A Dimensionally Adaptive Neutron Kinetics Algorithm for Efficient Nuclear Plant Safety Analysis Calculations

C. J. Jackson

Institut für Reaktorsicherheit

Forschungszentrum Karlsruhe
Technik und Umwelt
Wissenschaftliche Berichte
FZKA 5922

A Dimensionally Adaptive Neutron Kinetics Algorithm for Efficient Nuclear Plant Safety Analysis Calculations*

Christopher John Jackson

Institut für Reaktorsicherheit

Forschungszentrum Karlsruhe GmbH, Karlsruhe

1997

Abstract

The coupled RELAP5/PANBOX code has been developed for the analysis of postulated nuclear plant accidents which lead to significant changes in core reactivity and power distribution. The simple point kinetics model in RELAP5 is replaced with the core simulation program PANBOX, by using a general interface routine for RELAP developed at Siemens. Instead of the simple point kinetics model, RELAP5/PANBOX now solves the time dependent, three-dimensional multigroup diffusion equations to simulate the reactor core. With the new code, transients in which local or asymmetric reactivity contributions are important can now be more accurately analyzed. In order to be able to identify such transients, the option has also been created to examine reactivity contributions from separate physical effects. In particular, the reactivity contribution of the redistribution of the neutron flux can now be calculated. When this contribution is relatively large, it is necessary to use a three-dimensional neutron kinetics model.

The main goal of this work was the development of a dimensionally adaptive neutron kinetics algorithm, that automatically and adaptively switches between three-dimensional, one-dimensional, and point kinetics models. In order to develop this algorithm, it was first necessary to develop point- and one-dimensional models that were consistent with the three-dimensional model. The point kinetics model which was implemented was taken directly out of the literature. Perturbation theory is used to calculate the core reactivity more accurately, and is formulated using the nodal expansion method (NEM) solution of the three-dimensional flux. The continuous form of the one-dimensional model is, similar to the point kinetics model, derived from the factorization of the three-dimensional flux into a shape function and a one-dimensional amplitude function. The three dimensional neutron diffusion equations are then integrated over the plane perpendicular to the reactor axis. The resulting one-dimensional differential equations are then discretized with the NEM. Correction factors, which resemble the well-known heterogeneity or discontinuity factors, are defined in order to hold the one-dimensional solution equivalent to the three-dimensional solution.

The three-dimensional and one-dimensional models, and also perturbation theory, are closely tied with the NEM solution. It is therefore necessary to discuss the theoretical foundations of the NEM. Therefore, the consistency of the NEM is proved, and stability criteria for the different kinetics models are developed.

Criteria for activation of the different kinetics models are also derived. The criteria for switching from the three-dimensional to the one-dimensional model and from the one-dimensional to the point kinetics model are determined with the time variation of the shape function. The criteria are motivated from the fact that the one-dimensional and point kinetics models are derived from the three-dimensional model using the adiabatic quasi-static approximation. The time variation of the shape functions can not be accurately determined during time periods in which a lower dimensional models is active. For this reason, the reactivation of the three-dimensional model is triggered by different complementary criteria. One criterion is determined from the extrapolation of the last known time derivative of the shape function. An error estimation procedure, adapted from those used for finite element methods for the NEM, is also derived. Additional criteria are developed which are based on the reactivity and absolute changes in the reactivity.

Example calculations have shown that the adaptive algorithm produces satisfactorily accurate results, with from 30% to 70% less computation time then reference cases calculated with only a three-dimensional model. The transients which were used as examples were all characterized by a relatively large redistribution of the neutron flux. The dimensionally adaptive algorithm would likely use even less computation time for transients with less flux redistribution. The results generated by the adaptive algorithm were all slightly shifted in time in comparison to the reference calculations. If these shifts in time are tolerable, then the adaptive algorithm can be considered to deliver very accurate results.

Ein effizientes und in den räumlichen Dimensionen adaptives Verfahren für neutronenkinetische Berechnungen im Rahmen von Reaktorsicherheitsanalysen

Zusammenfassung

Das gekoppelte Rechenprogramm RELAP5/PANBOX ist für die Analyse von postulierten Störfällen in Kernkraftwerken entwickelt worden, die zu signifikanten Reaktivitätsänderungen und Leistungsumverteilungen im Reaktorkern führen. Über eine allgemeine von Siemens für RELAP5 entwickelte Schnittstelle ist im Rahmen dieser Arbeit das einfache punktkinetische Modell von RELAP5 durch den Kernreaktorsimulator PANBOX ersetzt worden. Statt des einfachen punktkinetischen Modells löst jetzt RELAP5/PANBOX die zeitabhängige dreidimensionale Mehrgruppen-Neutronendiffusionsgleichung zur Simulation des Reaktorkerns. Mit diesem Rechenprogramm können jetzt Störfälle, in denen lokale oder asymmetrische Reaktivitätsbeiträge wichtig sind, genauer analysiert werden. Um solche Störfälle identifizieren zu können, ist auch die Möglichkeit geschaffen worden, einzelne Komponenten der Reaktivität zu untersuchen. Insbesondere kann der Reaktivitätsbeitrag der Umverteilung des Neutronenflusses berechnet werden. Wenn dieser Beitrag relativ groß ist, dann ist ein dreidimensionales neutronenkinetisches Modell notwendig.

Das Hauptziel dieser Arbeit war die Entwicklung eines dimensional-adaptiven Kernmodells, das zwischen dreidimensionalen, eindimensionalen und punktkinetischen Modellen automatisch umschalten kann. Um das Verfahren zu realisieren, mußten zuerst punkt- und eindimensionale neutronenkinetische Modelle entwickelt werden, die mit der dreidimensionalen Lösung übereinstimmen. Die hier implementierte Version des punktkinetischen Modells konnte direkt aus der Literatur übernommen werden. Die Störungstheorie, die verwendet wird, um die Reaktivität genauer zu bestimmen, basiert auf der Lösung des dreidimensionalen Flusses mit der Nodalen Entwicklungs-Methode (NEM). Die stetige Form des eindimensionalen Modells ist, ähnlich wie beim punktkinetischen Modell, durch die Separation des dreidimensionalen Neutronflusses in eine Formfunktion und eine eindimensionale Amplitudenfunktion gewonnen worden. Die dreidimensionalen neutronenkinetischen Diffusionsgleichungen werden dann über die Ebene senkrecht zur Reaktorachse integriert. Die resultierenden eindimensionalen stetigen Gleichungen werden weiter mit NEM diskretisiert. Korrekturfaktoren, die den schon bekannten Heterogenitäts- oder Diskontinuitätsfaktoren ähneln, werden definiert, um die eindimensionale Lösung äquivalent mit der dreidimensionalen Lösung zu halten.

Dreidimensionale und eindimensionale Modelle und auch die Störungstheorie sind eng mit NEM verbunden. Es ist deshalb erforderlich, die theoretischen Grundlagen von NEM zu erörtern. Insbesondere wird die Konsistenz von NEM hier bewiesen; auch sind Stabilitätskriterien für die verschiedenen kinetischen Modelle entwickelt worden.

Kriterien zur Aktivierung der verschiedenen Modelle sind auch entwickelt worden. Die Kriterien zur Umschaltung vom dreidimensionalen auf das eindimensionale Modell bzw. vom eindimensionalen auf das punktkinetische Modell werden durch die Formfunktionen festgelegt. Die Kriterien ergeben sich daraus, daß eindimensionale und punktkinetische Modelle durch die quasi-statische adiabatische Näherung aus der dreidimensionalen Lösung abgeleitet worden sind. Während der Zeit, in der eine niedrigdimensionale Lösung aktiviert ist, kann die Zeitableitung der entsprechenden Formfunktion nur ungenau bestimmt werden. Deshalb kann die Wiedereinschaltung des dreidimensionalen Modells von verschiedenen sich ergänzenden Kriterien ausgelöst werden. Ein Kriterium wird durch die Extrapolation der letzten bekannten Zeitableitung der Formfunktion bestimmt. Auch ist ein Fehlerschätzer entwickelt worden, der von der Finiten Elemente Methode auf NEM adaptiert wurde. Zusätzlich sind Kriterien entwickelt worden, die auf absoluten und relativen Änderungen der Reaktivität beruhen.

Beispielrechnungen haben gezeigt, daß das adaptive Verfahren ausreichend genaue Ergebnisse liefert, und zwar mit 30% bis zu 70% weniger Rechenzeit. Die Störfälle, die als Beispiele benutzt worden sind, werden alle durch eine relativ große Umverteilung des Flusses charakterisiert. Für Störfälle mit geringerer Umverteilung werden die Rechenzeiten vermutlich noch kleiner. Die Ergebnisse, die mit Hilfe des adaptiven Verfahrens gewonnen werden, werden im Vergleich mit dreidimensionalen Referenzrechnungen meist zeitlich etwas verschoben. Sind diese Verschiebungen tolerierbar, können die Ergebnisse des adaptiven Verfahrens als sehr genau betrachtet werden.

Table of Contents

III. Notation

Lowercase Roman

$a^m_{1gu}, a^m_{2gu}, a^m_{3gu}, a^m_{4gu}$ – NEM flux expansion coefficients, for group g, direction u, and node m

a^m_u – the dimension of the node with index m, parallel to the u axis, $u \in \{x, y, z, \xi, \eta, \varsigma\}$

d^m_{gL}, d^m_{gR} – heterogeneity ('discontinuity') factors for node m, group g

e_g – absolute error in the flux

$f^{/m}_{gL}, f^{/m}_{gR}$ – correction factors to force equivalence between 1–D and 3–D NEM for node of index m, group g

j^{-m}_{gui} – 3–D model face averaged negative partial current of group g, on the nodal face Θ^m_{ui}

j^{+m}_{gui} – 3–D model face averaged positive partial current of group g, on the nodal face Θ^m_{ui}

t – time

$\vec{r}$ – position vector in 3–D space

s_i – eigenvalues of the matrix under discussion for the stability analysis

v_g – velocity for neutron group g

x, y – Cartesian coordinates perpindicular to z

z – axial coordinate

Uppercase Roman

A – matrix operator used for stability analysis

A – combined neutron absorption and scattering operator,

$$A\phi(\vec{r}, E, t) \equiv \Sigma_r(\vec{r}, E, t)\phi_r(\vec{r}, E, t) - \int_0^\infty \Sigma_s(\vec{r}, E' \rightarrow E, t)\phi(\vec{r}, E', t)dE'$$

A_0 – A for steady–state calculation

A_g, A_{g0} – the multigroup form of A and A_0

$\overline{B}_g$ – radial buckling coefficient for the axial kinetics model

B, B$_0$, B$_1$ – matrix operators used for the stability analysis

C(Δt) – amplification matrix used for the stability analysis

C_i – precursor amplitude function for precursor group i

C^j_i – C_i at time step j

C^{PK}_i – precursor amplitude for the point kinetics equations

$C^L_{1gu}, C^L_{2gu}, C^L_{3gu}, C^L_{4gu}$ – coefficients of the NEM outgoing partial current equation for the

3–D model, for group g, direction u, on the left side of the node (node index m is suppressed)

C_{1gu}^{R}, C_{2gu}^{R}, C_{3gu}^{R}, C_{4gu}^{R} – as above, for right side of the node

C_{1gL}, C_{2gL}, C_{3gL}, C_{4gL}, C_{1gR}, C_{2gR}, C_{3gR}, C_{4gR} – coefficients of the NEM outgoing current equations for the 1–D model, see Appendix C.

D – matrix operators used for the stability analysis

D_g^m – homogenized diffusion coefficient for the node of index m and energy group g

E – energy

$F_{gg'}^m$ – fission source operator for the node of index m, which gives the source of fission neutrons in group g from a flux of neutrons in group g', $F_{gg'}^m \equiv \sum_{j=1}^{N_J} \chi_g^j \nu^j \Sigma_{fg'}^{j,m}$

G – matrix operator used for the stability analysis

H– matrix operator used for the stability analysis

I – the unit matrix

J_{gz}^{-}, J_{gz}^{+} – continuous partial currents of the one–dimensional axial kinetics model

J_{gzl}^{-m}, J_{gzl}^{+m}– 1–D model average negative and positive partial currents of group g, on the left face of the node of index m

J_{gzr}^{-m}, J_{gzr}^{+m}– 1–D model average negative and positive partial currents of group g, on the right face of the node of index m

L – neutron diffusion operator $L\phi(\vec{r}, E, t) \equiv \nabla \cdot D(\vec{r}, E, t)\nabla\phi(\vec{r}, E, t)$

L_0 – L for steady–state calculation

L(Δt)

N – matrix operator used in stability analysis

N_g – envelope function or 'axial flux' of energy group g for 1–D model

N_g^m – the average of N_g for plane m of the axial grid $m \in (1, \ldots, N_z)$ in the 1–D NEM model

N – number of nodes in the 3–D NEM discretization of S

N_G – number of energy groups

N_I – number of neutron precursor groups

N_J – number of fissible isotopes in the core

N_x – maximum number of nodes in the x direction

N_y – maximum number of nodes in the y direction

N_z – number of nodes in the z direction

P – neutron amplitude function for the point kinetics equations

P^j – P at time step j

P(Δt) – matrix operator used in stability analysis

P – neutron production operator, $P\phi(\vec{r},E,t) \equiv \sum_{j=1}^{N_J} \chi^j(E) \int_0^\infty \nu\Sigma_f^j(\vec{r},E',t)\phi(\vec{r},E',t)dE'$

P_0 – **P** for steady state calculation

Q(Δt) – bounded matrix used in the stability analysis

R – matrix operator used in stability analysis

R – the space of real numbers

S – the boundary of **V**

T – total diffusion or transport operator $T=P-A-L$

T_0 – **T** for steady–state calculation

T_{min}^{1D}, T_{max}^{1D} – user–requested minimum and maximum 1–D model integration time

T_{min}^{PK}, T_{max}^{PK} – user–requested minimum and maximum PK model integration times

W – weight function

W_g – weight function for energy group g

W_i^C – weight function for the precursor concentrations in the axial 1–D model

V – the domain of the neutron diffusion equation: the core and the reflector

V_m – an axial partition of **V** consisting of all the nodes Ω in one axial plane, $m \in (1,...,N_z)$

Lowercase Greek

β – total delayed neutron precursor fraction for the point kinetics equations

β_i – delayed neutron precursor fraction for precursor group i for the point kinetics equations

β_i^j – delayed neutron precursor fraction for precursor group i and isotope j

$$\beta^j \equiv \sum_{i=1}^{N_I} \beta_i^j -$$

$\gamma_{g,ij}^{zr,n}$ – prolongation factor of the nodal face–averaged current for the node of index n, group g, on the right side of the node in the z direction, prolonged from time t_i to t_j

$\gamma_{g,ij}^{zl,n}$ – prolongation factor of the nodal face–averaged current for the node of index n, group g, on the left side of the node in the z direction, prolonged from time t_i to t_j

$\delta\rho_{\Delta\phi,\gamma,J}$ – calculated change in reactivity due to changes in flux shape, leakage, and normalization

$\delta\rho_{ppm}, \delta\rho_{ft}, \delta\rho_{mt}, \delta\rho_{md}, \delta\rho_{other}$ – calculated changes in reactivity from changes in boron ppm, fuel temperature, moderator temperature, moderator density, and other effects

$\delta\rho_{ppm\Delta\phi}, \delta\rho_{ft\Delta\phi}, \delta\rho_{mt\Delta\phi}, \delta\rho_{md\Delta\phi}, \delta\rho_{other\Delta\phi}$ – higher order calculated changes in reactivity

from the related effects and changes in flux shape, combined

ϱ – spectral radius of a matrix

ϵ_G – user–specified tolerable global error

$\epsilon_G^{LD \to 3D}$ – user–specified tolerable global error for switching from lower dimensional model to 3–D model

ϵ_L – user–specified tolerable local error

ε_g – relative error in the flux

η – the coordinate within a node parallel to the y axis, normalized by a_y^m

$\vec{\theta}_k$ – vector of phase angles of mode k for the local mode stability analysis

θ_u – the uth [with $u \in \{x, y, z\}$] component of $\vec{\theta}_k$, with the k index omitted

$\vec{\kappa}_m$ – the position vector of the node m, in terms of discrete co–ordinates:

$$\vec{\kappa}_m \equiv (i,j,k) \in (1..N_x, 1..N_y, 1..N_z)$$

λ – eigenvalue of the stationary solution of the neutron diffusion equation

λ_i – decay constant of neutron precursor group i

ν^j – number of neutrons released per fission for isotope j

ξ – the coordinate within a node parallel to the x axis, normalized by a_x^m

ξ_i^{PK} – precursor shape function of precursor group i for the point kinetics equations

ξ_i^{1D} – precursor shape function of precursor group i for the axial kinetics equations

$\tilde{\xi}_{ki}^n$ – amplitude of the kth mode of the ith precursor group at time step n for the local mode stability analysis

ρ – reactivity, ρ_0 – reactivity from steady–state calculation

ρ_{max} – maximum absolute reactivity at which lower dimensional models can be active

ς – the coordinate within a node parallel to the z axis, normalized by a_z^m

ϕ – neutron flux

ϕ_g – neutron flux of energy group g

ϕ_g^m – nodal averaged flux in node m for energy group g

φ_g^m – polynomial expansion of the 3–D flux for energy group g and the node of index m, in the local normalized coordinates of the node.

χ_{pg}^j – fraction of neutrons born through fission in group g from fissionable isotope j

χ_{dg}^i – fraction of neutrons born in fission group g from precursor group i

$\tilde{\chi}_{kgLu}$ – amplitude of the kth mode of the gth group of the partial currents on the left side of each node in the direction u

$\tilde{\chi}_{kgRu}$ – amplitude of the kth mode of the gth group of the partial currents on the right side

of each node in the direction u

ψ – flux shape function

ψ_0 – flux shape function from steady–state calculation

ψ_g^{PK} – flux shape function of group g for the point kinetics model

ψ_g^{1D} – flux shape function of group g for the axial kinetics model

$\bar{\psi}_{kg}^n$ – amplitude of the kth mode of the gth group of the flux at time step n for the local mode stability analysis

ω^m – dynamic frequency found from the exponential transform method

Uppercase Greek

Γ – surface integral for perturbation theory calculations of the reactivity

$\Gamma_{g,ij}^n$ – prolongation factor of the nodal averaged flux for the node of index n, group g, prolonged from time t_i to t_j

$\Gamma_{g,ij}^{zr,n}$ – prolongation factor of the nodal face–averaged flux for the node of index n, group g, on the right side of the node in the z direction, prolonged from time t_i to t_j

$\Gamma_{g,ij}^{zl,n}$ – prolongation factor of the nodal face–averaged flux for the node of index n, group g, on the left side of the node in the z direction, prolonged from time t_i to t_j

Δt – time step size, $\Delta t = t - t_0$

$\Delta A, \Delta L, \Delta P, \Delta T$ – changes in A, L, P, T since the steady–state calculation

$\Delta A_{ppm}, \Delta L_{ppm}, \Delta P_{ppm}$ – the respective contributions to $\Delta A, \Delta L, \Delta P$ from changes in boron concentration

$\Delta A_{ft}, \Delta L_{ft}, \Delta P_{ft}$ – the respective contributions to $\Delta A, \Delta L, \Delta P$ from changes in fuel temperature

$\Delta A_{mt}, \Delta L_{mt}, \Delta P_{mt}$ – the respective contributions to $\Delta A, \Delta L, \Delta P$ from changes in moderator temperature

$\Delta A_{md}, \Delta L_{md}, \Delta P_{md}$ – the respective contributions to $\Delta A, \Delta L, \Delta P$ from changes in moderator density

$\Delta A_{other}, \Delta L_{other}, \Delta P_{other}$ – the respective contributions to $\Delta A, \Delta L, \Delta P$ from other changes

$\Delta \rho_{\Delta \phi}$ – first order change in reactivity due to change in flux shape

$\Delta \rho_{\Delta T}$ – first order change in reactivity due to change in the operator T

$\Delta \rho_{\Delta T \Delta \phi}$ – higher order changes in reactivity due to changes in both the flux shape and T

$\Delta \rho_\gamma$ – change in reactivity due to a change of the normalization factor

$\Delta \rho^+$ – maximum change in reactivity when reactivity is positive before reactivation of the

3-D model

$\Delta\rho^-$ – maximum change in reactivity when reactivity is negative before reactivation of the 3-D model

Λ – neutron lifetime

$\Pi^n_{i,pq}$ – prolongation factor for the node averaged precursor concentration of node index n, precursor group i, from time t_p to t_q

Σ^m_{rg} – homogenized macroscopic removal cross section of node m and energy group g

$\Sigma^m_{gg'}$ – homogenized macroscopic scattering cross section of node m, for neutrons scattering from energy group g' to g

$\Sigma^{j,m}_{fg'}$ – homogenized macroscopic fission cross section of isotope j, node m, and for neutrons of energy group g',

$\overline{\Sigma}$ – cross sections of the one dimensional model, see definitions (5.17)

Θ^m_{ul} – the face of node Ω^m perpindicular to $u \in \{\xi, \eta, \varepsilon\}$, on the left side (u=0) of the node

Θ^m_{ur} – the face of node Ω^m perpindicular to $u \in \{\xi, \eta, \varepsilon\}$, on the right side (u=1) of the node

Φ^m_g – polynomial expansion of the 3-D flux for energy group g and the node of index m in the coordinates of V

Ψ_{gu} – transverse integrated 1-D nodal expansion function of the neutron flux, for energy group g, and direction $u \in \{x,y,z,\xi,\eta,\varsigma\}$

Ψ^m_g – 1-D nodal expansion function for the neutron flux of the axial kinetics model, for group g and axial level m

Ψ^m_{gl}, Ψ^m_{gr} – values of Ψ^m_g at the left ($u=0$) and right ($u=1$) hand sides of the axial level of index m

$\Omega = (\Omega^1, ..., \Omega^N), \{\Omega^m \subset S, m \in I, 1 \leq m \leq N\}$ the set of N nodes into which S is partitioned.

Ω^m – the node of index m in the set Ω

Ω^m_{ui} – the member of Ω which neighbours Ω^m on side ui

Other Definitions

$\left(\frac{\partial\epsilon}{\partial t}\right)^{1D}_G, \left(\frac{\partial\epsilon}{\partial t}\right)^{1D}_L$ – estimated rates of global and local error accumulation for the 1-D model

$\left(\frac{\partial\epsilon}{\partial t}\right)^{PK}_G, \left(\frac{\partial\epsilon}{\partial t}\right)^{PK}_G$ – estimated rates of global and local error accumulation for the PK model

Additional Superscripts

$*$ – adjoint

h – an approximation to the function without the h subscript

$3D$ – quantity from the reference 3D calculation

AML – quantity from a calculation with the adaptive algorithm

1. Introduction

A major task in the safety analysis of nuclear power plants is the simulation of postulated accident scenarios. These scenarios hypothesize undesired disturbances or failures in thermalhydraulic systems, control systems, plant machinery, or the reactor core. It is the task of the safety analyst to assess the consequences of these initially local disturbances on the plant as a whole. Among the most serious of accident scenarios are those which cause the reactivity of the core to increase. If the reactor becomes prompt supercritical, then the resulting rapid power excursion may ultimately cause fuel rods, and in the worst case, the reactor containment, to fail. A number of postulated accident scenarios which affect the core can be initiated in some other part of the power plant. The computer simulation and analysis of these accidents for light water reactors became possible with transient codes like RELAP,[1] TRAC,[2] and CATHARE.[3] The predecessors of the current versions of these codes were initially limited by the computer resources, and to a lesser extent by the available numerical methods, of the 1970s. For these reasons, the neutron kinetics models of these codes have remained simplified point or one–dimensional neutron kinetics models until only recently. Inclusion of three–dimensional neutron kinetics models into such codes significantly extends their range of applicability, and can greatly improve their accuracy for postulated accident scenarios in which the power shape of the core varies significantly in time. Several codes capable of performing coupled three–dimensional neutron kinetics and plant thermalhydraulics now exist.[4,5,6,7]

Despite the current state of the art in both numerical methods and computer hardware, the computational overhead of using a three–dimensional neutron kinetics model may still prohibit its utilization. This is especially true if a code user deems that a point kinetics or one–dimensional model is accurate enough for the application at hand, or if many calculations must be performed for simulation times on the order of hundreds or thousands of seconds. Indeed, it is recognized that during many postulated transients, there are large periods of simulation time in which a three–dimensional neutron kinetics model is *not* necessary. During these periods, use of a point kinetics or one–dimensional kinetics model would be sufficient. An efficient code would only activate the three–dimensional model when it was necessary, and would use a lower dimensional model during other periods of the transient. The goal of this dissertation is to develop an algorithm which automatically and adaptively switches between three–dimensional, one–dimensional, and point neutron kinetics models.

Published methods for finding efficient approximations to neutron kinetics problems began in the late 1950s. In their classic textbook, Weinberg and Wigner[8] present a derivation of the reactor kinetics equations. They begin with the time dependent neutron diffusion equation

1

for a bare homogeneous reactor. Because of this simplified geometry, they were able to write the flux solution as a superposition of orthogonal spatial modes. The time dependence of these modes form the reactor kinetics equations, which are in structure similar to the point kinetics equations. In the same year, Henry[9] showed how the point kinetics equations could be derived directly from the time dependent transport equation, and presented a method for approximating the point kinetics coefficients from the stationary flux solution of a heterogeneous core. Henry's method is based on the idea of separating the flux into a shape function and an amplitude function. The shape function is then approximated to be time independent. This procedure is refined and repeated by Henry in later works,[10,11] and has come to be known as the *adiabatic quasi–static approximation.*

Computer hardware of the 1960's enabled the use of finite difference schemes to solve the space dependent neutron diffusion equation. However, with the technology available, only solutions of one–dimensional problems were at this time tractable, and time dependent problems in three dimensions remained far out of range for efficient simulation. To overcome this difficulty, flux synthesis methods were introduced by Kaplan et. al.[12] and Yasinsky.[13] These methods synthesized asymptotic flux shapes with time–dependent amplitude functions. The time–dependent amplitude functions were generally found using either weighted residual methods or a variational principle. In the late 1960's, Ott[14] and Ott and Meneley[15] significantly extended Henry's adiabatic quasi–static approximation method by recalculating the flux shape using either the *quasi–static*[14] or the *improved quasi–static*[15] approximation. In each of these cases, an equation for the shape function is derived from the time dependent neutron diffusion equation: in the quasi–static case, the time derivative of the shape function is neglected; in case of the improved quasi–static method, the time derivative is considered, but the equation is integrated implicitly over large time steps. Quasi–static and flux synthesis methods were presented in a unified form in the note of Kessler[16] which also seems to be one of the earliest works analyzing space dependent reactor kinetics coupled with a space dependent fuel temperature model.

In 1970, Reed and Hansen[17] applied alternating direction finite difference methods to solve time dependent neutron kinetics problems in two dimensions. Although their computing resources were still too inadequate to treat three–dimensional problems of large heterogeneous cores, they did develop the exponential transform method, which is examined in chapter 3 of this thesis. In 1973, Ferguson and Hansen[18] extended this method to three dimensions, and used a semi–implicit time integration scheme. With the computer technology of the time, this method was still too inefficient to be applied to production engineering applications. In the same year, Kang and Hansen[19] presented the first application of finite element methods to the time dependent neutron diffusion equation. With this method, they were able to direct-

ly calculate transients in two spatial dimensions. In 1976, Buckner and Stewart[20] presented another finite difference scheme for three dimensions: their work concentrated on the iterative method which was used to solve the finite difference equations.

Despite the advances being made in computer technology, it was apparent in the 1970s that finite difference and even finite element methods were too expensive to be used for the accurate analysis of transients in large reactor cores. For this reason, considerable effort was being made to develop accurate coarse mesh or so called 'nodal' methods. The origin and evolution of these methods is fully described in the detailed review article of Dorning.[21] The first coarse–mesh method efficient and accurate enough for general applications was the nodal expansion method (NEM) presented in 1975 by Finnemann.[22] Further extensions to the NEM were presented in subsequent years by Finnemann et. al.[23,24,25] Ideas from the NEM were then later adapted by Lawrence and Dorning[26,27,28] in their development of the nodal Green's function method. Both the NEM and the Green's function method are to this day considered state–of–the–art. They form the basis for dozens of other 'nodal methods' found in the literature.

From the beginning of the 1980s, the development of more efficient methods to solve reactor kinetics equations in three dimensions became less focussed. Coarse mesh finite difference methods (CMFD) were developed at the MIT,[29,30,31,32] but their range of accurate application remains questionable. Other approaches concentrate on simplifying the two–group diffusion equations through various approximations.[33,34] Further developments of the quasi–static approximation have been more of a theoretical nature, with no identifiable advantages in efficiency.[35,36,37] (Integration of the improved quasi–static method into modern nodal codes does of course improve their efficiency.[38]) Coarse mesh rebalancing[39] and other multi–level methods[40] have been extensively applied to accelerate the convergence of the nodal methods.[41] Time discretization procedures have also been improved: the stiffness confinement method, developed by Chao and Risher[42] is an extension of the exponential transform method developed earlier by Reed and Hansen.[17] Crouzet and Turinsky[43] have also recently developed an adaptive time–step method which selects optimal time step sizes based on the temporal truncation error of the implicit scheme. The parallelization of neutron kinetics codes has also demonstrated significant wall–clock speedups provided that several computer processors are available for a calculation.[44,45,46] Finally, a totally different approach has been taken by Favorite and Stacey[47] who have effectively developed second–order perturbation theory expressions for the coefficients of the point kinetics equations. Their method is reasonably accurate and efficient provided that a pre–calculated sensitivity library has been generated and stored.

Critical examination of these recent advances in the efficiency of neutron kinetics codes shows that a reduction of 50% in the required computing time is a good achievement. Combination of the efficient numerical methods, such as nodal methods, the improved quasi–static approximation, multi–level coarse mesh rebalancing, adaptive time step selection, and code parallelization, can indeed result in an efficient code. Such an approach is highly advantageous when compared to cruder approximation techniques such as the simplification of the two–group diffusion equations. These simplifications are only useful if the code knows when they are valid. By applying an adaptive approach to the adiabatic quasi–static approximation initially presented by Henry,[9,10,11] this thesis converts an outdated and not always applicable method into a useful tool for reactor safety analysis.

As a starting point, the core simulation code PANBOX has been coupled to the best–estimate plant transient code RELAP5. The current version of the PANBOX code system[48] (PANBOX 2) is capable of calculating three–dimensional neutron kinetics transients using various coarse mesh nodal methods, such as the polynomial[23] and analytical[49] nodal expansion methods. In–core thermalhydraulic conditions may be calculated by PANBOX using an internal module based on COBRA 3–PC.[50] Typically, one one–dimensional thermalhydraulic channel is defined per fuel assembly, although coarser channels may be used, and subchannel analysis may be performed for specified assemblies. The one–dimensional thermalhydraulic solution is augmented by a crossflow model between channels. The features of PANBOX make it an appropriate state of the art code system for the simulation and safety analysis of pressurized light water reactor cores. RELAP5 is a code for the analysis of the thermal–hydraulic behaviour of light water systems. Originally designed for the analysis of loss–of–coolant accidents (LOCAs) in pressurized water reactors, the range of validity of the code has over many years been expanded to cover a wide range of postulated accident scenarios. RELAP5 models two–phase flow using a nonequilibrium, nonhomogeneous, six–equation model. Boron concentration and non–condensible gases may also be simulated with a separate equation for each material. RELAP5 also has the ability to simulate heat transfer to and from materials adjacent to the fluid. A one–dimensional temperature distribution in these materials is calculated by solution of the Fourier heat conduction equation. Equipment controllers, balance–of–plant equipment (e.g., pumps and turbines), and lumped–parameter representations of other processes may also be crudely simulated with the code. The neutron kinetics model in the currently released version of RELAP5 is the point kinetics model. The point kinetics coefficients can be made dependent on the thermalhydraulic state of the core, thus permitting simulation of feedback between thermalhydraulic and neutron kinetic behaviour. It is this simple point kinetics model in RELAP5 which is replaced by the three–dimensional multigroup model of PANBOX.

Chapter 2 of this thesis describes the interface which has been developed between RELAP5 and PANBOX. Three different coupling options are possible, depending on how the user wishes to use the COBRA modules internal to PANBOX. The usefulness of a three–dimensional neutron kinetics capability is then demonstrated by the calculation of a boron dilution transient. Through a reactivity edit option, developed here for RELAP/PANBOX to aid in the explanation of transient phenomena, it is shown that changes in the flux shape can have a major effect on the evolution of a transient. Indeed, it is when the flux shape is changing that a spatially dependent model is needed. When the flux shape is not changing, however, then the point kinetics model is sufficient. This observation motivates the need for a dimensionally adaptive algorithm.

The three–dimensional neutron kinetics model of PANBOX uses the NEM discretization of the multigroup diffusion equation. In later chapters, it will be seen that coefficients for the point and one–dimensional models are ultimately dependent on the three–dimensional flux solution found with the NEM. For the point kinetics model, perturbation theory expressions are derived from the nodal flux expansions of the NEM. The one–dimensional model is also discretized with the NEM. These observations show that the computed solutions of the NEM are essential to the adaptive algorithm. Therefore, the consistency and stability[51,52] of the NEM are addressed in Chapter 3.

The point kinetics model for the algorithm is developed in Chapter 4. Here, the point kinetics equations are derived in the classical manner from the three–dimensional multigroup neutron diffusion equation, as first done by Henry.[9] Implementation of this point kinetics model directly in PANBOX allows the point kinetics coefficients to be calculated from the three–dimensional flux solution and the three–dimensional neutron cross section data base. The question of how perturbation theory can be used to calculate the reactivity is addressed, and a method is presented for how the operator formulation, originally identified by Cacuci et. al.,[53,54] of perturbation theory may be implemented with solutions of the NEM. This contrasts the methods proposed by previous authors,[55,56,57,58,59,60] and rigorously examines the method sketched by Delmolino.[61]

Chapter 5 presents the one–dimensional axial kinetics model which has been derived for this dissertation. As per the point kinetics equations, the one–dimensional model is derived directly from the three–dimensional multigroup neutron diffusion equation. The NEM discretization is applied to the continuous representation of the one–dimensional model, and correction factors are defined to force equivalence between the three– and one–dimensional solutions. These correction factors are compared to the heterogeneity or 'discontinuity' factors developed for homogenization procedures.[62,63,64] In contrast to previously existing

one– dimensional diffusion models in the literature,[65,66,67,68,69] this is the only known model derived directly from the three–dimensional flux solution of a heterogeneous core.

The multi–level algorithm is developed and described in Chapter 6. The mechanics of switching from one model to another are detailed, and prolongation operators[40] are defined for approximating the three–dimensional flux. Criteria for switching from three–dimensional to one–dimensional and one–dimensional to point kinetics models are derived. An error estimation procedure based on the work of Ainsworth and Oden[70,71,72] is developed to determine criteria for when the three–dimensional model should be reactivated. Some reactivity–based switching criteria are also presented to complement the performance of the error estimator.

In Chapter 7, some sample problems are calculated. Through the calculation of control rod ejection, main steam line break, and boron dilution transients, it is shown that the adaptive algorithm can save 30%–70% of CPU time, while preserving much of the accuracy of a fully three–dimensional reference calculation. The accuracy is acceptable if the user can tolerate small shifts in time of the calculated results. Conclusions and future work are presented in Chapter 8.

2. Coupled Thermalhydraulics/Neutron Kinetics Calculations

2.1 Description of RELAP5/PANBOX

The first development version of the coupled RELAP5/PANBOX system was described by Knoll and Müller.[73] Further development in the coupling was necessary to bring the code system to a stage where it was user–friendly and applicable to a wider range of problems. Herein is briefly described the features of the version developed for use with this thesis. The coupling of the two codes is achieved via the interface EUMOD[74] (External User MODels), developed previously at Siemens AG. EUMOD is a set of subroutines which enables the user to link external codes to RELAP5. Under this system, PANBOX becomes a subroutine of RELAP5 which is called at the end of every RELAP5 time step. The flow logic is depicted in Figure 2.1.

Figure 2.1: Illustration of available options of the coupled RELAP5/PANBOX system.

The coupling of external codes to RELAP5 via EUMOD is restricted in that the codes may be coupled only explicitly, i.e. external routines can be called only at the end of every RELAP5 time step, and no iteration is performed between the RELAP5 solution and any itera-

tive solution of the external user model. This coupling procedure is assumed to be accurate and stable for the selected time step sizes. The RELAP5 time step size is chosen as the smallest requested by either PANBOX or by the stability criteria internal to RELAP5. The PANBOX adaptive time step criteria is based on changes in the fast flux and the changes in fuel temperature. Additional stability–based time step restrictions for the NEM are derived in chapter 3 of this thesis. Thus, both neutronic and thermalhydraulic behaviour are considered for the selection of time step size, although the interplay between the phenomena is not considered. For the time step sizes of interest, no numerical instability problems have been detected with this explicit coupling.

In the first development version of RELAP5/PANBOX, only the core averaged thermalhydraulic boundary conditions of core pressure, inlet temperature, and inlet mass flow rate were passed from RELAP5 to PANBOX. With these boundary conditions, PANBOX calculated core thermalhydraulics using its internal COBRA–based modules. This option has been extended to pass the boundary conditions of more than one RELAP5 core channel to PANBOX. The boundary conditions are transferred to the appropriate COBRA channels by a mapping procedure, as illustrated in Figure 2.2 for a RELAP5 nodalization that models the core using four channels. The power distribution calculated in PANBOX is collapsed back onto the RE-

Figure 2.2: Depiction of the first coupling option for RELAP/PANBOX: cross section update from COBRA.

LAP5 nodalization with the same mapping procedure. This option has the advantage that feedback effects are simulated using a thermalhydraulic model which is very detailed spatially. Additionally, thermal margins such as DNB ratios may be calculated by the COBRA code. The problem with this coupling procedure is that the COBRA solution algorithm converges

quite slowly, leading to long computing times, and often diverges in low flow or low pressure conditions. This rendered the RELAP5/PANBOX system unable to calculate transients which began in, or evolved into these core conditions, such as after a small break LOCA event.

To provide the user with a more flexible system, two more options were implemented in the interface routines of the codes. Option 2 utilizes no COBRA calculation at all. Rather, the thermalhydraulic and fuel temperature data from RELAP5 are used to update the neutronic cross sections. This option is depicted in Figure 2.3 for a four channel RELAP5 nodalization of the core. Because no COBRA solution is calculated, CPU demands for this option are sig-

Figure 2.3: Depiction of the second coupling option for RELAP/PANBOX: no COBRA calculation, cross section update from RELAP data.

nificantly less than for the other two options. The drawbacks of this method are that the feedback effects are simulated using only a very coarse thermalhydraulic mesh, and RELAP5 has no inherent routines for calculating thermal margins.

Option 3, depicted in Figure 2.4, also uses RELAP5 data to update the PANBOX cross sections; however, the core boundary conditions and PANBOX power distributions are used in a 'parallel' COBRA calculation. This COBRA calculation has no influence on the neutron kinetic or RELAP5 calculations, but is useful for computing thermal margins needed for reactor licensing. In future, this option can be extended so that the COBRA calculation is only activated only during those periods of the transient when safety margins need to be calculated.

The channel to fuel assembly mappings for the coupled calculations are input by the user. With these mappings, the three dimensional power profiles from PANBOX are collapsed onto the much coarser nodalization in RELAP5. The same mapping is used to transfer RE-

9

Figure 2.4: Depiction of coupling option 3 of RELAP/PANBOX: cross section update from RELAP, parallel COBRA calculation for thermal margins.

LAP5 thermalhydraulic data to the neutronic nodes, and thermalhydraulic boundary conditions to the COBRA channels. In addition to the radial mapping of channels, appropriate axial interpolation of data is performed automatically when the axial mesh sizes of the RELAP5 and PANBOX nodalizations differ.

Not shown in figures 2.2–2.4 is the transfer of boron concentrations from RELAP5 to PANBOX. This is performed in a manner similar to the transfer of fuel temperature, with the added consideration that the total amount of boron in the core is conserved in the interpolation procedure.

The new coupling options 2 and 3 involve a significant amount of data exchange between the two codes. The verification and validation of this data exchange was performed by using RELAP5/PANBOX to calculate cases A1 and A2 of the NEACRP control rod ejection benchmark problem.[75] The results of these calculations were presented in reference 76, and were found to agree quite well with the reference solution[77] and the solution calculated with the coupled RELAP/NESTLE code.[7]

2.2 Proof of the Usefulness of RELAP5/PANBOX: a sample calculation

In this section, it will be shown why a 3–D neutron kinetics model is necessary for the calculation of some transients. To do this, a sample calculation of a boron dilution transient has been

selected; and a method for quantitatively explaining the phenomena of the 3–D neutron kinetics solution is also presented.

In the simple point kinetics approximation of the standalone RELAP5 code, coupling between the core and the thermalhydraulic system is achieved by the input of fuel temperature, moderator density and boron concentration reactivity coefficients. In contrast, with the RELAP5/PANBOX code, coupling between the core and the thermalhydraulic system is accomplished via the dependence of the macroscopic cross sections on local thermalhydraulic conditions and boron concentrations. The point kinetics model produces results which are intuitively understandable: the contributions to the reactivity from fuel temperature, coolant density, and boron density are readily obtained and interpreted. In contrast, when the reactor kinetics are calculated using a three–dimensional multigroup neutron diffusion model, these reactivity contributions are no longer apparent . Thus, it was found that the incorporation of a three–dimensional kinetics model into RELAP5 created a new challenge for the analyst to explain transient phenomena, despite the fact that the transient could be calculated with greater accuracy.

To help explain transient phenomena, a 'reactivity edit option' was developed for PANBOX. The option uses the neutron flux distribution at each time point to calculate the total core reactivity, as well as the contributions to this reactivity from changes in the core state. These changes include changes in fuel temperature, changes in moderator density, and changes in neutron flux distribution. In section 2.2.1, these contributions are derived from the general definition of reactivity. In section 2.2.2, it is described how these contributions are calculated in PANBOX. Finally, the method is demonstrated in section 2.2.3 with results from the calculation of a boron dilution transient.

2.2.1 Definition of Reactivity and Reactivity Contributions

The general definition of reactivity is[10]

$$\rho(t) = \frac{< W(\vec{r},E), T(\vec{r},E,t)\psi(\vec{r},E,t) >}{< W(\vec{r},E), P(\vec{r},E,t)\psi(\vec{r},E,t) >} \tag{2.1}$$

where P is the fission source operator, and T is the total diffusion– or transport– theory operator,

$$T(\vec{r},E,t) \equiv P(\vec{r},E,t) - A(\vec{r},E,t) - L(\vec{r},E,t)$$

The operator A contains absorption and scattering terms and the operator L accounts for leakage effects. W is some weighting function (to be chosen later), ψ is the neutron flux shape, and $< \cdot, \cdot >$ denotes integration over all space and energy. The neutron flux shape is normalized from the neutron flux through

$$\psi(\vec{r},E,t) = \frac{\phi(\vec{r},E,t)}{\langle W(\vec{r},E), \phi(\vec{r},E,t)\rangle}$$

For an initial flux distribution ϕ_0, which satisfies the eigenvalue problem

$$\left[(1 - \rho_0)P_0(\vec{r},E) - A_0(\vec{r},E) - L_0(\vec{r},E)\right]\phi_0(\vec{r},E) = 0 \tag{2.2}$$

and the usual specified boundary conditions, equation (2.1) takes the form

$$\rho_0 = \frac{< W(\vec{r},E), T_0(\vec{r},E)\psi_0(\vec{r},E) >}{< W(\vec{r},E), P_0(\vec{r},E)\psi_0(\vec{r},E) >} \tag{2.3}$$

which is stationary for all non–trivial choices of W. During a transient calculation, the operators P, A, and L all change due to changes in neutron cross sections. These changes can be expressed as contributions from various sources. For example,

$$\Delta P(\vec{r},E,t) \equiv P(\vec{r},E,t) - P_0(\vec{r},E) = \Delta P_{ppm} + \Delta P_{ft} + \Delta P_{mt} + \Delta P_{md} + \Delta P_{other}$$

$$\Delta A(\vec{r},E,t) \equiv A(\vec{r},E,t) - A_0(\vec{r},E) = \Delta A_{ppm} + \Delta A_{ft} + \Delta A_{mt} + \Delta A_{md} + \Delta A_{other} \tag{2.4}$$

$$\Delta L(\vec{r},E,t) \equiv L(\vec{r},E,t) - L_0(\vec{r},E) = \Delta L_{ppm} + \Delta L_{ft} + \Delta L_{mt} + \Delta L_{md} + \Delta L_{other}$$

where the subscripts stand for changes in boron ppm, changes in fuel temperature, changes in moderator temperature, changes in moderator density, and other changes, respectively. It is assumed here that these different contributions to the cross sections are separable effects, which is the approximation made in most three–dimensional neutron kinetics codes. For example, if cross sections are determined by partial derivatives with respect to the various effects, then definitions (2.4) are valid. The goal of this section is to identify a quantitative measure of how these operator changes contribute to changes in reactivity.

The total change in reactivity from the initial condition is equation (2.3) subtracted from equation (2.1):

$$\Delta\rho(t) \equiv \rho(t) - \rho_0 = \frac{< W(\vec{r},E), T(\vec{r},E,t)\psi(\vec{r},E,t) >}{< W(\vec{r},E), P(\vec{r},E,t)\psi(\vec{r},E,t) >} - \frac{< W(\vec{r},E), T_0(\vec{r},E)\psi_0(\vec{r},E) >}{< W(\vec{r},E), P_0(\vec{r},E)\psi_0(\vec{r},E) >}$$

The expression on the right hand side is conveniently divided into four main contributions

$$\Delta\rho(t) = \Delta\rho_{\Delta\phi}(t) + \Delta\rho_{\Delta T}(t) + \Delta\rho_{\Delta T\Delta\phi} + \Delta\rho_\gamma(t)$$

These are defined as

$$\Delta\rho_{\Delta\phi}(t) \equiv \frac{< W(\vec{r},E), T_0(\vec{r},E)\psi(\vec{r},E,t) >}{< W(\vec{r},E), P_0(\vec{r},E)\psi(\vec{r},E,t) >} - \frac{< W(\vec{r},E), T_0(\vec{r},E)\psi_0(\vec{r},E) >}{< W(\vec{r},E), P_0(\vec{r},E)\psi_0(\vec{r},E) >} \tag{2.5}$$

$$\Delta\rho_{\Delta T}(t) \equiv \frac{< W(\vec{r},E), \Delta T(\vec{r},E,t)\psi_0(\vec{r},E) >}{< W(\vec{r},E), P(\vec{r},E,t)\psi(\vec{r},E,t) >} \tag{2.6}$$

$$\Delta\rho_{\Delta T\Delta\phi}(t) \equiv \frac{<W(\vec{r},E),\Delta T(\vec{r},E,t)\Delta\psi(\vec{r},E,t)>}{<W(\vec{r},E),P(\vec{r},E,t)\psi(\vec{r},E,t)>} \qquad (2.7)$$

$$\Delta\rho_\gamma(t) \equiv \gamma\frac{<W(\vec{r},E),T_0(\vec{r},E)\psi(\vec{r},E,t)>}{<W(\vec{r},E),P_0(\vec{r},E)\psi(\vec{r},E,t)>} \qquad (2.8)$$

where

$$1 + \gamma \equiv \frac{1}{1 + \dfrac{<W(\vec{r},E),\Delta P(\vec{r},E,t)\psi(\vec{r},E,t)>}{<W(\vec{r},E),P_0(\vec{r},E,t)\psi_0(\vec{r},E)>}} \qquad (2.9)$$

The meaning of the first three terms is clear: (2.5) is the reactivity change due only to changes in the neutron flux shape, (2.6) is the reactivity change due to direct changes in cross sections, and (2.7) are the combined changes due to changes in cross section and flux shape. The meaning of the $\Delta\rho_\gamma$ term becomes clear once a suitable weight function W is chosen.

The goal in choosing the weight function is to make the different reactivity components (2.5) to (2.8), at least as strongly dependent on the changes in operators as on the changes in flux. It is seen directly from (2.6) and (2.7) that $\Delta\rho_{\Delta T}$ and $\Delta\rho_{\Delta T\Delta\phi}$ already have this strong dependence on ΔT. Expanding (2.9) in a series, the leading terms of γ are

$$\gamma = -\frac{<W(\vec{r},E),\Delta P(\vec{r},E,t)\psi(\vec{r},E,t)>}{<W(\vec{r},E),P_0(\vec{r},E)\psi_0(\vec{r},E)>} + \frac{<W(\vec{r},E),\Delta P(\vec{r},E,t)\psi(\vec{r},E,t)>^2}{<W(\vec{r},E),P_0(\vec{r},E)\psi_0(\vec{r},E)>^2} - \cdots$$

which shows that $\Delta\rho_\gamma$ is also dependent on ΔP to first order. Therefore, only (2.5) does not have this first order dependence on the change in the operator. The natural choice for the weight function is therefore the solution to the adjoint of equation (2.2):

$$\left[(1 - \rho_0)P_0^*(\vec{r},E) - A_0^*(\vec{r},E) - L_0^*(\vec{r},E)\right]\phi^*(\vec{r},E) = 0$$

(Appropriate boundary conditions must also be chosen for the adjoint problem.) When this weight function is used, equation (2.5) may be expressed as

$$\begin{aligned}
\Delta\rho_{\Delta\phi}(t) &= \frac{<T_0^*(\vec{r},E)\phi^*(\vec{r},E),\psi(\vec{r},E,t)>}{<P_0^*(\vec{r},E)\phi^*(\vec{r},E),\psi(\vec{r},E,t)>} - \frac{<\phi^*(\vec{r},E),T_0(\vec{r},E)\psi_0(\vec{r},E)>}{<\phi^*(\vec{r},E),P_0(\vec{r},E)\psi_0(\vec{r},E)>} \\
&= \rho_0 - \rho_0 \\
&= 0
\end{aligned}$$

for all nontrivial ϕ^* and ψ_0. Thus, the use of the adjoint function as a weight function eliminates the first order reactivity contribution due to flux change. This is indeed the usual argument for using the adjoint function as a weight function for perturbation theory calculations of the reactivity. The main difference between this method and perturbation theory lies in its intended use. The contributions calculated here serve to explain the physical phenomena of the transient; they do not serve to estimate the reactivity, which is known. The use of the

adjoint weight function additionally casts light on the meaning of $\Delta \varrho_\gamma$, which now may be written as

$$\Delta \rho_\gamma(t) = \gamma \rho_0$$
$$= -\rho_0 \frac{< W(\vec{r},E), \Delta P(\vec{r},E,t) \psi(\vec{r},E,t) >}{< W(\vec{r},E), P_0(\vec{r},E) \psi_0(\vec{r},E) >} + \mathcal{O}(\rho_0 \Delta P^2)$$

That is, $\Delta \varrho_\gamma$ may be interpreted as a shift of the initial core reactivity ϱ_0 due to a change in the production operator. When the initial core reactivity is zero, $\Delta \varrho_\gamma$ is also zero.

The two main reactivity contributions $\Delta \varrho_{\Delta T}$ and $\Delta \varrho_{\Delta T \Delta \phi}$ may also be split up into contributions from the different core physics phenomena. Using (2.4) in (2.6), for example, the following different contributions are defined:

$$\Delta \rho_{ppm}(t) \equiv \frac{< W(\vec{r},E), \Delta T_{ppm}(\vec{r},E,t) \psi_0(\vec{r},E) >}{< W(\vec{r},E), P(\vec{r},E,t) \psi(\vec{r},E,t) >}$$

$$\Delta \rho_{fl}(t) \equiv \frac{< W(\vec{r},E), \Delta T_{fl}(\vec{r},E,t) \psi_0(\vec{r},E) >}{< W(\vec{r},E), P(\vec{r},E,t) \psi(\vec{r},E,t) >}$$

$$\Delta \rho_{mt}(t) \equiv \frac{< W(\vec{r},E), \Delta T_{mt}(\vec{r},E,t) \psi_0(\vec{r},E) >}{< W(\vec{r},E), P(\vec{r},E,t) \psi(\vec{r},E,t) >}$$

$$\Delta \rho_{md}(t) \equiv \frac{< W(\vec{r},E), \Delta T_{md}(\vec{r},E,t) \psi_0(\vec{r},E) >}{< W(\vec{r},E), P(\vec{r},E,t) \psi(\vec{r},E,t) >}$$

$$\Delta \rho_{other}(t) \equiv \frac{< W(\vec{r},E), \Delta T_{other}(\vec{r},E,t) \psi_0(\vec{r},E) >}{< W(\vec{r},E), P(\vec{r},E,t) \psi(\vec{r},E,t) >}$$

such that

$$\Delta \rho_{\Delta T}(t) = \Delta \rho_{ppm}(t) + \Delta \rho_{fl}(t) + \Delta \rho_{mt}(t) + \Delta \rho_{md}(t) + \Delta \rho_{other}(t)$$

Similarly, using these contributions in (2.7) the second–order terms are defined as

$$\Delta \rho_{effect \Delta \phi}(t) \equiv \frac{< W(\vec{r},E), \Delta T_{effect}(\vec{r},E,t) \Delta \psi(\vec{r},E,t) >}{< W(\vec{r},E), P(\vec{r},E,t) \psi(\vec{r},E,t) >} \quad , \quad effect \in \begin{Bmatrix} ppm \\ fl \\ mt \\ md \\ other \end{Bmatrix}$$

such that

$$\Delta \rho_{\Delta T \Delta \phi}(t) = \Delta \rho_{ppm \Delta \phi}(t) + \Delta \rho_{fl \Delta \phi}(t) + \Delta \rho_{mt \Delta \phi}(t) + \Delta \rho_{md \Delta \phi}(t) + \Delta \rho_{other \Delta \phi}(t)$$

The reactivity shift term $\Delta \varrho_\gamma$ is not separable because of its nonlinear terms in ΔP. However, if the fission cross sections do not change much during a transient, this term may be negligible compared to the others.

2.2.2 Implementation in PANBOX

In PANBOX, equation (2.2) is solved with the nodal expansion method (NEM).[23,24,41] Under the NEM discretization, auxiliary variables are defined for the neutron leakage $L\phi$. Equation (2.2) is then discretized as the 'nodal balance equation' or 'zeroth moment equation' which has the form

$$\left[(1 - \rho_0)P_0^m - A_0^m\right]\vec{\phi}_0^m - \vec{J}_{net,0}^m = 0 \tag{2.10}$$

Here, the subscript m denotes the node number, and the vector notation denotes a vector of multigroup fluxes and net leakages. $\vec{J}_{net,0}^m$ is determined with the NEM outgoing current and auxiliary moment equations, as described in references 23, 24, and 41. What is important for the purposes of the implementation of the reactivity edit option, is that under the NEM discretization, the L operator does not appear explicitly in equation (2.10). While the auxiliary NEM equations could be involved in this analysis – in order to isolate the reactivity contributions from changes in the L operator – it has instead been chosen to lump these different effects together. Thus, the following reactivity contributions have been defined for the implementation in PANBOX

$$\delta\rho_{\Delta\phi,\gamma,J}(t) \equiv \frac{< W, [P_0 - A_0]\phi(t) - J_{out}(t) >}{< W, P(t)\phi(t) >} - \frac{< W, [P_0 - A_0]\phi_0 - J_{out,0} >}{< W, P_0\phi_0 >}$$

$$= \Delta\rho_{\Delta\phi} + \Delta\rho_\gamma - \frac{< W, \Delta L(t)\psi_0 >}{< W, P(t)\psi(t) >} - \frac{< W, \Delta L(t)\Delta\psi(t) >}{< W, P(t)\psi(t) >}$$

$$\left.\begin{array}{l} \delta\rho_{effect}(t) \equiv \dfrac{< W, [\Delta P_{effect}(t) - \Delta A_{effect}(t)]\psi_0 >}{< W, P(t)\psi(t) >} \\[4mm] \delta\rho_{effect\Delta\phi}(t) \equiv \dfrac{< W, [\Delta P_{effect}(t) - \Delta A_{effect}(t)]\Delta\psi(t) >}{< W, P(t)\psi(t) >} \end{array}\right\}, \quad effect \in \left\{\begin{array}{l} ppm \\ ft \\ mt \\ md \\ other \end{array}\right\}$$

Thus, the entire change in reactivity since the beginning of the transient is expressed as a sum of all the components:

$$\Delta\rho(t) = \delta\rho_{\Delta\phi,\gamma,J} + \delta\rho_{ppm} + \delta\rho_{ft} + \delta\rho_{mt} + \delta\rho_{md} + \delta\rho_{other}$$

$$+ \delta\rho_{ppm\Delta\phi} + \delta\rho_{ft\Delta\phi} + \delta\rho_{mt\Delta\phi} + \delta\rho_{md\Delta\phi} + \delta\rho_{other\Delta\phi}$$

In the next section, it will be shown how these reactivity components may be used to explain core phenomena during a boron dilution transient. Interpretation of these phenomena demonstrates the usefulness of a coupled code system like RELAP5/PANBOX.

2.2.3 Boron Dilution Calculation

Under natural circulation flow conditions, boron dilution transients can exhibit a strong coupling between the plant thermalhydraulics and core neutron kinetics phenomena. The reason

for this is twofold: firstly, the core power is the driving force of the coolant flow; and secondly, the local boron concentration in the coolant has a direct influence on the core reactivity.

The postulated accident scenario evolves according to the following sequences. A small break loss of coolant accident (LOCA) has occurred, and the core is shut down with an automatic scram. Steam which formed in the upper plenum of the reactor during the LOCA event is transported through the primary coolant system to the steam generators, where it recondenses. This steam entrains relatively few boron particles, and the subsequent condensate has a very low boron concentration. As all the steam condenses, a large slug of unboronated water forms in the primary coolant system. Eventually, decay heat from the core leads to the establishment of natural circulation conditions, and the slug of deboronated water is swept uniformly into all coolant channels of the core. As a hypothetical scenario, conditions are assumed such that the slug is large enough and has little enough boron so that the reactor will be rendered prompt critical.

Figure 2.5: Power history and input reactivity of example transient.

The calculated total power and reactivity (in pcm=10^{-5}) of the core are shown as functions of time in Figure 2.5. Figure 2.6 shows the important contributions to the change in reactivity over time. As expected, when the boron dilution begins, the largest contribution to reactivity is due to changes in boron concentration. When the reactor reaches a prompt critical state, the strong power surge gives rise to an increase in the fuel temperature and a reduction in the

16

moderator density. The feedback effects are seen in the strong negative reactivity compo-
nents during this time period. Of great interest are the reactivity contributions between times
t_1 and t_2. The contributions are plotted along with the total reactivity in Figure 2.7. Reactiv-

Figure 2.6: Important reactivity contributions of the transient.

Figure 2.7: Total reactivity and reactivity contributions.

ity contributions are translated to zero at the time when the reactor first becomes critical so that the phenomena are more clearly examined.

From Figure 2.7, it is possible to study which feedback mechanisms are most important in abating the power surge. As the time of maximum reactivity is approached (t_3), increases in the fuel temperature decrease the core reactivity. There is also some reduction of reactivity due to changes in the moderator density, however this effect is somewhat delayed until the energy generated during the surge is transferred from the fuel to the moderator. At first sur-

prising is the reactivity contribution due to boron between times t_3 and t_4. Despite the fact that the boron concentration in the core is decreasing, the reactivity contribution of boron also decreases.

Figure 2.8 shows the separate contributions of $\delta\varrho_{ppm}$, $\delta\varrho_{ppm\Delta\phi}$ translated to zero from the time of maximum reactivity (t_3). It is seen from Figure 2.8 that the reduction in reactivity

Figure 2.8: Separation of the reactivity contributions due to changes in boron concentration.

contribution due to boron comes from the second order term $\delta\varrho_{ppm\Delta\phi}$, which is the reactivity contribution of the change in boron concentration combined with the change in flux shape. Why this reactivity contribution is negative can be seen in Figure 2.9, where the core axial boron distribution and axial flux shape are plotted at times t_3 and t_4. The flux distribution in the core changes such that the net neutron absorption due to boron *increases*. Thus, the change in the flux shape is the primary phenomenon which abates the power surge. It is essential to note that this effect *could not* be accounted for if only first order perturbation theory had been used to calculate the reactivities for a point neutron kinetics model, since $\delta\varrho_{ppm\Delta\phi}$ would be neglected in that case. Therefore, for some transients, *changes in the neutron flux shape* have a crucial influence on the reactivity of the core. In this case, the perturbation is largely uniform in the radial direction of the core, and a one–dimensional kinetics model

Figure 2.9: Axial flux and axial boron distributions at time of maximum reactivity and time of recriticality.

would likely describe transient reasonably well. However, three dimensional neutron kinetics calculations would definitely be needed if the diluted boron slug entered only a fraction of the total number of channels in the core.

2.3 Summary

The simple reactivity edit option which has been developed here highlights the necessity of space dependent models for the simulation of some postulated accident scenarios. In the case presented, the abatement of a power surge during a hypothetical boron dilution transient can be fully explained with the calculation of the various reactivity contributions. Here, it was shown that the change in the flux shape during the calculated boron dilution transient is the largest contribution which brings the reactor back to a subcritical state after the power surge. This observation identifies the importance that a space dependent neutron kinetics model be used to analyze such a transient; it also demonstrates the value of an analysis tool like RE-LAP5/PANBOX. However, it would be wrong to claim that a three–dimensional kinetics model is necessary for the calculation of all postulated accident scenarios. In fact, even with transients that require a three–dimensional kinetics model during some time period, it may not be necessary to use this model the during the entire transient. When the flux shape is not changing, a point kinetics model can be used; when only the axial flux shape is changing, a one–dimensional model can be used. The previous calculation highlights the importance

of the flux shape, and changes in flux shape, on the development of the transient. This concept will reappear in chapters 4, 5, and 6.

3. Consistency and Stability Considerations

Before the adaptive multi–level algorithm is developed, this chapter will address some theoretical considerations of the numerical discretization method used in PANBOX, the NEM. Both consistency and stability of the NEM will be examined. The fact that the NEM is consistent will be used in the perturbation theory formulation found in Chapter 4. The consistency and stability of the NEM are furthermore fundamental to its convergence.

It is well known that a discretization of a partial differential equation (PDE) initial value problem (IVP) should be consistent, convergent and stable. Consistent discretizations reduce to the original PDEs in the limit as the discretized parameters are reduced to zero. Stable schemes have solutions which are in some way bounded. When the solution of the discretized equations converges to the solution of the IVP as the discretized variables are reduced to zero, then the discretization is also convergent. Lax's Equivalence Theorem states the following relationship between consistency, stability and convergence:

> 'Given a properly posed initial–value problem and a finite–difference approximation
> to it that satisfies the consistency condition, stability is the necessary and sufficient
> condition for convergence.'[51]

In this chapter, it will be shown that the NEM is consistent with the multigroup neutron diffusion equation. Furthermore, stability conditions will be derived for various neutron kinetics models, including the NEM.

3.1 Consistency of the Nodal Expansion Method

Huang and Zhang[78] have shown that the M2B2 variant of the NEM is a special case of the generalized primal hybrid finite element method. In this section, a simpler proof is presented that a more general class of nodal expansion methods is consistent with the multigroup diffusion equation. i.e. the equations of the NEM converge to the continuous form of the multigroup diffusion equation as the mesh spacing is reduced to zero.

3.1.1 Notation and Preliminaries

Consider the NEM equations to be defined on a simply connected Cartesian geometry domain $V \subset R^3$ with a boundary $S \subset R^2$ consisting of a finite number of smooth planes. The position vector within this domain is $(x,y,z)^T \in V$. The domain V is partitioned into N subdomains Ω^m, $\{\Omega^m \subset V, m \in I, 1 \leq m \leq N\}$. The set of these subdomains is $\Omega = (\Omega^1,...,\Omega^N)$. The subdomains are hereafter referred to as 'nodes'. The nodes are rectangular prisms, of dimension (a_x^m, a_y^m, a_z^m), with vertices located in V at (x^m, y^m, z^m), $(x^m + a_x^m, y^m, z^m)$, $(x^m, y^m + a_y^m, z^m), (x^m, y^m, z^m + a_z^m)$, $(x^m + a_x^m, y^m + a_y^m, z^m)$, $(x^m + a_x^m, y^m, z^m + a_z^m)$, and $(x^m + a_x^m, y^m + a_y^m, z^m + a_z^m)$. In each node Ω^m a coordinate space is defined by $\{(\xi, \eta, \varsigma), 0 \leq \xi \leq 1, 0 \leq \eta \leq 1, 0 \leq \varsigma \leq 1\}$. The coordinates in V and Ω^m are related by the transformation

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_x^m & 0 & 0 \\ 0 & a_y^m & 0 \\ 0 & 0 & a_z^m \end{bmatrix} \begin{bmatrix} \xi \\ \eta \\ \varsigma \end{bmatrix} + \begin{bmatrix} x^m \\ y^m \\ z^m \end{bmatrix}$$

Each node has six rectangular boundaries, denoted by $\Theta_{\xi l}^m$, $\Theta_{\xi r}^m$, $\Theta_{\eta l}^m$, $\Theta_{\eta r}^m$, $\Theta_{\varsigma l}^m$, and $\Theta_{\varsigma r}^m$. The boundary of a node Ω^m is denoted by $\partial\Omega^m = \{\Theta_{\xi l}^m, \Theta_{\xi r}^m, \Theta_{\eta l}^m, \Theta_{\eta r}^m, \Theta_{\varsigma l}^m, \Theta_{\varsigma r}^m\}$. These boundaries are planes perpendicular to the coordinate direction indicated by the first subscript, and located at the left ($u=0$, $u \in \{\xi,\eta,\varsigma\}$) or right ($u=1$, $u \in \{\xi,\eta,\varsigma\}$) side of the node, as indicated by the second subscript: l, or r, respectively. A node may also have up to six neighbour nodes. Neighbour nodes of node Ω^m are denoted by $\Omega_{\xi l}^m$, $\Omega_{\xi r}^m$, $\Omega_{\eta l}^m$, $\Omega_{\eta r}^m$, $\Omega_{\varsigma l}^m$, and $\Omega_{\varsigma r}^m$. $\{\Omega_{\xi l}^m, \Omega_{\xi r}^m, \Omega_{\eta l}^m, \Omega_{\eta r}^m, \Omega_{\varsigma l}^m, \Omega_{\varsigma r}^m\} \subset \Omega$. The intersection of a node with its neighbour node is the nodal face: $\partial\Omega^m \cap \partial\Omega_{ui}^m = \Theta_{ui}^m$, for $u=\xi,\eta,\varsigma$, $i=l,r$. The intersection of any two nodal faces is denoted by $\Theta^{mn} \equiv \partial\Omega^m \cap \partial\Omega^n$, where $\Theta^{mn} = \Theta_{ui}^m$ if $\Omega_{ui}^m = \Omega^n$, otherwise $\Theta^{mn} = \{0\}$. The intersection of a node with S is $\Theta^{m0} \equiv \partial\Omega^m \cap S$.

Within each node Ω^m, there is identified a set of system parameters: D_g^m, Σ_{rg}^m ($g=1,..,N_G$), $\Sigma_{gg'}^m$, and $F_{gg'}^m$ ($g=1,...,N_G$; $g'=1,...,N_G$).

For a general class of nodal expansion methods, the following quantities are defined in each node:

(a) the nodal averaged flux for group g, $\{g=1,...,N_G\}$: ϕ_g^m;

(b) two nodal face–averaged partial currents for each group g and face ui: j_{gui}^{+m}, and j_{gui}^{-m}, $u \in \{\xi,\eta,\varsigma\}$, $i \in \{l,r\}$; and

(c) 1–D "transverse integrated" nodal expansion functions for each group g $\{g=1...,G\}$ and direction u $\{u=\xi,\eta,\varsigma\}$: $\Psi_{gu}^m(u)$, $0 \le u \le 1$. These functions are continuous and possess continuous first derivatives with respect to u on Ω^m. This means that they belong to the first Sobolev space of functions on Ω^m, $\Psi_{gu}^m \in H^1$.

A general class of nodal expansion methods solves the following system of equations:

$$\frac{D_g^m}{a_u^m} \frac{d\Psi_{gu}^m(u)}{du}\bigg|_{u=0} + j_{gul}^{+m} - j_{gul}^{-m} = 0 \tag{3.1}$$

$$\frac{D_g^m}{a_u^m} \frac{d\Psi_{gu}^m(u)}{du}\bigg|_{u=1} + j_{gur}^{+m} - j_{gur}^{-m} = 0 \tag{3.2}$$

24

$$\int_0^1 \Psi_{gu}^m(u)\,du = \phi_g^m \tag{3.3}$$

$$\Psi_{gu}^m(0) = 2\left[j_{gul}^{+m} + j_{gul}^{-m}\right] \tag{3.4}$$

$$\Psi_{gu}^m(1) = 2\left[j_{gur}^{+m} + j_{gur}^{-m}\right] \tag{3.5}$$

$$\sum_{u=\xi,\eta,\varsigma} \frac{1}{a_u}\left[j_{gur}^{+m} - j_{gur}^{-m} + j_{gul}^{-m} - j_{gul}^{+m}\right] + \Sigma_{rg}^m\phi_g^m = \sum_{g'=1}^{N_G}\left[\Sigma_{gg'}^m\phi_{g'}^m + \frac{1}{\lambda}F_{gg'}^m\phi_{g'}^m\right] \tag{3.6}$$

(where $a_\xi \equiv a_x^m$, etc.). The solutions of individual nodes are coupled by

$$j_{gul}^{+m} = j_{gur}^{+n}, \quad \text{for } \Omega^n = \Omega_{u0}^m \tag{3.7}$$

and

$$j_{gul}^{-m} = j_{gur}^{-n}, \quad \text{for } \Omega^n = \Omega_{u0}^m \tag{3.8}$$

for $u=\xi,\eta,\varsigma$. Boundary conditions are given by

$$j_{gul}^{+m} = 0, \quad \forall \; \Theta_{ul}^m \bigcap S \neq \{0\} \tag{3.9}$$

and

$$j_{gur}^{-m} = 0, \quad \forall \; \Theta_{ur}^m \bigcap S \neq \{0\} \tag{3.10}$$

for $u=\xi,\eta,\varsigma$.

3.1.2 Consistency with the Multigroup Neutron Diffusion Equation

First it will be shown that the solutions of equations (3.1) to (3.6) reduce to the solution of the continuous variable multigroup neutron diffusion equation in the limit as the dimensions of the node go to zero. The 3–D nodal expansion function $\varphi_g^m(\xi,\eta,\varsigma)$ is defined on Ω^m as,

$$\varphi_g^m(\xi,\eta,\varsigma) \equiv \Psi_{g\eta}^m(\eta) + \Psi_{g\varsigma}^m(\varsigma) + \Psi_{g\xi}^m(\xi) - 2\phi_g^m \tag{3.11}$$

This function obeys the relation

$$a_u\int_0^1\int_0^1 \hat{u}\cdot\nabla\varphi_g^m dvdw = \frac{d\Psi_{gu}^m(u)}{du}, \quad u,v,w = cyclic(\xi,\eta,\varsigma) \tag{3.12}$$

Substitution of equation (3.12) into equations (3.1) and (3.2) yields

$$D_g\int_0^1\int_0^1 \hat{u}\cdot\nabla\varphi_g^m|_{u=0}dvdw + j_{gul}^{+m} - j_{gul}^{-m} = 0 \tag{3.13}$$

and

$$D_g \int_0^1 \int_0^1 \hat{u} \cdot \nabla \varphi_g^m \big|_{u=1} dv \, dw + j_{gur}^+ - j_{gur}^- = 0 \tag{3.14}$$

Substitution of the 3–D expansion function (3.11) into (3.3) yields,

$$\int_0^1 \int_0^1 \int_0^1 \varphi_g^m d\xi \, d\eta \, d\varsigma = \phi_g^m \tag{3.15}$$

Equations (3.13) through (3.15) may be substituted into equation (3.6), resulting in

$$-\sum_{u=\xi,\eta,\varsigma} \frac{1}{a_u} \left[D_g \int_0^1 \int_0^1 \hat{u} \cdot \nabla \varphi_g^m \big|_{u=1} dv \, dw - D_g \int_0^1 \int_0^1 \hat{u} \cdot \nabla \varphi_g^m \big|_{u=0} dv \, dw \right]$$

$$+ \Sigma_{rg}^m \int_0^1 \int_0^1 \int_0^1 \varphi_g^m d\xi \, d\eta \, d\varsigma = \sum_{g'=1}^{N_G} \left[\Sigma_{gg'}^m + \frac{1}{\lambda} F_{gg'}^m \right] \int_0^1 \int_0^1 \int_0^1 \varphi_{g'}^m d\xi \, d\eta \, d\varsigma$$

This equation is multiplied by the volume of the node. The first sum on the LHS of the equation is rewritten as a surface integral. The other integrals are written as volume integrals, yielding the result

$$-D_g \int_{\partial\Omega^m} \hat{s} \cdot \nabla \Phi_g^m(x,y,z) dS + \Sigma_{rg} \int_{\Omega^m} \Phi_g^m(x,y,z) dV = \tag{3.16}$$

$$= \sum_{g'=1}^{N_G} \left[\Sigma_{gg'}^m + \frac{1}{\lambda} F_{gg'}^m \right] \int_{\Omega^m} \Phi_g^m(x,y,z) \, dV$$

where

$$\Phi_g^m(x,y,z) \equiv \varphi_g^m \left(\frac{x - x^m}{a_x^m}, \frac{y - y^m}{a_y^m}, \frac{z - z^m}{a_z^m} \right). \tag{3.17}$$

Because the expansion functions Ψ are smooth and continuous within Ω^m, so are the functions Φ_g^m. Therefore, the divergence theorem can be applied to equation (3.16) which becomes

$$\int_{\Omega^m} \left[-D_g \nabla \cdot \nabla \Phi_g^m(x,y,z) + \Sigma_{rg} \Phi_g^m(x,y,z) - \sum_{g'=1}^{N_G} \left(\Sigma_{gg'}^m + \frac{1}{\lambda} F_{gg'}^m \right) \Phi_g^m(x,y,z) \right] dV = 0$$

In the limit as the nodal volume goes to zero, this reduces to the multigroup diffusion equation in a homogeneous region:

$$-D_g \nabla \cdot \nabla \Phi_g^m(x,y,z) + \Sigma_{rg} \Phi_g^m(x,y,z) - \sum_{g'=1}^{N_G} \left(\Sigma_{gg'}^m + \frac{1}{\lambda} F_{gg'}^m \right) \Phi_g^m(x,y,z) = 0 \tag{3.18}$$

$$(x,y,z) \in \Omega^m$$

Since equation (3.18) has been derived for any homogeneous node in general, it is equally valid for all nodes.

3.1.3 Consistency of the Interface Conditions

The coupling conditions between the nodes are now examined. Integration of the expansion function (3.11) over the nodal boundaries yields

$$\int_0^1 \int_0^1 \varphi_g^m\big|_{u=0}\, dv\, dw = \Psi_{gu}^m(0)$$

(3.19)

and

$$\int_0^1 \int_0^1 \varphi_g^m\big|_{u=1}\, dv\, dw = \Psi_{gu}^m(1)$$

(3.20)

$$u, v, w = cyclic(\xi, \eta, \varsigma)$$

The 1–D expansion function boundary conditions, equations (3.4) and (3.5), along with the nodal interface conditions, equations (3.7) and (3.8) may be combined to find

$$\Psi_{gu}^m(0) = \Psi_{gu}^m(1) \quad \text{for} \quad \Omega^n = \Omega_{ul}^m, \quad u \in \{\xi, \eta, \varsigma\}$$

(3.21)

The 3–D nodal expansion function of the 'ul' neighbour node $\Omega^n = \Omega_{ul}^m$ is defined as

$$\varphi_g^n(\xi, \eta, \varsigma) \equiv \Psi_{g\eta}^m(\eta) + \Psi_{g\varsigma}^m(\varsigma) + \Psi_{g\xi}^m(\xi) - 2\phi_g^n$$

(3.22)

Equations (3.19) to (3.22) are combined to yield the following interface condition in terms of the 3–D expansion functions:

$$\int_0^1 \int_0^1 \varphi_g^m\big|_{u=0}\, dv\, dw = \int_0^1 \int_0^1 \varphi_g^n\big|_{u=1}\, dv\, dw$$

(3.23)

By defining Φ_g^n in the same manner as Φ_g^m [see equation (3.17)], equation (3.23) may be written as

$$\int_{\Theta_{ul}^m = \Theta^{mn}} \Phi_g^m(x, y, z)\, dS = \int_{\Theta_{ur}^n = \Theta^{mn}} \Phi_g^n(x, y, z)\, dS \quad u = x, y, z$$

(3.24)

In the limit as the mesh size goes to zero, this becomes

$$\Phi_g^m(x, y, z)\big|_{\Theta^{mn}} = \Phi_g^n(x, y, z)\big|_{\Theta^{mn}} \quad u = x, y, z$$

(3.25)

which is one of the two interface conditions of the multigroup diffusion equation between two regions. The other boundary condition can be found by combining the remaining 1–D expansion function boundary conditions, equations (3.1), (3.2), with the interface conditions, equations (3.7) and (3.8), to obtain

$$\frac{D_g^m}{a_u^m} \frac{d\Psi_{gu}^m(u)}{du}\bigg|_{u=0} = \frac{D_g^n}{a_u^n} \frac{d\Psi_{gu}^n(u)}{du}\bigg|_{u=1} \quad \text{for} \quad \Omega^n = \Omega_{ul}^m$$

(3.26)

The relationship (3.12) is used in this equation to obtain

$$D_g^m \int_0^1 \int_0^1 \hat{u} \cdot \nabla \varphi_g^m dv dw = D_g^n \int_0^1 \int_0^1 \hat{u} \cdot \nabla \varphi_g^n dv dw \tag{3.27}$$

$$u, v, w = cyclic(\xi, \eta, \varsigma)$$

In terms of x,y,z, this reduces to:

$$\int_{\Theta_{ul}^m = \Theta^{mn}} D_g^m \hat{u} \cdot \nabla \Phi_g^m(x, y, z) dS = \int_{\Theta_{ur}^n = \Theta^{mn}} D_g^n \hat{u} \cdot \nabla \Phi_g^n(x, y, z) dS \tag{3.28}$$

$$u = x, y, z$$

As the mesh spacing reduces to zero, equation (3.28) becomes

$$D_g^m \hat{u} \cdot \nabla \Phi_g^m(x, y, z)|_{\Theta_{ul}^m = \Theta^{mn}} = D_g^n \hat{u} \cdot \nabla \Phi_g^n(x, y, z)|_{\Theta_{ur}^n = \Theta^{mn}} \quad u = x, y, z$$

which is the second interface condition of the multigroup neutron diffusion equation between two homogeneous regions.

3.1.4 Boundary Conditions

Equations (3.1), (3.2), (3.4) and (3.5) may be combined to yield the following equations for the partial currents in terms of the 1–D expansion functions:

$$j_{gul}^{+m} = \frac{1}{4} \Psi_{gu}^m(0) - \frac{1}{2} \frac{D_g^m}{a_u^m} \frac{d\Psi_{gu}^m(u)}{du}\bigg|_{u=0}$$

$$j_{gur}^{-m} = \frac{1}{4} \Psi_{gu}^m(1) + \frac{1}{2} \frac{D_g^m}{a_u^m} \frac{d\Psi_{gu}^m(u)}{du}\bigg|_{u=1} \tag{3.29}$$

These equations are substituted into (3.9) and (3.10) and then integrated over Θ_{ul}^m and Θ_{ur}^m, respectively. The relationships (3.12), (3.19) and (3.20) can then be used to cast the boundary conditions in terms of the 3D expansion function, Φ_g^m. The result is

$$\int_{\Theta_{ul}^m} \left[\frac{1}{4} \Phi_g^m - \frac{D_g^m}{2} \hat{u} \cdot \nabla \Phi_g^m \right] dS = 0, \quad \forall \ \Theta_{ul}^m \bigcap S \neq \{0\} \tag{3.30}$$

and

$$\int_{\Theta_{ur}^m} \left[\frac{1}{4} \Phi_g^m + \frac{D_g^m}{2} \hat{u} \cdot \nabla \Phi_g^m \right] dS = 0, \quad \forall \ \Theta_{ur}^m \bigcap S \neq \{0\} \tag{3.31}$$

As the mesh size is reduced to zero, these equations reduce to

$$\left(\frac{1}{4} \Phi_g^m - \frac{D_g^m}{2} \hat{u} \cdot \nabla \Phi_g^m \right)_{u=0} = 0, \quad \forall \ \Theta_{ul}^m \bigcap S \neq \{0\} \tag{3.32}$$

and

$$\left(\frac{1}{4} \Phi_g^m + \frac{D_g^m}{2} \hat{u} \cdot \nabla \Phi_g^m \right)_{u=1} = 0, \quad \forall \ \Theta_{ur}^m \bigcap S \neq \{0\} \tag{3.33}$$

Equations (3.32) and (3.33) are the zero–incoming partial current boundary conditions for the multigroup neutron diffusion equation.

3.1.5 Summary

It has been shown that in the limit as the mesh spacing reduces to zero, the governing equations, interface conditions, and boundary conditions of the NEM equations reduce to those of the continuous neutron diffusion equation.

The above derivation is quite general. It may be applied to many different nodal expansion methods: as long as the expansion functions meet the conditions given in section 3.1.1, any such expansion function may be used. While this derivation shows that a general class of NEMs are *consistent* with the multigroup diffusion equation, it says nothing about the *accuracy* of the solution when coarse mesh nodes are used. Accuracy estimates of the M2B2 variant of the NEM in slab geometry have recently been derived by Penland, Azmy and Turinsky.[79] While their analysis also proves consistency, it only does this for one specific form of the NEM, and only for slab geometry. The simple development performed here is more general, but gives no accuracy estimate. Regardless of the accuracy on coarse meshes, it is assured that the 3–D flux expansions converge to the solution of the multigroup diffusion equation in the limit as the mesh spacing is reduced to zero.

3.2 Considerations for Stability Analysis

Consider a linear discretization in which $\vec{u}^n$ represents the vector of dependent variables to be solved for at time step n, given the initial conditions at $\vec{u}^0$ at time $t=t_0$. The difference equations for a one–step time integration method may be explicitly written in matrix form as

$$\mathbf{B}_1 \vec{u}^{n+1} = \mathbf{B}_0 \vec{u}^n. \tag{3.34}$$

The matrices $\mathbf{B}_0$ and $\mathbf{B}_1$ have elements which will depend partly on the time step Δt and, should the associated continuous variable problem be time and space dependent, the spatial discretization parameters. For the analysis of the numerical scheme, it is usually assumed that the spatial mesh size is functionalized to the time step size. For instance, in Cartesian geometry, it is usually assumed that $(\Delta x, \Delta y, \Delta z) = [g_1(\Delta t), g_2(\Delta t), g_3(\Delta t)]$. Supposing that $\mathbf{B}_1$ has an inverse (however, not supposing that the inverse is known explicitly), the amplification matrix may be defined as

$$\mathbf{C}(\Delta t) \equiv \mathbf{B}_1^{-1} \mathbf{B}_0 . \tag{3.35}$$

In this way, equation (3.34) may be written as

$$\vec{u}^{n+1} = \mathbf{C}(\Delta t) \vec{u}^n. \tag{3.36}$$

Definition 3.1:[51] The approximation $\mathbf{C}(\Delta t)$ is stable over the integration period T, if for some $\tau > 0$, the infinite set of operators

$$C(\Delta t)^n, \quad \begin{array}{l} 0 < \Delta t \le \tau, \\ 0 \le n\Delta t \le T, \end{array} \tag{3.37}$$

is uniformly bounded.

Essentially, this means that for a stable scheme, there exists a maximum value to which any component in the solution can be amplified over the specified time period T, provided that the time step falls within the range of values between 0 and τ. This concept of stability was formulated particularly with conservative physical systems in mind, although it can be useful for analyzing non–conservative systems. Proof of stability follows automatically if the bound $\|C(\Delta t)\| \le 1$. However, in other cases the proof may be more difficult. Should the system contain fixed source terms, or should the physical system be non–conservative, as is the case with reactor kinetics, then the following theorem of Kreiss and Strang is useful.

Theorem 3.1 [*The Kreiss–Strang Theorem*]:[51] If the difference system

$$\vec{u}^{n+1} = C(\Delta t)\vec{u}^n \tag{3.38}$$

is stable, and $Q(\Delta t)$ is a bounded family of operators, then the difference system

$$\vec{u}^{n+1} = [C(\Delta t) + \Delta t Q(\Delta t)]\vec{u}^n \tag{3.39}$$

is also stable.

The proof of Theorem 3.1 is given in reference 51. It is a very useful theorem, since it allows for some growth of the solution over the time of integration T without the requirement that Q^n be bounded. The growth, however, is bounded. Often it is possible to write an amplification matrix in the form of $[C(\Delta t)+\Delta t Q(\Delta t)]$, where it is easier to prove that $C^n(\Delta t)$ and $Q(\Delta t)$ are bounded than proving that $[C(\Delta t)+\Delta t Q(\Delta t)]^n$ is bounded. This will be demonstrated in sections 3.3.1 and 3.3.2.

Although Definition 3.1 is a strict definition of stability, it is often impractical to apply it to engineering problems of interest (see, for example the discussions in references 51 and 52). A somewhat weaker definition of stability was originally formulated by von Neumann in 1938, and remains a powerful analysis tool to this day.

Definition 3.2: Let ϱ be the spectral radius of the amplification matrix $C(\Delta t)$. Then the von Neumann stability criterion is

$$\varrho \le 1 + O(\Delta t). \tag{3.40}$$

It is shown in reference 51 that the von Neumann stability criterion is a necessary condition for stability defined by Definition 3.1. However, it is not always necessary and sufficient. *It is necessary and sufficient* when the amplification matrix $C(\Delta t)$ is a *normal matrix* (a matrix A is normal when $AA^* = A^*A$).[51]

Not to be confused with the von Neumann sufficient condition for stability is the *von Neumann method* for stability analysis. This method will be called the *local mode analysis* method, to avoid confusion with the von Neumann criterion. Local mode analysis may be performed on a linear

system of algebraic equations. Because of the linearity, error components of the solution obey the homogeneous form of the system of equations. For example, consider the system of equations

$$\mathbf{B}_1 \vec{u}^{n+1} = \mathbf{B}_0 \vec{u}^n + q .$$

(3.41)

Here the vector $\vec{u}$ represents the exact solution of the discretized equations. Suppose only an approximate solution to (3.41) is known. This approximate solution, which is denoted as $\vec{u}^h$, has an error component $\vec{e}$ such that

$$\vec{u}^n = \left(\vec{u}^h\right)^n + \vec{e}^n$$

(3.42)

When the approximate solution is used to solve (3.41) for $\left(\vec{u}^h\right)^{n+1}$ at the time step $n+1$, i.e.

$$\mathbf{B}_1 \left(\vec{u}^h\right)^{n+1} = \mathbf{B}_0 \left(\vec{u}^h\right)^n + \vec{q} ,$$

(3.43)

then the error components will evolve according to the equation

$$\mathbf{B}_1 \vec{e}^{n+1} = \mathbf{B}_0 \vec{e}^n .$$

(3.44)

It is because the operators $\mathbf{B}_0$ and $\mathbf{B}_1$ are linear that the error term can be separated out from the exact solution. The error can then be decomposed into a discrete Fourier series, provided that the boundary conditions are periodic. Then the evolution of Fourier modes may be analysed individually. The numerical scheme is stable when the growth of all Fourier modes is bounded. In practise, the local mode analysis is also used when nonperiodic boundary conditions are present: it is considered that the Fourier series representation is still good 'away from the boundary,' and in practical applications this approximate treatment of the boundary conditions does not usually detriment the validity of the stability criteria.[52]

In this chapter, three separate methods of deriving stability criteria will be considered. The first method utilizes Definition 3.1 and Theorem 3.1 to derive sufficient stability conditions for the point kinetics equations. Unfortunately, this method turns out to be too impractical to apply to one and three dimensional neutron kinetics models. The second method utilizes the *von Neumann necessary conditions* for stability, applied to the point kinetics equations. The third method, *local mode analysis,* applied to the discretized NEM equations, examines the equations only at an isolated (but general) node. It ignores the effects of boundary conditions and inhomogeneous parameters. In sections 3.2.1 and 3.2.2, sufficient stability conditions are derived for the point kinetics equations. In section 3.3.3, necessary von Neumann conditions are derived using the matrix method. Section 3.3.4 compares these conditions to numerical experiments. It will be shown that the sufficient conditions yield time step sizes too conservative to be of practical use, and that the necessary conditions yield time step sizes adequate for practical use. This motivates the approach used in section 3.4, where local mode analysis is combined with the matrix method in order to derive necessary stability conditions for the one and three dimensional NEM models. Before these stability issues are addressed, the time discretization techniques of PANBOX will be examined.

3.2.1 Time Discretization Techniques Used in PANBOX 2

PANBOX makes use of the exponential transformation fully implicit time discretization method, which was originally developed for an alternating semi–implicit scheme for the multigroup diffusion equation in two–dimensional geometry by Reed and Hansen in the late 1960s.[17]

The exponential transform technique as applied to the NEM is described in Appendix A. With this technique, the time derivative of the nodal averaged flux is approximated as,

$$\frac{d\phi_g^m(t)}{dt} = \frac{(1 + \omega^m \Delta t)\phi_g^m(t) - e^{\omega^m \Delta t}\phi_g^m(t_0)}{\Delta t} \tag{3.45}$$

The discretized form of the neutron precursor equation is

$$c_i^m(t) = c_i^m(t_0)e^{-\lambda_i \Delta t} + \frac{1}{\lambda}\left[\frac{1 - e^{-(\omega^m + \lambda_i)\Delta t}}{\omega^m + \lambda_i}\sum_{g'=1}^{N_G}\sum_{j=1}^{N_J}\beta_i^j \nu \Sigma_{fg'}^{j,m}\phi_{g'}^m(t)\right]. \tag{3.46}$$

The dynamic frequencies, ω^m are determined in the iterative solution of the NEM equations by

$$\omega^m = \frac{1}{\Delta t}\ln\left[\frac{\displaystyle\sum_{g=1}^{N_G}\phi_g^m(t)}{\displaystyle\sum_{g=1}^{N_G}\phi_g^m(t_0)}\right]. \tag{3.47}$$

The exponential transform technique was adapted to the NEM by Finnemann[24], however no theoretical stability analysis was performed for the discretization. Reed and Hansen[17] had earlier performed a rigorous stability analysis of their alternating semi–implicit scheme using Definition 3.1. It must be noted that the ω^m's are not constant in time. More importantly, due to the dependence of ω^m on the nodal averaged flux, the expressions (3.45) and (3.46) are nonlinear in the flux. Therefore, the scheme must first be linearized before a linear stability analysis can be performed. Reed and Hansen linearized their equations by assuming that the ω^m's remained constant in time. In a later paper,[18] Ferguson and Hansen extended the alternating semi–implicit scheme to three dimensions, however they did not significantly extend the theoretical analysis performed by Reed and Hansen. They relied on numerical experiments to demonstrate the stability of the numerical scheme. Later, Buckner and Stewart[20] applied the exponential transformation technique to a direct finite volume discretization of the multigroup neutron diffusion equations in three–dimensional geometry. They attempted to generalize the theoretical stability analysis performed in reference 17, however, they were unable to prove stability in the general case or derive any kind of applicable stability criterion.

It has often been assumed in the literature that implicit treatment of the new independent variable [T in Appendix A] will yield a stable numerical scheme. The experimental computational evidence[17,18,20,24] certainly supports this assertion. These numerical investigations are further necessary to investigate the nonlinear effects which cannot be analysed in a linear stability

analysis. However, despite the presence of a large data base of experience with this method, theoretically–derived stability criteria are still highly desirable. Sections 3.3 and 3.4 give derivations of stability criteria for this time integration method applied to point kinetics and one and three dimensional NEM models. Of these neutron kinetics models, the simplest to analyze is the point kinetics. It, therefore, is where the analysis will begin.

3.3 Stability Analysis of the Point Kinetics Model

The point kinetics equations with I precursor groups may be written in the form,

$$\frac{dP}{dt} = \frac{(\rho - \beta)}{\Lambda}P + \sum_{i=1}^{I} \lambda_i C_i , \tag{3.48}$$

$$\frac{dC_i}{dt} = \frac{\beta_i}{\Lambda}P - \lambda_i C_i , \qquad i = 1,...,I. \tag{3.49}$$

where

$$\beta = \sum_{i=1}^{I} \beta_i \tag{3.50}$$

In the most accurate representations, the terms ρ and β_i are dependent themselves on P through thermalhydraulic coupling. This dependence makes the system of equations (3.48) and (3.49) nonlinear. However, in the following analysis, the system of equations (3.48) and (3.49) will be linearized by treating the ρ and the β_i's to be piecewise constant functions of time, independent of P.

3.3.1 Exponential Transform Method Discretization

When the exponential transform discretization is applied to the point kinetics equations, equations (3.48) and (3.49) become

$$\left(\omega + \frac{1}{\Delta t}\right)P^{j+1} - \frac{e^{\omega\Delta t}}{\Delta t}P^j = \frac{(\rho - \beta)}{\Lambda}P^{j+1} + \sum_{i=1}^{6} \lambda_i C_i^{j+1} \tag{3.51}$$

and

$$C_i^{j+1} = C_i^j e^{-\lambda\Delta t} + \frac{\beta_i}{\Lambda}\frac{1 - e^{-(\omega+\lambda_i)\Delta t}}{\omega + \lambda_i}P^{j+1} \qquad i = 1,...,6. \tag{3.52}$$

Equations (3.51) and (3.52) may be written in matrix form as

$$
\begin{bmatrix}
\left[1 + \Delta t\left(\omega - \dfrac{\rho - \beta}{\Lambda}\right)\right] & -\Delta t\lambda_1 & -\Delta t\lambda_2 & \cdots & -\Delta t\lambda_6 \\[2ex]
-\dfrac{\beta_1}{\Lambda}\left[\dfrac{1 - e^{-(\omega + \lambda_1)\Delta t}}{\omega + \lambda_1}\right] & 1 & 0 & \cdots & 0 \\[2ex]
-\dfrac{\beta_2}{\Lambda}\left[\dfrac{1 - e^{-(\omega + \lambda_2)\Delta t}}{\omega + \lambda_2}\right] & 0 & 1 & \cdots & 0 \\[2ex]
\vdots & \vdots & \vdots & & \vdots \\[2ex]
-\dfrac{\beta_6}{\Lambda}\left[\dfrac{1 - e^{-(\omega + \lambda_6)\Delta t}}{\omega + \lambda_6}\right] & 0 & 0 & \cdots & 1
\end{bmatrix}
\begin{bmatrix}
P^{j+1} \\[2ex]
C_1^{j+1} \\[2ex]
C_2^{j+1} \\[2ex]
\vdots \\[2ex]
C_6^{j+1}
\end{bmatrix}
=
$$

$$
=
\begin{bmatrix}
e^{\omega \Delta t} & 0 & 0 & \cdots & 0 \\[1ex]
0 & e^{-\lambda_1 \Delta t} & 0 & \cdots & 0 \\[1ex]
0 & 0 & e^{-\lambda_2 \Delta t} & \cdots & 0 \\[1ex]
\vdots & \vdots & \vdots & & \vdots \\[1ex]
0 & 0 & 0 & \cdots & e^{-\lambda_6 \Delta t}
\end{bmatrix}
\begin{bmatrix}
P^{j} \\[2ex]
C_1^{j} \\[2ex]
C_2^{j} \\[2ex]
\vdots \\[2ex]
C_6^{j}
\end{bmatrix}
\qquad (3.53)
$$

Equation (3.53) may be written in the more compact notation

$$
\mathbf{A}\vec{u}^{\,j+1} = \mathbf{B}\vec{u}^{\,j} \qquad\qquad (3.54)
$$

where the matrices $\mathbf{A}$ and $\mathbf{B}$ are shown respectively in equation (3.53).

3.3.2 Sufficient Conditions for Stability

The first sufficient stability condition will be given by the following theorem:

Theorem 3.2: Let the matrix $\mathbf{R}$ be defined as

$$
\mathbf{R} \equiv
\begin{bmatrix}
\Delta t\left(\omega - \dfrac{\rho - \beta}{\Lambda}\right) & -\Delta t\lambda_1 & -\Delta t\lambda_2 & \cdots & -\Delta t\lambda_6 \\[2ex]
-\dfrac{\beta_1}{\Lambda}\left[\dfrac{1 - e^{-(\omega + \lambda_i)\Delta t}}{\omega + \lambda_i}\right] & 0 & 0 & \cdots & 0 \\[2ex]
-\dfrac{\beta_2}{\Lambda}\left[\dfrac{1 - e^{-(\omega + \lambda_i)\Delta t}}{\omega + \lambda_i}\right] & 0 & 0 & \cdots & 0 \\[2ex]
\vdots & \vdots & \vdots & & \vdots \\[2ex]
-\dfrac{\beta_6}{\Lambda}\left[\dfrac{1 - e^{-(\omega + \lambda_i)\Delta t}}{\omega + \lambda_i}\right] & 0 & 0 & \cdots & 0
\end{bmatrix}
$$

then the scheme given by (3.53) is stable if $\|R\|<1$.

Proof

The proof will show that if $\|R\|<1$, then Theorem 3.1 is satisfied. First, the difference system of equation (3.54) is cast into the form of (3.39), by writing (3.54) in the form of

$$\vec{u}^{j+1} = A^{-1}B\,\vec{u}^{j}\ . \tag{3.55}$$

For this case, the matrix A is relatively simple to invert – it may be done, for instance, by row reduction. However, direct inversion of A is not useful in this situation because it is difficult to show that the resulting matrix $A^{-1}B$ results in a stable system of equations. Instead, since if $\|R\|<1$, then the inverse of A may be expressed as

$$
\begin{aligned}
A^{-1} &= (I + R)^{-1} \\
&= I - R + R^2 - R^3 + ...
\end{aligned}
\tag{3.56}
$$

The matrix B may be written in the form of

$$B = I + \varDelta t P(\Delta t) \tag{3.57}$$

where P is bounded (since P is diagonal and its individual diagonal elements are all bounded). Therefore

$$A^{-1}B = I + \varDelta t P(\Delta t)\left[I - R + R^2 - R^3 + ...\right] + \left[-R + R^2 - R^3 + ...\right] \tag{3.58}$$

I^n has an upper bound of 1 for all n, so it satisfies the stability requirements for the matrix C in equation (3.39), Theorem 3.1. What remains to be shown is that the remaining terms,

$$\varDelta t P(\Delta t)\left[I - R + R^2 - R^3 + ...\right] + \left[-R + R^2 - R^3 + ...\right] \tag{3.59}$$

can be cast into the form $\varDelta t Q(\Delta t)$, such that $Q(\Delta t)$ is bounded. The matrix R can be written in the form

$$R = \varDelta t N$$

where

$$
N = \begin{bmatrix}
\left(\omega - \dfrac{\rho - \beta}{\varLambda}\right) & -\lambda_1 & -\lambda_2 \ ... \ -\lambda_6 \\[2em]
-\dfrac{\beta_1}{\varLambda}\left[1 - \dfrac{(\omega + \lambda_1)\varDelta t}{2!} + \dfrac{(\omega + \lambda_1)^2 \varDelta t^2}{3!} - ...\right] & 0 & 0 \ \cdots \ 0 \\[2em]
-\dfrac{\beta_2}{\varLambda}\left[1 - \dfrac{(\omega + \lambda_2)\varDelta t}{2!} + \dfrac{(\omega + \lambda_2)^2 \varDelta t^2}{3!} - ...\right] & 0 & 0 \ \cdots \ 0 \\[2em]
\vdots & \vdots & \vdots \quad \vdots \\[2em]
-\dfrac{\beta_6}{\varLambda}\left[1 - \dfrac{(\omega + \lambda_6)\varDelta t}{2!} + \dfrac{(\omega + \lambda_6)^2 \varDelta t^2}{3!} - ...\right] & 0 & 0 \ ... \ 0
\end{bmatrix}
$$

Let the bound of $\mathbf{R}$ be

$$\| \mathbf{R} \| = \Delta t \| \mathbf{N} \| = a\Delta t < 1 \tag{3.60}$$

then

$$
\begin{aligned}
\| \mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ... \| \quad &\leq \quad \| \mathbf{I} \| + \| - \mathbf{R} \| + \| \mathbf{R}^2 \| + \| - \mathbf{R}^3 \| + ... \\
&\leq \quad \| \mathbf{I} \| + \| \mathbf{R} \| + \| \mathbf{R}^2 \| + \| \mathbf{R}^3 \| + ... \\
&\leq \quad \| \mathbf{I} \| + \| \mathbf{R} \| + \| \mathbf{R} \|^2 + \| \mathbf{R} \|^3 + ... \\
&\leq \quad 1 + a\Delta t + (a\Delta t)^2 + (a\Delta t)^3 + ... \\
&= \quad \frac{1}{1 - a\Delta t} \, . \tag{3.61}
\end{aligned}
$$

Therefore, the matrix series $\mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ...$ is bounded if $\|\mathbf{R}\| < 1$. Note also that the matrix series $-\mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ...$ may be written in the form

$$
\begin{aligned}
- \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ... \quad &= \quad - \mathbf{R}\left[\mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ... \right] \\
&= \quad - \Delta t \mathbf{N}\left[\mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ... \right]
\end{aligned}
$$

and

$$
\begin{aligned}
\| \mathbf{N}\left[\mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ... \right] \| \quad &\leq \quad \| \mathbf{N} \| \, \| \mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ... \| \\
&= \quad \frac{a}{1 - a\Delta t} \, . \tag{3.62}
\end{aligned}
$$

i.e. $\| \mathbf{N} [\mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ...] \|$ is bounded. Thus (3.59) may be written as

$$
\begin{aligned}
\Delta t \mathbf{P}(\Delta t)\left[\mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ... \right] + \left[- \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ... \right] \\
= \Delta t \left\{ \mathbf{P}(\Delta t)\left[\mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ... \right] - \mathbf{N}[\mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ...] \right\} \\
\equiv \Delta t \mathbf{K}(\Delta t) \tag{3.63}
\end{aligned}
$$

The product of bounded matrices is bounded, and the sum of bounded matrices is bounded. Since (3.61) and (3.62) show that $[\mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ...]$ and $-\mathbf{N}[\mathbf{I} - \mathbf{R} + \mathbf{R}^2 - \mathbf{R}^3 + ...]$ are bounded, and since $\mathbf{P}(\Delta t)$ is bounded, then $\mathbf{K}(\Delta t)$ is bounded. Then, using (3.59) and (3.63) the difference system (3.54) becomes

$$\vec{u}^{\,j+1} = \left[\mathbf{I} + \Delta t \mathbf{K}(\Delta t) \right] \vec{u}^{\,j} \, . \tag{3.64}$$

Since $\mathbf{K}(\Delta t)$ and $\mathbf{I}^n$ are bounded, the difference system satisfies the conditions of the Kreiss –Strang theorem, Theorem 3.1, proving that the difference system is stable if $\|\mathbf{R}\| < 1$.

$\square$

$\|\mathbf{R}\| < 1$ is therefore a sufficient condition for the stability of the difference system. We note that the choice of matrix norm here is

$$\| \mathbf{R} \| = \max_{\vec{v} \neq 0} \frac{|\mathbf{R}\vec{v}|}{|\vec{v}|} = \max_{\vec{v} \neq 0} \left[\frac{(\mathbf{R}\vec{v}, \mathbf{R}\vec{v})}{(\vec{v}, \vec{v})} \right]^{\frac{1}{2}} \tag{3.65}$$

This norm is reasonably straightforward to evaluate. It is

$$\| \mathbf{R} \| = \sqrt{\varrho(\mathbf{R}^T\mathbf{R})} \ , \tag{3.66}$$

where $\varrho(\mathbf{R}^T\mathbf{R})$ is the spectral radius of the matrix $\mathbf{R}^T\mathbf{R}$. The eigenvalues of $\mathbf{R}^T\mathbf{R}$ are determined as

$$s_1 = s_2 = s_3 = s_4 = s_5 = 0$$

$$s_6, s_7 = \frac{1}{2}\left\{ b^2 + \sum_{i=1}^{6}(c_i^2 + d_i^2) \pm \sqrt{\left[b^2 + \sum_{i=1}^{6}(c_i^2 + d_i^2)\right]^2 + 4\sum_{i=1}^{6} b_i^2 \sum_{i=1}^{6} d_i^2 } \right\} \tag{3.67}$$

where

$$b \equiv \Delta t\left(\omega - \frac{\rho - \beta}{\Lambda}\right)$$

$$c_i \equiv \frac{\beta_i}{\Lambda}\left[\frac{1 - e^{-(\omega + \lambda_i)\Delta t}}{\omega + \lambda_i} \right]$$

$$d_i \equiv \Delta t \lambda_i$$

Thus, $\|\mathbf{R}\|$ may be calculated directly, and it may be immediately determined if the sufficient stability condition $\|\mathbf{R}\| < 1$ is satisfied.

3.3.3 Alternate Sufficient Conditions

In the last section, a *sufficient* condition for stability was derived for the difference system given by equation (3.54). The problem with sufficient conditions is that they may specify a time step which is too small to be of practical value. The ideal condition is both necessary and sufficient. Arriving at such a condition, however, is not always possible when the amplification matrix (in this case $\mathbf{A}^{-1}\mathbf{B}$) is non–unitary (as is true in this case). Instead of trying to find a necessary and sufficient condition for stability, an alternative sufficient condition will be derived here which turns out to admit larger time steps than the one derived in the last section.

Theorem 3.3: Let the matrix S be defined as

$$\mathbf{S} \equiv \begin{bmatrix} 0 & \dfrac{-\Delta t \lambda_1}{1 + \Delta t\left[\omega - \dfrac{(\rho - \beta)}{\Lambda}\right]} & \cdots & \dfrac{-\Delta t \lambda_6}{1 + \Delta t\left[\omega - \dfrac{(\rho - \beta)}{\Lambda}\right]} \\ -\dfrac{\beta_1}{\Lambda}\left[\dfrac{1 - e^{-(\omega + \lambda_1)\Delta t}}{\omega + \lambda_1}\right] & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ -\dfrac{\beta_6}{\Lambda}\left[\dfrac{1 - e^{-(\omega + \lambda_6)\Delta t}}{\omega + \lambda_6}\right] & 0 & \cdots & 0 \end{bmatrix}$$

then the difference system (3.54) is stable if

$$\| \mathbf{S} \| < 1 \qquad (3.68)$$

and

$$\omega - \frac{(\rho - \beta)}{\Lambda} > 0 \; . \qquad (3.69)$$

are satisfied.

Proof

First the matrix $\mathbf{A}$ is written in the form

$$\mathbf{A} = \mathbf{D}(\mathbf{I} + \mathbf{S}) \qquad (3.70)$$

where

$$\mathbf{D} \equiv
\begin{bmatrix}
1 + \Delta t \left[\omega - \dfrac{(\rho - \beta)}{\Lambda} \right] & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & & \vdots \\
0 & 0 & 0 & \cdots & 0
\end{bmatrix}$$

Then the difference equation (3.54) may be rewritten as

$$\vec{u}^{\,j+1} = (\mathbf{I} + \mathbf{S})^{-1} \mathbf{D}^{-1} \mathbf{B} \, \vec{u}^{\,j} \; . \qquad (3.71)$$

$\mathbf{D}^{-1}$ is found by direct inversion, and the matrix $\mathbf{D}^{-1}\mathbf{B}$ is given by

$$\mathbf{D}^{-1}\mathbf{B} =
\begin{bmatrix}
\dfrac{e^{\omega \Delta t}}{1 + \Delta t \left[\omega - \dfrac{(\rho - \beta)}{\Lambda} \right]} & 0 & 0 & \cdots & 0 \\
0 & e^{-\lambda_1 \Delta t} & 0 & \cdots & 0 \\
0 & 0 & e^{-\lambda_2 \Delta t} & \cdots & 0 \\
\vdots & \vdots & \vdots & & \vdots \\
0 & 0 & 0 & \cdots & e^{-\lambda_6 \Delta t}
\end{bmatrix}$$

The exponential terms may be expanded in series expansions and the matrix $\mathbf{D}^{-1}\mathbf{B}$ is split into the form

$$\mathbf{D}^{-1}\mathbf{B} = \mathbf{G} + \Delta t \mathbf{H}$$

where

$$G \equiv \begin{bmatrix} \dfrac{1}{1 + \varDelta t\left[\omega - \frac{(\rho - \beta)}{\varLambda}\right]} & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

Here, H is bounded and G^n is bounded for positive n due to the inequality (3.69). By expanding $(I+S)^{-1}$ in the same way as $(I+R)^{-1}$ is expanded in equation (3.56), the amplification matrix becomes

$$A^{-1}B = (I + S)^{-1}(G + \varDelta tH)$$

$$= (I - S + S^2 - S^3 + ...)(G + \varDelta tH) \quad \text{for } \| S \| < 1$$

$$= G + \varDelta t(I - S + S^2 - S^3 + ...)H + (- S + S^2 - S^3 + ...)G \qquad (3.72)$$

Following the same argument for S that was made for R in section 3.3.2, the matrix T can be defined by

$$T \equiv \frac{1}{\varDelta t}S \ .$$

Then the norm of S is given by

$$\| S \| = \varDelta t \| T \| = b\varDelta t < 1 \qquad (3.73)$$

and the following relationships hold:

$$- S + S^2 - S^3 + ... = - \varDelta t T\left[I - S + S^2 - S^3 + ...\right] \qquad (3.74)$$

$$\left\| T\left[I - S + S^2 - S^3 + ...\right] \right\| \leq \frac{b}{1 - b\varDelta t}$$

$$\left\| I - S + S^2 - S^3 + ... \right\| \leq \frac{1}{1 - b\varDelta t}$$

Using (3.74) in (3.72), the amplification matrix becomes

$$A^{-1}B = G + \varDelta t\left[(I - S + S^2 - S^3 + ...)H - T(I - S + S^2 - S^3 + ...)G\right].$$

Since $(I-S+S^2-S^3+...)$, H, $T(I-S+S^2-S^3+...)$ and G are all bounded matrices, then the matrix $L(\Delta t)$ is defined as

$$L(\Delta t) \equiv (I - S + S^2 - S^3 + ...)H - T(I - S + S^2 - S^3 + ...)G$$

which is also a bounded matrix. Then the difference system of equation (3.54) may be written as

$$\vec{u}^{j+1} = \left[G + \varDelta tL(\Delta t)\right]\vec{u}^j \ . \qquad (3.75)$$

It has therefore been shown that, if the inequalities (3.68) and (3.69) are satisfied, then G^n is bounded for positive n and $L(\Delta t)$ is bounded. Therefore, under these conditions, the difference system (3.75) [and hence (3.54)] satisfies the conditions of the Kreiss–Strang Theorem (Theorem

3.1) and the difference system is stable.

$\square$

Remark: The bound of S is calculated from the spectral radius of $S^T S$, the same way as in equation (3.66) for **R**. The eigenvalues of $S^T S$ are

$$s_1 = s_2 = s_3 = s_4 = s_5 = 0$$

$$s_6 = \frac{\Delta t^2}{\left[1 + \Delta t\left(\omega - \frac{(\rho - \beta)}{\Lambda}\right)\right]^2} \sum_{i=1}^{6} \lambda_i^2$$

$$s_7 = \sum_{i-1}^{6} \left(\frac{\beta_i}{\Lambda}\right)^2 \left[\frac{1 - e^{-(\omega + \lambda_i)\Delta t}}{\omega + \lambda_i}\right]^2$$

Thus, the sufficient stability condition is inequality (3.69) combined with $\max(|s_6|, |s_7|) < 1$.

3.3.4 Von Neumann Necessary Condition for Stability

To apply the von Neumann stability criteria to the exponential transform discretization of the point kinetics equations, the matrix method[52] will be used. Equations (3.51) and (3.52) are rewritten as

$$\frac{(1 + \omega\Delta t)\mathbf{I}\vec{u}^{n+1} - e^{\omega\Delta t}\mathbf{I}\vec{u}^{n}}{\Delta t} = \mathbf{S}\,\vec{u}^{n+1} \tag{3.76}$$

where the matrix S is here defined as

$$\mathbf{S} \equiv \begin{bmatrix} \dfrac{(\rho - \beta)}{\Lambda} & \lambda_1 & \cdots & \lambda_6 \\[2ex] \dfrac{\beta_1}{\Lambda}\left[\dfrac{e^{-(\omega+\lambda_1)\Delta t} - 1}{\Delta t(\omega + \lambda_1)}\right] & \dfrac{1 + \omega\Delta t - e^{(\omega+\lambda_1)\Delta t}}{\Delta t} & \cdots & 0 \\[2ex] \vdots & \vdots & & \vdots \\[2ex] \dfrac{\beta_6}{\Lambda}\left[\dfrac{e^{-(\omega+\lambda_6)\Delta t} - 1}{\Delta t(\omega + \lambda_6)}\right] & 0 & \cdots & \dfrac{1 + \omega\Delta t - e^{(\omega+\lambda_6)\Delta t}}{\Delta t} \end{bmatrix} \tag{3.77}$$

Let the eigenvalues and eigenvectors of S be s_i and $\vec{\mu}_i$, respectively. The vectors $\vec{u}^n$ and $\vec{u}^{n+1}$ can be decomposed into linear combinations of the eigenvectors $\vec{\mu}_i$, in the form

$$\vec{u}^j = \sum_i b_i^j \vec{\mu}_i$$

and

$$\vec{u}^{j+1} = \sum_i b_i^{j+1} \vec{\mu}_i$$

Since

$$\mathbf{S}\vec{\mu}_i = s_i \vec{\mu}_i$$

the amplification of a single component of $\vec{u}$ may be written as

40

$$\frac{(1 + \omega \Delta t) b_i^{n+1} - e^{\omega \Delta t} b_i^n}{\Delta t} = s_i b_i^{n+1}$$

$$\Rightarrow b_i^{n+1} = \frac{e^{\omega \Delta t}}{1 + \Delta t(\omega - s_i)} b_i^n \tag{3.78}$$

If $(\omega - s_i) > 0$ then the amplification factor satisfies.

$$\left| \frac{b_i^{n+1}}{b_i^n} \right| \leq 1 + \mathcal{O}(\Delta t).$$

If $(\omega - s_i) < 0$, then provided that $|\Delta t(\omega - s_i)| < 1$, the denominator of the right hand side of (3.78) can be expanded in a Taylor series. Then,

$$\left| \frac{b_i^{n+1}}{b_i^n} \right| \leq 1 + s_i \Delta t + \mathcal{O}(\Delta t^2)$$

$$\leq 1 + \mathcal{O}(\Delta t) \tag{3.79}$$

This is equivalent to the von Neumann stability criterion, Definition 3.2. The necessary stability criteria are therefore summarized as

$$\Delta t \leq \frac{1}{|\omega - s_i|} \quad \text{for} \quad (\omega - s_i) < 0 \tag{3.80}$$

$$0 \leq \Delta t \leq \infty \quad \text{for} \quad (\omega - s_i) \geq 0$$

Note that the eigenvalues of S are most practically found through numerical solution methods. The condition (3.80) must obviously apply for all eigenvalues.

3.3.5 Comparison of stability criteria versus numerical experiments

In order to verify the stability criteria derived in sections 3.3.1–3.3.3, the stability boundaries predicted in these sections are compared against some numerical experiments. In all cases, the problem is solved using the precursor coefficients shown in Table 3.1, a delayed neutron lifetime of $\Lambda = 2.345\text{E}{-}5$ s, and variable reactivity. These values are typical of a PWR.

i	β_i	λ_i (s^{-1})
1	3.802	2.060E–4
2	1.4005	8.070E–4
3	0.3268	2.085E–3
4	0.1241	9.990E–4
5	0.0315	1.119E–3
6	0.0128	1.620E–4

Table 3.1: Point Kinetics Coefficients

The stability boundaries using the first sufficient condition (Theorem 3.2, section 3.3.2), the second sufficient condition (Theorem 3.3, section 3.3.3) and the von Neumann necessary condition (section 3.3.4) are shown in Figure 3.1. Time step sizes to the left of the curves are predicted by the respective criteria to be stable. Numerical tests were performed by using

equations (3.51) and (3.52) to integrate the point kinetics equations over a finite time interval. The initial condition was a steady state condition, found with $\rho=0.0$ and $P=1.0$. The transients were initiated by using an instantaneous insertion of reactivity at time $t=0.0$ s, and time steps from $\Delta t=0.01$ s to $\Delta t=10.0$ s were used to integrate the equations (3.51) and (3.52).

Figure 3.1: Maximum time step size to satisfy stability criteria.

In analyzing the numerical solutions, it can be difficult to identify when instability effects begin to occur. In contrast to the behaviour of unstable schemes for hyperbolic problems, the difference between stability effects and accuracy effects are difficult to identify with this discretization of the point kinetics equations. The criterion chosen here was that when the solution of P at the end time differed by more than 50% from the solution using the next smallest time step, then the scheme was deemed unstable. As can be seen from Figure 3.1, the von Neumann necessary conditions for stability agree well with the stability regime which was found numerically. The sufficient stability conditions are seen to be too conservative, and would lead to excessive computation time if implemented in a production code.

3.4 Local Mode Analysis for the 1-D and 3-D NEM

In this section, stability conditions for the one and three dimensional forms of the NEM are derived. Section 3.3 demonstrated that although sufficient stability conditions can be derived for the exponential transform discretization of the point kinetics equations, these conditions yield admissible time steps which are often too small to be of practical value. It was also shown that the von Neumann necessary conditions are adequate for predicting an admissible time step size. This is also the experience found by other authors in a wide spectrum of examples.[52] For these

42

reasons, and for the fact that the calculation of sufficient conditions would be far too compute–intensive, only the von Neumann stability conditions are derived for the NEM discretization of the time dependent neutron diffusion and delayed precursor equations.

The time dependent NEM–M2 equations are a coupled system of *nonlinear* equations. The nonlinearities come into play not only from the dynamic frequencies in the discretization of the time derivative, but also from the higher order moment equations, which are derived through the use of several nonlinear approximations. The higher order flux moment coefficients, a_{3gu} and a_{4gu}, of the NEM–M2 discretization can be treated as corrections to the outgoing current equations of the NEM–M0 discretization. In the NEM–M2 formulation, the current equations take the form of

$$
j_{gul}^{-m} = C_{1gu}^{L}\left(\phi_{g}^{m,n} + a_{4gu}^{m,n}\right) + C_{2gu}^{L}j_{gul}^{+m} + C_{3gu}^{L}j_{gur}^{-m} - C_{4gu}^{L}a_{3gu}^{m,n}
$$

$$
j_{gul}^{+m} = C_{1gu}^{R}\left(\phi_{g}^{m,n} + a_{4gu}^{m,n}\right) + C_{3gu}^{R}j_{gul}^{+m} + C_{2gu}^{R}j_{gur}^{-m} + C_{4gu}^{R}a_{3gu}^{m,n}
$$

(3.81)

whereas in the NEM–M0 formulation, they take the form of

$$
j_{gul}^{-m} = C_{1gu}^{L}\phi_{g}^{m,n} + C_{2gu}^{L}j_{gul}^{+m} + C_{3gu}^{L}j_{gur}^{-m}
$$

$$
j_{gur}^{+m} = C_{1gu}^{R}\phi_{g}^{m,n} + C_{3gu}^{R}j_{gul}^{+m} + C_{2gu}^{R}j_{gur}^{-m}
$$

(3.82)

The approximation will be made here that the coefficients terms a_{3gu} and a_{4gu} are such that they can be considered as small corrections to (3.82), and that these corrections are linearly dependent on the nodal averaged flux. This approximation enables consideration of the moments equations to be left out of the stability analysis, and the outgoing current equations (3.81) may be written as

$$
j_{gul}^{-m} = C_{1gu}^{L\prime}\phi_{g}^{m,n} + C_{2gu}^{L}j_{gul}^{+m} + C_{3gu}^{L}j_{gur}^{-m}
$$

$$
j_{gur}^{+m} = C_{1gu}^{R\prime}\phi_{g}^{m,n} + C_{3gu}^{R}j_{gul}^{+m} + C_{2gu}^{R}j_{gur}^{-m}
$$

(3.83)

with

$$
C_{1gu}^{L\prime} \equiv \frac{C_{1gu}^{L}\left(\phi_{g}^{m,n} + a_{4gu}^{m,n}\right) - C_{4gu}^{L}a_{3gu}^{m,n}}{\phi_{g}^{m,n}}
$$

$$
C_{1gu}^{R\prime} \equiv \frac{C_{1gu}^{R}\left(\phi_{g}^{m,n} + a_{4gu}^{m,n}\right) + C_{4gu}^{R}a_{3gu}^{m,n}}{\phi_{g}^{m,n}}
$$

(3.84)

For the stability analysis, the approximation is made that the coefficients $C_{1gu}^{L\prime}$ and $C_{1gu}^{R\prime}$ are constant in time. Furthermore, as in section 3.3, the approximation is made that the dynamic frequencies are constant in time. These approximations linearize the NEM equations to the form of the NEM–M0 approximation, which is written as equations (3.83) along with

$$\frac{(1 + \omega\Delta t)\phi_g^{m,n} - e^{\omega\Delta t}\phi_g^{m,n-1}}{v_g\Delta t} + \Sigma_{rg}\phi_g^{m,n} = \sum_{g<g'}\Sigma_{gg'}\phi_{g'}^{m,n}$$

$$+ \frac{1}{\lambda}\sum_{g'=1}^{G}\sum_{j=1}^{G}(1 - \beta^j)\chi_{pg}^j\nu\Sigma_{fg'}^j\phi_{g'}^{m,n} + \sum_i\chi_{dg}^i\lambda_i c_i^{m,n} + \sum_u\frac{1}{a_u}\left[j_{gul}^{+m} - j_{gul}^{-m} + j_{gur}^{-m} - j_{gur}^{+m}\right]$$

$$(3.85)$$

and

$$c_i^{m,n} = c_i^{m,n-1}e^{-\lambda_i\Delta t} + \frac{1}{\lambda}\sum_{g'}\sum_j\beta_i^j\nu\Sigma_{fg'}^j\phi_{g'}^m\left(\frac{1 - e^{-(\omega+\lambda_i)\Delta t}}{\omega + \lambda_i}\right) \qquad (3.86)$$

Now the procedure of *local mode analysis* is applied. This method approximates that all coefficients are not spatially varying, and ignores the effects of boundary conditions. Despite these approximations, it is the most practical method of finding stability criteria for space dependent problems. The dependent variables are expanded as Fourier series:

$$\phi_g^{m,n} = \sum_k \tilde{\psi}_{kg}^n e^{i\vec{\theta}_k \cdot \vec{\kappa}_m} \qquad\qquad c_i^{m,n} = \sum_k \tilde{\xi}_{ki}^n e^{i\vec{\theta}_k \cdot \vec{\kappa}_m}$$

$$j_{gul}^{-m} = \sum_k \tilde{\chi}_{kgLu} e^{i\vec{\theta}_k \cdot \vec{\kappa}_m} \qquad\qquad j_{gur}^{+m} = \sum_k \tilde{\chi}_{kgRu} e^{i\vec{\theta}_k \cdot \vec{\kappa}_m} \qquad (3.87)$$

$$j_{gul}^{+m} = \sum_k \tilde{\chi}_{kgLu} e^{i\vec{\theta}_k \cdot \vec{\kappa}_m}\left(e^{-i\theta_{ku}}\right) \qquad j_{gur}^{-m} = \sum_k \tilde{\chi}_{kgRu} e^{i\vec{\theta}_k \cdot \vec{\kappa}_m}\left(e^{i\theta_{ku}}\right)$$

Here, the summation is over all Fourier modes and the $\tilde{\psi}$, $\tilde{\chi}$, and $\tilde{\xi}$ coefficients are the amplitudes of the respective modes k. $\vec{\theta}_k$ is the vector of phase angles for the mode: $\vec{\theta}_k$ ranges from $(-\pi, -\pi, -\pi)$ to (π, π, π) for the 3–D NEM, with discrete steps of $(\pi/N_x, 0, 0)$, $(0, \pi/N_y, 0)$ and $(0, 0, \pi/N_z)$. $\vec{\kappa}_m$ is the position vector of the node, $\vec{\kappa}_m = (i, j, k)$. Further clarification of the method of local mode analysis may be found in Hirsch.[52]

The following terms are now defined to simplify the development:

$$\Sigma_{Fgg'} \equiv \Sigma_{gg'} + \frac{1}{\lambda}\sum_j(1 - \beta^j)\chi_{pg}^j\nu\Sigma_{fg'}^j$$

$$\Sigma_{Cig} \equiv \frac{1}{\lambda}\sum_j\beta_i^j\nu\Sigma_{fg}^j\left(\frac{1 - e^{-(\omega+\lambda_i)\Delta t}}{\omega + \lambda_i}\right)$$

The Fourier expansions (3.87) are substituted into the neutron and precursor balance equations [(3.88) and (3.85)]. With the utilization of the orthogonality property of the Fourier modes, the amplitude equations for one mode (omitting the subscript of the mode, k) may be written as

$$\frac{(1 + \omega\Delta t)\tilde{\psi}_g^n - e^{\omega\Delta t}\tilde{\psi}_g^{n-1}}{v_g\Delta t} + \Sigma_{rg}\tilde{\psi}_g^n = \sum_{g'}\Sigma_{Fgg'}\tilde{\psi}_{g'}^n + \sum_i\chi_{dg}^i\lambda_i\tilde{\xi}_i^n$$

$$+ \sum_u\frac{1}{a_u}\left[\tilde{\chi}_{gLu}^n(e^{-i\theta_u} - 1) + \tilde{\chi}_{gRu}^n(e^{i\theta_u} - 1)\right] \qquad (3.89)$$

44

and

$$\tilde{\xi}_i^n = \tilde{\xi}_i^{n-1} e^{-\lambda_i \Delta t} + \sum_g \Sigma_{Cig} \tilde{\psi}_g^n \qquad (3.90)$$

Similarly, the partial current equations (3.83) may be solved for the amplitude of the current modes in terms of the amplitude of the flux modes:

$$\tilde{\chi}_{gLu} = \left\{ \frac{C_{1gu}^{L\prime}\left(1 - C_{3gu}^R e^{-i\theta_u}\right) + C_{1gu}^{R\prime} C_{2gu}^R e^{i\theta_u}}{1 - \left(C_{3gu}^L e^{i\theta_u} + C_{3gu}^R e^{-i\theta_u}\right) + C_{3gu}^L C_{3gu}^R - C_{2gu}^L C_{2gu}^R} \right\} \tilde{\psi}_g^n$$

$$\equiv R_{gLu}(\theta_u)\tilde{\psi}_g^n$$

$$\tilde{\chi}_{gRu} = \left\{ \frac{C_{1gu}^{L\prime} C_{2gu}^L e^{-i\theta_u} + C_{1gu}^{R\prime}\left(1 - C_{3gu}^L e^{i\theta_u}\right)}{1 - \left(C_{3gu}^L e^{i\theta_u} + C_{3gu}^R e^{-i\theta_u}\right) + C_{3gu}^L C_{3gu}^R - C_{2gu}^L C_{2gu}^R} \right\} \tilde{\psi}_g^n$$

$$\equiv R_{gRu}(\theta_u)\tilde{\psi}_g^n$$

These expressions may be substituted into the balance equation. With further manipulations the nodal balance and precursor equations may be written as

$$\frac{(1 + \omega\Delta t)\tilde{\psi}_g^n - e^{\omega z\Delta t}\tilde{\psi}_g^{n-1}}{v_g\Delta t} + \Sigma_{rg}\tilde{\psi}_g^n = \sum_{g\prime}\Sigma_{Fgg\prime}\tilde{\psi}_{g\prime}^n + \sum_i \chi_{dg}^i \lambda_i \tilde{\xi}_i^n + \sum_u L_{gu}(\theta_u)\tilde{\psi}_g^n \qquad (3.91)$$

and

$$\frac{(1 + \omega\Delta t)\tilde{\xi}_i^n - e^{\omega\Delta t}\tilde{\xi}_i^{n-1}}{\Delta t} = \frac{1 + \omega\Delta t - e^{(\omega + \lambda_i)\Delta t}}{\Delta t}\tilde{\xi}_i^n + \sum_g \frac{e^{(\omega + \lambda_i)\Delta t}\Sigma_{Cig}}{\Delta t}\tilde{\psi}_g^n \qquad (3.92)$$

with the definition

$$L_{gu}(\theta_u) \equiv \frac{1}{a_u}\left[R_{gLu}(\theta_u)(e^{-i\theta_u} - 1) + R_{gRu}(\theta_u)(e^{i\theta_u} - 1) \right]$$

Most PANBOX data bases for PWR analysis use two energy groups and six precursor groups. For these cases, the vector of mode amplitudes is defined as

$$\vec{u}^n = \left[\tilde{\psi}_1^n, \tilde{\psi}_2^n, \tilde{\xi}_1^n, \tilde{\xi}_2^n, ..., \tilde{\xi}_6^n \right]^T$$

and equations (3.91) and (3.92) may be written as

$$\frac{(1 + \omega\Delta t)\vec{u}^n - e^{\omega\Delta t}\vec{u}^{n-1}}{\Delta t} = S\vec{u}^n \qquad (3.93)$$

where the matrix S is defined by

45

$$S \equiv \begin{bmatrix} a_{11} & a_{12} & v_1\lambda_1 & v_1\lambda_2 & \cdots & v_1\lambda_6 \\ a_{21} & a_{22} & 0 & 0 & \cdots & 0 \\ b_{11} & b_{12} & c_1 & 0 & \cdots & 0 \\ b_{21} & b_{22} & 0 & c_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ b_{61} & b_{62} & 0 & 0 & \cdots & c_6 \end{bmatrix}$$

$$a_{11} \equiv v_1\left[\Sigma_{F11} + \sum_u L_{1u}(\theta_u) - \Sigma_{r1}\right] \qquad a_{12} \equiv v_1\Sigma_{12}$$

$$a_{21} \equiv v_2\Sigma_{21} \qquad\qquad a_{22} \equiv v_2\left[\sum_u L_{2u}(\theta_u) - \Sigma_{r2}\right]$$

$$b_{ig} \equiv \frac{e^{(\omega+\lambda_i)\Delta t}\Sigma_{Cig}}{\Delta t} \qquad\qquad c_i \equiv \left[\frac{1 + \omega\Delta t - e^{(\omega+\lambda_i)\Delta t}}{\Delta t}\right]$$

The procedure shown in section 3.3.4 is now applied to equation (3.93). The eigenvectors and eigenvalues of S are given as s_i and $\vec{\mu}_i$, respectively. The vectors $\vec{u}^n$ and $\vec{u}^{n-1}$ can be decomposed into linear combinations of the eigenvectors $\vec{\mu}_i$, in the form

$$\vec{u}^j = \sum_i d_i^j \vec{\mu}_i$$

and the amplification of the eigenmodes are given by

$$d_i^j = \frac{e^{\omega\Delta t}}{1 + \Delta t(\omega - s_i)} d_i^{j-1} \tag{3.94}$$

The stability criteria are derived as in section 3.3.4, and are summarized as

$$\Delta t \le \frac{1}{|\omega - s_i|} \quad \text{for} \quad (\omega - s_i) < 0$$
$$0 \le \Delta t \le \infty \quad \text{for} \quad (\omega - s_i) \ge 0 \tag{3.95}$$

These criteria are calculated for each material region of the data base. Because of the different burnup histories and thermalhydraulic conditions of materials within the core, this usually means that the stability criteria must be evaluated for each coarse mesh node. For a full core calculation, calculation of the stability criteria involves finding the eigenvalues of several thousand 7x7 matrices. To save computation time, stability criteria for all the nodes is evaluated only at the first time step. The least stable nodes are then identified as the ones for which the criteria (3.95) admit the smallest time step. During the transient calculation, the stability criteria is recalculated only for these least stable nodes.

46

3.5 Summary

In this chapter, some theoretical foundations for the numerical methods used in this thesis have been set. In section 3.1 it was shown that the NEM is consistent with the multigroup neutron diffusion equation. Section 3.2 introduced stability analysis and time integration techniques used in PANBOX. Section 3.3 derived three separate stability criteria for the PANBOX time integration method applied to the point kinetics equations, and compared these criteria to numerical experiments. From the results of these experiments, it was concluded that the necessary stability criteria are appropriate for practical use. Section 3.4 builds on the analysis of section 3.3 to derive necessary stability criteria for the NEM by using local mode analysis. Because the numerical methods are stable and consistent, it is expected that the solutions which will be calculated are also convergent to the solutions of the PDEs which are being modelled. Furthermore, the consistency of the NEM is used in the next chapter to develop an operator formulation of perturbation theory.

4. Point Kinetics Model

The point kinetics equations, already presented in section 3.3, are well known as the simplest kinetics model of a nuclear reactor. The derivation of the point kinetics equations in this chapter is similar to that of Henry[9] who seems to be the first person to have derived the point kinetics equations by separating the neutron flux into a shape function and an amplitude function. Henry's formulation was later used by Ott[14] to formulate the quasi–static approximation, which has proven to be an efficient and accurate method for approximating the solution to the time–dependent neutron diffusion equation.[15,35,36,80] While some variant forms of the point kinetics equations have been proposed (see for example, Becker[81]), the advantages of these methods are not necessarily relevant to this thesis. For this reason, the classical textbook formulation of the point kinetics equations will be derived.[10,11,82] The derivation is included in section 4.1, and not in the appendix, for reasons of continuity.

During many accident scenarios, the reactivity of the core is the greatest varying parameter in the point kinetics model. Accurate determination of the core reactivity is crucial to achieve an accurate description of the transient. It is well known that first order perturbation[8,83] or sensitivity theory[53,54] can be used to make reasonably accurate approximations of the reactivity. What was not known, until recently, were the technical details of how perturbation theory approximations can be implemented in a code utilizing the NEM. This is the main technical contribution of this chapter, and will be presented in section 4.2.

4.1 Derivation of the Point Kinetics Equations

Derivation of the point kinetics equations begins with the space–time continuous multigroup neutron diffusion equations, coupled with the delayed neutron precursor equations:

$$\frac{1}{v_g}\frac{\partial \phi_g}{\partial t} - \nabla \cdot D_g \nabla \phi_g + \Sigma_{rg}\phi_g = \sum_{g'=1}^{N_G} \Sigma_{gg'}\phi_{g'} + \sum_{g'=1}^{N_G} \Sigma_{pgg'}\phi_{g'} + \sum_{i=1}^{N_I} \chi_{dg}^i \lambda_i c_i, \qquad (4.1)$$

$$g = 1,...,N_G$$

$$\frac{\partial c_i}{\partial t} = \frac{1}{\lambda}\sum_{g'=1}^{N_G}\sum_{j=1}^{N_I} \beta_i^j \nu \Sigma_{fg'}^j \phi_{g'} - \lambda_i c_i, \quad i = 1,..,N_I \qquad (4.2)$$

where

$$\phi_g \equiv \phi_g(\vec{r},t) \qquad g = 1,..,N_G$$
$$c_i \equiv c_i(\vec{r},t) \qquad i = 1,..,N_I$$

and

$$\Sigma_{pgg'} \equiv \frac{1}{\lambda}\sum_{j=1}^{N_I} \chi_{pg}^j (1 - \beta^j)\nu\Sigma_{fg'}^j$$

49

$$\beta^j \equiv \sum_{i=1}^{N_I} \beta^j_i$$

The flux is separated into a time dependent amplitude function P and a time and space dependent shape function ψ_g^{PK}

$$\phi_g(\vec{r},t) = \psi_g^{PK}(\vec{r},t)P(t) \tag{4.3}$$

and the neutron precursor concentrations are separated in a similar way,

$$c_i(\vec{r},t) = \xi_i^{PK}(\vec{r},t)C_i^{PK}(t) \tag{4.4}$$

Equations (4.3) and (4.4) are substituted into (4.1). The neutron diffusion equation is then multiplied by a weight function (see the next section for the choice of a weight function) W_g, and integrated over the volume of the core.

$$\frac{\partial}{\partial t}\left[P(t)\int_V \sum_{g=1}^{N_G} \frac{W_g(\vec{r})\psi_g^{PK}(\vec{r},t)}{v_g}dV\right] = \frac{[\rho(t)-\beta(t)]}{\Lambda(t)}P(t) + \sum_{i=1}^{N_I}\lambda_i C_i^{PK}(t)\int_V \sum_{g=1}^{N_G} W_g(\vec{r})\chi_{dg}^i\xi_i^{PK}(\vec{r},t)dV \tag{4.5}$$

where the reactivity is defined as

$$\rho(t) \equiv \frac{\int_V \sum_{g=1}^{N_G} W_g(\vec{r})\left[\nabla\cdot\nabla D_g\nabla\psi_g^{PK} - \Sigma_{rg}\psi_g^{PK} + \sum_{g'=1}^{N_G}\left[\Sigma_{gg'} + \Sigma_{pgg'} + \frac{1}{\lambda}\sum_{i=1}^{N_I}\sum_{j=1}^{N_J}\chi_{dg}^i\beta_i^j\nu\Sigma_{fg'}^j\right]\psi_{g'}^{PK}\right]}{\int_V \sum_{g=1}^{N_G} W_g(\vec{r})\sum_{g'=1}^{G}\left[\Sigma_{pgg'} + \frac{1}{\lambda}\sum_{i=1}^{N_I}\sum_{j=1}^{N_J}\chi_{dg}^i\beta_i^j\nu\Sigma_{fg'}^j\right]\psi_{g'}^{PK}} \tag{4.6}$$

the delayed neutron fractions are defined as

$$\beta_i(t) \equiv \frac{\int_V \sum_{g=1}^{N_G} W_g(\vec{r})\left[\frac{1}{\lambda}\sum_{g'=1}^{N_G}\sum_{j=1}^{N_J}\chi_{dg}^i\beta_i^j\nu\Sigma_{fg'}^j\psi_{g'}^{PK}\right]dV}{\int_V \sum_{g=1}^{N_G} W_g(\vec{r})\sum_{g'=1}^{N_G}\left[\Sigma_{pgg'} + \frac{1}{\lambda}\sum_{i=1}^{N_I}\sum_{j=1}^{N_J}\chi_{dg}^i\beta_i^j\nu\Sigma_{fg'}^j\right]\psi_{g'}^{PK}dV} \qquad \beta(t) = \sum_{i=1}^{I}\beta_i(t) \tag{4.7}$$

and the neutron lifetime are defined as

$$\Lambda(t) \equiv \left[\int_V \sum_{g=1}^{N_G} W_g(\vec{r})\sum_{g'=1}^{N_G}\left[\Sigma_{pgg'} + \frac{1}{\lambda}\sum_{i=1}^{N_I}\sum_{j=1}^{N_J}\chi_{dg}^i\beta_i^j\nu\Sigma_{fg'}^j\right]\psi_{g'}^{PK}dV\right]^{-1} \tag{4.8}$$

The separation of variables (4.3) and (4.4) admits non–unique solutions of P, ψ_g^{PK}, C_i^{PK}, and ξ_i^{PK}. In order to admit a unique solution, the following constraints are applied to the shape functions:

$$\int_V \sum_{g=1}^{N_G} \frac{W_g(\vec{r})\psi_g^{PK}(\vec{r}, t)}{v_g} dV = 1 \qquad (4.9)$$

$$\int_V \sum_{g=1}^{N_G} W_g(\vec{r})\chi_{dg}^i \xi_i^{PK}(\vec{r}, t) dV = 1 \qquad (4.10)$$

These constraints reduce equation (4.5) to

$$\frac{dP}{dt} = \frac{\rho(t) - \beta(t)}{\Lambda(t)} P(t) + \sum_{i=1}^{N_I} \lambda_i C_i^{PK}(t) \qquad (4.11)$$

which is the first equation of the point kinetics model. The precursor equations of the point kinetics model are derived by multiplying equation (4.2) by $W_g(\vec{r}, t)\chi_{dg}^i$, summing over the energy groups, and integrating over all space. The result yields

$$\frac{\partial}{\partial t}\left[C_i^{PK}(t) \int_V \sum_{g=1}^{N_G} W_g(\vec{r})\chi_{dg}^i \xi_i^{PK}(\vec{r}, t) dV \right] = \frac{\beta_i(t)}{\Lambda(t)} P(t) - \lambda_i C_i^{PK}(t) \int_V \sum_{g=1}^{N_G} W_g(\vec{r})\chi_{dg}^i \xi_i^{PK}(\vec{r}, t) dV$$

which, upon application of the constraint (4.10), becomes

$$\frac{dC_i^{PK}}{dt} = \frac{\beta_i(t)}{\Lambda(t)} P(t) - \lambda_i C_i^{PK}(t) \qquad (4.12)$$

Equations (4.11) and (4.12) are the well known point kinetics equations. They are ordinary differential equations which may be integrated in time, given initial conditions and the time–dependent coefficients ρ, β_i, λ_i and Λ. Initial conditions may be found from an initial three dimensional solution by multiplying (4.3) and (4.4) by $W_g(\vec{r})/v_g$ and $W_g(\vec{r})\chi_{dg}^i$ respectively, summing over all energy groups, and integrating of the volume of the core. Due to the normalizations (4.9) and (4.10) the initial conditions become,

$$P(t_0) = \int_V \sum_{g=1}^{N_G} \frac{W_g(\vec{r})\phi_g(\vec{r}, t_0)}{v_g} dV \qquad (4.13)$$

and

$$C_i^{PK}(t_0) = \int_V \sum_{g=1}^{N_G} W_g(\vec{r})\chi_{dg}^i c_i(\vec{r}, t_0) dV \qquad (4.14)$$

The point kinetics parameters are found by approximating the integrals in equations (4.6) to (4.8). The main approximation used in this calculation is the adiabatic quasi–static approximation, in which the flux shape function is approximated to be slowly varying in time, such that

$$\psi_g^{PK}(\vec{r}, t) \approx \psi_g^{PK}(\vec{r}, t_0) \qquad (4.15)$$

The weight function which is chosen is the solution of the equations adjoint to the time–independent neutron diffusion equation, so that the reactivity is determined using perturbation theory. The implementation of perturbation theory within the NEM is the subject of the next section.

4.2 Adjoint method of sensitivity theory for the NEM

There is been some confusion in the literature[55,56,57 58,59,60] about how perturbation or sensitivity theory formulations can be implemented in codes which use variants of the NEM. The authors of these papers have all distinguished between the so called *mathematical* and *physical* adjoint equations of the NEM, terms which were posed by Lawrence in 1984.[55] The *physical* adjoint equations were deemed as the equations adjoint to the differential multigroup diffusion equations, discretized with the NEM. The *mathematical* adjoint equations was the name given to the adjoint of the discretized NEM equations.

The terminology *physical* and *mathematical* adjoint is misleading, because all equations are inherently *mathematical*. The problem which confronted Lawrence was in fact first solved in 1980 by Cacuci et. al.,[53] when they observed that in general, two different classes of adjoint equations could be formulated from a set of discretized equations. They defined two different formulations: the *operator formulation* is based on the discretization of the equation adjoint to the forward differential equation; and the *matrix formulation* is based on the adjoint of the discretized equations. For the example of the thermalhydraulic equations studied by Cacuci et. al.,[53] both adjoint formulations were consistent with the continuous variable adjoint equation, however the formulations did have different truncation errors.

Although in some cases it is favourable to use the matrix formulation, some serious difficulties have arisen with this formulation applied to the NEM. The first difficulty was that it took nine years from Lawrence's identification of the matrix formulation[55] before someone found a convergent iterative solution scheme for his equations.[59,60] Further problems occur because there are many variants of the NEM: the original NEM–M2B2 variant developed by Finnemann[23] has its own characteristics which are not addressed in the solution scheme for the variant developed by Yang, Taiwo and Khalil.[59,60] The main advantage of the operator formalism, on the other hand, is that the existing NEM solution scheme may be applied *without* having to wait nine years before it converges. None of the earlier works[55,56,57 58,59,60] which developed a matrix perturbation theory formulation for the NEM properly considered the possibility of using the operator formulation. Most of these authors recognized the fact that the perturbation theory formulas involve dot products of flux gradients integrated over the nodes, and that the use of the nodal averaged fluxes to approximate these gradients would be too inaccurate.[55,56,60] However, what previous authors did not recognize is that the NEM solution provides not only average nodal fluxes, but also *flux expansions* within the node. It was shown in chapter 3 that these flux expansions converge to the solution of the multigroup neutron diffusion equations as the mesh spacing reduces to zero. Therefore, they *can* be used in an operator theory formulation.

It must be noted that a very similar formulation to the one presented here has been recently published in the doctoral dissertation of Delmolino.[61] Delmolino implies that the use of flux expansions is necessary only in the implementation of generalized perturbation theory, and he uses simple inner products of the nodal averaged flux to determine perturbations in the reactivity. However, he shows no results for perturbations in the diffusion coefficient: it is this type of perturbation in which the accuracy of his proposed method breaks down. Delmolino also does not identify how the inner products of the gradients are calculated in his formulation. Additionally, he does not make the important theoretical connection that the flux expansions converge to the solution of the multigroup neutron diffusion equation. Without this consistency consideration, blind use of the operator formalism can be erroneous. The methods presented in the next section were developed independently of Delmolino's findings, and the differences described above distinguish this work from his.

4.2.1 Operator Formulation for the NEM

Here, the eigenvalue perturbation expression is derived from the differential multigroup forward and adjoint equations. As suggested by Delmolino's results,[61] the methodology may also be used with more general perturbation and sensitivity theory analyses. The initial, unperturbed forward multigroup neutron diffusion equation is written in compact form as

$$\nabla \cdot D_{g0} \nabla \phi_{g0} - A_{g0}\vec{\phi}_0 + \frac{1}{\lambda_0}F_{g0}\vec{\phi}_0 = 0, \quad g = 1,...,N_G. \tag{4.16}$$

Here, the vector of multigroup fluxes is

$$\vec{\phi}_0 \equiv (\phi_{10}, \phi_{20}, ..., \phi_{G0})^T \tag{4.17}$$

and the operators A_g and F_g are given as

$$A_{g0}\vec{\phi}_0 \equiv \Sigma_{rg0}\phi_{g0} - \sum_{g'=1, g' \neq g}^{N_G} \Sigma_{gg'0}\phi_{g'0} \tag{4.18}$$

and

$$F_{g0}\vec{\phi}_0 \equiv \chi_g \sum_{g'=1}^{N_G} \nu\Sigma_{fg'0}\phi_{g'0} \tag{4.19}$$

The perturbed forward equation is

$$\nabla \cdot D_g \nabla \phi_g - A_g\vec{\phi} + \frac{1}{\lambda}F_g\vec{\phi} = 0, \quad g = 1,...,N_G. \tag{4.20}$$

The perturbations are thus defined as

$$\delta\phi_g \equiv \phi_g - \phi_{g0}, \qquad \delta A_g \equiv A_g - A_{g0},$$

$$\delta D_g \equiv D_g - D_{g0}, \qquad \delta F_g \equiv F_g - F_{g0}, \tag{4.21}$$

$$\delta\left(\frac{1}{\lambda}\right) \equiv \frac{1}{\lambda} - \frac{1}{\lambda_0}.$$

The unperturbed adjoint equation is

$$\nabla \cdot D_{g0}^{*}\nabla\phi_{g0}^{*} - A_{g0}^{*}\vec{\phi}_{0}^{*} + \frac{1}{\lambda_{0}}F_{g0}^{*}\vec{\phi}_{0}^{*} = 0, \quad g = 1,...,N_{G}; \tag{4.22}$$

where

$$\vec{\phi}_{0}^{*} \equiv (\phi_{10}^{*}, \phi_{20}^{*}, ..., \phi_{N_{G}0}^{*})^{T}$$

$$A_{g0}^{*}\vec{\phi}_{0}^{*} \equiv \Sigma_{rg0}\phi_{g0}^{*} - \sum_{g'=1, g'\neq g}^{N_{G}} \Sigma_{g'g0}\phi_{g'0}^{*}$$

$$F_{g0}^{*}\vec{\phi}_{0}^{*} \equiv \nu_{g}\Sigma_{fg0}\sum_{g'=1}^{N_{G}} \chi_{g'}\phi_{g'0}^{*}$$

An exact expression for the perturbation in the eigenvalue may be obtained by multiplying equation (4.20) by ϕ_{g}^{*}, summing over all groups g, and integrating over the volume of the core. The result is

$$\delta\left(\frac{1}{\lambda}\right) = \frac{-\int_{V} \sum_{g=1}^{N_{G}} \phi_{g0}^{*}\left[\nabla \cdot D_{g}\nabla\phi_{g} - A_{g}\vec{\phi} + \frac{1}{\lambda_{0}}F_{g}\vec{\phi}\right]dV}{\int_{V} \sum_{g=1}^{N_{G}} \phi_{g0}^{*}F_{g}\vec{\phi}\,dV}. \tag{4.23}$$

The adjoint equation (4.22) is now multiplied by ϕ_{g}, summed over all groups g, and integrated over volume. The right hand side of the resulting equation is still equal to zero, and thus may be added to the numerator of equation (4.23) without any loss of generality. Using the definitions of the operators in (4.21), the exact expression for the perturbation becomes

$$\delta\left(\frac{1}{\lambda}\right) = \frac{-\int_{V} \sum_{g=1}^{N_{G}}\left[\phi_{g0}^{*}\nabla \cdot D_{g}\nabla\phi_{g} - \phi_{g}\nabla \cdot D_{g0}\nabla\phi_{g0}^{*} - \phi_{g0}^{*}\delta A_{g}\vec{\phi} + \frac{1}{\lambda}\phi_{g0}^{*}\delta F_{g}\vec{\phi}\right]dV}{\int_{V} \sum_{g=1}^{N_{G}} \phi_{g0}^{*}F_{g}\vec{\phi}\,dV}. \tag{4.24}$$

Using the vector relation

$$f\nabla \cdot \vec{g} = \nabla \cdot (f\vec{g}) - (\nabla f) \cdot \vec{g} ,$$

this becomes

$$\delta\left(\frac{1}{\lambda}\right) = \frac{-\int_{V} \sum_{g=1}^{N_{G}}\left[-\delta D_{g}\nabla\phi_{g0}^{*} \cdot \nabla\phi_{g} - \phi_{g0}^{*}\delta A_{g}\vec{\phi} + \frac{1}{\lambda}\phi_{g0}^{*}\delta F_{g}\vec{\phi}\right]dV}{\int_{V} \sum_{g=1}^{N_{G}} \phi_{g0}^{*}F_{g}\vec{\phi}\,dV} - \Gamma(\vec{\phi}_{g}^{*},\vec{\phi}_{g}) \tag{4.25}$$

where

$$\Gamma(\vec{\phi}_g^*, \vec{\phi}_g) \equiv \frac{\displaystyle\int_{\partial\Omega} \sum_{g=1}^{N_G} \hat{n}_s \cdot \left[\phi_{g0}^* D_g \nabla\phi_g - \phi_g D_{g0}\nabla\phi_{g0}^*\right] dA}{\displaystyle\int_V \sum_{g=1}^{N_G} \phi_{g0}^* F_g \vec{\phi} \, dV} \tag{4.26}$$

The first order accurate perturbation expression may be written as

$$\delta\left(\frac{1}{\lambda}\right) \approx \frac{-\displaystyle\int_V \sum_{g=1}^{N_G}\left[-\delta D_g \nabla\phi_{g0}^* \cdot \nabla\phi_{g0} - \phi_{g0}^*\delta A_g\vec{\phi}_0 + \frac{1}{\lambda}\phi_{g0}^*\delta F_g\vec{\phi}_0\right] dV}{\displaystyle\int_V \sum_{g=1}^{N_G} \phi_{g0}^* F_{g0}\vec{\phi}_0 \, dV} - \Gamma(\vec{\phi}_{g0}^*, \vec{\phi}_{g0}) \tag{4.27}$$

The first term of equation (4.27) is the commonly used perturbation theory expression which can be found in most textbooks on reactor core analysis.[8,10,82] The second term, Γ, is an integration over the boundary of the volume being considered, *and over all interface surfaces on which the fluxes or currents are discontinuous*. While the true solution to the PDE does not have such discontinuities, an approximation to the solution may possess such discontinuities. In the operator formulation, these possible discontinuities must be considered.

The first term of equation (4.27) involves the inner product of forward and adjoint flux gradients, and the inner product of forward and adjoint fluxes. It is assumed that all coefficients do not vary over the volume of a coarse mesh node. This assumption neglects higher order burnup[84] or control rod corrections,[24] where the cross sections are approximated as polynomials or piecewise constant functions within the node. The integrals over the entire reactor volume in equation (4.27) may be represented as sums of integrals over nodal volumes. For example,

$$\int_V \sum_{g=1}^{N_G} \phi_{g0}^*\delta A_g\vec{\phi}_0 \, dV = \sum_{m=1}^{N} \sum_{g=1}^{N_G}\left[\delta\Sigma_{rg}^m \int_{\Omega^m} \phi_{g0}^*\phi_{g0} \, dV - \sum_{g'=1}^{N_G} \delta\Sigma_{gg'}^m \int_{\Omega^m} \phi_{g0}^*\phi_{g'0} \, dV\right], \tag{4.28}$$

where the superscript m denotes the node index, and N is the total number of nodes in the discretization of the core. Similarly, the expression for the δF operator is

$$\int_V \sum_{g=1}^{N_G} \phi_{g0}^*\delta F_g\vec{\phi}_0 \, dV = \sum_{m=1}^{N_T} \sum_{g=1}^{N_G} \sum_{g'=1}^{N_G} \delta\left(\chi_g \nu\Sigma_{fg'}\right)^m \int_{\Omega^m} \phi_{g0}^*\phi_{g'0} \, dV \tag{4.29}$$

and the expression for the F operator has a form similar to (4.29). From equations (4.28) and (4.29) it is seen that the inner products of the forward and adjoint fluxes must be evaluated over nodal volumes. Additionally, because of the first term in the numerator of (4.27), the inner products of the forward and adjoint flux gradients must also be evaluated over the nodal volumes. These integrals can all be approximated by expanding the nodal flux in a series expansion, using the one–dimensional flux expansions of the NEM. In Cartesian geometry, the nodal flux then

takes the form

$$\phi_{g0}\big|_{\Omega^m} \approx \Phi_g^m(x,y,z) = \Psi_{gx}^m\left(\frac{x - x_0^m}{a_x}\right) + \Psi_{gy}^m\left(\frac{y - y_0^m}{a_y}\right) + \Psi_{gz}^m\left(\frac{z - z_0^m}{a_z}\right) - 2\phi_g^m \quad (4.30)$$

for the forward flux, and

$$\phi_{g0}^*\big|_{\Omega^m} \approx \Phi_g^{*m}(x,y,z) = \Psi_{gx}^{*m}\left(\frac{x - x_0^m}{a_x}\right) + \Psi_{gy}^{*m}\left(\frac{y - y_0^m}{a_y}\right) + \Psi_{gz}^{*m}\left(\frac{z - z_0^m}{a_z}\right) - 2\phi_g^{*m} \quad (4.31)$$

for the adjoint flux. The functions Ψ are the one–dimensional transverse integrated flux expansions of the form

$$\Psi_{gu}^m(u) = \phi_{g0}^m + \sum_{i=1}^{4} a_{igu}^m h_i(u) \quad (4.32)$$

for the forward expansion functions, and

$$\Psi_{gu}^{*m}(u) = \phi_{g0}^{*m} + \sum_{i=1}^{4} a_{igu}^{*m} h_i(u) \quad (4.33)$$

for the adjoint expansion functions. The NEM polynomials h_i are given in Appendix B. It was shown in chapter 3 that when the functions Ψ are determined with the NEM, then the flux expansions of equations (4.30) and (4.31) reduce to the solution of the forward and adjoint multigroup diffusion equations, as the nodal mesh spacing is reduced to zero. These expansions are therefore consistent with the solution to the continuous equations used to derive the perturbation expression of equation (4.27). If the expansions are used to evaluate the integrals in (4.27), then the result of the integration will converge to the true result of (4.27) as the mesh spacing is reduced to zero. The gradients of (4.30) and (4.31) are evaluated to be

$$\nabla \Phi_g^m = \frac{\partial \Psi_{gx}^m}{\partial x}\hat{x} + \frac{\partial \Psi_{gy}^m}{\partial y}\hat{y} + \frac{\partial \Psi_{gz}^m}{\partial z}\hat{z}, \quad (4.34)$$

and

$$\nabla \Phi_g^{*m} = \frac{\partial \Psi_{gx}^{*m}}{\partial x}\hat{x} + \frac{\partial \Psi_{gy}^{*m}}{\partial y}\hat{y} + \frac{\partial \Psi_{gz}^{*m}}{\partial z}\hat{z}. \quad (4.35)$$

With these expressions, the necessary integrals over the nodal volumes are evaluated to be

$$I_1^m(\phi_{g0}^*, \phi_{g'0}) \equiv \int_{\Omega^m} dV\left\{\Phi_g^{*m}\Phi_{g'}^m\right\} = a_x a_y a_z \left[\phi_{g0}^{*m}\phi_{g'0}^m + \sum_{u=x,y,z} \left\{ \frac{1}{3}a_{1gu}^{*m}a_{1g'u}^m + \frac{1}{5}a_{2gu}^{*m}a_{2g'u}^m + \right.\right.$$

$$+ \frac{6}{35}a_{3gu}^{*m}a_{3g'u}^m + \frac{2}{35}a_{4gu}^{*m}a_{4g'u}^m + \frac{1}{5}(a_{3gu}^{*m}a_{1g'u}^m + a_{1gu}^{*m}a_{3g'u}^m)$$

$$\left.\left. - \frac{3}{35}(a_{4gu}^{*m}a_{2g'u}^m + a_{2gu}^{*m}a_{4g'u}^m)\right\}\right]. \quad (4.36)$$

and

$$I_2^m(\phi_{g0}^*, \phi_{g0}) \equiv \int\limits_{\Omega^m} dV \left\{ \nabla \Phi_g^{*m} \cdot \nabla \Phi_g^m \right\} =$$

$$= a_x a_y a_z \sum_{u=x,y,z} \left\{ \frac{1}{a_u^2} \left[4a_{1gu}^m a_{1gu}^{m*} + 12a_{2gu}^m a_{2gu}^{m*} + \frac{36}{5} a_{3gu}^m a_{3gu}^{m*} + \frac{36}{7} a_{4gu}^m a_{4gu}^{m*} \right] \right\} \quad (4.37)$$

The next step in this development is the evaluation of Γ, definition (4.26). The NEM continuity conditions between two neighbouring nodes n and m are such that the following equations are satisfied by the flux expansions:

$$\int\limits_{\Theta^{mn}} \Phi_g^m dA = \int\limits_{\Theta^{mn}} \Phi_g^n dA \quad (4.38)$$

and

$$\left(D_g^m \nabla \Phi_g^m \cdot \hat{u} \right)\Big|_{\Theta^{mn}} = \left(D_g^n \nabla \Phi_g^n \cdot \hat{u} \right)\Big|_{\Theta^{mn}} \quad (4.39)$$

Because of these interface conditions, $\Gamma=0$ provided that the boundary conditions of the adjoint problem are appropriately chosen. It should be emphasized that the interface conditions (4.38) and (4.39) may not be met by all 'nodal' schemes. Very likely, they are not satisfied by many finite element schemes. Also, care should also be taken when constructing the expansions (4.30) and (4.31) in the case where the 3–D NEM method uses discontinuity factors. In all of these cases, it may be true that Γ is non–zero.

With the assumption that $\Gamma=0$ (as is true in this case) and that the coefficients are constant within a node, equations (4.36) and (4.37) may now be used in equation (4.27) to calculate first–order accurate eigenvalue responses. The final form of (4.27) then becomes

$$\delta\left(\frac{1}{\lambda}\right) \approx \frac{\sum_{m=1}^{N_T} \sum_{g=1}^{N_G} \left[\delta D_g^m I_2^m(\phi_{g0}^*, \phi_{g0}) + \sum_{g'=1}^{N_G} \left(\delta A_{gg'}^m - \frac{1}{\lambda} \delta F_{gg'}^m \right) I_1^m(\phi_{g0}^*, \phi_{g'0}) \right]}{\sum_{m=1}^{N_T} \sum_{g=1}^{N_G} \sum_{g'=1}^{N_G} F_{gg'}^m I_1^m(\phi_{g0}^*, \phi_{g'0})} \quad (4.40)$$

The form of (4.40) is particularly convenient to calculate during a point kinetics calculation, since the integrals $I_1^m(\Phi_g^{*m}, \Phi_g^m)$ and $I_2^m(\Phi_g^{*m}, \Phi_g^m)$ may be calculated and stored at the beginning of the period during which the point kinetics model is to be used.

4.3 Example Calculations Using the Operator Formulation

An appropriate method to test the accuracy of the above formulation is to use the exact expression for the perturbation, given by (4.25). In this case, the perturbed forward flux is recalculated, and equation (4.40) may be expressed as

$$\delta\left(\frac{1}{\lambda}\right) = \frac{\sum\limits_{m=1}^{N_T}\sum\limits_{g=1}^{N_G}\left[\delta D_g^m I_2^m(\phi_{g0}^*,\phi_g) + \sum\limits_{g'=1}^{N_G}\left(\delta A_{gg'}^m - \frac{1}{\lambda}\delta F_{gg'}^m\right)I_1^m(\phi_{g0}^*,\phi_{g'})\right]}{\sum\limits_{m=1}^{N}\sum\limits_{g=1}^{N_G}\sum\limits_{g'=1}^{N_G} F_{gg'}^m I_1^m(\phi_{g0}^*,\phi_{g'})}$$

(4.41)

The result of (4.41) can be compared with the known change in eigenvalue between the unperturbed and the perturbed forward calculations. If the flux expansions were exactly the flux solutions of the PDE, then there would be no difference between (4.41) and the change in eigenvalue calculated from the forward solutions. Thus, comparison of (4.41) with the known perturbation of the eigenvalue indicates how accurately the operator formulation performs with the selected NEM mesh size. This method of calculating exact changes in the eigenvalue is known as 'exact perturbation theory,' and is described by Williams.[85]

The methodology described above has been implemented into the PANBOX code. The calculations presented here have been made using a 2–group NEM calculation for a typical PWR core database in octant symmetry. Both the adjoint and forward NEM solutions were calculated with a convergence criteria of 10^{-5} for both nodal fluxes and eigenvalue. The calculations were performed as follows: first a critical boron concentration search was made in a hot full power core condition, and all necessary components of the forward solution were stored. Then the physical adjoint solution was calculated, and the results from that calculation were also stored. Perturbations in cross sections were made in the input deck by selecting nodes and cross sections to be perturbed. The changes in the cross sections were then used with the forward and adjoint solutions to estimate the first order response, as described in the last section. A second forward calculation was made to calculate the new eigenvalue. Finally, an exact perturbation theory calculation using the first term of equation (4.41) was made.

Perturbation in 10 selected nodes.	λ determined by the operator formulation	λ determined by second forward calculation	λ determined by equation (4.41)
−40%	1.00020993	1.000291	1.00029111
−30%	1.00015748	1.000198	1.00019896
−20%	1.00010502	1.000122	1.00012195
−15%	1.00007880	1.000088	1.00008798
−10%	1.00005257	1.000056	1.00005639
+10%	0.99994767	0.999951	0.99995106
+15%	0.99992144	0.999929	0.99992877
+20%	0.99989522	0.999908	0.99990791
+30%	0.99984276	0.999870	0.99986959
+40%	0.99979031	0.999836	0.99983543

Table 4.1: Results of perturbation of the diffusion coefficient in 10 selected nodes.

Fig. 4.1: First order perturbation theory results for perturbation of the diffusion coefficient in 10 selected nodes.

Table 4.1 shows the results of perturbations made in the diffusion coefficient. Of particular interest here are the *exact* perturbation theory calculations: the agreement is excellent, and this confirms that the method of approximating the nodal fluxes and flux gradients is accurate in this instance. The first order perturbation theory calculations are also compared against the perturbed forward calculations in figure 4.1.

4.4 Summary

In this chapter, the point kinetics model was derived from the multigroup neutron diffusion equation by separating the neutron flux into a shape function and an amplitude function. The main technical contribution of this chapter was to show how the operator formulation of sensitivity theory, as originally defined by Cacuci et. al.,[53] can be implemented with the solution of the NEM equations. The contributions are distinct from those of Delmolino,[61] who laid down no theoretical foundations with respect to the consistency of the NEM, and did not properly consider the potential importance of the Γ term (4.26).

5. Axial Kinetics Model

In this chapter, a one–dimensional approximation to the time–dependent multigroup diffusion equation is derived. The continuous variable representation of this one–dimensional model is then discretized with the NEM. The resulting discretized equations, hereafter referred to as the 'axial kinetics model,' are purposely cast into forms resembling the NEM discretization of the three–dimensional multigroup diffusion equation. In this way, existing numerical solution algorithms and PANBOX subroutines can be used to solve the equations of the axial kinetics model.

5.1 Continuous variable derivation

A review of the recent literature[65,66,67,68,69] shows a common trend in the development of one–dimensional kinetics models for coupled neutronics/thermal–hydraulic code systems. The axial kinetics models are generally derived from a one–dimensional form of the differential multigroup neutron diffusion equation[65,66,67] and implemented in the thermalhydraulics code. Then the users of the program are obliged to devise a method to choose one–dimensional coefficients such that an agreement exists between the one–dimensional solution and a three–dimensional solution from a core analysis or design code.[68,69] This approach is unsatisfactory from two perspectives. Firstly, it requires that the user spend time and resources to develop a methodology to produce appropriate coefficients for the one–dimensional model. Secondly, it increases the possibility that the user will obtain unrealistic results, because modified coefficients may not be correct coefficients.

It will be shown in this chapter that if the three–dimensional solution is considered in the derivation of the one–dimensional model, then agreement between the three– and one–dimensional solutions can be imposed automatically. Not only does this save user time and resources, but it also eliminates the chance that unwanted errors are introduced in the process of coefficient generation for the axial model.

5.1.1 Governing Equations

In accordance with the above discussion, the starting point for the derivation of a one–dimensional model in the axial direction are the space–time continuous multigroup neutron diffusion equations, coupled with the delayed neutron precursor equations:

$$\frac{1}{v_g}\frac{\partial \phi_g}{\partial t} - \nabla \cdot D_g \nabla \phi_g + \Sigma_{rg}\phi_g = \sum_{g'=1}^{N_G} \Sigma_{gg'}\phi_{g'} + \sum_{g'=1}^{N_G} \Sigma_{pgg'}\phi_{g'} + \sum_{i=1}^{N_I} \chi_{dg}^i \lambda_i c_i, \qquad (5.11)$$

$$g = 1,\ldots,N_G$$

$$\frac{\partial c_i}{\partial t} = \frac{1}{\lambda} \sum_{g'=1}^{N_G} \sum_{j=1}^{N_J} \beta_i^j \nu \Sigma_{fg'}^j \phi_{g'} - \lambda_i c_i \ , \quad i = 1,..,N_I \tag{5.12}$$

where

$$\phi_g \equiv \phi_g(\vec{r},t) \qquad g = 1,..,N_G$$

$$c_i \equiv c_i(\vec{r},t) \qquad i = 1,..,N_I$$

$$\Sigma_{pgg'} \equiv \frac{1}{\lambda} \sum_{j=1}^{N_J} \chi_{pg}^j (1 - \beta^j) \nu \Sigma_{fg'}^j$$

$$\beta^j \equiv \sum_{i=1}^{N_I} \beta_i^j$$

A one–dimensional model may be formally derived by separating the neutron flux into a shape function $\psi_g^{1D}(\vec{r},t)$ and an envelope function $N_g(z,t)$ as follows:

$$\phi_g(\vec{r},t) = N_g(z,t)\psi_g^{1D}(\vec{r},t) \tag{5.13}$$

It should be noted that this separation involves no ansatz. The shape function $\psi_g^{1D}(\vec{r},t)$ may vary as a function of z, so the flux is not assumed to be separable in the z direction. The purpose of this separation will become apparent later, when the one–dimensional analogue of the adiabatic *quasi–static* approximation is made, and $\psi_g^{1D}(\vec{r},t)$ is approximated to vary weakly in time.

The neutron precursor concentrations are separated in a way similar to the neutron flux:

$$c_i(\vec{r},t) = C_i^{1D}(z,t)\xi_i^{1D}(\vec{r},t) \tag{5.14}$$

The expressions (5.13) and (5.14) are now substituted into equations (5.11) and (5.12). Equation (5.11) is multiplied with a weight function $W_g(\vec{r})$ and integrated over the plane perpendicular to the z axis, represented as $A_\vdash$. Integration yields:

$$\frac{1}{v_g}\left[\frac{\partial N_g}{\partial t} \int_{A_\vdash} W_g(\vec{r})\psi_g^{1D}(\vec{r},t)dA_\vdash + N_g(z,t) \int_{A_\vdash} W_g(\vec{r})\frac{\partial \psi_g^{1D}}{\partial t}dA_\vdash\right]$$

$$- \int_{A_\vdash} W_g(\vec{r})\nabla \cdot \left(D_g\psi_g^{1D}\frac{\partial N_g}{\partial z}\right)dA_\vdash - \int_{A_\vdash} W_g(\vec{r})\nabla \cdot \left(N_g D_g \nabla \psi_g^{1D}\right)dA_\vdash + \Sigma_{rg}N_g =$$

$$= \sum_{g'=1}^{N_G} \Sigma_{gg'}N_{g'} + \sum_{g'=1}^{N_G} \Sigma_{pgg'}N_{g'} + \sum_{i=1}^{N_I} \chi_{dg}^i \lambda_i C_i^{1D}(z,t) \tag{5.15}$$

62

Equation (5.12) is also multiplied by a weight function W_i^C and also integrated over $A_\vdash$, giving

$$\frac{\partial C_i^{1D}}{\partial t} \int_{A_\vdash} W_i^C(\vec{r})\xi_i^{1D}(\vec{r},t)dA_\vdash + C_i^{1D}(z,t) \int_{A_\vdash} W_i^C(\vec{r})\frac{\partial \xi_i^{1D}}{\partial t}dA_\vdash =$$

$$= \sum_{g'=1}^{N_G} \overline{\Sigma}_{\beta g'}^i N_{g'} - \sum_{i=1}^{I} \lambda_i C_i^{1D}(z,t) \int_{A_\vdash} W_i^C(\vec{r})\xi_i^{1D}(\vec{r},t)dA_\vdash \qquad (5.16)$$

The new coefficients defined in this process are as follows:

$$\overline{\Sigma}_{rg}(\vec{r},t) \equiv \int_{A_\vdash} W_g(\vec{r})\Sigma_{rg}(\vec{r},t)\psi_g^{1D}(\vec{r},t)dA_\vdash$$

$$\overline{\Sigma}_{gg'}(\vec{r},t) \equiv \int_{A_\vdash} W_g(\vec{r})\Sigma_{gg'}(\vec{r},t)\psi_{g'}^{1D}(\vec{r},t)dA_\vdash$$

$$\overline{\Sigma}_{pgg'}(\vec{r},t) \equiv \int_{A_\vdash} W_g(\vec{r})\Sigma_{pgg'}(\vec{r},t)\psi_{g'}^{1D}(\vec{r},t)dA_\vdash \qquad (5.17)$$

$$\overline{\Sigma}_{\beta g'}^i(\vec{r},t) \equiv \frac{1}{\lambda} \int_{A_\vdash} W_i^C(\vec{r}) \sum_{j=1}^{N_J} \beta_i^j \nu \Sigma_{fg'}^j(\vec{r},t)\psi_{g'}^{1D}(\vec{r},t)dA_\vdash$$

$$\overline{\chi}_{dg}^i \equiv \int_{A_\vdash} \chi_{dg}^i W_g(\vec{r})\xi_i^{1D}(\vec{r},t)dA_\vdash$$

Equations (5.11) and (5.12) have been transformed from a set of linear equations with a unique solution to a set of nonlinear equations. While the solution of the neutron flux should remain unique, the individual functions $\psi_g^{1D}(\vec{r},t)$, $N_g(z,t)$, $\xi_i^{1D}(\vec{r},t)$ and $C_i^{1D}(z,t)$ are no longer unique. In order to impose a unique solution upon the envelope functions N_g and C_i^{1D}, the following constraints are imposed on the shape functions:

$$\int_{A_\vdash} W_g(\vec{r})\psi_g^{1D}(\vec{r},t)dA_\vdash = 1$$

$$\int_{A_\vdash} W_i^C(\vec{r})\xi_i^{1D}(\vec{r},t)dA_\vdash = 1 \qquad (5.18)$$

These constraints are effectively normalizations of the shape function. They reduce the one–dimensional model equations (5.15) and (5.16) to

$$\frac{1}{v_g}\left[\frac{\partial N_g}{\partial t} + N_g(z,t)\int_{A_\vdash} W_g(\vec{r})\frac{\partial \psi_g^{1D}}{\partial t}dA_\vdash\right] - \int_{A_\vdash} W_g(\vec{r})\nabla\cdot\left(D_g\psi_g^{1D}\frac{\partial N_g}{\partial z}\hat{k}\right)dA_\vdash$$

$$-\int_{A_\vdash} W_g(\vec{r})\nabla\cdot\left(N_g D_g\nabla\psi_g^{1D}\right)dA_\vdash + \overline{\Sigma}_{rg}N_g = \sum_{g'=1}^{N_G}\overline{\Sigma}_{gg'}N_{g'} + \sum_{g'=1}^{N_G}\overline{\Sigma}_{pgg'}N_{g'}$$

$$+\sum_{i=1}^{I}\overline{\chi}_{dg}^i\lambda_i C_i^{1D}(z,t) \quad (5.19)$$

and

$$\frac{\partial C_i^{1D}}{\partial t} + C_i^{1D}(z,t)\int_{A_\vdash} W_i^C(\vec{r})\frac{\partial \xi_i^{1D}}{\partial t}dA_\vdash = \sum_{g'=1}^{N_G}\overline{\Sigma}_{\beta g'}^i N_{g'} - \sum_{i=1}^{I}\lambda_i C_i^{1D}(z,t) \quad (5.20)$$

For the development of the one–dimensional model, the adiabatic quasi–static approximation is made which approximates the shape functions to be so slowly varying in time that the time derivatives of the shape functions are negligible. Mathematically, this approximation is expressed as

$$\frac{\partial \psi_g^{1D}}{\partial t} \approx 0$$

$$\frac{\partial \xi_i^{1D}}{\partial t} \approx 0$$

which reduces equations (5.19) and (5.20) to

$$\frac{1}{v_g}\frac{\partial N_g}{\partial t} - \overset{\textcircled{1}}{\int_{A_\vdash} W_g(\vec{r})\nabla\cdot\left(D_g\psi_g^{1D}\frac{\partial N_g}{\partial z}\hat{k}\right)dA_\vdash} - \overset{\textcircled{2}}{\int_{A_\vdash} W_g(\vec{r})\nabla\cdot\left(N_g D_g\nabla\psi_g^{1D}\right)dA_\vdash}$$

$$+ \overline{\Sigma}_{rg}N_g = \sum_{g'=1}^{N_G}\overline{\Sigma}_{gg'}N_{g'} + \sum_{g'=1}^{N_G}\overline{\Sigma}_{pgg'}N_{g'} + \sum_{i=1}^{N_I}\overline{\chi}_{dg}^i\lambda_i C_i^{1D}(z,t) \quad (5.21)$$

and

$$\frac{\partial C_i^{1D}}{\partial t} = \sum_{g'=1}^{N_G}\overline{\Sigma}_{\beta g'}^i N_{g'} - \sum_{i=1}^{N_I}\lambda_i C_i^{1D}(z,t) \quad (5.22)$$

Terms ① and ② from equation (5.21) must now be further evaluated. Assuming that the diffusion coefficient is homogeneous in the z–direction (this assumption will be justified in section 5.2), term ① becomes

64

$$\int_{A_\vdash} W_g(\vec{r}) \nabla \cdot \left(D_g \psi_g^{1D} \frac{\partial N_g}{\partial z} \hat{k} \right) dA_\vdash = \int_{A_\vdash} W_g D_g \frac{\partial}{\partial z} \left(\psi_g^{1D} \frac{\partial N_g}{\partial z} \right) dA_\vdash$$

Term ② may be written as,

$$\int_{A_\vdash} W_g(\vec{r}) \nabla \cdot \left(N_g D_g \nabla \psi_g^{1D} \right) dA_\vdash = \int_{A_\vdash} W_g D_g \frac{\partial}{\partial z} \left(N_g \frac{\partial \psi_g^{1D}}{\partial z} \right) dA_\vdash - \overline{B}_g N_g(z, t)$$

The radial leakage coefficient is defined in Cartesian coordinates as

$$\overline{B}_g \equiv - \int_{A_\vdash} W_g \left[\frac{\partial}{\partial x} \left(D_g \frac{\partial \psi_g^{1D}}{\partial x} \right) + \frac{\partial}{\partial y} \left(D_g \frac{\partial \psi_g^{1D}}{\partial y} \right) \right] dA_\vdash \qquad (5.23)$$

The freely chosen weight function is now restricted to have no z– dependence. Then, equation (5.21) may be written as

$$\frac{1}{v_g} \frac{\partial N_g}{\partial t} - \frac{\partial^2}{\partial z^2} \left[N_g \int_{A_\vdash} W_g D_g \psi_g^{1D} dA_\vdash \right] + \overline{B}_g N_g(z, t)$$

$$+ \overline{\Sigma}_{rg} N_g = \sum_{g'=1}^{N_G} \overline{\Sigma}_{gg'} N_{g'} + \sum_{g'=1}^{N_G} \overline{\Sigma}_{pgg'} N_{g'} + \sum_{i=1}^{N_I} \overline{\chi}_{dg}^i \lambda_i C_i^{1D}(z, t) \qquad (5.24)$$

Most one–dimensional models in the literature implicitly make the assumption that the shape function has no z dependence. In this case equation (5.24) reduces to the simple form of

$$\frac{1}{v_g} \frac{\partial N_g}{\partial t} - \overline{D}_g \frac{\partial^2 N_g}{\partial z^2} + \left(\overline{B}_g + \overline{\Sigma}_{rg} \right) N_g(z, t) =$$

$$= \sum_{g'=1}^{N_G} \overline{\Sigma}_{gg'} N_{g'} + \sum_{g'=1}^{N_G} \overline{\Sigma}_{pgg'} N_{g'} + \sum_{i=1}^{N_I} \overline{\chi}_{dg}^i \lambda_i C_i^{1D}(z, t) \qquad (5.25)$$

where

$$\overline{D}_g \equiv \int_{A_\vdash} W_g(\vec{r}) D_g \psi_g^{1D}(\vec{r}) dA_\vdash$$

However, as indicated by equation (5.13), the shape function is dependent on z in the general case, and thus the discretization of the continuous variable equations must begin from equation (5.24) and not equation (5.25).

5.1.2 Boundary and Interface Conditions

The interface conditions of the diffusion approximation are that the neutron flux and the component of neutron current normal to the interface are continuous for all energy groups. This is equivalent to stating that positive and negative partial currents are continuous across the interfaces. The partial currents in the z direction are given by

$$j_{gz}^+ = \frac{\phi_g}{4} - \frac{D_g}{2}\frac{\partial \phi_g}{\partial z} \tag{5.26}$$

$$j_{gz}^- = \frac{\phi_g}{4} + \frac{D_g}{2}\frac{\partial \phi_g}{\partial z} \tag{5.27}$$

The partial currents of the one–dimensional axial model are *defined* as

$$J_{gz}^+ \equiv \int_{A_\vdash} W_g j_{gz}^+ dA_\vdash = \frac{N_g}{4} - \frac{\overline{D}_g}{2}\frac{\partial N_g}{\partial z} - \frac{N_g}{2}\int_{A_\vdash} W_g D_g \frac{\partial \psi_g^{1D}}{\partial z}dA_\vdash \tag{5.28}$$

$$J_{gz}^- \equiv \int_{A_\vdash} W_g j_{gz}^- dA_\vdash = \frac{N_g}{4} + \frac{\overline{D}_g}{2}\frac{\partial N_g}{\partial z} + \frac{N_g}{2}\int_{A_\vdash} W_g D_g \frac{\partial \psi_g^{1D}}{\partial z}dA_\vdash \tag{5.29}$$

Definitions (5.28) and (5.29) are a crucial part of the derivation. Continuity of the partial currents of the three dimensional model also implies that these one–dimensional model partial currents must be kept continuous across any interfaces. The one–dimensional model expressions for the partial currents, equations (5.28) and (5.29) have an additional term in comparison to the expressions for the partial currents of the three dimensional model, equations (5.26) and (5.27). The extra term accounts for the varying z–dependence of the shape function. This term arises from the explicit treatment of the three–dimensional flux shape in the one–dimensional model, and is the primary distinguishing feature of this derivation from that of other works.[65,66,67]

Boundary conditions of the three dimensional model are treated in a similar way: where incoming partial currents are required to be zero, the corresponding condition must also hold true for the incoming partial currents of the one–dimensional model. Boundary conditions on surfaces normal to the z–direction are considered implicitly in the radial leakage coefficient, equation (5.23).

5.2 Nodal Expansion Method Discretization

Discretization of the continuous variable form of the one–dimensional model with the NEM follows the general procedure outlined in references 23, 24, and 25. Here, the parts of the derivations distinct to the axial kinetics model are described. Details external

to the distinct features of the axial kinetics model may be found in Appendix C and the references.[23,24,25] The full set of discretized equations which describe the M2–B2 variant of the NEM include the nodal balance equation, the outgoing partial current equations, and the moments equations. Derivation of each of these equations is described below in separate subsections.

5.2.1 Nodal Balance Equation

The NEM shares characteristics of both finite volume and finite element methods. The discretization requires that the integral of equations (5.22) and (5.24) be satisfied over homogeneous regions called nodes. The nodal averaged axial fluxes and precursor concentrations are defined as

$$N_g^m(t) \equiv \frac{1}{\Delta z_m} \int_{z_{m-1}}^{z_m} N_g(z, t) dz \tag{5.30}$$

$$C_i^m(t) \equiv \frac{1}{\Delta z_m} \int_{z_{m-1}}^{z_m} C_i^{1D}(z, t) dz \tag{5.31}$$

Here, z_{m-1} and z_m are the lower and upper coordinate boundaries of the homogenized node in question, and $\Delta z_m = z_m - z_{m-1}$. Equations (5.22) and (5.24) are integrated over the axial node to yield,

$$\frac{dC_i^m}{dt} = \sum_{g'=1}^{N_G} \overline{\Sigma}_{\beta g'}^{i,m} N_{g'}^m - \sum_{i=1}^{N_I} \lambda_i C_i^m(t) \tag{5.32}$$

and

$$\frac{1}{v_g} \frac{dN_g^m}{dt} - \frac{1}{\Delta z_m} \frac{\partial}{\partial z} \left[N_g \int_{A_\vdash} W_g D_g \psi_g^{1D} dA_\vdash \right]_{z=z_m} + \frac{1}{\Delta z_m} \frac{\partial}{\partial z} \left[N_g \int_{A_\vdash} W_g D_g \psi_g^{1D} dA_\vdash \right]_{z=z_{m-1}}$$

$$+ (\overline{\Sigma}_{rg}^m + \overline{B}_g^m) N_g^m = \sum_{g'=1}^{N_G} \overline{\Sigma}_{gg'}^m N_{g'}^m + \sum_{g'=1}^{N_G} \overline{\Sigma}_{pgg'}^m N_{g'}^m + \sum_{i=1}^{N_I} \overline{\chi}_{dg}^i \lambda_i C_i^m(t) \tag{5.33}$$

Equations (5.28) and (5.29) can be used to cast the leakage terms of equation (5.33) in terms of the partial currents of the axial model. The incoming and outgoing partial currents of a node of index m are defined by

$$J_{gzr}^{+m} \equiv \left[\frac{N_g}{4} - \frac{1}{2}\frac{\partial}{\partial z}\left[N_g \int_{A_\vdash} W_g D_g \psi_g^{1D} dA_\vdash\right]\right]_{z=z_m} \tag{5.34}$$

$$J_{gzr}^{-m} \equiv \left[\frac{N_g}{4} + \frac{1}{2}\frac{\partial}{\partial z}\left[N_g \int_{A_\vdash} W_g D_g \psi_g^{1D} dA_\vdash\right]\right]_{z=z_m} \tag{5.35}$$

The partial currents on the left hand side of the node, J_{gzl}^{+m} and J_{gzl}^{-m}, are similarly defined, except that they are evaluated at $z = z_{m-1}$ instead of $z = z_m$. From these definitions, it can be seen that the relationships

$$J_{gzr}^{-m} - J_{gzr}^{+m} = \frac{\partial}{\partial z}\left[N_g \int_{A_\vdash} W_g D_g \psi_g^{1D} dA_\vdash\right]_{z=z_m} \tag{5.36}$$

and

$$J_{gzl}^{-m} - J_{gzl}^{+m} = \frac{\partial}{\partial z}\left[N_g \int_{A_\vdash} W_g D_g \psi_g^{1D} dA_\vdash\right]_{z=z_{m-1}} \tag{5.37}$$

hold.

Using equations (5.36) and (5.37) in equation (5.33), the time dependent axial nodal balance equation is derived as

$$\frac{1}{v_g}\frac{dN_g^m}{dt} + \frac{1}{\Delta z_m}(J_{gul}^{-m} + J_{gur}^{+m} - J_{gul}^{+m} - J_{gur}^{-m}) + (\overline{\Sigma}_{rg}^m + \overline{B}_g^m)N_g^m =$$

$$= \sum_{g'=1}^{N_G} \overline{\Sigma}_{gg'}^m N_{g'}^m + \sum_{g'=1}^{N_G} \overline{\Sigma}_{pgg'}^m N_{g'}^m + \sum_{i=1}^{N_I} \overline{\chi}_{dg}^i \lambda_i C_i^m \tag{5.38}$$

Equation (5.38) is discretized in time using the exponential transformation method outlined in the Appendix A. The result yields the following expression for the time derivative

$$\frac{dN_g^m}{dt} = \frac{(1 + \omega^m \Delta t)N_g^m(t) - e^{\omega^m \Delta t}N_g^m(t_0)}{\Delta t} \tag{5.39}$$

which may be substituted into the balance equation (5.38) to yield the following implicit equation for the node–averaged flux:

68

$$\frac{1}{\Delta z_m}(J_{gul}^{-m} + J_{gur}^{+m} - J_{gul}^{+m} - J_{gur}^{-m}) + (\overline{\Sigma}_{Rg}^m + \overline{B}_g^m)N_g^m =$$

$$= \sum_{g'=1}^{N_G} \overline{\Sigma}_{gg'}^m N_{g'}^m + \sum_{g'=1}^{N_G} \overline{\Sigma}_{pgg'}^m N_{g'}^m + \sum_{i=1}^{N_I} \overline{\chi}_{dg}^i \lambda_i C_i^m(t) + S_g^m \qquad (5.40)$$

where

$$\overline{\Sigma}_{Rg}^m \equiv \frac{(1 + \omega^m \Delta t)}{v_g \Delta t} + \Sigma_{rg}$$

$$S_g^m \equiv \frac{e^{\omega^m \Delta t}}{v_g \Delta t} N_g^m(t_0)$$

The advantage to have the nodal balance equation in this form is that it retains the same structure as the nodal balance equation of the three dimensional model.[23,24,25] Thus, existing solution routines may be used once the coefficients and the effective source terms have been determined.

The time discretization of the precursor equations is shown in Appendix A.

5.2.2 Outgoing Current Equations

The second stage in the NEM discretization of the one–dimensional model is the determination of the outgoing current equations. To begin the derivation, $N_g(z,t)$ (referred to as the axial flux) is expanded in the NEM polynomials to a degree of fourth order

$$N_g(z, t) \approx \Psi_g^m(u, t) \equiv N_g^m(t) + \sum_{i=1}^{4} a_{ig}^m(t) h_i(u) , \qquad (5.41)$$

where

$$u = \frac{z - z^{m-1}}{\Delta z^m} \qquad (5.42)$$

In the rest of this subsection, the time dependence of the variables will be suppressed. The first two expansion coefficients a_{1g}^m and a_{2g}^m are given as follows:[23,24,25]

$$a_{1g}^m = \frac{\Psi_{gr}^m - \Psi_{gl}^m}{2} \qquad (5.43)$$

$$a_{2g}^m = N_g^m - \frac{\Psi_{gr}^m + \Psi_{gl}^m}{2} \qquad (5.44)$$

where

$$\Psi^m_{gr} \equiv \Psi^m_g(u = 1) = N_g(z_m)$$

$$\Psi^m_{gl} \equiv \Psi^m_g(u = 0) = N_g(z_{m-1})$$

Expressions for the higher order expansion coefficients, a^m_{3g} and a^m_{4g}, are derived in the following section. The first derivative of (5.41) may be evaluated at z_m and z_{m-1} to obtain the following expressions:

$$D_g \frac{\partial N_g}{\partial z}\bigg|_{z_{m-1}} \approx D_g \frac{\partial \Psi^m_g}{\partial z}\bigg|_{z_{m-1}} = \frac{D_g}{\Delta z_m}\left(2a^m_{1g} + 6a^m_{2g} - 6a^m_{3g} + 6a^m_{4g}\right)$$

$$D_g \frac{\partial N_g}{\partial z}\bigg|_{z_m} \approx D_g \frac{\partial \Psi^m_g}{\partial z}\bigg|_{z_m} = \frac{D_g}{\Delta z_m}\left(2a^m_{1g} - 6a^m_{2g} - 6a^m_{3g} - 6a^m_{4g}\right)$$

(5.45)

The following factors f_{gL} and f_{gR} are defined:

$$f_{gL} \equiv \left[\int_{A_\vdash} W_g D_g \frac{\partial \psi^{1D}_g}{\partial z} dA_\vdash\right]_{z_{m-1}} \qquad f_{gR} \equiv \left[\int_{A_\vdash} W_g D_g \frac{\partial \psi^{1D}_g}{\partial z} dA_\vdash\right]_{z_m}$$

Then the partial currents of the node of index m may be written in the form,

$$J^{+m}_{gzl} = \left(\frac{1}{4} - \frac{f_{gL}}{2}\right)\Psi_{gl} - \frac{\overline{D}_g}{2}\frac{\partial N_g}{\partial z}\bigg|_{z_{m-1}} \qquad J^{-m}_{gzl} = \left(\frac{1}{4} + \frac{f_{gL}}{2}\right)\Psi_{gl} + \frac{\overline{D}_g}{2}\frac{\partial N_g}{\partial z}\bigg|_{z_{m-1}}$$

$$J^{+m}_{gzr} = \left(\frac{1}{4} - \frac{f_{gR}}{2}\right)\Psi_{gr} - \frac{\overline{D}_g}{2}\frac{\partial N_g}{\partial z}\bigg|_{z_m} \qquad J^{-m}_{gzr} = \left(\frac{1}{4} + \frac{f_{gR}}{2}\right)\Psi_{gr} + \frac{\overline{D}_g}{2}\frac{\partial N_g}{\partial z}\bigg|_{z_m}$$

(5.46)

Equations (5.43), (5.44), and (5.46) may be substituted into equations (5.45) to arrive at the following coupled equations for the partial currents:

$$J^{+m}_{gzl} - J^{-m}_{gzl} = -\frac{\overline{D}^m_g}{\Delta z_m}\left[-2\Psi_{gr} - 4\frac{\Psi_{gl}}{f'_{gL}} + 6N^m_g - 6a^m_{3g} + 6a^m_{4g}\right]$$

$$J^{+m}_{gzr} - J^{-m}_{gzr} = -\frac{\overline{D}^m_g}{\Delta z_m}\left[4\frac{\Psi_{gr}}{f'_{gR}} + 2\Psi_{gl} + 6N^m_g - 6a^m_{3g} + 6a^m_{4g}\right]$$

(5.47)

here, the new coefficients are defined as

$$f'_{gL} \equiv \frac{4}{4 - \dfrac{\Delta z_m f_{gL}}{\overline{D}_g^m}} \qquad\qquad f'_{gR} \equiv \frac{4}{4 - \dfrac{\Delta z_m f_{gR}}{\overline{D}_g^m}} \qquad\qquad (5.48)$$

Equations (5.47) look similar to the equations which are used to derive outgoing current equations using heterogeneity factors, or so called 'discontinuity factors' to force agreement between solution of heterogeneous assemblies and the solutions of the homogeneous problems derived from them.[64] If d_{gL} and d_{gR} are the heterogeneity factors, then the corresponding partial current equations in the heterogeneity factor formulation take the form of

$$J^{+m}_{gzl} - J^{-m}_{gzl} = -\frac{\overline{D}_g^m}{\Delta z_m}\left[-2\frac{\Psi_{gr}}{d_{gR}} - 4\frac{\Psi_{gl}}{d_{gL}} + 6N_g^m - 6a^m_{3g} + 6a^m_{4g} \right]$$

$$J^{+m}_{gzr} - J^{-m}_{gzr} = -\frac{\overline{D}_g^m}{\Delta z_m}\left[4\frac{\Psi_{gr}}{d_{gR}} + 2\frac{\Psi_{gl}}{d_{gL}} + 6N_g^m - 6a^m_{3g} + 6a^m_{4g} \right] \qquad (5.49)$$

Both sets of partial current equations (5.47) and (5.49) involve correction factors to the 'standard' partial current equations of the NEM. It is instructive to compare how equations (5.47) and (5.49) have been derived. Equations (5.49) stem from the 'equivalence theory' developed by Koebke[62,63] and Smith.[64] Using the equivalence theory, extra degrees of freedom are introduced in the nodal solution of a homogenized reactor region so that the leakage and reaction rates can be forced to match that of the solution to the corresponding heterogeneous problem. Equations (5.47) also stem from a homogenization process – the process of collapsing a three–dimensional model to one dimension. The definitions of the partial currents of the one dimensional model, definitions (5.28) and (5.29), contain an extra term not found in the definition of partial currents for the three dimensional model [equations (5.26) and (5.27)]. It is this extra term that gives rise directly to the 'correction factors' in equation (5.47). In contrast, the heterogeneity factors of equations (5.49) have been introduced in almost an ad hoc manner, to provide extra degrees of freedom with which to match two solutions.

PANBOX has the built in capability of solving the NEM equations with or without heterogeneity factors. Therefore, it is advantageous to relate the two sets of factors, and then calculate heterogeneity factors from the correction factors. The heterogeneity factors can then be used in the existing PANBOX routines without them having to be changed. Equations (5.47) and (5.49) will have the same solutions provided that the heterogeneity factors satisfy the following relationships:

$$\frac{1}{d_{gR}} = \frac{1}{3}\left[\frac{4}{f'_{gR}} - 1\right] + \frac{2}{3}\frac{\Psi_{gl}}{\Psi_{gr}}\left[1 - \frac{1}{f'_{gl}}\right]$$

$$\frac{1}{d_{gL}} = \frac{1}{3}\left(\frac{4}{f_{gL}'} - 1\right) + \frac{2}{3}\frac{\Psi_{gr}}{\Psi_{gl}}\left(1 - \frac{1}{f_{gR}'}\right)$$

(5.50)

The heterogeneity factors are calculated with equation (5.50), during the inner iterations of a time step. This yields results equivalent to those as if the partial current equations (5.47), with the correction factors, were solved for the outcoming currents. The common industry practice is to hold heterogeneity factors constant in time during a transient calculation. While Gehin and Henry[32] proposed to update heterogeneity factors periodically during a transient calculation, their method relied on recalculating the heterogeneous problem to do so. In contrast, a relationship between the standard heterogeneity factors used in the industry and a more rigorously derived correction factor based on the shape function is used here to update the heterogeneity factors.

The final form of the outgoing current equations is found by eliminating one outgoing partial current from each of equations (5.47) to yield two equations which take the form:

$$J_{gzl}^{-m} = C_{1gL}^m(N_g^m + a_{4g}^m) + C_{2gL}^m J_{gzl}^{+m} + C_{3gL}^m J_{gzr}^{-m} - C_{4gL}^m a_{3g}^m$$

$$J_{gzr}^{+m} = C_{1gR}^m(N_g^m + a_{4g}^m) + C_{3gR}^m J_{gzl}^{+m} + C_{2gR}^m J_{gzr}^{-m} + C_{4gR}^m a_{3g}^m$$

(5.51)

The constants C_{igs}^m, $i=1,..,4$; $s=L,R$; are given in Appendix C. These equations, like the nodal balance equation (5.38) also retain the same form as that of the three dimensional model.

5.2.3 Moment Equations

If $a_{3g}^m=a_{4g}^m=0$, then equations (5.38) and (5.51) form the M0 approximation of the NEM, in which the flux is expanded in quadratic polynomials across the nodes. To achieve greater accuracy, auxiliary equations must be derived to determine the higher order flux moments a_{3g}^m and a_{4g}^m. These moments appear as small correction terms in the outgoing current equations (5.51). For the determination of the higher order axial flux moments a_{3g}^m and a_{4g}^m, the polynomial expansion of the axial flux [equation (5.41)] is substituted into equation (5.24), to yield

$$\frac{1}{v_g}\frac{\partial \Psi_g^m}{\partial t} - \frac{\partial^2}{\partial z^2}\left[\Psi_g^m \int_{A_\vdash} W_g D_g \psi_g^{1D} dA_\vdash\right] + L_g(z)$$

$$+ \overline{\Sigma}_{rg}\Psi_g^m = \sum_{g'=1}^{N_G} \overline{\Sigma}_{gg'}\Psi_{g'}^m + \sum_{g'=1}^{N_G} \overline{\Sigma}_{pgg'}\Psi_{g'}^m + \sum_{i=1}^{I} \overline{\chi}_{dg}^i \lambda_i C_i(z,t) \qquad (5.52)$$

The radial leakage $\overline{B}_g N_g(z,t)$ is here approximated as a quadratic polynomial $L_g(z)$, in keeping with the B2 transverse leakage approximation of the NEM. Furthermore, *for the moments equations*, the approximation is made that the shape function is not a strong function of z locally within the node, so that the second order partial derivative with respect to z is approximated as

$$\frac{\partial^2}{\partial z^2}\left[\Psi_g^m \int_{A_\vdash} W_g D_g \psi_g^{1D} dA_\vdash\right] \approx \overline{D}_g^m \frac{\partial^2 \Psi_g}{\partial z^2} \qquad (5.53)$$

Since approximation (5.53) may introduce some small inconsistencies between the 3–D and 1–D discretized solutions, these inconsistencies are corrected by adjusting the correction factors derived in subsection 5.2.2. This procedure of forcing an equivalent solution between the 3–D and the 1–D model solutions is described in section 5.3.

Using approximation (5.53), the one dimensional local diffusion equation may be written in terms of u as

$$\frac{1}{v_g}\frac{\partial \Psi_g^m}{\partial t} - \frac{\overline{D}_g^m}{\Delta z^2}\frac{\partial^2 \Psi_g^m}{\partial u^2} + L_g(u) + \Sigma_{rg}\Psi_g^m =$$

$$= \sum_{g'=1}^{N_G} \overline{\Sigma}_{gg'}\Psi_{g'}^m + \sum_{g'=1}^{N_G} \overline{\Sigma}_{pgg'}\Psi_{g'}^m + \sum_{i=1}^{N_I} \overline{\chi}_{dg}^i \lambda_i C_i(u,t) \qquad (5.54)$$

In terms of the NEM polynomials, $L_g(z)$ may be written in the following form:

$$L_g(u) = b_{0g} + b_{1g}h_1(u) + b_{2g}h_2(u)$$

where

$$b_{0g} = \overline{B}_g^m N_g^m$$

$$b_{1g} = \frac{1}{2}(L_{gr} - L_{gl})$$

$$b_{2g} = b_{0g} - \frac{1}{2}(L_{gr} + L_{gl})$$

and

$$L_{gl} = \frac{\Delta z^{m-1}\overline{B}_g^m N_g^m + \Delta z^m \overline{B}_g^{m-1} N_g^{m-1}}{\Delta z^{m-1} + \Delta z^m}$$

$$L_{gr} = \frac{\Delta z^{m+1}\overline{B}_g^m N_g^m + \Delta z^m \overline{B}_g^{m+1} N_g^{m+1}}{\Delta z^{m+1} + \Delta z^m}$$

This formulation is consistent with the M2–B2 variant of the NEM.[23,24,25]

$\partial \Psi_g / \partial t$ and $C_i(z,t)$ are approximated in the standard way for the moments equations (see Appendix C), and the resulting time–discrete moment equation is multiplied by the weight function $h_1(u)$, and integrated over the node. This gives rise to the first discrete moment equation,

$$\left[60\frac{\overline{D}_g^m}{\Delta z^2} + \overline{\Sigma}_g^m\right]a_{3g}^m = -\frac{5}{3}\Sigma_g a_{1g}^m + \sum_{g'=1}^{N_G} \overline{\Sigma}_{gg'}^m\left(\frac{5}{3}a_{1g'}^m + a_{3g'}^m\right) + \\ + \sum_{g'=1}^{N_G} \overline{\Sigma}_{qgg'}\left(\frac{5}{3}a_{1g'}^m + a_{3g'}^m\right) - \frac{5}{3}b_{1g} \tag{5.55}$$

Multiplying the time–discrete moment equation by the weight function $h_2(u)$ and integrating over the node gives rise to the second discrete moment equation,

$$\left[140\frac{\overline{D}_g^m}{\Delta z^2} + \overline{\Sigma}_g^m\right]a_{4g}^m = \frac{7}{3}\Sigma_g a_{2g}^m + \sum_{g'=1}^{N_G} \overline{\Sigma}_{gg'}^m\left(-\frac{7}{3}a_{2g'}^m + a_{4g'}^m\right) + \\ + \sum_{g'=1}^{N_G} \overline{\Sigma}_{qgg'}\left(-\frac{7}{3}a_{2g'}^m + a_{4g'}^m\right) + \frac{7}{3}b_{2g} \tag{5.56}$$

The new coefficients $\overline{\Sigma}_g^m$ and $\overline{\Sigma}_{qgg'}^m$ are given in Appendix C.

5.3 Restriction of Variables and Coefficients

In this section, the details of computing the one–dimensional coefficients and initial conditions of the axial kinetics model are described. The procedure is outlined in Figure 5.1. The prerequisite to the activation of the 1–D axial kinetics model is the calculation of the 3–D kinetics model prior to that period. Thus, the beginning of every axial kinetics time integration period begins with a solution of the 3–D NEM equations for the first time step. The purpose of this first 3–D kinetics calculation is to calculate the shape function, the initial conditions of the axial model, and the correction factors for the 1–D model.

Nodal averaged quantities for the 1–D model are calculated by using the nodal averaged quantities of the 3–D model. Let V_m be the set of the 3–D model nodes belonging to axial plane m of the 1–D model. ϕ_g^n, c_i^n and W_g^n, $\Omega^n \in V_m$ are respectively the nodal aver-

Beginning of time step
$\Delta t = t - t_0$

Beginning of 1–D calculation period?

Yes → Calculate time step with 3–D model →

Calculate nodal averaged envelope and shape functions from 3–D variables; calculate 1–D partial currents

$$\phi_g \to N_g, \psi_g$$
$$\phi_{g,0} \to N_{g0}$$
$$j_{gzs} \to J_{gzs}$$

No ↓

Calculate effective 1–D cross sections from 3–D data base
$$\Sigma_g \to \overline{\Sigma}_g$$

Solve for N_g by calculating time step with the 1–D model NEM equations

Solve for heterogeneity factors, d_{gL} and d_{gR} during the iteration procedure, using the time–independent values of f_{gR}' f_{gL}'.

Calculate effective 1–D cross sections from 3–D data base
$$\Sigma_g \to \overline{\Sigma}_g$$

Solve for heterogeneity factors, d_{gL} and d_{gR}, and from them f_{gR}' f_{gL}'.

Next time step

Figure 5.1 Outline of the strategy for determining 1–D coefficients from 3–D data.

aged fluxes, precursor concentrations and weight functions on the mesh of the 3–D NEM model. The 3–D model nodal areas transverse to the z direction are given as $A_{\vdash}^n$, $\Omega^n \in V_m$. Then the shape function is determined by

$$\psi_g^n = \frac{\phi_g^n}{\sum\limits_{l, \Omega^l \in S_m} W_g^l \phi_g^l A_{\vdash}^l}, \quad \forall \, \Omega^n \in V_m$$

The axial model neutron flux and precursor concentration functions are initially determined by

$$N_g^m = \sum\limits_{l, \Omega^l \in S_m} W_g^l \phi_g^l A_{\vdash}^l$$

and

$$C_i^m = \sum\limits_{l, \Omega^l \in S_m} W_g^l c_g^l A_{\vdash}^l$$

These discretized expressions are consistent with the continuous variable derivation,

75

equations (5.13), (5.14) and (5.18). The partial currents of the one–dimensional model are determined through the use of definitions (5.28) and (5.29) to give

$$J^{m\pm}_{gzs} = \sum_{l,\Omega^l \in S_m} W^l_g j^{l\pm}_{gzs} A^l_\vdash$$

The one–dimensional cross sections and diffusions coefficients are all determined in a similar way according to

$$\overline{\Sigma}^m_g = \sum_{l,\Omega^l \in S_m} W^l_g \psi^l_g \Sigma^l_g A^l_\vdash$$

or

$$\overline{\Sigma}^m_{gg'} = \sum_{l,\Omega^l \in S_m} W^l_g \psi^l_{g'} \Sigma^l_{gg'} A^l_\vdash$$

depending on whether the coefficient acts on the in–group or out–of–group flux, respectively. When the above restriction procedure is performed, the one–dimensional nodal balance equation (5.40) is satisfied automatically to within the same accuracy as the original 3–D model equations are solved. However, it is not true that the outgoing current equations (5.51) together with the moment equations, (5.55) and (5.56), will be automatically satisfied. They can be satisfied if

(a) the NEM–M0 approximation is used in both 3–D and 1–D models and the correction factors (5.48) are accurately determined; or

(b) the NEM–M2 approximation is used in both 3–D and 1–D models and the correction factors (5.48) are adjusted so that the satisfaction of the equations (5.51), (5.55) and (5.56) is forced.

Condition (a) is unsatisfactory from the point of view of computation costs: the NEM–M0 (quadratic expansions) approximation is not accurate enough to be used in coarse mesh approximations to model reactor cores. Use of finer meshes may be too expensive for general applications. NEM–M2 (quartic expansions) is the preferred method, and it has been demonstrated in the past that it is accurate enough for many coarse mesh applications. Adjustment of the correction factors eliminates the inconsistencies introduced by assumption (5.53) for the moments equations. Method (b) is therefore chosen and the correction factors (5.48) are adjusted so that all of the axial kinetics model equations are satisfied at the time of transition from the 3–D to the 1–D model. After this time of transition, the adjusted correction factors are approximated as constant in time,

although the heterogeneity factors are updated by (5.57). Calculation of the correction factors is described in the following subsection.

5.3.1 Calculation of Correction Factors

At the time of transition from the 3–D to the 1–D model, the correction factors $f_{gL}^{m\,\prime}$ and $f_{gR}^{m\,\prime}$ must be determined so that the restricted one–dimensional model parameters N_g^m, $J_{gzs}^{m\pm}$, and C_i^m satisfy the one–dimensional model equations (5.51), (5.55) and (5.56). This is done by substituting (5.43) and (5.44) into the moment equations (5.55) and (5.56) and using the resulting equations with (5.49) [from which equations (5.51) are derived] to form a system of 4G equations with 4G unknowns for each axial node. The unknowns are d_{gL}^m, d_{gR}^m, a_{3gu}^m, and a_{4gu}^m, $g=1,...,N_G$. For the usual practical case of $N_G=2$, solution of these equations involves inverting an 8x8 matrix which can be performed efficiently by direct inversion techniques, for instance through L–U decomposition.[90] Equations (5.50) can then be used to calculate $f_{gL}^{\prime m}$ and $f_{gR}^{\prime m}$ from the heterogeneity factors d_{gL}^m and d_{gR}^m. This procedure is repeated for each axial node.

5.4 Solution Procedure

Once the shape function, correction factors, and initial conditions of the one–dimensional model have been generated at the time of transition from the 3–D model, subsequent time steps are calculated using these quantities to generate coefficients for the one–dimensional model. Equations (5.40), (5.51), (5.55) and (5.56) are the NEM equations which must be solved in these subsequent time steps.

The one–dimensional NEM model equations have been purposely constructed to retain the same form as the NEM equations of the three–dimensional model. This has the distinct advantage that the same solution algorithms may be used for the one–dimensional model as have been developed for the three–dimensional model. As previously reported for the three–dimensional model, a coarse mesh rebalancing scheme greatly improves the convergence rate of the one–dimensional model equations. In the one–dimensional case, the coarse mesh rebalancing scheme described by Finnemann et. al.[41] immediately reduces to a tridiagonal system of equations on the first coarse mesh rebalancing grid. These equations can be solved directly, so that iterative solution schemes are unnecessary on the coarse mesh rebalancing grid.

5.5 Test Cases and Discussion

Three test cases will be presented in the following subsections. For the first test case, the solution of the axial 1–D model will be compared against the analytical and 3–D model solutions of a homogeneous reactor in slab geometry. The two subsequent test cases use real PWR databases with both axially uniform and local perturbations to initiate the transient. Results of the 1–D axial model using both heterogeneity and correction factors are compared against the 3–D PANBOX model.

5.5.1 Analytical Test Case

The first suitable test of the 1–D model is to calculate a case where the solution has only one–dimensional dependence. For this case, a 3–D data base was generated with reflective boundary conditions in the radial direction, zero incoming current boundary conditions in the axial direction, and with spatially constant cross sections throughout the core. No thermalhydraulic coupling was used, and a critical condition was imposed so that the solution remained stationary in time. For the PANBOX calculation, the 3–D NEM model was first used to generate

Node index / Centre height (cm)	Group 1 nodal averaged flux			Group 2 nodal averaged flux		
	Analytical	3D NEM	1D NEM	Analytical	3D NEM	1D NEM
1 / 15.0	2.427414656	2.4303	2.4303	0.6898091435	0.6861	0.6861
2 / 45.0	6.434524059	6.4350	6.4350	1.828526974	1.8289	1.8289
3 / 75.0	10.16707516	10.1670	10.1671	2.889222383	2.8892	2.8892
4 / 105.0	13.46581459	13.4660	13.4660	3.826640129	3.8267	3.8267
5 / 135.0	16.18997574	16.1911	16.1910	4.600776672	4.6011	4.6011
6 / 165.0	18.22334099	18.2269	18.2260	5.178606987	5.1796	5.1794
7 / 195.0	19.47912407	19.4835	19.4826	5.535468102	5.5367	5.5365
8 / 225.0	19.90376091	19.9078	19.9072	5.656138897	5.6573	5.6571
9 / 255.0	19.47912407	19.4828	19.4821	5.535468102	5.5365	5.5363
10 / 285.0	18.22334099	18.2260	18.2253	5.178606987	5.1794	5.1792
11 / 315.0	16.18997574	16.1904	16.1904	4.600776672	4.6009	4.6009
12 / 345.0	13.46581459	13.4655	13.4655	3.826640129	3.8266	3.8266
13 / 375.0	10.16707516	10.1667	10.1668	2.889222383	2.8891	2.8891
14 / 405.0	6.434524059	6.4347	6.4348	1.828526974	1.8288	1.8289
15 / 435.0	2.427414656	2.4302	2.4302	0.6898091435	0.6861	0.6861

Table 5.1 Comparison of analytical, 3D–NEM and 1D–NEM solutions of a one–dimensional two group diffusion problem.

a steady–state solution and initial conditions. Several time steps were integrated with the 3–D NEM model and then the 1–D NEM model was activated. During time integration of both the 3–D NEM and the 1–D NEM models, the flux distribution does not change appreciably. Axially averaged fluxes of both calculations are compared against the analytical solution in Table 5.1.

The correction factors for the 1–D–NEM model were all calculated to be equal to 1.0, within the accuracy of the convergence criteria (here 10^{-6}). This result is exactly as expected, because for the truly one dimensional solution, the shape function has no z–dependence, and correction factors, which account for the z–dependence of the shape function, should be equal to unity.

5.5.2 Transient Initiated by a Radially Uniform Perturbation

This section compares a 3–D calculation with two separate 1–D calculations. The first 1–D calculation uses time independent correction factors, while the second uses time independent heterogeneity factors, as they are described in section 5.2.2. The initial condition of the transient is a critical PWR reactor core operating at 1000 MW thermal power with all rods out. The initial power is peaked heavily towards the top of the core. The transient is initiated at t=0.5 seconds by decreasing the boron concentration from 1050 ppm to 950 ppm at the core inlet. An infinite slug of 950 ppm boronated water is assumed to traverse the core at a steady rate until it reaches the core exit at 10.5 seconds. It is noted that this transient is not realistic, since the actual transport of the boron–diluted slug would occur at the (much faster) speed of the coolant. However, the example is a good one for demonstration and discussion of the 1–D axial kinetics model.

In the case of the 1–D calculations, the axial kinetics model was activated at time t=0.5 seconds – at the beginning of the perturbation – and not reactivated. Tables 5.2 and 5.3 show the correction and heterogeneity factors, respectively, of the two axial model calculations. These factors are based on the steady–state flux shape, since the axial model is initialized as the transient is initiated. Important to note is the fact that the factors are here not equal to unity, as was true in section 5.5.2, because the shape function now has a z–dependence. Furthest from unity are the factors of the second energy group in the node at the top of the core. Here, the shape function is strongly varying due to the control rod tips protruding into the top of the core.

Node index/	Group 1 correction factors		Group 2 correction factors	
Centre height (cm)	Left	Right	Left	Right
1 / 15.0	1.00000	1.00000	1.0000	1.0000
2 / 45.0	0.995397	1.00263	0.999137	1.00821
3 / 75.0	0.999094	1.00074	0.997651	1.00130
4 / 105.0	0.996862	1.00226	0.996047	1.00240
5 / 135.0	0.997396	1.00156	0.997278	1.00157
6 / 165.0	0.996421	1.00007	0.999602	1.00022
7 / 195.0	0.999780	1.00015	0.995955	1.00036
8 / 225.0	1.00021	0.999830	1.000	1.00001
9 / 255.0	1.00059	0.999020	1.00032	0.999380
10 / 285.0	1.00186	0.997402	1.00154	0.997545
11 / 315.0	1.00204	0.997280	1.00191	0.996726
12 / 345.0	1.00029	0.999690	1.00057	0.999041
13 / 375.0	1.00219	0.996905	1.00408	0.992661
14 / 405.0	1.00087	0.998051	1.00596	0.999892
15 / 435.0	0.999412	1.00000	0.923327	0.919940

Table 5.2 Correction factors for 1–D model of boron dilution transient.

Figure 5.1: Total Power and Axial Offset During a Boron Dilution Event

Figure 5.1 shows the total core power and the relative axial power offset as a function of time during the transient for all three calculations. The two 1–D calculations are nearly indistinguishable, indicating that both the methods of using heterogeneity and correction factors are comparable in this case. The total core power of the 3–D model calculation differs from that of the 1–D model calculation, although the qualitative prediction of the transient by the 1–D model calculations is quite good. This is most likely due to the 3–D dependence of feedback effects in the heterogeneous core, as the deviation between the results does not occur until 2.5 seconds from initiation of the transient.

The 1–D models do predict the relative axial power offset quite well. In addition, the axially averaged power shapes, shown in figure 5.2 are very similar for all three calculations. The good agreement between the axial power shapes of the 1–D and 3–D models is expected for this kind of transient, since the perturbations (excluding feedback effects) are uniform in the radial directions.results does not occur until 2.5 seconds from initiation of the transient.

Node index/ Centre height (cm)	Group 1 correction factors		Group 2 correction factors	
	Left	Right	Left	Right
1 / 15.0	1.00000	1.00000	1.00000	1.00000
2 / 45.0	1.01421	0.995830	1.009596	0.988799
3 / 75.0	1.00184	0.998547	1.004246	0.997027
4 / 105.0	1.00605	0.995280	1.007241	0.994628
5 / 135.0	1.00470	0.996421	1.004851	0.996342
6 / 165.0	1.00053	0.999703	1.000695	0.999481
7 / 195.0	1.00041	0.999666	1.000809	0.999285
8 / 225.0	0.999596	1.00035	1.00004	0.999927
9 / 255.0	0.998482	1.00166	0.999108	1.00102
10 / 285.0	0.995691	1.00465	0.996224	1.00426
11 / 315.0	0.995540	1.00505	0.995339	1.00570
12 / 345.0	0.999416	1.00063	0.998649	1.00170
13 / 375.0	0.995765	1.00643	0.991449	1.01417
14 / 405.0	0.998495	1.00483	0.992059	1.00545
15 / 435.0	1.00078	0.996705	1.107997	0.691086

Table 5.3 Heterogeneity factors for 1D model of boron dilution transient.

5.5.3 Transient Initiated by Local Perturbations

The third transient is similar to that of section 5.5.2. In this case, the initial conditions is that of a critical PWR core operating at 1000 MW power with two rod banks inserted

3–D Calculation

1D Calculation with constant correction factors

1D Calculation with constant heterogeneity factors

Figure 5.2: Axial Power Shape During a Boron Dilution Event

fully in. This is obviously also not a realistic operating condition, but serves as a good comparision with the previous transient. At time t=0.5 seconds, the rod banks begin to be steadily withdrawn, until they reach the upper stop at t=10.5 seconds. The heterogeneity and correction factors are not shown for this calculation, as they do not qualitatively differ from those of section 5.5.2. The total core power and relative axial power offset of the 3–D and 1–D calculations are shown in figure 5.3. Again, there is no large difference between the two 1–D calculations. This indicates that holding either heterogeneity or correction factors constant in time is a good approximation relative to the approximation made that the shape function remains constant in time. As was true of the transient described in section 5.2.2, the power prediction of the 1–D model in this calculation is qualitatively good compared to that of the 3–D model. It is seen, though, that the relative axial power offset and the axial power shapes (shown in Figure 5.4) calculated by the 1–D models diverge from that of the 3–D model.

Figure 5.3: Total Power and Axial Offset During a Rod Withdrawal

The larger differences between the axial power shapes of the 1D and 3–D models is expected in this case. This is because the shape functions change more strongly with the local perturbations of a bank withdrawal, as compared with the radially uniform perturbations of the transient in section 5.5.2.

3–D Calculation

1D Calculation with constant correction factors

1D Calculation with constant heterogeneity factors

Figure 5.4: Axial Power Shape During a Rod Withdrawal

5.6 Summary

In this chapter, an axial kinetics model was developed from the three–dimensional neutron multigroup diffusion equation. The distinguishing feature of this development was the separation of the three–dimensional neutron flux into a shape function and an envelope function. This parallels the derivation of the point kinetics model in chapter 4, where the model was derived by separating the flux into a shape function and an amplitude function. The NEM equations of the axial kinetics model were kept in the same form as that of the NEM equations for the 3–D model. To force equivalence between the 3–D and 1–D model solutions, correction factors were introduced. These correction factors were compared with the heterogeneity factors used for coarse mesh homogenization in most industry codes. In the transient calculations performed, no significant differences could be found between the results of the 1–D calculations using either correction factors or heterogeneity factors.

6. Adaptive Multi-Level Algorithm

The previous two chapters derived point and one–dimensional (axial) neutron kinetics models which, together with the three–dimensional NEM model in PANBOX, form the three 'levels' of the adaptive multi–level algorithm. This chapter first considers *how* the algorithm will switch from one level to another. Secondly, criteria will be derived for *when* such switching should occur. The purpose of switching to lower dimensional models is to save computation time, while the purpose of switching back to the three dimensional model is to retain or restore accuracy. Switching from one model to another essentially entails finding initial conditions for the newly activated model. These initial conditions can be found from the previously activated model. In addition to initial conditions, it is necessary at all time steps to have an approximation of the three–dimensional multigroup flux. The flux is used to determine the power distribution in the core, which is then passed as a heat generation term to the thermalhydraulic models. Because it is always necessary to have an approximation of the three dimensional flux, it is convenient to consider all level–to–level switching modes [depicted in Figure 6.1(a)] as combinations of switching between the three–dimensional kinetics model and either the axial or point kinetics models. Thus, a switch from the axial to the point kinetics models is calculated as a switch first from the axial to the three–dimensional model, and then an immediate switch from the three dimensional to the point kinetics model. The reduced set of switching modes is depicted in Figure 6.1(b).

(a) possible modes of switching   (b) reduced set of switching
                                              modes

Figure 6.1: Modes of switching for adaptive multi–level algorithm.

6.1 Mechanics of Switching Between Models

An adaptive calculation always begins from the three dimensional model. This is necessary because the respective flux shape functions for the point and axial kinetics models are determined from the three–dimensional multigroup flux. A switch to either of the lower dimen-

average flux quantities of the lower dimensional model. The γ prolongation factors are the factor changes in the z component of the net currents. During a 1–D model calculation time the Γ factors are computed by

$$\Gamma^n_{g,ij} = \frac{N^m_g(t_j)}{N^m_g(t_i)}, \quad \forall\, \Omega^n \in V_m$$

$$\Gamma^{zr,n}_{g,ij} = \frac{J^{m+}_{gzr}(t_j) + J^{m-}_{gzr}(t_j)}{J^{m+}_{gzr}(t_i) + J^{m-}_{gzr}(t_i)}, \quad \forall\, \Omega^n \in V_m \tag{6.3}$$

$$\Gamma^{zl,n}_{g,ij} = \frac{J^{m+}_{gzl}(t_j) + J^{m-}_{gzl}(t_j)}{J^{m+}_{gzl}(t_i) + J^{m-}_{gzl}(t_i)}, \quad \forall\, \Omega^n \in V_m$$

and the γ prolongation factors are

$$\gamma^{zr,n}_{g,ij} = \frac{J^{m+}_{gzr}(t_j) - J^{m-}_{gzr}(t_j)}{J^{m+}_{gzr}(t_i) - J^{m-}_{gzr}(t_i)}, \quad \forall\, \Omega^n \in V_m \tag{6.4}$$

$$\gamma^{zl,n}_{g,ij} = \frac{J^{m+}_{gzl}(t_j) - J^{m-}_{gzl}(t_j)}{J^{m+}_{gzl}(t_i) - J^{m-}_{gzl}(t_i)}, \quad \forall\, \Omega^n \in V_m \tag{6.5}$$

During a PK model calculation time, these prolongation factors are determined by,

$$\Gamma^n_{g,ij} = \Gamma^{zr,n}_{g,ij} = \Gamma^{zl,n}_{g,ij} = \gamma^{zr,n}_{g,ij} = \gamma^{zl,n}_{g,ij} = \frac{P(t_j)}{P(t_i)}, \quad \forall\, \Omega^n \in V, g \in \{1,\dots,N_G\} \tag{6.6}$$

The 3–D fluxes are prolonged with the equation,

$$\phi^n_g(t_j) = \Gamma^n_{g,ij}\phi^n_g(t_i) \tag{6.7}$$

The partial currents in the radial direction are prolonged in the same way. For Cartesian geometry, the prolongation of the partial currents in the radial direction is given by

$$j^{n\pm}_{gus}(t_j) = \Gamma^n_{g,ij}j^{n\pm}_{gus}(t_i), \quad \forall\, u \in \{x,y\}, s \in \{l,r\} \tag{6.8}$$

In the z direction, changes in the net axial currents and nodal face averaged fluxes are prolonged to the 3–D grid, resulting in the following equations for the partial currents:

$$j^{n+}_{gzs}(t_j) = \frac{1}{2}\left[\left(\Gamma^{zr,n}_{g,ij} + \gamma^{zr,n}_{g,ij}\right)j^{n+}_{gzs}(t_i) + \left(\Gamma^{zr,n}_{g,ij} - \gamma^{zr,n}_{g,ij}\right)j^{n-}_{gzs}(t_i)\right] \tag{6.9}$$

$$j^{n-}_{gzs}(t_j) = \frac{1}{2}\left[\left(\Gamma^{zr,n}_{g,ij} - \gamma^{zr,n}_{g,ij}\right)j^{n+}_{gzs}(t_i) + \left(\Gamma^{zr,n}_{g,ij} + \gamma^{zr,n}_{g,ij}\right)j^{n-}_{gzs}(t_i)\right] \quad \forall\, s \in \{r,l\} \tag{6.10}$$

Equations (6.7) to (6.10) prolong the change in time of either lower dimensional models to the average fluxes and partial currents of the 3–D NEM model. This prolongation procedure is consistent with the separation of the flux into shape and amplitude or envelope functions,

(4.3) or (5.13), as well as with the definition of the 1–D model partial currents (5.28) and (5.29). For the point kinetics model, equations (6.9) and (6.10) reduce to the form of (6.8).

In addition to prolonging the 3–D fluxes and currents, the 3–D precursor concentrations must also be prolonged. This is done in the same fashion as the nodal averaged flux. The precursor prolongation factors are defined as

$$\Pi^n_{i,pq} = \frac{C^{PK}_i(t_p)}{C^{PK}_i(t_q)},$$ (6.11)

for the point kinetics model and

$$\Pi^n_{i,pq} = \frac{C^{1D,m}_i(t_p)}{C^{1D,m}_i(t_q)}, \quad \forall\, n \in S_m$$ (6.12)

for the 1–D model. The 3–D nodal averaged precursor concentrations in the NEM discretization are prolonged by

$$c^n_i(t_q) = \Pi^n_{i,pq} c^n_i(t_p)$$ (6.13)

The prolongation of the precursors, equation (6.13), is consistent with the separation of the precursors into a shape function and an amplitude or envelope function, equations (4.4) or (5.14), respectively.

The prolongation methods described above provide an approximation to the 3–D flux solution which is based on the flux shape from the last known solution of the 3–D NEM model and the current solution of either the 1–D or point kinetics model. This approximation to the 3–D flux can be used to recalculate nodal averaged powers, to perform interpolations of fuel pin powers, or to approximate new initial conditions for the 3–D model. Additionally, in section 6.3, it will be shown how this approximate reconstructed 3–D solution is used to estimate the error made by the 1–D or point kinetics models. These errors accumulate if the lower dimensional models are activated during time periods in which the quasi–static approximation does not hold. Before this aspect is examined, criteria will be developed to determine when the quasi–static approximation of the lower dimensional models *is* satisfied. This will motivate criteria for the activation of the lower dimensional models.

6.2 Adaptivity from Higher to Lower Dimensional Models

In both derivations of the point and the axial kinetics models, the approximation is made that the time variation of the shape functions, $\psi^{PK}_g(\vec{r}, t)$ and $\psi^{1D}_g(\vec{r}, t)$, is negligible. For most transients in real postulated accident scenarios, the shape functions will very rarely be time independent. The heterogeneity of both the reactor core and disturbances to the core will cause the shape functions to vary in time during most transient periods. If a lower dimensional model is activated during a time when the shape function does vary strongly in time, then

the approximation of the 3–D flux solution found with the lower dimensional model will diverge from the flux solution which would have been calculated with the 3–D NEM kinetics model. To determine when a switch to the 1–D model is allowable, $\partial \psi_g^{1D}(\vec{r}, t)/\partial t$ is monitored during calculations of the 3–D model. From the value of $\partial \psi_g^{1D}(\vec{r}, t)/\partial t$, an estimate can be made of how quickly the flux solution from the 1–D model would accumulate error should it be activated. From this estimated rate of error production, the code can calculate the time period after which a user–given error tolerance ϵ would be exceeded by the 1–D model. If this calculated time period, T_{est}^{1D}, exceeds the user–selected *minimum* 1–D integration period, T_{min}^{1D}, then the 1–D model can be activated. Otherwise, the 3–D model must remain active. This idea is illustrated in Figure 6.2. Similar criteria will also be developed for the switch from the 1–D axial to the point kinetics model.

Figure 6.2: Illustration of switching criteria from 3D to 1D models.

6.2.1 Criterion for Activating the Axial Kinetics Model

Consider a situation in which the 1–D kinetics model has been activated for n time steps. The approximation to the 3–D flux is given by

$$\phi_g^h(\vec{r}, t_n) = \psi_g^{1D}(\vec{r}, t_0)N_g^h(z, t_n) \tag{6.14}$$

The superscript h denotes that the flux and envelope function solutions are only approximations of the solution which could have been calculated by the 3–D NEM neutron kinetics model. Had the 3–D NEM solution been calculated, it could be separated into a shape and envelope function as

$$\phi_g(\vec{r}, t_n) = \psi_g^{1D}(\vec{r}, t_n)N_g(z, t_n) \tag{6.15}$$

The space dependent error of the 1–D model is defined as

$$e_g(\vec{r}, t_n) \equiv \phi_g - \phi_g^h = \psi_g^{1D}(\vec{r}, t_n) N_g(z, t_n) - \psi_g^{1D}(\vec{r}, t_0) N_g^h(z, t_n) \qquad (6.16)$$

The approximation is made that the shape function $\psi_g^{1D}(\vec{r}, t_0)$ is close to $\psi_g^{1D}(\vec{r}, t)$ so that the axial kinetics model coefficients are accurate enough that $N_g^h \approx N_g$. Then

$$e_g(\vec{r}, t_n) \approx N_g^h(z, t_n) \left[\psi_g^{1D}(\vec{r}, t_n) - \psi_g^{1D}(\vec{r}, t_0) \right] \qquad (6.17)$$

$\psi_g^{1D}(\vec{r}, t_n)$ is expanded in the following Taylor series:

$$\psi_g^{1D}(\vec{r}, t_n) = \psi_g^{1D}(\vec{r}, t_0) + \left. \frac{\partial \psi_g^{1D}}{\partial t} \right|_{t_0} \Delta t_n + \dots \qquad (6.18)$$

With (6.18), equation (6.17) may be written as

$$e_g(\vec{r}, t_n) \approx N_g^h(z, t_n) \left. \frac{\partial \psi_g^{1D}}{\partial t} \right|_{t_0} \Delta t_n \qquad (6.19)$$

The *relative* error in the flux is therefore

$$\varepsilon_g(\vec{r}, t_n) \equiv \frac{e_g(\vec{r}, t_n)}{\phi_g(\vec{r}, t_n)} \approx \frac{1}{\psi_g^{1D}(\vec{r}, t_n)} \left. \frac{\partial \psi_g^{1D}}{\partial t} \right|_{t=t_0} \Delta t_n$$

$$\approx \left(\frac{1}{\psi_g^{1D}} \frac{\partial \psi_g^{1D}}{\partial t} \right)_{t=t_0} \left(1 - \frac{\Delta t_n}{\psi_g^{1D}} \frac{\partial \psi_g^{1D}}{\partial t} + \dots \right)_{t=t_0} \Delta t_n$$

$$\approx \left(\frac{\Delta t_n}{\psi_g^{1D}} \frac{\partial \psi_g^{1D}}{\partial t} \right)_{t=t_0} \qquad (6.20)$$

to first order in $\partial \psi_g^{1D}/\partial t$. Using equation (6.20), the rate of accumulation of global error in an *L2* norm is defined by

$$\left(\frac{\partial \varepsilon}{\partial t} \right)_G^{1D} \equiv \left[\frac{1}{V N_G} \sum_{g=1}^{N_G} \int_V \left(\frac{1}{\psi_g^{1D}} \frac{\partial \psi_g^{1D}}{\partial t} \right)^2 dV \right]^{\frac{1}{2}} \qquad (6.21)$$

and the rate of accumulation of local error in an *L1* norm is

$$\left(\frac{\partial \varepsilon}{\partial t} \right)_L^{1D} \equiv \max \left[\frac{1}{\psi_g^{1D}} \frac{\partial \psi_g^{1D}}{\partial t} \right] \qquad (6.22)$$

Let the user–selected tolerable local *L1* relative error be ϵ_L and the tolerable global *L2* relative error be ϵ_G. Definitions (6.21) and (6.22) can be used to approximate the time periods T_G^{1D} and T_L^{1D} after which the errors ϵ_G and ϵ_L, respectively, would develop. The time period

after which an error ϵ_G would develop is therefore approximated as

$$T_G^{1D} = \epsilon_G \bigg/ \left(\frac{\partial \varepsilon}{\partial t}\right)_G^{1D} \tag{6.23}$$

and the time period after which a local error ϵ_L can develop is approximated as

$$T_L^{1D} = \epsilon_L \bigg/ \left(\frac{\partial \varepsilon}{\partial t}\right)_L^{1D} \tag{6.24}$$

The smallest time period after which either of the tolerable error bounds can be exceeded is therefore

$$T_{est}^{1D} = \min(T_L^{1D}, T_G^{1D}) \tag{6.25}$$

Therefore, T_{est}^{1D} is the estimated allowable integration period for the 1–D model. Determination of T_{est}^{1D} relies on knowing $(\partial \psi_g^{1D}/\partial t)/\psi_g^{1D}$. Because the time dependence of the shape function is not known when the 1–D model is activated, $(\partial \psi_g^{1D}/\partial t)/\psi_g^{1D}$ can be tracked only during the periods when the 3–D model is activated. If $\partial \psi_g^{1D}/\partial t$ is a slowly varying function in time, the behaviour of $\partial \psi_g^{1D}/\partial t$ during a 3–D model integration period can be used to infer its behaviour during the subsequent 1–D model integration period. If

$$T_{est}^{1D} \geq T_{\min}^{1D} \tag{6.26}$$

for a number of consecutive time steps on the 3–D level, then the 1–D model can be activated and the 3–D model deactivated. An alternate statement of criterion (6.26) is, using (6.23) and (6.24),

$$\left(\frac{\partial \varepsilon}{\partial t}\right)_G^{1D} T_{\min}^{1D} \leq \epsilon_G \tag{6.27}$$

and

$$\left(\frac{\partial \varepsilon}{\partial t}\right)_L^{1D} T_{\min}^{1D} \leq \epsilon_L \tag{6.28}$$

$T_{\min}^{1D}$ is the minimum 1–D integration period, selected by the user. It should be chosen to be greater than the maximum allowable time step size. A value of $T_{\min}^{1D}$ on the order of the time step size will mean that the algorithm will switch often back and forth between the 3–D and 1–D models. Larger values of $T_{\min}^{1D}$ will reduce the frequency of transitions between models, and will tend to activate the 1–D model only as the transient approaches an asymptotic state.

The algorithm implemented in PANBOX activates the 1–D model if (6.26) is fulfilled for three consecutive checks. Then, the 1–D integration period is defined by

$$T^{1D} \equiv \min\left(T_{est}^{1D}, T_{\max}^{1D}\right) \tag{6.29}$$

$T_{\max}^{1D}$ is the user selected maximum 1–D integration period. Its selection ensures an occasion-

al return to the 3-D model. This is useful, for example when

$$\left(\frac{\partial \varepsilon}{\partial t}\right)_G^{1D}, \left(\frac{\partial \varepsilon}{\partial t}\right)_L^{1D} \approx 0 \tag{6.30}$$

If the 1-D model is activated at time $t = t_0$, then the 3-D model will be reactivated at time $t_{3D} \equiv t_0 + T^{1D}$.

6.2.2 Criteria for Activating the Point Kinetics Model

The criteria for switching to the point kinetics model are similar to those developed in the last section. In this case, the point kinetics shape function can be written in terms of the 1-D model shape and amplitude functions as,

$$\psi_g^{PK}(\vec{r}, t) = \frac{N_g(z, t)\psi_g^{1D}(\vec{r}, t)}{P(t)} \tag{6.31}$$

The 1-D/PK shape function is defined as

$$\psi_g^{1D/PK}(z, t) \equiv \frac{N_g(z, t)}{P(t)} \tag{6.32}$$

so that

$$\psi_g^{PK}(\vec{r}, t) = \psi_g^{1D/PK}(z, t)\,\psi_g^{1D}(\vec{r}, t) \tag{6.33}$$

The point kinetics analogue of equation (6.17) is

$$e_g(\vec{r}, t_n) \approx P^h(t_n)\left[\psi_g^{PK}(\vec{r}, t_n) - \psi_g^{PK}(\vec{r}, t_0)\right] \tag{6.34}$$

so that the relative error is approximated as in (6.20) by

$$\varepsilon_g(\vec{r}, t_n) \approx \frac{1}{\psi_g^{PK}(\vec{r}, t_0)}\frac{\partial \psi_g^{PK}}{\partial t}\bigg|_{t_0} \Delta t_n \tag{6.35}$$

Using equation (6.33), $\partial \psi_g^{PK}/\partial t$ may be expressed as

$$\frac{\partial \psi_g^{PK}(\vec{r}, t)}{\partial t} = \frac{\partial \psi_g^{1D/PK}(z, t)}{\partial t}\psi_g^{1D}(\vec{r}, t) + \psi_g^{1D/PK}(z, t)\frac{\partial \psi_g^{1D}(\vec{r}, t)}{\partial t} \tag{6.36}$$

Thus,

$$\varepsilon_g(\vec{r}, t_n) \approx \left[\frac{1}{\psi_g^{1D/PK}}\frac{\partial \psi_g^{1D/PK}}{\partial t} + \frac{1}{\psi_g^{1D}}\frac{\partial \psi_g^{1D}}{\partial t}\right]_{t=t_0} \Delta t_n \tag{6.37}$$

$$\lesssim \left[\frac{1}{\psi_g^{1D/PK}}\frac{\partial \psi_g^{1D/PK}}{\partial t}\bigg|_{t=t_0} + \frac{\epsilon_L}{T^{1D}}\right]\Delta t_n$$

The approximated rate of accumulation of global error in an *L2* norm is given by

$$\left(\frac{\partial \varepsilon}{\partial t}\right)_G^{PK} \equiv \left[\frac{1}{VN_G}\sum_{g=1}^{N_G}\int_V\left[\frac{1}{\psi_g^{1D/PK}}\frac{\partial \psi_g^{1D/PK}}{\partial t} + \frac{\epsilon_L}{T^{1D}}\right]^2 dV\right]^{\frac{1}{2}}$$

$$= \left[\frac{1}{L_zN_G}\sum_{g=1}^{N_G}\int_0^{L_z}\left[\frac{1}{\psi_g^{1D/PK}}\frac{\partial \psi_g^{1D/PK}}{\partial t} + \frac{\epsilon_L}{T^{1D}}\right]^2 dz\right]^{\frac{1}{2}} \tag{6.38}$$

and the rate of accumulation of local error in an *L1* norm is approximated as

$$\left(\frac{\partial \varepsilon}{\partial t}\right)_L^{PK} \equiv \max\left|\frac{1}{\psi_g^{1D/PK}}\frac{\partial \psi_g^{1D/PK}}{\partial t}\right| + \frac{\epsilon_L}{T^{1D}} \tag{6.39}$$

The smallest time period after which an error ϵ_G can develop is therefore

$$T_G^{PK} = \epsilon_G \bigg/ \left(\frac{\partial \varepsilon}{\partial t}\right)_G^{PK} \tag{6.40}$$

and the smallest time period after which a local error ϵ_L can develop is

$$T_L^{PK} = \epsilon_L \bigg/ \left(\frac{\partial \varepsilon}{\partial t}\right)_L^{PK} \tag{6.41}$$

The smallest time period in which neither one of the error bounds can be exceeded is

$$T_{est}^{PK} = \min(T_L^{PK}, T_G^{PK}) \tag{6.42}$$

Similar to the criteria presented in the last subsection, if

$$T_{est}^{PK} \geq T_{min}^{PK} \tag{6.43}$$

for three consecutive calculations of T_{est}^{PK}, then the point kinetics model can be activated and the axial kinetics model deactivated. An alternate expression of (6.43) is,

$$\left(\frac{\partial \varepsilon}{\partial t}\right)_G^{PK}T_{min}^{PK} \leq \epsilon_G \tag{6.44}$$

and

$$\left(\frac{\partial \varepsilon}{\partial t}\right)_L^{PK}T_{min}^{PK} \leq \epsilon_L \tag{6.45}$$

The point kinetics integration period is defined by

$$T^{PK} \equiv \min\left(T_{est}^{PK}, T_{max}^{PK}\right) \tag{6.46}$$

T_{max}^{PK} is the user selected maximum PK integration period. If the PK model is activated at time $t = t_1$, then the 3–D model will be reactivated at time t_{3D}, given by

$$t_{3D} = \min\left[(t_0 + T^{1D}), (t_1 + T^{PK})\right] \tag{6.47}$$

The algorithm presented here requires the axial kinetics model always to be activated before the point kinetics model is activated. This requirement, combined with criterion (6.43), are the criteria for activation of the point kinetics model.

6.2.3 Alternative Adaptive Criteria

The adaptive criteria presented in sections 6.2.1 and 6.2.2 are by no means unique. For example, different criteria were presented in reference 92. The tolerable errors in sections 6.2.1 and 6.2.2 are measured according to the norm

$$(\varepsilon)_G = \left[\frac{1}{VN_G} \sum_{g=1}^{N_G} \int_V \varepsilon_g^2(\vec{r}, t) dV\right]^{\frac{1}{2}} \tag{6.48}$$

for the global error, and according to

$$(\varepsilon)_L = \max\left[\varepsilon_g(\vec{r}, t)\right] \tag{6.49}$$

for the local error. In contrast, the measures presented previously[92] were in norms

$$(\varepsilon)_G' = \left[\frac{\sum_{g=1}^{N_G} \int_V e_g^2(\vec{r}, t) dV}{\sum_{g=1}^{N_G} \int_V \phi_g^2(\vec{r}, t) dV}\right]^{\frac{1}{2}} \tag{6.50}$$

and

$$(\varepsilon)_L' = \frac{\max\left[e_g(\vec{r}, t)\right]}{\bar{\phi}} \tag{6.51}$$

Since $\varepsilon_g = e_g/\phi_g$, these measures are not completely dissimilar. The error measures of (6.48) and (6.49) weight relative errors more strongly than absolute errors, whereas the measures (6.50) and (6.51) weight the absolute errors more strongly. The strongly weighted *relative* criteria from sections 6.2.1 and 6.2.2 were chosen because the error measures to be evaluated tend to be smoother functions of time than those of (6.50) and (6.51). Additionally, the criteria presented earlier[92] approximated that the time dependence of the 1–D shape function ψ_g^{1D} was completely negligible by the time the point kinetics model is switched on. For more generality, this approximation has not been made in subsection 6.2.2.

6.3 Adaptivity from Lower to Higher Dimensional Models

In addition to determining when the lower dimensional kinetic models should be activated, the criteria developed in section 6.2 estimate a time point t_{3D} when the 3–D kinetics model should be reactivated. The time t_{3D} is determined mainly by how rapidly the shape functions are evolving, but it can also be limited by the maximum user–requested integration periods T^{1D}_{max} and T^{PK}_{max}. Unfortunately, the time dependence of the shape function is not known during time periods when the 3–D model is deactivated. During a time period in which the lower dimensional models are calculated, the time–dependent behaviour of the shape function is inferred from the previous period in which the 3–D model was used. While this inference may be satisfactory when the neutron flux evolves asymptotically, it may not be true if the neutron cross sections are strongly perturbed during a period in which the lower dimensional models are activated.

With the use of a coupled thermalhydraulics/neutron kinetics simulator like RELAP5/PAN-BOX, such perturbations are expected during some part of most postulated accident scenarios. Changes in fuel temperature, moderator density and temperature, poison concentration, and control rod position all perturb the neutron cross sections. The effects of such perturbations must somehow be assessed, so that the 3–D model can be switched back on when the lower dimensional models become too inaccurate. This concept is illustrated in Figure 6.3.

Figure 6.3: Illustration of reason to switch from 1D to 3D models earlier than t_{3D}.

The following sections deal with estimating the error during integration of the lower dimensional models. When the error becomes too great, the 3–D kinetics model must be switched back on.

$$f_g = \sum_{i=1}^{N_I} \chi_{dg}^i \lambda_i c_i(\vec{r},t) + \frac{e^{\omega \Delta t}}{v_g \Delta t} \phi_g(\vec{r},t_0)$$ (6.55)

Errors in the precursor concentrations, e_i^C, and the neutron flux, e_g, from a previous time step will cause equation (6.53) to look like

$$-\nabla \cdot D_g \nabla \phi_g^h + \Sigma_{Rg} \phi_g^h = \sum_{g'=1}^{N_G} \Sigma_{Tgg'} \phi_{g'}^h + f_g - r_g + \sum_{i=1}^{N_I} \chi_{dg}^i \lambda_i e_i^C + \frac{e^{\omega \Delta t}}{v_g \Delta t} e_g$$ (6.56)

These errors are neglected for the following reasons:

(a) Errors in the precursor concentrations will tend to be very small compared to the precursor concentrations themselves. This is especially true if the initial conditions of the transient are at operating power, because then the initial precursor concentrations will be relatively high. Errors in the flux are transferred over into errors in the precursors, but it takes time on the order of the precursor half–life before the relative errors are noticeable. Transients beginning from low power with a subsequent power increase will be influenced much more by the errors accumulating in the precursor concentrations.

(b) The last term of equation (6.56) also tends to be small in comparison to f_g. The term becomes more important the higher the reactivity, and the smaller the time step.

(c) To consider these last two terms, the spatial dependence of the error must be known. Calculation of this spatial dependence is too expensive.

For these reasons, only the error in the time–discrete form of the diffusion equation is to be estimated, with equation (6.54). Section 6.4 presents a method for how the error accumulation in time can be approximated.

6.3.2 Mathematical Preliminaries

Recalling the notation introduced in chapter 3, let $V \subset R^3$ be an open bounded domain with a boundary $S \subset R^2$ consisting of a finite number of smooth planes. The domain V is partitioned into N nodes Ω^k, $\{\Omega^k \subset V, k \in I, 1 \leq k \leq N\}$. The set of these nodes is $\Omega = (\Omega^1,...,\Omega^N)$. The boundary of a node Ω^k is denoted by $\partial \Omega^k$, and the intersection of two nodes is given by $\Theta^{kl} \equiv \partial \Omega^k \bigcap \partial \Omega^l$. The intersection of a node with S is $\Theta^{k0} \equiv \partial \Omega^k \bigcap S$. The unit normal on Θ^{kl}, is denoted as $\hat{n}$, and points from the node with the lower index to the node with the higher index. The partition Ω is such that

1. $N < \infty$.
2. $\overline{V} = \overline{\Omega}$.
3. If $\Omega^k \neq \Omega^l$ then $\Omega^k \bigcap \Omega^l$ is empty.
4. Ω^k are Lipschitzian domains with piecewise smooth boundaries $\partial \Omega^k$.

Let the function space T be defined as

$$T \subset [H^0(P)]^{N_G} \bigcap [H^1(\Omega)]^{N_G} \mu \in T \Rightarrow D_g^l \frac{\partial u_g^l}{\partial \hat{n}}\bigg|_{\Theta^{lk}} = D_g^k \frac{\partial u_g^k}{\partial \hat{n}}\bigg|_{\Theta^{lk}} , \quad g = 1,...,N_G. \quad (6.57)$$

Here, u_g^k is the g–th component of the restriction of u to the single node Ω^k. H^0 and H^1 are the first and second Sobolev spaces with the usual norms. It is noted that Ainsworth and Oden[70,71,72] worked with the first Sobolev space $[H^1(P)]^{N_G}$ instead of T. However, it has not been shown that the solution to the multigroup diffusion equation in a heterogeneous medium belongs to $[H^1(P)]^{N_G}$: this is due to the fact that the partial derivatives of the flux are not required to be continuous across interfaces. Introduction of T does not, however, change the results of Ainsworth and Oden. This is because all of the equations in their development which deal with the continuity of $\partial u/\partial \hat{n}$ have a D_g term as a multiplier of the $\partial u/\partial \hat{n}$. Therefore, introduction of T merely extends the range of applicability of their method since $[H^1(P)]^{N_G} \subset T$.

Let $X \subset T$, $Y \subset T$ and $B : X \times Y \to R$ denote the bilinear form

$$B(u,w) \equiv \int_S \sum_{g=1}^{N_G} \left\{ \nabla w_g \cdot D_g \nabla u_g + \Sigma_{Rg} w_g u_g - \sum_{g'=1}^{N_G} \Sigma_{Tgg'} w_g u_{g'} \right\} dV \quad (6.58)$$

where $u = (u_1,...,u_{N_G})$, $w = (w_1,...,w_{N_G})$. Let $L : Y \to R$ be

$$L(w) \equiv \int_S \sum_{g=1}^{N_G} f_g w_g \, dV \quad (6.59)$$

where f_g is a function given on V.

The time–discretized form of the multigroup diffusion equation, (6.52),

$$- \nabla \cdot D_g \nabla \phi_g + \Sigma_{Rg} \phi_g = \sum_{g'=1}^{N_G} \Sigma_{Tgg'} \phi_{g'} + f_g \quad (6.60)$$

has boundary conditions

$$\frac{\partial \phi_g}{\partial \hat{n}} = 0, \quad on \ S_1 \quad (6.61)$$

$$\phi_g = 0, \quad on \ S_2 \quad (6.62)$$

and interface conditions

$$D_g^k \frac{\partial \phi_g^k}{\partial \hat{n}}\bigg|_{\Theta^{kl}} = D_g^l \frac{\partial \phi_g^l}{\partial \hat{n}}\bigg|_{\Theta^{kl}} \quad (6.63)$$

100

$$\phi_g^k\big|_{\ominus u} = \phi_g^l\big|_{\ominus u} \qquad (6.64)$$

Assuming that there exists a unique solution $\phi \in X$ of

$$B(\phi, w) = L(w), \quad \forall\ w \in Y \qquad (6.65)$$

then the solution of (6.65) is the weak solution of equations (6.60) to (6.64).

For general values of the coefficients D_g, Σ_{Rg} and $\Sigma_{Tgg'}$, no proof of existence of a unique solution of (6.65) is available. Kang and Hansen[19] give a flawed argument: they rationalize that since $B(\phi, \phi)$ is positive for subcritical systems, then $B(u, v)$ is positive definite. However, their argument contradicts the definition of positive definiteness, and thus does not prove existence of a solution. Dautray and Lions analyze the problem more rigorously.[96] They are able to show that unique solutions exist for sufficiently subcritical systems, but not for the general case. Their analysis examines the coercivity of the bilinear operator $B(u, v)$. The concept of coercivity is important not only for the proof of existence of a solution, but also for the application of the error estimation theorems developed by Ainsworth and Oden.[70,71,72]

The bilinear operator $B(u,w)$ is said to be weakly coercive on $X \times Y$ if there exists a positive constant γ such that

$$\sup_{u \in X} \frac{B(u, w)}{\|\ u\ \|_X} \geq \gamma\ \|\ w\ \|_Y, \quad \forall\ u \in X, w \in Y \qquad (6.66)$$

Here $\|\ u\ \|_X = \left[(u, u)_X\right]^{\frac{1}{2}}$ and $\|\ w\ \|_Y = \left[(w, w)_Y\right]^{\frac{1}{2}}$, where $(\cdot, \cdot)_X$ and $(\cdot, \cdot)_Y$ are inner products on X and Y. Many residual–based error estimation methods[93,94,95,70,71,72] require condition (6.66), because it can be used as a starting point to find a strict upper bound of the error. For example, consider that $\phi^h \in X$ is the approximation to ϕ. Then the error $e = \phi - \phi^h$ satisfies

$$B(e, w) = L(w) - B(\phi^h, w), \quad \forall\ w \in Y \qquad (6.67)$$

If $\xi \in Y$ is defined so that

$$(\xi, v)_Y = B(e, v), \quad \forall\ v \in Y \qquad (6.68)$$

then it can easily be shown[71] that

$$\|\ e\ \|_X \leq \frac{1}{\gamma}\ \|\ \xi\ \|_Y \qquad (6.69)$$

Inequality (6.69) shows that estimating an upper bound of $\|\ \xi\ \|_Y$ is equivalent to estimating an upper bound of $\|\ e\ \|_X$. Although the original bilinear operator $B(u,w)$ is non–symmetric,

estimation of the upper bound of $\| \xi \|_Y$ is practical, because the inner product $(\cdot, \cdot)_Y$ is a symmetric positive definite bilinear operator and the RHS of (6.68), given by (6.67), is known.

The difficulty in applying this error estimation procedure to the neutron diffusion equation lies in finding appropriate inner products $(\cdot, \cdot)_X$ and $(\cdot, \cdot)_Y$ such that the coercivity condition (6.66) is satisfied. While possible for highly subcritical cores, (6.66) is not readily fulfilled for geometries and coefficients D_g, Σ_{Rg} and $\Sigma_{Tgg'}$ that reflect real physical situations. The approach taken here is to divide $B(u,w)$ into a symmetric and non–symmetric part. The symmetric part of $B(u,w)$ becomes the norm in which the error is estimated. The asymmetric part will be treated as an additional residual source term, using a local approximation technique. With this method, the error estimate is no longer a rigorous upper bound of the true error.

6.3.3 Approximation of the Asymmetric Terms

The operator $B(u,w)$ is split into two operators,

$$B(u, w) = a(u, w) - b(u, w) \tag{6.70}$$

where

$$a(u, w) \equiv \sum_{g=1}^{N_G} a_g(u, w) \ , \tag{6.71}$$

$$a_g(u, w) \equiv \int_S \left\{ \nabla w_g \cdot D_g \nabla u_g + \Sigma_{Rg} w_g u_g \right\} dV$$

and

$$b(u, w) \equiv \int_S \sum_{g=1}^{N_G} \sum_{g'=1}^{N_G} \Sigma_{Tgg'} w_g u_{g'} dV \tag{6.72}$$

An equation similar to (6.67) is now written as

$$a(e, w) = L(w) - B(\phi^h, w) + b(e, w) \ , \quad \forall \ v \in Y \tag{6.73}$$

The strong form of (6.73) is

$$-\nabla \cdot D_g \nabla e_g + \Sigma_{Rg} e_g = f_g + \nabla \cdot D_g \nabla \phi_g^h - \Sigma_{Rg} \phi_g^h + \sum_{g'=1}^{N_G} \Sigma_{Tgg'} \left(\phi_{g'}^h + e_{g'} \right) \tag{6.74}$$

with boundary and interface conditions on e as in (6.61) to (6.64). The bilinear operator $a(u,w)$ is symmetric and positive definite. Because of this, if the right hand side of (6.74) is known, then an upper bound of $a(e,e)$ may be estimated according to the method presented by Ainsworth and Oden.[72] However, the right hand side of (6.74) can only be found if the

102

local values of the error e are first known. Direct calculation of the error e is of no practical value, since it is as expensive as solving the NEM equations of the 3–D model. In fact, if the error e was known, there would be no value in estimating $a(e,e)$, because $a(e,e)$ is simply a conveniently calculated measure of e. If e were known exactly, then *any* measure of the error could be constructed quite easily.

The method of Ainsworth and Oden[72] solves partial differential equations on a local basis using the residuals of the elliptic equations as source terms. The solutions of these partial differential equations are then used to calculate local error indicators, the sum of which is the global error estimate. The local residuals, r_g^k, are given for the neutron diffusion equation by

$$r_g^k = \left\{ f_g + \nabla \cdot D_g \nabla \phi_g^h - \Sigma_{R_g} \phi_g^h + \sum_{g'=1}^{N_G} \Sigma_{Tgg'} \phi_{g'}^h \right\}_{\Omega^k} \tag{6.75}$$

The approach taken here is to first approximate e locally, in order to find an approximation of the right hand side of (6.74). This local approximation of e is denoted as e^h. With e^h, the augmented residual is defined as

$$R_g^k \equiv r_g^k + \left\{ \sum_{g'=1}^{N_G} \Sigma_{Tgg'} e_{g'}^h \right\}_{\Omega^k} \tag{6.76}$$

so that the strong formulation of the error equation (6.74), is approximated as

$$-\nabla \cdot D_g \nabla e_g + \Sigma_{R_g} e_g \approx R_g^k \tag{6.77}$$

Given the augmented residuals, an upper bound of $a(e,e)$ may be estimated according to the procedure described in reference 72.

Determination of e^h is performed by node–wise *group rebalancing*. The procedure of group rebalancing solves for the nodal averaged flux with the approximation that either
(a) the leakage term is accurate,

$$\Xi_g^{1k} \equiv \int_{\Omega^k} \nabla \cdot D_g \nabla \phi_g dV \approx \int_{\Omega^k} \nabla \cdot D_g \nabla \phi_g^h dV \tag{6.78}$$

or (b) the leakage term is accurate relative to the flux,

$$\Xi_g^{2k} \equiv \frac{1}{\phi_g^k} \int_{\Omega^k} \nabla \cdot D_g \nabla \phi_g dV \approx \frac{1}{\left(\phi_g^k\right)^h} \int_{\Omega^k} \nabla \cdot D_g \nabla \phi_g^h dV \tag{6.79}$$

In case (a) the group rebalanced fluxes are found by solving

$$\Sigma_{Rg}\overline{\phi}_g^a = \sum_{g'=1}^{N_G} \Sigma_{Tgg'}\overline{\phi}_{g'}^a + f_g + \Xi_g^1, \quad g = 1,...,N_G \qquad (6.80)$$

for $\overline{\phi}_g^a$ in each node. In case (b) the group rebalanced fluxes are found by solving

$$\left(\Sigma_{Rg} - \Xi_g^2\right)\overline{\phi}_g^b = \sum_{g'=1}^{N_G} \Sigma_{Tgg'}\overline{\phi}_{g'}^b + f_g, \quad g = 1,...,N_G \qquad (6.81)$$

for $\overline{\phi}_g^b$ in each node. The two different methods are used because equation (6.81) tends to be accurate in fueled regions, but is very sensitive to rounding errors in unfueled regions. Equation (6.80) is less accurate in fueled regions, but does not exhibit the singular–like behaviour of (6.81) in unfueled regions. To avoid overprediction of the local errors by equation (6.81), the group rebalanced flux for any given node is taken to be the minimum of the two different solutions:

$$\overline{\phi}_g^{reb} \approx \min(\overline{\phi}_g^a, \overline{\phi}_g^b) \qquad (6.82)$$

and the approximate error is determined according to

$$e_g^h = \overline{\phi}_g^{reb} - \phi_g^h \qquad (6.83)$$

Results of equation (6.83) are used in (6.76) to determine the augmented residuals, R_{Kg}.

The group rebalancing procedure has been developed here to provide a local estimate of the error for the purposes of determining the augmented residuals. The global flux problem, given by equations (6.60) to (6.64), is reduced to a series of isolated nodal problems by approximating that the leakage is good, and that any error in the approximate solution lies in the way the flux is distributed between the different energy groups. The augmented residuals are then used in equation (6.77), and the method of Ainsworth and Oden is used to find a global error estimate of $a_g(e,e)$. In contrast to the group rebalancing procedure, estimation of $a_g(e,e)$ decouples the global problem into a series of local problems through the proper choice of a Lagrangian multiplier (see reference 72). The two methods are here combined to complement one another: the local group rebalancing finds the augmented residuals and decouples the group dependence of the global estimate; the global estimate provides an upper bound of the error, only provided that the R_g^k's are accurate.

6.3.4 Global Error Estimate

The development of the global error estimator is fully described in reference 72. Here, only the details distinct to this application are discussed. The residuals of the neutron diffusion equation are found by expanding the approximation to the nodal flux, ϕ_g^h, and the source term f_g in the NEM polynomials. These expansions are used in (6.75), which is subsequently used

in (6.76) to find a polynomial expansion for the augmented residual R_g^k. The expansion is given as

$$R_g^k(\vec{r}) = \sum_{u=x,y,z} \sum_{i=0}^{4} r_{igu} h_i\left(\frac{u}{a_u}\right) \tag{6.84}$$

It is noted that for this application, the nodal boundary residuals (defined in reference 72) are equal to zero, because the approximation of the flux ϕ_g^h exactly satisfies the interface conditions (6.63) and (6.64). This greatly simplifies the choice of boundary conditions for the local error indicator problems. The local problems are given as:[72]

(a) on nodes $\Omega^k : \partial\Omega^k \cap S_1 \neq \emptyset$, find $\tau_g^k \in H^1(\Omega^k)$ such that

$$\left.\begin{array}{ll} -\nabla^2 \tau_g^k = R_g^k & \text{in } \Omega^k \\[2mm] \dfrac{\partial \tau_g^k}{\partial \hat{n}} = 0 & \text{on } \partial\Omega^k \backslash S_1 \\[2mm] \tau_g^k = 0 & \text{on } \partial\Omega^k \cap S_1 \end{array}\right\} \tag{6.85}$$

(b) on nodes $\Omega^k : \partial\Omega^k \cap S_1 = \emptyset$, find $\tau_g^k \in H^1(\Omega^k)$ such that

$$\left.\begin{array}{ll} -\nabla^2 \tau_g^k = \left\{R_g^k - \displaystyle\sum_{u=x,y,z} r_{0gu}\right\} & \text{in } \Omega^k \\[2mm] \dfrac{\partial \tau_g^k}{\partial \hat{n}} = 0 & \text{on } \partial\Omega^k \end{array}\right\} \tag{6.86}$$

The local error indicators are given by

$$\left[\eta_g^k(\vec{p}_g)\right]^2 = \int_{\Omega^k} \left\{ \frac{1}{D_g}\vec{p}_g \cdot \vec{p}_g + \frac{1}{\Sigma_{Rg}}\left(\nabla \cdot \vec{p}_g + R_{Kg}\right)^2 \right\} dV \tag{6.87}$$

where

$$\vec{p}_g \equiv \nabla \tau_g^k \tag{6.88}$$

The global error estimate is

$$a_g(e, e) \leq \sum_{k=1}^{N} \left(\eta_g^k\right)^2 \tag{6.89}$$

The origins of equations (6.85) to (6.89) are found in reference 72. Using the polynomial expansion of equation (6.84), problems (6.85) and (6.86) have been solved analytically. The analytical solutions may be substituted into (6.87) to yield:

105

(a) on nodes $\Omega^k : \partial\Omega^k \bigcap S_1 \neq \emptyset$,

$$\left[\eta_g^k(\vec{p}_g)\right]^2 = \int_{\Omega^k}\left\{\left[\frac{1}{D_g}\vec{p}_g \cdot \vec{p}_g\right]\right\}dV$$

$$= \frac{V_k}{D_g}\sum_{u=x,y,z}\left\{\quad \frac{1}{3}r_{0gu}^2 + \frac{1}{30}r_{1gu}^2 + \frac{1}{210}r_{2gu}^2 + \frac{1}{70}r_{3gu}^2 + \frac{1}{770}r_{4gu}^2\right.$$
$$- \left(\frac{1}{6}r_{1gu} - \frac{1}{30}r_{2gu} + \frac{1}{10}r_{3gu} + \frac{1}{70}r_{4gu}\right)r_{0gu}$$
$$\left.+ \frac{3}{70}r_{1gu}r_{3gu} - \frac{1}{210}r_{2gu}r_{4gu}\right\}a_u^2 \tag{6.90}$$

and (b) on nodes $\Omega^k : \partial\Omega^k \bigcap S_1 = \emptyset$,

$$\left[\eta_g^k(\vec{p}_g)\right]^2 = \int_{\Omega^k}\left\{\left[\frac{1}{D_g}\vec{p}_g \cdot \vec{p}_g + \frac{1}{\Sigma_{Rg}}\left[\sum_{u=x,y,z}r_{0gu}\right]^2\right]\right\}dV$$

$$= \frac{V_k}{D_g}\sum_{u=x,y,z}\left\{\quad \frac{1}{30}r_{1gu}^2 + \frac{1}{210}r_{2gu}^2 + \frac{1}{70}r_{3gu}^2 + \frac{1}{770}r_{4gu}^2\right.$$
$$\left.+ \frac{3}{70}r_{1gu}r_{3gu} - \frac{1}{210}r_{2gu}r_{4gu}\right\}a_u^2 + \frac{V_K}{\Sigma_{Rg}}\left[\sum_{u=x,y,z}r_{0gu}\right]^2 \tag{6.91}$$

The relative global error estimate is defined to be

$$\varepsilon_{est} \equiv \max_{g=1,\ldots,N_G}\sqrt{\frac{1}{a_g(\phi^h,\phi^h)}\sum_{k=1}^{N}\left(\eta_g^k\right)^2} \gtrsim \sqrt{\frac{a_g(e,e)}{a_g(\phi^h,\phi^h)}} \tag{6.92}$$

When $\varepsilon_{est} \geq \epsilon_G^{LD\to3D}$, then the error produced by the lower dimensional kinetics model is too great, and the 3–D kinetics model must be reactivated. Here, $\epsilon_G^{LD\to3D}$ is the tolerable error before a switch from lower dimensional to 3–D kinetics models is necessary. $\epsilon_G^{LD\to3D}$ is distinct from ϵ_G presented in section 6.2, because of the different measures of error which are used. However, the measures do produce qualitatively similar results and $\epsilon_G^{LD\to3D} = \epsilon_G$ has been used quite successfully with this algorithm (c.f. Chapter 7).

6.4 Reactivity Based Criteria

The error estimator developed in the last section neither accounts for error accumulation in time, nor does it provide a true upper bound measure of the time–discrete error. For these reasons, further switching criteria are developed to act as an accuracy safeguard for when the performance of the error estimator is poor. These criteria are all based on the reactivity, and

are therefore treated here together. Section 6.4.1 describes how the error accumulation in time is approximated. Section 6.4.2 describes criteria based on changes in reactivity, and section 6.4.3 describes an absolute reactivity criterion.

6.4.1 Approximation of the Time Dependence of the Error

To account for the time dependence of the error, it is recognized that the error in the precursor concentrations has not yet been considered. Errors in the approximate neutron flux will propagate to the precursor concentrations, which then reappear as slowly decaying error sources in the neutron diffusion equation. This implies that the time dependence of the error in the precursor concentration, and its influence on the neutron flux, must be estimated. The simplest model for analyzing the interplay between the flux and precursor concentrations is the point kinetics equations.

The approximation is made that the relative error in the reactivity is equal to ε_{est}, the result of the global error estimator of section 3.3. Although it is obvious from section 6.3 that ε_{est} is *not* a measure of the error in the reactivity, it *is* a relative measure of the error in the flux. Since it is the error in the flux shape which gives rise to any possible error in the reactivity, use of ε_{est} is not completely unjustified. Most importantly, ε_{est} has already been calculated, so use of it spares computing time needed to calculate some other measure of the error.

During a period where either the 1–D or point kinetics models is activated, two additional systems of point kinetics equations are also integrated: each set of equations uses the calculated reactivity of the core plus or minus a factor of ε_{est}. These systems of equations are

$$\frac{dP^L}{dt} = \frac{[(1 - \varepsilon_{est})\rho - \beta]}{\Lambda} P^L(t) + \sum_{i=1}^{I} \lambda_i C_i^L(t)$$
$$\frac{dC_i^L}{dt} = \frac{\beta_i}{\Lambda} P^L(t) - \lambda_i C_i^L(t)$$

(6.93)

and

$$\frac{dP^H}{dt} = \frac{[(1 + \varepsilon_{est})\rho - \beta]}{\Lambda} P^H(t) + \sum_{i=1}^{I} \lambda_i C_i^H(t)$$
$$\frac{dC_i^H}{dt} = \frac{\beta_i}{\Lambda} P^H(t) - \lambda_i C_i^H(t)$$

(6.94)

The initial conditions of these equations are determined at time t=t_0, using

$$P^L(t_0) = P^H(t_0) = P(t_0)$$
$$C_i^L(t_0) = C_i^H(t_0) = C_i^{PK}(t_0)$$

(6.95)

where equations (4.13) and (4.14) are used to find $P(t_0)$ and $C_i^{PK}(t_0)$.

The systems of equations (6.93) and (6.94) are used to try to bracket the real value of $P(t)$ with $P^L(t)$ and $P^H(t)$. The relative error in the reactivity is approximated as ε_{est}, found by the global error estimate at each time point.

The time–accumulated error at any time t is approximated by

$$\varepsilon_{est,t} \approx \frac{1}{2P(t)} |P^H(t) - P^L(t)| \approx \frac{|P^H - P^L|}{P^H + P^L} \tag{6.96}$$

If $\varepsilon_{est,t} \geq \epsilon_G^{LD \to 3D}$, then the accumulation of the error is too great, and the 3–D model must be reactivated.

6.4.2 Criteria Based on Changes in Reactivity

Because a PWR core is heterogeneous, even uniformly distributed cross–section perturbations can give rise to a significant change in the shape function. The magnitude of a core perturbation can be measured quite well by the change in reactivity $\Delta\rho(t) \equiv \rho(t) - \rho(t_0)$. As an extra safeguard to preserve the accuracy of an adaptive calculation, the code user can specify a maximum $\Delta\rho$ during a lower kinetics period. Once the reactivity has changed by $\Delta\rho$, the 3–D model is reactivated. Because, from a reactor safety standpoint, changes in reactivity are less important when the core is subcritical than when the core is supercritical, two separate user parameters are specified: $\Delta\rho^+$ and $\Delta\rho^-$. The 3–D model is reactivated when

$$\begin{aligned} |\Delta\rho(t)| &> \Delta\rho^+, \ \forall\, \rho \geq 0 \\ |\Delta\rho(t)| &> \Delta\rho^-, \ \forall\, \rho < 0 \end{aligned} \tag{6.97}$$

6.4.3 Criteria Based on Absolute Reactivity

In addition to the above criteria, it is supposed that there is some threshold reactivity above which the user always wishes to use a 3–D model. For example, as the core approaches a prompt–supercritical state, the 3–D model should always be activated to fully capture the rapid evolution of the evolving spatial modes. The user can specify a maximum reactivity, ρ_{max}, so that when $\rho > \rho_{max}$ the 3–D model is reactivated and stays activated.

7. Example Calculations

In the following sections, several postulated accident scenarios are calculated to demonstrate the performance of the algorithm. The transients calculated with the dimensionally adaptive algorithm are compared with a calculation using only the 3D neutron kinetics model. Both the required CPU time and the accuracy of the adaptive algorithm with respect to the 3D reference are examined. The behaviour of the adaptive algorithm is also examined according to how the adaptive parameters presented in the last chapter are chosen.

The adaptive algorithm is controlled by the following parameters: ϵ_G, ϵ_L, T_{min}^{1D}, T_{max}^{1D}, T_{min}^{PK}, T_{max}^{PK}, $\epsilon_G^{LD \to 3D}$, ρ_{max}, $\Delta\rho^+$, and $\Delta\rho^-$. To reduce the number of variant adaptive algorithms to be tested, the following conventions relating the parameters have been chosen.

$$T_{min} \equiv T_{min}^{1D} = T_{min}^{PK}$$

$$T_{max} \equiv T_{max}^{1D} = T_{max}^{PK}$$

$$\epsilon_G = \epsilon_G^{LD \to 3D} = (\epsilon_L - 0.05)$$

For both the adaptive and reference calculations, the nodal averaged fluxes are stored on disk as a function of time, and these fluxes can be compared according to the following measures:

$$ERR_G = \left\{ \frac{\displaystyle\sum_{m=1}^{N}\sum_{g=1}^{N_G}\left[\left(\phi_g^m\right)^{3D} - \left(\phi_g^m\right)^{AML}\right]^2 V^m}{\displaystyle\sum_{m=1}^{N}\sum_{g=1}^{N_G}\left[\left(\phi_g^m\right)^{3D}\right]^2 V^m} \right\}^{\frac{1}{2}}$$

$$ERR_{MAX} = \frac{\max\left|\left(\phi_g^m\right)^{3D} - \left(\phi_g^m\right)^{AML}\right|}{\dfrac{1}{V_{TOT}}\displaystyle\sum_{m=1}^{N}\sum_{g=1}^{N_G}\left(\phi_g^m\right)^{3D} V^m}$$

These errors are compared against the errors predicted by the error estimator, although it is important to note that the measures of error are not the same.

7.1 Control Rod Ejection Transients

The rapid ejection of a single control assembly is a postulated accident scenario which is not characterized by strong coupling between plant and core phenomena. For this reason, a coupled program system like RELAP5/PANBOX is not necessary to calculate such transients. Indeed, the standalone PANBOX core simulator can be used to calculate such tran-

sients, and the PANBOX calculations of the NEACRP control rod ejection benchmark problem[75] recently produced good results.[77] The NEACRP benchmark problem was calculated with the RELAP5/PANBOX system to verify the data exchange between the two codes.[76] A control rod ejection problem also serves as a good test of the adaptive algorithm, because the transient is characterized by very strong local perturbations in the flux shape as the control rod is ejected out of the core. Following the ejection, the power increase in the core gives rise to a global, non–uniform increase in fuel temperature, which also perturbs the cross sections through the Doppler effect. Ideally, the algorithm should select the 3D model during the movement of the control rod, and lower dimensional models should be selected in the asymptotic period of the transient.

Calculations of cases A1 and A2 of the NEACRP control rod ejection problem are presented here. Case A1 is the ejection of a fully inserted central control assembly from a reactor core at hot zero power conditions. Case A2 is the ejection of a partially inserted assembly from a reactor core at hot full power conditions. It is known that the results of case A1 are highly sensitive to mesh size and time step size, whereas case A2 is in comparison not so sensitive.

7.1.1 Control Rod Ejection from Hot Zero Power Conditions

Because rod ejection transients require relatively little computing time, and because the strong local and global perturbations serve as a good test for the algorithm, the sensitivity of the results of case A1 to some of the adaptive algorithm parameters has been examined. The calculated results of this study are depicted in Appendix D. Here, the last calculation shown in Appendix D is described in detail. Table 7.1.1 presents the parameters chosen for this calculation.

ϵ_G	0.15
ϵ_L	0.20
T_{min}	0.1 s
T_{max}	5.0 s
ρ_{max}	$2.0
$\Delta\rho^+$	$0.1
$\Delta\rho^-$	$10.0

Table 7.1.1: Parameters for case A1 of the NEACRP control rod ejection transient.

The calculation was performed from a restart of a 100 s calculation used to generate a converged stationary condition. This restart calculation begins at 100 seconds without any perturbations; between 101.0 and 101.1 seconds, the central control assembly is ejected from the core. The calculation is continued to 111.0 seconds, when the transient is in an asymptotic

state. The total core power, maximum nodal power peaking factor, and axial offset of the adaptive calculation is compared with those of a 3D reference calculation in figure 7.1.1. It is seen that the adaptive algorithm models the transient quite well.

Figure 7.1.1: Comparison of Adaptive and Reference Calculation of HZP Rod Ejection

Figure 7.1.2 shows which models are activated at various periods of the transient. It is seen that first the 1D and then the point kinetics model is activated at the beginning, stationary

part of the calculation. The 3D model is reactivated when the rod is ejected. The algorithm switches back to the lower dimensional models once the rod is out of the core, but makes frequent returns to the 3D model until the transient reaches an asymptotic state, where the point kinetics model is activated during most time steps.

Figure 7.1.2:   Dimension of Model Activated by Adaptive Algorithm During HZP Rod Ejection Transient

Figure 7.1.3 shows the power, model dimension, errors, and error estimate of the adaptive calculation between times 101.0 seconds and 102.0 seconds. When the rod ejection begins, the adaptive algorithm is using the point kinetics model. The error estimator detects some of the error generated by the movement of the control rod during the point kinetics calculation, but does not quantitatively detect the error which develops. It is the change in reactivity criterion, $\Delta \rho^+$ which finally reactivates the 3D model. Once the 3D model is reactivated, the rate of change of the flux shape function is so great that the lower dimensional models are not reactivated until after the control rod has stopped moving. For the next 0.9 seconds, the lower dimensional models are activated with brief returns to the 3D model when the error estimator predicts that the error is too great. It is noted here that the return to the 3D model does not correct the actual errors which are developing. This is because the errors are primarily in the *amplitude* of the flux, and not in the *shape* of the flux. It is seen in the power plot of figure 7.1.3 that the adaptive algorithm predicts the power surge to occur approximately

112

0.012 seconds later than that of the reference model. It is this time discrepancy, due to the

Figure 7.1.3: Behaviour of Adaptive Algorithm During HZP Rod Ejection Transient

delayed activation of the 3D model during the rod ejection, which is the main source of the calculated error. Figure 7.1.4 shows the powers and global errors, with the results of the adaptive algorithm shifted –0.012 seconds in time. This demonstrates that the results of the adaptive algorithm are actually very close to that of the 3D calculation. If a slight delay in

the calculation of the power surge is acceptable, then the adaptive algorithm yields good results. The savings in CPU time in this case was 74.1%.

Figure 7.1.4: Global Error when the data from the adaptive calculation is shifted –0.012 seconds in time.

Table 7.1.2 presents selected results of the sensitivity study of this problem from Appendix D. In all of these cases, T_{max}=5.0 seconds and $\Delta\rho^- = \$10.0$. T_{min}, ϵ_G, $\Delta\rho^+$, and ρ^{max} are all varied. The CPU times and savings in CPU are presented in the table, along with the peak maximum calculated error after the return to power. Here, all the calculations exhibit a behaviour similar to the adaptive calculation presented above, in that most of the error is due to a shift in time of the power peak. Regardless, the maximum error ERR_{MAX} in the time period after the return to power is also shown in the table. The strong peak of error which occurs during the actual rod ejection is not considered: because it occurs at near zero

Parameter	Case 1	Case 2
ϵ_G	0.15	0.05
ϵ_L	0.20	0.15
T_{min}	0.1 s	0.1 s
T_{max}	5.0 s	5.0 s
ρ_{max}	$2.0	$2.0
$\Delta\rho^+$	$0.1	$0.1
$\Delta\rho^-$	$10.0	$10.0

Table 7.1.3: Parameters used for Case A2 of the NEACRP rod ejection transient.

Figure 7.1.5:   Behaviour of adaptive algorithm during transient initiated by a rod ejection from hot full power. ϵ_G=0.15

7.2 Main Steam Line Break at Hot Zero Power

The hot zero power main steam line break is a postulated accident scenario in which a main steam line on the secondary coolant loop breaks completely at the exit of a steam generator nozzle. This causes rapid depressurization of all four steam generators which feed the main steam mixing header. Flow through the three intact steam generators is eventually stopped through the closure of isolation valves, however the steam generator with the broken line will continue to release steam. This results in a relatively fast depressurization and asymmetric cooling of the primary coolant system, and hence the reactor core. The cooling of the reactor fuel rods gives rise to an increase in reactor reactivity through the moderator temperature coefficient and the Doppler effect. For conservative evaluations, these postulated accident scenarios are analyzed with the most reactive control rod stuck out of the core. Additional conservatism is applied by offsetting the initial reactivity of the subcritical core.

The asymmetric cooling of the core, as well as the presence of the stuck control rod, can give rise to large changes in the neutron flux shape during the MSLB event. It is for this reason that a program system like RELAP5/PANBOX is useful for analyzing this accident scenario. The greater accuracy afforded by the 3D neutron kinetics models can replace some of the conservatism used in the point kinetics calculations. For example, Feltus has shown that if the HZP MSLB is calculated with a 3D neutron kinetics code, then the 3D kinetics do not predict recriticality of the core; however, a point kinetics calculation with conservatively chosen coefficients usually does predict a recriticality. It remains a licensing question as to whether or not lifting some of the conservatism becomes an acceptable practise when a 3D kinetics code is used.

In the series of calculations presented here, the conservatism has been included so that a recriticality is predicted by the calculation of the MSLB. This means that the averaged neutron flux changes by orders of magnitude, with a significant change in the flux distribution over the course of the transient. It is thus an appropriate test for the adaptive algorithm. Table 7.2.1 shows the parameters which were used for the adaptive algorithm. Cases 1 to 3 test the algorithm with varying values of ϵ_G. T_{min} and T_{max} are set to 4.0 and 15.0 seconds respectively. These are much larger than the periods for the rod ejection case, because the transient evolves more slowly and is calculated for a longer period of simulation time (400 seconds in this case). ρ_{max} is set to 0.9, which yielded accurate results for the case of the rod ejection, but reduced potential savings in CPU. The total core power, maximum nodal power peaking factor, and axial offset of the adaptive calculation are compared with the respective values of the 3D reference calculation in figure 7.2.1.

Parameter	Case 1	Case 2	Case 3
ϵ_G	0.15	0.10	0.05
ϵ_L	0.20	0.15	0.10
T_{min}	4.0 s	4.0 s	4.0 s
T_{max}	15.0 s	15.0 s	15.0 s
ρ_{max}	$0.9	$0.9	$0.9
$\Delta\rho^+$	$10.0	$10.0	$10.0
$\Delta\rho^-$	$10.0	$10.0	$10.0

Table 7.2.1: Adaptive parameters for HZP main steam line break.

Figure 7.2.1: Adaptive vs. 3D Reference calculation for HZP MSLB.

Figures 7.2.2 and 7.3.3 shows how the algorithm switches from level to level as the value of ϵ_G is varied. The case with ϵ_G=0.15 activates the 1D and point kinetics models for longer periods than the other two cases. As ϵ_G is decreased from 0.15 to 0.10, the point kinetics model is deactivated approximately 5 seconds sooner before the initial power surge. With ϵ_G=0.05, there is a continuous switching process between the 1D and 3D kinetics models, until the flux shape function starts to evolve strongly as a function of time, beginning at

Figure 7.2.2: Model Dimensions chosen by the adaptive algorithm during the first 50 seconds of the HZP MSLB transient for different values of ϵ_G.

approximately t=6.5 seconds. After the initial power surge has occurred, and ρ_{max} drops below the $0.9 threshold, the adaptive algorithms switch back to the 1D model. The 3D models are occasionally reactivated with a frequency which decreases as ϵ_G increases, and the point kinetics model is reactivated only by the ϵ_G=0.15 calculation at about t=43 seconds. As the transient becomes asymptotic (Figure 7.2.3), the ϵ_G=0.10 and 0.05 calculations also activate

Figure 7.2.3: Model Dimensions chosen by the adaptive algorithm during the start of the asymptotic period of the HZP MSLB transient for different values of ϵ_G.

error before the return to power is not important for safety calculations, it has been included here to show how the errors increase with increasing ϵ_G for this case. In all cases, the maximum error after the return to power occurs during the first power surge, where the shift of the peak in time is the greatest source of error, as it was with the transient initiated by the control rod ejection at HZP.

CASE	Global Error Criterion	Maximum Error Criterion	Greatest Actual Global Error	Greatest Actual Maximum Error
Adaptive – Case A	0.15	0.20	18.6% 10.1%	26.2% 10.7%
Adaptive – Case B	0.10	0.15	5.9% 11.1%	7.2% 11.9%
Adaptive – Case C	0.05	0.10	1.7% 5.0%	1.1% 5.0%

Table 7.2.2   Error Criteria and Maximum Actual Errors over Transient for Adaptive Main Steam Line Break Calculation

The performance of the error estimator is similar for all three cases, so only the first case will be examined here. Figure 7.2.5 depicts the total core power, the dimension of the model which is activated, ERR$_G$, ERR$_{MAX}$, as well as the error estimate for the first 40 seconds of this case. The point kinetics model is activated for about the first 15 seconds of the simulation. At time $t_{3D}=14.76$ seconds, the 3D model is reactivated due to the end of the lower kinetics period. The error estimate at this time is 13.6%, compared with a global error of 18.6%. From this point in the transient until the power surge, the lower kinetics models are not reactivated due to the strong temporal changes in the shape function. After the power surge, the 1D model is activated. The error estimator predicts a development of the error which is qualitatively matched by the development in time of ERR$_G$ and ERR$_{MAX}$. This results in an occasional reactivation of the 3D model, occurring less frequently as the transient approaches an asymptotic state.

The total savings in CPU as well as the savings in CPU of only the neutron kinetics routines are shown in Table 7.2.3 for the three different adaptive cases. Total savings of 36.9% to 50.6% were achieved for this parameter range.

CASE	GlobalError Criterion	Total CPU time (s)	Savings in total CPU	CPU used by neutronics (s)	Savings in neutronics CPU
Reference 3D	——	2.687E+4	——	2.298E+4	——
Adaptive – Case 1	0.15	1.328E+4	50.6%	0.945E+4	58.9%
Adaptive – Case 2	0.10	1.502E+4	44.1%	1.110E+4	51.7%
Adaptive – Case 3	0.10	1.695E+4	36.9%	1.305E+4	43.2%

Table 7.2.3 Savings in CPU times for Adaptive Main Steam Line Break Calculations

7.3 Boron Dilution Transient

The event sequence of a postulated boron dilution transient has been described in section 2.2.3, where it was demonstrated that spatially dependent kinetics models are needed to accurately model this kind of event. The boron dilution transient differs from that of the MSLB transient in that in this case, the perturbations to the cross sections are to a large extent axially uniform. Thus, it is expected that the adaptive algorithm would make efficient use of the 1D kinetics model during the traversal of the deboronized slug through the core. Table 7.3.1 shows the adaptive parameters for the two adaptive cases which were calculated. In these cases, ρ_{max} is decreased with ϵ_G to select the 3–D model more often.

Parameter	Case 1	Case 2
ϵ_G	0.15	0.05
ϵ_L	0.20	0.10
T_{min}	4.0 s	4.0 s
T_{max}	15.0 s	15.0 s
ρ_{max}	$0.95	$0.9
$\Delta\rho^+$	$10.0	$10.0
$\Delta\rho^-$	$10.0	$10.0

Table 7.3.1: Adaptive parameters for boron dilution transient.

Figure 7.3.1 shows the total power, maximum nodal power peaking factor, axial offset, and core averaged boron concentration for the whole transient. The differences between the reference and adaptive calculations are difficult to see on this scale, so the same quantities during the power surge, between times t=5338.0 s and t=5345.0 s, are shown in Figure 7.3.2. The results of the adaptive cases compare quite well with the reference: as seen for both the rod ejection and the MSLB calculations, the main discrepancies are a slight shift in time. Figure 7.3.3 shows which models are selected by the adaptive algorithm during the transient. It is seen in both cases that the point kinetics model is selected for the first part of the boron dilution, with occasional returns to the 3–D model. As the reactivity of the core increases, the algorithm stops switching from the 1–D model to the PK model: this happens earlier for the ϵ_G=0.05 case than for the ϵ_G=0.15 case. During the power surge, the 3– D model is selected more and more frequently to update the shape functions. Figure 7.3.4 shows the core power, the relative error in the average flux, ERR$_G$, the error estimate, and the chosen model dimension for Case 1. It is seen that the error in the average flux follows that of ERR$_G$. This implies that the errors are not due to errors in flux shape, but rather the magnitude of the flux. Again, this is due to a small shift of the results in time. Figure 7.3.5 shows a detail of the

125

Figure 7.3.2: Detail of Adaptive vs. 3D Reference Calculation for boron dilution transient.

Figure 7.3.4: Error evolution during detail of the transient.

Figure 7.3.5: Detail of the power peak during the boron dilution transient..

power peak for the reference and two adaptive cases. If the small shifts in time can be tolerated, then the algorithm is quite accurate for these cases.

8. Conclusions

The coupled RELAP5/PANBOX code has been developed for the analysis of nuclear plant accidents in which the reactivity of the core changes significantly. With this code, accidents in which local or asymmetric positive reactivity contributions occur can now be more accurately analyzed. A reactivity edit option has also been developed to help explain transient phenomena: in particular, it can identify when changes in flux shape are important during a transient, and thus whether or not RELAP5/PANBOX is useful for the transient being calculated.

Consistent point and one–dimensional kinetics models have also been developed. The point kinetics model uses perturbation theory for accurate determination of the reactivity, and a method for using the operator formulation of perturbation theory with the NEM has been developed. The one–dimensional axial kinetics model has been developed directly from the three–dimensional neutron kinetics equations using an approach unified with the development of the point kinetics model. Correction factors have been defined so that the discretization of the axial kinetics model provides equivalent solutions to the three dimensional NEM model.

Adaptive criteria have been developed to determine during which times of a transient the various models should be activated. Criteria for switching from the three dimensional to the one dimensional and the one dimensional to the point kinetics models are based on the shape function. These criteria were motivated from the observation that both lower dimensional models employ the adiabatic quasi–static approximation. Criteria for switching back to the three dimensional model are based on a global error estimation procedure developed for finite element analysis and also on changes and absolute values of reactivity.

Calculations show that the algorithm can produce quite accurate results, while decreasing total CPU time by roughly 30% to 70%. The calculated examples were all for transients where significant three–dimensional changes do occur in the core. For calculations of more benign transients, the savings will be greater. The results of the adaptive algorithm tend to be slightly shifted in time compared to the results of the respective reference calculations, which are calculated totally with a 3D model. If these shifts in time can be tolerated, then the algorithm may be considered very accurate.

Future work should concentrate on improving the performance of the error estimator, or even the whole adaptive strategy. Development of a newly improved quasi–static method would also augment the performance and the accuracy of the algorithm: an improved quasi–static method would serve to reduce some of the error which accumulates in time in the precursor

concentrations, and would also permit larger time step sizes during time periods when the three–dimensional kinetics model is primarily selected.

References

1. V. H. Ransom et al., *"RELAP5/MOD3 Code Manual,"* NUREG/CR–5535, EGG– 2596, Rev. 4, EG&G Idaho, Inc. (Jan. 1992).

2. "TRAC–PF1/MOD1: An Advanced Best–Estimate Computer Program for Pressurized Water Reactor Thermalhydraulic Analysis," NUREG/CR–3858, LA–10157–MS, Los Alamos National Laboratoroy (1986).

3. F. Barre, M. Parent, B. Brun, "Advanced Numerical Methods for Thermalhydraulics," *Nuclear Engineering and Design,* **145**, 147–158 (1993).

4. U. Grundmann, D. Lucas, U. Rohde, "Coupling of the Thermalhydraulic Code ATHLET with the Neutron Kinetic Core Model DYN3D," *International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses,* pp. 257–263, Portland, Oregon, April 30–May 4 (1995).

5. E. K. Puska, H. Kontio, 'Three Dimensional Core in APROS Plant Analyzer,' *International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses,* pp. 264–273, Portland, Oregon, April 30–May 4, (1995).

6. R. Kyrki–Rajamäki, T. Stenins, 'Calculation of Local Boron Dilution Accidents with the HEXTRAN Code,' *International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses,* pp. 274–283, Portland, Oregon, April 30–May 4, (1995).

7. J.L. Judd, W.L. Weaver, T. Downar, H.G. Joo, 'A Three–Dimensional Nodal Neutron Kinetics Capability for RELAP5', *ANS Reactor Physics Topical Meeting,* pp. 132–240, Knoxville, April (1994).

8. A. M. Weinberg, E. P. Wigner, *The Physical Theory of Neutron Chain Reactors,* The University of Chicago Press, Chicago (1958).

9. A. F. Henry "The Application of Reactor Kinetics to the Analysis of Experiments," *Nuclear Science and Engineering,* **3**, 52–70 (1958); see also A. F. Henry, N. J. Curlee, "Verification of a Method for Treating Neutron Space–Time Problems," *Nuclear Science and Engineering,* **4**, 727–744 (1958).

10. A. F. Henry, *Nuclear Reactor Analysis,* MIT Press, Cambridge, Massachusets (1975).

11. A. F. Henry, *Naval Reactor Physics Handbook,* Vol. I, pp. 855–864, A. Radkowsky, Ed., USAEC, Washington (1964).

12. S. Kaplan, O. J. Marlowe, J. Bewick, "Application of Synthesis Techniques to Problems Involving Time Dependence," *Nuclear Science and Engineering,* **18**, 163–176 (1964).

13. J. B. Yasinsky, "Numerical Studies of Combined Space–Time Synthesis," *Nuclear Science and Engineering,* **34**, 158–168 (1968).

14. K. Ott, "Quasistatic Treatment of Spatial Phenomena in Reactor Dynamics," *Nuclear Science and Engineering,* **26**, 563 (1966).

15. K. Ott, D. A. Meneley, "Accuracy of the Quasistatic Treatment of Spatial Reactor Kinetics," *Nuclear Science and Engineering,* **36**, 402–411 (1969).

32. J. C. Gehin, A. F. Henry, "Time Dependent Discontinuity Factors for Transient Nodal Models," *Proc. Joint Int. Conf. on Math. Methods and Supercomputing in Nuclear Applications*, Vol. 1, pp. 496–506, Karlsruhe, 19–23 April (1993).

33. B. O. Cho, A. H. Robinson, "An Efficient Computational Technique for Thermal Reactor Transients," *Nuclear Science and Engineering*, 113, 264–270 (1993).

34. Y. A. Chao, P. Huang, "Theory and Performance of the Fast–Running Multidimensional Pressurized Water Reactor Kinteics Code, SPNOVA–K," *Nuclear Science and Engineering*, 103, 415–419 (1989).

35. J. Devooght, E. Mund, "Generalized Quasi–Static Method for Nuclear Reactor Space–Time Kinetics," *Nuclear Science and Engineering*, 76, 10–17 (1980).

36. J. Mika, "Mathematical Foundations of the Quasistatic Approximation in Reactor Physics," *Annals of Nuclear Energy*, 9, 585–589, (1982).

37. A. Monier, "Application of Piecewise Polynomial Collocation to the Unconstrained Quasi–Static Method for Canada Deuterium Uranium (CANDU) Reactor Space–Time Kinetics,' *Nuclear Science and Engineering*, 120, 91–101 (1995).

38. T. A. Taiwo, H. S. Khalil, 'DIF3D–K: A Nodal Kinetics Code for Solving the Time–Dependent Diffusion Equation,' *International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses*, pp. 1171–1173, Portland, Oregon, April 30–May 4, (1995).

39. R. Froehlich, "A Theoretical Foundation for Coarse Mesh Variation Techniques," GA–7870, General Atomic (1967).

40. A. Brandt, "Multi–Level Adaptive Solutions to Boundary–Value Problems," *Mathematics of Computation*, 31, No. 138, pp. 333–390 (1977); see also A. Brandt, 'Multigrid Techniques: 1984 Guide with Applications to Fluid Dynamics,' *GMD–Studien Nr. 85*, Gesellschaft für Mathematik und Datenverarbeitung, Bonn (1984).

41. H. Finnemann, R. Böer, R. Müller, Z. I. Kim, "Multi–Level Techniques for the Acceleration of Nodal Reactor Calculations," *Proc. Int. Top. Mtg. Advances in Mathematics, Computations, and Reactor Physics*, Vol. 3, pp. 11.1:1–1–11.1:1–12 Pittsburgh (1991).

42. Y. A. Chao, D. H. Risher, "Using Dynamic Frequencies for Overcoming the Stiffness of the Neutronic Space–Time Kinetics Calculations," *Transactions of the American Nuclear Society*, 68, 430 (1993).

43. N. Crouzet, P. J. Turinsky, "A Second–Derivative–Based Adaptive Time–Step Method for Spatial Kinetics Calculations," *Nuclear Science and Engineering*, 123, 206–214 (1996).

44. A. Qaddouri, R. Roy, B. Goulard, 'Multigroup Flux Solvers Using PVM,' *International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses*, pp. 1554–1562, Portland, Oregon, April 30–May 4, (1995).

45. H. G. Joo, T. J. Downar, "Incomplete Domain Decomposition Preconditioning for Coarse Mesh Neutron Diffusion Problems," *International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses*, pp. 1554–1562, Portland, Oregon, April 30–May 4, (1995).

61. P. M. Delmolino, "Development of a Generalized Adjoint Formulation for Use in an Advanced Nodal Code," Ph.D. Thesis, Department of Mechanical Engineering, University of Massachusetts Lowell, (1996).

62. K. Koebke, "Equivalent, Homogenized and Condensed Nuclear Parameters," *Transactions of the American Nuclear Society*, **31**, 248 (1979).

63. K. Koebke, 'Advances in Homogenization and Dehomogenization,' *Proc. Int. Top. Mtg. on Advances in Math. Methods for the Solution of Nuclear Engineering Problems*, Vol. 2, pp. 59–73, München, April 27–29, (1981).

64. K. S. Smith, A. F. Henry, R. A. Loretz, "The Determination of Homogenized Diffusion Theory Parameters for Coarse Mesh Nodal Analysis," *Proc. of the Conference 1980 Advances in Reactor Physics and Shielding*, Sun Valley, Idaho, Sept. 14–17 (1980).

65. W. K. Terry, D. W. Nigg, "One–Dimensional Diffusion Theory in RELAP5," *Nuclear Science and Engineering*, **120**, 110–123 (1995).

66. P. R. Engrand et. al., "Coccinelle 1D: A One–Dimensional Neutron Kinetic Code Using Time–Step Size Control," *Proceedings of the International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses*, Vol. 1, pp. 247–256, Portland, Oregon, April 30–May 4, (1995).

67. D. W. Nigg, "One–Dimensional Nodal Neutronics Routines for the TRAC–BD1 Thermal–Hydraulics Program," *EGG–PB2–6379, Rev. 1*, Idaho National Engineering Laboratory, July, (1984).

68. J. L. Muñoz–Cobo et. al., "Consistenet Generation and Functionalization of One–Dimensional Cross Sections for TRAC–BF1," *Nuclear Technology*, **107**, 125–137 (1994).

69. J. T. Cronin, K. S. Smith, "Homogenization and Functionalization of One–Dimensional Cross Sections for Retran," *Nuclear Technology*, **100**, 174–183 (1992).

70. M. Ainsworth, J. T. Oden, "A procedure for a posteriori error estimation for h–p finite element methods," *Computational Methods in Applied Mechanical Engineering*, **101**, 73–96, (1992).

71. M. Ainsworth, J. T. Oden, "A Posteriori Error Estimators for Second Order Elliptic Systems: Part 1. Theoretical Foundations and A Posteriori Error Analysis," *Comp. Math. Applic.*, **25**, 101–113, (1993).

72. M. Ainsworth, J. T. Oden, "A unified approach to a posteriori error estimation using element residual methods," *Numer. Math.*, **65**, 23–50 (1993).

73. A. Knoll, R. Müller, "Coupling RELAP5 and PANBOX 2: A Three–Dimensional Space–Time Kinetics Application with RELAP5," *International Conference on New Trends in Nuclear System Thermalhydraulics*, pp. 110–114, Pisa, Italy, May 30–June 2 (1994).

74. T. Rothe, 'EUMOD: An Interface Between RELAP5/MOD2 and External User Models.' *Proc. SCS Multi Conference, Simulators IX*, pp. 63–70, Orlando, Florida, April 6–9, (1992).

75. H. Finnemann and A. Galati, "NEACRP 3–D LWR Core Transient Benchmark, Final Specifications," *NEACRP–L–335 (Revision 1)*, October 1991 (January 1992).

76. C. J. Jackson, H. Finnemann, "Verification of the Coupled RELAP5/PANBOX2 System with the NEACRP LWR Core Transient Benchmark," *International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses*, pp. 297–306, Portland, Oregon, April 30–May 4, (1995).

77. H. Finnemann, H. Bauer, A Galati, and R. Martinelli, "Results of LWR Core Transient Benchmarks," *NEA/NSC/DOC(93)25*, (October 1993). See also H. Finnemann et. al., "Results of LWR Core Transient Benchmarks," *International Conference on Mathematical Methods and Supercomputing in Nuclear Application*, Vol. 2, pp. 243–258, Karlsruhe, April (1993).

78. Huang Aixiang, Zhang Bo, "The Convergence for Nodal Expansion Method," *Applied Mathematics: A Journal of Chinese Universities*, Vol. 8 Ser. B No. 2, 135–149, Dec. (1993).

79. R. C. Penland, Y. Y. Azmy, P. J. Turinsky, "Error Analysis of the Quartic Nodal Expansion Method for Slab Geometry," *Proceedings of the International Conference on Mathematics and Computations, Reactor Physics, and Environmental Analyses*, pp. 1418–1427, Vol. 1, Portland, Oregon, April 30–May 4, (1995).

80. H. L. Dodds, Jr., "Accuracy of the Quasistatic Method for Two–Dimensional Thermal Reactor Transients with Feedback," *Nuclear Science and Engineering*, **59**, 271–276 (1976).

81. M. Becker, "A Generalized Formulation of Point Nuclear Reactor Kinetics Equations," *Nuclear Science and Engineering*, **31**, 458–464 (1968).

82. J. J. Duderstadt, L. J. Hamilton, *Nuclear Reactor Analysis*, John Wiley & Sons, New York, (1976).

83. E. P. Wigner, "Effects of Small Perturbations on Pile Period," *Chicago Report CP–G 3048*, 1945.

84. M. R. Wagner, K. Koebke, H.–J. Winter, "A Nonlinear Extension of the Nodal Expansion Method," *Proceedings of the ANS/ENS Intl. Topical Mtg.*, v. 2, p. 43, Munich, April (1981).

85. M. L. Williams, "Perturbation Theory for Nuclear Reactor Analysis," pp. 63–188 in *CRC Handbook of Nuclear Reactor Calculations*, Yigal Ronen, Ed., CRC Press, Florida, (1986).

86. C. Jackson, H. Finnemann, "Explanation of Core Physics Phenomena During a Boron Dilution Transient," presented at *OECD/CSNI Specialist Meeting on Boron Dilution Reactivity Transients*, State College, U.S.A., October 18–20, 1995. (**No Proceedings Published.**)

87. C. Jackson, H. Finnemann, D. Cacuci, R. Böer, "Two–Level Algorithm for Efficient Space–Time Reactor Core Calculations," *Transactions of the American Nuclear Society*, **70**, 354–355, (1994).

88. C. Jackson, H. Finnemann, R. Böer, "Multilevel Methods for Coupled Space–Time Kinetics and Systems Calculations," *Proceedings of the 15th Annual Conference of the Canadian Nuclear Society*, pp. H.1–H.10, Montreal, Quebec, June 14–18 (1994).

89. C. H. Kim, J. Y. Cho, "Approximate Three–Dimensional Correction Factors in Two–Dimensional Reactor Calculations," *Nuclear Science and Engineering*, **118**, 108–121 (1994).

90. W. H. Press, S. A. Teukolsky, W. T. Veterling, B. P. Flannery, *Numerical Recipes, The Art of Scientific Computing*, Cambridge University Press, Cambridge, (1992).

91. J. Devooght, "Solutions of Quasisingular Problems with Contraints Arising in Quasistatic Approximations," *ANS Proceedings of the Topical Meeting on Computational Mehtods in Nuclear Engineering*, pp. 3.135–3.146, Williamsburg, Virginia, April 23–25, (1979).

92. C. J. Jackson, D. G. Cacuci, H. B. Finnemann, "Efficiency Enhancment of RELAP/PANBOX Using Adaptive Spatial Kinetics," *Transactions of the American Nuclear Society*, **75**, 323–325 (1996).

93. I. Babuska, W. C. Rheinboldt, "Error Estimates for Adaptive Finite Element Computations," *SIAM J. Numer. Anal.*, **15**, No. 4, 736–754 (1978).

94. M. Bietermann, I. Babuska, "The Finite Element Method for Parabolic Equations," *Numer. Math.*, **40**, 339–371 (1982).

95. R. E. Bank, A. Weiser, "Some A Posteriori Error Estimators for Elliptic Partial Differential Equations," *Math. Comp.*, **44**, No. 170, 283–301 (1985).

96. R. Dautray, J. L. Lions, eds., *Mathematical Analysis and Numerical Methods for Science and Technology, Vol. 2, Functional and Variational Methods*, Springer–Verlag, Berlin (1988).

Appendix A: The Exponential Transformation

The exponential transform method[17] is a time discretization technique in which the node averaged flux is transformed according to

$$\phi_g^m(t) = e^{\omega^m(t-t_0)}T_g^m(t) \tag{A.1}$$

The ω^m is called a dynamic frequency, and is calculated according to equation (3.47). The partial derivative of the flux with respect to time is therefore expressed as

$$\frac{d\phi_g^m}{dt} = \omega^m \phi_g^m(t) + e^{\omega^m(t-t_0)}\frac{dT_g^m(t)}{dt} \tag{A.2}$$

Equation (A.2) is discretized with the first order Euler implicit method, yielding

$$\frac{d\phi_g^m}{dt} = \omega^m \phi_g^m(t) + e^{\omega^m(t-t_0)}\frac{T_g^m(t) - T_g^m(t_0)}{\Delta t}$$

$$= \frac{(1 + \omega^m \Delta t)\phi_g^m(t) - e^{\omega \Delta t}\phi_g^m(t_0)}{\Delta t} \tag{A.3}$$

The precursor equations may be expressed in integral form as

$$c_i^m(t) = c_i^m(t_0)e^{-\lambda \Delta t} + \frac{1}{\lambda}\sum_{g'}\sum_j \beta_i^j \int_{t_0}^t \nu \Sigma_{fg'}^j \phi_{g'}^m(t')e^{\lambda_i(t'-t)}dt' \tag{A.4}$$

Substitution of the exponential transform, equation (A.1), into (A.4) yields

$$c_i^m(t) = c_i^m(t_0)e^{-\lambda \Delta t} + \frac{1}{\lambda}\sum_{g'}\sum_j \beta_i^j \int_{t_0}^t \nu \Sigma_{fg'}^j e^{\omega^m(t'-t_0)}e^{\lambda_i(t'-t)}T_{g'}(t')dt' \tag{A.5}$$

Approximation of the integral yields,

$$c_i^m(t) = c_i^m(t_0)e^{-\lambda \Delta t} + \frac{1}{\lambda}\sum_{g'}\sum_j \beta_i^j \nu \Sigma_{fg'}^j \frac{1 - e^{-(\omega^m + \lambda_i)\Delta t}}{\omega^m + \lambda_i}\phi_{g'}^m(t) \tag{A.6}$$

141

Appendix B: The NEM Polynomials

The NEM[22,23,24] polynomials are given as

$$h_0(u) = 1$$
$$h_1(u) = 2u - 1$$
$$h_2(u) = 6u(1 - u) - 1 \tag{B.1}$$
$$h_3(u) = 6u(1 - u)(2u - 1)$$
$$h_4(u) = 6u(1 - u)(5u^2 - 5u + 1)$$

These polynomials obey the following integral relationships:

$$\int_0^1 h_i(v)dv = 0 \quad i = 1,\ldots,4; \qquad \int_0^1 h_1^2(v)dv = \frac{1}{3} \qquad \int_0^1 h_2^2(v)dv = \frac{1}{5}$$

$$\int_0^1 h_3^2(v)dv = \frac{6}{35} \qquad \int_0^1 h_4^2(v)dv = \frac{2}{35} \qquad \int_0^1 h_1 h_3 dv = \frac{1}{5} \qquad \int_0^1 h_2 h_4 dv = -\frac{3}{35} \tag{B.2}$$

$$\int_0^1 h_1 h_2 dv = \int_0^1 h_1 h_4 dv = \int_0^1 h_2 h_3 dv = \int_0^1 h_3 h_4 dv = 0$$

and the following differential relationships:

$$\frac{dh_1}{du} = 2; \qquad \frac{dh_2}{du} = -6h_1; \qquad \frac{dh_3}{du} = 6h_2; \qquad \frac{dh_4}{du} = 10h_3 - 6h_1$$

$$\frac{d^2h_2}{du^2} = -12; \qquad \frac{d^2h_3}{du^2} = -36h_1; \qquad \frac{d^2h_4}{du^2} = 60h_2 - 12$$

$$\tag{B.3}$$

The transverse integrated fluxes, expanded as

$$\Psi_{gu} = \sum_{u=0}^{4} a_{igu}h_i(u) \tag{B.4}$$

therefore have a second derivative equal to

$$\frac{d^2\Psi_{gu}}{du^2} = -12(a_{2gu} + a_{4gu}) - 36a_{3gu}h_1 + 60a_{4gu}h_2 \tag{B.5}$$

143

Appendix C: Details of the Axial Kinetics Model

C.1 Coefficients of the Outgoing Partial Current Equations

The coefficients of the outgoing partial current equations are found to be

$$C_{1gR} = \frac{6D_{gz}}{DET}\left[1 + 4\frac{D_{gz}}{d_{gL}}\right]$$

$$C_{1gL} = \frac{6D_{gz}}{DET}\left[1 + 4\frac{D_{gz}}{d_{gR}}\right]$$

$$C_{2gR} = \frac{1}{DET}\left[\left(1 + 8\frac{D_{gz}}{d_{gL}}\right)\left(1 - 8\frac{D_{gz}}{d_{gR}}\right) + 16\frac{D_{gz}^2}{d_{gL}d_{gR}}\right]$$

$$C_{2gL} = \frac{1}{DET}\left[\left(1 + 8\frac{D_{gz}}{d_{gR}}\right)\left(1 - 8\frac{D_{gz}}{d_{gL}}\right) + 16\frac{D_{gz}^2}{d_{gL}d_{gR}}\right]$$

$$C_{3gR} = \frac{-8D_{gz}}{d_{gL}\ DET}$$

$$C_{3gL} = \frac{-8D_{gz}}{d_{gR}\ DET}$$

$$C_{4gR} = \frac{6D_{gz}}{DET}\left[1 + 12\frac{D_{gz}}{d_{gL}}\right]$$

$$C_{4gL} = \frac{6D_{gz}}{DET}\left[1 + 12\frac{D_{gz}}{d_{gR}}\right]$$

(C.1)

where

$$D_{gz} \equiv \frac{D_g}{a_z}$$

$$DET \equiv \left(1 + 8\frac{D_{gz}}{d_{gL}}\right)\left(1 + 8\frac{D_{gz}}{d_{gR}}\right) - 16\frac{D_{gz}^2}{d_{gL}d_{gR}}$$

(C.2)

and the node index m has been suppressed.

145

C.2 Approximations and Abbreviations of the Moments Equations

The time derivatives of the 1D flux expansions are approximated by

$$\frac{\partial \Psi_g^m}{\partial t} = \frac{1}{N_g^m} \frac{dN_g^m}{dt} \Psi_g^m \tag{C.3}$$

which, upon using the exponential transform method (see Appendix A) yields

$$\frac{1}{v_g} \frac{\partial \Psi_g^m}{\partial t} = \frac{1}{v_g \Delta t} \left[\omega \Delta t + 1 - \frac{N_g^m(t_0)}{N_g^m(t)} e^{\omega \Delta t} \right] \Psi_g^m \tag{C.4}$$

The same approximation is made for the 1D precursor concentrations

$$\frac{\partial C_i(u, t)}{\partial t} = \frac{1}{C_i^m} \frac{dC_i^m}{dt} C_i(u, t) \tag{C.5}$$

which yields

$$C_i(u, t) = C_i^m \frac{\displaystyle\sum_{g'=1}^{G} \overline{\Sigma}_{\beta g'}^i \Psi_{g'}^m}{\displaystyle\sum_{g'=1}^{G} \overline{\Sigma}_{\beta g'}^i N_{g'}^m} \tag{C.6}$$

Using these expressions in equation (5.54), the new coefficients in (5.55) and (5.56) become

$$\overline{\Sigma}_g^m \equiv \overline{\Sigma}_g^m + \frac{1 + \omega \Delta t}{v_g \Delta t} - \frac{N_g^m(t_0)}{N_g^m(t)} \frac{e^{\omega \Delta t}}{v_g \Delta t} \tag{C.7}$$

and

$$\overline{\Sigma}_{qgg'}^m = \overline{\Sigma}_{pgg'}^m - \sum_i \overline{\chi}_{dg}^i \overline{\Sigma}_{\beta g'}^i \left[\frac{\omega + \lambda_i e^{(\omega + \lambda_i) \Delta t}}{\omega + \lambda_i} \right] + \sum_i \frac{\overline{\chi}_{dg}^i \overline{\Sigma}_{\beta g'}^i \lambda_i C_i^m(t_0)}{\displaystyle\sum_{g'} \overline{\Sigma}_{\beta g'}^i N_{g'}^m(t)} \tag{C.8}$$

146

Appendix D: Parameter Study of Rod Ejection

$\rho_{\max} = 0.9$ $T_{\min} = 1.0$ $\epsilon_L = 0.10$

$\Delta\rho^{+} = 10.0$ $T_{\max} = 5.0$ $(ERR_{\max})_{max,t} = 5.4\%$

$\Delta\rho^{-} = 10.0$ $\epsilon_G = 0.05$ $CPU\ Savings = 14.9\%$

Figure D.1: Case 1 of Control Rod Ejection Transient from Hot Zero Power

Figure D.2: Case 2 of Control Rod Ejection Transient from Hot Zero Power

Figure D.3: Case 3 of the Control Rod Ejection Transient from Hot Zero Power

$\rho_{max} = 0.9$ $T_{min} = 0.1$ $\epsilon_L = 0.20$

$\Delta\rho^+ = 10.0$ $T_{max} = 5.0$ $(ERR_{max})_{max,t} > 100.0\%$

$\Delta\rho^- = 10.0$ $\epsilon_G = 0.15$ $CPU\ Savings = 37.4\%$

Figure D.4: Case 4 of the Control Rod Ejection Transient from Hot Zero Power

150

Figure D.5: Case 5 of the Control Rod Ejection Transient from Hot Zero Power

$\rho_{max} = 2.0 \qquad T_{min} = 0.1 \qquad\qquad \epsilon_L = 0.10$

$\Delta\rho^+ = 10.0 \qquad T_{max} = 5.0 \qquad (ERR_{max})_{max,t} = 19.1\%$

$\Delta\rho^- = 10.0 \qquad \epsilon_G = 0.05 \qquad CPU\ Savings = 69.6\%$

Figure D.6: Case 6 of the Control Rod Ejection Transient from Hot Zero Power

$\rho_{\max} = 2.0 \qquad T_{\min} = 1.0 \qquad\qquad \epsilon_L = 0.20$

$\Delta\rho^+ = 0.1 \qquad T_{\max} = 5.0 \qquad (ERR_{\max})_{max,t} = 20.6\%$

$\Delta\rho^- = 10.0 \qquad \epsilon_G = 0.15 \qquad CPU\ Savings = 70.7\%$

Figure D.7: Case 7 of the Control Rod Ejection Transient from Hot Zero Power

Figure D.8: Case 8 of the Control Rod Ejection Transient from Hot Zero Power

$$\rho_{max} = 2.0 \qquad T_{min} = 0.1 \qquad \epsilon_L = 0.20$$

$$\Delta\rho^+ = 0.1 \qquad T_{max} = 5.0 \qquad (ERR_{max})_{max,t} = 23.6\%$$

$$\Delta\rho^- = 10.0 \qquad \epsilon_G = 0.15 \qquad CPU\ Savings = 74.1\%$$