

## **INFORMATION TO USERS**

The most advanced technology has been used to photograph and reproduce this manuscript from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

**The quality of this reproduction is dependent upon the quality of the copy submitted.** Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.



University Microfilms International  
A Bell & Howell Information Company  
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA  
313 761-4700 800 521-0600



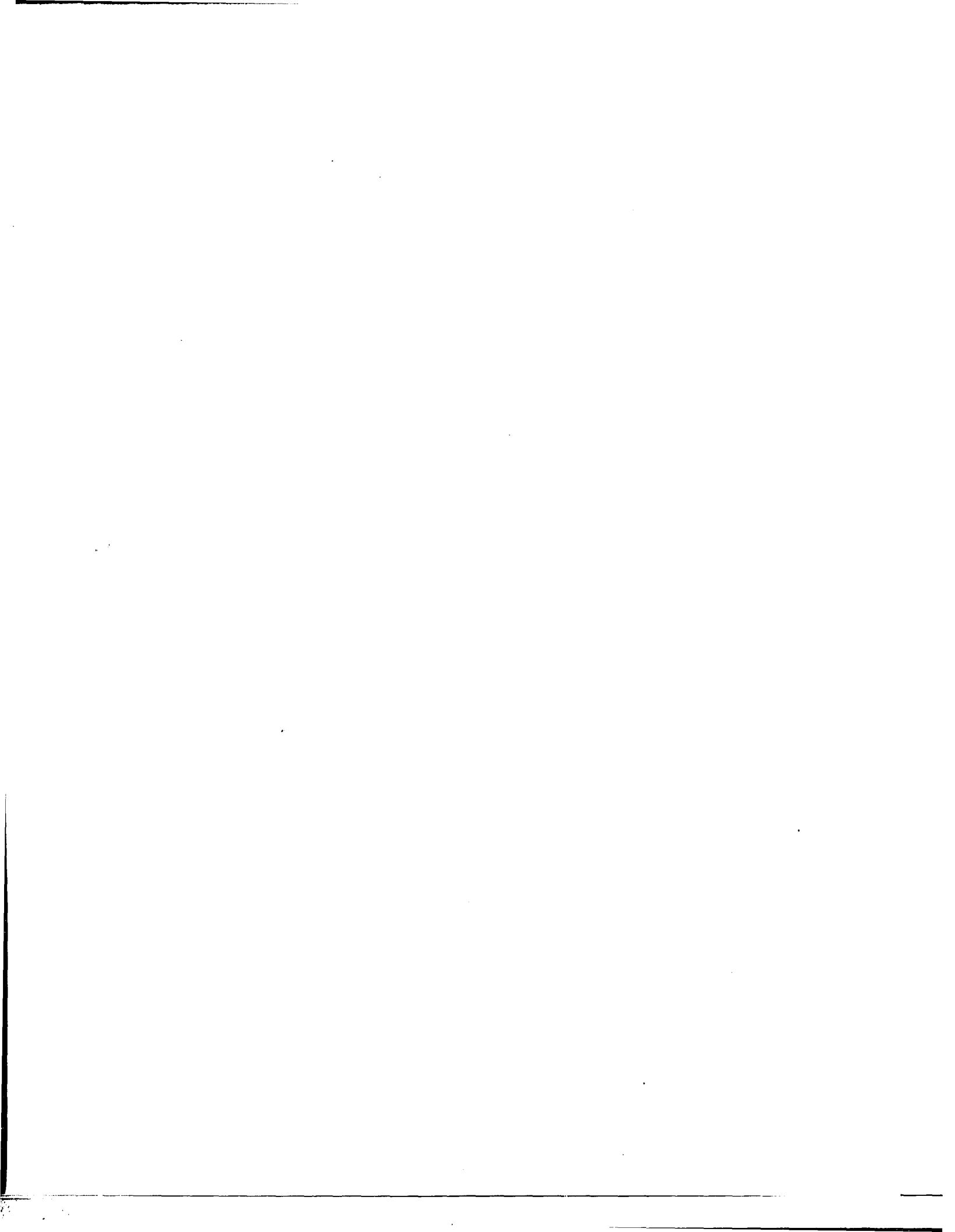
**Order Number 9032242**

**A three-dimensional transient neutronics routine for the  
TRAC-PF1 reactor thermal hydraulic computer code**

**Bandini, Bernard Ronald, Ph.D.**

**The Pennsylvania State University, 1990**

**U·M·I**  
300 N. Zeeb Rd.  
Ann Arbor, MI 48106



The Pennsylvania State University

The Graduate School

College of Engineering

**A THREE-DIMENSIONAL TRANSIENT  
NEUTRONICS ROUTINE FOR THE  
TRAC-PF1 REACTOR THERMAL  
HYDRAULIC COMPUTER CODE**

A Thesis in

**Nuclear Engineering**

by

Bernard Ronald Bandini

Submitted in Partial Fulfillment  
of the Requirements  
for the Degree of

**Doctor of Philosophy**

**May 1990**

We approve the thesis of Bernard Ronald Bandini.

Date of Signature

Anthony J. Baratta

Anthony J. Baratta  
Professor of Nuclear Engineering  
Thesis Adviser, Chair of Committee

March 19, 1990

Gordon E. Robinson

Gordon E. Robinson  
Associate Professor of Nuclear Engineering

March 19, 1990

John H. Mahaffy

John H. Mahaffy  
Research Associate, Applied Research Laboratory  
Assistant Professor of Nuclear Engineering

March 19, 1990

Nicholas J. Salamon

Nicholas J. Salamon  
Professor of Engineering  
Science and Mechanics

19 Mar 1990

Robert G. Steinke

Robert G. Steinke  
Professor, University of California  
Special Member

March 19, 1990

Edward H. Klevans

Edward H. Klevans  
Professor of Nuclear Engineering  
Head of the Department of  
Nuclear Engineering

March 19, 1990

## ABSTRACT

No present light water reactor accident analysis code employs both high state of the art neutronics and thermal-hydraulics computational algorithms. Adding a modern three-dimensional neutron kinetics model to the present TRAC-PF1/MOD2 code would create a fully up to date pressurized water reactor accident evaluation code. After reviewing several options, it was decided that the Nodal Expansion Method would best provide the basis for this multidimensional transient neutronic analysis capability. Steady-state and transient versions of the Nodal Expansion Method were coded in both three-dimensional Cartesian and cylindrical geometries. In stand-alone form this method of solving the few group neutron diffusion equations was shown to yield efficient and accurate results for a variety of steady-state and transient benchmark problems. The Nodal Expansion Method was then incorporated into TRAC-PF1/MOD2. The combined NEM/TRAC code results agreed well with the EPRI-ARROTTA core-only transient analysis code when modelling a severe PWR control rod ejection accident.

## TABLE OF CONTENTS

<b>LIST OF FIGURES . . . . .</b>	<b>viii</b>
<b>LIST OF TABLES . . . . .</b>	<b>x</b>
<b>ACKNOWLEDGMENTS . . . . .</b>	<b>xiv</b>
<b>Chapter 1. INTRODUCTION . . . . .</b>	<b>1</b>
1.1 <b>Overview . . . . .</b>	<b>1</b>
1.2 <b>Features of Present Generation Reactor Thermal-Hydraulic Codes . . . . .</b>	<b>3</b>
1.2.1 <b>RETRAN . . . . .</b>	<b>3</b>
1.2.2 <b>RELAP . . . . .</b>	<b>5</b>
1.2.3 <b>TRAC-PWR . . . . .</b>	<b>6</b>
1.2.4 <b>TRAC-BWR . . . . .</b>	<b>8</b>
1.2.5 <b>Other Codes . . . . .</b>	<b>8</b>
1.3 <b>Summary . . . . .</b>	<b>11</b>
<b>Chapter 2. TRANSIENT NEUTRONIC ANALYSIS . . . . .</b>	<b>12</b>
2.1 <b>Introduction . . . . .</b>	<b>12</b>
2.2 <b>The Point Kinetics Method . . . . .</b>	<b>14</b>
2.3 <b>Spatially Dependent Kinetics Methods . . . . .</b>	<b>15</b>
2.3.1 <b>Space-Time Factorization Techniques . . . . .</b>	<b>16</b>
2.3.2 <b>Complete Finite Difference Formulations . . . . .</b>	<b>18</b>
2.3.3 <b>Finite Element Formulations . . . . .</b>	<b>19</b>
2.3.4 <b>Nodal Methods . . . . .</b>	<b>19</b>
2.3.4.1 <b>Empirical Methods . . . . .</b>	<b>20</b>
2.3.4.2 <b>The Analytic Nodal Method . . . . .</b>	<b>21</b>
2.3.4.3 <b>Polynomial Methods . . . . .</b>	<b>23</b>
2.4 <b>Summary . . . . .</b>	<b>25</b>
<b>Chapter 3. DERIVATION OF THE STEADY-STATE NODAL EXPANSION METHOD . . . . .</b>	<b>27</b>
3.1 <b>Introduction . . . . .</b>	<b>27</b>
3.2 <b>Cartesian Geometry . . . . .</b>	<b>27</b>
3.2.1 <b>Derivation of Three-Dimensional Numerics . . . . .</b>	<b>27</b>
3.2.1.1 <b>The Nodal Balance Equation . . . . .</b>	<b>29</b>
3.2.1.2 <b>The Transverse Integration Procedure . . . . .</b>	<b>30</b>
3.2.1.3 <b>Polynomial Approximation of the One-Dimensional Equations . . . . .</b>	<b>31</b>

3.2.1.4 The Response Matrix Equations . . . . .	34
3.2.1.5 Calculation of the Transverse Leakage Moments . . . . .	37
3.2.1.6 Boundary Conditions . . . . .	40
3.2.2 Implementation and Acceleration of the Computational Algorithm . . . . .	41
3.2.2.1 Basic Overview of the Solution Procedure . . . . .	41
3.2.2.2 Solving the Response Matrix Equations . . . . .	44
3.2.2.3 Coarse-Mesh Rebalance Acceleration . . . . .	47
3.2.2.4 Asymptotic Extrapolation Acceleration . . . . .	51
3.3 Cylindrical Geometry . . . . .	53
3.3.1 Derivation of Three-Dimensional Numerics . . . . .	53
3.3.1.1 The Cylindrical Nodal Balance Equation . . . . .	54
3.3.1.2 The Transverse Integration Procedure . . . . .	55
3.3.1.3 Polynomial Approximation of the One-Dimensional Equations . . . . .	57
3.3.1.4 The Response Matrix Equations . . . . .	62
3.3.1.5 Calculation of the Transverse Leakage Moments . . . . .	65
3.3.2 Implementation of the Cylindrical Computational Algorithm . . . . .	65
3.4 Conclusions . . . . .	68
Chapter 4. BENCHMARKING OF THE STEADY-STATE NODAL EXPANSION METHOD . . . . .	69
4.1 Introduction . . . . .	69
4.2 Accuracy Requirements . . . . .	69
4.3 Cartesian Geometry . . . . .	71
4.3.1 Two-Dimensional Benchmarking . . . . .	71
4.3.1.1 Two-Dimensional IAEA Benchmark Problem . . . . .	73
4.3.1.2 Two-dimensional LRA Benchmark Problem . . . . .	81
4.3.2 Three-Dimensional Benchmarking . . . . .	88
4.3.2.1 Three-Dimensional IAEA Benchmark Problem . . . . .	89
4.3.2.2 Three-Dimensional LRA Benchmark Problem w/o Control Rod . . . . .	93
4.3.2.3 Three-Dimensional LMW Benchmark Problem . . . . .	94
4.4 Cylindrical Geometry . . . . .	100
4.4.1 Dodds' Two-Dimensional R-Z Benchmark Problem . . . . .	100
4.4.1.1 Fourth-Order Radial Expansion, First Current Derivative Approximation . . . . .	105

4.4.1.2 Fourth-Order Radial Expansion, Second Current Derivative Approximation . . . . .	111
4.4.2 Pedroni's R - $\theta$ Benchmark Problem . . . . .	116
4.5 Conclusions . . . . .	122
<b>Chapter 5. DERIVATION AND VALIDATION OF THE TRANSIENT NODAL EXPANSION METHOD . . . . .</b>	<b>123</b>
5.1 Introduction . . . . .	123
5.2 The Backwards Time-Differencing Technique . . . . .	123
5.2.1 Derivation of the Time-Dependent Numerics . . . . .	123
5.2.2 Implementation and Acceleration of the Computational Algorithm . . . . .	126
5.2.3 Two-Dimensional Benchmarking . . . . .	129
5.2.3.1 TWIGL Seed-Blanket Problem . . . . .	129
5.2.3.2 Dodds' R-Z Transient Sample Problem . . . . .	141
5.2.4 Three-Dimensional Benchmarking . . . . .	144
5.2.4.1 Three-Dimensional LMW Benchmark Problem w/o Feedback . . . . .	145
5.2.4.2 Three-Dimensional LMW Benchmark Problem w/ Thermal-Hydraulic Feedback . . . . .	148
5.3 The Exponential Time-Differencing Method . . . . .	155
5.3.1 Derivation of the Time-Dependent Numerics . . . . .	155
5.3.2 Implementation and Acceleration of the Computational Algorithm . . . . .	158
5.3.3 Benchmarking Calculations . . . . .	158
5.4 Conclusions . . . . .	161
<b>Chapter 6. INTEGRATING THE NODAL EXPANSION METHOD INTO TRAC-PF1/MOD2 . . . . .</b>	<b>164</b>
6.1 Introduction . . . . .	164
6.2 Features of the EPRI-ARROTTA Code . . . . .	165
6.3 Combined NEM/TRAC Coding . . . . .	172
6.4 The S. Levy PWR Test Problem . . . . .	176
6.4.1 Steady-State Comparisons . . . . .	177
6.4.2 Transient Comparisons . . . . .	193
6.5 Conclusions . . . . .	212
<b>Chapter 7. SUMMARY AND CONCLUSIONS . . . . .</b>	<b>215</b>

7.1	Overview of the Investigation . . . . .	215
7.2	Recommendations For Future Work . . . . .	218
REFERENCES	. . . . .	221
Appendix A.	RESPONSE MATRIX ELEMENTS . . . . .	227
Appendix B.	DESCRIPTION OF BENCHMARK PROBLEMS . . . . .	233

## LIST OF FIGURES

6.1 EPRI-ARROTTA Steady-State Outlet Coolant Temperature Rise . . . . .	183
6.2 EPRI-ARROTTA and NEM/TRAC Average Assembly Moderator Temperature as a Function of Core Height . . . . .	186
6.3 EPRI-ARROTTA and NEM/TRAC Hottest Assembly Moderator Temperature as a Function of Core Height . . . . .	187
6.4 EPRI-ARROTTA and NEM/TRAC Average Assembly Fuel Temperature as a Function of Core Height . . . . .	188
6.5 EPRI-ARROTTA and NEM/TRAC Fuel Pellet Thermal Conductivities as a Function of Temperature . . . . .	191
6.6 EPRI-ARROTTA and NEM/TRAC Hottest Assembly Fuel Temperature as a Function of Core Height . . . . .	192
6.7 EPRI-ARROTTA and NEM/TRAC Total Core Power as a Function of Time	195
6.8 EPRI-ARROTTA and NEM/TRAC Core Average Moderator Temperature as a Function of Time . . . . .	196
6.9 EPRI-ARROTTA and NEM/TRAC Average Outlet Moderator Temperature as a Function of Time . . . . .	197
6.10 EPRI-ARROTTA and NEM/TRAC Core Average Fuel Temperature as a Func- tion of Time . . . . .	198
6.11 EPRI-ARROTTA and NEM/TRAC Transient Average Fuel Temperature as a Function of Height at 0.09 Seconds . . . . .	203
6.12 EPRI-ARROTTA and NEM/TRAC Transient Assembly X=5, Y=5 Fuel Tem- perature as a Function of Height at 0.09 Seconds . . . . .	204
6.13 EPRI-ARROTTA and NEM/TRAC Transient Assembly X=5, Y=7 Fuel Tem- perature as a Function of Height at 0.09 Seconds . . . . .	205
6.14 EPRI-ARROTTA and NEM/TRAC Transient Average Fuel Temperature as a Function of Height at 0.30 Seconds . . . . .	209
6.15 EPRI-ARROTTA and NEM/TRAC Transient Assembly X=5, Y=5 Fuel Tem- perature as a Function of Height at 0.30 Seconds . . . . .	210

6.16 EPRI-ARROTTA and NEM/TRAC Transient Assembly X=5, Y=7 Fuel Temperature as a Function of Height at 0.30 Seconds . . . . .	211
6.17 EPRI-ARROTTA Transient Outlet Coolant Temperature Rise at 0.30 Seconds . . . . .	214

## LIST OF TABLES

4.1 Reference, Base-Case, and Fine Mesh Assembly Power Distributions and Percentage Errors for the Two-Dimensional IAEA Benchmark Problem . . . . .	75
4.2 Summary of Results for the 2-D IAEA Benchmark Problem . . . . .	76
4.3 Reference, Base-Case, and Lower Order Polynomial Expansion Assembly Power Distributions and Percentage Errors for the Two-Dimensional IAEA Benchmark Problem . . . . .	78
4.4 Reference, Base-Case, and Flat Leakage Assembly Power Distributions and Percentage Errors for the Two-Dimensional IAEA Benchmark Problem . . .	80
4.5 Reference, Base-Case, and Fine Mesh Assembly Power Distributions and Percentage Errors for the Two-Dimensional LRA Benchmark Problem . . . . .	83
4.6 Summary of Results for the 2-D LRA Benchmark Problem . . . . .	84
4.7 Reference, Base-Case, and Lower Order Polynomial Expansion Assembly Power Distributions and Percentage Errors for the Two-Dimensional LRA Benchmark Problem . . . . .	86
4.8 Reference, Base-Case, and Flat Leakage Assembly Power Distributions and Percentage Errors for the Two-Dimensional LRA Benchmark Problem . . .	87
4.9 Reference, Base-Case, and Alternate Assembly Power Distributions and Percentage Errors for the Three-Dimensional IAEA Benchmark Problem . . . .	90
4.10 Summary of Results for the 3-D IAEA Benchmark Problem . . . . .	92
4.11 Reference, Base-Case, and Fine Mesh Assembly Power Distributions and Percentage Errors for the Three-Dimensional LRA Benchmark Problem with Control Rod Inserted . . . . .	95
4.12 Summary of Results for the 3-D LRA Benchmark Problem with Control Rod Inserted . . . . .	96
4.13 Reference, Base-Case, and Alternate Assembly Power Distributions and Percentage Errors for the Three-Dimensional LMW Benchmark Problem . . . .	98
4.14 Summary of Results for the 3-D LMW Benchmark Problem . . . . .	99

4.15 Reference, Second-Order Fine-Mesh, and Third-Order (r dir) Coarse-Mesh Nodal Power Distributions and Percentage Errors for Dodds' Two-Dimensional Benchmark Problem . . . . .	102
4.16 Summary of Results for Dodds' R-Z Benchmark Problem . . . . .	103
4.17 Reference, Fine and Coarse Mesh Fourth-Order (1st Approx) Nodal Power Distributions and Percentage Errors for Dodds' Two-Dimensional Benchmark Problem . . . . .	109
4.18 Summary of Results for Dodds' R-Z Benchmark Problem . . . . .	110
4.19 Reference, Fine and Coarse Mesh Fourth-Order (2nd Approx) Nodal Power Distributions and Percentage Errors for Dodds' Two-Dimensional Benchmark Problem . . . . .	114
4.20 Summary of Results for Dodds' R-Z Benchmark Problem . . . . .	115
4.21 Reference, Fine and Coarse Mesh Fourth-Order (2nd Approx) Nodal Power Distributions and Percentage Errors for Pedroni's Two-Dimensional R-θ Benchmark Problem . . . . .	119
4.22 Summary of Results for Pedroni's R-θ Benchmark Problem . . . . .	120
5.1 Reference, Very Coarse, Coarse, and Fine Mesh Fourth-Order Steady-State Nodal Power Distributions and Percentage Errors for the TWIGL Seed-Blanket Two-Dimensional Transient Benchmark Problem . . . . .	131
5.2 Summary of Steady-State Results for the TWIGL Seed-Blanket Benchmark Problem . . . . .	132
5.3 Reference, Very Coarse, Coarse, and Fine Mesh Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Ramp Transient Benchmark Problem . . . . .	134
5.4 Reference, Very Coarse, Coarse, and Fine Mesh Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Step Transient Benchmark Problem . . . . .	135
5.5 Reference, Very Coarse, Coarse, and Fine Mesh Fourth-Order Asymptotic Nodal Power Distributions and Percentage Errors for the TWIGL Seed-Blanket Two-Dimensional Transient Step and Ramp Benchmark Problems . . . . .	137

5.6 Reference, 1, 10 and 50 ms Time Step Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Ramp Transient Benchmark Problem . . . . .	138
5.7 Reference, 1, 10 and 50 ms Time Step Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Step Transient Benchmark Problem . . . . .	139
5.8 Reference, Fine and Coarse Mesh Transient Total Power Vs. Time and No. of Outer Iterations for Dodds' R-Z Transient Problem with Various Time Step Sizes . . . . .	143
5.9 Reference, Coarse and Fine Axial Mesh Transient Total Power Vs. Time and No. of Outer Iterations for the 3-D LMW Transient Problem without Feedback for Various Time Step Sizes . . . . .	146
5.10 Reference and Very Fine Axial Mesh Transient Total Power Vs. Time and No. of Outer Iterations for the 3-D LMW Transient Problem without Feedback for Various Time Step Sizes . . . . .	149
5.11 Reference, Coarse and Fine Axial Mesh Transient Total Power Vs. Time and No. of Outer Iterations for the 3-D LMW Transient Problem with WIGL Thermal-Hydraulic Feedback for Various Time Step Sizes . . . . .	153
5.12 Reference, 1, 10 and 50 ms Time Step Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Ramp Transient Benchmark Problem . . . . .	160
5.13 Reference, 1, 10 and 50 ms Time Step Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Step Transient Benchmark Problem . . . . .	162
6.1 EPRI-ARROTTA and NEM/TRAC Axially Averaged Assembly Power Distributions and Percentage Differences for the Steady-State S. Levy PWR Test Problem Without Control Rods, Soluable Boron, or Thermal-Hydraulic Feedback . . . . .	179
6.2 EPRI-ARROTTA and NEM/TRAC Axially Averaged Assembly Power Distributions and Percentage Differences for the Steady-State S. Levy PWR Test Problem With Control Rods, Soluable Boron, and Thermal-Hydraulic Feedback . . . . .	181
6.3 EPRI-ARROTTA and NEM/TRAC Steady-State Coolant Temperature Rises (K) and Percentage Differences . . . . .	184

6.4 EPRI-ARROTTA and NEM/TRAC Axially Averaged Assembly Power Distributions and Percentage Differences for the S. Levy PWR Test Problem at 0.09 Seconds Into the Transient . . . . .	201
6.5 EPRI-ARROTTA and NEM/TRAC Axially Averaged Assembly Power Distributions and Percentage Differences for the S. Levy PWR Test Problem at 0.30 Seconds Into the Transient . . . . .	207
6.6 EPRI-ARROTTA and NEM/TRAC Transient Coolant Temperature Rises (K) and Percentage Differences at 0.30 Seconds . . . . .	213

## ACKNOWLEDGMENTS

The author wishes to thank his advisor, Prof. Anthony J. Baratta, for his encouragement and assistance during the course of this work.

The author would like to acknowledge the valuable discussions with Dr. Robert G. Steinke of the Los Alamos National Laboratory. The author also wishes to thank Dr. Antonio Dias of S. Levy Incorporated for providing and interpreting the EPRI-ARROTTA results for the S. Levy Benchmark Problem.

The encouragement of and valuable discussions with the author's fellow graduate students, Jeff Borkowski, Dave Knott, Kevin Smith, Joe Staudenmeier, and Mike Willis are gratefully acknowledged.

The author also wishes to thank the U. S. Department of Energy, the Penn State Nuclear Engineering Department, and the Los Alamos National Laboratory for providing funding during the course of his graduate studies.

Finally, the author wishes to thank his parents, Bernard and Mary Lou Bandini, his grandparents, Stella and the late Nato Bandini, and the rest of his family for their lasting love and support.

## CHAPTER 1

### INTRODUCTION

#### 1.1 Overview

Large commercial light-water reactors (LWR's) were introduced in the 1950's. Even before their introduction, researchers attempted to predict primary and secondary system thermal hydraulic behavior during both moderate and severe accidents. In the early years of light-water reactors, the available computer systems and computational methods required to simulate LWR system transients were limited. Therefore most predictions of transient LWR thermal-hydraulic behavior were based upon extrapolations of experimental data obtained from such facilities as the Experimental Boiling Water Reactor (1). As a result of the uncertainties involved with these predictions, early reactors were designed with large thermal-hydraulic safety margins.

In the late 1960's electric utilities, for economic reasons, began to demand reactors which produced power more economically, i.e. reactors which would extract more heat from a given amount of fissile material. The major reactor vendors, Westinghouse and General Electric, responded to these consumer demands by purchasing large up to date computer systems and developing complex numerical algorithms which would allow more accurate prediction of reactor thermal-hydraulic behavior under transient conditions. The major problem facing vendors in their quest for more economical or higher power density cores was proving to the Atomic Energy Commission (AEC), now Nuclear Regulatory Commission, that their design would have acceptable behavior during so-called design basis accidents. These hypothetical accidents include such things as the double ended break of a primary coolant line or the loss of off-site power with the resulting temporary shutdown of main coolant pumps. This AEC licensing requirement, in conjunction with the relative lack of high-speed computing power, meant that the

first vendor-developed reactor thermal-hydraulic codes tended to be fast-running codes which provided very conservative solutions to design-basis large-break loss-of-coolant accidents (large break LOCA's).

In the early 1970's, individual utilities began demanding their own means of analyzing hypothetical thermal-hydraulic transients. They desired this capability both to double check the original vendor analysis and to perform their own licensing analysis. Since the vendor's conservative thermal-hydraulic analysis codes were considered proprietary, the utilities began to develop their own methodology. When developing their codes, the utilities strived for a best-estimate capability. That is, coding which attempted to produce realistic rather than conservative results. This capability would ultimately allow utilities to run their reactors at more efficient higher power levels. The newly formed, utility-owned Electric Power Research Institute (EPRI) assisted the utilities in this development effort. After distributing several crude single-utility developed thermal-hydraulic codes such as MOXY (2), EPRI in conjunction with the Nuclear Regulatory Commission (NRC) commissioned the development of the original RELAP (3) and RETRAN (4) series of codes. In keeping with utility desires, RELAP and RETRAN were developed with best-estimate capability.

Concurrent with the development of RELAP and RETRAN, the NRC initiated a project to develop a more advanced best-estimate thermal-hydraulic code. The result of this study, performed at Los Alamos National Laboratory (LANL), was the original TRAC code for pressurized water reactor (PWR) analysis (5). TRAC achieved its high degree of accuracy by employing separate mass, momentum, and energy equations for the vapor and liquid phases in a three-dimensional model of the reactor vessel.

The TMI-2 accident in 1979 drastically altered this emphasis on studying design-basis large-break loss-of-coolant accidents. Up until this time it was believed that a

doubled-ended main-coolant-line break was the most severe credible accident that a light water reactor could undergo. TMI-2 proved that a long term small-break loss-of-coolant accident (small-break LOCA) could have consequences more severe than a large-break LOCA. The long time frame involved in analyzing small-break LOCA's meant that a large improvement in the computational efficiency of existing reactor thermal-hydraulic codes was needed.

In the years after the TMI-2 accident, the three major best-estimate reactor thermal-hydraulic codes were improved in both efficiency and accuracy. The developers of RELAP and RETRAN attained improved accuracy by respectively incorporating separate vapor and liquid field equations and a dynamic interphase slip model. The upgrading of both codes also included adding improved heuristic component models such as plenums and pressurizers. Concurrent with the improvements to RELAP and RETRAN, the TRAC code was extended to BWR analysis (6) through the addition of a jet pump model and a radiative heat transfer model.

Each of these four major non-proprietary thermal-hydraulic codes has a distinct set of advantages and disadvantages. These advantages and disadvantages and a description of equations solved and solution techniques used will be discussed in later sections of this chapter.

## 1.2 Features of Present Generation Reactor Thermal-Hydraulic Codes

### 1.2.1 RETRAN

The RETRAN thermal-hydraulic computer code was developed in the mid 1970's by Energy Incorporated (EI) of Idaho Falls, Idaho for the Electric Power Research Institute (EPRI). The initial code produced from this work was designated RETRAN-01.

The basic two-phase thermal-hydraulic model employed in RETRAN-01 was the Homogeneous Equilibrium Model (HEM). The HEM two-phase model analytically solves control-volume averaged algebraic forms of three basic fluid equations; a mixture mass conservation equation, a mixture momentum balance equation, and a mixture internal energy equation. As a result, the HEM model treats the two-phase mixture as a single compressible fluid. Both vapor and liquid phases are assumed to be at saturation temperature. In addition, both phases are assumed to be traveling at the same velocity. The boundary conditions for these cell-averaged algebraic equations are either given explicitly or are the output from or input to adjoining fluid cells. Thus the three balance equations in all connecting nodes form a system of algebraic equations which readily lend themselves to computerized numerical solution. These basic fluid equations are supplemented by a variety of specialized analytical and/or empirical models for such complex nuclear steam supply system (NSSS) components as pressurizers, pumps, steam separators, valves, etc.

To account for the time-dependent heat generation rate in the reactor core, RETRAN-01 contained transient neutronic and decay heat models. In this case transient neutronics were modelled using the point kinetics model with thermal-hydraulic feedback. The point kinetics model assumes that the three-dimensional core neutron flux profile is invariant with respect to time. This fixed flux profile is input by the user to facilitate calculations of overall time-dependent thermal-hydraulic reactivity feedback.

The use of the one-dimensional Homogeneous Equilibrium Model and the point kinetics approximation limited RETRAN-01's applicability to basically one-dimensional PWR transients, large break LOCA's, and a very limited number of BWR transients.

In order to overcome these shortcomings and make RETRAN applicable to a wider variety of transients, Energy Incorporated undertook the development of RETRAN-02

(7). The major upgrade in going from RETRAN-01 to RETRAN-02 was the introduction of the dynamic slip model. The dynamic slip model relaxes the HEM requirement of equal vapor and liquid velocities. This new model adds a fourth differential equation for the spatially dependent phase velocity difference to the three HEM mixture equations. This treatment is equivalent to using separate continuity equations for vapor and liquid. The dynamic slip model allows RETRAN to better model such physical phenomena as counter-current flow and buoyancy driven bubble rise. In addition, RETRAN-02 has the capability of solving the mixture momentum equation in more than one dimension. This so-called vector momentum capability allows RETRAN to better analyze multi-dimensional flow fields in such components as vessels and plenums.

El personnel also upgraded the point kinetics transient neutronics model of RETRAN-01 to a one-dimensional transient neutronics model in RETRAN-02. This one-dimensional transient neutronics capability is based on Ott and Meneley's finite-difference based Improved Quasi-Static Method (8). This model allows the axial neutron flux profile as well as the overall reactor power to vary during a transient. This 1-D neutronics capability allows RETRAN-02 to model thermal-hydraulic feedback and external reactivity changes which vary axially within the core.

These new models and some other more subtle improvements allow RETRAN-02 to better model a wider variety of transients. RETRAN-02 can now satisfactorily model small break LOCA's, anticipated transients without scram (ATWS's), and a substantial range of BWR transients.

### 1.2.2 RELAP

The RELAP series of thermal-hydraulic computer codes have evolved since the mid 1960's at the Idaho National Engineering Laboratory (INEL). RELAP was originally

developed at INEL under the direction of the Atomic Energy Commission (AEC) in 1966. It was originally designed for in-house analysis of thermal-hydraulic transients in several experimental PWR's at the Idaho Test Site. Like RETRAN, the initial versions of RELAP relied upon the Homogenous Equilibrium Model (HEM) to approximate general two-phase flow conditions. In addition, as in RETRAN-01, RELAP modelled reactor power changes and thermal-hydraulic feedback with the simple point kinetics model.

In the early 1980's INEL embarked upon a major upgrade of RELAP's modelling capability. In RELAP5/MOD1 the Homogenous Equilibrium Model was discarded in favor of a full two fluid, five equation model (9). Both liquid and vapor phases now had separate continuity and momentum equations, while retaining a single energy equation. The two phases could now possess independent velocities and temperatures without the use of any RETRAN-02 type two-phase slip models. The only limitation of this model is that one of the two phases must remain at saturation temperature. In the next RELAP upgrade, RELAP5/MOD2, even this minor limitation was eliminated through the introduction of an additional phasic energy equation (10). But since RELAP5 was mainly intended for use in analyzing small experimental PWR's, the developers did not feel a need to improve upon the simple point kinetics core neutronics model.

The complete six-equation two-fluid model used in RELAP5/MOD2 counts this code among the high state of the art in reactor thermal-hydraulic analysis. But the use of this code is mainly limited to the analysis of non-spatially dependent PWR transients because of its simple point kinetics transient neutronics model.

### 1.2.3 TRAC-PWR

The original TRAC thermal-hydraulic computer code (TRAC-P1) was developed in the mid 1970's at Los Alamos National Laboratory (LANL). Its purpose was to perform

advanced best-estimate thermal-hydraulic calculations for pressurized water reactors. The original TRAC used a drift flux model to account for general two-phase flow situations in one-dimensional components. This model, while superior to the HEM originally used in RETRAN and RELAP, does not account for such two-phase phenomena as counter-current flow as well as a full six equation two-fluid model. Another limitation of the original TRAC-P1 code was its use of the point kinetics transient core neutronics approximation. But TRAC did possess an advanced three-dimensional six equation reactor vessel thermal-hydraulic model. This model gave TRAC the capability to follow complex transient two-phase vessel flow with unprecedented modelling accuracy.

By 1981 LANL researchers released an upgraded version of TRAC (TRAC-PF1) (11). This version of TRAC contained a full six equation two-fluid thermal-hydraulic model in one-dimensional components very similar to that used in the vessel component. Two additional mass continuity equations were also added to TRAC at this time. These new equations were used to follow a non-condensable gas in the vapor field and a dissolved solute in the liquid field. The numerical solution method used in TRAC was also upgraded to the Stability Enhancing Two-Step (SETS) method. These improvements greatly enhanced TRAC's ability to accurately and efficiently model a variety of PWR transients such as small and large break LOCA's, ATWS's, and refloods.

The major drawback of TRAC-PF1 is the fact that it still contains only a simple point kinetics transient neutronics capability. This shortcoming, coupled with the fact that TRAC-PF1 contains no empirical jet pump or core radiation heat transport models, severely limits the use of TRAC-PF1 for BWR transient analysis.

#### 1.2.4 TRAC-BWR

In the late 1970's personnel at INEL were given the task of upgrading the TRAC-PF1 code to perform BWR analysis. The TRAC-BF1 code is the latest result of their efforts (12). This code retains the TRAC-PF1 general six equation two-phase flow methodology while adding a one-dimensional transient neutronics capability, a jet pump model, and a radiative heat transport model. The one-dimensional transient neutronics model used in TRAC-BF1 is the two-group Analytic Nodal Method developed by K. S. Smith at the Massachusetts Institute of Technology in 1979 (13). This and other transient neutronics models will be discussed in subsequent chapters.

When TRAC-BF1 was introduced it was the most up to date best-estimate thermal-hydraulic code in existence. But it is still somewhat lacking in the area of transient neutronics. In some severe transients such as single loop steam line breaks, the core coolant conditions tend to become highly asymmetrical. When reactor power has been shut down TRAC-BF1 readily solves this problem with its 3-D vessel flow capability. If core power has not been shut down, TRAC-BF1 can only evaluate a time invariant three-dimensional power shape with reactivity feedback in its neutronics model. Thus in order for TRAC-BF1 to become a truly up to date reactor thermal-hydraulic code, a three-dimensional (3-D) transient neutronics capability must be incorporated with its 3-D two-phase vessel flow model.

#### 1.2.5 Other Codes

In addition to the major non-proprietary general reactor thermal-hydraulic computer codes such as RETRAN, RELAP, and TRAC, several other more specialized codes have been developed to perform reactor thermal-hydraulic analysis. Two of the most significant of these specialized codes are the Modular Modelling System (MMS) developed by

Babcock and Wilcox Corp. (14) and RAMONA-3B originally developed by Scandpower of Norway and significantly upgraded by Brookhaven National Laboratory (BNL) (15).

MMS is a significant departure from other thermal-hydraulic codes in that it is written in an advanced computer simulation language rather than standard FORTRAN coding. This unique feature allows MMS tremendous flexibility in modelling all types of water based two-phase systems including both fossile and nuclear power plants. But in spite of this flexibility and ease of use, MMS lacks up to date two-phase thermal-hydraulic solution methodology. The original version of MMS (MMS-01) used the HEM model. Later this model was upgraded in MMS-02 to an algebraic slip model which to a first order accounted for interphase velocity differences. But this treatment can not compare to the accuracy offered by the full six equation two-phase methodology employed in RELAP5 and TRAC.

Another distinct disadvantage of MMS is its very crude transient neutronics calculational methodology. MMS uses the prompt jump approximation to account for time variations in neutron flux. This solution only approaches the point kinetics approximation at times long after the initial reactivity perturbation. At times nearer to the perturbation the prompt jump approximation gives very poor results. The MMS thermal-hydraulic time steps and thus the times when reactivity feedback is introduced are usually very short compared to the time necessary for the prompt approximation to yield reasonable results. Thus the resulting MMS reactivity feedback values can be highly inaccurate. MMS does, however, allow the option of improving the spatial accuracy of the reactivity feedback mechanism. The core can be divided into either 1, 3, or 12 predefined neutronic zones. In each of these zones an independent amount of time-dependent reactivity feedback is calculated. Even though this allows some spatial asymmetry in the calculation of feedback reactivity, the method still relies on the prompt

jump approximation. The model also requires that the user input leakage coefficients which describe the neutron transfer between adjacent nodes. These coefficients can only be properly calculated with an expensive full core steady-state neutronic analysis performed external to MMS.

The main advantage of MMS over more traditional FORTRAN-based thermal-hydraulic codes is its ease of operation. MMS can therefore be used for thermal-hydraulic scoping studies by users who have relatively little training. But in spite of its ease of operation, MMS is not an advanced two-fluid thermal-hydraulic code like TRAC or RELAP5. As a result, MMS should only be used as a scoping code, that is, one used to quickly analyze the general effects of proposed changes to a reactor thermal-hydraulic system.

The second of the major specialized thermal-hydraulic codes to be discussed is RAMONA-3B. RAMONA-3B was developed to analyze BWR system transients. The code is specialized in that it has the capability of modelling a BWR system using only a single loop. Like RETRAN-02 and MMS, RAMONA-3B uses a slip model to account for phase velocity differences in general two-phase flow. As previously stated, this approximation is somewhat better than the HEM model, but it is definitely inferior to a full two-fluid model. In addition, RAMONA-3B is only capable of modelling momentum transfer in one dimension, usually axially.

The major advantage of using RAMONA-3B is the fact that does contain a three-dimensional transient neutronics capability. The neutronic model used is a 1.5 energy group finite-difference diffusion theory approximation. This method, while not as accurate and efficient as present two-group nodal diffusion theory methods, is superior to any found in the five previously discussed thermal-hydraulic codes. This superior neutronic capability allows RAMONA-3B to model to reasonable degree of accuracy

highly asymmetric BWR accidents such as rod drop incidents. But RAMONA-3B can not approach the degree of asymmetric thermal-hydraulic calculational accuracy offered by the two TRAC codes.

### 1.3 Summary

The state of the art in best-estimate reactor thermal-hydraulic calculational methodology has advanced greatly over the past decade. Two phase flow modelling has advanced from the crude HEM model in the early RETRAN and RELAP codes to the much more accurate six equation two-phase flow model presently found in TRAC and RELAP5. In addition, transient neutronics capability has advanced from the simple prompt jump and point kinetics approximations in MMS, TRAC-PF1, and RELAP5 to the 1-D transient neutronic capability of RETRAN-02/MOD2 and TRAC-BF1 to finally the full three-dimensional transient neutronic capability of RAMONA-3B.

A logical next step would then be the integration of the superior thermal-hydraulic capability of TRAC with the three-dimensional transient neutronic capability of RAMONA-3B. This would enable one to more accurately predict certain types of severe transient behavior better than is now possible with existing codes. To accomplish this, it is proposed that an advanced three-dimensional transient neutronic capability be incorporated into the TRAC-PF1 reactor thermal-hydraulic computer code. The next chapter will discuss a variety of methods by which this advanced neutronic capability may be realized. A case will then be presented for incorporating a three-dimensional version of the Nodal Expansion Method (16) into TRAC-PF1. Chapters 3 and 5 will present, respectively, derivations of the three-dimensional steady-state and transient versions of the Nodal Expansion Method, while Chapter 6 will present the results of incorporating this coding into TRAC-PF1.

## CHAPTER 2

### TRANSIENT NEUTRONIC ANALYSIS

#### 2.1 Introduction

The equation which exactly describes the time-dependent distribution of neutrons in a medium is the Boltzmann transport equation. This equation accounts for the continuous spatial, angular, energy, and temporal distribution of neutrons in a medium. Since the Boltzmann equation can not be solved analytically except in a few grossly simplified cases, several means of reducing this equation to a numerically solvable form have been developed. In most of these simplification schemes several of the continuous variables are discretized. In the most common simplification, the discrete ordinates method, the energy, angular, and spatial variables are each separately discretized. The multi-group, time-independent discrete ordinates method has been used successfully to accurately calculate the neutron flux distribution in a wide variety of physical geometries. But in large three-dimensional geometries the discrete ordinates method with its large number of unknown variables becomes prohibitively time consuming and expensive to evaluate. The cost would be even more prohibitive if time dependence needed to be taken into account.

Therefore a simplification of the Boltzmann transport equation in which the angular dependence of the neutron flux has been integrated out has found widespread acceptance in the nuclear industry. This simplification is known as neutron diffusion theory. This method is a very satisfactory approximation to the Boltzmann transport equation in the large fissile geometries which exist in the cores of nuclear reactors. The general multi-group time-dependent neutron diffusion and precursor equations can be expressed

as:

$$\frac{1}{V_g} \frac{\partial}{\partial t} \phi_g(r, t) = \nabla^2 D_g \phi_g(r, t) - \Sigma_{rg} \phi_g(r, t) + \sum_{g'=1}^G \Sigma_{sg' \rightarrow g} \phi_{g'}(r, t) + \chi_{pg} (1 - \beta) \sum_{g'=1}^G \nu_{g'} \Sigma_{fg'} \phi_{g'}(r, t) + \chi_{dg} \sum_{d=1}^D \lambda_d C_d(r, t) \quad (2.1)$$

$$\frac{\partial}{\partial t} C_d(r, t) = \beta_d \sum_{g=1}^G \nu_g \Sigma_{fg} \phi_g(r, t) - \lambda_d C_d(r, t) \quad (2.2)$$

where

$V_g$   $\equiv$  energy group g neutron velocity

$\phi_r(r, t)$   $\equiv$  time and space dependent neutron flux in energy group g

$D_g$   $\equiv$  group g diffusion coefficient

$\Sigma_{rg}$   $\equiv$  group g removal cross-section

$\Sigma_{sg' \rightarrow g}$   $\equiv$  probability of a neutron scattering from group g' into group g

$\chi_{pg}$   $\equiv$  fraction of prompt fission neutrons entering group g

$\chi_{dg}$   $\equiv$  fraction of delayed fission neutrons entering group g

$\nu_{g'}$   $\equiv$  average number of neutrons produced by a group g' fission event

$\Sigma_{fg'}$   $\equiv$  group g' fission cross-section

$C_d(r, t)$   $\equiv$  time and space dependent group d neutron precursor concentration

$\beta_d$   $\equiv$  fraction of fission events which produce group d neutron precursors

$$\beta \equiv \sum_{d=1}^D \beta_d$$

$\lambda_d$   $\equiv$  decay constant for neutron precursor group d

A variety of means of solving these equations have been developed over the past three decades. A review of the most well known and widely used of these solution methods will be presented in the following sections of this chapter.

## 2.2 The Point Kinetics Method

In the lowest order solution to the transient neutron diffusion equations, the spatial dependence of the neutron flux is assumed to remain constant with time. This approximate solution is known as the point kinetics method. With point kinetics one needs only to solve for the time dependence of the neutron population. The actual point kinetics equations can be written as follows:

$$\frac{dn(t)}{dt} = \left( \frac{\rho(t) - \beta}{\Lambda} \right) n(t) + \sum_{i=1}^I \lambda_i C_i(t) \quad (2.3)$$

$$\frac{dC_i(t)}{dt} = \frac{\beta_i}{\Lambda} n(t) - \lambda_i C_i(t), i = 1, I \quad (2.4)$$

where

$n(t)$   $\equiv$  total time-dependent neutron population

$\rho(t) = \frac{k(t) - 1}{k(t)}$   $\equiv$  reactivity

$\beta$   $\equiv$  total delayed neutron fraction

$\Lambda$   $\equiv$  mean neutron generation time

$\lambda_i$   $\equiv$  decay constant of neutron precursor group  $i$

$C_i(t)$   $\equiv$  concentration of neutron precursor group  $i$

These coupled time-dependent equations can be derived from the time-dependent diffusion equations if one assumes a constant spatial distribution for the neutron flux. The two new parameter factors in the point kinetics equations, reactivity ( $\rho$ ), and the mean generation time ( $\Lambda$ ) can be defined as time-dependent flux shape weighted spatial integrals of various macroscopic reaction cross sections.

The point kinetics approximation is known to yield accurate results when used to calculate the time-dependent neutron flux in small tightly coupled systems. But

the method sacrifices much of its accuracy when used in large loosely coupled systems such as that which exists in large light water reactors. The inaccuracy becomes even greater if the LWR core is subjected to highly asymmetric reactivity excursions such as single steam line breaks or control rod drop incidents. Therefore, for a large number of transients, one must account for the changing spatial distribution of the neutron flux in order to obtain an accurate kinetics solution.

### 2.3 Spatially Dependent Kinetics Methods

In order to perform space-time kinetics one must develop some inexpensive means of calculating the time-independent flux distribution in a nuclear reactor. A typical PWR core consists of approximately 40,000 to 50,000, 12 foot tall, 1 cm diameter zircalloy clad UO<sub>2</sub> fuel pins. These fuel pins are in turn surrounded by a borated pressurized light water moderator/coolant. As one can imagine, calculating the detailed energy dependent neutron flux distribution in this large, complex, highly heterogeneous geometry would be an almost insurmountable task. Fortunately many techniques have been developed in the last few decades to greatly simplify this problem. Most of these techniques involve methods of generating equivalent few-group (usually two) diffusion theory parameters which represent regions in the reactor core that are larger than a single fuel pin (usually an entire fuel assembly). The use of this relatively small number of diffusion theory parameters greatly simplifies the core-wide diffusion theory flux calculation. Since assembly homogenization is a widely accepted technique in fuel management and transient analysis, all space-time kinetics methods developed in this chapter will assume the availability of homogenized two-group diffusion theory parameters for arbitrarily large core regions.

### 2.3.1 Space-Time Factorization Techniques

As previously stated, the point kinetics approximation requires that the general space-time kinetics solution be separable into independent functions of space and time. The point kinetics approximation also requires that the space dependent function remain constant throughout the transient of interest. Two other approximate space-time kinetics solution techniques have been derived in which the space and time dependences are assumed separable. But in these methods the spatial dependency of the flux is allowed to change with time. The least complex of these two space-time kinetics approximations is known as the adiabatic method (17).

In the adiabatic method, the fluxes' spatial dependency is calculated by a standard time-independent diffusion theory eigenvalue calculation. These eigenvalue calculations are only performed when significant changes in core composition are known to have occurred. These changes may include fuel movement, control rod movement, or significant variations in the coolant void distribution. The flux shapes obtained in these static calculations are first multiplied by a specified diffusion theory parameter, then weighted by the initial steady-state adjoint flux solution, then finally integrated over the core volume in order to obtain the time-dependent point kinetics parameters. These point kinetics parameters are then used to determine the amplitude of the neutron flux. The spatial dependence of the general space-time dependent neutron flux is assumed to change much more slowly than its amplitude. Therefore the time step for numerically solving the point kinetics amplitude equation is usually much smaller than the time between space function updates.

The major approximation in the adiabatic method is the assumption that the reactor core neutron population instantly adjusts to any change in core reactivity. That is, the effects of spatially dependent reactivity perturbations are felt instantaneously throughout

the reactor. This approximation is the result of ignoring terms which contain the time derivatives of both the amplitude and space functions in the exact equation for the spatial dependence of the flux. As a result of these approximations, the adiabatic method only works well for transients in which the spatial dependence of the flux changes very slowly with time. Thus the adiabatic method would work poorly when modelling severe spatially dependent transients.

The more exact method of separable space-time kinetics is known as the quasi-static method (18). As in the adiabatic method, it is assumed that the general space-time dependent neutron diffusion equation can be broken up into separable functions of space and time. Again, the time dependence of the flux can be expressed in terms of the standard point kinetics equations. But now the equation for the spatial dependence of the flux includes a term containing the time derivative of the fluxes' spatial dependence. In the quasi-static approximation the first of the two derivative terms is kept intact while the second, the time derivative of the space function itself, is ignored. With the inclusion of the term containing the derivative of the fluxes' time dependence, the calculation of the time and spatial dependencies of the flux becomes more coupled than it was in the adiabatic approximation. No longer can the fluxes' spatial dependence be calculated with an existing steady-state diffusion theory code. In the quasi-static method, a new code which simultaneously solves for both the spatial and time dependencies of the flux must be developed. As a result of its increased accuracy, the quasi-static approximation can be used in transients where the fluxes' spatial dependence varies more rapidly. But in very severe spatially dependent transients, the flux shape changes so rapidly that the space function must be updated at almost the same frequency as point kinetics numerical time step. In this situation the more complex quasi-static equations are more time consuming to solve than a direct space-time kinetics finite difference calculation.

An additional advancement of the quasi-static method has also been developed. This method is known as the Improved Quasi-static Method (8). This method differs from the usual quasi-static approximation in that the term in the equation for the spatial dependence of the flux containing the shape function time derivative is approximated by a backwards difference formulation rather than being ignored. This enhancement allows the Improved Quasi-static Method to follow highly spatially dependent space-time transients more accurately than the normal quasi-static method. But in spite of its improved accuracy, the Improved Quasi-static Method will also consume more computational time than will a direct finite difference formulation when changes in the spatial flux distribution are occurring very rapidly.

### 2.3.2 Complete Finite Difference Formulations

The general space-time multigroup diffusion equations (Eqn's 2.1 and 2.2) can be directly solved using finite difference techniques. In this well known method, differential equations are converted to discrete algebraic equations by performing Taylor's Series expansions over a finite mesh spacing. First and second order derivatives in these expansions are approximated by simple two and three point difference approximations. Some of the advantages of using the direct finite difference solution method are that the coefficient matrices are easily calculated and that the approximation is guaranteed to converge to the exact solution in the limit of small mesh spacing. The major disadvantage of the method is that very small spacings are required in order to achieve acceptable accuracy. For example, as many as 25 mesh points may be required in order to represent a single homogenized PWR fuel assembly in the X-Y plane. This mesh distribution will yield a reactor assembly power distribution accurate to within 5%. One can therefore see that the finite difference mesh spacing requirement results in an intolerably large

number of unknowns, especially if one is interested in a full three-dimensional transient solution.

### 2.3.3 Finite Element Formulations

Several researchers have applied the finite element method to the solution of the neutron diffusion equations (19). In this method the unknown quantity is expanded in a series of piecewise continuous polynomials over an arbitrary mesh spacing in each unknown variable. The coefficients of these piecewise continuous polynomials are then calculated by weighted residual methods. The finite element method has advantages over the finite difference method in modelling media with irregularly shaped boundaries. Here the finite element method allows conditions on these irregularly shaped boundaries to be specified with fewer approximations. Otherwise the finite element method possesses approximately the same mesh spacing requirements and computational efficiency as does the finite difference method.

### 2.3.4 Nodal Methods

Nodal methods of numerically solving the neutron diffusion equations have been in use for many years. The nodal method equations are derived from the general time-dependent diffusion equation by integrating over large homogeneous nodes. In three-dimensional Cartesian coordinates this integration procedure yields the following node l nodal balance equations:

$$\frac{1}{\Delta x} (J_{gx+}^l - J_{gx-}^l) + \frac{1}{\Delta y} (J_{gy+}^l - J_{gy-}^l) + \frac{1}{\Delta z} (J_{gz+}^l - J_{gz-}^l) + A_g^l \bar{\phi}_g^l = \bar{Q}_g^l \quad (2.5)$$

where

$$\begin{aligned} \bar{\phi}_g^l &= \frac{1}{V^l} \int_{-\Delta x/2}^{\Delta x/2} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} \phi_g^l(x, y, z) dx dy dz \\ &\equiv \text{group } g \text{ node volume average flux} \end{aligned}$$

$$\bar{Q}_g^l = \frac{1}{V^l} \int_{-\Delta x/2}^{\Delta x/2} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} Q_g^l(x, y, z) dx dy dz$$

$\equiv$  group g node volume average source

$$\frac{1}{\Delta r} (J_{gr+}^l - J_{gr-}^l) = \frac{1}{V^l} \int_{-\Delta x/2}^{\Delta x/2} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{\partial}{\partial r} j_{gr}^l(x, y, z) dx dy dz$$

$J_{gr\pm}^l \equiv$  average r-directed net current on node faces  $\pm \frac{\Delta r}{2}$

$$V^l = \Delta x \Delta y \Delta z \equiv$$
 volume of node l

In order to solve these one-dimensional nodal balance equations numerically, one must develop some relationship between the node average fluxes and the node surface currents. With this relationship one can, in three-dimensions, develop a simple set of relationships which involve only the node average fluxes and/or partial currents in adjacent nodes. Employing any coupling relationship of higher order than a simple linear flux gradient over a node enables one to use larger mesh spacings than those which can be used in the finite difference approximation. The various nodal approximations in existence can be differentiated by the scheme used to couple the node average flux and the node boundary currents. These coupling schemes can be divided into three broad categories. These categories are discussed in the subsections that follow.

#### 2.3.4.1 Empirical Methods

In the first and earliest category of nodal methods, the coupling between node volume average fluxes and node face averaged currents is determined empirically. That is, one must directly input such data as neutron trans-emission (the ratio of outgoing partial current on a node face to the incoming partial current on a perpendicular face) and reflection (the ratio of outgoing to incoming partial current on a single node face) parameters at all nodal interfaces for each energy group. These parameters are then used in such a way as to eliminate unknown node boundary currents in favor of unknown node

average fluxes. These nodal coupling parameters are usually derived from the results of limited one-dimensional finite difference calculations or from experimental observations. The basic problem with this approach is that the input nodal coupling parameters are not automatically updated with changing flux levels and material compositions. Therefore the accuracy of these methods becomes suspect for problems different from those for which the nodal parameters were derived. The coupling parameters could be updated more often, but the expense of re-evaluating these parameters during a general time-dependent calculation quickly becomes prohibitive. Nonetheless, empirical nodal codes have seen much use in BWR fuel management and in general LWR run-time simulation. The most well known computer codes employing empirical nodal methodology are FLARE (20), PRESTO (21), and SIMULATE-E (22).

#### 2.3.4.2 The Analytic Nodal Method

In his 1979 thesis at the Massachusetts Institute of Technology, K. S. Smith presented a novel nodal scheme called the Analytic Nodal Method (ANM) (13). In the development of his method of solving the three-dimensional, Cartesian coordinate, steady-state and transient nodal diffusion equations, Smith first applied a technique known as transverse integration. Here the three-dimensional neutron diffusion equation is reduced to three one-dimensional equations by successively integrating over the two directions transverse to the direction of interest. For example, the steady-state x-directed transverse integrated group diffusion equation in node l can be written as follows:

$$\begin{aligned} & \frac{d}{dx} D_g^l \frac{d}{dx} \phi_{gx}^l(x) - \Sigma_{rg}^l \phi_{gx}^l(x) + \sum_{g'=1}^G \Sigma_{sg' \rightarrow g}^l \phi_{g'x}^l(x) \\ & + \frac{\chi_g}{k} \sum_{g'=1}^G \nu_{g'} \Sigma_{fg'}^l \phi_{g'x}^l = \frac{1}{\Delta y} L_{gy}^l(x) + \frac{1}{\Delta z} L_{gz}^l(x) \end{aligned} \quad (2.6)$$

where

$$\phi_{gx}^l(x) = \frac{1}{\Delta y \Delta z} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} \phi_g^l(x, y, z) dz dy \equiv \text{transverse integrated flux}$$

$$L_{gy}^l(x) = \frac{1}{\Delta z} \int_{-\Delta z/2}^{\Delta z/2} \left( -D_g^l \frac{\partial}{\partial y} \phi_g^l(x, y, z) \Big|_{-\Delta y/2}^{\Delta y/2} \right) dz \equiv y \text{ transverse leakage}$$

$$L_{gz}^l(x) = \frac{1}{\Delta y} \int_{-\Delta y/2}^{\Delta y/2} \left( -D_g^l \frac{\partial}{\partial z} \phi_g^l(x, y, z) \Big|_{-\Delta z/2}^{\Delta z/2} \right) dz \equiv z \text{ transverse leakage}$$

In order to solve this transverse integrated equation, the x-dependence of the two transverse leakage terms must be represented by some function. Here as in most advanced nodal methods, the x-dependencies of the y and z directed transverse leakages are represented by quadratic polynomial expansions. The coefficients of these expansions are chosen such that the average x-directed leakage in three x-adjacent nodes is preserved.

What is unique about Smith's nodal technique is that the three transverse integrated group diffusion equations are solved analytically within a single node. These analytical solutions are then integrated separately over two adjacent nodes in the direction of interest in order to obtain an exact algebraic relationship between the node average fluxes and the face-averaged net currents. It can be noted that the Analytic Nodal Method yields an exact solution in one dimension. In multiple dimensions the only approximation introduced by this method is the quadratic leakage approximation. An important computational advantage of the Analytic Nodal Method is the fact that in three dimensions it involves only six unknown face average currents per node per energy group. In contrast, other advanced nodal methods of similar accuracy have twelve unknown partial currents per node per energy group. As a result, the Analytic Nodal Method may in fact be up to two times as computationally efficient as the partial current based nodal methods.

In spite of its speed and accuracy, the Analytic Nodal Method does suffer from several shortcomings when compared to other advanced nodal methods. These shortcomings are basically the result of the highly complex ANM coupling equations. These coupling equations prevent the ANM from being readily extended to more than two energy groups (13). In addition, since coupling equations containing Bessel functions of two by two matrices would have to be derived, extending the ANM from Cartesian to cylindrical geometry would prove a very formidable task.

#### 2.3.4.3 Polynomial Methods

This category of nodal methods encompasses a wide variety of recently developed modern nodal coupling schemes. In each of these schemes, the solution to the transverse integrated diffusion equations in each node is approximated by a polynomial expansion. This expansion can in general be expressed as follows:

$$\phi_{gx}^l(x) = \bar{\phi}_g^l + \sum_{n=1}^N a_{gx_n}^l f_n(x), -\frac{\Delta x}{2} \leq x \leq \frac{\Delta x}{2} \quad (2.7)$$

The choice of basis functions ( $f_n(x)$ ) and the methods of determining the expansion coefficients ( $a_{gx_n}$ ) are what differentiates the various polynomial methods from one another. In addition, just as in the Analytic Nodal Method, the various polynomial methods typically expand the x variation of the y and z leakage terms in the transverse integrated diffusion equation in quadratic polynomials.

The earliest of the polynomial methods is known as the Nodal Expansion Method (NEM) (16). This method was developed in West Germany in the late 1970's by H. Finnemann and his co-workers at Kraftwerk Union (KWU). In this scheme the first two basis functions are chosen to be:

$$f_1(x) = \frac{x}{\Delta x} \quad (2.8)$$

$$f_2(x) = 3\left(\frac{x}{\Delta x}\right)^2 - \frac{1}{4} \quad (2.9)$$

When this set of basis functions is used, the first two expansion coefficients are simply:

$$a_{gx1}^l = \phi_{gx+}^l - \phi_{gx-}^l \quad (2.10)$$

$$a_{gx2}^l = \phi_{gx+}^l + \phi_{gx-}^l - 2\bar{\phi}_g^l \quad (2.11)$$

where

$\bar{\phi}_g^l \equiv$  node volume average flux

$\phi_{gx\pm}^l \equiv$  x-direction transverse integrated flux evaluated at  $\frac{\pm\Delta x}{2}$

Basis functions of higher order are chosen such that they go to zero at the x-directed node boundaries. The expansion coefficients for these basis functions are chosen by a weighted residual procedure. Here the transverse integrated diffusion equations (Eqn 2.6) are first multiplied by a polynomial weight function  $w_n(x)$ , then integrated over the x extent of a node. Writing the node average flux in terms of its original polynomial expansion allows one to directly relate node average fluxes to the face averaged partial currents, enabling the solution to proceed similarly to that of standard finite difference calculation.

In another polynomial method developed by Sims at MIT (23), the basis functions in Equation 2.7 are chosen such that the first four expansion coefficients turn out to be the node face averaged partial currents;  $J_{gx+}^{out,l}, J_{gx+}^{in,l}, J_{gx-}^{out,l}, J_{gx-}^{in,l}$ .

In a third polynomial method, the NODLEG method developed by Maeder (24), the basis functions are chosen to be Legendre polynomials. The expansion coefficients in this case thus are simply Legendre moments.

A fourth polynomial nodal method applies a different approach for finding basis functions and expansion coefficients. This approach, developed at the University of Illinois by R. D. Lawrence, is known as the Nodal Green's Function Method (NGFM)

(25). In this method the transverse integrated diffusion equation in a given node is first converted from a differential into an integral equation. This conversion is performed by calculating the Green's function for the one-dimensional diffusion removal operator in the transverse integrated diffusion equation. These integral equations yield a local analytical relationship for the shape of the group flux within a node. In order to facilitate a numerical solution, these integral equations must be converted to some type of algebraic relationship between node average fluxes and the face averaged partial currents. This conversion is performed by first expanding the analytically generated one-dimensional group flux profiles in a series of quadratic polynomials. The coefficients in this expansion are calculated by means of standard weighted residual techniques. These polynomial flux profile approximations are finally integrated over the node width in order to obtain the sought after algebraic relationship between the node average fluxes and the node face averaged currents.

The final widely used polynomial nodal method is the QUABOX/CUBBOX coarse mesh method developed by S. Langenbuch and his associates in the mid 1970's (26). Basically this method can be distinguished from other polynomial nodal methods by the fact that node centerpoint rather node average fluxes are the principle unknowns. In addition, this method does not rely upon transverse integration of the neutron diffusion equation in multidimensional problems. Instead the flux distribution in a two or three dimensional node cell is expanded in a general multidimensional non-separable polynomial. The coefficients of this polynomial are then determined through the use of asymmetric weighted residual techniques.

#### 2.4 Summary

After a careful review of present spatially dependent transient neutronic methods, it has been decided that the Nodal Expansion Method holds the most promise for use in

TRAC-PF1. The major advantages of the NEM over other solution techniques are its adaptability to a very coarse mesh spacing, its conceptual simplicity, and its ability to be easily extended to cylindrical geometry and to more than two energy groups. The NEM has already been extended to the solution of the transient few-group diffusion equations (27). But the present technique uses a simple backward time difference approximation for both the transient flux and precursor equations (Eqn's 2.1 and 2.2). An improvement to the precursor portion of this time differencing scheme, a technique based upon Stacy's Time-Integrated Method (28) will first be tested. Later an exponential transformation scheme similar to that employed by Buckner and Stewart (29) will be developed and tested. Either or both of these time variable discretization methods should greatly increase the maximum acceptable time step size in a transient calculation.

The next chapter will present a derivation of the steady-state Nodal Expansion Method in both Cartesian and cylindrical geometries. Chapter 4 will present the results to several two and three dimensional steady-state benchmark problems. Chapter 5 will provide both the derivation and calculational results for the two above mentioned transient NEM approximations. Chapter 6 will include the results of coupling the transient NEM into TRAC-PF1. Finally, Chapter 7 will provide the conclusions reached during this study and recommendations for future work.

## CHAPTER 3

### DERIVATION OF THE STEADY-STATE NODAL EXPANSION METHOD

#### 3.1 Introduction

In this chapter the numerics of the Nodal Expansion Method will be derived in both Cartesian and Cylindrical geometries. In addition, the means by which these basic algorithms will be implemented in the Fortran computer language are described.

#### 3.2 Cartesian Geometry

##### 3.2.1 Derivation of Three-Dimensional Numerics

The following section outlines the details of deriving the two energy group, fourth order, quadratic leakage Nodal Expansion Method (NEM) in three-dimensional Cartesian Geometry. The derivation of the Cartesian geometry NEM in two dimensions, with a different number of energy groups, or with a different order polynomial flux/leakage expansion is conceptually similar to the derivation to be outlined.

The derivation begins by introducing the general form of the steady-state multi-group neutron diffusion equations.

$$\nabla^2 D_g \phi_g - \Sigma_{rg} \phi_g + \sum_{g'=1}^G \Sigma_{sg' \rightarrow g} \phi_{g'} + \frac{1}{K} \chi_g \sum_{g'=1}^G \nu_{g'} \Sigma_{fg'} \phi_{g'} = 0 \quad (3.1)$$

where

$\phi_g$   $\equiv$  neutron flux in group g

$D_g$   $\equiv$  group g diffusion coefficient

$\Sigma_{rg}$   $\equiv$  group g removal cross section

$\Sigma_{sg' \rightarrow g}$  ≡ probability of a neutron scattering from group  $g'$  into group  $g$

$K$  ≡ multiplicative eigenvalue

$\chi_g$  ≡ fraction of fission neutrons entering group  $g$

$\nu_{g'}$  ≡ average number of neutrons produced by a group  $g'$  fission event

$\Sigma_{fg'}$  ≡ group  $g'$  fission cross section

Equation 3.1 can be written in two energy groups within Cartesian node I with constant neutronic properties whose dimensions are  $\Delta x, \Delta y, \Delta z$  as:

$$\begin{aligned} & -D_g^I \frac{\partial^2}{\partial x^2} \phi_g^I(x, y, z) - D_g^I \frac{\partial^2}{\partial y^2} \phi_g^I(x, y, z) - D_g^I \frac{\partial^2}{\partial z^2} \phi_g^I(x, y, z) \\ & + A_g^I \phi_g^I(x, y, z) = Q_g^I(x, y, z) \\ & (x, y, z) \in V^I, g = 1, 2 \end{aligned} \quad (3.2)$$

where

$$A_1^I = \Sigma_{a1}^I + \Sigma_{12}^I - \frac{1}{K} \nu \Sigma_{f1}^I$$

$$A_2^I = \Sigma_{a2}^I$$

$$Q_1^I(x, y, z) = \frac{1}{K} \nu \Sigma_{f2}^I \phi_2^I(x, y, z)$$

$$Q_2^I(x, y, z) = \Sigma_{12}^I \phi_1^I(x, y, z)$$

$$V^I = \Delta x \Delta y \Delta z \equiv \text{Volume of node I}$$

$\Sigma_{ag}$  ≡ Group  $g$  absorption cross-section

$\Sigma_{12}$  ≡ Group 1 to 2 scattering cross-section

Using Fick's Law, which in the  $x$ -direction may be written as,

$$j_{gx}^I(x, y, z) = -D_g^I \frac{\partial}{\partial x} \phi_g^I(x, y, z) \quad (3.3)$$

where

$j_{gx}^I(x, y, z)$  ≡  $x$ -component of the net neutron current

Equation 3.2 may be rewritten as:

$$\begin{aligned} \frac{\partial}{\partial x} j_{gx}^l(x, y, z) + \frac{\partial}{\partial y} j_{gy}^l(x, y, z) + \frac{\partial}{\partial z} j_{gz}^l(x, y, z) + A_g^l \phi_g^l(x, y, z) \\ = Q_g^l(x, y, z) \\ (x, y, z) \in V^l \end{aligned} \quad (3.4)$$

### 3.2.1.1 The Nodal Balance Equation

Assuming the coordinate origin to be at the center of cell I, Equation 3.4 can be integrated over the volume of the cell to obtain a local neutron balance equation. This balance equation can be expressed as:

$$\begin{aligned} \frac{1}{\Delta x} (J_{gx+}^l - J_{gx-}^l) + \frac{1}{\Delta y} (J_{gy+}^l - J_{gy-}^l) + \frac{1}{\Delta z} (J_{gz+}^l - J_{gz-}^l) + A_g^l \bar{\phi}_g^l \\ = \bar{Q}_g^l \end{aligned} \quad (3.5)$$

where

$$\begin{aligned} \bar{\phi}_g^l &= \frac{1}{V^l} \int_{-\Delta x/2}^{\Delta x/2} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} \phi_g^l(x, y, z) dx dy dz \\ &\equiv \text{node volume average flux} \\ \bar{Q}_g^l &= \frac{1}{V^l} \int_{-\Delta x/2}^{\Delta x/2} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} Q_g^l(x, y, z) dx dy dz \\ &\equiv \text{node volume average source} \\ \frac{1}{\Delta x} (J_{gx+}^l - J_{gx-}^l) &= \frac{1}{V^l} \int_{-\Delta x/2}^{\Delta x/2} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{\partial}{\partial x} j_{gx}^l(x, y, z) dx dy dz \\ J_{gx\pm}^l &\equiv \text{average x-directed net current on node faces } \pm \frac{\Delta x}{2} \\ \frac{1}{\Delta y} (J_{gy+}^l - J_{gy-}^l) &= \frac{1}{V^l} \int_{-\Delta x/2}^{\Delta x/2} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{\partial}{\partial y} j_{gy}^l(x, y, z) dx dy dz \\ J_{gy\pm}^l &\equiv \text{average y-directed net current on node faces } \pm \frac{\Delta y}{2} \\ \frac{1}{\Delta z} (J_{gz+}^l - J_{gz-}^l) &= \frac{1}{V^l} \int_{-\Delta x/2}^{\Delta x/2} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{\partial}{\partial z} j_{gz}^l(x, y, z) dx dy dz \\ J_{gz\pm}^l &\equiv \text{average z-directed net current on node faces } \pm \frac{\Delta z}{2} \end{aligned}$$

### 3.2.1.2 The Transverse Integration Procedure

In order to solve for the spatial neutron flux distribution in a medium consisting of neutronically homogeneous nodes, one must derive some relationship between the node average flux and the face averaged net currents. In the Nodal Expansion Method this coupling relationship is provided by a series of three consistently derived one-dimensional polynomial flux expansions. In order to implement these one-dimensional polynomial flux expansions, the diffusion equation (Eqn 3.4) must first be transversely integrated. The transverse integration approximation, widely used in the development of modern multi-dimensional nodal methods (13,16,23,25), requires that Equation 3.4 be spatially integrated over the two dimensions transverse to the particular direction of interest. This approximation technique is primarily motivated by the simple observation that it is generally much easier to solve three one-dimensional equations than it is to solve a single three-dimensional equation. The x-directed transverse integrated diffusion equation within node l takes on the following form:

$$\frac{d}{dx} j_{gx}^l(x) + A_g^l \phi_{gx}^l(x) = Q_{gx}^l(x) - \frac{1}{\Delta y} L_{gy}^l(x) - \frac{1}{\Delta z} L_{gz}^l(x) \quad (3.6)$$

where

$$\begin{aligned} \phi_{gx}^l(x) &= \frac{1}{\Delta y \Delta z} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} \phi_g^l(x, y, z) dz dy \\ \frac{d}{dx} j_{gx}^l(x) &= \frac{1}{\Delta y \Delta z} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{\partial}{\partial x} j_{gx}^l(x, y, z) dz dy \\ Q_{gx}^l(x) &= \frac{1}{\Delta y \Delta z} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} Q_g^l(x, y, z) dz dy \\ L_{gy}^l(x) &= \frac{1}{\Delta z} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{\partial}{\partial y} j_{gy}^l(x, y, z) dz dy \\ &\equiv y\text{-direction transverse leakage} \end{aligned}$$

$$L_{gz}^l(x) = \frac{1}{\Delta y} \int_{-\Delta y/2}^{\Delta y/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{\partial}{\partial z} j_{gz}^l(x, y, z) dz dy \\ \equiv z\text{-direction transverse leakage}$$

The transverse integration procedure yields similar results for the y and z coordinate directions.

### 3.2.1.3 Polynomial Approximation of the One-Dimensional Equations

In the Nodal Expansion Method, the one-dimensional flux which appears in the x-directed transverse integrated diffusion equation is expanded in a series of polynomials as:

$$\phi_{gx}^l(x) = \bar{\phi}_g^l + \sum_{n=1}^N a_{gx n}^l f_n(x) \quad (3.7)$$

The first four basis functions in the NEM can be expressed as follows:

$$f_1(x) = \frac{x}{\Delta x} \quad (3.8)$$

$$f_2(x) = 3\left(\frac{x}{\Delta x}\right)^2 - \frac{1}{4} \quad (3.9)$$

$$f_3(x) = \frac{x}{\Delta x} \left(\frac{x}{\Delta x} - \frac{1}{2}\right) \left(\frac{x}{\Delta x} + \frac{1}{2}\right) = \left(\frac{x}{\Delta x}\right)^3 - \frac{1}{4}\left(\frac{x}{\Delta x}\right) \quad (3.10)$$

$$f_4(x) = \left(\left(\frac{x}{\Delta x}\right)^2 - \frac{1}{20}\right) \left(\frac{x}{\Delta x} - \frac{1}{2}\right) \left(\frac{x}{\Delta x} + \frac{1}{2}\right) \\ = \left(\frac{x}{\Delta x}\right)^4 - \frac{3}{10}\left(\frac{x}{\Delta x}\right)^2 + \frac{1}{80} \quad (3.11)$$

These basis functions can be shown to satisfy the following criteria:

$$\frac{1}{\Delta x} \int_{-\Delta x/2}^{\Delta x/2} f_n(x) dx = 0, n = 1, 4 \quad (3.12)$$

$$f_n\left(\frac{\pm \Delta x}{2}\right) = 0, n = 3, 4 \quad (3.13)$$

Similar polynomial expansions and basis functions are used to represent the y and z directed transverse integrated fluxes.

One must now solve for the x-directed expansion coefficients,  $a_{gx}^l$ . The first two expansion coefficients can be found by simply evaluating Equation 3.7 at the endpoints of the node ( $\frac{\pm\Delta x}{2}$ ). This evaluation results in:

$$a_{gx1}^l = \phi_{gx+}^l - \phi_{gx-}^l \quad (3.14)$$

$$a_{gx2}^l = \phi_{gx+}^l + \phi_{gx-}^l - 2\bar{\phi}_g^l \quad (3.15)$$

where

$$\phi_{gx\pm}^l \equiv \text{x-direction transverse integrated flux evaluated at } \frac{\pm\Delta x}{2}$$

The expansion coefficients  $a_{gx3}^l$  and  $a_{gx4}^l$  in the NEM are determined by a weighted residual procedure. This implies that Equation 3.6 is first multiplied through by a weight function,  $w_n(x)$ . The resulting product is integrated over the x extent of the node then set equal to zero. Previous numerical studies by Finnemann et al. (16) have shown that the most accurate weighting scheme is the moments weighting method. Here the weight functions used to find  $a_{gx3}^l$  and  $a_{gx4}^l$  are the first two basis functions. That is,  $w_1(x) \equiv f_1(x)$  and  $w_2(x) \equiv f_2(x)$ . The moments weighting scheme can be expressed in equation form as follows:

$$\begin{aligned} & \langle w_n(x), \frac{d}{dx} j_{gx}^l(x) + A_g^l \phi_{gx}^l(x) - Q_{gx}^l(x) \\ & + \frac{1}{\Delta y} L_{gy}^l(x) + \frac{1}{\Delta z} L_{gz}^l(x) \rangle = 0 \end{aligned} \quad (3.16)$$

The flux-like term obtained in performing the moments weighting with  $w_1(x) = \frac{x}{\Delta x}$  can be expressed as:

$$\bar{\phi}_{gx1}^l \equiv \langle w_1(x), \phi_{gx}^l(x) \rangle = \frac{1}{\Delta x} \int_{-\Delta x/2}^{\Delta x/2} \frac{x}{\Delta x} \phi_{gx}^l(x) dx \quad (3.17)$$

Substituting the polynomial flux expansion and basis functions given in Eqn's 3.7 through 3.11 into Equation 3.17, one obtains the following relationship for  $a_{gx3}^l$  .

$$a_{gx3}^l = 10a_{gx1}^l - 120\bar{\phi}_{gx1}^l \quad (3.18)$$

When weighting with  $w_2(x) = 3(\frac{x}{\Delta x})^2 - \frac{1}{4}$ , the flux-like term results in the following definition for  $a_{gx4}^l$ .

$$a_{gx4}^l = 35a_{gx2}^l - 700\bar{\phi}_{gx2}^l \quad (3.19)$$

where

$$\bar{\phi}_{gx2}^l \equiv \langle w_2(x), \phi_{gx}^l(x) \rangle$$

The first and most difficult step in the evaluation of the moments weighting expressions (Eqn 3.16) involves calculating the first or current derivative containing terms. Upon performing integration by parts these terms can be expressed as:

$$\langle w_1(x), \frac{d}{dx} j_{gx}^l(x) \rangle = \frac{1}{2\Delta x} (J_{gx+}^l + J_{gx-}^l) + \frac{D_g^l}{\Delta x^2} (\phi_{gx+}^l - \phi_{gx-}^l) \quad (3.20)$$

and

$$\begin{aligned} \langle w_2(x), \frac{d}{dx} j_{gx}^l(x) \rangle &= \frac{1}{2\Delta x} (J_{gx+}^l - J_{gx-}^l) \\ &+ \frac{3D_g^l}{\Delta x^2} (\phi_{gx+}^l + \phi_{gx-}^l) - \frac{6D_g^l}{\Delta x^2} \bar{\phi}_g^l \end{aligned} \quad (3.21)$$

The entire moment equations (Eqn's 3.16) can now be expressed as:

$$\begin{aligned} \bar{\phi}_{gx1}^l &= -\frac{1}{A_g^l} \left( \frac{1}{2\Delta x} (J_{gx+}^l + J_{gx-}^l) + \frac{D_g^l}{\Delta x^2} (\phi_{gx+}^l - \phi_{gx-}^l) \right. \\ &\quad \left. - \bar{Q}_{gx1}^l + \frac{1}{\Delta y} L_{gyx1}^l + \frac{1}{\Delta z} L_{gzz1}^l \right) \end{aligned} \quad (3.22)$$

and

$$\begin{aligned} \bar{\phi}_{gx2}^l &= -\frac{1}{A_g^l} \left( \frac{1}{2\Delta x} (J_{gx+}^l - J_{gx-}^l) + \frac{3D_g^l}{\Delta x^2} (\phi_{gx+}^l + \phi_{gx-}^l - 2\bar{\phi}_g^l) \right. \\ &\quad \left. - \bar{Q}_{gx2}^l + \frac{1}{\Delta y} L_{gyx2}^l + \frac{1}{\Delta z} L_{gzz2}^l \right) \end{aligned} \quad (3.23)$$

where by definition

$$\bar{Q}_{gx_n}^l \equiv \langle w_n(x), Q_{gx}^l(x) \rangle$$

$$L_{gyx_n}^l \equiv \langle w_n(x), L_{gy}^l(x) \rangle$$

$$L_{gzz_n}^l \equiv \langle w_n(x), L_{gz}^l(x) \rangle$$

Similar operations performed on the y and z direction transverse integrated diffusion equations will yield complementary expressions for  $a_{gyn}^l, n = 1, 4, a_{gz_n}^l, n = 1, 4, \bar{\phi}_{gyn}^l, n = 1, 2$ , and  $\bar{\phi}_{gz_n}^l, n = 1, 2$ .

### 3.2.1.4 The Response Matrix Equations

At this point one has enough information to produce the final partial current equations, that is a 6x6 matrix of equations which, in each node, express the outgoing partial currents as a function of incoming partial currents and intra-node sources/sinks. The expression for these partial current matrices can be derived from the Fick's Law expressions for the partial currents on the node boundaries. On the two faces of node l normal to the x-axis, Fick's Law takes on the following form:

$$J_{gx+}^{out,l} = J_{gx+}^{in,l} - D_g^l \frac{d}{dx} \phi_{gx}^l(x) |_{x=\Delta x/2} \quad (3.24)$$

at  $x = \frac{\Delta x}{2}$ , and

$$J_{gx-}^{out,l} = J_{gx-}^{in,l} + D_g^l \frac{d}{dx} \phi_{gx}^l(x) |_{x=-\Delta x/2} \quad (3.25)$$

at  $x = -\frac{\Delta x}{2}$ . The incoming and outgoing partial currents are related to the face averaged flux and net current as follows:

$$J_{gx+}^{out,l} - J_{gx+}^{in,l} = J_{gx+}^l \quad (3.26)$$

$$J_{gx-}^{out,l} - J_{gx-}^{in,l} = -J_{gx-}^l \quad (3.27)$$

$$2 \left( J_{gx\pm}^{out,l} + J_{gx\pm}^{in,l} \right) = \phi_{gx\pm}^l \quad (3.28)$$

The first step in producing the final partial current matrices from Eqn's 3.24 and 3.25 and the four analogous expressions on the y and z directed faces of node l involves replacing the fluxes with the polynomial expansion of Equation 3.7. Performing this substitution and differentiation, then substituting the the expressions for the polynomial expansion coefficients (Eqn's 3.14, 3.15, 3.18, and 3.19), results in:

$$J_{gx+}^{out,l} = J_{gx+}^{in,l} - \frac{D_g^l}{\Delta x} \left( 16\phi_{gx+}^l + 4\phi_{gx-}^l - 20\bar{\phi}_g^l - 60\bar{\phi}_{gx1}^l - 140\bar{\phi}_{gx2}^l \right) \quad (3.29)$$

and

$$J_{gx-}^{out,l} = J_{gx-}^{in,l} + \frac{D_g^l}{\Delta x} \left( -4\phi_{gx+}^l - 16\phi_{gx-}^l + 20\bar{\phi}_g^l - 60\bar{\phi}_{gx1}^l + 140\bar{\phi}_{gx2}^l \right) \quad (3.30)$$

Next one can substitute for the node average fluxes  $\bar{\phi}_g$  and the flux moments  $\bar{\phi}_{gx_n}$ , n=1,2 in Eqn's 3.29 and 3.30 using the neutron balance equation (Eqn 3.5) and the flux moment equations (Eqn's 3.22 and 3.23). After much algebraic manipulation these substitutions result in:

$$\begin{aligned} J_{gx+}^{out,l} &= J_{gx+}^{in,l} - \frac{D_g^l}{\Delta x} \left( 16\phi_{gx+}^l \left( 1 + \frac{30D_g^l}{A_g^l \Delta x^2} \right) + 4\phi_{gx-}^l \left( 1 + \frac{90D_g^l}{A_g^l \Delta x^2} \right) \right. \\ &\quad + J_{gx+}^l \left( \frac{120}{A_g^l \Delta x} \left( 1 + \frac{7D_g^l}{A_g^l \Delta x^2} \right) \right) + J_{gx-}^l \left( \frac{-60}{A_g^l \Delta x} \left( 1 + \frac{14D_g^l}{A_g^l \Delta x^2} \right) \right) \\ &\quad + (J_{gy+}^l - J_{gy-}^l) \left( \frac{20}{A_g^l \Delta y} \left( 1 + \frac{42D_g^l}{A_g^l \Delta x^2} \right) \right) \\ &\quad + (J_{gz+}^l - J_{gz-}^l) \left( \frac{20}{A_g^l \Delta z} \left( 1 + \frac{42D_g^l}{A_g^l \Delta x^2} \right) \right) + \bar{Q}_g^l \left( \frac{-20}{A_g^l} \left( 1 + \frac{42D_g^l}{A_g^l \Delta x^2} \right) \right) \\ &\quad + \bar{Q}_{gx1}^l \left( \frac{-60}{A_g^l} \right) + \bar{Q}_{gx2}^l \left( \frac{-140}{A_g^l} \right) + L_{gyx1}^l \left( \frac{60}{A_g^l \Delta y} \right) \\ &\quad \left. + L_{gyx2}^l \left( \frac{140}{A_g^l \Delta y} \right) + L_{gxx1}^l \left( \frac{60}{A_g^l \Delta z} \right) + L_{gxx2}^l \left( \frac{140}{A_g^l \Delta z} \right) \right) \end{aligned} \quad (3.31)$$

and

$$\begin{aligned}
 J_{gx-}^{out,l} = & J_{gx-}^{in,l} + \frac{D_g^l}{\Delta x} \left( -4\phi_{gx+}^l \left( 1 + \frac{90D_g^l}{A_g^l \Delta x^2} \right) - 16\phi_{gx-}^l \left( 1 + \frac{30D_g^l}{A_g^l \Delta x^2} \right) \right. \\
 & + J_{gx+}^l \left( \frac{-60}{A_g^l \Delta x} \left( 1 + \frac{14D_g^l}{A_g^l \Delta x^2} \right) \right) + J_{gx-}^l \left( \frac{120}{A_g^l \Delta x} \left( 1 + \frac{7D_g^l}{A_g^l \Delta x^2} \right) \right) \\
 & + (J_{gy+}^l - J_{gy-}^l) \left( \frac{-20}{A_g^l \Delta y} \left( 1 + \frac{42D_g^l}{A_g^l \Delta x^2} \right) \right) \\
 & + (J_{gz+}^l - J_{gz-}^l) \left( \frac{-20}{A_g^l \Delta z} \left( 1 + \frac{42D_g^l}{A_g^l \Delta x^2} \right) \right) + \bar{Q}_g^l \left( \frac{20}{A_g^l} \left( 1 + \frac{42D_g^l}{A_g^l \Delta x^2} \right) \right) \\
 & + \bar{Q}_{gx1}^l \left( \frac{-60}{A_g^l} \right) + \bar{Q}_{gx2}^l \left( \frac{140}{A_g^l} \right) + L_{gyx1}^l \left( \frac{60}{A_g^l \Delta y} \right) \\
 & \left. + L_{gyx2}^l \left( \frac{-140}{A_g^l \Delta y} \right) + L_{gzz1}^l \left( \frac{60}{A_g^l \Delta z} \right) + L_{gzz2}^l \left( \frac{-140}{A_g^l \Delta z} \right) \right)
 \end{aligned} \tag{3.32}$$

If the expressions for the partial currents (Eqn's 3.26, 3.27, and 3.28) are substituted into Eqn's 3.31 and 3.32 these two partial current equations and their y and z direction analogues may be written in matrix form as:

$$[\mathbf{A}] \cdot J_g^{out,l} = [\mathbf{C}] \cdot J_g^{in,l} + [\mathbf{B}_1] \cdot Q_g^l + [\mathbf{B}_2] \cdot L_g^l \tag{3.33}$$

where

$$\mathbf{A} = \begin{pmatrix} a_1 & a_2 & a_3 & a_3 & a_4 & a_4 \\ a_2 & a_1 & a_3 & a_3 & a_4 & a_4 \\ a_7 & a_7 & a_5 & a_6 & a_8 & a_8 \\ a_7 & a_7 & a_6 & a_5 & a_8 & a_8 \\ a_{11} & a_{11} & a_{12} & a_{12} & a_9 & a_{10} \\ a_{11} & a_{11} & a_{12} & a_{12} & a_{10} & a_9 \end{pmatrix}$$

$$\mathbf{J}_g^{out,l} = (J_{gx+}^{out,l}, J_{gx-}^{out,l}, J_{gy+}^{out,l}, J_{gy-}^{out,l}, J_{gz+}^{out,l}, J_{gz-}^{out,l})^T$$

$$\mathbf{C} = \begin{pmatrix} c_1 & c_2 & c_3 & c_3 & c_4 & c_4 \\ c_2 & c_1 & c_3 & c_3 & c_4 & c_4 \\ c_7 & c_7 & c_5 & c_6 & c_8 & c_8 \\ c_7 & c_7 & c_6 & c_5 & c_8 & c_8 \\ c_{11} & c_{11} & c_{12} & c_{12} & c_9 & c_{10} \\ c_{11} & c_{11} & c_{12} & c_{12} & c_{10} & c_9 \end{pmatrix}$$

$$\mathbf{J}_g^{\text{in},l} = \left( J_{gx+}^{in,l}, J_{gx-}^{in,l}, J_{gy+}^{in,l}, J_{gy-}^{in,l}, J_{gz+}^{in,l}, J_{gz-}^{in,l} \right)^T$$

$$\mathbf{B}_1 = \begin{pmatrix} b_1 & b_2 & b_3 & 0 & 0 & 0 & 0 \\ b_1 & -b_2 & b_3 & 0 & 0 & 0 & 0 \\ b_8 & 0 & 0 & b_9 & b_{10} & 0 & 0 \\ b_8 & 0 & 0 & -b_9 & b_{10} & 0 & 0 \\ b_{13} & 0 & 0 & 0 & 0 & b_{14} & b_{15} \\ b_{13} & 0 & 0 & 0 & 0 & -b_{14} & b_{15} \end{pmatrix}$$

$$\mathbf{Q}_g^l = \left( \bar{Q}_g^l, \bar{Q}_{gx1}^l, \bar{Q}_{gx2}^l, \bar{Q}_{gy1}^l, \bar{Q}_{gy2}^l, \bar{Q}_{gz1}^l, \bar{Q}_{gz2}^l \right)^T$$

$$\mathbf{B}_2 = \begin{pmatrix} -b_4 & b_5 & -b_6 & b_7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ b_4 & b_5 & b_6 & b_7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -b_4 & b_5 & -b_{11} & b_{12} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & b_4 & b_5 & b_{11} & b_{12} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -b_6 & b_7 & -b_{11} & b_{12} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & b_6 & b_7 & b_{11} & b_{12} \end{pmatrix}$$

$$\mathbf{L}_g^l = (L_{gyx1}^l, L_{gyx2}^l, L_{gzz1}^l, L_{gzz2}^l, L_{gxy1}^l, L_{gxy2}^l, L_{gzy1}^l, L_{gzy2}^l, L_{gxz1}^l, L_{gxz2}^l, L_{gyz1}^l, L_{gyz2}^l)^T$$

The coefficient a's, b's, and c's are expressed in terms of known quantities in Appendix A.

### 3.2.1.5 Calculation of the Transverse Leakage Moments

At this point all members of the vector quantities on the right-hand side of the response matrix equation (Eqn 3.33) are known from the outgoing currents in neighboring nodes, the specified problem boundary conditions, or the flux balance and flux moment equations (Eqn's 3.5, 3.22, and 3.23). All quantities that is, except the transverse leakage moments ( $L_g^l$ ). In the quadratic leakage approximation to the Nodal Expansion Method, these transverse leakage moments are calculated from various previous iterate incoming and outgoing partial currents based on the assumption that, for example, the x-direction spatial dependence of  $L_{gy}^l(x)$  in node l can be approximated by a quadratic polynomial  $p_{gy}^l(x)$ , where

$$p_{gy}^l(x) \equiv \bar{L}_{gy}^l + p_{gy1}^l f_1(x) + p_{gy2}^l f_2(x) \quad (3.34)$$

with

$$\bar{L}_{gy}^l \equiv \text{average y-directed leakage in node } l$$

and the expansion functions  $f_1(x)$  and  $f_2(x)$  defined as in Equations 3.8 and 3.9.

The expansion coefficients in this approximation are calculated based upon the assumption that  $p_{gy}^l(x)$  extends not only over node  $l$ , but over its two neighboring nodes in the  $x$ -direction, designated as  $l+$  and  $l-$ . Equation 3.34 is constrained to yield the average  $y$ -directed leakages in nodes  $l+$  and  $l-$  when integrated over either of these two nodes. The specific expansion coefficients obtained in this exercise are then used only to specify the quadratic leakage approximation in node  $l$ . Separate calculations are performed to determine the expansion coefficients in nodes  $l+$  and  $l-$ .

The details of calculating the leakage moments  $L_{gyx1}^l$  and  $L_{gyx2}^l$  are as follows. First Equation 3.34 is integrated over the  $x$ -extents of nodes  $l-$  and  $l+$ , i.e.:

$$\bar{L}_{gy}^{l-} = \frac{1}{\Delta x^{l-}} \int_{-\Delta x^l/2 - \Delta x^{l-}}^{-\Delta x^l/2} p_{gy}^l(x) \quad (3.35)$$

and

$$\bar{L}_{gy}^{l+} = \frac{1}{\Delta x^{l+}} \int_{\Delta x^l/2}^{\Delta x^l/2 + \Delta x^{l+}} p_{gy}^l(x) \quad (3.36)$$

Upon substituting for  $p_{gy}^l(x)$  from Eqn 3.34 and  $f_1(x)$  and  $f_2(x)$  from Eqn's 3.8 and 3.9, then performing the indicated integrations, the expansion coefficients  $p_{gy1}^l$  and  $p_{gy2}^l$  turn out to be:

$$\begin{aligned} p_{gy1}^l = & DD^l(\Delta x^l) \left( (\bar{L}_{gy}^{l+} - \bar{L}_{gy}^l) (\Delta x^l + 2\Delta x^{l-}) (\Delta x^l + \Delta x^{l-}) \right. \\ & \left. + (\bar{L}_{gy}^l - \bar{L}_{gy}^{l-}) (\Delta x^l + 2\Delta x^{l+}) (\Delta x^l + \Delta x^{l+}) \right) \end{aligned} \quad (3.37)$$

and

$$\begin{aligned} p_{gy2}^l = & DD^l(\Delta x^l)^2 \left( (\bar{L}_{gy}^{l+} - \bar{L}_{gy}^l) (\Delta x^l + \Delta x^{l-}) \right. \\ & \left. + (\bar{L}_{gy}^l - \bar{L}_{gy}^{l-}) (\Delta x^l + \Delta x^{l+}) \right) \end{aligned} \quad (3.38)$$

where

$$DD^l = \frac{1}{(\Delta x^l + \Delta x^{l+})(\Delta x^l + \Delta x^{l-})(\Delta x^l + \Delta x^{l+} + \Delta x^{l-})}$$

The definitions of the leakage moments from Equations 3.22 and 3.23 can now be used to evaluate the quantities  $L_{gyx1}^l$  and  $L_{gyx2}^l$ . The procedure for deriving these leakage moments is as follows:

$$L_{gyx1}^l \equiv \langle w_1(x), L_{gy}^l(x) \rangle = \frac{1}{\Delta x^l} \int_{-\Delta x^l/2}^{\Delta x^l/2} \left( \frac{x}{\Delta x^l} \right) p_{gy}^l(x) dx \quad (3.39)$$

and

$$L_{gyx2}^l \equiv \langle w_2(x), L_{gy}^l(x) \rangle = \frac{1}{\Delta x^l} \int_{-\Delta x^l/2}^{\Delta x^l/2} \left( 3 \left( \frac{x}{\Delta x^l} \right)^2 - \frac{1}{4} \right) p_{gy}^l(x) dx \quad (3.40)$$

The results of the above integrations are simply:

$$L_{gyx1}^l = \frac{p_{gy1}^l}{12} \quad (3.41)$$

and

$$L_{gyx2}^l = \frac{p_{gy2}^l}{60} \quad (3.42)$$

Upon evaluating the other ten leakage moments on the right-hand side of Equation 3.33 one obtains relationships similar to Eqn's 3.37, 3.38, 3.41, and 3.42. To obtain numerical values for these various transverse leakage moments, one must express the center cell and next-neighbor cell face averaged leakages in Eqn's 3.37 and 3.38 in terms of the values of the partial currents on particular faces of these various cells. For example:

$$\bar{L}_{gy}^l = J_{gy+}^{out,l} - J_{gy+}^{in,l} + J_{gy-}^{out,l} - J_{gy-}^{in,l} \quad (3.43)$$

### 3.2.1.6 Boundary Conditions

Up until this point, the fourth order Nodal Expansion Method has been shown to be a means by which the outgoing partial currents in a single neutronically homogeneous cell can be calculated based on known incoming partial currents and internal sources/sinks. In a heterogeneous medium with many homogeneous Cartesian nodes, one must somehow uniquely describe the neutron current at the boundaries of edge and interior nodes. This task is accomplished through the specification of node boundary conditions.

In the interior of the medium, neighboring nodes are usually coupled together by assuming that the partial currents are continuous across nodal interfaces. That is, the assumed outgoing surface averaged partial currents of one node are taken as the known incoming partial currents for nodes with common faces.

At the surfaces of nodes which form the outer problem boundary, one can, in this analysis, invoke one of three possible boundary conditions. The first of these boundary conditions involves the specification of pure reflection. Here, as is the case when only a fraction of a neutronically symmetric medium is being modelled, the incoming partial current on the surface of reflection is assumed to be equal to the outgoing partial current of the same surface, i.e.  $J_{g(x,y,z)\pm}^{in,l} = J_{g(x,y,z)\pm}^{out,l}$ . The second outer surface boundary condition involves specifying a vacuum boundary. This boundary condition physically describes a situation in which neutrons escaping the outer boundaries of a convex system have no chance of returning. This condition is numerically specified as:  $J_{g(x,y,z)\pm}^{in,l} = 0$ . The final outer boundary condition, the zero flux boundary condition, is used to exactly reproduce the rather artificial condition which is used to approximate a vacuum boundary in a traditional finite difference method diffusion theory code. This boundary condition, reproduced in the NEM for sample problem comparison purposes

only, takes the following form. The flux/partial current relationship of Eqn 3.28 is set equal to 0 at the problem boundary. This implies that the incoming partial current is equal to the negative of the outgoing partial current, i.e.:  $J_{g(x,y,z)\pm}^{in,l} = -J_{g(x,y,z)\pm}^{out,l}$ .

The quadratic transverse leakage approximation described in section 3.2.1.5 also requires some special treatment for nodes bordering the problem outer boundaries. For example, at the positive (negative) x-direction problem boundary,  $\bar{L}_{gy}^{l+}$ , ( $\bar{L}_{gy}^{l-}$ ) does not exist. This problem is easily remedied at boundaries of reflection by simply specifying  $\bar{L}_{gy}^{l+}(\bar{L}_{gy}^{l-}) \equiv \bar{L}_{gy}^l$  and  $\Delta x^{l+}(\Delta x^{l-}) \equiv \Delta x^l$ . At either zero incoming current or zero flux outer boundaries, the quadratic leakage approximation is constructed such that  $\bar{L}_{gy}^{l+}(\bar{L}_{gy}^{l-}) \equiv 0$  and  $\Delta x^{l+}(\Delta x^{l-}) \equiv 0$ , if the node in question is beyond the boundaries of the system. This boundary node leakage specification, while conceptually exact for the zero flux boundary condition, is only approximately true when the no incoming current boundary condition is invoked. The quadratic leakage approximation is treated similarly on the y and z directed problem outer boundaries.

### 3.2.2 Implementation and Acceleration of the Computational Algorithm

#### 3.2.2.1 Basic Overview of Solution Procedure

The multigroup Nodal Expansion Method outlined in previous sections of this chapter can be numerically solved by the traditional inner/outer iteration multigroup diffusion theory method. That is, for each group, inner iterations or multiple sweeps through the mesh with a known internal source are performed in order to invert the within-group diffusion removal matrix. Outer or fission source iterations are then performed around the inner iterations in order to calculate the correct values of the problem multiplicative eigenvalue ( $K_{eff}$ ) and the space and energy dependent fission neutron source distribution.

Specifically the unaccelerated multigroup Nodal Expansion Method equations are solved in the following sequence. After the problem specifications are read in and processed,  $K_{eff}$  is initially set to 1.0 and a guess, initially a constant, is made for the vector representing the entire right-hand sides of the response matrix equations (Eqn 3.33) for group one in all nodes. Next, while the second and third or internal source terms of Equation 3.33 remain constant, inner iterations are performed on group one using a so-called red-black iterative scheme. This scheme involves first classifying all nodes in the medium as either red nodes or black nodes. This node classification, which appears as a three-dimensional checkerboard, has the property that no two red or black nodes have a common face.

In group one, the inner iteration scheme first involves solving Equation 3.33 for the outgoing partial currents, by means to be detailed in the following subsection, for all nodes of a single color, i.e. the black nodes. Using the continuity of partial current internal boundary condition, the outgoing partial currents from the black nodes are, except on the problem outer boundaries, defined as the incoming partial currents for the red nodes. Using this modified source, Equation 3.33 is then solved for the group one outgoing partial currents from all red nodes. This two-step sequence defines a single group one inner iteration. A fixed number of these two-step inner iterations, usually four, are performed on group one, all the while updating the distribution of incoming partial currents at each half step.

After completion of the group one inner iterations, the node neutron balance equation (Eqn 3.5) is invoked to obtain the node volume average neutron flux. In addition, Equations 3.22 and 3.23 and their y and z direction counterparts, are used to calculate all first and second order flux moments. These known fluxes and flux moments are then used (in Eqn 3.2) to calculate the group 1 to 2 downscatter source and source moments.

Finally, Equations 3.41 and 3.42 and their 10 directional pair counterparts are used to calculate the group one transverse leakage moments to be used in the second outer iteration.

One next proceeds to perform inner iterations on group two. Since this is still the first outer iteration, one only has information on the downscatter source and source moments. Therefore, all incoming currents and transverse leakage moments on the right-hand side of Equation 3.33 are initially set to zero. Next, a fixed number of the previously described red-black inner iterations are performed in order to partially converge the group 2 spatially dependent partial currents. One then calculates the group 2 node average fluxes and flux moments. Finally, the new group 1 sources and source moments and the next-iteration group 2 transverse leakage moments are calculated.

The completion of this task signals the end of one outer iteration. At the end of this or any later outer iteration several tasks are performed to determine whether to proceed to the next outer iteration. First the net fission source in each node is calculated as:

$$\bar{S}^l = \sum_{g=1}^2 \nu \Sigma_{fg}^l \bar{\phi}_g^l \Delta x^l \Delta y^l \Delta z^l \quad (3.44)$$

This fission source vector is then summed over the entire system in order to obtain a global fission source ( $\bar{S}_{total}$ ). Next a new value of the multiplicative eigenvalue,  $K_{eff}$ , is calculated using the following relationship:

$$K_{eff}^n = K_{eff}^{n-1} \cdot \frac{\bar{S}_{total}^n}{\bar{S}_{total}^{n-1}} \quad (3.45)$$

where

$n, n - 1 \equiv$  outer iteration indices

$\bar{S}_{total}^n \equiv$  total fission source calculated at outer iteration n

Finally, if the outer iteration just completed is not the first, several tests are made to determine if the solution has converged. The first test involves observing the absolute change in  $K_{eff}$  since the last outer iteration. To pass the test,  $K_{eff}$  must not change by more than one part in ten million (seven significant digits unchanged). The next two tests evaluate the change in the fission source vector since the last outer iteration. In these tests, the change in the  $L_2$  and  $L_\infty$  norms of the vector must both be less than one part in 100,000 (five significant digits unchanged). Only after all three of these convergence criteria have been met is the problem considered fully converged. If full convergence has not been achieved, one proceeds to perform another outer iteration. At this time, the newly calculated value of  $K_{eff}$  is used in determining the next iteration values of the  $a'_i$ 's in Equation 3.33, while the final values of the incoming partial currents and transverse leakage moments are used in the next outer iteration.

### 3.2.2.2 Solving the Response Matrix Equations

As discussed in the previous subsection, a detailed discussion of the means by which the response matrix equations (Eqn's 3.33) are solved in three-dimensional Cartesian geometry for each node at each inner iteration will be presented.

To begin with, Equation 3.33 will first be written in simplified notation as:

$$[\mathbf{A}] \cdot \mathbf{J} = \mathbf{b} \quad (3.46)$$

where

$$\mathbf{A} = \begin{pmatrix} a_1 & a_2 & a_3 & a_3 & a_4 & a_4 \\ a_2 & a_1 & a_3 & a_3 & a_4 & a_4 \\ a_7 & a_7 & a_5 & a_6 & a_8 & a_8 \\ a_7 & a_7 & a_6 & a_5 & a_8 & a_8 \\ a_{11} & a_{11} & a_{12} & a_{12} & a_9 & a_{10} \\ a_{11} & a_{11} & a_{12} & a_{12} & a_{10} & a_9 \end{pmatrix}$$

$$\mathbf{J} = (J_1, J_2, J_3, J_4, J_5, J_6)^T$$

$$\mathbf{b} = (b_1, b_2, b_3, b_4, b_5, b_6)^T$$

Here the entire right-hand side of Equation 3.33 has been multiplied through and replaced by the  $\mathbf{b}$  vector in Equation 3.46. One can now perform a series of row operations on Equation 3.46 without affecting the values of the solution vector  $\mathbf{J}$ . These row operations, which involve adding linear combinations of rows one and two, three and four, and five and six to one another, result in:

$$[\mathbf{A}'] \cdot \mathbf{J} = \mathbf{b}' \quad (3.47)$$

where

$$\mathbf{A}' = \begin{pmatrix} a_1 + a_2 & a_1 + a_2 & 2a_3 & 2a_3 & 2a_4 & 2a_4 \\ a_2 - a_1 & a_1 - a_2 & 0 & 0 & 0 & 0 \\ 2a_7 & 2a_7 & a_5 + a_6 & a_5 + a_6 & 2a_8 & 2a_8 \\ 0 & 0 & a_6 - a_5 & a_5 - a_6 & 0 & 0 \\ 2a_{11} & 2a_{11} & 2a_{12} & 2a_{12} & a_9 + a_{10} & a_9 + a_{10} \\ 0 & 0 & 0 & 0 & a_{10} - a_9 & a_9 - a_{10} \end{pmatrix}$$

$$\mathbf{b}' = (b_1 + b_2, b_2 - b_1, b_3 + b_4, b_4 - b_3, b_5 + b_6, b_6 - b_5)^T$$

If matrix Equation 3.47 is expanded into its six constitutive coupled linear equations, one can, by observation, rewrite these equations in terms of a new solution vector as:

$$[\mathbf{A}''] \cdot \mathbf{J}' = \mathbf{b}' \quad (3.48)$$

where

$$\mathbf{A}'' = \begin{pmatrix} a_1 + a_2 & 0 & 2a_3 & 0 & 2a_4 & 0 \\ 0 & a_2 - a_1 & 0 & 0 & 0 & 0 \\ 2a_7 & 0 & a_5 + a_6 & 0 & 2a_8 & 0 \\ 0 & 0 & 0 & a_6 - a_5 & 0 & 0 \\ 2a_{11} & 0 & 2a_{12} & 0 & a_9 + a_{10} & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{10} - a_9 \end{pmatrix}$$

$$\mathbf{J}' = (J_1 + J_2, J_1 - J_2, J_3 + J_4, J_3 - J_4, J_5 + J_6, J_5 - J_6)^T$$

Upon observing the independence of rows 1 and 2, 3 and 4, and 5 and 6, one may easily rewrite the 6x6 matrix Equation 3.48 as:

$$[\mathbf{A33}] \cdot \mathbf{J3} = \mathbf{b3} \quad (3.49)$$

where

$$\mathbf{A33} = \begin{pmatrix} a_1 + a_2 & 2a_3 & 2a_4 \\ 2a_7 & a_5 + a_6 & 2a_8 \\ 2a_{11} & 2a_{12} & a_9 + a_{10} \end{pmatrix}$$

$$\mathbf{J3} = (J_1 + J_2, J_3 + J_4, J_5 + J_6)^T$$

$$\mathbf{b3} = (b_1 + b_2, b_3 + b_4, b_5 + b_6)^T$$

and

$$[\mathbf{A33}'] \cdot \mathbf{J3}' = \mathbf{b3}' \quad (3.50)$$

where

$$\mathbf{A33}' = \begin{pmatrix} a_2 - a_1 & 0 & 0 \\ 0 & a_6 - a_5 & 0 \\ 0 & 0 & a_{10} - a_9 \end{pmatrix}$$

$$\mathbf{J3}' = (J_1 - J_2, J_3 - J_4, J_5 - J_6)^T$$

$$\mathbf{b3}' = (b_2 - b_1, b_4 - b_3, b_6 - b_5)^T$$

These simplified 3x3 matrices are the ones that are actually used in finding the six face averaged outgoing partial currents for each node at each inner iteration and each energy group.

The actual response matrix solution process proceeds as follows. First, at the beginning of each outer iteration the  $\mathbf{A33}$  matrix, which depends on node dimensions, cross sections, and the iteration n eigenvalue estimate ( $K_{eff}^n$ ) is calculated for all nodes at each energy group. Next the  $\mathbf{A33}$  matrices are numerically decomposed into lower triangular matrices (  $\mathbf{L33}$  ) and upper triangular matrices (  $\mathbf{U33}$  ) using a standard

Gaussian Elimination routine with scaled partial pivoting (30). During the actual red-black mesh sweeps, the  $\mathbf{J3}$  vector in Equation 3.49 is continuously updated by simple back substitution. The six separate values of the outgoing partial current from each node are easily obtained from length 3  $\mathbf{J3}$  vectors by employing the three independent relationships that make up Equation 3.50. These relationships yield:

$$J_1 = \frac{(J_1 + J_2)}{2} + \frac{b_1 - b_2}{2(a_1 - a_2)} \quad (3.51)$$

$$J_2 = \frac{(J_1 + J_2)}{2} + \frac{b_2 - b_1}{2(a_1 - a_2)} \quad (3.52)$$

$$J_3 = \frac{(J_3 + J_4)}{2} + \frac{b_3 - b_4}{2(a_5 - a_6)} \quad (3.53)$$

$$J_4 = \frac{(J_3 + J_4)}{2} + \frac{b_4 - b_3}{2(a_5 - a_6)} \quad (3.54)$$

$$J_5 = \frac{(J_5 + J_6)}{2} + \frac{b_5 - b_6}{2(a_9 - a_{10})} \quad (3.55)$$

$$J_6 = \frac{(J_5 + J_6)}{2} + \frac{b_6 - b_5}{2(a_9 - a_{10})} \quad (3.56)$$

As one can readily observe, solving Equations 3.49 and 3.51 through 3.56 for the outgoing partial currents in each node requires much less computational effort per iteration than would a direct solution of the original 6x6 response matrix equation given in Eqn 3.46.

### 3.2.2.3 Coarse-Mesh Rebalance Acceleration

During the course of executing various steady-state sample problems it was discovered that the basic NEM solution methodology inevitably requires an excessive number of outer iterations, approximately 350 in a typical problem, in order to meet the previously mentioned convergence criteria on  $K_{eff}$  and the fission source vector. As a result, methods to speed up or accelerate the convergence of the outer or fission source iterations were reviewed. Upon reviewing a variety of iterative acceleration methods, it

was decided that coarse-mesh rebalancing in conjunction with asymptotic extrapolation would be the most effective method for speeding up the basic NEM. Both the coarse-mesh rebalancing method and the asymptotic extrapolation method have proven to be very effective means of accelerating the convergence of iterative numerical solutions to the neutron transport equation, the neutron diffusion equation, and a variety of fluid dynamics problems (25).

The coarse-mesh rebalancing technique basically works by computing a system eigenvalue and a series of neutron flux/partial current scale factors. When applied, these scale factors will force a neutron balance over a set of cells, each of which is arbitrarily made up of one or more of the originally defined fine mesh cells. The coarse-mesh rebalancing method can be derived by first heuristically defining the overall neutron balance equation for an arbitrary coarse node  $m$ :

$$\begin{aligned} & - \sum_{l \in S_m} \Delta^2(x, y, z) \sum_{g=1}^2 J_g^{in,l} + \sum_{m \in S_m} \Delta^2(x, y, z) \sum_{g=1}^2 J_g^{out,l} \\ & + \sum_{m \in V_m} \Delta x \Delta y \Delta z \sum_{g=1}^2 \Sigma_{ag}^m \bar{\phi}_g^m = \frac{1}{K_{eff}} \sum_{m \in V_m} \Delta x \Delta y \Delta z \sum_{g=1}^2 \nu \Sigma_{fg}^m \bar{\phi}_g^m \end{aligned} \quad (3.57)$$

The first term on the left-hand side of this eigenvalue equation represents the net influx of neutrons into coarse node  $m$  from neighboring nodes. The second term represents the net neutron outflow across the faces of cell  $m$ . The last term on the left represents the total volumetric absorption losses in cell  $m$ . Finally, the term on the right-hand side of Equation 3.57 represents the net volumetric fission production rate in cell  $m$ . This balance equation will not be exactly satisfied if the outer iterations have not converged. This is a result of the numerical procedure in which a previous iteration fission source is used to calculate present iteration fluxes and partial currents.

The balance equation can, however, be satisfied artificially by defining a set of rebalance factors. These factors modify the coarse-mesh fluxes and outgoing partial currents as follows:

$$\bar{\phi}_g^{lm} \equiv \bar{\phi}_g^m \cdot f^m \quad (3.58)$$

$$J_{g(x,y,z)}^{lout,m} \equiv J_{g(x,y,z)}^{out,m} \cdot f^m \quad (3.59)$$

These rebalance factors are calculated by the following eigenvalue equation:

$$\begin{aligned} & -f^l \cdot \sum_{l \in S_m} \Delta^2(x, y, z) \sum_{g=1}^2 J_{g(x,y,z)}^{in,l} \\ & + f^m \cdot \sum_{m \in S_m} \Delta^2(x, y, z) \sum_{g=1}^2 J_{g(x,y,z)}^{out,m} \\ & + f^m \cdot \sum_{m \in V_m} \Delta x \Delta y \Delta z \sum_{g=1}^2 \Sigma_{ag}^m \bar{\phi}_g^m \\ & = f^m \cdot \frac{1}{\lambda} \sum_{m \in V_m} \Delta x \Delta y \Delta z \sum_{g=1}^2 \nu \Sigma_{fg}^m \bar{\phi}_g^m \end{aligned} \quad (3.60)$$

The calculated rebalance factors will all tend towards 1 as the outer iterations converge.

This equation can be written in simplified matrix form as:

$$[M] \underline{f} = \frac{1}{\lambda} [P] \underline{f} \quad (3.61)$$

where

$\underline{f}$   $\equiv$  vector of rebalance factors

$[M]$   $\equiv$  seven stripe matrix representing L.H.S. of Eqn 3.60

$[P]$   $\equiv$  diagonal matrix representing R.H.S. of Eqn 3.60

This eigenvalue problem can be solved quite efficiently by the Weilant method of fractional iteration (31). Here the well known power iteration method for the numerical solution of eigenvalue problems (31) is applied to a so-called shifted eigenvalue problem:

$$[M'] = \frac{1}{\lambda'} [P] \underline{f} \quad (3.62)$$

where

$$[M'] \equiv [M] - \frac{1}{\lambda_e} [P]$$

$$\frac{1}{\lambda'} \equiv \frac{1}{\lambda} - \frac{1}{\lambda_e}$$

Here  $\lambda_e$  is an estimate of  $K_{eff}$  or the problem fundamental mode eigenvalue. In this analysis  $\lambda_e$  is defined such that:

$$\lambda_e = 1.05 \cdot K_{eff}^{n-1} \quad (3.63)$$

where

$K_{eff}^{n-1} \equiv$  the estimate of  $K_{eff}$  from the previous outer iteration

The convergence rate of solving Equation 3.62 by the power iteration method is determined by the dominance ratio ( $\sigma$ ) of the matrix  $[M']^{-1} [P]$ , where  $\sigma \equiv |\frac{\lambda_1}{\lambda_0}|$ . Here  $\lambda_0$  and  $\lambda_1$  represent, respectively, the largest and second largest eigenvalues of the matrix under consideration. It can be shown (31), that for  $\lambda_e > K_{eff}$ , the dominance ratio of  $[M']^{-1} [P]$  is much smaller than that of  $[M]^{-1} [P]$ . Therefore, the power iteration method will converge much faster for Equation 3.62 than it will for Equation 3.61. The power iteration method is employed in the solution of Equation 3.62 by using the following sequence:

$$\underline{S}^{(t)} = [M']^{-1} [P] \underline{f}^{(t-1)} \quad (3.64)$$

$$\lambda'^{(t)} = \frac{\| [P] \cdot \underline{S}^{(t)} \|_1}{\| [P] \cdot \underline{f}^{(t-1)} \|_1} \quad (3.65)$$

$$\underline{f}^{(t)} = \frac{1}{\lambda'^{(t)}} \underline{S}^{(t)} \quad (3.66)$$

where

$\| \cdot \|_1 \equiv L_1$  vector norm

$t \equiv$  power iteration index

Finally, after convergence of the power iterations has been achieved, as verified by testing the  $L_\infty$  norm of the  $\underline{S}$  vector components between successive iterations, all fluxes and outgoing partial currents within the fine nodes contained within coarse node  $m$  are multiplied by  $f^m$  while the new estimate of the multiplicative eigenvalue ( $K_{eff}$ ) at outer iteration  $n$  becomes  $\lambda^n$ , where  $\frac{1}{\lambda^n} = \frac{1}{\lambda^f} + \frac{1}{\lambda_e}$ .

In operation, using a coarse mesh of approximately 8 fine nodes per coarse node, the coarse-mesh rebalancing method reduces the number of outer iterations required for problem convergence by approximately a factor of five. This increase in overall computational efficiency much more than compensates for the overhead required to iteratively solve for the coarse-mesh rebalance factors and eigenvalue.

### 3.2.2.4 Asymptotic Extrapolation Acceleration

As discussed in the preceding subsection, an asymptotic source extrapolation procedure is used in conjunction with coarse-mesh rebalancing to accelerate convergence of the outer iterations. This rather simple extrapolation procedure is based on the assumption that the fission source vector  $\bar{S}_{total}^n$  converges to its fully converged solution  $\bar{S}_{total}^\infty$  with asymptotic behavior, i.e.:

$$\bar{S}_{total}^\infty = \bar{S}_{total}^n + \underline{R}\sigma^n \quad (3.67)$$

where

$\bar{S}_{total}^n \equiv$  fission source vector at outer iteration  $n$

$\sigma^n \equiv$  the dominance ratio of the matrix which yields

the multigroup diffusion theory flux solution

$\underline{R} \equiv$  an unknown vector

one can estimate the dominance ratio  $\sigma^n$  by the following relationship:

$$\sigma^n \cong \frac{\|\bar{S}_{total}^n - \bar{S}_{total}^{n-1}\|_2}{\|\bar{S}_{total}^{n-1} - \bar{S}_{total}^{n-2}\|_2} \quad (3.68)$$

where

$$\|x\|_2 \equiv L_2 \text{ norm of vector } x$$

Combining Equations 3.67 and 3.68 one can produce an improved estimate of the fission source vector  $\bar{S}_{total}^n$ :

$$\bar{S}'_{total}^n = \bar{S}_{total}^n + \omega^n (\bar{S}_{total}^n - \bar{S}_{total}^{n-1}) \quad (3.69)$$

where

$$\omega^n \equiv \frac{\sigma^n}{1 - \sigma^n}$$

This asymptotic extrapolation factor is applied to both iteration n outgoing partial currents and node average sources only when true asymptotic convergence behavior has been substantiated. Therefore extrapolation is only performed when the following criteria:

$$\min [\varepsilon_\omega^n, \varepsilon_\omega^{n-1}] < 0.1 \quad (3.70)$$

where

$$\varepsilon_\omega^n = \left| \frac{\omega^n - \omega^{n-1}}{\omega^n} \right|$$

is satisfied and five outer iterations have passed since the most recent extrapolation. This latest calculated value of  $\sigma^n$  is then applied to the iteration n node flux moments and outgoing partial vectors before the start of iteration n+1 in accordance with:

$$\bar{\phi}_g'^{l(n)} = \bar{\phi}_g^{l(n)} + \omega^n (\bar{\phi}_g^{l(n)} - \bar{\phi}_g^{l(n-1)}) \quad (3.71)$$

and

$$J_g'^{out,l(n)} = J_g^{out,l(n)} + \omega^n (J_g^{out,l(n)} - J_g^{out,l(n-1)}) \quad (3.72)$$

When applied to an actual NEM calculation, asymptotic extrapolation reduces the number of outer iterations by about a factor of two over and above the speedup due to coarse-mesh rebalancing. Therefore the combination of coarse-mesh rebalancing and asymptotic extrapolation, for most realistic problems, reduces the number of outer iterations by at least an order of magnitude, greatly increasing the computational efficiency of the Nodal Expansion Method.

### 3.3 Cylindrical Geometry

#### 3.3.1 Derivation of Three-Dimensional Numerics

The following section outlines the details of deriving the three-dimensional Nodal Expansion Method in cylindrical geometry. This derivation will mainly concentrate on differences between the cylindrical geometry methodology and the previously derived methodology in Cartesian geometry.

To begin this derivation, the two energy group diffusion equation within node 1 must be rewritten in cylindrical geometry. Equation 3.2 can therefore be rewritten as:

$$\begin{aligned} -D_g^l \frac{1}{r} \frac{\partial}{\partial r} \left( r \frac{\partial}{\partial r} \phi_g^l(r, \theta, z) \right) - D_g^l \frac{1}{r^2} \frac{\partial^2}{\partial \theta^2} \phi_g^l(r, \theta, z) \\ - D_g^l \frac{\partial^2}{\partial z^2} \phi_g^l(r, \theta, z) + A_g^l \phi_g^l(r, \theta, z) = Q_g^l(r, \theta, z) \end{aligned} \quad (3.73)$$

$(r, \theta, z) \in V^l, g = 1, 2$

where

$$\begin{aligned} A_1^l &= \Sigma_{a1}^l + \Sigma_{12}^l - \frac{1}{K} \nu \Sigma_{f1}^l \\ A_2^l &= \Sigma_{a2}^l \\ Q_1^l(r, \theta, z) &= \frac{1}{K} \nu \Sigma_{f2}^l \phi_2^l(r, \theta, z) \\ Q_2^l(r, \theta, z) &= \Sigma_{12}^l \phi_1^l(r, \theta, z) \end{aligned}$$

$$V^l = \frac{1}{2} (R_{out}^2 - R_{in}^2) \Delta\theta \Delta z \equiv \text{Volume of node } l$$

$\Sigma_{ag}$   $\equiv$  Group g absorption cross-section

$\Sigma_{12}$   $\equiv$  Group 1 to 2 scattering cross-section

$R_{in}, R_{out}$   $\equiv$  inner/outer radius of node l

In cylindrical geometry Fick's law may be used to express the three components of the current vector as:

$$j_{gr}^l(r, \theta, z) = -D_g^l \frac{\partial}{\partial r} \phi_g^l(r, \theta, z) \quad (3.74)$$

$$j_{g\theta}^l(r, \theta, z) = -D_g^l \frac{1}{r} \frac{\partial}{\partial \theta} \phi_g^l(r, \theta, z) \quad (3.75)$$

$$j_{gz}^l(r, \theta, z) = -D_g^l \frac{\partial}{\partial z} \phi_g^l(r, \theta, z) \quad (3.76)$$

Equation 3.73 thus may be rewritten as:

$$\begin{aligned} & \frac{1}{r} \frac{\partial}{\partial r} (r j_{gr}^l(r, \theta, z)) + \frac{1}{r} \frac{\partial}{\partial \theta} j_{g\theta}^l(r, \theta, z) + \frac{\partial}{\partial z} j_{gz}^l(r, \theta, z) \\ & + A_g^l \phi_g^l(r, \theta, z) = Q_g^l(r, \theta, z) \end{aligned} \quad (3.77)$$

$$(r, \theta, z) \in V^l$$

### 3.3.1.1 The Cylindrical Nodal Balance Equation

Assuming the coordinate origin to be at the center of cell l, Equation 3.77 can be integrated over node l to obtain:

$$\begin{aligned} & \frac{1}{R \Delta r} \left( \left( R + \frac{\Delta r}{2} \right) J_{gr+}^l - \left( R - \frac{\Delta r}{2} \right) J_{gr-}^l \right) \\ & + \frac{1}{R \Delta \theta} (J_{g\theta+}^l - J_{g\theta-}^l) + \frac{1}{\Delta z} (J_{gz+}^l - J_{gz-}^l) + A_g^l \bar{\phi}_g^l = \bar{Q}_g^l \end{aligned} \quad (3.78)$$

where

$R \equiv$  radius from coordinate centerline to a point midway between  $R_{in}$  and  $R_{out}$

$$\bar{\phi}_g^l = \frac{1}{V^l} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta\theta/2}^{\Delta\theta/2} \int_{-\Delta z/2}^{\Delta z/2} \phi_g^l(r, \theta, z) r dr d\theta dz$$

$\equiv$  node volume average flux

$$\bar{Q}_g^l = \frac{1}{V^l} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta\theta/2}^{\Delta\theta/2} \int_{-\Delta z/2}^{\Delta z/2} Q_g^l(r, \theta, z) r dr d\theta dz$$

$\equiv$  node volume average source

$$\begin{aligned} & \frac{1}{R\Delta r} \left( \left( R + \frac{\Delta r}{2} \right) J_{gr+}^l - \left( R - \frac{\Delta r}{2} \right) J_{gr-}^l \right) \\ &= \frac{1}{V^l} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta\theta/2}^{\Delta\theta/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{1}{r} \left( \frac{\partial}{\partial r} r j_{gr}^l(r, \theta, z) \right) r dr d\theta dz \end{aligned}$$

$J_{gr\pm}^l \equiv$  average  $r$ -directed net current on node faces  $R \pm \frac{\Delta r}{2}$

$$\begin{aligned} & \frac{1}{R\Delta\theta} (J_{g\theta+}^l - J_{g\theta-}^l) \\ &= \frac{1}{V^l} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta\theta/2}^{\Delta\theta/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{1}{r} \frac{\partial}{\partial\theta} j_{g\theta}^l(r, \theta, z) r dr d\theta dz \end{aligned}$$

$J_{g\theta\pm}^l \equiv$  average  $\theta$ -directed net current on node faces  $\pm \frac{\Delta\theta}{2}$

$$\begin{aligned} & \frac{1}{\Delta z} (J_{gz+}^l - J_{gz-}^l) \\ &= \frac{1}{V^l} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta\theta/2}^{\Delta\theta/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{\partial}{\partial z} j_{gz}^l(r, \theta, z) r dr d\theta dz \end{aligned}$$

$J_{gz\pm}^l \equiv$  average  $z$ -directed net current on node faces  $\pm \frac{\Delta z}{2}$

For central nodes, i.e. where  $R_{in} = 0$  and  $R \equiv \frac{\Delta r}{2}$ , one can see that  $J_{gr-}^l$  drops out of the nodal balance equation.

### 3.3.1.2 The Transverse Integration Procedure

In order to develop a relationship between the node average flux and the face average net currents in Equation 3.78, one must first transversely integrate Equation

3.77 in directions perpendicular to the direction of interest. The r-directed transverse integrated diffusion equation thus becomes:

$$\frac{1}{r} \frac{d}{dr} (r j_{gr}^l(r)) + A_g^l \phi_{gr}^l(r) = Q_{gr}^l(r) - \frac{1}{r \Delta \theta} L_{g\theta}^l(r) - \frac{1}{\Delta z} L_{gz}^l(r) \quad (3.79)$$

where

$$\begin{aligned} \phi_{gr}^l(r) &= \frac{1}{\Delta \theta \Delta z} \int_{-\Delta \theta/2}^{\Delta \theta/2} \int_{-\Delta z/2}^{\Delta z/2} \phi_g^l(r, \theta, z) dz d\theta \\ \frac{1}{r} \frac{d}{dr} (r j_{gr}^l(r)) &= \frac{1}{\Delta \theta \Delta z} \int_{-\Delta \theta/2}^{\Delta \theta/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{1}{r} \frac{\partial}{\partial r} (r j_{gr}^l(r, \theta, z)) dz d\theta \\ Q_{gr}^l(r) &= \frac{1}{\Delta \theta \Delta z} \int_{-\Delta \theta/2}^{\Delta \theta/2} \int_{-\Delta z/2}^{\Delta z/2} Q_g^l(r, \theta, z) dz d\theta \\ L_{g\theta}^l(r) &= \frac{1}{\Delta z} \int_{-\Delta \theta/2}^{\Delta \theta/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{\partial}{\partial \theta} j_{g\theta}^l(r, \theta, z) dz d\theta \\ &\equiv \theta\text{-direction transverse leakage} \\ L_{gz}^l(r) &= \frac{1}{\Delta \theta} \int_{-\Delta \theta/2}^{\Delta \theta/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{\partial}{\partial z} j_{gz}^l(r, \theta, z) dz d\theta \\ &\equiv z\text{-direction transverse leakage} \end{aligned}$$

The  $\theta$ -direction transverse integrated diffusion equation can be written as:

$$\frac{1}{R} \frac{d}{d\theta} j_{g\theta}^l(\theta) + A_g^l \phi_{g\theta}^l(\theta) = Q_{g\theta}^l(\theta) - \frac{1}{\Delta r} L_{gr}^l(\theta) - \frac{1}{\Delta z} L_{gz}^l(\theta) \quad (3.80)$$

where

$$\begin{aligned} \phi_{g\theta}^l(\theta) &= \frac{1}{R \Delta r \Delta z} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta z/2}^{\Delta z/2} \phi_g^l(r, \theta, z) dz dr \\ \frac{1}{R} \frac{d}{d\theta} j_{g\theta}^l(\theta) &= \frac{1}{R \Delta r \Delta z} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{1}{r} \frac{\partial}{\partial \theta} j_{g\theta}^l(r, \theta, z) dz dr \\ Q_{g\theta}^l(\theta) &= \frac{1}{R \Delta r \Delta z} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta z/2}^{\Delta z/2} Q_g^l(r, \theta, z) dz dr \end{aligned}$$

$$L_{gr}^l(\theta) = \frac{1}{R\Delta z} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{1}{r} \frac{\partial}{\partial r} (r j_{gr}^l(r, \theta, z)) dz dr \\ \equiv r\text{-direction transverse leakage}$$

$$L_{gz}^l(\theta) = \frac{1}{R\Delta r} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta z/2}^{\Delta z/2} \frac{\partial}{\partial z} j_{gz}^l(r, \theta, z) dz dr \\ \equiv z\text{-direction transverse leakage}$$

Finally, the z-direction transverse integrated diffusion equation can be written as:

$$\frac{d}{dz} j_{gz}^l(z) + A_g^l \phi_{gz}^l(z) = Q_{gz}^l(z) - \frac{1}{R\Delta\theta} L_{g\theta}^l(z) - \frac{1}{\Delta r} L_{gr}^l(z) \quad (3.81)$$

where

$$\phi_{gz}^l(z) = \frac{1}{R\Delta r\Delta\theta} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta\theta/2}^{\Delta\theta/2} \phi_g^l(r, \theta, z) d\theta dr \\ \frac{d}{dz} j_{gz}^l(z) = \frac{1}{R\Delta r\Delta\theta} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta\theta/2}^{\Delta\theta/2} \frac{\partial}{\partial z} j_{gz}^l(r, \theta, z) d\theta dr \\ Q_{gz}^l(z) = \frac{1}{R\Delta r\Delta\theta} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta\theta/2}^{\Delta\theta/2} Q_g^l(r, \theta, z) d\theta dr \\ L_{gr}^l(z) = \frac{1}{R\Delta\theta} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta\theta/2}^{\Delta\theta/2} \frac{1}{r} \frac{\partial}{\partial r} (r j_{gr}^l(r, \theta, z)) d\theta dr \\ \equiv r\text{-direction transverse leakage} \\ L_{g\theta}^l(z) = \frac{1}{\Delta r} \int_{R-\Delta r/2}^{R+\Delta r/2} \int_{-\Delta\theta/2}^{\Delta\theta/2} \frac{1}{r} \frac{\partial}{\partial\theta} j_{g\theta}^l(r, \theta, z) d\theta dr \\ \equiv \theta\text{-direction transverse leakage}$$

### 3.3.1.3 Polynomial Approximation of the One-Dimensional Equations

As was the case in Cartesian coordinates, The Nodal Expansion Method requires that the one-dimensional fluxes in Equations 3.79, 3.80, and 3.81 be expanded in a series of polynomials, i.e.:

$$\phi_{gr}^l(r) = \bar{\phi}_g^l + \sum_{n=1}^N a_{grn}^l f_n(r) \quad (3.82)$$

$$\phi_{g\theta}^l(\theta) = \bar{\phi}_g^l + \sum_{n=1}^N a_{g\theta n}^l f_n(\theta) \quad (3.83)$$

$$\phi_{gz}^l(z) = \bar{\phi}_g^l + \sum_{n=1}^N a_{gz n}^l f_n(z) \quad (3.84)$$

Since the three coordinate directions in cylindrical geometry are not equivalent, one must develop separate sequences of four basis functions for the r,  $\theta$ , and z directions.

For the r direction these basis functions are:

$$f_1(r) = \frac{r - R}{\Delta r} \quad (3.85)$$

$$f_2(r) = 3 \left( \frac{r - R}{\Delta r} \right)^2 - \frac{1}{4} \quad (3.86)$$

$$f_3(r) = \left( \frac{r - R}{\Delta r} \right)^3 - \frac{1}{4} \left( \frac{r - R}{\Delta r} \right) \quad (3.87)$$

$$f_4(r) = \left( \frac{r - R}{\Delta r} \right)^4 - \frac{3}{10} \left( \frac{r - R}{\Delta r} \right)^2 + \frac{1}{80} \quad (3.88)$$

These basis functions satisfy:

$$\frac{1}{\Delta r} \int_{R-\Delta r/2}^{R+\Delta r/2} f_n(r) dr = 0, n = 1, 4 \quad (3.89)$$

$$f_n \left( R \pm \frac{\Delta r}{2} \right) = 0, n = 3, 4 \quad (3.90)$$

It would be more correct in cylindrical geometry to include a factor of r within the integral of Equation 3.89. An approximation was made in deleting this factor such that a single set of r-direction basis functions (Eqn's 3.85, 3.86, 3.87, and 3.88) could be used in all homogeneous nodes, including those at the problem center.

The basis functions in the  $\theta$  direction are:

$$f_1(\theta) = \frac{\theta}{\Delta\theta} \quad (3.91)$$

$$f_2(\theta) = 3 \left( \frac{\theta}{\Delta\theta} \right)^2 - \frac{1}{4} \quad (3.92)$$

$$f_3(\theta) = \left(\frac{\theta}{\Delta\theta}\right)^3 - \frac{1}{4} \left(\frac{\theta}{\Delta\theta}\right) \quad (3.93)$$

$$f_4(\theta) = \left(\frac{\theta}{\Delta\theta}\right)^4 - \frac{3}{10} \left(\frac{\theta}{\Delta\theta}\right)^2 + \frac{1}{80} \quad (3.94)$$

These basis functions satisfy:

$$\frac{1}{\Delta\theta} \int_{-\Delta\theta/2}^{\Delta\theta/2} f_n(\theta) d\theta = 0, n = 1, 4 \quad (3.95)$$

$$f_n\left(\frac{\pm\Delta\theta}{2}\right) = 0, n = 3, 4 \quad (3.96)$$

The basis functions in the z-direction are the same as Equations 3.91 through 3.94 for the  $\theta$  direction with  $z$  and  $\Delta z$  substituted for  $\theta$  and  $\Delta\theta$ , respectively.

One can easily obtain the first two expansion coefficients in any of the three coordinate directions by simply evaluating Equation 3.82, 3.83, or 3.84 at the respective endpoints:  $R \pm \frac{\Delta r}{2}, \pm \frac{\Delta\theta}{2}$ , or  $\pm \frac{\Delta z}{2}$ . These first two expansion coefficients can be expressed as:

$$a_{g(r,\theta,z)}^l = \phi_{g(r,\theta,z)+}^l - \phi_{g(r,\theta,z)-}^l \quad (3.97)$$

$$a_{g(r,\theta,z)}^l = \phi_{g(r,\theta,z)+}^l + \phi_{g(r,\theta,z)-}^l - 2\bar{\phi}_g^l \quad (3.98)$$

where

$$\phi_{g(r,\theta,z)\pm}^l \equiv (r, \theta, z) \text{-direction transverse integrated flux evaluated at node endpoints}$$

In order to find the higher (third and fourth) order flux expansion coefficients in the  $r$ ,  $\theta$ , and  $z$  directions, as in Cartesian geometry, one employs the moments weighting scheme. Moments weighting can be expressed in the  $r$  direction as:

$$\begin{aligned} & \langle w_n(r), \frac{1}{r} \frac{d}{dr} (r j_{gr}^l(r)) + A_g^l \phi_{gr}^l(r) - Q_{gr}^l(r) \\ & + \frac{1}{r\Delta\theta} L_{g\theta}^l(r) + \frac{1}{\Delta z} L_{gz}^l(r) \rangle = 0 \end{aligned} \quad (3.99)$$

In the  $\theta$  direction moments weighting is expressed as:

$$\begin{aligned} & \langle w_n(\theta), \frac{1}{R} \frac{d}{d\theta} j_{g\theta}^l(\theta) + A_g^l \phi_{g\theta}^l(\theta) - Q_{g\theta}^l(\theta) \\ & + \frac{1}{\Delta r} L_{gr}^l(\theta) + \frac{1}{\Delta z} L_{gz}^l(\theta) \rangle = 0 \end{aligned} \quad (3.100)$$

While, in the  $z$  direction, moments weighting can be expressed as:

$$\begin{aligned} & \langle w_n(z), \frac{d}{dz} j_{gz}^l(z) + A_g^l \phi_{gz}^l(z) - Q_{gz}^l(z) \\ & + \frac{1}{R \Delta \theta} L_{g\theta}^l(z) + \frac{1}{\Delta r} L_{gr}^l(z) \rangle = 0 \end{aligned} \quad (3.101)$$

To obtain expressions for the third and fourth order expansion coefficients one needs only to evaluate the flux-like term in the moments weighting expressions, i.e.:

$$\bar{\phi}_{grn}^l \equiv \langle w_n(r), \phi_{gr}^l(r) \rangle = \frac{1}{\Delta r} \int_{R-\Delta r/2}^{R+\Delta r/2} w_n(r) \phi_{gr}^l(r) dr \quad (3.102)$$

$$\bar{\phi}_{g\theta n}^l \equiv \langle w_n(\theta), \phi_{g\theta}^l(\theta) \rangle = \frac{1}{\Delta \theta} \int_{\Delta \theta/2}^{\Delta \theta/2} w_n(\theta) \phi_{g\theta}^l(\theta) d\theta \quad (3.103)$$

$$\bar{\phi}_{gz n}^l \equiv \langle w_n(z), \phi_{gz}^l(z) \rangle = \frac{1}{\Delta z} \int_{\Delta z/2}^{\Delta z/2} w_n(z) \phi_{gz}^l(z) dz \quad (3.104)$$

Substituting in the polynomial flux expansions and basis functions then performing the above integrations yields the following for the third and fourth order expansion coefficients:

$$a_{g(r,\theta,z)3}^l = 10a_{g(r,\theta,z)1}^l - 120\bar{\phi}_{g(r,\theta,z)1}^l \quad (3.105)$$

$$a_{g(r,\theta,z)4}^l = 35a_{g(r,\theta,z)2}^l - 700\bar{\phi}_{g(r,\theta,z)2}^l \quad (3.106)$$

One must now evaluate the entire expressions for the flux moments. The  $\theta$  and  $z$  direction flux moments can be easily evaluated since the integrations needed to evaluate

these quantities have the same form as those previously found in the x direction of Cartesian geometry. Therefore the  $\theta$  and z moment equations can be written as:

$$\begin{aligned}\bar{\phi}_{g\theta 1}^l &= -\frac{1}{A_g^l} \left( \frac{1}{2R\Delta\theta} (J_{g\theta+}^l + J_{g\theta-}^l) + \frac{D_g^l}{R^2\Delta\theta^2} (\phi_{g\theta+}^l - \phi_{g\theta-}^l) \right. \\ &\quad \left. - \bar{Q}_{g\theta 1}^l + \frac{1}{\Delta z} L_{gz\theta 1}^l + \frac{1}{\Delta r} L_{gr\theta 1}^l \right) \end{aligned}\quad (3.107)$$

$$\begin{aligned}\bar{\phi}_{g\theta 2}^l &= -\frac{1}{A_g^l} \left( \frac{1}{2R\Delta\theta} (J_{g\theta+}^l - J_{g\theta-}^l) + \frac{3D_g^l}{R^2\Delta\theta^2} (\phi_{g\theta+}^l + \phi_{g\theta-}^l - 2\bar{\phi}_g^l) \right. \\ &\quad \left. - \bar{Q}_{g\theta 2}^l + \frac{1}{\Delta z} L_{gz\theta 2}^l + \frac{1}{\Delta r} L_{gr\theta 2}^l \right) \end{aligned}\quad (3.108)$$

$$\begin{aligned}\bar{\phi}_{gz 1}^l &= -\frac{1}{A_g^l} \left( \frac{1}{2\Delta z} (J_{gz+}^l + J_{gz-}^l) + \frac{D_g^l}{\Delta z^2} (\phi_{gz+}^l - \phi_{gz-}^l) \right. \\ &\quad \left. - \bar{Q}_{gz 1}^l + \frac{1}{R\Delta\theta} L_{g\theta z 1}^l + \frac{1}{\Delta r} L_{gr z 1}^l \right) \end{aligned}\quad (3.109)$$

$$\begin{aligned}\bar{\phi}_{gz 2}^l &= -\frac{1}{A_g^l} \left( \frac{1}{2\Delta z} (J_{gz+}^l - J_{gz-}^l) + \frac{3D_g^l}{\Delta z^2} (\phi_{gz+}^l + \phi_{gz-}^l - 2\bar{\phi}_g^l) \right. \\ &\quad \left. - \bar{Q}_{gz 2}^l + \frac{1}{R\Delta\theta} L_{g\theta z 2}^l + \frac{1}{\Delta r} L_{gr z 2}^l \right) \end{aligned}\quad (3.110)$$

Because of the integrations involved, evaluating the r-direction moment equations proves to be a more difficult task. The first or current derivative term needed to evaluate the first r moment equation can be expressed as:

$$\begin{aligned}&\langle w_1(r), \frac{1}{r} \frac{d}{dr} (rj_{gr}^l(r)) \rangle \\ &= \frac{1}{\Delta r} \int_{R-\Delta r/2}^{R+\Delta r/2} \left( \frac{r-R}{\Delta r} \right) \left( \frac{1}{r} \frac{d}{dr} (rj_{gr}^l(r)) \right) dr \end{aligned}\quad (3.111)$$

performing integration by parts one obtains without any approximation:

$$\begin{aligned}&\langle w_1(r), \frac{1}{r} \frac{d}{dr} (rj_{gr}^l(r)) \rangle \\ &= \frac{1}{2\Delta r} (J_{gr+}^l + J_{gr-}^l) - \frac{R}{\Delta r^2} \int_{R-\Delta r/2}^{R+\Delta r/2} \frac{1}{r} j_{gr}^l(r) dr \end{aligned}\quad (3.112)$$

At this point some assumption must be made in order to solve the unknown integral in Equation 3.112. After some initial trial and error it was discovered that the assumption

which yields the most accurate results is to simply define  $\frac{1}{r}$  within a node as the constant  $\frac{1}{R}$ , where  $R \equiv$  the average radius of the node. Using this assumption Equation 3.112 becomes:

$$\langle w_1(r), \frac{1}{r} \frac{d}{dr} (rj_{gr}^l(r)) \rangle = \frac{1}{2\Delta r} (J_{gr+}^l + J_{gr-}^l) + \frac{D_g^l}{\Delta r^2} (\phi_{gr+}^l - \phi_{gr-}^l) \quad (3.113)$$

Using this result, the first order r-direction moment equation can be written as:

$$\begin{aligned} \bar{\phi}_{gr1}^l &= -\frac{1}{A_g^l} \left( \frac{1}{2\Delta r} (J_{gr+}^l + J_{gr-}^l) + \frac{D_g^l}{\Delta r^2} (\phi_{gr+}^l - \phi_{gr-}^l) \right) \\ &\quad - \bar{Q}_{gr1}^l + \frac{1}{R\Delta\theta} L_{g\theta r1}^l + \frac{1}{\Delta z} L_{gz r1}^l \end{aligned} \quad (3.114)$$

Upon observation, one can easily see that approximating  $\frac{1}{r}$  as  $\frac{1}{R}$  results in a first order r moment equation with the same form as those in the  $\theta$  (Eqn 3.107) and  $z$  (Eqn 3.109) directions. A second order r moment equation will be derived in the next chapter using various approximations to solve unknown integrals.

### 3.3.1.4 The Response Matrix Equations

At this point one has enough information to produce the response matrix equations for a cylindrical geometry NEM scheme which uses a fourth order polynomial flux expansion in the  $\theta$  and  $z$  directions and a third order polynomial flux expansion in the  $r$  direction. As was the case in Cartesian geometry, the response matrix equations are derived from the Fick's law expressions for the partial currents on the node boundaries. Fick's law takes on the following forms on the six faces of node l:

$$J_{gr+}^{out,l} = J_{gr+}^{in,l} - D_g^l \frac{d}{dr} \phi_{gr}^l(r) |_{r=R+\Delta r/2} \quad (3.115)$$

$$J_{gr-}^{out,l} = J_{gr-}^{in,l} + D_g^l \frac{d}{dr} \phi_{gr}^l(r) |_{r=R-\Delta r/2} \quad (3.116)$$

$$J_{g\theta+}^{out,l} = J_{g\theta+}^{in,l} - \frac{D_g^l}{R} \frac{d}{d\theta} \phi_{g\theta}^l(\theta) |_{\theta=\Delta\theta/2} \quad (3.117)$$

$$J_{g\theta-}^{out,l} = J_{g\theta-}^{in,l} + \frac{D_g^l}{R} \frac{d}{d\theta} \phi_{g\theta}^l(\theta) |_{\theta=-\Delta\theta/2} \quad (3.118)$$

$$J_{gz+}^{out,l} = J_{gz+}^{in,l} - D_g^l \frac{d}{dz} \phi_{gz}^l(z) |_{z=\Delta z/2} \quad (3.119)$$

$$J_{gz-}^{out,l} = J_{gz-}^{in,l} + D_g^l \frac{d}{dz} \phi_{gz}^l(z) |_{z=-\Delta z/2} \quad (3.120)$$

As was the case in Cartesian geometry, the incoming and outgoing partial currents are related to face averaged fluxes and net currents by the following:

$$J_{g(r,\theta,z)+}^{out,l} - J_{g(r,\theta,z)+}^{in,l} = J_{g(r,\theta,z)+}^l \quad (3.121)$$

$$J_{g(r,\theta,z)-}^{out,l} - J_{g(r,\theta,z)-}^{in,l} = -J_{g(r,\theta,z)+}^l \quad (3.122)$$

$$2 \left( J_{g(r,\theta,z)\pm}^{out,l} + J_{g(r,\theta,z)\pm}^{in,l} \right) = \phi_{g(r,\theta,z)\pm}^l \quad (3.123)$$

In order to produce the final partial current equations, one first substitutes the spatial derivatives of the polynomial flux expansions given in Equations 3.82, 3.83, and 3.84 into the Fick's law expressions of Equations 3.115 through 3.120. Next the remaining expansion coefficients are eliminated by using the relationships given by Equations 3.97, 3.98, 3.105, and 3.106. This series of substitutions finally results in the following partial current equations:

$$J_{gr+}^{out,l} = J_{gr+}^{in,l} - \frac{D_g^l}{\Delta r} \left( 9\phi_{gr+}^l - 3\phi_{gr-}^l - 6\bar{\phi}_g^l - 60\bar{\phi}_{gr1}^l \right) \quad (3.124)$$

$$J_{gr-}^{out,l} = J_{gr-}^{in,l} + \frac{D_g^l}{\Delta r} \left( 3\phi_{gr+}^l - 9\phi_{gr-}^l + 6\bar{\phi}_g^l - 60\bar{\phi}_{gr1}^l \right) \quad (3.125)$$

$$J_{g\theta+}^{out,l} = J_{g\theta+}^{in,l} - \frac{D_g^l}{R\Delta\theta} \left( 16\phi_{g\theta+}^l + 4\phi_{g\theta-}^l - 20\bar{\phi}_g^l - 60\bar{\phi}_{g\theta1}^l - 140\bar{\phi}_{g\theta2}^l \right) \quad (3.126)$$

$$J_{g\theta-}^{out,l} = J_{g\theta-}^{in,l} + \frac{D_g^l}{R\Delta\theta} \left( -4\phi_{g\theta+}^l - 16\phi_{g\theta-}^l + 20\bar{\phi}_g^l - 60\bar{\phi}_{g\theta1}^l + 140\bar{\phi}_{g\theta2}^l \right) \quad (3.127)$$

$$J_{gz+}^{out,l} = J_{gz+}^{in,l} - \frac{D_g^l}{\Delta z} \left( 16\phi_{gz+}^l + 4\phi_{gz-}^l - 20\bar{\phi}_g^l - 60\bar{\phi}_{gz1}^l - 140\bar{\phi}_{gz2}^l \right) \quad (3.128)$$

$$J_{gz-}^{out,l} = J_{gz-}^{in,l} + \frac{D_g^l}{\Delta z} \left( -4\phi_{gz+}^l - 16\phi_{gz-}^l + 20\bar{\phi}_g^l - 60\bar{\phi}_{gz1}^l + 140\bar{\phi}_{gz2}^l \right) \quad (3.129)$$

At a glance one can see that the  $r$ -direction partial current equations are different from those in the  $\theta$  and  $z$  directions. This discrepancy is a direct result of only using a third

order rather than a fourth order flux expansion in the  $r$  direction. If the equations for the node average flux (Eqn 3.78) and the flux moments (Eqn's 3.107, 3.108, 3.109, 3.110, and 3.114) are substituted into Equations 3.124 through 3.129, the final form of the 6x6 response matrix equations is obtained. These response matrix equations may be written as:

$$[\mathbf{A}] \cdot J_g^{out,l} = [\mathbf{C}] \cdot J_g^{in,l} + [\mathbf{B}_1] \cdot Q_g^l + [\mathbf{B}_2] \cdot L_g^l \quad (3.130)$$

where

$$\mathbf{A} = \begin{pmatrix} a_1 + \Delta & a_2 - \Delta & a_3 & a_3 & a_4 & a_4 \\ a_2 + \Delta & a_1 - \Delta & a_3 & a_3 & a_4 & a_4 \\ a_{7R} & a_{7L} & a_5 & a_6 & a_8 & a_8 \\ a_{7R} & a_{7L} & a_6 & a_5 & a_8 & a_8 \\ a_{11R} & a_{11L} & a_{12} & a_{12} & a_9 & a_{10} \\ a_{11R} & a_{11L} & a_{12} & a_{12} & a_{10} & a_9 \end{pmatrix}$$

$$\mathbf{J}_g^{out,l} = (J_{gr+}^{out,l}, J_{gr-}^{out,l}, J_{g\theta+}^{out,l}, J_{g\theta-}^{out,l}, J_{gz+}^{out,l}, J_{gz-}^{out,l})^T$$

$$\mathbf{C} = \begin{pmatrix} c_1 + \Delta & c_2 - \Delta & c_3 & c_3 & c_4 & c_4 \\ c_2 + \Delta & c_1 - \Delta & c_3 & c_3 & c_4 & c_4 \\ c_{7R} & c_{7L} & c_5 & c_6 & c_8 & c_8 \\ c_{7R} & c_{7L} & c_6 & c_5 & c_8 & c_8 \\ c_{11R} & c_{11L} & c_{12} & c_{12} & c_9 & c_{10} \\ c_{11R} & c_{11L} & c_{12} & c_{12} & c_{10} & c_9 \end{pmatrix}$$

$$\mathbf{J}_g^{in,l} = (J_{gr+}^{in,l}, J_{gr-}^{in,l}, J_{g\theta+}^{in,l}, J_{g\theta-}^{in,l}, J_{gz+}^{in,l}, J_{gz-}^{in,l})^T$$

$$\mathbf{B}_1 = \begin{pmatrix} b_1 & b_2 & 0 & 0 & 0 & 0 & 0 \\ b_1 & -b_2 & 0 & 0 & 0 & 0 & 0 \\ b_8 & 0 & 0 & b_9 & b_{10} & 0 & 0 \\ b_8 & 0 & 0 & -b_9 & b_{10} & 0 & 0 \\ b_{13} & 0 & 0 & 0 & 0 & b_{14} & b_{15} \\ b_{13} & 0 & 0 & 0 & 0 & -b_{14} & b_{15} \end{pmatrix}$$

$$\mathbf{Q}_g^l = (\bar{Q}_g^l, \bar{Q}_{gr1}^l, \bar{Q}_{gr2}^l, \bar{Q}_{g\theta1}^l, \bar{Q}_{g\theta2}^l, \bar{Q}_{gz1}^l, \bar{Q}_{gz2}^l)^T$$

$$\mathbf{B}_2 = \begin{pmatrix} -b_4 & 0 & -b_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ b_4 & 0 & b_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -b_4 & b_5 & -b_{11} & b_{12} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & b_4 & b_5 & b_{11} & b_{12} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -b_6 & b_7 & -b_{11} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & b_6 & b_7 & b_{11} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & b_6 & b_7 & b_{12} \end{pmatrix}$$

$$\mathbf{L}_g^l = (L_{g\theta r1}^l, L_{g\theta r2}^l, L_{gzr1}^l, L_{gzr2}^l, L_{gr\theta 1}^l, L_{gr\theta 2}^l, L_{gz\theta 1}^l, L_{gz\theta 2}^l, L_{grz1}^l, L_{grz2}^l, L_{g\theta z1}^l, L_{g\theta z2}^l)^T$$

The coefficient a's, b's, and c's are expressed in terms of known quantities in Appendix A.

### 3.3.1.5 Calculation of the Transverse Leakage Moments

At this point one must calculate the first and second order transverse leakage moments. In cylindrical geometry, as in Cartesian geometry, the quadratic leakage approximation implies that the leakage quantities in Equations 3.79, 3.80, and 3.81 can be approximated by quadratic polynomials. As before, the coefficients of these polynomials are calculated by preserving the average leakage (as determined from previous iterate partial currents transverse to the direction of interest) in three nodes which are adjacent in the direction of interest. The actual relationships between the previous iterate partial currents and the twelve cylindrical geometry transverse leakage moments can be derived by exactly the same methodology as that described in sub-section 3.2.1.5.

In cylindrical geometry, the transverse leakage moments end up possessing exactly the same form as those in Cartesian geometry with the respective variables  $\Delta r^l$ ,  $R\Delta\theta^l$ , and  $\Delta z^l$  substituted into Equations 3.37 and 3.38 in place of  $\Delta x^l$ .

### 3.3.2 Implementation of the Cylindrical Computational Algorithm

In cylindrical geometry the Nodal Expansion Method proceeds through exactly the same inner/outer iteration numerical procedure as was used in Cartesian geometry. In addition, the outer iteration acceleration techniques of coarse-mesh rebalancing and asymptotic extrapolation function exactly as before. The only difference between the three-dimensional cylindrical and the three-dimensional Cartesian solution methodology lies in the means by which the 6x6 nodal response matrices are solved. The details of

solving the cylindrical geometry response matrices are outlined in the following discussion.

As in Cartesian geometry, the general cylindrical geometry response matrix equation (Eqn 3.130) is first written in simplified form:

$$[\mathbf{A}] \cdot \mathbf{J} = \mathbf{b} \quad (3.131)$$

where

$$\mathbf{A} = \begin{pmatrix} a_1 + \Delta & a_2 - \Delta & a_3 & a_3 & a_4 & a_4 \\ a_2 + \Delta & a_1 - \Delta & a_3 & a_3 & a_4 & a_4 \\ a_{7R} & a_{7L} & a_5 & a_6 & a_8 & a_8 \\ a_{7R} & a_{7L} & a_6 & a_5 & a_8 & a_8 \\ a_{11R} & a_{11L} & a_{12} & a_{12} & a_9 & a_{10} \\ a_{11R} & a_{11L} & a_{12} & a_{12} & a_{10} & a_9 \end{pmatrix}$$

$$\mathbf{J} = (J_1, J_2, J_3, J_4, J_5, J_6)^T$$

$$\mathbf{b} = (b_1, b_2, b_3, b_4, b_5, b_6)^T$$

Here the entire right-hand side of Equation 3.130 has been multiplied through and replaced by the  $\mathbf{b}$  vector in Equation 3.131. One can now perform a series of row operations on Equation 3.131. These row operations, which do not affect the value of the members of the solution vector  $\mathbf{J}$ , involve adding rows one and two, three and four, and five and six to one another. These operations result in:

$$[\mathbf{A}'] \cdot \mathbf{J} = \mathbf{b}' \quad (3.132)$$

where

$$\mathbf{A}' = \begin{pmatrix} a_1 + a_2 + 2\Delta & a_1 + a_2 - 2\Delta & 2a_3 & 2a_3 & 2a_4 & 2a_4 \\ a_2 - a_1 & a_1 - a_2 & 0 & 0 & 0 & 0 \\ 2a_{7R} & 2a_{7L} & a_5 + a_6 & a_5 + a_6 & 2a_8 & 2a_8 \\ 0 & 0 & a_6 - a_5 & a_5 - a_6 & 0 & 0 \\ 2a_{11R} & 2a_{11L} & 2a_{12} & 2a_{12} & a_9 + a_{10} & a_9 + a_{10} \\ 0 & 0 & 0 & 0 & a_{10} - a_9 & a_9 - a_{10} \end{pmatrix}$$

$$\mathbf{b}' = (b_1 + b_2, b_2 - b_1, b_3 + b_4, b_4 - b_3, b_5 + b_6, b_6 - b_5)^T$$

If Equation 3.132 is split into its six constitutive coupled linear equations, these equations can be written in terms of a new solution vector as:

$$[\mathbf{A}''] \cdot \mathbf{J}' = \mathbf{b}' \quad (3.133)$$

where

$$\mathbf{A}'' = \begin{pmatrix} a_1 + a_2 + 2\Delta & a_1 + a_2 - 2\Delta & 2a_3 & 0 & 2a_4 & 0 \\ a_2 - a_1 & a_1 - a_2 & 0 & 0 & 0 & 0 \\ 2a_{7R} & 2a_{7L} & a_5 + a_6 & 0 & 2a_8 & 0 \\ 0 & 0 & 0 & a_6 - a_5 & 0 & 0 \\ 2a_{11R} & 2a_{11L} & 2a_{12} & 0 & a_9 + a_{10} & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{10} - a_9 \end{pmatrix}$$

$$\mathbf{J}' = (J_1, J_2, J_3 + J_4, J_3 - J_4, J_5 + J_6, J_5 - J_6)^T$$

Upon observation, one notices the linear independence of rows 3 and 4, and 5 and 6.

Therefore the  $6 \times 6$  matrix equation given by Equation 3.133 may be rewritten as:

$$[\mathbf{A}44] \cdot \mathbf{J}4 = \mathbf{b}4 \quad (3.134)$$

where

$$\mathbf{A}44 = \begin{pmatrix} a_1 + a_2 + 2\Delta & a_1 + a_2 - 2\Delta & 2a_3 & 2a_4 & 0 & 0 \\ a_2 - a_1 & a_1 - a_2 & 0 & 0 & 0 & 0 \\ 2a_{7R} & 2a_{7L} & a_5 + a_6 & 2a_8 & 0 & 0 \\ 2a_{11R} & 2a_{11L} & 2a_{12} & a_9 + a_{10} & 0 & 0 \end{pmatrix}$$

$$\mathbf{J}4 = (J_1, J_2, J_3 + J_4, J_5 + J_6)^T$$

$$\mathbf{b}4 = (b_1 + b_2, b_2 - b_1, b_3 + b_4, b_5 + b_6)^T$$

and

$$[\mathbf{A}22] \cdot \mathbf{J}2 = \mathbf{b}2 \quad (3.135)$$

where

$$\mathbf{A}22 = \begin{pmatrix} a_6 - a_5 & 0 \\ 0 & a_{10} - a_9 \end{pmatrix}$$

$$\mathbf{J}2 = (J_3 - J_4, J_5 - J_6)^T$$

$$\mathbf{b2} = (b_4 - b_3, b_6 - b_5)^T$$

The 4x4 and 2x2 matrix equations are the ones actually used to calculate the six face averaged partial currents for each node at each inner iteration and energy group. At the beginning of each outer iteration, the **A44** matrix, which depends on node dimensions, cross sections, and the iteration n eigenvalue estimate, is first calculated for all nodes and energy groups. These **A44** matrices are then numerically decomposed into upper ( **U44** ) and lower ( **L44** ) triangular matrices through the use of a Gaussian elimination routine (31). During the red-black inner iteration mesh sweeps, the **J4** vectors are updated by simple back substitution. The final values of  $J_3$  through  $J_6$  are obtained through the use of the two independent relationships of Equation 3.135. These relationships yield the following equations for  $J_3$  through  $J_6$ :

$$J_3 = \frac{(J_3 + J_4)}{2} + \frac{b_3 - b_4}{2(a_5 - a_6)} \quad (3.136)$$

$$J_4 = \frac{(J_3 + J_4)}{2} + \frac{b_4 - b_3}{2(a_5 - a_6)} \quad (3.137)$$

$$J_5 = \frac{(J_5 + J_6)}{2} + \frac{b_5 - b_6}{2(a_9 - a_{10})} \quad (3.138)$$

$$J_6 = \frac{(J_5 + J_6)}{2} + \frac{b_6 - b_5}{2(a_9 - a_{10})} \quad (3.139)$$

### 3.4 Conclusions

This chapter has provided detailed derivations of the basic fourth-order Nodal Expansion Method in both Cartesian and cylindrical geometries. A description of the iterative means by which the resulting equations are solved is also provided. In addition, two numerical techniques which greatly speed the convergence of the outer or source iterations, coarse-mesh rebalancing and asymptotic extrapolation, are outlined. The results of using these steady-state numerical algorithms on a wide variety of two- and three-dimensional calculational benchmarks will be provided in the next chapter.

## CHAPTER 4

### BENCHMARKING OF THE STEADY-STATE NODAL EXPANSION METHOD

#### 4.1 Introduction

This chapter will provide the computational results of a series of Nodal Expansion Method calculations. The results of several two- and three-dimensional steady-state benchmark problems will be compared with reference solutions.

#### 4.2 Accuracy Requirements

Before the results of the steady-state Nodal Expansion Method calculations can be presented, one must first gain an understanding of the accuracy requirements of a typical coupled neutronic/ thermal-hydraulic reactor simulation. If one assumes that the results of such a simulation are to be compared with experimental data, it would be a misuse of computational resources to enforce accuracy requirements that are much stiffer than the typical uncertainty in experimental readings. A simple but informative example will serve to quantify the accuracy requirements of the neutronic portion of a simulation.

Assume for the moment that one is modelling the heavily instrumented Loss-of-Flow Test (LOFT) facility at steady-state. From the facility instrumentation analysis document (32) one knows that temperature of the sub-cooled coolant exiting one of the heated channels (typical of full-scale PWR) can be measured to within  $\pm 2^{\circ}\text{F}$  at  $600^{\circ}\text{F}$  (589 K). At steady-state the average coolant temperature at the exit of an assembly can, to a first approximation, be assumed to be only a function of the assembly inlet coolant flow rate and temperature and the total amount of heat energy generated within that assembly. This approximation holds true if one assumes that cross flow between

assemblies is negligible and that all energy produced by an assembly is deposited within that same assembly. Based upon these assumptions, the average enthalpy of the coolant exiting a typical assembly can be expressed as:

$$h_{out}(T_{out}) = h_{in}(T_{in}) + \frac{q}{m} \quad (4.1)$$

where

$h_{out}$   $\equiv$  outlet coolant enthalpy (subcooled)

$h_{in}$   $\equiv$  inlet coolant enthalpy (subcooled)

$q$   $\equiv$  total heat produced in an assembly

$m$   $\equiv$  coolant inlet mass flow rate

Assuming a typical PWR (or the LOFT reactor) at steady-state operating conditions, one can use Equation 4.1 and the sub-cooled water temperature/ enthalpy correlations (steam tables) to calculate a representative value of the coolant outlet temperature.

Assume:

$$T_{in} = 564.82K (557^{\circ}F)$$

$$P_{in} = 15.5mPa (2250PSIA)$$

$$m = 118.5 \frac{Kg}{sec} \left( 9.407 \times 10^5 \frac{lb}{hr} \right)$$

$$q = 15.22 \times 10^3 Kw \left( 5.196 \times 10^7 \frac{BTU}{hr} \right)$$

From the Steam Tables:

$$h_{in} = 1294.8 \frac{KJ}{Kg}$$

From Equation 4.1:

$$h_{out} = 1423.2 \frac{KJ}{Kg}$$

From the Steam Tables:

$$T_{out} = 586.4K (595.9^{\circ}F)$$

One now assumes an error in measured outlet temperature of + 2 °F. Thus the assembly coolant outlet temperature could in fact be as high as 597.9 °F (587.5 K). Repeating the process outlined above, one finds that to obtain this new temperature, a total assembly power of  $16.07 \times 10^3$  KW ( $5.483 \times 10^7 \frac{BTU}{hr}$ ) is required. This value is 5.6% greater than the originally assumed total assembly power.

This simple example has shown that the power produced in a single LOFT assembly may be in error by more than 5% before its effect on the steady-state assembly outlet coolant temperature can be definitively measured. Thus when simulating the LOFT reactor, one only has to predict assembly powers to within an accuracy of approximately 5%. For the neutronic benchmark calculations performed in this and later chapters, one can conservatively assume that errors in predicted assembly power of less than approximately 3% will be satisfactory in light of their use in future coupled neutronics/thermal-hydraulics simulations.

#### 4.3 Cartesian Geometry

##### 4.3.1 Two-Dimensional Benchmarking

In the previous chapter, the fourth-order quadratic transverse leakage expansion version of the NEM was derived in three-dimensional Cartesian geometry. In this subsection of this chapter NEM results will be presented for two two-dimensional Cartesian geometry sample problems. The derivation of the fourth-order, quadratic leakage NEM

in 2-D Cartesian geometry follows the same lines as the 3-D derivation except for the lack of z-dimension flux and transverse leakage moments. Since the NEM in two dimensions will only have four outgoing partial currents per node per energy group, the NEM response matrices will only consist of four equations in four unknowns. The form of these response matrices will consist of exactly the 16 coefficients in the upper left-hand quadrant of the 3-D response matrix given by Equation 3.44. One can again perform row operations, similar to those performed in subsection 3.2.2.2, to reduce the 4x4 response matrix equation to two 2x2 matrix equations similar to Equations 3.49 and 3.50. Since a general 2x2 matrix equation has a simple analytical solution, the outgoing partial currents for each node are simply determined analytically at each inner iteration.

For the sake of comparison, this subsection also presents NEM results for second and third order polynomial flux expansion approximations. In these lower order approximations, the polynomial expansion of the one-dimensional transverse integrated diffusion equations (Eqn's 3.7) are limited to only two or three terms, respectively. The derivation of the response matrix follows a course very similar to that presented in previous subsections for the fourth order flux expansion. For example, in two dimensions, all three flux expansion approximations yield 4x4 response matrix equations of similar structure. In addition, the third-order expansion yields the same first-order flux moment equations as does the fourth-order flux expansion. But, the third order expansion has no second-order flux moments. Similarly, the second-order expansion has no flux moments or flux moment equations at all. As a result, the second-order NEM should have the same order of spatial accuracy as does a traditional mesh-centered finite difference approximation to the neutron diffusion equation.

In addition to varying the order of the polynomial flux expansion approximation, the type of transverse leakage approximation was also varied in the course of executing

the two-dimensional steady-state benchmark problems. Here, in addition to the previously derived quadratic leakage approximation, a so-called flat transverse approximation is tested in the upcoming subsections. In the flat transverse leakage approximation, for example, the  $x$  variation in the  $y$ -directed transverse leakage,  $L_{gy}^l(x)$ , is assumed to be adequately represented by the constant average  $y$ -directed leakage in that node, i.e.  $L_{gy}^l(x) \cong \bar{L}_{gy}^l$ . Using this approximation, all first and/or second order transverse leakage moments become identically zero, regardless of the order of the polynomial flux expansion approximation. In contrast, when the order of the one-dimensional polynomial flux expansion approximation is varied, the complexity of the quadratic leakage expansion changes accordingly. For example, since the second order flux expansion approximation has no flux or transverse leakage moments, no transverse leakage approximation is necessary. If a third-order flux expansion is used, the quadratic leakage approximation is only needed to express first-order transverse leakage moments.

In the following two subsections, a variety of NEM results are presented for two well-documented two-dimensional steady-state computational benchmarks, the IAEA 2-D Benchmark Problem (33) and the LRA 2-D Steady-State Benchmark Problem (25).

#### 4.3.1.1 Two-dimensional IAEA Benchmark Problem

The IAEA PWR Test Problem is a highly simplified two or three-dimensional two energy group static test problem. The detailed macroscopic cross-section and geometric specifications for this problem appear in Section B.1 of Appendix B. Basically the IAEA benchmark problem models a one-eighth core symmetric two-zone reactor made up of 177 fuel assemblies, each having planar dimensions of 20 x 20 cm and an active height of 340 cm. The core is reflected both radially and axially by 20 cm of water. The nine control rods in this problem are modelled as smeared absorbers within their respective fuel assemblies. The existence of inserted control rods and a water reflector introduce

severe local flux perturbations which make this problem a good test of the speed and accuracy of the Nodal Expansion Method.

To serve as a point of comparison, a base-case NEM solution was prepared. This two-dimensional, quarter-core symmetric solution used an x-y mesh spacing of 20 x 20 cm, or one mesh per assembly, a fourth-order polynomial flux expansion, and the quadratic approximation to the transverse leakage terms.

This base-case solution is first compared to a solution obtained using a planar mesh spacing of 10 x 10 cm or four mesh per assembly. The reference normalized assembly power distribution (25), the results of the base (20 x 20 cm) and fine-mesh (10 x 10 cm) solutions, and the percentage error distribution in these solutions with respect to the reference solution appear in Table 4.1. The values displayed in the percentage error distributions are determined by simply calculating the percentage difference between the NEM and the reference normalized assembly power for any given assembly. Other details of these and other solutions to the two-dimensional IAEA sample problem appear in Table 4.2.

As can be seen in Table 4.1, both the base-case and the fine-mesh NEM solutions yield normalized assembly powers very near those of the reference solution. The largest error in assembly power in the base-case solution is an error of approximately 2% in one of the low power assemblies adjacent to the water reflector. This error is well within the realm of acceptability for use in a multidimensional kinetics calculation, especially when one considers that a finite difference solution of similar accuracy would require a mesh spacing of 1.25 cm or less (13). It is also readily apparent from Tables 4.1 and 4.2 that halving the mesh spacing results in a more accurate solution (maximum error of 0.25%). But this increase in accuracy comes with a penalty of more than a factor of three in computation time. Since the errors introduced in cross-section preparation

**Table 4.1 Reference, Base-Case, and Fine-Mesh Assembly Power Distributions and Percentage Errors for the Two-Dimensional IAEA Benchmark Problem**

0.7549	0.7358	0.6921			
0.7450(-1.31)	0.7250(-1.47)	0.6800(-1.75)			
0.7541(-0.11)	0.7350(-0.11)	0.6916(-0.07)			
0.9343	0.9504	0.9752	0.8461	0.5972	
0.9283(-0.64)	0.9429(-0.79)	0.9672(-0.82)	0.8365(-1.13)	0.5912(-1.00)	
0.9341(-0.02)	0.9499(-0.05)	0.9748(-0.04)	0.8457(-0.05)	0.5971(-0.02)	
0.9351	1.0361	1.0705	0.9064	0.6856	0.5849
0.9379(0.30)	1.0351(-0.10)	1.0689(-0.15)	0.9027(-0.41)	0.6843(-0.19)	0.5729(-2.05)
0.9360(0.10)	1.0358(-0.03)	1.0702(-0.03)	0.9064(0.00)	0.6861(0.07)	0.5848(-0.02)
0.6100	1.0697	1.1792	0.9670	0.4706	
0.6097(-0.05)	1.0769(0.67)	1.1843(0.43)	0.9723(0.55)	0.4662(-0.93)	
0.6088(-0.20)	1.0708(0.10)	1.1793(0.01)	0.9673(0.03)	0.4700(-0.13)	
1.2107	1.3149	1.3451	1.1929		
1.2207(0.83)	1.3230(0.62)	1.3544(0.69)	1.1991(0.52)		
1.2122(0.12)	1.3149(0.00)	1.3451(0.00)	1.1929(0.00)		
1.4537	1.4799	1.4694			
1.4634(0.67)	1.4912(0.76)	1.4796(0.69)			
1.4546(0.06)	1.4801(0.01)	1.4693(-0.01)			
1.3097	1.4351				
1.3221(0.95)	1.4453(0.71)				
1.3125(0.21)	1.4351(0.00)				
0.7456					
0.7510(0.72)					
0.7438(-0.24)					

**Reference**

**Base-Case (% Error)**

**10x10 cm (% Error)**

**Table 4.2 Summary of Results for the 2-D IAEA Benchmark Problem**

Case	Keff	Cray XMP CPU Time (sec)	Number of outer iterations
Reference	1.0296	-	-
Base Case	1.0295	0.758	32
Fine Mesh (10x10 cm)	1.0296	2.427	35
Third Order Expansion	1.0290	0.764	33
Second Order Expansion	1.0288	0.551	25
Flat Transverse Leakage	1.0299	0.739	33
Fine-Mesh Rebalancing	1.0295	0.612	22
<b>Two Inner Iterations</b>			
per outer	1.0295	0.626	36
No CM Rebalancing	1.0295	6.918	375
No Asymtotic Extrap.	1.0295	1.393	62

and processing are on the order of approximately 5 to 10%, (34) this extreme level of accuracy in the spatial solution is unnecessary.

Next, in Table 4.3, the reference fourth-order expansion solution is compared with one mesh per assembly solutions obtained with third and second order polynomial flux expansions. As is readily observable from the table, the lower order expansions have very large errors in their assembly power distributions, up to 14% for the third order expansion, and up to 22% for the second order expansion. These errors are in fact much too large to be acceptable, even in a multidimensional kinetics calculation. But in spite of the much lower level of accuracy, one can see from Table 4.2 that these cases use as much or slightly more CPU time as does the much more accurate fourth-order solution. This observation highlights the fact that calculating higher order flux moments requires very little additional CPU time in a NEM calculation. Thus, it has been demonstrated that expanding the one-dimensional fluxes in higher-order polynomials is a much more efficient means of increasing the accuracy of a diffusion theory solution than is a fine-mesh finite difference solution of equal accuracy.

Table 4.4 shows a comparison between the base calculation and a similar calculation in which a flat rather than quadratic approximation is used for the transverse leakage terms. As is readily apparent, the flat leakage approximation gives approximately the same assembly power distribution as does the quadratic approximation, especially for nodes not on the problem boundaries. This implies that the flux within the interior nodes is readily separable into independent x and y components. The accuracy of the flat leakage approximation does suffer in nodes with large non-separable flux gradients, such as in the control rod assemblies and assemblies near the problem boundaries. Table 4.2 shows that the quadratic leakage approximation imposes only a small execution time penalty on the problem solution. Therefore the quadratic leakage approximation

**Table 4.3 Reference, Base-Case, and Lower Order Polynomial Expansion Assembly Power Distributions and Percentage Errors for the Two-Dimensional IAEA Benchmark Problem**

0.7549	0.7358	0.6921			
0.7450(-1.31)	0.7250(-1.47)	0.6800(-1.75)			
0.6826(-9.58)	0.6689(-9.09)	0.5985(-13.52)			
0.6583(-12.80)	0.6264(-14.87)	0.5417(-21.73)			
0.9343	0.9504	0.9752	0.8461	0.5972	
0.9283(-0.64)	0.9429(-0.79)	0.9672(-0.82)	0.8365(-1.13)	0.5912(-1.00)	
0.9120(-2.39)	0.9247(-2.70)	0.9386(-3.75)	0.7785(-7.99)	0.5197(-12.98)	
0.8935(-4.37)	0.8920(-6.14)	0.8970(-8.02)	0.7132(-15.71)	0.4683(-21.58)	
0.9351	1.0361	1.0705	0.9064	0.6856	0.5849
0.9379(0.30)	1.0351(-0.10)	1.0689(-0.15)	0.9027(-0.41)	0.6843(-0.19)	0.5729(-2.05)
0.9481(1.39)	1.0369(0.08)	1.0641(-0.60)	0.8991(-0.81)	0.6781(-1.09)	0.5107(-12.69)
0.9475(1.33)	1.0357(-0.04)	1.0539(-1.55)	0.8716(-3.84)	0.6360(-7.23)	0.4493(-23.18)
0.6100	1.0697	1.1792	0.9670	0.4706	
0.6097(-0.05)	1.0769(0.67)	1.1843(0.43)	0.9723(0.55)	0.4662(-0.93)	
0.6138(0.62)	1.1053(3.33)	1.2036(2.07)	0.9913(2.51)	0.4544(-3.44)	
0.6097(-0.05)	1.1369(6.28)	1.2380(4.99)	1.0079(4.23)	0.4418(-6.12)	
1.2107	1.3149	1.3451	1.1929		
1.2207(0.83)	1.3230(0.62)	1.3544(0.69)	1.1991(0.52)		
1.2768(5.46)	1.3717(4.32)	1.3963(3.81)	1.2306(3.16)		
1.3395(10.64)	1.4232(8.24)	1.4332(6.55)	1.2591(5.55)		
1.4537	1.4799	1.4694			
1.4634(0.67)	1.4912(0.76)	1.4796(0.69)			
1.5255(4.94)	1.5544(5.03)	1.5361(4.54)			
1.6216(11.55)	1.6188(9.39)	1.5794(7.49)			
1.3097	1.4351				
1.3221(0.95)	1.4453(0.71)				
1.3916(6.25)	1.5133(5.45)				
1.4761(12.71)	1.5893(10.74)				
0.7456					
0.7510(0.72)					
0.7799(4.60)					
0.7892(5.85)					
	Reference				
	Base-Case (% Error)				
	Third Order Expansion (% Error)				
	Second Order Expansion (% Error)				

represents an efficient means for increasing the accuracy of a nodal solution to the neutron diffusion equation.

Finally, Table 4.2 summarizes the results of several additional IAEA calculations in which various coding options are employed. Since all of these cases yield the same assembly power distributions as does the base case, no power distributions are tabulated. In the first of these calculations, the number of red-black inner iterations performed during each outer iteration is reduced from 4 to 2. For this specific problem, reducing the number of inner iterations noticeably increases problem computational efficiency. That is, since the within-group fluxes are sufficiently well converged at each outer iteration after only two inner iterations per group, additional iterations only serve to waste CPU time. For other problems 2 inner iterations may not sufficiently converge the within-group partial current solution at any given outer iteration. This may in turn cause the entire solution to diverge.

It was shown in previous trial and error code debugging calculations that performing an odd number of inner iterations per outer iteration (3 or 5) does not improve overall problem convergence beyond using the next lower even number of inner iterations. As a result, the use of four inner iterations was generally settled upon as the smallest fixed number of inner iterations per outer iteration which would in all cases ensure overall problem convergence.

In another calculation, fine-mesh rather than coarse-mesh rebalancing was employed. That is, all nodes in the coarse-mesh rebalancing calculation were set to contain only a single fine node. This modification resulted in a surprising increase in computational efficiency. Using fine-mesh rebalancing greatly reduced the number of outer iterations from 32 to 22, resulting in a problem speed-up which greatly overshadowed

**Table 4.4 Reference, Base-Case, and Flat Leakage Assembly Power Distributions and Percentage Errors for the Two-Dimensional IAEA Benchmark Problem**

Reference	Base-Case	Flat Leakage	(% Error)	(% Error)
0.7549	0.7358	0.6921		
0.7450(-1.31)	0.7250(-1.47)	0.6800(-1.75)		
0.7534(-0.20)	0.7312(-0.63)	0.6799(-1.76)		
0.9343	0.9504	0.9752	0.8461	0.5972
0.9283(-0.64)	0.9429(-0.79)	0.9672(-0.82)	0.8365(-1.13)	0.5912(-1.00)
0.9358(0.16)	0.9432(-0.76)	0.9678(-0.76)	0.8386(-0.89)	0.5953(-0.32)
0.9351	1.0361	1.0705	0.9064	0.6856
0.9379(0.30)	1.0351(-0.10)	1.0689(-0.15)	0.9027(-0.41)	0.6843(-0.19)
0.9396(0.48)	1.0336(-0.24)	1.0676(-0.27)	0.9069(0.06)	0.6949(1.36)
0.9326	1.0336	1.0705	0.9064	0.6856
0.9379(0.30)	1.0351(-0.10)	1.0689(-0.15)	0.9027(-0.41)	0.6843(-0.19)
0.9396(0.48)	1.0336(-0.24)	1.0676(-0.27)	0.9069(0.06)	0.6949(1.36)
0.6100	1.0697	1.1792	0.9670	0.4706
0.6097(-0.05)	1.0769(0.67)	1.1843(0.43)	0.9723(0.55)	0.4662(-0.93)
0.5966(-2.20)	1.0777(0.75)	1.1940(1.26)	0.9746(0.79)	0.4609(-2.06)
1.2107	1.3149	1.3451	1.1929	
1.2207(0.83)	1.3230(0.62)	1.3544(0.69)	1.1991(0.52)	
1.2220(0.93)	1.3173(0.18)	1.3440(-0.08)	1.1943(0.12)	
1.4537	1.4799	1.4694		
1.4634(0.67)	1.4912(0.76)	1.4796(0.69)		
1.4764(1.56)	1.4809(0.07)	1.4586(-0.73)		
1.3097	1.4351			
1.3221(0.95)	1.4453(0.71)			
1.3231(1.02)	1.4391(0.28)			
0.7456				
0.7510(0.72)				
0.7314(-1.90)				

### Reference

### Base-Case (% Error)

### Flat Leakage (% Error)

the small increase in computational effort required at each outer iteration to calculate rebalance factors for each fine node.

The last two calculations were run to show the general accelerating effects of coarse-mesh rebalancing and asymptotic extrapolation. In the first of these calculations, both coarse-mesh rebalancing and asymptotic extrapolation were shut off entirely. This unaccelerated case required almost an order of magnitude more outer iterations and CPU time to achieve convergence than did the base calculation. In the second calculation just the asymptotic extrapolation acceleration was turned off. Here the number of outer iterations and CPU time increased by about a factor of two. As these two runs dramatically demonstrate, coarse-mesh rebalancing and asymptotic extrapolation are very powerful means by which to inexpensively speed convergence of an iterative nodal solution to the multigroup neutron diffusion equation.

#### 4.3.1.2 Two-Dimensional LRA Benchmark Problem

The LRA BWR Test Problem is a highly simplified two or three dimensional two energy group static or kinetic benchmark problem. The detailed macroscopic cross-section and geometric specifications for this problem appear in Section B.2 of Appendix B. The BWR reactor being modelled has a two zone core containing 312 fuel elements each having planar dimensions of 15 x 15 cm and an active height of 300 cm. The core is reflected both radially and axially by 30 cm of water. The inserted control rods in this problem are modelled as smeared absorbers in four adjacent fuel assemblies. Several other control rods are modelled in their fully removed positions, causing the flux distribution to exhibit severe local perturbations, which in turn provide a good test of the Nodal Expansion Method.

As was the case for the IAEA benchmark problem, a base-case NEM solution of the LRA problem was first prepared. Here the LRA problem was solved in two dimensions, using quarter-core symmetry and a mesh spacing of  $15 \times 15$  cm or one mesh per assembly. In addition, this calculation used a fourth-order polynomial flux expansion and the quadratic approximation to the transverse leakage terms.

The results of this base-case solution are first compared to the results of a similar NEM solution obtained with a mesh spacing of  $7.5 \times 7.5$  cm or four mesh per assembly. The reference assembly power distribution (13), the results of the base case ( $15 \times 15$  cm) and fine-mesh ( $7.5 \times 7.5$  cm) solutions, and the percentage errors in each of these solutions with respect to the reference appear in Table 4.5. Other more global details of these and other solutions to the two-dimensional LRA benchmark problem appear in Table 4.6. Many of the observations made with respect to the results of the two-dimensional IAEA benchmark problem apply to the LRA problem.

First, Table 4.5 shows both base-case and fine-mesh solutions being acceptably close to the reference assembly power distribution. The maximum error in the base-case solution, a 2.78% overshoot, occurs at the centermost assembly, which models a relatively highly reactive assembly with fully removed control rods. This maximum error seems to be an artifact of the fourth-order, quadratic leakage NEM solution, since no other assemblies, even those near the water reflector, have errors in power greater than 2%. Regardless, the error in the base-case solution is, as was the case for the 2-D IAEA problem, well within the realm of acceptability for a multidimensional kinetics solution. It can also be seen from Table 4.5 that the fine-mesh solution yields extremely accurate results, a maximum assembly power error of 11 parts in 10,000. This accuracy comes with a penalty of an approximately 400% increase in computation time, a price not worth paying in light of cross-sectional errors inherent in a kinetics type solution.

**Table 4.5 Reference, Base-Case, and Fine-Mesh Assembly Power Distributions and Percentage Errors for the Two-Dimensional LRA Benchmark Problem**

0.9242	0.8672	0.8268	0.8530	0.9325	0.9716	0.8465	
0.9192(-0.54)	0.8640(-0.37)	0.8215(-0.64)	0.8457(-0.86)	0.9232(-1.00)	0.9591(-1.29)	0.8298(-1.97)	
0.9235(-0.08)	0.8665(-0.08)	0.8260(-0.10)	0.8521(-0.11)	0.9315(-0.11)	0.9708(-0.08)	0.8466(0.01)	
1.4810	1.2810	1.1730	1.2210	1.4220	1.6790	1.6210	1.3280
1.4837(0.18)	1.2856(0.36)	1.1709(-0.18)	1.2167(-0.35)	1.4173(-0.33)	1.6673(-0.70)	1.6044(-1.02)	1.3062(-1.64)
1.4810(0.00)	1.2811(0.01)	1.1727(-0.03)	1.2210(0.00)	1.4213(-0.05)	1.6786(-0.02)	1.6202(-0.05)	1.3295(0.11)
1.6610	1.1510	0.9667	1.0220	1.3390	2.0510	2.1610	
1.6831(1.33)	1.1516(0.05)	0.9669(0.02)	1.0199(-0.21)	1.3326(-0.48)	2.0523(0.06)	2.1575(-0.16)	
1.6619(0.05)	1.1505(-0.04)	0.9667(0.00)	1.0221(0.01)	1.3386(-0.03)	2.0513(0.01)	2.1609(0.00)	
1.3860	0.9398	0.7826	0.8432	1.1520	1.8520		
1.4060(1.44)	0.9430(0.34)	0.7846(0.26)	0.8430(-0.02)	1.1484(-0.31)	1.8559(0.21)		
1.3864(0.03)	0.9398(0.00)	0.7828(0.03)	0.8434(0.02)	1.1517(-0.03)	1.8526(0.03)		
0.7902	0.6705	0.6181	0.6782	0.8643			
0.7947(0.57)	0.6780(1.12)	0.6218(0.60)	0.6804(0.32)	0.8668(0.29)			
0.7904(0.03)	0.6707(0.03)	0.6183(0.03)	0.6783(0.01)	0.8643(0.00)			
0.5118	0.4904	0.4921	0.5524				
0.5155(0.72)	0.4955(1.04)	0.4963(0.85)	0.5558(0.62)				
0.5122(0.08)	0.4906(0.04)	0.4922(0.02)	0.5526(0.04)				
0.4130	0.4067	0.4240					
0.4173(1.04)	0.4117(1.23)	0.4286(1.08)					
0.4134(0.10)	0.4070(0.07)	0.4242(0.05)					
0.4402	0.3995						
0.4456(1.23)	0.4066(1.78)						
0.4406(0.09)	0.3996(0.10)						
0.6122							
0.6292(2.78)							
0.6131(0.15)							

**Reference**

**Base-Case (% Error)**

**7.5x7.5 cm (% Error)**

Table 4.6 Summary of Results for the 2-D LRA Benchmark Problem

Case	Keff	Cray XMP CPU Time (sec)	Number of outer iterations
Reference	0.99636	-	-
Base Case	0.99629	1.029	22
Fine Mesh (7.5x7.5 cm)	0.99637	3.602	18
3 <sup>rd</sup> order expansion	0.99578	1.025	22
2 <sup>nd</sup> order expansion	0.99666	0.905	20
Flat Transverse Leakage	0.99679	1.005	22
6x6-Mesh Rebalancing	0.99629	1.415	37
No CM Rebalancing	0.99629	10.081	303
No Asymtotic Extrap.	0.99629	1.763	37

Next, in Table 4.7, the base-case solution is compared to the results of similar one mesh cell per assembly solutions which employ lower order polynomial flux expansions. As can be seen from the table, the lower order flux expansions in general give larger assembly power errors than does the base-case fourth-order expansion. But, surprisingly, these errors are not nearly as large as those obtained in running the 2-D IAEA sample problem. This is most likely the result of the LRA problem using a  $15 \times 15$  cm mesh rather than a  $20 \times 20$  cm mesh. This smaller mesh spacing greatly increases the accuracy of the more finite-difference-like second and third order polynomial flux expansions. Table 4.6 shows that the increased accuracy of the base-case fourth-order expansion comes with only a very minor computational penalty, again showing the efficiency inherent in expanding the one-dimensional fluxes in higher order polynomials.

Finally, in Table 4.8, the base-case solution is compared to a similar calculation in which the flat rather than quadratic approximation is made for the transverse leakage terms. As was the case for the 2-D IAEA problem, the flat leakage approximation yields approximately the same order of error in the normalized assembly power distribution as does the quadratic leakage approximation. Again, this implies that the within-node flux distribution in the LRA problem can for the most part be accurately represented by separable functions of  $x$  and  $y$ . As can be seen in Table 4.8, this assumption breaks down near the material and problem boundaries, where the cross terms in the true two-dimensional polynomial flux expansion are not negligible.

Finally, Table 4.6 gives the global results of three other NEM solutions to the 2-D LRA benchmark problem. The assembly power distributions obtained from all three of these solutions are exactly the same as that of the base case, hence they are not shown. In the first of these calculations, a  $6 \times 6$  coarse-mesh rebalancing mesh is used rather than the  $11 \times 11$  fine-mesh rebalancing of the base case. As was shown for the 2-D

**Table 4.7 Reference, Base-Case, and Lower-Order Polynomial Expansion Assembly Power Distributions and Percentage Errors for the Two-Dimensional LRA Benchmark Problem**

0.9242	0.8672	0.8268	0.8530	0.9325	0.9716	0.8465
0.9192(-0.54)	0.8640(-0.37)	0.8215(-0.64)	0.8457(-0.86)	0.9232(-1.00)	0.9591(-1.29)	0.8298(-1.97)
0.8997(-2.65)	0.8411(-3.01)	0.7955(-3.79)	0.8149(-4.47)	0.8867(-4.91)	0.9253(-4.77)	0.7740(-8.56)
0.9030(-2.29)	0.8563(-1.26)	0.8171(-1.17)	0.8418(-1.31)	0.9195(-1.39)	0.9412(-3.13)	0.7853(-7.23)
1.4810	1.2810	1.1730	1.2210	1.4220	1.6790	1.6210
1.4837(0.18)	1.2856(0.36)	1.1709(-0.18)	1.2167(-0.35)	1.4173(-0.33)	1.6673(-0.70)	1.6044(-1.02)
1.4987(1.20)	1.3020(1.64)	1.1809(0.67)	1.2205(-0.04)	1.4143(-0.54)	1.6481(-1.84)	1.5710(-3.08)
1.4828(0.12)	1.2998(1.47)	1.1846(0.99)	1.2304(0.77)	1.4309(0.63)	1.6566(-1.33)	1.5758(-2.79)
1.6610	1.1510	0.9667	1.0220	1.3390	2.0510	2.1610
1.6831(1.33)	1.1516(0.05)	0.9669(0.02)	1.0199(-0.21)	1.3326(-0.48)	2.0523(0.06)	2.1575(-0.16)
1.7199(3.55)	1.1573(0.55)	0.9751(0.87)	1.0229(0.09)	1.3179(-1.58)	2.0508(-0.01)	2.1395(-0.99)
1.7273(3.99)	1.1458(-0.45)	0.9595(-0.74)	1.0091(-1.26)	1.3155(-1.76)	2.0855(1.68)	2.1727(0.54)
1.3860	0.9398	0.7826	0.8432	1.1520	1.8520	
1.4060(1.44)	0.9430(0.34)	0.7846(0.26)	0.8430(-0.02)	1.1484(-0.31)	1.8559(0.21)	
1.4453(4.28)	0.9556(1.68)	0.7985(2.03)	0.8532(1.19)	1.1447(-0.63)	1.8667(0.79)	
1.4577(5.17)	0.9430(0.34)	0.7783(-0.55)	0.8330(-1.21)	1.1375(-1.26)	1.9009(2.64)	
0.7902	0.6705	0.6181	0.6782	0.8643		
0.7947(0.57)	0.6780(1.12)	0.6218(0.60)	0.6804(0.32)	0.8668(0.29)		
0.8084(2.30)	0.6927(3.31)	0.6332(2.44)	0.6890(1.59)	0.8723(0.93)		
0.7985(1.05)	0.6844(2.07)	0.6241(0.97)	0.6813(0.46)	0.8696(0.61)		
0.5118	0.4904	0.4921	0.5524			
0.5155(0.72)	0.4955(1.04)	0.4963(0.85)	0.5558(0.62)			
0.5322(3.99)	0.5088(3.75)	0.5074(3.11)	0.5654(2.35)			
0.5126(0.16)	0.4977(1.49)	0.5001(1.63)	0.5594(1.27)			
0.4130	0.4067	0.4240				
0.4173(1.04)	0.4117(1.23)	0.4286(1.08)				
0.4328(4.79)	0.4249(4.48)	0.4404(3.87)				
0.4136(0.15)	0.4117(1.23)	0.4313(1.72)				
0.4402	0.3995					
0.4456(1.23)	0.4066(1.78)					
0.4611(4.75)	0.4216(5.53)					
0.4421(0.43)	0.4058(1.58)					
0.6122						
0.6292(2.78)						
0.6618(8.10)						
0.6446(5.29)						
Reference						
Base-Case (% Error)						
Third Order Expansion (% Error)						
Second Order Expansion (% Error)						

**Table 4.8 Reference, Base-Case, and Flat Leakage Assembly Power Distributions and Percentage Errors for the Two-Dimensional LRA Benchmark Problem**

0.9242	0.8672	0.8268	0.8530	0.9325	0.9716	0.8465
0.9192(-0.54)	0.8640(-0.37)	0.8215(-0.64)	0.8457(-0.86)	0.9232(-1.00)	0.9591(-1.29)	0.8298(-1.97)
0.9013(-2.48)	0.8546(-1.45)	0.8193(-0.91)	0.8522(-0.09)	0.9392(0.72)	0.9768(0.54)	0.8429(-0.43)
1.4810	1.2810	1.1730	1.2210	1.4220	1.6790	1.6210
1.4837(0.18)	1.2856(0.36)	1.1709(-0.18)	1.2167(-0.35)	1.4173(-0.33)	1.6673(-0.70)	1.6044(-1.02)
1.4631(-1.21)	1.2704(-0.83)	1.1610(-1.02)	1.2179(-0.25)	1.4366(1.03)	1.6975(1.10)	1.6434(1.38)
1.6610	1.1510	0.9667	1.0220	1.3390	2.0510	2.1610
1.6831(1.33)	1.1516(0.05)	0.9669(0.02)	1.0199(-0.21)	1.3326(-0.48)	2.0523(0.06)	2.1575(-0.16)
1.6739(0.78)	1.1356(-1.34)	0.9504(-1.69)	1.0101(-1.16)	1.3393(0.02)	2.1007(2.42)	2.2191(2.69)
1.3860	0.9398	0.7826	0.8432	1.1520	1.8520	
1.4060(1.44)	0.9430(0.34)	0.7846(0.26)	0.8430(-0.02)	1.1484(-0.31)	1.8559(0.21)	
1.3981(0.87)	0.9297(-1.07)	0.7703(-1.57)	0.8328(-1.23)	1.1502(-0.16)	1.8924(2.18)	
0.7902	0.6705	0.6181	0.6782	0.8643		
0.7947(0.57)	0.6780(1.12)	0.6218(0.60)	0.6804(0.32)	0.8668(0.29)		
0.7809(-1.18)	0.6695(-0.15)	0.6166(-0.24)	0.6786(0.06)	0.8694(0.59)		
0.5118	0.4904	0.4921	0.5524			
0.5155(0.72)	0.4955(1.04)	0.4963(0.85)	0.5558(0.62)			
0.5017(-1.97)	0.4885(-0.39)	0.4944(0.47)	0.5560(0.65)			
0.4130	0.4067	0.4240				
0.4173(1.04)	0.4117(1.23)	0.4286(1.08)				
0.4049(-1.96)	0.4043(-0.59)	0.4253(0.31)				
0.4402	0.3995					
0.4456(1.23)	0.4066(1.78)					
0.4340(-1.41)	0.3971(-0.60)					
0.6122						
0.6292(2.78)						
0.6162(0.65)						

## Reference

### Base-Case (% Error)

### Flat Leakage (% Error)

IAEA problem, fine-mesh rebalancing turns out to be much more efficient and effective than coarse-mesh rebalancing. Here fine-mesh rebalancing reduces the number of outer iterations required for convergence by almost one half, while reducing computation time requirements by almost one third.

The last two calculations demonstrate the vast increases in efficiency that can be obtained with coarse(fine)-mesh rebalancing and asymptotic extrapolation. In the first of these two calculations both coarse(fine)-mesh rebalancing and asymptotic extrapolation were turned off. As was the case for the IAEA problem, this action increased the number of outer iterations and the problem computation time by about an order of magnitude. In the second case only the asymptotic extrapolation routine was turned off. Here, similar to the IAEA problem, both the number of outer iterations and the problem computation time increased by about a factor of two over the base case. These two runs further show the computational efficiency that can be gained from using the coarse(fine)-mesh rebalancing and asymptotic extrapolation methods to accelerate the basic Nodal Expansion Method.

Finally, as was done for the IAEA problem, an attempt was made with the LRA problem to perform only two inner iterations for each group at each outer iteration. For the LRA problem this resulted in the iterative technique becoming unstable, eventually reaching a computational overflow condition.

#### 4.3.2 Three-Dimensional Benchmarking

In the upcoming sub-sections of this chapter, several steady-state eigenvalue calculations are made with a three-dimensional Cartesian geometry version of the NEM coding. This coding solves the three-dimensional, fourth order, quadratic leakage expansion NEM equations developed in previous sections of this chapter. It also has the

capability of solving the three-dimensional NEM equations using only a second or a third order polynomial approximation for the one-dimensional intra-node fluxes.

#### 4.3.2.1 Three-Dimensional IAEA Benchmark Problem

The first in the series of three-dimensional steady-state eigenvalue calculations were assembly normalized power calculations for the 3-D version of the IAEA benchmark problem. This problem, details of which appear in section B.1 of Appendix B, is similar to the 2-D IAEA problem except for the existence of 20 cm axial water reflectors. In addition, this problem also contains an additional partially inserted control rod within the central fuelled region.

As was the case in two dimensions, a base-case solution was first prepared. This three-dimensional, quarter-core symmetric solution used a 20 cm mesh cell width in the x, y, and z directions, a fourth-order polynomial flux expansion and the quadratic approximation to the transverse leakage terms. This solution used four red-black inner iterations per outer iteration, since using only two inner iterations per outer caused the solution to diverge. The one-eighth-core symmetric x-y assembly power distribution obtained from this calculation is compared with the reference Richardson extrapolated solution obtained with the VENTURE (35) finite-difference diffusion theory code in Table 4.9. As can be seen from the table, the fourth-order, quadratic leakage NEM gives a very accurate solution using one mesh per assembly in the x-y plane. The error in this solution is generally well below 1% , except for some low-power assemblies near the radial water reflector where the error climbs to a maximum of 1.62% . This error distribution is very similar to, but slightly smaller than, the error distribution obtained with the base-case solution to the 2-D IAEA benchmark problem.

**Table 4.9 Reference, Base-Case, and Alternate Assembly Power Distributions and Percentage Errors for the Three-Dimensional IAEA Benchmark Problem**

0.7770	0.7570	0.7110			
0.7690(-1.03)	0.7481(-1.18)	0.7008(-1.43)			
0.7784(0.18)	0.7584(0.18)	0.7127(0.24)			
0.7780(0.13)	0.7549(-0.28)	0.7010(-1.41)			
0.7043(-9.36)	0.6899(-8.86)	0.6166(-13.28)			
0.9590	0.9760	1.0000	0.8660	0.6110	
0.9542(-0.50)	0.9687(-0.75)	0.9928(-0.72)	0.8584(-0.88)	0.6067(-0.70)	
0.9601(0.11)	0.9758(-0.02)	1.0006(0.06)	0.8677(0.20)	0.6128(0.29)	
0.9619(0.30)	0.9694(-0.68)	0.9942(-0.58)	0.8609(-0.59)	0.6107(-0.05)	
0.9371(-2.28)	0.9497(-2.69)	0.9632(-3.68)	0.7985(-7.79)	0.5332(-12.73)	
0.9530	1.0550	1.0890	0.9230	0.7000	0.5970
0.9554(0.25)	1.0533(-0.16)	1.0871(-0.17)	0.9194(-0.39)	0.6993(-0.10)	0.5873(-1.62)
0.9536(0.06)	1.0539(-0.10)	1.0884(-0.06)	0.9231(0.01)	0.7012(0.17)	0.5994(0.40)
0.9565(0.37)	1.0517(-0.31)	1.0863(-0.25)	0.9240(0.11)	0.7099(1.41)	0.5863(-1.79)
0.9656(1.32)	1.0551(0.01)	1.0821(-0.63)	0.9156(-0.80)	0.6927(-1.04)	0.5233(-12.35)
0.6100	1.0720	1.1810	0.9720	0.4760	
0.6097(-0.05)	1.0767(0.44)	1.1843(0.28)	0.9759(0.40)	0.4718(-0.88)	
0.6089(-0.18)	1.0704(-0.15)	1.1790(-0.17)	0.9707(-0.13)	0.4758(-0.04)	
0.5956(-2.36)	1.0769(0.46)	1.1948(1.17)	0.9780(0.62)	0.4659(-2.12)	
0.6144(0.72)	1.1060(3.17)	1.2043(1.97)	0.9954(2.41)	0.4598(-3.40)	
1.1930	1.2910	1.3110	1.1780		
1.2002(0.60)	1.2949(0.30)	1.3174(0.49)	1.1819(0.33)		
1.1913(-0.14)	1.2868(-0.33)	1.3077(-0.25)	1.1757(-0.20)		
1.2007(0.65)	1.2887(-0.18)	1.3070(-0.31)	1.1768(-0.10)		
1.2581(5.46)	1.3455(4.22)	1.3615(3.85)	1.2144(3.09)		
1.4220	1.4320	1.3680			
1.4279(0.41)	1.4385(0.45)	1.3736(0.41)			
1.4184(-0.25)	1.4269(-0.36)	1.3642(-0.28)			
1.4416(1.38)	1.4283(-0.26)	1.3517(-1.19)			
1.4923(4.94)	1.5048(5.08)	1.4282(4.40)			
1.2810	1.3970		<b>Reference</b>		
1.2889(0.62)	1.4023(0.38)		<b>Base-Case (% Error)</b>		
1.2790(-0.16)	1.3918(-0.37)		<b>10x10x20(10) cm (% Error)</b>		
1.2887(0.60)	1.3953(-0.12)		<b>Flat Leakage</b>		
1.3608(6.23)	1.4728(5.43)		<b>Third Order Expansion</b>		

Other global specifics for this and other solutions to the 3-D IAEA benchmark problem appear in Table 4.10. As shown in the table, this three-dimensional, fine-mesh rebalanced solution uses up almost 50 times more CPU time than does the equivalent fine-mesh rebalanced two-dimensional solution. This large discrepancy can be explained by the need in 3-D to calculate six rather than four face-averaged partial currents per node per inner iteration for each energy group. In addition, the 3-D calculation has a factor of 19 more mesh cells than does the 2-D calculation, which by itself significantly contributes to this substantial increase in CPU time.

Next, for comparison, a NEM calculation was run with an x-y plane mesh spacing of 10 cm or four mesh per assembly. The axial mesh spacing for this calculation remained at 20 cm, except in the axial reflectors where a mesh spacing of 10 cm was used. The assembly power distribution obtained from this fine-mesh calculation and the percentage errors in these results with respect to the reference solution appear as the third entry in Table 4.9. As shown in the table, the results of this calculation lie very near the reference results, most of which are in error by well less than one-half of one percent. Table 4.10 shows that this increase in accuracy comes with a price of an approximately 250% increase in CPU time, too expensive to outweigh the largely insignificant increase in accuracy over the base one-mesh-per-assembly solution.

Next, Table 4.9 shows the results of a one-mesh-per-assembly calculation performed with a flat rather than quadratic approximation to the transverse leakage terms. As was the case for the 2-D IAEA problem, the errors in this solution are essentially the same as for the base calculation, except for those nodes near the problem boundaries or those near where large cross-section changes occur. As before, this phenomena is the result of the intra-node flux being less non-separable in nodes near large flux gradients.

**Table 4.10 Summary of Results for the 3-D IAEA Benchmark Problem**

Case	Keff	Cray XMP CPU Time (sec)	Number of outer iterations
Reference	1.0290	-	-
Base Case	1.0290	29.049	24
<b>Fine Mesh</b>			
(10x10x20(10) cm)	1.0291	75.253	24
Flat Transverse Leakage	1.0295	28.812	25
Third Order Expansion	1.0284	29.748	24
Coarse-Mesh Rebalancing	1.0290	30.628	48

In the next 3-D IAEA calculation, a third rather than a fourth order polynomial approximation is made for the one-dimensional fluxes. As shown in Table 4.9, this lower order flux expansion approximation gives much worse results than does the base-case solution. In addition, Table 4.10 shows that the use of this simplified intra-node flux approximation results in essentially no savings in CPU time, again confirming the efficiency inherent in using higher order flux expansion approximations.

Finally, as shown in Table 4.10, the base-case calculation was run using coarse-mesh rather than fine-mesh rebalancing. Here, the use of a 6x6x7 coarse mesh results in the need to perform twice the number of outer iterations to achieve problem convergence. But, as seen in Table 4.10, the extra CPU time used by this large increase in the number of outer iterations is almost totally eclipsed by the CPU time needed to calculate the more than sixfold increase in the number of rebalance factors. This demonstrates that for some three-dimensional problems, it may be more computationally efficient to use coarse rather than fine mesh rebalancing.

#### 4.3.2.2 Three-Dimensional LRA Benchmark Problem w/o Control Rod

The second in the series of three-dimensional, Cartesian geometry, steady-state, eigenvalue calculations were two calculations of the quarter-core symmetric assembly power distribution for the three-dimensional LRA benchmark problem with control rod withdrawn. This 3-D benchmark problem, whose specifications appear in section B.2 of Appendix B, is very similar to the 2-D LRA benchmark problem except for the existence of 30 cm thick axial water reflectors and an off-center control rod. The base-case NEM assembly power distribution for this model was a fourth-order, quadratic leakage calculation with 15x15 cm cells in the x-y plane (one mesh per assembly) and 25 cm nodes in the axial direction, except in the reflectors where a mesh spacing of 15 cm was

used. The results of this quarter-core symmetric solution and the reference fine-mesh Analytic Nodal Method solution by Smith (13) appear in Table 4.11.

Upon observation, one immediately notices the severe power tilt brought on by the withdrawal of the off-center control rod. One also notices that the base one-mesh-per-assembly NEM calculation follows this highly asymmetric flux distribution rather well. Percentage errors in the assembly power, in general, do not rise above 1.5%, except for a few edge assemblies where the error rises to almost 2%. As was the case with the 2-D LRA problem, the centermost assembly exhibits an anomalous error in assembly power of about 3%. The global results of this calculation appear in Table 4.12. One can see from this table that the 3-D LRA problem uses almost 62 times the CPU time as its 2-D counterpart.

Next, the 3-D LRA assembly power distribution was calculated using four mesh per assembly in the x-y plane. Axial meshing for this calculation remained at 25 cm in the core with 15 cm mesh in the water reflectors. The normalized assembly power distribution obtained from this fine-mesh calculation appears as the third entry in Table 4.11. As shown in the table, the use of four per assembly significantly lowers the errors in the power distribution. The maximum error with respect to the reference solution is now only 0.25%, with the average error being much less. But, as shown in Table 4.12, this exceptional accuracy comes with a significant 2.5 fold increase in computation time. The CPU time required to converge this fine-mesh, quarter-core symmetric calculation is a prohibitive 158+ seconds.

#### 4.3.2.3 Three-Dimensional LMW Benchmark Problem

The final runs in the series of 3-D, Cartesian geometry, steady-state, eigenvalue calculations were performed on a steady-state version of a 3-D transient sample prob-

**Table 4.11 Reference, Base-Case, and Fine-Mesh Assembly Power Distributions and Percentage Errors for the Three-Dimensional LRA Benchmark Problem with Control Rod Inserted**

0.3589	0.3851	0.4701	0.6425	0.8968	1.1358	1.1805		
0.3584(-0.14)	0.3852(0.03)	0.4689(-0.26)	0.6404(-0.33)	0.8933(-0.39)	1.1283(-0.66)	1.1600(-1.74)		
0.3590(0.03)	0.3852(0.03)	0.4701(0.00)	0.6423(-0.03)	0.8963(-0.06)	1.1349(-0.08)	1.1781(-0.20)		
0.5686	0.5631	0.6702	0.9427	1.4256	2.0807	2.4821	2.5842	
0.5717(0.55)	0.5672(0.73)	0.6717(0.21)	0.9434(0.07)	1.4292(0.25)	2.0770(-0.18)	2.4688(-0.54)	2.5399(-1.71)	
0.5692(0.11)	0.5637(0.11)	0.6706(0.06)	0.9431(0.04)	1.4260(0.03)	2.0801(-0.03)	2.4805(-0.06)	2.5792(-0.19)	
0.6290	0.4984	0.5571	0.8155	1.4146	2.6809	3.6004	4.0389	2.6072
0.6402(1.78)	0.5011(0.54)	0.5596(0.45)	0.8164(0.11)	1.4126(-0.14)	2.6908(0.37)	3.6068(0.18)	4.0277(-0.28)	2.5664(-1.56)
0.6301(0.17)	0.4988(0.08)	0.5574(0.05)	0.8157(0.02)	1.4143(-0.02)	2.6820(0.04)	3.6002(-0.01)	4.0369(-0.05)	2.6022(-0.19)
0.5222	0.4053	0.4524	0.6795	1.2292	2.4359	3.4237	4.1094	2.8021
0.5324(1.95)	0.4085(0.79)	0.4554(0.66)	0.6814(0.28)	1.2292(0.00)	2.4474(0.47)	3.4359(0.36)	4.1099(0.01)	2.7837(-0.66)
0.5232(0.19)	0.4057(0.10)	0.4528(0.09)	0.6799(0.06)	1.2292(0.00)	2.4373(0.06)	3.4245(0.02)	4.1082(-0.03)	2.7995(-0.09)
0.2996	0.2911	0.3515	0.5230	0.8620	1.4071	1.9793	2.4908	1.7871
0.3027(1.03)	0.2953(1.44)	0.3546(0.88)	0.5263(0.63)	0.8678(0.67)	1.4082(0.08)	1.9768(-0.13)	2.4868(-0.16)	1.7742(-0.72)
0.3000(0.13)	0.2915(0.14)	0.3518(0.09)	0.5233(0.06)	0.8624(0.05)	1.4070(-0.01)	1.9785(-0.04)	2.4907(0.00)	1.7854(-0.10)
0.1923	0.2093	0.2645	0.3824	0.5807	0.8522	1.1774	1.5305	1.1188
0.1945(1.14)	0.2121(1.34)	0.2674(1.10)	0.3858(0.89)	0.5844(0.64)	0.8546(0.28)	1.1769(-0.04)	1.5251(-0.35)	1.1101(-0.78)
0.1926(0.16)	0.2096(0.14)	0.2648(0.11)	0.3827(0.08)	0.5810(0.05)	0.8526(0.05)	1.1775(0.01)	1.5308(0.02)	1.1180(-0.07)
0.1478	0.1630	0.2060	0.2908	0.4252	0.6016	0.8101	1.0397	0.7568
0.1499(1.42)	0.1654(1.47)	0.2086(1.26)	0.2939(1.07)	0.4286(0.80)	0.6044(0.47)	0.8116(0.19)	1.0386(-0.11)	0.7521(-0.62)
0.1480(0.14)	0.1633(0.18)	0.2062(0.10)	0.2911(0.10)	0.4256(0.09)	0.6019(0.05)	0.8104(0.04)	1.0401(0.04)	0.7565(-0.04)
0.1458	0.1465	0.1767	0.2490	0.3762	0.5587	0.7219	0.8448	0.5899
0.1481(1.58)	0.1494(1.98)	0.1792(1.41)	0.2520(1.20)	0.3810(1.28)	0.5617(0.54)	0.7236(0.24)	0.8480(0.38)	0.5879(-0.34)
0.1461(0.21)	0.1468(0.20)	0.1769(0.11)	0.2493(0.12)	0.3766(0.11)	0.5591(0.07)	0.7222(0.04)	0.8455(0.08)	0.5898(-0.02)
0.1928	0.1517	0.1675	0.2390	0.4008	0.7307	0.9064	0.8378	0.5363
0.1988(3.11)	0.1540(1.52)	0.1696(1.25)	0.2413(0.96)	0.4040(0.80)	0.7430(1.68)	0.9197(1.47)	0.8394(0.19)	0.5336(-0.50)
0.1933(0.26)	0.1520(0.20)	0.1677(0.12)	0.2393(0.13)	0.4012(0.10)	0.7321(0.19)	0.9078(0.15)	0.8384(0.07)	0.5364(0.02)

### Reference

Base-Case (% Error)

7.5x7.5x25(15) cm (% Error)

**Table 4.12 Summary of Results for the 3-D LRA Benchmark Problem With Control Rod Inserted**

Case	Keff	Cray XMP CPU Time (sec)	Number of outer iterations
Reference	0.9964	-	-
Base Case	1.0152	61.860	27
Fine Mesh (7.5x7.5x25(15) cm)	1.0155	158.237	29

lem developed by Langenbuch et. al. (36). This problem, referred to as the LMW (Langenbuch-Maurer-Werner) sample problem, models a highly simplified light water reactor. This model, steady-state and transient details of which are given in Section B.3 of Appendix B, simulates a two-zone, one-eighth-core symmetric reactor with 77 assemblies. These assemblies have x-y dimensions of 20x20 cm and an active height of 160 cm. The core is reflected in both the radial and axial directions by 20 cm of water. In the steady-state LMW calculation, five control rods, modelled as homogenized assemblies, are parked in the upper axial reflector, while four other control rods are inserted from the upper axial reflector to the midplane of the core. The base-case NEM calculation for this model uses a mesh spacing of 20 cm in x, y, and z directions, a fourth-order polynomial flux expansion, and the quadratic approximation to the transverse leakage terms.

The assembly power distribution obtained from this calculation is shown in Table 4.13. As can be seen, the calculated assembly powers are in excellent agreement with the reference solution, a four mesh per assembly Analytic Nodal Method calculation (13). The maximum error in this calculation, whose assembly power distribution is much more representative of a typical operating LWR, is only 1.22%, while the average error is generally much lower. Table 4.14 shows the global results of this and other steady-state LMW calculations. Remarkably, this base-case calculation consumed less than 4.5 seconds of CPU time, significantly less than all previous 3-D eigenvalue calculations.

In the second LMW calculation, the x-y mesh spacing was halved to 10 cm, while the axial mesh remained at 20 cm, except in the axial reflectors where the mesh spacing was 10 cm. As shown in Table 4.13, this calculation yields extremely accurate results, i.e. very close to those of the reference ANM solution which used the same mesh

**Table 4.13 Reference, Base-Case, and Alternate Assembly Power Distributions and Percentage Errors for the Three-Dimensional LMW Benchmark Problem**

0.7268	0.7083	0.6274	0.4345
0.7226(-.58)	0.7041(-.59)	0.6254(-.32)	0.4292(-1.22)
0.7262(-.08)	0.7076(-.10)	0.6267(-.11)	0.4337(-.18)
0.7226(-.58)	0.7041(-.59)	0.6254(-.32)	0.4292(-1.22)
0.9801	1.0833	0.9801	0.8597
0.9806(0.05)	1.0846(0.12)	0.9821(0.20)	0.8607(0.12)
0.9802(0.01)	1.0835(0.02)	0.9802(0.01)	0.8592(-0.06)
0.9805(0.04)	1.0846(0.12)	0.9821(0.20)	0.8606(0.10)
1.4401	1.3959	1.1228	
1.4427(0.18)	1.3984(0.18)	1.1248(0.18)	
1.4407(0.04)	1.3964(0.04)	1.1230(0.02)	
1.4428(0.19)	1.3985(0.19)	1.1247(0.17)	
1.6544	1.5894		
1.6581(0.22)	1.5928(0.21)		
1.6553(0.05)	1.5902(0.05)		
1.6582(0.23)	1.5929(0.22)		
1.5542			
1.5577(0.23)			
1.5552(0.06)			
1.5576(0.22)			

**Reference**

**Base-Case (% Error)**

**10x10x20(10) cm (% Error)**

**20x20x10 cm (% Error)**

**Table 4.14 Summary of Results for the 3-D LMW Benchmark Problem**

Case	Keff	Cray XMP CPU Time (sec)	Number of outer iterations
Reference	0.99966	-	-
Base Case	0.99958	4.444	19
<b>Fine Mesh</b>			
(10x10x20(10) cm)	0.99963	40.059	24
<b>Fine Mesh</b>			
(20x20x10 cm)	0.99960	13.170	41
<b>Two Inner Iterations</b>			
per Outer	0.99958	8.186	40

spacing. But to obtain this level of accuracy, one must pay a steep price, in this case almost an order of magnitude more CPU time than the base case.

Next, in order to separate the effects of x-y plane mesh spacing and axial mesh spacing on accuracy, a NEM calculation was run in which the x-y plane mesh spacing remained at 20 cm, but the axial mesh spacing was reduced to 10 cm. The results of this calculation appear as the final entry in Table 4.13. In spite of the approximate tripling of CPU time, this calculation yields almost exactly the same results as the base calculation. This observation indicates that the axial flux shape in the LMW problem is rather smooth, a shape that can be very accurately modelled by a coarse-mesh fourth-order NEM approximation.

Finally, the base-case LMW calculation was repeated using only two inner iterations per outer iteration. As shown in Table 4.14, this resulted in a significant increase in both outer iterations and CPU time. These results indicate that using fewer inner iterations per outer iteration, even if it does not cause solution divergence, may in fact increase problem CPU time.

#### 4.4 Cylindrical Geometry

##### 4.4.1 Dodds' Two-Dimensional R-Z Benchmark Problem

In order to test the accuracy of the Nodal Expansion Method in cylindrical geometry, the steady-state nodal power distribution in a R-Z geometry, two energy group sample problem was first calculated. This sample problem, the eighth in the Argonne Benchmark Problem Book (33), will be referred to as the Dodds' Problem, after the researcher who initially described it, H. L. Dodds Jr. of the University of Tennessee. The Dodds' problem is a highly simplified R-Z geometry two-group model of a Plutonium/Tritium production reactor at the Savannah River Laboratory. This model, a

description of which appears in Section B.4 of Appendix B, consists of seven fuel materials and two axial reflector regions in a geometry 240 cm in radius and 525 cm in height. The reference steady-state two-group flux distribution for this model was obtained by Dodds using a finite difference diffusion theory code with a uniform radial mesh spacing of 8 cm and a uniform axial mesh spacing of 18.75 cm. The normalized fast and thermal  $31 \times 29$  point flux distribution reported by Dodds was converted to an equivalent  $6 \times 14$  cell volume averaged power distribution for purposes of comparison with NEM results.

The first steady-state NEM calculation of the power distribution in the Dodds' problem was an R- $\theta$ -Z geometry model which used only a second order polynomial expansion in each of the three coordinate directions. This model used the same R and Z mesh spacing, 8 cm and 18.75 cm respectively, as did the reference finite difference calculation. Three 10 degree mesh were used in the  $\theta$  direction, but reflective boundary conditions were placed on both  $\theta$  boundaries in order to yield an equivalent 2-D R-Z flux distribution. The axially symmetric nodal power distribution resulting from this calculation is compared with the power distribution derived from Dodds' results in Table 4.15. Global results of this and other NEM solutions appear in Table 4.16.

As can be seen from Table 4.15, the second-order NEM solution is in good agreement (errors below 2.5%) with the reference finite difference solution, except near the axial water reflectors where errors of almost 9% appear. These large nodal power discrepancies are most likely the result of using a cell-averaged rather than a point-centered spatial numerical approximation to the neutron diffusion equation. The differences between these two approximations is greatly exaggerated near boundaries where large changes in the group diffusion coefficient occur. The cell-averaged (equivalent to the second-order NEM) method is believed to be the more accurate approximation since

**Table 4.15 Reference, Second Order Fine-Mesh and Third Order (r dir) Coarse-Mesh Nodal Power Distributions and Percentage Errors for Dodds' Two-Dimensional Benchmark Problem**

0.6067	0.6069	0.6101	0.5953	0.4566	0.1714
0.6550(7.96)	0.6559(8.07)	0.6594(8.07)	0.6454(8.41)	0.4967(8.79)	0.1853(8.10)
0.6382(5.19)	0.6392(5.31)	0.6428(5.35)	0.6301(5.84)	0.4863(6.51)	0.1827(6.58)
0.6302(3.87)	0.6318(4.10)	0.6363(4.29)	0.6275(5.41)	0.4898(7.28)	0.1875(9.38)
1.2727	1.2755	1.2849	1.2526	0.9604	0.3614
1.3003(2.17)	1.3009(1.99)	1.3057(1.62)	1.2756(1.83)	0.9802(2.06)	0.3656(1.16)
1.2895(1.32)	1.2905(1.17)	1.2960(0.86)	1.2686(1.28)	0.9781(1.85)	0.3673(1.63)
1.2737(0.08)	1.2759(0.03)	1.2832(-0.13)	1.2646(0.96)	0.9871(2.78)	0.3777(4.50)
1.6740	1.6740	1.6777	1.6242	1.2330	0.4651
1.6761(0.12)	1.6746(0.04)	1.6747(-0.18)	1.6225(-0.10)	1.2340(0.08)	0.4630(-0.45)
1.6716(-0.14)	1.6706(-0.20)	1.6718(-0.35)	1.6231(-0.07)	1.2383(0.43)	0.4681(0.65)
1.6513(-1.36)	1.6517(-1.33)	1.6550(-1.35)	1.6172(-0.43)	1.2485(1.26)	0.4803(3.27)
1.7789	1.7756	1.7681	1.6928	1.2484	0.4883
1.7487(-1.70)	1.7442(-1.77)	1.7365(-1.79)	1.6607(-1.89)	1.2228(-2.05)	0.4783(-2.04)
1.7493(-1.67)	1.7452(-1.71)	1.7387(-1.66)	1.6664(-1.56)	1.2315(-1.35)	0.4847(-0.73)
1.7280(-2.86)	1.7254(-2.83)	1.7209(-2.67)	1.6595(-1.96)	1.2383(-0.81)	0.4971(1.81)
1.7969	1.7931	1.7807	1.6919	1.2329	0.4874
1.7621(-1.94)	1.7553(-2.11)	1.7418(-2.19)	1.6527(-2.32)	1.2048(-2.28)	0.4741(-2.72)
1.7643(-1.81)	1.7580(-1.96)	1.7457(-1.97)	1.6603(-1.87)	1.2145(-1.49)	0.4811(-1.28)
1.7427(-3.02)	1.7379(-3.08)	1.7276(-2.98)	1.6528(-2.31)	1.2203(-1.02)	0.4922(0.99)

### Reference

Second Order Expansion (% Error)

Third Order r, Fourth Order  $\theta z$  (20cm  $\times$  10°  $\times$  18.75cm) (% Error)

Third Order r, Fourth Order  $\theta z$  (40cm  $\times$  10°  $\times$  37.5cm) (% Error)

**Table 4.16 Summary of Results for Dodds' R-Z Benchmark Problem**

Case	Keff	Cray XMP CPU Time (sec)	Number of outer iters
Reference	0.86705	-	-
Second Order	0.86821	120.02	70
3rd r, 4th $\theta$ z (20cm x 10° x 18.75cm)	0.86723	49.649	64
3rd r, 4th $\theta$ z (40cm x 10° x 37.5cm)	0.86735	10.274	52

no type of diffusion coefficient averaging needs to be performed, as is necessary in the point approximation at material boundaries.

The second NEM calculation of the nodal power distribution in Dodds' problem used a third-order polynomial flux expansion approximation in the  $r$  direction and fourth-order flux expansions in the  $\theta$  and  $z$  directions. In addition, this model employed a radial mesh spacing of 20 cm and an axial mesh spacing of 18.75 cm. Again, this problem used three  $\theta$  mesh of  $10^\circ$  each with reflecting boundary conditions on both plus and minus  $\theta$  faces. The nodal power results of this calculation are generally in better agreement with Dodds' results than were the second-order NEM results which used a 2.5 times smaller radial mesh spacing. In addition, this more accurate calculation uses only about 40% as much CPU time as does the second-order fine-mesh calculation. This observation again shows the computational efficiency that can be gained by expanding the one-dimensional intranode fluxes in higher order polynomials.

The third NEM calculation again used a third-order expansion in the  $r$ -direction and a fourth-order expansion in the  $\theta$  and  $z$  directions. But, this time the radial mesh spacing was set at 40 cm while the axial mesh spacing was set at 37.5 cm. The power distribution resulting from this maximum allowable node size calculation are shown as the final entry in Table 4.15. As shown in the table, the results of this calculation, except near the axial reflectors, are in error by no more than about 3%. This level of error may be slightly larger than acceptable for a three-dimensional kinetics calculation. To rectify this situation and allow for more accurate calculation of flux in the radial direction, cylindrical NEM coding which uses a fourth-order polynomial flux expansion in the radial direction will be derived in the following sub-section.

#### 4.4.1.1 Fourth-Order Radial Expansion, First Current Derivative Approximation

From Section 3.3.1.3 it is known that the moments weighting scheme is used to calculate the fourth-order polynomial flux expansion coefficient and the second-order r direction moment equation. Second-order moments weighting in the r direction can be expressed as:

$$\begin{aligned} & \langle 3 \left( \frac{r - R}{\Delta r} \right)^2 - \frac{1}{4}, \frac{1}{r} \frac{d}{dr} (r j_{gr}^l(r)) + A_g^l \phi_{gr}^l(r) - Q_{gr}^l(r) \\ & + \frac{1}{r \Delta \theta} L_{g\theta}^l(r) + \frac{1}{\Delta z} L_{gz}^l(r) \rangle = 0 \end{aligned} \quad (4.2)$$

The fourth-order expansion coefficient in the r direction can be found by evaluating the flux-like term in Equation 4.2, i.e.:

$$\begin{aligned} \bar{\phi}_{gr2}^l & \equiv \langle 3 \left( \frac{r - R}{\Delta r} \right)^2 - \frac{1}{4}, \phi_{gr}^l(r) \rangle \\ & = \frac{1}{\Delta r} \int_{R-\Delta r/2}^{R+\Delta r/2} \left( 3 \left( \frac{r - R}{\Delta r} \right)^2 - \frac{1}{4} \right) \phi_{gr}^l(r) dr \end{aligned} \quad (4.3)$$

Upon substituting in the fourth-order polynomial expansion approximation for the r direction flux (Eqn 3.82) and then performing the indicated integration, one obtains the following expression for the fourth-order flux expansion coefficient:

$$a_{gr4}^l = 35a_{gr2}^l - 700\bar{\phi}_{gr2}^l \quad (4.4)$$

In order to fully determine the second-order r moment equation, one must first evaluate the first or current derivative term of Equation 4.2. The integration required to evaluate this term can be expressed as:

$$\begin{aligned} & \langle 3 \left( \frac{r - R}{\Delta r} \right)^2 - \frac{1}{4}, \frac{1}{r} \frac{d}{dr} (r j_{gr}^l(r)) \rangle \\ & = \frac{1}{\Delta r} \int_{R-\Delta r/2}^{R+\Delta r/2} \left( 3 \left( \frac{r - R}{\Delta r} \right)^2 - \frac{1}{4} \right) \left( \frac{1}{r} \frac{d}{dr} (r j_{gr}^l(r)) \right) dr \end{aligned} \quad (4.5)$$

Without any approximation, this integral can be reduced to the following form:

$$\begin{aligned} & \left\langle 3 \left( \frac{r-R}{\Delta r} \right)^2 - \frac{1}{4}, \frac{1}{r} \frac{d}{dr} (r j_{gr}^l(r)) \right\rangle \\ &= \frac{1}{2\Delta r} (J_{gr+}^l - J_{gr-}^l) + \frac{3D_g^l}{2\Delta r^2} (\phi_{gr+}^l + \phi_{gr-}^l - 2\bar{\phi}_g^l) \\ &+ \frac{3D_g^l R}{\Delta r^3} (\phi_{gr+}^l - \phi_{gr-}^l) + \left( \frac{3R^2}{\Delta r^3} - \frac{1}{4\Delta r} \right) \int_{R-\Delta r/2}^{R+\Delta r/2} \frac{1}{r} j_{gr}^l(r) dr \end{aligned} \quad (4.6)$$

An assumption must now be made in order to evaluate the final unknown integral in Equation 4.6. As was done for the first-order  $r$  direction flux moment equation,  $\frac{1}{r}$  can be simply approximated within node  $l$  as  $\frac{1}{R}$ . This approximation leads to the following form for the second-order  $r$  moment equation:

$$\begin{aligned} \bar{\phi}_{gr2}^l &= -\frac{1}{A_g^l} \left( \frac{1}{2\Delta r} (J_{gr+}^l - J_{gr-}^l) + \frac{D_g^l}{2\Delta r^2} \left( \left( \frac{3}{\Delta r} + \frac{1}{2R} \right) \phi_{gr+}^l \right. \right. \\ &\quad \left. \left. + \left( \frac{3}{\Delta r} - \frac{1}{2R} \right) \phi_{gr-}^l \right) - \frac{3D_g^l}{\Delta r^2} \bar{\phi}_g^l - \bar{Q}_{gr2}^l + \frac{1}{R\Delta\theta} L_{g\theta r2}^l + \frac{1}{\Delta z} L_{gz r2}^l \right) \end{aligned} \quad (4.7)$$

At this point the  $r$  direction response matrix equations must be re-evaluated to reflect the fact that a fourth-order polynomial flux expansion approximation is now being used in the  $r$  direction. Equations 3.124 and 3.125 may now be rewritten as:

$$J_{gr+}^{out,l} = J_{gr+}^{in,l} - \frac{D_g^l}{\Delta r} (16\phi_{gr+}^l + 4\phi_{gr-}^l - 20\bar{\phi}_g^l - 60\bar{\phi}_{gr1}^l - 140\bar{\phi}_{gr2}^l) \quad (4.8)$$

$$J_{gr-}^{out,l} = J_{gr-}^{in,l} + \frac{D_g^l}{\Delta r} (-4\phi_{gr+}^l - 16\phi_{gr-}^l + 20\bar{\phi}_g^l - 60\bar{\phi}_{gr1}^l + 140\bar{\phi}_{gr2}^l) \quad (4.9)$$

The expressions for the node average flux (Eqn 3.78) and the  $r$  direction flux moments (Eqn's 3.114 and 4.7) can now substituted into Equations 4.8 and 4.9. If the resulting equations are combined with previously derived fourth-order  $\theta$  and  $z$  direction equations, 6x6 response matrix equations of the following form result:

$$[\mathbf{A}] \cdot J_g^{out,l} = [\mathbf{C}] \cdot J_g^{in,l} + [\mathbf{B}_1] \cdot Q_g^l + [\mathbf{B}_2] \cdot L_g^l \quad (4.10)$$

where

$$\mathbf{A} = \begin{pmatrix} a_{1R} + \Delta & a_{2L} - \Delta & a_3 & a_3 & a_4 & a_4 \\ a_{2R} + \Delta & a_{1L} - \Delta & a_3 & a_3 & a_4 & a_4 \\ a_{7R} & a_{7L} & a_5 & a_6 & a_8 & a_8 \\ a_{7R} & a_{7L} & a_6 & a_5 & a_8 & a_8 \\ a_{11R} & a_{11L} & a_{12} & a_{12} & a_9 & a_{10} \\ a_{11R} & a_{11L} & a_{12} & a_{12} & a_{10} & a_9 \end{pmatrix}$$

$$\mathbf{J}_g^{out,l} = \left( J_{gr+}^{out,l}, J_{gr-}^{out,l}, J_{g\theta+}^{out,l}, J_{g\theta-}^{out,l}, J_{gz+}^{out,l}, J_{gz-}^{out,l} \right)^T$$

$$\mathbf{C} = \begin{pmatrix} c_{1R} + \Delta & c_{2L} - \Delta & c_3 & c_3 & c_4 & c_4 \\ c_{2R} + \Delta & c_{1L} - \Delta & c_3 & c_3 & c_4 & c_4 \\ c_{7R} & c_{7L} & c_5 & c_6 & c_8 & c_8 \\ c_{7R} & c_{7L} & c_6 & c_5 & c_8 & c_8 \\ c_{11R} & c_{11L} & c_{12} & c_{12} & c_9 & c_{10} \\ c_{11R} & c_{11L} & c_{12} & c_{12} & c_{10} & c_9 \end{pmatrix}$$

$$\mathbf{J}_g^{in,l} = \left( J_{gr+}^{in,l}, J_{gr-}^{in,l}, J_{g\theta+}^{in,l}, J_{g\theta-}^{in,l}, J_{gz+}^{in,l}, J_{gz-}^{in,l} \right)^T$$

$$\mathbf{B}_1 = \begin{pmatrix} b_1 & b_2 & b_3 & 0 & 0 & 0 & 0 & 0 \\ b_1 & -b_2 & b_3 & 0 & 0 & 0 & 0 & 0 \\ b_8 & 0 & 0 & b_9 & b_{10} & 0 & 0 & 0 \\ b_8 & 0 & 0 & -b_9 & b_{10} & 0 & 0 & 0 \\ b_{13} & 0 & 0 & 0 & 0 & b_{14} & b_{15} & \\ b_{13} & 0 & 0 & 0 & 0 & -b_{14} & b_{15} & \end{pmatrix}$$

$$\mathbf{Q}_g^l = \left( \bar{Q}_g^l, \bar{Q}_{gr1}^l, \bar{Q}_{gr2}^l, \bar{Q}_{g\theta1}^l, \bar{Q}_{g\theta2}^l, \bar{Q}_{gz1}^l, \bar{Q}_{gz2}^l \right)^T$$

$$\mathbf{B}_2 = \begin{pmatrix} -b_4 & b_5 & -b_6 & b_7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ b_4 & b_5 & b_6 & b_7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -b_4 & b_5 & -b_{11} & b_{12} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & b_4 & b_5 & b_{11} & b_{12} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -b_6 & b_7 & -b_{11} & b_{12} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & b_6 & b_7 & b_{11} & b_{12} \end{pmatrix}$$

$$\mathbf{L}_g^l = \left( L_{g\theta r1}^l, L_{g\theta r2}^l, L_{g z r1}^l, L_{g z r2}^l, L_{g r \theta 1}^l, L_{g r \theta 2}^l, L_{g z \theta 1}^l, L_{g z \theta 2}^l, L_{g r z 1}^l, L_{g r z 2}^l, L_{g \theta z 1}^l, L_{g \theta z 2}^l \right)^T$$

The coefficients of these modified matrices are expressed in terms of known quantities in Appendix A.

The transverse leakage moments for this new fourth-order cylindrical geometry model are calculated exactly as before. In addition, the cylindrical geometry computational algorithm which reduces the 6x6 response matrix equation to an equivalent 4x4 matrix equation remains valid.

The accuracy of this cylindrical geometry NEM algorithm with fourth-order polynomial expansions for the r,  $\theta$ , and z direction fluxes will now be tested on Dodds' steady-state R-Z geometry benchmark problem.

The first calculation to be performed with this first fourth-order r direction flux expansion approximation was Dodds' R-Z benchmark problem with a radial mesh spacing of 20 cm and an axial mesh spacing of 18.75 cm. The axially symmetric nodal power results of this calculation are compared to Dodds' reference solution in Table 4.17. The global results of this calculation appear in Table 4.18. As shown in the first of these tables, this calculation, in general, gives slightly worse results for the same mesh spacing than did the NEM coding with only a third-order r direction flux expansion.

The next calculation was similar to the first except that the radial mesh spacing was increased to 40 cm while the axial mesh spacing was increased to 37.5 cm. The nodal power results of this calculation appear as the third entry in Table 4.17. As was the case with the smaller mesh spacing, the new fourth-order radial flux expansion coding gives worse results than the third order radial flux expansion coding with the same mesh spacing.

Based upon these unfortunate results, one must conclude that the approximation used in calculating the integral which appears in the first or current derivative term (Eqn 4.6) of the second-order r direction flux moment equation may not be the most

**Table 4.17 Reference, Fine and Coarse Mesh Fourth-Order (1st Approx) Nodal Power Distributions and Percentage Errors for Dodds' Two-Dimensional Benchmark Problem**

0.6067	0.6069	0.6101	0.5953	0.4566	0.1714
0.6356(4.76)	0.6368(4.92)	0.6409(5.04)	0.6314(6.06)	0.4890(7.10)	0.1815(5.88)
0.6198(2.16)	0.6222(2.51)	0.6285(3.01)	0.6325(6.25)	0.5004(9.60)	0.1835(7.05)
1.2727	1.2755	1.2849	1.2526	0.9604	0.3614
1.2841(0.90)	1.2857(0.80)	1.2922(0.57)	1.2712(1.48)	0.9833(2.39)	0.3650(0.99)
1.2526(-1.58)	1.2565(-1.49)	1.2674(-1.36)	1.2745(1.75)	1.0074(4.90)	0.3699(2.35)
1.6740	1.6740	1.6777	1.6242	1.2330	0.4651
1.6647(-0.56)	1.6645(-0.57)	1.6668(-0.65)	1.6265(0.14)	1.2438(0.87)	0.4659(0.17)
1.6238(-3.00)	1.6266(-2.83)	1.6348(-2.56)	1.6305(0.39)	1.2690(2.92)	0.4734(1.79)
1.7789	1.7756	1.7681	1.6928	1.2484	0.4883
1.7420(-2.08)	1.7388(-2.07)	1.7336(-1.95)	1.6708(-1.30)	1.2344(-1.12)	0.4842(-0.83)
1.6990(-4.49)	1.6990(-4.31)	1.6998(-3.86)	1.6760(-0.99)	1.2489(0.04)	0.4970(1.79)
1.7969	1.7931	1.7807	1.6919	1.2329	0.4874
1.7569(-2.23)	1.7515(-2.32)	1.7407(-2.25)	1.6649(-1.59)	1.2163(-1.35)	0.4814(-1.22)
1.7134(-4.65)	1.7113(-4.65)	1.7066(-4.16)	1.6699(-1.30)	1.2260(-0.56)	0.4954(1.65)

### Reference

Fourth Order r (1st Approx), Fourth Order  $\theta$  z (20cm x 10° x 18.75cm) (% Error)

Fourth Order r (1st Approx), Fourth Order  $\theta$  z (40cm x 10° x 37.5cm) (% Error)

**Table 4.18 Summary of Results for Dodds' R-Z Benchmark Problem**

Case	Keff	Cray XMP CPU	Number of outer iters
		Time (sec)	
Reference	0.86705	-	-
4th r (1st), 4th $\theta$ z (20cm x $10^\circ$ x 18.75cm)	0.86731	50.753	64
4th r (1st), 4th $\theta$ z (40cm x $10^\circ$ x 37.5cm)	0.86766	12.201	60

accurate approximation possible. In the next sub-section an alternative approximation will be used to calculate this current derivative term.

#### 4.4.1.2 Fourth-Order Radial Flux Expansion, Second Current Derivative Approximation

From Equation 4.5 in section 4.4.1.1 the current derivative term in the second order  $r$  moment equation (Eqn 4.2) can be expressed as:

$$\begin{aligned} & \left\langle 3 \left( \frac{r - R}{\Delta r} \right)^2 - \frac{1}{4}, \frac{1}{r} \frac{d}{dr} (r j_{gr}^l(r)) \right\rangle \\ &= \frac{1}{\Delta r} \int_{R - \Delta r/2}^{R + \Delta r/2} \left( 3 \left( \frac{r - R}{\Delta r} \right)^2 - \frac{1}{4} \right) \left( \frac{1}{r} \frac{d}{dr} (r j_{gr}^l(r)) \right) dr \end{aligned} \quad (4.11)$$

In the course of obtaining the first-order radial direction moment equation, the approximation  $\frac{1}{r} \equiv \frac{1}{R}$  was used. This approximation yielded a first-order  $r$  moment equation (Eqn 3.114) whose form was the same as the first-order  $\theta$  and  $z$  direction moments equations. This approximation yielded acceptable results for Dodds' R-Z benchmark problem with a 20 cm radial and 18.75 cm axial mesh spacing.

In an attempt to achieve greater spatial accuracy while using large homogenous nodes, the radial direction intra-node flux was next approximated by a fourth-order rather than a third-order polynomial expansion. In deriving this higher order methodology, a non-analytic integral was encountered in the development of an expression for the second-order  $r$  direction flux moment. Following the precedent established in the development of the third-order  $r$  moment equation, the approximation  $\frac{1}{r} \equiv \frac{1}{R}$  was used to make this integral analytic. But the resulting fourth-order NEM methodology did not give a better nodal power distribution for Dodds' R-Z benchmark problem. Actually the results were generally worse than those obtained with the third-order  $r$  direction polynomial expansion.

Based upon its success in deriving the first-order radial moment equation, it was decided to allow the current derivative term in the second-order radial moment equation to take on the same form as the equivalent term in the  $\theta$  and  $z$  direction second-order flux moment equations. The current derivative term is then written as:

$$\begin{aligned} & \left\langle 3 \left( \frac{r - R}{\Delta r} \right)^2 - \frac{1}{4}, \frac{1}{r} \frac{d}{dr} (r j_{gr}^l(r)) \right\rangle \\ & \equiv \frac{1}{2\Delta r} (J_{gr+}^l - J_{gr-}^l) + \frac{3D_g^l}{\Delta r^2} (\phi_{gr+}^l + \phi_{gr-}^l) - \frac{6D_g^l}{\Delta r^2} \bar{\phi}_g^l \end{aligned} \quad (4.12)$$

The above form of the current derivative term can be obtained from the integral in Equation 4.11 by simply substituting a constant  $R$  for the radial variable which twice appears in the derivative term of Equation 4.12.

The second-order radial direction moment equation can now be expressed as:

$$\begin{aligned} \bar{\phi}_{gr2}^l = & -\frac{1}{A_g^l} \left( \frac{1}{2\Delta r} (J_{gr+}^l + J_{gr-}^l) + \frac{3D_g^l}{\Delta r^2} (\phi_{gr+}^l + \phi_{gr-}^l - 2\bar{\phi}_g^l) \right. \\ & \left. - \bar{Q}_{gr2}^l + \frac{1}{R\Delta\theta} L_{g\theta r2}^l + \frac{1}{\Delta z} L_{gz r2}^l \right) \end{aligned} \quad (4.13)$$

One immediately notices that this form of the second-order radial moment equation is completely equivalent to those in the  $\theta$  and  $z$  directions (Eqn's 3.108 and 3.110).

Substituting this new moment equation into the previously derived fourth-order  $r$  direction response matrix equations (Eqn's 4.8 and 4.9), one obtains the following 6x6 nodal response matrix equations:

$$[\mathbf{A}] \cdot J_g^{out,l} = [\mathbf{C}] \cdot J_g^{in,l} + [\mathbf{B}_1] \cdot Q_g^l + [\mathbf{B}_2] \cdot L_g^l \quad (4.14)$$

Where the  $\mathbf{A}$  and  $\mathbf{C}$  matrices take on the form as those in Equation 3.130 and the  $\mathbf{B}_1$  and  $\mathbf{B}_2$  matrices take on the same form as in Equation 4.10. Some elements of these matrices do however differ from their previous forms. These modified elements appear in Appendix A.

Again, as was the case with the previous fourth-order radial direction approximation, the twelve first and second order transverse leakage moments are calculated exactly as prescribed in the derivation of the Cartesian geometry algorithm, with the variables  $\Delta r^l$ ,  $R\Delta\theta^l$ , and  $\Delta z^l$  substituted for  $\Delta x^l$ . In addition, the previously derived cylindrical geometry computational algorithm, where the 6x6 response matrix equation is reduced to an equivalent 4x4 matrix equation, remains valid.

The accuracy of this new fourth-order cylindrical geometry algorithm will be tested first on Dodds' R-Z problem. Later the algorithm will be tested on a separate two-dimensional R- $\theta$  sample problem.

The first calculation to be performed with this second fourth-order radial direction flux expansion approximation was Dodds' R-Z sample problem with a radial mesh spacing of 20 cm and an axial mesh spacing of 18.75 cm. The axially symmetric nodal power distribution resulting from this calculation is compared to the reference solution in Table 4.19. The global results of this calculation appear in Table 4.20.

As shown in Table 4.19, this new fine-mesh calculation gives an error distribution very much the same as that obtained with the third order radial direction expansion calculation shown in Table 4.15. This error distribution, however, is much smaller than the one obtained with the first approximation to the second-order  $r$  moment equation (Table 4.17). One also notices from Table 4.20 that this 12x3x28 mesh calculation consumes approximately 40% less CPU time than its counterparts in Tables 4.16 and 4.18. This decrease in run time is the result of substantial coding efficiency upgrades, some of which are made possible when similar flux moment equations are used in all three coordinate directions.

**Table 4.19 Reference, Fine and Coarse Mesh Fourth-Order (2nd Approx) Nodal Power Distributions and Percentage Errors for Dodds' Two-Dimensional Benchmark Problem**

0.6067	0.6069	0.6101	0.5953	0.4566	0.1714
0.6380(5.16)	0.6391(5.30)	0.6428(5.35)	0.6301(5.84)	0.4863(6.51)	0.1828(6.64)
0.6299(3.82)	0.6315(4.05)	0.6362(4.27)	0.6274(5.39)	0.4896(7.23)	0.1879(9.62)
1.2727	1.2755	1.2849	1.2526	0.9604	0.3614
1.2892(1.30)	1.2903(1.16)	1.2960(0.86)	1.2686(1.28)	0.9780(1.84)	0.3674(1.65)
1.2732(0.04)	1.2753(-0.02)	1.2829(-0.16)	1.2645(0.95)	0.9866(2.73)	0.3785(4.73)
1.6740	1.6740	1.6777	1.6242	1.2330	0.4651
1.6713(-0.16)	1.6704(-0.22)	1.6717(-0.36)	1.6231(-0.07)	1.2382(0.42)	0.4682(0.67)
1.6507(-1.39)	1.6510(-1.37)	1.6547(-1.37)	1.6170(-0.44)	1.2479(1.21)	0.4814(3.51)
1.7789	1.7756	1.7681	1.6928	1.2484	0.4883
1.7490(-1.68)	1.7451(-1.72)	1.7387(-1.66)	1.6664(-1.56)	1.2315(-1.35)	0.4848(-0.71)
1.7273(-2.90)	1.7246(-2.87)	1.7208(-2.67)	1.6591(-1.99)	1.2381(-0.82)	0.4978(1.95)
1.7969	1.7931	1.7807	1.6919	1.2329	0.4874
1.7639(-1.84)	1.7578(-1.97)	1.7458(-1.96)	1.6603(-1.87)	1.2145(-1.49)	0.4812(-1.26)
1.7421(-3.05)	1.7372(-3.12)	1.7274(-2.99)	1.6524(-2.33)	1.2201(-1.04)	0.4930(1.16)

### Reference

Fourth Order r (2nd Approx), Fourth Order  $\theta$  z (20cm x  $10^o$  x 18.75cm) (% Error)

Fourth Order r (2nd Approx), Fourth Order  $\theta$  z (40cm x  $10^o$  x 37.5cm) (% Error)

**Table 4.20 Summary of Results for Dodds' R-Z Benchmark Problem**

Case	Keff	Cray XMP CPU Time (sec)	Number of outer iters
Reference	0.86705	-	-
4th r (2nd), 4th $\theta$ z (20cm x 10° x 18.75cm)	0.86723	30.421	59
4th r (2nd), 4th $\theta$ z (40cm x 10° x 37.5cm)	0.86736	7.196	54

The next calculation to be performed was again Dodd's R-Z benchmark. Here the radial mesh was increased to 40 cm while the axial mesh spacing was increased to 37.5 cm. The results of this calculation also appear in Table 3.19. Again, as was the case with the finer mesh spacing, the new fourth-order radial expansion calculation gives approximately the same nodal power error distribution as did the third-order radial expansion calculation. Just as before, this new second-order  $r$  flux moment approximation yielded more accurate results than did the original second-order  $r$  flux moment approximation. From Table 4.20 one also notices that this latest calculation consumes approximately 30 to 40% less CPU time than the equivalent calculations in Tables 4.16 and 4.18. Again this is the result of coding efficiency upgrades.

From the results of the above calculations, one can see that while the latest fourth-order radial flux expansion approximation yields no better results than the original third order approximation, it does run much more quickly. Because of the efficiency and accuracy inherent in running this new fourth-order radial flux expansion coding, this is the coding that will be used in all future cylindrical geometry NEM calculations. The radial mesh spacing used in these calculations, however, will have to remain relatively small in order to preserve spatial solution accuracy.

#### 4.4.2 Pedroni's R - $\theta$ Benchmark Problem

Since Dodds' benchmark problem was only a two-dimensional R-Z test calculation, the accuracy of the fourth-order NEM approximations in the  $\theta$  direction has not yet been tested. As a result, a cylindrical geometry test problem which exhibited  $\theta$  direction dependencies was searched for. An R- $\theta$  geometry test problem was discovered in a 1985 article by Pedroni and Pollachini in the Italian journal Energia Nucleare (37). This problem, a detailed description of which appears in Section B.5 of Appendix B, is a highly simplified, one-group, one-eighth core symmetric model of a light-water

reactor. The model consists of four types of fissile materials placed in a 15 sector R- $\theta$  geometry lattice. The reference flux calculation for this configuration was a finite difference VENTURE (35) calculation with 80 radial mesh and 256 equally spaced  $\theta$  mesh. In addition, Pedroni and his associates ran two coarse-mesh diffusion theory flux calculations on this model to test the speed and accuracy of their unique analytically based nodal approximation. The first of these calculations employed 10 radial nodes and 32 equally spaced  $\theta$  nodes. The second calculation employed 5 radial nodes and 16  $\theta$  nodes. As detailed in his journal article, Pedroni's 10x32 mesh calculation displayed a maximum node average flux error of 1.6% as compared to the base fine-mesh VENTURE calculation while his 5x16 mesh calculation had a maximum error of 5.8% .

At this point several one-group R- $\theta$ -Z geometry NEM calculations were run on Pedroni's model to test the accuracy of the previously derived  $\theta$  direction flux approximations. To perform these calculations, a one-group version of the cylindrical geometry NEM coding had to be derived. Only minor FORTRAN modifications were required to produce a one-group NEM code. All looping over groups had to be confined to only group one, while the group one source moments had to be set equal to the group one rather than group two fission source moments.

The first R- $\theta$ -Z NEM flux calculation for the Pedroni problem used 10 mesh cells in the radial direction, 32 equally spaced mesh cells in the  $\theta$  direction, and three 10 cm z-direction mesh with reflected boundaries on both top and bottom. After running this calculation, it soon became obvious that Pedroni had used an albedo boundary condition at the problem outer radial boundary, rather than a zero-flux or no-incoming-current boundary condition. Since an albedo boundary condition has not yet been formulated for the Nodal Expansion Method, a reflector material was placed at the problem outer radial boundary. The characteristics of this reflector; thickness, diffusion coefficient, and

absorption cross-section, were then adjusted to reduce the differences in node average fluxes between this calculation and Pedroni's fine-mesh VENTURE calculation. The reflector which best modelled Pedroni's albedo condition contained two radial mesh with a total width of 17.5 cm, a diffusion coefficient of 1.57 cm, and a macroscopic absorption cross-section of  $0.0212 \text{ cm}^{-1}$ .

The normalized node average fluxes produced by this and other more coarse NEM calculations are compared to Pedroni's reference VENTURE calculation in Table 4.21, while the global results of these calculations can be found in Table 4.22. One notices from the second entry in Table 4.21 that careful choice of a proper reflector has reduced the maximum error in the nodal flux distribution to slightly more than 2%. This error is only slightly worse than the 1.6% maximum error obtained by Pedroni using his nodal method with similar mesh spacing. The greater error in the NEM calculation is most likely the result of using a physical reflector material rather than an albedo boundary condition. In addition, one notices in Table 4.22 that the 16x32x3 mesh calculation required a much larger than usual number of outer iterations to achieve convergence. This is a result of an anomaly in which coarse-mesh rebalancing thwarted rather than enhanced problem convergence in this calculation. As a result, the coarse-mesh rebalancing routine was turned off during this calculation.

In the second calculation of the flux in Pedroni's model, the number of  $\theta$  mesh was reduced from 32 to 16 while the number of radial mesh remained at 10 in the fuel region and 2 in the 17.5 cm reflector. The results of this calculation appear as the third entry in Table 4.21. As can be seen in the table, halving the number of  $\theta$  mesh has almost no effect on the resulting one-group flux distribution. Using 32 mesh in the  $\theta$  direction is thus unnecessary in this model. In addition, one notices from Table 4.22 that this 16  $\theta$  mesh calculation uses only one-fourth as much CPU time as the previous calculation.

**Table 4.21 Reference, Fine and Coarse Mesh Fourth-Order (2nd Approx) Nodal Power Distributions and Percentage Errors for Pedroni's Two-Dimensional R- $\theta$  Benchmark Problem**

1.0675	1.0206	0.8752	1.1134	0.7701
1.0831(1.47)	1.0212(0.06)	0.8645(-1.23)	1.1117(-0.15)	0.7860(2.07)
1.0833(1.48)	1.0212(0.06)	0.8644(-1.23)	1.1116(-0.16)	0.7860(2.07)
1.0850(1.64)	1.0247(0.40)	0.8522(-2.63)	1.1140(0.06)	0.7986(3.71)
1.0835(1.50)	1.0268(0.61)	0.8509(-2.78)	1.1127(-0.06)	0.7986(3.70)
1.0541	1.0873	1.3605	1.2255	0.7536
1.0710(1.61)	1.0798(-0.69)	1.3367(-1.75)	1.2176(-0.64)	0.7635(1.32)
1.0712(1.62)	1.0797(-0.70)	1.3366(-1.76)	1.2177(-0.64)	0.7636(1.33)
1.0799(2.45)	1.0677(-1.80)	1.3214(-2.87)	1.2191(-0.52)	0.7702(2.20)
1.0784(2.31)	1.0672(-1.85)	1.3227(-2.78)	1.2231(-0.19)	0.7733(2.62)
1.0296	0.9886	1.3788	0.7904	0.4848
1.0507(2.05)	0.9770(-1.17)	1.3568(-1.59)	0.7870(-0.43)	0.4832(1.73)
1.0510(2.07)	0.9770(-1.17)	1.3566(-1.61)	0.7870(-0.44)	0.4931(1.71)
1.0724(4.15)	0.9535(-3.56)	1.3577(-1.53)	0.7900(-0.06)	0.4837(1.83)
1.0711(4.03)	0.9519(-3.71)	1.3607(-1.32)	0.7885(-0.25)	0.4907(1.22)

### Reference

**Fourth Order R- $\theta$ -Z (12x32x3 mesh) (% Error)**

**Fourth Order R- $\theta$ -Z (12x16x3 mesh) (% Error)**

**Fourth Order R- $\theta$ -Z (6x16x3 mesh) (% Error)**

**Fourth Order R- $\theta$ -Z (6x4x3 mesh) (% Error)**

**Table 4.22 Summary of Results for Pedroni's R-θ Benchmark Problem**

Case	Keff	Cray XMP CPU	Number of outer iterations
Reference	-	-	-
<b>4<sup>th</sup> R-θ-Z</b>			
(12x32x3 mesh)	1.0435	48.011	170
<b>4<sup>th</sup> R-θ-Z</b>			
(12x16x3 mesh)	1.0435	16.949	111
<b>4<sup>th</sup> R-θ-Z</b>			
(6x16x3 mesh)	1.0437	4.720	61
<b>4<sup>th</sup> R-θ-Z</b>			
(6x4x3 mesh)	1.0438	1.007	44

This large reduction in computation time is the result of both halving the total number of mesh cells and the fact that coarse-mesh rebalancing has slightly enhanced problem convergence in this case.

In the third NEM calculation of the flux in Pedroni's model, the number of radial mesh cells was reduced from 12 to 6 or one only radial mesh cell per sector shaped assembly. The number of  $\theta$  mesh used in this model remained 16. As shown in the fourth entry on Table 4.21, the maximum node average flux error as compared to VENTURE has risen slightly to more than 4%, a value slightly higher than would be acceptable in a kinetics calculation. As was the case with Dodds' R-Z problem, these results demonstrate the need to use relatively small mesh spacings in the radial direction. But, the maximum flux error in this calculation is not as large as the almost 6% maximum error reported by Pedroni for his nodal solution with equivalently sized mesh cells. One can therefore conclude that the fourth-order cylindrical geometry NEM methodology may be a more accurate numerical approximation to the neutron diffusion equation than Pedroni's analytically based model. Table 4.22 shows that the CPU time used by this 6x16x3 mesh calculation is about one-fourth of that used by the finer 12x16x3 mesh calculation. Again this is due to a combination of halving the number of mesh cells and greater convergence acceleration by the coarse-mesh rebalancing algorithm.

In the fourth and final NEM flux calculation for Pedroni's R- $\theta$  model, the number of  $\theta$  mesh was cut by a factor of four, a reduction from 16 to 4 equally spaced mesh cells. For this calculation the number of radial mesh remained at its minimum possible number, six. The node average flux results of this calculation appear as the fifth and final entry in Table 4.21. As shown in the table, the results of this 4  $\theta$  mesh calculation are very close to those obtained with the previous 16  $\theta$  mesh calculation. This again demonstrates the accuracy of the fourth-order polynomial flux expansion approximation

in the  $\theta$  direction. Finally, upon observing Table 4.22, one notices an almost factor of five decrease in computation time for the  $6 \times 4 \times 3$  mesh problem compared to the  $6 \times 16 \times 3$  mesh problem. This decrease can be attributed to a combination of reducing the number of mesh cells by a factor of four and an additional improvement in the efficiency of coarse-mesh rebalancing acceleration.

The results of various NEM calculations on both Dodds' R-Z Benchmark Problem and Pedroni's R- $\theta$  Benchmark problem indicate that the final cylindrical geometry NEM coding with fourth-order polynomial approximations for the one-dimensional fluxes in the R,  $\theta$ , and Z directions will provide sufficiently accurate three-dimensional coarse-mesh spatial flux and/or power distributions for a multi-dimensional kinetics calculation.

#### 4.5 Conclusions

This chapter has included a large number of calculational results to a total of seven two and three-dimensional steady-state sample problems. The results of these calculations substantiate the suitability of the fourth-order Nodal Expansion Method for calculating light water reactor assembly power distributions in either Cartesian or cylindrical geometries using a very coarse mesh spacing.

## CHAPTER 5

### DERIVATION AND VALIDATION OF THE TRANSIENT NODAL EXPANSION METHOD

#### 5.1 Introduction

In this chapter the numerics of the transient version of the Nodal Expansion Method will be derived and tested. At first, the transient time differencing scheme will be based on a simple but stable backwards finite differencing technique. Later, a more advanced time differencing scheme, based upon an exponential time transformation technique developed at the Savannah River Laboratory by M. R. Buckner and J. W. Stewart (29), will be examined and tested.

#### 5.2 The Backwards Time Differencing Technique

##### 5.2.1 Derivation of the Time-Dependent Numerics

The derivation of the backwards-differenced, time-dependent Nodal Expansion Method begins by first introducing the two energy group, multidimensional transient neutron diffusion and precursor equations within node 1 which has constant neutronic properties.

$$\begin{aligned} \frac{1}{V_1} \frac{\partial}{\partial t} \phi_1^l(r, t) - D_1^l \nabla^2 \phi_1^l(r, t) + (\Sigma_{a1}^l + \Sigma_{12}^l - (1 - \beta) \nu \Sigma_{f1}^l) \phi_1^l(r, t) \\ = (1 - \beta) \nu \Sigma_{f2}^l \phi_2^l(r, t) + \sum_{d=1}^D \lambda_d C_d^l(r, t) \end{aligned} \quad (5.1)$$

$$\frac{1}{V_2} \frac{\partial}{\partial t} \phi_2^l(r, t) - D_2^l \nabla^2 \phi_2^l(r, t) + \Sigma_{a2}^l \phi_2^l(r, t) = \Sigma_{12}^l \phi_1^l(r, t) \quad (5.2)$$

$$\frac{\partial}{\partial t} C_d^l(r, t) = \beta_d (\nu \Sigma_{f1}^l \phi_1^l(r, t) + \nu \Sigma_{f2}^l \phi_2^l(r, t)) - \lambda_d C_d^l(r, t), d = 1, D \quad (5.3)$$

where

$V_g \equiv$  energy group g neutron velocity

$\phi_r^l(r, t) \equiv$  time and space dependent neutron flux in energy group g

$D_g^l \equiv$  group g diffusion coefficient

$\Sigma_{ag}^l \equiv$  group g absorption cross-section

$\Sigma_{12}^l \equiv$  group 1 to 2 scattering cross-section

$\nu \equiv$  average number of neutrons produced by a fission event

$\Sigma_{fg}^l \equiv$  group g fission cross-section

$C_d^l(r, t) \equiv$  time and space dependent group d neutron precursor concentration

$\beta_d \equiv$  fraction of fission events which produce group d neutron precursors

$$\beta \equiv \sum_{d=1}^D \beta_d$$

$\lambda_d \equiv$  decay constant for neutron precursor group d

$D \equiv$  total number of delayed neutron precursor groups

To begin the derivation, the time derivatives of the fluxes in Equations 5.1 and 5.2 can be approximated by a simple first-order finite difference expression:

$$\frac{\partial}{\partial t} \phi_g^l(r, t) = \frac{\phi_g^l(r) - \phi_g^{l, old}(r)}{\Delta t} \quad (5.4)$$

where

$\phi_g^{l, old}(r) \equiv$  spatially dependent flux at time  $t^{old}$

$\phi_g^l(r) \equiv$  spatially dependent flux at time  $t^{new}$

$$\Delta t \equiv t^{new} - t^{old}$$

One must now develop some approximate expression for the delayed neutron source contribution to the fast group equation (Eqn. 5.1). Here the so-called time-integrated

approximation to the delayed neutron precursor equations (25,28) is employed. This approximation can be derived by integrating Equation 5.3 from  $t^{old}$  to  $t^{new}$ . This integration results in:

$$C_d^l(r, t^{new}) = C_d^l(r, t^{old}) e^{-\lambda_d \Delta t} + \beta_d \int_{t^{old}}^{t^{new}} (\nu \Sigma_{f1}^l \phi_1^l(r, t) + \nu \Sigma_{f2}^l \phi_2^l(r, t)) e^{-\lambda_d(t^{new}-t)} dt \quad (5.5)$$

In order to solve the above integral, one must develop some type of functional form for the time dependence of the neutron fission source between  $t^{old}$  and  $t^{new}$ . In the time-integrated approximation, one simply assumes that the fission source varies linearly between time steps (28), i.e.:

$$\nu \Sigma_{fg}^l \phi_g^l(r, t) = \frac{\nu \Sigma_{fg}^l \phi_g^l(r) - \nu \Sigma_{fg}^{l, old} \phi_g^{l, old}(r)}{\Delta t} (t - t^{old}) + \nu \Sigma_{fg}^{l, old} \phi_g^{l, old}(r) \quad (5.6)$$

Incorporating this approximation into Equation 5.5, one obtains:

$$C_d^l(r, t^{new}) = C_d^l(r, t^{old}) e^{-\lambda_d \Delta t} + \frac{\beta_d}{\lambda_d} \left( (\nu \Sigma_{f1}^l \phi_1^l(r) + \nu \Sigma_{f2}^l \phi_2^l(r)) \left( 1 - \frac{1 - e^{-\lambda_d \Delta t}}{\lambda_d \Delta t} \right) \right) + \frac{\beta_d}{\lambda_d} \left( \left( \nu \Sigma_{f1}^{l, old} \phi_1^{l, old}(r) + \nu \Sigma_{f2}^{l, old} \phi_2^{l, old}(r) \right) \cdot \left( \frac{1 - e^{-\lambda_d \Delta t}}{\lambda_d \Delta t} - e^{-\lambda_d \Delta t} \right) \right) \quad (5.7)$$

When Equations 5.4 and 5.7 are substituted in Equations 5.1 and 5.2, equations of the following form result:

$$-D_g^l \nabla^2 \phi_g^l(r) + A_g^l \phi_g^l(r) = Q_g^l(r), g = 1, 2 \quad (5.8)$$

where

$$A_1^l = \Sigma_{a1}^l + \Sigma_{12}^l + \frac{1}{V_1 \Delta t} - (1 - \beta) \nu \Sigma_{f1}^l + \sum_{d=1}^D \beta_d \nu \Sigma_{f1}^l \left( 1 - \frac{1 - e^{-\lambda_d \Delta t}}{\lambda_d \Delta t} \right)$$

$$A_2^l = \Sigma_{a2}^l + \frac{1}{V_2 \Delta t}$$

$$\begin{aligned}
Q_1^l(r) &= \frac{\phi_1^{l,old}(r)}{V_1 \Delta t} + (1 - \beta) \nu \Sigma_{f2}^l \phi_2^l(r) \\
&\quad + \sum_{d=1}^D \lambda_d \left( C_d^{l,old}(r) e^{-\lambda_d \Delta t} + \frac{\beta_d}{\lambda_d} \left( \nu \Sigma_{f2}^l \phi_2^l(r) \left( 1 - \frac{1 - e^{-\lambda_d \Delta t}}{\lambda_d \Delta t} \right) \right. \right. \\
&\quad \left. \left. + \left( \nu \Sigma_{f1}^{l,old} \phi_1^{l,old}(r) + \nu \Sigma_{f2}^{l,old} \phi_2^{l,old}(r) \right) \left( \frac{1 - e^{-\lambda_d \Delta t}}{\lambda_d \Delta t} - e^{-\lambda_d \Delta t} \right) \right) \right) \\
Q_2^l(r) &= \frac{\phi_2^{l,old}(r)}{V_2 \Delta t} + \Sigma_{12}^l \phi_1^l(r)
\end{aligned}$$

As one can readily observe, Equation 5.8 is simply a fixed source version of Equations 3.2 or 3.73. As a result, diffusion theory flux solutions to Equation 5.8 can be obtained in either Cartesian or cylindrical geometry using the Nodal Expansion Method spatial approximations outlined in the previous chapter.

An additional feature does, however, appear in the transient version of the NEM. This feature is an outgrowth of expressing the spatially dependent neutron flux and source distributions within a node in terms of node average fluxes/sources and first and second order flux/source moments. In the transient NEM, these expressions result in analogous expansion of the intra-node spatially dependent precursor concentration in terms of a node average group d precursor concentration ( $\bar{C}_d^l$ ) and group d precursor concentration moments ( $\bar{C}_{d(x,y,z)(1,2)}^l$ ). These node average precursor concentrations and concentration moments can be easily related to old and new time node average fluxes and flux moments through equations similar to Equation 5.7 .

### 5.2.2 Implementation and Acceleration of the Computational Algorithm

The backwards time-differenced or fully implicit transient version of the Nodal Expansion Method is solved at each time step using essentially the same numerical strategy as that described for the steady-state NEM. The basic difference between the solution methodologies lies in the fact that part of the source terms in a transient calculation

( $Q_1^l(r), Q_2^l(r)$ ) are initially known, thereby fixing the total source magnitude and eliminating the need to calculate an eigenvalue. The new time flux distribution does, however, need to be calculated by the previously described inner/outer iteration strategy. This is true because both the new time group flux distributions and the partial contribution of the new time sources and source moments are unknown.

Before proceeding to calculate the two-group flux distribution at time step one, one needs to have already obtained time zero values for the node average fluxes and flux moments. In addition, one should have also obtained values for the initial node average precursor concentrations and concentration moments. The initial time zero fluxes are calculated by performing an initial steady-state eigenvalue calculation. If the eigenvalue produced by this calculation is not exactly 1.0, the production (fission) cross-sections in all materials and energy groups are divided by K to force the initial material configuration to an exact steady state. The initial precursor concentrations are calculated from this steady-state condition by setting the time-derivative in Equation 5.3 to zero. The initial precursor concentrations can then be described by:

$$C_d^l(r, 0) = \frac{\beta_d}{\lambda_d} \left( \nu \Sigma_{f1}^{l,0} \phi_1^l(r, 0) + \nu \Sigma_{f2}^{l,0} \phi_2^l(r, 0) \right) \quad (5.9)$$

At each succeeding time step one must again have available previous time values for the fluxes and precursor concentrations. The old time fluxes are simply those calculated at the previous time step, while the previous time precursor concentrations are calculated from one and two step previous fluxes using Equation 5.7.

The final difference in numerical implementation between the static and the fully implicit transient versions of the Nodal Expansion Method lies in the means by which the coarse-mesh rebalancing factors are calculated. The equation which defines the

coarse-mesh rebalancing factors ( $f^m$ ) can be written in Cartesian geometry for the transient case as follows:

$$\begin{aligned}
 & -f^l \sum_{l \in S_m} \Delta^2(x, y, z) \sum_{g=1}^2 J_{g(x,y,z)}^{in,l} + f^m \sum_{m \in S_m} \Delta^2(x, y, z) \sum_{g=1}^2 J_{g(x,y,z)}^{out,m} \\
 & + f^m \sum_{m \in V_m} \Delta x \Delta y \Delta z \sum_{g=1}^2 \left( \Sigma_{ag}^m - (1 - \beta) \nu \Sigma_{fg}^m + \frac{1}{V_g \Delta t} \right. \\
 & \quad \left. - \sum_{d=1}^D \beta_d \nu \Sigma_{fg}^m \left( 1 - \frac{1 - e^{-\lambda_d \Delta t}}{\lambda_d \Delta t} \right) \right) \bar{\phi}_g^m \\
 & = \sum_{m \in V_m} \Delta x \Delta y \Delta z \sum_{g=1}^2 \left( \frac{1}{V_g \Delta t} + \sum_{d=1}^D \beta_d \nu \Sigma_{fg}^{m,old} \right. \\
 & \quad \left. \cdot \left( \frac{1 - e^{-\lambda_d \Delta t}}{\lambda_d \Delta t} - e^{-\lambda_d \Delta t} \right) \right) \bar{\phi}_g^{m,old} \\
 & + \sum_{m \in V_m} \Delta x \Delta y \Delta z \sum_{d=1}^D \lambda_d \bar{C}_d^{m,old} e^{-\lambda_d \Delta t}
 \end{aligned} \tag{5.10}$$

One can see that Equation 5.10 represents a fixed source rather than an eigenvalue equation. A neutron balance like Equation 5.10 can be written for each coarse-mesh node. If these balance equations are combined in matrix format, the following formulation results:

$$[M] \underline{f} = \underline{c} \tag{5.11}$$

where

$\underline{f} \equiv$  vector of rebalance factors

$[M] \equiv$  seven stripe matrix representing L. H. S. of Eqn 5.10

$\underline{c} \equiv$  constant vector representing R. H. S. of Eqn 5.10

The simple matrix equation shown in Equation 5.11 can be solved by a single pass through a Gaussian elimination routine, rather than the multiple passes required to solve Equation 3.61 by the Weilant Method of Fractional Iteration.

### 5.2.3 Two-Dimensional Benchmarking

The fully implicit backwards differenced transient NEM methodology will now be tested on two two-dimensional transient reactor sample problems. The first of these problems is a simplified X-Y geometry transient without feedback developed by Hageman and Yasinsky at the Bettis Atomic Laboratory for use in testing their finite-difference based two-dimensional reactor kinetics code TWIGL (38). This problem will hereafter be referred to as the TWIGL seed-blanket problem. The second two-dimensional sample problem is a non-feedback transient version of Dodds' R-Z sample problem initially described in Chapter 4.

#### 5.2.3.1 TWIGL Seed-Blanket Problem

The TWIGL seed-blanket problem is a simplified X-Y geometry test problem which models a 160 cm square unreflected seed-blanket reactor consisting of three fissile material types. A complete description of the physical configuration and material specifications of this sample problem appear in Section B.6 of Appendix B. As shown in the specifications, two different delayed super-critical transients are initiated by either a step or a ramp change in the thermal absorption cross-section of material type 2. The reference transient total power versus time solutions for these two situations were originally obtained by Hageman and Yasinsky (38), while the reference steady-state and asymptotic normalized assembly power distributions were obtained by R. D. Lawrence (25).

When the TWIGL seed-blanket problem was solved using the backwards differenced Nodal Expansion Method, three different X-Y spatial mesh distributions were used. These mesh distributions, respectively designated as very coarse, coarse, and fine, are

defined as follows:

Region	Very Coarse	Coarse	Fine
$0 \leq x,y \leq 24\text{cm}$	24cm	12cm	8cm
$24 \leq x,y \leq 56\text{cm}$	32cm	16cm	8cm
$56 \leq x,y \leq 80\text{cm}$	24cm	12cm	8cm

Solutions to the TWIGL seed-blanket transient were actually obtained using three-dimensional transient NEM coding. As a result, for each of the three X-Y mesh spacings, three 1.0 cm z-mesh were employed. To obtain a solution dependent on only the x and y problem dimensions, reflected boundary conditions were used at all + z and - z node boundaries.

As previously discussed, steady-state eigenvalue calculations first had to be performed to initialize the TWIGL seed-blanket transient solutions. The one-eighth-core symmetric steady-state normalized nodal power distributions produced using these three progressively larger mesh spacings are compared to Lawrence's reference fine-mesh solution in Table 5.1. The number of outer iterations required to converge these steady-state solutions along with the calculated steady-state eigenvalues are displayed in Table 5.2. As shown in Table 5.1, all three mesh distributions yield extremely accurate steady-state nodal power distributions. Even with the 3x3x3 very coarse mesh distribution, the maximum error in assembly power was only slightly greater than 1%.

Table 5.3 shows samples of the time-dependent normalized power produced for the TWIGL seed-blanket ramp perturbed transient by the backwards time-differenced NEM coding for the three chosen mesh spacings. All three solutions used a transient time step of 5 milliseconds, consequently, each transient solution was run for 1000 time steps. Table 5.3 also contains a tabulation of the number of transient outer iterations

**Table 5.1 Reference, Very Coarse, Coarse, and Fine Mesh Fourth-Order Steady-State Nodal Power Distributions and Percentage Errors for the TWIGL Seed-Blanket Two Dimensional Transient Benchmark Problem**

0.3916	0.2862	0.0932
0.3918(0.05)	0.2864(0.07)	0.0932(0.00)
0.3932(0.41)	0.2873(0.38)	0.0933(0.11)
0.3936(0.51)	0.2881(0.66)	0.0932(0.00)
2.0268	1.5771	
2.0263(-0.02)	1.5768(-0.02)	
2.0241(-0.13)	1.5747(-0.15)	
2.0274(0.03)	1.5627(-0.91)	
1.2628		
1.2638(0.08)		
1.2682(0.43)		
1.2777(1.18)		

#### Reference

Fine (10x10x3) mesh (% Error)

Coarse (6x6x3) mesh (% Error)

Very Coarse (3x3x3) mesh (% Error)

**Table 5.2 Summary of Steady-State Results for the TWIGL Seed-Blanket Benchmark Problem**

Case	Keff	Initial outer iterations
Reference	0.913214	
Very Coarse (3x3x3) Mesh	0.913200	26
Coarse (6x6x3) Mesh	0.913128	43
Fine (10x10x3) Mesh	0.913201	40

required for solution convergence at the displayed time steps, and a summary of the CPU time required to complete each of the transient solutions.

As shown in Table 5.3, both the coarse and fine mesh calculations yield almost the same time-dependent reactor power. In addition, neither calculation is in error with respect to the reference power distribution at the displayed times by greater than one-fourth of one percent. But, as shown, the coarse-mesh solution achieves this level of accuracy using less than one-half the computation time of the fine mesh calculation. The results of the very coarse mesh calculation, while using only one-third as much CPU time as the equivalent coarse-mesh calculation, undershoot the reference normalized core powers by as much as 1%. This level of error in integrated power is possibly too large to be acceptable in a typical transient calculation.

Table 5.4 shows the same information as does Table 5.3, except for a step rather than a ramp cross-section perturbation. As shown in the table, all three NEM step transient calculations provide integrated power results closer to the reference than was the case for the ramp cross-section perturbation. In this case the maximum error using coarse or fine mesh is below one-tenth of one percent, while the maximum error using obtained with very coarse mesh is below 0.8%. Larger errors quite possibly could be found for any of these calculations in the 0 to 0.1 second time frame, except that the reference powers at these times are not known. Finally, one notices that the step perturbation calculations use 20 to 40% less CPU time than do the equivalent ramp perturbation calculations. This is mainly the result of the step perturbation time steps, after the initial large cross-section change, requiring fewer outer iterations to achieve convergence.

At the final ( $100^{th}$ ) time step for each of the above six TWIGL seed-blanket ramp or step perturbation transient calculations, a normalized assembly power distribution

**Table 5.3 Reference, Very Coarse, Coarse, and Fine Mesh Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Ramp Transient Benchmark Problem**

Power (no. of outers)				
Time (sec)	V. Coarse	Coarse	Fine	Ref
0.05	1.124 (14)	1.125 (11)	1.125 (8)	1.125
0.1	1.306 (14)	1.309 (11)	1.309 (8)	1.307
0.2	1.948 (14)	1.961 (11)	1.962 (8)	1.957
0.3	2.059 (5)	2.074 (4)	2.075 (4)	2.074
0.4	2.076 (5)	2.091 (4)	2.092 (4)	2.096
0.5	2.093 (5)	2.108 (4)	2.110 (4)	2.109
CPU time	14.61	48.71	119.4	-

**Table 5.4 Reference, Very Coarse, Coarse, and Fine Mesh Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Step Transient Benchmark Problem**

Power (no. of outers)					
Time (sec)	V. Coarse	Coarse	Fine	Ref	
0.05	2.018 (7)	2.031 (6)	2.033 (6)	-	
0.1	2.045 (6)	2.059 (4)	2.060 (4)	2.061	
0.2	2.062 (5)	2.077 (4)	2.078 (4)	2.078	
0.3	2.079 (5)	2.094 (4)	2.095 (4)	2.095	
0.4	2.096 (5)	2.112 (4)	2.113 (4)	2.113	
0.5	2.113 (5)	2.129 (4)	2.131 (4)	2.131	
CPU time	10.18	33.59	93.56	-	

was again output. These six asymptotic assembly power distributions are compared to Lawrence's reference solution in Table 5.5. As is readily apparent from the table, step and ramp perturbations with equivalent mesh spacings yield exactly the same asymptotic power distributions. This is to be expected, since asymptotic power distributions depend only on the values of the final assembly cross-sections, not the time sequence by which these final cross sections are reached. In addition, Table 5.5 shows that the errors in the asymptotic assembly powers are very near those obtained at steady state. As a result, the normalized assembly power distribution which exists long after the initiating transient cross-section perturbation appears to be solely a function of the NEM spatial rather than temporal discretization approximations.

Tables 5.6 and 5.7 show the time-dependent TWIGL seed-blanket core powers obtained using time step sizes greater or less than 5 milliseconds. All calculations in these tables use a 6x6x3 coarse mesh spatial discretization, chosen because of its good balance between spatial accuracy and computation time requirements.

Table 5.6 shows the results of TWIGL seed-blanket ramp perturbation performed using respective time step sizes of 1, 10, and 50 milliseconds. As can be readily observed in the table, the use of a time step size of 1 rather than 5 ms decreases the maximum error in integrated power to below 0.1% at the expense of more than triple the CPU time. This relatively insignificant increase in accuracy does not justify the great increase in computation time requirements. In contrast, the 10 ms solution, on average, yields only slightly greater errors than did the 5 ms coarse-mesh ramp perturbation solution of Table 5.3. Beside, this solution was obtained using 40% less CPU time than the 5 ms calculation. Clearly it is advantageous to increase the time step size for the ramp perturbation from 5 to 10 ms.

**Table 5.5 Reference, Very Coarse, Coarse, and Fine Mesh Fourth-Order Asymptotic Nodal Power Distributions and Percentage Errors for the TWIGL Seed-Blanket Two Dimensional Transient Step and Ramp Benchmark Problems**

0.3881	0.2914	0.0955
0.3883(0.05)	0.2915(0.03)	0.0955(0.00)
0.3883(0.05)	0.2915(0.03)	0.0955(0.00)
0.3897(0.41)	0.2924(0.34)	0.0956(0.10)
0.3897(0.41)	0.2924(0.34)	0.0956(0.10)
0.3903(0.57)	0.2929(0.51)	0.0956(0.10)
0.3903(0.57)	0.2929(0.51)	0.0956(0.10)
2.0027	1.6239	
2.0018(-0.04)	1.6238(-0.01)	
2.0018(-0.04)	1.6238(-0.01)	
1.9995(-0.16)	1.6218(-0.13)	
1.9995(-0.16)	1.6218(-0.13)	
2.0016(-0.05)	1.6117(-0.75)	
2.0016(-0.05)	1.6117(-0.75)	
1.2362		
1.2368(0.05)		
1.2368(0.05)		
1.2412(0.40)		
1.2412(0.40)		
1.2508(1.18)		
1.2508(1.18)		

#### Reference

Fine (10x10x3) mesh Ramp (% Error)

Fine (10x10x3) mesh Step (% Error)

Coarse (6x6x3) mesh Ramp (% Error)

Coarse (6x6x3) mesh Step (% Error)

Very Coarse (3x3x3) mesh Ramp (% Error)

Very Coarse (3x3x3) mesh Step (% Error)

**Table 5.6 Reference, 1, 10, and 50 ms Time Step Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Ramp Transient Benchmark Problem**

Power (no. of outer iterations)				
Time (sec)	1ms	10ms	50ms	Ref
0.05	1.125 (6)	1.125 (11)	1.128 (15)	1.125
0.1	1.308 (6)	1.309 (11)	1.312 (15)	1.307
0.2	1.959 (7)	1.962 (11)	1.977 (15)	1.957
0.3	2.074 (4)	2.073 (4)	2.071 (6)	2.074
0.4	2.091 (4)	2.091 (4)	2.091 (5)	2.096
0.5	2.108 (4)	2.108 (4)	2.109 (5)	2.109
CPU time	174.3	27.33	9.442	-

**Table 5.7 Reference, 1, 10, and 50 ms Time Step Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Step Transient Benchmark Problem**

Power (no. of outers)				
Time (sec)	1ms	10ms	50ms	Ref
0.05	2.042 (5)	2.016 (7)	1.872 (15)	-
0.1	2.060 (4)	2.058 (5)	2.026 (13)	2.061
0.2	2.077 (4)	2.076 (4)	2.072 (5)	2.078
0.3	2.095 (4)	2.094 (4)	2.090 (5)	2.095
0.4	2.112 (4)	2.111 (4)	2.108 (5)	2.113
0.5	2.130 (4)	2.129 (4)	2.125 (5)	2.131
CPU time	150.4	19.61	7.749	-

Finally, Table 5.6 shows the results of a TWIGL seed-blanket ramp perturbation calculation in which the problem time step size was increased to 50 ms. Here the normalized integrated core power is in error with respect to the reference solution by up to 1%. In spite of using a factor of three less CPU time than the 10 ms calculation, the overall error in this solution appears too large. The use of a time step size somewhere between 10 and 50 ms should yield the optimum balance between solution accuracy and computer usage for this calculation.

Table 5.7 presents the same information as did Table 5.6, except for the TWIGL seed-blanket step cross-section perturbation. This table shows that the 1 and 10 ms time step calculations yield results which compare very well to the previous coarse-mesh step perturbation calculation with 5 ms time steps. As was the case with the ramp perturbation, Table 5.7 shows that using a 1 ms time step yields only a slight increase in solution accuracy with a great increase in CPU time, while using a 10 ms time step decreases accuracy very slightly with a significant savings in CPU time. When the time step size is increased to 50 ms, significant errors in integrated core power begin to appear. These rather large errors, much too large to be acceptable, are readily apparent if one compares the step perturbation integrated power at time 0.05 s obtained using various size time steps. This error does in fact decrease to an acceptable level the further one goes from initial cross-section perturbation. From these results it is apparent that the optimal time step size for the TWIGL seed-blanket step perturbation should be smaller than was the case for the ramp perturbation.

In conclusion, several interesting features of a backwards differenced transient NEM coding have been discovered from running the simple TWIGL seed-blanket calculations with ramp and step cross-section perturbations. First one can limit the spatial discretization error and its effect on the error in total integrated power throughout the

transient calculation by setting an appropriate maximum assembly power error criterion on the initial steady-state eigenvalue calculation. Secondly, the choice of a time step size has little effect on the error in the normalized assembly power distribution later in the transient. Finally, it would be beneficial from an accuracy and computation time standpoint to use a variable time step size in a transient calculation, small steps near where relatively large cross-section changes occur and large steps where relatively minor perturbations occur. The size of this variable time step can be chosen based on some maximum allowable change in nodal power during a time step.

#### 5.2.3.2 Dodds' R-Z Transient Sample Problem

The second two-dimensional transient benchmark problem was a transient version of Dodds' R-Z geometry benchmark, initially described in Section 4.3.1 of Chapter 4. The spatially dependent transient for this configuration is the result of 1.0 second long ramp changes in the thermal absorption cross-section in problem regions 3, 7 , and 11. In regions 3 and 7 this ramp involves a total positive 3% cross-section change, while region 11 has a total negative 3% change. The perturbed thermal absorption cross-sections then retain their 1.0 second values for the duration of the 4 second transient.

Four transient NEM calculations were run on Dodds' R-Z problem. Two of these calculations used a spatial mesh distribution consisting of 12 radial mesh and 28 z mesh, while the other two used a coarser mesh distribution consisting of 6 radial mesh and 14 z mesh. These mesh distributions are the same as those described in Section 4.3.1 for the steady-state NEM calculations. Transient time step sizes of 0.5 and 0.05 seconds were then employed with each of these two mesh distributions. As described in Chapter 4, all Dodds' R-Z calculations used three 10 degree  $\theta$  mesh with reflected boundary conditions at the positive and negative extrema. Finally, all transient calculations used the more

accurate second approximation (Section 4.4.1.2) for calculating the second-order radial direction flux moments.

The results, total core power at selected times extending out to 4.0 seconds, of these four transient calculations appear in Table 5.8. Steady-state fluxes and precursor concentrations for these transient calculations were obtained from the steady-state Dodds' R-Z calculations performed in Chapter 4. If one compares the most accurate transient NEM calculation of Table 5.8, 12x3x28 mesh and a 0.05 second time step, to Dodds' reference solution, one observes that the NEM calculation yields total core powers which are slightly larger than the reference solution at any given time. This discrepancy becomes more and more pronounced as the transient continues, rising to an error of more than 8% at four seconds into the transient.

The second transient NEM calculation used the same 12x3x28 mesh with a factor of 10 larger time step, 0.5 seconds. The results of this calculation, shown in the third column of Table 5.8, are at all times larger than the results of both the reference calculation and the 0.05 second time step calculation. The results of the present calculation differ by as much as 4.5% from the previous NEM calculation. Clearly, in spite of the savings in CPU time, a 0.5 second time step is too large for this model, especially during the first 1.0 seconds where the ramp cross-section changes occur.

The third Dodds' R-Z problem transient calculation employs a coarse 6x3x14 mesh and a 0.05 second time step size. As shown in Table 5.8, this calculation gives almost the same results, a maximum difference of less than 1%, as the similar fine-mesh NEM calculation. In addition, these results are obtained at a savings of almost a factor of four in CPU time. Therefore, if one is only interested in calculating total core power, using a coarse 6x3x14 mesh will provide a significant gain in computational efficiency.

**Table 5.8 Reference, Fine and Coarse Mesh Transient Total Power vs. Time and No. of Outer Iterations for Dodds' R-Z Transient Problem with Various Time Step Sizes**

Time (sec)	Power (no. ofouters)					Ref
	12x3x28 Mesh 0.05 Sec	12x3x28 Mesh 0.5 Sec	6x3x14 Mesh 0.05 Sec	6x3x14 Mesh 0.5 Sec		
0.2	1.041 (36)	-	1.041 (23)	-		1.039
0.4	1.114 (37)	-	1.114 (23)	-		1.108
0.6	1.222 (38)	-	1.223 (27)	-		1.209
0.8	1.387 (36)	-	1.389 (29)	-		1.360
1.0	1.651 (43)	1.725 (117)	1.656 (32)	1.731 (54)		1.596
1.2	1.815 (16)	-	1.823 (15)	-		1.752
1.4	1.906 (12)	-	1.915 (12)	-		1.828
1.6	1.984 (10)	-	1.994 (12)	-		1.895
1.8	2.058 (10)	-	2.070 (12)	-		1.960
2.0	2.132 (10)	2.154 (21)	2.145 (12)	2.167 (25)		2.027
3.0	2.497 (11)	2.523 (23)	2.517 (12)	2.543 (21)		2.335
4.0	2.883 (11)	2.913 (20)	2.911 (12)	2.942 (20)		2.659
CPU time	865.7	211.8	203.9	41.71		-

The final transient NEM calculation used the 6x3x14 spatial mesh and a coarse 0.5 second time step. This calculation again overshoots the 0.05 second time step NEM calculation with a similar mesh distribution by up to 4.5%. These results reinforce the previous conclusion that a 0.5 second time step leads to unacceptable errors in calculated total power.

#### 5.2.4 Three-Dimensional Benchmarking

At this time the backwards-differenced transient version of the Nodal Expansion Method will be tested on two versions of a true three-dimensional transient benchmark problem. This test problem is the transient version of the 3-D LMW problem, on which steady-state calculations were performed in Section 4.2.2.3 of Chapter 4. In the transient LMW problem, the steady-state configuration is altered by the vertical motion of four control rod assemblies. At first, the initially partially inserted control assemblies at x-y plane positions 4,1 and 1,4 are withdrawn at a rate of 3 cm per second. Later, at 7.5 seconds into the transient, control rods at positions 1,1 and 3,3 are inserted into the core at 3 cm per second. At this rate the final control rod positions, shown in the third figure in Section B.3, Appendix B, will be reached at 47.5 seconds into the transient. After these final positions are reached, the transient condition will be followed for another 12.5 sec., for a total problem time of 60 seconds.

This transient calculation will be performed both with and without a thermal-hydraulic feedback mechanism. The cross-section feedback mechanism to be used consists of the highly simplified one-dimensional WIGL channel model initially developed by Westinghouse (39). This feedback model, and the means by which the model interacts with the steady-state and transient neutronics of the LMW problem will be detailed in upcoming subsections of this chapter.

#### 5.2.4.1 Three-Dimensional LMW Transient Benchmark Problem w/o Feedback

A total of five examples of the 3-D LMW transient benchmark problem were run without a thermal-hydraulic feedback mechanism. These backwards-differenced transient NEM calculations differ from one another only in axial mesh spacing and/or transient time step size.

The first of these transient NEM calculations used an x-y plane mesh spacing of 20 cm, an axial mesh spacing of 20 cm, and a transient time step size of 0.5 seconds. An edited version of the normalized total core power produced by this calculation is compared with a fine space-time meshed CUBBOX (26) reference solution, run by K. S. Smith (13), in Table 5.9. As shown in the table, the NEM power lags the reference solution at intermediate problem times by as much as 5.5%. Since the steady-state nodal power distribution for this particular mesh spacing differs from the reference by less than 0.6% (Table 4.13), the source of error in the transient solution is believed to be related to either the choice of time step size or the control rod motion algorithm.

K. S. Smith experienced errors of similar magnitude while running Analytic Nodal Method calculations employing this same (20 cm) mesh spacing (13). The source of error in both Smith's solutions and the present NEM calculations seems to stem from a phenomena known as control rod cusping. Control rod cusping results from the inaccurate means by which the absorption cross sections in cells with partially inserted control rods are calculated. Since both the Analytic Nodal Method and the NEM require constant nodal cross sections, the control rod motion algorithms simply set the absorption cross sections in the cells with partially inserted rods equal to a weighted average of the cross sections in fully controlled and fully uncontrolled cells. In both the present calculation and in Smith's, this weighting involves linearly interpolating between fully controlled and fully uncontrolled cross-sections based upon the fractional

**Table 5.9 Reference, Coarse and Fine Axial Mesh Transient Total Power vs. Time and No. of Outer Iterations for the 3-D LMW Transient Problem without Feedback for Various Time Step Sizes**

Power (no. of outers)				
Time (sec)	20x20x20 cm 0.5 Sec	20x20x10 cm 0.5 Sec	20x20x10 cm 2.5 Sec	Ref
5.0	1.121 (11)	1.125 (15)	1.127 (15)	1.129
10.0	1.316 (12)	1.336 (14)	1.340 (15)	1.347
20.0	1.661 (10)	1.694 (10)	1.695 (13)	1.737
30.0	1.322 (9)	1.356 (10)	1.352 (15)	1.399
40.0	0.7864 (10)	0.8014 (10)	0.8008 (14)	0.8260
50.0	0.4904 (7)	0.4983 (7)	0.4987 (11)	0.5100
60.0	0.3767 (6)	0.3826 (5)	0.3828 (9)	0.3907
CPU time	309.6	635.1	172.9	-

distance a control rod is inserted into a given cell. The controlled and uncontrolled cross sections should in fact be weighted by the actual node axial flux distribution. Weighting by insertion fraction simply means assuming a fictitious and inaccurate flat nodal flux distribution in the axial direction.

A second 0.5 sec. time step NEM calculation was run in an attempt to mitigate the errors which arise from control rod cusping. Here the axial mesh spacing was halved to 10 cm in the belief that a flat cross-section weighting function is more accurate over shorter distances. The results of this calculation appear as the second entry in Table 5.9. As shown in the table, reducing the axial mesh spacing definitely reduces the solution error at intermediate problem times. The maximum error in total core power has been lowered from 5.5% to 3.1%, at the expense of more than doubling the problem computation time.

A third NEM calculation was run to determine the effects of time step discretization error on the time-dependent total core power. Here a NEM calculation was run using a 10 cm axial mesh spacing and a 2.5 sec. time step. The results of this calculation appear as the third entry in Table 5.9. As shown in the table, this dramatic increase in time step size has little effect on the calculated total core power. This underscores the low level of time step discretization error that can be achieved in an obviously delayed super-critical transient when using the time-integrated approximation (28) for delayed neutron precursor production.

Next, a fourth NEM calculation was run to study the control rod cusping error reduction possible by a further decrease in axial mesh size. In this calculation the in-core axial mesh spacing was reduced from 10 cm to 5 cm, while the reflector axial mesh spacing remained at 10 cm. The normalized total core power resulting from this 0.5 sec. time step calculation appears as the first entry in Table 5.10. This further reduction in

axial mesh spacing diminishes control rod cusping to a point where the maximum error in total power is reduced to 2.4%. This slight increase in accuracy, however, comes with a penalty of an approximate factor of two increase in problem computation time as compared to the 10 cm axial mesh calculation.

A fifth and final NEM calculation was run to determine the effects of time step discretization error on the 5 cm axial mesh calculation. Here the time step size was again increased to 2.5 sec. The results of this calculation appear as the second entry in Table 5.10. As discovered previously with an axial mesh spacing of 10 cm, a five-fold in time step size has little effect on the 5 cm axial mesh problem. In addition, this calculation consumes about a factor of four less CPU time than the equivalent 0.5 sec. time step calculation, again highlighting the time domain efficiency gains possible with the time-integrated precursor approximation.

#### 5.2.4.2 Three-Dimensional LMW Transient Benchmark Problem w/ Thermal-Hydraulic Feedback

As previously discussed, a simplified WIGL (39) based thermal-hydraulic feedback mechanism for the transient LMW sample problem will be developed in this subsection. The WIGL model only accounts for the vertical time-dependent transport of energy within a non-boiling coolant. Effects due to fluid momentum transport, pressure variations, and boiling are ignored. The WIGL model accounts for three primary thermal-hydraulic properties within a given homogeneous node: average fuel temperature, average coolant temperature, and average coolant density. The time-dependent energy balances on the fuel and coolant temperatures within node (i,j,k) with constant coolant flow rate can be written as:

$$\frac{d\bar{T}_{i,j,k}^f}{dt} = \frac{1}{\rho_f C_f} \left( (1 - r) q'''_{i,j,k} - \frac{V_{i,j,k}^c A_h h_o U}{V_{i,j,k}^f h_o + U} (\bar{T}_{i,j,k}^f - \bar{T}_{i,j,k}^c) \right) \quad (5.12)$$

**Table 5.10 Reference and Very Fine Axial Mesh Transient Total Power vs. Time and No. of Outer Iterations for the 3-D LMW Transient Problem without Feedback for Various Time Step Sizes**

Time (sec)	Power (no. of outers)		
	20x20x5 cm 0.5 Sec	20x20x5 cm 2.5 Sec	Ref
5.0	1.127 (14)	1.129 (14)	1.129
10.0	1.339 (14)	1.343 (14)	1.347
20.0	1.705 (10)	1.705 (13)	1.737
30.0	1.366 (11)	1.363 (13)	1.399
40.0	0.8059 (11)	0.8058 (13)	0.8260
50.0	0.5008 (6)	0.5014 (10)	0.5100
60.0	0.3844 (5)	0.3848 (9)	0.3907
CPU time	1163	289.8	-

$$\frac{d\bar{T}_{i,j,k}^c}{dt} = \left( \frac{\partial \rho^c H}{\partial \bar{T}^c} \right)^{-1} \left( \frac{A_h h_o U}{h_o + U} (\bar{T}_{i,j,k}^f - \bar{T}_{i,j,k}^c) + \frac{2W_{i,j,k}^r C^c}{V_{i,j,k}^c} (T_{i,j,k}^b - \bar{T}_{i,j,k}^c) + r q_{i,j,k}''' \frac{V_{i,j,k}^f}{V_{i,j,k}^c} \right) \quad (5.13)$$

$$T_{i,j,k}^b = 2\bar{T}_{i,j,k-1}^c - T_{i,j,k-1}^b \quad (5.14)$$

where

$\bar{T}_{i,j,k}^f$  ≡ average fuel temperature in node (i,j,k)

$\bar{T}_{i,j,k}^c$  ≡ average coolant temperature in node (i,j,k)

$T_{i,j,k}^b$  ≡ inlet coolant temperature to node (i,j,k)

$\rho^f$  ≡ fuel density

$\rho^c$  ≡ coolant density

$C^f$  ≡ specific heat of fuel

$C^c$  ≡ specific heat of coolant

$r$  ≡ fraction of fission power deposited directly into coolant

$q_{i,j,k}'''$  ≡ volumetric heat generation rate within node (i,j,k)

$V_{i,j,k}^f$  ≡ fuel volume in node (i,j,k)

$V_{i,j,k}^c$  ≡ coolant volume in node (i,j,k)

$A_h$  ≡ heat transfer area per unit coolant volume

$h_o$  ≡ convective heat transfer coefficient

$W_o$  ≡ total mass flow rate of coolant into core

$\frac{\partial \rho^c H}{\partial \bar{T}^c}$  ≡ energy required to raise the temperature of a unit  
volume of coolant by one temperature unit

$U$  ≡ inverse of resistance to heat flow through fuel, gap, and clad

$W_{i,j,k}^r$  ≡ coolant mass flow rate through node (i,j,k)

In order to simplify the thermal solution process, the time derivatives in Equations 5.12 and 5.13 are expressed by backwards difference approximations. These fully implicit temperature equations are then solved using the same temporal mesh as the neutronic equations. The TRAC-PF1 subroutine RHOLIQ (11) was used to evaluate subcooled coolant density at the calculated coolant temperature.

The cross-section feedback resulting from these calculated fuel and coolant thermal states is accomplished by a simple linear feedback model. This cross-section feedback model can be expressed as:

$$\Sigma_{\alpha i,j,k} = \Sigma_{\alpha i,j,k}^r + \left( \frac{\partial \Sigma_\alpha}{\partial \bar{T}^c} \right) (\bar{T}_{i,j,k}^c - \bar{T}^{c,r}) + \left( \frac{\partial \Sigma_\alpha}{\partial \bar{T}^f} \right) \cdot (\bar{T}_{i,j,k}^f - \bar{T}^{f,r}) + \left( \frac{\partial \Sigma_\alpha}{\partial \rho^c} \right) (\rho_{i,j,k}^c - \rho^r) \quad (5.15)$$

where

$\alpha \equiv$  any two-group macroscopic cross section

$\bar{T}^{c,r}, \bar{T}^{f,r}, \rho^r \equiv$  reference conditions

For the purposes of this benchmark calculation, this linear feedback model is assumed to be valid over the entire range of thermal-hydraulic conditions.

Steady-state thermal-hydraulic conditions are calculated from Equations 5.12 and 5.13 by setting the time derivatives to zero. Several iterations between the steady-state neutronic and thermal-hydraulic routines are usually necessary to achieve consistency between the calculated temperatures and densities and the nodal fission power levels. During the transient time steps the neutronic and thermal-hydraulic solutions are decoupled by using the previous time step thermal solution as the feedback mechanism in the present time step.

A total of four coupled NEM/WIGL calculations were run on the 3-D LMW problem. All four calculations used the representative values for the WIGL thermal-hydraulic parameters given in Section B.3 of Appendix B. The first NEM calculation employed an axial mesh spacing of 20 cm and a 0.5 second transient time step. The time-dependent total normalized core power resulting from this calculation appear in the first column of Table 5.11. The reference solution to the 3-D transient LMW problem with feedback was taken to be a 20 cm axial mesh QUANDRY (13) calculation with a 0.25 second time step. This calculation, unlike the CUBBOX (26) reference solution for the LMW problem without feedback, contained no correction for the previously discussed control rod cusping problem.

The 20 cm axial mesh NEM calculation required six steady-state iterations between the neutronic and thermal-hydraulic routines in order to achieve a consistent coupled solution. In the process, the system eigenvalue dropped from 0.9996 to 0.9847, indicative of the feedback reactivity generated in taking a generic PWR from a hot standby condition (zero power) to full power. The time-dependent normalized core powers generated by this NEM calculation are very near those of the reference solution, a maximum error of 1.6% occurring at the end of the transient. Upon comparing the results of this calculation to those of similar calculations without feedback, Tables 5.9 and 5.10, one can see that cross-section feedback acts to significantly dampen the power oscillation resulting from time-dependent control rod motion.

In the second transient NEM calculation with feedback, the axial mesh spacing was reduced to 10 cm. The results of this 0.5 second time step calculation appear in the second column of Table 5.11. As shown in the table, the results of this fine axial mesh calculation are well within 1% of those obtained with 20 cm axial mesh. This trivial

**Table 5.11 Reference, Coarse and Fine Axial Mesh Transient Total Power vs. Time and No. of Outer Iterations for the 3-D LMW Transient Problem with WIGL Thermal-Hydraulic Feedback for Various Time Step Sizes**

Power (no. of outers)					
	20x20x20 cm 0.5 Sec	20x20x10 cm 0.5 Sec	20x20x20 cm 0.25 Sec	20x20x20 cm 1.0 Sec	Ref
K Init	0.984735	0.984684	0.984735	0.984735	0.98355
Time (sec)					
2.5	1.025 (12)	1.027 (15)	1.024 (10)	-	1.024
5.0	1.041 (12)	1.040 (14)	1.039 (10)	1.043 (14)	1.037
7.5	1.051 (13)	1.052 (13)	1.050 (13)	-	1.048
10.0	1.055 (13)	1.060 (13)	1.054 (10)	1.056 (16)	1.051
12.5	1.063 (13)	1.062 (13)	1.062 (11)	-	1.059
15.0	1.058 (13)	1.061 (12)	1.058 (12)	1.057 (10)	1.054
17.5	1.051 (12)	1.057 (12)	1.051 (12)	-	1.049
20.0	1.049 (8)	1.046 (9)	1.049 (8)	1.048 (9)	1.047
22.5	1.026 (12)	1.030 (9)	1.027 (10)	-	1.023
25.0	1.010 (12)	1.012 (9)	1.011 (10)	1.007 (14)	1.010
27.5	0.9954 (12)	0.9930 (11)	0.9966 (12)	-	0.9962
30.0	0.9655 (11)	0.9685 (11)	0.9675 (11)	0.9614 (11)	0.9686
32.5	0.9443 (11)	0.9443 (11)	0.9462 (11)	-	0.9486
35.0	0.9208 (9)	0.9207 (9)	0.9229 (9)	0.9166 (11)	0.9286
37.5	0.8961 (11)	0.8976 (11)	0.8983 (11)	-	0.9048
40.0	0.8743 (11)	0.8744 (9)	0.8764 (11)	0.8701 (11)	0.8831
42.5	0.8547 (11)	0.8539 (11)	0.8567 (9)	-	0.8658
45.0	0.8363 (11)	0.8359 (9)	0.8382 (9)	0.8326 (12)	0.8485
47.5	0.8185 (11)	0.8202 (9)	0.8204 (9)	-	0.8317
50.0	0.8324 (8)	0.8335 (9)	0.8328 (7)	0.8319 (9)	0.8458
52.5	0.8336 (6)	0.8344 (5)	0.8338 (4)	-	0.8458
55.0	0.8326 (5)	0.8334 (5)	0.8328 (4)	0.8322 (6)	0.8458
57.5	0.8316 (5)	0.8324 (5)	0.8318 (4)	-	0.8447
60.0	0.8306 (5)	0.8315 (4)	0.8308 (4)	0.8302 (6)	0.8442
CPU time	338.1	669.0	586.8	200.0	-

increase in accuracy can in no way overcome the approximately 100% computation time penalty associated with this calculation.

The third NEM transient calculation used a 20 cm axial mesh spacing and a 0.25 second time step. The results of this calculation are contained in the third column of Table 5.11. As can be seen in the table, this calculation gives almost exactly the same answers as those of the original 0.5 second time step NEM calculation. These results, however, require about 75% more CPU time to obtain than do those of the 0.5 sec. calculation.

The final transient NEM calculation with the WIGL thermal-hydraulic model uses a 20 cm axial mesh spacing and a 1.0 second time step. The results of this calculation appear in the fourth column of Table 5.11. Again, the table shows very good agreement between the results of this calculation and those of the 0.5 sec. time step NEM calculation and thus K. S. Smith's reference QUANDRY calculation. In addition, these results are obtained using only about 40% of the CPU time required by the 0.5 sec. time step calculation. The results of these calculations with various size time steps re-emphasizes the temporal accuracy inherent in the time-integrated neutron precursor production approximation.

The use of time step sizes in this transient of larger than approximately 1 second will most likely yield results which deviate unacceptably from the reference solution. These errors are primarily the result of a breakdown in the decoupled means by which transient cross-section feedback is calculated. The results obtained using large time steps could be made much more accurate if the core temperature and fission power solutions were iterated upon at any given time step. But this coupled solution methodology, because of its iterative nature, might require more computation time than a similarly accurate decoupled solution with a much smaller time step.

### 5.3 The Exponential Time-Differencing Method

#### 5.3.1 Derivation of the Time-Dependent Numerics

The derivation of the exponential time-differenced transient version of the Nodal Expansion Method originates from the node 1, two-energy-group neutron diffusion and precursor equations (Equations 5.1, 5.2, and 5.3). But, in contrast to the previous backward-differencing scheme, the space and time dependent neutron flux variable,  $\phi_g^l(r, t)$ , is exchanged for a variable with much weaker time dependence,  $\psi_g^l(r, t)$ . The relationship between  $\phi_g^l(r, t)$  and  $\psi_g^l(r, t)$  can be expressed as:

$$\phi_g^l(r, t) = e^{\Omega^l(t - t^{old})} \psi_g^l(r, t) \quad (5.16)$$

where

$t^{old} \equiv$  time at beginning of discrete time step

$\Omega^l \equiv$  node dependent frequencies

This transformation attempts to separate out the dominant mode exponential time dependency from the transient flux distribution. As a result, with the proper choice of frequencies, the function  $\psi_g^l(r, t)$  should vary much more slowly with time than  $\phi_g^l(r, t)$ . Thus the truncation error developed in time differencing  $\psi_g^l(r, t)$  should be much less than that developed in time differencing  $\phi_g^l(r, t)$ .

Employing this transformation, Equations 5.1 through 5.3 may be rewritten as:

$$\begin{aligned} \frac{1}{V_1} \frac{\partial}{\partial t} \psi_1^l(r, t) - D_1^l \nabla^2 \psi_1^l(r, t) + \left( \frac{\Omega^l}{V_1} + \Sigma_{a1}^l + \Sigma_{12}^l \right. \\ \left. - (1 - \beta) \nu \Sigma_{f1}^l \right) \psi_1^l(r, t) = (1 - \beta) \nu \Sigma_{f2}^l \psi_2^l(r, t) \\ + e^{\Omega^l(t - t^{old})} \sum_{d=1}^D \lambda_d C_d^l(r, t) \end{aligned} \quad (5.17)$$

$$\frac{1}{V_2} \frac{\partial}{\partial t} \psi_2^l(r, t) - D_2^l \nabla^2 \psi_2^l(r, t) + \left( \frac{\Omega^l}{V_2} + \Sigma_{a2}^l \right) \psi_2^l(r, t) = \Sigma_{12}^l \psi_1^l(r, t) \quad (5.18)$$

$$\begin{aligned} \frac{\partial}{\partial t} C_d^l(r, t) &= e^{\Omega^l(t-t^{old})} \beta_d (\nu \Sigma_{f1}^l \psi_1^l(r, t) \\ &\quad + \nu \Sigma_{f2}^l \psi_2^l(r, t)) - \lambda_d C_d^l(r, t), d = 1, D \end{aligned} \quad (5.19)$$

Where, all coefficients and variables have been previously described. In accordance with Buckner and Stewart's proscription (29), the time derivatives of  $\psi$  in Eqn's 5.17 and 5.18 are approximated by a first-order finite-difference expression:

$$\frac{\partial}{\partial t} \psi_g^l(r, t) = \frac{\psi_g^l(r) - \psi_g^{l,old}(r)}{\Delta t} \quad (5.20)$$

where

$\psi_g^{l,old}(r) \equiv$  spatially dependent  $\psi$ -function at time  $t^{old}$

$\psi_g^l(r) \equiv$  spatially dependent  $\psi$ -function at time  $t^{new}$

$$\Delta t \equiv t^{new} - t^{old}$$

One must develop an expression for the delayed neutron source contribution to Equation 5.17. Here, Buckner and Stewart (29) simply performed a backwards differencing operation on Equation 5.19. Backwards time-differencing Equation 5.19 results in:

$$\begin{aligned} C_d^l(r, t^{new}) &= \frac{1}{1 + \Delta t \lambda_d} \left( C_d^l(r, t^{old}) + \beta_d \Delta t e^{-\Omega^l \Delta t} \right. \\ &\quad \left. \cdot (\nu \Sigma_{f1}^l \psi_1^l(r, t^{new}) + \nu \Sigma_{f2}^l \psi_2^l(r, t^{new})) \right) \end{aligned} \quad (5.21)$$

where

$C_d^{l,old}(r) \equiv$  spatially dependent group d precursor concentration at time  $t^{old}$

$C_d^l(r) \equiv$  spatially dependent group d precursor concentration at time  $t^{new}$

When Equations 5.20 and 5.21 are substituted into Equations 5.17 and 5.18, equations of the following form result:

$$-D_g^l \nabla^2 \psi_g^l(r) + A_g^l \psi_g^l(r) = Q_g^l(r), g = 1, 2 \quad (5.22)$$

where

$$\begin{aligned} A_1^l &= \Sigma_{a1}^l + \Sigma_{12}^l + \frac{1}{V_1 \Delta t} + \frac{\Omega^l}{V_1} - (1 - \beta) \nu \Sigma_{f1}^l + \Delta t \nu \Sigma_{f1}^l \sum_{d=1}^D \frac{\beta_d \lambda_d}{1 + \lambda_d \Delta t} \\ A_2^l &= \Sigma_{a2}^l + \frac{1}{V_2 \Delta t} + \frac{\Omega^l}{V_1} \\ Q_1^l(r) &= \frac{\psi_1^{l,old}(r)}{V_1 \Delta t} + (1 - \beta) \nu \Sigma_{f2}^l \psi_2^l(r) \\ &\quad + \sum_{d=1}^D \frac{\lambda_d}{1 + \lambda_d \Delta t} \left( C_d^{l,old}(r) e^{-\Omega^l \Delta t} + \beta_d \Delta t \nu \Sigma_{f2}^l \psi_g^l(r) \right) \\ Q_2^l(r) &= \frac{\psi_2^{l,old}(r)}{V_2 \Delta t} + \Sigma_{12}^l \psi_1^l(r) \end{aligned}$$

Equations 5.22, like the previous transient Equations 5.8, are simply fixed source versions of the steady-state eigenvalue equations (Eqn's 3.2 or 3.73). Therefore the function  $\psi_g^l(r)$  can be solved for using the Nodal Expansion Method spatial approximations described in Chapter 3. The actual transient two-group neutron flux distribution is obtained at the end of each time step from the calculated  $\psi_g^l(r)$  distribution through the use of the exponential transformation described by Equation 5.16. Updated values for the spatial distribution of  $\Omega^l$ 's or frequencies, initially set to zero, are best obtained from the past history of the spatially dependent thermal flux levels. New values for the frequencies can be expressed as:

$$\Omega^{l,j+1} = \frac{1}{\Delta t_j} \log \left( \frac{\bar{\phi}_2^{l,j}}{\bar{\phi}_2^{l,j-1}} \right) \quad (5.23)$$

where

$j \equiv$  time step indicator

As was the case with the previously described backwards differenced transient NEM methodology, the exponential method also requires the existence of both node average group d precursor concentrations ( $\bar{C}_d^l$ ) and group d precursor concentration moments ( $\bar{C}_{d(x,y,z)(1,2)}^l$ ).

### 5.3.2 Implementation and Acceleration of Computational Algorithm

The exponential time-differencing method employs the exact same solution methodology as did the previously described fully implicit method. That is, a fixed source type calculation is performed to calculate the  $\psi_g^l(r)$  distribution at each time step using an inner/outer iteration strategy. The equation which defines the coarse-mesh rebalancing factors is, however, slightly different from that used previously. The equation for these factors can now be expressed in Cartesian geometry as:

$$\begin{aligned}
 & -f^l \sum_{l \in S_m} \Delta^2(x, y, z) \sum_{g=1}^2 J_{g(x,y,z)}^{in,l} + f^m \sum_{m \in S_m} \Delta^2(x, y, z) \sum_{g=1}^2 J_{g(x,y,z)}^{out,m} + f^m \\
 & \sum_{m \in V_m} \Delta x \Delta y \Delta z \sum_{g=1}^2 \left( \Sigma_{ag}^m - (1 - \beta) \nu \Sigma_{fg}^m + \frac{1}{V_g \Delta t} + \frac{\Omega^m}{V_g} - \sum_{d=1}^D \frac{\beta_d \lambda_d \Delta t \nu \Sigma_{fg}^m}{1 - \lambda_d \Delta t} \right) \bar{\psi}_g^m \\
 & = \sum_{m \in V_m} \Delta x \Delta y \Delta z \sum_{g=1}^2 \frac{1}{V_g \Delta t} \bar{\psi}_g^{m,old} + \sum_{m \in V_m} \Delta x \Delta y \Delta z \sum_{d=1}^D \frac{\lambda_d}{1 + \lambda_d \Delta t} \bar{C}_d^{m,old} e^{-\Omega^m \Delta t}
 \end{aligned} \tag{5.24}$$

If equations like Equation 5.24 are written for all nodes in the system and combined in matrix format, a fixed source equation of the same form as Equation 5.11 is produced. As before, this matrix equation can be solved by a single pass through a standard Gaussian Elimination routine.

### 5.3.3 Benchmarking Calculations

The exponential method was first tested on the two-dimensional TWIGL Seed-Blanket transient benchmark problems previously described in Subsection 5.2.3.1. Both

ramp and step perturbed calculations were run using various size time steps. All 2-D TWIGL benchmark calculations employed the previously described 6x6x3 three-dimensional coarse-mesh spatial discretization.

The total power calculated by the NEM exponential method for the ramp perturbed TWIGL problem using various size time steps appears in Table 5.12. Upon comparing the results of the 1 ms time step calculation with those of the similar calculation in Table 5.6, one can see that the exponential method and the original fully implicit method yield the exact same results using about the same amount of computation time. However, when the time step size is raised to 10 ms, the results of the exponential method began to diverge slightly from those obtained using the fully implicit method, especially during the earlier portions of the transient. Finally, when the time step size is raised to 50 ms, the exponential method is shown to yield highly erroneous results, differing from the reference solution by as much as 7.5%. The reasons for this inaccuracy at large time step sizes are unclear. The most likely explanation is the possibly large truncation error inherent in the backwards differencing approximation used to determine the time-dependent precursor concentrations.

The same series of exponential method calculations were then performed for the TWIGL Seed-Blanket step perturbed problem. The results of this series of coarse-mesh NEM calculations appears in Table 5.13. As was the case with the ramp calculation, this table shows that the exponential method gives essentially the same results as the original fully implicit method (Table 5.7) when a time step size of 1 ms is used. When a time step size of 10 ms is employed, Table 5.13 actually shows a slight improvement in calculational accuracy over the fully implicit method. Just as before, the performance of the exponential method degrades significantly when larger time step sizes are used.

**Table 5.12 Reference, 1, 10, and 50 ms Time Step Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Ramp Transient Benchmark Problem**

Power (no. of outers)				
Time (sec)	1ms	10ms	50ms	Ref
0.05	1.125 (6)	1.124 (12)	1.128 (15)	1.125
0.1	1.308 (6)	1.313 (13)	1.332 (15)	1.307
0.2	1.959 (7)	1.966 (11)	2.041 (15)	1.957
0.3	2.074 (4)	2.071 (5)	2.103 (9)	2.074
0.4	2.091 (4)	2.092 (4)	2.169 (12)	2.096
0.5	2.108 (4)	2.109 (4)	2.269 (14)	2.109
CPU time	173.3	28.02	11.15	-

With a time step size of 50 ms, the exponential method is in error at times by more than 100%. Again, specific reasons for these large errors are generally unclear.

Further attempts were made to calculate the time-dependent core power distribution in the 3-D LMW Problem Without Feedback using the exponential method. However, all of these attempts failed, producing very large errors even with a rather small 0.5 second time step size. As a result, Buckner and Stewart's exponential method (29) can by no means compare in temporal accuracy with the previously described fully implicit method with time-integrated precursors (28).

#### 5.4 Conclusions

In this chapter two transient solution techniques have been derived and tested. The first technique involves using a simple backwards difference approximation for the flux derivatives and the time-integrated precursor production approximation (28). The second technique, known as the exponential method, first transforms the time-dependent flux into a new variable whose temporal dependence is assumed to be much weaker. After performing this transformation, which factors out the spatially dependent exponential time dependence, a fully implicit first-order finite-difference approximation is used to discretize both the transformed flux equations and the neutron precursor equations. In both methods, the flux distribution at each time step is iteratively solved for using a fixed source version of the Nodal Expansion Method.

A variety of two- and three-dimensional benchmark calculations were performed using one or both of these methods. The results of these calculations clearly show that the fully implicit/time-integrated precursor method is superior to the exponential method. For the initial 2-D TWIGL Seed-Blanket calculations, this method yielded very accurate results, even for large time step sizes. In contrast, the exponential method

**Table 5.13 Reference, 1, 10, and 50 ms Time Step Fourth-Order Transient Total Power Vs. Time for the TWIGL Seed-Blanket Two-Dimensional Step Transient Benchmark Problem**

Power (no. of outers)					
Time (sec)	1ms	10ms	50ms	Ref	
0.05	2.037 (5)	1.910 (7)	1.875 (15)	-	
0.1	2.060 (4)	2.062 (7)	2.568 (16)	2.061	
0.2	2.077 (4)	2.078 (4)	3.121 (17)	2.078	
0.3	2.095 (4)	2.095 (4)	3.844 (18)	2.095	
0.4	2.112 (4)	2.113 (4)	4.207 (18)	2.113	
0.5	2.130 (4)	2.130 (4)	4.335 (18)	2.131	
CPU time	148.8	22.78	14.51	-	

gave large power errors when large time step sizes were used. In addition, the fully implicit/time-integrated precursor method performed admirably when used to calculate the transient flux distributions in Dodds' R-Z problem and the 3-D LMW benchmark problem both with and without thermal-hydraulic cross-section feedback.

## CHAPTER 6

### INTEGRATING THE NODAL EXPANSION METHOD INTO TRAC-PF1/MOD2

#### 6.1 Introduction

Up until this point the Nodal Expansion Method has only been run as a stand-alone neutronics package or in conjunction with the highly simplified WIGL(39) thermal-hydraulic feedback model. As a result of its successful stand-alone implementation, the NEM-based multidimensional neutronics coding can now be combined with the TRAC-PF1/MOD2 modular thermal-hydraulics package.

The degree of success obtained in integrating this multidimensional neutronics package into TRAC-PF1/MOD2 will be measured by running a coupled neutronic/thermal-hydraulic sample problem. This problem consists of a single control rod withdrawal incident in a Westinghouse pressurized water reactor initially at full power. This quarter-core-symmetric, core-region-only, coupled neutronic/ thermal-hydraulic calculation, known here as the S. Levy test problem, was initially developed by personnel at S. Levy Incorporated to test the capabilities of the EPRI-ARROTTA accident evaluation code (40).

This test calculation will be performed using problem specific NEM/TRAC coding. That is, no attempt will be made at this time to develop a fully tested version of TRAC-PF1/MOD2 coupled to a generally applicable 3-D neutron kinetics capability. Providing this capability would require the development of a general neutron cross-section feedback model, a general control rod position model, in addition to TRAC consistent array storage, input, output, and dump/restart files. The additional effort

required to provide TRAC-PF1/MOD2 with a general 3-D neutron kinetics capability is beyond the scope of this effort.

In the next subsection, the features of the EPRI-ARROTTA transient analysis code will be described. Following this, there will appear a brief description of the means by which the 3-D transient NEM neutronics coding was integrated into TRAC-PF1/MOD2. Lastly, the final sections of this chapter will present the results of running the Westinghouse PWR transient test problem with both EPRI-ARROTTA and NEM/TRAC.

## 6.2 Features of the EPRI-ARROTTA Code

The EPRI-ARROTTA code was developed for the Electric Power Research Institute (EPRI) by S. Levy Incorporated in 1985 (40). The in-core thermal-hydraulic portion of EPRI-ARROTTA was obtained from the BEAGL-02 coupled neutronic/ thermal-hydraulic code previously developed by Brookhaven National Laboratory (41). The multidimensional transient neutronics package used in EPRI-ARROTTA was the three-dimensional, Cartesian geometry, Analytic Nodal Method developed by K. S. Smith (13).

The methodology employed in the EPRI-ARROTTA code has been proven accurate by comparison both to experimental data and to the results of the Nuclear Regulatory Commission licensed coupled neutronic/ thermal-hydraulic code HERMITE (42). Specifically, EPRI-ARROTTA and HERMITE were both used to calculate the transient response of a one-half-core symmetric PWR at stand-by (hot zero power) to a control rod ejection incident (43). The results obtained by running the two space-time kinetics codes were shown to be in good agreement throughout the extent of the transient. Since this simulated PWR transient is in many ways very similar to the forthcoming S. Levy transient test problem, EPRI-ARROTTA's capability to perform well during

the latter control rod ejection accident can be inferred from its close comparison with HERMITE.

The intent of S. Levy Incorporated in developing the EPRI-ARROTTA code was to create a fast running code capable of simulating control rod related reactivity transients which might occur during normal light water reactor operations. Simulations of these types of transients must be run as part of the Nuclear Regulatory Commission pre-startup licensing procedure. As a result of the relatively limited scope of EPRI-ARROTTA's intended usage, certain thermal-hydraulic model simplifications were made to enhance computational efficiency. The details of this model are compared to those of the general thermal-hydraulic model of TRAC-PF1/MOD2 in the following paragraphs.

The BEAGL-02 and EPRI-ARROTTA two-phase fluid flow model can be described as a four-equation model. These four equations include; a mixture mass conservation equation, two energy balance equations for the liquid and vapor phases, and an algebraic slip relationship. Momentum effects, as represented by the liquid and vapor phase momentum balance equations, are neglected in this model. As a consequence EPRI-ARROTTA is only permitted to model non-recirculating parallel channel flow within the core of boiling or pressurized water reactors, a scenario where the effects of the fluid's momentum are usually small. In this model the core inlet mass flow rate must be specified as a boundary condition for each fuel channel.

If this four equation model were used to simulate general PWR transients, it would suffer from several limitations. First, this model cannot account for coolant cross flow between assemblies. Cross flow, while negligible under most normal reactor conditions, becomes significant when the power produced in one channel remains much greater than that produced in one or more of its neighbors. Secondly, this four equation model does not take into account the effects of gravitational head on the primary coolant, a small

but sometimes non-negligible effect in a 12 foot tall PWR core, especially at low flow rates with two-phase fluid. Lastly, the BEAGL-02/ EPRI-ARROTTA thermal-hydraulic model cannot account for the effects of fluid acceleration. Fluid acceleration effects can become significant during a depressurization transient which involves significant instantaneous boiling. Here the coolant velocity increase which should occur above a region of rapidly expanding coolant is in actuality damped by fluid inertia. The EPRI-ARROTTA model will ignore this effect by forcing the mass flow rate to remain constant within a channel. As a result of these model limitations, EPRI-ARROTTA would be best at modelling limited hydrodynamic in-core transients initiated from a relatively high flow rate. These are exactly the modelling scenarios for which EPRI-ARROTTA was designed.

In order to transport heat from its source as volumetric fission within the solid (usually UO<sub>2</sub>) fuel pellets, to the exterior of the fuel rod cladding, which is in direct contact with the flowing coolant, a transient heat-conduction model must be employed. EPRI-ARROTTA employs an analytic radial heat conduction model for this purpose. Here the general radial direction transient heat-conduction equation is solved analytically in the fuel pellet, gas gap, and cladding. The solution of these equations yields expressions for the time-dependent average fuel temperature, and more importantly, the external cladding temperature, which in turn is coupled to the bulk coolant conditions by a variety of two-phase flow convection correlations.

One possible limitation of this conduction model is that heat flow axially within the fuel rods is ignored. Axial heat conduction, while negligible during the transients EPRI-ARROTTA was designed to simulate, may become significant during depressurization transients. These types of transients sometimes result in a clear axial division between

vapor and liquid regions. Such a situation is likely to occur when emergency core cooling systems are reflooding a core following a large-break loss-of-coolant accident.

A more serious drawback of the EPRI-ARROTTA analytical heat conduction model is the fact that it assumes spatially constant thermal properties within the fuel pellet, gas gap, and cladding. These thermal properties are chosen as a function of the average temperature in each of the three regions. This approximation can lead to an inaccurate fuel rod radial temperature profile, especially within the fuel pellet. Within a typical operating PWR, the temperature difference found in moving radially from the pellet centerline to the pellet edge may be as much as 600 K. Over this wide temperature range the thermal conductivity of uranium dioxide may in fact vary by as much as a factor of two. Using a single value of pellet thermal conductivity will result in some error in the pellet radial temperature profile.

Finally, EPRI-ARROTTA employs a simple linear model for calculating the fuel pellet dimensional variation associated with thermal expansion and contraction. Again this model is based on the average temperature within the pellet.

EPRI-ARROTTA employs another BEAGL-02 model for providing cross-section thermal-hydraulic feedback to its three-dimensional QUANDRY-based (13) transient neutronics. This model consists of three primary feedback mechanisms and one secondary feedback mechanism. The primary feedback mechanisms are based upon variations in moderator density, moderator temperature, and fuel temperature, while the secondary mechanism is based upon variations in coolant soluble-boron concentration. This cross-section feedback mechanism can be expressed as:

$$\begin{aligned}\Sigma = & R \cdot \left( A + B \cdot (X - 1) + C \cdot (X - 1)^2 \right) \\ & + (1 - R) \cdot \left( A' + B' \cdot (X - 1) + C' \cdot (X - 1)^2 \right)\end{aligned}$$

$$\begin{aligned}
 & + D \cdot (T_m - T'_m) \\
 & + E \cdot (\sqrt{T_f} - \sqrt{T'_f}) \\
 & + \sigma_b \cdot \mu \cdot ppm \cdot 1.102 \times 10^{-8} \cdot V_f
 \end{aligned} \tag{6.1}$$

where

$\Sigma \equiv$  any two-group macroscopic cross section or diffusion coefficient

$R \equiv$  node control fraction or fraction by volume of control

rods which are inserted into a node

$A, B, C \equiv$  zeroth, first, and second order coefficients which express the variation in macroscopic cross section as a function of cell averaged moderator density for a completely controlled cell

$A', B', C' \equiv$  same as above except for a completely uncontrolled cell

$X = \frac{\rho}{\rho'} \equiv$  actual cell-averaged moderator density  $\left( \frac{lb}{ft^3} \right)$   
divided by a reference moderator density  $\left( \frac{lb}{ft^3} \right)$

$D \equiv$  the change in macroscopic cross-section resulting from  
a unit change in cell-averaged moderator temperature

$T_m \equiv$  cell averaged moderator temperature ( $^{\circ}F$ )

$T'_m \equiv$  reference cell averaged moderator temperature ( $^{\circ}F$ )

$E \equiv$  the change in macroscopic cross-section resulting from a unit  
change in the square root of the absolute fuel temperature

$T_f \equiv$  cell averaged fuel temperature ( $^{\circ}F$ )

$T'_f \equiv$  reference cell averaged fuel temperature ( $^{\circ}F$ )

$\sigma_b \equiv$  boron-10 microscopic cross section (*barns*)

$\mu \equiv$  cell-averaged moderator density ( $\frac{g}{cm^3}$ )

$ppm \equiv$  coolant soluble boron concentration ( parts per million by weight )

$V_f \equiv$  coolant volume fraction in the cell

Cross-section feedback based upon the coolant soluble boron concentration is said to be a secondary mechanism since its short-term time variation is the result of changes in moderator density but not from a direct change in the moderator boron concentration.

Each of the four feedback-mechanism parameters shown in Equation 6.1 do not apply to all seven cell-specific, two-group diffusion-theory parameters. For instance, while the moderator density and temperature feedback mechanisms are assumed to apply to all seven parameters, the fuel temperature based feedback mechanism applies to only the fast group parameters. In addition, the coolant soluble boron feedback parameter only affects the fast and thermal absorption and the fast to thermal downscatter cross sections.

Finally, EPRI-ARROTTA employs a mechanism to track the time dependent movement of individual control rods or control-rod banks. This model simply determines the axial position of the tips of all control rods at the end of any discrete transient time step. If the tip of one or more rods is determined to lie within a homogeneous cell, the control fraction (R) for that cell is calculated from a simple algebraic relationship based upon the distance the rod is inserted into the cell. The relationship between rod insertion distance and control fraction is usually chosen to be linear. That is, the control fraction is assumed to be equal to the fraction of the total distance which the control rod is inserted into a given cell.

EPRI-ARROTTA uses an explicit coupling mechanism to determine the multidimensional core neutronic and thermal-hydraulic solution during a transient. That is,

assuming that the core neutronic and thermal-hydraulic conditions are known at the end of time step N, one must follow a specified once-through procedure to obtain the new conditions at the end of time step N+1. First, if applicable, all control rods are moved to their end of time step positions. Based upon these final positions, the control fractions (R) within all homogeneous cells are calculated. Next, new macroscopic node cross sections are calculated using Equation 6.1. The core thermal-hydraulic conditions employed in Equation 6.1 are those obtained at the end of time step N. The newly calculated macroscopic cross-sections are then passed to the transient Analytic Nodal Method neutronic routines where the step N+1 nodal power densities are calculated. Finally, this new distribution of nodal power densities is transferred to the thermal-hydraulic routines to calculate all step N+1 thermal-hydraulic quantities. At this point in time, step N+1 is considered to be complete.

If implicit rather than explicit coupling was employed, one would use the latest step N+1 thermal-hydraulic properties to re-calculate the step N+1 cross sections. These cross sections would then be used to re-evaluate the step N+1 nodal power densities. This iterative process would continue until the change in all step N+1 quantities fell below some specified criterion. Under most circumstances this time consuming iterative process is unnecessary. If one chooses a reasonably small time step, the single iteration explicit coupling process will provide results indistinguishable from those obtained by the multi-iteration implicit process. This observation was proven in the previous chapter for the 3-D LMW transient problem with WIGL-based thermal-hydraulic feedback.

In the case of a steady-state neutronic/ thermal-hydraulic analysis, which must be performed prior to any transient calculation to provide initial conditions, the neutronic and thermal-hydraulic routines must be implicitly coupled. That is, in order to achieve consistency between the neutronic and thermal-hydraulic solutions, one must at time

zero iterate several times between the cross-section feedback routine, the steady-state multidimensional neutronic routines and the steady-state thermal-hydraulic routines.

### 6.3 Combined NEM/TRAC coding

This sub-section will provide the details of the means by which the previously described transient NEM coding was coupled with the TRAC-PF1/MOD2 thermal-hydraulic routines. In coupling the NEM neutronics with TRAC-PF1/MOD2, care was taken to ensure that this new coding performed similarly to the previously established EPRI-ARROTTA methodology. This approach will lessen the reasons for discrepancy between NEM/TRAC and EPRI-ARROTTA results in comparison calculations.

As previously detailed in Chapter 1, TRAC-PF1/MOD2 uses a full six equation model to approximate the conditions of two-phase flow. This model, because of its versatility, is superior to the four-equation thermal-hydraulic model used in EPRI-ARROTTA. If limited to calculating fluid conditions in a reactor core with prescribed inlet conditions, TRAC's six equation model and EPRI-ARROTTA's four equation model should yield similar results. The results obtained by these two methods will differ only during transients initiated from very low levels of coolant flow.

TRAC-PF1/MOD2 employs an alternative methodology for solving the heat conduction equation within the confines of the fuel rods. Rather than solve the radial heat conduction equation analytically, TRAC uses a finite difference technique to approximate the solution to the two-dimensional (axial and radial direction) fuel-rod heat-conduction equation. While a finite-difference solution to the heat-conduction equation only approaches the analytic solution in the limit of very fine mesh spacing, the TRAC solution methodology does provide two important advantages over EPRI-ARROTTA.

The primary advantage of the TRAC finite-difference heat conduction solution is that it uses different fuel-pellet thermal properties for each of the radial heat conduction nodes. In this model, local fuel material thermal properties are determined based upon the volume-averaged temperature which exists in the vicinity of a given radial/axial conduction node. This model will approach the true heat conduction solution in the limit of small mesh spacing.

The second advantage is that TRAC allows fuel heat conduction in both the axial and radial directions. But, in order to make the NEM/TRAC coding more consistent with EPRI-ARROTTA, TRAC's axial conduction option was switched off for all comparison calculations.

Similar to EPRI-ARROTTA, TRAC also provides a model for calculating the fuel-pellet dimensional variations which occur as the result of thermal expansion and contraction. Since the pellet expansion models employed by EPRI-ARROTTA and TRAC-PF1/MOD2 are somewhat different, both of these models will be shut off in comparison calculations.

In the process of coupling the TRAC and NEM coding, one must import some suitable spatially dependent cross-section feedback mechanism. Because of the desire to make the EPRI-ARROTTA and NEM/TRAC models as similar as feasible for comparison purposes, the EPRI-ARROTTA cross-section feedback mechanism (given by Eqn 6.1) was employed in NEM/TRAC without modification.

The NEM/TRAC coding also employs the EPRI-ARROTTA model to calculate the core-wide control fraction ( $R$ ) distribution. This simple model allows NEM/TRAC to track the time-dependent location of the tips of all movable control rods. As in EPRI-ARROTTA, this information is readily converted to a core-wide distribution of

control fractions ( $R$ ) by a simple algebraic relationship. In NEM/TRAC only the linear relationship between control-rod tip positions and cell control fractions is employed. This relationship is identical to the control-rod model used in the 3-D LMW transient calculations of Chapter 5. As before, this simple control rod model will introduce a small degree of control rod cusping error into the solution.

The NEM/TRAC coding uses a slightly different coupling scheme in the time domain. Before the details of this coupling scheme are introduced, one must gain a general understanding of the progression of a TRAC-PF1/MOD2 thermal-hydraulic time step.

A TRAC time step can be divided into three distinct stages: the pre-pass stage, the outer-iteration stage, and the post-pass stage. In the pre-pass stage, which occurs at the beginning of time step  $N+1$ , TRAC, among other things, calculates condition dependent material properties and convective heat-transfer coefficients, based upon the conditions existing at the end of time step  $N$ . In the outer iteration stage, TRAC solves the 1-D and 3-D fluid-dynamic equations. Finally, in the post-pass stage to complete the time step  $N+1$  solution, TRAC solves the fuel-rod heat conduction equations.

The transient NEM coding has been integrated into the general TRAC transient thermal-hydraulic solution procedure. First, routines to collect and store the latest node-dependent moderator and fuel rod conditions were placed in the outer-iteration and post-pass stages. The control-rod movement, cross-section feedback, and NEM routines were then placed into TRAC such that they would be called between the existing outer-iteration and post-pass stages. The overall transient calculational sequence proceeds as follows: Assume for the moment that time step  $N$  has just been completed. At this point, time step  $N$  values for the node dependent moderator and fuel conditions will be contained in storage. Time step  $N+1$  now begins. First, in the pre-pass stage,

new condition dependent material properties are calculated. Secondly, the overall fluid dynamic equations are solved in the TRAC outer-iteration stage. These time step N+1 equations are solved based upon the fuel-rod surface temperatures and energy fluxes calculated in the time step N post-pass. After the outer-iteration stage has been completed, time step N+1 node dependent moderator conditions will be determined. The newly added routines are evaluated next. Here, the control rods are moved to their end of time step N+1 positions. Node dependent control fractions are then calculated from these rod tip positions. The cross-section feedback routines are now evaluated. Time step N+1 cross sections are calculated based on nodal-cell-volume-averaged time step N fuel temperatures and time step N+1 moderator conditions. At this point the transient NEM neutronic routines are evaluated. These routines calculate the step N+1 nodal power distribution. Finally, to complete the time step N+1 solution, TRAC's normal post-pass phase is evaluated. Here, the just calculated time step N+1 nodal power distribution will serve as the heat source for the fuel rod heat conduction calculations. Time step N+1 is then complete.

In the case of the steady-state calculation, the NEM/TRAC coding follows a similar computational procedure. When a steady-state calculation is being performed, NEM/TRAC will iterate between the steady-state neutronics and thermal-hydraulics routines until the two modules yield a consistent set of nodal powers and thermal properties. In most cases, especially when performing once-through core-flow calculations to compare to EPRI-ARROTTA results, the steady-state neutronic routines will consume much more CPU time than do the thermal-hydraulic routines. As a result of this practical consideration, the NEM/TRAC coding has been set up to perform only one steady-state neutronics calculation for every five steady-state thermal-hydraulics calculations.

#### 6.4 The S. Levy PWR Test Problem

In the following two sub-sections the results of running both NEM/TRAC and EPRI-ARROTTA on a simplified transient PWR test problem will be compared. The detailed input description for this test problem, run with EPRI-ARROTTA by Antonio Dias of S. Levy Incorporated (44), appears in Section B.7 of Appendix B. This input basically simulates a radially and axially reflected, one-eighth-core symmetric Westinghouse pressurized water reactor with 193 fuel assemblies. Each of these fuel assemblies in turn contains 264 fuel rods, 24 control-rod guide tubes, and 1 instrumentation tube. This reactor is initially operating at a steady-state thermal power of 3411 MW with 21 partially inserted control rods when four of these control rods are quickly ejected from the core. The resulting severe power transient is then followed for a total of 0.3 seconds.

When this problem was evaluated by EPRI-ARROTTA, several model simplifications were made which might cause these results to deviate somewhat from experimental observations, if this experiment could in fact be safely performed. One of these simplifications is that fission energy is assumed deposited within the fuel pellets uniformly over their radii. This neglects the fuel's self-shielding of thermal neutrons, which generate most of the neutronic power. Secondly, all fission energy in this model is assumed deposited within the fuel pellets, when in fact a non-insignificant amount is deposited in the fuel cladding, assembly structure, control rods, and coolant. Next, the thermal expansion of the fuel pellets in this model has been neglected, a significant effect, when the fuel and cladding come into contact and affect heat transfer across the gas gap. Finally, because bulk coolant boiling is not expected in this calculation, the cross-section feedback mechanism which is based upon changes in moderator density (Eqn 6.1) has been turned off.

During the development of the NEM/TRAC model for the S. Levy PWR transient test problem, care was taken to ensure consistency with the existing EPRI-ARROTTA calculation. That is, certain TRAC-PF1/MOD2 thermal-hydraulic models were, if feasible, degraded in order to achieve results similar to EPRI-ARROTTA. To this end the TRAC models for fuel-pellet thermal expansion and axial fuel- rod heat conduction were turned off. For the same reason, TRAC's three-dimensional vessel fluid flow model was effectively disabled by placing zero flow areas between all nodes in the X-Y plane. In addition, the gravity term in the NEM/TRAC calculation was set to zero to eliminate the hydraulic head buoyancy effect. Finally, as in the EPRI-ARROTTA calculation, all fission energy in the NEM/TRAC model was deposited uniformly within only the fuel pellet. Otherwise, all TRAC thermal-hydraulic models were used without modification in this PWR transient calculation.

In the next two sub-sections various steady-state and transient results of performing this PWR calculation on both EPRI-ARROTTA and NEM/TRAC will be compared and contrasted.

#### 6.4.1 Steady-State Comparisons

In this sub-section comparisons of both steady-state neutronic and thermal-hydraulic results will be made. The steady-state neutronics results from EPRI-ARROTTA and NEM/TRAC will be compared before introducing control rods, soluble boron, or thermal-hydraulic feedback. In performing this comparison, one can isolate the differences between the two steady-state neutronics calculations. If these discrepancies are assumed not to change when control rods, soluble boron, and thermal-hydraulic feedback are included in the calculation, one can then infer what proportion of the calculational differences result solely from these added effects. The axially averaged one-quarter-core symmetric assembly power distributions produced by EPRI-ARROTTA

and NEM/TRAC are compared in Table 6.1. As can be seen in this table, the assembly powers in these two calculations agree to within 3% for all assemblies. This level of agreement is, in general, slightly worse than that obtained with respect to reference solutions when 3-D, neutronic only, steady-state benchmark calculations were run in Chapter 4. Since a reference solution does not exist for the S. Levy test problem, it is impossible to know how much each of the respective calculations is in error. The differences between the two solutions should, however, be low enough to allow valid comparisons to be made when the thermal-hydraulic feedback mechanisms are invoked. In addition, Table 6.1 shows the global eigenvalues which result from these two steady-state neutronics calculations. These eigenvalues agree to within four significant digits.

Now the effects of control rods and soluble boron are considered in the steady-state neutronics calculations. In addition, the respective EPRI-ARROTTA and NEM/TRAC thermal-hydraulic feedback models were activated. In both cases, converging the steady-state neutronic and thermal-hydraulic solutions required many iterations. In the case of EPRI-ARROTTA, 264 neutronic outer iterations and 46 thermal-hydraulic iterations were required to achieve steady-state convergence, i.e., a maximum change in individual nodal power of less than 0.01%. This calculation consumed approximately 1900 seconds of CPU time on an IBM 3083 mainframe computer. In contrast, the NEM/TRAC calculation required 38 neutronic outer iterations and 200 thermal-hydraulic iterations for the maximum nodewise change in liquid temperature between iterations to drop below 0.01%. This second calculation used more than 1300 seconds of CPU time on a Cray XMP supercomputer, approximately 95% of which was consumed in the neutronics routines. The great difference in the ratio of the number of neutronic to thermal-hydraulic time steps in the two calculations is the result of using different steady-state iteration strategies.

**Table 6.1 EPRI-ARROTTA and NEM/TRAC Axially Averaged Assembly Power Distributions and Percentage Differences for the Steady-State S. Levy PWR Test Problem Without Control Rods, Soluble Boron, or Thermal-Hydraulic Feedback**

0.9747 0.9510(-2.44)	0.9860 0.9663(-1.99)	0.8878 0.8698(-2.02)	0.7636 0.7434(-2.64)				
0.8976 0.9846(-0.34)	1.0354 1.0239(-1.11)	0.8637 0.8631(-0.07)	0.9855 0.9721(-1.36)	1.0278 1.0067(-2.06)	0.7746 0.7527(-2.82)		
1.0004 1.0016(0.12)	0.9223 0.9295(0.78)	0.9760 0.9745(-0.16)	0.9092 0.9129(0.41)	0.9399 0.9324(-0.80)	0.9333 0.9167(-1.78)	0.7748 0.7527(-2.85)	
1.0103 1.0229(1.25)	1.1282 1.1352(0.62)	0.9787 0.9904(1.19)	1.0358 1.0378(0.20)	1.1874 1.1893(0.15)	0.9407 0.9324(-0.88)	1.0292 1.0067(-2.18)	
1.2005 1.2133(1.07)	1.0697 1.0870(1.62)	1.1749 1.1856(0.91)	1.0138 1.0264(1.24)	1.0362 1.0378(0.16)	0.9102 0.9129(0.30)	0.9878 0.9721(-1.59)	0.7657 0.7434(-2.92)
1.0625 1.0826(1.89)	1.2025 1.2178(1.27)	1.0827 1.1021(1.79)	1.1752 1.1856(0.88)	0.9796 0.9904(1.10)	0.9764 0.9745(-0.19)	0.8661 0.8631(-0.35)	0.8915 0.8698(-2.43)
1.0781 1.0942(1.49)	1.0316 1.0533(2.10)	1.2029 1.2178(1.24)	1.0696 1.0870(1.62)	1.1284 1.1352(0.60)	0.9227 0.9295(0.74)	1.0356 1.0239(-1.12)	0.9857 0.9663(-1.97)
0.9672 0.9892(2.27)	1.0785 1.0942(1.45)	1.0613 1.0826(2.01)	1.2007 1.2133(1.06)	1.0098 1.0229(1.30)	1.0015 1.0016(0.01)	0.8958 0.8946(-0.14)	0.9719 0.9510(-2.15)

ARROTTA ( $K_{eff} = 1.1192$ )

NEM/TRAC (% Difference) ( $K_{eff} = 1.1190$ )

The nodewise distribution of several key neutronic and thermal-hydraulic quantities must be compared to ensure the reasonableness of the new steady-state NEM/TRAC calculation. One of the first ways of comparing these calculations will involve observing the final steady-state assembly power distributions. These normalized power distributions and their percentage differences are contained in Table 6.2. These new assembly power distributions include the effects of inserted control rods, 975 ppm of soluble boron, and most important of all, thermal-hydraulic cross-section feedback.

Upon comparing Table 6.2 with Table 6.1, one recognizes that the assembly power distribution has been shifted, generally towards the outermost assemblies. This phenomena is mainly the result of partially inserting 21 highly absorbing control rods into the central region of the core. One also notices in comparing Table 6.2 with 6.1 that  $K_{eff}$  has dropped significantly. This drop of approximately  $0.13 \frac{\Delta k}{k}$  is the result of both the addition of control rods and 975 ppm of soluble boron to the core and the effects of the thermal-hydraulic cross-section feedback mechanism. Table 6.2 also shows that the values of  $K_{eff}$  obtained by EPRI-ARROTTA and NEM/TRAC differ by almost 0.15%, much more than they had previously, but quite acceptable considering the greatly increased computational complexity. Finally, one notices a slight general increase in the distribution of assembly power differences in going from Table 6.1 to Table 6.2. Again, these increases are primarily a result of the greatly increased complexity of the coupled neutronic/ thermal-hydraulic calculations.

Next, the outlet coolant temperature distribution will be examined. This quantity depends on only the coolant inlet temperature, the coolant inlet mass flow rate, the coolant temperature/enthalpy correlations, the reactor steady-state power level, and the aforementioned assembly power distribution. The rise (above inlet temperature) in the assembly outlet coolant temperatures obtained by EPRI-ARROTTA is presented in

**Table 6.2 EPRI-ARROTTA and NEM/TRAC Axially Averaged Assembly Power Distributions and Percentage Differences for the Steady-State S. Levy PWR Test Problem With Control Rods, Soluble Boron, and Thermal-Hydraulic Feedback**

1.1438 1.1095(-2.99)	1.2614 1.2329(-2.26)	1.2293 1.2086(-1.68)	1.0657 1.0420(-2.22)				
0.8007 0.7897(-1.38)	1.2492 1.2355(-1.10)	1.0912 1.0898(-0.12)	1.3544 1.3394(-1.11)	1.3859 1.3635(-1.62)	1.0515 1.0254(-2.48)		
1.0349 1.0394(0.43)	0.9837 0.9928(0.93)	1.1122 1.1152(0.26)	0.9864 0.9924(0.60)	1.0425 1.0380(-0.42)	1.2390 1.2199(-1.54)	1.0517 1.0254(-2.50)	
0.9680 0.9853(1.79)	1.1075 1.1213(1.24)	0.9203 0.9375(1.86)	0.8870 0.8960(1.02)	0.6290 0.6304(0.23)	1.0436 1.0380(-0.53)	1.3879 1.3635(-1.76)	
1.0463 1.0660(1.89)	0.8833 0.9036(2.30)	0.9401 0.9559(1.68)	0.8018 0.8161(1.79)	0.8876 0.8960(0.94)	0.9879 0.9924(0.46)	1.3585 1.3394(-1.41)	1.0704 1.0420(-2.65)
0.7838 0.8097(3.30)	0.8632 0.8831(2.31)	0.6071 0.6187(1.91)	0.9412 0.9559(1.57)	0.9221 0.9375(1.66)	1.1141 1.1152(0.09)	1.0946 1.0898(-0.43)	1.2376 1.2086(-2.34)
0.6610 0.6803(2.91)	0.6663 0.6860(2.96)	0.8647 0.8831(2.12)	0.8845 0.9036(2.16)	1.1084 1.1213(1.16)	0.9846 0.9928(0.84)	1.2501 1.2355(-1.17)	1.2614 1.2329(-2.26)
0.3131 0.3228(3.10)	0.6619 0.6803(2.78)	0.7886 0.8097(2.67)	1.0476 1.0660(1.76)	0.9677 0.9853(1.82)	1.0359 1.0394(0.33)	0.7989 0.7897(-1.15)	1.1385 1.1095(-2.54)

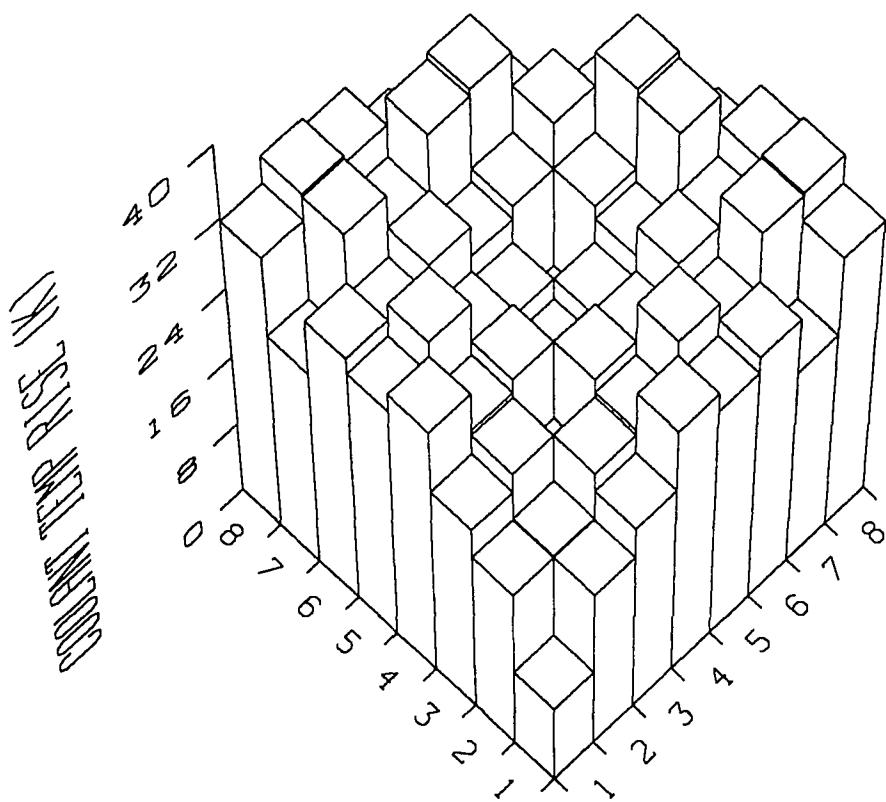
ARROTTA ( $K_{eff} = 0.97566$ )

NEM/TRAC (% Difference) ( $K_{eff} = 0.97429$ )

graphical form in Figure 6.1. Upon observing this figure, one notices that, as expected, the assemblies with the highest steady-state coolant exit temperature are those with the highest assembly normalized powers. Table 6.3 shows this same data in tabular format along with the equivalent data from the steady-state NEM/TRAC calculation. At the primary system pressure of 15.5 mPa (2250 PSI), the coolant would have to reach a temperature of 618 K to begin bulk boiling. This translates to a 53 K temperature rise from the coolant inlet to outlet. As is evident from the data shown in Table 6.3, no bulk boiling occurs in even the hottest channels of the core. In addition, there should only be a minimal amount of sub-cooled boiling in the core. Table 6.3 also shows that the NEM/TRAC outlet coolant temperatures are, on average, slightly lower than those obtained with EPRI-ARROTTA. Since both codes specify the same coolant inlet mass flow rates and temperatures and steady-state reactor power levels, this global difference can only be attributed to NEM/TRAC generally using larger values for specific heat at a given sub-cooled temperature than does EPRI-ARROTTA.

At this point it would be instructive to observe how the coolant temperature distribution varies axially within the core. Figure 6.2 shows how the coolant temperature in the average X-Y plane assembly steadily rises as it travels up the core. Upon inspection, one immediately notices that the average coolant temperature in the first axial node is somewhat higher in NEM/TRAC than in EPRI-ARROTTA. This is a result of differences in the heat conduction/ convection coupling strategies used in the two codes. EPRI-ARROTTA adds partial-cell flow-path integrated energy flux to the coolant at the axial center of a cell while NEM/TRAC adds full-cell flow-path integrated energy flux to the same location. As a result, NEM/TRAC will add heat to the coolant integrated to a half-cell higher than does EPRI-ARROTTA. One can also see that the average coolant temperature initially rises more steeply in the NEM/TRAC calculation than in the EPRI-ARROTTA calculation. This small difference in initial slope is a function of

Figure 6.1 EPRI-ARROTTA Steady-State Outlet Coolant Temperature Rise



**Table 6.3 EPRI-ARROTTA and NEM/TRAC Steady-State Coolant Temperature Rises (K) and Percentage Differences**

-Inlet Temperature = 564.8 K

-Saturation Temperature = 618.0 K (53.15 K rise)

31.64 30.04(-5.04)	34.58 32.92(-4.79)	33.79 32.37(-4.19)	29.62 28.30(-4.48)				
22.72 22.45(-1.20)	34.28 33.07(-3.53)	30.34 29.67(-2.18)	36.83 35.39(-3.91)	37.55 35.88(-4.45)	29.21 27.83(-4.72)		
28.82 28.28(-1.89)	27.50 27.06(-1.58)	30.80 30.10(-2.29)	27.55 27.02(-1.94)	28.98 28.11(-3.01)	33.94 32.52(-4.16)	29.21 27.83(-4.72)	
27.06 26.83(-0.86)	30.65 30.19(-1.52)	25.82 25.62(-0.75)	24.94 24.52(-1.65)	18.00 17.37(-3.49)	29.01 28.11(-3.10)	37.60 35.88(-4.57)	
29.07 28.82(-0.86)	24.84 24.77(-0.27)	26.34 26.18(-0.61)	22.67 22.51(-0.74)	24.95 24.52(-1.71)	27.59 27.02(-2.07)	36.93 35.39(-4.17)	29.75 28.30(-4.87)
22.20 22.32(0.55)	24.31 24.31(0.00)	17.43 17.82(2.23)	26.37 26.18(-0.72)	25.86 25.62(-0.92)	30.85 30.10(-2.45)	30.42 29.67(-2.47)	33.99 32.37(-4.77)
18.87 18.86(-0.06)	19.02 19.06(0.23)	24.36 24.31(-0.18)	24.87 24.77(-0.40)	30.67 30.19(-1.59)	27.52 27.06(-1.66)	34.30 33.07(-3.60)	34.58 32.92(-4.79)
9.14 9.01(-1.46)	18.90 18.86(-0.18)	22.32 22.32(-0.02)	29.11 28.82(-0.99)	27.05 26.83(-0.82)	28.85 28.28(-1.96)	22.67 22.45(-0.98)	31.50 30.04(-4.64)

## ARROTTA

## NEM/TRAC (% Difference)

both variations in the calculated plane averaged axially dependent nodal power distributions and the sub-cooled temperature/ enthalpy correlation. Higher up the core these curves begin to converge, then finally cross one another approximately 3/4 of the way up the core.

Figure 6.3 shows the same moderator temperature verses core height data for the highest powered assembly in the core, i.e. the assembly at position X=5, Y=7. Although the coolant heats up more in this assembly, the general axial trend is the same as in the average assembly. That is, the coolant temperature initially rises more steeply in the NEM/TRAC calculation. This higher rate of increase again slackens and the two curves cross approximately 2/3 of the way up the core. Again the disparity between the two curves is the result of differences in both the hot assembly axial-power profile and the sub-cooled temperature/ enthalpy correlation.

The final thermal-hydraulic quantity of interest is the node average fuel temperature. The node average fuel temperature is defined as an average over both the radius of individual fuel rods and the planar and axial extent of a specific neutronic/ thermal-hydraulic node. The axial variation of the node average fuel temperature in the average assembly can be seen in Figure 6.4.

Upon studying Figure 6.4, one notices that, in general, the NEM/TRAC calculation yields higher node-average fuel temperatures than does the EPRI-ARROTTA calculation. The reasons behind this discrepancy may be four-fold. First, there may be a difference in the calculated average axial power profiles. The contribution of this factor is lessened if one notices that the planar averaged NEM/TRAC fuel temperatures lie above those of EPRI-ARROTTA by approximately the same amount. The second contributor is the average moderator temperature at a given height. This quantity, as shown in Figure 6.2, may be a small contributor to NEM/TRAC's higher average

Figure 6.2 EPRI-ARROTTA and NEM/TRAC Average Assembly Moderator Temperature as a Function of Core Height

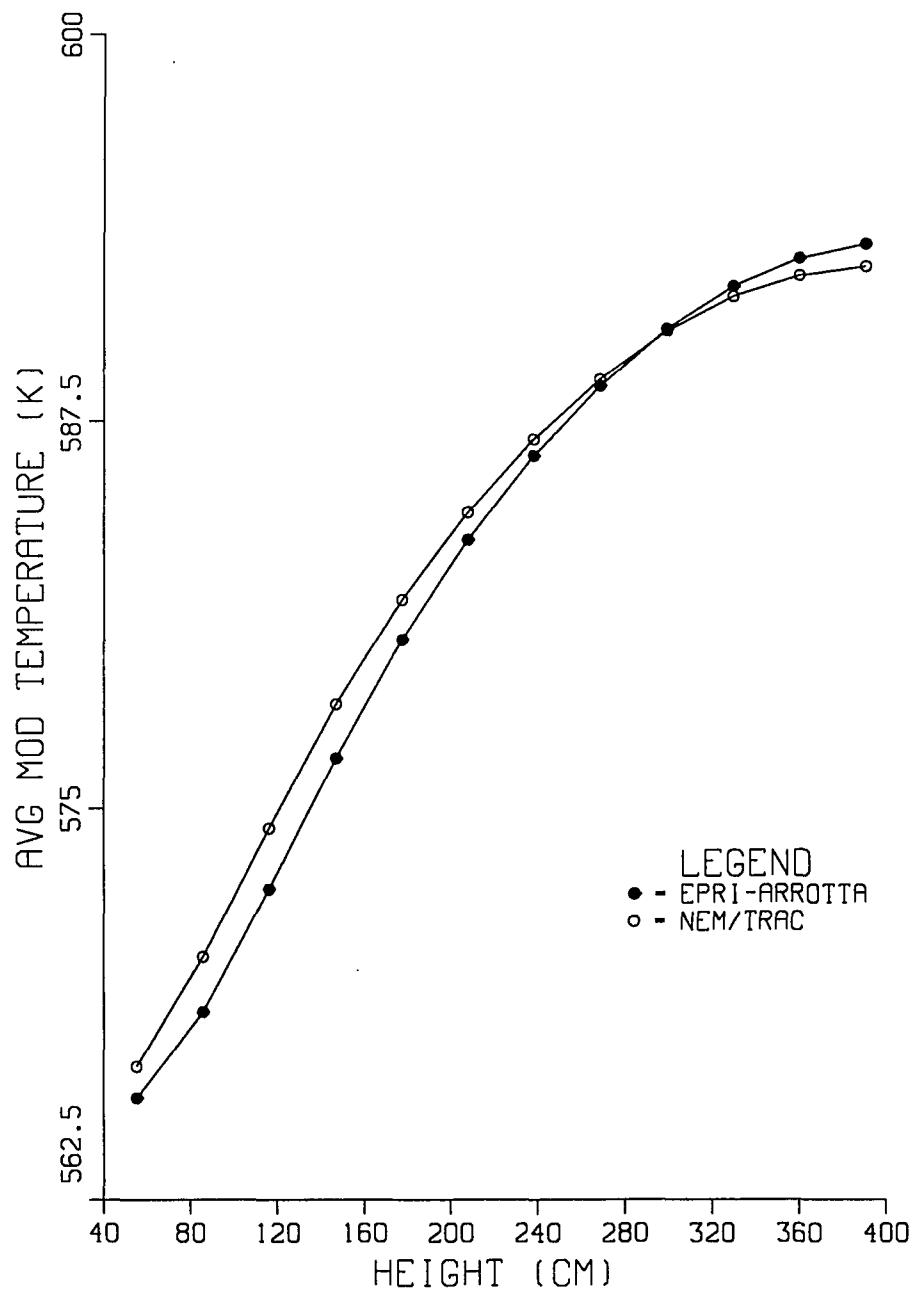


Figure 6.3 EPRI-ARROTTA and NEM/TRAC Hottest Assembly Moderator Temperature as a Function of Core Height

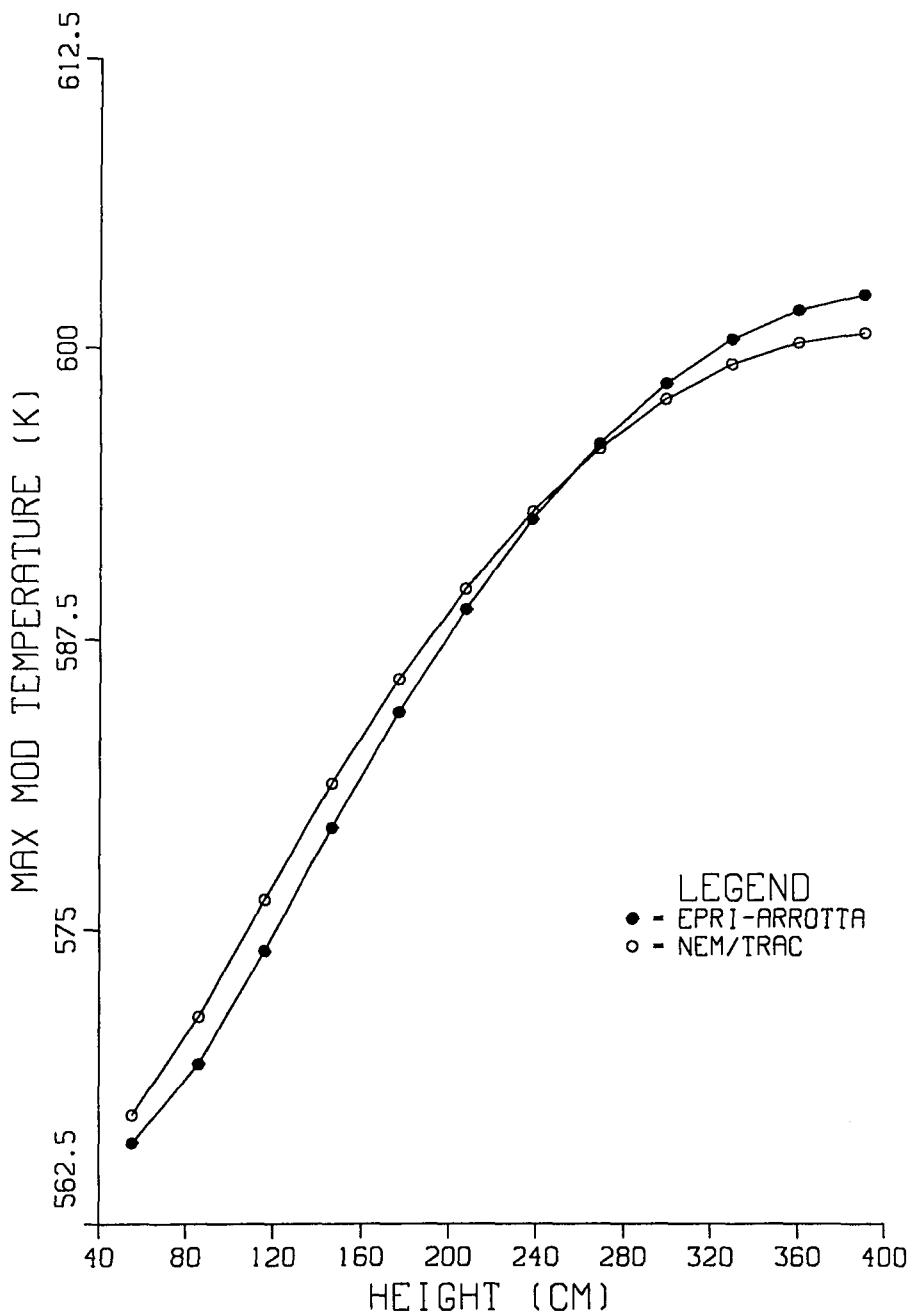
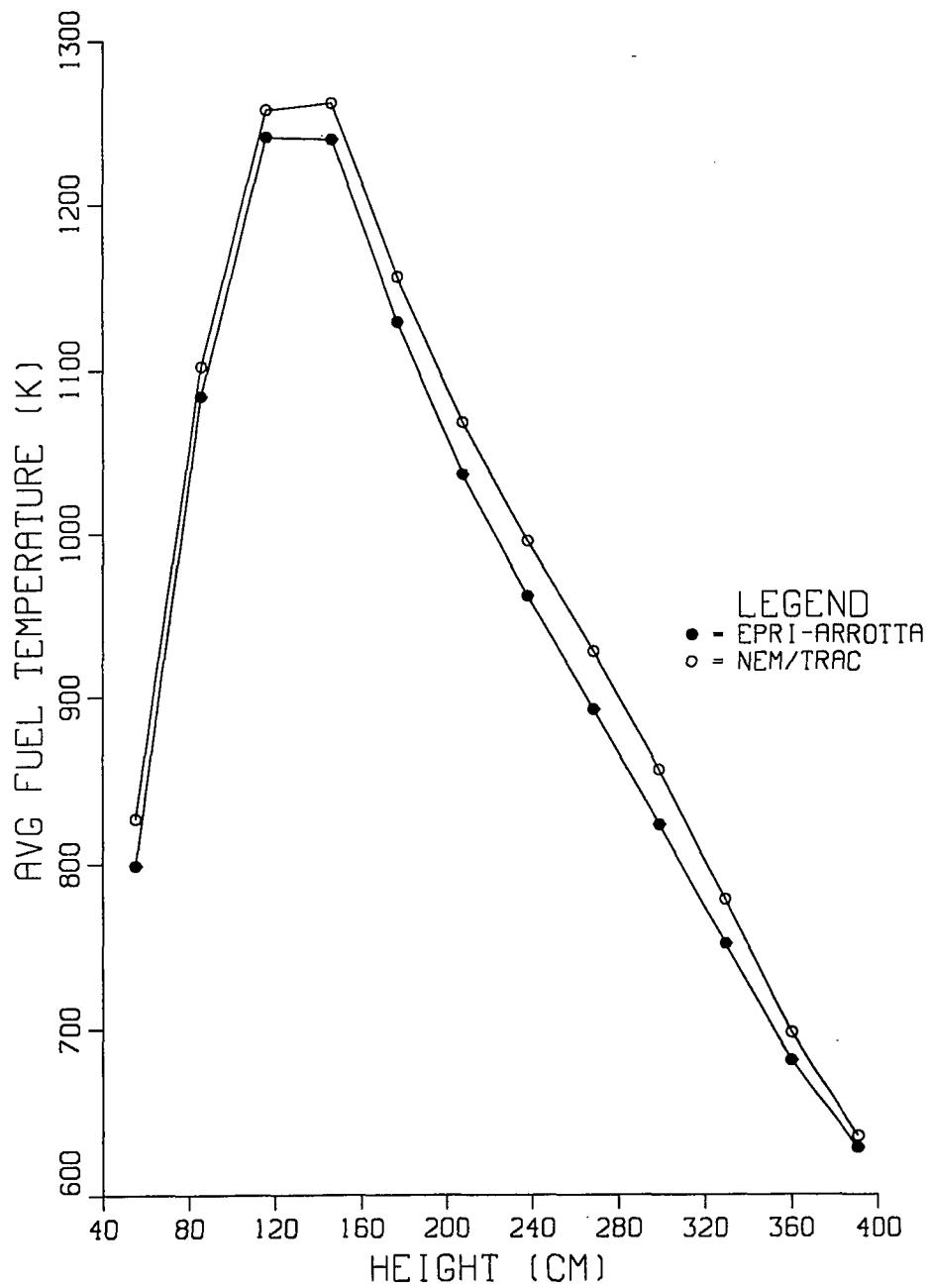


Figure 6.4 EPRI-ARROTTA and NEM/TRAC Average Assembly Fuel Temperature as a Function of Core Height



fuel temperature. Next, code specific differences in the fuel rod material thermal conductivities and convective heat transfer coefficients may be a cause of the higher fuel temperatures in NEM/TRAC. Finally, the higher fuel temperatures are probably the result of NEM/TRAC using separate temperature dependent UO<sub>2</sub> thermal conductivities for each of its ten radial heat-conduction nodes, whereas EPRI-ARROTTA uses a single average thermal conductivity. The higher NEM/TRAC fuel temperature is most likely not a function of fuel-rod gas-gap thermal conductivity because the same constant values are specified in both codes. In addition, this discrepancy is not the result of either fuel rod dimensions or the number of powered rods per assembly, since the same values for these quantities are specified in both codes.

To better understand the cause of NEM/TRAC's higher fuel temperatures, a study of the fuel pellet thermal conductivity correlations used in NEM/TRAC and EPRI-ARROTTA was performed. In EPRI-ARROTTA the fuel-pellet thermal conductivity is related to fuel temperature as follows:

$$K_f \left( \frac{BTU}{HR - FT - ^\circ F} \right) = \frac{3.9781}{0.69261 + \frac{T(^{\circ}F)}{1000}} + 0.00602366 \left( 0.46 + \frac{T(^{\circ}F)}{1000} \right)^3 \quad (6.2)$$

A completely different correlation is used in NEM/TRAC. This correlation is:

$$K_f \left( \frac{W}{M - K} \right) = 105.26 \cdot (1 - (1 - FTD) \cdot (2.58 - 5.8 \times 10^{-4} T(^{\circ}C))) \cdot \left( \frac{40.4}{464.0 + T(^{\circ}C)} + 1.216 \times 10^{-4} \cdot e^{(1.867 \times 10^{-3} \cdot T(^{\circ}C))} \right) \quad (6.3)$$

for T(^{\circ}C) ≤ 1650° C

and

$$K_f \left( \frac{W}{M - K} \right) = 105.26 \cdot (1 - (1 - FTD) \cdot (2.58 - 5.8 \times 10^{-4} T(^{\circ}C))) \cdot \left( 0.0191 + 1.216 \times 10^{-4} \cdot e^{(1.867 \times 10^{-3} \cdot T(^{\circ}C))} \right) \quad (6.4)$$

for  $T(^{\circ}C) > 1650^{\circ}C$

where

$$FTD \equiv \text{fraction of theoretical UO}_2 \text{ density}$$

Figure 6.5 shows an SI-unit plot of these two fuel conductivity correlations over a typical range of fuel-pellet operating temperatures (600 K to 2600 K). As is readily apparent from the figure, NEM/TRAC employs a lower thermal conductivity over the entire range of normal steady-state fuel pellet operating temperatures (600 K to 2000 K). As a result, differences in fuel thermal conductivity should be a major contributor to NEM/TRAC's higher average fuel temperatures, as depicted in Figure 6.4.

Next, the axially dependent temperatures in the hottest assembly, X=5, Y=7, will be studied. Plots of these quantities, as calculated by NEM/TRAC and EPRI-ARROTTA, are displayed in Figure 6.6. Both fuel temperature curves in this figure generally have the same shape as those in Figure 6.4, except that the peak temperatures in this hottest assembly are approximately 200 K higher than those in the average assembly. In contrast to the average assembly, EPRI-ARROTTA calculates higher average fuel temperatures than NEM/TRAC at some elevations within the hottest assembly. This phenomena can be readily understood if one observes the steady-state normalized assembly power distribution given in Table 6.2. Here one can see that NEM/TRAC calculates a power level in the hottest assembly that is 1.68% lower than that of EPRI-ARROTTA.

As will be seen in later sections of this chapter, NEM/TRAC's higher average fuel temperatures will, because of the fuel-temperature cross-section feedback mechanism (Eqn 6.1), have an effect on all spatially dependent neutronic and thermal-hydraulic quantities in a transient calculation.

Figure 6.5 EPRI-ARROTTA and NEM/TRAC Fuel Pellet Thermal Conductivities as a Function of Temperature

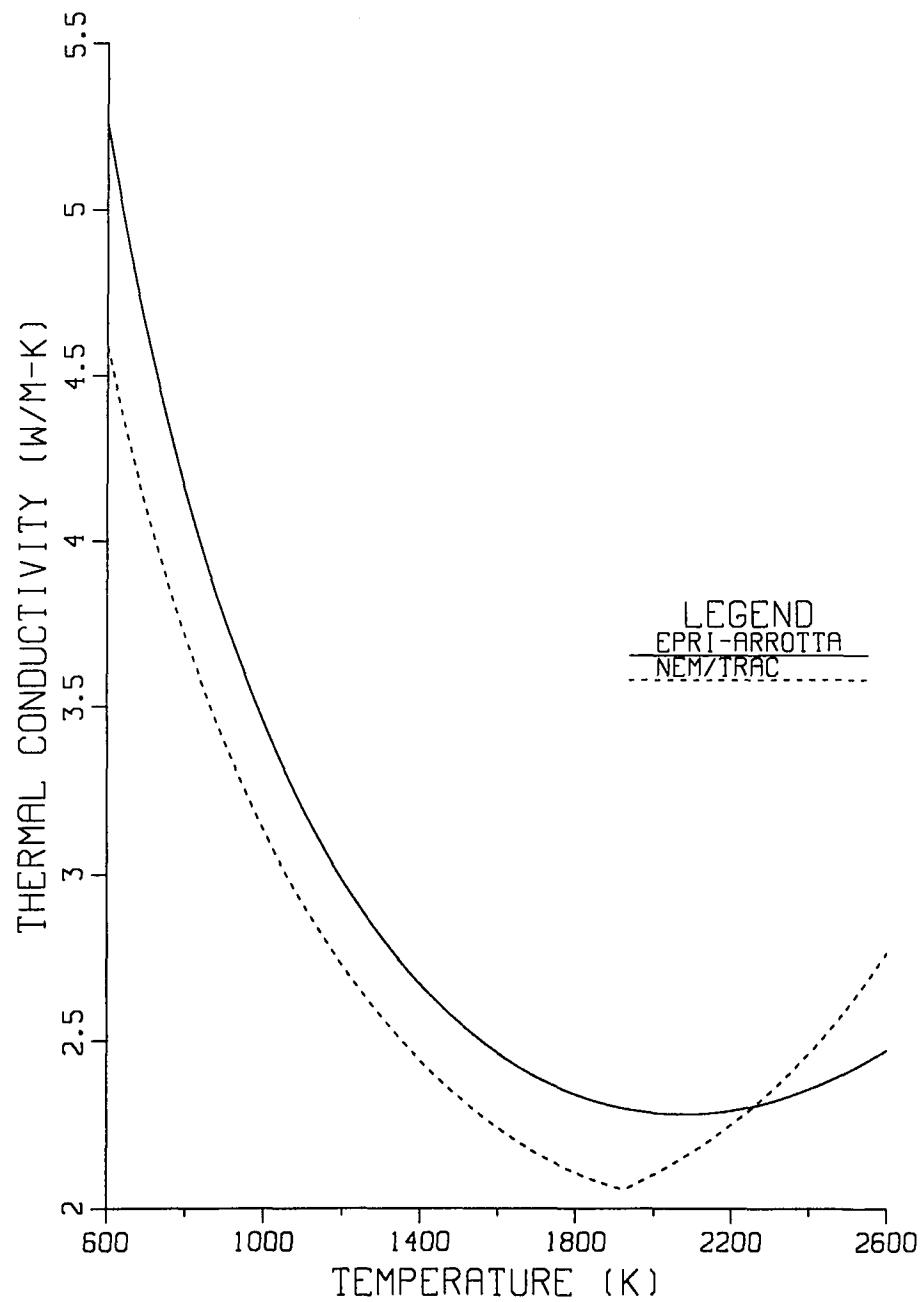
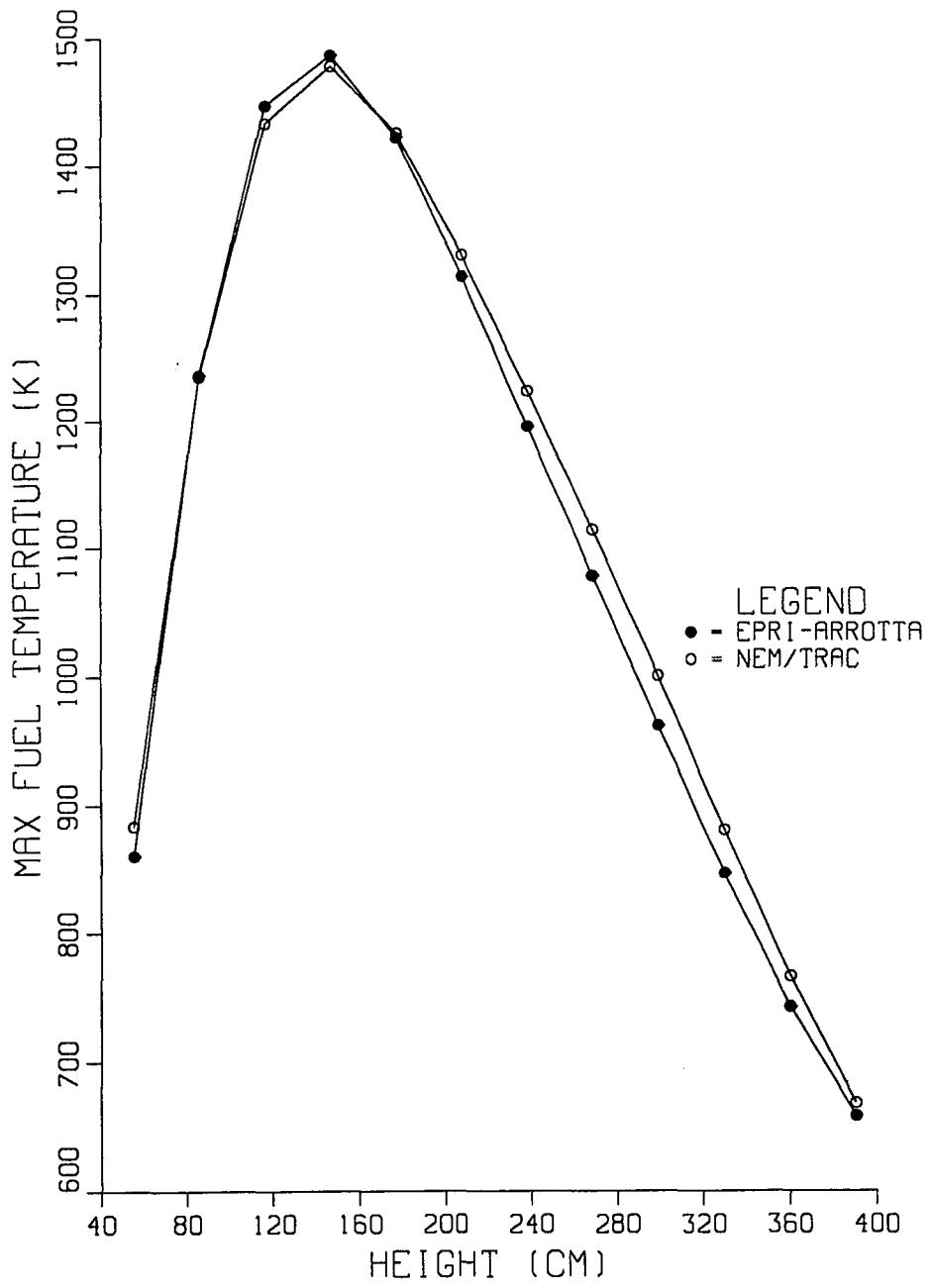


Figure 6.6 EPRI-ARROTTA and NEM/TRAC Hottest Assembly Fuel Temperature as a Function of Core Height



#### 6.4.2 Transient Comparisons

In this section the results of the transient portion of the S. Levy test problem with NEM/TRAC and EPRI-ARROTTA will be compared and contrasted. As previously stated, the transient portion of the S. Levy test problem is initiated by rapidly removing (at 35 m/s) the partially inserted control rod at position X=5, Y=5. Since this Westinghouse PWR is being modelled with one-quarter-core symmetry, a total of four control rods are actually being ejected from the entire core. This control rod ejection results in a severe prompt-critical power excursion, which is quickly damped by the fuel-temperature (doppler) cross-section feedback mechanism. As a result of the speed at which this power excursion occurs, the amount of time spent following the transient can be relatively small, in this case only 0.3 seconds. In following this transient NEM/TRAC and EPRI-ARROTTA both employ a fixed time step size of 0.002 seconds, which results in a total of 150 time steps.

NEM/TRAC used a large amount of CPU time to perform the transient portion of the S. Levy calculation. Almost 2900 CPU seconds on a Cray XMP supercomputer were required to perform this calculation. Approximately 99% of that time was consumed in the NEM transient neutronics routines. S. Levy Incorporated personnel claimed that the transient portion of the EPRI-ARROTTA calculation consumed approximately 1200 seconds of CPU time on an IBM 3083 mainframe computer (44). A study by Dongarra (45) establishes that a Cray XMP performs typical numerically intensive FORTRAN calculations at speeds of approximately 66 million floating point operations per second (66 mflops) while an IBM 3083 performs similar calculations at 19 mflops, a factor of 3.5 more slowly. As a consequence, the more problem specific EPRI-ARROTTA coding executes the transient portion of the S. Levy test problem approximately 8.5 times more efficiently than NEM/TRAC.

The just discussed NEM/TRAC and EPRI-ARROTTA steady-state calculations serve to provide initial conditions for these transient calculations. This means that many features of the steady-state results will manifest themselves in some form during the transient calculation.

The first calculated quantity to be compared in this transient calculation will be the total reactor power. A plot of the variation of this quantity with respect to transient time is shown in Figure 6.7. Upon studying this figure, one notices that the time-dependent reactor power levels produced by NEM/TRAC and EPRI-ARROTTA agree with one another quite closely. This level of agreement becomes more significant if one observes that reactor power varies by approximately a factor of 50 during the short duration of this transient.

In spite of the previously discussed level of agreement, one first notices that NEM/TRAC and EPRI-ARROTTA predict peak transient power levels that differ by approximately 4%. Possible reasons for this disagreement at approximately 0.09 seconds into the transient will now be studied. The predominant reasons behind this discrepancy can most likely be attributed to differences in thermal-hydraulic cross-section feedback. Since both codes employ the same feedback mechanism (Eqn 6.1) and feedback coefficients, any differences in feedback must be the result of NEM/TRAC generally having either higher node average fuel temperatures or lower moderator densities than does EPRI-ARROTTA. These possibilities are explored in Figures 6.8 through 6.10.

Figure 6.8 displays the time-dependence of the core-average moderator temperature throughout the transient. As can be seen in this figure, this power transient is so rapid that very little moderator heatup occurs over the entire extent of either calculation. This behavior is to be expected both as a result of system thermal inertia (i.e., the storage of heat energy within the fuel material) and as a result the relatively small amount of

Figure 6.7 EPRI-ARROTTA and NEM/TRAC Total Core Power as a Function of Time

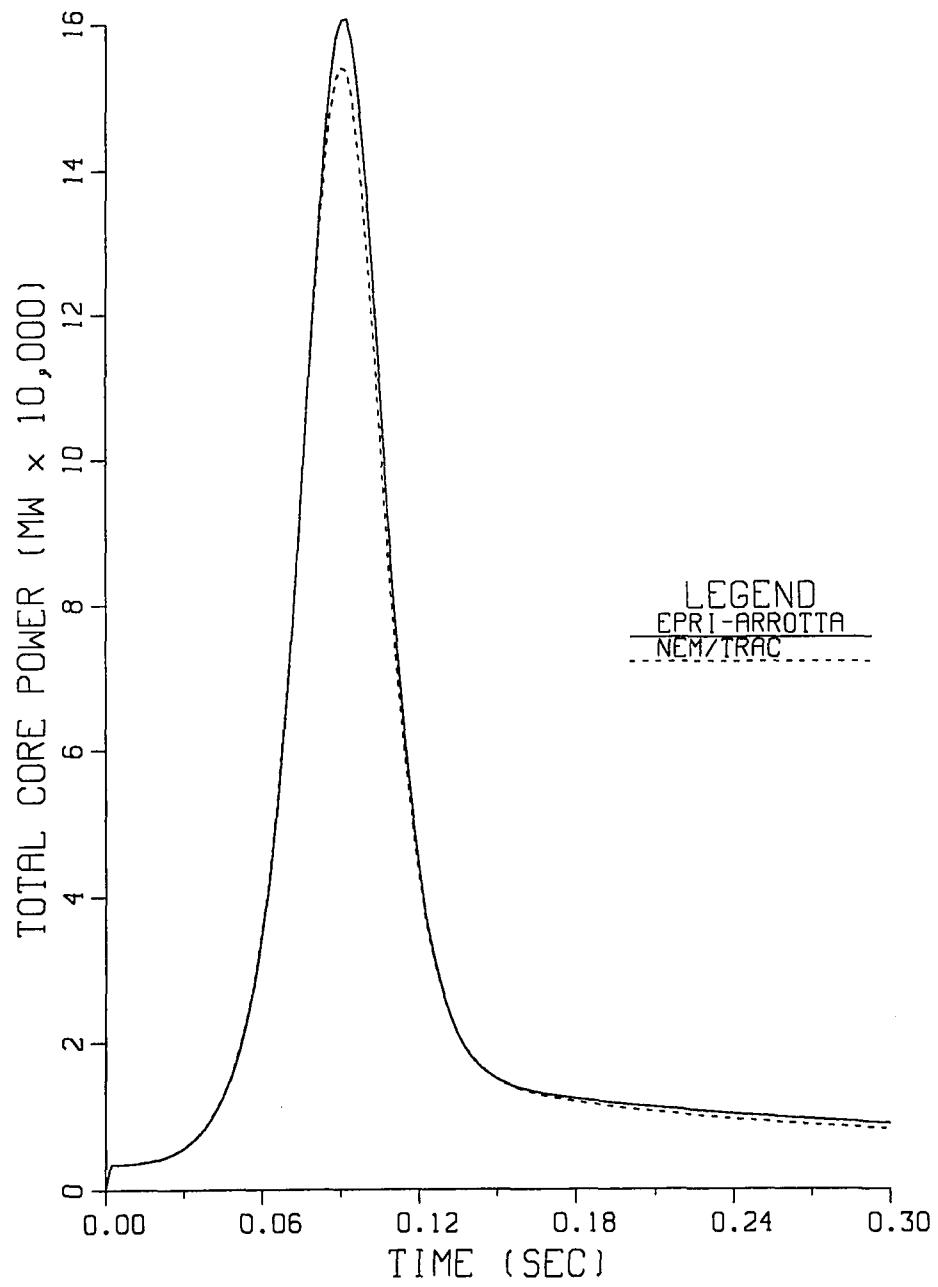


Figure 6.8 EPRI-ARROTTA and NEM/TRAC Core Average Moderator Temperature as a Function of Time

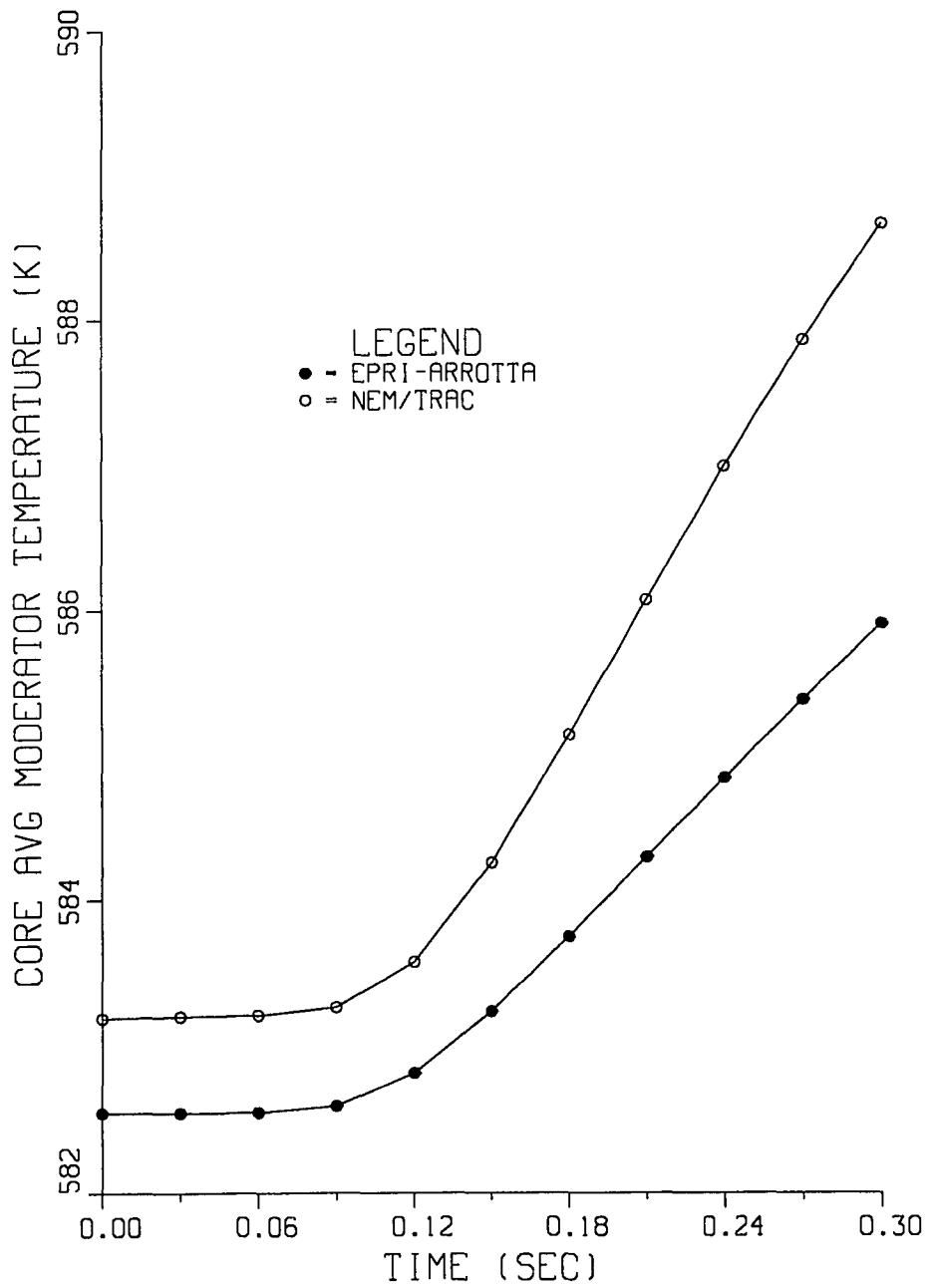


Figure 6.9 EPRI-ARROTTA and NEM/TRAC Average Outlet Moderator Temperature as a Function of Time

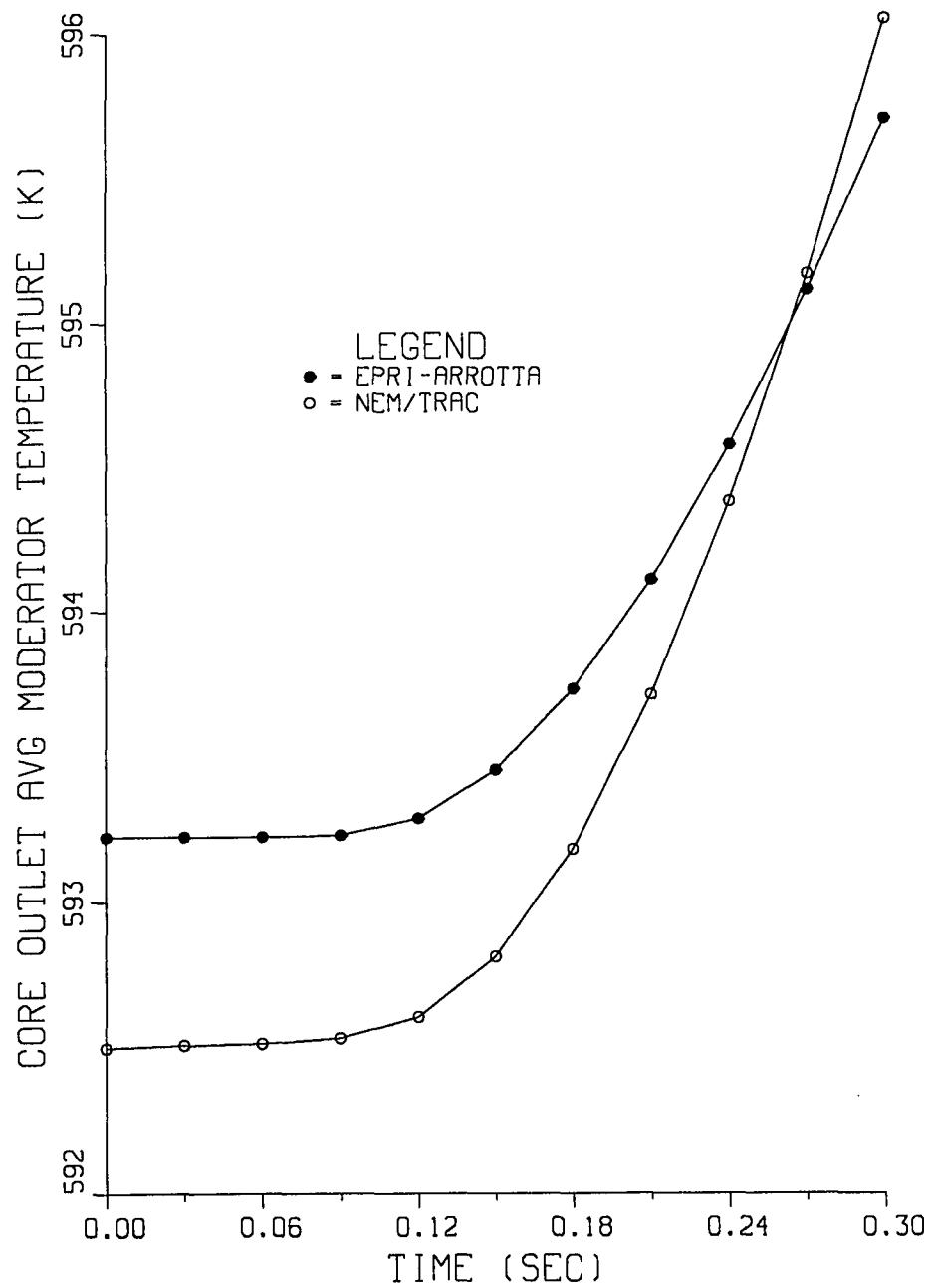
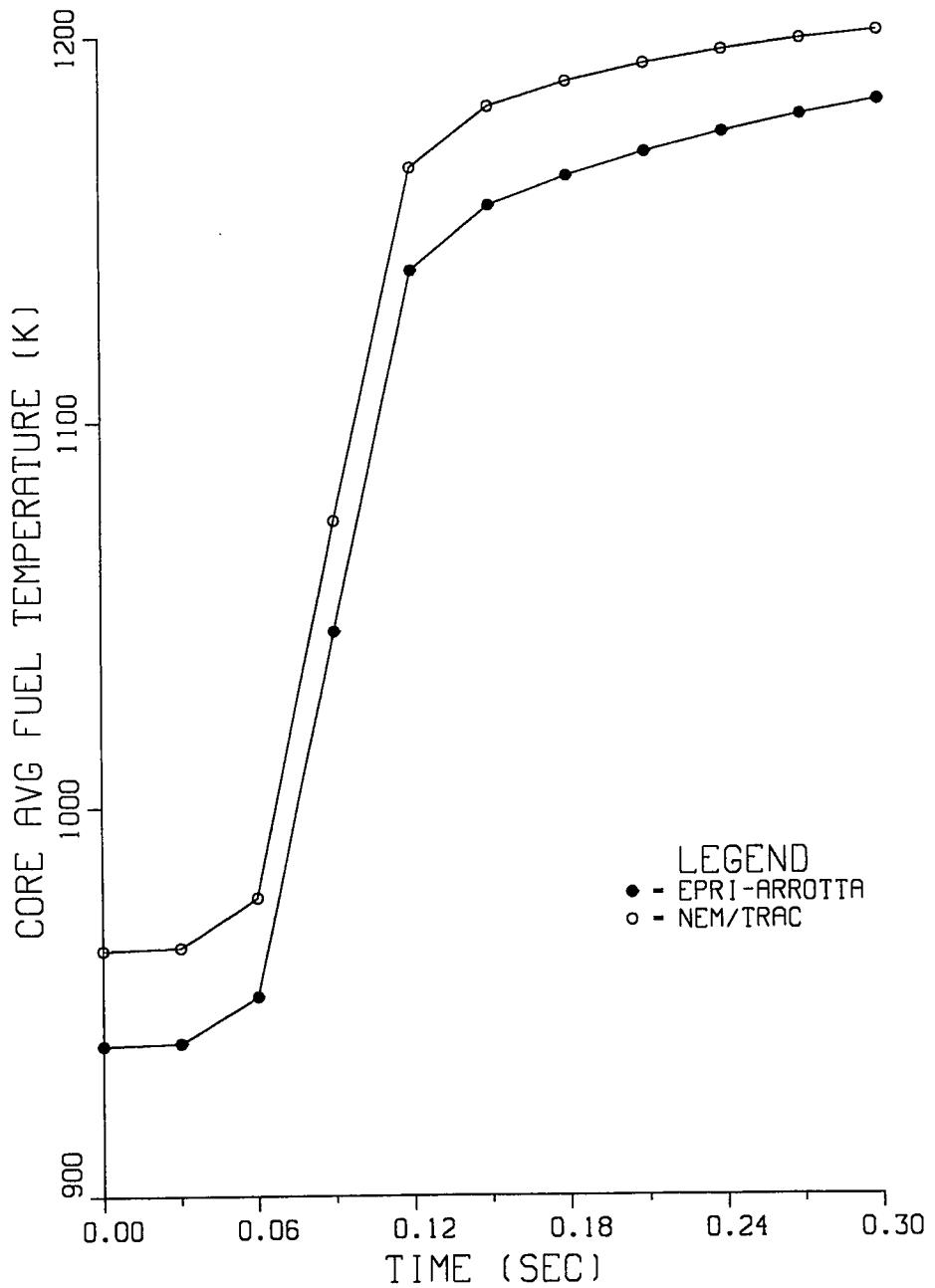


Figure 6.10 EPRI-ARROTTA and NEM/TRAC Core Average Fuel Temperature as a Function of Time



heat energy produced by the very rapid power excursion (from 3411 MW to 154000 MW). At 0.09 seconds into the transient, the approximate time when peak power has been reached, one notices that, on average, almost no moderator heatup has yet taken place. This means that, other than a slight offset carried over from steady-state, the moderator-density feedback mechanism will have little effect on the differences in the calculated peak transient core-power levels.

As can be seen in Figure 6.9, the average outlet coolant temperature rises only slightly over the course of the transient. The initial upward trend in this quantity lags behind a similar rise in the core average moderator temperature. This delay is the result of the time necessary for coolant to flow from the heated core to the outlet. In addition, throughout most of the transient NEM/TRAC yields slightly higher average outlet moderator temperatures than does EPRI-ARROTTA. This apparent contradiction with Figure 6.8 can be understood if one reviews Figure 6.2 and notices that at steady-state NEM/TRAC generally predicts higher moderator temperatures than EPRI-ARROTTA. Everywhere, that is, except at the outlet. Figure 6.9 reinforces the conclusion that the moderator density feedback mechanism has little effect on the timing or the magnitude of the transient power spike seen in Figure 6.7

In contrast to the core-average and outlet-average moderator temperatures, the core average fuel temperature rises significantly (approximately 250 K) during the course of this transient. This phenomenon is readily apparent upon viewing Figure 6.10. The prompt or doppler cross-section feedback which results from this significantly increased fuel temperature is the mechanism which reverses the power spike at approximately 0.09 seconds. The power rise will be reversed when the doppler (fuel temperature) feedback mechanism adds enough negative reactivity to counteract the positive reactivity being

added by the location X=5, Y=5 control rod withdrawal. The core average fuel temperature continues to rise beyond 0.09 seconds as heat energy continues to be added to the fuel faster than it can be transferred to the flowing coolant. A quasi-equilibrium state is reached at approximately 0.15 seconds into the transient as the core-energy production rate and the rate of heat transfer to the flowing coolant begin to come into balance at a new but significantly higher (approximately 400%) steady-state power level.

In addition to the very fast rise in core-average fuel temperature, one also notices that the NEM/TRAC calculation consistently predicts higher fuel temperatures (by approximately 25 K) than does the EPRI-ARROTTA calculation. This almost constant temperature offset will provide increased doppler cross-section feedback to the NEM/TRAC calculation. This feedback, in turn, contributes significantly to its lower (and slightly earlier) calculated peak reactor power level.

Other quantities of interest at the 0.09 second power peak are the assembly normalized power distribution and the axial dependence of the node-average fuel temperature in the average and hottest assemblies. The axially averaged assembly power distributions produced by NEM/TRAC and EPRI-ARROTTA are displayed in Table 6.4. Upon viewing this table, one immediately notices the significant power shift towards the peripheral assemblies. This power shift is especially pronounced in assembly X=5, Y=5, the location of the ejected control rod. At that time this assembly has replaced assembly X=5, Y=7 as the assembly with the highest power. When these two assembly power distributions are compared, one notices that differences between NEM/TRAC and EPRI-ARROTTA are less pronounced than they were at steady-state (Table 6.2).

Figure 6.11 shows the axial variation in the average-assembly fuel temperature at 0.09 seconds into the transient. One notices that these new curves have almost exactly the same shape as the steady-state curves in Figure 6.4, except that the peak

**Table 6.4 EPRI-ARROTTA and NEM/TRAC Axially Averaged Assembly Power Distributions and Percentage Differences for the S. Levy PWR Test Problem at 0.09 Seconds Into the Transient**

0.8889 0.8678(-2.38)	1.0204 1.0017(-1.83)	1.0808 1.0661(-1.36)	1.0369 1.0146(-2.15)					
0.6139 0.6095(-0.72)	1.0333 1.0274(-0.57)	1.0026 1.0047(0.21)	1.4320 1.4199(-0.85)	1.6631 1.6369(-1.58)	1.3308 1.2983(-2.44)			
0.8509 0.8575(0.77)	0.8519 0.8616(1.14)	1.1042 1.1090(0.44)	1.2223 1.2310(0.71)	1.5575 1.5520(-0.35)	1.7107 1.6886(-1.29)	1.3314 1.2983(-2.49)		
0.8112 0.8259(1.81)	0.9756 0.9878(1.25)	0.9523 0.9679(1.64)	1.2925 1.2996(0.55)	1.7774 1.7831(0.32)	1.5593 1.5520(-0.47)	1.6687 1.6369(-1.90)		
0.8610 0.8769(1.85)	0.7578 0.7744(2.19)	0.9250 0.9384(1.45)	0.9929 1.0073(1.45)	1.2934 1.2996(0.48)	1.2246 1.2310(0.52)	1.4380 1.4199(-1.26)	1.0428 1.0146(-2.70)	
0.6213 0.6422(3.37)	0.6978 0.7134(2.23)	0.5179 0.5272(1.80)	0.9261 0.9384(1.33)	0.9539 0.9679(1.46)	1.1069 1.1090(0.19)	1.0071 1.0047(-0.23)	1.0904 1.0661(-2.23)	
0.5081 0.5231(2.95)	0.5184 0.5333(2.88)	0.6993 0.7134(2.02)	0.7591 0.7744(2.02)	0.9766 0.9878(1.15)	0.8533 0.8616(0.97)	1.0359 1.0274(-0.82)	1.0241 1.0017(-2.18)	
0.2380 0.2455(3.14)	0.5092 0.5231(2.74)	0.6259 0.6422(2.60)	0.8621 0.8769(1.72)	0.8108 0.8259(1.86)	0.8525 0.8575(0.59)	0.6139 0.6095(-0.72)	0.8875 0.8678(-2.22)	

ARROTTA  
NEM/TRAC (% Difference)

is now about 200 K higher. Again NEM/TRAC yields higher average fuel temperatures than EPRI-ARROTTA, mostly as a result of NEM/TRAC's lower value of fuel thermal conductivity (Figure 6.5).

After a quick review of Table 6.4, one might expect the assembly at  $X=5$ ,  $Y=5$  to have the highest fuel temperatures in the core. Figure 6.12 shows that this is not so. In reality the maximum fuel temperature in assembly  $X=5$ ,  $Y=5$  is approximately 250 K lower than that in the average assembly. The explanation for this discrepancy is that fuel temperature is dependent upon the total integrated heat input, not the present power level. At 0.09 seconds, the transient is so short-lived that not nearly enough heat has been added to raise the fuel temperature in this assembly beyond its lower than average steady-state level. As a result of this phenomena, the assembly at position  $X=5$ ,  $Y=7$  has the core's highest fuel temperatures, in spite of its no longer having the highest power level. The present axial distribution of fuel temperature in this assembly can be seen in Figure 6.13. Again the curves in this figure look very much the same as those at steady-state (Figure 6.6), except with an almost 300 K higher peak. At some elevations EPRI-ARROTTA again has higher fuel temperatures than NEM/TRAC. As was the case at steady-state, this is the result of EPRI-ARROTTA's sustained higher power level in assembly  $X=5$ ,  $Y=7$ . (Tables 6.2 and 6.4).

Another point of discrepancy between the transient total core power calculated by EPRI-ARROTTA and NEM/TRAC begins to manifest itself after the prompt transient has subsided, i.e., beyond approximately 0.15 seconds. Here NEM/TRAC begins to yield increasingly smaller transient total core power levels than EPRI-ARROTTA. This difference in power levels eventually builds to a level of almost 8.5% at 0.30 seconds. One of the most obvious sources of this discrepancy is a variation in the amount of cross-section feedback calculated by the two codes. One can test this hypothesis by

Figure 6.11 EPRI-ARROTTA and NEM/TRAC Transient Average Fuel Temperature as a Function of Height at 0.09 Seconds

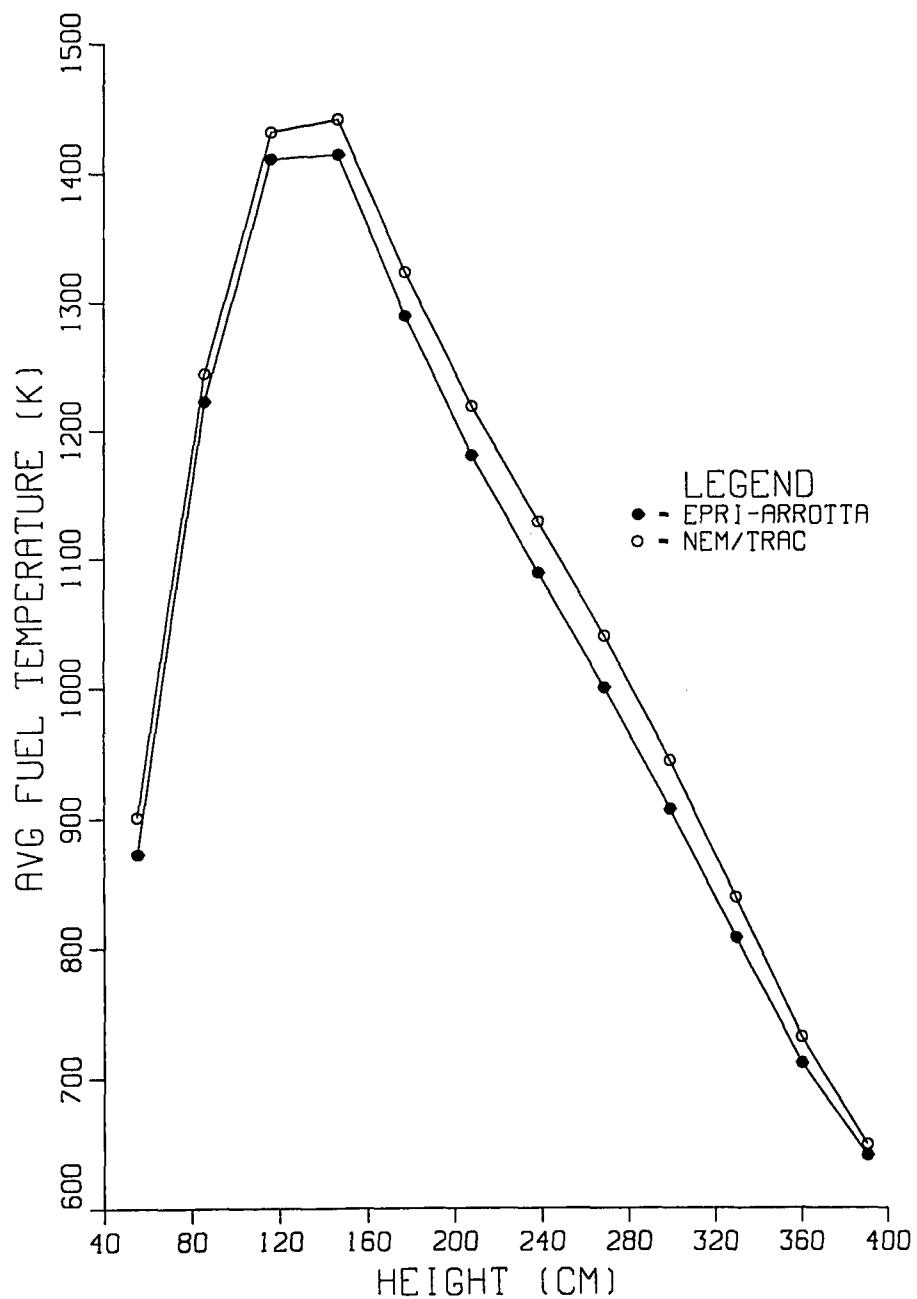


Figure 6.12 EPRI-ARROTTA and NEM/TRAC Transient Assembly X=5, Y=5 Fuel Temperature as a Function of Height at 0.09 Seconds

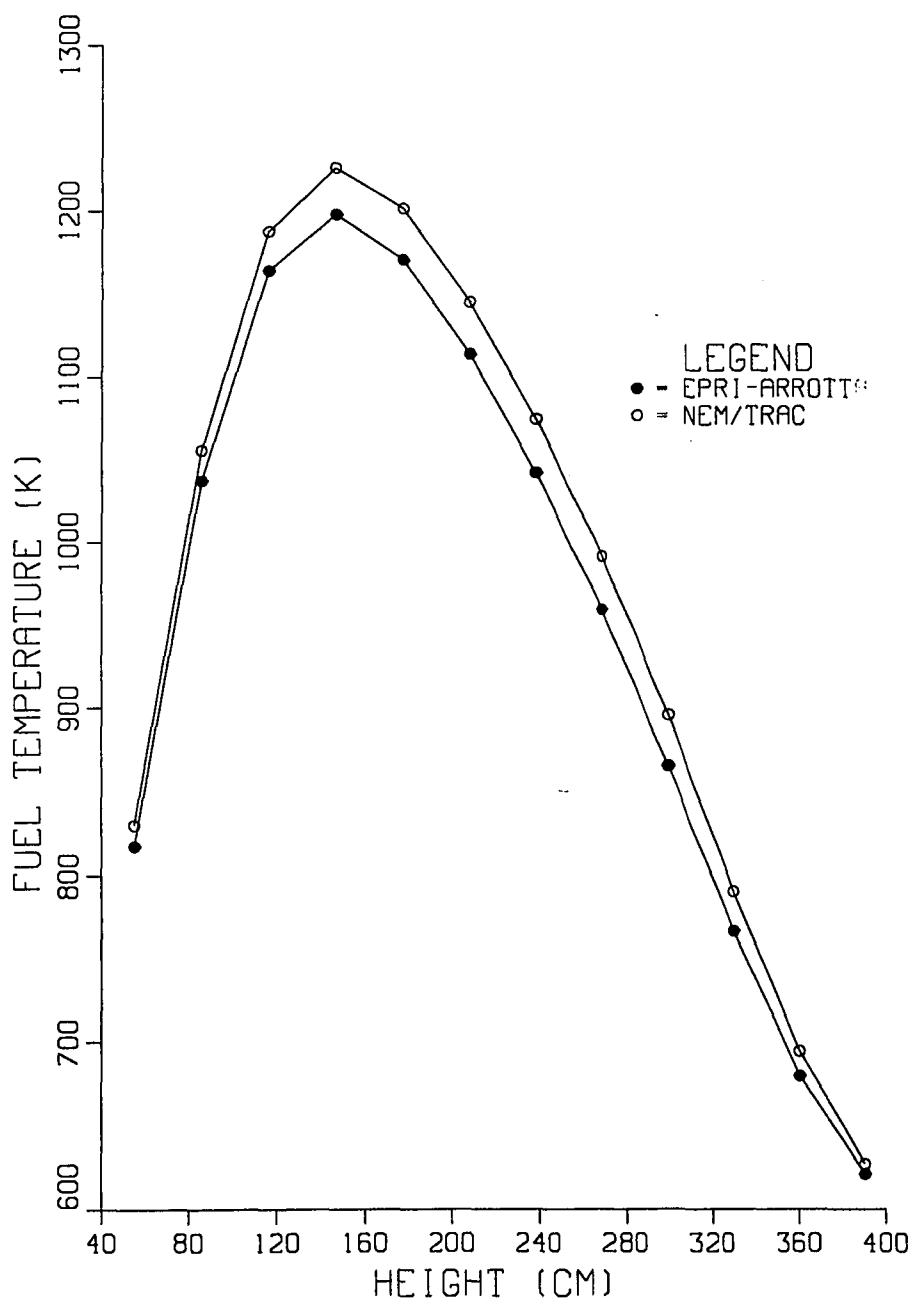
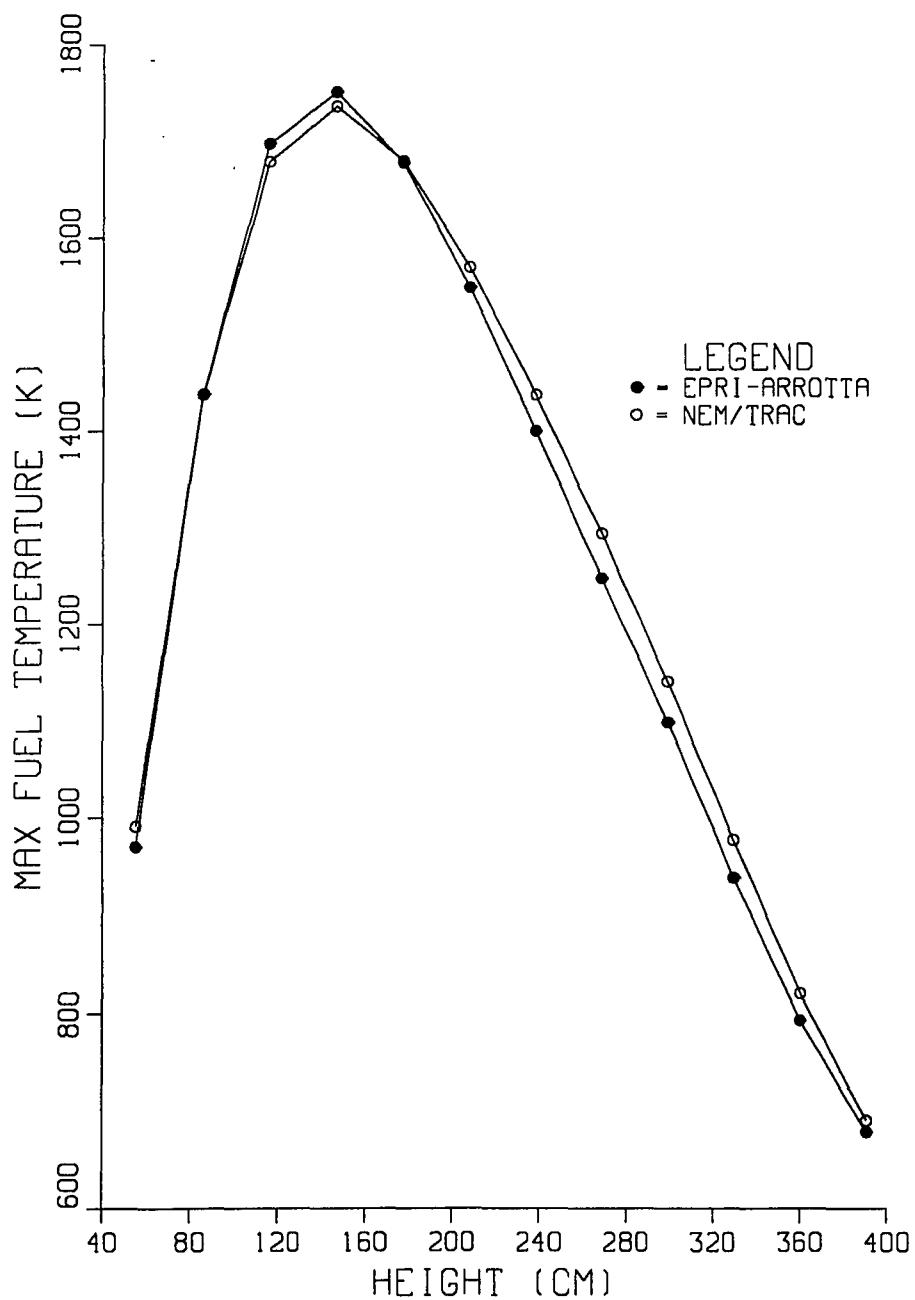


Figure 6.13 EPRI-ARROTTA and NEM/TRAC Transient Assembly X=5, Y=7 Fuel Temperature as a Function of Height at 0.09 Seconds



reviewing Figures 6.8 and 6.10. One immediately notices from these figures that both core-average fuel temperature and core-average moderator temperature are at all times higher in the NEM/TRAC calculation than in the EPRI-ARROTTA calculation. These consistently higher fuel and moderator temperatures are indeed the most likely cause of NEM/TRAC's lower predicted power levels. This difference in power levels, however, should begin to narrow at times beyond 0.30 seconds as NEM/TRAC's consistently lower power level begins to eventually result in lower fuel and moderator temperatures. These lower temperatures will then interact through the cross-section feedback mechanism to reduce the rate of power decay.

Other quantities of interest at 0.30 seconds into the transient will now be observed. The first of these quantities is the assembly power distribution. This distribution is displayed in Table 6.5. Upon comparing this data with that at 0.09 seconds (Table 6.4), one immediately notices that the bulk of the power generation is shifting back towards the centermost assemblies. This is the result of thermal-hydraulic feedback beginning to lower the large powers in the outermost assemblies. One also notices in Table 6.5 that the NEM/TRAC and EPRI-ARROTTA power distributions disagree more than they did at either 0.09 seconds or at steady-state. This higher level of disagreement is again the result of differences in fuel and moderator temperatures between the two calculations. As has already been established, NEM/TRAC's generally higher fuel and moderator temperatures produce greater cross-section feedback than EPRI-ARROTTA. This effect is especially pronounced in the hotter outermost assemblies. As a result, NEM/TRAC's power distribution at 0.30 seconds is skewed more toward the center assemblies than is the case for EPRI-ARROTTA.

Other quantities of interest at 0.30 seconds include the axial fuel temperature distributions in the average and hottest assemblies. The axial fuel temperature distribution

**Table 6.5 EPRI-ARROTTA and NEM/TRAC Axially Averaged Assembly Power Distributions and Percentage Differences for the S. Levy PWR Test Problem at 0.30 Seconds Into the Transient**

0.9119 0.8923(-2.15)	1.0314 1.0118(-1.90)	1.0700 1.0486(-2.00)	1.0048 0.9756(-2.91)				
0.6415 0.6437(0.34)	1.0509 1.0457(-0.49)	0.9930 0.9908(-0.23)	1.3741 1.3474(-1.94)	1.5522 1.5039(-3.11)	1.2325 1.1826(-4.05)		
0.8872 0.9004(1.48)	0.8761 0.8903(1.62)	1.1028 1.1055(0.24)	1.1819 1.1804(-0.13)	1.4652 1.4370(-1.93)	1.5813 1.5317(-3.14)	1.2325 1.1826(-4.05)	
0.8542 0.8767(2.64)	1.0150 1.0335(1.83)	0.9652 0.9837(1.92)	1.2667 1.2670(0.02)	1.7005 1.6854(-0.89)	1.4660 1.4370(-1.98)	1.5543 1.5039(-3.24)	
0.9203 0.9467(2.87)	0.8038 0.8302(3.29)	0.9565 0.9775(2.19)	0.9965 1.0131(1.67)	1.2671 1.2670(0.00)	1.1827 1.1804(-0.20)	1.3770 1.3474(-2.15)	1.0081 0.9756(-3.22)
0.6778 0.7108(4.87)	0.7570 0.7856(3.78)	0.5562 0.5763(3.62)	0.9572 0.9775(2.13)	0.9660 0.9837(1.83)	1.1038 1.1055(0.16)	0.9949 0.9908(-0.41)	1.0756 1.0486(-2.51)
0.5630 0.5898(4.75)	0.5715 0.5985(4.73)	0.7583 0.7856(3.60)	0.8042 0.8302(3.24)	1.0146 1.0335(1.86)	0.8760 0.8903(1.64)	1.0508 1.0457(-0.48)	1.0307 1.0118(-1.84)
0.2661 0.2798(5.16)	0.5639 0.5898(4.60)	0.6821 0.7108(4.21)	0.9205 0.9467(2.84)	0.8525 0.8767(2.84)	0.8870 0.9004(1.51)	0.6396 0.6437(0.64)	0.9073 0.8923(-1.65)

ARROTTA  
NEM/TRAC (% Difference)

in the average assembly at 0.30 seconds appears in Figure 6.14. These distributions look quite similar to those at 0.09 seconds (Figure 6.11) except for an approximately 100 K higher peak. This upward shifting of the curve is the direct result of the continuing addition of heat to the fuel material at a rate faster than the rate of convective heat transfer. One also notices from Figure 6.14 that the NEM/TRAC and EPRI-ARROTTA average assembly fuel temperatures are now more closely aligned than they were at 0.09 seconds. This is a direct result of the EPRI-ARROTTA calculation producing slightly more power than the NEM/TRAC calculation in the time period from about 0.18 to 0.30 seconds. A glimpse of this behavior can also be seen in Figure 6.10 as the difference between the two core-average fuel temperatures begins to narrow in the time frame from 0.18 to 0.30 seconds.

One notices in Table 6.5 that the assembly where the control-rod pull occurs, ( $X=5$ ,  $Y=5$ ), is by far the highest powered assembly in the core. Since this assembly has had the highest power since before 0.09 seconds, one must expect it to now have the hottest fuel temperature. A review of Figure 6.15 quickly disproves this hypothesis. Again the fuel in this assembly remains slightly cooler than that in the average assembly. Not enough total heat energy has yet been added to this assembly to heat it above average.

The hottest fuel in the core remains in assembly  $X=5$ ,  $Y=7$ . As shown in Figure 6.16, the peak temperature of the fuel in this assembly has risen by over 200 K from that which existed at 0.09 seconds. EPRI-ARROTTA again predicts generally higher fuel temperatures in this assembly than NEM/TRAC. Again this is the result of EPRI-ARROTTA's consistently higher predicted assembly  $X=5$ ,  $Y=7$  power level (Tables 6.2, 6.4, and 6.5).

Finally, one can study the coolant outlet temperature distribution which exists at 0.30 seconds into the transient. These final coolant temperatures are displayed as

Figure 6.14 EPRI-ARROTTA and NEM/TRAC Transient Average Assembly Fuel Temperature as a Function of Height at 0.30 Seconds

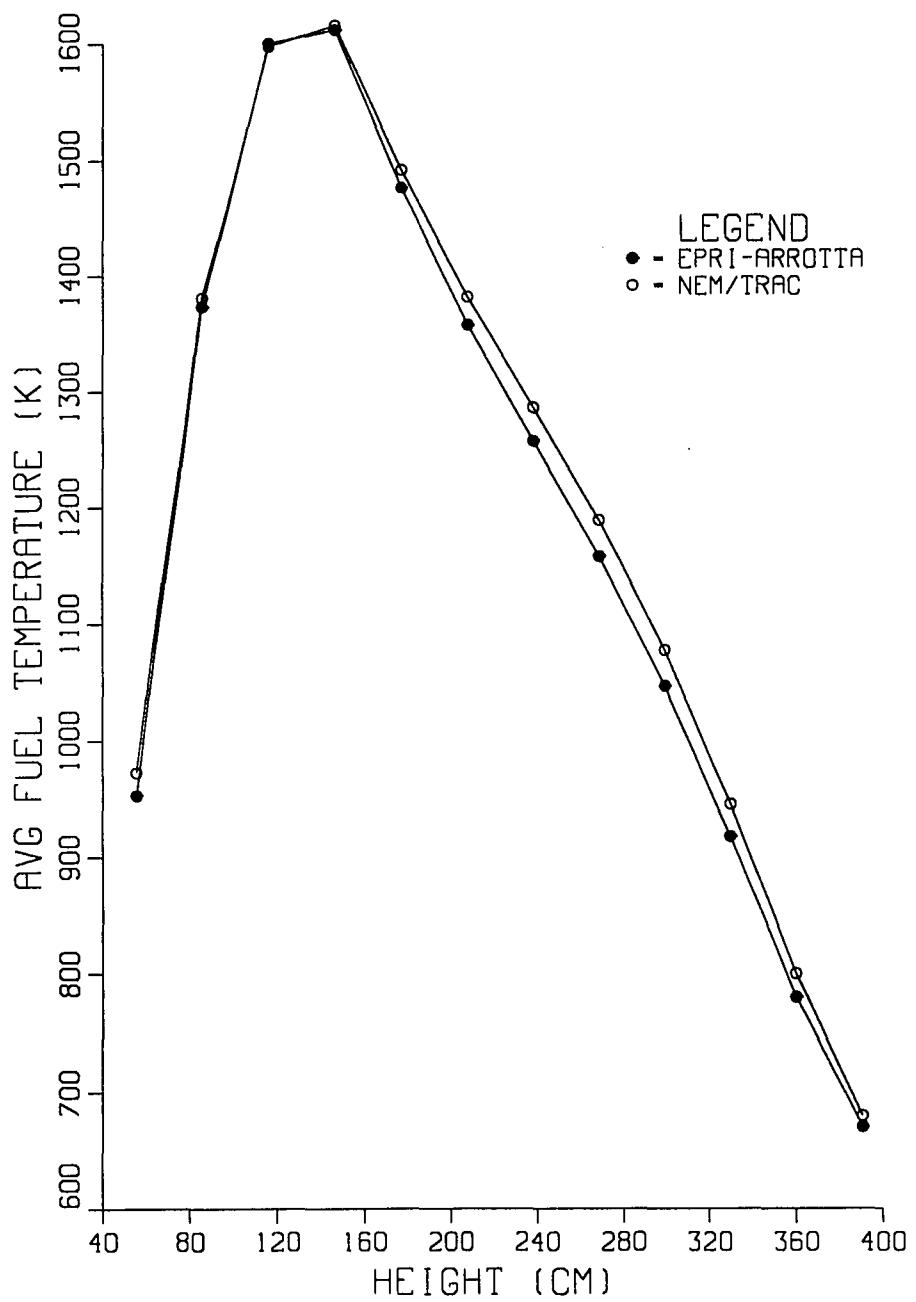


Figure 6.15 EPRI-ARROTTA and NEM/TRAC Transient Assembly X=5, Y=5 Fuel Temperature as a Function of Height at 0.30 Seconds

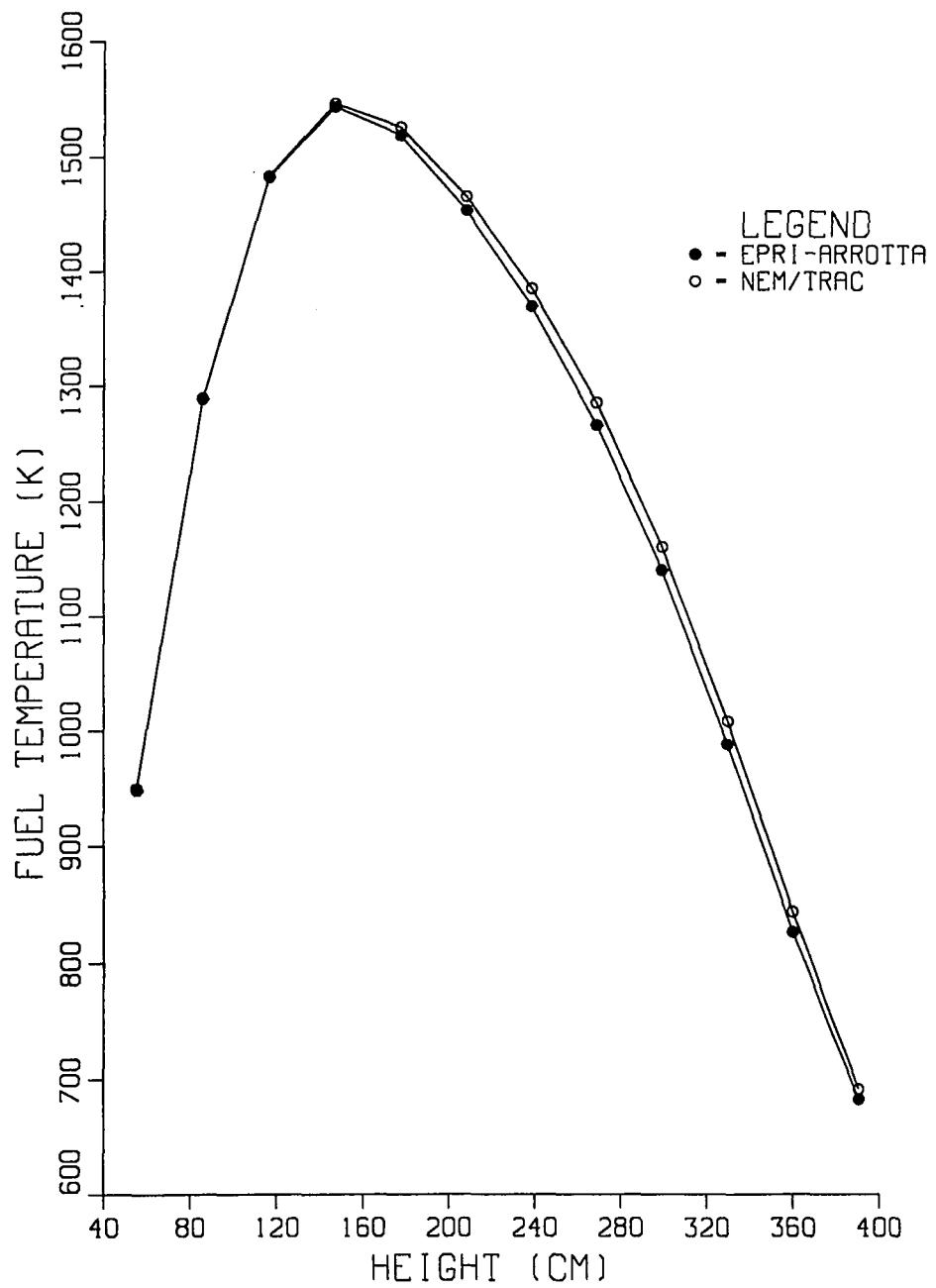
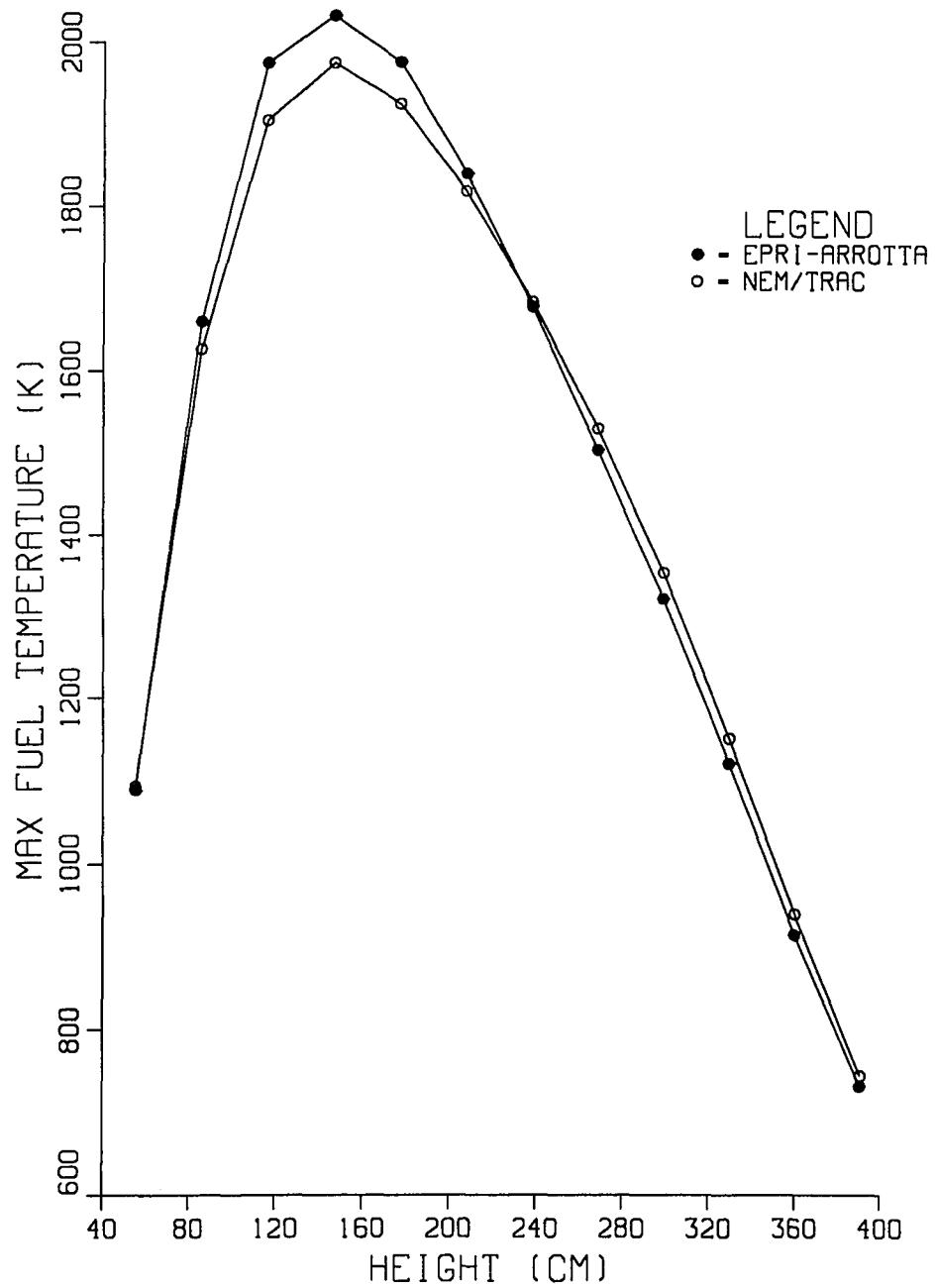


Figure 6.16 EPRI-ARROTTA and NEM/TRAC Transient Assembly X=5, Y=7 Fuel Temperature as a Function of Height at 0.30 Seconds



coolant temperature rises in Table 6.6 and Figure 6.17. One immediately notices upon comparing this figure and table with their steady-state counterparts (Figure 6.1 and Table 6.3) that, in general, only 0.5 to 7.0 K rises in outlet coolant temperature have occurred in any given assembly. One also notices that little shift has occurred in the general assembly coolant-outlet temperature distribution. Both of these observations are the result of the relatively short time extent of the transient. In the approximately 0.15 seconds since the power transient, the fuel and coolant temperature distributions have had little time to equilibrate, meaning that the outlet coolant temperatures still reflect their steady-state conditions to a large degree. One also notices at 0.30 seconds that neither the EPRI-ARROTTA nor the NEM/TRAC outlet coolant temperatures have come anywhere near the saturation temperature (618 K). No bulk boiling and very little subcooled boiling has yet occurred in even the hottest assemblies of the core.

## 6.5 Conclusions

The EPRI-ARROTTA and NEM/TRAC calculations of the S. Levy sample problem have yielded similar results. They have predicted the same transient core power in a severe reactivity excursion with few discrepancies. These discrepancies, most notably NEM/TRAC's lower predicted powers at the power peak (0.09 seconds) and at quasi-equilibrium (0.30 seconds), are the result of NEM/TRAC's generally higher predicted fuel temperatures. These higher fuel temperatures, which result in greater cross-section feedback, are in turn the result of NEM/TRAC using lower values of fuel thermal conductivity than EPRI-ARROTTA. The higher fuel temperatures are also in part due to EPRI-ARROTTA using a single volume-averaged fuel-temperature dependent value of thermal conductivity for the entire radial extent of the fuel pellet.

**Table 6.6 EPRI-ARROTTA and NEM/TRAC Transient Coolant Temperature Rises (K) and Percentage Differences at 0.30 Seconds**

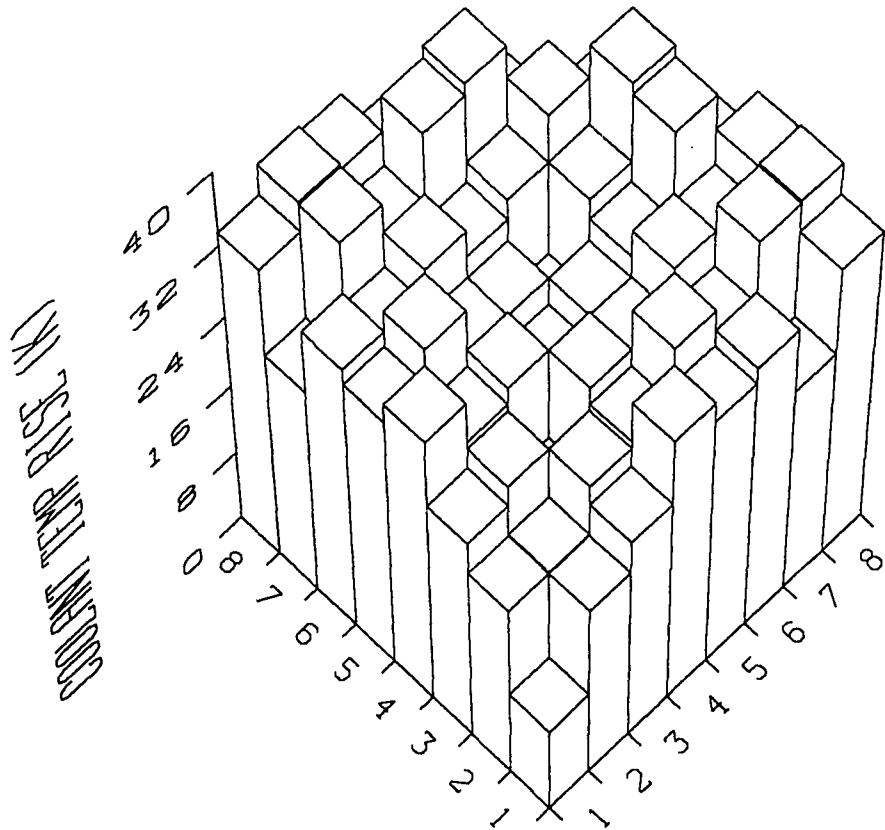
-Inlet Temperature = 564.8 K

-Saturation Temperature = 618.0 K (53.15 K rise)

33.45 32.54(-2.72)	36.69 35.92(-2.09)	36.11 35.74(-1.02)	32.10 31.75(-1.09)					
23.84 23.99(0.63)	36.41 36.12(-0.79)	32.46 32.75(0.87)	39.89 40.14(0.61)	41.13 41.49(0.86)	32.50 32.55(0.14)			
30.82 31.06(0.78)	29.58 29.95(1.26)	33.42 33.85(1.28)	30.61 31.37(2.49)	32.75 33.64(2.73)	37.80 38.45(1.72)	32.51 32.55(0.10)		
29.15 29.74(2.04)	33.05 33.59(1.65)	28.30 29.08(2.75)	28.26 29.17(3.22)	22.64 23.80(5.13)	32.78 33.64(2.63)	41.18 41.49(0.74)		
31.25 31.88(2.03)	26.82 27.51(2.59)	28.66 29.39(2.54)	25.32 26.12(3.16)	28.28 29.17(3.16)	30.65 31.37(2.36)	40.00 40.14(0.35)	32.22 31.75(-1.48)	
23.86 24.62(3.21)	26.06 26.72(2.56)	18.54 19.30(4.14)	28.69 29.39(2.44)	28.35 29.08(2.57)	33.47 33.85(1.13)	32.56 32.75(0.58)	36.32 35.74(-1.61)	
20.29 20.77(2.38)	20.42 20.96(2.61)	26.10 26.72(2.38)	26.85 27.51(2.46)	33.07 33.59(1.58)	29.60 29.95(1.18)	36.44 36.12(-0.85)	36.69 35.92(-2.10)	
9.88 9.92(0.45)	20.31 20.77(2.24)	24.00 24.62(2.59)	31.28 31.88(1.92)	29.14 29.74(2.08)	30.84 31.06(0.70)	23.80 23.99(0.82)	33.31 32.54(-2.32)	

ARROTTA  
NEM/TRAC (% Difference)

Figure 6.17 EPRI-ARROTTA Transient Outlet Coolant Temperature Rise at 0.30 Seconds



## **CHAPTER 7**

### **SUMMARY AND CONCLUSIONS**

#### **7.1 Overview of the Investigation**

The objective of this research project was to improve the present state of the art in computational methods for studying short term transient scenarios in water-cooled reactor systems.

In Chapter 1, the features and shortcomings of several present generation reactor transient analysis codes were studied. Here it was concluded that the TRAC code, in both pressurized and boiling water reactor versions, employed the most accurate and up-to-date thermal-hydraulic modelling capability. In addition, it was concluded that RAMONA-3B contained the best transient neutronics modelling capability with a fully three-dimensional, space-time neutronics algorithm. At this point, it was decided to combine the superior thermal-hydraulic capability of TRAC with the three-dimensional neutronic capability of RAMONA-3B. It was determined that the most expedient way of accomplishing this task would be to add an efficient three-dimensional neutron diffusion kinetics capability to the existing TRAC-PWR (TRAC-PF1/MOD2) code.

In Chapter 2, a variety of transient neutronics analysis techniques were examined. Of the methods for numerically solving the spatially dependent transient neutron diffusion equations in a homogenized reactor environment, it was determined that modern nodal methods provided the optimum trade-off between accuracy and efficiency. Specifically, the Nodal Expansion Method was chosen as the optimum means of providing TRAC-PF1/MOD2 with a multidimensional transient neutronics capability.

Chapter 3 presented a detailed derivation of the steady-state Nodal Expansion Method in both Cartesian and cylindrical geometries.

The results of using this just derived Nodal Expansion Method algorithm to solve for the two-energy-group flux distribution in a variety of two- and three-dimensional static reactor benchmark problems were presented in Chapter 4. Here, the Nodal Expansion Method was proven to be an accurate and efficient means of solving the multidimensional, two-group, static, neutron diffusion equations. It was shown that if homogenized cross-sections are available for assembly sized regions in a LWR (approximately 20 X 20 X 20 cm), the fourth-order, quadratic leakage, Nodal Expansion Method can be expected to yield assembly averaged power distributions with a maximum error of approximately 2%. This level of accuracy was achieved using approximately two orders of magnitude fewer mesh points than a similarly accurate standard finite difference solution. This large relaxation in mesh spacing requirements translates into an almost two order of magnitude increase in computational efficiency.

In Chapter 5, the transient version of the Nodal Expansion Method was derived and tested. Here, two time domain discretization methods were examined. In the first method, known as the backwards time-differencing technique, the time derivative terms in the two-group neutron diffusion equations were simply replaced by a first-order fully implicit finite difference approximation. In addition, the production terms in the neutron precursor equations were expressed in terms of a linear combination of old and new time fluxes. These auxiliary equations were then solved by integration over the extent of a single time step, a procedure known as the time-integrated method. This fully implicit transient solution technique was shown to be capable of yielding excellent results for several two- and three-dimensional sample problems, both with and without a thermal-hydraulic feedback mechanism. In addition, the accuracy of the method was shown to decrease only slightly as progressively larger time steps were used.

The second time domain discretization technique developed in Chapter 5 was the exponential time-differencing method. Here the space and time dependent flux in the neutron diffusion equations was replaced by a new space/time function multiplied by a time dependent exponential. This substitution was made in an effort to factor from the flux the majority of its time dependence. A time derivative finite difference approximation should in this case be more accurate than that performed with the first time discretization scheme. This is the result of the very weak time dependence of the new flux-like quantity. The neutron precursor equations in this new time-differencing technique were solved by using a simple first-order, fully implicit, finite-difference, time-derivative approximation. The results obtained with this exponential time-differencing model were disappointing. On a simple two-dimensional transient benchmark calculation without feedback, the exponential model yielded much worse results than the original backwards differenced model when large time steps were employed. As a result, the backwards differenced model with time-integrated precursor equations was chosen as the best model to use in conjunction with the TRAC-PF1/MOD2 thermal-hydraulic code.

In Chapter 6, the Nodal Expansion Method space-time neutronics coding was integrated into TRAC-PF1/MOD2. In order to test the operation of the combined NEM/TRAC coding, a Cartesian geometry control rod withdrawal transient was run on a one-quarter core symmetric Westinghouse PWR. The results of this calculation were compared to those obtained using the EPRI-ARROTTA code, a three-dimensional coupled neutronic/ thermal-hydraulic code with core-only modelling capability. The agreement between the NEM/TRAC and EPRI-ARROTTA calculations was very good, proving the inherent workability of a three-dimensional neutronics module in TRAC-PF1/MOD2.

## 7.2 Recommendations for Future Work

During the course of this research, several potential problems of interest have been left either unresolved or untouched. Many of these items may warrant further investigation. This section describes these potential future research areas.

(1) First, the Nodal Expansion Method multidimensional neutronics coding must be fully integrated into TRAC-PF1/MOD2. That is, the NEM coding must first be made to communicate directly with the normal TRAC input, output, restart, and graphics files. Secondly, the significant additional array storage required by the new neutronics routines must be integrated into the present TRAC-PF1/MOD2 variably dimensioned data structure. Next, generally applicable control rod movement and thermal-hydraulic cross-section feedback algorithms must be developed and incorporated into the existing TRAC file and array-storage structure.

The final code resulting from this integration should be applicable to a wide variety of problems. That is, the user should be able to easily select either a 0, 1, 2, or 3 dimensional neutron kinetics model in either Cartesian or cylindrical geometries. In addition, the user should be able to easily assign vessel thermal-hydraulic nodes and heat structures (fuel rods) to equivalent neutronic nodes in order to properly account for cross-section feedback and reactor power profiles. Finally, the detailed multidimensional cross-section information required by the chosen feedback mechanism should be made compatible with the standard output of existing neutron cross-section processing codes.

(2) Secondly, a variety of improvements could be made to the Nodal Expansion Method transient neutronic algorithm itself. One might, for instance, add a new geometry option to the NEM coding, i.e. hexagonal-Z geometry. Next, one might decide to increase the number of allowable neutron energy groups to greater than two. This

option would prove useful in calculations involving fast and intermediate spectrum reactors, such as those cooled by gas or liquid metals.

One might also want to introduce some transport-theory corrections into the diffusion-theory solution produced by the Nodal Expansion Method. One of the most straight forward ways of introducing these transport-theory corrections is to incorporate the idea of assembly discontinuity factors into the basic NEM algorithm. These factors, first introduced by K. S. Smith (46), fully account for the differences between diffusion and transport theory within any node of a given calculation. These transport corrections may prove very important in core regions where diffusion theory is inaccurate, such as within or near highly voided or highly absorbing media.

Some means of accounting for the true distribution of prompt gamma-ray heating might also be included in the NEM/TRAC algorithm. Approximately 10% of the recoverable heat energy in a reactor system results from the interaction of prompt (fission and neutron capture) gamma rays with the reactor internal structure. Some of these interactions may occur at significant distances from the fission site which either directly or indirectly led to the formation of the gamma ray. As a result, the true nodal power distribution may be somewhat different from that presently obtained by assuming all energy is deposited at the fission sites. The NEM generated nodal power distribution may, to a first order, be corrected to account for the true gamma-ray heating profile by a single off-line problem-specific gamma-ray transport calculation ( $S_n$  or Monte-Carlo). In this calculation, one would use the space and energy dependent fission gamma-ray source distribution and material configuration typical of the reactor being modelled to obtain a matrix which represents the fraction of gamma-ray power born in one specific neutronic node which gets deposited in each of its neighboring nodes. One would then

use this matrix to modify the calculated nodal power distribution at all times in a given transient calculation.

Some consideration must also be given to improving the speed of the transient NEM algorithm. There are several possible ways of improving the NEM algorithm's speed. One is to adopt some form of the quasi-static method (8). If a form of the quasi-static method were adopted, one could then update the spatially dependent power distribution much less frequently, resulting in a considerable savings in CPU time. Also, other more modern iterative-acceleration techniques could be tried. These techniques basically include variations of the Chebychev acceleration techniques (47). Finally, some of the FORTRAN for the NEM algorithm could be rewritten to enhance its vectorization and parallel-processing potential, which will in turn lead to some degree of computational speed-up.

Finally, one should develop some means for significantly reducing the variable storage requirements of the Nodal Expansion Method. For instance, one could rewrite the solution algorithm to save only outgoing nodal partial currents. Incoming partial currents could be produced at any time from the problem boundary conditions or from the outgoing partial currents in neighboring nodes. Likewise, the node and energy group dependent  $C$ ,  $B_1$ , and  $B_2$  matrices in Equation 3.33 could be re-evaluated whenever needed rather than being stored. Similar operations could be performed in many other portions of the NEM coding. The decision to implement any of these storage reduction measures will depend on whether or not the savings in storage costs outweighs any additional cost in CPU time necessitated by that change in the coding.

## **REFERENCES**

1. "The Experimental Boiling Water Reactor," ANL-5607, Argonne National Laboratory, (1957).
2. D. R. Evans, "The MOXY Core Heat Transfer Code: Model Description and Users Guide," PG-R-76-003, E. G. and G. Idaho Inc., (1976).
3. W. H. Rettig et al., "RELAP3: A Computer Program for Reactor Blowdown Analysis," USAEC IN-1321, Idaho Nuclear Corp, (1970).
4. K. V. Moore et al., "RETRAN: A Program for One-Dimensional Transient Thermal Hydraulic Analysis of Complex Fluid Flow Systems," CCM-5, Electric Power Research Institute, (1978).
5. "TRAC-P1: An Advanced Best-Estimate Computer Code for PWR LOCA Analysis: Volume 1; Methods, Models, User Information, and Programming Details," LA-7279-MS, Los Alamos National Laboratory, (1984).
6. J. Spore et al., "TRAC-BD1: An Advanced Best-Estimate Computer Program for Boiling Water Reactor Loss-of-Coolant Analysis," NUREG/CF-2178, E. G. and G. Idaho Inc., (1981).
7. "RETRAN-02: A Program for Transient Thermal Hydraulic Analysis of Complex Fluid Flow Systems," NP-1850-CCMA, Electric Power Research Institute, (1984).
8. K. O. Ott and D. A. Meneley, "Accuracy of the Quasistatic Treatment of Spatial Reactor Kinetics," Nucl. Sci. Eng. (36), p. 402 (1969).

9. V. H. Ransom et al., "RELAP5/MOD1 Code Manual. Volume 1: System Models and Numerical Methods," NUREG/CR-1826, EGG-2070, E. G. and G. Idaho Inc., (1982).
10. V. H. Ransom et al., "RELAP5/MOD2 Code Manual. Volume 1: Code Structure, System Models, and Solution Methods," EGG-SAAM-6377, E. G. and G. Idaho Inc., (1982).
11. "TRAC-PF1/MOD1: An Advanced Best-Estimate Computer Program for Pressurized Water Reactor Thermal Hydraulic Analysis," NUREG/CR-3858, Los Alamos National Laboratory, (1986).
12. W. L. Weaver et al., "TRAC-BF1 Manual: Extensions to TRAC-BF1/MOD1," NUREG/ CR-4391, E. G. and G. Idaho Inc., (1986).
13. K. S. Smith, "An Analytic Nodal Method for Solving the Two-Group, Multidimensional, Static and Transient Nodal Diffusion Equations," Thesis, Massachusetts Institute of Technology, Dept. of Nucl. Eng., (1979).
14. "A Modular Modelling System (MMS): A Code for the Dynamic Simulation of Fossile and Nuclear Power Plants," Babcock and Wilcox Inc., (1985).
15. W. Wulff et al., "A Description and Assessment of RAMONA-3B MOD.0 Cycle 4: A Computer Code with Three-Dimensional Neutron Kinetics for BWR System Transients," NUREG/CR-3664, Brookhaven National Laboratory, (1984).
16. H. Finnemann et al., "Interface Current Techniques for Multidimensional Reactor Calculations," Atomkerenergie (30), p. 123 (1977).

17. A. F. Henry and N. J. Curlee, "Verification of a Method for Treating Neutron Space-Time Problems," *Nucl. Sci. Eng.* (4), p. 727 (1958).
18. K. Ott and J. T. Madell, "Quasistatic Treatment of Spatial Phenomena in Reactor Dynamics," *Nucl. Sci. Eng.* (26), p. 563 (1966).
19. C. M. Kang and K. F. Hansen, "Finite Element Methods for Reactor Analysis," *Nucl. Sci. Eng.* (51), p. 456 (1973).
20. D. L. Delp et al., "FLARE, A Three-Dimensional Boiling Water Reactor Simulator," GEAP-4598, General Electric Company, (1964).
21. S. Borresen, "A Simplified, Coarse Mesh, Three-Dimensional Diffusion Scheme for Calculating the Gross Power Distribution in a Boiling Water Reactor," *Nucl. Sci. Eng.* (44), p. 37 (1972).
22. "SIMULATE-E: A Nodal Core Analysis Program for Light Water Reactors," NP-2792-CCM, Electric Power Research Institute, (1983).
23. R. Sims, "A Coarse Mesh Nodal Diffusion Method Based on Response Matrix Considerations," Thesis, Massachusetts Institute of Technology, Dept. of Nucl. Eng., (1977).
24. C. Maeder, "A Nodal Diffusion Method with Legendre Polynomials," ANS Topical Meeting on Advances in Reactor Physics, p. 121 (1978).
25. R. D. Lawrence, "A Nodal Green's Function Method for Multidimensional Neutron Diffusion Calculations," Thesis, University of Illinois, Nuclear Engineering Program, (1979).

26. S. Langenbuch, "Coarse-Mesh Flux Expansion Method for the Analysis of Space-Time Effects in Large Light Water Reactor Cores," *Nucl. Sci. Eng.* (63), p. 437 (1977).
27. H. Finnemann, "A Consistent Nodal Method for the Analysis of Space-Time Effects in Large LWR's," MRR145, Proc. of the Joint NEACRP/CSN Specialists Meeting, p. 131 (1975).
28. W. M. Stacy and C. H. Adams, "The Time-Integrated Method: A Quasistatic Model for Space-Time Neutronics," *Trans. Am. Nucl. Soc.* (10), p. 251 (1967).
29. M. R. Buckner and J. W. Stewart, "Multidimensional Space-Time Nuclear Reactor Kinetics Studies Part 1: Theoretical," *Nucl. Sci. Eng.* (59), p. 289 (1976).
30. J. J. Dongarra et al., "Linpack User's Guide," SIAM Philadelphia, (1979).
31. E. L. Wachpress, "Iterative Solution of Elliptic Systems," Prentice-Hall Inc., Englewood Cliffs, NJ., (1966).
32. G. D. Lassahn, "LOFT Experimental Uncertainty Analyses, Vol. 13, Temperature Measurements," NUREG/CR-0169, EGG-2037, (1982).
33. "Argonne Code Center: Benchmark Problem Book," ANL-7416, Suppl. 2, Argonne National Laboratory, (1977).
34. B. R. Bandini and A. J. Baratta, "Determination of the End State of the Three Mile Island Unit 2 Accident Using Neutron Transport Analysis," *Nucl. Tech.* (81), p. 370 (1988).
35. D. R. Vondy, T. B. Fowler, and G. W. Cunningham, "VENTURE: A Code Block for Solving Multigroup Neutronics Problems Applying the Finite Difference Diffusion

- Theory Approximation to Neutron Transport, Version II, ORNL-5062/R1, Oak Ridge National Laboratory, (1977).
36. S. Langenbuch, W. Maurer, and W. Werner, "Coarse-Mesh Nodal Diffusion Method for the Analysis of Space-Time Effects in Large Light Water Reactors," Nucl. Sci. Eng. (63), p. 437 (1977).
  37. G. Pedroni and L. Pollachini, "Effective Few-Node Computational Model for R,  $\theta$ , Z Neutron Diffusion," Energia Nucleare (2), p. 54 (1985).
  38. L. A. Hageman and J. B. Yasinsky, "Comparison of Alternating Direction Time Differencing Methods with Other Implicit Methods for the Solution of the Neutron Group Diffusion Equations," Nucl. Sci. Eng. (38), p. 8 (1969).
  39. A. V. Vota, N. J. Curlee, Jr., and A. F. Henry, "WIGL3: A Program for the Steady-State and Transient Solution of the One-Dimensional, Two-Group, Space-Time Diffusion Equations Accounting for Temperature, Xenon, and Control Feedback," WAPD-TM-788, (1969).
  40. L. D. Eisenhart, "ARROTTA: Advanced Rapid Reactor Operational Transient Analysis Computer Code Documentation Package, Volume 1: Theory and Numerical Analysis, and Volume 2: User's Manual," Electric Power Research Institute, (1989).
  41. "BEAGL-02: A Computer Code for Calculating Rapid LWR Core Transients," EPRI NP-3243-CCM, Electric Power Research Institute, (1984).
  42. P. E. Rohan, S. G. Wagner, and S. E. Ritterbusch, "HERMITE: A Multidimensional Space-Time Kinetics Code for PWR Transients," CENPD-188-A, Combustion Engineering, Inc., (1976).

43. P. E. Rohan and S. G. Wagner, "ARROTTA-HERMITE Code Comparison," EPRI NP-6614, Electric Power Research Institute, (1989).
44. A. Dias, Personal Communication, (1989).
45. J. J. Dongarra, "Performance of Various Computers Using Standard Linear Equation Software in a Fortran Environment," Technical Memorandum 23, Argonne National Laboratory, (1989).
46. K. S. Smith, "Spatial Homogenization Methods for Light Water Reactors," Thesis, Massachusetts Institute of Technology, Dept. of Nucl. Eng., (1980).
47. A. F. Henry and H. C. Honenk, "The Physics of Water Reactors," AX-0890988; Computer Applications Technology, (1989).

## APPENDIX A

### RESPONSE MATRIX ELEMENTS

In this Appendix, explicit expressions for the elements of the three-dimensional Nodal Expansion Method response matrices will be presented.

The elements of the fourth-order Cartesian geometry response matrix equations (Eqn 3.33) can be expressed in terms of known quantities as follows:

$$a_1(c_1) = 1 + (-) \frac{32D_g^l}{\Delta x} \left( 1 + \frac{30D_g^l}{A_g^l \Delta x^2} \right) + \frac{120D_g^l}{A_g^l \Delta x^2} \left( 1 + \frac{7D_g^l}{A_g^l \Delta x^2} \right)$$

$$a_2(c_2) = + (-) \frac{8D_g^l}{\Delta x} \left( 1 + \frac{90D_g^l}{A_g^l \Delta x^2} \right) + \frac{60D_g^l}{A_g^l \Delta x^2} \left( 1 + \frac{14D_g^l}{A_g^l \Delta x^2} \right)$$

$a_5(c_5) \equiv$  same as  $a_1(c_1)$  with  $\Delta y$  in place of  $\Delta x$

$a_6(c_6) \equiv$  same as  $a_2(c_2)$  with  $\Delta y$  in place of  $\Delta x$

$a_9(c_9) \equiv$  same as  $a_1(c_1)$  with  $\Delta z$  in place of  $\Delta x$

$a_{10}(c_{10}) \equiv$  same as  $a_2(c_2)$  with  $\Delta z$  in place of  $\Delta x$

$$a_3 = c_3 = \frac{20D_g^l}{A_g^l \Delta x \Delta y} d_1$$

$$a_4 = c_4 = \frac{20D_g^l}{A_g^l \Delta x \Delta z} d_1$$

$$a_7 = c_7 = \frac{20D_g^l}{A_g^l \Delta x \Delta y} d_2$$

$$a_8 = c_8 = \frac{20D_g^l}{A_g^l \Delta y \Delta z} d_2$$

$$a_{11} = c_{11} = \frac{20D_g^l}{A_g^l \Delta x \Delta z} d_3$$

$$a_{12} = c_{12} = \frac{20D_g^l}{A_g^l \Delta y \Delta z} d_3$$

$$b_1 = \frac{20D_g^l}{A_g^l \Delta x} d_1$$

$$b_8 = \frac{20D_g^l}{A_g^l \Delta y} d_2$$

$$b_{13} = \frac{20D_g^l}{A_g^l \Delta z} d_3$$

$$b_2 = \frac{60D_g^l}{A_g^l \Delta x}$$

$$b_9 = \frac{60D_g^l}{A_g^l \Delta y}$$

$$b_{14} = \frac{60D_g^l}{A_g^l \Delta z}$$

$$b_3 = \frac{7}{3} b_1$$

$$b_{10} = \frac{7}{3} b_9$$

$$b_{15} = \frac{7}{3} b_{14}$$

$$b_4 = \frac{b_2}{\Delta y}$$

$$b_6 = \frac{b_2}{\Delta z}$$

$$b_{11} = \frac{b_9}{\Delta z}$$

$$b_5 = \frac{-b_3}{\Delta y}$$

$$b_7 = \frac{-b_3}{\Delta z}$$

$$b_{12} = \frac{-b_{10}}{\Delta z}$$

$$d_1 = 1 + \frac{42D_g^l}{A_g^l \Delta x^2}$$

$d_2 \equiv$  same as  $d_1$  with  $\Delta y$  in place of  $\Delta x$

$d_3 \equiv$  same as  $d_1$  with  $\Delta z$  in place of  $\Delta x$

The elements of the third-order  $r$ , fourth-order  $\theta$ - $z$  cylindrical geometry response matrix equations (Eqn 3.130) can be expressed in terms of known quantities as follows:

$$\Delta = \frac{3D_g^l}{A_g^l R \Delta r}$$

$$\begin{aligned}
a_1(c_1) &= 1 + (-) \frac{6D_g^l}{\Delta r} \left( 3 + \frac{20D_g^l}{A_g^l \Delta r^2} \right) + \frac{36D_g^l}{A_g^l \Delta r^2} \\
a_2(c_2) &= - (+) \frac{6D_g^l}{\Delta r} \left( 1 + \frac{20D_g^l}{A_g^l \Delta r^2} \right) - \frac{24D_g^l}{A_g^l \Delta r^2} \\
a_5(c_5) &= 1 + (-) \frac{32D_g^l}{R \Delta \theta} \left( 1 + \frac{30D_g^l}{A_g^l R^2 \Delta \theta^2} \right) + \frac{120D_g^l}{A_g^l R^2 \Delta \theta^2} \left( 1 + \frac{7D_g^l}{A_g^l R^2 \Delta \theta^2} \right) \\
a_6(c_6) &= + (-) \frac{8D_g^l}{R \Delta \theta} \left( 1 + \frac{90D_g^l}{A_g^l R^2 \Delta \theta^2} \right) + \frac{60D_g^l}{A_g^l R^2 \Delta \theta^2} \left( 1 + \frac{14D_g^l}{A_g^l R^2 \Delta \theta^2} \right) \\
a_9(c_9) &= 1 + (-) \frac{32D_g^l}{\Delta z} \left( 1 + \frac{30D_g^l}{A_g^l \Delta z^2} \right) + \frac{120D_g^l}{A_g^l \Delta z^2} \left( 1 + \frac{7D_g^l}{A_g^l \Delta z^2} \right) \\
a_{10}(c_{10}) &= + (-) \frac{8D_g^l}{\Delta z} \left( 1 + \frac{90D_g^l}{A_g^l \Delta z^2} \right) + \frac{60D_g^l}{A_g^l \Delta z^2} \left( 1 + \frac{14D_g^l}{A_g^l \Delta z^2} \right) \\
a_3 = c_3 &= \frac{6D_g^l}{A_g^l R \Delta r \Delta \theta} \\
a_4 = c_4 &= \frac{6D_g^l}{A_g^l \Delta r \Delta z} \\
a_{7R} = c_{7R} &= \frac{20D_g^l}{A_g^l R \Delta \theta} \left( \frac{1}{\Delta r} + \frac{1}{2R} \right) d_2 \\
a_{7L} = c_{7L} &= \frac{20D_g^l}{A_g^l R \Delta \theta} \left( \frac{1}{\Delta r} - \frac{1}{2R} \right) d_2 \\
a_8 = c_8 &= \frac{20D_g^l}{A_g^l R \Delta \theta \Delta z} d_2 \\
a_{11R} = c_{11R} &= \frac{20D_g^l}{A_g^l \Delta z} \left( \frac{1}{\Delta r} + \frac{1}{2R} \right) d_3 \\
a_{11L} = c_{11L} &= \frac{20D_g^l}{A_g^l \Delta z} \left( \frac{1}{\Delta r} - \frac{1}{2R} \right) d_3 \\
a_{12} = c_{12} &= \frac{20D_g^l}{A_g^l R \Delta \theta \Delta z} d_3 \\
b_1 &= \frac{6D_g^l}{A_g^l \Delta r} \\
b_8 &= \frac{20D_g^l}{A_g^l R \Delta \theta} d_2
\end{aligned}$$

$$b_{13} = \frac{20D_g^l}{A_g^l \Delta z} d_3$$

$$b_2 = \frac{60D_g^l}{A_g^l \Delta r}$$

$$b_3 = \frac{140D_g^l}{A_g^l \Delta r}$$

$$b_9 = \frac{60D_g^l}{A_g^l R \Delta \theta}$$

$$b_{14} = \frac{60D_g^l}{A_g^l \Delta z}$$

$$b_{10} = \frac{7}{3} b_9$$

$$b_{15} = \frac{7}{3} b_{14}$$

$$b_4 = \frac{b_2}{R \Delta \theta}$$

$$b_6 = \frac{b_2}{\Delta z}$$

$$b_{11} = \frac{b_9}{\Delta z}$$

$$b_5 = \frac{-b_3}{R \Delta \theta}$$

$$b_7 = \frac{-b_3}{\Delta z}$$

$$b_{12} = \frac{-b_{10}}{\Delta z}$$

$$d_2 = 1 + \frac{42D_g^l}{A_g^l R^2 \Delta \theta^2}$$

$$d_3 = 1 + \frac{42D_g^l}{A_g^l \Delta z^2}$$

The elements of the fourth-order  $r$  (1st approx.), fourth-order  $\theta$ - $z$  cylindrical geometry response matrix equations (Eqn 4.10) can be expressed in terms of known quantities as follows:

$$\Delta = \frac{10D_g^l}{A_g^l R \Delta r} d_1$$

$$\begin{aligned}
a_{1R}(c_{1R}) &= 1 + (-) \frac{4D_g^l}{\Delta r} \left( 8 + \frac{135D_g^l}{A_g^l \Delta r^2} \right) + \frac{60D_g^l}{A_g^l \Delta r^2} \left( 2 + \frac{7D_g^l}{A_g^l \Delta r^2} \right) + (-) \frac{70(D_g^l)^2}{A_g^l R \Delta r^2} \\
a_{1L}(c_{1L}) &= 1 + (-) \frac{4D_g^l}{\Delta r} \left( 8 + \frac{135D_g^l}{A_g^l \Delta r^2} \right) + \frac{60D_g^l}{A_g^l \Delta r^2} \left( 2 + \frac{7D_g^l}{A_g^l \Delta r^2} \right) - (+) \frac{70(D_g^l)^2}{A_g^l R \Delta r^2} \\
a_{2R}(c_{2R}) &= + (-) \frac{4D_g^l}{\Delta r} \left( 2 + \frac{75D_g^l}{A_g^l \Delta r^2} \right) + \frac{60D_g^l}{A_g^l \Delta r^2} \left( 1 + \frac{7D_g^l}{A_g^l \Delta r^2} \right) + (-) \frac{70(D_g^l)^2}{A_g^l R \Delta r^2} \\
a_{2L}(c_{2L}) &= + (-) \frac{4D_g^l}{\Delta r} \left( 2 + \frac{75D_g^l}{A_g^l \Delta r^2} \right) + \frac{60D_g^l}{A_g^l \Delta r^2} \left( 1 + \frac{7D_g^l}{A_g^l \Delta r^2} \right) - (+) \frac{70(D_g^l)^2}{A_g^l R \Delta r^2} \\
a_3 = c_3 &= \frac{20D_g^l}{A_g^l R \Delta r \Delta \theta} d_1 \\
a_4 = c_4 &= \frac{20D_g^l}{A_g^l \Delta r \Delta z} d_1 \\
b_1 &= \frac{20D_g^l}{A_g^l \Delta r} d_1 \\
d_1 &= 1 + \frac{21D_g^l}{A_g^l \Delta r^2}
\end{aligned}$$

All other elements in matrix Equation 4.10 are the same as those found above in Equation 3.130.

The elements of the fourth-order  $r$  (2nd approx.), fourth-order  $\theta$ -z cylindrical geometry response matrix equations (Eqn 4.14) can be expressed in terms of known quantities as follows:

$$\begin{aligned}
\Delta &= \frac{10D_g^l}{A_g^l R \Delta r} d_1 \\
a_1(c_1) &= 1 + (-) \frac{32D_g^l}{\Delta r} \left( 1 + \frac{30D_g^l}{A_g^l \Delta r^2} \right) + \frac{120D_g^l}{A_g^l \Delta r^2} \left( 1 + \frac{7D_g^l}{A_g^l \Delta r^2} \right) \\
a_2(c_2) &= + (-) \frac{4D_g^l}{\Delta r} \left( 1 + \frac{90D_g^l}{A_g^l \Delta r^2} \right) + \frac{60D_g^l}{A_g^l \Delta r^2} \left( 1 + \frac{14D_g^l}{A_g^l \Delta r^2} \right) \\
a_3 = c_3 &= \frac{20D_g^l}{A_g^l R \Delta r \Delta \theta} d_1
\end{aligned}$$

$$a_4 = c_4 = \frac{20D_g^l}{A_g^l \Delta r \Delta z} d_1$$

$$b_1 = \frac{20D_g^l}{A_g^l \Delta r} d_1$$

$$d_1 = 1 + \frac{42D_g^l}{A_g^l \Delta r^2}$$

All other elements in matrix equation 4.14 retain the same form as they had in Equation 3.130.

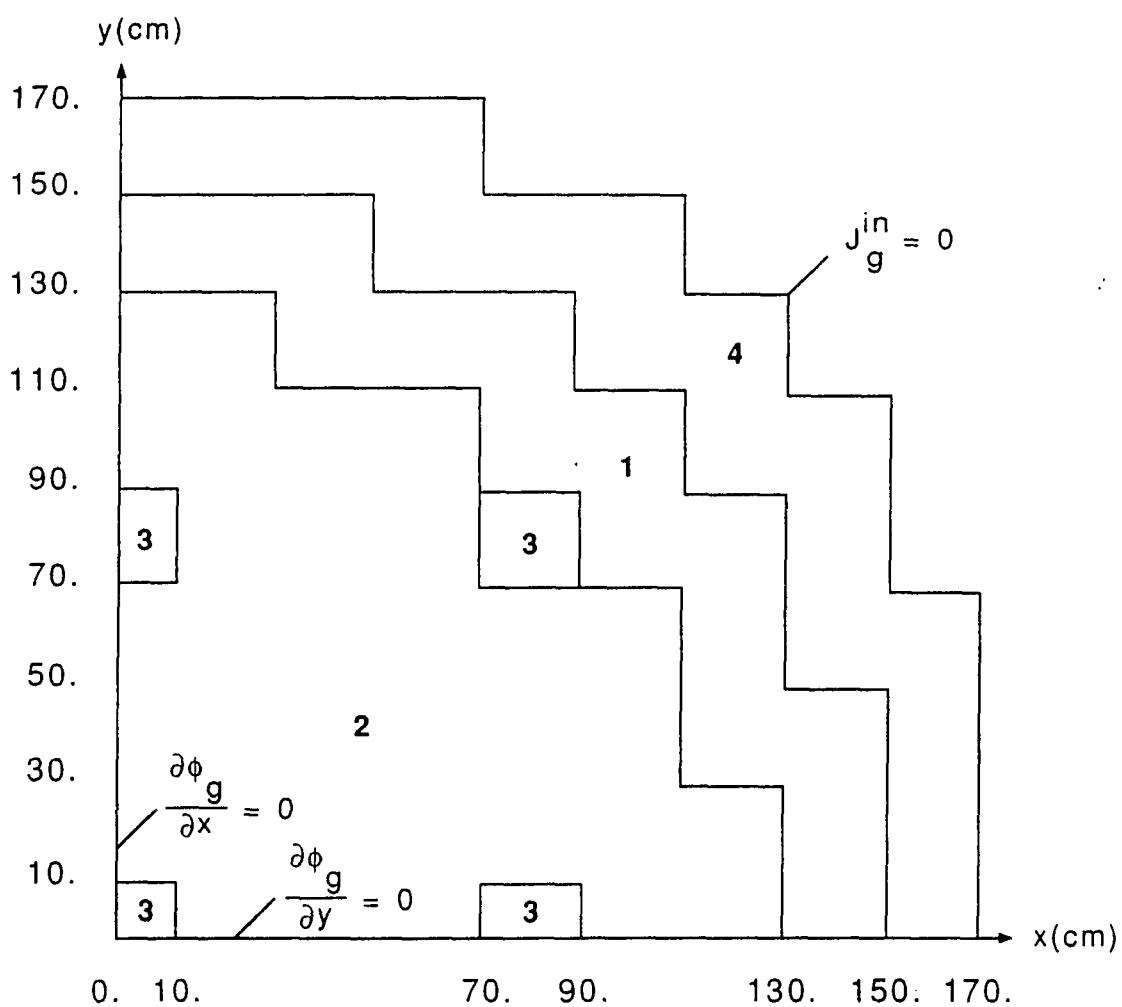
## **APPENDIX B**

### **DESCRIPTION OF BENCHMARK PROBLEMS**

- B.1    The IAEA PWR Steady-State Benchmark Problem**
- B.2    The LRA BWR Steady-State Benchmark Problem**
- B.3    The LMW LWR Transient Benchmark Problem**
- B.4    Dodds' R-Z Geometry Transient Benchmark Problem**
- B.5    Pedroni's R- $\theta$  Geometry Steady-State Benchmark Problem**
- B.6    The TWIGL Seed-Blanket Transient Benchmark Problem**
- B.7    The S. Levy PWR Transient Test Problem**

### B.1 The IAEA PWR Steady-State Benchmark Problem

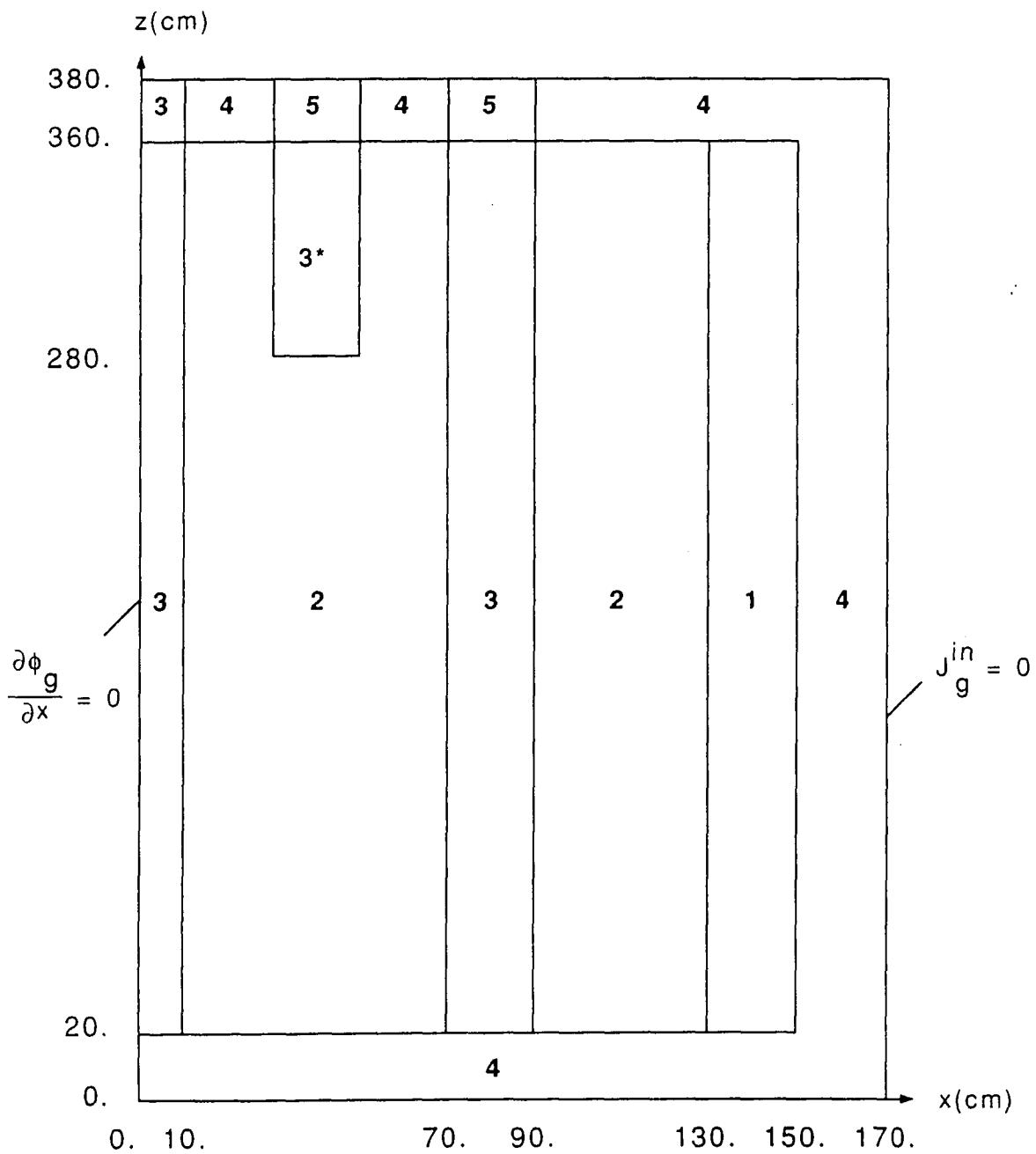
Quadrant of Reactor Horizontal Cross Section at Axial Midplane



Vertical Reactor Cross Section at  $y = 0$ 

\* Position of partially inserted rod:

$$30.0 \leq x \leq 50.0, 30.0 \leq y \leq 50.0$$



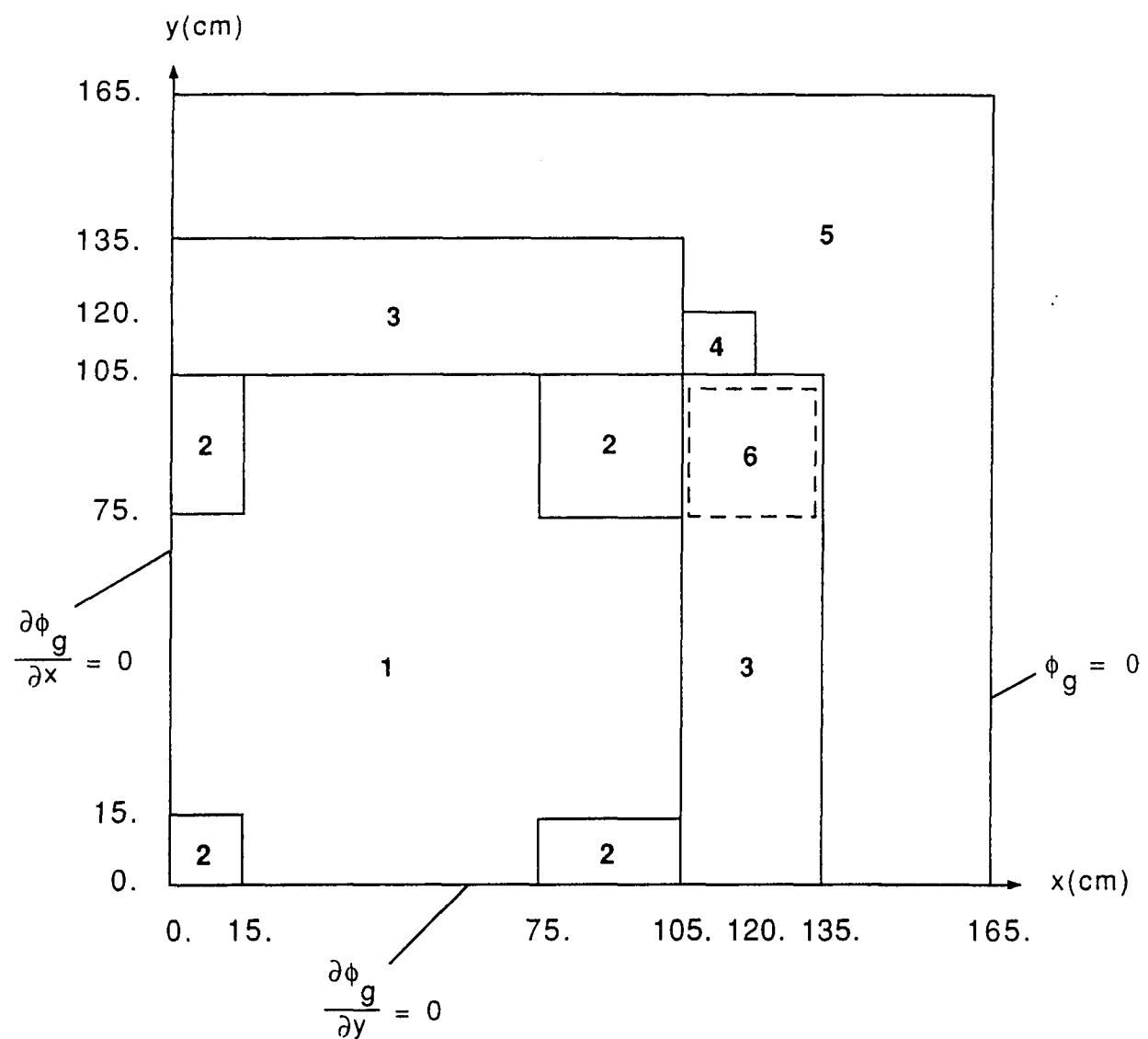
**Material Constants:**

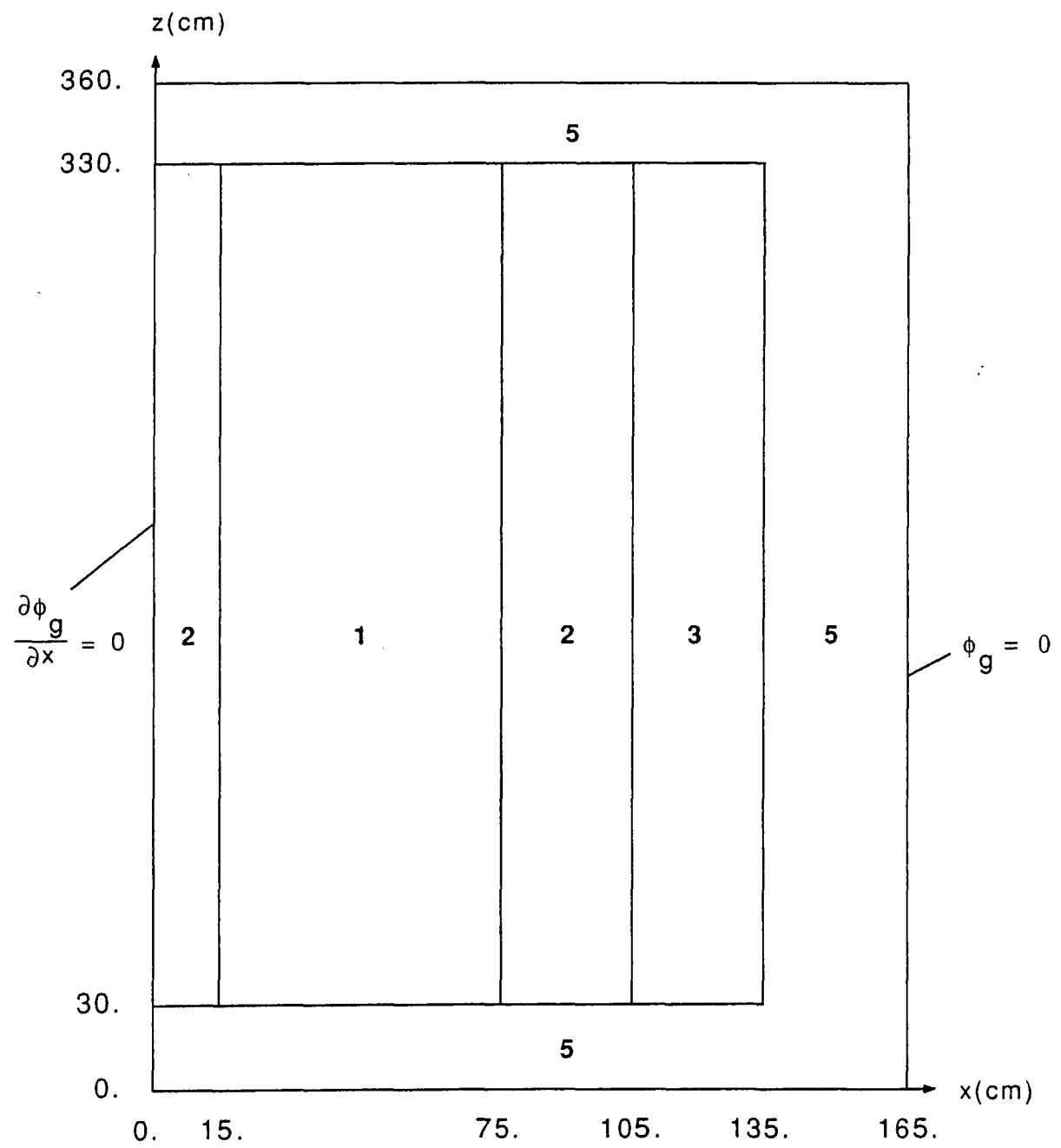
<b>Composition</b>	<b>Group</b>	$D_g$	$\Sigma_{ag}$	$\nu \Sigma_{fg}$	$\Sigma_{12}$
1	1	1.5	0.01	0.0	
	2	0.4	0.08	0.135	0.02
2	1	1.5	0.01	0.0	
	2	0.4	0.085	0.135	0.02
3	1	1.5	0.01	0.0	
	2	0.4	0.13	0.135	0.02
4	1	2.0	0.0	0.0	
	2	0.3	0.01	0.0	0.04
5	1	2.0	0.0	0.0	
	2	0.3	0.055	0.0	0.04

$$B_z^2 = 8.0 \times 10^{-5} \text{ for all compositions and groups}$$

## B.2 The LRA BWR Steady-State Benchmark Problem

Quadrant of Reactor Horizontal Cross Section at Axial Midplane



Vertical Reactor Cross Section at  $y = 0$ 

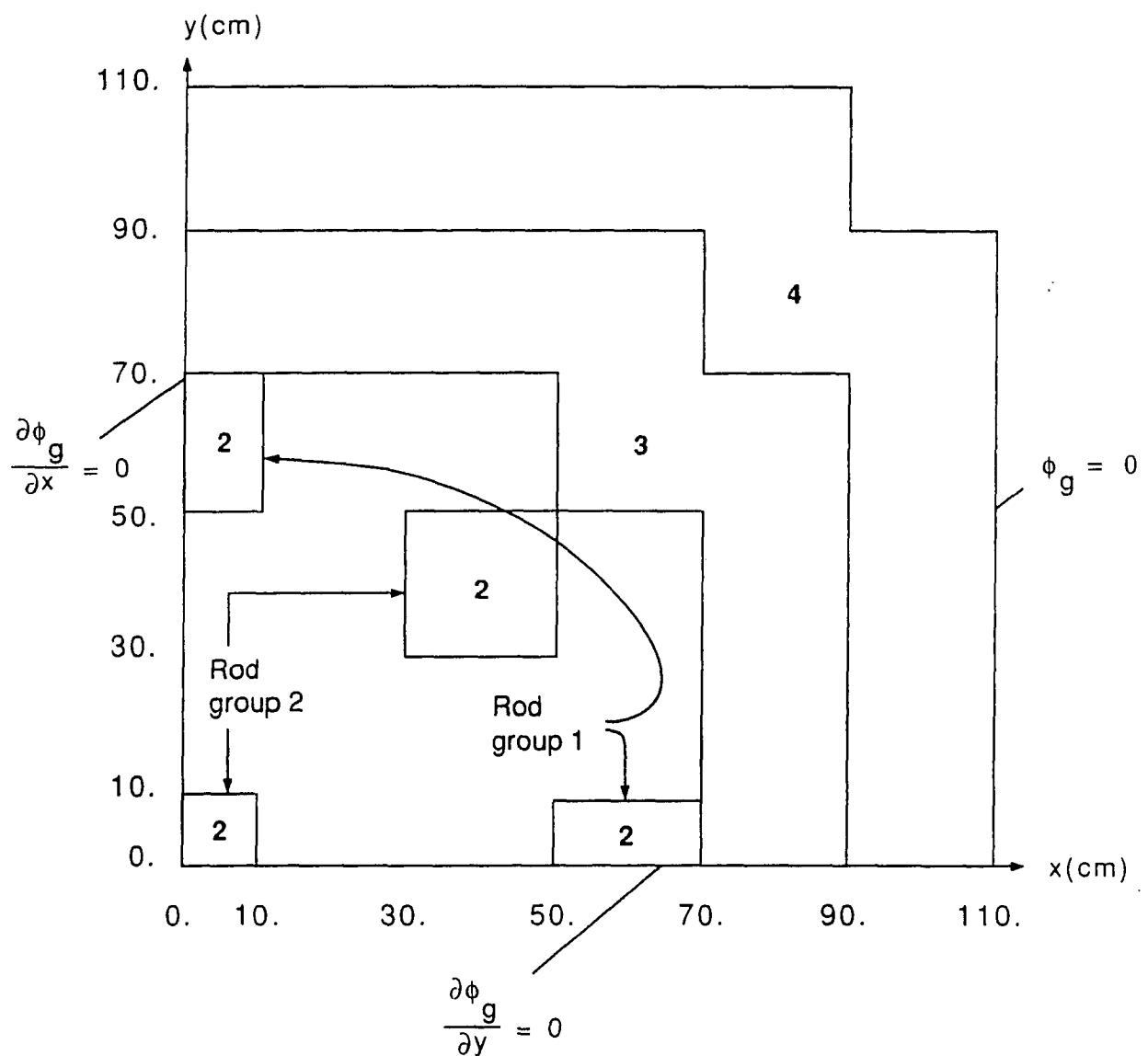
## Material Constants:

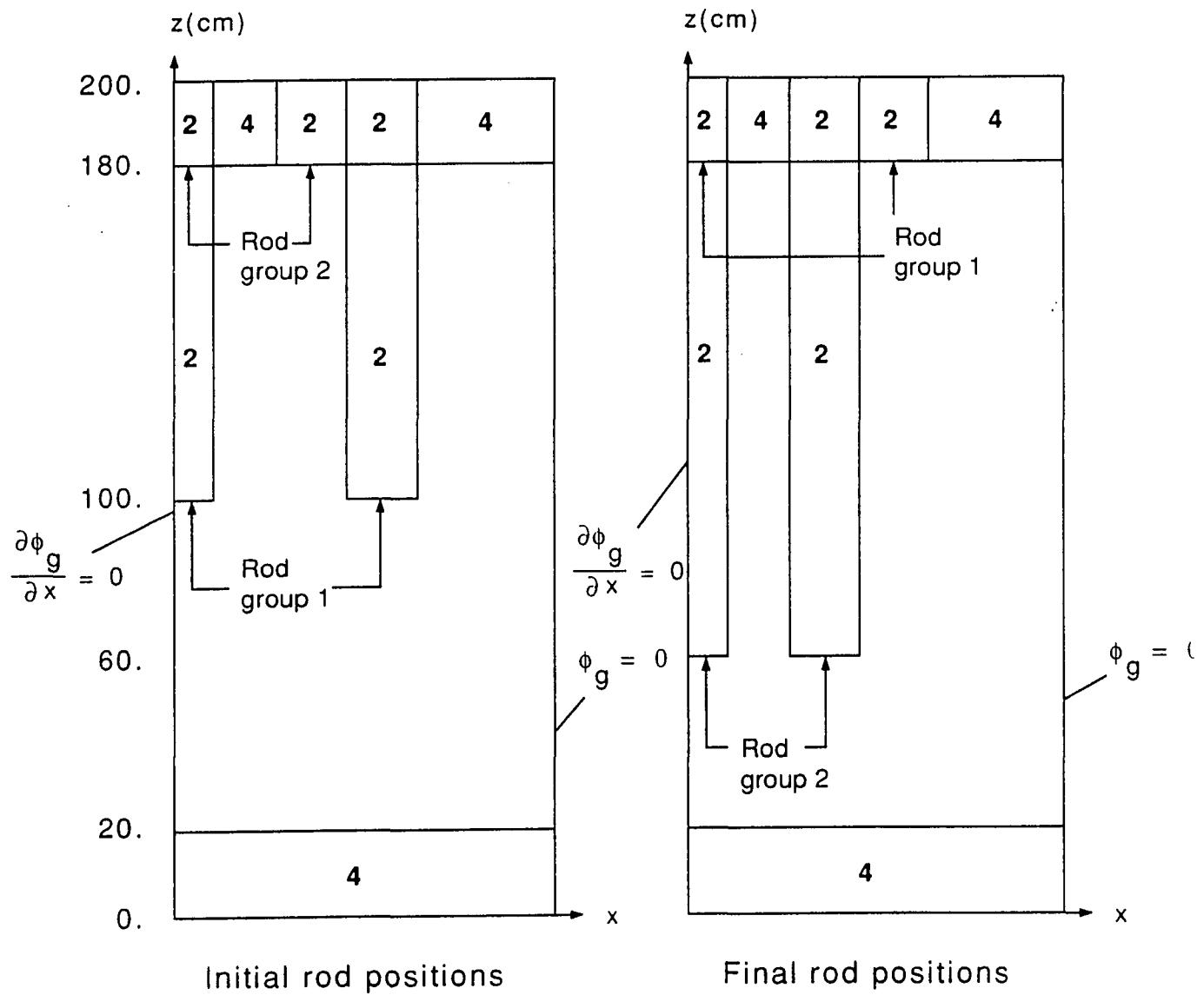
Composition	Group	$D_g$	$\Sigma_{ag}$	$\nu \Sigma_{fg}$	$\Sigma_{12}$
1	1	1.255	0.008252	0.004602	
	2	0.211	0.1003	0.1091	0.02533
2	1	1.268	0.007181	0.004609	
	2	0.1902	0.07047	0.08675	0.02767
3	1	1.259	0.008002	0.004663	
	2	0.2091	0.08344	0.1021	0.02617
4	1	1.259	0.008002	0.004663	
	2	0.2091	0.073324	0.1021	0.02617
5	1	1.257	0.0006034	0.0	
	2	0.1592	0.01911	0.0	0.04754
6	1	1.259	0.008002	0.004663	
	2	0.2091	0.08344	0.1021	0.02617

$B_z^2 = 1.0 \times 10^{-4}$  for all compositions and groups

### B.3 The LMW LWR Transient Benchmark Problem

Quadrant of Reactor Horizontal Cross Section at Axial Midplane



Vertical Reactor Cross Section at  $y = 0$ 

## Material Constants:

Composition	Group	$D_g$	$\Sigma_{ag}$	$\nu \Sigma_{fg}$	$\Sigma_{12}$
1	1	1.423913	0.01040206	0.006477691	
	2	0.356306	0.08766217	0.1127328	0.0175555
2	1	1.423913	0.01095206	0.006477691	
	2	0.356306	0.09146217	0.1127328	0.0175555
3	1	1.425611	0.01099263	0.007503284	
	2	0.350574	0.09925634	0.1378004	0.01717768
4	1	1.634227	0.002660573	0.0	
	2	0.264002	0.04936351	0.0	0.02759693

$$V_1 = 1.25 \times 10^7 \text{ cm / s}$$

$$V_2 = 2.5 \times 10^5 \text{ cm / s}$$

**Delayed Neutron Data:**

<b>Family, d</b>	$\beta_d$	$\lambda_d$
1	0.000247	0.0127
2	0.0013845	0.0317
3	0.001222	0.115
4	0.0026455	0.311
5	0.000832	1.40
6	0.000169	3.87

**Energy Conversion Factor:**

$$3.204 \times 10^{-11} \text{ W-s / Fission}$$

**Perturbation:**

Rod group 1 removed at 3 cm / s,  $0 \leq t \leq 26.666$  sec.

Rod group 2 inserted at 3 cm / s,  $7.5 \leq t \leq 47.5$  sec.

**WIGL Thermal Hydraulic Parameters:**

$$C^f = 2.46 \times 10^6 \text{ ergs / (gm - K)}$$

$$C^c = 5.43 \times 10^7 \text{ ergs / (gm - K)}$$

$$\rho^f = 1.03 \text{ gm / cm}^3$$

$$W_o = 2.2 \times 10^6 \text{ gm / sec}$$

$$h_o = 2.71 \times 10^7 \text{ ergs / (cm}^2\text{-sec-K)}$$

$$A_h = 2.95 \text{ cm}^{-1}$$

$$U = 2.2 \times 10^6 \text{ ergs / (cm}^2\text{-sec-K)}$$

$$V^c/(V^c + V^f) = 0.559$$

$$r = 0.0$$

$$\frac{\partial \rho^c H}{\partial \bar{T}^c} = 1.60 \times 10^7 \text{ ergs / (cm}^{-3}\text{-K)}$$

**Pressure =  $1.53 \times 10^7$  pascals**

**Coolant Inlet Temperature = 533 K**

**Initial Power = 184.8 Mw(t) (quarter core)**

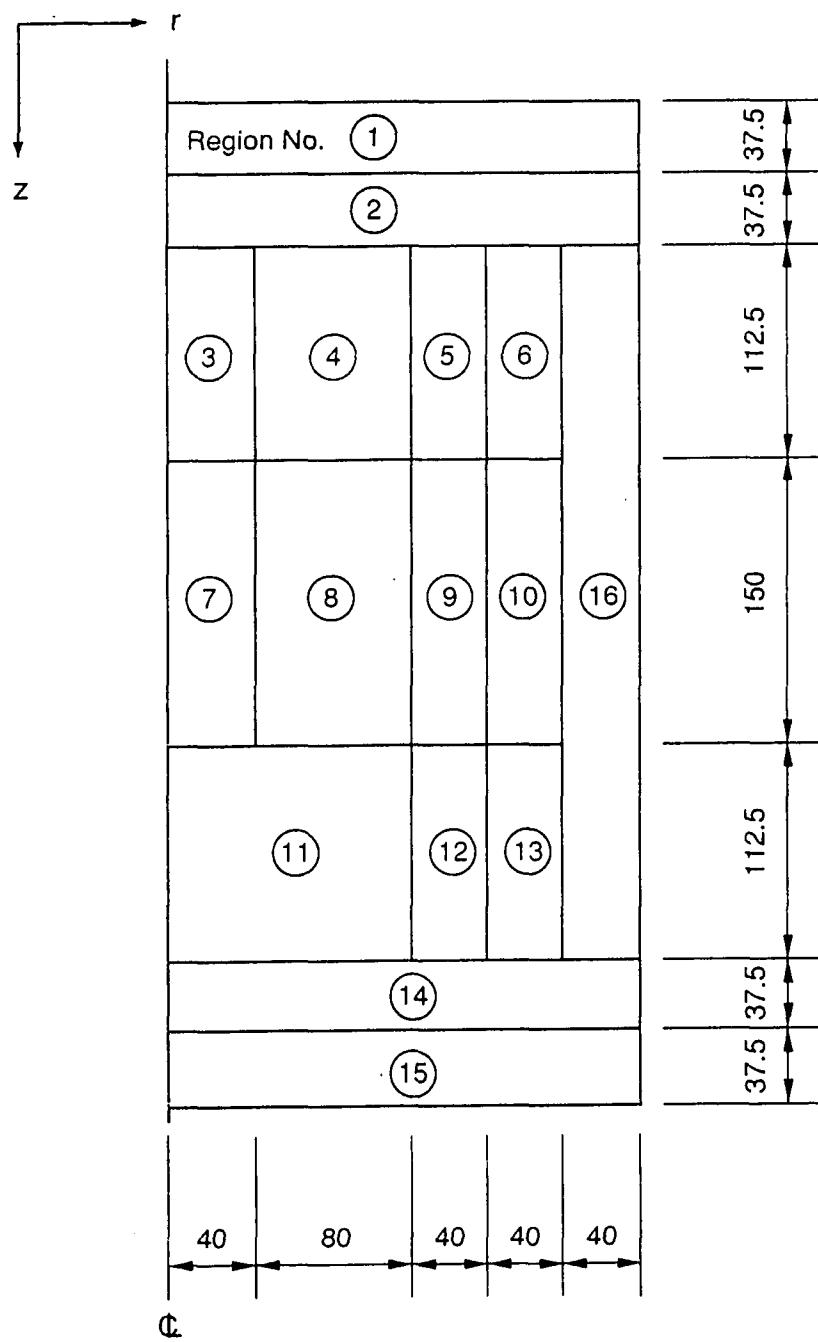
**Reference Density = 0.7961 gm / cm<sup>-3</sup>**

**Reference Coolant and Fuel Temperatures = 533 K**

## Macroscopic Cross Section Derivatives:

Parameter, $\Sigma$	$\frac{\partial \Sigma}{\partial \rho^c}$	$\frac{\partial \Sigma}{\partial T^c}$	$\frac{\partial \Sigma}{\partial T^f}$
$D_1^{-1}$	0.41	$-8.0 \times 10^{-5}$	$-6.6 \times 10^{-6}$
$D_2^{-1}$	2.7	$-1.3 \times 10^{-3}$	$-2.6 \times 10^{-6}$
$\Sigma_{a1}$	$2.83 \times 10^{-3}$	$3.0 \times 10^{-6}$	$3.3 \times 10^{-7}$
$\Sigma_{a2}$	$1.4 \times 10^{-2}$	$-8.2 \times 10^{-6}$	$-3.8 \times 10^{-7}$
$\Sigma_{f1}$	0.0	0.0	0.0
$\Sigma_{f2}$	$1.7 \times 10^{-2}$	$-8.3 \times 10^{-6}$	$-1.0 \times 10^{-6}$
$\Sigma_{12}$	$2.4 \times 10^{-2}$	$-1.5 \times 10^{-6}$	$8.5 \times 10^{-8}$

## B.5 Dodds' R-Z Geometry Transient Benchmark Problem



## Material Constants:

Region	Group	$D_g$	$\Sigma_{ag}$	$\nu \Sigma_{fg}$	$\Sigma_{12}$
1, 15	1	1.0684	0.002	0.0	
	2	0.32051	0.0033	0.0	0.026
2, 14	1	1.3495	0.00001	0.0	
	2	0.87032	0.019	0.0	0.012
3, 4, 11	1	1.3052	0.0024399	0.0011776	
	2	0.88857	0.013063	0.013268	0.0080351
5, 12	1	1.3052	0.0024399	0.0011776	
	2	0.88857	0.012623	0.013268	0.0080351
6, 13	1	1.3052	0.0024399	0.0011776	
	2	0.88857	0.012183	0.013268	0.0080351
7, 8	1	1.3052	0.0024399	0.0011776	
	2	0.88857	0.013453	0.013268	0.0080351
9	1	1.3052	0.0024399	0.0011776	
	2	0.88857	0.012973	0.013268	0.0080351

**Material Constants (cont.):**

Region	Group	$D_g$	$\Sigma_{ag}$	$\nu\Sigma_{fg}$	$\Sigma_{12}$
10	1	1.3052	0.0024399	0.0011776	
	2	0.88857	0.012933	0.013268	0.0080351
16	1	1.2997	0.0025639	0.0012875	
	2	0.87951	0.013065	0.014246	0.0079061

$$V_1 = 1.0 \times 10^7 \text{ cm / s}$$

$$V_2 = 2.2 \times 10^5 \text{ cm / s}$$

**Delayed Neutron Data:**

Family, d	$\beta_d$	$\lambda_d$
1	0.000247	0.0127
2	0.001384	0.0317
3	0.001222	0.115
4	0.002645	0.311
5	0.000832	1.40
6	0.000169	3.87

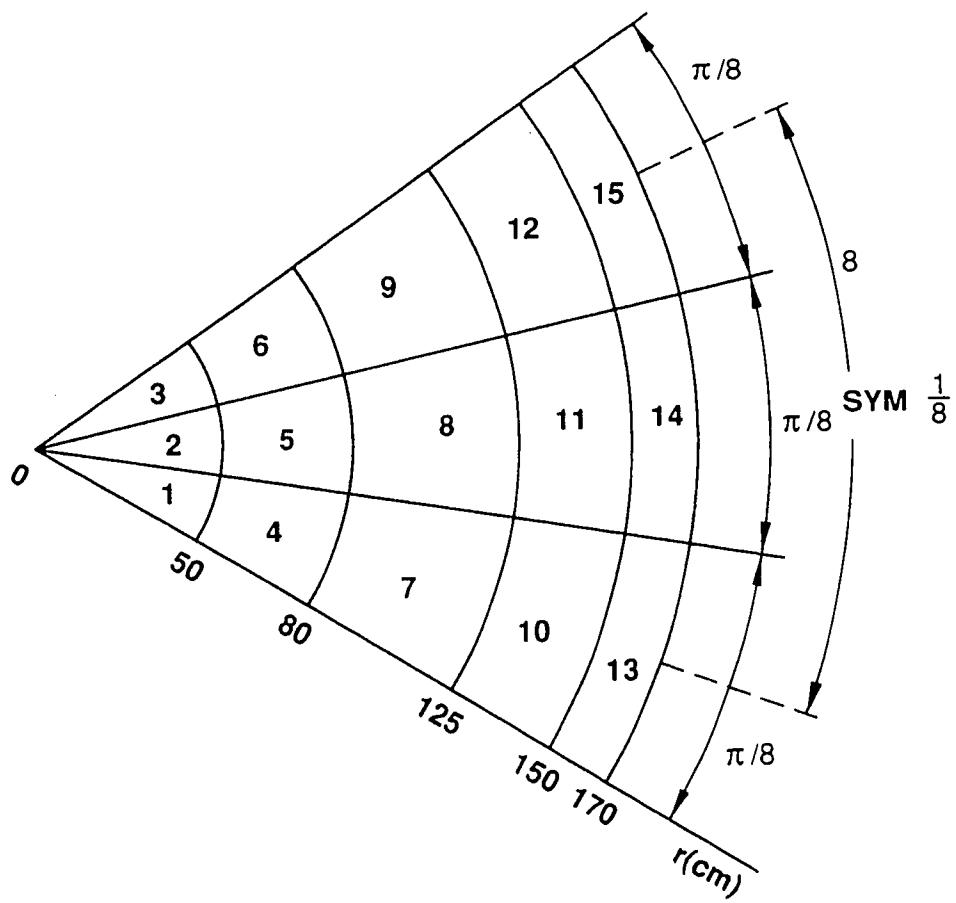
Perturbation:

In Regions 3 and 7 ;

$$\frac{\Sigma_{a2}(t)}{\Sigma_{a2}(0)} = 1 + 1.03 \cdot t, t \leq 1 \text{ or } 1.03 \cdot t, t \geq 1$$

In Region 11 ;

$$\frac{\Sigma_{a2}(t)}{\Sigma_{a2}(0)} = 1 - 1.03 \cdot t, t \leq 1 \text{ or } 0.97 \cdot t, t \geq 1$$

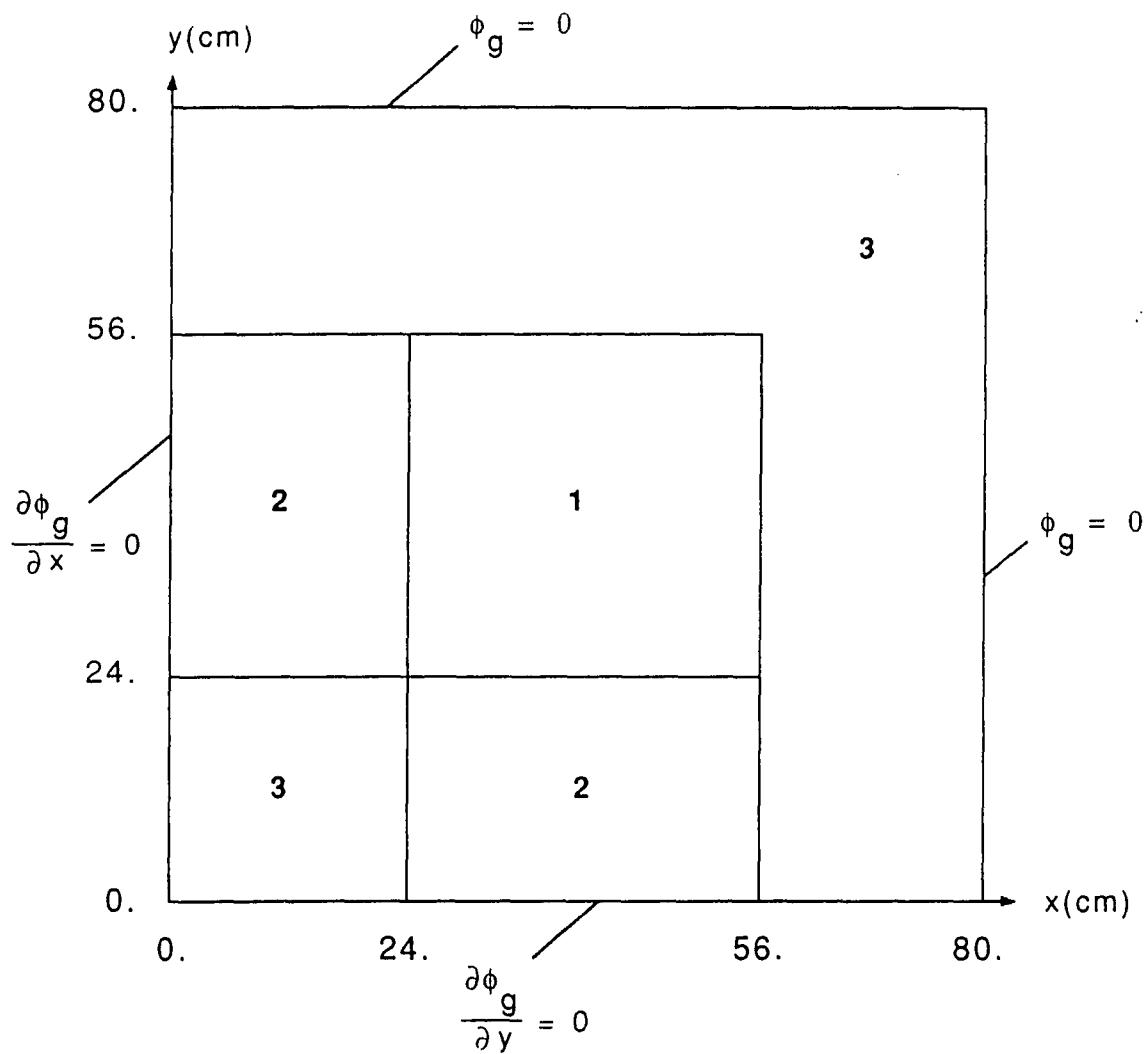
B.5 Pedroni's R- $\theta$  Geometry Steady-State Benchmark Problem

**Material Constants:**

Region	Group	$D_g$	$\Sigma_{ag}$	$\nu \Sigma_{fg}$
1	1	1.50	0.025	0.0263
2	1	1.50	0.025	0.0263
3	1	1.50	0.025	0.0263
4	1	1.50	0.0278	0.025
5	1	1.50	0.025	0.0263
6	1	1.50	0.025	0.0263
7	1	1.50	0.025	0.0275
8	1	1.50	0.025	0.0275
9	1	1.50	0.0278	0.025
10	1	1.50	0.0278	0.025
11	1	1.50	0.025	0.0275
12	1	1.50	0.025	0.0275
13	1	1.50	0.025	0.03
14	1	1.50	0.025	0.03
15	1	1.50	0.025	0.03

### B.6 The TWIGL Seed-Blanket Transient Benchmark Problem

Quadrant of TWIGL reactor



**Material Constants:**

<b>Composition</b>	<b>Group</b>	$D_g$	$\Sigma_{ag}$	$\nu\Sigma_{fg}$	$\Sigma_{12}$
1	1	1.4	0.01	0.007	
	2	0.4	0.15	0.2	0.01
2	1	1.4	0.01	0.007	
	2	0.4	0.15	0.2	0.01
3	1	1.3	0.008	0.003	
	2	0.5	0.05	0.06	0.01

$$V_1 = 1.0 \times 10^7 \text{ cm / s}$$

$$V_2 = 2.0 \times 10^5 \text{ cm / s}$$

**Delayed Neutron Data:**

<b>Family, d</b>	$\beta_d$	$\lambda_d$
1	0.0075	0.08

Perturbations: (in composition 1)

Step :

$$\frac{\Sigma_{a2}(t)}{\Sigma_{a2}(0)} = 0.97666, t = 0$$

Ramp :

$$\frac{\Sigma_{a2}(t)}{\Sigma_{a2}(0)} = 1 - 0.11667 \cdot t, t \leq 0.2 \text{ or } 0.97666 \cdot t, t \geq 0.2$$

## B.7 The S. Levy PWR Transient Test Problem

## Composition Maps

### Planes No. 1 and No. 2

### **Planes No. 3 Through No. 12**

**Planes No. 15 and No. 16**

## Problem Parameters

Total Problem Power	3411.2 MW
Soluable Boron Level	975 PPM
Fuel Assembly Dimensions	21.608 cm X 21.608 cm
Active Fuel Height	365.76 cm
Lower Reflector Height	40 cm
Upper Reflector Height	40 cm
Control Rod Group 2 Withdrawal Rate	3546.52 cm/sec
System Pressure	1.551+7 Pa (2250 PSIA)
Coolant Flow Rate	4100.3 Kg/m <sup>2</sup> -sec (839.81 lb/ft <sup>2</sup> -sec)
Coolant Inlet Temperature	564.82 K (557 °F)
Fuel Rod Cell Pitch	1.26 cm (0.041333 ft)
Fraction of Assembly	
Not Fuel Rod or Coolant	0.04015
Fuel Pellet Radius	0.3922 cm (0.012867 ft)
Fuel-Clad Gap Width	8.75-3 cm (2.87-4 ft)
Clad Thickness	5.718-2 cm (1.875-3 ft)
Fuel Pellet Density	10280 Kg/m <sup>3</sup> (641.75 lb/ft <sup>3</sup> )
Gap Thermal Conductivity	5678 W/m <sup>2</sup> -K (1000 Btu/hr-ft <sup>2</sup> - °F)
Number of Delayed Groups	6
Reference Fuel Temperature	949.8 K (1249.95 °F)
Reference Moderator Density	0.702643 Kg/m <sup>3</sup> (43.8646 lb/ft <sup>3</sup> )

## Delayed Neutron Data:

Family, d	$\beta_d$	$\lambda_d$
1	2.46358-4	1.27535-2
2	1.45914-3	3.17843-2
3	1.33969-3	1.19458-1
4	2.94599-3	3.18858-1
5	1.05456-3	1.40265+0
6	2.50605-4	3.92952+0

## Neutron Velocities:

$$V_1 = 1.706 \times 10^7 \text{ cm / s}$$

$$V_2 = 4.132 \times 10^5 \text{ cm / s}$$

## Initial Control Rod Positions (cm above base of problem)

Rod Group	Assemblies	Position
1	1,1	49.47
2	5,5	49.47
3	1,7 7,1 3,3	163.77
4	3,7 7,3	346.33

## Fuel Type 1 Cross sections

Moderator Volume Fraction = 0.61950e+00

Group	Type	A'	B' (Unrodded)	C'
1	$\Sigma_{tr}$	0.23093e+00	0.11251e+00	0.35299e-02
	$\Sigma_a$	0.76856e-02	0.12403e-02	-0.47597e-03
	$\Sigma_{12}$	0.17901e-01	0.19644e-01	0.63919e-03
	$\nu\Sigma_f$	0.43727e-02	0.59992e-03	-0.26850e-03
	$\kappa\Sigma_f$	0.57857e-13	0.83270e-14	-0.37966e-14
2	$\Sigma_{tr}$	0.87397e+00	0.85364e+00	0.34697e+00
	$\Sigma_a$	0.48737e-01	0.18515e-01	0.21109e-01
	$\nu\Sigma_f$	0.68675e-01	0.18314e-01	0.27654e-01
	$\kappa\Sigma_f$	0.90866e-12	0.24845e-12	0.36276e-12
Group	Type	A	B (Roddled)	C
1	$\Sigma_{tr}$	0.23682e+00	0.11251e+00	0.35299e-02
	$\Sigma_a$	0.13035e-01	0.12403e-02	-0.47597e-03
	$\Sigma_{12}$	0.15180e-01	0.19644e-01	0.63919e-03
	$\nu\Sigma_f$	0.43062e-02	0.59992e-03	-0.26850e-03
	$\kappa\Sigma_f$	0.56619e-13	0.83270e-14	-0.37966e-14
2	$\Sigma_{tr}$	0.86829e+00	0.85364e+00	0.34697e+00
	$\Sigma_a$	0.75309e-01	0.18515e-01	0.21109e-01
	$\nu\Sigma_f$	0.69927e-01	0.18314e-01	0.27654e-01
	$\kappa\Sigma_f$	0.91941e-12	0.24845e-12	0.36276e-12
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	-0.56529e-04	0.00000e+00	
	$\Sigma_a$	0.23866e-04	0.76871e+02	
	$\Sigma_{12}$	-0.14093e-04	0.81530e+02	
	$\nu\Sigma_f$	-0.19962e-05	0.00000e+00	
	$\kappa\Sigma_f$	-0.26796e-16	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.19563e+04	

## Fuel Type 2 Cross sections

Moderator Volume Fraction = 0.59405e+00

Group	Type	A'	B'	C'
		(Unrodded)		
1	$\Sigma_{tr}$	0.23113e+00	0.10689e+00	0.35871e-02
	$\Sigma_a$	0.88623e-02	0.13024e-02	-0.64845e-03
	$\Sigma_{12}$	0.16509e-01	0.18564e-01	0.40392e-03
	$\nu\Sigma_f$	0.53888e-02	0.66917e-03	-0.35725e-03
	$\kappa\Sigma_f$	0.71252e-13	0.94073e-14	-0.50463e-14
2	$\Sigma_{tr}$	0.85690e+00	0.80269e+00	0.29521e+00
	$\Sigma_a$	0.75007e-01	0.22321e-01	0.21861e-01
	$\nu\Sigma_f$	0.98118e-01	0.27485e-01	0.31547e-01
	$\kappa\Sigma_f$	0.12973e-11	0.37359e-12	0.41284e-12
Group	Type	A	B	C
		(Rodded)		
1	$\Sigma_{tr}$	0.23113e+00	0.10689e+00	0.35871e-02
	$\Sigma_a$	0.88623e-02	0.13024e-02	-0.64845e-03
	$\Sigma_{12}$	0.16509e-01	0.18564e-01	0.40392e-03
	$\nu\Sigma_f$	0.53888e-02	0.66917e-03	-0.35725e-03
	$\kappa\Sigma_f$	0.71252e-13	0.94073e-14	-0.50463e-14
2	$\Sigma_{tr}$	0.85690e+00	0.80269e+00	0.29521e+00
	$\Sigma_a$	0.75007e-01	0.22321e-01	0.21861e-01
	$\nu\Sigma_f$	0.98118e-01	0.27485e-01	0.31547e-01
	$\kappa\Sigma_f$	0.12973e-11	0.37359e-12	0.41284e-12
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	-0.56340e-04	0.00000e+00	
	$\Sigma_a$	0.23302e-04	0.69261e+02	
	$\Sigma_{12}$	-0.13502e-04	0.38956e+02	
	$\nu\Sigma_f$	-0.23637e-05	0.00000e+00	
	$\kappa\Sigma_f$	-0.31817e-16	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.18664e+04	

## Fuel Type 3 Cross sections

Moderator Volume Fraction = 0.60042e+00

Group	Type	A'	B' (Unrodded)	C'
1	$\Sigma_{tr}$	0.23073e+00	0.10826e+00	0.39298e-02
	$\Sigma_a$	0.86807e-02	0.12903e-02	-0.62819e-03
	$\Sigma_{12}$	0.16660e-01	0.18697e-01	0.45823e-03
	$\nu\Sigma_f$	0.53884e-02	0.66144e-03	-0.35038e-03
	$\kappa\Sigma_f$	0.71275e-13	0.92890e-14	-0.49515e-14
2	$\Sigma_{tr}$	0.86106e+00	0.81343e+00	0.30334e+00
	$\Sigma_a$	0.71620e-01	0.22069e-01	0.21224e-01
	$\nu\Sigma_f$	0.98055e-01	0.27022e-01	0.31633e-01
	$\kappa\Sigma_f$	0.12970e-11	0.36728e-12	0.41395e-12
Group	Type	A	B (Rodded)	C
1	$\Sigma_{tr}$	0.23073e+00	0.10826e+00	0.39298e-02
	$\Sigma_a$	0.86807e-02	0.12903e-02	-0.62819e-03
	$\Sigma_{12}$	0.16660e-01	0.18697e-01	0.45823e-03
	$\nu\Sigma_f$	0.53884e-02	0.66144e-03	-0.35038e-03
	$\kappa\Sigma_f$	0.71275e-13	0.92890e-14	-0.49515e-14
2	$\Sigma_{tr}$	0.86106e+00	0.81343e+00	0.30334e+00
	$\Sigma_a$	0.71620e-01	0.22069e-01	0.21224e-01
	$\nu\Sigma_f$	0.98055e-01	0.27022e-01	0.31633e-01
	$\kappa\Sigma_f$	0.12970e-11	0.36728e-12	0.41395e-12
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	-0.56126e-04	0.00000e+00	
	$\Sigma_a$	0.23293e-04	0.69984e+02	
	$\Sigma_{12}$	-0.13439e-04	0.43901e+02	
	$\nu\Sigma_f$	-0.23553e-05	0.00000e+00	
	$\kappa\Sigma_f$	-0.31692e-16	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.18769e+04	

## Fuel Type 4 Cross sections

Moderator Volume Fraction = 0.60996e+00

Group	Type	A'	B'	C'
		(Unrodded)		
1	$\Sigma_{tr}$	0.22921e+00	0.11030e+00	0.38737e-02
	$\Sigma_a$	0.88049e-02	0.13219e-02	-0.57045e-03
	$\Sigma_{12}$	0.16311e-01	0.18620e-01	0.59081e-03
	$\nu\Sigma_f$	0.62453e-02	0.71350e-03	-0.38756e-03
	$\kappa\Sigma_f$	0.82695e-13	0.10039e-13	-0.55618e-14
2	$\Sigma_{tr}$	0.86779e+00	0.82163e+00	0.30054e+00
	$\Sigma_a$	0.77714e-01	0.24455e-01	0.19878e-01
	$\nu\Sigma_f$	0.12152e+00	0.32065e-01	0.32595e-01
	$\kappa\Sigma_f$	0.16090e-11	0.43607e-12	0.42457e-12
Group	Type	A	B	C
		(Roddled)		
1	$\Sigma_{tr}$	0.22921e+00	0.11030e+00	0.38737e-02
	$\Sigma_a$	0.88049e-02	0.13219e-02	-0.57045e-03
	$\Sigma_{12}$	0.16311e-01	0.18620e-01	0.59081e-03
	$\nu\Sigma_f$	0.62453e-02	0.71350e-03	-0.38756e-03
	$\kappa\Sigma_f$	0.82695e-13	0.10039e-13	-0.55618e-14
2	$\Sigma_{tr}$	0.86779e+00	0.82163e+00	0.30054e+00
	$\Sigma_a$	0.77714e-01	0.24455e-01	0.19878e-01
	$\nu\Sigma_f$	0.12152e+00	0.32065e-01	0.32595e-01
	$\kappa\Sigma_f$	0.16090e-11	0.43607e-12	0.42457e-12
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	-0.54969e-04	0.00000e+00	
	$\Sigma_a$	0.22823e-04	0.67943e+02	
	$\Sigma_{12}$	-0.12906e-04	0.40190e+02	
	$\nu\Sigma_f$	-0.25985e-05	0.00000e+00	
	$\kappa\Sigma_f$	-0.34855e-16	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.18525e+04	

## Fuel Type 5 Cross sections

Moderator Volume Fraction = 0.58769e+00

Group	Type	A'	B' (Unrodded)	C'
1	$\Sigma_{tr}$	0.23049e+00	0.10563e+00	0.40216e-02
	$\Sigma_a$	0.94222e-02	0.13906e-02	-0.71070e-03
	$\Sigma_{12}$	0.15730e-01	0.18103e-01	0.44380e-03
	$\nu\Sigma_f$	0.62518e-02	0.74157e-03	-0.42845e-03
	$\kappa\Sigma_f$	0.82679e-13	0.10464e-13	-0.60642e-14
2	$\Sigma_{tr}$	0.85326e+00	0.78296e+00	0.27325e+00
	$\Sigma_a$	0.89871e-01	0.25703e-01	0.19748e-01
	$\nu\Sigma_f$	0.12205e+00	0.34203e-01	0.32101e-01
	$\kappa\Sigma_f$	0.16141e-11	0.46522e-12	0.41852e-12
Group	Type	A	B (Rodded)	C
1	$\Sigma_{tr}$	0.23049e+00	0.10563e+00	0.40216e-02
	$\Sigma_a$	0.94222e-02	0.13906e-02	-0.71070e-03
	$\Sigma_{12}$	0.15730e-01	0.18103e-01	0.44380e-03
	$\nu\Sigma_f$	0.62518e-02	0.74157e-03	-0.42845e-03
	$\kappa\Sigma_f$	0.82679e-13	0.10464e-13	-0.60642e-14
2	$\Sigma_{tr}$	0.85326e+00	0.78296e+00	0.27325e+00
	$\Sigma_a$	0.89871e-01	0.25703e-01	0.19748e-01
	$\nu\Sigma_f$	0.12205e+00	0.34203e-01	0.32101e-01
	$\kappa\Sigma_f$	0.16141e-11	0.46522e-12	0.41852e-12
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	-0.55822e-04	0.00000e+00	
	$\Sigma_a$	0.22830e-04	0.65296e+02	
	$\Sigma_{12}$	-0.13076e-04	0.26349e+02	
	$\nu\Sigma_f$	-0.26413e-05	0.00000e+00	
	$\kappa\Sigma_f$	-0.35522e-16	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.18021e+04	

## Fuel Type 6 Cross sections

Moderator Volume Fraction = 0.61950e+00

Group	Type	A'	B' (Unrodded)	C'
1	$\Sigma_{tr}$	0.22872e+00	0.11217e+00	0.36416e-02
	$\Sigma_a$	0.85654e-02	0.13126e-02	-0.57662e-03
	$\Sigma_{12}$	0.16537e-01	0.18814e-01	0.71674e-03
	$\nu\Sigma_f$	0.62437e-02	0.70490e-03	-0.37029e-03
	$\kappa\Sigma_f$	0.82709e-13	0.99026e-14	-0.53041e-14
2	$\Sigma_{tr}$	0.87274e+00	0.83447e+00	0.30919e+00
	$\Sigma_a$	0.73641e-01	0.24356e-01	0.19150e-01
	$\nu\Sigma_f$	0.12136e+00	0.31356e-01	0.32981e-01
	$\kappa\Sigma_f$	0.16076e-11	0.42633e-12	0.43106e-12
Group	Type	A	B (Rodded)	C
1	$\Sigma_{tr}$	0.23434e+00	0.11217e+00	0.36416e-02
	$\Sigma_a$	0.13540e-01	0.13126e-02	-0.57662e-03
	$\Sigma_{12}$	0.13841e-01	0.18814e-01	0.71674e-03
	$\nu\Sigma_f$	0.61193e-02	0.70490e-03	-0.37029e-03
	$\kappa\Sigma_f$	0.80714e-13	0.99026e-14	-0.53041e-14
2	$\Sigma_{tr}$	0.87047e+00	0.83447e+00	0.30919e+00
	$\Sigma_a$	0.10149e+00	0.24356e-01	0.19150e-01
	$\nu\Sigma_f$	0.12405e+00	0.31356e-01	0.32981e-01
	$\kappa\Sigma_f$	0.16362e-11	0.42633e-12	0.43106e-12
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	-0.54441e-04	0.00000e+00	
	$\Sigma_a$	0.22851e-04	0.68191e+02	
	$\Sigma_{12}$	-0.12895e-04	0.46010e+02	
	$\nu\Sigma_f$	-0.25839e-05	0.00000e+00	
	$\kappa\Sigma_f$	-0.34646e-16	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.18471e+04	

## Fuel Type 7 Cross sections

Moderator Volume Fraction = 0.59405e+00

Group	Type	A'	B' (Unrodded)	C'
1	$\Sigma_{tr}$	0.23012e+00	0.10695e+00	0.39005e-02
	$\Sigma_a$	0.92471e-02	0.13688e-02	-0.68947e-03
	$\Sigma_{12}$	0.15902e-01	0.18247e-01	0.52696e-03
	$\nu\Sigma_f$	0.62498e-02	0.73162e-03	-0.41883e-03
	$\kappa\Sigma_f$	0.82683e-13	0.10317e-13	-0.59055e-14
2	$\Sigma_{tr}$	0.85762e+00	0.79465e+00	0.28198e+00
	$\Sigma_a$	0.86263e-01	0.25175e-01	0.18967e-01
	$\nu\Sigma_f$	0.12183e+00	0.33550e-01	0.32099e-01
	$\kappa\Sigma_f$	0.16117e-11	0.45634e-12	0.41887e-12
Group	Type	A	B (Rodded)	C
1	$\Sigma_{tr}$	0.23012e+00	0.10695e+00	0.39005e-02
	$\Sigma_a$	0.92471e-02	0.13688e-02	-0.68947e-03
	$\Sigma_{12}$	0.15902e-01	0.18247e-01	0.52696e-03
	$\nu\Sigma_f$	0.62498e-02	0.73162e-03	-0.41883e-03
	$\kappa\Sigma_f$	0.82683e-13	0.10317e-13	-0.59055e-14
2	$\Sigma_{tr}$	0.85762e+00	0.79465e+00	0.28198e+00
	$\Sigma_a$	0.86263e-01	0.25175e-01	0.18967e-01
	$\nu\Sigma_f$	0.12183e+00	0.33550e-01	0.32099e-01
	$\kappa\Sigma_f$	0.16117e-11	0.45634e-12	0.41887e-12
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	-0.55713e-04	0.00000e+00	
	$\Sigma_a$	0.22839e-04	0.66040e+02	
	$\Sigma_{12}$	-0.13033e-04	0.30182e+02	
	$\nu\Sigma_f$	-0.26277e-05	0.00000e+00	
	$\kappa\Sigma_f$	-0.35345e-16	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.18185e+04	

## Fuel Type 8 Cross sections

Moderator Volume Fraction = 0.61950e+00

Group	Type	A'	B'	C'
		(Unrodded)		
1	$\Sigma_{tr}$	0.22957e+00	0.11229e+00	0.35918e-02
	$\Sigma_a$	0.81499e-02	0.12683e-02	-0.55119e-03
	$\Sigma_{12}$	0.17086e-01	0.19117e-01	0.68513e-03
	$\nu\Sigma_f$	0.53871e-02	0.64482e-03	-0.31974e-03
	$\kappa\Sigma_f$	0.71337e-13	0.90241e-14	-0.45613e-14
2	$\Sigma_{tr}$	0.87247e+00	0.84230e+00	0.32406e+00
	$\Sigma_a$	0.62501e-01	0.21830e-01	0.21196e-01
	$\nu\Sigma_f$	0.97834e-01	0.25689e-01	0.32285e-01
	$\kappa\Sigma_f$	0.12955e-11	0.34904e-12	0.42367e-12
Group	Type	A	B	C
		(Rodded)		
1	$\Sigma_{tr}$	0.23535e+00	0.11229e+00	0.35918e-02
	$\Sigma_a$	0.13280e-01	0.12683e-02	-0.55119e-03
	$\Sigma_{12}$	0.14386e-01	0.19117e-01	0.68513e-03
	$\nu\Sigma_f$	0.52900e-02	0.64482e-03	-0.31974e-03
	$\kappa\Sigma_f$	0.69706e-13	0.90241e-14	-0.45613e-14
2	$\Sigma_{tr}$	0.86875e+00	0.84230e+00	0.32406e+00
	$\Sigma_a$	0.89760e-01	0.21830e-01	0.21196e-01
	$\nu\Sigma_f$	0.99831e-01	0.25689e-01	0.32285e-01
	$\kappa\Sigma_f$	0.13155e-11	0.34904e-12	0.42367e-12
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	-0.55067e-04	0.00000e+00	
	$\Sigma_a$	0.23251e-04	0.71605e+02	
	$\Sigma_{12}$	-0.13301e-04	0.59339e+02	
	$\nu\Sigma_f$	-0.23334e-05	0.00000e+00	
	$\kappa\Sigma_f$	-0.31286e-16	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.18927e+04	

## Fuel Type 9 Cross sections

Moderator Volume Fraction = 0.59564e+00

Group	Type	A'	B' (Unrodded)	C'
1	$\Sigma_{tr}$	0.23000e+00	0.10738e+00	0.38393e-02
	$\Sigma_a$	0.91866e-02	0.13472e-02	-0.74041e-03
	$\Sigma_{12}$	0.15949e-01	0.18318e-01	0.60989e-03
	$\nu\Sigma_f$	0.62506e-02	0.73328e-03	-0.41343e-03
	$\kappa\Sigma_f$	0.82705e-13	0.10332e-13	-0.58905e-14
2	$\Sigma_{tr}$	0.85906e+00	0.79872e+00	0.28534e+00
	$\Sigma_a$	0.84888e-01	0.24927e-01	0.18374e-01
	$\nu\Sigma_f$	0.12191e+00	0.33324e-01	0.32223e-01
	$\kappa\Sigma_f$	0.16130e-11	0.45318e-12	0.42049e-12
Group	Type	A	B (Rodded)	C
1	$\Sigma_{tr}$	0.23000e+00	0.10738e+00	0.38393e-02
	$\Sigma_a$	0.91866e-02	0.13472e-02	-0.74041e-03
	$\Sigma_{12}$	0.15949e-01	0.18318e-01	0.60989e-03
	$\nu\Sigma_f$	0.62506e-02	0.73328e-03	-0.41343e-03
	$\kappa\Sigma_f$	0.82705e-13	0.10332e-13	-0.58905e-14
2	$\Sigma_{tr}$	0.85906e+00	0.79872e+00	0.28534e+00
	$\Sigma_a$	0.84888e-01	0.24927e-01	0.18374e-01
	$\nu\Sigma_f$	0.12191e+00	0.33324e-01	0.32223e-01
	$\kappa\Sigma_f$	0.16130e-11	0.45318e-12	0.42049e-12
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	-0.55398e-04	0.00000e+00	
	$\Sigma_a$	0.22859e-04	0.66678e+02	
	$\Sigma_{12}$	-0.13043e-04	0.31578e+02	
	$\nu\Sigma_f$	-0.26246e-05	0.00000e+00	
	$\kappa\Sigma_f$	-0.35251e-16	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.18358e+04	

## Lower Axial Reflector (Type 10) Cross sections

Moderator Volume Fraction = 0.10000e+01

Group	Type	A'	B' (Unrodded)	C'
1	$\Sigma_{tr}$	0.27394e+00	0.12367e+00	0.00000e+00
	$\Sigma_a$	0.22583e-02	-0.19523e-03	0.00000e+00
	$\Sigma_{12}$	0.31360e-01	0.34230e-01	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
2	$\Sigma_{tr}$	0.12243e+01	0.11955e+01	0.00000e+00
	$\Sigma_a$	0.41545e-01	0.15436e-01	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
Group	Type	A	B (Roddled)	C
1	$\Sigma_{tr}$	0.27394e+00	0.12367e+00	0.00000e+00
	$\Sigma_a$	0.22583e-02	-0.19523e-03	0.00000e+00
	$\Sigma_{12}$	0.31360e-01	0.34230e-01	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
2	$\Sigma_{tr}$	0.12243e+01	0.11955e+01	0.00000e+00
	$\Sigma_a$	0.41545e-01	0.15436e-01	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	0.00000e+00	0.00000e+00	
	$\Sigma_a$	0.00000e+00	0.18592e+02	
	$\Sigma_{12}$	0.00000e+00	0.00000e+00	
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.68650e+03	

## Inner Radial Reflector (Type 11) Cross sections

Moderator Volume Fraction = 0.10000e+01

Group	Type	A'	B' (Unrodded)	C'
1	$\Sigma_{tr}$	0.26771e+00	0.00000e+00	0.00000e+00
	$\Sigma_a$	0.19399e-02	0.00000e+00	0.00000e+00
	$\Sigma_{12}$	0.28934e-01	0.00000e+00	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
2	$\Sigma_{tr}$	0.12787e+01	0.00000e+00	0.00000e+00
	$\Sigma_a$	0.18890e-01	0.00000e+00	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
Group	Type	A	B (Rodded)	C
1	$\Sigma_{tr}$	0.26771e+00	0.00000e+00	0.00000e+00
	$\Sigma_a$	0.19399e-02	0.00000e+00	0.00000e+00
	$\Sigma_{12}$	0.28934e-01	0.00000e+00	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
2	$\Sigma_{tr}$	0.12787e+01	0.00000e+00	0.00000e+00
	$\Sigma_a$	0.18890e-01	0.00000e+00	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	0.00000e+00	0.00000e+00	
	$\Sigma_a$	0.00000e+00	0.46481e+02	
	$\Sigma_{12}$	0.00000e+00	0.00000e+00	
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.21771e+04	

## Outer Radial Reflector (Type 12) Cross sections

Moderator Volume Fraction = 0.10000e+01

Group	Type	A'	B' (Unrodded)	C'
1	$\Sigma_{tr}$	0.22401e+00	0.00000e+00	0.00000e+00
	$\Sigma_a$	0.39311e-03	0.00000e+00	0.00000e+00
	$\Sigma_{12}$	0.44508e-01	0.00000e+00	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
2	$\Sigma_{tr}$	0.12969e+01	0.00000e+00	0.00000e+00
	$\Sigma_a$	0.10124e-01	0.00000e+00	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
Group	Type	A	B (Rodded)	C
1	$\Sigma_{tr}$	0.22401e+00	0.00000e+00	0.00000e+00
	$\Sigma_a$	0.39311e-03	0.00000e+00	0.00000e+00
	$\Sigma_{12}$	0.44508e-01	0.00000e+00	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
2	$\Sigma_{tr}$	0.12969e+01	0.00000e+00	0.00000e+00
	$\Sigma_a$	0.10124e-01	0.00000e+00	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	0.00000e+00	0.00000e+00	
	$\Sigma_a$	0.00000e+00	0.72435e+02	
	$\Sigma_{12}$	0.00000e+00	0.00000e+00	
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.23374e+04	

## Upper Axial Reflector (Type 13) Cross sections

Moderator Volume Fraction = 0.10000e+01

Group	Type	A'	B' (Unrodded)	C'
1	$\Sigma_{tr}$	0.15495e+00	0.13674e+00	0.00000e+00
	$\Sigma_a$	0.35715e-03	0.40749e-03	0.00000e+00
	$\Sigma_{12}$	0.29924e-01	0.28571e-01	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
2	$\Sigma_{tr}$	0.93404e+00	0.10094e+01	0.00000e+00
	$\Sigma_a$	0.83648e-02	0.83648e-02	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
Group	Type	A	B (Rodded)	C
1	$\Sigma_{tr}$	0.15495e+00	0.13674e+00	0.00000e+00
	$\Sigma_a$	0.35715e-03	0.40749e-03	0.00000e+00
	$\Sigma_{12}$	0.29924e-01	0.28571e-01	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
2	$\Sigma_{tr}$	0.93404e+00	0.10094e+01	0.00000e+00
	$\Sigma_a$	0.83648e-02	0.83648e-02	0.00000e+00
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	0.00000e+00
Group	Type	Fuel Temp Feedback	Boron Micros	
1	$\Sigma_{tr}$	0.00000e+00	0.00000e+00	
	$\Sigma_a$	0.00000e+00	0.45973e+02	
	$\Sigma_{12}$	0.00000e+00	0.00000e+00	
	$\nu\Sigma_f$	0.00000e+00	0.00000e+00	
	$\kappa\Sigma_f$	0.00000e+00	0.00000e+00	
2	$\Sigma_a$	0.00000e+00	0.16274e+04	

## VITA

Bernard Ronald Bandini was born in Pittsburgh, Pennsylvania, on March 27, 1961. He attended the SS. Cyril and Methodius parochial school and several schools in the Charleroi Area School District until his graduation from Charleroi Area High School in June, 1979. He then enrolled at The Pennsylvania State University where he received his Bachelor of Science degree in Nuclear Engineering in March, 1983. As an undergraduate, he was elected to memberships in the Tau Beta Pi and Alpha Nu Sigma honorary societies.

He began post-graduate studies in Nuclear Engineering at The Pennsylvania State University in March, 1983, where he received his Master of Science degree in Nuclear Engineering in May, 1988. During his graduate studies he was supported by a U. S. Department of Energy Nuclear Science and Engineering Fellowship, several Nuclear Engineering Department research assistantships, and a Graduate Research Assistantship at the Los Alamos National Laboratory. He has co-authored the following publications:

D. H. Berwald et al., "Updated Reference Design of a Liquid-Metal-Cooled Tandem Mirror Fusion Breeder," *Fusion Tech.* (12), p. 30 (1987).

B. R. Bandini and A. J. Baratta, "Determination of the End State of the Three Mile Island Unit 2 Accident Using Neutron Transport Analysis," *Nucl. Tech.* (81), p. 370 (1988).

H.-Y. Wu, M.-Y. Hsiao, A. J. Baratta, B. R. Bandini, and E. L. Tolman, "Analysis of the Source Range Monitor During the First Four Hours of the Three Mile Island Unit 2 Accident," *Nucl. Tech.* (84), p. 169 (1989).

B. R. Bandini and A. J. Baratta, "Potential For Recriticality of the Relocated Core," *Nucl. Tech.* (87), p. 926 (1989).

A. J. Baratta, A. Haghigat, and B. R. Bandini, "The Overall Source Range Monitor Response During the Three Mile Island Unit 2 Accident," *Nucl. Tech.* (87), p. 1013 (1989).