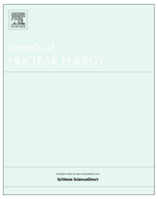
Annals of Nuclear Energy 65 (2014) 370–375

Contents lists available at ScienceDirect

Annals of Nuclear Energy

journal homepage: www.elsevier.com/locate/anucene



Technical note

Solving the static-neutron diffusion equation in 2D-Cartesian geometry with Lagrange interpolation

Barbaro Quintero-Leyva [*]

119 SW 6th Avenue, Miami, FL 33130, United States

ARTICLE INFO

Article history:
Received 10 July 2013
Received in revised form 6 November 2013
Accepted 7 November 2013
Available online 11 December 2013

Keywords:
Multivariate interpolation
Neutron diffusion
Lagrange interpolation
Progressive polynomial approximation
Nuclear reactor

ABSTRACT

Cell centered (LAPc) and cell edge (LAPe) algorithms were developed to solve the static neutron diffusion equation in 2D Cartesian geometry using Lagrange interpolation with the progressive polynomial approximation. Two benchmark problems were used to test the algorithms: the two-group TWIGL problem and a one-group IAEA benchmark problem. The LAP algorithms showed to be more accurate than a finite difference method (FDM) and for about the same level of accuracy, the LAP numerical methods have an efficiency advantage because they have to solve for less number of unknowns. The LAP algorithms showed more sensitivity to the mesh size than what QUANDRY results showed. Even though the FDMs algorithm, for the calculation of k_{eff}, showed systematically to be less accurate than QUANDRY, LAPc, and that LAPe, it was the only one that did not produce negative flux in any location for all the mesh structures analyzed in the IAEA problem. Other variants of the Lagrange interpolation polynomial did not show systematically good reliability and/or accuracy.

1. Introduction

Even with the current computation speeds, the search for more efficient algorithms to solve realistic nuclear reactor problems is needed. Because the finite difference method (FDM), one of the first approach used in solving large nuclear reactor problems, requires a very large number of mesh points to accurately model many two and three dimensional problems other numerical methods have been developed to overcome this difficulty, among them, the analytic nodal method has been implemented with significant better accuracy and efficiency than the low order FDM. In this work numerical algorithms are developed as an effort in this important search.

A brief description of the numerical methods used in QUANDRY (based on an Analytic Nodal Method) and in FDMs (based on a Finite Difference Method) which will be used for comparison, is presented next without the details of the equations to be used.

The Analytic Nodal Method (ANM) used in QUANDRY, as described in (Smith, 1979), is based on the integration of the SND equation over an arbitrary volume (a node with spatially homogenized cross sections) inside the reactor. In this way an exact neutron balance equation is obtained with two unknowns: the face-averaged current and the node-averaged flux, so to be able to solve the problem and still be within the neutron diffusion theory, a relationship between those unknowns has to be established. In QUANDRY that relationship is obtained by integrating the SND equation over the directions transverse to the direction of interest. In doing that a spatially dependent-transverse leakage source term appears in the resulting one-dimensional SND equation. The assumption (a quadratic shape) made to determine that source term is the only approximation of the ANM apart from the assumption of homogenized nodes. For a two-dimensional problem QUANDRY needs to solve for 3*N* (*N* is the number of spatial nodes) unknowns per group: *N* averaged-nodal fluxes, *N* surface-averaged net leakages in *x* direction and *N* surface-averaged net leakages in *y* direction. The resulting system of algebraic equations is transformed in such a way that more spatial coupling is present in the node-average flux terms, then, it is solved iteratively using outer/inner iteration schemes. To accelerate the convergence rate of the outer iterations the eigenvalue shift method is used. For the inner iterations, a modified block Gauss–Seidel iteration is performed. At each inner iteration the flux coefficient matrix is inverted using the Cyclic Chebyshev Semi-Iterative method (a variant of the block successive over-relaxation method). It was shown that the ANM is guaranteed to give the exact solution of the two-group SND equations in the limit of infinitesimal small spatial mesh. The algebraic complexity in determining the leakage term limits the ANM to two energy groups (Lawrence, 1986).

The finite difference method used in the FDMs code is based on the integration of the SND equation over an arbitrary area (node with spatially homogenized cross sections) determined by the partitions made on the x and *y* axis. In doing that the second order derivatives (leakage terms) are reduced to a first order which are

[*] Tel.: +1 786 301 5915.
E-mail address: doserate2002@yahoo.com

0306-4549/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.anucene.2013.11.013

approximated with a finite difference in terms of the fluxes at two consecutive points. The terms representing the neutron current are eliminated by applying continuity of it at the interface between two consecutive regions (unlike the ANM where the neutron current is left as an unknown). The integral of the terms containing the flux is approximated by assuming a constant flux inside the integration domain. The code uses an edge-centered algorithm so the unknowns are the fluxes at the corner of the cells and the number of unknowns per group is smaller than N (the number of cells/nodes). The neutron flux and the eigenvalue are determined using the successive method (Duderstadt and Hamilton, 1976), one group at a time successively. The resulting system of algebraic linear equations for each group was solved with the Gauss elimination method (unlike the ANM that uses iterative procedures). Because the values of the cross sections at the common point of 4 adjacent cells are used, an area-weighted average is to be done. It is well established that the finite difference method gives the exact solution of the multi-group SND equation in the limit of infinitesimal small spatial mesh.

Table 1 shows published results of k_{eff} calculation for the TWIGL benchmark problem for QUANDRY (Smith, 1979), along with the results of TWIGL (Yasinski et al., 1968), POW (Pollard, 1977), and FDMs (Quintero-Leyva, 2012, full core) codes. Notice that the k_{eff} value of QUANDRY (quarter core result) is not very close to the rest for the same mesh size, but it is more accurate as the trend of FDMs indicates. From that table can be seen that for about the same level of accuracy FDMs has to solve for 6241 unknowns per group, while QUANDRY has to solve for only 1200 (per group for a full core) for a ratio of about 5 in favor QUANDRY's efficiency. On the other hand, for the same mesh size the ratio is about 3 in favor of FMDs efficiency but it is accompanied with the lower accuracy.

That observation, and a hope of finding an approximation to the connection between average neutron flux on each side of a surface and the surface-average net neutron current, could be to some extent equivalent to finding a reliable flux distribution in the region of interest, inspired this work. Therefore, algorithms that do not give the neutron currents (main reason for the indicated ratio of 3) as unknowns (but provide a better level of accuracy than that of the common finite difference method) were developed.

There are many methods in the literature, apart from the FDM, that have only the neutron flux as unknown: flux synthesis (Larsen, 1971), Wachspress et al. (1972) referenced in (Christensen, 1985), flux expansion (Langenbush et al., 1977a, 1977b) referenced in (Sutton and Aviles, 1996), finite elements using Lagrange polynomials referenced in (Christensen, 1985) or using basic polynomial interpolation equations (Kaper et al., 1972), etc. In this work, a 2D-Lagrange interpolation with the unknown flux as the coefficients of the interpolation equation, and without applying any variational principle, is used to solve the static neutron diffusion (SND) equations.

The method used in the LAP codes (cell centered or edge centered) is based on integrating the SND equations in a way similar to the FDMs' one but instead of using the finite difference to approximate the derivative, it is obtained by taking the analytic derivative of the bivariate Lagrange interpolation polynomial equations representing the neutron flux across 4 or more cells. The integral of the terms containing the flux is obtained by analytically integrating the Lagrange polynomial. The neutron flux and the eigenvalue are determined using the same approach used in FDMs. Lagrange polynomial greater than second degree (in the independent variable in question: x or y) can be used by adapting the progressive polynomial approximation developed in (Quintero-Leyva, 2009) for point kinetics to the current diffusion problem. This is done by progressively increasing the degree of the polynomial when going away from the external borders and progressively decreasing it when approaching the external boundaries of the reactor. For the edge-centered algorithm (as in FDMs) the number of unknowns per group is smaller than N (the number of cells/nodes) and for the cell-centered algorithm it is N.

Table 1
Reported k_{eff} calculation for the 2D-TWIGL benchmark problem.

Code	K_{eff}	Mesh size	Unknowns
TWGL	0.914193	8.0	
POW	0.914194	8.0	
QUANDRY	0.91321	8.0	1200
FDMs	0.9141935	8.0	361
	0.9136486	4.0	1521
	0.9133551	2.0	6241

2. Numerical solution of the SND equation in 2D-Cartesian geometry

The neutron diffusion differential equation for steady state can be written as (Duderstadt and Hamilton, 1976):

$$-\nabla(D_g\nabla\phi_g)+\sum_{R,g}\phi_g=\sum_{g'\neq g}^{G}\Sigma_{S,g'\to g}\phi'_g+\frac{1}{k_{eff}}\chi_g\sum_{g'=1}^{G}\nu\Sigma_{F,g'}\phi_{g'}$$

where $\sum_{R,g}$ is the neutron macroscopic removal cross section (absorption and out scatters plus, if considered, the leakage through the non-modeled dimensions).

The rest of the terms are well described in the literature. The independent variables were dropped for convenience when possible.

For two dimensions in Cartesian geometry the above equation becomes:

$$-\frac{\partial}{\partial x}\left(D_g(x,y)\frac{\partial\phi_g(x,y)}{\partial x}\right)-\frac{\partial}{\partial y}\left(D_g(x,y)\frac{\partial\phi_g(x,y)}{\partial y}\right)$$
$$+\Sigma_{R,g}(x,y)\phi_g(x,y)=S_g(x,y) \qquad (1)$$

$$S_g(x,y)=\sum_{g'\neq g}^{G}\Sigma_{S,g'\to g}(x,y)\phi_{g'}(x,y)+\frac{1}{k_{eff}}\chi_g\sum_{g'=1}^{G}\nu\Sigma_{F,g'}(x,y)\phi_{g'}(x,y)$$

Integrating Eq. (1) between $x_{i-1/2}$ and $x_{i+1/2}$ in x direction, and between $y_{j-1/2}$ and $y_{j+1/2}$ in y direction, applying continuity of current and flux, considering an average value of the nuclear constants inside each rectangular cell, and assuming a neutron flux distribution represented by a Lagrange interpolation as

$$\phi_{g,i,j}(x,y)=\sum_{m=-N_m}^{N_m}\sum_{l=-N_l}^{N_l}\phi_{gi+l,j+m}\delta_{i+l,j+m}(x,y) \qquad (2)$$

the following system of algebraic linear equations (SALE) is obtained for cell Edge and cell Centered algorithms:

$$\sum_{m=-N_m}^{N_m}\sum_{l=-N_l}^{N_l}\phi_{gi+l,,j+m}\left(-b^{inty,derx}_{Dg,i+l,j+m}-b^{intx,dery}_{Dg,i+l,j+m}+b^{y,x}_{Rg,i+l,j+m}\right)$$
$$=S_{Sg,i+l,j+m}+S_{Fg,i+l,j+m} \qquad (3)$$

where, $\phi_{gi+l,,j+m}$: Discrete values of the neutron flux at the points of interest (the unknowns)

$$b^{inty,derx}_{Dg,i+l,j+m}=\sum_{q=0}^{1}\left[-D_1T^{j+q/2,i-1/2}_{i+l,j+m}+D_2T^{j+q/2,i+1/2}_{i+l,j+m}\right]$$

$$\text{Edge}:\ D_1=D_{gi,j+q}\quad D_2=D_{g,i+1,j+q}$$

$$\text{Center}:\ D_1=\frac{D_{gi-1,j}+D_{gi,j}}{2}\quad D_2=\frac{D_{gi,j}+D_{gi+1,j}}{2}$$