sum_{g=1}^{G} \sigma_{Sm,ag}^l \phi_g^l} \tag{9.26}$$

for the Pm^{149}-Sm^{149} chain.

For the transient Xenon calculation, forward-differencing Eq. (9.18) through Eq. (9.22) results in the following equations for updating the fission product number densities:

$$N_I^l(t+\Delta t) = N_I^l(t) + \Delta t \left[\gamma_I^l \sum_{g=1}^{G} \Sigma_{fg}^l(t) \phi_g^l(t) - \lambda_I^l N_I^l(t) \right] \tag{9.27}$$

$$N_{Xe}^l(t+\Delta t) = N_{Xe}^l(t) + \Delta t \left[\gamma_{Xe}^l \sum_{g=1}^{G} \Sigma_{fg}^l(t) \phi_g^l(t) + \lambda_I^l N_I^l(t) - \lambda_{Xe}^l N_{Xe}^l(t) - \sum_{g=1}^{G} \sigma_{Xe,ag}^l(t) \phi_g^l(t) N_{Xe}^l(t) \right] \tag{9.28}$$

for the I^{135}-Xe^{135} chain, and

$$N_{Pm}^{l}(t+\Delta t) = N_{Pm}^{l}(t) + \Delta t\left[\gamma_{Pm}^{l}\sum_{g=1}^{G}\Sigma_{fg}^{l}(t)\phi_{g}^{l}(t) - \lambda_{Pm}^{l}N_{Pm}^{l}(t)\right] \tag{9.29}$$

$$N_{Sm}^{l}(t+\Delta t) = N_{Sm}^{l}(t) + \Delta t\left[\lambda_{Pm}^{l}N_{Pm}^{l}(t) - \sum_{g=1}^{G}\sigma_{Sm,ag}^{l}(t)\phi_{g}^{l}(t)N_{Sm}^{l}(t)\right] \tag{9.30}$$

for the Pm^{149}-Sm^{149} chain.

The equilibrium number densities shown in Eq. (9.23) through Eq. (9.26) are updated during the steady-state calculation at the same frequency as the nonlinear updates (both two-node and thermal-hydraulic), and the time-dependent number densities shown in Eq. (9.27) through Eq. (9.30) are updated at the beginning of each transient time step. Once these number densities have been calculated, the macroscopic absorption cross section is updated as follows:

$$\Sigma_{ag}^{l} = \Sigma_{ag}^{l} + \Delta\Sigma_{Xe,ag}^{l} + \Delta\Sigma_{Sm,ag}^{l} \tag{9.31}$$

where,

$$\Delta\Sigma_{Xe,ag}^{l} = \sigma_{Xe,ag}^{l}N_{Xe}^{l} \qquad \textit{and} \qquad \Delta\Sigma_{Sm,ag}^{l} = \sigma_{Sm,ag}^{l}N_{Sm}^{l} \tag{9.32}$$

9.7 One-Dimensional Kinetics

The 1D kinetics equation can be derived by integrating the 3D time-dependent neutron diffusion equation over the radial domain. The solution of the 1D kinetics equation is relatively simple because it involves only a block tridiagonal linear system which can be solved directly by the Gaussian elimination scheme. In order to retain good spatial solution accuracy, the nonlinear analytic nodal method (ANM) implemented in the PARCS code is to be used in the 1D solver. Due to the use of the ANM, the axial node size of the 1D module can be chosen as large as 20-30 cm, resulting in only 12 to 18 axial nodes. Hence the computational burden in the solution of the 1D kinetics equation is trivial. The methods for eigenvalue calculation and temporal discretization are chosen to be the same as the PARCS code in order to maintain consistency. In the following section, the 1D kinetics equation is derived and planar averaged group constants are defined. During the derivation, the current conservation factor (CCF) is introduced which guarantees the same axial neutron currents as the 3D reference values are obtained from the 1D equation. With the planar averaged group constants and the CCF's, it becomes possible to reproduce the 3D reference solution from the 1D model. The specific expressions for the CCF is then derived in Section 9.7.2. In order to obtain planar cross sections as functions of state parameters, two cross section functionalization schemes are considered here: the TRAC-B type polynomial representation and a generalized tabular representation, which are described in Section 9.7.3. The detailed solution methods for the eigenvalue and transient fixed source problems, as well as the temporal discretization methods are omitted here since they are already described in Chapter 3.

9.7.1 Derivation of One-Dimensional Kinetics Equation

The 3D time-dependent two-group neutron diffusion equation in Cartesian coordinate can be written in standard notation as:

$$\frac{1}{v_g}\frac{\partial \phi_g}{\partial t} = Q_g - \left(\frac{\partial J_{gx}}{\partial x} + \frac{\partial J_{gy}}{\partial y} + \frac{\partial J_{gz}}{\partial z} + \Sigma_{rg}\phi_g\right) \tag{9.33}$$

where

$$J_{gu} = -D_g \frac{\partial \phi}{\partial u} \tag{9.34}$$

and

$$Q_g = \begin{cases} (1-\beta)(\nu\Sigma_{f1}\phi_1 + \nu\Sigma_{f2}\phi_2) + \sum_{k=1}^{K}\lambda_k C_k \ , & g = 1 \\ \Sigma_{12}\phi_1 \ , & g = 2 \end{cases} . \tag{9.35}$$

Before integrating Eq.(9.33) over the radial domain, the flux is factorized into two independent functions, which are defined as the 1D flux (φ) and radial shape function (Φ), respectively, as follows:

$$\phi_g(x, y, z, t) = \varphi_g(z, t)\Phi_g(x, y, z, t) \quad . \tag{9.36}$$

Integration of the left hand side (LHS) of Eq.(9.33) can then be performed using the factorized flux:

$$\begin{aligned} \int_A \frac{1}{v_g}\frac{\partial \phi_g}{\partial t} dA &= \int_A \frac{1}{v_g}\left(\Phi_g \frac{\partial \varphi_g}{\partial t} + \varphi_g \frac{\partial \Phi_g}{\partial t}\right) dA \\ &= \frac{\partial \varphi_g}{\partial t}\int_A \frac{\Phi_g}{v_g} dA + \varphi_g \frac{\partial}{\partial t}\int_A \frac{\Phi_g}{v_g} dA \end{aligned} \quad . \tag{9.37}$$

In the above derivation, it was assumed that the neutron velocities are time-independent. Since Eq.(7.44) is an arbitrary factorization, it is necessary to impose a constraint on the radial shape function to make the factorization unique. The constraint is chosen such that Eq.(9.37) can be simplified. Specifically,

$$\int_A \frac{\Phi_g}{v_g} dA = \frac{A}{\overline{v_g}(z)} \equiv \int_A \frac{1}{v_g(r)} dA \tag{9.38}$$

where A is the area of the radial domain. The second term in Eq.(9.37) vanishes because the integral term is constant over time and the LHS term reduces to:

$$\int_A \frac{1}{v_g}\frac{\partial \phi_g}{\partial t} dA = \frac{A}{\overline{v_g}}\frac{\partial \varphi_g}{\partial t} \ . \tag{9.39}$$

The integration of the removal term on the right hand side (RHS) of Eq.(9.33) becomes:

$$\int_A \Sigma_{rg}\phi_g dA = \varphi_g \int_A \Sigma_{rg}\Phi dA_g = A\bar{\Sigma}_{rg}\varphi_g \tag{9.40}$$

where the planar averaged removal cross section is defined as:

$$\bar{\Sigma}_{rg} \equiv \frac{1}{A}\int_A \Sigma_{rg}\Phi_g dA. \tag{9.41}$$

The other types of planar averaged cross sections appearing in the source terms (Q_g) can be defined similarly.

The integration of the radial leakage term is simplified by Gauss' Theorem as:

$$\int_A \left(\frac{\partial J_{gx}}{\partial x} + \frac{\partial J_{gy}}{\partial y}\right) dA = \oint_B J_{gx} dy + \oint_B J_{gy} dx = \varphi_g \left(\oint_B -D_g \frac{\partial \Phi_g}{\partial x} dy + \oint_B -D_g \frac{\partial \Phi_g}{\partial y} dx\right) = A\Sigma_{Lg}\varphi_g \tag{9.42}$$

where Σ_{Lg} is the leakage cross section defined as:

$$\Sigma_{Lg} = \frac{1}{A\varphi_g}\left(\oint_B J_{gx} dy + \oint_B J_{gy} dx\right) \tag{9.43}$$

The integration of the axial leakage term proceeds first by decomposing the axial current term as:

$$J_{gz} = -D_g \frac{\partial \phi_g}{\partial z} = -D_g \left(\varphi_g \frac{\partial \Phi_g}{\partial z} + \Phi_g \frac{\partial \varphi_g}{\partial z}\right) . \tag{9.44}$$

The integration now yields:

$$\int_A \frac{\partial J_{gz}}{\partial z} dA = \frac{\partial}{\partial z}\int_A -D_g \left(\Phi_g \frac{\partial \varphi_g}{\partial z} + \varphi_g \frac{\partial \Phi_g}{\partial z}\right) dA = -\frac{\partial}{\partial z}\int_A D_g \Phi_g dA \frac{\partial \varphi_g}{\partial z} - \frac{\partial}{\partial z}\left(\varphi_g \int_A D_g \frac{\partial \Phi_g}{\partial z} dA\right) . \tag{9.45}$$

The first term on the RHS of Eq.(9.45) can be simplified by introducing the planar averaged diffusion coefficient defined as:

$$\overline{D}_g \equiv \frac{1}{A}\int_A D_g \Phi_g dA . \tag{9.46}$$

The second term is more complicated. This term would be zero if $\frac{\partial \Phi_g}{\partial z} = 0$, which implies that the radial flux shape is uniform over the axial direction, which is not the normal case. This term can be retained explicitly using the following definition of the shape dependent current:

$$\tilde{J}_g \equiv -\frac{\varphi_g}{A}\int_A D_g \frac{\partial \Phi_g}{\partial z} dA , \tag{9.47}$$

Eq. (9.45) then reduces to:

$$\int_A \frac{\partial J_{gz}}{\partial z} dA = A\frac{\partial}{\partial z}(\overline{J}_g + \tilde{J}_g) \tag{9.48}$$

where

$$\overline{J}_g = -\overline{D}_g \frac{\partial \varphi_g}{\partial z} \tag{9.49}$$

Using Eqs. (9.39), (9.40), (9.42), and (9.48), the 1D kinetics equation can be obtained as follows (after removing A from both sides):

$$\frac{1}{\overline{V}_g} \frac{\partial \varphi_g}{\partial t} = \overline{Q}_g - \left(\hat{\Sigma}_{rg} \varphi_g + \frac{\partial}{\partial z} \hat{J}_g \right) \tag{9.50}$$

where the effective removal cross section, $\hat{\Sigma}_{Rg}$, is defined as follows by adding the radial leakage cross section:

$$\hat{\Sigma}_{rg} \equiv \overline{\Sigma}_{rg} + \Sigma_{Lg} \tag{9.51}$$

and the total axial current, $\hat{J}_g$, combines both components of the current, i.e. :

$$\hat{J}_g \equiv \overline{J}_g + \tilde{J}_g. \tag{9.52}$$

When a flux distribution is available from a reference 3D calculation, the planar averaged group constants can be obtained by evaluating Eq. (9.38), Eq. (9.43) and Eq. (9.46) for each plane. If the current due to the difference in the radial shape ($\tilde{J}_g$) is neglected, which is small compared to the current due to the difference in the 1D flux ($\overline{J}_g$), then Eq. (9.50) can be solved for the 1D flux, φ_g. In such case, however, it is not possible to reproduce exactly the 3D base values of the eigenvalue and axial flux distribution in the 1D calculation. The problem arises when $\tilde{J}_g$ is explicitly considered. Since this term does not contain the derivative of the 1D flux as identified in Eq. (9.47), inclusion of this term makes Eq. (9.50) no longer a diffusion equation. Moreover, because this term involves $\frac{\partial \Phi}{\partial z}^g$ it can not be readily evaluated in normal 3D nodal calculations in which the intranodal expansion is obtained for the normal flux and not for the shape and 1D flux separately. Thus, defining an exact collapsed group constant for the integral in Eq. (9.47) is not possible. In order to overcome this problem in the framework of the nonlinear nodal method, the concept of flux discontinuity is introduced here such that the total axial current (J_g) determined in a 3D reference calculation is preserved in the 1D nodal calculation. With the discontinuity factor whose definition is detailed in the next section, it now becomes possible to reproduce exactly the 3D results in the 1D calculation for the reference conditions. The 1D model can then be applied to other perturbed states.

9.7.2 The Current Conservation Factor

If the planar averaged group constants, fluxes, and interface currents from a 3D nodal calculation are assumed to be known for two neighboring planes, it is possible to formulate a two-node

problem to determine the nodal coupling relation which can be used to represent the interface current in terms of two node (or planar) averaged fluxes as:

$$\hat{J}_g = -\tilde{D}_g(\varphi_g^t - \varphi_g^b) - \hat{D}_g(\varphi_g^t + \varphi_g^b) \tag{9.53}$$

where the superscript *t* and *b* stand for top and bottom node of the two nodes, respectively. In normal two-node problem, the interface current as well as the correctional nodal coupling coefficient (CNCC), $\hat{D}_g$, is the free parameter to be determined from the two-node nodal calculation. In order to solve the normal two-node problem, four constraints are imposed per group. These are two node average fluxes and the flux and current continuity at the interface. In the present problem, however, the interface current is not a free parameter, rather it is considered as an additional constraint.

The two-group ANM solution for a node is given by the following (refer to Chapter 4 for the details of the derivation of the ANM solution):

$$\begin{bmatrix}\varphi_1(z)\\ \varphi_2(z)\end{bmatrix} = \begin{vmatrix}\varphi_1^H(z)+\varphi_1^P(z)\\ \varphi_2^H(z)+\varphi_2^P(z)\end{vmatrix} = \begin{bmatrix}r & s\\ 1 & 1\end{bmatrix}\begin{bmatrix}a_{21}sn(\kappa x)+a_{22}cn(\kappa x)\\ a_{23}sn(\mu x)+a_{24}cn(\mu x)\end{bmatrix} + \begin{bmatrix}c_{10}+c_{11}f_1(\xi)+c_{12}f_2(\xi)\\ c_{20}+c_{21}f_1(\xi)+c_{22}f_2(\xi)\end{bmatrix} \tag{9.54}$$

In the above equation, the *c* coefficients are determined by the transverse leakage and/or the transient fixed source, which are zero in the 1D steady-state case. In a two-node problem, there are then eight coefficients (2*4 *a*'s) to be determined. In order to determine them uniquely, the eight constraint conditions must be specified. These are the flux continuity conditions for two groups, the four node-average flux constraints (2 nodes x 2 groups), and two current continuity conditions (2 groups). The flux continuity conditions are given by:

$$\zeta_g^b \varphi_g^b\left(\frac{h_x^l}{2}\right) = \zeta_g^t \varphi_g^t\left(-\frac{h_x^r}{2}\right) \tag{9.55}$$

where ζ_g is the discontinuity factor that is assumed to be known in normal two-node calculations.

If the two interface currents (one for each group) are added as the additional condition in the two-node problem, then two additional unknowns should be introduced. For this purpose, it is possible to introduce the current conservation factor (CCF), ε_g, and represent the discontinuity factors as follows:

$$\zeta_g^b = 1-\varepsilon_g; \qquad \zeta_g^t = 1+\varepsilon_g \tag{9.56}$$

The current conservation factor thus becomes an additional unknown for each group. The ten unknowns can be simultaneously determined by imposing the ten constraints.

The planar cross sections defined by Eq. (9.41), Eq. (9.43), or Eq. (9.46) can be obtained for each plane after a 3D steady-state calculation is performed. Since these cross sections are func-

tions of state parameters such as coolant density and boron concentration, several 3D calculations need to be performed to generate planar cross sections at various states. The planar cross sections then can be properly functionalized for use in the 1D calculations. The following section provides descriptions of the two cross section representation schemes implemented in the 1D module.

9.7.3 TRAC-B Polynomial Scheme

In TRAC-B, the planar cross sections are represented by the following polynomial form:

$$\Sigma(\alpha, T_f, T_m, \xi) = \xi(a_1 + a_2\alpha + a_3\alpha^2) + (1-\xi)(a_4 + a_5\alpha + a_6\alpha^2) + a_7(\sqrt{T_f} - \sqrt{T_{f0}}) + a_8(T_m - T_{m0}) + a_9 B \tag{9.57}$$

where

α = void fraction,
T_f = Doppler temperature in K,
B = boron concentration in ppm, and
ξ = control rod fraction of Bank.

The T-H state variables used are the planar averaged values which are determined by the 3D TRAC results at each time point. This representation is considered adequate for BWR's. The only limitation would be that no distinction can be made in the worth of different control banks.

For general applications of the 1D kinetics module to include PWRs, the following cross section representation scheme is introduced which combines a tabular and polynomial representation:

$$\Sigma(\rho, T_f, B, \xi) = \Sigma(\rho, T_f) + \sum_{i=1}^{I} \xi_i \Delta\Sigma_{Ri}(\rho, T_f) + (a_1 + a_2\Delta\rho)\Delta B \tag{9.58}$$

where

ρ = coolant density in g/cc,
ξ_i = control rod fraction of Bank *i*,
$\Delta\Sigma_{Ri}$ = cross section change due to insertion of control bank i,
$\Delta\rho$ = coolant density change from the reference value in g/cc, and
ΔB = boron concentration change from the reference value in ppm.

In this representation, the base and control rod cross sections are given in a 2-dimensional table form in which the independent variables are coolant density and fuel temperature. These two independent variables were chosen because the coolant density and fuel temperature can undergo large variations during the transient calculation. The use of tables makes it possible to include cross term effects and to extend the range of application of the cross section set. The boron concentration effect here is separated out because it is presumed that the cross section sets are generated for a given burnup state and that the range of ppm variation around the critical boron concentration at that burnup would be small. However, in order to retain accuracy, the density dependence of the ppm derivative of the *macroscopic* cross section is included. The tables are to be provided for each type of two-group cross sections or a total of nine types. Note that a separate

delta cross section table is required for each bank. The void history effects characteristic of BWRs are assumed to be incorporated into the base cross section table.

9.7.4 Quasi-Static One-Dimensional Kinetics Calculation

The conventional way of obtaining planar cross sections during the 1D calculation is to use directly one of the two functionalization schemes presented above. It requires, however, a preparation of either the polynomial coefficients or the table entries beforehand, which can require considerable effort. When a 3D PARCS model is available, it is possible to use the cross section evaluation routines of the 3D module directly. In such case, only the 3D flux calculation function of the 3D module is replaced by the 1D module and the rest of the 3D modules can be executed as if it is the normal 3D case. The radial power distributions needed in the 3D TRAC calculation can be obtained by multiplying the prespecified normalized radial power distribution by the planar power calculated by the 1D module. As long as the radial flux shape and the leakage cross section are prespecified, the planar cross section can be obtain through the normal collapsing procedure. Hence the preparation of 1D cross sections a priori is not necessary. Instead, the normalized flux distributions as well as the leakage cross sections need to be specified in the input. The information on radial shapes can be specified as time-dependent. A radial shape function can be assumed invariant or changing linearly over a specified time interval. This approach is thus referred to as the quasi-static 1D calculation.

In principle, the 3D transient results can be exactly reproduced using the quasi-static approach if the accurate time-dependent radial shapes are provided with fine time step sizes. Exact reproduction of the 3D results, however, is not practical because accurate radial shape functions can only be possible by running the 3D transient case a priori. Instead, the radial shapes generated for a few steady-states can be used as a good approximation. It is expected that the states for which the radial shapes need to be precalculated are only the sates involving a large change in either the control rod configuration or flow distribution that causes a significant change in the radial flux shape.

The solution sequences for 1D steady-state and transient calculations are basically the same as the 3D. There are, however, two additional steps in the 1D calculation. One is the radial averaging of the T-H variables in case of the normal 1D calculations or the radial cross section collapsing in the case of the quasi-static calculation. The other is the generation of 3D power shapes for use in the system T-H code. The steady-state and transient solution sequences are shown in Figures 7.3 and 7.4, respectively. Both figures show different paths for the two modes of the 1D calculation - the normal mode and the quasi-static mode.

After input processing and initialization, the steady-state calculation begins with evaluating planar cross sections at the initial T-H condition. In the normal mode, the T-H state variables are radially-averaged with volume weighting while in the quasi-static mode, the node-wise cross sections are radially-collapsed with flux-volume weighting. One set of the T-H volume weighting factors should be specified in the input for the normal mode, whereas the radial flux shapes are specified for each plane for the quasi-static mode. Once the planar cross sections are determined, the coarse-mesh finite difference (CMFD) linear system can be formulated for the eigenvalue problem. The CMFD linear system is tridiagonal so it can be solved by Gaussian elimination in the normal mode for which new solution routines will be written. In case of the quasi-static case,

the existing PARCS Krylov CMFD solver will be used although the problem is one-dimensional. The reason for this choice is that it is necessary and also possible to use most of the existing routines for the quasi-static calculation and thus there is no need for introducing changes to the CMFD solver.

Once the CMFD solution is obtained for a given eigenvalue, the eigenvalue is updated by using the Wielandt shift method. The RHS source term and the diagonal term in the linear system are then updated accordingly. After a fixed number of the fission source iterations, the nodal update is invoked to update the nodal coupling coefficient. During the nodal updates, two-node ANM calculations are performed using the node-average flux distribution obtained from the CMFD solution. The axial discontinuity factors are used in the two-node calculations. After the nodal update, a T-H calculation is performed. The 3D power distributions needed for system T-H calculations are obtained by multiplying the 1D power by the normalized radial power shape function. The new T-H field variables are used to update the planar cross sections through either averaging of T-H variables or radial collapsing of node-wise cross sections, depending on the execution mode. In the 1D cross section evaluation, one of the two cross section representations schemes represented by Eq.(7.66) or Eq. (7.67) Eq. (9.58) is chosen as an input option.

The transient solution sequence begins with incorporating the external perturbations such as control rod movements into the transient fixed source problem. The temporal discretization scheme used for the 1D module is also the same as the 3D method. Specifically, the 1D kinetics equation is discretized by the theta method combined with the exponential transformation method. The temporal discretization yields a transient fixed source problem (TFSP). In the formulation of the linear system for a TFSP, the nodal coupling coefficients determined at a prior time point are used. The linear system is updated and solved again if a nodal update is required at the current time step. The need for a nodal update is determined by the conditional nodal update scheme which monitors the change in local cross sections. In this scheme, the nodal update is invoked only when the relative cross section change from the last nodal update is greater than a specified criterion. With regards to the planar cross section evaluation, there is no difference between steady-state and transient solution sequences. It should be noted, however, that the cross section evaluation. as well as the T-H calculation, is performed only once per time step. In the quasi-static mode, linear changes are to be made in the radial shapes if at least two shapes are available at different time points.

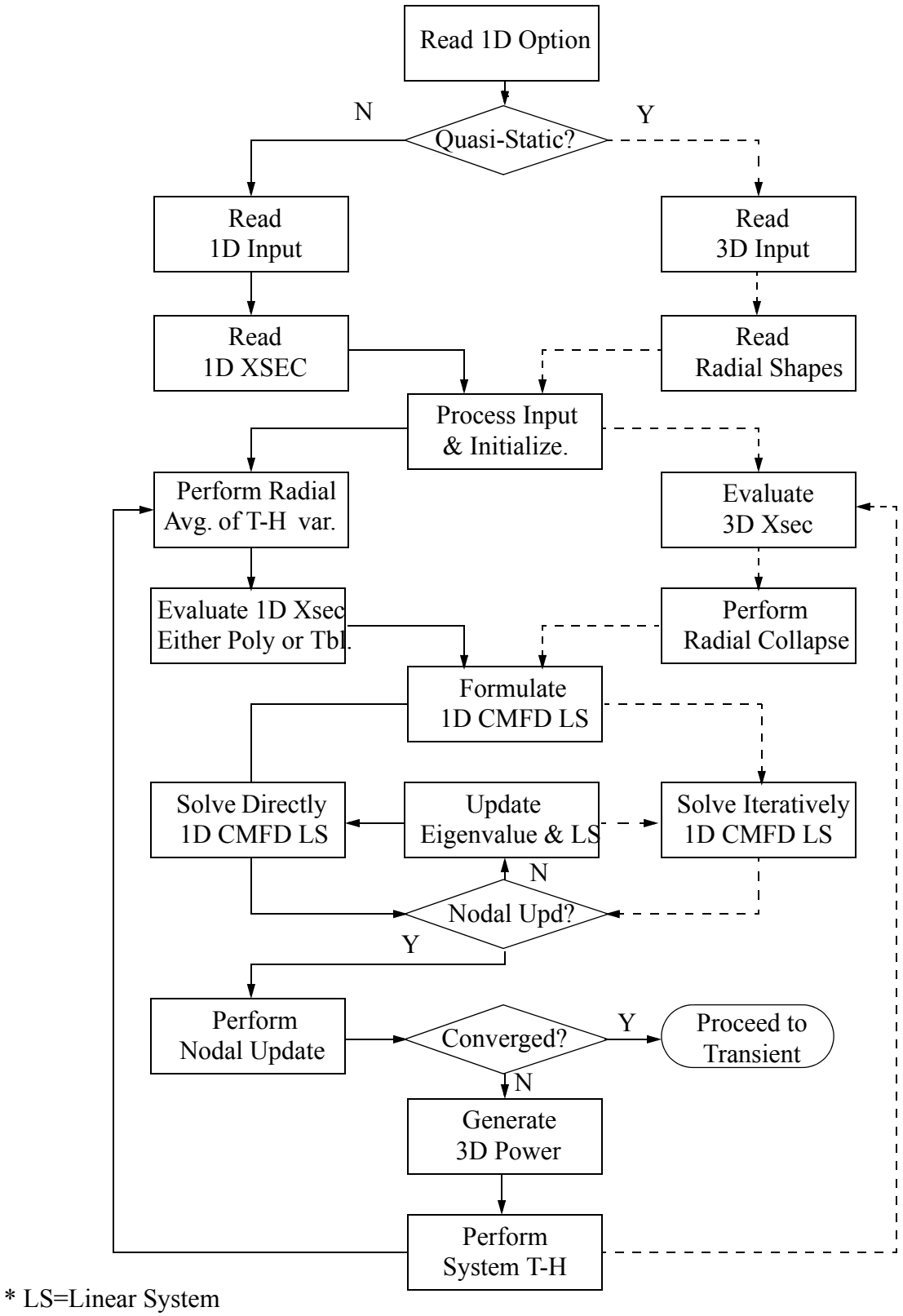


Figure 9.3: Steady-State 1-D Solution Sequence

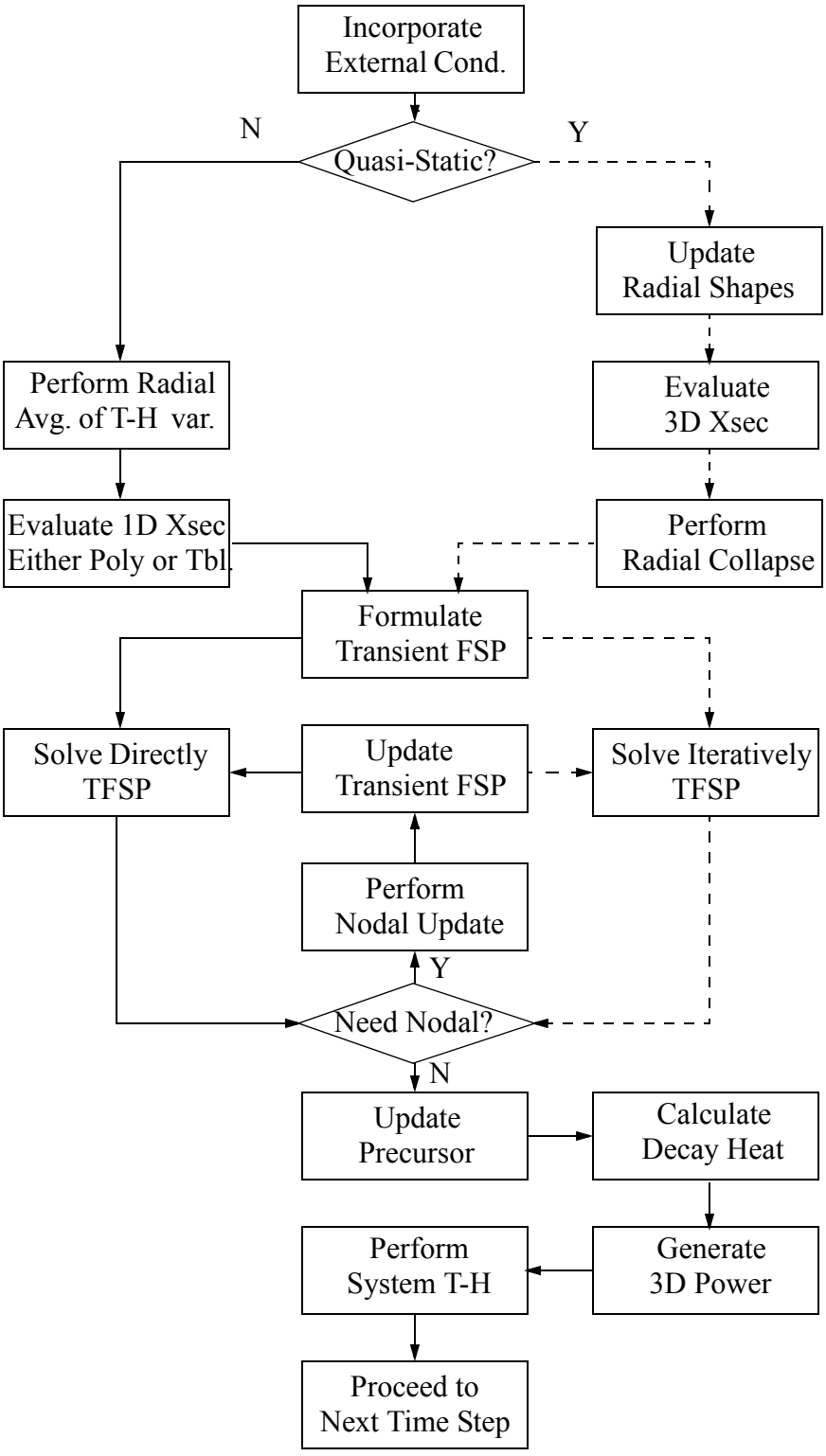


Figure 9.4: Transient 1-D Solution Sequence

9.8 Point Kinetics

Methods were implemented in PARCS to determine the point kinetics parameters for exact point kinetics and the various approximations of the exact point kinetics equations. This section will describe the basis for the methods implemented in PARCS, beginning with the formulation of the exact point kinetics equations which will reproduce the same solution as the PARCS spatial kinetics solution.

9.8.1 Derivation of Exact Point Kinetics Equations

The 3D time-dependent multi-group neutron diffusion equation and precusors equations can be written in standard notation as:

$$\frac{1}{v_g(\mathbf{r})}\frac{\partial \phi_g(\mathbf{r},t)}{\partial t} = \nabla \bullet D_g(\mathbf{r},t)\nabla \phi_g(\mathbf{r},t) + \sum_{g'} \Sigma_{g,g'}(\mathbf{r},t)\phi_{g'}(\mathbf{r},t) - \Sigma_{tg}(\mathbf{r},t)\phi_g(\mathbf{r},t) + \chi_g(\mathbf{r})S^F(\mathbf{r},t) \tag{9.59}$$

$$+ \sum_k \chi_{dk,g}(\mathbf{r})\left(\lambda_k(\mathbf{r})C_k(\mathbf{r},t) - \beta_k(\mathbf{r})S^F(\mathbf{r},t)\right), \quad g = 1,2...G$$

$$\frac{\partial C_k(\mathbf{r},t)}{\partial t} = \beta_k(\mathbf{r})S^F(\mathbf{r},t) - \lambda_k(\mathbf{r})C_k(\mathbf{r},t), \qquad k = 1,2...N_d \tag{9.60}$$

where S^F is fission source,

$$S^F(\mathbf{r},t) = \frac{1}{k_{eff}^s}\sum_{g'} \nu\Sigma_{f,g'}(\mathbf{r},t)\phi_{g'}(\mathbf{r},t), \tag{9.61}$$

and χ_g is average fission spectrum.

$$\chi_g = \chi_{p,g} + \sum_k \beta_k(\chi_{dk,g} - \chi_{p,g}), \tag{9.62}$$

Before integrating Eq.(9.59) and Eq.(9.60) over the space and energy domain, the flux is factorized into two independent functions, which are defined as the magitude function (p) and shape function (ψ), respectively, as follows:

$$\phi_g(r,\ t) = p(t)\psi_g(r,\ t) \quad . \tag{9.63}$$

Because Eq.(9.63) is an arbitrary factorization, it is necessary to impose the following constraint on the shape function which makes the factorization unique:

$$\sum_g \iiint_V \phi_g^*(\mathbf{r})\frac{1}{v_g(\mathbf{r})}\psi_g(\mathbf{r},t)d\mathbf{r} = \text{constant} \tag{9.64}$$

where ϕ_g^* is the adjoint flux from the steady state adjoint calcualtion. The notation can be simplified by defining a simple notation for integrating over the spacial and energy domain as,

$$< f_g(\mathbf{r},t) > \equiv \sum_g \iiint_V f_g(\mathbf{r},t) d\mathbf{r} \tag{9.65}$$

Substituting equaiton (9.63) into equation (9.59) yields

$$\begin{aligned} \frac{1}{v_g(\mathbf{r})}\left(\psi_g(\mathbf{r},t)\frac{dp(t)}{dt} + p(t)\frac{\partial \psi_g(\mathbf{r},t)}{\partial t}\right) &= p(t)\left[\nabla \bullet D_g(\mathbf{r},t)\nabla\psi_g(\mathbf{r},t) + \sum_{g'}\Sigma_{g,g'}(\mathbf{r},t)\psi_{g'}(\mathbf{r},t) - \Sigma_{tg}(\mathbf{r},t)\psi_g(\mathbf{r},t)\right] \\ &+ \chi_g(\mathbf{r})S^F(\mathbf{r},t) + \sum_k \chi_{dk,g}(\mathbf{r})\left(\lambda_k(\mathbf{r})C_k(\mathbf{r},t) - \beta_k(\mathbf{r})S^F(\mathbf{r},t)\right) \end{aligned} \tag{9.66}$$

Multiplying Eq. (9.66) by adjoint flux and integrating over the space and energy domain yields:

$$\Lambda_0 F_0 \frac{dp(t)}{dt} = \rho(t)p(t)F(t) - \beta^{eff}(t)p(t)F(t) + \sum_k \lambda_k(t)\zeta_k(t)F_0 \tag{9.67}$$

where $F(t)$ is adjoint weighted quasi-stationary fission source,

$$F(t) = < \phi_g^*(\mathbf{r})\chi_g(\mathbf{r})\hat{S}^F(\mathbf{r},t) >, \tag{9.68}$$

and

$$\hat{S}^F(\mathbf{r},t) = \frac{S^F(\mathbf{r},t)}{p(t)} = \frac{1}{k_{eff}^s}\sum_{g'} \nu\Sigma_{f,g'}(\mathbf{r},t)\psi_{g'}(\mathbf{r},t), \tag{9.69}$$

The neutron generation time is defined as:

$$\Lambda(t) = \frac{< \phi_g^*(\mathbf{r})\dfrac{1}{v_g(\mathbf{r})}\psi_g(\mathbf{r},t) >}{F(\mathbf{r},t)} \tag{9.70}$$

where F_0 and Λ_0 are $F(t)$ and $\Lambda(t)$ at time zero, respectivly.

The reduced precursor concentration is defined as:

$$\zeta_k(t) = \frac{< \phi_g^*(\mathbf{r})\chi_{dk,g}(\mathbf{r})C_k(\mathbf{r},t) >}{F_0}, \tag{9.71}$$

The aversage decay constant of precusor k is given as:

$$\lambda_k(t) = \frac{<\phi_g^*(\mathbf{r})\lambda_k(\mathbf{r})\chi_{dk,g}(\mathbf{r})C_k(\mathbf{r},t)>}{\zeta_k(t)F_0}, \tag{9.72}$$

$$\beta_k^{eff}(t) = \frac{<\phi_g^*(\mathbf{r})\chi_{dk,g}(\mathbf{r})\beta_k(\mathbf{r})\hat{S}^F(\mathbf{r},t)>,}{F(t)} \tag{9.73}$$

and

$$\beta^{eff}(t) = \sum_k \beta_k^{eff}(t) \tag{9.74}$$

The reacitivity is defined as:

$$\rho(t) = \frac{1}{F(t)} <\phi_g^*(\mathbf{r})\left(M_g - F_g\right)\psi(\mathbf{r},t)> \tag{9.75}$$

where:

$$M_g\left[\psi(\mathbf{r},t)\right] = \nabla\bullet\left(D_g\nabla\psi_g(\mathbf{r},t)\right) + \sum_{g'}\Sigma_{g,g'}(\mathbf{r},t)\psi_{g'}(\mathbf{r},t), \qquad F_g\left[\psi(\mathbf{r},t)\right] = \chi_g(\mathbf{r})\hat{S}^F(\mathbf{r},t) \tag{9.76}$$

Multiply equation (9.60) by adjoint flux and $\chi_{dk,g}$, then integrate over the space and energy domain, we obtained the reduced precursor equation:

$$F_0\frac{d\zeta_k(t)}{dt} = \beta_k^{eff}(t)p(t)F(t) - \lambda_k(t)\zeta_k(t)F_0 \tag{9.77}$$

Equations (9.67) and (9.77) can be rewritten as standard exact point kinetic equations,

$$\frac{dp(t)}{dt} = \frac{\rho(t)-\beta^{eff}(t)}{\Lambda(t)}p(t) + \frac{1}{\Lambda_0}\sum_k \lambda_k(t)\zeta_k(t) \tag{9.78}$$

and

$$\frac{d\zeta_k(t)}{dt} = \frac{\Lambda_0}{\Lambda(t)}\beta_k^{eff}(t)p(t) - \lambda_k(t)\zeta_k(t),\ k = 1,2...N_d \tag{9.79}$$

If the time dependent shape functions are available, the exact point kinetic equations (9.78) and (9.79) can reproduce same core fission power as spatial kinetics predicted.

$$p_f(t) = p_f(0)\frac{p(t) < \kappa\Sigma_{fg}(r,t)\psi_g(r,t) >}{p(0) < \kappa\Sigma_{fg}(r,0)\psi_g(r,0) >}, \tag{9.80}$$

The time dependent shape functions can be obtained from spatial kinetics calculation, or determined from solving following equations (9.81) and (9.82) for shape function.

$$\frac{1}{v_g(\mathbf{r})}\frac{\partial\psi_g(\mathbf{r},t)}{\partial t} = \nabla \bullet D_g(\mathbf{r},t)\nabla\psi_g(\mathbf{r},t) + \sum_{g'}\Sigma_{g,g'}(\mathbf{r},t)\psi_{g'}(\mathbf{r},t) - \left(\Sigma_{tg}(\mathbf{r},t) + \frac{1}{p(t)}\frac{dp(t)}{dt}\right)\psi_g(\mathbf{r},t) \tag{9.81}$$

$$+ \chi_g(\mathbf{r})\hat{S}^F(\mathbf{r},t) + \sum_k \chi_{dk,g}(\mathbf{r})\left(\lambda_k(\mathbf{r})\hat{C}_k(\mathbf{r},t) - \beta_k(\mathbf{r})\hat{S}^F(\mathbf{r},t)\right), \qquad g = 1,2...G$$

$$\frac{\partial\hat{C}_k(\mathbf{r},t)}{\partial t} = \beta_k(\mathbf{r})\hat{S}^F(\mathbf{r},t) - \left(\lambda_k(\mathbf{r}) + \frac{1}{p(t)}\frac{dp(t)}{dt}\right)\hat{C}_k(\mathbf{r},t), \qquad k = 1,2...N_d \tag{9.82}$$

where

$$\hat{C}_k(\mathbf{r},t) = \frac{C_k(\mathbf{r},t)}{p(t)} \tag{9.83}$$

Equations (9.81) and (9.82) are almost same as spatial kinetics equations (9.59) and (9.60) except the terms with time derivative of magnitude function. As shape function changes slower than flux distribution, larger time steps can be applied to equations (9.81) and (9.82) than the time steps which required to solve origional spatial kinetic equaitons (9.59) and (9.60).

9.8.2 Conventional Point Kinetics

In the conventional point kinetics formulation, the time dependent shape functions are not evaluated and the initial shape function is used as an approximation of time dependent shape fucntions throughout the transient. This approximation leads to conventional point kinetics equations formulation:

$$\frac{dp(t)}{dt} = \frac{\rho(t) - \beta^{eff}}{\Lambda}p(t) + \frac{1}{\Lambda}\sum_k \lambda_k\zeta_k(t) \tag{9.84}$$

and

$$\frac{d\zeta_k(t)}{dt} = \beta_k^{eff} p(t) - \lambda_k \zeta_k(t),\ k = 1, 2 \ldots N_d \tag{9.85}$$

The initial value of the generation time and the precursor decay contants are used through out the transient.

The shape function approximation is the major approximation of convential point kinetics. The additional approximation in conventional point kinetics is related to the reacitivity evaluation. Equation (9.86) is often used for evaluating reacitivity in conventional point kinetics,

$$\rho(t) = \rho_{cr}(t) + \sum_i \frac{\partial \rho}{\partial x_i}\left(x_i(t) - x_i(0)\right) \tag{9.86}$$

where ρ_{cr} is the control rod reactivity which is pre-evaluted from steady-state solutions with different control rod positions, x_i are core average parameters, such as coolant density, fuel temperature, soluble boron concentration, etc. The reactivity coefficients are often evaluated with uniform perturbed parameters from the steady state value.

$$\frac{\partial \rho}{\partial x_i} = \frac{1}{F(0)\Delta x_i} < \phi_g^*(\mathbf{r})\left(M_g(\bar{x}_i + \Delta x_i) - F_g(\bar{x}_i + \Delta x_i)\right)\psi(\mathbf{r}, t) > \tag{9.87}$$

where $\bar{x}_i$ is steady state distribution of parameter, and Δx_i is an arbitary uniform perturbation. It should be noted that the reacitivity coefficients evaluated with this method generally can not accuratelyrepresent the nonlinear effect of parameters over a wide range.

During the transient calculation, the core average parameters are often computed with weight which proportional to square of power if the adjoint function is not available:

$$x_i(t) = \frac{\iiint_V \bar{x}_i(\mathbf{r}, t) S^2(\mathbf{r}, 0) d\mathbf{r}}{\iiint_V S^2(\mathbf{r}, 0) d\mathbf{r}} \tag{9.88}$$

Additional point kinetics options were introduced in PARCS in order to provide the analyst with a means to quantify the error introduced by the three major approximations: 1) approximating time dependent shape functions with the initial shape function, 2) evaluating reacitivity with core average paremeters and precomputed reacivity coefficients, 3) core average parameters with square of power weighting.

$$\rho(t) = \frac{1}{F(0)} < \phi_g^*(\mathbf{r}) \left(M_g(t) - F_g(t) \right) \psi(\mathbf{r}, 0) > \tag{9.89}$$

The first option is to evaluate reacitivity with the time dependent cross section instead of precomputed reacitivity coefficients.

This option reduces the error in the point kinetics even when the time dependent shape functions are not available. The difference between this option and exact point kinetics is solely caused by approximating time dependent shape functions with initial shape function. If there is are not large variations in the flux shape in transient, this option will provide accurate transient results.

In the second option, the reactivity is evaluated with core average paremeters and precomputed reacivity coefficients, but the core average parameters are calculated with adjoint weighting:

$$x_i(t) = \frac{< \phi_g^*(\mathbf{r}) \vec{x}_i(\mathbf{r}, t) \chi_g \left(\mathbf{r} \right) \widehat{S}^F(\mathbf{r}, 0) >}{F(0)} \tag{9.90}$$

The difference between this option and conventional point kinetics is averaging the parameters with or without adjoint weighting. And the difference between this option and first option shows the error introduced by evaluating reacitivity with core average paremeters and precomputed reacivity coefficients.

9.8.3 Solution method for Point Kinetics

The same solution methods are used to solve the exact and the conventional point kinetics equations. The precursors are analytically integrated with the assumption of quadratic function variation during the integration time step. The integration equations are the same as these shown in Chapter 2 for the solution of the spatial kinetics equations. The theta method and exponential tranformation which are also described in Chapter 2 are also applied to amplitude equation.

10. CALCULATION CONTROL LOGIC

In the previous Chapters the individual computational methods required to perform various calculations using the PARCS code have been presented. These methods were incorporated into PARCS in a modular form and in order to solve a spatial kinetics problem the computational modules should be properly coordinated during the solution process. This Chapter describes the coordination and control logic of the methods as well as several special functions available to make the solution process more efficient.

Although transient problems can be initiated by a fixed source calculation (e.g. simulation of start up tests), transient calculations are most often preceded by a steady-state calculation to establish the initial conditions for the transient. The most distinct aspect of a steady-state calculation is the logic to calculate the steady-state eigenvalue. In section 2.1, an overview was provided of the fission source iteration and the Wielandt Shift acceleration method used to perform the eigenvalue calculation. The following section provides a more detailed description of the the eigenvalue calculation control logic and is followed by the transient calculation control method in Section 10.2.

10.1 Eigenvalue Calculation Control

An eigenvalue calculation involves a nested outer/inner iteration. Two nonlinear updates, nodal and T/H, are performed in the outer iteration which is accelerated by the Wielandt shift method. As noted in Chapter 2, the inner iteration consists of a source problem, Eq. (2.4), which is solved in PARCS by the Krylov subspace method. The control logic of the eigenvalue calculation that involves the inner, outer, and the nodal and T/H updates is depicted in Figure 10.1. The functions of each module and the conditionals which are represented by a box and by a diamond, respectively, are described in detail in the following subsections.

10.1.1 Initialization

After the input processing, a steady-state T/H calculation is performed to initialize the state variables. The relative power distribution to be used in the T/H calculation is obtained either by using an input axial flux shape or a default chopped cosine shape. After the T/H calculation, the nodal cross sections are initialized to the input power level and the fission source shape is obtained using the nodal cross sections and the input flux shape. Other initializations include setting the eigenvalue to unity, the CNCCs to zero, and the decay heat parameters and the Xe/Sm concentrations. All the counters and control flags are also properly initialized.

10.1.2 Calculate FDM Nodal Coupling Coefficients

The **F**inite difference **N**odal **C**oupling **C**oefficients (FNCC) are calculated using the nodal diffusion coefficients according to Eq. (3.2) for every nodal interface. The boundary conditions are also incorporated when calculating the FNCC at the external boundaries of the problem domain. The following general relation is used to calculate the FNCC at the boundaries:

$$\tilde{D}_g = \frac{2\alpha D_g}{2D_g + \alpha h} \tag{10.1}$$

where the albedo α is given as follows:

$$\alpha = \begin{cases} 0 & if \quad J = 0 \\ 10^{30} & if \quad \phi = 0 \\ 0.5 & if \quad J_{in} = 0 \end{cases} \tag{10.2}$$

The FNCC are recomputed whenever the nodal cross sections are updated due to T/H feedback.

10.1.3 Setup CMFD Linear System

Once the nodal cross sections and nodal coupling coefficients are determined, the CMFD linear system can be constructed using Eqs. (3.9) and (3.10). In the case of Wielandt shift, the reciprocal eigenvalue, λ, appearing in Eq. (3.9) is set to $\frac{1}{k_s}$.

10.1.4 Build Preconditioner

The BILU preconditioners are then constructed for each plane to be used in BILU3D preconditioning as described in Section 3.2.

10.1.5 Initialize BiCGSTAB and Construct Source

The source vector (s) appearing on the RHS of Eq. (2.6) is then constructed. The initial residual vector (r_0^i), which provides the starting point of the subsequent Krylov subspaces, is initialized using the source and the most recent nodal flux vector as:

$$r_0^n = s_n - A_n \phi_{n-1} \tag{10.3}$$

where *n* is the outer iteration index. The scalar and vector variables are also initialized according to the BiCGSTAB algorithm given in Section 3.2.1.

10.1.6 Perform a BiCGSTAB Iteration

One iteration step of the BiCGSTAB algorithm is then performed. This involves two solutions of the preconditioner equation as well as four inner products and two matrix vector products. At the end of the calculation, the flux and the residual vector is updated.

10.1.7 Inner Convergence

The convergence of the inner iteration is checked using the following conditional on the residual reduction factor:

$$\sigma \equiv \frac{\left\| r_i^n \right\|_2}{\left\| r_0^n \right\|_2} < \varepsilon_{erf} \quad . \tag{10.4}$$

The criterion on the error reduction factor is set in the input and a value from 0.005 to 0.04 is recommended with a typical value of 0.01. The inner iteration is also considered converged if the number of the inner iterations are over the specified limit (N_{in}). It has been observed that one inner iteration is sufficient ($N_{in} = 1$) in most practical calculations regardless of the fission source acceleration option. This is due to the substantial error reduction achieved with the BiCGSTAB algorithm. In the case of an irregular convergence, it is recommended that N_{in} is set to a larger value, e.g. 3 or 5.

10.1.8 Perform Wielandt Shift

After finishing the inner iteration, the expression in Eq. (2.14) is evaluated in the Wielandt option to update the eigenvalue. The new value of the shifted eigenvalue is then used to modify the diagonal elements of the linear system representing the LHS of Eq. (2.6). A multiplier to the fission source which is used to construct the source of the RHS of Eq. (2.6) is also updated using the shifted eigenvalue.

10.1.9 Conditional on Nodal Update

The nodal update is normally invoked with the same frequency as the T/H updates unless the nodal option is not turned on. However, the nodal update can be performed less frequently than the T/H depending upon the input parameter N_{Multi_TH}. The nodal update frequency is determined by $N_{nod} = N_{upd} \times N_{Multi\_TH}$ and the following conditional should be satisfied to invoke the nodal update:

$$mod(n, N_{upd}) = 0 \quad ? \tag{10.5}$$

At the later stages of the outer iteration, the nodal update can be invoked without satisfying the above conditional if all the convergence tests are satisfied. The recommended value of N_{Multi_TH} is one, however a larger value might lead to a faster convergence if the T/H feedback is dominant and the core is less heterogeneous, e.g. a Hot-Full-Power (HFP) , All-Rod-Out (ARO) problem.

10.1.10 Perform Nodal Update

The manner of performing the nodal update depends on the choice of solution kernel. When the two-node ANM kernel presented in Section 4.2 is used, then it is applied to every interior nodal interface in the problem domain. There are roughly $3 \times M$ (M being the total number of nodes) two-node problems to be solved since the one-node problems are also solved for the external boundaries. Before solving the one- or two-node problems, the node-wise ANM calculation parameters such as $_{\infty}$, κ, μ, r and are computed for each of the nodes. The net current at every interface is then calculated and used to determine the transverse leakages. The transverse leakages are then fitted into a quadratic polynomial for each direction at every node.

As discussed in Section 4.2, once the local nodal parameters are calculated, the nodal k_∞ and the core k_{eff} are compared to decide which nodal kernel to use for the two-node problem. For each node of interest:

$$\delta \equiv \left| 1 - \frac{k_\infty}{k_{eff}} \right| < \varepsilon_k \tag{10.6}$$

If this condition is satisfied then NEM is used, otherwise the standard ANM kernel is employed.

The nodal update method for the other solution kernels is similar.

10.1.11 Critical Boron Concentration Search

If the user chooses to perform a critical boron concentration search, the boron concentration is then updated only if a two-node or thermal-hydraulic update is performed. If a CMFD-only calculation is being performed the CBC search is turned off.

10.1.12 Conditional on T/H Update

The T/H update is invoked periodically unless the feedback option is turned off or the fuel temperature convergence is satisfied. The periodic update is performed every N_{TH} outer iterations by evaluating:

$$mod(n, N_{TH}) = 0 \tag{10.7}$$

where *mod* is the remainder function. N_{TH} is set equal to N_{upd} which is specified in the input as the cycle for nonlinear update. The T/H update is also invoked if all convergence parameters satisfy the specified criteria even if the condition in Eq. (10.7) is not satisfied. This occurs when the nonlinear iteration is almost converged at the later stages of the outer iteration.

10.1.13 Perform Steady-State T/H

When the T/H update is invoked, the power distribution is normalized and the relative nodal powers are calculated. The relative power distribution is then multiplied by the average power density that is calculated such that the specified power level can be achieved. The steady-state T/H calculation is performed first for coolant to determine the coolant bulk condition for the give heat source. Then the heat conduction calculation is performed to determine the fuel temperatures based on the coolant bulk condition.

10.1.14 Equilibrium Xenon/Samarium Calculation

The steady-state Xenon/Samarium number densities are calculated if the user chooses either the “equilibrium” or the “transient” Xenon/Samarium option. The update of these number densities is performed if either a two-node or thermal-hydraulic calculation is performed. Similar to

the CBC search, if a CMFD-only calculation is being performed the Xenon/Samarium option is turned off.

10.1.15 Update Nodal Macroscopic Cross Sections

After the T/H state parameters consisting of coolant density, coolant temperature, and fuel temperature are updated, the nodal macroscopic cross sections are updated. The functional dependencies of each type of cross section on the T/H state parameter is specified in the input.

10.1.16 Global Convergence Check

The convergence of the global nonlinear iteration for the eigenvalue calculation is determined based on the following parameters:

$$\delta_k = \left| k_{eff}^{n+1} - k_{eff}^{n} \right| \quad , \quad \delta_{L2} = \frac{\left\| \psi_{n+1} - \psi_n \right\|_2}{\left\langle \psi_{n+1}, \psi_n \right\rangle^2} \quad , \tag{10.8}$$

$$\delta_{L\infty} = max \left| \frac{\psi_{n+1}^{m} - \psi_n^{m}}{\psi_{n+1}^{m}} \right| \quad , \quad \delta_{Dop} = max \left| \frac{T_{D,\,t+1}^{m} - T_{D,\,t}^{m}}{T_{D,\,t+1}^{m}} \right|$$

where $T_{D,\,t}^{m}$ is the Doppler temperature at node *m* and T/H update index *t*. The convergence check is performed only after the first CMFD solution is obtained after a nodal update to ensure the convergence of the nonlinear iteration. All of the four convergence parameters should satisfy the respective convergence criterion to terminate the outer iteration. The four convergence criteria are specified in the input and the recommended values are $\varepsilon_k = 5 \times 10^{-6}$, $\varepsilon_{L2} = 5 \times 10^{-5}$, $\varepsilon_{L\infty} = 5 \times 10^{-4}$, and $\varepsilon_{Dop} = 10^{-3}$, respectively.

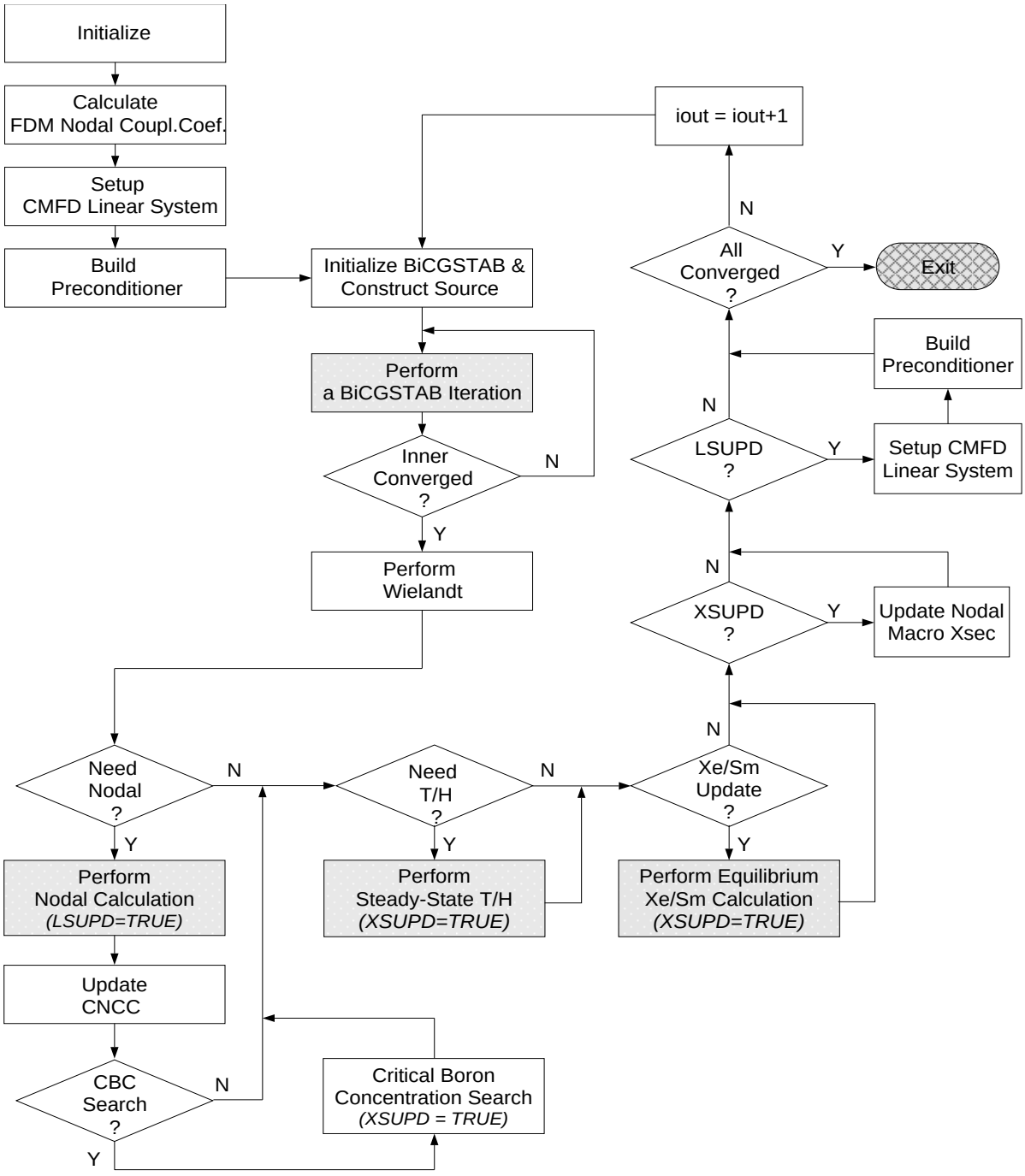


Figure 10.1: Eigenvalue Calculation Flow

10.2 Transient Calculation Control

Upon completion of the steady-state calculation, the core has been initialized and all conditions have been established for the transient calculation. These conditions include the k_{eff}, flux

distribution, T/H state variables, corrective nodal coupling coefficients (CNCC), Xe/Sm concentrations, Decay Heat levels, and the precursor densities. The transient calculation begins by constructing the transient fixed source and solving the transient fixed source problem (TFSP) at the first time point, which is then repeated at each subsequent time step.

As described in Section 2.2, a TFSP is formulated by employing the analytic precursor integration, the theta method temporal differencing, and the CMFD spatial differencing. The solution of a TFSP is then performed by a nonlinear iteration which involves the solution of CMFD problems and two-node problems. The nonlinear iteration, however, can be avoided in most of the transient time steps since the CNCCs can be updated only occasionally during a transient calculation by employing a conditional nodal update scheme. This considerably reduces the computational time since only the CMFD calculation needs be performe at most time steps. The feasibility of a conditional nodal update scheme was confirmed by an observation that the CNCC remains essentially unchanged from the prior value unless there is strong local perturbation introduced in the current time step. In the following subsection, an efficient scheme of conditional nodal update is presented which is intended to reduce the transient calculation time by not performing unnecessary nodal calculations.

In almost all transients, T/H feedback effects play an important role in determining the overall transient behavior. In the transient calculation, the interaction between neutronics and T/H field variables are treated in a slightly different way than the steady-state calculation. The transient fixed source problem formulated by the theta method requires that the coefficient matrix be formed at the end of a time step. This necessitates the evaluation of nodal cross sections at the end of the time step, which in turn requires the T/H calculation at the end of the time step. However, since a flux solution is not yet available at the end of the time step, some sort of extrapolation is necessary for the estimate of the power distribution to be used in T/H calculation. Also, to achieve convergence of both the neutronics and T/H field variables, a nonlinear iterative procedure is employed which involves alternating the neutronics and T/H calculations. In Subsection 10.2.2, the coordination of the neutronics and T/H transient calculations is described.

In the application of iterative solution methods, the efficiency of the calculation can depend on the method of determining solution convergence. Historically, convergence of transient neutronics calculations was determined by comparing the solutions of two successive iterates. In order to estimate the true error, however, additional information was required that dictates the asymptotic error behavior. For classical iterative methods, the spectral radius of the iteration matrix is one such information. In practice, however, such additional information is not usually used since it was very expensive to compute. In the Krylov subspace methods, a simple relation dictating the asymptotic error behavior is not readily available. Instead, during each iteration the residual becomes available as part of the Krylov algorithm and can be used as an absolute measure of error. In Subsection 10.2.3, the method for checking convergence during the transient calculation is described and then the overall transient calculation flow is explained in Subsection 10.2.4.

10.2.1 Nodal Coupling Coefficient Update Strategy

The corrective nodal coupling coefficients (CNCC) depend on various core characteristics which include the core size, fuel loading pattern, control rod configuration, power level, etc. Once

determined for the initial steady-state, the variation in the CNCCs is usually small during the transient unless there is a significant change in core characteristics such as the movement of control rods. Therefore, the computation time can be reduced by only reevaluating the CNCCs when a significant change is introduced during the transient.

The success of this strategy depends on a successful conditional nodal update scheme. Since the local cross section change is known prior to the solution of a transient fixed source problem (TFSP), it is the logical candidate to determine whether a nodal update is necessary. An analysis was performed to examine the cross section as the basis for conditional CNCC using the control rod ejection transient, which involves a strong local change in the cross sections.

A control rod ejection event from an initially HZP condition results in a severe change in both the magnitude and the shape of the neutron flux because the worth of the ejected control rod can be potentially large enough to cause a super-prompt critical event. Since the radial flux distribution changes significantly, resulting in a strong peak at the ejected rod location, this event can serve as a good test case to examine the change in the CNCC. In the following subsection, the NEACRP PWR benchmark problem Case C1[25] which is a peripheral control rod ejection at HZP is used as the test case to examine the changes in the corrective nodal coupling coefficients during the transient.

Assessment of Coupling Coefficient During a CEA

The relative changes of the corrective nodal coupling coefficients for the x-direction from their steady-state values at three different radial and axial locations are shown in Figure 10.2. The control rod is being ejected at a radial location on the x-axis corresponding to Fuel Assembly (FA) 16 and the rod ejection is completed at 0.1 seconds. FA 8 is located at the center of the core and FA 2 is at the opposite location of FA 16 on the x-axis. Thus the distance between FA 16 and FA 2 is two times the distance between FA 16 and FA 8. The planes in Figure 10.2 are numbered from the bottom of the core and Planes 6, 14, and 10 are located at the middle of the lower half, the middle of the upper half, and the middle of the entire core, respectively. The data plotted in Figure 10.2 were obtained from a case that solves the converged two-node problems at every time step with a time step size of 10 ms.

As seen in the plots for FA 14, the correctional nodal coupling coefficient changes significantly in the node in which the control rod is being removed. For Groups 1 and 2, it changes by about 70% and 175%, respectively, from the steady-state values which were obtained with the rod inserted in the node. Once the rod is removed, however, the corrective nodal coupling coefficient remains essentially constant. At the center of the core (FA 8), the change in the corrective nodal coupling coefficients is much smaller than at FA 16, and the change is larger for Group 1 (~4%) than for Group 2 (0.8%). At the location far away from the perturbed location (FA 2), the change is very small (~1.2% for Group 1, ~0.3% for Group 2). However, it should be noted that although the changes are small, relatively large changes occur during the period of rod movement (0 to 0.1 second). This reflects the characteristic of the elliptic transient fixed source problem: perturbations are propagated instantaneously through the system. As indicated in the Figure, the extent of influence is larger for Group 1 because of the larger mean free path of fast neutrons. As shown in the lower right plot, changes occur in the CCNCs for FAs 8 and 2 after a small plateau from 0.1 to 0.25 seconds. These are due to the Doppler feedback which as the result of the rise in fuel temperature after about 0.25 secs. However, the changes due to the Doppler effect are much smaller than the external perturbation (control rod motion in this case).

Conditional Update Algorithm

Based on the above observations and other studies, it was concluded that a nodal update should be performed for all the nodes if there is a significant local change in the cross section, either due to a material motion or T/H feedback. The following describes a conditional nodal update scheme based on cross section change.

(1) Find the maximum change in nodal removal (absorption+scattering) cross section when the nodal cross sections at the current time point are available after incorporating the external perturbation and estimating the T/H condition. Namely,

$$\Delta\Sigma_{max} = max\left|\frac{\Sigma_{rg}^{m} - \hat{\Sigma}_{rg}^{m}}{\hat{\Sigma}_{rg}^{m}}\right| \tag{10.9}$$

where $\hat{\Sigma}_{rg}^{m}$ is the removal cross section of Group *g* at Node *m* when the previous nodal update was performed.

(2) If the maximum change is greater than the specified criterion ($\Sigma_{max} > \varepsilon_{Xsec}$), turn the nodal flag on.

(3) If the nodal flag is on, perform at least one nodal update whether or not the convergence of the CMFD solution is achieved before testing whether to invoke a nodal update. Otherwise, perform the nodal update only when the CMFD solution does not converge in N_{nodal}^{0} iterations, which is specified in the input. Terminate in both cases if the CMFD solution is converged.

(4) Whenever a nodal update is done, update $\hat{\Sigma}_{rg}^{m}$ for use in later time steps.

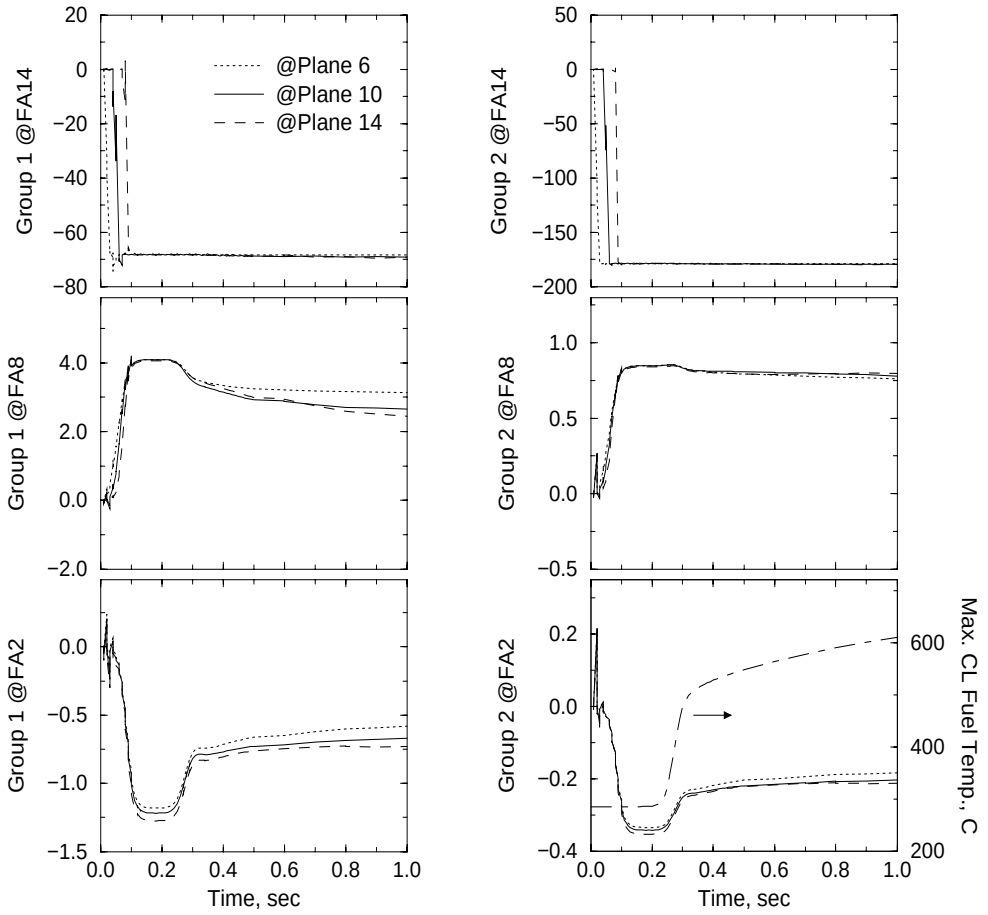


Figure 10.2: Assessment of Changes in the Corrective Nodal Coupling Coefficients

10.2.2 Transient T/H Calculation

In a LWR, there are two primary T/H feedback mechanisms that affect the neutronics solutions during a transient: the Doppler effect resulting from changes in the fuel temperature and the moderator/coolant effect resulting from changes in the water number density. The Doppler effect originating in the fuel is considered a prompt reactivity feedback, whereas the moderator/coolant density effect is considered delayed because there is a time delay on the order of seconds in the heat transfer from the fuel to the coolant. This impacts the numerical solution since the Doppler effect needs to be incorporated in the iteration process to resolve the nonlinearity in a problem involving T/H feedback.

Since the cross sections needed to formulate a TFSP should be obtained at the end of the time step, the transient heat conduction calculation should advance first to the end of the time step

to determine the change in the fuel temperature during the time step. This calculation requires an estimate of the power distribution at the end of the time step. Since the new power distribution is not known, the current power distribution is extrapolated based on the two most recent time step values.

The fuel temperature distribution obtained from the first heat conduction solution can now be used to calculate the nodal cross sections. It is also used to check convergence of the Doppler temperature by comparing the relative change to a user specified criterion:

$$\Delta T_d^{max} = max\left|\frac{T_d^m - \hat{T}_d^m}{T_d^m}\right| < \varepsilon_{Dop} \tag{10.10}$$

where $\hat{T}_d^m$ is the old Doppler temperature at Node *m* and is reset to the most recent value after the comparison is made. After the first heat conduction solution at each time step, the relative change represents the change from the previous time point. Later in the time step, it represents the change in the two consecutive estimates at the end of the time step. If the criterion in Eq. (10.10) is not satisfied, the heat conduction calculation will be performed after a partial convergence is achieved in the neutronics solution which will provide the new power distribution at the end of the time step. Otherwise, no more heat conduction calculations will be performed and the solution proceeds to achieve neutronics convergence.

After both the heat conduction and the neutronics converges, the heat conduction calculation is done once more with the converged flux distribution. The last heat conduction calculation determines the heat flux to be used in the subsequent heat convection calculation for the coolant. The heat convection solution determines the coolant condition at the end of the time step and the coolant condition will be used in the next time step calculation to update the cross sections and also to provide the boundary condition to the heat conduction calculations.

10.2.3 Convergence Checking

In solving a TFSP in which the right hand side is fixed, the convergence can be checked by examining relative magnitudes of the residual vector, which is defined as $r_i = s - A\phi_i$ for the *i-th* iterate of the solution.

A measure of the global error in the *i-th* iterate solution is provided using the 2-norm as:

$$\delta_2^i \equiv \frac{\|r_i\|_2}{\|s\|_2} \tag{10.11}$$

based on the magnitude of the effective source vector. Given a B-norm of a vector with $B=A^TA$ as $\|v\|_B = \langle v, Bv\rangle^{1/2} = \langle v, A^TAv\rangle^{1/2} = \|Av\|_2$, it can be shown that the second measure of error is the same as

$$\delta_2^i = \frac{\|\phi_i - \phi^*\|_B}{\|\phi^*\|_B} = \frac{\|A\phi_i - A\phi^*\|_2}{\|A\phi^*\|_2} \tag{10.12}$$

where ϕ^* is the exact solution. Since $A=M-F$ where *M* and *F* are is the loss and the fission operators, respectively, and $A\phi$ is the net loss rate (loss-fission), the above definition gives the relative

error in net loss (negative production) rates instead of fission rates. A measure of local error can be defined in a similar way using the infinite norm as:

$$\delta_{\infty}^{i} \equiv max\left|\frac{r_i^m}{s_i^m}\right| \quad . \tag{10.13}$$

Using the measures of error defined by Eq. (10.12) and Eq. (10.13) has the advantage that it can represent a true error without estimating the asymptotic convergence behavior. It has been observed that the global convergence check is sufficient in most cases and the default value of the global relative residual convergence criterion (ε_{r2}) is 0.001.

10.2.4 Transient Calculation Flow

The transient calculation at a time point consists of three stages: formulation of a TFSP, solution of a TFSP, and update of the state variables. At the first stage, the nodal cross sections are estimated and the TFSP is formulated by employing the temporal differencing scheme. At the second stage, a TFSP is solved by an iterative procedure which involves the nodal and heat conduction calculations as well as the CMFD calculation. The last stage is to update both the delayed neutron and decay heat precursors and the T/H variables using the converged flux distribution. The three stages are depicted in Figure 10.3 which shows the overall transient calculation flow. In the following, the functions of the new modules are described which do not appear in the steady-state calculation, and which are not mentioned in the previous subsections.

The formulation of the TFSP begins by incorporating external perturbations (e.g. control rod motion and boron concentration change), changes in the thermal condition, and changes in the Xenon/Samarium number densities into the nodal cross sections. At this point, the linear system given in Eq. (3.5) is obtained by performing the temporal discretization procedure described in Section 2.2, which involves the calculation of the precursor integration parameters, Eq. (2.24), and the construction of the effective source, Eq. (2.26) and Eq. (2.32). The inverse period is evaluated at each node using the previous two flux solutions. The inverse period is used to obtain the initial guess of the flux by the exponential extrapolation and/or to perform the exponential transform if specified by the user. When finished, the primary output of this module is the effective source, prompt neutron fraction, theta differencing parameters (first two terms in the RHS of Eq. (2.31)), and the initial flux guess.

After convergence is achieved, the delayed neutron precursor density is updated using Eq. (2.23), and the decay heat precursor density is updated. The new precursor densities will then be used to compute the delayed neutron source and decay heat source at the next time step.

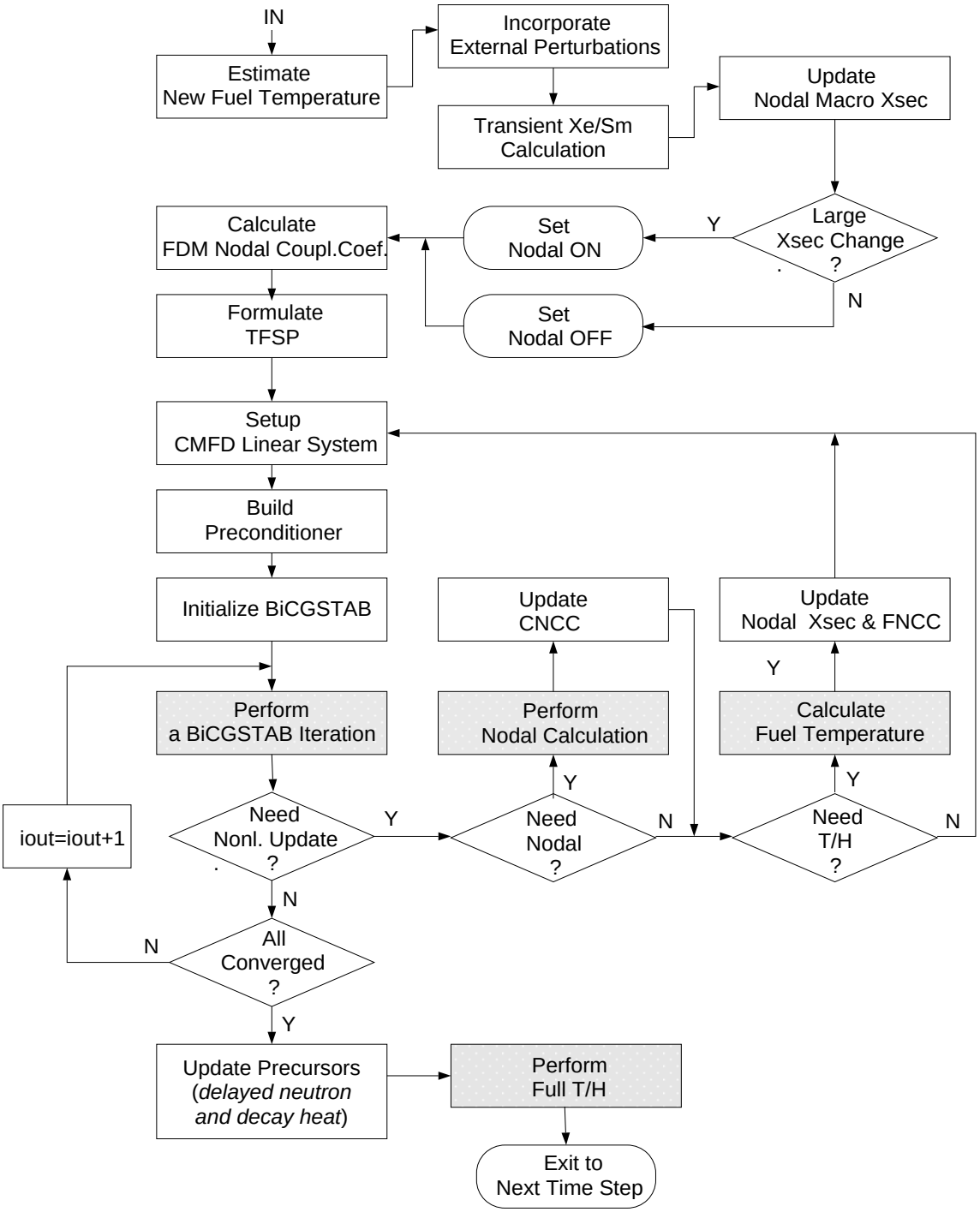


Figure 10.3: Transient Calculation Algorithm

11. REFERENCES

[1] K. Smith, "Nodal Method Storage Reduction by Nonlinear Iteration," *Trans. Am. Nucl. Soc.*, **44**, 265 (1983).

[2] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, 1996, *pp*. 144-229.

[3] H. Finnemann, F. Bennewitz, M. R. Wagner, "Interface Current Techniques for Multidimensional Reactor Calculations," *Atomkernenergie*, **30**, 123-128 (1977).

[4] K. Smith, "An Analytic Nodal Method for Solving the 2-Group, Multi-Dimensional, Static and Transient Neutron Diffusion Equations," *Nuc. Eng. Thesis*, Dept. of Nuc. Eng., MIT (1979).

[5] H. G. Joo, G. Jiang, and T. J. Downar, "A Hybrid ANM/NEM Interface Current Technique for the Nonlinear Nodal Calculation," *Proc. ANS Conf. Math. Comp.*, Saratoga, NY, Oct. 1997.

[6] Rempe, K., Smith, K., and Henry, A., "SIMULATE-3 Pin Power Reconstruction: Methodology and Benchmarking," *Proc. Int. Reac. Phys. Conf.*, Jackson Hole, Wyoming, Sep. 1988, Vol. III, p. 19 (1988).

[7] Boer, R and Finnemann, H, "Fast Analytical Flux Reconstruction Method for Nodal Space-Time Nuclear Reactor Analysis," *annl. nucl. Ener.*, **91**, 123 (1985).

[8] Brantley, P. and Larsen E., "The Simplified SP3 Approximation," *Nucl. Sci. Eng.*, **134**, 1-21 (2000).

[9] Cho, J.Y., et al., "Hexagonal CMFD Formulation Employing the Triangular Polynomical Expansion Method," Mathematics and Computation International ANS Meeting, Salt Lake City, September, 2001.

[10] P. J. Turinsky et. al., "NESTLE: A Few-Group Neutron Diffusion Equation Solver Utilizing the Nodal Expansion Method for Eigenvalue, Adjoint, Fixed-Source Steady-State and Transient Problems," EGG-NRE-11406 (1994).

[11] T. M. Sutton, "Wielandt Iteration as Applied to the Nodal Expansion Method," *Nucl. Sci. Eng.*, **98**, pp. 169-173 (1988).

[12] H. A. Van Der Vorst, "BI-CGSTAB: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems," SIAM J. Sci. Stat. Comput. 13, pp. 631-644 (1992).

[13] M. R. Hestenes and E. Stiefel, "Methods of Conjugate Gradients for Solving Linear Systems," *J. Res. Nat. Bur. Standards*, **49**, pp. 409-436 (1952).

[14] O. Axelsson, "Incomplete block matrix factorization preconditioning method. The ultimate answer?," *J. Comp. Appl. Math.* **12&13**, pp. 3-18 (1985).

[15] P. R. Engrand, G. I. Maldonado, R. Al-Chalabi and P. J. Turinsky, "Non-Linear Iteration Strategy for NEM: Refinement and Extension," *Trans. Am. Nucl. Soc.*, **65**, p. 221 (1992).

[16] H. G. Joo, G. Jiang, and T. J. Downar, "Stabilization Techniques for the Nonlinear Analytic Nodal Method," *Nucl. Sci. Eng.*, **130**, 1-13 (1998).

[17] Noh, J. M. and Cho, N. Z., "A New Approach of Analytic Basis Function Expansion to Neutron Diffusion Nodal Calculation," *Nucl. Sci. Eng.*, **116**, 165 (1994).

[18] Cho, J. et al., "Hexagonal CMFD for Triangle-Based Polynomial Expansion Method," ANS Topical Meeting: M&C2001, Salt Lake City, Utah, September, 2001.

[19] DeHart, M., et al., "TRITON: An Advanced Lattice Code for MOX Fuel Analysis," ANS Topical Meeting: Advances In Nuclear Fuel Management III, October, 2003.

[20] Giust, F. and Stammler, R., HELIOS v 1.8 User Guide and Manual, Studsvik/Scandpower, (2003).

[21] PARCS Website: https://engineering.purdue.edu/PARCS

[22] Smith, K., et. al., SIMULATE-3 Methodology, Studsvik/SOA-92-02, (1992).

[23] Umbarger, J. A., et. al., SIMULATE-3 User's Manual, Studsvik/SOA-92/01, (1992).

[24] Dunn, F., "Decay Heat Calculations for Transient Analysis," ANS Trans., 60, 633 (1989).

[25] Finnemann, H., "NEACRP 3D LWR Core Transient Benchmark - Final Specifications," NEACRP-L-335 (Revision 1), (1992).

[26] B. R. Bandini, A Three-Dimensional Transient Neutronics Routine for the TRAC-PF1 Reactor Thermal Hydraulic Computer Code, Ph. D. Dissertation, Pennsylvania State University, 1990.

[27] R. D. Lawrence, "Progress in Nodal Methods for the Solution of the Neutron Diffusion and Transport Equations," Progress in Nuclear Energy, 17, 3, 1986, pp. 271.

[28] K. S. Smith, "Nodal Method Storage Reduction By Non-Linear Iteration," Transactions of the American Nuclear Society, Detroit, MI, June 1983, 44, pp. 265.

[29] K. S. Smith, "QPANDA: An Advanced Nodal Method for LWR Analyses," Transactions of the American Nuclear Society, June 1985, pp. 265.

[30] K. S. Smith and K. R. Rempe, "Testing and Applications of the QPANDA Nodal Method," Proceedings of the International Topical Meeting on Advances in Reactor Physics, Pittsburgh, PA, 1987, 2, pp. 861.

[31] P. R. Engrand, G. I. Maldanado, R. Al-Chalabi, and P. J. Turinsky, "Non-Linear Iterative Strategy for NEM: Refinement and Extension," Transactions of the American Nuclear Society, Boston, MA, June 1992, 65, pp. 221.