Computer Physics Communications 42 (1986) 197–216
North-Holland, Amsterdam

GRENADE – A COARSE-MESH REACTOR PHYSICS PROGRAM TO SOLVE THE STATIC DIFFUSION EQUATION FOR NEUTRONS

T.A. BEU *, D.I. SIMIONOVICI and V.N. ANGHEL

Institute for Nuclear Power Reactors, P.O. Box 78, Pitesti, Romania

Received 15 June 1985; in final form 7 March 1986

PROGRAM SUMMARY

Title of program: GRENADE

Catalogue number: AALJ

Program obtainable from: CPC Program Library, Queen's University of Belfast, N. Ireland (see application form in this issue)

Computer: CDC Cyber 170/720; *Installation*: Institute for Nuclear Power Reactors, Pitesti, Romania

Operating system: NOS 1P4 552/552

Programming language used: FORTRAN IV

High speed storage required: about 26 Kwords program length + variable storage (see section 4.2)

Number of bits in a word: 60

Number of lines in combined program and test deck: 2864

Keywords: diffusion equation, nodal method, eigenvalue problem, neutron, fission source iteration, extrapolation, coarse-mesh rebalancing

Nature of physical problem
The program is designed to solve the static multigroup diffusion equation for neutrons in multidimensional problems, assuming Cartesian geometry. The program yields flux- and power distributions and the effective neutron multiplication factor (k_{eff}).

Method of solution
GRENADE (GREen's Function Nodal Algorithm for the Diffusion Equation) is based on the linear form of the nodal balance equation written in terms of the average net interface currents across the surface of a subdomain (node). Green's functions for the one-dimensional in-group diffusion-removal operator are used to generate a coupled set of one-dimensional integral equations defined over a node. These integral equations represent an exact (local) solution to the coupled set of one-dimensional differential equations obtained by spatially integrating the multidimensional diffusion equation over directions transverse to each coordinate direction.

The integral equations are approximated using a weighted residual procedure. The resulting matrix equations, when solved in conjunction with the linear form of the nodal balance equation, provide the necessary additional relationships between the net interface currents and the flux within a node.

Restrictions on the complexity of the problem
The size of the reactor model that can be handled is limited by computer core capacity.

Typical running time
On CDC CYBER 170/720 the solution of the two-dimensional two-group IAEA benchmark problem [2] with a 9×9 points mesh grid took about 9 s CPU time.

References
[1] R.D. Lawrence and J.J. Dorning, Nucl. Sci. Eng., 76 (1980) 218.
[2] Argonne Code Center, Benchmark Problem Book (ANL-7416, Argonne National Laboratory, 1977).

* Present address: Institute of Isotopic and Molecular Technology, R-3400 Cluj-Napoca 5, P.O. Box 700, Romania; author to whom correspondence should be addressed.

LONG WRITE-UP

1. Introduction

The calculation of the power produced throughout a nuclear reactor typically begins with the evaluation of equivalent homogenized group diffusion parameters for relatively large subregions of the reactor. These parameters can then be used in the subsequent computation of the global power distribution by solving the diffusion equation for this "homogenized-assembly" reactor model. The use of finite difference techniques for the solution of such global problems turns out to be extremely computer time and memory consuming, due to the very large number of unknowns required to achieve acceptable accuracy. One very efficient approach for the solution of such problems, that has been in use lately in different variants, is the nodal method [1].

GRENADE (Green's Function Nodal Algorithm for the Diffusion Equation), the algorithm on which GRENADE computer code is based, is in many aspects similar to NGFM (Nodal Green's Function Method) [2]. The description of GRENADE, given in section 2, follows that of ref. [2]. The main difference between GRENADE and NGFM consists in the usage of different nodal Green's functions. In addition, NGFM uses incoming and outgoing partial currents at the node surfaces, while GRENADE uses half as many net currents. This aspect makes GRENADE more efficient in terms of computer time and memory.

2. Description of the Green's function nodal algorithm for the diffusion equation

The time independent diffusion equation can be written in the standard multigroup form:

$$-\nabla \cdot D_g(\boldsymbol{r})\ \nabla\phi_g(\boldsymbol{r}) + \Sigma_g^{\mathrm{r}}(\boldsymbol{r})\phi_g(\boldsymbol{r}) = Q_g(\boldsymbol{r}), \tag{1}$$

where D_g is the diffusion coefficient, ϕ_g is the flux and Σ_g^{r} is the removal cross section:

$$\Sigma_g^{\mathrm{r}}(\boldsymbol{r}) = \Sigma_g^{\mathrm{a}}(\boldsymbol{r}) + \sum_{\substack{g'=1 \\ g'\neq g}}^{G} \Sigma_{gg'}(\boldsymbol{r}),$$

with Σ_g^{a} denoting the absorption cross section and $\Sigma_{gg'}$ the scattering cross section. The total group source is defined by the following relations:

$$\begin{aligned} Q_g(\boldsymbol{r}) &= Q_g^{\mathrm{s}}(\boldsymbol{r}) + \frac{\chi_g}{k_{\mathrm{eff}}} Q^{\mathrm{f}}(\boldsymbol{r}), \\ Q_g^{\mathrm{s}}(\boldsymbol{r}) &= \sum_{\substack{g'=1 \\ g'\neq g}}^{G} \Sigma_{g'g}(\boldsymbol{r})\phi_{g'}(\boldsymbol{r}), \\ Q^{\mathrm{f}}(\boldsymbol{r}) &= \sum_{g'=1}^{G} \left(\nu\Sigma_{g'}^{\mathrm{f}}(\boldsymbol{r})\right)\phi_{g'}(\boldsymbol{r}), \end{aligned} \tag{2}$$

where Q_g^{s} is the scattering source, Q^{f} is the total fission source, χ_g is the fission spectrum, $(\nu\Sigma_g^{\mathrm{f}})$ is the production cross section. Eq. (1) is solved subject to suitable boundary conditions and with the usual continuity conditions across internal surfaces.

2.1. The nodal equations

Assuming Cartesian geometry, we partition the reactor volume V in K disjoint homogeneous boxes (nodes) of volume V_k, $k = 1, 2, \ldots, K$. Integrating eq. (1) over a box V_k yields the nodal balance equation, which can be written in terms of the face-averaged net currents:

$$\sum_{u=x,y,z} \frac{1}{2a_u^k}\left[J_{gu}^k(a_u^k) - J_{gu}^k(-a_u^k)\right] + \Sigma_g^{r,k}\bar{\phi}_g^k = \bar{Q}_g^k, \quad g = 1, 2, \ldots, G;\ k = 1, 2, \ldots, K. \tag{3}$$

a_u^k, $u = x, y, z$, denote the node half widths, i.e.

$$V_k = \left[-a_x^k, a_x^k\right] \times \left[-a_y^k, a_y^k\right] \times \left[-a_z^k, a_z^k\right].$$

The average (nodal) flux is defined by:

$$\bar{\phi}_g^k \equiv \frac{1}{V_k}\int_{V_k} \mathrm{d}^3 r \phi_g(\boldsymbol{r}).$$

The face-averaged net currents across the node surfaces perpendicular to the u-direction are given by:

$$J_{gu}^k(-a_u^k) = -D_g^k\left[\frac{\mathrm{d}}{\mathrm{d}u}\phi_{gu}^k(u)\right]_{u=-a_u^k},$$

$$J_{gu}^k(a_u^k) = -D_g^k\left[\frac{\mathrm{d}}{\mathrm{d}u}\phi_{gu}^k(u)\right]_{u=a_u^k}, \tag{4}$$

where $\phi_{gu}^k(u)$ is a partially integrated flux:

$$\phi_{gu}^k(u) \equiv \frac{1}{4a_v^k a_w^k}\int_{-a_v^k}^{a_v^k} \mathrm{d}v \int_{-a_w^k}^{a_w^k} \mathrm{d}w\ \phi_g^k(u, v, w), \quad u = x, y, z;\ v \neq w \neq u.$$

In order to solve eq. (3) for the nodal fluxes, one needs additional equations relating the surface net currents to the interior fluxes. These relationships can be derived from the three one-dimensional equations obtained by integrating eq. (1) for each node over two transverse directions:

$$-D_g^k \frac{\mathrm{d}^2}{\mathrm{d}u^2}\phi_{gu}^k(u) + \Sigma_g^{r,k}\phi_{gu}^k(u) = Q_{gu}^k(u) - L_{gu}^k(u), \quad u = x, y, z;\ g = 1, 2, \ldots, G;\ k = 1, 2, \ldots, K, \tag{5}$$

where the u-dependent net leakage in the two directions perpendicular to the u-direction is defined as:

$$L_{gu}^k(u) \equiv -\frac{1}{4a_v^k a_w^k}\int_{-a_v^k}^{a_v^k} \mathrm{d}v \int_{-a_w^k}^{a_w^k} \mathrm{d}w\ D_g^k\left(\frac{\partial^2}{\partial v^2} + \frac{\partial^2}{\partial w^2}\right)\phi_g^k(u, v, w).$$

Using eq. (4), the averaged value of the transverse leakage can be written in terms of the face-averaged net currents:

$$\bar{L}_{gu}^k \equiv \frac{1}{2a_u^k}\int_{-a_u^k}^{a_u^k} \mathrm{d}u\ L_{gu}^k(u) = \frac{1}{2a_v^k}\left[J_{gv}^k(a_v^k) - J_{gv}^k(-a_v^k)\right] + \frac{1}{2a_w^k}\left[J_{gw}^k(a_w^k) - J_{gw}^k(-a_w^k)\right]. \tag{6}$$

2.2. The local integral equations

Eq. (5) will be converted to an integral equation using the Green's function for the one-dimensional diffusion removal operator. The Green's function satisfies the equation:

$$-D_g^k \frac{\mathrm{d}^2}{\mathrm{d}u^2} G_{gu}^k(u, u_0) + \Sigma_g^{\mathrm{r},k} G_{gu}^k(u, u_0) = \delta(u - u_0), \quad -a_u^k \leqslant u, u_0 \leqslant a_u^k. \tag{7}$$

On the assumption of satisfying Neumann type boundary conditions at $u = \pm a_u^k$ the nodal Green's function has the form:

$$G_{gu}^k(u, u_0) = \begin{cases} \dfrac{\cosh\left(\omega_g^k\left(a_u^k + u\right)\right)\cosh\left(\omega_g^k\left(a_u^k - u_0\right)\right)}{2D_g^k\omega_g^k \sinh\left(2\omega_g^k a_u^k\right)}, & u < u_0, \\ \dfrac{\cosh\left(\omega_g^k\left(a_u^k - u\right)\right)\cosh\left(\omega_g^k\left(a_u^k + u_0\right)\right)}{2D_g^k\omega_g^k \sinh\left(2\omega_g^k a_u^k\right)}, & u > u_0, \end{cases}$$

with $(\omega_g^k)^2 = \Sigma_g^{\mathrm{r},k}/D_g^k$. Multiplying eq. (5) with $G_{gu}^k(u, u_0)$ and eq. (7) with $\phi_{gu}^k(u)$, subtracting the resulting equations and then integrating for $u \in [-a_u^k, a_u^k]$ one obtains the integral equation:

$$\phi_{gu}^k(u) = \int_{-a_u^k}^{a_u^k} \mathrm{d}u_0\, G_{gu}^k(u, u_0)\left(Q_{gu}^k(u_0) - L_{gu}^k(u_0)\right) + G_{gu}^k\left(u, -a_u^k\right)J_{gu}^k - G_{gu}^k\left(u, a_u^k\right)J_{gu}^{k+1}. \tag{8}$$

Assuming that the index k refers to the u-direction, and that $J_{gu}^k(a_u^k) \equiv J_{gu}^{k+1}(-a_u^{k+1})$ for all k, the following notations were used in eq. (8):

$$J_{gu}^k \equiv J_{gu}^k\left(-a_u^k\right), \quad k = 1, 2, \ldots, K,$$
$$J_{gu}^{K+1} \equiv J_{gu}^K\left(a_u^K\right).$$

We can obtain additional equations for the net currents by imposing the natural condition of partial integrated flux continuity across internal surfaces:

$$\phi_{gu}^{k-1}\left(a_u^{k-1}\right) = \phi_{gu}^k\left(-a_u^k\right), \quad k = 2, 3, \ldots, K$$

or, from eq. (8):

$$A_{gu}^k J_{gu}^{k-1} + B_{gu}^k J_{gu}^k + C_{gu}^k J_{gu}^{k+1} = \tilde{Q}_{gu}^k, \quad u = x, y, z;\ g = 1, 2, \ldots, G;\ k = 2, 3, \ldots, K, \tag{9}$$

where

$$\begin{aligned} A_{gu}^k &= -G_{gu}^{k-1}\left(-a_u^{k-1}, a_u^{k-1}\right), \\ B_{gu}^k &= G_{gu}^{k-1}\left(a_u^{k-1}, a_u^{k-1}\right) + G_{gu}^k\left(a_u^k, a_u^k\right), \\ C_{gu}^k &= -G_{gu}^k\left(-a_u^k, a_u^k\right) = A_{gu}^{k+1}, \\ \tilde{Q}_{gu}^k &= \int_{-a_u^{k-1}}^{a_u^{k-1}} \mathrm{d}u_0\, G_{gu}^{k-1}\left(a_u^{k-1}, u_0\right)\left(Q_{gu}^{k-1}(u_0) - L_{gu}^{k-1}(u_0)\right) \\ &\quad - \int_{-a_u^k}^{a_u^k} \mathrm{d}u_0\, G_{gu}^k\left(-a_u^k, u_0\right)\left(Q_{gu}^k(u_0) - L_{gu}^k(u_0)\right). \end{aligned} \tag{10}$$

Describing the boundary conditions for the u-direction by means of two constants, α_u and β_u, defined by the relations

$$\alpha_u = \frac{J^1_{gu}}{\phi^1_{gu}(-a^1_u)}, \quad \beta_u = \frac{J^{K+1}_{gu}}{\phi^K_{gu}(a^K_u)},$$

one obtains from eq. (8) two more equations which complete the set of linear equations for the net currents (9). The coefficients for these additional equations are:

$$\begin{aligned}
B^1_{gu} &= G^1_{gu}(a^1_u, a^1_u) - 1/\alpha_u, \\
C^1_{gu} &= -G^1_{gu}(-a^1_u, a^1_u) = A^2_{gu}, \\
\tilde{Q}^1_{gu} &= -\int_{-a^1_u}^{a^1_u} \mathrm{d}u_0\, G^1_{gu}(-a^1_u, u_0)\big(Q^1_{gu}(u_0) - L^1_{gu}(u_0)\big), \\
A^{K+1}_{gu} &= -G^K_{gu}(-a^K_u, a^K_u), \\
B^{K+1}_{gu} &= G^K_{gu}(a^K_u, a^K_u) + 1/\beta_u, \\
\tilde{Q}^{K+1}_{gu} &= \int_{-a^K_u}^{a^K_u} \mathrm{d}u_0\, G^K_{gu}(a^K_u, u_0)\big(Q^K_{gu}(u_0) - L^K_{gu}(u_0)\big).
\end{aligned} \tag{11}$$

The values of $-\alpha_u$ and β_u for reflective-, vacuum-, extrapolated vacuum- and $J_{\mathrm{in}} = 0$ boundary type conditions are, respectively, 0, ∞, 0.4695 and 0.5.

2.3. Spatial approximations

Eq. (8) and (9) are reduced to matrix equations using a standard weighted residual procedure to approximate the spatial dependence. The partially integrated group fluxes, group sources and the transverse leakages are approximated as conveniently truncate orthogonal polynomial series:

$$\phi^k_{gu}(u) = \sum_{n=1}^{3} \phi^k_{gun} p^k_{un}(u), \tag{12a}$$

$$Q^k_{gu}(u) = \sum_{n=1}^{3} Q^k_{gun} p^k_{un}(u), \tag{12b}$$

$$L^k_{gu}(u) = \sum_{n=1}^{3} L^k_{gun} p^k_{un}(u), \tag{12c}$$

with the polynomials defined as:

$$p^k_{u1}(u) = 1, \quad p^k_{u2}(u) = u, \quad p^k_{u3}(u) = u^2 - \tfrac{1}{3}(a^k_u)^2.$$

The leading coefficients in the expansions (12) are clearly the corresponding average values over the node. A set of equations for the flux coefficients can be obtained by substituting eqs. (12) into eq. (8), multiplying the resultant equation by one of the expansion polynomials and then integrating for $u \in [-a^k_u, a^k_u]$. The result can be written:

$$N^k_{um}\phi^k_{gum} = \sum_{n=1}^{3} G^k_{gumn}\big(Q^k_{gun} - L^k_{gun}\big) + G^{k-}_{gum}J^k_{gu} - G^{k+}_{gum}J^{k+1}_{gu}, \tag{13}$$

$$u = x, y, z;\ g = 1, 2, \ldots, G;\ k = 1, 2, \ldots, K;\ m = 1, 2, 3,$$

where N_{um}^k is the square normalization constant for the polynomial $p_{um}^k(u)$ and the Green's function matrix elements are given by:

$$
\begin{aligned}
&G_{gumn}^k \equiv \int_{-a_u^k}^{a_u^k} \mathrm{d}u\, p_{um}^k(u) \int_{-a_u^k}^{a_u^k} \mathrm{d}u_0\; p_{un}^k(u_0) G_{gu}^k(u,\, u_0) = \delta_{mn} \tilde{G}_{gum}^k, \\
&\tilde{G}_{gu1}^k = 2a_u^k A, \\
&\tilde{G}_{gu2}^k = \tfrac{2}{3}\left(a_u^k\right)^3 A + \frac{2}{\left(\omega_g^k\right)^2}\left(\omega_g^k a_u^k \cosh\left(\omega_g^k a_u^k\right) - \sinh\left(\omega_g^k a_u^k\right)\right) C_2, \\
&\tilde{G}_{gu3}^k = \tfrac{8}{45}\left(a_u^k\right)^5 A + \frac{4}{\left(\omega_g^k\right)^3}\left(\left(\tfrac{1}{3}\left(\omega_g^k a_u^k\right)^2 + 1\right) \sinh\left(\omega_g^k a_u^k\right) - \omega_g^k a_u^k \cosh\left(\omega_g^k a_u^k\right)\right) C_3,
\end{aligned}
\tag{14}
$$

and, respectively, by:

$$
\begin{aligned}
&G_{gum}^{k\pm} \equiv \int_{-a_u^k}^{a_u^k} \mathrm{d}u\, p_{um}^k(u) G_{gu}^k\left(u,\, \pm a_u^k\right), \\
&G_{gu1}^{k+} = G_{gu1}^{k-} = A, \\
&G_{gu2}^{k+} = -G_{gu2}^{k-} = a_u^k A + \sinh\left(\omega_g^k a_u^k\right) C_2, \\
&G_{gu3}^{k+} = G_{gu3}^{k-} = \left(a_u^k\right)^2 A + B + \cosh\left(\omega_g^k a_u^k\right) C_3,
\end{aligned}
\tag{15}
$$

where

$$
\begin{aligned}
&A = \frac{1}{D_g^k\left(\omega_g^k\right)^2}, \\
&B = \frac{1}{D_g^k\left(\omega_g^k\right)^4}\left(2 - \tfrac{1}{3}\left(\omega_g^k a_u^k\right)^2\right), \\
&C_2 = \frac{1}{D_g^k\left(\omega_g^k\right)^3 \cosh\left(\omega_g^k a_u^k\right)}, \\
&C_3 = -\frac{1}{D_g^k\left(\omega_g^k\right)^4 \sinh\left(\omega_g^k a_u^k\right)}.
\end{aligned}
\tag{16}
$$

The partial integrated total source coefficients and the r.h.s. terms for eqs. (9) can now be written:

$$
Q_{gun}^k = \frac{\chi_g}{k_{\mathrm{eff}}} \sum_{g'=1}^{G} \left(\nu \Sigma_{g'}^{\mathrm{f},k}\right) \phi_{g'un}^k + \sum_{\substack{g'=1 \\ g' \neq g}}^{G} \Sigma_{g'g}^{\mathrm{s},k} \phi_{g'un}^k,
\tag{17}
$$

$$
\begin{aligned}
&\tilde{Q}_{gu}^1 = -\sum_{m=1}^{3} G_{gum}^{1-}\left(Q_{gum}^1 - L_{gum}^1\right), \\
&\tilde{Q}_{gu}^k = \sum_{m=1}^{3} \left[G_{gum}^{(k-1)+}\left(Q_{gum}^{k-1} - L_{gum}^{k-1}\right) - G_{gum}^{k-}\left(Q_{gum}^k - L_{gum}^k\right)\right], \quad k = 2, 3, \ldots, K, \\
&\tilde{Q}_{gu}^{K+1} = \sum_{m=1}^{3} G_{gum}^{K+}\left(Q_{gum}^K - L_{gum}^K\right).
\end{aligned}
\tag{18}
$$

2.4. Transverse leakages

The expansion coefficients for the transverse leakages are computed using the quadratic approximation of Bennewitz et al. [3]. It is supposed that the quadratic polynomial defined by eq. (12c) also extends over the two adjacent nodes in the u-direction. The leading coefficient in this expansion is simply the average leakage $\bar{L}_{gu}^k$ in the center node, while the two remaining coefficients are calculated by requiring that the expansion yields the proper average values, $\bar{L}_{gu}^{k-1}$ and $\bar{L}_{gu}^{k+1}$, over the adjacent nodes, i.e.

$$\frac{1}{2a_u^{k-1}} \int_{-a_u^k - 2a_u^{k-1}}^{-a_u^k} \mathrm{d}u\, L_{gu}^k(u) = \bar{L}_{gu}^{k-1},$$

$$\frac{1}{2a_u^{k+1}} \int_{a_u^k}^{a_u^k + 2a_u^{k+1}} \mathrm{d}u\, L_{gu}^k(u) = \bar{L}_{gu}^{k+1}.$$

Evaluating the integrals involved in these equations, the second and the third expansion coefficients can be expressed as:

$$\begin{aligned} L_{gu2}^k &= \left(\bar{L}_{gu}^{k-1} - \bar{L}_{gu}^k\right)\tilde{C}_{u4}^k - \left(\bar{L}_{gu}^{k+1} - \bar{L}_{gu}^k\right)\tilde{C}_{u2}^k, \\ L_{gu3}^k &= -\left(\bar{L}_{gu}^{k-1} - \bar{L}_{gu}^k\right)\tilde{C}_{u3}^k + \left(\bar{L}_{gu}^{k+1} - \bar{L}_{gu}^k\right)\tilde{C}_{u1}^k, \end{aligned} \tag{19}$$

where

$$\tilde{C}_{ui}^k = C_{ui}^k/\Delta, \quad i = 1, 2, 3, 4, \quad \Delta = C_{u1}^k C_{u4}^k - C_{u2}^k C_{u3}^k$$

and

$$\begin{aligned} C_{u1}^k &= -\left(a_u^k + a_u^{k-1}\right), \\ C_{u2}^k &= \tfrac{2}{3}\left(a_u^k + a_u^{k-1}\right)\left(a_u^k + 2a_u^{k-1}\right), \\ C_{u3}^k &= a_u^k + a_u^{k+1}, \\ C_{u4}^k &= \tfrac{2}{3}\left(a_u^k + a_u^{k+1}\right)\left(a_u^k + 2a_u^{k+1}\right). \end{aligned} \tag{20}$$

The transverse leakage expansion coefficients for boundary nodes are computed such that the average leakage over a fictitious adjacent node situated beyond the boundary is zero for vacuum type boundary conditions and has the value of the average leakage in the boundary node for reflective type boundary conditions.

3. Numerical implementation of GRENADE

3.1. General aspects

The eigenvalue problem is solved using a standard fission source iteration procedure. The one-dimensional net current equations (9) for the x-, y- and z-lines of the three-dimensional mesh are solved using single Gauss–Seidel iterations. Actually, alternating directions iterations are used, i.e. the program performs NITXY alternating solutions of the x- and y-direction equations for each plane of the reactor model, solving the z-direction current equations separately. At each outer (fission source) iteration this entire procedure is repeated NITXYZ times in order to achieve global convergence of the currents to a certain extent. Both NITXY and NITXYZ are specified in the input. It was found that for most practical

problems the program works most satisfactorily with NITXY = NITXYZ = 1. The transverse leakage contribution to the effective group source is updated using the most recently available values for the currents. The leading transverse leakage coefficient is computed from eq. (6), while the other coefficients are calculated using eqs. (19). Once the currents are computed, the expansion coefficients for the partially integrated fluxes are then updated using eq. (13) and are normalized to the nodal fluxes obtained from the nodal balance equation (3). The normalization procedure is intended to counteract the unbalancing that might occur during the alternating directions iterations. The entire procedure is repeated for each energy group.

The effective neutron multiplication factor is computed by means of the Rayleigh quotient [4,5]. This procedure can be summarized as follows:

$$k_{\text{eff}}^{(0)} = 1,$$

$$k_{\text{eff}}^{(n)} = \frac{\int \mathrm{d}V \left(Q^{\mathrm{f}(n)}\right)^2}{\int \mathrm{d}V\, Q^{\mathrm{f}(n-1)} Q^{\mathrm{f}(n)}} k_{\text{eff}}^{(n-1)}, \quad n = 1, 2, 3, \ldots,$$

where $Q^{f(n)}$ is the total fission source at the nth outer iteration.

The outer iterations are accelerated using cyclic Chebyshev extrapolation in conjunction with coarse-mesh rebalancing.

3.2. The Chebyshev extrapolation

The extrapolation scheme used is

$$Q_{\text{ex}}^{\mathrm{f}(n)} = \omega^{(n)} Q^{\mathrm{f}(n)} + (1 - \omega^{(n)}) Q^{\mathrm{f}(n-1)}.$$

For a cycle length of 6, the Chebyshev sequence would give:

$$\omega_i = \frac{1}{1 - \frac{1}{2}\sigma\left[\cos(\pi m_i/12) + 1\right]}$$

with $m_i = 1, 3, 5, 7, 9, 11$. Here σ represents an estimate for the dominance ratio of the iteration process. At each outer iteration this estimate is computed as:

$$\sigma = \frac{\int \mathrm{d}V \left(Q^{\mathrm{f}(n)} - Q^{\mathrm{f}(n-1)}\right)}{\int \mathrm{d}V \left(Q^{\mathrm{f}(n-1)} - Q^{\mathrm{f}(n-2)}\right)}.$$

In the case of the Lebedev extrapolation scheme [6], actually used by the program, the sequence m_i is rearranged in order to avoid numerical instabilities: $m_i = 3, 9, 5, 7, 1, 11$.

The program enters an extrapolation cycle only when certain convergence smoothness conditions are met, i.e. when the relative error in the dominance ratio estimate at two successive outer iterations is less than 0.1. The dominance ratio estimate used within an extrapolation cycle is the one used for the first extrapolation step.

3.3. The coarse mesh rebalancing

The rebalance equations are constructed on a course mesh obtained by partitioning the reactor configuration into an array of cells, such that each coarse mesh cell boundary coincides with a node boundary, and each coarse mesh cell contains at least one "material" node. After each completed

extrapolation cycle, improved estimates for fluxes and currents may be obtained by scaling these with the rebalance factors:

$$\hat{\phi}_g^k = f^k \phi_g^k, \quad \hat{J}_{gu}^k = \tfrac{1}{2}\left(f^k + f^{k-1}\right) J_{gu}^k.$$

For the nodes contained within a coarse mesh cell, the rebalance factor is the same.

An equation for the rebalance factors is obtained by requiring that the rebalanced solution satisfies the coarse mesh balance equation obtained by summing the nodal balance equation (3) over all energy groups and over all nodes contained within a coarse mesh cell. Assuming that the space collapsing is carried out only in the x–y plane obtained by collapsing all x–y planes of the reactor model, the equation for the rebalance factors takes the form:

$$a_{mn} f^{m,n-1} + b_{mn}^a f^{m-1,n} + b_{mn}^b f^{m,n} + b_{mn}^c f^{m+1,n} + c_{mn} f^{m,n+1} = k'^{-1}_{\text{eff}} t_{mn} f^{m,n},$$

where the indices m and n range over the number of x- and y-coarse mesh intervals, respectively. Making the notations:

$$A_{lm}^n = a_{mn}\delta_{lm},$$

$$B_{lm}^n = \begin{cases} b_{mn}^a, \text{ for } l = m-1, \\ b_{mn}^b, \text{ for } l = m, \\ b_{mn}^c, \text{ for } l = m+1, \\ 0, \text{ for } l \neq m,\ m \pm 1, \end{cases}$$

$$C_{lm}^n = c_{mn}\delta_{lm},$$

$$T_{lm}^n = t_{mn}\delta_{lm},$$

we obtain the following block tridiagonal matrix system:

$$A^n f^{n-1} + B^n f^n + C^n f^{n+1} = k'^{-1}_{\text{eff}} T^n f^n,$$

where f^n defines the rebalance factor vector for the nth coarse mesh y-line. This eigenvalue problem for k'_{eff} is solved using an iteration procedure similar to the fission source iteration. At each iteration the inversion of the LHS operator is carried out using a forward elimination–backward substitution technique.

4. Program structure and operation

4.1. General remarks

The program GRENADE is self-contained (it does not call any external user library subroutines) and it consists of subprograms written exclusively in FORTRAN IV. The only machine dependent coding in this program concerns the mass storage I/O subroutines (OPENMS, WRITMS, READMS, CLOSMS), the operating system interface subroutines returning current date (DATE) and CPU time (SECOND), and the dynamic memory allocation system subprograms (CMMALF, LOCF) of the CDC CYBER 170/720 computer.

The bulk of data kept in the high speed memory is organized in arrays corresponding to a certain direction, to a certain energy group and to a certain x–y plane. Data for all directions, energy groups and x–y planes are stored on mass storage random files.

The operation of the program assumes the Cartesian coordinate system to be oriented such that the

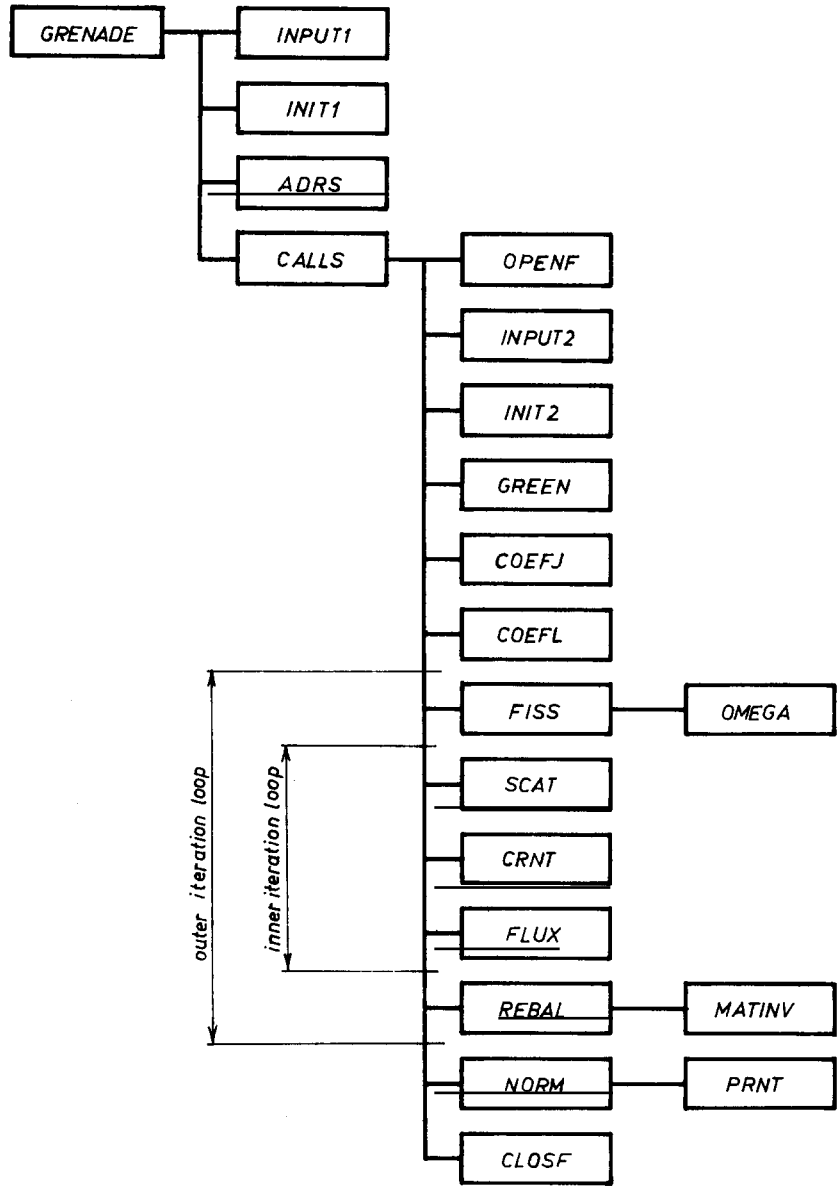

Fig. 1. GRENADE simplified flowchart.

x-axis is pointing to the bottom of the page, the y-axis is pointing to the right edge of the page and the z-axis is pointing to the onlooker.

4.2. Program description

A simplified flowchart of the GRENADE program is shown in fig. 1.

The main program GRENADE calls only four of the program subroutines: INPUT1, INIT1, ADRS and CALLS (which takes over the program flow control). It contains the array declaring statement

```
DIMENSION A(1)
```

for the variable A, which is subject to dynamic memory allocation. The dynamic memory allocation sequence is:

```
CALL SYMLIB
CALL CMMALF(NW + 1, 0, 0, IA)
LA = IA - LOCF(A(1)) + 1
```

with NW (the number of words required by the arrays used within the program) being previously calculated in subroutine INIT1. For machines having no dynamic memory allocation facilities, array A should be dimensioned according to NW and the above sequence should be replaced by:

```
LA = 1
```

It is apparent that LA corresponds to the first word location of the array A. The main program also provides start-of-execution and end-of-execution CPU time and writes the total execution time to the output file.

Subroutine INPUT1 writes GRENADE banner and current date to output file. It reads and writes the job title. It also reads necessary data for dynamic memory allocation: the number of spatial dimensions of the problem (IDM), the number of x-, y- and z-mesh intervals (IM, JM and KM), the number of x- and y-coarse mesh intervals (ICM, JCM), the number of materials (NMAT) and the number of energy groups (IGM). For one-dimensional cases it redefines JM = KM = 1 and for two-dimensional cases it sets KM = 1.

Subroutine INIT1 computes from the data read by INPUT1 the number of words of core required by the arrays used by the program, according to the relation:

$$\mathrm{NW} = 19 \times \mathrm{IM} \times \mathrm{JM} + 10 \times (\mathrm{IM} + \mathrm{JM}) + (9 + 10 \times \mathrm{IGM} \times \mathrm{IDM}) \times \mathrm{KM} + (4 + \mathrm{IGM}) \times \mathrm{IGM} \times \mathrm{NMAT} + (1 + 10 \times \mathrm{IDM}) \times \mathrm{IGM} + (\mathrm{ICM} + \mathrm{JCM}) \times (\mathrm{ICM} + 1) + 27.$$

It also computes array lengths, which are stored in COMMON block NDIM.

Subroutine ADRS calculates first word addresses for the program arrays, relative to the first word address of the array A (subject to dynamic memory allocation).

The main task subroutine CALLS fulfils is to control the overall flow of the program, calling the bulk of the program subroutines and equivalencing the first word addresses of the arrays with the corresponding components of array A.

Subroutine OPENF opens the mass storage random files. It also generates record indices and stores them in the record key array IND (this is a three-dimensional, direction-, plane- and group-dependent array).

Subroutine INPUT2 reads the bulk of input data and writes them to the output file:

NIT	maximum number of outer iterations,
NITXY	number of x–y alternating directions iterations,
NITXYZ	number of x–y–z alternating directions iterations,
NITR	maximum number of iterations for the rebalancing eigenvalue problem,
IOPBN	variable indicating optional use of straight/polygonal boundaries,
IOPEX	variable indicating optional fission source extrapolation,
IOPRB	variable indicating optional coarse mesh rebalancing,
IOPRS	variable indicating whether the calculation is a restarted one,
IOPWR	variable indicating whether a short or a long output report is desired (see section 6),
EPSF	flux convergence criterion,
EPSK	k_{eff} convergence criterion,

IALF(ID)	left side boundary condition type for direction ID,
IBET(ID)	right side boundary condition type for direction ID,
X(I)	x-direction mesh spacing,
Y(J)	y-direction mesh spacing,
Z(K)	z-direction mesh spacing,
MAPZ(I, J)	material map for a certain x–y plane,
MI(1, J), MI(2, J)	minimum and maximum I, respectively, for the "material" nodes on the Jth x-line,
MJ(1, I), MJ(2, I)	minimum and maximum J, respectively, for the "material" nodes on the Ith y-line,
IIC(IC)	ordering number of the x-mesh interval at which the ICth x-coarse mesh interval begins (the program sets IIC(1) = 1),
JJC(JC)	ordering number of the y-mesh interval at which the JCth y-coarse mesh interval begins (the programs sets JJC(1) = 1),
ASIGM(M, IG)	absorption cross section for material M and group IG,
FSIGN(M, IG)	production cross section ($\nu\Sigma_g^{f,k}$),
SSIGM(M, IG, IG1)	scattering cross section from group IG to group IG1,
HI(IG)	fission spectrum.

As it is obvious from the above explanations, arrays MI and MJ describe the boundaries of the x–y planes (all identically shaped). Such a detailed description is necessary for cases for which the reactor model is not simply a rectangular box, but has "polygonal" boundaries involving the x- and y-directions. Arrays IIC and JJC describe the rectangular x–y rebalancing map. The coarse mesh spacings should be chosen such that each coarse mesh cell contains at least one "material" node (special care should be taken for the case of "polygonal" boundary problems). Failure to meet this requirement leads to fatal program errors. Variables IOPBN, IOPEX, IOPRB, IOPRS and IOPWR are stored in COMMON block OPTN.

Subroutine INIT2 initializes boundary condition constants α_u and β_u according to the boundary condition type specified by IALF(ID) and IBET(ID). The resulting variables, ALFA(ID) and BETA(ID), are stored together with IALF(ID) and IBET(ID) in COMMON block BNDR. INIT2 also constructs removal cross sections, RSIGM(M, IG), from absorption and scattering cross sections. If the problem has straight boundaries (IOPBN = 0), INIT2 sets

$$\mathrm{MI}(1, \mathrm{J}) = 1, \quad \mathrm{J} = 1, \mathrm{JM},$$
$$\mathrm{MI}(2, \mathrm{J}) = \mathrm{IM},$$
$$\mathrm{MJ}(1, \mathrm{I}) = 1, \quad \mathrm{I} = 1, \mathrm{IM},$$
$$\mathrm{MJ}(2, \mathrm{I}) = \mathrm{JM}.$$

If the calculation is not a restarted one (IOPRS = 0), INIT2 initializes flux, fission source and currents. The average nodal fluxes are initialized to 1, while the other components together with the fission source components and the currents are zeroed. INIT2 also constructs normalization factors for the nodal expansion polynomials.

Subroutine GREEN computes Green's function matrix elements GG(N, I, J), N = 1, 2, 3, according to eqs. (14) and GM(N, I, J), N = 1, 2, 3, given by eqs. (15).

Subroutine COEFJ constructs current equation coefficients A_{gu}^k and B_{gu}^k given by eqs. (10) and (11). For the x-direction these coefficients are stored as ABX(1, I, J) and ABX(2, I, J), respectively. In a similar manner array ABY is used for the y direction and array ABZ for the z direction.

Subroutine COEFL generates transverse leakage coefficients $\tilde{C}_{ui}^k$ from eqs. (20) and stores them in arrays (CLK(4, I, J) for the x and y directions and CLZ(4, K) for the z direction, respectively.

Subroutine FISS computes fission source, extrapolates it if desired, calculates the effective neutron multiplication factor k_{eff} (KEFF in the program) and tests if k_{eff} and the flux have converged to the

desired precision. The Chebyshev extrapolation factor is provided by function OMEGA. The COMMON blocks ITER and FSIT contain fission source iteration related data: IT (outer iteration loop counter), NBIT (control variable for the extrapolation and rebalancing steps), NIT, NITXY, NITXYZ, NITR, TIME1 (start-of-execution CPU time), EPSF, EPSK, ERRF (relative flux error), ERRK (relative k_{eff} error), KEFF and REBC (rebalancing constant). Definitions for ERRF, ERRK and REBC are given in section 6.

Subroutine SCAT generates the scattering source and the total group source QFSL.

Subroutine CRNT computes the net interface currents (JIX, JIY and JIZ) for the three perpendicular directions, by solving the corresponding one-dimensional current equations (9). This subroutine also controls the x–y and x–y–z alternating directions iterations. The transverse leakage contribution to the effective group source is updated using the most recently available values for the currents. The effective group source, QFSL, is stored for further use in subroutine FLUX.

Subroutine FLUX constructs flux coefficients (FI(N, I, J), N = 1, 2, 3) according to eq. (13). It also scales the flux coefficients to the average flux computed from the nodal balance equation (3).

Subroutine REBAL generates coarse mesh rebalance factors and scales the flux coefficients and currents. For the accomplishment of the matrix inversion, subroutine REBAL calls subroutine MATINV.

Subroutine NORM normalizes the flux such that the core-averaged power equals 1. For this normalization scheme, the average assembly power coincides with the average assembly fission source. NORM also normalizes the currents accordingly. For IOPWR = 1, this subroutine writes by means of subroutine PRNT the average flux and power distributions to the output file.

4.3. File names used by GRENADE

Besides INPUT (TAPE1) and OUTPUT (TAPE2), the program GRENADE makes use of 11 local files. We give a brief description of the contents of these files.

TAPE3 flux coefficients,
TAPE4 net interface currents,
TAPE5 fission source coefficients,
TAPE6 fission + scattering source coefficients,
TAPE7 $\tilde{G}^{k}_{gun}$ Green's function matrix elements,
TAPE8 G^{k+}_{gun} Green's function matrix elements,
TAPE9 current equation coefficients,
TAPE10 material map,
TAPE11 effective group source coefficients,
TAPE12 leakage coefficients $\tilde{C}^{k}_{ui}$,
TAPE13 factorization matrices in the block matrix inversion procedure.

4.4. Restarted calculations

Any GRENADE calculation can be continued, i.e. restarted from the stage at which it ended, provided the necessary files are stored. Such "continuation" runs (selected by setting the variable IOPRS = 1) may, for example, be necessary for improving the convergence on flux distribution. The files to be saved for such cases are TAPE3, TAPE4 and TAPE5, namely those on which, respectively, flux, currents and fission source reside, quantities which are otherwise being initialized in subroutine INIT2.

Also flux-, currents and fission source distributions can be used as a first guess in a calculation with a changed material configuration, as long as the mesh configuration remains identical.

5. Input preparation

The card input consists of 18 card types, which are sequentially described in table 1.

Table 1
GRENADE card input ([x] is here defined as identical to x if x is an integer, otherwise the value of x rounded to the next higher integer)

Card type	Format	Columns	Variable	Comments
1	8A9	1–72	TITLE(I)	(TITLE(I), I = 1, 8) title of calculation
2	24I3	1–3	IDM	(1/2/3)
		4–6	IM	
		7–9	JM	if IDM.GE.2
		10–12	KM	if IDM.EQ.3
		13–15	ICM	
		16–18	JCM	
3	24I3	1–3	NMAT	
		4–6	IGM	
4	24I3	1–3	NIT	
		4–6	NITXY	typical value = 1
		7–9	NITXYZ	typical value = 1
		10–12	NITR	typical value = 20
5	24I3	1–3	IOPBN	the value for the following items is
		4–6	IOPEX	1 if the corresponding option is
		7–9	IOPRB	desired, otherwise 0
		10–12	IOPRS	
		13–15	IOPWR	
6	6E12.5	1–12	EPSF	typical value = 10^{-4}
		13–24	EPSK	typical value = 10^{-6}
7	24I3	1–3	IALF(1)	0 – reflective, 1 – vacuum,
		4–6	IBET(1)	2 – extrapolated vacuum, 3 – $J_{in} = 0$
		7–9	IALF(2)	if IDM.GE.2
		10–12	IBET(2)	if IDM.GE.2
		13–15	IALF(3)	if IDM.EQ.3
		16–18	IBET(3)	if IDM.EQ.3
8	12F6.2	1–72	X(I)	(X(I), I = 1, IM); [IM/12] cards
9	12F6.2	1–72	Y(J)	if IDM.GE.2 (Y(J), J = 1, JM); [JM/12] cards
10	12F6.2	1–72	Z(K)	if IDM.EQ.3 (Z(K), K = 1, KM); [KM/12] cards
11	36I2	1–72	MAPZ(I, 1)	if IDM.EQ.1 (MAPZ(I, 1), I = 1, IM;); [IM/36] cards
12	36I2	1–72	MAPZ(I, J)	if IDM.GT.1 (MAPZ(I, J), J = 1, JM); [JM/36] cards; card type should be repeated for I = 1, IM
13	24I3	1–72	MI(N, J)	if IOPBN.EQ.1 (MI(N, J), J = 1, JM); [JM/24] cards; card type should be repeated for N = 1, 2

Table 1 (continued)

Card type	Format	Columns	Variable	Comments
14	24I3	1–72	MJ(N, I)	if IOPBN.EQ.1 (MJ(N, I), I = 1, IM); [IM/24] cards; card type should be repeated for N = 1, 2
15	24I3	1–72	IIC(IC)	if IOPRB.EQ.1 (IIC(IC), IC = 1, ICM); [ICM/24] cards
16	24I3	1–72	JJC(JC)	if IOPRB.EQ.1 (JJC(JC), JC = 1, JCM); [JCM/24] cards
17	6E12.5	1–12	ASIGM(M, IG)	
		13–24	FSIGN(M, IG)	
		25–36	DIF(M, IG)	
		36–72	SSIGM(M, IG, IG1)	(SSIGM(M, IG, IG1), IG1 = 1, IGM) if IGM.GT.3, scattering cross sections are read from [(IGM-3)/6] more cards with format 6E12.5; this card type should be repeated for IG = 1, IGM, and for each IG, for M = 1, NMAT
18	6E12.5	1–72	HI(IG)	(HI(IG), IG = 1, IGM); [IGM/6] cards

6. Output interpretation

The output generated by a GRENADE execution is organized into two main sections: INPUT DATA and OUTPUT DATA.

The input data section begins right after the GRENADE banner and the current date. The title card input is listed. Further GRENADE lists in a structured manner the quantities IDM, IM, JM, KM, ICM, JCM, IGM, NMAT, NIT, NITXY, NITXYZ, IOPEX, IOPRB, EPSF and EPSK. Next, the program prints the number of words of core required by the arrays used by the program. Further, the code lists the boundary condition types, the mesh spacings for the three perpendicular directions, the material map (in increasing order of the plane number), the boundary limits for the x- and y-directions (for polygonal boundary cases) and the rebalancing map for the x- and y-directions. The input data section ends with the material properties, listed in a tabular manner, and the fission spectrum. If the removal cross section turns out to be zero, it is set to 10^{-20} and a warning message is issued.

The output data section comprises two sub-sections: the iteration-governed output and the final flux- and power distribution output. The latter actually appears on the output report only if IOPWR = 1.

The iteration-governed output is most important since it shows how the flux iterations proceed. At each outer iteration, a line of output is generated by subroutine FISS. It gives the present values of IT (the current iteration number), TIME (CPU time since the start-of-execution), KEFF, ERRK (the eigenvalue convergence) defined as

$$\mathrm{ERRK} = |1 - k_{\mathrm{eff}}^{(n)}/k_{\mathrm{eff}}^{(n-1)}|,$$

ERRF (the flux convergence), given by

$$\mathrm{ERRF} = \max_i |1 - \phi_i^{(n)}/\phi_i^{(n-1)}|,$$

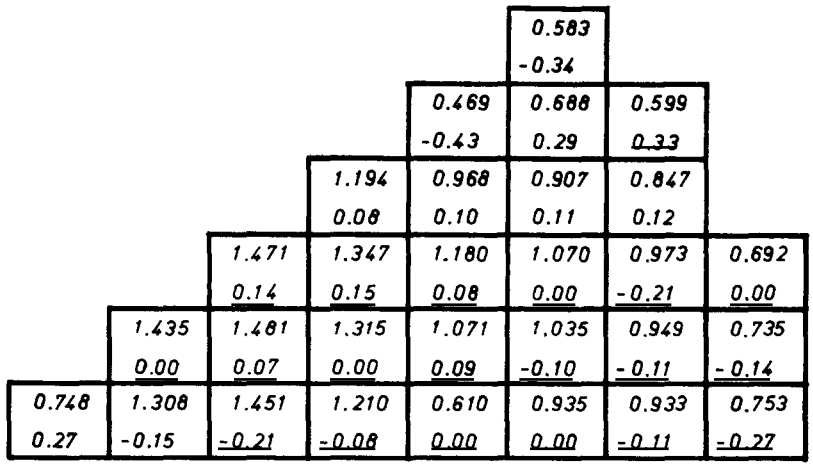

Fig. 2. Assembly-averaged power distribution and relative errors (in %) for the two-dimensional IAEA benchmark problem, with one node per assembly.

where ϕ_i are the fluxes, REBC (rebalancing constant) defined as

$$\text{REBC} = \frac{1}{N} \sum_{i=1}^{N} (1 - f_i)^2,$$

where f_i are the rebalancing factors, and OMG (the Chebyshev extrapolation factor).

7. Test runs

The test run deck comprises the input for the two-group IAEA benchmark problems, which are simplified two- or three-dimensional models of a pressurized water reactor [7].

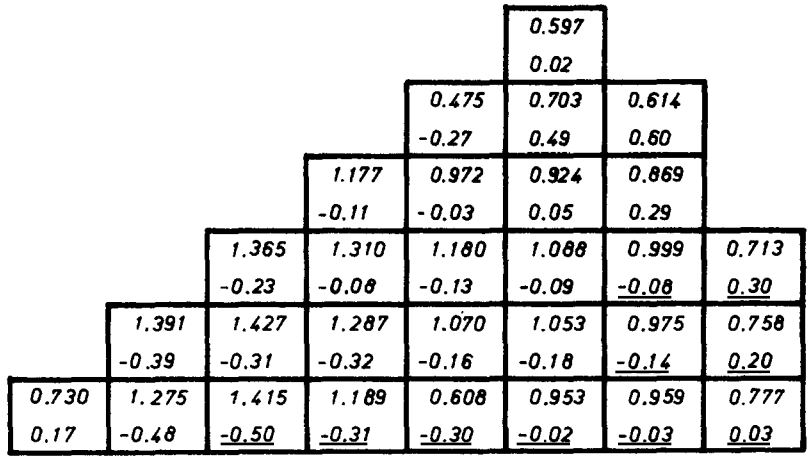

Fig. 3. Axial-averaged assembly-averaged power distribution and relative errors (in %) for the three-dimensional IAEA benchmark problem, with a 20 cm mesh spacing.

The GRENADE assembly-averaged power density output for the two-dimensional IAEA problem, for a 20 cm mesh spacing (one node per assembly), together with the corresponding relative errors (in %) to the IAEA reference values, are presented in fig. 2. For this problem the GRENADE k_{eff} is 1.029560, while the reference value is 1.029585. Taking into account the computer features (especially those concerning the number of operations per second) it is to be noted that the GRENADE execution time (9.3 s on the CYBER 170/720) is smaller than those reported in the literature [2] and the maximum error in the assembly averaged power distribution is smaller as well (except the Partial Current Balance Method which is slower, however). Fig. 3 shows the axial-averaged assembly-averaged power density distribution for the three-dimensional IAEA problem together with the corresponding errors to the IAEA reference values. For this problem the accuracy of GRENADE again compares favourably with that of other solutions quoted in ref. [2].

Acknowledgements

The authors acknowledge the continous support and the helpful comments of C. Costescu throughout the elaboration period of this program.

This work was supported by the Institute for Nuclear Power Reactors, Pitesti, Romania.

References

[1] J.J. Dorning, Proc. Conf. Computational Methods of Nuclear Engineering, Williamsburg, Virginia, 23–25 April 1979, CONF-790402, American Nuclear Society (1979) p. 3.
[2] R.D. Lawrence and J.J. Dorning, Nucl. Sci. Eng. 76 (1980) 218.
[3] F. Bennewitz, H. Finneman and M. Wagner, Trans. Am. Nucl. Sci. 22 (1975) 250.
[4] E.L.Wachspress, Iterative Solution of Elliptic Systems (Prentice Hall, Englewood Cliffs, New Jersey, 1966).
[5] R.S. Varga, Matrix Iterative Analysis (Prentice Hall, Englewood Cliffs, New Jersey, 1961).
[6] V.I. Lebedev and S.A. Finogenov, J. Comput. Math. Math. Phys. 16 (1976) 895 (in Russian).
[7] Argonne Code Center, Benchmark Problem Book (ANL-7416, Argonne National Laboratory, 1977).

TEST RUN OUTPUT

```
*************************************************************
*                                                           *
*   GGGG    RRRRR    EEEEEE  NN  NN     AA     DDDDD    EEEEEE  *
*  GGGGGG   RRRRRR   EEEEEE  NNN NN    AAAA    DDDDDD   EEEEEE  *
*  GG       RR  RR   EE      NNNNNN   AA  AA   DD  DD   EE      *
*  GG GGG   RR  RR   EEEEE   NN NNN   AA  AA   DD  DD   EEEEE   *
*  GG  GG   RRRRR    EE      NN  NN   AAAAAA   DD  DD   EE      *
*  GGGGGG   RR  RR   EEEEEE  NN  NN   AA  AA   DDDDDD   EEEEEE  *
*   GGGG    RR  RR   EEEEEE  NN  NN   AA  AA   DDDDD    EEEEEE  *
*                                                           *
*************************************************************

   GREEN'S FUNCTION NODAL ALGORYTHM FOR DIFFUSION EQUATION
                          85/05/13.

**********************
* I N P U T   D A T A *
**********************

2D BENCHMARK PROBLEM: "BENCHMARK PROBLEM BOOK",ANL-7416(1977)

IDM     1/2/3=NO.OF SPATIAL DIMENSIONS                          2
IM      NO. OF X-MESH INTERVALS                                 9
JM      NO. OF Y-MESH INTERVALS                                 9
KM      NO. OF Z-MESH INTERVALS                                 1
ICM     NO. OF X-COARSE-MESH INTERVALS                          3
JCM     NO. OF Y-COARSE-MESH INTERVALS                          3
IGM     NO. OF ENERGY GROUPS                                    2
NMAT    NO. OF MATERIALS                                        4
NIT     MAXIMUM NO. OF OUTER ITERATIONS                        50
NITXY   NO.OF X-Y ALT. DIR. ITERATIONS                          1
NITXYZ  NO.OF X-Y-Z ALT. DIR. ITERATIONS                        1
IOPEX   0/1=NO/FISS. SOURCE EXTRAPOLATION                       1
IOPRB   0/1=NO/COARSE-MESH REBALANCING                          1
EPSF    FLUX ERROR CRITERION                        1.00000E-04
EPSK    KEFF ERROR CRITERION                        1.00000E-06

WORDS OF CORE REQUIRED                                       1909

BOUNDARY CONDITIONS
(0/1/2/3=REFLECTIVE/VACUUM/EXTRAP. VACUUM/JIN=0)
X DIRECTION      2  0
Y DIRECTION      0  2

MESH SPACING
X
20.00 20.00 20.00 20.00 20.00 20.00 20.00 20.00 10.00
Y
10.00 20.00 20.00 20.00 20.00 20.00 20.00 20.00 20.00

MATERIAL MAP
PLANE   1
  4  4  4  4  0  0  0  0  0
  1  1  1  4  4  4  0  0  0
  2  2  1  1  1  4  4  0  0
  2  2  2  2  2  1  1  4  4  0
  3  3  2  2  2  3  1  1  4  0
  3  3  3  2  2  2  1  4  4
  2  2  2  2  2  2  2  1  1  4
  2  2  2  2  2  2  2  1  4
  3  2  2  2  3  2  2  1  4

BOUNDARY LIMITS
  1   1   1   1   2   2   3   4   6
  9   9   9   9   9   9   9   9   9
  1   1   1   1   1   1   1   1   1
  4   6   7   8   8   9   9   9   9
```

REBALANCING MAP

```
  1   4   7
  1   4   7
```

GROUP	MATERIAL	ASIGM	FSIGN	DIF		SSIGM
1	1	1.012000E-02	0.	1.500000E+00	0.	2.000000E-02
1	2	1.012000E-02	0.	1.500000E+00	0.	2.000000E-02
1	3	1.012000E-02	0.	1.500000E+00	0.	2.000000E-02
1	4	1.600000E-04	0.	2.000000E+00	0.	4.000000E-02
2	1	8.003200E-02	1.350000E-01	4.000000E-01	0.	0.
2	2	8.503200E-02	1.350000E-01	4.000000E-01	0.	0.
2	3	1.300320E-01	1.350000E-01	4.000000E-01	0.	0.
2	4	1.002400E-02	0.	3.000000E-01	0.	0.

FISSION SPECTRUM

```
  1.0000   0.0000
```

```
*************************
* O U T P U T   D A T A *
*************************
```

OUTER ITER.	TIME (SECONDS)	EIGENVALUE (KEFF)	EIGENVALUE CONVERGENCE	FLUX CONVERGENCE	REBALANCE CONSTANT	EXTRAPOLATION FACTOR
1	.81	1.000000000	1.00000E+00	1.00000E+00	0.	1.00000E+00
2	.99	1.027478131	2.75954E-02	9.13141E+00	0.	1.00000E+00
3	1.17	1.024563629	5.43504E-03	2.24667E-01	0.	1.00000E+00
4	1.35	1.025544830	1.04078E-03	9.25331E-02	0.	1.00000E+00
5	1.53	1.026374610	7.58331E-04	5.81800E-02	0.	1.00000E+00
6	1.71	1.026944473	7.46477E-04	4.28477E-02	0.	1.00000E+00
7	1.90	1.027346630	7.34120E-04	3.53773E-02	0.	1.00000E+00
8	2.08	1.032155076	4.44832E-03	3.25253E-02	0.	6.22756E+00
9	2.27	1.028069688	4.21906E-03	2.34598E-01	0.	1.16825E+00
10	2.46	1.030217234	1.16195E-03	5.28909E-02	0.	2.62460E+00
11	2.65	1.028114046	2.84036E-03	3.25778E-02	0.	1.57345E+00
12	2.84	1.067858758	1.10163E-02	2.31358E-02	0.	3.00216E+01
13	3.03	1.028612637	3.70023E-02	2.56400E-01	0.	1.01704E+00
14	3.35	1.029575571	1.16221E-03	8.24172E-02	1.25427E-02	1.00000E+00
15	3.53	1.029515283	3.93684E-05	8.30418E-02	0.	1.00000E+00
16	3.72	1.029553085	8.67232E-05	1.38179E-02	0.	1.02977E+00
17	3.91	1.029552537	4.96194E-05	4.78861E-03	0.	1.00499E+00
18	4.10	1.029557290	5.80485E-05	2.45118E-03	0.	1.02178E+00
19	4.29	1.029559345	5.72760E-05	1.88219E-03	0.	1.01271E+00
20	4.48	1.029560781	5.92261E-05	1.47825E-03	0.	1.03444E+00
21	4.66	1.029561789	5.76396E-05	1.23121E-03	0.	1.00058E+00
22	4.93	1.029245913	5.62506E-04	9.96101E-04	2.25423E-04	5.90630E+00
23	5.13	1.029638095	4.19839E-04	1.24398E-02	0.	1.16621E+00
24	5.32	1.029449619	8.38350E-05	2.06407E-03	0.	2.58098E+00
25	5.51	1.029634801	2.38866E-04	2.14810E-03	0.	1.56412E+00
26	5.69	1.028086555	6.89480E-04	7.70312E-04	0.	2.30587E+01
27	5.88	1.029601554	1.47170E-03	1.11521E-02	0.	1.01686E+00
28	6.14	1.029514736	9.44497E-05	2.85676E-03	1.41792E-04	1.00000E+00
29	6.32	1.029557537	3.87412E-05	9.27057E-04	0.	1.00000E+00
30	6.50	1.029558059	3.01214E-06	4.56461E-04	0.	1.00000E+00
31	6.68	1.029558248	3.30556E-06	3.66806E-04	0.	1.00000E+00
32	6.87	1.029558370	3.16507E-06	3.13668E-04	0.	1.00000E+00
33	7.05	1.029558489	2.88422E-06	2.62798E-04	0.	1.00000E+00
34	7.24	1.029559120	1.14628E-05	2.21987E-04	0.	4.50072E+00
35	7.43	1.029558895	1.63388E-06	8.67394E-04	0.	1.15400E+00
36	7.62	1.029559967	4.46397E-07	2.18180E-04	0.	2.34499E+00
37	7.81	1.029559377	6.31458E-07	1.56631E-04	0.	1.50990E+00
38	8.00	1.029563155	6.13653E-06	9.66312E-05	0.	9.59140E+00
39	8.18	1.029560077	1.82418E-06	3.39295E-04	0.	1.01577E+00
40	8.44	1.029559578	2.72755E-06	7.60460E-05	4.56104E-06	1.00000E+00
41	8.62	1.029560350	1.83898E-06	2.06296E-05	0.	1.00000E+00
42	8.80	1.029560285	9.95309E-07	1.52019E-05	0.	1.00000E+00

*** FLUX FOR GROUP 1 ***

PLANE 1

	1	2	3	4	5	6	7	8	9
1	3.36303E+00	3.21126E+00	2.42035E+00	7.47359E-01	0.	0.	0.	0.	0.
2	2.02779E+01	1.98308E+01	1.66251E+01	5.91933E+00	2.44493E+00	6.15050E-01	0.	0.	0.
3	2.93704E+01	2.98078E+01	2.90407E+01	2.25875E+01	1.44448E+01	3.95121E+00	7.00412E-01	0.	0.
4	2.99464E+01	3.26872E+01	3.36767E+01	2.85117E+01	2.08482E+01	1.39378E+01	3.95148E+00	6.15123E-01	0.
5	2.65630E+01	3.42495E+01	3.72394E+01	3.09334E+01	2.08454E+01	2.08499E+01	1.44467E+01	2.44531E+00	0.
6	3.86821E+01	4.15449E+01	4.25128E+01	3.77183E+01	3.09355E+01	2.85156E+01	2.25915E+01	5.92052E+00	7.47528E-01
7	4.58129E+01	4.67231E+01	4.64066E+01	4.25145E+01	3.72430E+01	3.36822E+01	2.90467E+01	1.66289E+01	2.42091E+00
8	4.18675E+01	4.53166E+01	4.67241E+01	4.15473E+01	3.42534E+01	3.26931E+01	2.98143E+01	1.98355E+01	3.21204E+00
9	3.25620E+01	4.18679E+01	4.58143E+01	3.86844E+01	2.65662E+01	2.99521E+01	2.93770E+01	2.02828E+01	3.36385E+00

*** FLUX FOR GROUP 2 ***

PLANE 1

	1	2	3	4	5	6	7	8	9
1	7.96932E+00	7.57315E+00	5.85856E+00	2.85055E+00	0.	0.	0.	0.	0.
2	5.57697E+00	5.44446E+00	5.12474E+00	1.24733E+01	6.06835E+00	2.25874E+00	0.	0.	0.
3	6.90733E+00	7.02580E+00	7.20847E+00	6.26880E+00	4.43596E+00	8.33130E+00	2.59331E+00	0.	0.
4	6.92695E+00	7.66487E+00	7.92287E+00	6.71509E+00	5.09788E+00	4.31446E+00	8.33186E+00	2.25900E+00	0.
5	4.52144E+00	7.93044E+00	8.73758E+00	7.17174E+00	3.47432E+00	5.09831E+00	4.43654E+00	6.06930E+00	0.
6	8.96351E+00	9.74240E+00	9.98072E+00	8.84557E+00	7.17222E+00	6.71600E+00	6.26991E+00	1.24758E+01	2.85118E+00
7	1.07489E+01	1.09693E+01	1.08935E+01	9.98110E+00	8.73844E+00	7.92415E+00	7.20994E+00	5.12590E+00	5.85992E+00
8	9.69243E+00	1.06263E+01	1.09695E+01	9.74295E+00	7.93134E+00	7.66625E+00	7.02733E+00	5.44576E+00	7.57499E+00
9	5.54595E+00	9.69252E+00	1.07492E+01	8.96405E+00	4.52198E+00	6.92825E+00	6.90888E+00	5.57831E+00	7.97127E+00

*** POWER DENSITY ***

PLANE 1

	1	2	3	4	5	6	7	8	9
1	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	7.52890E-01	7.35003E-01	6.91840E-01	0.	0.	0.	0.	0.	0.
3	9.32489E-01	9.48482E-01	9.73143E-01	8.46288E-01	5.98855E-01	0.	0.	0.	0.
4	9.35138E-01	1.03476E+00	1.06959E+00	9.06538E-01	6.88214E-01	5.82452E-01	0.	0.	0.
5	6.10395E-01	1.07061E+00	1.17957E+00	9.68185E-01	4.69034E-01	6.88271E-01	5.98933E-01	0.	0.
6	1.21007E+00	1.31522E+00	1.34740E+00	1.19415E+00	9.68250E-01	9.06660E-01	8.46438E-01	0.	0.
7	1.45110E+00	1.48085E+00	1.47063E+00	1.34745E+00	1.17969E+00	1.06976E+00	9.73342E-01	6.91997E-01	0.
8	1.30848E+00	1.43454E+00	1.48089E+00	1.31530E+00	1.07073E+00	1.03494E+00	9.48690E-01	7.35178E-01	0.
9	7.48704E-01	1.30849E+00	1.45114E+00	1.21015E+00	6.10468E-01	9.35314E-01	9.32699E-01	7.53073E-01	0.

*** THIS CASE WAS PROCESSED BY GRENADE IN 9.18 SECONDS ***