ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Argonne, Illinois 60439

APPLICATIONS OF FINITE ELEMENT METHODS
IN REACTOR MATHEMATICS.
NUMERICAL SOLUTION OF THE
NEUTRON DIFFUSION EQUATION

by

Hans G. Kaper, Gary K. Leaf,
and Arthur J. Lindeman

Applied Mathematics Division

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

February 1972

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

TABLE OF CONTENTS

	<u>Page</u>
ABSTRACT.	7
I. DESCRIPTION OF THE PROBLEM.	9
A. Introduction.	9
B. Statement of Objectives	11
C. The Multigroup Formalism and the Power Iteration Method	11
II. FORMULATION OF A GALERKIN-TYPE APPROXIMATION PROCEDURE.	16
A. Introduction.	16
B. Statement of the Problem and Properties of the Solution	18
C. The Galerkin Approximation Procedure.	25
III. SOLUTION OF THE APPROXIMATE PROBLEM	32
A. Choice of a Finite-dimensional Subspace	32
B. Construction of a Basis	33
C. Solution of the Approximate Problem	38
D. Generation of the Coefficients.	39
IV. DESCRIPTION OF THE RESEARCH COMPUTER PROGRAM HOD.	46
V. A TIMING COMPARISON STUDY	48
A. Introduction.	48
B. Description of the Computer Program DARC2D.	48
C. Scope and Limitations of the Studies.	50
D. A Homogeneous Reactor Configuration	54
E. A Simple Two-zone Reactor Configuration	62
F. A 1,000 MWe Liquid Metal Fast Breeder Reactor Configuration	72
G. A Loosely Coupled Reactor Configuration	78
H. Conclusions	92
VI. SUMMARY AND CONCLUSIONS	95
APPENDIX.	96
ACKNOWLEDGEMENT	97
REFERENCES.	98

LIST OF FIGURES

<u>No.</u>	<u>Title</u>	<u>Page</u>
1.	Homogeneous reactor configuration (quarter reactor) with basic triangulation $\tau_{6,1}$; the coordinates of P are $x = 20.005$, $y = 20.055$ (in units of 1 cm).	56
2.	Two-zone reactor configuration (quarter reactor) with basic triangulation $\tau_{6,1}$	63
3.	Rectangular meshes (π_{16} and π_{64}) for the finite difference solution on the two-zone configuration	66
4.	Triangulation $\tau_{8,1}$ of the two-zone configuration; the coordinates of P are $x = 30.2$, $y = 30.2$ (in units of 1 cm) . .	69
5.	Triangulation $\tau_{22,1}$ of the two-zone configuration; P_1 : $x = 30.2$, $y = 30.2$, P_2 : $x = 45.0$, $y = 45.0$, P_3 : $x = 35.0525$, $y = 0$, P_4 : $x = 0$, $y = 35.0525$ (in units of 1 cm)	70
6.	Typical Oxide-Fueled LMFBR Configuration (quarter reactor) . .	73
7.	Triangulation $\tau_{17,1}$ of the LMFBR configuration	74
8.	Triangulation $\tau_{24,1}$ of the LMFBR configuration	75
9.	Loosely coupled reactor configuration (full reactor)	79
10.	Triangulation $\tau_{11,1}$ of the loosely coupled configuration (quarter reactor); P_1 : $x = 54$, $y = 20$, P_2 : $x = 75$, $y = 25$, P_3 : $x = 20$, $y = 20$, P_4 : $x = 38$, $y = 0$, P_5 : $x = 72$, $y = 0$ (in units of 1 cm)	82
11.	Triangulation $\tau_{13,1}$ of the loosely coupled configuration (quarter reactor); the coordinates of P_1 , P_2 , P_3 , and P_4 are the same as in Fig. 10, P_5 : $x = 70$, $y = 0$, P_6 : $x = 80$, $y = 8$	83
12.	Triangulation $\tau_{22,1}$ of the loosely coupled configuration (quarter reactor); P_1 : $x = 40$, $y = 20$, P_2 : $x = 60$, $y = 20$, P_3 : $x = 50$, $y = 30$, P_4 : $x = 80$, $y = 20$, P_5 : $x = 70$, $y = 30$, P_6 : $x = 60$, $y = 40$, P_7 : $x = 40$, $y = 40$, P_8 : $x = 20$, $y = 20$, P_9 : $x = 20$, $y = 0$, P_{10} : $x = 40$, $y = 0$, P_{11} : $x = 50$, $y = 0$, P_{12} : $x = 70$, $y = 0$ (in units of 1 cm).	84

LIST OF FIGURES (Contd.)

<u>No.</u>	<u>Title</u>	<u>Page</u>
13.	Triangulation $\tau_{26,1}$ of the loosely coupled configuration (quarter reactor); the coordinates of P_1 through P_{12} are the same as in Fig. 12, P_{13} : $x = 55$, $y = 45$, P_{14} : $x = 45$, $y = 45$, P_{15} : $x = 35$, $y = 25$, P_{16} : $x = 5$, $y = 5$, P_{17} : $x = 80$, $y = 10$ (in units of 1 cm).	87
14.	Flux in the first energy group along the 45° line from the lower left-hand corner to the midpoint of the loosely coupled reactor configuration. The unmarked curve represents the benchmark flux obtained with the quintic element on 208 triangles.	
	X DARC2D data, 441 mesh cells	
	Δ HOD data, quadratic elements on 104 triangles. .	90

LIST OF TABLES

<u>No.</u>	<u>Title</u>	<u>Page</u>
I.	Results for HOD on the Homogeneous Reactor Configuration.	57
II.	Benchmark Calculation of k_{eff} for the Two-Zone Reactor Configuration.	64
III.	Timing Results for DARC2D on the Two-Zone Reactor Configuration.	66
IV.	Timing/Results for HOD on a Fixed Triangulation ($\tau_{6,2}$) of the Two-Zone Reactor Configuration	67
V.	A Comparison of Execution Times for HOD and DARC2D on the Two-Zone Reactor Configuration	71
VI.	Mesh Structures for Finite Difference Approximations of the 1,000 MW LMFBR Configuration	76
VII.	Timing Results and Memory Requirements for DARC2D on the 1,000 MW LMFBR Configuration	77
VIII.	Execution Times and Memory Requirements for HOD on the 1,000 MW LMFBR Configuration	78
IX.	Benchmark Calculation Using Quintic Elements for the Loosely Coupled Configuration.	85
X.	Mesh Structures for Finite Difference Approximations of the Loosely Coupled Configuration.	85
XI.	Timing Results for DARC2D on the Loosely Coupled Configuration.	85
XII.	Execution Times for HOD on the Loosely Coupled Configuration.	86

APPLICATIONS OF FINITE ELEMENT METHODS IN REACTOR
MATHEMATICS. NUMERICAL SOLUTION OF THE
NEUTRON DIFFUSION EQUATION

by

Hans G. Kaper, Gary K. Leaf, Arthur J. Lindeman

ABSTRACT

In this report we investigate the applicability of a particular class of high-order approximation procedures for the numerical solution of the multigroup neutron diffusion equation on two-dimensional reactor configurations. This class is based on the use of variational principles and finite-dimensional subspaces whose elements are piecewise polynomial functions obtained by a Lagrange-type interpolation procedure over a triangulation of the domain (finite element method).

On the basis of a research High Order Diffusion computer program (HOD) we evaluate each approximation procedure and compare its performance with that of a low-order approximation procedure based on the use of finite difference methods, for the computation of neutron distributions and inventories in reactor design studies.

Our conclusions are as follows.

- 1) The use of high-order approximation procedures presents a significant advancement in computational capability for the numerical solution of the multigroup diffusion equation on fast reactor configurations.

- 2) With high-order approximation procedures based on finite element methods it is possible to obtain, at a reasonable cost, solutions to the multigroup diffusion equation which are sufficiently accurate that any errors can be attributed either to the diffusion theory approximation or other approximations in the reactor model, rather than to the numerical approximation procedure.

3) Solutions to the multigroup diffusion equation which are obtained with the finite element method provide flux inventories whose accuracy is comparable to the accuracy of the multiplication factor.

I. DESCRIPTION OF THE PROBLEM

A. Introduction

In the numerical solution of any equation of mathematical physics, such as the neutron diffusion equation, one may identify three problems:

- 1) The formulation of an approximation procedure; for example, an approximation procedure with a truncation error of a specified order.
- 2) The development of a particular technique for solving the system of algebraic equations which results from the application of the approximation procedure to the equation under investigation; for example, a direct or an iterative solution technique.
- 3) The implementation of accelerative devices in the solution technique, wherever possible; for example, successive over-relaxation or Chebyshev extrapolation.

Given the size of the fast memory of the computer, and given the required degree of accuracy of the solution to the particular physical problem, there are two basic approaches. Low-order approximation procedures, for example, those based on finite differencing, yield large systems of algebraic equations. The size of these systems is such that generally not all data can be held in the fast memory and thus they are more amenable to solution by iterative techniques. On the other hand, for high-order approximation procedures, the systems of algebraic equations are generally of moderate size, the storage of all data in the fast memory may become feasible, and a direct solution technique may be attempted.

Each approach has its advantages and disadvantages. A low-order approximation procedure coupled with an iterative solution technique requires relatively little arithmetic per data point and has a simple instruction set; however, it requires extensive data management and the

use of accelerative devices becomes almost mandatory. A high-order approximation procedure coupled with a direct solution technique requires more arithmetic per data point and has a more extensive instruction set and larger memory requirement. However, there is little need for data management and the ultimate program may, therefore, be considered as a subroutine. Furthermore, no accelerative devices are required. Thus, neither approach can *a priori* be discarded in favor of the other.

In reactor mathematics, an approach based on a low-order approximation procedure coupled with an iterative solution technique has been the traditional one, mainly because fast memories of scientific computers were too small to allow the application of direct solution techniques to any realistic reactor problem. However, digital computer hardware has evolved to a point where a fast-memory size of, say, 65,000 60-bit words is not uncommon and the direct solution of fairly large systems of algebraic equations has become feasible. This study, therefore, departs from the traditional low-order methods and investigates the use of a particular class of high-order approximation procedures. These procedures are based on the use of variational principles and subspaces generated by finite element methods. The use of finite element methods introduces a band structure into the resulting system of algebraic equations which makes an attempt to direct inversion realistic.

Current studies for liquid metal fast breeder reactors are based on core designs involving subassemblies which are hexagonal prisms arranged in concentric rings. In a detailed analysis of the neutron distribution there is, therefore, a natural distinction between the axial coordinate (i.e., the coordinate in the direction of the prism axis) and the coordinates in the transverse directions, since every transverse cut produces

the same geometrical pattern. This suggests analyzing the three-dimensional neutron distribution by means of a series of two-dimensional problems, each based on the same hexagonal pattern in a plane. Therefore, we will direct ourselves to solution techniques for the multigroup diffusion equation which are applicable to a broad class of two-dimensional reactor configurations, including those with a hexagonal structure. In accordance with the standard fast reactor core designs, we will allow material properties to vary on a subassembly basis.

B. Statement of Objectives

The primary objectives of the investigation are

- 1) to develop a class of high-order approximation procedures based on the use of finite element methods for the numerical solution of the neutron diffusion equation for two-dimensional reactor configurations;
- 2) to evaluate each approximation procedure on the basis of its intrinsic qualities through the development and use of a research computer program;
- 3) to determine whether these procedures provide a useful alternative to existing procedures based on the use of finite difference methods for the computation of neutron distributions in reactor design studies.

C. The Multigroup Formalism and the Power Iteration Method

In the multigroup formulation, the neutron diffusion equation is represented by a coupled system of differential equations for the *scalar flux*, ϕ .

$$\begin{aligned}
-\nabla \cdot D_g(\underline{r}) \nabla \phi_g(\underline{r}) + \sum_{g'}^R(\underline{r}) \phi_{g'}(\underline{r}) = & \sum_{\substack{g'=1 \\ g' \neq g}}^G \sum_{g''}^S(\underline{r}) \phi_{g''}(\underline{r}) \\
+ \sum_{g'=1}^G \chi_{g',g}(\underline{r}) (v \sum_{f} \phi_f)(\underline{r}) \phi_{g'}(\underline{r}), & \quad g = 1 \dots G,
\end{aligned} \tag{1}$$

for $\underline{r} \in \Omega$. If the exterior boundary of Ω is denoted by $\partial_e \Omega$, the following general type of boundary condition is associated with Eq. (1):

$$a(\underline{r}) (D_g \frac{\partial \phi_g}{\partial n})(\underline{r}) + b(\underline{r}) \phi_g(\underline{r}) = 0 \quad \underline{r} \in \partial_e \Omega. \tag{2}$$

The boundary condition (2) includes the cases of a free boundary ($a=0$), a reflective boundary ($b=0$), and an extrapolated boundary condition ($a>0, b>0$); a and b are non-negative and do not vanish simultaneously. All symbols in Eqs. (1) and (2) have their usual meaning [1]. If the reactor is modeled as a finite number of subassemblies, each of which is characterized by specific material properties, then the bulk coefficients $D_g(\underline{r})$, $\sum_{g'}^R(\underline{r})$, $\sum_{g''}^S(\underline{r})$, $\chi_{g',g}(\underline{r})$ and $(v \sum_f \phi_f)(\underline{r})$, as well as the boundary coefficients $a(\underline{r})$ and $b(\underline{r})$, are piecewise continuous functions of position and are constant throughout each subassembly. In addition to the boundary condition (2), the usual interface conditions must be satisfied. If the collection of material interfaces within Ω is denoted by $\partial_i \Omega$, then the interface conditions are,

$$\phi_g(\underline{r}) \text{ and } (D_g \frac{\partial \phi_g}{\partial n})(\underline{r}) \text{ are continuous across } \partial_i \Omega, \tag{3}$$

where $\underline{n}$ is the unit vector which is locally normal to $\partial_i \Omega$, and $\partial/\partial n = \underline{n} \cdot \nabla$.

To solve the generalized eigenvalue problem (1)-(3) for the multiplication factor--i.e., the dominant characteristic value, $k=k_{\text{eff}}$ --and the corresponding positive eigenfunction $\{\phi_g(\underline{r}): g=1, \dots, G\}$ we adopt the usual power iteration method. Starting with a positive but otherwise arbitrary estimate $\{\phi_g^{(0)}\}$ for the group fluxes and a positive estimate $k^{(0)}$ for the multiplication factor, we generate successive estimates $\{\phi_g^{(n)}\}$ and $k^{(n)}$, $n=1, 2, \dots$, according to the following scheme:

$$\begin{aligned} -\nabla \cdot D_g(\underline{r}) \nabla \phi_g^{(n)}(\underline{r}) + \sum_g R_g(\underline{r}) \phi_g^{(n)}(\underline{r}) &= \sum_{g' < g} \chi_{g',g}(\underline{r}) \phi_{g'}^{(n)}(\underline{r}) \\ &+ \sum_{g' > g} \chi_{g',g}(\underline{r}) \phi_{g'}^{(n-1)}(\underline{r}) + \frac{1}{k^{(n-1)}} \sum_{g'=1}^G \chi_{g',g}(\underline{r}) (v \sum_f \chi_{f,g'}(\underline{r}) \phi_{g'}^{(n-1)}(\underline{r})), \end{aligned} \quad (4)$$

$g=1, \dots, G$, and

$$k^{(n)} = k^{(n-1)} \frac{\langle \phi^{(n)}, \phi^{(n)} \rangle}{\langle \phi^{(n)}, \phi^{(n-1)} \rangle} \quad (5)$$

where $\langle, \rangle$ denotes any appropriate inner product.

At each step of this power iteration procedure we thus solve G uncoupled self-adjoint elliptic boundary value problems of the form

$$-\nabla \cdot D(\underline{r}) \nabla \phi(\underline{r}) + \sum (\underline{r}) \phi(\underline{r}) = f(\underline{r}) \quad \underline{r} \in \Omega, \quad (6)$$

where f is a known function. The unknown ϕ is subject to the same boundary and interface conditions (2) and (3),

$$a(\underline{r}) (D \frac{\partial \phi}{\partial n})(\underline{r}) + b(\underline{r}) \phi(\underline{r}) = 0 \quad \underline{r} \in \partial_e \Omega, \quad (7)$$

$$\phi(\underline{r}) \quad \text{and} \quad \left(D \frac{\partial \phi}{\partial n}\right)(\underline{r}) \quad \text{continuous across } \partial_1 \Omega. \quad (8)$$

In the following study, the domain Ω is a bounded set in the two-dimensional Euclidian plane, E_2 ; Ω is the union of a finite number (K) of contiguous subdomains Ω_i ,

$$\Omega = \bigcup_{i=1}^K \Omega_i,$$

where each Ω_i corresponds to the interior of a subassembly. We do not allow for the presence of "black" (i.e., totally absorbing) subassemblies or of "holes" in the interior of the reactor domain Ω . The boundary $\partial\Omega$ of Ω , which is the union of the exterior boundary $\partial_e \Omega$ and the interfaces $\partial_1 \Omega$,

$$\partial\Omega = \partial_e \Omega \cup \partial_1 \Omega,$$

is assumed to be piecewise rectilinear, and therefore piecewise smooth in the usual sense.

As stated in the introduction, the traditional approach to the numerical solution of equations of the form (6) in reactor computations has been through the use of finite difference methods. Normally, a five-point difference approximation is applied to the differential operator. This particular approximation is but one example of what are commonly known in approximation theory as low-order methods, i.e., methods which give rise to a truncation error of low degree in the mesh size. Higher-order approximations could be achieved, for example, through the use of more sophisticated finite differencing techniques--e.g., the nine-point

difference approximation. However, such an approach, when applied to equations of the form (6), destroys the self-adjoint character of the equation and has found little application for this reason.

An alternative means of systematically achieving high-order approximations, which conserves the self-adjoint character of the one-group diffusion equation, is provided by the Ritz-Galerkin technique of variational calculus. The first step of the procedure is to reformulate the problem and to look for a generalized solution of the boundary value problem in some appropriate Hilbert space. The second step consists of the choice of a particular finite-dimensional subspace of the Hilbert space, in which an approximate solution of the boundary value problem is subsequently sought.

In the following study we will adopt the latter approach for the construction of high-order approximation procedures for the numerical solution of the boundary value problem (6)-(8). In particular, we will employ finite element methods in the construction of the finite-dimensional subspaces.

II. FORMULATION OF A GALERKIN-TYPE APPROXIMATION PROCEDURE

A. Introduction

In this chapter we will develop a Galerkin-type approximation procedure for the solution of the elliptic boundary-value problem described by Eqs. (6), (7), and (8). Before doing so, however, it is appropriate to comment on the problem of solving this type of boundary-value problem with piecewise continuous data, as it is neither a trivial nor a classical problem. In order to provide some insight into this matter, we give some perspective on the historical development of this area.

The solution of elliptic boundary-value problems for which all the data (coefficients, inhomogeneous term, and boundary) are smooth, i.e., at least twice continuously differentiable, is well understood--see, for example, Courant and Hilbert [2]. The properties of the solutions are known; the solution is "classical" in the sense that it exists and is at least twice continuously differentiable everywhere in the domain. If the restrictions on the data are relaxed somewhat it is appropriate to recast the boundary-value problem into a so-called "weak" form. In the case of a symmetric differential operator, this weak statement is equivalent to a variational formulation of the problem. The solution may still be classical in the above sense; however, the techniques for demonstrating its existence, uniqueness, and regularity properties require the use of methods of functional analysis, notably the use of spaces of functions with generalized derivatives (Sobolev spaces). Thus, for example, if the coefficients and the inhomogeneous term are Hölder continuous (the definition of Hölder continuity is given below, p. 21), and the boundary is piecewise smooth, then the variational solution is still classical, see, e.g., Sec. 5.8 of [3].

In the problems we are considering, the restrictions on the data must be relaxed even further, as the coefficients are only piecewise constant and the source term corresponds to a piecewise continuous function. In this case, the solution is no longer classical, and a study of the nature of the solution involves an even more elaborate use of the methods of functional analysis. We know, for example, that in the case of Dirichlet boundary conditions (i.e., $a(r) \equiv 0$ in Eq. (7)) the solution of the diffusion equation (6) is still Hölder continuous in the closure $\bar{\Omega}$ of the domain Ω . Major contributors to the study of such equations include Morrey [4], Murthy and Stampachia [5], and Ladyzhenskaja and Ural'tseva [6]. In Section B we formulate the boundary-value problem described by Eqs. (6), (7), and (8) in differential form, and state the results of Ladyzhenskaja and Ural'tseva which are relevant to our own problem, particularly those regarding the regularity properties of the solution (Theorem 1).

In Section C we develop the Galerkin-type approximation procedure, utilizing the techniques employed by Ciarlet, Schultz, and Varga [7] for the general case of monotone operator equations in reflexive Banach spaces. These techniques necessitate the restatement of the original problem in a weaker form. The existence and uniqueness of the solution in an appropriate Hilbert space are provided by the results of Ref. 7. The approximation procedure consists of finding a weak solution in a finite-dimensional subspace of the Hilbert space. Again, the existence and uniqueness of an approximate solution are provided by the results of Ref. 7. Finally, we obtain from Ref. 7 an important inequality which we then use to obtain error estimates for the approximate solutions.

B. Statement of the Problem and Properties of the Solution

As noted in Chapter I, the reactor domain Ω is a bounded region in the two-dimensional Euclidean plane, E_2 . It is the union of a finite number of contiguous subdomains, $\Omega = \bigcup_{i=1}^K \Omega_i$, such that within each such subdomain the material properties do not vary. The boundary $\partial\Omega = \bar{\Omega} - \Omega$ is piecewise rectilinear and therefore piecewise smooth in the usual sense. In fact, between two vertices, i.e., in the interior of each edge, the boundary is infinitely smooth. $\partial\Omega$ is the union of the exterior boundary, $\partial_e\Omega$, and the interfaces, $\partial_i\Omega$.

For typographical reasons we will deviate slightly from the notation of Chapter I. If E_2 denotes the two-dimensional Euclidean plane, a point P in E_2 with coordinates x_1, x_2 will be indicated by the symbol x . Similarly, a gradient with respect to x will be indicated by the symbol $\frac{\partial}{\partial x}$. The measure in E_2 is the usual two-dimensional Lebesgue measure (dx) and the measure on $\partial\Omega$ is the usual one-dimensional Lebesgue measure (ds) . Notice that the boundary $\partial\Omega$ is a set of two-dimensional measure zero in E_2 and that the vertices form, in turn, a set of one-dimensional measure zero on $\partial\Omega$. On $\partial_e\Omega$, the unit normal vector n (directed towards the exterior of Ω) is defined almost everywhere. The component of the spatial gradient $(\partial/\partial x)$ in the direction of n is $n \cdot (\partial/\partial x)$, which we abbreviate by $(\partial/\partial n)$.

We are then faced with the following boundary-value problem: to find a function u which satisfies the equation

$$-\frac{\partial}{\partial x} \cdot D(x) \frac{\partial}{\partial x} u(x) + \int(x)u(x) = f(x) \quad x \in \Omega, \quad (9)$$

which satisfies the boundary condition

$$a(x) \left(D \frac{\partial u}{\partial n} \right) (x) + b(x)u(x) = 0 \quad x \in \partial_e \Omega, \quad (10)$$

and which satisfies the interface conditions

$$u(x) \text{ and } \left(D \frac{\partial u}{\partial n} \right) (x) \text{ are continuous across } \partial_1 \Omega. \quad (11)$$

In Eq. (9), the coefficients D and Σ are bounded and piecewise constant,

$$D(x) = D_i, \quad \Sigma(x) = \Sigma_i \quad \text{for all } x \in \Omega_i. \quad (12a)$$

As we have excluded the presence of "black" subassemblies in the interior of the domain Ω , all diffusion coefficients are strictly positive. Moreover, in each subdomain Ω_i the removal cross section is strictly positive. Hence, there exist two positive constants D_0 and Σ_0 such that

$$0 < D_0 = \min_{(i)} D_i, \quad 0 < \Sigma_0 = \min_{(i)} \Sigma_i. \quad (12b)$$

The inhomogeneous term f in Eq. (9) is continuous in each subdomain Ω_i , and hence piecewise continuous in Ω .

In the boundary condition (10) the coefficients a and b are bounded and piecewise constant:

$$a(x) = a_i, \quad b(x) = b_i \quad \text{for all } x \in (\partial_e \Omega)_i, \quad (13a)$$

where $(\partial_e \Omega)_i = \partial_e \Omega \cap \partial \Omega_i$. Moreover,

$$a_i \geq 0, \quad b_i \geq 0, \quad a_i + b_i > 0 \quad i = 1, \dots, K. \quad (13b)$$

We will now formulate the problem (9)-(11) in precise form.

Let $L_2(\Omega)$ be the Hilbert space of functions which are square integrable over Ω . The inner product on $L_2(\Omega)$ is given by

$$(u, v) = \int_{\Omega} u(x)v(x)dx$$

and the norm by

$$||u|| = (u, u)^{\frac{1}{2}}.$$

We define $D(L)$ as the set of functions in $L_2(\Omega)$ which have the following properties:

- (i) they are twice continuously differentiable in Ω ;
- (ii) the functions and their first derivatives are continuous on $\partial\Omega$;
- (iii) they satisfy the boundary conditions (10) and (11).

We now define the linear differential operator L by the expression

$$(Lu)(x) = -\frac{\partial}{\partial x} \cdot D(x) \frac{\partial}{\partial x} u(x) \quad (14)$$

on the domain $D(L)$. Furthermore, we define the multiplicative operator $\tilde{\Gamma}$ on $L_2(\Omega)$ by

$$(\tilde{\Gamma}u)(x) = \tilde{\Gamma}(x)u(x). \quad (15)$$

Then we have the following

Problem. To find the function $u \in D(L)$ which satisfies the equation

$$(L + \int)u = f, \quad f \in L_2(\Omega). \quad (16)$$

In Ref. 6, Chapter 3, Sec. 16, a problem of the same type is discussed. We shall state below the relevant results when applied to Eq. (9). For the sake of completeness we first recall some definitions from functional analysis.

Definition 1. $W_2^1(\Omega)$ denotes the Hilbert space of all elements of $L_2(\Omega)$ that have generalized derivatives of the first order in $L_2(\Omega)$. The inner product in $W_2^1(\Omega)$ is defined by

$$\langle u, v \rangle_{1,2} = \int_{\Omega} \left(uv + \frac{\partial u}{\partial v} \cdot \frac{\partial v}{\partial x} \right) dx$$

and the norm by

$$\|u\|_{1,2} = \langle u, u \rangle_{1,2}^{1/2}.$$

When the domain Ω has a piecewise smooth boundary, as it does in the problem under consideration, $W_2^1(\Omega)$ can be characterized as the closure in the norm $\| \cdot \|_{1,2}$ of the set of infinitely differentiable functions which vanish in a neighborhood of the boundary $\partial\Omega$ of Ω .

We say that a function $u(x)$ is Hölder continuous in the set $\bar{\Omega}$ with modulus K and exponent α ($K > 0$, $0 < \alpha \leq 1$) if at any two points x' and x'' in $\bar{\Omega}$,

$$|u(x') - u(x'')| \leq K|x' - x''|^\alpha.$$

Here $|x' - x''|$ is the Euclidean distance in E_2 .

Definition 2. $C_{l,\alpha}(\bar{\Omega})$ is the Banach space of all functions u that are l times continuously differentiable in $\bar{\Omega}$ and whose l -th derivatives are Hölder continuous in $\bar{\Omega}$ with exponent α , and for which the value of

$$|u|_{l,\alpha,\Omega} = \sum_{k=0}^{l-1} \sum_{(k)} \max_{x \in \bar{\Omega}} |u^{(k)}(x)| + \sum_{(l)} \sup_{x \in \bar{\Omega}} \frac{|u^{(l)}(x') - u^{(l)}(x'')|}{|x' - x''|^\alpha}$$

is finite. Here, $u^{(k)}$ denotes an arbitrary derivative of order k of u with respect to x , and $\sum_{(k)}$ denotes summation over all possible derivatives of order k . The quantity $|u|_{l,\alpha,\Omega}$ defines a norm in $C_{l,\alpha}(\bar{\Omega})$. $C_{l,\alpha}(\Omega)$ is the set of functions belonging to $C_{l,\alpha}(\bar{\Omega}')$ for all $\Omega' \subset \Omega$.

The boundary value problem (16) is uniformly elliptic, i.e., for any real $\xi = (\xi_1, \xi_2)$,

$$0 < \nu \xi_1 \xi_1 \leq D(x) \delta_{ij} \xi_i \xi_j \leq \mu \xi_1 \xi_1$$

with $\nu = \min_{(1)} D_i$ and $\mu = \max_{(1)} D_i$. (Here, we have used the Einstein summation convention: if in a term the same index occurs more than once, a summation over this index must be performed.) Now, compare Eqs. (9)-(11) with the following integral identity:

$$\begin{aligned} & \int_{\Omega} \left(D(x) \frac{\partial u}{\partial x} \cdot \frac{\partial v}{\partial x} + \sum_{(1)} f(x) u(x) v(x) \right) dx + \int_{\partial_e \Omega} \frac{b(x)}{a(x)} u(x) v(x) ds \\ &= \int_{\Omega} f(x) v(x) dx, \end{aligned} \quad (17)$$

for any $v \in D(L)$, where

$$\partial_e' \Omega = \{x: x \in \partial_e \Omega, a(x) > 0\}.$$

If $u \in D(L)$ and all the coefficients in Eq. (17) are smooth, then it is easy to see that the problem (16) and Eq. (17) are equivalent. However, the identity (17) is meaningful for functions u possessing first-order derivatives only and for piecewise continuous functions D , $\sum$ and f . The principal term in it is $\int_{\Omega} D \frac{\partial u}{\partial x} \cdot \frac{\partial v}{\partial x} dx$. The quadratic form $\int_{\Omega} D \frac{\partial u}{\partial x} \cdot \frac{\partial u}{\partial x} dx$ corresponding to it is defined for an arbitrary function $u \in W_2^1(G)$ because of the uniform ellipticity of the boundary value problem. Therefore, a natural class of generalized solutions of Eqs. (9)-(11) is the class of generalized solutions in $W_2^1(\Omega)$.

For $u, v \in W_2^1(\Omega)$ we define the bilinear form

$$\begin{aligned} a(u, v) = & \int_{\Omega} \left(D(x) \frac{\partial u}{\partial x} \cdot \frac{\partial v}{\partial x} + \sum(x) u(x) v(x) \right) dx \\ & + \int_{\partial_e' \Omega} \frac{b(x)}{a(x)} u(x) v(x) ds. \end{aligned} \quad (18)$$

Then we have the following

Generalized Problem. To find the function $u \in W_2^1(\Omega)$ which satisfies the equation

$$a(u, v) = (f, v) \quad \text{for all} \quad v \in W_2^1(\Omega). \quad (19)$$

A solution to this generalized problem is called a *generalized solution* of the original problem.

The boundary value problem (16) is very similar to a problem of diffraction theory discussed by Ladyzhenskaja and Ural'tseva in Ref. 6, Chapter 3, Sec. 16. These authors studied the steady-state diffraction problem in which there are several media of different substances and in which the equilibrium state is described by an equation of the form (9) or a generalization thereof. Furthermore, the dependent variable satisfies boundary and interface conditions of precisely the form (10) and (11). This analogy enables us to deduce the results pertinent to our problem as special cases. The following theorem is adapted from Theorems 16.1 and 16.2 of Ref. 6.

Theorem 1. The boundary-value problem (16) has a unique generalized solution $u(x)$ in $W_2^1(\Omega)$; $u(x)$ belongs to the classes $C_{0,\alpha}(\bar{\Omega})$ and $C_{2,\alpha}(\Omega_i)$, where $i = 1, \dots, K$. Moreover, $D(x) \frac{\partial u}{\partial x}$ exists and is Hölder continuous in each Ω_i up to the boundary $\partial\Omega_i$, except possibly at the vertices on $\partial\Omega_i$. The solution satisfies all conditions of the problem in the form (9), (10), (11). Finally, $D(x) \frac{\partial u}{\partial x}$ exists and is Hölder continuous in Ω up to $\partial\Omega$, except possibly at the vertices on $\partial\Omega$.

This theorem answers the question as to the existence of a solution to our problem and gives the continuity properties of the solution. We will utilize these results in the construction of the finite-dimensional subspaces in which the approximate solution will be sought.

Remark 1. The two theorems 16.1 and 16.2 in Ref. 6, from which Theorem 1 is adapted, are stated for the case of the Dirichlet problem. In the text

(page 206, English edition), the authors state that "the cases of the other boundary conditions can be considered in a similar manner." The boundary conditions which are discussed in Ref. 6 are those of the second and third kind, or, in our notation, those of the form $D \frac{\partial u}{\partial n} + bu = 0$ on $\partial_e \Omega$. If $b \equiv 0$, we have the second kind; if $b \neq 0$, we have the third kind. Note that this form does not contain boundary conditions of "mixed type," and $\frac{\partial u}{\partial n} + bu = 0$. The difference is that the mixed-type boundary conditions permit a Dirichlet condition $u = 0$ on a portion of the boundary, whereas the boundary conditions of the third kind as presented in Ref. 6 do not. Since the results of Theorems 16.1 and 16.2 in Ref. 6, as condensed in Theorem 1, hold for Dirichlet problems and for Robin problems (boundary conditions of the third kind), we have assumed the results of this theorem to be valid for the mixed-type problem as well.

C. The Galerkin Approximation Procedure

The results of this section are based on the work of Ciarlet, Schultz, and Varga [7]. The latter authors study the Galerkin method for approximating the solutions of a wide class of non-linear elliptic boundary-value problems.

The following two lemmas will be useful for the discussion. They are special cases of Sobolev's imbedding theorems - see [8], Chapter I, Sections 8.2 and 11.5.

Lemma 1. If $u \in W_2^1(\Omega)$, then $u \in L_2(\Omega)$; moreover, the imbedding operator which maps an element u of $W_2^1(\Omega)$ into itself as an element of $L_2(\Omega)$ is completely continuous (and *a fortiori* bounded).

Lemma 2. Let Γ be a piecewise smooth curve lying entirely in $\bar{\Omega}$. If $u \in W_2^1(\Omega)$, then $u \in L_2(\Gamma)$; moreover, the imbedding operator which maps an

element u of $W_2^1(\Omega)$ into itself as an element of $L_2(\Gamma)$ is completely continuous (and *a fortiori* bounded).

In the following lemma we state some relevant properties of the bilinear form $a(u, v)$.

Lemma 3. If the coefficients of the boundary-value problem (16) satisfy the inequalities (12) and (13), then the bilinear form $a(u, v)$, defined in Eq. (18) has the following properties:

- (i) $a(u, v)$ is a bilinear symmetric functional on $W_2^1(\Omega)$;
- (ii) there exists a constant C such that

$$|a(u, v)| \leq C \|u\|_{1,2} \|v\|_{1,2}$$

for all $u, v \in W_2^1(\Omega)$;

- (iii) there exists a positive constant γ such that

$$a(u, u) \geq \gamma \|u\|_{1,2}^2$$

for all $u \in W_2^1(\Omega)$.

Proof.

- (i) Obvious.
- (ii) We first estimate the integral over the domain Ω , which we denote by $a_\Omega(u, v)$,

$$a_\Omega(u, v) = \int_\Omega \left(D(x) \frac{\partial u}{\partial x} + \frac{\partial v}{\partial x} + \sum_1(x) u(x) v(x) \right) dx.$$

Set

$$M = \max_{x \in \Omega} \{D(x), \sum_1(x)\} = \max_{(1)} \{\sum_1, D_1\}.$$

Then, using elementary inequalities, we find

$$\begin{aligned}
|a_{\Omega}(u,v)|^2 &\leq M^2 \int_{\Omega} \left| \frac{\partial u}{\partial x} \cdot \frac{\partial v}{\partial x} + uv \right|^2 dx \\
&\leq M^2 \int_{\Omega} \left(\left| \frac{\partial u}{\partial x} \right|^2 + |u|^2 \right) \left(\left| \frac{\partial v}{\partial x} \right|^2 + |v|^2 \right) dx.
\end{aligned}$$

This can be further estimated by means of Hölder's inequality,

$$\begin{aligned}
|a_{\Omega}(u,v)|^2 &\leq M^2 \int_{\Omega} \left(\left| \frac{\partial u}{\partial x} \right|^2 + |u|^2 \right) dx \int_{\Omega} \left(\left| \frac{\partial v}{\partial x} \right|^2 + |v|^2 \right) dx \\
&= M^2 \|u\|_{1,2}^2 \|v\|_{1,2}^2.
\end{aligned}$$

Now, consider the term in $a(u,v)$ involving the surface integral, which we denote by $a_{\partial\Omega}(u,v)$,

$$a_{\partial\Omega}(u,v) = \int_{\partial_e' \Omega} \frac{b(x)}{a(x)} u(x)v(x) ds.$$

By definition, a is positive on $\partial_e' \Omega$. Moreover, b is bounded; hence, there exists a constant M' such that

$$M' = \max_{x \in \partial_e' \Omega} \frac{b(x)}{a(x)} = \max_{(1)} \frac{b_1}{a_1}.$$

Thus,

$$|a_{\partial\Omega}(u,v)|^2 \leq M'^2 \int_{\partial_e' \Omega} |uv|^2 ds,$$

or, using Hölder's inequality,

$$|a_{\partial\Omega}(u,v)|^2 \leq M'^2 \int_{\partial_e' \Omega} |u|^2 ds \int_{\partial_e' \Omega} |v|^2 ds = M'^2 \|u\|_{L_2(\partial_e' \Omega)}^2 \|v\|_{L_2(\partial_e' \Omega)}^2.$$

From Lemma 2 it follows that the imbedding of $W_2^1(\Omega)$ into

$L_2(\partial_e' \Omega)$ is bounded; hence, there exists a constant M'' such that

$$|a_{\partial\Omega}(u,v)|^2 \leq M'^2 M''^2 \|u\|_{1,2}^2 \|v\|_{1,2}^2.$$

Since $a(u,v) = a_\Omega(u,v) + a_{\partial\Omega}(u,v)$ it follows that there exists a constant $C = (M+M'M'')$ such that

$$|a(u,v)| \leq C \|u\|_{1,2} \|v\|_{1,2}$$

for all $u, v \in W_2^1(\Omega)$.

(iii) The inequality follows from elementary inequalities on the basis of the inequalities (12), with $\gamma = \min \{D_0, \int_0\}$.

It should be noted that under the weaker assumption that $\int(x) \geq 0$, $\int(x) \not\equiv 0$ on Ω , one can use Lemma 1 in order to obtain the weaker result that $a(u,u)$ is positive definite on $W_2^1(\Omega)$ with respect to the $L_2(\Omega)$ -norm, i.e., $a(u,u) \geq \gamma' \|u\|^2$ for some $\gamma' > 0$. This result is, however, not adequate for the direct application of the approach of Ciarlet, et al.

We shall now define the Galerkin-type approximation procedure. Let M_N be any finite-dimensional subspace of $W_2^1(\Omega)$. Then we have the following

Approximate Problem. To find the function $\hat{u} \in M_N$ which satisfies the equation

$$a(\hat{u}, v) = (f, v) \quad \text{for all } v \in M_N. \quad (20)$$

A solution to this approximate problem is called an *approximate solution* to the generalized problem.

In order to apply the results of Ciarlet, et al. we reconsider the definitions of generalized and approximate solution. From Lemma 3 we have the result that $a(u,v)$ is a symmetric, bounded, positive-definite bilinear functional on the Hilbert space $W_2^1(\Omega)$. By the Lax-Milgram Theorem (cf. [9], Chapter 3, Sec. 7) there exists a bounded linear operator

S on $W_2^1(\Omega)$ with a bounded linear inverse S^{-1} such that

$$a(u, v) = \langle S^{-1}u, v \rangle_{1,2}$$

and

$$\|S^{-1}\|_{1,2} \leq C, \quad \|S\|_{1,2} \leq \frac{1}{\gamma},$$

where C and γ are the constants appearing in Lemma 3. From Lemma 2, for fixed f , $h_f(v) = (f, v)$ is a bounded linear functional on $W_2^1(\Omega)$; hence, by the Riesz representation theorem, there exists an element $g \in W_2^1(\Omega)$ such that

$$(f, v) = \langle g, v \rangle_{1,2} \quad \text{for all } v \in W_2^1(\Omega).$$

If we set

$$Tu = S^{-1}u - g, \quad u \in W_2^1(\Omega),$$

then u is a solution of the generalized problem (19) if and only if

$$\langle Tu, v \rangle_{1,2} = 0 \quad \text{for all } v \in W_2^1(\Omega).$$

In the same way, $\hat{u}$ is a solution of the approximate problem (20) if and only if

$$\langle \hat{T}u, v \rangle_{1,2} = 0 \quad \text{for all } v \in M_N.$$

Now, before applying the results of Ciarlet, et al. we must verify certain properties to be satisfied by the operator T .

Lemma 4. The operator T satisfies the hypotheses H_1 , H_2^1 , and H_3^1 of Ref. 7:

- (i) T is finitely continuous;
- (ii) T is strongly monotone,

$$\langle Tu - Tv, u - v \rangle_{1,2} \geq \gamma \|u - v\|_{1,2}^2;$$

- (iii) T is Lipschitz continuous,

$$\|Tu - Tv\|_{1,2} \leq C \|u - v\|_{1,2}.$$

Here, C and γ are the same constants as in Lemma 3.

Proof.

- (i) An immediate consequence of the fact that S^{-1} is a linear bounded operator.
- (ii) Use the definition of T , the definition of S^{-1} , and the inequality $a(u-v, u-v) \geq \gamma \|u-v\|_{1,2}^2$.
- (iii) Use the definition of T , Schwarz's inequality, and the estimate $\|S^{-1}\|_{1,2} \leq C$.

The following theorem is a composite of Lemma 2.1, Theorem 2.1 and Corollary 2.1 of Ref. 7.

Theorem 2. The generalized problem (19) has a unique solution u in $W_2^1(\Omega)$. For any finite-dimensional subspace M_N of $W_2^1(\Omega)$, the approximate problem (20) has a unique solution $\hat{u}$ in M_N . The following estimate holds for the error of the approximate solution in the Sobolev norm:

$$\|\hat{u}-u\|_{1,2} \leq \gamma \|g\|_{1,2} \inf_{w \in M_N} \|w-u\|_{1,2}. \quad (21)$$

Furthermore, let $\{M_N\}_{N=1}^{\infty}$ be any sequence of finite-dimensional subspaces in $W_2^1(\Omega)$ for which

$$\lim_{N \rightarrow \infty} \left(\inf_{w \in M_N} \|w-u\|_{1,2} \right) = 0,$$

where u is the generalized solution in $W_2^1(\Omega)$. Then, if $\hat{u}_N$ is the approximate solution in M_N ,

$$\lim_{N \rightarrow \infty} \|\hat{u}_N - u\|_{1,2} = 0.$$

It should be noted that the existence and uniqueness, as well as the regularity properties, of the generalized solution in $W_2^1(\Omega)$ were already obtained in Theorem 1. Thus the value of Theorem 2 is mainly in the inequality (21).

This result completes our analysis of a Galerkin-type approximation procedure for the diffusion equation.

III. SOLUTION OF THE APPROXIMATE PROBLEM

A. Choice of a Finite-dimensional Subspace

In the previous chapter we have formulated the Galerkin-type approximation procedure; the next step therefore consists of the choice of a finite-dimensional subspace, cf. Eq. (20), and the construction of a basis for it.

Obviously, we have a considerable freedom in the choice of a finite-dimensional subspace. A large category of subspaces is generated by piecewise polynomial functions of a specified degree over the domain Ω . That is, with any such subspace there is associated a finite partition of Ω into polygonal subdomains of elementary shape (triangles, rectangles), and a specified integer, say m , such that, if u is an element of the subspace, then u is equal to a polynomial function of degree m in each polygonal subdomain of the partition. When each polynomial is uniquely determined by its behavior within its associated polygon, this approximation technique is commonly referred to as the finite element method. We will adopt this method in our investigation. As it has been described in some detail by various authors--see, e.g., [10] and [11]--we present only those aspects that are relevant for the implementation in our computer program HOD (see Chapter IV). Our presentation follows the one given by George in Ref. 11.

First of all, we must make a choice as to the nature of the partition of Ω into polygonal subdomains. As mentioned above, the most common subdomain shapes are triangles and rectangles. Now, in order to preserve the capability of solving the diffusion equation on a hexagonal reactor configuration, we will not consider the case of a rectangular partition of Ω ,

but devote our attention exclusively to the case of a triangular partition of Ω . The triangulations allowed are almost entirely arbitrary, the only restriction being that every portion of a material interface lie on an edge of some triangle.

Having made this choice we see that the finite-dimensional subspaces M_N will consist of functions u which are piecewise polynomial on $\bar{\Omega}$, and which are polynomials (of a specified degree m) on each triangle of the triangulation of Ω . Moreover, on each triangle u is completely and uniquely determined by its local behavior, that is, by its behavior within the triangle.

B. Construction of a Basis

Next, we construct a basis for the finite-dimensional subspace, M_N . Suppose that Ω has been partitioned into triangles. Let m be the desired degree of the piecewise polynomial functions in M_N . In order to uniquely determine $u \in M_N$ from its local behavior we need, in each triangle, $v = \frac{1}{2}(m+1)(m+2)$ data, as a polynomial of degree m in two variables has precisely v degrees of freedom. These data can be taken as the values of the function or of its derivatives at a certain number of points in the triangle. Polynomials which are obtained in this manner are called interpolation polynomials. In the following study we will assume that only function values, and no derivative values, are used to determine the polynomials. This type of interpolation polynomials is commonly referred to as the Lagrange-type interpolation polynomial. Thus, in order to uniquely determine a Lagrange-type interpolation polynomial over a triangle we need v reference points, which we shall take at the nodes of a regular grid over the triangle--viz., three vertex nodes, $m-1$ equally spaced nodes on each

of the three sides of the triangle, and $v-3m = \frac{1}{2}(m-1)(m-2)$ interior nodes. The total number of nodes in $\bar{\Omega}$ is then

$$N = V + (m-1)S + \frac{1}{2}(m-1)(m-2)N_{\Delta},$$

where V , S , and N_{Δ} are, respectively, the number of vertices, the number of triangle sides, and the number of triangles in the triangulation. For a typical triangulation, having many more triangle sides and vertices in the interior than on the external boundary, one has $V \approx \frac{1}{2} N_{\Delta}$ and $S \approx \frac{3}{2} N_{\Delta}$, so

$$N \approx \frac{1}{2} m^2 N_{\Delta}. \quad (22)$$

Since we associate one degree of freedom with each node, the dimension of the finite-dimensional subspace is precisely N .

With each node (global index i , $i=1, \dots, N$) in $\bar{\Omega}$ we will now associate a basis function, u_i , which has minimal support--that is, u_i vanishes outside the union of the triangles to which this i^{th} node belongs; furthermore, u_i assumes the value 1 at the i^{th} node and the value 0 at all other nodes within its support. It is easily verified that, for all $m \geq 1$, u_i is continuous across interelement boundaries, so $u \in M_N$ implies $u \in C(\bar{\Omega})$.

Next, we address ourselves to the construction of the basis function u_i . Let Γ_i denote the set of indices (γ) of those triangles (T_{γ}) to which the i^{th} node belongs. (The triangles are thought of as closed sets.) Then,

$$\text{supp } (u_i) = \bigcup_{\gamma \in \Gamma_i} T_{\gamma}.$$

On each triangle T_Y , u_i is uniquely determined in terms of its local data. Let the indices of the nodes on T_Y be $\gamma_1, \gamma_2, \dots, \gamma_v$, in some specified order--for example, vertex nodes first in counterclockwise order, followed by the side nodes also in counterclockwise order, followed by the interior nodes (in no specified order). To simplify the notation we associate with each node, in addition to its global index γ_k , $k=1, 2, \dots, v$, a local index k (local with respect to T_Y); as long as we are working on T_Y we will then use the local index, instead of the global index, and write k instead of γ_k .

Now, a polynomial of degree m has the general form

$$p(x, y) = \sum_{k=1}^v a_k \phi_k = a^T \phi = \phi^T a,$$

where ϕ is the v -dimensional column vector whose elements are the monomial terms of the general m -th degree polynomial in two variables, $v = \frac{1}{2}(m+1)(m+2)$. We assume that the terms appear in orders of non-decreasing degree and, for consecutive terms of the same degree, in order of increasing powers of y . For example, for $m=3$ ($v=10$),

$$\phi^T = (1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2, y^3).$$

The vector a contains the coefficients of p . Our goal is to express p in terms of the nodal parameters, i.e., in terms of its values at the nodes in T_Y . Let the coordinates of the nodes be denoted by $Q_k = (x_k, y_k)$, $k=1, \dots, v$, and let the vector of the nodal parameters be denoted by q , i.e., $(q^Y)^T = (q_1, \dots, q_v)$. We define the functionals $l_1, \dots, l_v$ by

$$L_i[f] = f(Q_i)$$

and denote the vector of the functionals associated with T_Y by L^Y ,

$$(L^Y)^T = (L_1, \dots, L_v).$$

Then, for p to be a Lagrange-type interpolation polynomial over T_Y we must have the identity

$$q^Y = L^Y[p],$$

from which we derive

$$q^Y = L^Y[\phi^T a] = L^Y[\phi^T] a = C^Y a.$$

For example, for $m=3$ ($v=10$) the matrix C^Y has the form

$$C^Y = \begin{bmatrix} 1 & x_1 & y_1 & x_1^2 & x_1 y_1 & y_1^2 & x_1^3 & x_1^2 y_1 & x_1 y_1^2 & y_1^3 \\ 1 & x_2 & y_2 & x_2^2 & x_2 y_2 & y_2^2 & x_2^3 & x_2^2 y_2 & x_2 y_2^2 & y_2^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{10} & y_{10} & x_{10}^2 & x_{10} y_{10} & y_{10}^2 & x_{10}^3 & x_{10}^2 y_{10} & x_{10} y_{10}^2 & y_{10}^3 \end{bmatrix}.$$

The matrix C^Y is non-singular, so

$$a = (C^Y)^{-1} q^Y,$$

and thus

$$p = (q^\gamma)^T [(C^\gamma)^{-1}]^T \phi. \quad (23)$$

This result gives the polynomial p on T_γ in terms of the parameters which we have chosen to characterize it.

Now, we apply this result to the construction of the basis function u_i , which is associated with the node with global index i . On each triangle T_γ , $\gamma \in \Gamma_i$, to which the i^{th} node belongs, u_i is represented by a polynomial of degree m , say u_i^γ .

$$u_i(x, y) = u_i^\gamma(x, y) \quad \text{for} \quad (x, y) \in T_\gamma, \quad \gamma \in \Gamma_i. \quad (24a)$$

On T_γ , u_i^γ is the unique Lagrange-type interpolant which has the value 1 at i^{th} node and the value 0 at all other nodes. If the local index of the i^{th} node on triangle T_γ is k , then we obtain u_i^γ from Eq. (23) by taking for q^γ the v -dimensional unit vector e_k , which contains a 1 in the k^{th} position and 0 for all other entries. Thus

$$u_i^\gamma(x, y) = \sum_{k=1}^v [(C^\gamma)^{-1}]_{\ell k} \phi_\ell, \quad (24b)$$

where $k \equiv k(i, \gamma)$ is the position of the i^{th} node in T_γ .

This completes the construction of a basis for the finite-dimensional subspace M_N .

C. Solution of the Approximate Problem

From here on we can proceed as usual in a Galerkin approximation procedure. The solution $\hat{u}$ of the approximate problem (20) will be of the form

$$\hat{u} = \sum_{i=1}^N q_i u_i, \quad (25)$$

where $\{u_i\}_{i=1}^N$ is the basis of the space M_N constructed in the previous section. Notice that, due to the particular construction of the basis functions, we have the identity

$$\hat{u}(Q_i) = q_i, \quad i=1, \dots, N, \quad (26)$$

so the coefficients in the expansion (25) represent the values of $\hat{u}$ at the nodes. In finite element terminology, the q_i are the generalized coordinates.

Substituting the expansion (25) into Eq. (20), and taking for v successively each of the basis functions u_i of M_N , we obtain a linear system of equations for the generalized coordinates.

$$\sum_{j=1}^N a(u_i, u_j) q_j = (f, u_i), \quad i=1, \dots, N. \quad (27)$$

If we assume that all boundary conditions are natural (i.e., they are satisfied automatically because of the design of the Ritz-Galerkin approximation procedure), then we can solve Eq. (27) for the q 's and obtain the approximate solution $\hat{u}$ by substituting the q 's into Eq. (25). If not all

boundary conditions are natural, then some of the q 's are constrained to assume certain values or satisfy certain relations. The problem of generating the coefficients $a(u_i, u_j)$ in Eq. (27) and incorporating the essential boundary conditions is examined in the next section.

D. Generation of the Coefficients

We begin by observing that the bilinear form $a(u_i, u_j)$ is non-zero only if the domains of u_i and u_j have at least one triangle in common. Denote by Γ_{ij} the set of indices (γ) of those triangles (T_γ) which belong to the intersection of the domains of u_i and u_j . Then

$$a(u_i, u_j) = \sum_{\gamma \in \Gamma_{ij}} a^\gamma(u_i, u_j), \quad (28)$$

where

$$\begin{aligned} a^\gamma(u_i, u_j) = & \int_{T_\gamma} [D(\frac{\partial u_i}{\partial x} \frac{\partial u_j}{\partial x} + \frac{\partial u_i}{\partial y} \frac{\partial u_j}{\partial y}) + [u_i u_j] dx dy \\ & + \int_{\partial_e T_\gamma} \frac{b}{a} u_i u_j ds, \end{aligned}$$

with $\partial_e T_\gamma = \{(x, y) : (x, y) \in \partial_e \Omega \cap T_\gamma\}$. For T_γ with no side along $\partial_e \Omega$, the set $\partial_e T_\gamma$ is empty and the line integral is zero. On T_γ , u_i is represented by the function u_i^γ and u_j is represented by u_j^γ , where, according to Eq. (24b),

$$\begin{aligned} u_i^\gamma(x, y) &= \sum_{k=1}^v ((C^\gamma)^{-1})_{ik} \phi_k, & k &\equiv k(i, \gamma) \\ u_j^\gamma(x, y) &= \sum_{k'=1}^v ((C^\gamma)^{-1})_{j'k'} \phi_{k'}, & k' &\equiv k'(j, \gamma). \end{aligned}$$

So

$$\begin{aligned}
 a^Y(u_i, u_j) = & \sum_{l=1}^V \sum_{l'=1}^V ((C^Y)^{-1})_{lk} ((C^Y)^{-1})_{l'k} \\
 & \times \left\{ \int_{T_Y} \left[D \left(\frac{\partial \phi_l}{\partial x} \frac{\partial \phi_{l'}}{\partial x} + \frac{\partial \phi_l}{\partial y} \frac{\partial \phi_{l'}}{\partial y} \right) + \int \phi_l \phi_{l'} \right] dx dy \right. \\
 & \left. + \int_{\partial_e T_Y} \frac{b}{a} \phi_l \phi_{l'} ds \right\}.
 \end{aligned} \quad (29)$$

Similarly,

$$(f, u_i^Y) = \sum_{Y \in \Gamma_i} (f, u_i^Y) \quad (30)$$

with

$$(f, u_i^Y) = \sum_{l=1}^V ((C^Y)^{-1})_{lk} \int_{T_Y} f \phi_l dx dy. \quad (31)$$

In particular, if $f \in M_N$, so that f has the representation

$$f = \sum_{j=1}^N f_j u_j, \quad (32)$$

we have Eq. (30) with

$$(f, u_i^Y) = \sum_{l=1}^V \sum_{l'=1}^V ((C^Y)^{-1})_{lk} f_l \int_{T_Y} \phi_l \phi_{l'} dx dy. \quad (33)$$

The coefficients $a^Y(u_i, u_j)$ and the inner products (f, u_i^Y) are most efficiently calculated by mapping the triangle T_Y onto a standard canonical

triangle, say T_0 . Let T_Y have vertices (x_1, y_1) , (x_2, y_2) and (x_3, y_3) , and let T_0 have vertices $(0,0)$, $(1,0)$, and $(0,1)$. The linear transformation mapping T_0 (ξ, η plane) onto T_Y (x, y -plane) is

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} + \begin{pmatrix} x_2 - x_1 & -(x_1 - x_3) \\ y_2 - y_1 & -(y_1 - y_3) \end{pmatrix} \begin{pmatrix} \xi \\ \eta \end{pmatrix} = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} + J_Y \begin{pmatrix} \xi \\ \eta \end{pmatrix} \quad (34)$$

The inverse mapping is given by

$$\begin{pmatrix} \xi \\ \eta \end{pmatrix} = \frac{1}{|J_Y|} \begin{pmatrix} -x_1 y_3 + x_3 y_1 \\ -x_2 y_1 + x_1 y_2 \end{pmatrix} + \frac{1}{|J_Y|} \begin{pmatrix} -(y_1 - y_3) & x_1 - x_3 \\ -(y_2 - y_1) & x_2 - x_1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}, \quad (35)$$

where $|J_Y| = \det J_Y$. The ordering of the nodes on T_Y is preserved under this mapping, so we may associate the local index $k \equiv k(i, \gamma)$ with the node on T_Y , as well as with its image on T_0 . Let ϕ be the v -dimensional column vector whose elements are the monomial terms of the general m -th degree polynomial in ξ and η . For example, for $m=3$,

$$\phi^T = (1, \xi, \eta, \xi^2, \xi\eta, \eta^2, \xi^3, \xi^2\eta, \xi\eta^2, \eta^3).$$

The unique Lagrange interpolation polynomial on T_0 which has the value 1 at the k^{th} node and the value 0 at all other nodes is given by

$$\tilde{u}_k(\xi, \eta) = \sum_{\ell=1}^v (\tilde{C}^{-1})_{\ell k} \tilde{\phi}_\ell,$$

where $\tilde{C} = I[\tilde{\phi}^T]$. Of course, $\tilde{u}_k$ with $k \equiv k(i, \gamma)$ is the image in the (ξ, η) -plane of u_i^Y under the transformation (35). Thus,

$$\frac{\partial u_1^Y}{\partial x} = \frac{\partial \xi}{\partial x} \frac{\partial \tilde{u}_k}{\partial \xi} + \frac{\partial \eta}{\partial x} \frac{\partial \tilde{u}_k}{\partial \eta} = \sum_{\ell=1}^v (\tilde{C}^{-1})_{\ell k} \left(\frac{\partial \xi}{\partial x} \frac{\partial \tilde{\phi}_\ell}{\partial \xi} + \frac{\partial \eta}{\partial x} \frac{\partial \tilde{\phi}_\ell}{\partial \eta} \right)$$

$$\frac{\partial u_1^Y}{\partial y} = \frac{\partial \xi}{\partial y} \frac{\partial \tilde{u}_k}{\partial \xi} + \frac{\partial \eta}{\partial y} \frac{\partial \tilde{u}_k}{\partial \eta} = \sum_{\ell=1}^v (\tilde{C}^{-1})_{\ell k} \left(\frac{\partial \xi}{\partial y} \frac{\partial \tilde{\phi}_\ell}{\partial \xi} + \frac{\partial \eta}{\partial y} \frac{\partial \tilde{\phi}_\ell}{\partial \eta} \right).$$

Furthermore, for any function $w(x, y)$,

$$\int_{T_Y} w(x, y) dx dy = \int_{T_O} \tilde{w}(\xi, \eta) |J_Y| d\xi d\eta,$$

and

$$\int_{\partial_e T_Y} w(x, y) ds = \int_{\partial_e T_O} \tilde{w}(\xi, \eta) \left| \frac{ds}{d\sigma} \right| d\sigma,$$

where $\tilde{w}(\xi, \eta) = w(x(\xi, \eta), y(\xi, \eta))$, and $\partial_e T_O$ is the part of ∂T_O corresponding to $\partial_e T_Y$. Thus we obtain, after some algebraic manipulation,

$$\begin{aligned} a^Y(u_i, u_j) &= \sum_{\ell=1}^v \sum_{\ell'=1}^v (\tilde{C}^{-1})_{\ell k} (\tilde{C}^{-1})_{\ell' k'} \\ &\times \int_{T_O} \left[D^Y (g_1^Y \frac{\partial \tilde{\phi}_\ell}{\partial \xi} \frac{\partial \tilde{\phi}_{\ell'}}{\partial \xi} + g_2^Y \left\{ \frac{\partial \tilde{\phi}_\ell}{\partial \xi} \frac{\partial \tilde{\phi}_{\ell'}}{\partial \eta} + \frac{\partial \tilde{\phi}_\ell}{\partial \eta} \frac{\partial \tilde{\phi}_{\ell'}}{\partial \xi} \right\} \right. \\ &\quad \left. + g_3^Y \frac{\partial \tilde{\phi}_\ell}{\partial \eta} \frac{\partial \tilde{\phi}_{\ell'}}{\partial \eta} \right] + \sum g_4^Y \tilde{\phi}_\ell \tilde{\phi}_{\ell'} \Big] d\xi d\eta \\ &+ \int_{\partial_e T_O} \left(\frac{b^Y}{a^Y} \right) g_5^Y \tilde{\phi}_\ell \tilde{\phi}_{\ell'} d\sigma, \end{aligned}$$

where $k \equiv k(i, Y)$ and $k' \equiv k'(j, Y)$ and the g 's depend upon the mapping,

$$g_1^Y = (\xi_x^2 + \xi_y^2) |J_Y| = ((x_1 - x_3)^2 + (y_1 - y_3)^2) / |J_Y|,$$

$$g_2^Y = (\xi_x \eta_x + \xi_y \eta_y) |J_Y| = ((x_1 - x_3)(x_2 - x_1) + (y_1 - y_3)(y_2 - y_1)) / |J_Y|,$$

$$g_3^Y = (\eta_x^2 + \eta_y^2) |J_Y| = ((x_2 - x_1)^2 + (y_2 - y_1)^2) / |J_Y|,$$

$$g_4^Y = |J_Y|,$$

$$g_5^Y = \left| \frac{ds}{d\sigma} \right|.$$

Since the material parameters are constant over each triangle it is convenient to introduce the following matrices:

$$G_1 = (\tilde{C}^{-1})^T \int_{T_0} \tilde{\phi}_\xi \tilde{\phi}_\xi^T d\xi d\eta \tilde{C}^{-1},$$

$$G_2 = (\tilde{C}^{-1})^T \int_{T_0} (\tilde{\phi}_\xi \tilde{\phi}_\eta^T + \tilde{\phi}_\eta \tilde{\phi}_\xi^T) d\xi d\eta \tilde{C}^{-1},$$

$$G_3 = (\tilde{C}^{-1})^T \int_{T_0} \tilde{\phi}_\eta \tilde{\phi}_\eta^T d\xi d\eta \tilde{C}^{-1},$$

$$G_4 = (\tilde{C}^{-1})^T \int_{T_0} \tilde{\phi} \tilde{\phi}^T d\xi d\eta \tilde{C}^{-1}$$

$$G_5 = (\tilde{C}^{-1})^T \int_{\partial_e T_0} \tilde{\phi} \tilde{\phi}^T d\sigma \tilde{C}^{-1}.$$

The elements of these matrices are independent of the triangle T_Y and therefore need to be computed only once. Then,

$$\begin{aligned}
 a^Y(u_1, u_j) = & D^Y[g_1^Y(G_1)_{kk'} + g_2^Y(G_2)_{kk'} + g_3^Y(G_3)_{kk'}] \\
 & + \sum^Y g_4^Y(G_4)_{kk'} + (b^Y/a^Y)g_5^Y(G_5)_{kk'}.
 \end{aligned}
 \quad (36)$$

The inner product (f, u_1^Y) can be treated in a similar manner. One has

$$(f, u_1^Y) = |J_Y| \sum_{l=1}^V (\tilde{C}^{-1})_{lk} \int_{T_0} \tilde{f} \tilde{\phi}_l d\xi d\eta \quad (37)$$

with $\tilde{f}(\xi, \eta) = f(x(\xi, \eta), y(\xi, \eta))$, where $k \equiv k(i, \gamma)$. In particular, if $f \in M_N$ and $f = \sum_{j=1}^N f_j u_j$, then

$$(f, u_1^Y) = g_6^Y \sum_{l=1}^V (G_6)_{kl} f_l, \quad k \equiv k(i, \gamma), \quad (38)$$

with

$$G_6 = (\tilde{C}^{-1})^T \int_{T_0} \tilde{\phi} \tilde{\phi}^T d\xi d\eta$$

and

$$g_6^Y = |J_Y|.$$

This completes this discussion of the generation of the system of equations (27).

As we pointed out in the previous section, if all boundary conditions are natural, then Eq. (27) can be solved for the generalized coordinates $q_1, \dots, q_N$. If not all boundary conditions are natural, then some of the q 's must be constrained in order to satisfy the boundary conditions.

In the following studies we have confined ourselves to two cases: a Dirichlet boundary condition (exterior boundary), and a Neumann boundary condition (reflective boundary). The latter is natural for the diffusion equation. The former poses constraints on those generalized coordinates which are associated with the nodes on the boundary. These constraints are incorporated most easily in the following way. Let the system of equations (27) be denoted by

$$A_q = b.$$

We partition q into q_1 and q_2 with

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}.$$

If q_2 must satisfy $q_2 = \bar{q}_2$ we solve the following system in some permuted form,

$$\begin{pmatrix} A_{11} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = \begin{pmatrix} b_1 - A_{12}\bar{q}_2 \\ \bar{q}_2 \end{pmatrix},$$

where I is the identity matrix.

IV. DESCRIPTION OF THE RESEARCH COMPUTER PROGRAM HOD

In order to evaluate the applicability of finite element methods to multigroup diffusion calculations we developed a prototype High-Order Diffusion computer program, HOD, and implemented it on the IBM System 360 Model 50/75. Current capabilities of this program are limited to the calculation of k_{eff} and power integrals. Since HOD is oriented towards fast reactor calculations, only downscattering is permitted, although arbitrary depth in the scattering is allowed.

As pointed out in Chapter I, multigroup diffusion problems, when approached via the power iteration method, lead to the solution of a system of one-group diffusion equations of the type shown in Eq. (6). For the numerical solution of these equations we modified a finite element computer program designed by J. A. George. This program, which was developed for the solution of a self-adjoint elliptic boundary value problem (Cartesian geometry) with constant coefficients, was based on the use of Lagrange-type interpolation polynomials described in the previous chapter. HOD was constructed by modifying George's program to include the case of piecewise constant coefficients, and then incorporating this modified program into a power iteration procedure. The structure of HOD is compatible with the Argonne Reactor Computation (ARC) System. Material cross sections are obtained directly from standard ARC System data files. We now give a brief review of some details of HOD.

The flux vector is ordered with respect to decreasing energy groups. The triangulation of the reactor domain Ω is almost arbitrary, with the only restriction imposed that all material interfaces lie on an edge of some triangle. The structure of the resulting finite element coefficient

matrix is, in general, sparse and this aspect can be exploited with regard to storage and computation requirements. A reverse Cuthill-McKee re-ordering is used to reduce the profile for the coefficient matrix resulting in compact storage for the unknown generalized coordinates. For each energy group, the integrations required to obtain the coefficients $a(u_i, u_j)$ of the linear system (27) are carried out triangle by triangle via a transformation to a canonical triangle. Each of the G one-group diffusion equations is solved by a direct solution technique. More specifically, a Cholesky decomposition algorithm, adapted to account for the profile storage scheme, is used to solve the linear system. The interested reader is referred to Ref. 11 for additional details. The outer iterations are accelerated by means of a three term Chebyshev extrapolation procedure. We have employed the strategy of three unextrapolated iterations before an extrapolation sequence, five extrapolated iterations before testing for extrapolation payoff (maximum of eight extrapolated iterations), and three unextrapolated iterations between extrapolation sequences. The entire calculation is contained in fast memory.

Present output capability of HOD consists of the values of the generalized coordinates, i.e., the approximate solution at the nodes of the triangulation, and the power in each region of the reactor.

V. A TIMING COMPARISON STUDY

A. Introduction

In this chapter we present the results of a detailed comparison study of the efficiency of low-order finite difference approximations versus high-order finite element approximations, for the numerical solution of the multigroup diffusion equation. The comparisons are based on the execution times for a k_{eff} -calculation with a prescribed precision, for the programs DARC2D (the two-dimensional multigroup diffusion module in the Argonne Reactor Computation (ARC) System) and HOD. The comparisons refer to three different types of reactor configurations: a simple two-zone configuration with two energy groups, a multizone configuration (1,000 MWe LMFBR) with four energy groups, and a loosely coupled configuration with two energy groups.*

B. Description of the Computer Program DARC2D

The DARC2D program is the two-dimensional multigroup diffusion module in the Argonne Reactor Computation (ARC) System [12]. The modular structure of the ARC System permits a user to selectively call upon modules for geometry input preparation, cross section homogenization, various calculational capabilities (e.g., one-dimensional or two-dimensional diffusion, one-dimensional or two-dimensional transport) and output edits. DARC2D is the production program in the ARC System for two-dimensional diffusion calculations. The algorithms used in the program are similar to those used throughout the reactor community for fast reactor analysis. In the

*In the applications in this chapter the material constants are such that $\chi_{g',g} = \chi_g$ for $g' = 1, \dots, G$.

group-ordered capability of the program, the flux vector is ordered with respect to decreasing energy groups. For each energy group, the Laplacian is approximated by a five-point finite difference approximation, the flux is ordered with respect to vertical lines running from left to right, and, within each line, from the bottom to the top. At each outer iteration, each of the G one group diffusion equations is solved iteratively for $\phi_g^{(n)}$ by means of a block Gauss-Seidel splitting of the discretized diffusion operator (that is, the operator in the left member of Eq. (4)) with respect to single vertical lines of the mesh. These inner iterations are accelerated by a three-term Chebyshev extrapolation procedure. We have employed current default strategy for the extrapolation procedure which consists of three unextrapolated inner iterations before an extrapolation sequence, three extrapolated inner iterations before testing for extrapolation payoff (maximum of six extrapolated inner iterations), and three unextrapolated inner iterations between extrapolation sequences. The inner iterations are contained in fast memory insofar as no data management is required during these iterations. In addition, the finite difference coefficients and all cross sections for every group are contained in fast memory during all inner and outer iterations. However, for each outer iteration, the number of I/O requests (read and/or write requests) is about six times the number of groups. For all problems considered in this study the number of outer iterations is twenty or less and the number of groups is four or less. The outer iterations are also accelerated using a three-term Chebyshev extrapolation procedure. Similar to the inner iterations, we have employed the current default strategy of three unextrapolated outer iterations before an extrapolation sequence, five extrapolated outer iterations before testing for extrapolation payoff (maximum of eight extrapolated

outer iterations), and three unextrapolated outer iterations between extrapolation sequences. This strategy is exactly the same as the one implemented in HOD.

C. Scope and Limitations of the Studies

The primary goal of the following studies is to compare the efficiency of high-order approximations based on finite element methods vs. low-order approximations based on finite difference methods, for the numerical solution of the multigroup diffusion equations on fast reactor configurations. The execution time for a k_{eff} -calculation with a specified precision is adopted as an objective measure of efficiency. In this section we discuss some details of the comparisons.

The comparisons are made on the basis of results obtained with the programs DARC2D and HOD described above. The results refer to the three different types of reactor configurations mentioned in the Introduction. In addition, we have analyzed a one-region configuration in conjunction with a two energy group cross section set as a program check for HOD. In the latter case, the solution of the system of multigroup diffusion equations can be obtained directly without resort to any approximation.

To apply the approximation procedure based on the five-point finite difference formula we impose a (rectangular) mesh π on a given configuration. Corresponding to this mesh there is an approximation $k(\pi)$ to k_{eff} . We refer to the difference $k(\pi) - k_{\text{eff}}$ as the truncation error in k_{eff} for the finite difference method relative to the mesh π . Similarly, to apply the approximation procedure based on an interpolatory polynomial element of degree m , we impose a triangulation τ on the same configuration. Corresponding to this triangulation and to the degree m there is an approximation $k(m, \tau)$ to k_{eff} , and we refer to the difference $k(m, \tau) - k_{\text{eff}}$ as the truncation error in k_{eff} for the finite element method relative to the degree m and the triangulation τ .

To distinguish the various meshes π and triangulations τ employed on any particular configuration we have adopted the following notational convention: π_p denotes a rectangular mesh with p mesh cells, $\tau_{p,q}$ a triangulation with pq^2 triangles. The latter is obtained from a basic triangulation $\tau_{p,1}$ with p triangles by subdividing the edges of each triangle in $\tau_{p,1}$ into q equal parts, thus generating q^2 subtriangles within each triangle of $\tau_{p,1}$. If h is the length of the largest side of all triangles in the triangulation $\tau_{p,1}$, then the length of the largest side of all triangles in the triangulation $\tau_{p,q}$ is h/q . We refer to the index q as the refinement index.

For each configuration we established a benchmark value of k_{eff} by using a quintic polynomial element ($m=5$) on a sequence $\{\tau_{p,q}\}$ of successively refined triangulations of the domain Ω . The error criterion adopted in these benchmarking runs was 10^{-10} , i.e., the power iteration procedure was terminated when the inequality $|k^{(n)} - k^{(n-1)}|/k^{(n)} < 10^{-10}$ was satisfied by a pair of successive iterates, $k^{(n-1)}$ and $k^{(n)}$. The corresponding value $k^{(n)}$ was identified with $k(m, \tau_{p,q})$ for $m = 5$. The benchmark value for k_{eff} was subsequently established on the basis of the sequence $\{k(5, \tau_{p,q})\}$. All relative truncation errors are measured with respect to the value of k_{eff} thus obtained.

All execution times quoted in the paper are the execution times needed for the iterates $k^{(n)}$ to satisfy the inequality $|k^{(n)} - k^{(n-1)}|/k^{(n)} < 10^{-5}$. The value 10^{-5} was chosen because it is the typical value used in reactor engineering applications; it is, in fact, the default value for DARC2D. In order to have an unambiguous criterion we used this criterion and no other to determine execution times. The values for k thus obtained are listed in the tables with six decimal digits. The corresponding relative truncation errors are quoted in percent.

Both DARC2D and HOD use the same power (outer) iteration procedure, including essentially the same acceleration technique on the outer iterations. However, since DARC2D retains the flux vectors corresponding to the preceding two outer iterations on a disk, there is some data management involved in each outer iteration of DARC2D, whereas there is none in HOD. Furthermore, DARC2D uses an iterative method to solve the group-ordered diffusion equations; whereas HOD uses a direct method. Since all timing measurements made on these two programs record the times needed to complete all outer iterations, they reflect therefore the net effect of three distinct phenomena. DARC2D uses a finite difference approximation combined with some data management in the outer iterations and an inner iteration technique, while HOD uses a finite element approximation with a direct method for inverting the group-ordered equations and no data management. We have made no attempt to separate these effects. In fact, since the use of a finite difference method leads to systems of equations involving a large number of unknowns, data management and the use of iterative techniques with this method is natural on third generation computers. Moreover, for the majority of design configurations, the use of a finite element method does not lead to systems of equations of large order; hence, a core-contained program with a direct technique is quite appropriate.

Finally, a few remarks must be made concerning the accuracy of the execution times quoted in this paper. All programs were executed at the Computer Center of Argonne National Laboratory, which operates a system composed of an IBM 360 Model 75 as a main processor coupled to an IBM 360 Model 50 under an attached support processor (ASP). The program environment on the Model 75 is a multiprogramming system with a variable number

of tasks (MVT). The clock is a 60-cycle routine provided by the manufacturer. In such an environment it is extremely difficult to obtain accurate and consistent execution times due to the possible presence of several programs in the main processor, as well as to the interference of various external support processors through the Model 50. Faced with this situation, we have taken extra precautions to establish some degree of reliability in the reported execution times. For DARC2D reproducible times can be obtained, since the program, as compiled for production use, requires essentially all of the fast memory in the main processor; hence, program interference is minimal. On the other hand, HOD requires a variable amount of storage; hence, for the purpose of timing we required the program to request the entire memory. However, to dwell on the accuracy of the execution times quoted in this paper would be to entirely miss the most essential point. We are comparing two very different methods, for solving the same problem, using two particular computer programs to represent each method. The gains in computational speed are certainly qualitatively correct, in the sense that the magnitudes of the execution time ratios are representative for the class of high order methods vs. the class of finite difference methods. There are faster programs than DARC2D, and there are also programs which utilize other low-order finite difference schemes. On the other hand, there are also a substantial number of other elements besides the Lagrange-type polynomial elements used in HOD; in fact, there is evidence that the latter elements are by no means optimal [13].

D. A Homogeneous Reactor Configuration

The main purpose of the following analysis is to establish the credibility of NOD. For this purpose we consider a square, homogeneous quarter-reactor configuration in conjunction with a two-energy group cross-section set. $\Omega = \{(x,y): 0 < x < L, 0 < y < L\}$, with $L = 50.05$ cm; the material constants are listed in the Appendix under the material name CORE.* In this case, the multigroup diffusion equation is exactly solvable. With $\chi_{21} = \chi_{11} = \chi_1$ and $\chi_{12} = \chi_{22} = \chi_2$ the system of equations (1) takes the form

$$-D_1 \Delta \phi_1 + \sum_1^R \phi_1 = \frac{1}{k} \chi_1 [(\nu \sum_f)_1 \phi_1 + (\nu \sum_f)_2 \phi_2] \quad (39)$$

$$-D_2 \Delta \phi_2 + \sum_2^R \phi_2 = \sum_{12} \phi_1 + \frac{1}{k} \chi_2 [(\nu \sum_f)_1 \phi_1 + (\nu \sum_f)_2 \phi_2]$$

for $(x,y) \in \Omega$, with the boundary conditions

$$\frac{\partial \phi_g}{\partial x}(0,y) = \frac{\partial \phi_g}{\partial y}(x,0) = 0 \quad (40)$$

and

$$\phi_g(L,y) = \phi_g(x,L) = 0, \text{ for } (x,y) \in \partial\Omega, \quad g = 1,2. \quad (41)$$

If we set $B^2 = 2(\frac{\pi}{2L})^2$ and $E_g = D_g B^2 + \sum_g^R$ for $g = 1,2$, then the multiplication factor k_{eff} is given by the expression

* See footnote on p. 48.

$$k_{\text{eff}} = [\chi_1 (\nu \bar{\Sigma}_f)_1 E_2 + \chi_1 (\nu \bar{\Sigma}_f)_2 \bar{\Sigma}_{12} + \chi_2 (\nu \bar{\Sigma}_f)_2 E_1] / E_1 E_2.$$

The unnormalized eigenfunction (flux) corresponding to $k = k_{\text{eff}}$ has the form

$$\phi_g(x,y) = A_g \cos \frac{\pi x}{2L} \cos \frac{\pi y}{2L} \quad \text{for } g = 1, 2,$$

where $A_1 = \alpha A_2$ with $\alpha = \chi_1 E_2 / (\chi_2 E_1 + \chi_1 \bar{\Sigma}_{12})$. If the flux is normalized to one fission per second in the entire reactor, then

$A_2 = \frac{1}{8} B^2 / [\alpha (\bar{\Sigma}_f)_1 + (\bar{\Sigma}_f)_2]$. Substituting the values of the material parameters we find the following exact values for k_{eff} , A_1 , and A_2 ,

$$k_{\text{eff}} = 1.4653 \ 87384 \ 104,$$

$$A_1 = 0.00403 \ 5664 \ 84357,$$

$$A_2 = 0.01949 \ 13021 \ 0086.$$

For the program check of HOD we used a sequence $\{\tau_{p,q}\}$ with $p = 6$ and $q = 1, 2, \dots$, of triangulations of Ω generated by the basic triangulation $\tau_{6,1}$ into the six triangles shown in Fig. 1.

The results obtained with linear ($m=1$) through quintic ($m=5$) elements are displayed in Table I. The first column contains the triangulation, $\tau_{p,q}$; the second column the number of generalized coordinates per energy group, N (i.e., the order of the linear system (27)); the third column the calculated value of the multiplication factor, $k \equiv k(m, \tau_{p,q})$. The latter is assumed to be achieved when a relative

Fig. 1. Homogeneous reactor configuration (quarter reactor) with basic triangulation $\tau_{6,1}$; the coordinates of P are $x = 20.005$, $y = 20.055$ (in units of 1 cm).

TABLE I. Results for HOD on the Homogeneous Reactor Configuration

Triangulation Number of Unknowns			Relative Truncation Error (%)	p_k	Sup-norm of Truncation Error In Flux	p_ϕ
$\tau_{p,q}$	per Group,N	k				
<i>Linear</i>						
$\tau_{6,1}$	7	1.4225 1602	$-2.9 \cdot 10^0$	-	$0.1551 \cdot 10^{-1}$	-
$\tau_{6,2}$	19	1.4532 6222	$-8.3 \cdot 10^{-1}$	1.82	$0.5932 \cdot 10^{-2}$	1.39
$\tau_{6,3}$	37	1.4598 6024	$-3.8 \cdot 10^{-1}$	1.94	$0.2991 \cdot 10^{-2}$	1.69
$\tau_{6,4}$	61	1.4622 4695	$-2.1 \cdot 10^{-1}$	1.97	$0.1809 \cdot 10^{-2}$	1.75
$\tau_{6,5}$	91	1.4633 6726	$-1.4 \cdot 10^{-1}$	1.98	$0.1218 \cdot 10^{-2}$	1.77
$\tau_{6,6}$	127	1.4639 8039	$-9.6 \cdot 10^{-2}$	1.99	$0.8799 \cdot 10^{-3}$	1.79
$\tau_{6,7}$	169	1.4643 5174	$-7.1 \cdot 10^{-2}$	1.99	$0.6673 \cdot 10^{-3}$	1.79
<i>Quadratic</i>						
$\tau_{6,1}$	19	1.4633 9429	$-1.4 \cdot 10^{-1}$	-	$0.1858 \cdot 10^{-2}$	-
$\tau_{6,2}$	61	1.4652 5178	$-9.3 \cdot 10^{-3}$	3.88	$0.2200 \cdot 10^{-3}$	3.08
$\tau_{6,3}$	127	1.4653 6003	$-1.9 \cdot 10^{-3}$	3.95	$0.6265 \cdot 10^{-4}$	3.10
$\tau_{6,4}$	217	1.4653 7865	$-6.0 \cdot 10^{-4}$	3.97	$0.2594 \cdot 10^{-4}$	3.07
$\tau_{6,5}$	331	1.4653 8379	$-2.5 \cdot 10^{-4}$	3.98	$0.1314 \cdot 10^{-4}$	3.05
$\tau_{6,6}$	469	1.4653 8564	$-1.2 \cdot 10^{-4}$	3.97	$0.7545 \cdot 10^{-5}$	3.04
$\tau_{6,7}$	631	1.4653 8644	$-6.4 \cdot 10^{-5}$	3.98	$0.4723 \cdot 10^{-5}$	3.04
<i>Cubic</i>						
$\tau_{6,1}$	37	1.4653 59766 44	$-1.9 \cdot 10^{-3}$	-	$0.1761 \cdot 10^{-3}$	-
$\tau_{6,2}$	127	1.4653 86919 30	$-3.2 \cdot 10^{-5}$	5.89	$0.1870 \cdot 10^{-4}$	3.24
$\tau_{6,3}$	271	1.4653 87342 16	$-2.9 \cdot 10^{-6}$	5.93	$0.4058 \cdot 10^{-5}$	3.77
$\tau_{6,4}$	509	1.4653 87376 55	$-5.2 \cdot 10^{-7}$	5.96	$0.1323 \cdot 10^{-5}$	3.90
$\tau_{6,5}$	721	1.4653 87382 11	$-1.4 \cdot 10^{-7}$	5.97	$0.5476 \cdot 10^{-6}$	3.96
$\tau_{6,6}$	1027	1.4653 87383 43	$-4.6 \cdot 10^{-8}$	5.95	$0.2642 \cdot 10^{-6}$	4.00

TABLE I. Results for HOD on the Homogeneous Reactor Configuration (Contd.)

Triangulation Number of Unknowns					Relative Truncation Error (%)	p_k	Sup-norm of Truncation Error In Flux	p_ϕ
$\tau_{p,q}$	per Group,N	k						
<i>Quartic</i>								
$\tau_{6,1}$	61	1.4653	86986	41802	$-2.7 \cdot 10^{-5}$	-	$0.2911 \cdot 10^{-4}$	-
$\tau_{6,2}$	217	1.4653	87382	35456	$-1.2 \cdot 10^{-7}$	7.83	$0.1186 \cdot 10^{-5}$	4.62
$\tau_{6,3}$	469	1.4653	87384	03438	$-4.8 \cdot 10^{-9}$	7.95	$0.1806 \cdot 10^{-6}$	4.64
$\tau_{6,4}$	817	1.4653	87384	09716	$-4.7 \cdot 10^{-10}$	8.05	$0.6062 \cdot 10^{-7}$	-
$\tau_{6,5}$	1261	1.4653	87384	10323	$-5.5 \cdot 10^{-11}$	-	$0.3575 \cdot 10^{-7}$	-
<i>Quintic</i>								
$\tau_{6,1}$	91	1.4653	87382	16435	$-1.3 \cdot 10^{-7}$	-	$0.1854 \cdot 10^{-5}$	-
$\tau_{6,2}$	331	1.4653	87384	08191	$-1.5 \cdot 10^{-9}$	-	$0.4611 \cdot 10^{-7}$	5.33
$\tau_{6,3}$	721	1.4653	87384	04337	$-4.1 \cdot 10^{-9}$	-	$0.2467 \cdot 10^{-7}$	-
$\tau_{6,4}$	1261	1.4653	87383	98078	$-8.4 \cdot 10^{-9}$	-	$0.2497 \cdot 10^{-7}$	-

error of 10^{-15} or less is obtained in the iterates of k in the power iteration. The fourth column contains the relative percent truncation error measured against the exact value of k_{eff} --i.e., $(\Delta k/k_{\text{eff}}) \times 100$, with $\Delta k \equiv \Delta k(m, \tau_{p,q}) = k(m, \tau_{p,q}) - k_{\text{eff}}$. Column five contains the quantity $p_k \equiv p_k(m, \tau_{p,q})$, defined by

$$p_k(m, \tau_{p,q}) = \ln \left| \frac{\Delta k(m, \tau_{p,q})}{\Delta k(m, \tau_{p,q-1})} \right| / \ln \frac{q-1}{q},$$

which is an estimate for the exponent $p_k(m)$ in the asymptotic relation

$$|k(m, \tau) - k_{\text{eff}}| = O(h^{p_k(m)}) \quad \text{as } h \rightarrow 0.$$

Columns six and seven give data pertinent to the flux. Let $\phi_g(m, \tau_{p,q})$ denote the flux for energy group g ($g=1,2$) obtained with a polynomial element of degree m on the triangulation $\tau_{p,q}$. Column six then contains the value of the quantity

$$\|\phi_g(m, \tau_{p,q}) - \phi_g\|_{\infty} = \max_{g=1,2} \max_{(x,y) \in \Omega} |\phi_g(m, \tau_{p,q})(x,y) - \phi_g(x,y)|,$$

i.e., the sup-norm of the truncation error in the flux. Column seven contains the quantity $p_{\phi} \equiv p_{\phi}(m, \tau_{p,q})$, defined by

$$p_{\phi}(m, \tau_{p,q}) = \ln \left| \frac{\Delta \phi(m, \tau_{p,q})}{\Delta \phi(m, \tau_{p,q-1})} \right| / \ln \frac{q-1}{q},$$

which is an estimate for the exponent $p_{\phi}(m)$ in the asymptotic relation

$$\|\phi_g(m, \tau) - \phi_g\|_{\infty} = O(h^{p_g(m)}) \quad \text{as } h \rightarrow 0.$$

We make the following observations with respect to the results in columns five and seven.

The multigroup system of Eqs. (39)-(41) does not correspond to a self-adjoint boundary value problem. On the other hand, every outer iteration involves the solution of two one-group diffusion equations with a given right-hand side, and each of these equations corresponds in itself to a self-adjoint, second order elliptic boundary value problem with constant coefficients. Asymptotic rates of convergence for the truncation error of the solution for such problems are known for a large class of polynomial elements. For example, it is known that for quadratic elements ($m=2$) the truncation error of the solution, in the Sobolev norm--that is, $\|\phi_g(m, \tau) - \phi_g\|_{1,2}$ for $m=2$ --has a behavior $O(h^2)$ as $q \rightarrow \infty$. [14] More generally, it is known that the truncation error, in the Sobolev norm, will behave as $O(h^m)$ for a certain class of Hermite-type polynomial elements of degree m . [15] For the linear ($m=1$) and the quadratic ($m=2$) cases, this class of elements coincides with the class of Lagrange-type interpolatory polynomial elements used in this study. Since the Sobolev norm involves a derivative, one might the truncation error, when measured by a norm such as the supremum norm, to behave as $O(h^{m+1})$.

It is of interest to note that although we are dealing with a boundary value problem which is not self-adjoint, and although the elements used in this study are not of the Hermite type (except for $m=1$ and $m=2$), these theoretical estimates appear to be achieved by some of the

elements. For example, the calculated exponents for the quadratic and cubic element appear to be converging to values very near the theoretical values of 3 and 4, respectively. On the other hand, the calculated exponent for the linear element does not appear to be converging to its theoretical value of 2. In fact, the assumed functional behavior $p_\phi(1, \tau_{6,q}) = p_\phi(1)(1-A/q^2)$ with $A = 0.405832$ fits the data rather well, and predicts the value of $p_\phi(1)$ to be approximately 1.8. The calculated exponents for the quartic and quintic elements exhibit a deterioration under refinement which we can only ascribe, somewhat vaguely, to a loss of precision in the program.

Turning now to the results in column five, we observe that it has been shown that, for a single self-adjoint elliptic eigenvalue problem, if the truncation error for the dominant eigenfunction in the Sobolev norm behaves asymptotically as $O(h^m)$, then the truncation error for the corresponding dominant eigenvalue has a behavior $O(h^{2m})$ [16]. Again, although our system of equations is not self-adjoint, we observe from the data in column five that this behavior seems to be approached, at least by the lower degree elements. For the quartic and quintic element we again observe a deterioration in the value for the calculated exponent. In this case, however, the deterioration is not unexpected for the following reason. In the IBM System 360 floating point word length is limited to 64 bits. With expected values of 8 and 10 for the respective exponents, it is clear that the accuracy in k will reach the limit of this machine's precision after very few refinements in the triangulation. Thus we see from Table I that a loss of precision has occurred with a refinement index $q = 5$ in the quartic case, and $q = 2$ in the quintic case.

E. A Simple Two-zone Reactor Configuration

In this section we discuss the efficiency of high-order vs. low-order approximations for a simple two-zone reactor configuration. The discussion is based on timing comparisons between the two programs DARC2D and HOD for the square quarter-reactor configuration shown in Fig. 2. The external dimensions are the same as before: $\Omega = \{(x,y): 0 < x < L, 0 < y < L\}$ with $L = 50.05$ cm. However, the reactor is assumed to be built up of two zones (core and blanket) with different material properties: $\Omega = \Omega^I + \Omega^{II}$ with $\Omega^I = \{(x,y): 0 < x < 20.055, 0 < y < 20.055\}$ (x and y are measured in units of 1 cm). The two-group cross section set is listed in the Appendix under the material names CORE and BLANKET.*

To establish a benchmark value for the multiplication factor k_{eff} for this configuration we performed a series of calculations using HOD with linear ($m=1$) through quintic ($m=5$) elements. The triangulations of Ω were refinements of the basic triangulation shown in Fig. 2. The results for $m = 3, 4$, and 5 are given in Table II. Column one contains the degree of the element, m ; column two the triangulation, $\tau_{p,q}$; column three the number of unknown coefficients per group. Column four contains the value of $k \equiv k(m, \tau_{p,q})$ determined by using the 10^{-10} -criterion on the relative difference of two successive iterates. We adopted the following value as the benchmark value for k_{eff} for this configuration:

$$k_{eff} = 1.19739966.$$

* See footnote on p. 48.

Fig. 2. Two-zone reactor configuration (quarter reactor) with basic triangulation $r_{6,1}$.

TABLE II. Benchmark Calculation of k_{eff} for the Two-Zone
Reactor Configuration

Degree, m	Triangulation, $\tau_{p,q}$	Number of Unknowns/Group	k	Relative Error (%)
3	$\tau_{6,2}$	127	1.1974048	$+4.3 \cdot 10^{-4}$
	$\tau_{6,3}$	271	1.1973904	$-7.7 \cdot 10^{-4}$
	$\tau_{6,4}$	509	1.1973920	$-6.4 \cdot 10^{-4}$
	$\tau_{6,5}$	721	1.1973943	$-4.5 \cdot 10^{-4}$
4	$\tau_{6,2}$	217	1.1973889	$-9.0 \cdot 10^{-4}$
	$\tau_{6,3}$	469	1.1973940	$-4.7 \cdot 10^{-4}$
	$\tau_{6,4}$	817	1.1973966	$-2.6 \cdot 10^{-4}$
	$\tau_{6,5}$	1261	1.1973980	$-1.4 \cdot 10^{-4}$
5	$\tau_{6,2}$	331	1.1973939	$-4.8 \cdot 10^{-4}$
	$\tau_{6,3}$	721	1.1973974	$-1.9 \cdot 10^{-4}$
	$\tau_{6,4}$	1261	1.19739871	$-7.9 \cdot 10^{-5}$
	$\tau_{6,5}$	1951	1.19739932	$-2.8 \cdot 10^{-5}$
	$\tau_{6,6}$	2791	1.19739966	-

Column five of Table II contains the relative percent truncation error measured against this value for k_{eff} .

For our timing comparisons we first performed a series of timing runs using the finite difference code DARC2D on five different rectangular meshes, with respectively 4, 8, 20, 32, and 48 unequally spaced intervals along each edge of the square. Figure 3 shows the distribution of the mesh lines n_{16} and n_{64} . In both the x- and y-direction the intervals are generated by placing one-half of the mesh lines in the core region, and the remaining one-half in the blanket region, and by spacing them equally in both regions. We present the results in Table III. Column one contains the mesh, column two the number of unknowns per energy group (which is equal to the number of mesh cells), column three the value of k (determined with a relative error criterion of 10^{-5} on successive iterates), column four the relative percent truncation error in k , and column five the execution time.

The smallest relative error in k for this series, 0.03%, is obtained with 2,304 mesh cells, at a cost of 212.4 seconds. Since an error of this magnitude is representative for a large class of problems encountered in reactor engineering, we adopt this case as a standard for comparisons with the results of the finite element program MOD.

As can be seen from Table II, a 0.03% relative error is very gross for finite element approximations based on the use of elements of degree 3 and up. Thus, in order to conduct a timing comparison relative to this error we must employ a very crude triangulation. The basic triangulation shown in Fig. 2 is an example of a very crude triangulation; in fact, it is the crudest possible triangulation in the sense that the least number of triangles that can be achieved by any triangulation of this reactor configuration contains precisely six triangles. However,

Fig. 3. Rectangular meshes (π_{16} and π_{64}) for the finite difference solution on the two-zone configuration.

TABLE III. Timing Results for DARC2D on the Two-Zone Reactor Configuration

Mesh	Number of Unknowns/Group	k	Relative Error (%)	Execution Time (seconds)
π_{16}	16	1.256149	$4.9 \cdot 10^0$	12.5
π_{64}	64	1.211848	$1.2 \cdot 10^0$	14.9
π_{400}	400	1.199491	$1.7 \cdot 10^{-1}$	30.7
$\pi_{1,024}$	1,024	1.198201	$6.7 \cdot 10^{-2}$	90.0
$\pi_{2,304}$	2,304	1.197753	$3.0 \cdot 10^{-2}$	212.4

we encounter a fundamental difficulty with such a crude triangulation: most approximation procedures based on the use of the Ritz-Galerkin technique, regardless of whether they are of low order or of high order type, when they involve functions defined in a piecewise manner over a polygonal subdivision of the domain, require this subdivision to be sufficiently refined before they can be applied successfully. (We call a solution successful when it produces a flux which is positive at all the nodes.) Indeed, in this application we could not achieve a successful solution with any of the elements on the basic triangulation shown in Fig. 2. With one refinement ($q=2$) of the basic triangulation, successful solutions were, however, obtained with all elements. The timing results for quadratic through quintic elements on the fixed triangulation $\tau_{6,2}$ (24 triangles) are shown in Table IV. The first column contains the degree of the element, m , and the second the number of unknowns per group. With 10^{-5} as the criterion, the value for k , its relative percent error, and the execution time appear in columns three, four, and five respectively.

TABLE IV. Timing Results for HOD on a Fixed Triangulation ($\tau_{6,2}$) of the Two-Zone Reactor Configuration

Degree, m	Number of Unknowns/Group	k	Relative Error (%)	Execution Time (seconds)
2	61	1.197017	$-3.2 \cdot 10^{-2}$	5.2
3	127	1.197406	$+5.3 \cdot 10^{-4}$	17.4
4	217	1.197390	$-8.1 \cdot 10^{-4}$	44.7
5	331	1.197395	$-3.9 \cdot 10^{-4}$	97.2

From this table it is clear that the elements of degree three or more produce errors at least two orders of magnitude less than DARC2D on the finest mesh. For this reason they are not suitable for a direct timing comparison. With a relative error of -0.032% , the quadratic element is suitable for comparison. With an execution time of 5.2 seconds, as compared to 212.4 seconds for DARC2D, we see that HOD with this element is forty (40) times faster than DARC2D.

As noted, the triangulation $\tau_{6,2}$ is too fine for elements of degree 3 or greater. Moreover, $\tau_{6,2}$ is not optimal for the quadratic elements in the sense that a relative error of 0.03% can be achieved on fewer triangles. We made an attempt to find a triangulation on which an element such as the cubic element might achieve a value for the relative error comparable to 0.03% . Maintaining a symmetric triangulation, we constructed an eight-triangle system $\tau_{8,1}$ and used it in conjunction with the cubic, quartic, and quintic elements. The triangulation is shown in Fig. 4. The cubic element was successful on this triangulation. However, the quadratic element produced a relative error larger than 0.03% in absolute value. We achieved a relative error comparable to 0.03% with the quadratic element in conjunction with the 22-triangle system $\tau_{22,1}$ shown in Fig. 5.

Table V summarizes these results (the second entry is repeated from Table IV). In the last column we give execution time ratios, which are found by dividing the execution time needed by DARC2D to achieve the relative error of 0.030% (i.e., 212.4 seconds) by the execution time needed by HOD to achieve the listed relative error, truncated to the nearest integer.

Fig. 4. Triangulation $\tau_{8,1}$ of the two-zone configuration; the coordinates of P are $x = 30.2$, $y = 30.2$ (in units of 1 cm).

Fig. 5. Triangulation $\tau_{22,1}$ of the two-zone configuration;
 P_1 : $x = 30.2$, $y = 30.2$, P_2 : $x = 45.0$, $y = 45.0$, P_3 :
 $x = 35.0525$, $y = 0$, P_4 : $x = 0$, $y = 35.0525$ (in units
of 1 cm).

TABLE V. A Comparison of Execution Times for HOD and DARC2D on the Two-Zone Reactor Configuration

Degree, m	Triangu- lation $\tau_{p,q}$	Number of Unknowns/Group	k	Relative Error (%)	Execution Time (seconds)	Execution Time Ratio
2	$\tau_{22,1}$	53	1.197049	$-2.9 \cdot 10^{-2}$	4.1	51
2	$\tau_{6,2}$	61	1.197017	$-3.2 \cdot 10^{-2}$	5.2	40
3	$\tau_{8,1}$	46	1.197855	$+3.8 \cdot 10^{-2}$	4.2	50
4	$\tau_{8,1}$	77	1.197491	$+7.3 \cdot 10^{-3}$	7.2	29
5	$\tau_{8,1}$	116	1.197402	$+2.0 \cdot 10^{-4}$	17.0	12

We observe that, by reducing the number of triangles to 22, HOD with the quadratic element is fifty-one (51) times faster than DARC2D. With the cubic element, we can reduce the number of triangles to 8 while still achieving a comparable error with a time ratio of 50:1. The quartic element on 8 triangles, although only 29 times faster, produces a relative error which is reduced by a factor of four. The quintic element is twelve (12) times faster and reduces the relative error of the standard DARC2D result by two orders of magnitude.

From Table III we see that the default criterion in DARC2D of 10^{-5} on successive iterates of k, when used in conjunction with a mesh composed of 2,304 cells, produces an estimate for k which is accurate to almost three decimal places. Hence, we are using a rather gross error criterion for these timing comparisons. It is clear from Tables IV and V, by comparison of the errors and execution times with those shown in Table III, that the ratios of the execution times will be even more pronounced when a smaller error criterion is used. Thus, for this

configuration, the values quoted for the execution time ratios represent conservative estimates.

F. A 1,000 MWe Liquid Metal Fast Breeder Reactor Configuration

In this section we discuss the efficiency of high-order vs. low-order approximations for a mockup (in Cartesian geometry) which is typical of current oxide-fueled Liquid Metal Fast Breeder Reactor (LMFBR) designs. The discussion is based on timing comparisons for the quarter-reactor configuration shown in Fig. 6. The configuration exhibits the essential characteristics of an oxide-fueled LMFBR with a full power rating of 2,400 MWe. Fuel, sodium, and stainless steel volume fractions in the core are 0.42, 0.37, and 0.21, respectively. The dimensions of the configuration are indicated in Fig. 6. The four-group cross section set is listed in the Appendix under the material names CORE REGION I, CORE REGION II, BLANKET REGION I, BLANKET REGION II, and REFLECTOR REGION.*

To determine a benchmark value for the multiplication factor, k_{eff} , for this configuration we took the quintic element in conjunction with two triangulations, $\tau_{17,1}$ and $\tau_{24,1}$. These triangulations are shown in Figs. 7 and 8, respectively. On the basis of a convergence-criterion of 10^{-10} we found $k(5, \tau_{17,1}) = 1.0569952$ and $k(5, \tau_{24,1}) = 1.0569955$. As the triangulation $\tau_{24,1}$ involves some refinement with respect to the triangulation $\tau_{17,1}$ we adopt the value of $k(5, \tau_{24,1})$ as the benchmark value for k_{eff} .

$$k_{eff} = 1.0569955.$$

Relative truncation errors are measured against this value of k_{eff} .

* See footnote on p. 48.

Fig. 6. Typical Oxide-Fueled LMFBR Configuration (quarter reactor).

Fig. 7. Triangulation $\tau_{17,1}$ of the LMFBR configuration.

Fig. 8. Triangulation $\tau_{24,1}$ of the LMFBR configuration.

For DARC2D we used seven different rectangular meshes having 6×8 , 10×14 , 14×20 , 20×28 , 25×36 , 29×40 , and 33×44 mesh intervals, respectively. The distribution of the mesh intervals in the x- and y-directions for each of these meshes is given in Table VI. Within each region the intervals are equally spaced in both the x- and y-direction. The results are presented in Table VII. Column one contains the mesh, column two the number of unknowns per energy group (which is equal to the number of mesh cells), column three the value of k (determined with the error criterion 10^{-5}), column four the relative percent truncation error in k , and column five the execution time. In the last column, column six, we give the maximum number of 64-bit words used during the execution of the program.

TABLE VI. Mesh Structures for Finite Difference Approximations of the 1,000 MW LMFBR Configuration

Mesh	x-direction (cm)				y-direction (cm)					Total Number
	0	54.85	92.95	Total	0	73.5	103.9	138.7	Total	of
	to 54.85	to 92.95	to 123.45		to 73.5	to 103.9	to 138.7	to 169.2		Mesh Cells
n_{48}	2	2	2	6	2	2	2	2	8	48
n_{140}	6	2	2	10	8	2	2	2	14	140
n_{280}	8	4	2	14	10	4	4	2	20	280
n_{560}	10	6	4	20	12	6	6	4	28	560
n_{900}	13	7	5	25	14	8	8	6	36	900
$n_{1,160}$	15	8	6	29	15	9	9	7	40	1,160
$n_{1,452}$	16	10	7	33	16	10	10	8	44	1,452

Table VII. Timing Results and Memory Requirements for DARC2D on the 1,000 MW LMFBR Configuration

Mesh	Number of Unknowns/Group	k	Relative Error (%)	Execution Time (seconds)	Number of 64-bit Words Used
π_{48}	48	1.067176	$9.6 \cdot 10^{-1}$	30.7	38,900
π_{140}	140	1.064142	$6.8 \cdot 10^{-1}$	36.4	41,000
π_{280}	280	1.058874	$1.8 \cdot 10^{-1}$	50.6	44,200
π_{560}	560	1.057842	$8.0 \cdot 10^{-2}$	87.4	50,500
π_{900}	900	1.057838	$5.2 \cdot 10^{-2}$	176.5	58,200
$\pi_{1,160}$	1,160	1.057414	$4.0 \cdot 10^{-2}$	284.7	64,100
$\pi_{1,452}$	1,452	1.057298	$2.9 \cdot 10^{-2}$	317.4	70,700

The smallest relative error in k for this series, 0.029%, is obtained with 1,452 mesh cells, at a cost of 317.4 seconds and 70,000 64-bit words. We adopt this case as a standard for comparisons with the results of the finite element program HOD, for the same reason as in the foregoing study of the two-zone configuration.

The results for HOD with quadratic through quintic elements on the triangulation $\tau_{17,1}$, and for quadratic and quintic elements on the triangulation $\tau_{24,1}$, are given in Table VIII.

Upon comparison of the data of Tables VII and VIII we observe a significant reduction of both the storage requirement and execution time for the finite element approximations. HOD with quadratic elements on seventeen triangles is approximately twenty (20) times faster than DARC2D. In addition, the storage requirements are reduced from 70,700 words to

TABLE VIII. Execution Times and Memory Requirements for HOD on the 1,000 MW LMFBR Configuration

Degree, m	Triangulation, $r_{p,q}$	Number of Unknowns/Group	k	Relative Error (%)	Execution Time (seconds)	Number of 64-bit Words Used
2	$r_{17,1}$	46	1.056700	$-2.8 \cdot 10^{-2}$	16.5	48,000
2	$r_{24,1}$	63	1.056851	$-1.3 \cdot 10^{-2}$	23.0	48,400
3	$r_{17,1}$	94	1.057007	$+1.5 \cdot 10^{-3}$	46.2	49,800
4	$r_{17,1}$	159	1.056995	$+3.3 \cdot 10^{-4}$	110.8	53,300
5	$r_{17,1}$	241	1.056992	-	246.1	59,400
5	$r_{24,1}$	336	1.056992	-	370.5	64,400

48,000 words, which represents a saving of more than 30%. For the higher order elements a timing comparison cannot be made, since even on the triangulation $r_{17,1}$ which is almost as crude as one can go in the present configuration, the relative errors are at least an order of magnitude less than the relative error in the standard DARC2D result. If DARC2D were required to achieve a much smaller relative error we expect both the timing and storage ratios to improve further in favor of the finite element program HOD.

G. A Loosely Coupled Reactor Configuration

On the reactor configuration shown in Fig. 9 the neutron flux is roughly saddle shaped. It yields a surface with a complex curvature pattern, with curves on which the flux is very nearly zero, and with a very large maximum flux gradient. Such a flux behavior is substantially different from the behavior in the LMFBR configuration studied in the preceding section. We thus have a very different type of problem, for which another timing comparison is appropriate. We refer to the configuration of Fig. 9 as a loosely coupled configuration. We have studied it

Fig. 9. Loosely coupled reactor configuration (full reactor).

in conjunction with a two-group cross section set, which is listed in the Appendix under the material names CORE and BLANKET.*

In the preceding studies we have used the execution time for a program to achieve a prescribed relative accuracy in k as the sole criterion for our timing comparisons. We used this criterion because it provides a single parameter, which is directly and easily monitored, and which is defined in the same way for both programs. To the reactor designer, integrated flux inventories are of basic importance in any reactor calculation. As an example of such an inventory we have monitored in the present study the relative error in the power in each of the two regions.

The present problem is symmetric with respect to the two diagonals. However, since DARC2D does not have any facility for taking advantage of the 45°-symmetry, we were forced to perform the calculations on the entire reactor domain Ω . On the other hand, since HOD is based on the use of a triangular mesh it is trivial to make use of the symmetry of the problem. We have indeed performed all calculations with HOD on one quarter of the reactor. For the purpose of comparison, however, we have adjusted all the data quoted in this section to the full reactor domain. In particular, all execution times quoted in the tables were obtained by multiplying the execution times for the quarter reactor by four. Since, due to the symmetry, the full reactor problem involves somewhat fewer unknowns than four quarter-reactor problems, the listed execution times represent conservative estimates for the actual execution times.

To establish benchmark values for k_{eff} and the power in each of the two regions of the configuration, we performed four calculations using HOD

* See footnote on p. 48.

with quintic elements on the triangulations $\tau_{11,1}$, $\tau_{22,1}$, $\tau_{11,2}$, and $\tau_{13,2}$. The corresponding basic triangulations $\tau_{11,1}$, $\tau_{13,1}$, and $\tau_{22,1}$ are shown in Figs. 10, 11, and 12. The results of these calculations are shown in Table IX. We adopt the following benchmark values:

$$k_{\text{eff}} = 0.750692,$$

$$A_1 = 0.789028, \text{ and}$$

$$A_2 = 0.210972,$$

where A_1 and A_2 are the power in the core and the blanket, respectively, normalized to 1 fission per second for the entire reactor.

The results of a series of timing runs with DARC2D on three different meshes are given in Table XI. The meshes have 441, 1,225, and 2,500 mesh cells, respectively, and have the same distribution of mesh intervals in the x- and y-direction to account for the symmetry with respect to the 45°-line (see Table X). We observe that the errors for the 2500-point problem are still very crude. From the behavior of the error for each of the three meshes it is clear that even under the optimistic assumption that the convergence rate is quadratic it would require more than 100 mesh intervals on a side to reduce the error for the 2500-point problem by an order of magnitude. In the preceding sections, we have worked with relative errors of this order of magnitude ($\sim 0.024\%$); however, a 10,000-point problem is extremely expensive in computer time and storage. For this reason we have not attempted to reduce the error

Fig. 10. Triangulation $\tau_{11,1}$ of the loosely coupled configuration (quarter reactor); P_1 : $x = 54$, $y = 20$, P_2 : $x = 75$, $y = 25$, P_3 : $x = 20$, $y = 20$, P_4 : $x = 38$, $y = 0$, P_5 : $x = 72$, $y = 0$ (in units of 1 cm).

Fig. 11. Triangulation $\tau_{13,1}$ of the loosely coupled configuration (quarter reactor); the coordinates of P_1 , P_2 , P_3 , and P_4 are the same as in Fig. 10, P_5 : $x = 70$, $y = 0$, P_6 : $x = 80$, $y = 8$.

Fig. 12. Triangulation $r_{22,1}$ of the loosely coupled configuration (quarter reactor); P_1 : $x = 40$, $y = 20$, P_2 : $x = 60$, $y = 20$, P_3 : $x = 50$, $y = 30$, P_4 : $x = 80$, $y = 20$, P_5 : $x = 70$, $y = 30$, P_6 : $x = 60$, $y = 40$, P_7 : $x = 40$, $y = 40$, P_8 : $x = 20$, $y = 20$, P_9 : $x = 20$, $y = 0$, P_{10} : $x = 40$, $y = 0$, P_{11} : $x = 50$, $y = 0$, P_{12} : $x = 70$, $y = 0$ (in units of 1 cm).

Table IX. Benchmark Calculation Using Quintic Elements
for the Loosely Coupled Configuration

Triangulation, $\tau_{p,q}$	Number of Triangles (for full reactor)	Number of Unknowns/Group, N	k	Power	
				Core	Blanket
$\tau_{11,1}$	44	581	0.750677	0.788997	0.211003
$\tau_{22,1}$	88	1,151	0.750682	0.789020	0.210980
$\tau_{11,2}$	176	2,261	0.750692	0.789028	0.210972
$\tau_{13,2}$	208	2,661	0.750692	0.789028	0.210972

Table X. Mesh Structures for Finite Difference Approximations
of the Loosely Coupled Configuration

Mesh	x- and y-direction (cm)						Total Number of Mesh Cells
	0	10	30	70	90	Total	
	to 10	to 30	to 70	to 90	to 100		
π_{441}	2	5	7	5	2	21	441
$\pi_{1,225}$	5	7	11	7	5	35	1,225
$\pi_{2,500}$	6	10	18	10	6	50	2,500

Table XI. Timing Results for DARC2D on the Loosely Coupled Configuration

Mesh	Number of Unknowns/Group	k	Relative Error mk (%)	Execution Time (seconds)	Relative Error in Power (%)	
					Core	Blanket
π_{441}	441	0.760479	$1.3 \cdot 10^0$	25.4	$2.2 \cdot 10^0$	$-8.1 \cdot 10^0$
$\pi_{1,225}$	1,225	0.754593	$5.2 \cdot 10^{-1}$	75.4	$8.1 \cdot 10^{-1}$	$-3.0 \cdot 10^0$
$\pi_{2,500}$	2,500	0.752490	$2.4 \cdot 10^{-1}$	201.6	$3.8 \cdot 10^{-1}$	$-1.4 \cdot 10^0$

below the 0.24% value shown in Table XI. The timing comparisons with the polynomial elements in HOD are made relative to this very crude error.

The results for HOD with quadratic through quintic elements on various triangulations are given in Table XII.

TABLE XII. Execution Times for HOD on the Loosely Coupled Configuration

Degrees	Triangulation	Number of Triangles	Number of Unknowns/Group	k	Relative Error in k (%)	Execution Time (seconds)	Relative Error in Power (%)	
							Core	Blanket
2	$T_{26,1}$	104	229	0.748941	$-2.3 \cdot 10^{-1}$	16.8	$-2.6 \cdot 10^{-1}$	$9.7 \cdot 10^{-1}$
3	$T_{22,1}$	88	427	0.750640	$-6.9 \cdot 10^{-3}$	49.7	$-2.8 \cdot 10^{-2}$	$1.1 \cdot 10^{-1}$
4	$T_{13,1}$	52	441	0.750692	-	74.0	$-2.5 \cdot 10^{-2}$	$9.5 \cdot 10^{-2}$
5	$T_{11,1}$	44	581	0.750681	$-1.5 \cdot 10^{-3}$	145.6	$-3.9 \cdot 10^{-3}$	$1.5 \cdot 10^{-2}$

These triangulations involved 11, 13, 22, and 26 triangles, respectively, on one quarter of the configuration, as displayed in Figs. 10, 11, 12, and 13, respectively. Again, just as we observed in Section E, the polynomial elements will not produce a successful solution when the triangulation is too coarse. For example, the quartic element failed on 11 triangles; yet, when it worked on 13 triangles it produced a very small relative error in k. This pattern has been consistently observed in all four cases. The results in Table XII refer to the crudest of the four triangulations considered, on which the particular element still produced a successful solution.

Comparing the execution time for DARC2D on the 2500-point mesh in Table XI (i.e., 201.6 seconds) with the execution time for HOD with the

Fig. 13. Triangulation $\tau_{26,1}$ of the loosely coupled configuration (quarter reactor); the coordinates of P_1 through P_{12} are the same as in Fig. 12, P_{13} : $x = 55$, $y = 45$, P_{14} : $x = 45$, $y = 45$, P_{15} : $x = 35$, $y = 25$, P_{16} : $x = 5$, $y = 5$, P_{17} : $x = 80$, $y = 10$ (in units of 1 cm).

quadratic element (for the same error in k) we see that we have gained a factor of twelve (12). In view of the execution time ratios reported in earlier sections this is only a modest gain. However, it is very important to observe that with HOD it is relatively easy to combine a modest gain in execution time with a significant gain in accuracy. This point is illustrated very well by the results shown in Table XII for the quartic element. Here, the calculation is approximately three times faster than DARC2D on the 2500-point mesh and k has greater accuracy. In fact, the value for k is virtually exact. However, this is somewhat fortuitous since the iterations happened to be terminated at a point where the iterate $k^{(n)}$ coincided exactly with k_{eff} to six decimal places. When the iterations were carried further we found that $k^{(n)}$ was converging to a value of 0.750686. Hence, it would be more correct to say that the quartic element produced a value for k which is accurate to four decimal places. This should be compared with the accuracy of two decimal places which was achieved in the standard DARC2D result.

Thus far in the four studies presented in this paper we have concentrated on the behavior of the multiplication factor, k . As pointed out above, this was done in order to provide a simple and unambiguous basis for comparing the two methods. However, the value of the dominant eigenvalue is not the sole object of a reactor design calculation. Reaction rate inventories and, although to a far lesser extent (especially in diffusion calculations), flux plots through various sections of the reactor domain provide additional information. Using the power in each region as an example of a typical reaction rate inventory we have displayed the relative percent errors in the power in the last two columns of Table XII.

As can be seen from the benchmark table (Table IX) the error in the core region is of most interest since this region contains about seventy-nine percent of the power. We observe that, for both programs, the error in this power is of the same order of magnitude as the error in the dominant eigenvalue. This is in contrast to the behavior observed in the study of the homogeneous reactor configuration (Section D), where the value for k converged considerably faster than any measure of the flux. A more important observation is that for both programs the error in the power is comparable to the respective error in k . Thus the results of the timing comparisons on all three multizone configurations considered in this paper should remain the same if flux inventories are used as a basis for comparison instead of k_{eff} .

The pointwise behavior of the flux, based on diffusion theory, may be of some interest if, for example, one wishes to estimate the location of flux peaks, maximum flux levels, and the ratio of maximum to minimum flux levels. In order to compare the capability of the two methods to supply this type of information we have displayed, in Fig. 14, the flux of the first energy group taken along the 45°-line from the lower left-hand corner to the midpoint of the reactor configuration. For this figure we have picked two problems with approximately equal execution times. In the case of DARC2D, this was the 441-point problem, with an execution time of approximately 26 seconds, with 11 data points for the graph. In the case of MOD, we have chosen a quadratic element on 104 triangles which had an execution time of 16.8 seconds. The results have been plotted against the flux determined from the quintic element on 208 triangles which was used for benchmarking. The vertical scale in the figure was determined by normalizing the total flux to one fission per

Fig. 14. Flux in the first energy group along the 45° line from the lower left-hand corner to the midpoint of the loosely coupled reactor configuration. The unmarked curve represents the benchmark flux obtained with the quintic element on 208 triangles.

X DARC2D data, 441 mesh cells

Δ HOD data, quadratic elements on 104 triangles.

second in the reactor.' From the figure we see that the finite difference solution overestimates the peak flux value by about 3.6% and the location by about 2.4%. The quadratic element underestimates the peak value by about 1.8% and the location by about 0.093%.

Considering the flux plot relative to the benchmark curve we observe a behavior which is common to all finite element solutions in this study; namely, there is a pattern of underestimating the peak values and overestimating the values on the slopes. Hence, when flux inventories are calculated a compensation of errors occurs which appears to be missing in the finite difference solutions. This point is illustrated very well by the above example, where the quadratic element yields an error of 0.23% in k , 0.26% in the core power, but has a maximum error of 1.8% in the pointwise flux of group one. These errors are in the approximate ratios 1:1.1:7.8 and we see that the error in the power is comparable to the error on k , in spite of the fact that the error in the flux is almost eight times greater. On the other hand, the finite difference solution exhibits very little of this effect. In this case the error in k is 1.3%, the error in the core power is 2.2%, while the error in the flux is only 3.6%. These errors are in the approximate ratios 1:1.7:2.8, which does not reflect any significant compensation of errors.

Relative to graphical displays, there is another significant advantage which is inherent in the use of the finite element method. A finite difference solution provides a finite collection of data points with essentially no information concerning the behavior of the flux between the data points. In fact, the only assumed behavior between data points which is consistent with a low order finite difference approximation is

piecewise linear. With relatively few data points this is unsatisfactory from a graphical standpoint, as Fig. 14 shows. On the other hand, the finite element method uses, for example, piecewise polynomials of a specified degree, and therefore provides a complete graph, rather than just a collection of data points.

H. Conclusions

In this chapter we have made a detailed comparison study of the efficiency of high order approximation procedures based on finite element methods, versus low order approximation procedures based on finite difference methods, for the numerical solution of the static multigroup diffusion equation. The execution time for a k_{eff} -calculation with a prescribed accuracy has been adopted as an objective measure of efficiency. All calculations have been performed with the computer programs HOD--a prototype two-dimensional multigroup high order diffusion program based on the use of finite element methods--and DARC2D--the two-dimensional multigroup diffusion module in the Argonne Reactor Computation (ARC) System based on the use of finite difference methods--on the IBM System 360 Model 50/75.

Although no claim is made that either of the two programs used in this study represents the most efficient implementation, we do maintain that they are, at least, representative for the class of high order approximation procedures and the class of low order approximation procedures, respectively, for the numerical solution of the multigroup diffusion equation on fast reactor configurations.

On this basis we have made extensive studies on three different reactor configurations, to determine k_{eff} to both engineering and benchmark accuracy. One of these studies involved a mockup of a typical oxide-fueled

LMFBR, while the other two studies involved widely variant two-zone configurations. From these studies we can draw three conclusions.

First, the use of high order approximation procedures leads to substantial execution time savings. Execution time ratios on the two-zone configurations ranged in value from twelve to over fifty, and on the LMFBR mockup we achieved the same accuracy in the multiplication factor approximately twenty times faster with HOD than with DARC2D, while using thirty percent less storage. However, to dwell on the quantitative aspects of these comparisons would be to miss the most essential point, which is qualitative in nature: namely, that the use of finite element methods is not just a viable alternative to the use of finite difference methods, but is, indeed, a significant advancement in computational capability for the numerical solution of the multigroup diffusion equation on realistic fast reactor configurations.

Second, with high order approximation procedures based on finite element methods it is possible to obtain, with reasonable cost, solutions to the multigroup diffusion equation which are sufficiently accurate that any errors can be attributed to either the diffusion theory approximation or other approximations in the reactor model, rather than to the numerical approximation procedure. In other words, whereas a determination of the truncation error in k_{eff} is very expensive when finite difference methods are used, finite element methods provide this information at a very moderate cost. For problems requiring more than three decimal places of accuracy in k_{eff} the high degree polynomial elements can yield speed advantages considerably greater than the 50:1 ratio achieved in the simple two-zone configuration studied in this paper. This conclusion is particularly relevant in perturbation calculations, where high accuracy is a necessity.

Finally, we have observed that solutions obtained with the finite element method provide flux inventories whose accuracies are comparable to the accuracy of k_{eff} .

VI. SUMMARY AND CONCLUSIONS

In this study we have investigated the applicability of a particular class of high-order approximation procedures for the numerical solution of the multigroup neutron diffusion equation on two-dimensional reactor configurations. This class is based on the use of variational principles and finite-dimensional subspaces whose elements are piecewise polynomial functions obtained by a Lagrange-type interpolation procedure over a triangulation of the domain (finite element method).

On the basis of a research High Order Diffusion computer program (HOD) we have evaluated each approximation procedure and compared its performance with that of a low-order approximation procedure based on the use of finite difference methods, for the computation of neutron distributions and inventories in reactor design studies.

Our conclusions are as follows.

- 1) The use of high-order approximation procedures presents a significant advancement in computational capability for the numerical solution of the multigroup diffusion equation on fast reactor configurations.
- 2) With high-order approximation procedures based on finite element methods it is possible to obtain, with reasonable cost, solutions to the multigroup diffusion equation which are sufficiently accurate that any errors can be attributed either to the diffusion theory approximation or other approximations in the reactor model, rather than to the numerical approximation procedure.
- 3) Solutions to the multigroup diffusion equation which are obtained with the finite element method provide flux inventories whose accuracies are comparable to the accuracy of the multiplication factor.

APPENDIX

Table of Material Constants

Material Name Group, g	Diffusion Coefficient, D_g	Removal, Σ_g^R	Prompt nu x fission, $(\nu \Sigma_f)_g$	Fission Spectrum, χ_g	Fission, $(\Sigma_f)_g$	g' Scattering, $\Sigma_{g',g}$	g' Scattering, $\Sigma_{g',g}$	g' Scattering, $\Sigma_{g',g}$	
CORE									
1	2.680004513127601	$5.457703768309187 \cdot 10^{-2}$	$3.0834480000 \cdot 10^{-2}$	0.575	$1.23338 \cdot 10^{-2}$	1	$4.079207068309187 \cdot 10^{-2}$		
2	1.578767222771633	$1.4496087600000000 \cdot 10^{-2}$	$2.52000 \cdot 10^{-2}$	0.425	$1.00800 \cdot 10^{-2}$				
BLANKET									
1	1.549689314102148	$1.120250077677641 \cdot 10^{-1}$	$5.2464120000 \cdot 10^{-2}$	0.575	$2.09856 \cdot 10^{-2}$	1	$8.898901317976410 \cdot 10^{-2}$		
2	1.094165406735201	$9.007499895999997 \cdot 10^{-3}$	$3.42000 \cdot 10^{-4}$	0.425	$1.36800 \cdot 10^{-4}$				
CORE REGION I									
1	2.854941033016231	$4.440444949778234 \cdot 10^{-2}$	$1.781057170670027 \cdot 10^{-2}$	$5.8815320000 \cdot 10^{-1}$	$5.93749 \cdot 10^{-3}$	1	$3.767320442206254 \cdot 10^{-2}$		
2	1.501986443264424	$7.571444090409197 \cdot 10^{-3}$	$4.776868736367377 \cdot 10^{-3}$	$4.0819007000 \cdot 10^{-1}$	$1.63826 \cdot 10^{-3}$	1	$1.908431512133174 \cdot 10^{-4}$	2	$4.158162747460528 \cdot 10^{-3}$
3	$9.562632076442035 \cdot 10^{-1}$	$1.072834789278052 \cdot 10^{-2}$	$6.320160180885395 \cdot 10^{-3}$	$3.6383214000 \cdot 10^{-3}$	$2.19686 \cdot 10^{-3}$	1	$1.310473892443253 \cdot 10^{-8}$	2	$3.070836489032499 \cdot 10^{-7}$
4	$9.497600617630897 \cdot 10^{-1}$	$2.838577323096713 \cdot 10^{-2}$	$2.447808863627339 \cdot 10^{-2}$	$1.9472007900 \cdot 10^{-5}$	$8.51473 \cdot 10^{-3}$			3	$1.800536554552510 \cdot 10^{-3}$
CORE REGION II									
1	2.856441914832550	$4.431183195852026 \cdot 10^{-2}$	$1.950480516165939 \cdot 10^{-2}$		$6.43424 \cdot 10^{-3}$				
2	1.502985947891542	$8.021232936249597 \cdot 10^{-3}$	$6.107673098300966 \cdot 10^{-3}$	Same as	$2.09374 \cdot 10^{-3}$	1	$3.709379184033693 \cdot 10^{-2}$		
3	$9.554640902251283 \cdot 10^{-1}$	$1.145673323354558 \cdot 10^{-2}$	$8.089060752627117 \cdot 10^{-3}$	CORE REGION I	$2.81138 \cdot 10^{-3}$	1	$1.853462348181510 \cdot 10^{-4}$	2	$4.150747089615149 \cdot 10^{-3}$
4	$9.354613665764563 \cdot 10^{-1}$	$3.209386113767015 \cdot 10^{-2}$	$3.130573911881479 \cdot 10^{-2}$		$1.08885 \cdot 10^{-2}$	1	$1.364753876153921 \cdot 10^{-8}$	2	$3.081681401847571 \cdot 10^{-7}$
REFLECTOR REGION									
1	2.945422143505490	$3.087764674574141 \cdot 10^{-2}$	-	-	-				
2	1.876715410822115	$3.055280929404975 \cdot 10^{-3}$	-	-	-	1	$3.028213782102570 \cdot 10^{-2}$		
3	$9.081416343452814 \cdot 10^{-1}$	$2.851196636008086 \cdot 10^{-3}$	-	-	-	1	$7.204660122999878 \cdot 10^{-5}$	2	$2.825184566355386 \cdot 10^{-3}$
4	$8.099749625417491 \cdot 10^{-1}$	$4.502069424075260 \cdot 10^{-3}$	-	-	-	1	0.0	2	0.0
BLANKET REGION I									
1	2.717089206580792	$4.774695615393770 \cdot 10^{-2}$	$1.412598021460720 \cdot 10^{-2}$		$4.93474 \cdot 10^{-3}$				
2	1.423965467737726	$6.451098734277303 \cdot 10^{-3}$	$8.382636267774364 \cdot 10^{-4}$	Same as	$2.90263 \cdot 10^{-4}$	1	$4.196466033938123 \cdot 10^{-2}$		
3	$9.178666262609924 \cdot 10^{-1}$	$8.723799048978818 \cdot 10^{-3}$	$1.073451903654241 \cdot 10^{-3}$	CORE REGION I	$3.73750 \cdot 10^{-4}$	1	$2.210492662465704 \cdot 10^{-4}$	2	$4.316959853234866 \cdot 10^{-3}$
4	$9.655724219540306 \cdot 10^{-1}$	$1.617300394475686 \cdot 10^{-2}$	$4.204886143315687 \cdot 10^{-3}$		$1.46507 \cdot 10^{-3}$	1	0.0	2	$1.757794569949868 \cdot 10^{-7}$
BLANKET REGION II									
1	2.514063780951626	$5.349433043386938 \cdot 10^{-2}$	$1.730073402917920 \cdot 10^{-2}$		$6.02174 \cdot 10^{-3}$				
2	1.301935010859454	$7.345761885441624 \cdot 10^{-3}$	$1.358360792262098 \cdot 10^{-3}$	Same as	$4.69205 \cdot 10^{-4}$	1	$4.652246010326033 \cdot 10^{-2}$		
3	$8.614383224022450 \cdot 10^{-1}$	$1.026816690266032 \cdot 10^{-2}$	$1.767208951913162 \cdot 10^{-3}$	CORE REGION I	$6.15299 \cdot 10^{-4}$	1	$2.564734547136148 \cdot 10^{-4}$	2	$4.694027447275414 \cdot 10^{-3}$
4	$9.018909391592444 \cdot 10^{-1}$	$1.987415555410203 \cdot 10^{-2}$	$6.920408051746308 \cdot 10^{-3}$		$2.41121 \cdot 10^{-3}$	1	0.0	2	$2.086118268360045 \cdot 10^{-7}$
								3	$1.905178592496166 \cdot 10^{-3}$

ACKNOWLEDGEMENT

The authors wish to thank Dr. J. Alan George of the University of Waterloo for making available the computer program he developed as part of his doctoral dissertation at Stanford University.

REFERENCES

- [1] M. Clark, Jr. and K. F. Hansen, Numerical Methods of Reactor Analysis, Academic Press, New York, 1964.
- [2] R. Courant and D. Hilbert, Methods of Mathematical Physics, vol. I, Interscience, New York, 1953.
- [3] L. Bers and M. Schechter, "Elliptic equations," in Partial Differential Equations by L. Bers, M. Schechter and F. John, Lectures in Applied Mathematics, vol. III, Interscience Publishers, 1964.
- [4] C. B. Morrey, Jr., "Second order elliptic equations in several variables and Hölder continuity," Math. Zeitschr. 72, 146-164 (1959).
- [5] M. K. V. Murthy and G. Stampacchia, "Boundary value problems for some degenerate-elliptic operators," Annali di Matematica, Series 4, 80, 1-22 (1968).
- [6] O. A. Ladyzhenskaja and N. N. Ural'tseva, Linejnye i kvazilinejnye uravnenija elliptičeskogo tipa, Nauka, Moscow 1954; available in English translation, Academic Press, New York 1968 (Mathematics in Science and Engineering, vol. 46), and in French translation, Dunod, Paris, 1968 (Monographies Universitaires de Mathematiques, vol. 31).
- [7] P. G. Ciarlet, M. H. Schultz, R. S. Varga, "Numerical methods of high-order accuracy for nonlinear boundary value problems. V. Monotone operator theory," Numer. Math. 13, 51-77 (1969).
- [8] S. L. Sobolev, Nekotorye primeneniya funktsionalnogo analiza v matematičeskoj fizike, Fizmatgiz, Moscow 1950; available in English translation, American Mathematical Society, Providence (R.I.) 1963 (Translations of Mathematical Monographs, vol. 7), and in German translation, Akademie-Verlag, Berlin 1964 (Mathematische Lehrbücher und Monographien, I. Abteilung, Band 12).
- [9] K. Yosida, Functional analysis, Springer Verlag, Berlin, 1965.
- [10] C. A. Felippa and R. W. Clough, "The Finite Element Method in Solid Mechanics," in Numerical Solution of Field Problems in Continuum Physics, G. Birkhoff and R. S. Varga, Eds., SIAM-AMS Proceedings, Vol. II, American Mathematical Society, Providence, R. I. (1970).
- [11] J. A. George, Computer Implementation of the Finite Element Method, STAN-CS-71-208, Stanford University, Stanford, Cal. (1971).
- [12] L. C. Just, et al., The System Aspects and Interface Data Sets of the Argonne Reactor Computation (ARC) System, ANL-7711, Argonne National Laboratory, Argonne, Ill. (1971).

- [13] M. Zlámal, Numer. Math. 17, 394 (1970).
- [14] M. Zlámal, Numer. Math. 12, 394 (1968).
- [15] A. Ženíšek, Numer. Math. 15, 283 (1970).
- [16] G. Fix, J. Math. and Mech. 18, 645 (1969).