FUNDAMENTALS AND RECENT DEVELOPMENTS OF REACTOR PHYSICS METHODS

NAM ZIN CHO
Korea Advanced Institute of Science and Technology
Department of Nuclear and Quantum Engineering
373-1 Guseong-dong, Yuseong-gu, Daejeon, Korea
E-mail : nzcho@kaist.ac.kr

As a key and core knowledge for the design of various types of nuclear reactors, the discipline of reactor physics has been advanced continually in the past six decades and has led to a very sophisticated fabric of analysis methods and computer codes in use today. Notwithstanding, the discipline faces interesting challenges from next-generation nuclear reactors and innovative new fuel designs in the coming.

After presenting a brief overview of important tasks and steps involved in the nuclear design and analysis of a reactor, this article focuses on the currently-used design and analysis methods, issues and limitations, and current activities to resolve them as follows:

(1) Derivation of the multigroup transport equations and the multigroup diffusion equations, with representative solution methods thereof.
(2) Elements of modern (now almost three decades old) diffusion nodal methods.
(3) Limitations of nodal methods such as transverse integration, flux reconstruction, and analysis of UO2-MOX mixed cores. Homogenization and related issues.
(4) Description of the analytic function expansion nodal (AFEN) method.
(5) Ongoing efforts for three-dimensional whole-core heterogeneous transport calculations and acceleration methods.
(6) Elements of spatial kinetics calculation methods and coupled neutronics and thermal-hydraulics transient analysis.
(7) Identification of future research and development areas in advanced reactors and Generation-IV reactors, in particular, in very high temperature gas reactor (VHTR) cores.

1. INTRODUCTION

Nuclear reactors are engineering devices in which controlled nuclear fission chain reactions are maintained and from which the produced nuclear energy is extracted for useful uses, such as generation of electricity. In such a device, neutrons induce nuclear fission reactions with heavy nuclei called nuclear fuel. The constituent materials of a reactor are generally fuel, coolant/ moderator, structural materials, and fission control material. In general, these materials are arranged very heterogeneously due to neutronics, thermal-hydraulics, and structural conside-rations. In addition, these structural arrangement and the constituents may change depending on the life-cycle of the fuel or on the operational mode of the reactor, including accident conditions.

Although the discipline of reactor physics that deals with the design and analysis of such reactors encompasses several areas in science and engineering, the reactor physics has matured on its own and established a unique field; and thus in particular, reactor analysis and methods development may be characterized as a discipline concerning determination and prediction of the states of a reactor that sustains chain reaction by balancing neutron production by fission and loss by capture and leakage. More specifically and summarily, we can say that the objective of the reactor analysis is to determine :

i) neutron multiplication factors for various configurations of a reactor,

and

ii) neutron flux distributions (hence, power distributions that are generated), spatial and temporal, under various operating (including accident) conditions.

Thus, the results of reactor design and analysis become the base or springboard to other activities necessary in the

realization of a nuclear power plant.

Figure 1.1 shows a two-dimensional schematic of a typical configuration of a power reactor core and fuel assembly. Figure 1.2 shows a typical calculational flow involved in reactor core design and analysis. Note that major methods currently in use consist of transport theory methods and diffusion theory methods. The methods based on transport theory are used in cell and assembly local calculations, while the methods based on diffusion theory are used in whole-core global calculations. Detailed calculational flows (e.g., the number of groups and methods in each phase) depend on the specific reactor vendors and the fuel designers who provide initial/reloaded fuel design services.

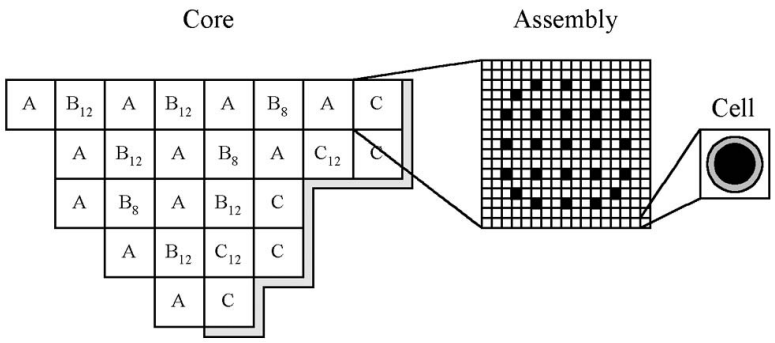


Fig 1.1 Typical configuration of 1/8 ractor core and fuel assembly.

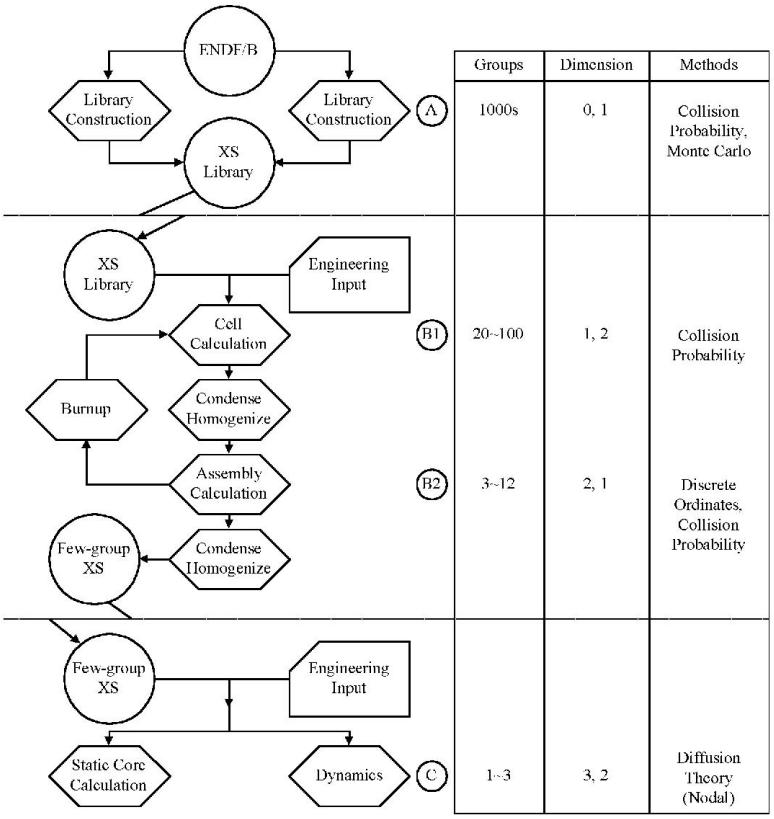


Fig 1.2 Calculational flow of reactor core design. (from [1])

The trend in recent years is to combine phase B1 and phase B2, or the cell calculation is done to condense the cross sections from multigroup to few-group, without homogenization. Another trend is that the method of characteristics (MOC) replaces the collision probability method (CPM) in these phases.

In the last few years, there has been a significant amount of work on coupled neutron kinetics and thermal-hydraulics transient analysis. Traditionally, the thermal-hydraulics and system transient analysis codes contain point kinetics model or at most one-dimensional kinetics model for core neutronics. The recent activities are to couple the state-of-the-art three-dimensional neutronics codes with the thermal-hydraulics system codes.

Also, there grows a strong interest in applying transport theory methods to three-dimensional whole-core heterogeneous calculation, eliminating diffusion theory methods. This amounts to combining phase B2 and phase C (at least static core calculation) under transport theory methods.

The topics covered in this review article on fundamentals and recent developments of reactor physics methods are obviously limited. They are neither comprehensive nor exhaustive. The topics selected certainly reflect the author's experience, interests and taste. However, the author tries to present tutorially the most fundamental principles and methods in the reactor analysis of today and attempts to provide recent developments in the topics selected by the author.

The article is organized as follows. Section 2 provides two fundamental governing equations in reactor physics and their multigroup approximations. Section 3 and Section 4 describe the basics of numerical methods for multigroup diffusion equations and multigroup transport equations, respectively. Section 5 provides fundamentals in three-dimensional reactor kinetics. Finally, Section 6 addresses challenging areas in reactor physics for future research and development.

2. GOVERNING EQUATIONS IN REACTOR PHYSICS

2.1 Starting Equations

One (perhaps the ultimate) objective of the reactor physics is to determine the distribution of the neutrons in a reactor. For that, we have to take into account the motion of the neutrons and their interactions with the host nuclei of various kinds. Thus, we need a mathematical model or theory to describe this particle transport phenomena.

2.1.1 Transport Equation

As a high-level model that describes the distribution of neutrons in a medium such as a reactor, we usually consider the following Boltzmann transport equation :

$$\begin{aligned}\frac{1}{\mathrm{v}}\frac{\partial}{\partial t}\varphi(\vec{r},E,\vec{\Omega},t)+\vec{\Omega}\cdot\nabla\varphi(\vec{r},E,\vec{\Omega},t)+\sigma_t(\vec{r},E)\varphi(\vec{r},E,\vec{\Omega},t)\\ =\int_{4\pi}d\vec{\Omega}'\int_0^\infty dE'\sigma_s(\vec{r},E'\to E,\vec{\Omega}'\cdot\vec{\Omega})\varphi(\vec{r},E',\vec{\Omega}',t)\\ +\frac{\chi(E)}{4\pi}\int_{4\pi}d\vec{\Omega}'\int_0^\infty dE'\ \nu\sigma_f(\vec{r},E')\varphi(\vec{r},E',\vec{\Omega}',t)+q_{ex}(\vec{r},E,\vec{\Omega},t)\end{aligned} \tag{2.1}$$

with appropriate initial and boundary conditions provided. In Eq.(2.1), the angular neutron flux φ is defined as

$$\varphi(\vec{r},E,\vec{\Omega},t)=\mathrm{v}n(\vec{r},E,\vec{\Omega},t) \tag{2.1a}$$

where $\mathrm{v}=\left(\frac{2E}{m}\right)^{1/2}$ and the angular neutron density n has the following meaning :

$$\begin{aligned}n(\vec{r},E,\vec{\Omega},t)\,d\vec{r}\,dE\,d\vec{\Omega}=\ &\text{expected number of neutrons in } d\vec{r},\ dE,\ d\vec{\Omega}\\ &\text{around the phase space point } \vec{r},\ E,\ \vec{\Omega} \text{ at time t.}\end{aligned} \tag{2.1b}$$

Other notations are standard, except that the lower case σ stands for macroscopic cross sections [2, 3].

In writing down Eq.(2.1), we have assumed that i) the medium is isotropic (e.g., the medium exhibits no polarization to neutrons), ii) the fission neutrons are emitted isotropically, and iii) all fission neutrons are emitted promptly (this can be relaxed by considering that some neutrons are delayed). It is also based on the assumption that the neutron is a point particle that is described classically by its position and velocity. The cross sections (the degrees of various reactions) are given by experimental data or by theoretical calculations, if experimental data are not available, with the help of quantum mechanics. Another article in this special issue of the journal deals specifically with nuclear data. The problem of finding solutions to Eq.(2.1) is nontrivial or defies elementary approaches of analytical methods, but requires sophisticated

numerical methods. This is due to i) the complicated energy and space-dependency of the cross sections, ii) the angular dependency of the scattering cross section $\sigma_s(\vec{\Omega}'\cdot\vec{\Omega})$, and iii) complexity due to the $\vec{\Omega}\cdot\nabla\varphi$ term, particularly in curvilinear coordinates.

It is customary to first represent the differential scattering cross section in Legendre components:

$$\sigma_s(\vec{r},E'\to E,\mu_0)=\sum_{l=0}^{\infty}\frac{2l+1}{4\pi}\sigma_{sl}(\vec{r},E'\to E)P_l(\mu_0)\ . \tag{2.2}$$

In the case of time-independent or steady-state situation, Eq.(2.1) becomes

$$\begin{aligned}&\vec{\Omega}\cdot\nabla\varphi(\vec{r},E,\vec{\Omega})+\sigma_t(\vec{r},E)\varphi(\vec{r},E,\vec{\Omega})\\&\quad=\int_{4\pi}d\vec{\Omega}'\int_0^{\infty}dE'\sum_{l=0}^{\infty}\frac{2l+1}{4\pi}\sigma_{sl}(\vec{r},E'\to E)P_l(\mu_0)\varphi(\vec{r},E',\vec{\Omega}')\\&\quad+\frac{\chi(E)}{4\pi}\int_{4\pi}d\vec{\Omega}'\int_0^{\infty}dE'\nu\sigma_f(\vec{r},E')\varphi(\vec{r},E',\vec{\Omega}')+q_{ex}(\vec{r},E,\vec{\Omega})\ .\end{aligned} \tag{2.3}$$

2.1.2 Diffusion Equation

Because the solutions to the transport equation (2.3) for realistic reactor problems are difficult to obtain, and also due to the fact that knowledge of the neutron density

$$N(\vec{r},E)=\int d\vec{\Omega}\ n(\vec{r},E,\vec{\Omega}) \tag{2.4}$$

or the neutron (scalar) flux

$$\phi(\vec{r},E)=\int d\vec{\Omega}\ \varphi(\vec{r},E,\vec{\Omega}) \tag{2.5}$$

is sufficient for most applications such as the fission rate distribution, one is interested in obtaining a governing equation with $N(\vec{r},E)$ or $\phi(\vec{r},E)$ as unknown. We begin by defining, in addition to Eq. (2.5), also neutron current density

$$\vec{J}(\vec{r},E)=\int_{4\pi}d\vec{\Omega}\ \vec{\Omega}\ \varphi(\vec{r},E,\vec{\Omega}), \tag{2.6}$$

and we will integrate Eq. (2.3) twice over the angular variable, resulting in two P_1 equations.

If we assume that the angular flux (which is the unknown) is only weakly dependent on the angle, i.e., linearly anisotropic (which would not be good in a highly absorbing medium or near the boundary or in a medium of rapid variation of cross sections):

$$P_1\ \text{Approximation}:\ \varphi(\vec{r},E,\vec{\Omega})=\frac{1}{4\pi}\phi(\vec{r},E)+\frac{3}{4\pi}\vec{\Omega}\cdot\vec{J}(\vec{r},E), \tag{2.7}$$

we can write formally as [4]

$$\vec{J}(\vec{r},E)=-D(\vec{r},E)\nabla\phi(\vec{r},E), \tag{2.8}$$

if we define the operator (called "diffusion coefficient"):

$$D(\vec{r},E)=\frac{1}{3}\left[\sigma_t(\vec{r},E)-\frac{\int_0^{\infty}dE'\sigma_{s1}(\vec{r},E'\to E)J_u(\vec{r},E')}{J_u(\vec{r},E)}\right]^{-1} \tag{2.9}$$

where $u=x,y,z$. Eq.(2.8) is called Fick's law. Finally, the two P_1 equations are combined to give

$$\begin{aligned}&-\nabla\cdot D(\vec{r},E)\nabla\phi(\vec{r},E)+\sigma_t(\vec{r},E)\phi(\vec{r},E)\\&\quad=\int_0^{\infty}dE'\sigma_{s0}(\vec{r},E'\to E)\phi(\vec{r},E')+\chi(E)\int_0^{\infty}dE'\ \nu\sigma_f(\vec{r},E')\phi(\vec{r},E')+Q_{ex}(\vec{r},E).\end{aligned} \tag{2.10}$$

This is called continuous-energy diffusion equation. Note that $D(\vec{r},E)$ in Eq.(2.9) depends on direction u and on J_u with energy coupling. The diffusion coefficient depends not only the cross sections but also on $\vec{J}$. The dependence on

$\vec{J}$ means that the diffusion coefficient depends on the solution introducing nonlinearity into the problem.

A conveniently workable implementation of Eq.(2.9) is not trivial and involves a varying degree of approximations. This is one of the sources for trouble in diffusion theory. A common implementation is to neglect energy transfer in the anisotropic scattering (which is obviously not a good assumption for light nuclei). Then Eq.(2.9) simplifies to

$$D(\vec{r},E)=\frac{1}{3}\left[\sigma_t(\vec{r},E)-\sigma_{s1}(\vec{r},E'\to E)\right]^{-1}, \tag{2.11}$$

or

$$\begin{aligned} D(\vec{r},E) &= \frac{1}{3}\left[\sigma_t(\vec{r},E)-\bar{\mu}_0\sigma_s(\vec{r},E)\right]^{-1} \\ &\equiv \frac{1}{3\sigma_{tr}} \end{aligned} \tag{2.12}$$

where $\bar{\mu}_0$ is the mean scattering angle.

Note that the direction dependency and energy coupling disappeared in Eq.(2.11) or (2.12). In a more rigorous implementation of determining multigroup diffusion coefficients, the complexity is simplified somewhat by the fine group (or continuous energy) solution of the P_1 equations for an infinite medium or the space-energy separability assumption of the flux within the group.

To our relief, in modern nodal methods (which we will discuss later), the direction dependency of the diffusion coefficient can be ignored and the diffusion coefficient itself can be determined arbitrarily (conveniently for practice) according to the equivalence theory for homogenization. But the burden is transferred to the discontinuity factors newly introduced in the equivalence theory.

2.2 Multigroup Approximation

The transport equation (2.3) and the diffusion equation (2.10) are of continuous form in independent variables. Except for extremely simple cases, it is not feasible to find exact solutions for them. We need to call for a variety of methods by which the governing equations are discretized and solved numerically. The first discretization we consider is the energy variable ; called multigroup approximation, that is common to virtually all deterministic methods.

Stochastic methods such as the Monte Carlo method can treat continuous energy variable, thus avoiding the complexities incurring from the multigroup approximation, e.g., resonance absorption and self-shielding. These methods are not dealt with in this article for the sake of space. Refer to references [5] through [9] for these methods of growing importance.

2.2.1 Multigroup Transport Equations

The discretization of the energy variable in the transport equation may proceed from Eq.(2.3). First, we divide the energy range into G intervals, where $E_G=0$ and E_0 is sufficiently large such that it covers the highest energy that any neutron can have. Then we obtain an approximation to the transport equation in terms of the group angular flux defined by

$$\varphi_g(\vec{r},\vec{\Omega})=\int_g dE\varphi(\vec{r},E,\vec{\Omega}),\quad g=1,2,\cdots,G. \tag{2.13}$$

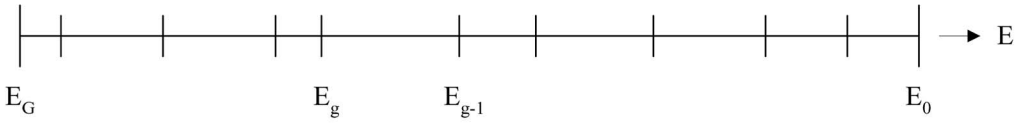


Figure 2.1 Division of the energy range into G energy groups.

Integrating Eq.(2.3) over energy group g, we have

$$\vec{\Omega}\cdot\nabla\int_g dE\varphi(\vec{r},E,\vec{\Omega})+\int_g dE\sigma(\vec{r},E)\varphi(\vec{r},E,\vec{\Omega})$$
$$=\sum_{g'=1}^{G}\int_g dE\int_{g'}dE'\int d\vec{\Omega}'\sigma_s(\vec{r},E'\to E,\vec{\Omega}'\cdot\vec{\Omega})\varphi(\vec{r},E',\vec{\Omega}') \tag{2.14}$$
$$+\int_g dE\chi(E)\sum_{g'=1}^{G}\int_{g'}dE'\ \nu\sigma_f(\vec{r},E')\phi(\vec{r},E')+\int_g dEq_{ex}(\vec{r},E,\vec{\Omega}).$$

Now we avoid the common use of the ad hoc energy separability but we proceed to use the Legendre moments of the within-group angular flux distribution. They may be available from solutions of a simpler problem that represents the problem under study more or less characteristically.

For simplicity, let us consider Eq.(2.3) in slab geometry. Using the Legendre addition theorem in the scattering term, we have

$$\mu\frac{\partial}{\partial x}\varphi(x,E,\mu)+\sigma(x,E)\varphi(x,E,\mu)$$
$$=\sum_{l=0}^{\infty}(2l+1)P_l(\mu)\int_0^{\infty}dE'\sigma_{sl}(x,E'\to E)\phi_l(x,E') \tag{2.15}$$
$$+\chi(E)\int_0^{\infty}dE'\nu\sigma_f(x,E')\phi(x,E')+q_{ex}(x,E,\mu).$$

Integrating Eq.(2.15) over energy group g, we have, corresponding to Eq.(2.14),

$$\mu\frac{\partial}{\partial x}\int_g dE\varphi(x,E,\mu)+\int_g dE\sigma(x,E)\varphi(x,E,\mu)$$
$$=\sum_{l=0}^{\infty}(2l+1)P_l(\mu)\sum_{g'=1}^{G}\int_g dE\int_{g'}dE'\sigma_{sl}(x,E'\to E)\phi_l(x,E') \tag{2.16}$$
$$+\int_g dE\chi(E)\sum_{g'=1}^{G}\int_{g'}dE'\nu\sigma_f(x,E')\phi(x,E')+\int_g dEq_{ex}(x,E,\mu).$$

Eq.(2.16) can be rewritten as

$$\mu\frac{\partial}{\partial x}\varphi_g(x,\mu)+\hat{\sigma}_g(x,\mu)\varphi_g(x,\mu)=\sum_{l=0}^{\infty}(2l+1)P_l(\mu)\sum_{g'=1}^{G}\sigma_{lgg'}(x)\phi_{lg'}(x)$$
$$+\chi_g\sum_{g'=1}^{G}\nu\sigma_{fg'}(x)\phi_{g'}(x)+q_g^e(x,\mu), \tag{2.17}$$

if the multigroup cross sections are now defined as

$$\hat{\sigma}_g(x,\mu)=\frac{\int_g dE\sigma(x,E)\varphi(x,E,\mu)}{\varphi_g(x,\mu)}, \tag{2.18}$$

$$\nu\sigma_{fg}(x)=\frac{\int_g dE\nu\sigma_f(x,E)\phi(x,E)}{\phi_g(x)}, \tag{2.19}$$

$$\sigma_{lgg'}(x)=\frac{\int_g dE\int_{g'}dE'\sigma_{sl}(x,E'\to E)\phi_l(x,E')}{\phi_{lg'}(x)}, \tag{2.20}$$

where

$$\phi_{lg'}(x)=\int_{g'}dE'\phi_l(x,E'). \tag{2.21}$$

The angular dependency in $\hat{\sigma}_g(x,\mu)$ is noted. This angular dependency is problematic for computation. To avoid the angular-dependency and to utilize the Legendre moments of the angular flux, the second term in Eq.(2.16) is given an alternative (and sophisticated) treatment as the following. Let us expand

$$\varphi(x,E,\mu)=\sum_{l=0}^{\infty}(2l+1)P_l(\mu)\phi_l(x,E), \tag{2.22}$$

where

$$\phi_l(x,E) = \int_{-1}^{1} \frac{d\mu}{2} P_l(\mu)\varphi(x,E,\mu). \tag{2.23}$$

Using Eq.(2.22), 2nd term in Eq.(2.16) becomes

$$\begin{aligned}\int_g dE\sigma(x,E)\varphi(x,E,\mu) &= \sum_{l=0}^{\infty}(2l+1)P_l(\mu)\int_g dE\sigma(x,E)\phi_l(x,E)\\ &= \sum_{l=0}^{\infty}(2l+1)P_l(\mu)\sigma_{lg}(x)\phi_{lg}(x),\end{aligned} \tag{2.24}$$

if we define

$$\begin{aligned}\sigma_{lg}(x) &= \frac{\int_g dE\sigma(x,E)\phi_l(x,E)}{\int_g dE\phi_l(x,E)}\\ &= \frac{\int_g dE\sigma(x,E)\phi_l(x,E)}{\phi_{lg}(x)}.\end{aligned} \tag{2.25}$$

Then, Eq.(2.16) becomes

$$\begin{aligned}\mu\frac{\partial}{\partial x}\varphi_g(x,\mu) = &\sum_{l=0}^{\infty}(2l+1)P_l(\mu)\sum_{g'=1}^{G}\left[\sigma_{lgg'}(x)-\sigma_{lg}(x)\delta_{gg'}\right]\phi_{lg'}(x)\\ &+\chi_g\sum_{g'=1}^{G}\nu\sigma_{fg'}(x)\phi_{g'}(x)+q_g^e(x,\mu).\end{aligned} \tag{2.26}$$

From Eq.(2.22),

$$\varphi_g(x,\mu) = \sum_{l=0}^{\infty}(2l+1)P_l(\mu)\phi_{lg}(x). \tag{2.27}$$

Multiplying this by $\sigma_g(x)$

$$\sigma_g(x)\varphi_g(x,\mu) = \sum_{l=0}^{\infty}\sigma_g(x)(2l+1)P_l(\mu)\phi_{lg}(x), \tag{2.28}$$

where $\sigma_g(x)$ is positive, not specified yet, but not dependent on μ. Adding Eqs. (2.26) and (2.28) by sides, we obtain one-dimensional multigroup transport question:

$$\begin{aligned}\mu\frac{\partial}{\partial x}\varphi_g(x,\mu)+\sigma_g(x)\varphi_g(x,\mu) = &\sum_{l=0}^{\infty}(2l+1)P_l(\mu)\sum_{g'=1}^{G}\hat{\sigma}_{lgg'}(x)\phi_{lg'}(x)\\ &+\chi_g\sum_{g'=1}^{G}\nu\sigma_{fg'}(x)\phi_{g'}(x)+q_g^e(x,\mu),\end{aligned} \tag{2.29}$$

where

$$\hat{\sigma}_{lgg'}(x) = \sigma_{lgg'}(x)+[\sigma_g(x)-\sigma_{lg}(x)]\delta_{gg'}. \tag{2.30}$$

A remaining question is how to specify $\sigma_g(x)$.

i) Consistent P Approximation

This approximation chooses

$$\sigma_g(x) = \sigma_{0g}(x), \tag{2.31}$$

so that $\sigma_g(x)$ is simply the flux-averaged total collision cross section from Eq.(2.25). For the name, see Section 5.4 of [3].

ii) Extended Transport Approximation

A more elegant technique is to choose $\sigma_g(x)$ to cause the first truncated scattering term, $l = L+1$, to be small, since the Legendre expansion in the scattering term is usually kept up to $l = L$:

$$(2L+3)P_{L+1}(\mu)\sum_{g'=1}^{G}\left\{\sigma_{L+1gg'}(x)+\left[\sigma_g(x)-\sigma_{L+1g}(x)\right]\delta_{gg'}\right\}\phi_{L+1g'}\simeq 0. \tag{2.32}$$

We expect in reactors, for most groups,

$$\sum_{g'=1}^{G}\sigma_{L+1gg'}(x)\phi_{L+1g'}(x)\simeq\sum_{g'=1}^{G}\sigma_{L+1g'g}(x)\phi_{L+1g}(x), \tag{2.33}$$

which is an expression of detailed balance. This would be strictly valid if there is no absorption. Substituting Eq.(2.33) into Eq.(2.32), we obtain extended transport approximation for $\sigma_g(x)$:

$$\sigma_g(x)=\sigma_{L+1,g}(x)-\sum_{g'=1}^{G}\sigma_{L+1,g'g}(x). \tag{2.34}$$

The extended transport approximation should be more effective in strongly anisotropic scattering problems. After all, determination of multigroup cross sections boils down to a simpler problem of approximately finding the moments of the within-group angular flux distribution : $\phi_l(x,E)$ or an asymptotic spectra $\psi_l(E)$. Finding $\phi_l(x,E)$ or even $\psi_l(E)$ is of course not trivial. For each energy interval that includes resonances (see Figure 2.2), all the physics of resonance absorption such as energy- and spatial- self shieldings should be considered.

Note that for $L=0$ ("simple" transport approximation) Eq.(2.34) becomes

$$\sigma_g=\sigma_{1g}-\sum_{g'=1}^{G}\sigma_{1g'g}=\sigma_{1g}-\overline{\mu}_g\sigma_{s0g}. \tag{2.35}$$

It will be interesting to note that Eq.(2.35) is closely related to multigroup diffusion coefficients.
In summary, a 3-D representation of Eq.(2.29) can be written as

$$\begin{aligned}\vec{\Omega}\cdot\nabla\varphi_g(\vec{r},\vec{\Omega})+\sigma_g(\vec{r})\varphi_g(\vec{r},\vec{\Omega})=&\sum_{g'=1}^{G}\sum_{l=0}^{\infty}\sigma_{lgg'}(\vec{r})\sum_{m=-l}^{l}Y_{lm}(\vec{\Omega})\phi_{lg'}^{m}(\vec{r})\\&+\chi_g\sum_{g'=1}^{G}\nu\sigma_{fg'}(\vec{r})\phi_{g'}(\vec{r})+q_g^e(\vec{r},\vec{\Omega}),\end{aligned} \tag{2.36}$$

where

$$\phi_{lg'}^{m}(\vec{r})=\int d\vec{\Omega}'\,Y_{lm}^{*}(\vec{\Omega}')\varphi_{g'}(\vec{r},\vec{\Omega}'), \tag{2.37}$$

and $\sigma_{lgg'}=\widehat{\sigma}_{lgg'}$ from Eq.(2.30).

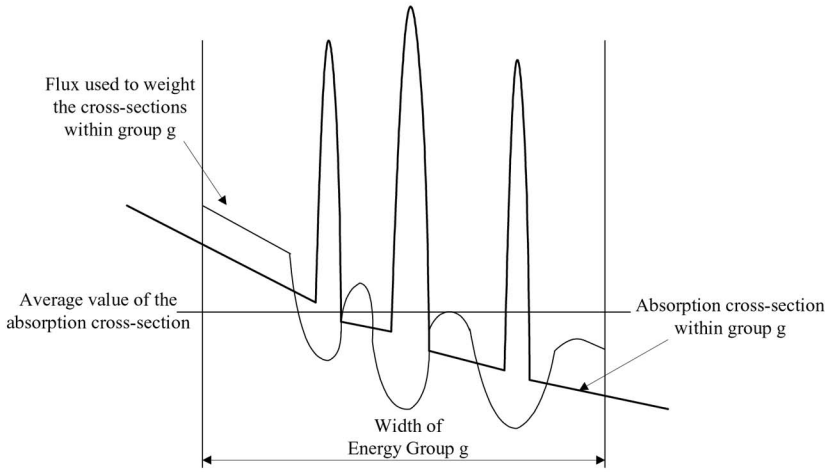


Figure 2.2 Absorption resonances.

Eq.(2.36) becomes for one-dimensional geometry (y- and z-uniform, infinite),

$$\mu\frac{\partial}{\partial x}\varphi_g(x,\mu)+\sigma_g(x)\varphi_g(x,\mu)=\sum_{g'=1}^{G}\sum_{l=0}^{\infty}\sigma_{lgg'}(x)P_l(\mu)(2l+1)\int_{-1}^{1}\frac{d\mu'}{2}P_l(\mu')\varphi_{g'}(x,\mu') +\chi_g\sum_{g'=1}^{G}\nu\sigma_{fg'}(x)\phi_{g'}(x)+q_g^e(x,\mu), \tag{2.38}$$

or, for two-dimensional x-y geometry (z-uniform, infinite),

$$\left[\mu\frac{\partial}{\partial x}+\eta\frac{\partial}{\partial y}+\sigma_g(x,y)\right]\varphi_g(x,y,\mu,\eta) =\sum_{g'=1}^{G}\sum_{l=0}^{\infty}\sigma_{lgg'}(x,y)\sum_{m=0}^{l}(2-\delta_{m0})Y_{lm}^e(\mu,\eta)\int d\vec{\Omega}Y_{lm}^e(\mu',\eta')\varphi_{g'}(x,y,\mu',\eta') +\chi_g\sum_{g'=1}^{G}\nu\sigma_{fg'}(x,y)\phi_{g'}(x,y)+q_g^e(x,y,\mu,\eta), \tag{2.39}$$

where

$$Y_{lm}^e=\sqrt{\frac{(2l+1)(l-m)!}{(l+m)!}}P_l^m(\mu)\cos(m\varpi),$$

$$Y_{lm}^o=\sqrt{\frac{(2l+1)(l-m)!}{(l+m)!}}P_l^m(\mu)\sin(m\varpi),$$

$$Y_{lm}=Y_{lm}^e+jY_{lm}^o.$$

2.2.2 Multigroup Diffusion Equations

The discretization of the energy variable in the diffusion equation may start with Eq.(2.10),

$$-\nabla\cdot D(\vec{r},E)\nabla\phi(\vec{r},E)+\Sigma_t(\vec{r},E)\phi(\vec{r},E)=\int_0^\infty dE'\,\Sigma_{s0}(\vec{r},E'\to E)\phi(\vec{r},E') +\chi(E)\int_0^\infty dE'\,\nu\Sigma_f(\vec{r},E')\phi(\vec{r},E')+Q_{ex}(\vec{r},E). \tag{2.40}$$

Integrating Eq.(2.40) over energy group g, we have

$$-\nabla D_g(\vec{r})\nabla\phi_g(\vec{r})+\Sigma_{rg}(\vec{r})\phi_g(\vec{r})=\sum_{\substack{g'=1\\g'\neq g}}^{G}\Sigma_{sg'\to g}(\vec{r})\phi_{g'}(\vec{r}) +\chi_g\sum_{g'=1}^{G}\nu\Sigma_{fg'}(\vec{r})\phi_{g'}(\vec{r})+S_g(\vec{r}), \tag{2.41}$$

where

$$\phi_g(\vec{r})=\int_g dE\phi(\vec{r},E),\quad \chi_g=\int_g dE\chi(E),$$
$$S_g(\vec{r})=\int_g dEQ_{ex}(\vec{r},E), \tag{2.42}$$

$$\Sigma_{\alpha g}(\vec{r})=\frac{\int_g dE\Sigma_\alpha(\vec{r},E)\phi(\vec{r},E)}{\int_g dE\phi(\vec{r},E)},\quad \alpha=t,f, \tag{2.43}$$

$$\Sigma_{sg'\to g}(\vec{r})=\frac{\int_g dE\int_{g'}dE'\,\Sigma_{so}(\vec{r},E'\to E)\phi(\vec{r},E')}{\int_{g'}dE'\phi(\vec{r},E')}, \tag{2.44}$$

$$\Sigma_{rg}(\vec{r})=\Sigma_{tg}(\vec{r})-\Sigma_{sg\to g}(\vec{r}), \tag{2.45}$$

and

$$D_g(\vec{r}) = \frac{1}{3\Sigma_{trg}(\vec{r})}, \tag{2.46}$$

where (in view of Eq.(2.9)),

$$\Sigma_{trg}(\vec{r}) = \frac{\int_g dE \int_0^\infty dE' \left[\Sigma_t(\vec{r},E)\delta(E'-E) - \Sigma_{s1}(\vec{r},E'\to E)\right]\vec{J}(\vec{r},E')}{\int_g dE \vec{J}(\vec{r},E)}. \tag{2.47}$$

$\phi(\vec{r},E)$ and $J(\vec{r},E)$ to be used in Eqs.(2.43) through (2.47) are usually obtained in the following procedure :

i) Find P_1 or transport asymptotic solutions $\psi(E)$ and $J(E)$ in fine group form of a simpler problem.

ii) Use $\psi(E)$ and $J(E)$ for $\phi(\vec{r},E)$ and $J(\vec{r},E)$, respectively.

iii) If we neglect energy transfer in Σ_{s1}, use of $J(E)$ in Eq.(2.47) leads to $D_g(\vec{r})$ of usual form. Otherwise, the multigroup diffusion coefficients become direction-dependent and group-coupled, $D_{gg',u}(\vec{r})$, $u = x, y, z$ (see Henry [10]). These multigroup coefficients cause complications in computation and are rarely in use.

These are the basis of the most treatments of reactor diffusion theory. Without external source, the multigroup diffusion equations become an eigenvalue problem:

$$-\nabla\cdot D_g(\vec{r})\nabla\phi_g(\vec{r}) + \Sigma_{rg}(\vec{r})\phi_g(\vec{r}) = \sum_{g'\neq g}^{G}\Sigma_{sg'\to g}(\vec{r})\phi_{g'}(\vec{r}) + \frac{\chi_g}{k}\sum_{g'=1}^{G}\nu\Sigma_{fg'}(\vec{r})\phi_{g'}(\vec{r}). \tag{2.48}$$

3. NUMERICAL METHODS FOR MULTIGROUP DIFFUSION EQUATIONS

This section provides key elements of the numerical methods that are the foundations of reactor analysis and design : finite difference methods and modern nodal methods for multigroup diffusion equations. The finite difference methods are basic numerical techniques in all areas of science and engineering. They have been taken up in reactor physics early on and have been a major workforce up to the 1970's. Now they have been almost completely replaced by the so-called modern nodal methods.

The modern nodal methods first appeared in mid-seventies, almost three decades ago, but still retaining the adjective "modern", perhaps for a lack of new methods thereafter. The nodal methods now have taken a firm place in the current production codes for reactor design as a main computational engine.

The modern nodal methods use very coarse meshes (~20cm×20cm/mesh) resulting in dramatic reduction in computing time compared to the finite difference methods. But they attain very high accuracy by careful treatment in discretizing the diffusion equations to enforce neutron balance.

3.1 Overall Structure of the Solution Framework

Before we discuss specific solution methods for the spatially discretized diffusion equations, we provide an overall structure of the numerical solution framework. Let us consider the multigroup or few-group diffusion equations in Eq.(2.48). For the sake of simplicity, we assume that there is no upscattering :

$$-\nabla\cdot D_g(\vec{r})\nabla\phi_g(\vec{r}) + \Sigma_{rg}(\vec{r})\phi_g(\vec{r}) = \sum_{g'=1}^{g-1}\Sigma_{sgg'}(\vec{r})\phi_{g'}(\vec{r}) + \frac{\chi_g}{k}\sum_{g'=1}^{G}\nu\Sigma_{fg'}(\vec{r})\phi_{g'}(\vec{r}). \tag{3.1}$$

If we let

$$S(\vec{r}) = \sum_{g'=1}^{G}\nu\Sigma_{fg'}(\vec{r})\phi_{g'}(\vec{r}), \quad S_g(\vec{r}) = \chi_g S(\vec{r}), \tag{3.2}$$

Eq.(3.1) becomes

$$-\nabla\cdot D_g(\vec{r})\nabla\phi_g(\vec{r}) + \Sigma_{rg}(\vec{r})\phi_g(\vec{r}) = \frac{1}{k}S_g(\vec{r}) + \sum_{g'=1}^{g-1}\Sigma_{sgg'}(\vec{r})\phi_{g'}(\vec{r}). \tag{3.3}$$

If we discretize Eq.(3.3) in "some way" with mesh index m and further let

$$\Phi = (\phi_1^m, \phi_2^m, \cdots, \phi_G^m)^T, \quad F_{gg'}^{mm} = \chi_g \nu\Sigma_{fg'}^m \tag{3.4}$$

then Eq.(3.3) becomes

$$M\Phi = \frac{1}{k} F\Phi, \tag{3.5}$$

for which a solution scheme is

$$A_g \phi_g^{(t+1)} = \frac{1}{k^{(t)}} S_g^{(t)} + \sum_{g'=1}^{g-1} \Sigma_{sgg'} \phi_{g'}^{(t+1)} \equiv Q_g^{(t)}, \quad g = 1, 2, \cdots, G, \tag{3.6}$$

$$\frac{1}{k^{(t+1)}} = \frac{1}{k^{(t)}} \frac{(F\Phi^{(t)}, F\Phi^{(t)})}{(F\Phi^{(t+1)}, F\Phi^{(t+1)})}. \tag{3.7}$$

Here, t is index of "outer iteration", the update procedure for ϕ_g and k in Eqs.(3.6) and (3.7) is called power method. For two- or three-dimensional problems, Eq.(3.6) itself is solved by some kind of iterations, called "inner iteration". Figure 3.1 shows an iterative solution scheme for Eq.(3.5) that is broken down into Eqs.(3.6) and (3.7).

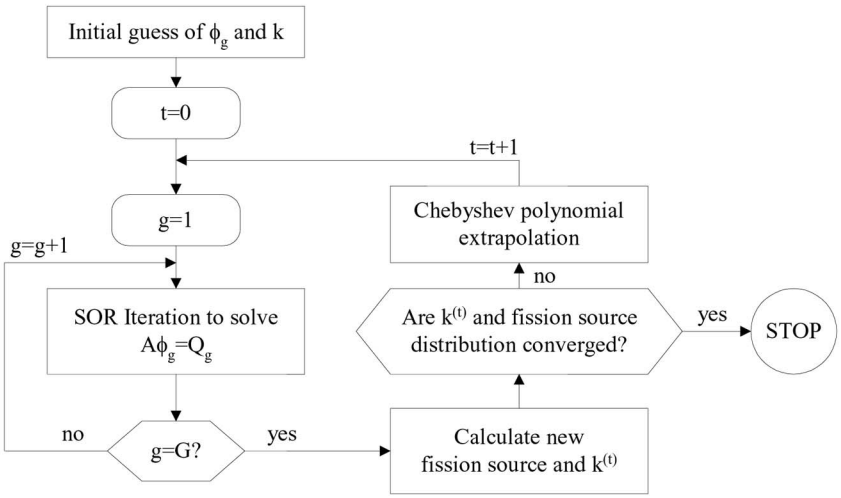


Figure 3.1 Iterative solution scheme for Eq.(3.5).

3.2 Finite Difference Methods

As an example, let us consider the following two-dimensional few-group neutron diffusion equations written in a form :

$$\nabla \cdot \vec{J}_g^m(x,y) + \Sigma_{rg}^m \phi_g^m(x,y) = \sum_{g'} \left(\Sigma_{gg'}^m + \frac{\chi_g}{k} \nu\Sigma_{fg'}^m \right) \phi_{g'}^m(x,y), \tag{3.8a}$$

$$\vec{J}_g^m(x,y) = -D_g^m \nabla \phi_g^m(x,y), \tag{3.8b}$$

where m stands for node (or mesh) m (see Figure 3.2). We assume that each node is homogeneous. If we integrate Eq.(3.8a) over node m, we have the following neutron balance equation :

$$\frac{1}{V_m} \sum_{ms=1}^{4} A_{ms} J_g(ms) + \Sigma_{rg}^m \bar{\phi}_g^m = \sum_{g'} \left(\Sigma_{gg'}^m + \frac{\chi_g}{k} \nu\Sigma_{fg'}^m \right) \bar{\phi}_{g'}^m. \tag{3.9}$$

A note on Eq.(3.9) is in order. Eq.(3.9) is a result of node integration of multigroup form of the first equation of P_1 equations in which no approximation was introduced in the angular treatment. Thus, Eq.(3.9) is exact within the multigroup formalism and does not contain spatial discretization and angular treatment errors.

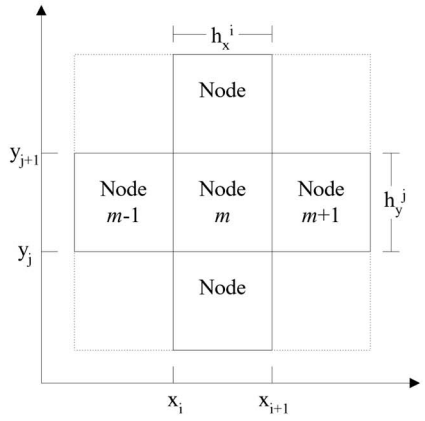


Figure 3.2 Node *m* and its neighbors.

Now we need to provide a relation of $J_g(x_{i+1})$ to $\overline{\phi}_g^m$ and $\overline{\phi}_g^{m+1}$, so that we can eliminate $J_g(ms)$ in Eq.(3.9). A finite difference method provides this relation as

$$J_g^{m,m+1} = -\frac{2D_g^m D_g^{m+1}}{h(D_g^m + D_g^{m+1})}(\overline{\phi}_g^{m+1} - \overline{\phi}_g^m), \tag{3.10}$$

which is a discretized approximation for the Fick's law Eq.(3.8b) (which itself is an approximation)

$$\vec{J}_g(x_{i+1}) = -D_g \nabla \phi_g\Big|_{x_{i+1}}$$

and enforcing continuities of the flux and currents at the interface. To see what is involved in inner iterations, let us consider the structure of matrix A in Eq.(3.6) for Figure 3.2. Eq.(3.6) is represented as (dropping g)

$$A\phi = Q \tag{3.11}$$

$$\left[\begin{array}{ccc|ccc|ccc}
a_{11}^C & a_{11}^R & 0 & a_{11}^T & 0 & 0 & 0 & 0 & 0 \\
a_{21}^L & a_{21}^C & a_{21}^R & 0 & a_{21}^T & 0 & 0 & 0 & 0 \\
0 & a_{31}^L & a_{31}^C & 0 & 0 & a_{31}^T & 0 & 0 & 0 \\
\hline
a_{12}^B & 0 & 0 & a_{12}^C & a_{12}^R & 0 & a_{12}^T & 0 & 0 \\
0 & a_{22}^B & 0 & a_{22}^L & a_{22}^C & a_{22}^R & 0 & a_{22}^T & 0 \\
0 & 0 & a_{32}^B & 0 & a_{32}^L & a_{32}^C & 0 & 0 & a_{32}^T \\
\hline
0 & 0 & 0 & a_{13}^B & 0 & 0 & a_{13}^C & a_{13}^R & 0 \\
0 & 0 & 0 & 0 & a_{23}^B & 0 & a_{23}^L & a_{23}^C & a_{23}^R \\
0 & 0 & 0 & 0 & 0 & a_{33}^B & 0 & a_{33}^L & a_{33}^C
\end{array}\right]
\left[\begin{array}{c}
\phi_{11} \\ \phi_{21} \\ \phi_{31} \\ \hline \phi_{12} \\ \phi_{22} \\ \phi_{32} \\ \hline \phi_{13} \\ \phi_{23} \\ \phi_{33}
\end{array}\right]
=
\left[\begin{array}{c}
Q_{11} \\ Q_{21} \\ Q_{31} \\ \hline Q_{12} \\ Q_{22} \\ Q_{32} \\ \hline Q_{13} \\ Q_{23} \\ Q_{33}
\end{array}\right] \tag{3.12}$$

Eq.(3.11) is solved by iteration (called inner iteration), e.g., by successive-line-overrelaxation (SLOR) :

$$B\tilde{\phi}^{(i)} = (L+U)\phi^{(i-1)} + Q, \tag{3.13}$$

$$\phi^{(i)} = \omega\tilde{\phi}^{(i)} + (1-\omega)\phi^{(i-1)}, \quad 1 < \omega < 2, \tag{3.14}$$

where

$$B=\left[\begin{array}{ccc|ccc|ccc} a_{11}^{C} & a_{11}^{R} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_{21}^{L} & a_{21}^{C} & a_{21}^{R} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{31}^{L} & a_{31}^{C} & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & a_{12}^{C} & a_{12}^{R} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{22}^{L} & a_{22}^{C} & a_{22}^{R} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a_{32}^{L} & a_{32}^{C} & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & a_{13}^{C} & a_{13}^{R} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a_{23}^{L} & a_{23}^{C} & a_{23}^{R} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{33}^{L} & a_{33}^{C} \end{array}\right], \tag{3.15}$$

$$L+U=\left[\begin{array}{ccc|ccc|ccc} 0 & 0 & 0 & -a_{11}^{T} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -a_{21}^{T} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -a_{31}^{T} & 0 & 0 & 0 \\ \hline -a_{12}^{B} & 0 & 0 & 0 & 0 & 0 & -a_{12}^{T} & 0 & 0 \\ 0 & -a_{22}^{B} & 0 & 0 & 0 & 0 & 0 & -a_{22}^{T} & 0 \\ 0 & 0 & -a_{32}^{B} & 0 & 0 & 0 & 0 & 0 & -a_{32}^{T} \\ \hline 0 & 0 & 0 & -a_{13}^{B} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -a_{23}^{B} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -a_{33}^{B} & 0 & 0 & 0 \end{array}\right]. \tag{3.16}$$

It is known [11, 12] that

$$\omega_{opt}=\frac{2}{1+\sqrt{1-\mu^{2}[B^{-1}(L+U)]}} \tag{3.17}$$

where μ stands for spectral radius, and that the spectral radius of SLOR is

$$\mu(\mathcal{L}_{\omega})=\omega_{opt}-1, \tag{3.18}$$

that governs convergence rate of the inner iteration ; the smaller, the faster. The spectral radius depends on the approximation methods such as the discretization schemes and the mesh sizes (in diffusion length) and on the acceleration methods used. Typically, the spectral radius increases as the mesh size decreases.

The convergence of k and ϕ_g of the power method used in outer iteration in Eqs.(3.6) and (3.7) is governed by the dominance ratio, defined as

$$\sigma=\left|\frac{k_1}{k_0}\right|, \tag{3.19}$$

where k_0 and k_1 are the fundamental-mode (largest) and first-mode (second largest) eigenvalues, respectively, of the matrix $M^{-1}F$. The dominance ratio depends on physical properties of the reactor rather than on the approximation methods. Typically, it depends on the size of the reactor in diffusion length and it is insensitive to the mesh size itself. Of course, the effective dominance ratio can be reduced by acceleration methods, such as the Chebyshev polynomial extrapolation methods. Like SLOR, these acceleration methods require some parameters to be estimated online.

Quite recently, parameter estimation-free iterative methods such as the Krylov subspace methods draw attention in the reactor physics and transport theory community [13, 14].

3.3 Modern Nodal Methods

3.3.1 Equivalence Homogenization Theory

Since modern nodal methods are based on a large coarse mesh (of an assembly size) called “node” whose properties are

constant, the first step in nodal methods is homogenization (possibly plus group condensation). The homogenization provides equivalent constant properties of a node that are physically heterogeneous. This is a process that converts a physical system that is difficult to analyze to a mathematical system (see Figure 3.3) that is easier to analyze but gives equivalent solutions for important parameters (such as reaction rates of the node and multiplication factor of the core).

The nodal equivalence is established between the global heterogeneous solution and the global homogeneous solution by preserving the following key quantities for each node :

i) All group reaction rates

$$\int_{V_i} \Sigma_{\alpha g}^{hom} \phi_g^{hom}(\vec{r}) d\vec{r} = \int_{V_i} \Sigma_{\alpha g}^{het} \phi_g^{het}(\vec{r}) d\vec{r}, \quad g = 1,2,\cdots,G \quad \alpha = t, f, g' \tag{3.20}$$

ii) All group surface currents (leakage rate out of the node)

$$-\int_{S_i^k} D_g^{hom} \frac{\partial}{\partial u} \phi_g^{hom}(\vec{r}) \cdot d\vec{S} = \int_{S_i^k} \vec{J}_{gu}^{het}(\vec{r}) \cdot d\vec{S}, \quad g = 1,2,\cdots,G \quad k = 1,2,\cdots,K \text{ surfaces of the node} \tag{3.21}$$

iii) Reactor eigenvalue

$$-\sum_{k=1}^{K} \int_{S_i^k} D_g^{hom} \nabla \phi_g^{hom}(\vec{r}) \cdot d\vec{S} + \int_{V_i} \Sigma_{tg}^{hom} \phi_g^{hom}(\vec{r}) d\vec{r} = \sum_{g'=1}^{G} \left[\int_{V_i} \Sigma_{g' \to g}^{hom} \phi_{g'}^{hom}(\vec{r}) d\vec{r} + \frac{\chi_g}{k^{hom}} \int_{V_i} \nu \Sigma_{fg'}^{hom} \phi_{g'}^{hom}(\vec{r}) d\vec{r} \right], \quad g = 1,2,\cdots,G \tag{3.22}$$

where $k^{hom} = k^{het}$.

Therefore, ideal homogenized parameters are given as

$$\Sigma_{\alpha g}^{hom,i} = \frac{\int_{V_i} \Sigma_{\alpha g}^{het} \phi_g^{het}(\vec{r}) d\vec{r}}{\int_{V_i} \phi_g^{hom}(\vec{r}) d\vec{r}}, \quad g = 1,2,\cdots,G \quad \alpha = t, f, g' \tag{3.23}$$

$$D_g^{hom,i} = \frac{-\int_{S_i^k} \vec{J}_g^{het}(\vec{r}) \cdot d\vec{S}}{\int_{S_i^k} \nabla \phi_g^{hom}(\vec{r}) \cdot d\vec{S}}, \quad g = 1,2,\cdots G \quad k = 1,2,\cdots,K \tag{3.24}$$

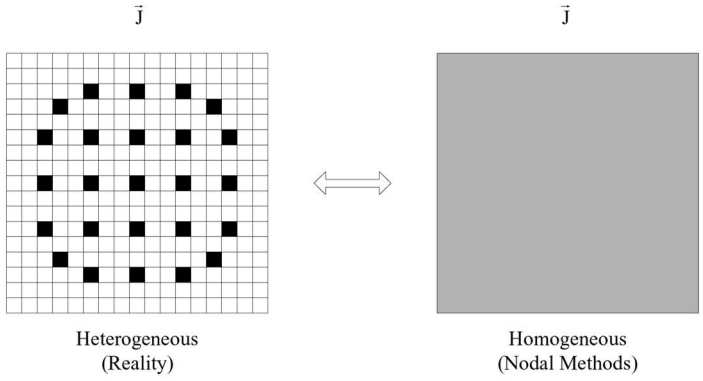


Figure 3.3 Homogenization and equivalence.

These ideal homogenized parameters contain three problems for applications to real systems - the first two for practicality reasons and third for theoretical reasons :

i) It requires knowledge of the global heterogeneous flux and current. This can be obtained by the whole core FDM solution which we like to avoid.

ii) It requires knowledge of homogeneous solution before homogenized parameters are determined, that introduces vicious-cycle nonlinearity between homogenization and homogenized solution.

iii) Eq.(3.24) must be valid for all of the K surfaces of the node, which is against the physics. This is because the leakages in the numerator in Eq.(3.24) are in general all different on the K surfaces and so are the ratios. The imposition of the conventional continuity conditions of homogeneous flux and net current on all nodal surfaces will give different $D_g^{hom,i}$ values for each surface. This contradicts our desire of generating spatially constant parameters (single value in a node for each parameter).

To overcome the difficulties above, Koebke [15] proposed the following equivalence theory method, generalized later by Smith [16] :

i) Cross sections are homogenized as in the conventional flux-volume weighting :

$$\Sigma_{\alpha g}^{hom,i} = \frac{\int_{V_i} \Sigma_{\alpha g}^{het} \phi_{Ag}(\vec{r}) d\vec{r}}{\int_{V_i} \phi_{Ag}(\vec{r}) d\vec{r}}, \quad g = 1,2,\cdots G \quad \alpha = t, f, g', \tag{3.25}$$

using the usual isolated assembly heterogeneous calculation with zero net current boundary condition, and

$$\int_{V_i} \phi_g^{hom}(\vec{r}) d\vec{r} = \int_{V_i} \phi_g^{het}(\vec{r}) d\vec{r} = \int_{V_i} \phi_{Ag}(\vec{r}) d\vec{r}.$$

$\phi_g^{het}(\vec{r})$ is approximated by $\phi_{Ag}(\vec{r})$ which is the result of the isolated single assembly calculation. $\phi_g^{het}(\vec{r})$ would have been obtained if real boundary conditions were used (which is not possible, because they are available only if a global heterogeneous calculation is done).

ii) Approximate the homogenized diffusion coefficients by (arbitrarily!)

$$D_g^{hom,i} = \frac{\int_{V_i} D_g^{het}(\vec{r}) \phi_{Ag}(\vec{r}) d\vec{r}}{\int_{V_i} \phi_{Ag}(\vec{r}) d\vec{r}}, \quad g = 1,2,\cdots, G \tag{3.26}$$

or

$$\frac{1}{D_g^{hom,i}} = \frac{\int_{V_i} \frac{1}{D_g^{het}(\vec{r})} \phi_{Ag}(\vec{r}) d\vec{r}}{\int_{V_i} \phi_{Ag}(\vec{r}) d\vec{r}}, \quad g = 1,2,\cdots, G \tag{3.27}$$

iii) Introduce two discontinuity factors (one heterogeneity factor in the case of Koebke, requiring some iterations) per direction and allow discontinuity of the homogeneous surface flux between the two nodes (but still enforcing continuity of the homogeneous surface current) :

$$\tilde{\phi}_{gi}^{hom}(x_{i+1/2}) \cdot f_{gi}^{x+} = \tilde{\phi}_{gi+1}^{hom}(x_{i+1/2}) \cdot f_{gi+1}^{x-}, \tag{3.28}$$

where the two discontinuity factors for each direction are defined as

$$f_{gi}^{x+} = \frac{\phi_{gi}^{het}(x_{i+1/2})}{\tilde{\phi}_{gi}^{hom}(x_{i+1/2})}, \tag{3.29}$$

$$f_{gi+1}^{x-} = \frac{\phi_{gi+1}^{het}(x_{i+1/2})}{\tilde{\phi}_{gi+1}^{hom}(x_{i+1/2})}. \tag{3.30}$$

Eq.(3.28) states that the heterogeneous surface flux is continuous, although the homogeneous surface flux may be discontinuous. If the results of the single-assembly calculation $\phi_{Ag}(\vec{r})$ are used as the heterogeneous surface fluxes in the numerators in Eqs.(3.29) and (3.30), the discontinuity factors are particularly called assembly discontinuity factors (ADFs). These ADFs are usually used in practice, which may be the single most important source of approximation in modern nodal methods. Figure 3.4 compares the diffusion coefficients and discontinuous factors for a 2-D problem between Koebke and Smith.

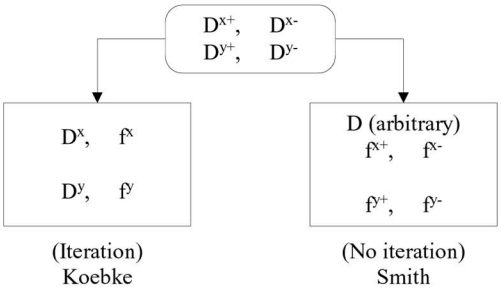


Figure 3.4 Comparison of D and f between Koebke and Smith.

Recall that diffusion coefficients should be originally direction-dependent (axis-dependent) as discussed in Section 2.1.2. This is so even in a homogeneous medium because of the solution dependency of the diffusion coefficient in its definition. But in homogenization of a heterogeneous medium the diffusion coefficient depends not only on axis (e.g., x-axis) but also on direction (e.g., $x+$ or $x-$) ; D^{x+} and D^{x-}. Therefore, we may say that the equivalence homogenization theory merely transfers this direction-dependency of the diffusion coefficients to the heterogeneity factors (from D^{x+} and D^{x-} to D^{x} and f^{x} in the case of Koebke, or to f^{x+} and f^{x-} in the case of Smith). Thus the real physics of the direction dependency of the diffusion coefficient is delegated to the discontinuity factors.

3.3.2 Modern Nodal Methods (Based on Transverse Integration Procedure)

Let us go back to Figure 3.2 and consider the nodes are set up for nodal methods, e.g., they are homogenized and of a coarse size. If we perform a transverse integration (e.g., in y-direction, see Figure 3.5) on Eq.(3.8), we obtain

$$-D_g^m \frac{d^2}{dx^2}\tilde{\phi}_g^m(x) + \Sigma_g^m \tilde{\phi}_g^m(x) = \sum_{g'}\left(\Sigma_{gg'}^m + \frac{\chi_g}{k}\nu\Sigma_{fg'}^m\right)\tilde{\phi}_{g'}^m(x) - \tilde{L}_{gx}^m(x), \tag{3.31}$$

where

$$\tilde{\phi}_g^m(x) = \frac{1}{h}\int_{-h/2}^{h/2}\phi_g^m(x,y)dy, \tag{3.32}$$

$$\tilde{L}_{gx}^m(x) = -\frac{D_g^m}{h}\int_{-h/2}^{h/2}\frac{\partial^2}{\partial y^2}\phi_g^m(x,y)dy. \tag{3.33}$$

$\tilde{L}_{gx}^m(x)$ is a transverse leakage term. A key step and also a very important approximation in modern nodal methods is to assume that $\tilde{L}_{gx}^m(x)$ can be represented by a low-order polynomial, typically a quadratic polynomial :

$$\tilde{L}_{gx}^m(x) = b_0 + b_1 x + b_2 x^2. \tag{3.34}$$

Then we can find the solution to Eq.(3.31) in a polynomial, typically in a 4-th order (called nodal expansion method (NEM) [17]), or even analytically (called analytic nodal method (ANM) [18]).

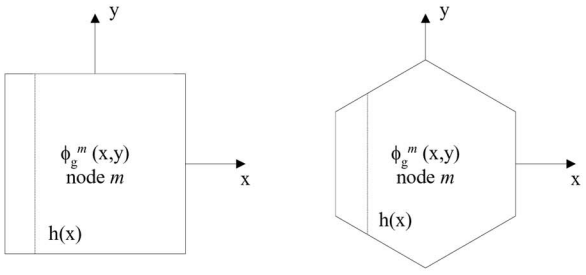


Figure 3.5 Transverse integration.

Then we can perform

$$J_{gx}^{m}(ms) = -D_{g}^{m}\frac{d}{dx}\tilde{\phi}_{g}^{m}(x)\bigg|_{s} \tag{3.35}$$

and relate $J_{gx}^{m}(ms)$ to the node-average fluxes of the two nodes. In this way, $J_{gx}^{m}(ms)$ can be eliminated from the neutron node balance equation (3.9). This coupling relation obtained in this way allows a large node size still maintaining high accuracy. (This was done by Eq.(3.10) in the case of finite difference methods.) The rest of the computational procedures in nodal methods are quite similar to those of the finite difference methods.

We can say that the success of modern nodal methods based on transverse integration hinges on :

- It is easier to guess the shape of $\tilde{L}_{gu}^{m}(u)$ than $\phi_{g}^{m}(x,y)$, because it is one-dimensional.
- The quadratic shape is O.K. for usual problems and for not too large nodes.

However, there are limitations in these methods due to transverse integration :

- The need to evaluate $\tilde{L}_{gu}^{m}(u)$.
- The polynomial (quadratic) approximation on $\tilde{L}_{gu}^{m}(u)$ is not viable for many problems.
- It is difficult to model the steep flux gradients between material interfaces.
- The loss of corner-point information.
- Resultant one-dimensional fluxes $\tilde{\phi}_{g}^{m}(x)$, $\tilde{\phi}_{g}^{m}(y)$ cannot be used in heterogeneous flux reconstruction. In the usual method of pinwise flux reconstruction (form function method), $\phi_{g}^{m,het}(x,y) = \phi_{Ag}^{m}(x,y)\phi_{g}^{m}(x,y)$, where $\phi_{Ag}^{m}(x,y)$ is the single-assembly heterogeneous calculation solution (see Section 3.3.1) and $\phi_{g}^{m}(x,y)$ is the homogeneous flux distribution, $\phi_{g}^{m}(x,y)$ should be obtained from the results of nodal calculation. To obtain the distributional $\phi_{g}^{m}(x,y)$ requires extra work with approximations.
- For nonrectangular nodes (see Figure 3.5), singular terms in transverse leakage occur, which have to be approximated by smooth functions :

$$\begin{aligned}\tilde{L}(x) = {} & \frac{2}{\sqrt{3}}[J_{s}(x,y_{s}(x)) - J_{s}(x,-y_{s}(x))] \\ & -\frac{D}{\sqrt{3}}\mathrm{sgn}(x)\left[\frac{d}{dx}\hat{\phi}(x,y_{s}(x)) + \frac{d}{dx}\hat{\phi}(x,-y_{s}(x))\right] \\ & -\frac{D}{\sqrt{3}}\delta(x)[\hat{\phi}_{s}(x,y_{s}(x)) - \hat{\phi}_{s}(x,-y_{s}(x)).\end{aligned} \tag{3.36}$$

  In an effort to avoid the singularity, Chao [19] introduced an approach of conformal mapping from a hexagon to a rectangle and then applied the transverse integration. But due to the appearance of a space-dependent mapping scale function that multiplies the coefficients of the diffusion equation, the singularity still remains (although it is weaker).
- In some formulations, convergence problem arises.

3.3.3 Analytic Function Expansion Nodal Method (Without Transverse Integration)

The analytic function expansion nodal (AFEN) method that has been developed in the last several years at Korea Advanced Institute of Science and Technology (KAIST) [20-22] is based on the insights during the study of the modern nodal methods :

i) It is believed and generally true that guessing the shape of one-dimensional $\tilde{L}_{gu}^{m}(u)$ is easier than that of two-dimensional $\phi_{g}^{m}(x,y)$ or three-dimensional $\phi_{g}^{m}(x,y,z)$.

ii) Thus, it is necessary to do the transverse integration and to assume $\tilde{L}_{gu}^{m}(u)$.

iii) Because of the transverse integration, a singularity occurs purely from the geometry for some nodes, e.g., hexagonal nodes.

iv) We realized that we can find analytic solutions even for two- and three-dimensional elliptic equations such as the diffusion equations in Eq.(3.8). We can write down the analytic solutions at least in a summation of terms, each term itself satisfying the diffusion equation exactly, which has implications to the accuracy of the results.
v) Thus, we do not need transverse integration, which is a cause of several problems identified above in transverse integrated nodal methods.

Let us write two-dimensional diffusion equation for homogeneous node *m* as

$$-D_g^m \nabla^2 \hat{\phi}_g^m(x,y) + \Sigma_g^m \hat{\phi}_g^m(x,y) = \sum_{g'} \left(\frac{\chi_g}{k} \nu\Sigma_{fg'}^m + \Sigma_{gg'}^m \right) \hat{\phi}_{g'}^m(x,y). \tag{3.37}$$

Define λ_g^m as eigenvalues of $(D^m)^{-1}\left\{\Sigma^m - (1/k)\chi\nu\Sigma_f^m\right\}$ and R^m as matrix with columns of eigenvectors, then the diffusion equation is decoupled into

$$-\nabla^2 \hat{\xi}_g^m(x,y) + \lambda_g^m \hat{\xi}_g^m(x,y) = 0, \tag{3.38}$$

where $\hat{\xi}^m(x,y) = (R^m)^{-1}\hat{\phi}^m(x,y)$. We can write the solution in terms of analytic basis functions as

$$\hat{\xi}^m(x,y) = \sum_{i=0}^{\infty} \left\{ A_{gi}^{'} SN\kappa_g^m(a_{gxi}^m x + a_{gyi}^m y) + B_{gi}^{'} CS\kappa_g^m(a_{gxi}^m x + a_{gyi}^m y) \right\}, \tag{3.39}$$

where

$$\begin{aligned} \kappa_g^m &= \sqrt{|\lambda_g^m|} \\ (a_{gxi}^m)^2 + (a_{gyi}^m)^2 &= 1 \\ SN\kappa_g^m(a_{gxi}^m x + a_{gyi}^m y) &= \begin{cases} \sinh \kappa_g^m(a_{gxi}^m x + a_{gyi}^m y), & \lambda_g^m > 0 \\ \sin \kappa_g^m(a_{gxi}^m x + a_{gyi}^m y), & \lambda_g^m < 0 \end{cases} \\ CS\kappa_g^m(a_{gxi}^m x + a_{gyi}^m y) &= \begin{cases} \cosh \kappa_g^m(a_{gxi}^m x + a_{gyi}^m y), & \lambda_g^m > 0 \\ \cos \kappa_g^m(a_{gxi}^m x + a_{gyi}^m y), & \lambda_g^m < 0 \end{cases} \end{aligned} \tag{3.40}$$

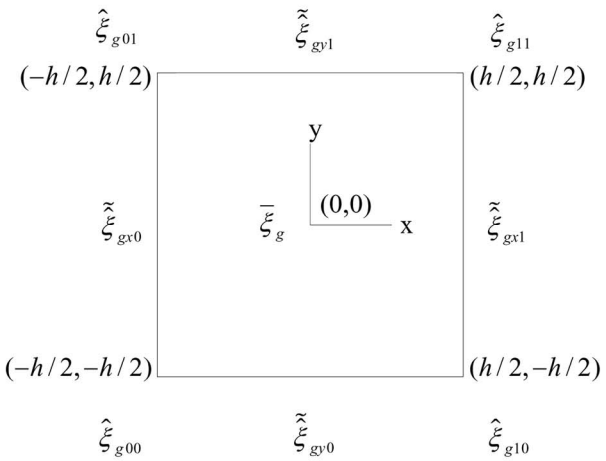


Figure 3.6 Nodal unknowns.

We now truncate the expansion with eight analytic basis functions and a constant :

$$\begin{aligned}\hat{\xi}_g^m(x,y) = C_g^m &+ A_{g1}^m SN\kappa_g^m x + A_{g2}^m CS\kappa_g^m x + A_{g3}^m SN\kappa_g^m y + A_{g4}^m CS\kappa_g^m y \\ &+ B_{g1}^m SN\frac{\sqrt{2}}{2}\kappa_g^m x SN\frac{\sqrt{2}}{2}\kappa_g^m y + B_{g2}^m SN\frac{\sqrt{2}}{2}\kappa_g^m x CS\frac{\sqrt{2}}{2}\kappa_g^m y \\ &+ B_{g3}^m CS\frac{\sqrt{2}}{2}\kappa_g^m x SN\frac{\sqrt{2}}{2}\kappa_g^m y + B_{g4}^m CS\frac{\sqrt{2}}{2}\kappa_g^m x CS\frac{\sqrt{2}}{2}\kappa_g^m y.\end{aligned} \tag{3.41}$$

The nine coefficients can be expressed in terms of nine nodal unknowns (one node-average flux, four node-interface fluxes, and four corner-point fluxes) shown in Figure 3.6. For example,

$$A_{g1}^m = \frac{\dfrac{\tilde{\hat{\xi}}_{gx1}^m - \tilde{\hat{\xi}}_{gx0}^m}{2} CS\frac{\sqrt{2}}{4}\kappa_g^m h - \dfrac{\hat{\xi}_{g11}^m + \hat{\xi}_{g10}^m - \hat{\xi}_{g01}^m - \hat{\xi}_{g00}^m}{4}\dfrac{2\sqrt{2}}{\kappa_g^m h} SN\frac{\sqrt{2}}{4}\kappa_g^m h}{SN\dfrac{\kappa_g^m h}{2}\left\{CS\dfrac{\sqrt{2}}{4}\kappa_g^m h - \dfrac{2\sqrt{2}}{\kappa_g^m h} SN\dfrac{\sqrt{2}}{4}\kappa_g^m h\right\}}. \tag{3.42}$$

Once all the coefficients of the Eq.(3.41) flux expansion are expressed in terms of the nodal unknowns and converted back to $\hat{\phi}^m(x,y) = R^m\hat{\xi}^m(x,y)$, we build as many solvable nodal coupling equations as the number of the unknowns to be determined. These are the nodal balance equations for each node, the continuity conditions on the neutron currents for interface-average fluxes, and the corner-point balance equations for corner-point fluxes. With all the nodal coupling equations for the nodal unknowns now available, a conventional iteration procedure, similar to the other nodal methods, may be used, that involves two levels of iterative schemes : inner iteration and outer iteration.

The AFEN method was applied to both rectangular and hexagonal core problems. All the results consistently indicate that AFEN outperforms the conventional nodal methods not only in core criticality but also in the flux distributions, especially in the region where large flux gradients exist. This superior performance of AFEN is mainly attributed to the fact that the nonseparable analytic function expansion of the intranodal flux distributions can accurately model the large flux gradients near the strong material discontinuity. Table 3.1 summarizes the key features of the AFEN method and Table 3.2 provides a comparison of problem size and convergence property in various nodal methods.

As for flux reconstruction, the results of the test, designed to demonstrate the ability of the AFEN method to reconstruct the pinwise flux distributions, show that the homogeneous flux distributions readily available from the AFEN calculation can be directly used in reconstructing heterogeneous pinwise flux distributions. Further developments are reported in [23-26].

Table 3.1 Key Features of the AFEN Method

Existing Nodal Methods	AFEN Method
Transverse integration ▸ Transverse leakage : quadratic ▸ Singularity : geometry	No transverse integration ▸ No need to assume TL shape ▸ Robust to geometry
Quadratic polynomial (1-D)	Analytic basis functions (2-D, 3-D)
Corner-point flux : not in the nodal coupling equations	Corner-point flux : in the nodal coupling equation

In discussing nodal methods, two additional works are worth noting. One is the PARCS code [27] which, for a cubic node problem, uses the ANM method and which, for a hexagonal prism problem, uses hybrid of axially NEM and radially triangular-based polynomial expansion nodal (TPEN) method, in which a hexagon is divided into six triangles. TPEN is an adaptation of the polynomial expansion nodal (PEN) method introduced in rectangular node problems [28-29]. The other is the so-called unified nodal method (UNM) [30,31], which is a collection of the NEM, ANM, and AFEN method in a transverse integration nodal method framework. The framework cannot handle the corner-point flux coupling equations in the AFEN method, so they should be considered separately, and thus the AFEN method is not “unified”. Moreover, the

transverse integration framework with the AFEN functions renders the iterative solution procedure susceptible to numerical instability and thus requires an extra underrelaxation scheme for convergence. The underrelaxation parameter is to be supplied by the user. If this is not properly given, the procedure could potentially lead to a false convergence.

Table 3.2 Comparison of Problem Size and Convergence Property in Various Methods (in Rectangular 2-D)

Method	Number of unknowns per node	Number of nodes per assembly	Spectral radius (μ)	Dominance ratio (σ)
FDM	1	17×17×22 =1156	Large	Large
NEM ANM	5(3*)	2×2=4	Significantly reduced	Slightly increase
AFEN	9(4*)	1	Significantly reduced	Slightly increase

* Net number of nodal unknowns per node in a core.

3.4 CMFD/CGR Acceleration Methods

3.4.1 Coarse Mesh Finite Difference (CMFD) Acceleration

As an alternative to the core-wise response matrix formalism solution scheme for the modern nodal methods, a nonlinear coarse mesh finite difference (CMFD) iterative solution scheme is widely used in the transverse integrated nodal methods such as NEM and ANM. This scheme turned out to be very effective in minimizing the memory requirement and computing time associated with higher-order nodal methods. It is also easily implemented into a mesh-centered FDM code. This nonlinear CMFD iterative scheme [32] uses the modified FDM current definition given for the interface of nodes m and m+1 with side length h,

$$\tilde{J}_g^{m+1/2} = -\frac{2D_g^m D_g^{m+1}}{h(D_g^m + D_g^{m+1})}(\bar{\phi}_g^{m+1} - \bar{\phi}_g^m) - \frac{\hat{D}_g^{m+1/2}}{h}(\bar{\phi}_g^{m+1} + \bar{\phi}_g^m), \tag{3.43}$$

instead of the original FDM definition, defined as the first term only on the right side. Here, D_g^m and $\bar{\phi}_g^m$ are the diffusion coefficient and the node-average flux of node m, respectively. The term $\hat{D}_g^{m+1/2}$ is the nonlinear correction factor defined at the interface of nodes m and m+1, which is a key element in the nonlinear CMFD iterative scheme. The bar ($-$) over a quantity denotes the quantity averaged over a node volume, and the tilde ($\sim$) denotes the quantity defined at a surface of a node. Although the nonlinear CMFD is usually referred to Ref.[33], Ref.[33] does not provide an expression of Eq.(3.43) or a similar one thereto, but Ref.[34] does.

Determining the nonlinear correction factor so that the interface current preserves the value of a higher-order nodal method makes the solution of this finite difference scheme equivalent to that of the higher-order nodal method itself. Note that the equivalence theory guarantees that even a lower-order method can reproduce the results of a higher-order nodal method if there exists a means for the lower-order method to preserve the interface currents of the higher-order method. For the nonlinear CMFD iterative scheme with the usual higher-order nodal methods that use the transverse integration, this is done by solving two-node problems consisting of neighboring nodes periodically, after a specified number of outer iterations of the FDM routine. Using the higher-order nodal method, the two-node problem is solved for the interface current of the two nodes with currently available node-average fluxes and transverse leakage shapes of both nodes as boundary conditions. The nonlinear correction factor at the interface is updated by equating the resultant higher-order interface current with the modified FDM current given by Eq.(3.43). Then, the FDM routine is continued, utilizing the updated nonlinear correction factor. The entire process is repeated until convergence of the effective multiplication factor and the node-average fluxes is achieved.

The process is summarized as follows :

Step 0 : Initialize $\hat{D}_g^{m+1/2}$ ($\hat{D}_g^{m+1/2} = 0$ or approximate $\hat{D}_g^{m+1/2}$ from some approximate core-wise

calculations).

Step 1 : Substitute Eq.(3.43) into Eq.(3.9) and solve the resulting finite difference equations to obtain $\bar{\phi}_g^m$ and $\tilde{J}_g^{m+1/2}$ for all nodes.

Step 2 : For a pair of two nodes (see Figure 3.7), solve the transverse integrated nodal equation by using $\bar{\phi}_g^m$, $\bar{\phi}_g^{m+1}$, and transverse leakages to obtain the interface net current $\tilde{J}_g^{m+1/2}$.

Step 3 : Repeat Step 2 for all pairs of two nodes in all directions.

Step 4 : Update $\hat{D}_g^{m+1/2}$ by Eq.(3.43).

Step 5 : Go to Step 1 until convergence.

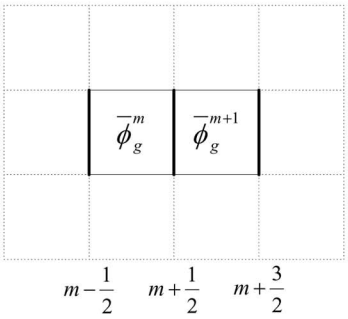


Figure.3.7 A scheme of the two-node nonlinear CMFD iteration.

3.4.2 Coarse Group Rebalance (CGR) Acceleration

Due to a larger number of nodal unknowns (addition of edge fluxes as nodal unknowns) of the AFEN method, computing time has been an issue of the method. To overcome this, a two-node AFEN scheme [35] was applied to two-dimensional geometry. As a more efficient iterative scheme for the AFEN method, a coarse group rebalance (CGR) method [36] is described in this section.

As in a standard way of any nodal method, the AFEN method is formulated in partial current form. All outgoing partial currents are written in terms of incoming partial currents and other nodal unknowns explicitly. The CGR acceleration scheme uses one correction factor per node to rebalance neutron sources during iteration, and the factor is used to balance the group integrated neutron source terms (outgoing partial currents, absorption, scattering and fission sources) in the node. The special structure ("nested" block symmetry) of the coefficient matrix of the outgoing partial currents allows their simultaneous update by using direct matrix inverse. Also, it is possible to use a simple update of the edge fluxes after rebalancing node neutron sources. The results show that the CGR acceleration scheme is very effective in reducing computing time and very stable in its convergence.

After constructing partial current formulation, we must derive rebalance equations. It starts from the neutron diffusion equation for node *m* :

$$-\begin{pmatrix} D_1^m & 0 \\ 0 & D_2^m \end{pmatrix}\nabla^2\begin{pmatrix} \phi_1^m \\ \phi_2^m \end{pmatrix}+\begin{pmatrix} \Sigma_{a1}^m+\Sigma_{1\to 2}^m & 0 \\ -\Sigma_{1\to 2}^m & \Sigma_{a2}^m \end{pmatrix}\begin{pmatrix} \phi_1^m \\ \phi_2^m \end{pmatrix}=\frac{1}{k}\begin{pmatrix} \nu\Sigma_{f1}^m & \nu\Sigma_{f2}^m \\ 0 & 0 \end{pmatrix}\begin{pmatrix} \phi_1^m \\ \phi_2^m \end{pmatrix}. \tag{3.44}$$

In the partial current formulation of a nodal method, Eq.(3.44) can be written as follows :

$$\sum_u \frac{1}{h_u^m}(\vec{J}_{ur}^{m,+}-\vec{J}_{ul}^{mr,-}-\vec{J}_{ul}^{m,-}+\vec{J}_{ur}^{ml,+})+\Sigma^m\vec{\phi}^m=\frac{1}{k}Q^m\vec{\phi}^m, \tag{3.45}$$

where

$$\Sigma^m=\begin{pmatrix} \Sigma_{a1}^m+\Sigma_{1\to 2}^m & 0 \\ -\Sigma_{1\to 2}^m & \Sigma_{a2}^m \end{pmatrix},\quad Q^m=\begin{pmatrix} \nu\Sigma_{f1}^m & \nu\Sigma_{f2}^m \\ 0 & 0 \end{pmatrix}.$$

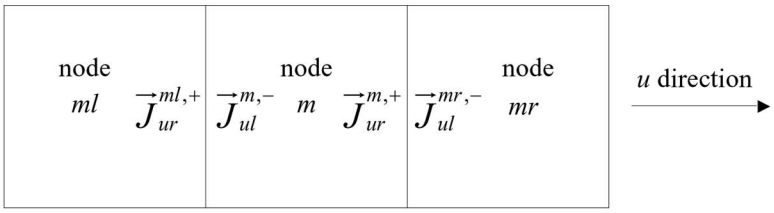


Figure 3.8 Partial currents related to node *m* in *u* direction in neutron balance equation.

By group summation of the neutron diffusion equations and using the partial current index in Figure 3.8, we can obtain the following equation :

$$R^m + \sum_{u=x,y,z} \frac{1}{h_u^m} \left\{ \sum_{s=l,r} \sum_{g=1,2} J_{usg}^{m,out} \right\} - \frac{1}{k} P^m = \sum_{u=x,y,z} \frac{1}{h_u^m} \sum_{g=1,2} \left\{ J_{ulg}^{mr,-} + J_{urg}^{ml,+} \right\}, \tag{3.46}$$

where

$$\sum_{s=l,r} \sum_{g=1,2} J_{usg}^{m,out} = J_{ur1}^{m,+} + J_{ur2}^{m,+} + J_{ul1}^{m,-} + J_{ul2}^{m,-},$$

$$P^m = \nu\Sigma_{f1}^m \phi_1^m + \nu\Sigma_{f2}^m \phi_2^m,$$

$$R^m = \Sigma_{a1}^m \phi_1^m + \Sigma_{a2}^m \phi_2^m.$$

Eq.(3.46) is an overall neutron balance equation for node *m*. The left-hand side of the equation is neutrons disappearing from node *m* through absorption, fission and outgoing currents, and the right-hand side is incoming neutrons from adjacent nodes. During iteration, the neutron balance equation is not satisfied, some nodes have more neutrons and other nodes less. We define a rebalance factor f^m for node *m* (same factor for each group), similarly to the case of coarse-mesh rebalance method in transport theory, to rebalance the neutrons :

$$\phi_g^m(x,y,z) = f^m \phi_{0,g}^m(x,y,z), \quad g = 1,2, \tag{3.47}$$

where

$\phi_{0,g}^m$: prebalanced flux (flux obtained from preceding AFEN outer iteration),

ϕ_g^m : rebalanced flux (flux obtained from coarse group rebalancing, used for the subsequent AFEN outer iteration).

Inserting Eq.(3.47) into Eq.(3.46), we obtain the following rebalance equations :

$$\left[R^m + \sum_{u=x,y,z} \frac{1}{h_u^m} \left\{ \sum_{s=l,r} \sum_{g=1,2} J_{usg}^{m,out} \right\} - \frac{1}{k} P^m \right] f^m = \sum_{u=x,y,z} \frac{1}{h_u^m} \sum_{g=1,2} \left\{ J_{ulg}^{mr,-} f^{mr} + J_{urg}^{ml,+} f^{ml} \right\}. \tag{3.48}$$

Eq.(3.48) has the same structure of FDM equations and can be solved as an eigenvalue (k) problem with the eigenvector (f^m). Calculated value of f^m is inserted into Eq.(3.47), that is, f^m is multiplied to all nodal unknowns in node *m*. Then, the rebalanced fluxes are used as the initial conditions for next AFEN outer iteration.

4. NUMERICAL METHODS FOR MULTIGROUP TRANSPORT EQUATIONS

Besides that the transport equation is the starting point to the derivation of the diffusion equation mostly used to obtain the knowledge of the neutron distribution in a reactor, usually it is the transport equation that is solved to use in the homogenization and condensation (at a cell level or an assembly level) prior to applying the diffusion equation (see Section 3.3.1). However, in a recent trend of more exotic and innovative reactor cores and fuels, we are strongly interested in applying the transport equation directly to the whole-core heterogeneous geometries. We may consider two most relevant basic methods for solving multigroup transport equations in this context: the discrete ordinates (S_N) method and the method of characteristics (MOC). Since the literature on the discrete ordinates method is vast and a recent seminal review article is

available [14], we describe in this section only the method of characteristics and a 2D/1D fusion method for 3-D whole-core transport calculation.

Due to the large amount of computing time required in whole-core transport calculations, effective acceleration methods are necessarily important. Thus this section also includes discussion of several acceleration methods, that are currently in use or contemplated to be used in the near future.

4.1 Method of Characteristics (MOC)

The method of characteristics [37, 38] is similar to the discrete ordinates method in that it considers only a finite number of discrete directions (with quadrature sets) and calculates mesh-average angular flux by sweeping. The key difference is in the way of calculating the mesh-average flux.

In MOC, for a given ordinate (direction of neutron travel traced), each of several rays parallel to the ordinate (that encompass the mesh) is traced to provide mesh-average ray angular flux and outgoing mesh-edge ray angular flux by analytic integration. These mesh-average ray angular fluxes are then summed for all the rays that pass through the mesh to provide mesh-average angular flux. The ray-wise integration allows flexibility of the mesh shapes. The meshes can take any shape and mixture of shapes as in Monte Carlo methods. This is in contrast to the discrete ordinates method.

4.1.1 Ray Tracing

To provide a key element of the method of characteristics, let us start from the within-group transport equation with the scattering term truncated after $l = L$ from Eq.(2.36) (dropping g) :

$$\vec{\Omega}\cdot\nabla\varphi(\vec{r},\vec{\Omega})+\sigma(\vec{r})\varphi(\vec{r},\vec{\Omega})=\sum_{l=0}^{L}\sigma_l(\vec{r})\sum_{m=-l}^{l}Y_{lm}(\vec{\Omega})\int d\vec{\Omega}'Y_{lm}^{*}(\vec{\Omega}')\varphi(\vec{r},\vec{\Omega}')+s(\vec{r},\vec{\Omega}), \tag{4.1}$$

where s contains neutron sources of group g from fission and scattering from other groups. Then major steps of the method of characteristics are as follows :

i) Choose a set of N discrete directions or rays $\vec{\Omega}_n$, $n=1,2,\cdots,N$, and corresponding quadrature weights $w_1,w_2,\cdots,w_N$ for numerical integrations over angle.

ii) Eq.(4.1) is evaluated at $\vec{\Omega}_n$:

$$\begin{aligned}\vec{\Omega}_n\cdot\nabla\varphi(\vec{r},\vec{\Omega}_n)+\sigma(\vec{r})\varphi(\vec{r},\vec{\Omega}_n)=&\sum_{l=0}^{L}\sigma_l(\vec{r})\sum_{m=-l}^{l}Y_{lm}(\vec{\Omega}_n)\int d\vec{\Omega}'Y_{lm}^{*}(\vec{\Omega}')\varphi(\vec{r},\vec{\Omega}')\\&+s(\vec{r},\vec{\Omega}_n)\end{aligned} \tag{4.2}$$

Let us now write the right-hand side of the within-group transport equation in Eq.(4.2) as $q_n(\vec{r})$, then we have

$$\vec{\Omega}_n\cdot\nabla\varphi_g(\vec{r},\vec{\Omega}_n)+\sigma_g(\vec{r})\varphi_g(\vec{r},\vec{\Omega}_n)=q_{g,n}(\vec{r}), \tag{4.3}$$

where

$$q_{g,n}(\vec{r})=\sum_{g'=1}^{G}\sum_{l=0}^{L}\sum_{m=-l}^{l}Y_{lm}(\vec{\Omega}_n)\sigma_{slgg'}(\vec{r})\phi_{lg'}^{m}(\vec{r})+\frac{\chi_g}{k}\sum_{g'=1}^{G}\nu\sigma_{fg'}(\vec{r})\phi_{g'}(\vec{r}), \tag{4.4}$$

$$\phi_{lg}^{m}(\vec{r})=\sum_{n=1}^{N}w_nY_{lm}^{*}(\vec{\Omega}_n)\varphi_g(\vec{r},\vec{\Omega}_n), \tag{4.5}$$

and $\{\vec{\Omega}_n,w_n\}$ are the chosen angular quadrature set. The angular quadrature set usually used in MOC is the product quadrature set (see Figure 4.1) to be discussed in detail later.

If we now assume that $q_{g,n}(\vec{r})$ is constant in a mesh (flat source approximation), then we have by analytic integration of Eq.(4.3),

$$\varphi_{g,n,l}^{out}=\varphi_{g,n,l}^{in}e^{-\sigma_gL_{n,l}/\sin\theta_n}+\frac{q_{g,n}}{\sigma_g}(1-e^{-\sigma_gL_{n,l}/\sin\theta_n}), \tag{4.6}$$

where $L_{n,l}$ is the track length of the l'th ray for direction $\vec{\Omega}_n$ in mesh (i,j), projected to the x-y plane in x-y-z (z-uniform, infinite) geometry. The mesh-average angular flux for the scattering source iteration is obtained by

$$\bar{\varphi}_{g,n} = \frac{q_{g,n}}{\sigma_g} + \frac{\sin\theta_n}{A_{ij}\sigma_g} \sum_{l \in ij'th\ mesh} \delta_n (\varphi^{in}_{g,n,l} - \varphi^{out}_{g,n,l}), \tag{4.7}$$

where δ_n is the spacing between adjacent rays for direction $\vec{\Omega}_n$ and A_{ij} is the area of mesh (i, j). To preserve the area A_{ij}, the projected track length $L_{n,l}$ is renormalized as follows :

$$L^*_{n,l} = L_{n,l} \frac{A_{ij}}{\sum_l \sum_n L_{n,l} \delta_n w^{\varphi}_n}, \tag{4.8}$$

where w^{φ}_n are the weights for azimuthal angles. Then Eqs.(4.6) and (4.7) and a general geometric tracking module (see Figure 4.2) are complete for transport calculation with MOC.

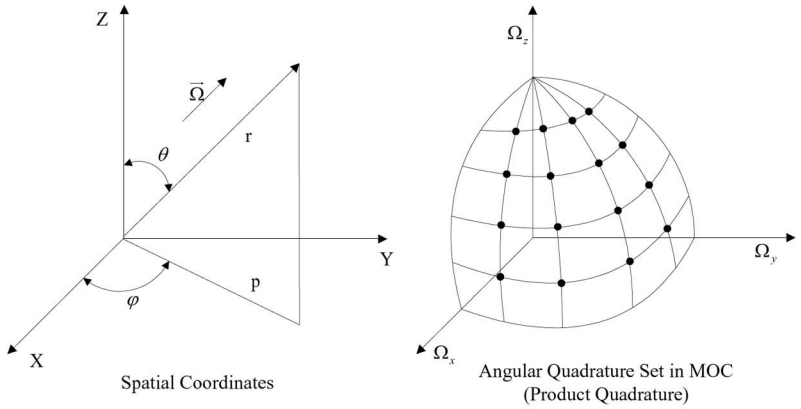


Figure 4.1 Angular quadrature set in MOC.

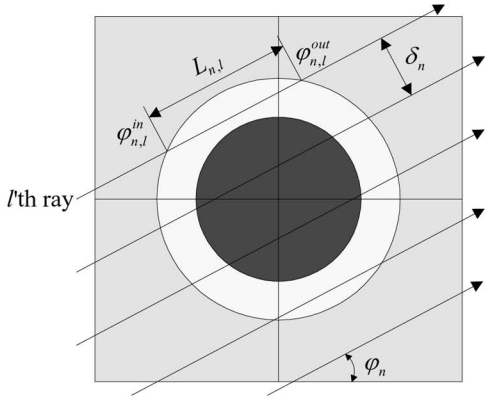
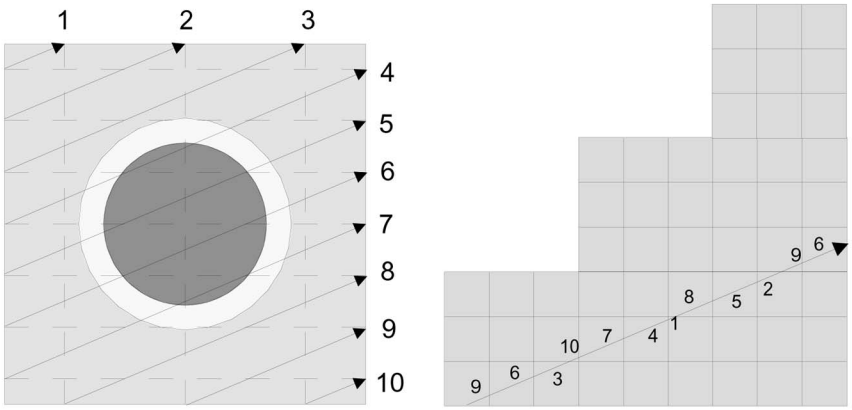


Figure 4.2 Geometric tracking module in MOC.

The capability of modular ray tracing is very desirable because it facilitates the whole-core ray sweeping. The modular ray tracing is a general geometric tracking procedure for determining track lengths and region numbers for unique cell types only, saving computer memory requirements and floating-point operations. The product quadrature set to be described in the next section leads to a modular ray tracing system with self-repeating ray distributions and returning to the same starting point after reflection. This makes whole-core ray sweeping possible with MOC without approximation on interfaces between modules (see Figure 4.3).



(a) Modular ray tracing (b) Direct ray sweeping of whole core

Figure 4.3 Whole-core ray sweeping in MOC.

4.1.2 Product Quadrature

A product quadrature set that allows modular ray tracing is provided as follows. The polar angles and azimuthal angles can be chosen independently. Leonard & McDaniel [39] provides an optimal scheme for polar angles such that the discrepancy in the region-to-region collision rates between a continuous form and a chosen discrete form is minimized. For example, in the case of two polar angles in an octant, their scheme gives the following optimal polar quadrature set :

$$\sin\theta_1 = 0.273658, \qquad w_1 = 0.139473, \tag{4.9}$$

$$\sin\theta_2 = 0.865714, \qquad w_2 = 0.860527. \tag{4.10}$$

The azimuthal quadrature that can be chosen is constrained by the requirement of modular ray tracing. The following is proposed by Filippone [40] :

$$\tan(\varphi_r) = (\Delta y/\Delta x)\times\frac{r}{(N_\varphi+1-r)}, \quad r = 1,2,\cdots,N_\varphi, \tag{4.11}$$

$$\delta_r = \frac{\Delta x}{r}\sin\varphi_r = \frac{\Delta y}{N_\varphi+1-r}\cos\varphi_r, \tag{4.12}$$

where $\Delta x,\ \Delta y$: size of a unit cell, N_φ : number of azimuthal angles, and $N_l(=N_\varphi+1)$ rays for each of N_φ azimuthal angles, with weights given as

$$\sum_r^{N_\varphi} w_r^\varphi \cos(2s\varphi_r) = \begin{cases} \pi/2 & \text{if } s = 0, \\ 0 & \text{if } s = 1,2,\cdots,N_\varphi - 1. \end{cases} \tag{4.13}$$

The CRX code [41] developed at KAIST uses only a subset N_φ' (typically 4 or 8, $< N_\varphi$) angles, achieving acceptable accuracy without excessive number of ordinates.

4.2 Whole-Core Transport Calculation

Although several 3-D deterministic transport methods have been developed, it is not easy to apply them to realistic whole-core problems [42-45] due to accuracy and memory problems. Among them, the discrete ordinates method is the most widely used to solve the 3-D transport equation. However, this method has difficulty in treating curvilinear geometry arising from heterogeneity. During the last several years, the method of characteristics (MOC) has been revisited and refined to apply to 2-D whole-core calculations [42-44]. MOC is known to provide accurate solutions in complex geometries and strongly heterogeneous problems.

A direct extension of MOC may be possible, but it will take tremendous amount of memory and long computing time. Recently, 2D/1D fusion method [46, 47] was devised for 3-D problems, that is, a synergistic combination of the method of characteristics for radial 2-D calculation and the S_N-like method for axial 1-D calculation. This approach takes advantage of the structure of a core that is usually simpler in axial direction than in radial direction. Each solver sends and receives

interface angular fluxes and thus requires some iterations. However, this new approach requires less memory, resulting in overall efficiency, and may constitute a practical method for the 3-D whole-core heterogeneous transport calculation.

As related works that followed are two hybrid methods [48, 49]. These methods use basically 1D diffusion calculations to provide the interface information for 2D MOC calculations, thus they are short of 3D transport methods.

4.2.1 2D/1D Fusion Method

Let us consider the following directional form of the within-group neutron transport equation:

$$\vec{\Omega}_n \cdot \nabla \varphi^g(\vec{r},\vec{\Omega}_n) + \sigma^g(\vec{r}) + \varphi^g(\vec{r},\vec{\Omega}_n) = q^g(\vec{r},\vec{\Omega}_n), \tag{4.14}$$

where standard notations are used. For 3-D cartesian coordinates, we obtain the following equation by integrating Eq.(4.14) over z-direction interval $(z_{k-1/2}, z_{k+1/2})$ of cell (i,j,k) plane:

$$\begin{aligned}\mu_n \frac{\partial}{\partial x}\varphi^g_{n,k}(x,y) + \eta_n \frac{\partial}{\partial y}\varphi^g_{n,k}(x,y) + \frac{\xi_n}{\Delta_k}\left[\varphi^g_{n,k+1/2}(x,y) - \varphi^g_{n,k-1/2}(x,y)\right] \\ + \sigma^g_k(x,y)\varphi^g_{n,k}(x,y) = q^g_{n,k}(x,y),\end{aligned} \tag{4.15}$$

where

$$\varphi^g_{n,k}(x,y) = \frac{1}{\Delta_k}\int_{z_{k-1/2}}^{z_{k+1/2}} dz \varphi^g_n(x,y,z),$$

$$\varphi^g_{n,k\pm1/2}(x,y) = \varphi^g_n(x,y,z_{k\pm1/2}), \quad \Delta_k = z_{k+1/2} - z_{k-1/2},$$

$$q^g_{n,k}(x,y) = \frac{1}{\Delta_k}\int_{z_{k-1/2}}^{z_{k+1/2}} dz\ q^g_n(x,y,z),$$

$$\sigma^g_k(x,y) = \frac{1}{\varphi^g_{n,k}(x,y)\Delta_k}\int_{z_{k-1/2}}^{z_{k+1/2}} dz\ \sigma^g(x,y,z)\varphi^g_n(x,y,z).$$

Note that we consider in piecewise homogeneous properties in axial direction cell by cell. Rewriting Eq.(4.15), the equation can be given in two-dimensional form as

$$\mu_n \frac{\partial}{\partial x}\varphi^g_{n,k}(x,y) + \eta_n \frac{\partial}{\partial y}\varphi^g_{n,k}(x,y) + \sigma^g_k(x,y)\varphi^g_{n,k}(x,y) = Q^g_{n,k}(x,y), \tag{4.16}$$

where

$$Q^g_{n,k}(x,y) = q^g_{n,k}(x,y) - \frac{\xi_n}{\Delta_k}\left[\varphi^g_{n,k+1/2}(x,y) - \varphi^g_{n,k-1/2}(x,y)\right]. \tag{4.17}$$

Eq.(4.16) is solved by the method of characteristics with heterogeneous geometry maintained. But the source term is modified by the z-directional angular fluxes in Eq.(4.17) at cell axial interfaces. To provide these interface angular fluxes, we now derive an axial 1D equation. Integrating Eq.(4.14) over x-y domain $(x_{i-1/2}, x_{i+1/2})$ and $(y_{j-1/2}, y_{j+1/2})$ of cell (i,j,k) column, we obtain

$$\begin{aligned}\frac{\mu_n}{\Delta_i}\left[\varphi^g_{n,i+1/2,j}(z) - \varphi^g_{n,i-1/2,j}(z)\right] + \frac{\eta_n}{\Delta_j}\left[\varphi^g_{n,i,j+1/2}(z) - \varphi^g_{n,i,j-1/2}(z)\right] \\ + \xi_n \frac{d}{dz}\varphi^g_{n,i,j}(z) + \sigma^g_{n,i,j}(z)\varphi^g_{n,i,j}(z) = q^g_{n,i,j}(z),\end{aligned} \tag{4.18}$$

where

$$\varphi^g_{n,i\pm1/2,j}(z) = \frac{1}{\Delta_j}\int_{y_{j-1/2}}^{y_{j+1/2}} dy\ \varphi^g_n(x_{i\pm1/2},y,z),$$

$$\varphi^g_{n,i,j\pm1/2}(z) = \frac{1}{\Delta_i}\int_{x_{i-1/2}}^{x_{i+1/2}} dx\ \varphi^g_n(x,y_{j\pm1/2},z),$$

$$\varphi^g_{n,i,j}(z) = \frac{1}{\Delta_i}\int_{x_{i-1/2}}^{x_{i+1/2}} dx \frac{1}{\Delta_j}\int_{y_{j-1/2}}^{y_{j+1/2}} dy\ \varphi^g_n(x,y,z),$$

$$\sigma^g_{n,i,j}(z) = \frac{1}{\varphi^g_{n,i,j}(z)}\frac{1}{\Delta_i}\int_{x_{i-1/2}}^{x_{i+1/2}} dx \frac{1}{\Delta_j}\int_{y_{j-1/2}}^{y_{j+1/2}} dy\ \sigma^g(x,y,z)\varphi^g_n(x,y,z).$$

Rewriting Eq.(4.18), we obtain then

$$\xi_n \frac{d}{dz}\varphi^g_{n,i,j}(z) + \sigma^g_{n,i,j}(z)\varphi^g_{n,i,j}(z) = Q^g_{n,i,j}(z), \tag{4.19}$$

where

$$Q^g_{n,i,j}(z) = q^g_{n,i,j}(z) - \frac{\mu_n}{\Delta_i}\left[\varphi^g_{n,i+1/2,j}(z) - \varphi^g_{n,i-1/2,j}(z)\right] + \frac{\eta_n}{\Delta_j}\left[\varphi^g_{n,i,j+1/2}(z) - \varphi^g_{n,i,j-1/2}(z)\right]. \tag{4.20}$$

To solve Eq.(4.19), we integrate Eq.(4.19) over z-directional interval $(z_{k-1/2}, z_{k+1/2})$ of cell (*i,j,k*) resulting in a balance equation,

$$\frac{\xi_n}{\Delta_k}(\varphi^g_{n,i,j,k+1/2} - \varphi^g_{n,i,j,k-1/2}) + \sigma^g_{n,i,j,k}\varphi^g_{n,i,j,k} = Q^g_{n,i,j,k}, \tag{4.21}$$

where

$$\varphi^g_{n,i,j,k\pm1/2} = \varphi^g_{n,i,j}(z_{k\pm1/2}),$$

$$\varphi^g_{n,i,j,k} = \frac{1}{\Delta_k}\int_{z_{k-1/2}}^{z_{k+1/2}} dz\ \varphi^g_{n,i,j}(z),$$

$$\begin{aligned}\sigma^g_{n,i,j,k} &= \frac{1}{\varphi^g_{n,i,j,k}}\frac{1}{\Delta_k}\int_{z_{k-1/2}}^{z_{k+1/2}} dz\ \sigma^g_{n,i,j}(z)\varphi^g_{n,i,j}(z) \\ &= \frac{1}{\varphi^g_{n,i,j,k}}\frac{1}{\Delta_i\Delta_j\Delta_k}\int_{z_{k-1/2}}^{z_{k+1/2}} dz\int_{x_{i-1/2}}^{x_{i+1/2}} dx\int_{y_{j-1/2}}^{y_{j+1/2}} dy\ \sigma^g(x,y,z)\varphi^g_n(x,y,z).\end{aligned}$$

Since we consider piecewise homogeneous properties in axial direction, $\sigma^g_{n,i,j,k}$ can be represented as

$$\sigma^g_{n,i,j,k} = \frac{\sum_{m\in cell(i,j,k)} \sigma^g_k(m)\varphi^g_{n,k}(m)A_m}{\sum_{m\in cell(i,j,k)} \varphi^g_{n,k}(m)A_m}, \tag{4.22}$$

where A_m is the area of radial computational mesh m and the average angular fluxes are obtained from the results of 2D MOC calculation.

To evaluate Eq.(4.22), it is necessary to save the angle-dependent total cross sections and use additional variables for the average angular fluxes of the computational meshes in addition to the average scalar fluxes. If we like to avoid this extra requirement, Eq.(4.22) may be approximated by

$$\sigma^g_{n,i,j,k} = \hat{\sigma}^g_{i,j,k}, \tag{4.23}$$

where

$$\hat{\sigma}^g_{i,j,k} = \frac{\sum_{m\in cell(i,j,k)} \sigma^g_k(m)\sum_n w_n\varphi^g_{n,k}(m)A_m}{\sum_{m\in cell(i,j,k)}\sum_n w_n\varphi^g_{n,k}(m)A_m} = \frac{\sum_{m\in cell(i,j,k)} \sigma^g_k(m)\phi^g_k(m)A_m}{\sum_{m\in cell(i,j,k)} \phi^g_k(m)A_m}.$$

Now the axial 1D equation (4.21) can be solved by various S_N-like methods using various auxiliary equations, e.g., diamond difference (DD), linear discontinuous (LD), or linear multiple balance (LMB) scheme. A simple choice is the DD scheme:

$$\varphi^{g}_{n,i,j,k} = \frac{1}{2}(\varphi^{g}_{n,i,j,k+1/2} + \varphi^{g}_{n,i,j,k-1/2}). \tag{4.24}$$

The resulting z-directional interface angular fluxes are then used to update the modifying source term in Eq.(4.16). A calculational flow is shown in Figure 4.4, which also implements p-CMFD acceleration to be described in Section 4.3.2.

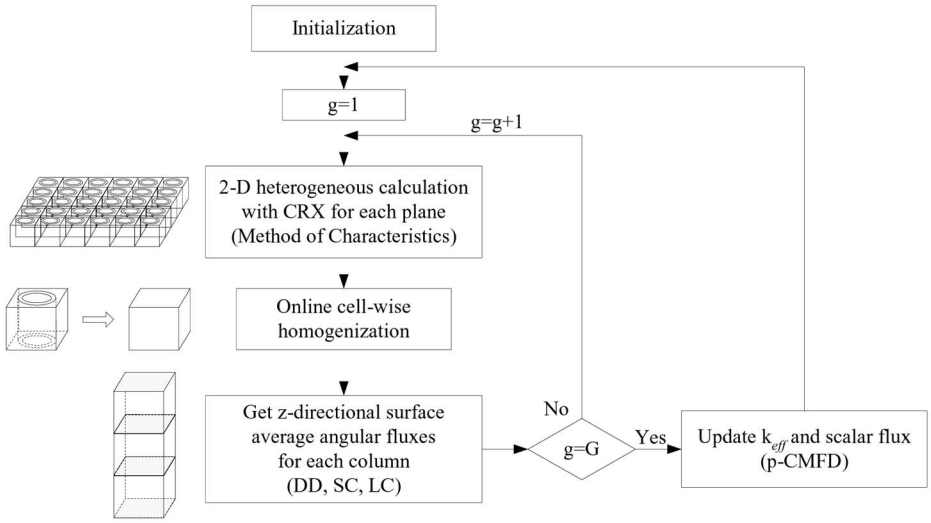


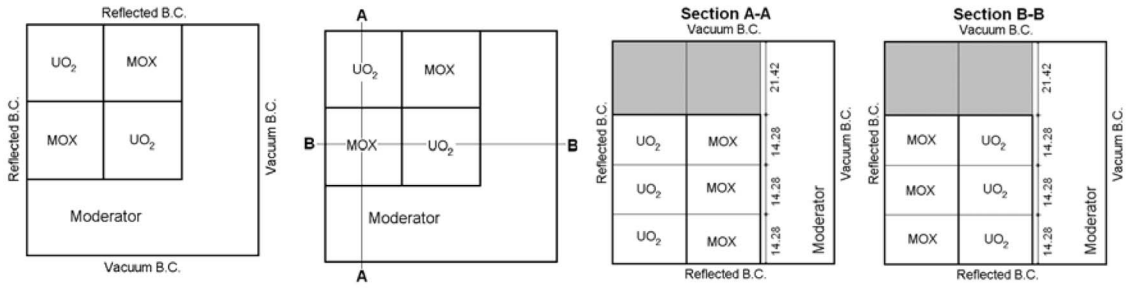
Figure 4.4 Calculational flow in the 2D/1D fusion method.

4.2.2. Numerical Results on OECD Benchmark Problem C5G7MOX

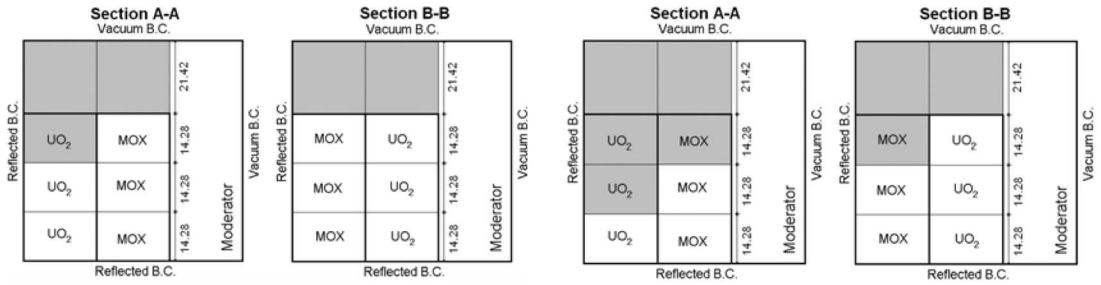
The OECD benchmark problem C5G7MOX consists of 3 configurations ; Unrodded, Rodded A, and Rodded B. The fuel cell is heterogeneous, that is, there is no fuel/coolant homogenization. In Unrodded configuration, there are no control rods in fuel assemblies but control rods are inserted only in the top reflector. In Rodded A configuration, control rods are inserted in the reflector and in 1/3 depth of the inner UO_2 assembly. In Rodded B configuration, control rods are inserted in 2/3 depth of the inner UO_2 assembly and 1/3 depth of the two MOX assemblies. Figure 4.5 shows the configurations of the benchmark problem with, control rods represented in shades. Their detail configurations with 7-group cross sections are described in Ref.[50].

In these calculations, we used 8 azimuthal/2 polar angles per octant (the 2 polar angles described as optimal in Ref.[39]), 50 rays per direction and cell. Each cell has 32 radial computational meshes and 18 axial meshes (3.57cm/axial mesh). Table 4.1 shows the results of the three configurations of the extended OECD benchmark problem, compared with the reference solution provided by OECD using Monte Carlo calculation.

The multiplication factors (k_{eff}) from the CRX code with 2D/1D fusion method are very close to those of the reference solution ; largest discrepancy is -0.065% for Rodded B configuration. In pinpower distributions, the largest discrepancy (-1.6%) in maximum pinpower occurs in slice 3 of Rodded A configuration.



(a) Top view of the benchmark problem (b) Unrodded configuration



(c) Rodded A configuration (d) Rodded B configuration

Figure 4.5 Three configurations of the benchmark problem. (shaded areas represent control rod insertion depth)

Table 4.1 Numerical Results of the Extended OECD Benchmark Problem

	Unrodded	Rodded A	Rodded B
k_{eff}	1.14335 (1.14308)*	1.12810 (1.12806)	1.07707 (1.07777)
slice1(a) Max. pinpower	1.109 (1.108)	1.191 (1.197)	1.182 (1.200)
slice2(b) Max. pinpower	0.882 (0.882)	0.829 (0.832)	0.556 (0.554)
slice3(c) Max. pinpower	0.490 (0.491)	0.309 (0.304)	0.218 (0.217)

* Values in parentheses are reference values provided by OECD
(a) Max. pinpower in 0.00<z<14.28 cm
(b) Max. pinpower in 14.28<z<28.56 cm
(c) Max. pinpower in 28.56<z<42.84 cm

4.3 Acceleration Methods

The within-group transport equation either in the discrete ordinates method or in the method of characteristics is solved by iterative methods. A simple iterative method is scattering source iteration (SI) that can be symbolically written as

$$\vec{\Omega}\cdot\nabla\varphi^{l+1}(\vec{r},\vec{\Omega})+\sigma(\vec{r})\varphi^{l+1}(\vec{r},\vec{\Omega})=\sigma_s(\vec{r})\phi^{l}(\vec{r})+q(\vec{r},\vec{\Omega}), \tag{4.25}$$

$$\phi^{l+1}(\vec{r})=\int d\vec{\Omega}\varphi^{l+1}(\vec{r},\vec{\Omega}), \tag{4.26}$$

where l is an iteration index.
For problems in which neutrons undergo few collisions or for problems that are “leaky”, the SI method converges rapidly. However, for problems that contain diffusive regions that are optically thick and scattering-dominated ($c=\sigma_s/\sigma\rightarrow 1$), the SI method converges very slowly. To accelerate the convergence, various acceleration methods are used in the following structure :

$$\vec{\Omega}\cdot\nabla\varphi^{l+1/2}(\vec{r},\vec{\Omega})+\sigma(\vec{r})\varphi^{l+1/2}(\vec{r},\vec{\Omega})=\sigma_s(\vec{r})\phi^{l}(\vec{r})+q(\vec{r},\vec{\Omega}), \tag{4.27}$$

$$\phi^{l+1}(\vec{r})=\mathcal{F}[\varphi^{l+1/2}(\vec{r},\vec{\Omega})]. \tag{4.28}$$

Eq.(4.27) is called high-order equation, which is the original transport equation for which solution is sought. The solution $\varphi^{l+1/2}$ is usually obtained by transport sweep. Eq.(4.28) is called acceleration or low-order equation, which is simpler than high-order equation. The solution ϕ^{l+1} is then substituted in the right-hand side of Eq.(4.27) and the process continues until ϕ^{l+1} converges. For details, see a recent review paper by Adams & Larsen [14].

4.3.1 Diffusion Synthetic Acceleration (DSA)

The most widely used acceleration method is based on the diffusion type operation for $\mathcal{F}$. In continuous form, they are given as follows :

Transport sweep (High-order equation) :

$$\vec{\Omega}\cdot\nabla\varphi^{l+1/2}(\vec{r},\vec{\Omega})+\sigma(\vec{r})\varphi^{l+1/2}(\vec{r},\vec{\Omega})=\sigma_s(\vec{r})\phi^{l}(\vec{r})+q(\vec{r},\vec{\Omega}), \tag{4.27}$$

$$\phi^{l+1/2}(\vec{r})=\int d\vec{\Omega}\varphi^{l+1/2}(\vec{r},\vec{\Omega}). \tag{4.29}$$

DSA equation (Low-order equation) :

$$-\nabla\cdot\frac{1}{3\sigma(\vec{r})}\nabla F^{l+1}(\vec{r})+\sigma_a(\vec{r})F^{l+1}(\vec{r})=\sigma_s(\vec{r})[\phi^{l+1/2}(\vec{r})-\phi^{l}(\vec{r})], \tag{4.30}$$

where

$$\sigma_a(\vec{r})=\sigma(\vec{r})-\sigma_s(\vec{r}),$$

$$F^{l+1}(\vec{r})=\phi^{l+1}(\vec{r})-\phi^{l+1/2}(\vec{r}).$$

Thus, $\phi(\vec{r})$ is updated by

$$\phi^{l+1}(\vec{r})=\phi^{l+1/2}(\vec{r})+F(\vec{r}). \tag{4.31}$$

The discretization in Eq.(4.30) must be "consistent" with that in Eq.(4.27) for unconditional stability (convergence of solutions for all mesh sizes, including optically thick meshes). The spatially differenced diffusion equation must be derived directly from the spatially differenced S_N equation. The following is a consistently discretized equations in 1-D geometry for diamond scheme in the high-order equation :

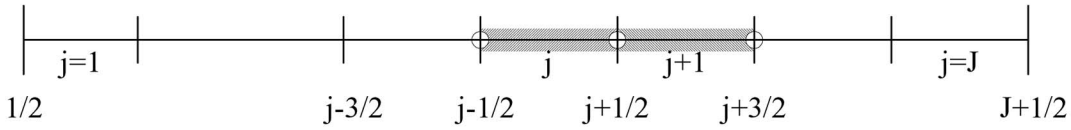


Figure 4.6 Three-point mesh-edge discretization.

High-order equation :

$$\frac{\mu_n}{h_n}(\varphi^{l+1/2}_{n,j+1/2}-\varphi^{l+1/2}_{n,j-1/2})+\sigma_j\varphi^{l+1/2}_{n,j}=\frac{\sigma_{sj}}{2}\phi^{l}_{j}+\frac{q_j}{2}, \tag{4.32}$$

$$\varphi^{l+1/2}_{n,j}=\frac{1}{2}(\varphi^{l+1/2}_{n,j+1/2}+\varphi^{l+1/2}_{n,j-1/2}), \tag{4.33}$$

$$\phi^{l+1/2}_{j}=\sum_{n=1}^{N}w_n\varphi^{l+1/2}_{n,j}. \tag{4.34}$$

DSA (Low-order) equation :

$$-\frac{1}{3\sigma_{j+1}h_{j+1}}(F^{l+1}_{j+3/2}-F^{l+1}_{j+1/2})+\frac{1}{3\sigma_j h_j}(F^{l+1}_{j+1/2}-F^{l+1}_{j-1/2})$$
$$+\frac{\sigma_{a,j+1}h_{j+1}}{4}(F^{l+1}_{j+3/2}+F^{l+1}_{j+1/2})+\frac{\sigma_{a,j}h_j}{4}(F^{l+1}_{j+1/2}+F^{l+1}_{j-1/2}) \tag{4.35}$$
$$=\frac{\sigma_{s,j+1}h_{j+1}}{2}(\phi^{l+1/2}_{j+1}-\phi^{l}_{j+1})+\frac{\sigma_{s,j}h_j}{2}(\phi^{l+1/2}_{j}-\phi^{l}_{j}),$$

$$\phi^{l+1}_j=\phi^{l+1/2}_j+\frac{1}{2}(F^{l+1}_{j+1/2}+F^{l+1}_{j-1/2}). \tag{4.36}$$

Note that Eq.(4.35) is a mesh-edge scheme and that the removal (σ_a) term is based on a three-point, rather than a conventional one-point mesh-edge (inconsistent) discretization. Therefore, in 2-D problems it becomes a nine-point discretization. Thus, the consistent derivation of such diffusion difference schemes is algebraically complicated in multidimensional geometries and non-diamond difference schemes and potentially difficult to solve. Figure 4.7 shows the spectral radii for DSA methods obtained from Fourier analysis on a model problem of infinite, homogeneous-medium, and uniform-mesh system.

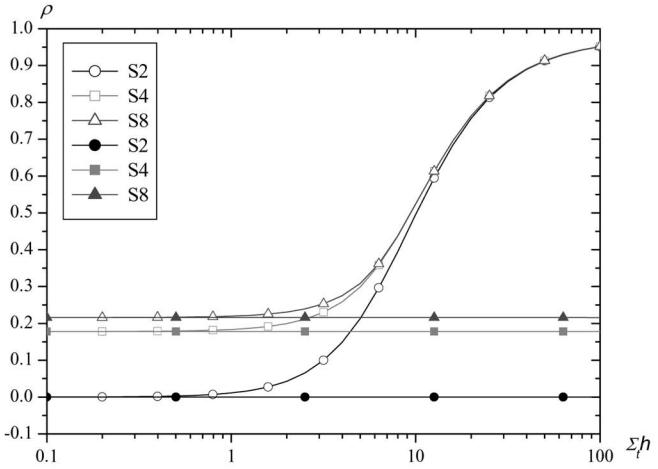


Figure 4.7 ρ_{DSA} versus $\Sigma_t h$. (c=0.98, "Inconsistent" and "Consistent" DSA diffusion discretizations) (From Ref.[14])

Recent studies [51, 52] show that the computation efficiency of S_N calculations with DSA methods lose their effectiveness when the problem contains strong discontinuities in the material properties. But when used to precondition a Krylov subspace method, as opposed to accelerating some iteration, a DSA method can be applied efficiently even if the discretization is not fully consistent [53].

4.3.2 Partial Current-Based Coarse Mesh Finite Difference (p-CMFD) Acceleration

We are particularly interested in a coarse-mesh acceleration of some kinds, since it is easy to overlay simplex coarse meshes over the unstructured fine meshes and work with acceleration equations in a simple, coarse mesh structure.

The coarse mesh finite difference (CMFD) acceleration method that is popular in the fast solution of nodal diffusion equations [32, 33], has been employed for the acceleration of the transport calculations with very good results [54-57], although the transport calculations were limited to the 2D problems. In the CMFD method, a current correction coefficient is introduced to preserve the interface net current (obtained from the transport sweep) between two coarse meshes. It is known that the convergence of CMFD is very fast in optically thin meshes but it becomes poor or divergent in optically thick meshes [58]. This may be due to the weak physical basis of the way the correction coefficient is introduced.

Recently, a partial current-based coarse mesh finite difference (p-CMFD) acceleration method [59], in which two correction coefficients are introduced at an interface between two coarse meshes such that partial currents are preserved, has been introduced, that shows better convergence than CMFD.

In the original CMFD method, the current at the right edge (i+1/2) of coarse mesh i is defined with a correction factor $\widehat{D}_{i+1/2}$ as

$$J_{i+1/2}^{l+1} = -\widetilde{D}_{i+1/2}\left(\phi_{i+1}^{l+1} - \phi_i^{l+1}\right) - \widehat{D}_{i+1/2}(\phi_i^{l+1} + \phi_{i+1}^{l+1}), \tag{4.37}$$

$$\widehat{D}_{i+1/2} = -\frac{J_{i+1/2}^{l+1/2} + \widetilde{D}_{i+1/2}\left(\phi_{i+1}^{l+1/2} - \phi_i^{l+1/2}\right)}{\phi_i^{l+1/2} + \phi_{i+1}^{l+1/2}}, \tag{4.38}$$

where $\widehat{D}_{i+1/2}$ is the coupling coefficient determined in the ordinary finite difference method and l is an iteration index. The current $J_{i+1/2}^{l+1/2}$ in Eq.(4.38) is obtained from transport calculation as follows :

$$J_{i+1/2}^{l+1/2} = J_{i+1/2}^{+,l+1/2} - J_{i+1/2}^{-,l+1/2}, \tag{4.39}$$

where

$$J_{i+1/2}^{+,l+1/2} = \sum_{\mu_n>0} w_n \mu_n \psi_{n,i+1/2}^{l+1/2}, \tag{4.39a}$$

$$J_{i+1/2}^{-,l+1/2} = \sum_{\mu_n<0} w_n \mid \mu_n \mid \psi_{n,i+1/2}^{l+1/2}. \tag{4.39b}$$

In contrast, in p-CMFD acceleration, partial currents at the right edge (i+1/2) of coarse mesh i are corrected with the CMR concept ; outgoing and incoming partial currents of any mesh cell are related with the corresponding cell average scalar fluxes, respectively, as

$$J_{i+1/2}^{+,l+1} = \frac{-\widetilde{D}_{i+1/2}\left(\phi_{i+1}^{l+1} - \phi_i^{l+1}\right) + 2\widehat{D}_{i+1/2}^{+}\phi_i^{l+1}}{2}, \tag{4.40}$$

$$J_{i+1/2}^{-,l+1} = \frac{\widetilde{D}_{i+1/2}\left(\phi_{i+1}^{l+1} - \phi_i^{l+1}\right) + 2\widehat{D}_{i+1/2}^{-}\phi_{i+1}^{l+1}}{2}, \tag{4.41}$$

and the two correction factors are defined to preserve the respective partial currents as

$$\widehat{D}_{i+1/2}^{+} = \frac{2J_{i+1/2}^{+,l+1/2} + \widetilde{D}_{i+1/2}\left(\phi_{i+1}^{l+1/2} - \phi_i^{l+1/2}\right)}{2\phi_i^{l+1/2}}, \tag{4.42}$$

$$\widehat{D}_{i+1/2}^{-} = \frac{2J_{i+1/2}^{-,l+1/2} - \widetilde{D}_{i+1/2}\left(\phi_{i+1}^{l+1/2} - \phi_i^{l+1/2}\right)}{2\phi_{i+1}^{l+1/2}}, \tag{4.43}$$

where the partial currents $J_{i+1/2}^{+,l+1/2}$ and $J_{i+1/2}^{-,l+1/2}$ are obtained from Eqs.(4.39a) and (4.39b), respectively. Then, the net current is obtained from Eqs.(4.40) and (4.41) as

$$J_{i+1/2}^{l+1} = J_{i+1/2}^{+,l+1} - J_{i+1/2}^{-,l+1} = -\widetilde{D}_{i+1/2}\left(\phi_{i+1}^{l+1} - \phi_i^{l+1}\right) - (\widehat{D}_{i+1/2}^{-}\phi_{i+1}^{l+1} - \widehat{D}_{i+1/2}^{+}\phi_i^{l+1}). \tag{4.44}$$

The net current relation Eq.(4.37) or Eq.(4.44) is then substituted into Eq.(3.9) with conventional flux-volume weighted cross sections, resulting in a finite difference form of equations.

It is interesting to note that either CMFD using Eq.(4.37) or p-CMFD using Eq.(4.44) is equivalent to a low-order equation of diffusion-type with, in effect, directional diffusion coefficients D^{x+} and D^{x-} discussed in Section 3.3.1. It does not require to deal with discontinuity factors or surface flux discontinuity directly. Because of the directionality of the diffusion coefficients, the low-order equation becomes a nonsymmetric matrix system for which a Krylov subspace method is a good solver.

The computational procedures of the p-CMFD method are similar to the original CMFD method and do not require any extra computation. In the p-CMFD method, net currents are automatically preserved and, since outgoing or incoming

partial current is corrected to be preserved by its own coarse mesh flux, it is more physically based. This indicates that p-CMFD should perform better in problems with strong flux gradients. Furthermore, the availability of the partial currents as part of the solution may be utilized for other uses.

To predict convergence behavior of the p-CMFD method, linearized Fourier analysis was performed for an infinite homogeneous one-dimensional model problem with the diamond difference (DD) discretization scheme for various *p*'s, the number of fine meshes per coarse mesh. Figure 4.8 shows the improved behavior of p-CMFD over CMFD.

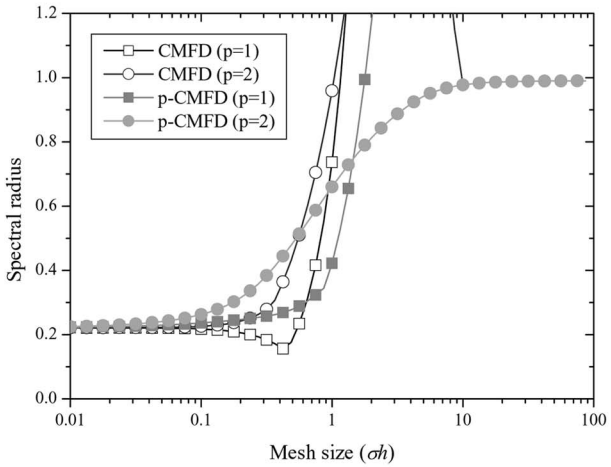


Figure 4.8 Results of Fourier analysis of CMFD and p-CMFD with DD, c=0.99, and S_{16}.

Figure 4.9 shows how CMFD and p-CMFD work in the transport calculation. We use the fusion method for transport calculation and solve low-order equation with BiCGSTAB which is one of the Krylov subspace methods.

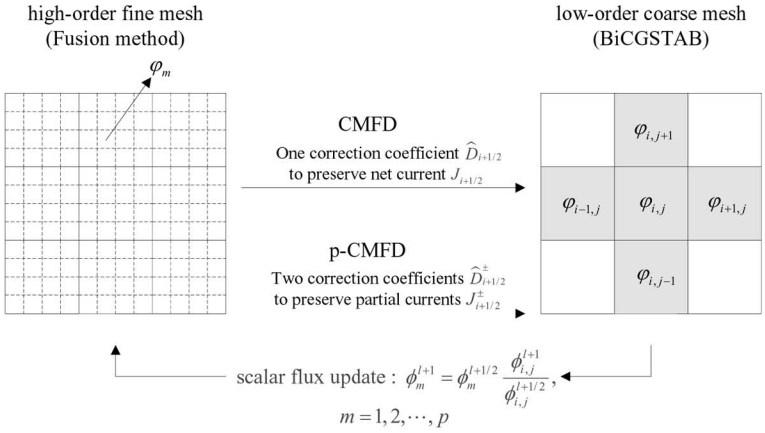


Figure 4.9 Transport calculation of 2D/1D fusion method with CMFD and p-CMFD methods.

4.3.3 Coarse-Mesh Angular Dependent Rebalance (CMADR) Acceleration

The method of angular dependent rebalance (ADR) factors was originally introduced in fine-mesh cases [60, 61] to provide an unconditionally stable acceleration scheme for discrete ordinates method. Then the coarse-mesh angular dependent rebalance (CMADR) method was developed in one- and two-dimensional geometries to accelerate conventional discretization scheme such as diamond difference (DD) scheme [62].

As the nuclear reactor core becomes more complex, heterogeneous, and geometrically irregular, the method of characteristics (MOC) is gaining its wide use in the neutron transport calculations. However, the long computer times required by MOC call for good acceleration methods. In this section, the concept of coarse-mesh angular dependent

rebalance (CMADR) acceleration is described and applied to the MOC calculations. The method is based on angular dependent rebalance factors defined on the coarse-mesh boundaries; a coarse-mesh consists of several fine meshes that may be (1) heterogeneous and (2) of mixed geometries with nonregular or unstructured mesh shapes. In addition, (3) the coarse-mesh boundaries may not coincide with the structural interfaces of the problem and can be chosen artificially for convenience. Thus, the CMADR acceleration method on the MOC scheme that allows the very desirable features (1), (2), and (3) above will be very useful for the analysis of complex reactors.

In MOC calculations, for a computational mesh *i* with flat source approximation, the outgoing angular flux along ray ℓ and the average angular flux over mesh i are given, respectively, as follows:

$$\psi^{l,\alpha+1/2}_{m,n,i+1/2} = \exp(-\frac{\sigma_i L^l_{m,n,i}}{\sin\theta_n})\psi^{l,\alpha+1/2}_{m,n,i-1/2} + \frac{q^{\alpha}_{m,n,i}}{\sigma_i}(1-\exp(-\frac{\sigma_i L^l_{m,n,i}}{\sin\theta_n})), \tag{4.45}$$

$$\overline{\psi}^{\alpha+1/2}_{m,n,i} = \frac{q^{\alpha}_{m,n,i}}{\sigma_i} + \frac{\sin\theta_n}{\sigma_i A_i}\sum_{l\in i}\delta^l_m(\psi^{l,\alpha+1/2}_{m,n,i-1/2} - \psi^{l,\alpha+1/2}_{m,n,i+1/2}), \tag{4.46}$$

where α is the iteration index, m and n are azimuthal and polar angle indices respectively, θ_n is n-th polar angle, $L^l_{m,n,i}$ is the track length of *l*-th ray of mesh *i* in *(m,n)* direction, A_i is the area of the mesh *i*, and δ^l_m is the ray spacing of ray l in *m*-th azimuthal angle.

To obtain the CMADR equations, a coarse mesh *(ni,nj)* composed of several computational meshes is considered as in Fig. 4.10. Now, coarse mesh index *(ni,nj)* will be omitted for simplicity. Using Eq. (4.45), the outgoing angular flux of the coarse mesh along the *l*-th ray is given as follows:

$$\psi^{l,\alpha+1/2}_{m,n,t} = T^l_{m,n,u\to t}\psi^{l,\alpha+1/2}_{m,n,u} + \sum_{i=1}^{N_m(u,t)} T^l_{m,n,i\to t} q^{\alpha}_{m,n,i}, \tag{4.47}$$

where *u* and *t* are incoming and outgoing edge indices, $N_m(u,t)$ is the number of meshes that affect outgoing edge *t* along the *l*-th ray in incoming edge *u*. Coefficients *T* are given as follows:

$$T^l_{m,n,u\to t} = \exp(-\sum_{i=1}^{N_m(u,t)}\sigma_i L^l_{m,n,i}/\sin\theta_n), \tag{4.48}$$

$$T^l_{m,n,i\to t} = \frac{[1-\exp(-\sigma_i L^l_{m,n,i}/\sin\theta_n)]}{\sigma_i}\exp(-\sum_{p=i+1}^{N_m(u,t)}\sigma_p L^l_{m,n,p}/\sin\theta_n). \tag{4.49}$$

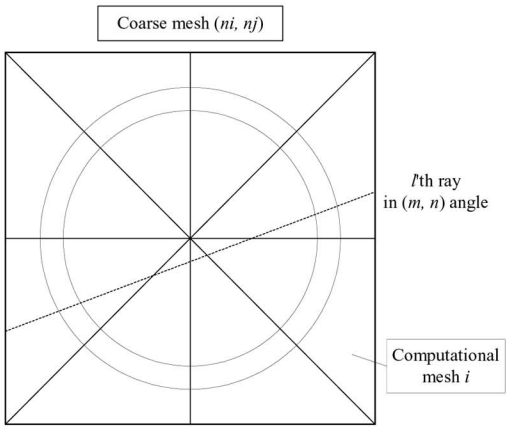


Figure 4.10 A coarse mesh and computational meshes

Then, *x*- and *y*-direction outgoing angular fluxes of the coarse mesh are given by

$$\sum_{l\in t_x}\delta_m^l\psi_{m,n,t_x}^{l,\alpha+1/2}=\sum_{l\in(u_x,t_x)}\delta_m^l T_{m,n,u_x\to t_x}^l\psi_{m,n,u_x}^{l,\alpha+1/2}+\sum_{l\in(u_y,t_x)}\delta_m^l T_{m,n,u_y\to t_x}^l\psi_{m,n,u_y}^{l,\alpha+1/2}+\sum_{i=1}^{N_m(x)}\sum_{l\in(i,t_x)}\delta_m^l T_{m,n,i\to t_x}^l q_{m,n,i}^{\alpha}, \tag{4.50}$$

$$\sum_{l\in t_y}\delta_m^l\psi_{m,n,t_y}^{l,\alpha+1/2}=\sum_{l\in(u_x,t_y)}\delta_m^l T_{m,n,u_x\to t_y}^l\psi_{m,n,u_x}^{l,\alpha+1/2}+\sum_{l\in(u_y,t_y)}\delta_m^l T_{m,n,u_y\to t_y}^l\psi_{m,n,u_y}^{l,\alpha+1/2}+\sum_{i=1}^{N_m(y)}\sum_{l\in(i,t_y)}\delta_m^l T_{m,n,i\to t_y}^l q_{m,n,i}^{\alpha}, \tag{4.51}$$

where $N_m(x)=N_m(u_x,t_x)+N_m(u_y,t_x)$ and $N_m(y)=N_m(u_x,t_y)+N_m(u_y,t_y)$.
Now the angular dependant rebalance factors are defined on the coarse mesh boundaries as:

$$f_{x,m,n,t_x}=\frac{\psi_{m,n,t_x}^{l,\alpha+1}}{\psi_{m,n,t_x}^{l,\alpha+1/2}}\cong f_{x,t_x}^{\gamma},\quad f_{y,m,n,t_y}=\frac{\psi_{m,n,t_y}^{l,\alpha+1}}{\psi_{m,n,t_y}^{l,\alpha+1/2}}\cong f_{y,t_y}^{\gamma} \tag{4.52}$$

where direction (m,n) is in quadrant γ.

The CMADR equations are obtained by replacing iteration indices in Eqs. (4.50) and (4.51) to $\alpha+1$, introducing rebalance factors, and summing over each quadrant with weight $W_{m,n}$ as follows:

$$\begin{aligned}f_{x,t_x}^{\gamma}&=AX^{\alpha+1/2,\gamma}f_{x,u_x}^{\gamma}+BX^{\alpha+1/2,\gamma}f_{y,u_y}^{\gamma}+CX^{\alpha+1,\gamma},\\ f_{y,t_y}^{\gamma}&=AY^{\alpha+1/2,\gamma}f_{x,u_x}^{\gamma}+BY^{\alpha+1/2,\gamma}f_{y,u_y}^{\gamma}+CY^{\alpha+1,\gamma},\end{aligned} \tag{4.53}$$

We use $\mu=\sin\theta_n\cos\varphi_m$ and $\eta=\sin\theta_n\sin\varphi_m$ as the weights $W_{m,n}$ of x- and y-direction edges, respectively.

In addition to Eq. (4.53) the mesh averaged angular flux can be expressed as follows:

$$\bar{\psi}_{m,n,i}^{\alpha+1/2}=\sum_{l\in(u_x,i)}\delta_m^l P_{m,n,u_x\to i}^l\psi_{m,n,u_x}^{l,\alpha+1/2}+\sum_{l\in(u_y,i)}\delta_m^l P_{m,n,u_y\to i}^l\psi_{m,n,u_y}^{l,\alpha+1/2}+\sum_{j=1}^{N_m(i)}\sum_{l\in(j,i)}\delta_m^l P_{m,n,j\to i}^l q_{m,n,i}^{\alpha}, \tag{4.54}$$

where $N_m(i)=N_m(u_x,i)+N_m(u_y,i)$ and coefficients P are given as follows:

$$P_{m,n,u\to i}^l=\frac{\sin\theta_n}{\sigma_i A_i}\left[1-\exp(-\sigma_i L_{m,n,i}^l/\sin\theta_n)\right]\exp(-\sum_{j=1}^{N_m(i-1)}\sigma_j L_{m,n,j}^l/\sin\theta_n), \tag{4.55}$$

$$\begin{aligned}P_{m,n,j\to i}^l=&\frac{\sin\theta_n}{\sigma_j\sigma_i A_i}\left[1-\exp(-\sigma_j L_{m,n,j}^l/\sin\theta_n)\right]\\ &\times\left[\exp(-\sum_{k=j+1}^{N_m(u,i-1)}\sigma_k L_{m,n,k}^l/\sin\theta_n)-\exp(-\sum_{k=j+1}^{N_m(u,i)}\sigma_k L_{m,n,k}^l/\sin\theta_n)\right],\end{aligned} \tag{4.56}$$

$$P_{m,n,i\to i}^l=\frac{1}{\delta_{m,i}\sigma_i}-\frac{\sin\theta_n}{\sigma_i^2 A_i}\left[1-\exp(-\sigma_i L_{m,n,i}^l/\sin\theta_n)\right]. \tag{4.57}$$

The update equations of CMADR are obtained by changing iteration indices in Eq. (4.54) to $\alpha+1$, introducing rebalance factors, and summing over each quadrant as follows:

$$\begin{aligned}\bar{\phi}_{m,n,i}^{\alpha+1,\gamma}&=DM^{\alpha+1/2,\gamma}f_{x,u_x}^{\gamma}+EM^{\alpha+1/2,\gamma}f_{y,u_y}^{\gamma}+FM^{\alpha+1/2,\gamma},\\ \bar{\phi}_{m,n,i}^{\alpha+1}&=\sum_{\gamma}\bar{\phi}_{m,n,i}^{\alpha+1,\gamma}.\end{aligned} \tag{4.58}$$

The resulting CMADR equations [Eqs. (4.51) and (4.58)] resemble S_2 transport equations and we can solve these equations by transport-like sweep or the Krylov subspace method. Also, the coefficients T and P are independent of

iteration index and flux shape so that these coefficients can be calculated and stored before the iteration. Moreover, if modular ray tracing is used in calculation, only several coefficients are stored according to cell types. We may use the CRX code for MOC, which uses modular ray tracing and BiCGSTAB method to reduce computing time to solve the CMADR equations. The CMADR method on MOC was tested successfully on several test problems and the results showed that it is effective in reducing the number of iterations and computing time. [63]

5. THREE-DIMENSIONAL REACTOR KINETICS

5.1 Methods for Spatial Reactor Kinetics

Although we can discuss spatial reactor kinetics from the transport equation [64, 65], let us start from the time-dependent multigroup diffusion equations as in usual practice :

$$[\mathrm{V}^{-1}]\frac{\partial\vec{\phi}}{\partial t} = \nabla[D]\nabla\vec{\phi} + [B]\vec{\phi} + (1-\beta)[\chi_0][F]^T\vec{\phi} + \sum_{d=1}^{D}\lambda_d C_d[\chi_d] + [S], \tag{5.1a}$$

$$\frac{\partial C_d}{\partial t} = -\lambda_d C_d + \beta_d[F]^T\vec{\phi}, \quad d = 1,2,.....,D, \tag{5.1b}$$

where

$\vec{\phi} = (\phi_1, \phi_2, \cdots\cdots, \phi_G)^T$,

and other notations are standard.

The major spatial reactor kinetics methods can be classified into two categories : space-time factorization methods and direct methods.

5.1.1 Space-Time Factorization Methods

Space-time factorization methods [65-67] involve a factorization of space and time dependence of the flux into i) an amplitude function which carries most of the time-dependence and ii) a shape function which carries all of the space-dependence and is only weakly varying in time. This factorization is an approach of multi-scale analysis in time. Consider an arbitrary vector weighting function (not a function of time),

$$\vec{W}(\vec{r}) = (w_1(\vec{r}), w_2(\vec{r}), \cdots, w_G(\vec{r}))^T$$

and form scalar products (spatial) of $\vec{W}$ with Eqs.(5.1a) and (5.1b),

$$\left(\vec{W}, [V^{-1}]\frac{\partial\vec{\phi}}{\partial t}\right) = \left(\vec{W}, \nabla[D]\nabla\vec{\phi}\right) + \left(\vec{W}, [B]\vec{\phi}\right) + (1-\beta)(\vec{W}, [\chi_0][F]^T\vec{\phi}) + \sum_{d=1}^{D}\lambda_d(\vec{W}, [\chi_d]C_d) + (\vec{W}, [S]), \tag{5.2a}$$

$$\left(\vec{W}, \frac{\partial C_d}{\partial t}\right) = \beta_d(\vec{W}, [F]^T\vec{\phi}) - \lambda_d(\vec{W}, C_d), \quad d = 1,2,\cdots,D. \tag{5.2b}$$

Now let us “factorize” the flux

$$\vec{\phi}(\vec{r},t) = n(t)\vec{\psi}(\vec{r},t). \tag{5.3}$$

Eq.(5.3) involves no approximation, is not separation of variables ; This splitting can be done in an infinite number of ways. We want the splitting to be such that $n(t)$ characterizes most of temporal behavior (amplitude function) and $\vec{\psi}(\vec{r},t)$

characterizes mainly spatial behavior (shape function) of $\vec{\phi}(\vec{r},t)$. Now we define the amplitude function $n(t)$ as follows :

$$n(t) = \left(\overline{W}(\vec{r}),[V^{-1}]\vec{\phi}(\vec{r},t)\right). \tag{5.4}$$

Then, we note from Eqs.(5.3) and (5.4) that

$$n(t) = \left(\overline{W}(\vec{r}),[V^{-1}]\vec{\phi}(\vec{r},t)\right) = \left(\overline{W}(\vec{r}),[V^{-1}]n(t)\vec{\psi}(\vec{r},t)\right) = n(t)\left(\overline{W}(\vec{r}),[V^{-1}]\vec{\psi}(\vec{r},t)\right). \tag{5.5}$$

Thus,

$$\begin{aligned}\left(\overline{W}(\vec{r}),[V^{-1}]\vec{\psi}(\vec{r},t)\right) = 1 \ &: \text{a constraint on the shape function } \vec{\psi}(\vec{r},t),\\ &: \text{normalization condition}\end{aligned} \tag{5.6}$$

The shape function is forced to describe temporal redistribution of neutrons within the reactor. Substitute Eq.(5.3) in Eq.(5.2a),

$$\begin{aligned}LHS &= \left(\overline{W},[V^{-1}]\frac{dn(t)}{dt}\vec{\psi}(\vec{r},t)\right) + \left(\overline{W},[V^{-1}]n(t)\frac{\partial\vec{\psi}(\vec{r},t)}{\partial t}\right)\\ &= \frac{dn(t)}{dt}\left(\overline{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right) + n(t)\left(\overline{W},[V^{-1}]\frac{\partial\vec{\psi}(\vec{r},t)}{\partial t}\right).\end{aligned} \tag{5.7}$$

Due to the constraint on the shape function in Eq.(5.6), the second term in the right hand side of Eq.(5.7) vanishes. Thus Eq.(5.2a) becomes

$$\begin{aligned}&\left(\overline{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)\frac{d}{dt}n(t) =\\ &\left\{\left(\overline{W},\nabla[D]\nabla\vec{\psi}(\vec{r},t)\right) + \left(\overline{W},[B]\vec{\psi}(\vec{r},t)\right) + (1-\beta)\left(\overline{W},[\chi_0][F]^T\vec{\psi}(\vec{r},t)\right)\right\}n(t)\\ &+\sum_{d=1}^{D}\lambda_d\left(\overline{W},[\chi_d]C_d\right) + \left(\overline{W},[S]\right).\end{aligned} \tag{5.8a}$$

Multiplying Eq.(5.2b) by $[\chi_d]$ and substituting Eq.(5.3),

$$\left(\overline{W},[\chi_d]\frac{\partial C_d}{\partial t}\right) = \beta_d\left(\overline{W},[\chi_d][F]^T\vec{\psi}(\vec{r},t)\right)n(t) - \lambda_d\left(\overline{W},[\chi_d]C_d\right). \tag{5.8b}$$

Rewriting Eqs.(5.8a) and (5.8b),

$$\begin{aligned}\frac{dn(t)}{dt} = &\left\{\frac{\left(\overline{W},\nabla[D]\nabla\vec{\psi}(\vec{r},t)\right) + \left(\overline{W},[B]\vec{\psi}(\vec{r},t)\right) + (1-\beta)\left(\overline{W},[\chi_0][F]^T\vec{\psi}(\vec{r},t)\right)}{\left(\overline{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)}\right.\\ &\left.+\frac{\sum_{d=1}^{D}\beta_d\left(\overline{W},[\chi_d][F]^T\vec{\psi}(\vec{r},t)\right)}{\left(\overline{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)} - \frac{\sum_{d=1}^{D}\beta_d\left(\overline{W},[\chi_d][F]^T\vec{\psi}(\vec{r},t)\right)}{\left(\overline{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)}\right\}n(t)\\ &+\frac{\sum_{d=1}^{D}\lambda_d\left(\overline{W},[\chi_d]C_d\right)}{\left(\overline{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)} + \frac{\left(\overline{W},[S]\right)}{\left(\overline{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)},\end{aligned} \tag{5.9a}$$

$$\frac{d}{dt}\frac{\left(\vec{W},[\chi_d]C_d\right)}{\left(\vec{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)}=\frac{\beta_d\left(\vec{W},[\chi_d][F]^T\vec{\psi}(\vec{r},t)\right)}{\left(\vec{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)}n(t)-\frac{\lambda_d\left(\vec{W},[\chi_d]C_d\right)}{\left(\vec{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)}. \tag{5.9b}$$

Recall that

i) $n(t)\equiv\left(\vec{W},[V^{-1}]\vec{\phi}(\vec{r},t)\right)$ from Eq.(5.4) : weighted integral over reactor (5.10)

volume of all neutron groups at time t.

Now let

ii) $\tilde{f}(t)\equiv\left(\vec{W},\left\langle(1-\beta)[\chi_0]+\sum_{d=1}^{D}\beta_d[\chi_d]\right\rangle[F]^T\vec{\psi}(\vec{r},t)\right)$: weighted integral (5.11)

over reactor volume of the prompt and delayed neutron fission source (production rate).

iii) $l(t)\equiv\dfrac{\left(\vec{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)}{\tilde{f}(t)}$: neutron generation time, (5.12)

:ratio of weighted neutrons at t to weighted rate of production of fission neutrons at the same time.

iv) $\bar{\beta}_d\equiv\dfrac{\left(\vec{W},\beta_d[\chi_d][F]^T\vec{\psi}(\vec{r},t)\right)}{\tilde{f}(t)}$: "effective" delayed neutron fraction, (5.13)

$$\bar{\beta}(t)=\sum_{d=1}^{D}\bar{\beta}_d(t).$$

v) $\bar{C}_d(t)\equiv\dfrac{\left(\vec{W},[\chi_d]C_d\right)}{\left(\vec{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)}$: "effective" d^{th} precursor. (5.14)

vi) $\bar{S}(t)\equiv\dfrac{\left(\vec{W},[S]\right)}{\left(\vec{W},[V^{-1}]\vec{\psi}(\vec{r},t)\right)}$: "effective" source. (5.15)

vii) $\rho(t)\equiv\dfrac{1}{\tilde{f}(t)}\left[\tilde{f}(t)-\left(\vec{W},\left\langle-\nabla[D]\nabla-[B]\right\rangle\vec{\psi}(\vec{r},t)\right)\right]$: "reactivity" (5.16)

$$=1-\frac{1}{k},$$

where

$$k\equiv\frac{\left(\vec{W},\left\langle(1-\beta)[\chi_0]+\sum_{d=1}^{D}\beta_d[\chi_d]\right\rangle[F]^T\vec{\psi}(\vec{r},t)\right)}{\left(\vec{W},\left\langle-\nabla[D]\nabla-[B]\right\rangle\vec{\psi}(\vec{r},t)\right)}, \tag{5.17}$$

or

$$\rho(t) \equiv \frac{\left(\vec{W},\left\langle (1-\beta)[\chi_0]+\sum_{d=1}^{D}\beta_d[\chi_d]\right\rangle[F]^T\vec{\psi}(\vec{r},t)\right)-\left(\vec{W},\langle -\nabla[D]\nabla-[B]\rangle\vec{\psi}(\vec{r},t)\right)}{\left(\vec{W},\left\langle (1-\beta)[\chi_0]+\sum_{d=1}^{D}\beta_d[\chi_d]\right\rangle[F]^T\vec{\psi}(\vec{r},t)\right)}$$
$$= \frac{\begin{pmatrix}\text{total weighted production rate of}\\ \text{neutrons due to fission at time } t\end{pmatrix}-\begin{pmatrix}\text{total weighted loss rate}\\ \text{due to leakage and absorption}\end{pmatrix}}{\begin{pmatrix}\text{total weighted production rate of neutrons}\\ \text{due to fission at time } t\end{pmatrix}}. \tag{5.18}$$

Using these definitions, Eq. (5.9) becomes

$$\frac{d}{dt}n(t) = \frac{\rho(t)-\bar{\beta}(t)}{l(t)}n(t)+\sum_{d=1}^{D}\lambda_d\bar{C}_d(t)+\bar{S}(t), \tag{5.19a}$$

$$\frac{d}{dt}\bar{C}_d(t) = -\lambda_d\bar{C}_d(t)+\frac{\bar{\beta}_d(t)}{l(t)}n(t). \tag{5.19b}$$

These are the (exact) point kinetics equations. Some comments are in order :

i) Each reactor model has the same form of equations, but different definitions of parameters.

ii) The form of the equations is independent of $\vec{W}$ and $\vec{\psi}$. $\vec{W}$ is not yet specified and $\vec{\psi}$ not determined.

iii) $\bar{\beta}_d$, $\bar{\beta}$, l are not physical properties to be looked up in handbook. They depend on spectrum, composition, weighting and shape functions, etc.

iv) The point kinetics equations (5.19) are exact with the $\vec{W}$ and $\vec{\psi}$ given, i.e., no assumptions or approximations have been made.

v) We imposed that most of the time behavior in $\vec{\phi}(\vec{r},t)$ be carried by the amplitude function $n(t)$, and spatial behavior by the shape function $\vec{\psi}(\vec{r},t)$.

vi) The practical utility of the kinetics equations depends on our ability to obtain a reasonably accurate value of $\vec{\psi}(\vec{r},t)$ without actually solving Eq.(5.1).

So far the weighting has not been specified. One frequently chooses

$$\vec{W}(\vec{r}) = \vec{\phi}_c^{+}(\vec{r}), \tag{5.20}$$

which is the adjoint solution to the initial steady state critical equation, based on a variational principle that says that the error in the reactivity $\rho(t)$ resulting from inaccurate shape function $\vec{\psi}(\vec{r},t)$ is reduced.

Now the equation for the shape function is obtained by substituting Eq.(5.3) in the multigroup diffusion equation Eq.(5.1) with $[S]=0$,

$$[V^{-1}]\left(\frac{dn(t)}{dt}\vec{\psi}(\vec{r},t)+n(t)\frac{\partial}{\partial t}\vec{\psi}(\vec{r},t)\right) =$$
$$\left(\nabla[D]\nabla+[B]+(1-\beta)[\chi_0][F]^T\right)n(t)\vec{\psi}(\vec{r},t)+\sum_{d=1}^{D}\lambda_d[\chi_d]C_d,$$

or

$$[V^{-1}]\frac{\partial}{\partial t}\vec{\psi}(\vec{r},t) = \left\{\nabla[D]\nabla+[B]+(1-\beta)[\chi_0][F]^T-[V^{-1}]\frac{1}{n(t)}\frac{dn(t)}{dt}\right\}\vec{\psi}(\vec{r},t)$$
$$+\frac{1}{n(t)}\sum_{d=1}^{D}\lambda_d[\chi_d]C_d. \tag{5.21a}$$

Eq.(5.1b) is solved for $C_d(\vec{r},t)$ as

$$C_d(\vec{r},t) = C_d(\vec{r},0)\exp(-\lambda_d t) + \int_0^t \exp[-\lambda_d(t-t')]\beta_d[F]^T n(t')\vec{\psi}(\vec{r},t')dt'. \tag{5.21b}$$

Let

$$S_d[n(t')\vec{\psi}(\vec{r},t');t] = \sum_{d=1}^{D} \lambda_d[\chi_d]C_d.$$

Then, Eq.(5.21a) becomes

$$[V^{-1}]\frac{\partial}{\partial t}\vec{\psi}(\vec{r},t) = \left\{\nabla[D]\nabla + [B] + (1-\beta)[\chi_0][F]^T - [V^{-1}]\frac{1}{n(t)}\frac{dn(t)}{dt}\right\}\vec{\psi}(\vec{r},t) + \frac{1}{n(t)}S_d[n(t')\vec{\psi}(\vec{r},t');t] \tag{5.22}$$

Equations (5.19), (5.22), and (5.6) are equivalent to the original time-dependent multigroup diffusion equations (5.1) ; No approximations have been made yet.

Several variants of the flux factorization approaches are obtained depending on how to approximate Eq.(5.22) for the shape function ; Since $\vec{\psi}$ is slowly varying, Δt can be much larger than the time step required to solve Eq.(5.19). The shape function is solved less frequently than the amplitude function. A noticeable deviation from unity of Eq.(5.6) with a calculated shape function indicates that the time step was large and that the shape function needs to be recalculated with a smaller time step. The following is a sequence of approximations in calculating the shape function (in an increasing order of approximations).

i) Improved Quasistatic Approximation :

$\partial\vec{\psi}/\partial t$ in Eq.(5.22) is approximated by a backward difference :

$$\frac{\partial}{\partial t}\vec{\psi}(\vec{r},t) = \frac{\vec{\psi}(\vec{r},t) - \vec{\psi}(\vec{r},t-\Delta t)}{\Delta t}$$

where $t-\Delta t$ is the time of the last shape function calculation. Thus Eq.(5.22) becomes

$$\left\{[V^{-1}]\frac{1}{\Delta t} - \nabla[D]\nabla - [B] - (1-\beta)[\chi_0][F]^T + [V^{-1}]\frac{1}{n(t)}\frac{dn(t)}{dt}\right\}\vec{\psi}(\vec{r},t) = \frac{1}{n(t)}S_d[n(t')\vec{\psi}(\vec{r},t');t] + \frac{1}{\Delta t}[V^{-1}]\vec{\psi}(\vec{r},t-\Delta t). \tag{5.23}$$

n and dn/dt are from Eq.(5.19). S_d is corrected for $t \leftarrow t-\Delta t$: extrapolation of $\vec{\psi}$ in Eq.(5.21b).

ii) Quasistatic Approximation :

$\partial\vec{\psi}/\partial t$ in Eq.(5.22) is neglected entirely, i.e., $\partial\vec{\psi}/\partial t = 0$. Thus, Eq.(5.22) becomes

$$\left\{\nabla[D]\nabla + [B] + (1-\beta)[\chi_0][F]^T - [V^{-1}]\frac{1}{n(t)}\frac{dn(t)}{dt}\right\}\vec{\psi}(\vec{r},t) = -\frac{1}{n(t)}S_d[n(t')\vec{\psi}(\vec{r},t');t]. \tag{5.24}$$

n and dn/dt are from Eq.(5.19). S_d is corrected for $t \leftarrow t-\Delta t$: extrapolation of $\vec{\psi}$ in Eq.(5.21b).

iii) Adiabatic Approximation :

$\partial\vec{\psi}/\partial t = 0$ and $dn/dt = 0$. Then,

$$(1-\beta)[\chi_0][F]^T\vec{\psi}(\vec{r},t) + \frac{1}{n(t)}S_d[n(t')\vec{\psi}(\vec{r},t');t] \;\rightarrow\; [\chi][F]^T\vec{\psi}(\vec{r},t) \tag{5.25}$$

The adiabatic approximation does not distinguish the shape of delayed neutron source from the shape of prompt source, thus neglecting the time retardation. Thus Eq.(5.22) becomes

$$\left\{\nabla[D]\nabla+[B]+\frac{[\chi][F]^T}{k}\right\}\vec{\psi}(\vec{r},t)=0 \quad : \quad \lambda\text{-mode} \tag{5.26}$$

iv) Point Reactor Model :

In this usual point kinetics model, the initial flux distribution is used as the shape function throughout the transient :

$$\vec{\psi}(\vec{r},t)=\vec{\phi}_c(\vec{r}). \tag{5.27}$$

The procedures are summarized as follows :

Step 1 : Find the adjoint solution $\vec{\phi}_c^{+}(\vec{r})$ to the initial steady state equation and use it as $\vec{W}(\vec{r})$:

$$\left\{\nabla[D]\nabla+[B]+\frac{[\chi][F]^T}{k}\right\}^T\vec{\phi}_c^{+}(\vec{r})=0$$

Step 2 : Set $\vec{\psi}(\vec{r},0)=\vec{\phi}_c(\vec{r})$: initial flux distribution.

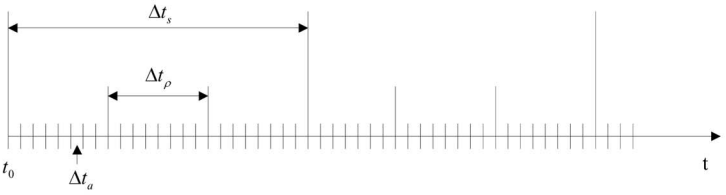
Step 3 : Calculate point kinetics parameters according to the definitions (5.11)through (5.16) at every Δt_ρ, incorporating system change.

Step 4 : Using these parameters, solve the point kinetics equations (5.19) for a duration with time step Δt_ρ.

Step 5 : Using the results, calculate the new shape function at Δt_ρ, using one of the approximations (5.23), (5.24), (5.26).For a point reactor model, skip this step.

Step 6 : Go to Step 3.

Figure 5.1 shows schematically the three time scales involved in factorization methods. Step 5 for shape function calculation is a basically spatial problem for which the modern nodal methods can be used instead of the fine-mesh finite difference methods. Then the flux distribution should be reconstructed for its detailed distribution to be used to evaluate the point kinetics parameters accurately.



Δt_a : amplitude function, *n(t)*
Δt_ρ : kinetics parameters, e.g., *ρ(t)*
Δt_s : shape function, $\vec{\psi}(\vec{r},t)$

Figure 5.1 Example of various time steps in the calculational procedures of the factorization methods.

5.1.2 Direct Methods

Direct methods are generally straightforward and conceptually simpler than the space-time factorization methods. Direct methods are characterized by a simultaneous space-time discretization for the solution of the time-dependent multigroup diffusion equations.

Space discretization can be based on finite difference methods, polynomial basis-coarse mesh methods, or modern nodal methods. For time discretization, various treatments are used such as explicit method, implicit method, "θ" method, or alternating direction implicit (ADI) method. To describe a direct method based on implicit time discretization and a modern nodal method for space discretization, we rewrite Eq.(5.1) for the two-group neutron fluxes and six delayed neutron precursor densities for node *m* :

$$\begin{aligned}[V^m]^{-1}\frac{\partial}{\partial t}\vec{\phi}^m(\vec{r},t)&=\nabla\cdot[D^m(t)]\nabla\vec{\phi}^m(\vec{r},t)-[A^m(\vec{r},t)]\vec{\phi}^m(\vec{r},t)\\&\quad+(1-\beta^m)\vec{\chi}_0[F^m(\vec{r},t)]^T\vec{\phi}^m(\vec{r},t)+\sum_{d=1}^{D}\lambda_d^m C_d^m(\vec{r},t)\vec{\chi}_d\end{aligned} \tag{5.28a}$$

and

$$\frac{\partial}{\partial t}C_d^m(\vec{r},t) = -\lambda_d^m C_d^m(\vec{r},t) + \beta_d^m [F^m(\vec{r},t)]^T \vec{\phi}^m(\vec{r},t), \quad d = 1,2,\cdots,D, \tag{5.28b}$$

where

$$\vec{\phi}^m(\vec{r},t) \equiv \left\{\vec{\phi}_1^m(\vec{r},t), \vec{\phi}_2^m(\vec{r},t)\right\}^T,$$

$$[V^m]^{-1} \equiv \begin{pmatrix} \frac{1}{v_1^m} & 0 \\ 0 & \frac{1}{v_2^m} \end{pmatrix},$$

$$[D^m(t)] \equiv \begin{pmatrix} D_1^m(t) & 0 \\ 0 & D_2^m(t) \end{pmatrix},$$

$$[A^m(\vec{r},t)] \equiv \begin{pmatrix} \Sigma_{a1}^m(\vec{r},t) + \Sigma_{1\to 2}^m & 0 \\ -\Sigma_{1\to 2}^m & \Sigma_{a2}^m(\vec{r},t) \end{pmatrix},$$

$$[F^m(\vec{r},t)]^T \equiv \left(\nu\Sigma_{f1}^m(\vec{r},t) \quad \nu\Sigma_{f2}^m(\vec{r},t)\right).$$

To treat space-dependent feedback, the absorption and fission cross sections are allowed to have spatial variations. For convenience, we separate these cross sections into a constant part (subscript 0) and a space-dependent part (subscript 1) as follows [24] :

$$[A^m(\vec{r},t)] = [A_0^m(t)] + [A_1^m(\vec{r},t)], \tag{5.29}$$

and

$$[F^m(\vec{r},t)] = [F_0^m(t)] + [F_1^m(\vec{r},t)]. \tag{5.30}$$

In most of the direct methods reported in the literature, i) exponential transformation [68] and ii) linearly varying assumption on fission rate within a time step [69] are used without introducing significant approximation but leading to much convenience. These allow much larger time steps to be used and analytic solutions for the precursor densities.

Eq.(5.1) is discretized in time by applying the exponential transformation :

$$\vec{\phi}^m(\vec{r},t) = \left[e^{\omega_n^m(t-t_n)}\right]\vec{\psi}^m(\vec{r},t), \quad (t_n \le t \le t_{n+1}), \tag{5.31}$$

where

$$\left[e^{\omega_n^m(t-t_n)}\right] \equiv \begin{pmatrix} e^{\omega_{n,1}^m(t-t_n)} & 0 \\ 0 & e^{\omega_{n,2}^m(t-t_n)} \end{pmatrix}$$

and

$$\omega_{n,g}^m = \frac{\ln\frac{\phi_g^m(t_n)}{\phi_g^m(t_{n-1})}}{t_n - t_{n-1}}.$$

Note that

$$\vec{\phi}^{m}(\vec{r},t_n)=\vec{\psi}^{m}(\vec{r},t_n)$$

and

$$\vec{\phi}^{m}(\vec{r},t_{n+1})=\left[e^{\omega_n^m\Delta t}\right]\vec{\psi}^{m}(\vec{r},t_{n+1}).$$

The value ω_n^m is obtained from the previous time-step calculation (between t_{n-1} and t_n). The last two conditions give the relations between $\vec{\phi}^{m}(\vec{r},t)$ and $\vec{\psi}^{m}(\vec{r},t)$ at time t_n and t_{n+1}. From these conditions, the time derivative of Eq.(5.31) can be derived as follows :

$$\frac{\partial}{\partial t}\vec{\phi}^{m}(\vec{r},t)=\left[\omega_n^m e^{\omega_n^m(t-t_n)}\right]\vec{\psi}^{m}(\vec{r},t)+\left[e^{\omega_n^m(t-t_n)}\right]\frac{\partial}{\partial t}\vec{\psi}^{m}(\vec{r},t). \tag{5.32}$$

The partial derivative $\partial\vec{\psi}^{m}(\vec{r},t)/\partial t$ is now approximated using an implicit Euler discretization, which yields the following form for the derivative of the flux :

$$\left[\frac{\partial}{\partial t}\vec{\phi}^{m}(\vec{r},t)\right]_{t=t_{n+1}}\approx\left[\omega_n^m e^{\omega_n^m\Delta t}\right]\vec{\psi}^{m}(\vec{r},t_{n+1})+\left[e^{\omega_n^m\Delta t}\right]\frac{\vec{\psi}^{m}(\vec{r},t_{n+1})-\vec{\psi}^{m}(\vec{r},t_n)}{\Delta t}. \tag{5.33}$$

By using Eq.(5.31), the precursor equation, Eq.(5.28b) is integrated analytically with the approximation that the quantity $[F^m(\vec{r},t)]^T\vec{\psi}^{m}(\vec{r},t)$ is linear in a time step. It becomes

$$\begin{aligned}C_d^m(\vec{r},t_{n+1})&=e^{-\lambda_d^m\Delta t}C_d^m(\vec{r},t_n)\\&\quad+\sum_{g=1}^{2}\alpha_{1,g}^{d,m}\nu\Sigma_{fg}^{m}(\vec{r},t_n)\vec{\psi}_g^{m}(\vec{r},t_n)+\sum_{g=1}^{2}\alpha_{2,g}^{d,m}\nu\Sigma_{fg}^{m}(\vec{r},t_{n+1})\vec{\psi}_g^{m}(\vec{r},t_{n+1}),\end{aligned} \tag{5.34}$$

where

$$\alpha_{1,g}^{d,m}=\frac{\beta_d^m}{\lambda_d^m+\omega_{n,g}^m}\left\{\frac{e^{\omega_{n,g}^m\Delta t}-e^{-\lambda_d^m\Delta t}}{\Delta t(\lambda_d^m+\omega_{n,g}^m)}-e^{-\lambda_d^m\Delta t}\right\}$$

and

$$\alpha_{2,g}^{d,m}=\frac{\beta_d^m}{\lambda_d^m+\omega_{n,g}^m}\left\{e^{\omega_{n,g}^m\Delta t}-\frac{e^{\omega_{n,g}^m\Delta t}-e^{-\lambda_d^m\Delta t}}{\Delta t(\lambda_d^m+\omega_{n,g}^m)}\right\}.$$

Substituting Eqs.(5.33) and (5.34) into Eq.(5.28a), we obtain the following time-discretized diffusion kinetics equation :

$$\begin{aligned}&-\nabla\cdot[D^m(t_{n+1})]\nabla\vec{\psi}^{m}(\vec{r},t_{n+1})+\{[A^m(\vec{r},t_{n+1})]-\vec{\chi}_0(1-\beta^m)[F^m(\vec{r},t_{n+1})]^T\}\vec{\psi}^{m}(\vec{r},t_{n+1})\\&\qquad+[\gamma^m(t_{n+1})]\vec{\psi}^{m}(\vec{r},t_{n+1})=[\gamma^m(t_n)]\vec{\psi}^{m}(\vec{r},t_n)+\vec{S}^{m}(\vec{r},t_n),\end{aligned} \tag{5.35}$$

where

$$[\gamma^m(t_n)]=[e^{-\omega_n^m\Delta t}]\left[[V^m]^{-1}\left[\frac{e^{\omega_n^m\Delta t}}{\Delta t}\right]+[M_1^m]\right],$$

$$[\gamma^m(t_{n+1})]=\left[[V^m]^{-1}\left([\omega_n^m]+\left[\frac{1}{\Delta t}\right]\right)+[M_2^m]\right],$$

$$\vec{S}^{m}(\vec{r},t_n)=[e^{-\omega_n^m\Delta t}]\sum_{d=1}^{D}\vec{\chi}_d e^{-\lambda_d^m\Delta t}\{\lambda_d^m C_d^m(\vec{r},t_n)-\beta_d^m[F^m(\vec{r},t_n)]^T\vec{\psi}^{m}(\vec{r},t_n)\},$$

$$[M_1^m] = \sum_{d=1}^{D} \vec{\chi}_d [(\alpha_{1,1}^{d,m}\lambda_d^m - \beta_d^m e^{-\lambda_d^m \Delta t})\nu\Sigma_{f1}^m(\vec{r},t_n),\ (\alpha_{1,2}^{d,m}\lambda_d^m - \beta_d^m e^{-\lambda_d^m \Delta t})\nu\Sigma_{f2}^m(\vec{r},t_n)]^T,$$

and

$$[M_2^m] = \sum_{d=1}^{D} \vec{\chi}_d [\alpha_{2,1}^{d,m}\lambda_d^m \nu\Sigma_{f1}^m(\vec{r},t_{n+1}),\ \alpha_{2,2}^{d,m}\lambda_d^m \nu\Sigma_{f2}^m(\vec{r},t_{n+1})].$$

Note that no approximations except the time derivative of $\psi^m(\vec{r},t)$ are used in deriving Eq.(5.35). Because the right hand side includes previous time-step variables fixed during the calculation, the equation is an inhomogeneous fixed-source problem. Also, because the right hand side of the equation has a complicated form, it is not effective to find an exact particular solution. In the analytic function expansion nodal (AFEN) method, Eq.(5.35) is solved, to be described below, by decomposing the solution into an analytic part and a correctional polynomial part.

Although there are kinetics versions of several other modern nodal methods [70], in the following we describe the kinetics calculation approach of the AFEN method that was discussed in Section 3.3.3 for static core analysis.

In contrast to the conventional nodal methods, there is no transverse-integration procedure in the AFEN method. This feature avoids the transverse-leakage approximation and requires a 3-D space-dependent solution scheme to solve the inhomogeneous equation of Eq.(5.35). We write the solution of Eq.(5.35) as a combination of the analytic part $\vec{\psi}_a^m(\vec{r},t_{n+1})$ and the polynomial part $\vec{\psi}_a^m(\vec{r},t_{n+1})$:

$$\vec{\psi}^m(\vec{r},t_{n+1}) = \vec{\psi}_a^m(\vec{r},t_{n+1}) + \vec{\psi}_p^m(\vec{r},t_{n+1}). \tag{5.36}$$

We choose the analytic part of Eq.(5.36) to consist of the analytic basis functions as follows :

$$\vec{\psi}_a^m(\vec{r},t_{n+1}) = \sum_i \sinh(\sqrt{[\Lambda^m]}\vec{\tau}_i\cdot\vec{r})\vec{\alpha}_{a0i}^m + \cosh(\sqrt{[\Lambda^m]}\vec{\tau}_i\cdot\vec{r})\vec{\alpha}_{a1i}^m\ , \tag{5.37}$$

where

$$[\Lambda^m] = [D^m(t_{n+1})]^{-1}([A_0^m(t_{n+1})] - \vec{\chi}_0[F_0^m(t_{n+1})]^T).$$

In 3-D geometry,

$$\vec{\psi}_a^m(x,y,z,t_{n+1}) = [R^m]\vec{\xi}^m(x,y,z) \tag{5.38}$$

and

$$\xi_\mu^m(x,y,z) = \alpha_{\mu 0}^m + g_\mu^m(x,y) + g_\mu^m(y,z) + g_\mu^m(z,x),\ \ \mu=1,2\ , \tag{5.39}$$

where

$$\begin{aligned} g_\mu^m(x,y) = {} & \alpha_{a\mu 1}^{m,xy} SN(\kappa_\mu^m x) + \alpha_{a\mu 2}^{m,xy} CS(\kappa_\mu^m x) \\ & + \alpha_{a\mu 3}^{m,xy} SN\left(\frac{\kappa_\mu^m x}{\sqrt{2}}\right) SN\left(\frac{\kappa_\mu^m y}{\sqrt{2}}\right) + \alpha_{a\mu 4}^{m,xy} SN\left(\frac{\kappa_\mu^m x}{\sqrt{2}}\right) CS\left(\frac{\kappa_\mu^m y}{\sqrt{2}}\right) \\ & + \alpha_{a\mu 5}^{m,xy} CS\left(\frac{\kappa_\mu^m x}{\sqrt{2}}\right) SN\left(\frac{\kappa_\mu^m y}{\sqrt{2}}\right) + \alpha_{a\mu 6}^{m,xy} CS\left(\frac{\kappa_\mu^m x}{\sqrt{2}}\right) CS\left(\frac{\kappa_\mu^m y}{\sqrt{2}}\right) \end{aligned}$$

$$\kappa_\mu^m = \sqrt{|\lambda_\mu^m|}$$

λ_μ^m, $[R^m]$ = eigenvalues and transformation matrix of $[\Lambda^m]$

$$SN = \begin{cases} \sinh\ , & \lambda_\mu^m > 0 \\ \sin\ , & \lambda_\mu^m < 0 \end{cases}$$

$$CS = \begin{cases} \cosh \ , & \lambda_\mu^m > 0 \\ \cos \ , & \lambda_\mu^m < 0 \end{cases}$$

The notations are standard in the AFEN methodology [24]. Note that $\vec{\psi}_a^m(\vec{r},t_{n+1})$ satisfies the following "quasi-static" equation :

$$\{-\nabla\cdot[D^m(t_{n+1})]\nabla+[A_0^m(t_{n+1})]-\vec{\chi}_0[F_0^m(t_{n+1})]^T\}\vec{\psi}_a^m(\vec{r},t_{n+1})=\vec{0}, \tag{5.40}$$

in view of the definition of Eq.(5.37). To find $\vec{\psi}_a^m(\vec{r},t_{n+1})$, we impose the following weighted integral equation on Eq.(5.35) :

$$\begin{aligned}&\int\omega(\vec{r})\{-\nabla\cdot[D^m(t_{n+1})]\nabla+[A^m(\vec{r},t_{n+1})]-\vec{\chi}_0[F^m(\vec{r},t_{n+1})]^T\}\times\{\vec{\psi}_a^m(\vec{r},t_{n+1})+\vec{\psi}_p^m(\vec{r},t_{n+1})\}dV_m\\&\quad-\int\omega(\vec{r})[[\gamma^m(t_n)]\{\vec{\psi}_a^m(\vec{r},t_n)+\vec{\psi}_p^m(\vec{r},t_n)\}-(\vec{\chi}_0\beta^m[F^m(\vec{r},t_{n+1})]^T+[\gamma^m(t_{n+1})]\\&\quad\times\{\vec{\psi}_a^m(\vec{r},t_{n+1})+\vec{\psi}_p^m(\vec{r},t_{n+1})\}]dV_m\\&\quad-\int\omega(\vec{r})\vec{S}^m(\vec{r},t_n)dV_m=\vec{0}.\end{aligned} \tag{5.41}$$

By using Eq.(5.40), Eq.(5.41) becomes

$$\begin{aligned}&\int\omega(\vec{r})\{-\nabla\cdot[D^m(t_{n+1})]\nabla+[A_0^m(t_{n+1})]-\vec{\chi}_0[F_0^m(t_{n+1})]^T\}\vec{\psi}_p^m(\vec{r},t_{n+1})dV_m\\&\quad-\int\omega(\vec{r})[[\gamma^m(t_n)]\{\vec{\psi}_a^m(\vec{r},t_n)+\vec{\psi}_p^m(\vec{r},t_n)\}-(\vec{\chi}_0\beta^m[F^m(\vec{r},t_{n+1})]^T+[\gamma^m(t_{n+1})]\\&\quad\times\{\vec{\psi}_a^m(\vec{r},t_{n+1})+\vec{\psi}_p^m(\vec{r},t_{n+1})\}]dV_m\\&\quad-\int\omega(\vec{r})\vec{S}^m(\vec{r},t_n)dV_m\\&\quad+\int\omega(\vec{r})[\{[A_1^m(\vec{r},t_{n+1})-\chi_0F_1^m(\vec{r},t_{n+1})\}\times\{\vec{\psi}_a^m(\vec{r},t_{n+1})+\vec{\psi}_p^m(\vec{r},t_{n+1})\}]dV_m=\vec{0}.\end{aligned} \tag{5.42}$$

In this equation, Legendre polynomials are used both for the weighting function $\omega(\vec{r})$ and for the basis functions of $\vec{\psi}_p^m(\vec{r},t_{n+1})$. The value $\vec{\psi}_p^m(\vec{r},t_{n+1})$ is written as follows :

$$\vec{\psi}_p^m(\vec{r},t_{n+1})=\sum_i\vec{\alpha}_{pi}^m(t_{n+1})P_i(\vec{r}), \tag{5.43}$$

where $P_i(\vec{r})$ is the *i*'th order Legendre polynomial. In 3-D geometry, the polynomial part of group g flux expanded to the quadratic order is

$$\begin{aligned}\vec{\psi}_p^m(\vec{r},t_{n+1})=&\alpha_{pg0}^m+\alpha_{pg1}^mP_1\left(\frac{2x}{h_x}\right)+\alpha_{pg2}^mP_1\left(\frac{2y}{h_y}\right)+\alpha_{pg3}^mP_1\left(\frac{2z}{h_z}\right)\\&+\alpha_{pg4}^mP_2\left(\frac{2x}{h_x}\right)+\alpha_{pg5}^mP_2\left(\frac{2y}{h_y}\right)+\alpha_{pg6}^mP_2\left(\frac{2z}{h_z}\right)\\&+\alpha_{pg7}^mP_1\left(\frac{2x}{h_x}\right)P_1\left(\frac{2y}{h_y}\right)+\alpha_{pg8}^mP_1\left(\frac{2y}{h_y}\right)P_1\left(\frac{2z}{h_z}\right)+\alpha_{pg9}^mP_1\left(\frac{2z}{h_z}\right)P_1\left(\frac{2x}{h_x}\right)\\&+\alpha_{pg10}^mP_1\left(\frac{2x}{h_x}\right)P_2\left(\frac{2y}{h_y}\right)+\alpha_{pg11}^mP_1\left(\frac{2y}{h_y}\right)P_2\left(\frac{2z}{h_z}\right)+\alpha_{pg12}^mP_1\left(\frac{2z}{h_z}\right)P_2\left(\frac{2x}{h_x}\right)\\&+\alpha_{pg13}^mP_2\left(\frac{2x}{h_x}\right)P_1\left(\frac{2y}{h_y}\right)+\alpha_{pg14}^mP_2\left(\frac{2y}{h_y}\right)P_1\left(\frac{2z}{h_z}\right)+\alpha_{pg15}^mP_2\left(\frac{2z}{h_z}\right)P_1\left(\frac{2x}{h_x}\right)\\&+\alpha_{pg16}^mP_2\left(\frac{2x}{h_x}\right)P_2\left(\frac{2y}{h_y}\right)+\alpha_{pg17}^mP_2\left(\frac{2y}{h_y}\right)P_2\left(\frac{2z}{h_z}\right)+\alpha_{pg18}^mP_2\left(\frac{2z}{h_z}\right)P_2\left(\frac{2x}{h_x}\right)\end{aligned} \tag{5.44}$$

In conventional nodal methods that use the transverse integration procedure, all the quantities that do not appear in the "steady-state" equations are approximated in terms of polynomials like transverse leakages [71]. But, in the AFEN kinetics calculation with the Galerkin scheme, we do not need such an approximation in Eq.(5.42). Weighted integrals of $\vec{\psi}_a^m(\vec{r},t_{n+1})$ in the equation can be evaluated in terms of the analytic basis functions used. Also, the weighted integral of $\vec{S}^m(\vec{r},t_n)$ can be evaluated exactly by saving the weighted integrals of delayed neutron precursors and fission sources at a previous time step. Thus, we can use low-order Legendre polynomials for the correction part as in Eq.(5.44) without incurring large errors. In addition, when there is no transient, the second and third integrands in Eq.(5.42), respectively, become zero, which makes Eq.(5.35) a steady-state equation (no null transient problem [71]).

Rearranging Eq.(5.42), we obtain the following equation for $\vec{\psi}_p^m(\vec{r},t_{n+1})$:

$$\begin{aligned}
&\int\omega(\vec{r})\{-\nabla\cdot[D^m(t_{n+1})]\nabla+[A^m(\vec{r},t_{n+1})]-\vec{\chi}_0(1-\beta^m)[F^m(\vec{r},t_{n+1})]^T-[\gamma^m(t_{n+1})]\}\\
&\quad\times\vec{\psi}_p^m(\vec{r},t_{n+1})dV_m\\
&\quad=\int\omega(\vec{r})[\gamma^m(t_n)]\{\vec{\psi}_a^m(\vec{r},t_n)+\vec{\psi}_p^m(\vec{r},t_n)\}-\int\omega(\vec{r})\vec{S}^m(\vec{r},t_n)dV_m\\
&\quad+\int\omega(\vec{r})\{A_1^m(\vec{r},t_{n+1})-\chi_0F_1^m(\vec{r},t_{n+1})+\vec{\chi}_0\beta^m[F^m(\vec{r},t_{n+1})]^T+[\gamma^m(t_{n+1})]\}\\
&\quad\times\vec{\psi}_a^m(\vec{r},t_{n+1})dV_m.
\end{aligned}\tag{5.45}$$

Equations (5.37), (5.43), and (5.45) show that the coefficients of the polynomial part at the current time step can be determined in terms of the previous time-step variables and the coefficients of the analytic part at the current time step.

The coefficients of the analytic part are determined in the same manner as used in the steady-state AFEN calculation. The procedure for the AFEN kinetics calculation at time step t_{n+1} can be written as follows :

Step 1 : Assume the coefficients of $\vec{\psi}_a^m(\vec{r},t_{n+1})$ for each node.

Step 2 : Calculate the coefficients of $\vec{\psi}_p^m(\vec{r},t_{n+1})$ from Eq.(5.45) for each node.

Step 3 : Calculate the coefficients of $\vec{\psi}_a^m(\vec{r},t_{n+1})$ with those of $\vec{\psi}_p^m(\vec{r},t_{n+1})$ obtained step 2, using the following nodal equations for total flux (by iteration)

a. nodal balance equation resulting from node volume integration of Eq.(5.35).
b. current continuity equation
c. corner-point leakage balance equation.

Step 4 : Go to step 2 until the total flux converges.

During iteration, the correction term need not be updated per every step 3 iteration for the coefficients of $\vec{\psi}_a^m(\vec{r},t_{n+1})$. Numerical tests show that calculation of step 2 per every ten times of step 3 iteration is sufficient for accuracy. Also, this makes the computing time increase due to the step 2 calculation marginal. After the step 3 iteration, we update the nodal fluxes by solving the CGR equation for acceleration.

In 2-D geometry, the Laboratorium für Reaktorregelung und Anlagensicherung (LRA) boiling water reactor (BWR) benchmark problem was tested. The LRA-2D quarter-core BWR problem consists of 78 fuel assemblies, each 15 cm on a side. A complete description, including adiabatic heating and Doppler feedback models, is given in Ref.[72].

The prompt-supercritical transient is initiated by linearly decreasing the thermal absorption cross section in a portion of the core for 2 sec, and then maintaining the cross section at the new value for the remainder of the transient. The problem has proven to be extremely difficult because it has severe flux tilts that occur during the transient and requires a highly accurate spatial neutronics model. For this problem, Smith [18] pointed out that temperature shape calculation is needed for coarse-mesh models.

Table 5.1 shows the results obtained from the developed AFEN kinetics code and six conventional nodal codes. Figure 5.2 shows the peak time and power of the cases shown in Table 5.1. We applied the space-dependent feedback calculation model. Reference solution is made by AFEN using the 4 node/assembly and fine time-step calculation with space-dependent feedback. Cases from 1 to 5 show nearly the same results as the reference case, which proves that almost all the errors induced from space-dependent feedback were removed. Case 6 obtained from the 4 node/assembly and no space-dependent feedback calculation shows ~0.8% errors. Results of the 1 node/assembly calculation with space-dependent feedback calculation (cases 2, 3, 4, and 5) are much more accurate than those of the cases with the 4 node/assembly

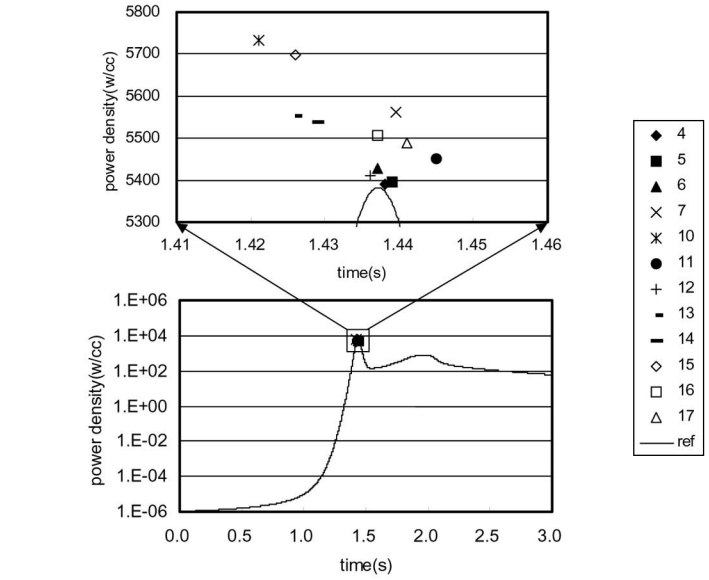
calculation without space-dependent feedback incorporated. Errors increase more (~3%) when we use the 1 node/assembly calculation without space-dependent feedback incorporated (cases 7, 8, and 9). Among the results, the AFEN kinetics method gives excellent accuracy even in the cases using the 1 node/assembly calculation. The 4 node/assembly calculation of Shober only has < 1.0% error in peak power density prediction. In case 14, although Smith used a homogenized cross section obtained from temperature shape calculation, the errors are still large compared with those of AFEN.

The results of thermal flux shapes in this benchmark problem show that almost >95% in total flux is described by the analytic part, and only a minor correction is added by the polynomial part. That is, most of the transient flux is described by the analytic basis functions.

Table 5.1 Comparison and Results of 2-D LRA BWR Problem

	case	N_m [a]	Feedback Calculation	Number of Time Steps	T_{peak} (s) (% error)	P_{peak} (W/cm^3) (% error)	Execution Time (s)
AFEN	Reference	484	Yes	9400	1.4373	5382.7	
	1	484	Yes	2600	1.4372 (-0.01)	5381.8 (-0.02)	
	2	121	Yes	2600	1.4407 (0.24)	5394.2 (0.21)	67.84
	3	121	Yes	1000	1.4405 (0.22)	5392.9 (0.19)	44.84
	4	121	Yes	329	1.4395 (0.15)	5387.8 (0.09)	27.93
	5[b]	121	Yes	329	1.4395 (0.15)	5388.7 (0.11)	25.67
	6	484	No	2600	1.4372 (-0.01)	5427.1 (0.82)	
	7	121	No	2600	1.4410 (0.26)	5562.8 (3.34)	60.50
	8	121	No	1000	1.4410 (0.26)	5562.0 (3.33)	44.57
	9	121	No	329	1.4395 (0.15)	5561.1 (3.31)	25.69
CUBBOX	10	121		1200	1.421 (-1.13)	5734 (6.53)	
IQSBOX	11	121		522	1.445 (0.54)	5451 (1.27)	
Shober	12	484		2600	1.436 (-0.09)	5411 (0.53)	
	13	121		1000	1.426 (-0.79)	5552 (3.15)	
QUANDRY	14[c]	121		329	1.429 (-0.58)	5538 (2.89)	
	15	121		329	1.426 (-0.79)	5699 (5.88)	
CONQUEST	16	121		1000	1.437 (-0.02)	5505 (2.28)	
SPANDEX	17	121		1579	1.441 (0.26)	5490 (2.00)	

[a] Number of total nodes.
[b] Linear order of the polynomial part used.
[c] Temperature shape calculated.



◆ 4 : AFEN (1 node/assembly, with space-dependent feedback, 329 time steps, quadratic correction)
■ 5 : AFEN (1 node/assembly, with space-dependent feedback, 329 time steps, linear correction)
▲ 6 : AFEN (4 node/assembly, without space-dependent feedback, 2600 time steps, quadratic correction)
+ 12 : Shober (4 node/assembly, 2600 time steps)
— ref. : AFEN (4 node/assembly, with space-dependent feedback, 9400 time steps, quadratic correction)

Figure 5.2 Peak time and power of the 2-D LRA BWR problem.

5.2 Coupled Thermal Hydraulics and Neutron Kinetics

Incorporation of full three-dimensional reactor core models into system transient analysis codes that customarily contain point- and /or one-dimensional neutron kinetics models would allow for more realistic and best-estimate calculation of interactions between the core behavior and the plant dynamics. In the last ten years, a great effort has been made in several countries and organizations in this direction [73]. The Nuclear Energy Agency (NEA) Nuclear Science Committee has developed a series of benchmark problems to instigate studies of computer codes used to obtain solutions for coupled thermal-hydraulics and three-dimensional neutron kinetics [74]. In addition to this series, the Organization of Economic Cooperation and Development (OECD) and the U.S. Nuclear Regulatory Commission (NRC) sponsored the PWR Main Steam Line Break (MSLB) benchmark problem [75] and the BWR Turbine Trip (TT) benchmark problem. The coupled codes used in these activities include RELAP5/NEM, TRAC-PF1, -BF1/NEM, RELAP5/PARCS, TRAC-M/PARCS, MARS/ MASTER, RELAP5/PANBOX, CATHARE/CRONOS, ATHLET/QUABOX-CUBBOX, ATHLET/DYN3D, SAS/DIF3DK, and so on (see the special issues of *Nuclear Technology*, Vol.142, No.2, May 2003, and *Nuclear Science and Engineering*, Vol.148, No.2, October 2004). Currently in progress is the VVER-1000 coolant transient benchmark problem [76] sponsored by the OECD/NEA Nuclear Science Committee and the U.S. Department of Energy (DOE).

The space-time kinetics models in most of code systems use the direct methods based on implicit temporal discretization and nodal spatial discretization. Exceptions are SAS/DIF3DK and RELAP5/PANBOX in which space-time factorization is used. It is worth noting that RELAP5/PANBOX uses dimensionally adaptive neutron kinetics based on sophisticated automatic switching algorithms [77, 78] to change between three-dimensional, one-dimensional, and point kinetic models during a transient, resulting in a substantial reduction in computing time without sacrificing accuracy.

Evidently, there are many parameters and issues to consider that will affect the accuracy of the solutions from the coupled code systems. The important issues and considerations that are identified in these studies may be classified as follows :

i) mode of coupling of the neutron kinetics code with the thermal-hydraulics code : internal vs external coupling, basically depending on whether or not in-core thermal-hydraulics calculation (e.g., fuel temperature) is done in the system analysis code.
ii) mapping of nodalizations between codes : neutron kinetics codes usually use one node/fuel assembly (sometimes four nodes/fuel assembly) but TH channels used tend to be much coarser.
iii) treatment of coolant mixing in the inlet and outlet plena.

iv) treatment of cross flows between channels.
v) cross section tabulation and interpolation procedures.
vi) temporal coupling and time step size, and coupled convergence strategies.

As a representative example, Figures 5.3, through 5.6 show options in coupling modes and nodalization mappings used in RELAP5/PANBOX.

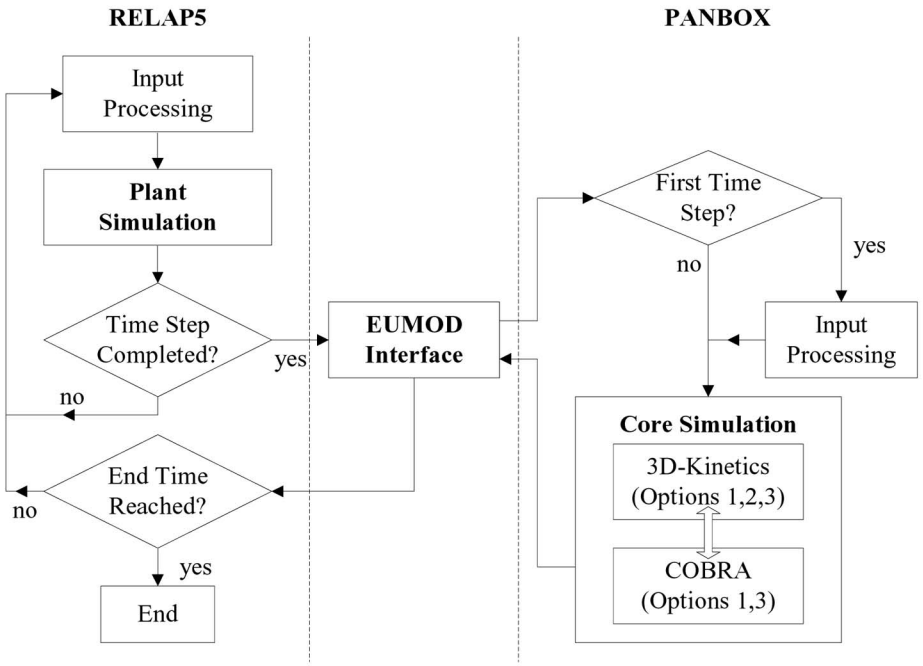


Figure 5.3 Simplified flow logic of RELAP5/PANBOX. (from Ref.[79])

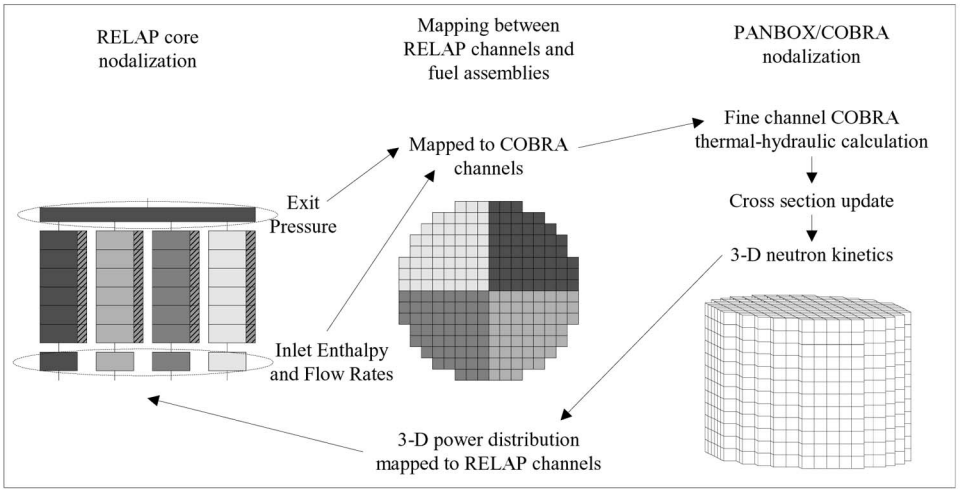


Figure 5.4 Depiction of the first coupling option for RELAP5/PANBOX : cross section update from COBRA. (from Ref.[77])

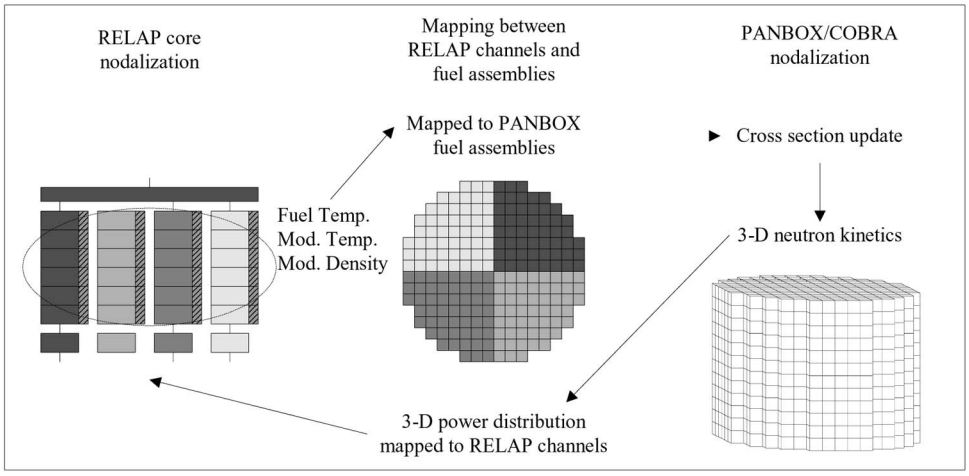


Figure 5.5 Depiction of the second coupling option for RELAP5/PANBOX : no COBRA calculation, cross section update from RELAP data. (from Ref.[77])

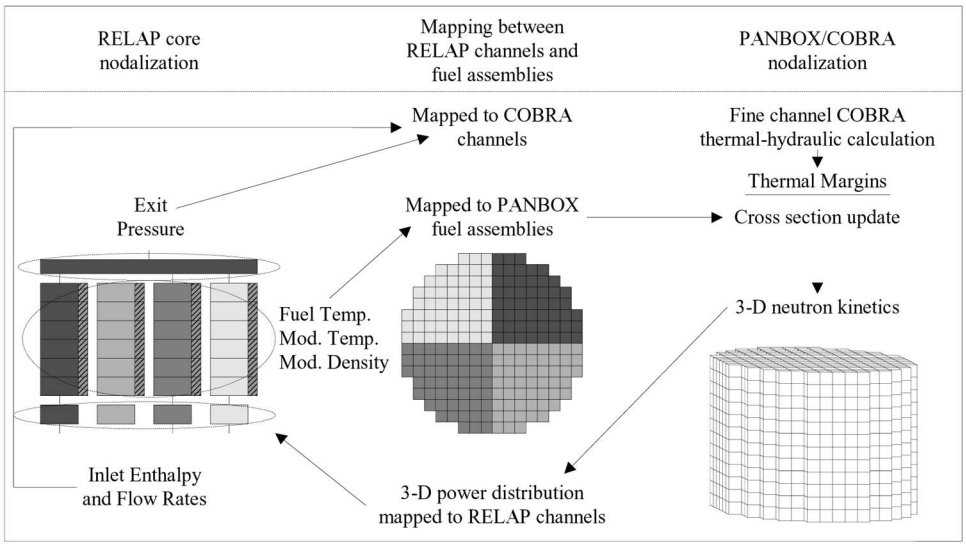


Figure 5.6 Depiction of coupling option 3 for RELAP5/PANBOX : cross section update from RELAP, parallel COBRA calculation for thermal margins. (from Ref.[77])

Although the coupling schemes in most of the codes are in general explicit, the sequence in which the system thermal-hydraulics, in-core thermal-hydraulics (i.e., heat conduction), and neutronics equations are solved could be modified to increase the "implicitness", and thus enhances the numerical stability, eventually leading to computer time reduction.

Kastanya and Turinsky [80, 81] proposed such a method based on a Newton-BiCGSTAB solver to treat strong nonlinearity due to thermal feedback in a BWR simulator. For example, flux and temperature (even multiplication factor in the case of steady state calculation) are related in a system of coupled nonlinear equations. In this approach, the flux and temperature are calculated simultaneously at the same time step in a Newton iteration. Gan, Xu and Downar [82] report preliminary results of applying this Newton-Krylov approach to the coupled code system TRAC-M/PARCS. They take an advantage of the fact that an attractive feature of the Krylov subspace methods is that only the action of the Jacobian on a vector is required and it is not necessary to form the Jacobian explicitly.

6. CHALLENGES FOR THE FUTURE

As we have seen above (at least partially), the major design tools of current generation nuclear reactors are built upon a very sophisticated fabric of analysis methods and computer codes. Although the current methods implemented in

production codes are generally adequate for designs of standard reactors and fuels, improvement and refinement are warranted in several areas for further enhancement of their economy and safety. In particular, for the design of evolutionary advanced reactors such as reactors loaded with annular fuels, axially graded fuels, UO_2-MOX fuel mixed cores with increased MOX fuel fraction, and the cores loaded with inert-matrix fuels, the areas considered to be worthwhile pursuing research and development are:

i) Use of parameter estimation-free methods such as Krylov subspace methods, replacing the currently used power method, successive overrelaxation (SOR), and so on, in reactor analysis codes.
ii) Incorporation of anisotropic scattering beyond the transport-corrected cross sections.
iii) Departure from the current practice of single-assembly calculation with zero net current boundary condition for homogenization.
iv) Heterogeneous treatment of the fuel cell without fuel/coolant homogenization.
v) Use of whole-core heterogeneous transport calculation, replacing the diffusion nodal methods currently in use. This is also related to items ii), iii), and iv) above. A good candidate for this whole-core heterogeneous transport calculation would be the method of characteristics (MOC) with good acceleration methods.
vi) Increasing use of Monte Carlo technique (or its advancement) for complicated geometry and continuous energy treatment (in particular, allowing more satisfactory treatment of resonance cross sections) to amend the multigroup deterministic methods.

In the face of new next-generation reactor concepts and innovative fuel designs under development, e.g., as in the so-called Generation IV nuclear energy systems, more challenging tasks are awaiting :

i) Power peaking control and rigorous reactivity analysis methods of subcritical accelerator-driven systems.
ii) Treatment of moving fuels in a pebble bed reactor or in a molten salt reactor.
iii) Double heterogeneity treatment for very high temperature gas reactor (VHTR) cores; randomly dispersed coated fuel particles in a pebble or in a fuel compact, random packing of pebbles in a pebble bed reactor, fuel compact channels in a graphite assembly of a prismatic reactor, neutron streaming between pebbles or along coolant channels.
iv) Burnup tracking of fuel pebbles in a pebble reactor, and shuffling schemes of fuel blocks in a prismatic reactor.

The items in the list above are not of course comprehensive and not necessarily in order of importance or priority.

As a key and core knowledge for the design of various types of nuclear reactors, the discipline of reactor physics and kinetics has advanced remarkably in the past five decades or so and will continue to do so in the future. The reactor physicists, senior and junior alike, face interesting challenges and thus rewarding opportunities in the years to come.

Note: This article is based on a revision of the lecture notes originally presented by the author at the Frederic Joliot / Otto Hahn Summer School in Reactor Physics, August 20-29, 2003, Karlsruhe, Germany.

Acknowledgment

This work was supported in part by the Ministry of Science and Technology of Korea through the National Research Laboratory (NRL) Program.

REFERENCES

[1] R.J.J. Stamm'ler and M.J. Abbate, Methods of Steady-State Reactor Physics in Nuclear Design, Academic Press, 1983.
[2] E.E. Lewis and W.F. Miller, Jr., Computational Methods of Neutron Transport, John Wiley & Sons, 1984.
[3] G.I. Bell and S. Glasstone, Nuclear Reactor Theory, Van Nostrand Reinhold, 1970.
[4] J.J. Duderstadt and L.J. Hamilton, Nuclear Reactor Analysis, John Wiley & Sons, 1976.
[5] R.J. Brissenden and A.R. Garlick, "Biases in the Estimation of and Its Error by Monte Carlo Methods," Ann. Nucl. Energy, 13, 63 (1986).
[6] E.M. Gelbard and A.G. Gu, "Biases in Monte Carlo Eigenvalue Calculations," Nucl. Sci. Eng., 117, 1 (1994).
[7] T. Ueki, T. Mori, and M. Nakagawa, "Error Estimates and Their Biases in Monte Carlo Eigenvalue Calculations," Nucl. Sci. Eng., 125, 1 (1997).
[8] R.N. Blomquist and E.M. Gelbard, "Alternative Implementations of the Monte Carlo Power Method, " Nucl. Sci. Eng., 141, 85 (2003).
[9] J.D. Densmore and E.W. Larsen, "Variational Variance Reduction for Monte Carlo Eigenvalue and Eigenfunction Problems," Nucl. Sci. Eng., 146, 121 (2004).
[10] A.F. Henry, Nuclear Reactor Analysis, The MIT Press, 1975.
[11] S. Nakamura, Computational Methods in Engineering and Science, John Wiley & Sons, 1997.
[12] E.L. Wachspress, Iterative Solution of Elliptic Systems, Prentice-Hall, 1966.
[13] G. Verdu, R. Miro, D. Ginestar, and V. Vidal, "The Implicit Restarted Arnoldi Method, An Efficient Alternative to Solve the Neutron Diffusion Equation," Ann. Nucl. Energy, 26, 579,

(1999).
[14] M.L. Adams and E.W. Larsen, "Fast Iterative Methods for Discrete-Ordinates Particle Transport Calculations," Progress in Nuclear Energy, 40, 3 (2002).
[15] K. Koebke, "A New approach to Homogenization and Group Condensation," Proc. Specialists' Mtg. Homogenization Methods in Reactor Physics, Lugano, Switzerland, November 13-15, 1978, IAEA-TECDOC-232, p. 303, International Atomic Energy Agency (1978).
[16] K.S. Smith, "Spatial Homogenization Methods for Light Water Reactor Analysis," PhD Thesis, Massachusetts Institute of Technology (1980).
[17] H. Finnemann, F. Bennewitz and M. Wagner, "Interface Current Techniques for Multidimensional Reactor Calculations," Atomkernenergie, 30, 123 (1977).
[18] K.S. Smith, "An Analytic Nodal Method for Solving the Two-Group, Multidimensional, Static and Transient Neutron Diffusion Equations," Thesis, Nuclear Engineering, Massachusetts Institute of Technology (1979).
[19] Y.A. Chao and N. Tsoulfanidis, "Conformal Mapping and Hexagonal Nodal Methods-I: Mathematical Foundation," Nucl. Sci. Eng., 121, 202 (1995).
[20] J.M. Noh and N.Z. Cho, " A New Approach of Analytic Basis Function Expansion to Neutron Diffusion Calculation," Nucl. Sci. Eng., 116, 165 (1994).
[21] N.Z. Cho and J.M. Noh, "Analytic Function Expansion Nodal Method for Hexagonal Geometry," Nucl. Sci. Eng., 121, 245 (1995).
[22] N.Z. Cho, Y.H. Kim, and K.W. Park, "Extension of Analytic Function Expansion Nodal Method to Multigroup Problems in Hexagonal-Z Geometry," Nucl. Sci. Eng., 126, 35 (1997).
[23] S.W. Woo, N.Z. Cho, and J.M. Noh, "The Analytic Function Expansion Nodal Method Refined with Transverse Gradient Basis Functions and Interface Flux Moments," Nucl. Sci. Eng., 139, 156 (2001).
[24] D.S. Kim and N.Z. Cho, "Kinetics Calculation Under Space-Dependent Feedback in Analytic Function Expansion Nodal Method via Solution Decomposition and Galerkin Scheme," Nucl. Sci. Eng., 140, 267 (2002).
[25] D.S. Kim and N.Z. Cho, "The Analytic Function Expansion Nodal (AFEN) Method with Half-Interface Averaged Fluxes in Mixed Geometry Nodes for Analysis of Pebble Bed Modular Reactor (PBMR) Cores," Journal of Nuclear Science and Technology, 40, 291 (2003).
[26] N.Z. Cho, et al, "Improved Features and Verification of an AFEN Method Code in Hexagonal-Z 3-D Geometry for Neutron Diffusion Calculation," submitted to American Nuclear Society Annual Meeting, San Diego, CA, June 2005.
[27] H.G. Joo, et al., "User Manual for the PARCS-v2.3 Beta Kinetics Core Simulator Module," Purdue University, March 2002.
[28] J.M. Noh and N.Z. Cho, "Intranodal Burnup Gradient Correction in Analytic Function Expansion Nodal (AFEN) Method," Proceedings of the International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analysis, Vol.2, pp.1435-1444, April 1995, Portland, U.S.A.
[29] N.Z. Cho and J.M. Noh, "Hybrid of AFEN and PEN Methods for Multigroup Diffusion Nodal Calculation," Trans. Am. Nucl. Soc., 73, 438 (1995).
[30] H.C. Lee and C.H. Kim, "Unified Formulation of Nodal Expansion Method and Analytic Nodal Method Solutions to Two-Group Diffusion Equations," Nucl. Sci. Eng., 138, 192 (2001).
[31] H.C. Lee and C.H. Kim, "Unified Nodal Method Formulation for Analytic Function Expansion Nodal Method Solution to Two-Group Diffusion Equations in Rectangular Geometry," Nucl. Sci. Eng., 140, 137 (2002).
[32] P.J. Turinsky, et al., "NESTLE: A Few-Group Neutron Diffusion Equation Solver," EGG-NRE-11406, Electric Power Research Center, North Carolina State University (1994).
[33] K.S. Smith, "Nodal Method Storage Reduction by Nonlinear Iteration," Trans. Am. Nucl. Soc., 44, 265 (1983).
[34] T.M. Sutton, "NODEX: A High Order NEM-Based Multigroup Nodal Code," Proc. Topical Meeting on Advances in Nuclear Engineering Computations and Radiation Shielding, 38:1-3:11. Santa Fe, New Mexico, April 9-13 (1989).
[35] K.S. Moon, N.Z. Cho, J.M. Noh, and S.G. Hong, "Acceleration of the Analytic Function Expansion Nodal Method by Two-Factor Two-Node Nonlinear Iteration," Nucl. Sci. Eng., 132, 194 (1999).
[36] D.S. Kim and N.Z. Cho, "Acceleration of Three-Dimensional AFEN Nodal Codes via Coarse Group Rebalance and Direct Matrix Inverse," Proc. Int. Conf. Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications, p.168, Madrid, Spain, September 27-30, 1999, and references therein.
[37] J.R. Askew, "A Characteristics Formulation of the Neutron Transport Equation in Complicated Geometries," AEEW-R-1108, U.K. Atomic Energy Authority (1972).
[38] M.J. Halsall, "CACTUS, A Characteristics Solutions to the Neutron Transport Equations in Complicated Geometries," AEEW-R-1291, U.K. Atomic Energy Authority (1980).
[39] A. Leonard and C.T. McDaniel, "Optimal Polar Angles and Weights," Trans. Am. Nucl. Soc., 73, 171 (1995).
[40] W.L. Filippone, S. Woolf, and R.J. Lavigne, "Particle Transport Calculations with the Method of Streaming Rays," Nucl. Sci. Eng., 77, 119 (1981).
[41] S.G. Hong and N.Z. Cho, "CRX: A Code for Rectangular and Hexagonal Lattices Based on the Method of Characteristics," Ann. Nucl. Energy, 25, 547 (1998).
[42] N.Z. Cho, et al., "Whole-Core Heterogeneous Transport Calculations and Their Comparison with Diffusion Results," Trans. Am. Nucl. Soc., 83, 292 (2000).
[43] S. Kosaka and E. Saji, "The Characteristics Transport Calculation for a Multi-Assembly System using Neutron Path Linking Technique," Proc. Int. Conf. Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications, p.1890, Madrid, Spain, September 27-30, 1999.
[44] G.S. Lee, N.Z. Cho, and S.G. Hong, "Acceleration and Parallelization of the Method of Characteristics for Lattice and Whole-Core Heterogeneous Calculations," Proc. PHYSOR 2000, Session II-C, Pittsburgh, USA, May 7-11, 2000.
[45] M.A. Smith, et al., "Whole-Core Neutron Transport Calculations Without Fuel-Coolant Homogenization," Proc. PHYSOR 2000, Session II-C, Pittsburgh, USA, May 7-11, 2000.
[46] N.Z. Cho, G.S. Lee, C.J. Park, "Fusion of Method of Characteristics and Nodal Method for 3-D Whole-Core

Transport Calculation," Trans. Am. Nucl. Soc., 86, 322 (2002).
[47] N.Z. Cho, et al., "Refinement of the 2-D/1-D Fusion Method for 3-D Whole-Core Transport Calculation," Trans. Am. Nucl. Soc., 87, 417 (2002).
[48] J.Y. Cho, et al., "Three-Dimensional Heterogeneous Whole Core Transport Calculation Employing Planar MOC Solutions," Trans. Am. Nucl. Soc., 87, 234 (2002).
[49] S. Kosaka and T. Takeda, "Diffusion-Like 3-D Heterogeneous Core Calculation with 2-D Characteristics Transport Correction by Non-Linear Iteration Technique," Int. Conf. Nuclear Mathematical and Computational Sciences (M&C 2003), Gatlinburg, USA, April 6-11, 2003, CD-ROM, American Nuclear Society (2003).
[50] E.E. Lewis, et al., Expert Group on 3-D Radiation Transport Benchmarks – Benchmark Specification for Deterministic MOX Fuel Assembly Transport Calculations without Spatial Homoginisation (3-D Extension C5G7 MOX), NEA/NSC/DOC (2003)6, 2003.
[51] Y. Azmy, "Impossibility of Unconditional Stability and Robustness of Diffusive Acceleration Schemes," American Nuclear Society Radiation Protection and Shielding Division Topical Meeting, p. 480, Nashville, TN, U.S.A., April 19-23, 1998.
[52] Y. Azmy, T. A. Wareing, and J. Morel, "Effect of Material Heterogeneity on the Performance of DSA for Even-Parity SN Methods," International Conference on Mathematics and Computation, Reactor Physics, and Environmental Analysis in Nuclear Applications, p.55, Madrid, Spain, September 27-30, 1999.
[53] J. S. Warsa, T. A. Wareing, J. E. Morel, J. M. McGhee, and R. B. Lehoucq, "Krylov Subspace Iterations for the Calculation of k-Eigenvalues with Sn Transport Codes," Int. Conf. Nuclear Mathematical and Computational Sciences (M&C 2003), Gatlinburg, USA, April 6-11, 2003, CD-ROM, American Nuclear Society (2003).
[54] K.S. Smith, J.D. Rhodes, III, "CASMO Characteristics Method for Two-Dimensional PWR and BWR Core Calculation," Trans. Am. Nucl. Soc., 83, 294, (2000).
[55] K.S. Smith, J.D. Rhodes, III, "Full-Core, 2-D, LWR Core Calculations with CASMO-4E," Proc. Int. Conf. New Frontiers of Nuclear Technology: Reactor Physics, Safety and High- Performance Computing (PHYSOR 2002), Seoul, Korea, October 7-10, 2002, CD-ROM, American Nuclear Society (2002).
[56] J.Y. Cho, et al., "Cell-Based CMFD Formulation for Acceleration of Whole-Core Method of Characteristics Calculations," Journal of the Korean Nuclear Society, 34, 250 (2002).
[57] H.G. Joo, et al., "Dynamic Implementation of the Equivalence Theory in the Heterogeneous Whole Core Transport Calculation," Proc. Int. Conf. New Frontiers of Nuclear Technology: Reactor Physics, Safety and High- Performance Computing (PHYSOR 2002), Seoul, Korea, October 7-10, 2002, CD-ROM, American Nuclear Society (2002).
[58] N.Z. Cho, C.J. Park, "A Comparison of Coarse Mesh Rebalance and Coarse Mesh Finite Difference Accelerations for the Neutron Transport Calculations," Int. Conf. Nuclear Mathematical and Computational Sciences (M&C 2003), Gatlinburg, USA, April 6-11, 2003, CD-ROM, American Nuclear Society (2003).
[59] N.Z. Cho, G.S. Lee, C.J. Park, "Partial Current-Based CMFD Acceleration of the 2D/1D Fusion method for 3D Whole-Core Transport Calculations," Trans. Am. Nucl. Soc., 88, 594, (2003).
[60] S.G. Hong and N.Z. Cho, "A Rebalance Approach to Nonlinear Iteration for Solving the Neutron Transport Equations," Ann. Nucl. Energy, 24, 147 (1997).
[61] S.G. Hong and N.Z. Cho, "Convergence Analysis of the Angular-Dependent Rebalance Iteration Method in X-Y Geometry," Trans. Am. Nucl. Soc., 80, 115 (1999).
[62] Y.R. Park and N.Z. Cho, "Coarse-Mesh Angular Dependent Rebalance Acceleration of the Discrete Ordinate Transport Calculations," Nucl. Sci. Eng., 148, 355 (2004).
[63] Y.R. Park and N.Z. Cho, "The MOC Neutron Transport Calculations Accelerated by Coarse-Mesh Angular Dependent Rebalance," Proc. Korean Nucl. Soc., Autumn Mtg., Yongpyong, October 2004.
[64] A. Pautz and A. Birkhofer, "A Time-Dependent Neutron Transport Code Coupled with the Thermal-Hydraulics Code ATHLET," Proc. Int. Conf. New Frontiers of Nuclear Technology: Reactor Physics, Safety and High- Performance Computing (PHYSOR 2002), Seoul, Korea, October 7-10, 2002, CD-ROM, American Nuclear Society (2002).
[65] A.F. Henry, "The Application of Reactor Kinetics to the Analysis of Experiments," Nucl . Sci. Eng., 3, 52 (1958).
[66] K. Ott, "Quasistatic Treatment of Spatial Phenomena in Reactor Dynamics," Nucl . Sci. Eng., 26, 563 (1966).
[67] K. Ott and D.A. Meneley, "Accuracy of the Quasistatic Treatment of Spatial Reactor Kinetics," Nucl . Sci. Eng., 36, 402 (1969).
[67] W.H. Reed and K.F. Hansen, "Alternating Direction Methods for the Reactor Kinetics Equations," Nucl . Sci. Eng., 41, 431 (1970).
[68] W.M. Stacey, Jr., Space-Time Nuclear Reactor Kinetics, Academic Press, 1969.
[69] T.M. Sutton and B.N. Aviles, "Diffusion Theory Methods for Spatial Kinetics Calculations," Prog. Nucle. Energy, 30,119 (1996).
[70] P.R. Engrand, G.I. Maldonado, R. Al-Chalabi, and P.J. Turinsky, "Nonlinear Iteration Strategy for NEM : Refinement and Extension," Trans. Am. Nucl. Soc., 65, 221, (1992).
[71] "Argonne Code Center : Benchmark Problem Book," ANL-7416, Suppl. 2, Argonne National Laboratory (1977).
[72] K.N. Ivanov and N.K. Todorova, "Using the OECD/NRC Pressurized Water Reactor Main Steam Line Break Benchmark to Study Current Numerical and Computational Issues of Coupled Calculations," Nucl. Technol., 142, 95 (2003).
[73] H.B. Finnemann and A. Galati, "NEACRP 3-DLWR Core Transient Benchmark, Final Specifications," NEACRP-L-335 (Rev. 1), Nuclear Energy Agency Committee on Reactor Physics (Jan. 1992).
[74] K.N. Ivanov, et al., "PWR MSLB Benchmark. Volume 1 : Final Specifications," NEA/NSC/DOC (99)8, Nuclear Energy Agency/Nuclear Science Committee (Apr. 1999).
[75] B. Ivanov, K. Ivanov, P. Groudev, M. Pavlova and V. Hadjiev, "VVER-1000 Coolant Transient Benchmark – PHASE I (V1000CT-1), Volume I : Final Specifications," NEA/NSC/DOC, 6 (2002).
[76] C.J. Jackson, D.G. Cacuci, and H.B. Finnemann, "Dimensionally Adaptive Neutron Kinetics for

Multidimensional Reactor Safety Transients – 1 : New Features of RELAP5/PANBOX," Nucl. Sci. Eng., 131, 143 (1999).
[77] C.J. Jackson, D.G. Cacuci, and H.B. Finnemann, "Dimensionally Adaptive Neutron Kinetics for Multidimensional Reactor Safety Transients – II : Dimensionally Adaptive Switching Algorithms," Nucl . Sci. Eng., 131, 164 (1999).
[78] R. Böer and A. Knoll, "Sensitivity Studies for Main Steam Line Break Exercises 2 and 3 with RELAP5/PANBOX," Nucl. Technol., 142, 137 (2003).
[79] D. Kastanya and P.J. Turinsky, "Development of a Newton-Krylov Solver in FORMOSA-B," Trans. Am. Nucl. Soc., 86, 226, (2002).
[80] D. Kastanya and P.J. Turinsky, "Implementation of a Newton-BiCGSTAB Solver to Treat the Strong Non-Linearity in the FORMOSA-B Boiling Water Reactor Core Simulator Code,"
[81] Int. Conf. Nuclear Mathematical and Computational Sciences (M&C 2003), Gatlinburg, USA, April 6-11, 2003, CD-ROM, American Nuclear Society (2003).
[82] J. Gan, Y. Xu, and T.J. Downar, "A Matrix-Free Newton Method for Coupled Neutronics Thermal-Hydraulics Reactor Analysis," Int. Conf. Nuclear Mathematical and Computational Sciences (M&C 2003), Gatlinburg, USA, April 6-11, 2003, CD-ROM, American Nuclear Society (2003).