Journal of Computational Physics 397 (2019) 108801

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A new numerical method for solving the Boltzmann transport equation using the PN method and the discontinuous finite elements on unstructured and curved meshes

Lahbib Bourhrara

DEN-Service d'Etudes de Réacteurs et de Mathématiques Appliquées (SERMA), CEA, Université Paris-Saclay, F-91191 Gif-sur-Yvette, France

ARTICLE INFO

Article history:
Received 11 January 2019
Received in revised form 26 April 2019
Accepted 1 July 2019
Available online 8 July 2019

Keywords:
Boltzmann transport equation
PN method
DG method
Discontinuous finite element method
Unstructured and non-conformal meshes
Curved meshes

ABSTRACT

This document presents a new numerical scheme dealing with the Boltzmann transport equation. This scheme is based on the expansion of the angular flux in a truncated spherical harmonics function and the discontinuous finite element method for the spatial variable. The advantage of this scheme lies in the fact that we can deal with unstructured, non-conformal and curved meshes. Indeed, it is possible to deal with distorted regions whose boundary is constituted by edges that can be either line segments or circular arcs or circles. In this document, we detail the derivation of the method for 2D geometries. However, the generalization to 2D extruded geometries is trivial.

1. Introduction

We propose in this work a new method of solving Boltzmann's transport equation based on the P_N method for the angular direction and the discontinuous finite element method in the spatial variable. The numerical scheme proposed is conservative, in the sense that the balance equation is respected per region of calculation. This method has the advantage of dealing with completely unstructured, non-conformal and curved meshes. The regions we consider in 2D are regions of arbitrary shape whose boundary is a finite union of straight line segments, arcs of circles or circles. Fig. 1 depicts an example of a mesh that can be handled by the method.

Recall that non-conforming meshes offer great flexibility for local refinement, and offer the possibility to deal with complex geometries. Indeed, we can refine a region or element of the mesh into two or more regions without any refinement of other regions. This cannot be done in general with conformal meshes. This makes the non-conformal meshes appreciable when using an adaptive mesh refinement (AMR). Methods dealing with non-conformal meshes automatically handle conformal meshes, the opposite is not true.

Two approximations are adopted: the first one is that the angular flux is assumed to be polynomial in space per region of calculation; and the second is that the expansion of the angular flux on spherical harmonics is truncated at a finite order N.

This method leads to matrix systems whose coefficients require calculations of integrals of polynomials over a region and over its surface as well as integrals on the angular variable over the unit sphere and over half-spheres. We will show

E-mail address: lahbib.bourhrara@cea.fr.

https://doi.org/10.1016/j.jcp.2019.07.001

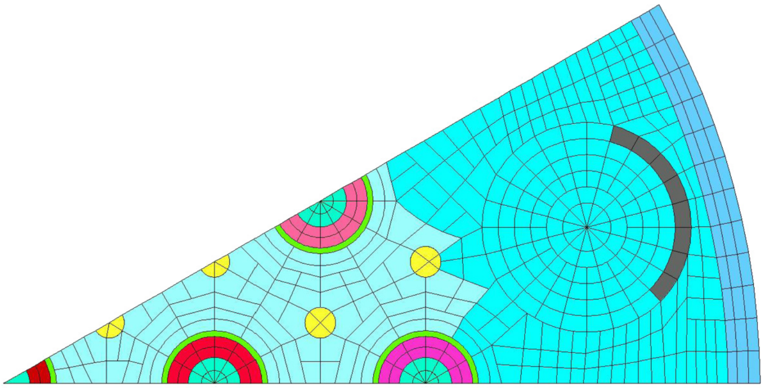

Fig. 1. Example of a 2D-mesh.

how all these integrals can be calculated exactly even for geometric regions of arbitrary shape. The computation of these integrals can be generalized without difficulty for 2D extruded geometries.

We recall that the P_N method has the advantage of not presenting the ray effects which affect the S_N method. This is because the P_N method approximates the angular flux for any direction, whereas the S_N method is limited to approximate the angular flux for a finite number of directions.

The P_N method has been widely studied in the literature and especially for the even- and odd-angular flux formulation, see for example [7], [10], [15] and [20]. However, these studies remain limited to structured and conformal meshes consisting of rectangles, triangles or hexagons for 2D geometries.

As a method of acceleration, one can use a preconditioner resulting from the same method but with a lower P_N order and/or a lower degree of polynomials in space. To our knowledge, such P_N/P_N acceleration has not been explored. This is just a suggestion for a possible acceleration, this proposal remains to be confirmed. Mathematically speaking, the acceleration by a lower order P_N can be seen as methods using pre-conditioner called AMG (Algebraic MultiGrid).

The novelty in this work lies in the fact that: on the one hand we use a variational formulation presented in [4] and to our knowledge, this formulation has not been explored numerically neither in P_N nor in S_N. On the other hand we deal with unstructured, non-conformal and curved meshes, such non-standard meshes are dealt mostly by the method of characteristics (MOC). Finally, we use a technique of integration of polynomials over regions of arbitrary shape using the divergence theorem.

The strong point of the method lies in the fact that we can calculate exactly the integrals defining the coefficients of the elementary matrices resulting from the adopted approximation, and this for arbitrarily-shaped regions. Moreover, the matrices involved in the approximation depend only on the geometry and not on cross-sections, and so not on energy groups, this makes the method reasonable in terms of memory storage.

The numerical scheme presented in this document has been implemented in C++ in the NYMO solver of the APOLLO3® code [21]. NYMO integrates two solvers NYMO-CG and NYMO-DG. These two solvers are based on the same variational formulation and both use the P_N approximation for the angular variable but they have different spatial approximations. NYMO-CG uses continuous finite element method and NYMO-DG uses discontinuous finite element method, see [11] for discontinuous Galerkin methods. NYMO's CG version deals with 1D, 2D and 3D geometries, but it is limited to structured and conformal meshes. This document discusses only the NYMO-DG version.

2. Multigroup transport problem

Consider the phase space $X = D \times S^2$ where D is the spatial domain and S^2 is the unit sphere. We use the variables $x \in D$ and $\omega \in S^2$ to denote the space and the angular variables, respectively. We denote by ∂D the boundary of D and by $n(x)$ the unit outward normal vector to D at $x \in \partial D$. We also denote by Γ and $\Gamma_\pm$ the sets:

$$\Gamma = \partial D \times S^2,$$
$$\Gamma_+ = \{(x,\omega) \in \Gamma, \quad \omega \cdot n(x) > 0\},$$
$$\Gamma_- = \{(x,\omega) \in \Gamma, \quad \omega \cdot n(x) < 0\}.$$

The neutron transport problem in its multigroup form consists to determine the multigroup flux $u = (u^g(x,\omega))_{g=1,\cdots,G}$ and eventually the associated eigenvalue λ such that:

$$\begin{cases} \omega \cdot \nabla u^g + \sigma^g u^g = H^g u + \frac{1}{\lambda} F^g u + q^g & \text{in } X, \quad (a) \\ u^g = f^g & \text{on } \Gamma_-, \quad (b) \end{cases} \tag{1}$$

where

$$(H^g u)(x,\omega) = \sum_{g'=1}^{G} \int_{S^2} \sigma_s^{g,g'}(x,\omega\cdot\omega') u^{g'}(x,\omega')\,\mathrm{d}\omega', \tag{2}$$

$$(F^g u)(x,\omega) = \sum_{\alpha} \chi_\alpha^g(x) \sum_{g'=1}^{G} \nu\sigma_{f,\alpha}^{g'}(x) \int_{S^2} u^{g'}(x,\omega')\,\mathrm{d}\omega'. \tag{3}$$

σ^g denotes the total cross-section and $\sigma_s^{g,g'}$ is the transfer cross-section from group g' to g. In Eq. (3) defining the fission operator F^g, the sum over α is done over the fissile isotopes. $\nu\sigma_{f,\alpha}^g$ and χ_α^g are respectively the fission production term, assumed isotropic, and the spectrum for isotope α. The terms $q^g(x,\omega)$ and $f^g(x,\omega)$ represent the external source and the incoming angular flux respectively.

Remark 2.1. It should be noted that u^g, in Eqs. (1a)-(3), is the flux in the group g and u denotes the multi-group flux: $u = (u^g)_{g=1,\cdots,G}$.

Remark 2.2. In the case where the external source q^g and the incoming flux f^g are both zero, the problem is said an eigenvalue problem, where u is the eigenvector associated with eigenvalue λ. In other cases we have $\lambda = 1$ and the problem is called a source problem.

Remark 2.3. The angular direction ω is described by two angles $\theta \in [0,\pi]$ and $\varphi \in [0,2\pi]$, θ and φ are the axial and azimuthal angles of ω, so that $\omega = (\sin\theta\cos\varphi, \sin\theta\sin\varphi, \cos\theta)$. In Eqs. (2) and (3) the integral over the sphere S^2 is defined by:

$$\int_{S^2} u(\omega)\,\mathrm{d}\omega = \frac{1}{4\pi}\int_0^{2\pi}\int_0^{\pi} u(\theta,\varphi)\sin\theta\,\mathrm{d}\theta\,\mathrm{d}\varphi.$$

3. Variational formulation

In the remainder of this paper, we will assume that the total cross-section σ^g is nonzero throughout the spatial domain D. Multiplying Eq. (1a) by $(v + \frac{1}{\sigma^g}(\omega\cdot\nabla v))$, where $v = v(x,\omega)$ is a test function, we obtain:

$$\begin{aligned}&\frac{1}{\sigma^g}(\omega\cdot\nabla u^g)(\omega\cdot\nabla v) + \sigma^g u^g v + u^g(\omega\cdot\nabla v) + (\omega\cdot\nabla u^g)v \\ &= \big(H^g u + \frac{1}{\lambda}F^g u + q^g\big)\big(v + \frac{1}{\sigma^g}(\omega\cdot\nabla v)\big).\end{aligned}$$

On integrating over the phase space X and after using of Green's formula, we obtain:

$$\begin{aligned}&\int_X \big(\frac{1}{\sigma^g}(\omega\cdot\nabla u^g)(\omega\cdot\nabla v) + \sigma^g u^g v\big)\,\mathrm{d}\omega\mathrm{d}x + \int_\Gamma u^g v\ (\omega\cdot\mathrm{n})\,\mathrm{d}\omega\mathrm{d}s \\ &= \int_X \big(H^g u + \frac{1}{\lambda}F^g u + q^g\big)\big(v + \frac{1}{\sigma^g}(\omega\cdot\nabla v)\big)\,\mathrm{d}\omega\mathrm{d}x.\end{aligned}$$

Splitting the integral over Γ into the sum of two integrals over Γ_+ and Γ_- and using the boundary conditions (1b), we obtain the variational formulation:

$$a^g(u^g, v) = h^g(u,v) + \frac{1}{\lambda}p^g(u,v) + L^g(v), \tag{4}$$

where

$$a^g(u^g,v) = \int_X \frac{1}{\sigma^g}(\omega\cdot\nabla u^g)(\omega\cdot\nabla v) + \sigma^g u^g v\,\mathrm{d}\omega\mathrm{d}x + \int_{\Gamma_+} u^g v\ (\omega\cdot\mathrm{n})\,\mathrm{d}\omega\mathrm{d}s, \tag{5}$$

$$h^g(u,v) = \int_X (H^g u)v + \frac{1}{\sigma^g}(H^g u)(\omega\cdot\nabla v)\,\mathrm{d}\omega\mathrm{d}x, \tag{6}$$

$$p^g(u,v) = \int_X (F^g u)v + \frac{1}{\sigma^g}(F^g u)(\omega\cdot\nabla v)\,\mathrm{d}\omega\mathrm{d}x, \tag{7}$$

$$L^g(v) = \int_X q^g v + \frac{1}{\sigma^g} q^g(\omega\cdot\nabla v)\,\mathrm{d}\omega\mathrm{d}x - \int_{\Gamma_-} f^g v\,(\omega\cdot\mathrm{n})\,\mathrm{d}\omega\mathrm{d}s. \tag{8}$$

In [4] it is established, when using an appropriate functional setting and with reasonable assumptions about the data of the problem, that the variational formulation (4) is equivalent to the transport problem (1a)-(1b). In other words, if u is solution of Eq. (4) then u is also solution of Eqs. (1a)-(1b) and vice versa. We refer to [4], [5] and [6] for more details about this variational formulation.

The variational formulation (4) can also be derived by using a second-order form of the transport equation called Self-Adjoint Angular Flux equation (SAAF). A simple derivation of the SAAF equation is given in [19]. See [19] for other references deriving the SAAF equation with different approaches.

In [14] (page 43, Eq. 3.24), the author derives a variational formulation for the SAAF equation, which resembles to the variational formulation (4) but without the integration over the angular direction. Slight modifications in the approach used in [14] will lead exactly to the formulation (4).

In [17] authors use the variational formulation (4) for non-void regions and a conservative least-squares method for void regions, the approximation is based on the classical Lagrange continuous finite elements method, and both S_N and P_N approximations are considered. See also [16] and [13] where the authors use the discontinuous finite elements method and the P_N method based on other variational formulations of the transport equation.

Remark 3.1. The formulation (4) is written for $v = 1$:

$$\int_X \sigma^g u^g\,\mathrm{d}\omega\mathrm{d}x + \int_{\Gamma_+} u^g\,(\omega\cdot\mathrm{n})\,\mathrm{d}\omega\mathrm{d}s = \int_X \left(H^g u + \frac{1}{\lambda}F^g u + q^g\right)\mathrm{d}\omega\mathrm{d}x - \int_{\Gamma_-} f^g\,(\omega\cdot\mathrm{n})\,\mathrm{d}\omega\mathrm{d}s.$$

This equation expresses the particle balance in the domain D, which shows that formulation (4) is conservative.

4. Discretization of the angular flux

We will denote by $y_\ell^k(\omega)$, see Appendix A, the real-valued spherical harmonic functions defined on the unit sphere S^2. The degree ℓ is zero or a positive integer and the order k is an integer varying from $-\ell$ to ℓ. The expansion of the flux $u^g(x,\omega)$ on the spherical harmonics is written as:

$$u^g(x,\omega) = \sum_{\ell=0}^{\infty}\sum_{k=-\ell}^{\ell} u_\ell^{k,g}(x)\,y_\ell^k(\omega).$$

The functions $u_\ell^{k,g}(x)$ are called the angular flux moments. The first two moments $u_0^{0,g}(x)$ and $u_1^{k,g}(x)$ are identical with the scalar flux and the current respectively. As a first approximation, we will assume that these flux moments are written as:

$$u_\ell^{k,g}(x) = \sum_{j=1}^{J} u_{\ell,j}^{k,g}\varphi_j(x),$$

where $(\varphi_j(x))_{j=1,\cdots,J}$ is a known basis of functions that depend only on the spatial variable x. So:

$$u^g(x,\omega) = \sum_{\ell=0}^{\infty}\sum_{k=-\ell}^{\ell}\sum_{j=1}^{J} u_{\ell,j}^{k,g}\varphi_j(x)\,y_\ell^k(\omega).$$

The P_N approximation of the angular flux consists of truncating the infinite sum over ℓ to an order N:

$$u^g(x,\omega) = \sum_{\ell=0}^{N}\sum_{k=-\ell}^{\ell}\sum_{j=1}^{J} u_{\ell,j}^{k,g}\,\varphi_j(x)\,y_\ell^k(\omega). \tag{9}$$

In the same way we approximate the source term $q^g(x,\omega)$ and the incoming flux $f^g(x,\omega)$ by expanding them on the basis of functions $\left(\varphi_j(x)y_\ell^k(\omega)\right)$:

$$q^g(x,\omega) = \sum_{\ell=0}^{N}\sum_{k=-\ell}^{\ell}\sum_{j=1}^{J} q_{\ell,j}^{k,g}\,\varphi_j(x)\,y_\ell^k(\omega), \tag{10}$$

$$f^g(x,\omega) = \sum_{\ell=0}^{N} \sum_{k=-\ell}^{\ell} \sum_{j=1}^{J} f_{\ell,j}^{k,g}\, \varphi_j(x) y_\ell^k(\omega). \tag{11}$$

Recall that the incoming flux $f^g(x,\omega)$ is defined only for $(x,\omega) \in \Gamma_-$.

In the following, we consider a partition of the domain D in a set of homogeneous and non-overlapping regions or elements D_r:

$$\mathsf{D} = \bigcup_r \mathsf{D}_r.$$

The choice of the basis of functions (φ_j) play a crucial role in the approximation of the spatial variable. In a first step, we assume that the functions (φ_j) are polynomials and linearly independent, they may be dependent on the region D_r. We specify later in Section 13 the choice of these polynomials used in the NYMO solver.

Remark 4.1. It should be noted that in the case of 2D geometries, for reasons of symmetry, the sum over k in Eqs. (9), (10) and (11) is reduced only to k having the same parity as ℓ, see [1] page 192. Likewise for 1D geometries, the sum over k is reduced to $k=0$. Thus, the number of degrees of freedom, for a given energy group g and a given region D_r, is $N^\star = J(N+1)^2$ in 3D, $N^\star = J(N+1)(N+2)/2$ in 2D and $N^\star = J(N+1)$ in 1D.

The idea to obtain the discretized problem consists in applying the variational formulation (4) for each region D_r of the domain D, which amounts to taking as phase space $\mathsf{X}_r = \mathsf{D}_r \times \mathsf{S}^2$ in (4). Then we apply Galerkin's method, for both spatial and angular variables, by replacing u^g, q^g and f^g by their approximations given by Eqs. (9), (10) and (11) as well as v by $\varphi_i y_n^m$, which gives for all (i,n,m):

$$a^g(u^g, \varphi_i y_n^m) = h^g(u, \varphi_i y_n^m) + \frac{1}{\lambda} p^g(u, \varphi_i y_n^m) + L^g(\varphi_i y_n^m). \tag{12}$$

We stress that the incoming flux f^g at the faces of region D_r is given either by the boundary conditions or by the flux in the adjacent regions to the region D_r. Equation (12) can be written in matrix form whose unknowns are $u_{\ell,j}^{k,g}$. In the following subsections we develop each terms of the above equation.

4.1. Calculation of the term $a^g(u^g, \varphi_i y_n^m)$

Let us start by splitting the term $a^g(u^g, v)$ of Eq. (5) into three terms:

$$a^g(u^g, v) = a_0^g(u^g, v) + a_1^g(u^g, v) + a^+(u^g, v),$$

with

$$a_0^g(u^g, v) = \int_{\mathsf{X}_r} \frac{1}{\sigma^g} (\omega \cdot \nabla u^g)(\omega \cdot \nabla v)\, \mathrm{d}\omega \mathrm{d}x,$$

$$a_1^g(u^g, v) = \int_{\mathsf{X}_r} \sigma^g u^g v\, \mathrm{d}\omega \mathrm{d}x,$$

$$a^+(u^g, v) = \int_{\Gamma_+} u^g v\, (\omega \cdot \mathsf{n})\, \mathrm{d}\omega \mathrm{d}s.$$

Here, $\mathsf{X}_r = \mathsf{D}_r \times \mathsf{S}^2$ where D_r is a calculation region and $\Gamma_+ = \{(x,\omega) \in \partial\mathsf{D}_r \times S^2,\ \omega \cdot \mathsf{n}(x) > 0\}$. In order to get the expressions of a_0^g, a_1^g and a^+ completely discretized, we use the approximation (9) for u^g and we replace v with $\varphi_i y_n^m$. After rearranging the terms and using the orthonormalization of the spherical harmonics, see Appendix A, we get:

$$a_0^g(u^g, \varphi_i y_n^m) = \sum_{\ell,k,j} A_r^0(i,n,m;\, j,\ell,k)\, \frac{1}{\sigma^g}\, u_{\ell,j}^{k,g}, \tag{13}$$

$$a_1^g(u^g, \varphi_i y_n^m) = \sum_{\ell,k,j} A_r^1(i,n,m;\, j,\ell,k)\, \sigma^g\, u_{\ell,j}^{k,g}, \tag{14}$$

$$a^+(u^g, \varphi_i y_n^m) = \sum_{\ell,k,j} A_r^+(i,n,m;\, j,\ell,k)\, u_{\ell,j}^{k,g}, \tag{15}$$

with

$$A_r^0(i,n,m;\,j,\ell,k)=\sum_{p=1}^{3}\sum_{q=1}^{3}\Big(\int\limits_{\mathsf{D}_r}\partial_p\varphi_i\partial_q\varphi_j\,\mathrm{d}x\Big)\Big(\int\limits_{\mathsf{S}^2}\omega_p\omega_q y_n^m y_\ell^k\,\mathrm{d}\omega\Big), \tag{16}$$

$$A_r^1(i,n,m;\,j,\ell,k)=(\delta_{n,\ell}\delta_{m,k})\int\limits_{\mathsf{D}_r}\varphi_i\varphi_j\,\mathrm{d}x, \tag{17}$$

$$A_r^+(i,n,m;\,j,\ell,k)=\sum_{\mathsf{f}\in\partial\mathsf{D}_r}\int\limits_{\mathsf{f}}\varphi_i\varphi_j\int\limits_{(\omega\cdot\mathsf{n})>0}y_n^m y_\ell^k\,(\omega\cdot\mathsf{n})\,\mathrm{d}\omega\mathrm{d}s. \tag{18}$$

Where ∂_1, ∂_2 and ∂_3 refer to ∂_x, ∂_y and ∂_z respectively. Similarly ω_1, ω_2 and ω_3 refer to ω_x, ω_y and ω_z. $\delta_{n,\ell}$ is the Kronecker symbol. The sum $\sum_{\ell,k,j}$ is an abbreviation for $\sum_{\ell=0}^{N}\sum_{k=-\ell}^{\ell}\sum_{j=1}^{J}$ with of course the restrictions on k for one- and two-dimensional geometries as mentioned in Remark 4.1. In the definition of matrix A_r^+ the sum over f is done over all faces of the boundary $\partial\mathsf{D}_r$ of region D_r.

Note that if n and ℓ are of opposite parity we have $A_r^0(i,n,m;\,j,\ell,k)=0$, because the integral over all directions of an odd function of ω vanishes.

If the functions (φ_j) are orthonormalized in the region D_r, then $A_r^1(i,n,m;\,j,\ell,k)=\delta_{n,\ell}\delta_{m,k}\delta_{i,j}$ and the matrix A_r^1 is reduced to the identity matrix.

The matrices A_r^0, A_r^1 and A_r^+ are symmetric and we will show later that the integrals defining the coefficients of these matrices can be calculated exactly for regions of arbitrary shape, for any degree of polynomials and at any order N.

4.2. Calculation of the scattering source $h^g(u,\varphi_i y_n^m)$

Let us split the term $h^g(u,v)$ of Eq. (6) into two terms:

$$h^g(u,v)=h_1^g(u,v)+h_2^g(u,v),$$

with

$$h_1^g(u,v)=\int\limits_{\mathsf{X}_r}(H^g u)v\,\mathrm{d}\omega\mathrm{d}x,$$

$$h_2^g(u,v)=\int\limits_{\mathsf{X}_r}\frac{1}{\sigma^g}(H^g u)(\omega\cdot\nabla v)\,\mathrm{d}\omega\mathrm{d}x.$$

The addition theorem leads to:

$$\sigma_s^{g,g'}(x,\omega\cdot\omega')=\sum_{n=0}^{a}\sigma_s^{n;g,g'}\sum_{m=-n}^{n}y_n^m(\omega)y_n^m(\omega'),$$

where a is the order of anisotropy in the region D_r. Thus, on using the discretized flux given by Eq. (9), the scattering operator (2) is written as:

$$(H^g u)(x,\omega)=\sum_{\ell,k,j}\varphi_j(x)y_\ell^k(\omega)\sum_{g'=1}^{G}\sigma_s^{\ell;g,g'}u_{\ell,j}^{k,g'}.$$

Substituting this expression of $(H^g u)$ into the definitions of h_1^g and h_2^g, we get:

$$h_1^g(u,\varphi_i y_n^m)=\sum_{\ell,k,j}A_r^1(i,n,m;\,j,\ell,k)\sum_{g'=1}^{G}\sigma_s^{\ell;g,g'}\,u_{\ell,j}^{k,g'}, \tag{19}$$

$$h_2^g(u,\varphi_i y_n^m)=\sum_{\ell,k,j}A_r^2(i,n,m;\,j,\ell,k)\sum_{g'=1}^{G}\frac{\sigma_s^{\ell;g,g'}}{\sigma^g}\,u_{\ell,j}^{k,g'}, \tag{20}$$

where the matrix A_r^1 is defined by (17) and the matrix A_r^2 is given by:

$$A_r^2(i,n,m;\,j,\ell,k)=\sum_{p=1}^{3}\int\limits_{\mathsf{D}_r}(\partial_p\varphi_i)\varphi_j\,\mathrm{d}x\int\limits_{\mathsf{S}^2}\omega_p y_n^m y_\ell^k\,\mathrm{d}\omega. \tag{21}$$

In Eqs. (19)-(20), $\sigma_s^{\ell;g,g'}$ is zero if ℓ is strictly greater than the order of anisotropy in the region D_r.

Note that if n and ℓ have the same parity we have $A_r^2(i,n,m;\,j,\ell,k)=0$, because the integral over the unit sphere of an odd function of ω vanishes.

Remark 4.2. It is possible, if we want to avoid inner iterations, to switch the scattering source term of the group g into itself in the left-hand side of Eq. (12), on writing:

$$(a^g - h_1^{g,g} - h_2^{g,g})(u^g, \varphi_i y_n^m) = h^{\star,g}(u, \varphi_i y_n^m) + \frac{1}{\lambda} p^g(u, \varphi_i y_n^m) + L^g(\varphi_i y_n^m),$$

with

$$h_1^{g,g}(u^g, \varphi_i y_n^m) = \sum_{\ell,k,j} A_r^1(i,n,m;\,j,\ell,k)\,\sigma_s^{\ell;g,g}\,u_{\ell,j}^{k,g},$$

$$h_2^{g,g}(u^g, \varphi_i y_n^m) = \sum_{\ell,k,j} A_r^2(i,n,m;\,j,\ell,k)\,\frac{\sigma_s^{\ell;g,g}}{\sigma^g}\,u_{\ell,j}^{k,g}.$$

The term $h^{\star,g} = h_1^{\star,g} + h_2^{\star,g}$ is written as $h^g = h_1^g + h_2^g$ by replacing $\sum_{g'=1}^{G}$ with $\sum_{g'=1;g'\neq g}^{G}$. On the other hand, we have:

$$(a_1^g - h_1^{g,g})(u^g, \varphi_i y_n^m) = \sum_{\ell,k,j} A_r^1(i,n,m;\,j,\ell,k)\,\sigma_a^{\ell,g}\,u_{\ell,j}^{k,g},$$

where $\sigma_a^{\ell,g} = \sigma^g - \sigma_s^{\ell;g,g}$ is the effective absorption section.

4.3. Calculation of the fission source $p^g(u, \varphi_i y_n^m)$

Let us split the term $p^g(u,v)$ defined by Eq. (7) into two terms:

$$p^g(u,v) = p_1^g(u,v) + p_2^g(u,v),$$

where

$$p_1^g(u,v) = \int_{\mathsf{X}_r} (F^g u)v \,\mathrm{d}\omega \mathrm{d}x,$$

$$p_2^g(u,v) = \int_{\mathsf{X}_r} \frac{1}{\sigma^g}(F^g u)(\omega\cdot\nabla v)\,\mathrm{d}\omega \mathrm{d}x.$$

The discretized terms p_1^g and p_2^g are written as:

$$p_1^g(u, \varphi_i y_n^m) = \sum_{\ell,k,j} F_r^1(i,n,m;\,j,\ell,k)\,\sum_{\alpha} \chi_\alpha^g \sum_{g'=1}^{G} \nu\sigma_{f,\alpha}^{g'} u_{0,j}^{0,g'}, \tag{22}$$

$$p_2^g(u, \varphi_i y_n^m) = \sum_{\ell,k,j} F_r^2(i,n,m;\,j,\ell,k)\,\sum_{\alpha} \frac{\chi_\alpha^g}{\sigma^g} \sum_{g'=1}^{G} \nu\sigma_{f,\alpha}^{g'} u_{0,j}^{0,g'}, \tag{23}$$

with

$$F_r^1(i,n,m;\,j,\ell,k) = \delta_{n,0}\delta_{m,0}\delta_{\ell,0}\delta_{k,0}\int_{\mathsf{D}_r} \varphi_i\varphi_j \,\mathrm{d}x, \tag{24}$$

$$F_r^2(i,n,m;\,j,\ell,k) = \delta_{n,1}\delta_{\ell,0}\delta_{k,0}\sum_{p=1}^{3}\int_{\mathsf{D}_r} (\partial_p\varphi_i)\varphi_j \,\mathrm{d}x \int_{\mathsf{S}^2} \omega_p y_n^m \,\mathrm{d}\omega. \tag{25}$$

4.4. Calculation of the source term $L^g(\varphi_i y_n^m)$

Let us split the term $L^g(v)$ defined by Eq. (8) into three terms:

$$L^g(v) = L_1^g(v) + L_2^g(v) - L_-^g(v),$$

where

$$L_1^g(v) = \int_{\mathsf{X}_r} q^g v \, \mathrm{d}\omega \mathrm{d}x,$$

$$L_2^g(v) = \int_{\mathsf{X}_r} \frac{1}{\sigma^g} q^g (\omega \cdot \nabla v) \, \mathrm{d}\omega \mathrm{d}x,$$

$$L_-^g(v) = \int_{\Gamma_-} f^g v \, (\omega \cdot \mathsf{n}) \, \mathrm{d}\omega \mathrm{d}s.$$

Using approximations (9), (10) and (11), we obtain after some algebra:

$$L_1^g(\varphi_i y_n^m) = \sum_{\ell,k,j} A_r^1(i,n,m;\, j,\ell,k)\, q_{\ell,j}^{k,g}, \tag{26}$$

$$L_2^g(\varphi_i y_n^m) = \sum_{\ell,k,j} A_r^2(i,n,m;\, j,\ell,k)\, \frac{1}{\sigma^g}\, q_{\ell,j}^{k,g}, \tag{27}$$

$$L_-^g(\varphi_i y_n^m) = \sum_{\mathsf{f}\in\partial \mathsf{D}_r} \sum_{\ell,k,j} A_{\mathsf{f}}^-(i,n,m;\, j,\ell,k)\, f_{\ell,j}^{k,g}, \tag{28}$$

where the matrices A_r^1 and A_r^2 are defined by Eqs. (17) and (21) and the matrix A_{f}^- is given by:

$$A_{\mathsf{f}}^-(i,n,m;\, j,\ell,k) = \int_{\mathsf{f}} \varphi_i \varphi_j \int_{(\omega\cdot \mathsf{n})<0} y_n^m y_\ell^k \, (\omega \cdot \mathsf{n}) \, \mathrm{d}\omega \mathrm{d}s. \tag{29}$$

If the functions (φ_j) are orthonormalized in the region D_r, then $L_1^g(\varphi_i y_n^m) = q_{n,i}^{m,g}$.

Note that the incoming flux $f_{\ell,j}^{k,g}$ in Eq. (28) can be given: either by the boundary conditions if the face f is a part of the boundary $\partial\mathsf{D}$ of the domain D or by the flux in the adjacent regions when the face f is an interface between two regions.

Remark 4.3. The term $L_-^g(\varphi_i y_n^m)$ is the only one linking the flux in region D_r to the fluxes of its adjacent regions, i.e. sharing the same face.

5. Solving the discretized problem

Before showing how one can calculate the coefficients of the matrices A_r^0, A_r^1, A_r^2, F_r^1, F_r^2, A_r^+ and A_{f}^-, we give in this section the strategy used in the NYMO code to solve the discretized problem.

The discretization of the transport problem presented in the above sections leads to a matrix problem per energy group g and region D_r which is written by considering Remark 4.2 and after numbering the degrees of freedom (j,ℓ,k):

$$A_r^g\, u^g = Q_r^g\, u, \tag{30}$$

with

$$A_r^g u^g = \left(A_r^0\, \mathsf{d}_r^{0,g} + A_r^1\, \mathsf{d}_r^{1,g} - A_r^2\, \mathsf{d}_r^{2,g} + A_r^+ \right) u^g + \sum_{\mathsf{f}\in\partial\mathsf{D}_r} A_{\mathsf{f}}^-\, \widetilde{u}_{\mathsf{f}}^g, \tag{31}$$

and where the matrices $\mathsf{d}_r^{0,g}$, $\mathsf{d}_r^{1,g}$ and $\mathsf{d}_r^{2,g}$ are diagonal:

$$\mathsf{d}_r^{0,g}(i,n,m;\, j,\ell,k) = \delta_{n,\ell}\delta_{m,k}\delta_{i,j} \left(\frac{1}{\sigma^g} \right),$$

$$\mathsf{d}_r^{1,g}(i,n,m;\, j,\ell,k) = \delta_{n,\ell}\delta_{m,k}\delta_{i,j} \left(\sigma_a^{\ell,g} \right),$$

$$\mathsf{d}_r^{2,g}(i,n,m;\, j,\ell,k) = \delta_{n,\ell}\delta_{m,k}\delta_{i,j} \left(\frac{\sigma_s^{\ell;g,g}}{\sigma^g} \right),$$

where $\sigma_s^{\ell;g,g}$ is zero if ℓ is strictly greater than the order of anisotropy in the region D_r. u^g is the flux in the group g and in the region D_r given by its components $(u_{\ell,j}^{k,g})$ and $\widetilde{u}_\mathsf{f}^g$ is either an incoming flux given by the boundary conditions or the flux in the region $\mathsf{D}_{\widetilde{r}}$ adjacent to the region D_r by the face f.

Q_r^g represents the source term regrouping the scattering, fission and eventually an external source:

$$Q_r^g u = A_r^1 \left(\widehat{s}^{\,g}\right) + A_r^2 \left(\frac{1}{\sigma^g}\widehat{s}^{\,g}\right) + \frac{1}{\lambda}\left(F_r^1 \left(\widehat{p}^{\,g}\right) + F_r^2 \left(\frac{1}{\sigma^g}\widehat{p}^{\,g}\right)\right) + A_r^1 \left(q^g\right) + A_r^2 \left(\frac{1}{\sigma^g} q^g\right), \tag{32}$$

where

$$\widehat{s}_{\ell,j}^{\,k,g} = \sum_{g'=1, g'\neq g}^{G} \sigma_s^{\ell;g,g'} u_{\ell,j}^{k,g'},$$

$$\widehat{p}_{\ell,j}^{\,k,g} = \delta_{\ell,0}\delta_{k,0} \sum_{\alpha} \chi_\alpha^g \sum_{g'=1}^{G} \nu\sigma_{f,\alpha}^{g'} u_{0,j}^{0,g'}.$$

The matrices A_r^0, A_r^1, A_r^2, A_r^+ and A_f^- depend on the region D_r, but not on the energy group g.

Now, by giving a global numbering of degrees of freedom (r, j, ℓ, k) and after assembling the matrices A_r^0, A_r^1, A_r^2, A_r^+, A_f^-, $\mathrm{d}_r^{0,g}$, $\mathrm{d}_r^{1,g}$ and $\mathrm{d}_r^{2,g}$ into global matrices A^0, A^1, A^2, A^+, A^-, $\mathrm{d}^{0,g}$, $\mathrm{d}^{1,g}$ and $\mathrm{d}^{2,g}$ we get the matrix system per energy group:

$$A^g\, u^g = Q^g\, u, \tag{33}$$

with

$$A^g u^g = \left(A^0 \mathrm{d}^{0,g} + A^1 \mathrm{d}^{1,g} - A^2 \mathrm{d}^{2,g} + A^+ + A^-\right) u^g. \tag{34}$$

The matrices $\mathrm{d}^{0,g}$, $\mathrm{d}^{1,g}$ and $\mathrm{d}^{2,g}$ are diagonal. The matrices A^0, A^1, A^2, A^+ and A^- do not depend on the energy group g. In addition, these matrices are sparse because the flux in a region D_r is coupled only to the fluxes of its neighbor regions. NYMO uses the Compressed Sparse Row format (CSR) to store only the nonzero coefficients of these matrices.

The matrix A^g is not symmetric and the resolution of the system (33) requires solvers dealing with non-symmetric matrices such as GMRES and BiCGSTAB. Moreover, such solvers are well suited to solve system (33) because they only need matrix-vector operation that can be computed using (34) by storing only matrices A^0, A^1, A^2, $A^+ + A^-$, $\mathrm{d}^{0,g}$, $\mathrm{d}^{1,g}$ and $\mathrm{d}^{2,g}$. This avoids the storage of matrices A^g which can be expensive in cases with many energy groups.

The matrix-vector operation with sparse matrices is easily parallelizable in shared memory using OpenMP.

Another level of parallelization in distributed memory can be used with MPI, by grouping calculation regions D_r into some agglomerations D_R of regions D_r. The assembly of the matrices A_r^0, A_r^1, A_r^2, A_r^+, A_f^-, $\mathrm{d}_r^{0,g}$, $\mathrm{d}_r^{1,g}$ and $\mathrm{d}_r^{2,g}$ can be done in distributed memory for each agglomeration D_R of regions, likewise the matrix-vector product can also be done in distributed memory. This approach is similar to the domain decomposition method where each agglomeration D_R of regions is considered as a subdomain.

In the NYMO code, the two solvers GMRES and BiCGSTAB are implemented and the matrix-vector operation is parallelized using OpenMP. However, the parallelization using MPI is not yet implemented. For the moment we use the most trivial preconditioner namely the diagonal of the matrix A^g. But one can consider preconditioners resulting from the same method but with a lower $\mathrm{P_N}$ order and/or a lower degree of polynomials in space.

Equation (33) has two levels of coupling multigroup fluxes (u^g) via both the scattering and the fission sources. These couplings are treated conventionally using outer and thermal iterations.

6. Calculation of integrals over the angular variable

The integrals over the angular variable intervening in the coefficients of matrices A_r^0, A_r^2, F_r^2, A_r^+ and A_f^- are

$$\int_{\mathsf{S}^2} \omega_p \omega_q y_n^m y_\ell^k \,\mathrm{d}\omega, \quad \int_{\mathsf{S}^2} \omega_p y_n^m y_\ell^k \,\mathrm{d}\omega \quad \text{and} \quad \int_{\pm(\omega\cdot\mathsf{n})>0} y_n^m y_\ell^k (\omega\cdot\mathsf{n}) \,\mathrm{d}\omega. \tag{35}$$

In order to explain how we can calculate these integrals, we start by defining the set E of functions with a single real variable $f(\theta)$ which are written

$$f(\theta) = a_0 + \sum_{i\geq 1} a_c^i \cos(i\,\theta) + \sum_{j\geq 1} a_s^j \sin(j\,\theta), \tag{36}$$

where the sums over integers i and j are finite and coefficients a_0, a_c^i and a_s^j are real numbers. Let us also define the set Y of two real variable functions $y(\theta, \varphi)$ by:

$$\mathsf{Y}=\{y(\theta,\varphi)=f(\theta)g(\varphi);\ \text{with} \quad f\in\mathsf{E}\ \text{and}\ g\in\mathsf{E}\}. \tag{37}$$

It's easy to see that the set E is a vector space. What will interest us more in the following is the fact that E is stable by multiplication: if $f_1(\theta)\in\mathsf{E}$ and $f_2(\theta)\in\mathsf{E}$ then $f_1(\theta)f_2(\theta)\in\mathsf{E}$. This results by linearization of product of functions $\cos(i\ \theta)$ and $\sin(j\ \theta)$.

Therefore the set Y is also stable by multiplication: if $y_1(\theta,\varphi)\in\mathsf{Y}$ and $y_2(\theta,\varphi)\in\mathsf{Y}$ then $y_1(\theta,\varphi)y_2(\theta,\varphi)\in\mathsf{Y}$.

Functions $y_n^m(\theta,\varphi)$, $\omega_p(\theta,\varphi)$ and $\sin\theta$ belong to the set Y. And since the latter is stable by multiplication, functions $(\omega_p\omega_q y_n^m y_\ell^k\sin\theta)$ and $(\omega_p y_n^m y_\ell^k\sin\theta)$ which appear under the integrals in (35) are also functions of Y. The integrand $\sin\theta$ results from taking into account that $\mathrm{d}\omega=(1/4\pi)\sin\theta\ \mathrm{d}\theta\ \mathrm{d}\varphi$.

Thus, integrals (35) are reduced to integrals in the form:

$$\int_{\theta_0}^{\theta_1}\int_{\varphi_0}^{\varphi_1} f(\theta)\ g(\varphi)\ \mathrm{d}\theta\ \mathrm{d}\varphi=\int_{\theta_0}^{\theta_1} f(\theta)\ \mathrm{d}\theta\int_{\varphi_0}^{\varphi_1} g(\varphi)\ \mathrm{d}\varphi,$$

with $f\in\mathsf{E}$ and $g\in\mathsf{E}$.

For integrals over the entire sphere S^2, we have $\theta_0=0$, $\theta_1=\pi$, $\varphi_0=0$ and $\varphi_1=2\pi$. For the integrals over the outgoing half-sphere $(\omega\cdot\mathsf{n}>0)$ and in the 2D case, we have $\theta_0=0$, $\theta_1=\pi$, $\varphi_0=\varphi_\mathsf{n}-\pi/2$ and $\varphi_1=\varphi_\mathsf{n}+\pi/2$ where φ_n is the measure of the angle formed by the x-axis and the normal n. For integrals over the incoming half-sphere $(\omega\cdot\mathsf{n}<0)$ we have $\varphi_0=\varphi_\mathsf{n}+\pi/2$ and $\varphi_1=\varphi_\mathsf{n}+3\pi/2$.

Remark 6.1. The two first integrals over the entire sphere S^2 in Eq. (35) have been studied in [3]. We find in this reference the analytic expressions of these integrals and even the C++ source code calculating them. Integrals over the half-sphere are not discussed in [3].

When n and ℓ are of opposite parity, the integrals over the half-sphere in Eq. (35) can be recast into integrals over the whole sphere:

$$\int_{\pm(\omega\cdot\mathsf{n})>0} y_n^m y_\ell^k(\omega\cdot\mathsf{n})\ \mathrm{d}\omega=\frac{1}{2}\int_{S^2} y_n^m y_\ell^k(\omega\cdot\mathsf{n})\ \mathrm{d}\omega=\frac{1}{2}\sum_{p=1}^{3}\mathsf{n}_p\int_{S^2}\omega_p y_n^m y_\ell^k\ \mathrm{d}\omega.$$

This results from the fact that the integrand $y_n^m y_\ell^k(\omega\cdot\mathsf{n})$ is even of ω. Thus, in the case where n and ℓ have opposite parity, the integrals over the half-sphere also have an analytic expression which can be found in [3].

Remark 6.2. It should be noted that the surface integrals involved in the coefficients of matrices A_r^+ and A_f^-:

$$\int_\mathsf{f}\varphi_i\varphi_j\int_{\pm(\omega\cdot\mathsf{n})>0} y_n^m y_\ell^k(\omega\cdot\mathsf{n})\ \mathrm{d}\omega\ \mathrm{d}s,$$

can be written, in the case where the face f is a line segment, as:

$$\left(\int_\mathsf{f}\varphi_i\varphi_j\ \mathrm{d}s\right)\times\left(\int_{\pm(\omega\cdot\mathsf{n})>0} y_n^m y_\ell^k(\omega\cdot\mathsf{n})\ \mathrm{d}\omega\right).$$

This results from the fact that the normal vector n is constant along the face f. However, the integral over the surface and the integral over the half-sphere remain coupled in the case where the face is a circular arc, but even in this case the integrals can be calculated exactly as we will see in the following.

7. Integral of a monomial over a 2D region

Coefficients of matrices A_r^0, A_r^1, A_r^2, F_r^1 and F_r^2 require the computation of the integrals over a region D_r of the polynomials: $\varphi_i\varphi_j$, $(\partial_p\varphi_i)(\partial_q\varphi_j)$ and $(\partial_p\varphi_i)\varphi_j$. As the computation of the integral of a polynomial leads to calculate the integral of its monomials, we will limit ourselves to explaining how to calculate the integral of a monomial X^iY^j over a region D_r.

Consider a region D_r having a boundary $\partial\mathsf{D}_r$ that can be partitioned into some finite number of edges:

$$\partial\mathsf{D}_r=\bigcup_{\mathsf{e}=1}^{N^\mathsf{e}}\mathcal{A}_\mathsf{e}.$$

Each edge $\mathcal{A}_\mathsf{e}$ can be line segments, arc of circles or circles. We will denote by $\mathsf{n}_\mathsf{e}=(\mathsf{n}_\mathsf{e}^x,\mathsf{n}_\mathsf{e}^y)$ the unit normal vector on the edge $\mathcal{A}_\mathsf{e}$ oriented outside of the region D_r.

In order to compute the integral of the monomial X^iY^j over the region D_r, consider the vector field:

$$\vec{F}(X,Y) = \big(\frac{1}{i+1}X^{i+1}Y^j,\ 0\big),$$

this field is chosen so that: $div\,\vec{F} = X^iY^j$. On using the divergence theorem:

$$\int\limits_{\mathsf{D}_r} X^iY^j\,\mathrm{d}x\,\mathrm{d}y = \int\limits_{\mathsf{D}_r} div\ \vec{F}\,\mathrm{d}x\,\mathrm{d}y = \int\limits_{\partial\mathsf{D}_r} \vec{F}\cdot\mathsf{n}\,\mathrm{d}s = \sum_{\mathsf{e}=1}^{N^{\mathsf{e}}}\int\limits_{\mathcal{A}_{\mathsf{e}}} \vec{F}\cdot\mathsf{n}_{\mathsf{e}}\,\mathrm{d}s.$$

Thus

$$\int\limits_{\mathsf{D}_r} X^iY^j\,\mathrm{d}x\,\mathrm{d}y = \frac{1}{i+1}\sum_{\mathsf{e}=1}^{N^{\mathsf{e}}}\int\limits_{\mathcal{A}_{\mathsf{e}}} X^{i+1}Y^j\,\mathsf{n}_{\mathsf{e}}^x\,\mathrm{d}s.$$

The last equation recasts the volume integral of a monomial X^iY^j over a region D_r into the surface integrals over edges $\mathcal{A}_{\mathsf{e}}$ constituting the boundary $\partial\mathsf{D}_r$. The following sections give the details for calculating the surface integrals:

$$I = \int\limits_{\mathcal{A}_{\mathsf{e}}} X^{i+1}Y^j\,\mathsf{n}_{\mathsf{e}}^x\,\mathrm{d}s.$$

The expressions of these integrals differ according to the shape of the edge: line segment, circular arc or circle.

7.1. Case where the edge is a line segment

Consider the case where the edge $\mathcal{A} = (A, B)$ is a line segment, with $A(x_A, y_A)$ and $B(x_B, y_B)$. By setting $\Delta_x = x_B - x_A$, $\Delta_y = y_B - y_A$, $L = \sqrt{(\Delta_x)^2 + (\Delta_y)^2}$ and noting that the normal vector $\mathsf{n}(\mathsf{n}^x, \mathsf{n}^y)$ is constant along the line segment, we obtain:

$$I = \int\limits_{\mathcal{A}} X^{i+1}Y^j\,\mathsf{n}^x\,\mathrm{d}s = \mathsf{n}^x L\int\limits_0^1 \big(\Delta_x s + x_A\big)^{i+1}\,\big(\Delta_y s + y_A\big)^j\,\mathrm{d}s.$$

Using the binomial formula:

$$I = \mathsf{n}^x L\sum_{i'=0}^{i+1}\sum_{j'=0}^{j}\binom{i+1}{i'}\binom{j}{j'}(x_A)^{i+1-i'}(\Delta_x)^{i'}(y_A)^{j-j'}(\Delta_y)^{j'}\int\limits_0^1 s^{i'+j'}\,\mathrm{d}s,$$

with

$$\binom{i}{i'} = \frac{i!}{i'!(i-i')!}.$$

Moreover, we have $\mathsf{n} = \frac{\kappa}{L}(\Delta_y, -\Delta_x)$ where $\kappa = 1$ if region D_r is at the left of the line segment (A, B) when going from A to B and $\kappa = -1$ otherwise. Thus $\mathsf{n}^x = \frac{\kappa\Delta_y}{L}$ and finally:

$$I = \kappa\,\Delta_y\sum_{i'=0}^{i+1}\binom{i+1}{i'}(x_A)^{i+1-i'}(\Delta_x)^{i'}\sum_{j'=0}^{j}\binom{j}{j'}(y_A)^{j-j'}(\Delta_y)^{j'}\frac{1}{i'+j'+1}. \tag{38}$$

7.2. Case where the edge is an arc of circle

In the case where $\mathcal{A}$ is an arc defined by its radius ρ, its center $C(x_C, y_C)$ and its endpoints (A, B) and denoting α the angle formed by the x-axis and the vector $\overrightarrow{CA}$ and β the angle formed by the x-axis and the vector $\overrightarrow{CB}$, we obtain

$$I = \int\limits_{\mathcal{A}} X^{i+1}Y^j\,\mathsf{n}^x\,\mathrm{d}s = \kappa\rho\int\limits_{\alpha}^{\beta} \big(x_C + \rho\cos s\big)^{i+1}\,\big(y_C + \rho\sin s\big)^j\cos s\,\mathrm{d}s.$$

In the above equation we used the fact that $\mathsf{n}^x = \kappa\cos s$, with $\kappa = 1$ if the region D_r is at the left of the arc when going from A to B in the trigonometric sense and $\kappa = -1$ otherwise. Using the binomial formula:

$$I=\kappa\rho\int_{\alpha}^{\beta}\left(\sum_{i'=0}^{i+1}\binom{i+1}{i'}(x_C)^{i+1-i'}\rho^{i'}\cos^{i'}s\right)\left(\sum_{j'=0}^{j}\binom{j}{j'}(y_C)^{j-j'}\rho^{j'}\sin^{j'}s\right)(\cos s)\,\mathrm{d}s,$$

and finally

$$I=\kappa\rho\sum_{i'=0}^{i+1}\binom{i+1}{i'}(x_C)^{i+1-i'}\rho^{i'}\sum_{j'=0}^{j}\binom{j}{j'}(y_C)^{j-j'}\rho^{j'}\int_{\alpha}^{\beta}(\cos^{i'+1}s)(\sin^{j'}s)\,\mathrm{d}s. \tag{39}$$

Thus, the computation of the integral I is reduced to integrate functions of the form $(\cos^N(s)\ \sin^M(s))$. In NYMO code, we proceed by linearization of the function $(\cos^N(s)\ \sin^M(s))$ to compute its integral. To do this, we write

$$\cos^N(s)\ \sin^M(s)=\Big(\frac{1}{2}(e^{is}+e^{-is})\Big)^N\Big(\frac{1}{2i}(e^{is}-e^{-is})\Big)^M.$$

By expanding the right-hand member using the binomial formula, we obtain:

$$\cos^N(s)\ \sin^M(s)=\frac{1}{2^{N+M}(i^M)}\sum_{n=0}^{N}\sum_{m=0}^{M}\binom{N}{n}\binom{M}{m}(-1)^{M-m}e^{i(2(n+m)-N-M)s}.$$

Noticing that $\binom{N}{n}=\binom{N}{N-n}$ and $\binom{M}{m}=\binom{M}{M-m}$ it's easy to see that:

$$\cos^N(s)\ \sin^M(s)=\begin{cases}\dfrac{(-1)^{M/2}}{2^{N+M}}\displaystyle\sum_{k=0}^{N+M}C_k^{N,M}\cos(2k-N-M)s & \text{if } M \text{ is even}\\[2ex] \dfrac{(-1)^{(M+1)/2}}{2^{N+M}}\displaystyle\sum_{k=0}^{N+M}C_k^{N,M}\sin(2k-N-M)s & \text{if } M \text{ is odd}\end{cases}$$

where coefficients $C_k^{N,M}$ are integers given by:

$$C_k^{N,M}=\sum_{n=0}^{N}\sum_{m=0}^{M}\binom{N}{n}\binom{M}{m}(-1)^m\delta_{k,n+m}.$$

In the case where the edge is a circle, the calculation of the integrals is done in a similar way to those of arc with $\alpha=0$ and $\beta=2\pi$.

Remark 7.1. The integral over an arc of circle of a monomial can also be approximated by subdividing the arc into several arcs sufficiently small so that these can be considered as straight line segments. Then we apply the formula of the integral of a monomial over a line segment as it is seen in Section 7.1.

Remark 7.2. The previous remark is also applicable for other types of edges that are neither line segments nor circular arcs.

8. Integral of a monomial over the boundary of a region

The coefficients of matrices A_r^+ and A_f^- require computations of integrals of polynomials over the incoming and the outgoing boundaries of a region D_r. The computation of these integrals leads to calculate the integrals of monomials X^iY^j over these boundaries. Thus, for an edge $\mathcal{A}$ of the boundary $\partial\mathsf{D}_r$ we have to calculate the integrals:

$$\mathcal{I}^+(i,n,m;\,j,\ell,k)=\int_{\mathcal{A}}X^iY^j\int_{(\omega\cdot\mathsf{n})>0}y_\ell^k y_n^m(\omega\cdot\mathsf{n})\,\mathrm{d}\omega\,\mathrm{d}s, \tag{40}$$

$$\mathcal{I}^-(i,n,m;\,j,\ell,k)=\int_{\mathcal{A}}X^iY^j\int_{(\omega\cdot\mathsf{n})<0}y_\ell^k y_n^m(\omega\cdot\mathsf{n})\,\mathrm{d}\omega\,\mathrm{d}s. \tag{41}$$

In the following subsections, we give the details to compute these integrals according to the shape of the edge $\mathcal{A}$: line segment or arc of circle.

8.1. Case where the edge is a line segment

In the case where the edge is a line segment, the surface integrals and the integrals over the angular direction, in Eqs. (40)-(41), are decoupled because the normal vector n is constant along the line segment:

$$\mathcal{I}^{\pm}(i,n,m;\, j,\ell,k) = \Big(\int_{\mathcal{A}} X^i Y^j \, \mathrm{d}s\Big)\Big(\int_{\pm(\omega\cdot\mathrm{n})>0} y_\ell^k y_n^m (\omega\cdot\mathrm{n})\, \mathrm{d}\omega\Big).$$

In this section, we give only the details for the calculations of the surface integrals, see Section 6 for integrals over the angular direction. Denote by $A(x_A, y_A)$ and $B(x_B, y_B)$ the endpoints of the line segment $\mathcal{A}$. By setting $\Delta_x = x_B - x_A$, $\Delta_y = y_B - y_A$ and $L = \sqrt{(\Delta_x)^2 + (\Delta_y)^2}$, we can write:

$$I = \int_{\mathcal{A}} X^i Y^j \, \mathrm{d}s = L \int_0^1 \left(\Delta_x s + x_A\right)^i \left(\Delta_y s + y_A\right)^j \mathrm{d}s.$$

Using again the binomial formula, we obtain:

$$I = L \sum_{i'=0}^{i} \binom{i}{i'} (x_A)^{i-i'} (\Delta_x)^{i'} \sum_{j'=0}^{j} \binom{j}{j'} (y_A)^{j-j'} (\Delta_y)^{j'} \int_0^1 s^{i'+j'} \, \mathrm{d}s.$$

Finally

$$I = L \sum_{i'=0}^{i} \binom{i}{i'} (x_A)^{i-i'} (\Delta_x)^{i'} \sum_{j'=0}^{j} \binom{j}{j'} (y_A)^{j-j'} (\Delta_y)^{j'} \frac{1}{i'+j'+1}. \tag{42}$$

8.2. Case where the edge is an arc of circle

In this section, we explain how the integrals (40) and (41) can be evaluated exactly in the case where $\mathcal{A}$ is an arc of a circle. But these integrals can also be approximated by applying Remark 7.1 and the previous section.

Consider the case where $\mathcal{A}$ is an arc defined by its radius ρ, its center $C(x_C, y_C)$ and its endpoints (A, B). Let α be the angle formed by the x-axis and the vector $\overrightarrow{CA}$ and β the angle formed by the x-axis and the vector $\overrightarrow{CB}$. The integrals (40) and (41) are then written:

$$\mathcal{I}^{+}(i,n,m;\, j,\ell,k) = \rho \int_\alpha^\beta a(s) \int_{\omega\cdot\mathrm{n}>0} y_\ell^k y_n^m (\omega\cdot\mathrm{n})\, \mathrm{d}\omega \, \mathrm{d}s,$$

$$\mathcal{I}^{-}(i,n,m;\, j,\ell,k) = \rho \int_\alpha^\beta a(s) \int_{\omega\cdot\mathrm{n}<0} y_\ell^k y_n^m (\omega\cdot\mathrm{n})\, \mathrm{d}\omega \, \mathrm{d}s,$$

where

$$a(s) = (x_C + \rho\cos s)^i (y_C + \rho \sin s)^j.$$

Since $\mathrm{n} = \kappa(\cos s, \sin s, 0)$ and $\omega = (\sin\theta\cos\varphi, \sin\theta\sin\varphi, \cos\theta)$, with $\kappa = 1$ if the region D_r is at the left of the arc when going from A to B in the trigonometric sense and $\kappa = -1$ otherwise, we obtain:

$$\begin{aligned}
\mathcal{I}^{\pm}(i,n,m;\, j,\ell,k) &= \frac{\rho}{4\pi} \int_\alpha^\beta a(s) \int_{\varphi_0^\pm(s)}^{\varphi_1^\pm(s)} \int_0^\pi y_\ell^k y_n^m (\omega\cdot\mathrm{n}) \sin\theta \, \mathrm{d}\theta \, \mathrm{d}\varphi \, \mathrm{d}s \\
&= \frac{\kappa\rho}{4\pi} \int_\alpha^\beta a(s) \cos s \int_{\varphi_0^\pm(s)}^{\varphi_1^\pm(s)} \cos\varphi \int_0^\pi y_\ell^k y_n^m \sin^2\theta \, \mathrm{d}\theta \, \mathrm{d}\varphi \, \mathrm{d}s \\
&\quad + \frac{\kappa\rho}{4\pi} \int_\alpha^\beta a(s) \sin s \int_{\varphi_0^\pm(s)}^{\varphi_1^\pm(s)} \sin\varphi \int_0^\pi y_\ell^k y_n^m \sin^2\theta \, \mathrm{d}\theta \, \mathrm{d}\varphi \, \mathrm{d}s
\end{aligned}$$

where φ_0^+, φ_1^+, φ_0^-, φ_1^- are given by:

$$\varphi_0^+(s) = s - \pi/2, \qquad \varphi_1^+(s) = s + \pi/2, \tag{43}$$

$$\varphi_0^-(s) = s + \pi/2, \qquad \varphi_1^-(s) = s + 3\pi/2. \tag{44}$$

Consider now the functions $\phi_x(\varphi)$ and $\phi_y(\varphi)$ defined by:

$$\phi_x(\varphi) = \cos\varphi \int_0^\pi y_n^m(\theta,\varphi) y_\ell^k(\theta,\varphi) \sin^2\theta \, \mathrm{d}\theta,$$

$$\phi_y(\varphi) = \sin\varphi \int_0^\pi y_n^m(\theta,\varphi) y_\ell^k(\theta,\varphi) \sin^2\theta \, \mathrm{d}\theta.$$

It's easy to see that $\phi_x \in \mathsf{E}$ and $\phi_y \in \mathsf{E}$, so their primitives Φ_x and Φ_y are written as:

$$\Phi_x(\varphi) = c_x\,\varphi + f_x(\varphi), \tag{45}$$

$$\Phi_y(\varphi) = c_y\,\varphi + f_y(\varphi), \tag{46}$$

where f_x and f_y are still functions of E, c_x and c_y being constants. So we have:

$$\begin{aligned} \mathcal{I}^\pm(i,n,m;\, j,\ell,k) = {} & \frac{\kappa\rho}{4\pi} \int_\alpha^\beta a(s)(\cos s)\left(\Phi_x(\varphi_1^\pm(s)) - \Phi_x(\varphi_0^\pm(s))\right) \mathrm{d}s \; + \\ & \frac{\kappa\rho}{4\pi} \int_\alpha^\beta a(s)(\sin s)\left(\Phi_y(\varphi_1^\pm(s)) - \Phi_y(\varphi_0^\pm(s))\right) \mathrm{d}s. \end{aligned} \tag{47}$$

Using (43), (44), (45) and (46), it is easy to see that the functions

$$\gamma_x^\pm(s) = \left(\Phi_x(\varphi_1^\pm(s)) - \Phi_x(\varphi_0^\pm(s))\right) \quad \text{and} \quad \gamma_y^\pm(s) = \left(\Phi_y(\varphi_1^\pm(s)) - \Phi_y(\varphi_0^\pm(s))\right)$$

are also functions of E. And since E is stable by multiplication, the functions

$$\left(a(s)(\cos s)(\Phi_x(\varphi_1^\pm(s)) - \Phi_x(\varphi_0^\pm(s))\right) \quad \text{and} \quad \left(a(s)(\sin s)(\Phi_y(\varphi_1^\pm(s)) - \Phi_y(\varphi_0^\pm(s))\right)$$

belong to E.

Thus the computation of the integrals $\mathcal{I}^\pm(i,n,m;\, j,\ell,k)$ is reduced to integrals of functions of E, things we know how to do. It is not easy to give an analytic expression for the integrals $\mathcal{I}^\pm(i,n,m;\, j,\ell,k)$, but their calculation is achievable by programming a computer to do the algebra.

Circles are treated in the same way as arcs with $\alpha = 0$ and $\beta = 2\pi$.

9. Integral of a monomial over a 2D extruded region

We shall briefly give here the main idea to compute the integral of a monomial over a 2D extruded region. To do this, assume that the region D_r is a vertical cylinder with horizontal bases:

$$\mathsf{D}_r = S \times [z_0, z_1],$$

where S is a flat horizontal surface of arbitrary shape. So

$$\begin{aligned} \int_{\mathsf{D}_r} X^i Y^j Z^k \, \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z &= \int_S \int_{z_0}^{z_1} X^i Y^j Z^k \, \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z \\ &= \frac{1}{k+1}(z_1^{k+1} - z_0^{k+1}) \int_S X^i Y^j \, \mathrm{d}x\, \mathrm{d}y. \end{aligned}$$

The integral $\int_S X^i Y^j \, \mathrm{d}x\, \mathrm{d}y$ can be computed as explained in the previous sections. Surface integrals can also be calculated without difficulty by distinguishing the lateral surfaces and the bases surfaces of the region.

10. Treatment of the reflecting boundary conditions

Let f be a perfectly reflecting surface of the boundary $\partial \mathsf{D}$ assumed to be a line segment, as is the case in real applications. Let $\mathsf{n}(\theta_{\mathsf{n}}, \varphi_{\mathsf{n}})$ denote the unit normal to the surface f oriented outside of the domain D. The reflected direction of $\omega(\theta, \varphi)$ by the surface f is $\omega^\star(\theta^\star, \varphi^\star)$ defined by

$$\omega^\star = \omega - 2(\omega \cdot \mathsf{n})\mathsf{n}. \tag{48}$$

The reflecting boundary conditions applied to the surface f is then written:

$$u(x, \omega) = u(x, \omega^\star), \qquad \text{for} \quad \omega \cdot \mathsf{n} < 0. \tag{49}$$

Thus the incoming flux in region D_r through the face f is given by:

$$I_{\mathsf{f}}^-(i,n,m) = \int_{\mathsf{f}} \int_{\omega\cdot\mathsf{n}<0} u(x,\omega^\star)\varphi_i(x) y_n^m(\omega)(\omega\cdot\mathsf{n})\, \mathrm{d}\omega\, \mathrm{d}s.$$

Using approximation (9)

$$I_{\mathsf{f}}^-(i,n,m) = \int_{\mathsf{f}} \int_{\omega\cdot\mathsf{n}<0} \left(\sum_{\ell,k,j} u_{\ell,j}^k y_\ell^k(\omega^\star)\varphi_j(x) \right) \varphi_i(x) y_n^m(\omega)(\omega\cdot\mathsf{n})\, \mathrm{d}\omega\, \mathrm{d}s,$$

thus

$$I_{\mathsf{f}}^-(i,n,m) = \sum_{\ell,k,j} A_{\mathsf{f}}^{-,\star}(i,n,m;\, j,\ell,k)\, u_{\ell,j}^k,$$

with

$$A_{\mathsf{f}}^{-,\star}(i,n,m;\, j,\ell,k) = \left(\int_{\mathsf{f}} \varphi_i\varphi_j\, \mathrm{d}s \right) \left(\int_{\omega\cdot\mathsf{n}<0} y_n^m(\omega) y_\ell^k(\omega^\star)(\omega\cdot\mathsf{n})\, \mathrm{d}\omega \right).$$

Using (48), it's easy to see that $\omega^\star = (\sin\theta^\star \cos\varphi^\star, \sin\theta^\star \sin\varphi^\star, \cos\theta^\star)$ with $\theta^\star = \theta$ and $\varphi^\star = \pi + 2\varphi_{\mathsf{n}} - \varphi$. Therefore the function $y_\ell^k(\omega^\star)$ is also a function of Y.

The computation of the coefficients of matrices $A_{\mathsf{f}}^{-,\star}$ is done in a similar way to the computation of the coefficients of A_{f}^- presented in Section 8.1. The matrices $A_{\mathsf{f}}^{-,\star}$ are not symmetric while the matrices A_{f}^- are.

11. Orthonormalization of a family of polynomials

The coefficients of matrices A_r^1 and F_r^1 require the computation of the integrals

$$\int_{\mathsf{D}_r} \varphi_i\varphi_j\, \mathrm{d}x\, \mathrm{d}y,$$

these coefficients become simpler if the polynomials family (φ_j) is orthonormalized on the region D_r with respect to the $L^2(\mathsf{D}_r)$-product, since the last integral is reduced to $\delta_{i,j}$. The use of orthonormalized basis of functions (φ_j) avoids the problem of obtaining a solution from an ill-conditioned matrix equation and also introduces the economy of avoiding the storage of the matrix A^1 and subsequent matrix-vector multiplication. Using an orthonormalized polynomials family (φ_j) in each region D_r is possible, but the family (φ_j) becomes necessarily dependent on the region D_r. Thus on both sides of an interface f the flux is developed on two different bases of polynomials. This changes the expression of the matrix A_{f}^-. In this section we recall how to orthonormalize a family of polynomials in a region and we revise the expression of the matrix A_{f}^- accordingly.

Consider a family of linearly independent polynomials $(\phi_j)_{j=1,\cdots,J}$. We recall here the Gram-Schmidt process for building a family of polynomials $(\varphi_j)_{j=1,\cdots,J}$ orthonormalized on a region D_r spanning the same space as the family (ϕ_j).

Assuming that the polynomials φ_j for $j = 1, \cdots, i$ have already been built, the construction of the polynomial φ_{i+1} is done by setting:

$$\widehat{\varphi}_{i+1} = \phi_{i+1} - \sum_{k=1}^{i} < \phi_{i+1}, \varphi_k >_r \varphi_k,$$

with

$$< \phi_{i+1}, \varphi_k >_r = \int_{\mathsf{D}_r} \phi_{i+1}(x)\, \varphi_k(x)\, \mathrm{d}x,$$

it is easy to check that $\widehat{\varphi}_{i+1}$ is orthogonal to all polynomials φ_j for $j = 1, \cdots, i$. φ_{i+1} is then obtained by normalization of $\widehat{\varphi}_{i+1}$:

$$\varphi_{i+1} = \frac{\widehat{\varphi}_{i+1}}{||\widehat{\varphi}_{i+1}||}.$$

The Gram-Schmidt algorithm requires only the calculation of the integrals $< \phi_{i+1}, \varphi_k >_r$, which we know how to do as seen in the previous sections.

Suppose now that we have two families of polynomials $(\varphi_j)_{j=1,\cdots,J}$ and $(\widetilde{\varphi}_j)_{j=1,\cdots,J}$ spanning the same space with (φ_j) orthonormalized in a region D_r and $(\widetilde{\varphi}_j)$ orthonormalized in a region $\mathsf{D}_{\widetilde{r}}$ adjacent to the region D_r. We can write:

$$\widetilde{\varphi}_i = \sum_{j=1}^{J} < \widetilde{\varphi}_i, \varphi_j >_r \varphi_j.$$

The flux in the region $\mathsf{D}_{\widetilde{r}}$ is written:

$$\widetilde{u}^g(x,\omega) = \sum_{\ell,k,j} \widetilde{u}^{k,g}_{\ell,j}\, \widetilde{\varphi}_j(x) y^k_\ell(\omega), \tag{50}$$

thus

$$\widetilde{u}^g(x,\omega) = \sum_{\ell,k,j} \widetilde{u}^{k,g}_{\ell,j} \Big(\sum_{j'=1}^{J} < \widetilde{\varphi}_j, \varphi_{j'} >_r \varphi_{j'}\Big) y^k_\ell. \tag{51}$$

Let us rewrite the expression of $L^g_-(\varphi_i y^m_n)$ where the basis of polynomials (φ_j) in the region D_r and the basis $(\widetilde{\varphi}_j)$ in its adjacent regions are different.

$$\begin{aligned} L^g_-(\varphi_i y^m_n) &= \int_{\Gamma_-} \widetilde{u}^g(x,\omega) \varphi_i y^m_n\, (\omega \cdot \mathsf{n})\, \mathrm{d}\omega \mathrm{d}s \\ &= \int_{\Gamma_-} \left(\sum_{\ell,k,j} \widetilde{u}^{k,g}_{\ell,j} \Big(\sum_{j'=1}^{J} < \widetilde{\varphi}_j, \varphi_{j'} >_r \varphi_{j'}\Big) y^k_\ell \right) \varphi_i y^m_n\, (\omega \cdot \mathsf{n})\, \mathrm{d}\omega \mathrm{d}s. \end{aligned}$$

Thus by writing $\int_{\Gamma_-} \cdot\, \mathrm{d}\omega\, \mathrm{d}s = \sum_{\mathsf{f} \in \partial \mathsf{D}_r} \int_{\mathsf{f}} \int_{(\omega\cdot\mathsf{n})<0} \cdot\, \mathrm{d}\omega\, \mathrm{d}s$:

$$L^g_-(\varphi_i y^m_n) = \sum_{\mathsf{f} \in \partial \mathsf{D}_r} \sum_{\ell,k,j} \widetilde{A}^-_{\mathsf{f}}(i,n,m;\, j,\ell,k) \widetilde{u}^{k,g}_{\ell,j},$$

with

$$\widetilde{A}^-_{\mathsf{f}}(i,n,m;\, j,\ell,k) = \sum_{j'=1}^{J} < \widetilde{\varphi}_j, \varphi_{j'} >_r \int_{\mathsf{f}} \varphi_i \varphi_{j'} \int_{(\omega\cdot\mathsf{n})<0} y^m_n y^k_\ell\, (\omega \cdot \mathsf{n})\, \mathrm{d}\omega \mathrm{d}s,$$

thus

$$\widetilde{A}^-_{\mathsf{f}}(i,n,m;\, j,\ell,k) = \sum_{j'=1}^{J} < \widetilde{\varphi}_j, \varphi_{j'} >_r A^-_{\mathsf{f}}(i,n,m;\, j',\ell,k).$$

12. Description of the mesh in NYMO-DG

The 2D mesh in NYMO-DG is first described by a cloud of numbered points given by their coordinates (x, y). This cloud of points is constituted by the endpoints of the edges and the centers of arcs and circles intervening in the mesh. The mesh is then described by the edges separating two regions of calculation. Each edge is defined according to its shape, line segment, arc of circle or circle:

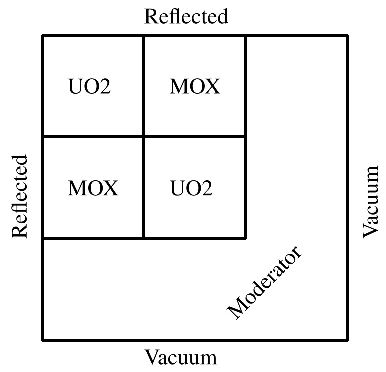

Fig. 2. C5G7 geometry.

- A line segment is defined by the numbers of its two endpoints (A, B) and the numbers of its two neighbors regions (r_L, r_R), where r_L is the number of the left region of the line segment when going from A to B. When the line segment is a part of the boundary ∂D, the line segment has only one neighbor region and in this case we assign the null value to the superfluous region, i.e. $r_L = 0$ or $r_R = 0$ depending on the case.
- An arc is defined by its radius ρ, the number of its center C, the numbers of its two endpoints (A, B) and the numbers of its two neighbors regions (r_L, r_R), where r_L is the left region to the arc when going from A to B in the counter-clockwise.
- A circle is defined by its radius ρ, the number of its center C and the numbers of its two neighbors regions (r_L, r_R), where r_L is the region lying inside the circle.

The computation of matrices coefficients A_r^0, A_r^1, A_r^2, F_r^1, F_r^2, A_r^+ and A_f^- is done by sweeping over all edges of the mesh and for each edge we calculate its contribution to the matrix coefficients for its two neighbors regions r_L and r_R.

13. Choice of polynomials in NYMO-DG

In NYMO-DG we use an orthonormalized polynomials family $(\varphi_j)_{1\le j\le J}$ in each region D_r. This family is constructed by orthonormalization of a family of translated monomials $(\phi_j = (X - x_0)^n (Y - y_0)^m)$, by setting up an one-to-one correspondence between the pair (n, m) and j. The point (x_0, y_0) is a local origin of region D_r. In NYMO, the choice of the local origin (x_0, y_0) of the region D_r is the barycenter of the points involved in the definition of the region: endpoints of edges and centers of arcs. The use of translated monomials allows for distant regions and identical up to an arbitrary translation to have identical elementary matrices.

When working with translated monomials, the integrals (38), (39), (42) and (47) remain valid by replacing x_A, y_A, x_C and y_C per $(x_A - x_0)$, $(y_A - y_0)$, $(x_C - x_0)$ and $(y_C - y_0)$ respectively.

The monomials family (ϕ_j) is specified by giving the highest degree d of its monomials as well as the mode of this degree: total $(n + m \le d)$ or partial $(n \le d$ and $m \le d)$.

14. C5G7 benchmark

In this section we present numerical results for the C5G7 benchmark in its 2D version, see [22]. This benchmark was proposed to provide a point of comparison for assessing the capabilities of transport codes to deal with reactor core problems without spatial homogenization. The geometry of the benchmark consists of 4×4 assemblies surrounded by moderator, see Fig. 2 where only a quarter of the geometry is presented for reasons of symmetry. Internal assemblies (MOX/UO2) are 17×17 cells. The cells comprise two material zones representing different fuels, guide tubes and instrumented tubes. The benchmark specifies the macroscopic cross-sections for seven energy groups and with isotropic collision.

The geometry of the cells is presented in Fig. 3. The side length of each pin cell is 1.26 cm and all of the fuel pins and guides tubes have 0.54 cm radius. The mesh of cells used in the calculations is given in Fig. 4. The calculations are done in eighth of the geometry, Fig. 5 provides the mesh of computation.

The authors of the benchmark provided the k-effective and the overall pin power distribution resulting from the Monte-Carlo transport code MCNP. This power distribution is used as a reference result for comparison with the results obtained by the NYMO code. The results of the NYMO code, presented in this section, use piecewise linear polynomials in space. The authors also specified a number of results to be compared to those of MCNP. The Tables 1 to 6 give these comparisons with NYMO code for different P_N orders but with the same spatial mesh.

Table 1 gives the k-effective solution for different P_N orders and the error relative in pcm compared to MCNP code.

The results of the k-effective in Table 1 do not give a convergence as desired when the order P_N grows. This is due in part to the fact that P_N calculations use the same spatial mesh, whereas for a convergence study one has to go up in P_N

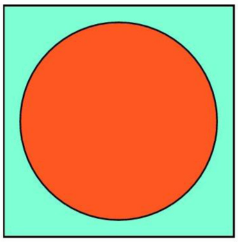

Fig. 3. Cell geometry.

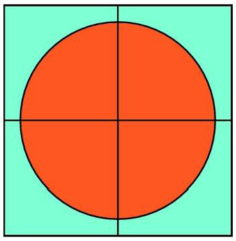

Fig. 4. Cell mesh.

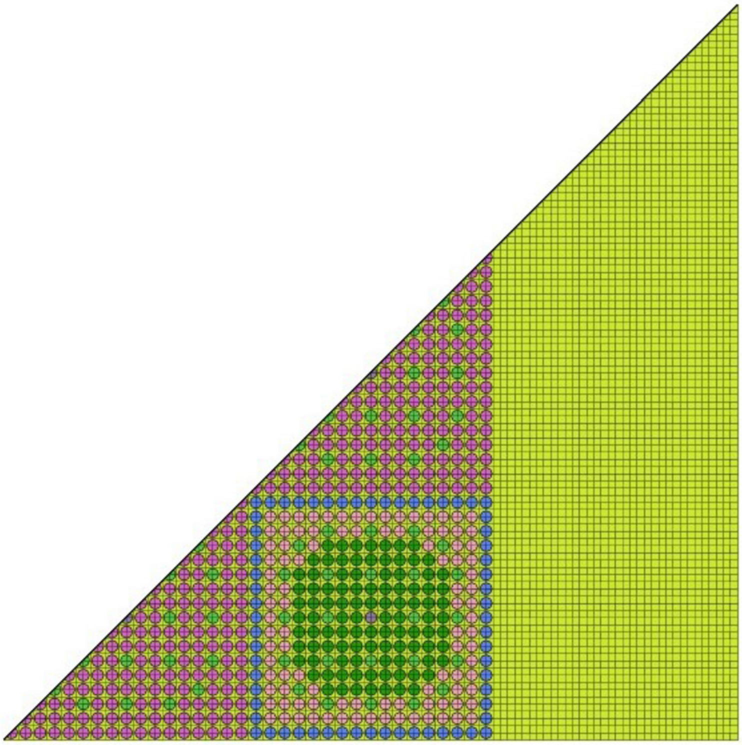

Fig. 5. NYMO mesh.

order and refine the spatial mesh simultaneously. This phenomenon deserves to be analyzed on problems with Cartesian meshes so that the spatial refinement is easy to realize. Nevertheless, when we take a close look at Table 2, the convergence on the power distribution seems acceptable. Difference of the results between P_N approximations with even and odd orders remains unexplained.

The P_N approximation is often used with an odd expansion order. It is established for the P_N equations resulting from the first-order transport equation that odd orders give better results, see Davison's book [9], Section 10.3.2. The superiority of odd orders over even is not true in the absolute for all P_N approximations, this depends on the variational formulation used, on the spatial approximations adopted and in particular on the approximations at the interfaces. The P_N method developed

Table 1
Keff resulting from NYMO code.

order P_N	keff	pcm error	cpu time (s)
P_2	1.18672	14	71
P_4	1.18684	24	226
P_6	1.18704	41	575
P_1	1.18500	−131	37
P_3	1.18517	−116	130
P_5	1.18527	−108	385
MCNP	1.18655	±8	

Table 2
Percent error results for specific pin powers.

order P_N	Max pin power	error (%)	Min pin power	error (%)	Max error(%)
P_2	2.477	−0.80	0.233	0.69	1.32
P_4	2.487	−0.41	0.232	0.32	1.08
P_6	2.498	−0.41	0.232	0.27	1.15
P_1	2.515	0.71	0.233	0.68	2.98
P_3	2.511	0.54	0.232	0.26	1.75
P_5	2.502	0.18	0.232	0.38	1.25
MCNP	2.498	±0.16	0.232	±0.58	

Table 3
Assembly power percent errors.

order P_N	Inner UO2	error (%)	MOX	error (%)	Outer UO2	error (%)
P_2	490.5	−0.46	212.4	0.33	140.6	0.61
P_4	491.3	−0.30	212.1	0.20	140.4	0.44
P_6	491.3	−0.30	212.2	0.22	140.4	0.43
P_1	495.1	0.48	210.8	−0.43	139.3	−0.36
P_3	494.2	0.29	210.8	−0.42	140.1	−0.25
P_5	493.2	0.09	211.3	−0.21	140.2	0.34
MCNP	492.8	±0.10	211.7	±0.18	139.8	±0.20

here use both odd and even orders indifferently. There are no theoretical results, for our method, favoring or discrediting odd order approximations over those of even order. Other authors [8] use the P_N approximation based on other variational formulation, have shown that their method can be used with both even and odd orders. See also [12] where the authors combine the P_N method with the least-squares method, in this reference, all numerical results are given only with even orders.

Table 2 gives the relative error in percent of the extrema of pin power distribution compared to the results given by MCNP. The positions of the pins achieving the maximum and the minimum of the pin power distribution coincide between MCNP and NYMO calculations.

Table 3 gives the relative error in percent per assembly power versus to the results given by MCNP.

Table 4 give the differences between MCNP and NYMO results for overall pin power distribution. These differences are calculated in three different ways called AVG, RMS and MRE:

$$AVG = \frac{1}{N}\sum_{n=1}^{N} |e_n|,$$

$$RMS = \sqrt{\frac{1}{N}\sum_{n=1}^{N} e_n^2},$$

$$MRE = \frac{1}{N \cdot p_{avg}}\sum_{n=1}^{N} |e_n| \cdot p_n.$$

See document [22] for the meanings of these deviations. N being the number of fuel pin, e_n is the relative error in percent of the calculated power in the nth fuel pin compared to the reference result. p_n is the reference pin power in the nth fuel pin. $p_{avg} = \frac{1}{N}\sum_{n=1}^{N} p_n$ is the average power.

Table 4
Pin power distribution error.

order P_N	AVG	RMS	MRE
P_2	0.52	0.60	0.50
P_4	0.34	0.42	0.33
P_6	0.35	0.44	0.34
P_1	0.90	1.11	0.76
P_3	0.42	0.53	0.37
P_5	0.31	0.40	0.25
MCNP	0.32	0.34	0.27

Table 5
Number of fuel pins within the reference confidence intervals.

order P_N	68%	90%	98%	99.8%
P_2	734	910	968	972
P_4	928	1038	1050	1052
P_6	894	1022	1044	1047
P_1	480	595	648	669
P_3	843	947	986	993
P_5	939	1030	1041	1041

Table 6
Percentage of fuel pins within the reference confidence intervals.

order P_N	68%	90%	98%	99.8%
P_2	69.5	86.2	91.7	92.0
P_4	87.9	98.3	99.4	99.6
P_6	84.7	96.8	98.9	99.1
P_1	45.5	56.3	61.4	63.4
P_3	79.8	89.7	93.4	94.0
P_5	88.9	97.5	98.6	98.6

Table 5 gives statistics of the number of fuel pins having a relative error in percent compared to MCNP less than 0.68, 0.90, 0.98 and 0.998. And Table 6 gives the same statistics but in percent. The total number of fuel pins is 1056 for the quarter of the geometry.

15. Conclusion

In this work, a new numerical scheme solving the transport equation is described in detail for 2D geometries. The method has the ability to deal with non-standard meshes (unstructured, non-conformal and curved), such meshes are considered for realistic calculations only by the method of characteristics MOC. The method presented is not limited neither in P_N order nor in degrees of polynomials. The coefficients of the elementary matrices resulting from the adopted approximations are evaluated exactly for regions of arbitrary shape.

The numerical results presented for the C5G7 benchmark are very satisfying for the precision of computation and the computation time. The method deserves to be studied more and especially for the choice of a preconditioner with an acceptable storage cost. Optimizing of the performance of the method by parallelization and acceleration using a good preconditioner would make the method very competitive. It would also be interesting to implement the method for 2D extruded geometries.

The motivation for the numerical scheme presented was firstly the consideration of complex geometries with regions having curved boundaries like those used in heterogeneous reactor calculations. The method is quite general so that it can also deal with other problems as shielding or source detector calculations. This of course remains to be demonstrated with appropriate test cases. However, the method as it is presented here cannot handle cases with void regions, a possible way to extend the method for void regions can be found in the work done in [17].

Acknowledgements

The author wishes to thank Richard Sanchez, Igor Zmijarevic, Gérald Samba and Fausto Malvagi for fruitful discussions. This work is partially supported by EDF and Framatome within the common project for the development of the Apollo3 code.

Appendix A. Spherical harmonics

In this appendix, we recall the definition of real-valued spherical harmonics y_n^m.

A.1. The Legendre polynomials

The Legendre polynomials $P_n(\mu)$, of degree n, are defined by Rodrigue's formula:

$$P_0(\mu) = 1,$$
$$P_n(\mu) = \frac{1}{2^n\, n!}\,\frac{d^n}{d\mu^n}\,(\mu^2-1)^n, \qquad n \ge 1.$$

They satisfy the following properties, see [18]:

$$\int_{-1}^{1} P_n(\mu)P_\ell(\mu)\,\mathrm{d}\mu = \frac{2}{2n+1}\delta_{n,\ell}, \qquad \forall n,\ell \ge 0,$$
$$P_n(-\mu) = (-1)^n P_n(\mu), \qquad \forall n \ge 0,$$
$$(2n+1)\mu P_n(\mu) = (n+1)P_{n+1}(\mu) + nP_{n-1}(\mu), \qquad \forall n \ge 1.$$

The last equation is a recursive formula on Legendre's polynomials, it can serve as an alternative definition for Legendre's polynomials with $P_0(\mu) = 1$ and $P_1(\mu) = \mu$.

A.2. Associated Legendre functions

The associated Legendre functions $P_n^m(\mu)$, of degree n and order m, are defined by:

$$P_n^0(\mu) = P_n(\mu), \qquad n \ge 0,$$
$$P_n^m(\mu) = (-1)^m(1-\mu^2)^{m/2}\,\frac{d^m}{d\mu^m}\,P_n(\mu), \qquad n \ge 0;\ 1 \le m \le n,$$

They satisfy the following properties, see [18]:

$$\int_{-1}^{1} P_n^m(\mu)P_\ell^m(\mu)\,\mathrm{d}\mu = \frac{2}{C_n^m}\delta_{n,\ell}, \qquad \forall n,m,\ell \ge 0,$$
$$P_n^m(-\mu) = (-1)^{n+m}P_n^m(\mu), \qquad \forall n,m \ge 0,$$

where,

$$C_n^m = (2n+1)\frac{(n-m)!}{(n+m)!}.$$

On the other hand, we have the recurrence formula, for $n \ge 1$ and $0 \le m \le n$:

$$(2n+1)\,\mu P_n^m(\mu) = (n-m+1)P_{n+1}^m(\mu) + (n+m)P_{n-1}^m(\mu).$$

A.3. Spherical harmonics

The real spherical harmonics $y_n^m(\omega)$, of degree n and order m, are finally defined by, see [2] page 315:

$$y_n^0(\omega) = (2n+1)^{1/2}\ P_n(\cos\theta), \qquad m = 0,$$
$$y_n^m(\omega) = (2C_n^m)^{1/2}\ P_n^m(\cos\theta)\ \cos(m\varphi), \qquad m = 1,\cdots,n,$$
$$y_n^{-m}(\omega) = (2C_n^m)^{1/2}\ P_n^m(\cos\theta)\ \sin(m\varphi), \qquad m = 1,\cdots,n,$$

where (θ,φ) are the polar coordinates of the direction ω. θ is the angle formed by ω and the z-axis, φ is the angle formed by the orthogonal projection of ω on the xy-plane and the x-axis, thus $\omega = (\sin\theta\cos\varphi, \sin\theta\sin\varphi, \cos\theta)$. The spherical harmonics y_n^m constitute a complete basis of the space $L^2(\mathrm{S}^2)$ of square integrable real functions. We finally have the following properties:

$$\int_{\mathrm{S}^2} y_n^m(\omega) y_\ell^k(\omega)\,\mathrm{d}\omega = \delta_{n,\ell}\,\delta_{m,k}, \tag{A.1}$$

$$P_n(\omega\cdot\omega') = \frac{1}{2n+1}\sum_{m=-n}^{n} y_n^m(\omega) y_n^m(\omega'), \tag{A.2}$$

$$y_n^m(-\omega) = (-1)^n y_n^m(\omega). \tag{A.3}$$

Relation (A.1) expresses the orthonormalization of the spherical harmonics, relation (A.2) is the addition formula and relation (A.3) expresses that the parity in ω of the harmonic spherical y_n^m is that of n. A proof of the addition formula (A.2) is presented in [18], page 128.

References

[1] Ron T. Ackroyd, Finite Element Methods for Particle Transport: Applications to Reactor and Radiation Physics, John Wiley & Sons Limited, 1997.
[2] Kendall Atkinson, Weimin Han, Theoretical Numerical Analysis. A Functional Analysis Framework, Springer, 2005.
[3] Andrew Scott Bielen, Spherical harmonics solutions to second order forms of the Boltzmann transport equation using particle transport code SCEPTRE, Thesis Master of Science, Pennsylvania State University. Department of Mechanical and Nuclear Engineering, 2008.
[4] L. Bourhrara, New variational formulations for the neutron transport equation, Transp. Theory Stat. Phys. 33 (2004) 93–124.
[5] L. Bourhrara, H^1 approximations of the neutron transport equation and associated diffusion equations, Transp. Theory Stat. Phys. 35 (2006) 89–108.
[6] L. Bourhrara, W^{N} approximations of neutron transport equation, Transp. Theory Stat. Phys. 38 (2009) 195–227.
[7] C.B. Carrico, E.E. Lewis, G. Palmiotti, Three dimensional variational nodal transport methods for cartesian, triangular, and hexagonal criticality calculations, Nucl. Sci. Eng. 111 (1992) 168.
[8] S. Van Criekingen, E.E. Lewis, R. Beauwens, Mixed-hybrid transport discretization using even and odd $\mathrm{P_N}$ expansions, Nucl. Sci. Eng. 152 (2) (2006) 149–163.
[9] B. Davison, Neutron Transport Theory, The International Series of Monographs on Physics, Oxford at the Clarendon Press, London, 1957.
[10] C.R.E. De Oliveira, An arbitrary geometry finite element method for multigroup neutron transport with anisotropic scattering, Prog. Nucl. Energy 28 (1/2) (1986) 227–236.
[11] D.A. Di Pietro, A. Ern, Mathematical Aspects of Discontinuous Galerkin Methods, Springer, 2012.
[12] C. Drumm, W. Fan, A. Bielen, J. Chenhall, Least squares finite elements algorithms in the SCEPTRE radiation transport code, in: International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M&C 2011, Rio de Janeiro, RJ, Brazil, May 8-12, 2011, Latin American Section (LAS)/American Nuclear Society ANS, 2011, on CD-ROM.
[13] Clifton Russell Drumm, Spherical harmonics ($\mathrm{P_N}$) methods in the SCEPTRE radiation transport code, No. SAND2015-2113C, Sandia National Lab (SNL-NM), Albuquerque, NM (United States), in: ANS MC2015 - Joint International Conference on Mathematics and Computation (M&C), Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC) Method, Nashville, TN, April 19-23, 2015.
[14] C.J. Gesh, Finite Element Methods for Second Order Forms of the Transport Equation, Thesis, Texas A&M University, 1999.
[15] S.E. Keller, C.R.E. De Oliveira, Two-dimensional c5g7 mox fuel assembly benchmark calculations using the FEM-PN code EVENT, Prog. Nucl. Energy 45 (2–4) (2004) 255–263.
[16] V.M. Laboure, R.G. McClarren, C.D. Hauck, Implicit filtered P_N for high-energy density thermal radiation transport using discontinuous Galerkin finite elements, J. Comput. Phys. 321 (2016) 624–643.
[17] V.M. Laboure, R.G. McClarren, Yaqi Wang, Globally conservative, hybrid self-adjoint angular flux and least-squares method compatible with voids, Nucl. Sci. Eng. 185 (2) (2017) 294–306.
[18] T.M. MacRobert, I.N. Sneddon, Spherical Harmonics: An Elementary Treatise on Harmonic Functions, with Applications, 3rd rev. ed., Pergamon Press, Oxford, England, 1967.
[19] J.E. Morel, J.M. McGhee, A self-adjoint angular flux equation, Nucl. Sci. Eng. 108 (1999) 247–266.
[20] G. Palmiotti, E.E. Lewis, C.B. Carrico VARIANT, Variational Anisotropic Nodal Transport for Multidimensional Cartesian and Hexagonal Geometry Calculations, Technical Report ANL-95/40, Argone National Laboratory, 1995.
[21] D. Schneider, F. Dolci, F. Gabriel, J.-M. Palau, et al., APOLLO3®: CEA/DEN deterministic multipurpose platform for core physics analysis, in: Proc. of Int. Mtg. on the Physics of Fuel Cycles and Advanced Nuclear Systems, PHYSOR 2016, Sun Valley, USA, 2016.
[22] M.A. Smith, E.E. Lewis, B.-C. Na, Benchmark on Deterministic Transport Calculations Without Spatial Homogenization. A 2-D/3-D MOX Fuel Assembly Benchmark, OECD/NEA report, NEA/NSC/DOC(2003)16, ISBN 92-64-02139-6, 2003.