NESTED ELEMENT METHOD IN MULTIDIMENSIONAL NEUTRON DIFFUSION CALCULATIONS

D. V. Altiparmakov
Boris Kidrič Institute of Nuclear Sciences
Nuclear Engineering Laboratory, OOUR-150
P.O.Box 522, 11001 Belgrade, Yugoslavia

A new numerical method is developed that is particularly efficient in solving the multidimensional neutron diffusion equation in geometrically complex systems. The needs for a generally applicable and fast running computer code have stimulated, here presented, the inroad of a nonclassical (R-function) numerical method into the nuclear field. By using the R-functions, the geometrical components of the diffusion problem are a priory analytically implemented into the approximate solution. The class of functions, to which the approximate solution belongs, is chosen as close to the exact solution class as practically acceptable from the time consumption point of view. That implies a drastic reduction of the number of degrees of freedom, compared to the other methods. Furthermore, the reduced number of degrees of freedom enables calculation of large multidimensional problems on small computers.

INTRODUCTION

The requirements for accuracy and computing speed of the nuclear reactor calculations have been and still are constantly growing. The following three approaches have been used to meet these requirements over the past years:

(i) Application of more powerful computers,
(ii) Development of more efficient numerical methods,
(iii) Parallel processing of the parts of calculations.

One of the most time consuming parts is the computation of the neutron flux distribution, i.e. the accurate solution of the multidimensional multigroup

neutron diffusion equation. Looking back at the last few decades one can see that tremendous efforts have been made to satisfactory solve this problem. These efforts have resulted in a variety of more or less fast running computer codes. From a large number of approximations which are in use in present day codes, one can easily single out the following three classes of methods[1-4] as the most important ones:

(i) Finite differences,
(ii) Finite elements,
(iii) Nodal methods.

The finite difference approximation is one of the most important tools in the numerical mathematics. Hence, it is the normal course of events that it has been used from the very first days of the code development for the nuclear reactor calculation purposes. The main advantage of the finite difference approximation is its mathematical simplicity. Unfortunately, having to deal with a very large number of unknowns, the classical finite difference approach becomes very time consuming in the case of large 3D problems. That fact has forced quite a number of attempts to develope alternative methods. However, it seems that the finite difference approximation is still unavoidable in several areas of the nuclear reactor calculations[5].

Under the same assumptions, the theoretical estimate of the convergence rates of finite elements is superior to that of finite differences. That fact has been numerically confirmed[6] in the case of moderate-sized test problems. In those calculations, the advantage of finite elements has been not only related to the reduced number of unknowns, but to the fact that direct solution technique was applied as well. However, the inefficiency of the direct solving of large multidimensional problems has been recognized[2-4] as the major disadvantage of the method. Significant improvements, which have been introduced in the last few years, have made the finite element codes competitive with the others.

From the large variety of benchmark calculations, it can be easily seen that the nodal methods are superior to the others, concerning the computing speed. However, most, though not all of them, are restricted to deal with rectangular configurations. It is not so clear how to extend the nodal approach to the other types of nuclear reactors. Moreover, although the nodal representation can be considered as natural in x-y plane, it is not so in the case of z-direction, even in the LWRs with rectangular fuel assemblies.

The needs for a generally applicable and fast running code have stimulated the development of the nested element method, which is presented in this paper. This method, in contrast to the others, already mentioned, belongs to

another class of methods, i.e. the nonclassical numerical methods of R-functions[7,8]. In order to follow the considerations along this paper, the main points of the R-function theory will be briefly reviewed in the next section.

AN INTRODUCTION TO THE R-FUNCTION THEORY

The early development of the R-function theory has been closely related to the solution of the boundary value problems. An approach to obtain efficient solution technique is to implement the geometrical properties of the problem analytically into the approximate solution. To this end, one has to solve the inverse problem of the analytical geometry, i.e. to construct such a function which describes the boundary of any given domain. Furthermore, that function has to be strictly positive inside, and negative everywhere outside the considered domain. That has been done in a general way by introducing the R-functions.

Let us explain the idea of the R-function in simple terms. It is such a function which properties are uniquely determined by the properties (not the magnitudes) of its arguments. From our point of view, the function properties which have to be considered are the following three: greater, equal and less than zero.

For a more rigorous approach to the R-function, let us consider the following three-valued predicate:

$$S_3(r) = \begin{cases} 2, & r \in \Omega \ ; \\ 1, & r \in \delta\Omega \ ; \\ 0, & r \notin \Omega \cup \delta\Omega \ , \end{cases} \qquad r=(x,y,z) \tag{1}$$

where Ω is a space domain, and $\delta\Omega$ its outer surface. A function $f(t_1,\ldots,t_n)$ $t \in (-\infty,\infty)$ is the R-function if there exists such a Boolean function $F(T_1,\ldots,T_n)$ that

$$S_3[f(t_1,\ldots,t_n)] = F[S_3(t_1),\ldots,S_3(t_n)] \ . \tag{2}$$

The related Boolean function has to have the same number of arguments like the R-function f itself.

A logical combination of predicates is a predicate itself. Starting from that fact and knowing the predicate equations of the components, one can easily construct the predicate equation for any complex domain by using a complete set of logical functions. To pass from the predicate to an analytical equation one just needs to choose the adequate R-functions from their infinite set.

In the three-valued Boolean algebra, a complete set of logical operations is the following one:

$$\{F(T),\ \bar{T},\ T_1 \wedge T_2\} \tag{3}$$

where $F(T) = \begin{cases} 1, & T=0 \\ 0, & (T=1) \cup (t=2) \end{cases}$,

$\bar{T}$ = negation,

$\wedge$ = conjunction.

Several complete systems of R-functions are given in sequence, where the R-conjunction has the following forms:

R_α-system: $$\omega_1 \wedge_\alpha \omega_2 \equiv \frac{1}{1+\alpha}\left(\omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2 - 2\alpha\omega_1\omega_2}\right), \tag{4}$$

R_1-system: $$\omega_1 \wedge_1 \omega_2 \equiv \frac{1}{2}(\omega_1 + \omega_2 - |\omega_1 - \omega_2|) \tag{5}$$

R_0^m-system: $$\omega_1 \wedge_0^m \omega_2 \equiv (\omega_1 + \omega_2 - \sqrt{\omega_1^2 + \omega_2^2}) \cdot (\omega_1^2 + \omega_2^2)^{m/2} \tag{6}$$

R_p-system: $$\omega_1 \underset{p}{\wedge} \omega_2 \equiv \omega_1 + \omega_2 - [\,|\omega_1|^p + |\omega_2|^p\,]^{1/p}; \qquad p>1 \tag{7}$$

R_c-system: $$\omega_1 \overset{m}{\underset{c}{\wedge}} \omega_2 \equiv \left(\frac{\omega_1 + \omega_2}{2}\right)^m \operatorname{sign}(\omega_1+\omega_2)^{m+1} - \left(\frac{\omega_1-\omega_2}{2}\right)^m \operatorname{sign}(\omega_1-\omega_2)^m \tag{8}$$

$\overset{0}{R}$-system: $$\omega_1 \overset{0}{\underset{n}{\wedge}} \omega_2 \equiv \begin{cases} \omega_1\omega_2(\omega_1^n+\omega_2^n)^{-\frac{1}{n}}, & \forall\ \omega_1>0,\ \omega_2>0\ ; \\ \omega_1, & \forall\ \omega_1\leq 0,\ \omega_2\geq 0\ ; \\ \omega_2, & \forall\ \omega_1\geq 0,\ \omega_2\leq 0\ ; \\ (-1)^{n+1}(\omega_1+\omega_2)^{\frac{1}{n}}, & \forall\ \omega_1<0,\ \omega_2<0\ . \end{cases} \tag{9}$$

In all of those systems the zero element is -1 and the R-negation has the following form:

$$\bar{\omega} \equiv -\omega \tag{10}$$

NESTED ELEMENT METHOD

For the sake of simplicity the scope of the method will be outlined for the case of approximate solution of the one-group inhomogeneous neutron diffusion equation:

$$-\nabla[D(r)\nabla\phi(r)] + \Sigma_r(r)\phi(r) = S(r), \quad r=\{x,y,z\} \in \Omega_0 \tag{11}$$

assuming that zero flux (Dirichlet type) boundary condition is imposed at the outer surface of the reactor model,

$$\phi(r) = 0 , \quad r \in \delta\Omega_0 \tag{12}$$

where

ϕ = neutron flux distribution,

S = neutron source density,

D = diffusion constant,

Σ_r = removal cross-section,

Ω_0 = space domain of the reactor model,

$\delta\Omega_0$ = outer surface of the reactor model.

Let us assume that the functions S, D an Σ_r are piecewisely continous. Thus, the diffusion solution belongs to the class of continual functions which satisfy the boundary condition (12) and have the continual (except at the internal boundaries) first and second derivatives. Moreover, at the internal boundaries (except in the singular corner points) the neutron current $(D\frac{\delta\phi}{\delta n})$ continuity has to be preserved.

The basic idea of the nested element method is to introduce an approximate solution, which belongs to a class of functions, as close to the exact solution as practically acceptable regarding the computer time consumption. That goal can be achieved by using the R-functions to construct the basis functions and consecutively applying a numerical integration to evaluate the matrix coefficients of the approximate system of linear algebraic equations. A similar method[9,10] has been already applied in the neutron diffusion calculations. However, in order to perform an analytical integration in the matrix coefficient evaluation, the basis functions have to be represented in the polynomial form. That fact implies that such a method can exclusively deal with simple geometrical domains. Moreover, by using the higher order polynomials one can expect that the roundoff error effects might become significant.

In order to construct the approximate solution let us assume that in the domain Ω_0 certain subdomains Ω_i (bounded by $\delta\Omega_i$) can be defined. There is no general rule to do that. However, the choice of subdomains should reflect the heterogeneous nature of the reactor model. More precisely, the greater the number of the subdomains introduced as a result of a more detailed description would give rise to a better approximation to the exact solution. In that sense, any homogeneous, not necessarily connected, material region (fuel assemblies of the same enrichment, fuel assemblies with control rods, reflector etc.) and any inhomogeneous region with the similar physical properties (reactor core without fuel assemblies containing control rods, reactor core etc.) can be represented by the corresponding subdomains. It is evident that some of the subdomains defined in such a way, can be nested into the others. Clearly, all of the subdomains are nested into the reactor model domain Ω_0. Let us adopt a common name for them - elements.

Using the R-functions, a function $\omega_i(r)$ with the following properies:

$$\omega_i(r) \begin{cases} > 0 , & r \in \Omega_i \\ = 0 , & r \in \delta\Omega_i \\ < 0 , & r \notin \Omega_i \cup \delta\Omega_i \end{cases} \tag{13}$$

can be related to any of the elements Ω_i. Let us now introduce the approximate solution in the following form:

$$\phi(r) = \sum_{i \in I} \sum_{j \in J_i} \phi_{ij} \frac{\omega_i(r)+|\omega_i(r)|}{2} \theta_{ij}(r) , \tag{14}$$

where

ϕ_{ij} = unknown coefficients,

θ_{ij} = trial functions,

i = element index,

I = set of element indices,

j = trial function index,

J_i = set of trial function indices related to the element i.

In simple terms, the diffusion solution is approximated with a linear combination of the functions which vanishe outside the related elements.

Any complete set of functions can be used as the set of trial functions. In that case, it has been shown[11] that the set $\omega(r)\theta_j(r)$ is a complete set on the related domain Ω, as well. Because of the simple mathematical form

the power functions $x^k y^m z^n$, i.e.

$$\Theta = \{1, x, y, z, xy, xz, yz, x^2, y^2, z^2, \ldots\} \tag{15}$$

are favourable to be used as the trial ones. The symmetry and the periodical properties of the problem have to be taken into account. For example, instead of (15), the following set has to be used in a 2D problem with x, y and diagonal symmetry:

$$\Theta = \{1, x^2+y^2, x^2y^2, x^4+y^4, x^2y^2(x^2+y^2), x^4y^4, \ldots\} . \tag{16}$$

By a proper choice of the functions ω_i, the class of functions to which the approximate solution belongs, can be made as close to the exact solution class as necessary. Let us considere the following two cases:

A. Approximate solution belongs to a less rigorous class than the exact solution (current continuity condition is not satisfied exactly at the internal boundaries).

B. Approximate solution belongs to the same class of functions as the exact solution.

In the both cases, the R-functions need to have the continual first and second derivatives, everywhere except in the singular (corner) points. From the R systems, already mentioned in the preceding section, the following four systems: $R_\alpha(\alpha \neq 1)$, R_0^m, $\underset{p}{R}$ and $\overset{0}{R}$ $(n>2)$ can satisfy that requirement. Because of its simplicity the choice of the R_0-system has been made, where:

$$R_\alpha(\alpha=0) \iff R_0^m(m=0) \iff \underset{p}{R}(p=2) . \tag{17}$$

In the case A, the domain functions ω_i have to satisfy the following additional requirement:

$$\text{grad}\ \omega_i(r) \neq 0 , \quad r \in \delta\Omega_i . \tag{18}$$

Using (18), the variational principle will determine the unknown coefficients ϕ_{ij} in such a manner that the current continuity condition is approximately satisfied at the internal boundaries.

In the case B, the current continuity at the internal boundaries has to be a priory satisfied. A general way to do that is given in reference 8 (Ch.3, §13), as follows: The variable $r=\{x,y,z\}$ in the approximate solution (14) has to be replaced by a variable $\rho=\{X,Y,Z\}$, where:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} - \sum_{\ell \in L} \frac{D_\ell - D}{D_\ell} \cdot \frac{\tilde{\omega}_\ell + |\tilde{\omega}_\ell|}{2} \cdot \begin{pmatrix} \frac{\delta \tilde{\omega}_\ell}{\delta x} \\ \frac{\delta \tilde{\omega}_\ell}{\delta y} \\ \frac{\delta \tilde{\omega}_\ell}{\delta z} \end{pmatrix} \quad . \tag{19}$$

In (19) $D > 0$ is an arbitrary quantity and $\tilde{\omega}_\ell$ is related to a homogeneous domain Ω_ℓ, where L is the set of indices of all homogeneous domains in the reactor model.

Beside (13) the domain functions ω_i and $\tilde{\omega}_\ell$ have to satisfy the following additional requirements:

$$\text{grad } \omega_i(r) \begin{cases} = 0 , & r \in \delta\Omega_i^{int} \\ \neq 0 , & r \in \delta\Omega_i^{out} \end{cases} \tag{20}$$

$$\text{grad } \tilde{\omega}_\ell(r) \begin{cases} = 1 , & r \in \delta\Omega_\ell^{int} \\ = 0 , & r \in \delta\Omega_\ell^{out} \end{cases} \tag{21}$$

where $\delta\Omega^{int}$ = internal part of the domain boundary $\delta\Omega$,

$\delta\Omega^{out}$ = outer part of the domain boundary $\delta\Omega$, i.e. $\delta\Omega^{out} \subset \delta\Omega_0$.

Both of the approximate solutions which have been considered, are based on complete sets of basis functions. However, some cases exist when an incomplete set may results in a satisfactory approximation. In that way, let us consider the following two approxim .

$$\phi(r) = \sum_{i \in I} \sum_{n \in N_i} \phi_{in} [\omega_i(r)]^n \quad , \quad N_i = \{0,1,2,\ldots\} \tag{22}$$

$$\phi(r) = \sum_{i \in I} \sum_{m \in M_i} \phi_{im} \omega_{im}(r) \quad , \quad M_i = \{0,1,2,\ldots\} \tag{23}$$

In (23) it is assumed that the domain functions ω_{im} are constructed by using different R-functions. At this stage it is not so clear how to do that. To use a complete system of R-functions which contain a variable parameter (for example R_0^m-system) it seems to be a natural approach.

EVALUATION OF THE MATRIX COEFFICIENTS

All of the approximate solutions which have been considered in the preceding section, can be written in the following general form:

$$\phi(r) = \sum_{i \in I} \sum_{j \in J_i} \phi_{ij} \xi_{ij}(r) \ . \tag{24}$$

In order to obtain the approximation skeleton $\{\phi_{ij}\}$, let us consider the following functional (weak formulation of the diffusion problem):

$$F(\psi) = \int_{\Omega} d\Omega \left\{ D(\nabla\psi)^2 + \Sigma\ \psi^2 - 2S\psi \right\} - \int_{\delta\Omega} d(\delta\Omega)\psi D \frac{\delta\psi}{\delta n} \ . \tag{25}$$

Using the method of Ritz and after a certain amount of simple algebra, the following system of linear algebraic equations can be obtained:

$$\sum_{i' \in I} \sum_{j' \in J_i} \phi_{i'j'} \sum_{\ell \in L} \int_{\Omega_\ell} d\Omega \left\{ D \left[\frac{\delta\xi_{ij}}{\delta x} \frac{\delta\xi_{i'j'}}{\delta x} + \frac{\delta\xi_{ij}}{\delta y} \frac{\delta\xi_{i'j'}}{\delta y} + \frac{\delta\xi_{ij}}{\delta z} \frac{\delta\xi_{i'j'}}{\delta z} \right] + \right.$$

$$\left. + \Sigma_r\ \xi_{ij}\, \xi_{i'j'} \right\} = \sum_{\ell \in L} \int_{\Omega_\ell} d\Omega S \xi_{ij}, \quad (i \in I,\ j \in J_i\ ). \tag{26}$$

As a consequence of the diffusion operator properties, the resulting system matrix is symmetric and positive definite. On the other hand, the matrix sparcity is influenced by the local character of some of the nested elements.

The integrals in (26), and the matrix coefficients consequently, can not be calculated analytically because of the complex mathematical form of the domain functions. In order to perform a numerical integration, any homogeneous domain Ω_ℓ of complex geometrical shape, has to be subdivided into several nonoverlaping simple subdomains, on which the cubature formulae can be applied efficiently. In that way, the calculation of the matrix and right hand side coefficients in (26) is reduced to the computation of the values of the basis functions and their derivatives in a set of space points - cubature nodes.

Let us consider the approximate solution A and B from the computational point of view. Depending on the domain complexity, quite a number of square root operations can appear during the computation of the domain functions. That is the most time consuming part of the computation. In the case A the domain functions Ω_{ij} are independent of the nuclear cross-sections. That fact implies

that the same values can be used in the matrix calculation of the geometrically identical problems. In contrast to that, in the case B, the domain functions must be recalculated if any changes of the nuclear cross-sections occur. For that reason, the approximate solution B seems to be impracticable in multigroup and burn-up calculations.

NUMERICAL RESULTS AND DISCUSSION

Based on the described method (approximation A), the prototypical computer code NESTEL has been written in FORTRAN IV. In that code, the fission source iteration technique is applied to solve the multigroup eigenvalue problem, while the method of Cholesky is used to obtain the solution of the one group systems of linear algebraic equations. The R_0 operations of conjunction, disjunction and negation are used to construct the domain functions. In that way, starting from some simple domains (half-space, infinite slab, infinite cylinder, etc.) the analytical formulae for very complex domains can be obtained. An illustration how the domain functions can be constructed is given in Figure 1. The calculation of the matrix coefficients is performed by using numerical integration on simple subdomains, where the cubature formulae are constructed as direct products of the Gaussian quadrature formulae[12]. In order to test the method and the code NESTEL itself, several calculations of the 2D IAEA benchmark problem[13] have been performed.

Four different sets of elements (defined in Table 1) have been used in the calculations. A geometrical interpretation of the elements is given in Figure 2. The results of calculations based on the approximation (23) (series of different domain functions) are given in Table 2. In that approximation, the domain functions have been constructed by using the R_0^m system (m=0,1,..) for the basic domains (rectangles), and the R_0 system for the complex domains, respectively. Assuming the zeroth order trial functions, the number of degrees of freedom of the approximation is independent on the number of space variables. The presented results show that a satisfactory accuracy (10^{-4} in k_{eff}) can be obtained by using either a less detailed description (S_5 and S_5') in conjunction with a higher order aproximation (4 terms), or a more detailed description (S_7 and S_9) and a lower number of terms (2). However, the latter approach is superior concerning the computing time. For that reason the element set S_9 has been used in the further calculations.

In contrast to (23), the approximation (22) (power series of the domain functions) is inapplicable. The corresponding results are given in Table 3, where a very poor convergence can be observed. Moreover, increasing the number of terms in the series, the order of cubature formulae must be increased as well.

The results of calculations which have been performed using the various number of trial functions per element, are given in Table 4. Another parameter in that table is the type of the basic domain functions. The R_0^1 function is a very close approximation to the analytical solution in the case of a square bare homogeneous reactor. For that reason, it has been used in the early development of the code NESTEL. However, afterwards it turn out that the R_0 system can yields better results than R_0^1. The presented results confirm that fact. Based on the R_0 system, a very low order approximation (13 unknowns) can result in a high accuracy (10^{-4} in k_{eff}).

In the presented method, the computational time is strongly dependent on the total number of cubature nodes. In that sense, it is necessary to determine the minimal number of nodes for a given accuracy. It is evident from the presented results (Tables 2 and 4) that the accuracy of k_{eff} higher than 0.5×10^{-3} can be achieved by using 4×4 nodes per numerical integration subdomain (square of 20cm 20cm).

One can notice that some of the results, which are related to the same approximations (Tables 2 and 4), are slightly different. Those differences are caused by the different starting domains (half-space or infinite slab, Figure 1) in construction of the domain functions. On the other hand, as a consequence of the different word lenght size, similar differences occur in the results of calculations carried out on different computers (CDC-3600 and VAX-11/780).

A comparison of the 2D IAEA benchmark results of various methods is given in Table 5. It is evident that for the same accuracy, the number of unknowns can be drasticaly reduced (more than one order of magnitude) by applying the nested elements compared to the finite element or nodal methods. At the same time, the computational time of the nested element method is very close to that of the nodal methods.

It is important to emphasize here, that all the calculations have been carried out using less than 32K memory (including the code itself) and without any auxiliary storage.

CONCLUSION

The main characteristics of the nested element method, presented in this paper, can be summarized as follows:

(a) Method of general application able to treat geometrically very complex configurations,

(b) The related mathematical formalism is independent on the number of space variables,

(c) A priory analytically implemented geometrical properties in the approximate solution result in a drastic reduction of the number of unknowns, and

(d) The reduced number of unknowns enables calculations of large problems on very small computers.

REFERENCES

[1] A. F. Henry, "Refinements in Accuracy of Coarse-Mesh Finite-Difference Solution of the Group-Diffusion Equations", IAEA/SM-154/21, Seminar on Numerical Reactor Calculations, Vienna, January 17-21, 1972.

[2] M. R. Wagner, "Current Trends in Multidimensional Static Reactor Calculations", Proc. of the Conference on Computational Methods in Nuclear Engineering", April 15-17, 1975, Charleston, South Carolina, Vol.I.

[3] C. H. Adams, "Current Trends in Methods for Neutron Diffusion Calculations", Nucl. Sci. & Eng., 64 (1977), p 552.

[4] R. Fröhlich, "Summary Discussion and State of the Art Review for Coarse-Mesh Computational Methods", Atomkernenergie, 30 (1977), p 152.

[5] S. P. Congdon, et al., "Three Dimensional Simulation of Large Power Reactors", Proc. of the ITM on Advances in Mathematical Methods for the Solution of Nuclear Engineering Problems, April 27-29, 1981, München, Vol.1, p 231.

[6] H. G. Kaper, et al., "A Timing Comparison Study for Some High Order Finite Element Approximation Procedures and a Low Order Finite Difference Approxximation Procedures for the Numerical Solution of the Multigroup Neutron Diffusion Equation", Nucl. Sci. & Eng., 49 (1972), p 27.

[7] V. L. Rvachev and V. A. Rvachev, "Neclassicheskiye Metody Teoriyi Pribl´izhenii v Krayevyh zadachah" ("Nonclassical Approximation Theory Methods in Boundary Value Problems"), Ch. 1 and 2, Naukova Dumka, Kiev, (1979).

[8] V. L. Rvachev, "Teoriya R-Functsiy i Nekotorye yeyo Prilozheniya" (R-Function Theory and Some Applications), Naukova Dumka, Kiev, (1982).

[9] I. S. Slesarev and A. M. Sirotkin, "Raschet Diffuzii Neytronov Metodom Vlozhenih Elementov" (Neutron Diffusion Calculation by the Inclusive Element Method), Zh.V.M.M.Ph., 16, (1976), p 399.

[10] I. S. Slesarev and A. M. Sirotkin, "Variatsionno- Raznostnie Schemi v Teoriyi Perenosa Neytronov" (Variational-Difference Schemes in Neutron Transport Theory), Atomizdat, Moscow, (1978).

[11] L. V. Kantorovich and V. I. Krylov, "Priblizhennye Metody Vysshego Analiza" (Approximate Methods of Higher Analysis), Gostehizdat, Moscow, (1962),

[12] V. I. Krilov and L. T. Shul´gina, "Spravochnaya Kniga po Chislenomu Integrirovaniyu" (Numerical Integration Handbook), Izd. Nauka, Moscow,(1966).

[13] ANL-7416, Argonne Code Center: Benchmark Problem Book, Suppl. 2, (1977).

[14] M. R. Wagner, et al., "Validation of the Nodal Expansion Method and the Depletion Program MEDIUM-2 by Benchmark Calculations and Direct Comparison with Experiment", Atomkernenergie, 30, (1977), p 129.

[15] D. M. Davierwalla, et al., "Finite Element and Nodal Methods in Diffusion", in IAEA Proc. "Transport Theora and Advanced Reactor Calculations", Ch. 5.

[16] R. D. Lawrence and J. J. Dorning, "A Nodal Green´s Function Method for Multidimensional Neutron Diffusion Calculations", Nucl. Sci. & Eng., 76, (1980), p 218.

[17] M. Makai and C. Maeder, "SEXI - A Fast Diffusion Program Based on Exponential and Trigonometric Interpolation", EIR-Bericht Nr. 401, (1980).

Table 1: Sets of Elements Used in 2D IAEA Benchmark Problem Calculations

Set	Elements	Domains	Description
S_5	1	$\Omega_1 \cup \Omega_2 \cup \Omega_3$	Fuel assemblies with control rod
	2	Ω_4	Fuel 2
	3	Ω_5	Fuel 1
	4	Ω_6	Reflector
	5	$\Omega_1 \cup \Omega_2 \cup \Omega_3 \cup \Omega_4 \cup \Omega_5 \cup \Omega_6$	Reactor
S_5'	1	$\Omega_1 \cup \Omega_2 \cup \Omega_3$	Fuel assemblies with control rods
	2	Ω_5	Fuel 1
	3	$\Omega_4 \cup \Omega_5$	Fuel 1 + fuel 2
	4	Ω_6	Reflector
	5	$\Omega_1 \cup \Omega_2 \cup \Omega_3 \cup \Omega_4 \cup \Omega_5 \cup \Omega_6$	Reactor
S_7	1	$\Omega_1 \cup \Omega_2 \cup \Omega_3$	Fuel assemblies with control rods
	2	Ω_4	Fuel 2
	3	Ω_5	Fuel 1
	4	$\Omega_4 \cup \Omega_5$	Fuel 1 + fuel 2
	5	$\Omega_1 \cup \Omega_2 \cup \Omega_3 \cup \Omega_4 \cup \Omega_5$	Reactor core
	6	Ω_6	Reflector
	7	$\Omega_1 \cup \Omega_2 \cup \Omega_3 \cup \Omega_4 \cup \Omega_5 \cup \Omega_6$	Reactor
S_9	1	Ω_1	Central fuel assembly with control rods
	2	Ω_2	Fuel assembly with control rods (x and y axis)
	3	Ω_3	Fuel assembly with control rods (diagonal)
	4	Ω_4	Fuel 2
	5	Ω_5	Fuel 1
	6	$\Omega_4 \cup \Omega_5$	Fuel 1 + fuel 2
	7	$\Omega_1 \cup \Omega_2 \cup \Omega_3 \cup \Omega_4 \cup \Omega_5$	Reactor core
	8	Ω_6	Reflector
	9	$\Omega_1 \cup \Omega_2 \cup \Omega_3 \cup \Omega_4 \cup \Omega_5 \cup \Omega_6$	Reactor

Table 2: 2D IAEA Benchmark Problem Calculations Using Various Sets of Elements and Various Number of Terms in the Series of Domain Functions

Set of elements	Number of terms	Number of cubature nodes	Number of unknowns per group	k_{eff}	Execution time (CDC-3600) (sec) matrix	iteration
S_5	1	4×4	5	1.02160	13.7	0.6
	2	4×4	10	1.02550	30.5	4.2
	3	4×4	15	1.03002	48.1	7.0
	4	4×4	20	1.02923	103.5	9.8
S'_5	1	4×4	5	1.02328	18.0	0.7
	2	4×4	10	1.02641	40.6	2.6
	3	4×4	15	1.02890	64.1	6.9
	4	4×4	20	1.02955	138.0	8.9
S_7	1	4×4	7	1.02661	26.3	4.4
	2	4×4	14	1.02964	59.5	9.1
S_9	1	4×4	9	1.02663	25.9	6.6
	1	5×5	9	1.02649	40.4	6.6
	2	4×4	18	1.02979	58.7	13.7
	2	5×5	18	1.02948	91.6	13.7

Table 3: 2D IAEA Benchmark Problem Calculations Using Element Set S_9 and Various Number of Terms in the Power Series of Domain Functions

Number of terms	Number of cubature nodes	Number of unknowns per group	k_{eff}	Computer	Execution time (sec)	
					matrix	iteration
1	4×4	9	1.02611	CDC-3600	23.3	4.9
	4×4	9	1.02614	VAX-11/780	7.2	0.7
	5×5	9	1.02586	VAX-11/780	36.5	22.1
	6×6	9	1.02583	VAX-11/780	71.4	0.4
2	4×4	18	1.02937	CDC-3600	27.8	16.0
	4×4	18	1.02940	VAX-11/780	9.2	1.4
	5×5	18	1.02809	VAX-11/780	13.4	2.4
3	4×4	27	1.03775	CDC-3600	34.6	6.6
	4×4	27	1.03772	VAX-11/780	10.2	8.6
	5×5	27	1.03188	VAX-11/780	29.8	22.0
	6×6	27	1.02798	VAX-11/780	25.8	15.1
	7×7	27	1.02846	VAX-11/780	83.4	8.3

Table 4: 2D IAEA Benchmark Problem Calculations Using two Types of R-Functions and Various Number of Trial Functions per Element (Element Set S_9)

R-function	Number of trial functions per element (elements: 1 2 3 4 5 6 7 8 9)	Number of cubature nodes	Number of unknowns per group	k_{eff}	Computer	Execution time (sec)
R_0^1	1 1 1 1 1 1 1 1 1	4×4	9	1.02046	CDC-3600	33.0
	1 1 1 1 1 1 1 1 1	6×6	9	1.02041	CDC-3600	
	1 1 1 1 1 1 3 1 3	4×4	13	1.02314	CDC-3600	
	1 1 1 1 1 1 3 1 6	4×4	16	1.03051	CDC-3600	80.0
	1 1 1 1 1 1 3 1 6	6×6	16	1.03043	CDC-3600	
	3 3 3 3 3 3 6 3 6	4×4	33	1.02954	CDC-3600	153.0
R_0	1 1 1 1 1 1 1 1 1	4×4	9	1.02611	CDC-3600	28.0
		4×4	9	1.02614	VAX-11/780	7.9
		5×5	9	1.02585	CDC-3600	40.6
		5×5	9	1.02586	VAX-11/780	
		6×6	9	1.02578	CDC-3600	56.2
		6×6	9	1.02583	VAX-11/780	
	1 1 1 1 1 1 3 1 3	4×4	13	1.02966	CDC-3600	46.5
		4×4	13	1.02983	VAX-11/780	13.5
		5×5	13	1.02945	CDC-3600	61.5
		5×5	13	1.02935	VAX-11/780	20.8
		6×6	13	1.02940	CDC-3600	81.0
		6×6	13	1.02968	VAX-11/780	21.5
		7×7	13	1.02972	VAX-11/780	31.0
		8×8	13	1.02947	VAX-11/780	39.4

TABLE 5: COMPARISON OF 2D IAEA BENCHMARK RESULTS

METHOD	PROGRAM	MESH WIDTH(cm)	APPROXIMATION	NUMBER OF UNKNOWNS	K_{eff}	COMPUTER	COMP. TIME (s) ACTUAL	COMP. TIME (s) CDC 6600	REF.
FINITE DIFFERENCES	FDM	1		12050[a]	1.029597	CDC-6600	400	400	2
	VENTURE	5	mesh centered	536[a]	1.02924	IBM-360/91	19.2	19.2	14
	- " -	2.5	- " -	2144[a]	1.02944	- " -	204	204	-"-
	- " -	1.25	- " -	8576[a]	1.02954	- " -	930	930	-"-
	- " -	0.667	- " -	23150[a]	1.02959	- " -	4800	4800	-"-
	- " -	extrapol.			1.02959	- " -			-"-
FINITE ELEMENTS	FEM2D		linear	182	1.0302	CDC-6600	6.8	6.8	2
	- " -		- " -	606	1.02973	- " -	35.8	35.8	-"-
	DIF6EN		linear	2319	1.02958	CDC-6600	400	400	15
	FINELM	20	triang. parabolic	236[a]	1.02977	CDC-6400	68	17	-"-
	- " -	20	- " - cubic	484[a]	1.02962	- " -	160	40	-"-
	TRIDENT	20	Lagrang. cubic	349	1.02969	IBM-360/91	6.3	6.3	-"-
	- " -	5	Serent. parabolic	1528	1.02958	- " -	34	34	-"-
NODAL	MEDIUM	10	NCPM	609	1.02921	CDC-6600	4.9	4.9	2
	- " -	10/1	NSM	609/2410	1.02991	- " -	15.5	15.5	-"-
	MEDIUM-2	20	NEM	175	1.03001	- " -	1.77	1.77	-"-
	- " -	10	-"-	609	1.02981	- " -	4.17	4.17	-"-
	- " -	20	NEM 5(2)		1.029657	CYBER 175	0.86	2.4	14
	- " -	10	- " -		1.029611	- " -	1.34	3.8	-"-
	- " -	3.333	- " -		1.029585	- " -	15.1	43	-"-
	NGFM	20			1.029595	- " -	1.8	5.1	16
	- " -	10			1.029599	- " -	4.4	12.5	-"-
	- " -	5			1.029584	- " -	16.8	47.8	-"-
	NODLEG	20	5(2)[b]		1.02959	CDC-6400	18.1	4.5	17
	QUANDRY	20	∞(2)		1.02962	IBM-370/168	1.2	1.2	-"-
	TWO STEP	20	∞(0)		1.02970	- " -	1.4	1.4	-"-
	SEXI	20	∞(0)		1.03005	CDC-6400	13.5	3.5	-"-
	MUSIC	20	∞		1.02986	CYBER 175	1.8	5.1	-"-
NESTED ELEMENTS	NESTEL	20	S_8:111111111;(4×4)	9	1.02611	CDC-3600	28.2	2.8	
	- " -	-"-	- " -	9	1.02614	VAX-11/780	7.9		
	- " -	-"-	S_8:111111313;(4×4)	13	1.02966	CDC-3600	46.5	4.6	
	- " -	-"-	- " -	13	1.02983	VAX-11/780	13.5		

[a] The number of unknowns has been assumed by the author of this paper according to the given data

[b] N(M) denotes: N = order of flux approximation, M = order of cross-leakage approximation, ∞ = analytical solution

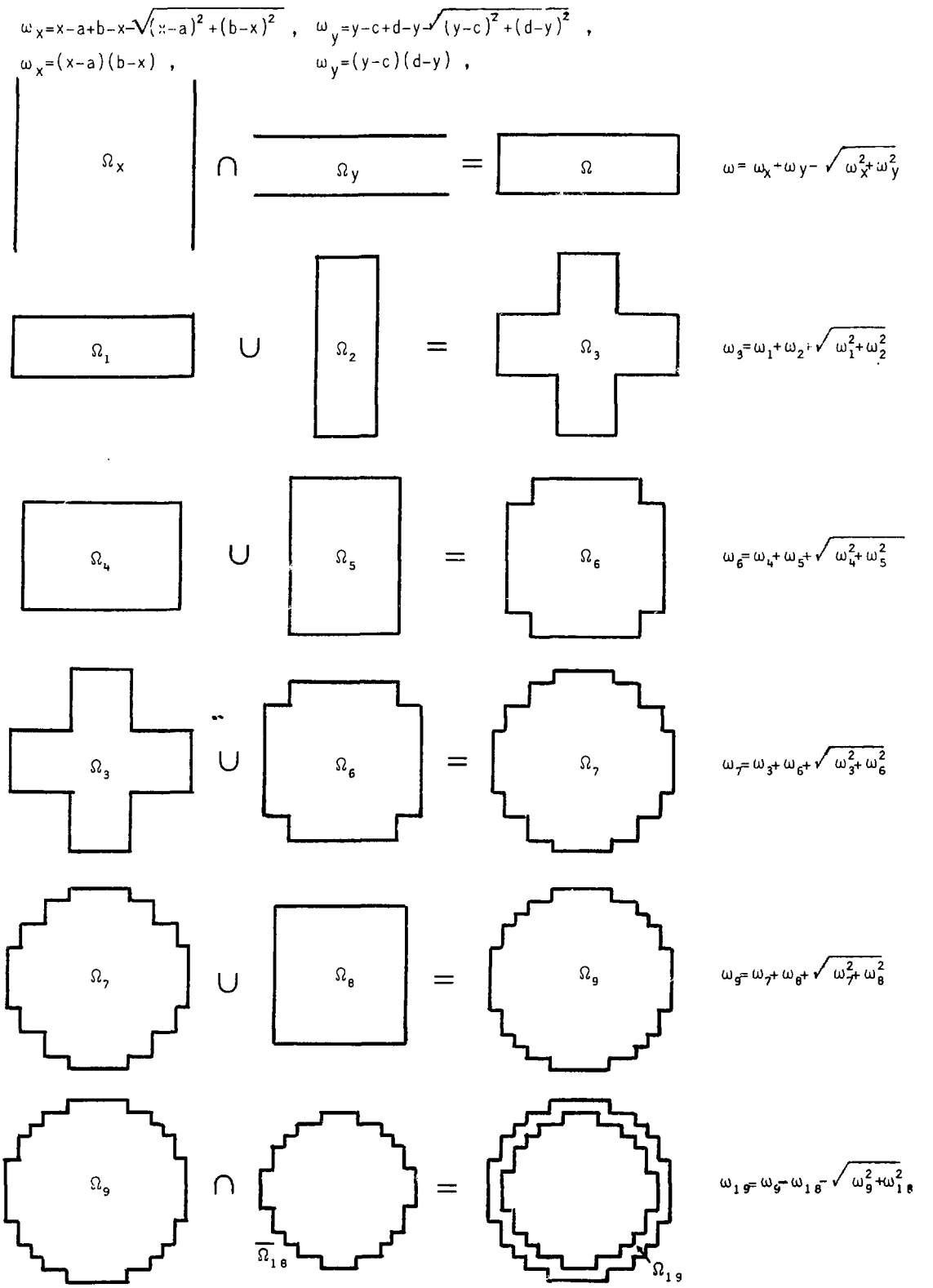

Figure 1: An Illustration how the Domain Functions can be Constructed (Example: The Domain of Fuel 1 in 2D IAEA Benchmark Problem)

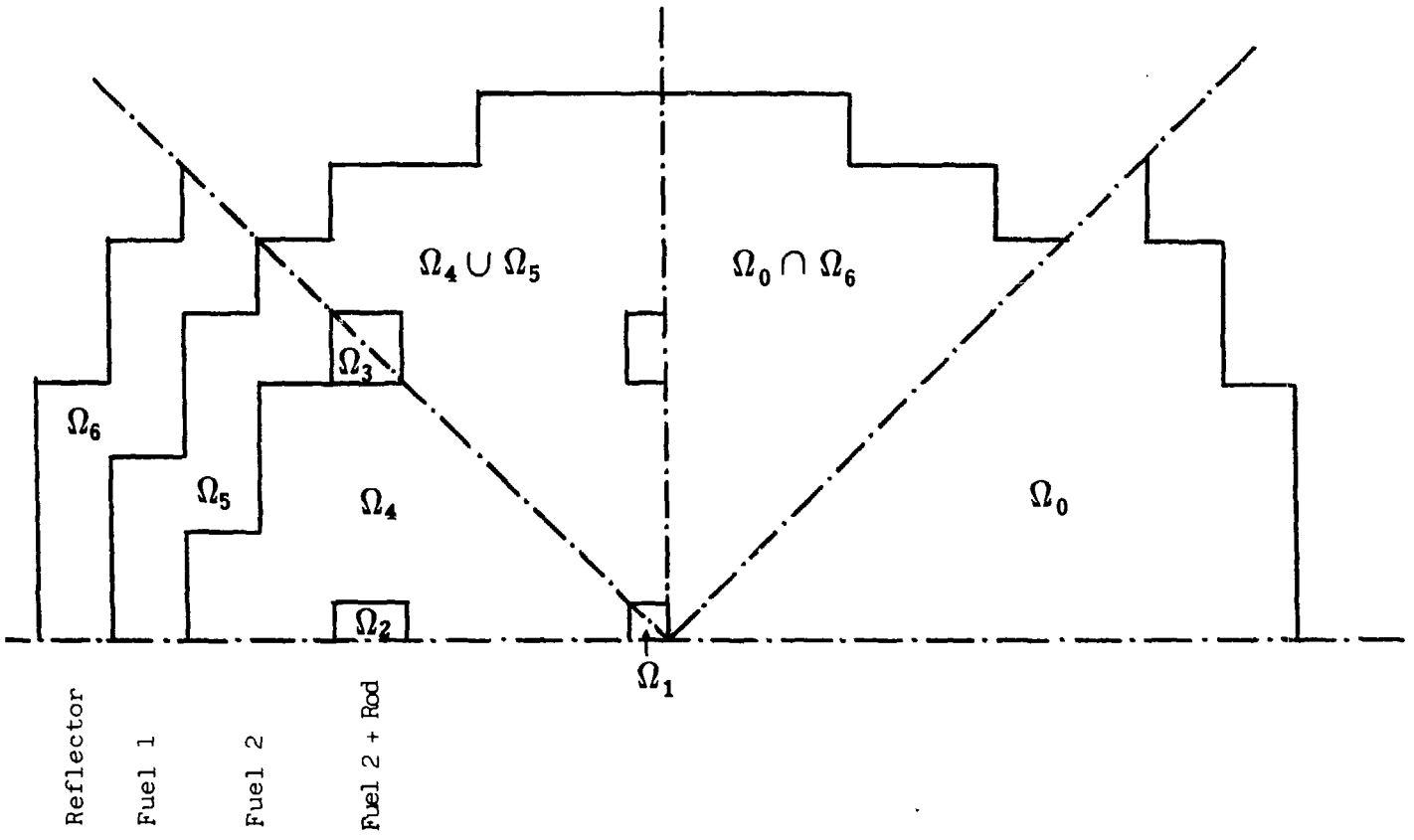

Figure 2: Nested Elements Used in 2D IAEA Benchmark Problem Calculations