Verification of the Current Coupling Collision Probability Method with Orthogonal Flux Expansion for the Case of Single Cell

Dzianis Litskevich[1], Bruno Merk, Seddon Atkinson

d.litskevich@liverpool.ac.uk
b.merk@liverpool.ac.uk
Seddon.Atkinson@liverpool.ac.uk

University of Liverpool, School of Engineering, Department of Mech, Materials and Aerospace Engineering , Liverpool, Merseyside L69 3BX

Abstract

Within this article, current coupling collision probability method with expansion of flux by orthogonal polynomials is verified and validated via comparison with analytical and Monte-Carlo solutions for the single cell with reflective boundary conditions. The results of the calculations for a single unit cell demonstrate good agreement with the results of Monte Carlo calculations. The comparison of the new method with the state-of-the-art flat flux approximation demonstrates either an improved quality of the result for identical cell discretisation or significantly increased computational efficiency to achieve an identical solution. Application of the numerical methods for evaluation of the coefficients of orthogonal polynomials increase the flexibility of the method significantly since higher orders of the polynomials for the neutron source/flux expansion can be applied now for much wider range of the 2D problems. Application of the orthogonal polynomials for the flux expansion allows a significant reduction of the number of calculation regions, while maintaining or improving the accuracy of the results. It also improves efficiency (in terms of the accuracy/run time ratio) of the calculations. Finally, it has a potential to relieve the user from determining the correct discretisation of the calculation regions, which is required in the case of the traditional flat flux approximation and can be quite challenging for the complex geometries.

Keywords: nuclear reactor simulation, neutron transport calculations, CCCP, orthogonal polynomials

1 Introduction

The operation of nuclear reactors requires detailed knowledge of important safety parameters, such as the spatial power distribution, the control rod worth, margin to departure from nucleate boiling (DNB), pin burnup etc. The widely used standard approach in design and safety calculations are coupled neutronics and thermal-hydraulics codes for the steady state and transient simulation. The neutronics calculations are typically performed at a nodal level using the diffusion approximation and assembly-homogenized sets of cross-sections while the thermal hydraulics relies on a channel model with fuel assembly sized channels. However, for the determination of the safety limits, which are based on local pin-based parameters, the knowledge of the power and temperature distribution on a nodal level is not sufficient. Local thermal hydraulic and neutronics safety

[1] Corresponding author. E-mail: d.litskevich@liverpool.ac.uk

parameters need to be predicted for the hottest fuel assembly in detail, down to the individual channels (hot channel analysis) and pins. To achieve this level of detail, local pin power distribution is necessary.

For practical reactor core applications, low-order transport approximations such as diffusion or SP_3 methods have been implemented on the nodal level for static and transient calculations with considerably less computational expense than full transport methods. However, using these approximations to produce a finer grid on pin level fails to provide accurate resolution of the heterogeneous fuel pin arrangements [1].

In nuclear reactors, there are several locations where the effect of neighbouring assemblies plays an important role and cannot be ignored such as the interface between fuel assemblies and reflector, the fuel assembly with control rods, and the interface between MOX and uranium fuel assemblies [2, 3].

Therefore, in recent years a number of projects were started focused on advanced multiphysical simulation of the nuclear reactors such as CASL [4] in the USA or DRD [5] in the UK. Interest has also increased in the development of the neutron transport codes which can solve the neutron transport equation for the full reactor core [6, 7, 8, 9], as they currently are mainly used to provide the two-dimensional cross section preparation. The straight forward extension of the methods used for cross-section preparation to full core analysis is problematic since the full-core transport solvers are very computationally expensive and additional cross section requirements significantly increase these costs. Full core computations using transport solvers require, as a rule, a significantly powerful computer cluster. Transient calculations are even more expensive and can require several weeks of cluster time for calculation of few seconds of transient behaviour [2].

To overcome the above limitations, other methodologies for pin-wise neutron flux calculations can be chosen. These methodologies can be combined in the frame of a multi-scale and multi-methodological approach. Pin-wise calculations, in this case, are performed by applying a transport solver using the heterogeneous fuel assembly geometry on an unstructured mesh with boundary conditions extracted from the 3D full core nodal diffusion solution. This combined nodal-transport approach was proposed in the 1970s by different researchers - Wagner, Koebke, Grill, Jonsson [8, 9, 10]. Further investigations in this field have been performed by Nissen [11]. The overall conclusion is that the combined nodal-transport approach is capable of providing high accuracy results with good computational performance.

To follow this strategy, one needs a transport solver, which can be used for the flux reconstruction on the pin level. The current coupling collision probability (CCCP) method [12, 13] seems to be a good choice for the development of such a solver. CCCP has several advantages which are important in the case of the pin power reconstruction: it is proven to be fast (HELIOS code [12]) and it allows a very detailed description of the boundary currents entering and leaving the assembly under consideration which is important for the task of the pin power reconstruction.

One of the drawbacks of the traditional CCCP method is its poor representation of the spatial distribution of the within group neutron flux/source. Traditionally, the CCCP method uses so-called flat flux approximation for the representation of the neutron sources within computational regions. Therefore, for the optically thick regions such as water reflectors, the detailed discretisation should be used to accurately represent the shape of the neutron source. Similar situation is observed for the thermal groups of the fuel. Thus, both fuel and reflector regions should be additionally discretised to capture the spatial effects of the neutron source. Unfortunately, the computational time of any type of the collision probability based methods is proportional to the second order of the number of

calculation regions. It means that doubling of the computational regions leads to the four time higher computational times. Therefore, the traditional collision probability methods become less computationally efficient as the number of the flat source regions increase. In addition, the spatial subdivision of the calculation regions can be complicated for the geometrically complex regions and the spatial convergence of the sources/fluxes cannot be guaranteed.

In order to overcome these issues, another approach can be used for better representation of the spatial distribution of the sources within calculation regions. For example, a linear source approximation was developed and tested for the method of characteristics [14] [15]. In this approach, the source is assumed to vary linearly over a given track's segment. It was demonstrated that the "linear source approximation improves the accuracy and the efficiency of the MOC calculation relative to the flat source scheme for a comparable level of accuracy" [15].

Similar approach can be used for the CCCP method. The flux/source within calculation regions can be expanded by orthogonal polynomials of two variables. This allows us to reduce the number of the calculation regions while maintaining the accuracy on the good level. This technique was previously tested for the case of the regular hexagonal lattice [1, 16]. In the presented work, this methodology was extended for the case of the unstructured mesh. In the case of the regular hexagonal lattice, the coefficients for the orthogonal polynomials can be calculated analytically, while in the case of the unstructured mesh, the corresponding coefficients should be evaluated numerically. To achieve this, the Gramm-Schmidt procedure combined with a triangulation algorithm is applied. The comparison of this new method to the flat flux approximation demonstrates either an improved quality of the results for identical cell discretisation or significantly increased computational efficiency to achieve a comparable accuracy.

In present study, verification of the developed solver is performed for the case of the single reactor cell. The results are compared with the reference Monte-Carlo simulations. Additional verification of the solver is performed using visualisation technique.

The paper is organised as follows. In section 2 the methodology is described. In section 3 implementation of the methodology is discussed. In section 4 results of the numerical experiments and visualisation of the flux's shape in the computational regions are presented, the performance of the proposed methodology is discussed. Finally, conclusions and remarks are given in section 5.

2 Methodology description

In traditional CCCP methods, neutron flux inside each calculation region is represented by only one value [12]. This approximation is called the flat-flux approximation. Therefore, the subdivision of each zone is essential [17] to achieve a non-flat neutron flux distribution which is required for an accurate result. The subdivision of each region leads to an increase in the number of elementary volumes (calculation regions). Consequently, this causes an increase in the required calculation time as well as in the memory demand. The calculation time for the collision probability method is proportional to N^2 where N is the number of regions [18].

An advanced approach for a more accurate representation of the flux inside the calculation regions has recently been proposed [16, 1]. According to this approach, the non-uniformity of the neutron flux distribution inside the calculation regions can be described with the help of the orthogonal polynomials of two variables. The general idea of the approach is very much comparable to the nodal methods applied to full core simulations, where a mathematical approximation represents the neutron flux distribution inside each of the calculation regions and nodes are coupled

via interface currents. As previously mentioned, this approach allows a significant increase of the size of the calculation regions without penalty in the calculation accuracy.

Mathematically, the expansion of the flux by orthogonal polynomials can be written in the following way:

$$\Phi\left(x,y\right) = \sum_{k=0}^{\infty} \Phi_k P_k(x,y)\,, \tag{1}$$

where $\Phi(x,y)$ is the neutron flux at the point with the coordinates x and y, Φ_k are the expansion coefficients, $P_k(x,y)$ are orthogonal polynomials. In the real case, the number of polynomials is limited by N members (in contrast to the exact representation where $N \to \infty$).

After that, the above flux expansion is introduced into the equations for the current coupling collision probability method instead of conventional flat flux approximation. The final set of equations for the current coupling collision probability method with orthogonal flux expansion involves now (in contrast to the flat flux approximation) additional terms related to the spatial moments of the flux. The detailed description of the methodology can be found in the works [1, 19, 20]. In the mentioned works, the expansion of the flux was applied for the case of the regular hexagonal cells. In the current study, the methodology was extended and tested for the case of the cell with arbitrary polygons with inscribed circles.

In order to expand the neutron flux by orthogonal polynomials, the polynomials should be orthogonalized within given calculation region. Let a be the set of the base monomials functions. In this case, the set of the monomials up to the n-th order will be written in the following way:

$$a = 1,x,y,x^2,xy,y^2,\ldots,x^{n-q}y^q,\ldots,y^n \qquad q = 0,\ldots,n \tag{2}$$

The i-th orthogonal polynomial can be written as follows:

$$P_i = \sum_{k=0}^{i} b_{ik} a_k\,, \tag{3}$$

where b_k is the coefficient, a_k is k-th monomial from the set of monomials.

Equation (3) can be written in the matrix form for the n first base polynomials:

$$\mathbf{P} = \mathbf{B}\mathbf{A}, \tag{4}$$

where $\mathbf{P} = (P_1,P_2,P_3,\ldots,P_n)$ is $n \times 1$ column vector containing resulting orthogonal polynomials; $\mathbf{A} = (a_0,a_1,a_2,\ldots,a_n)$ is $n \times 1$ column vector containing base functions (monomials) and $\mathbf{B}$ is matrix containing orthogonal coefficients:

$$\mathbf{B} = \begin{pmatrix} b_{1,0} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ b_{n,0} & \cdots & b_{n,n} \end{pmatrix} \tag{5}$$

$\mathbf{B}$ is lower triangular matrix.

Therefore, the task of the polynomial orthogonalisation can be formulated as finding the elements of the $\mathbf{B}$ matrix. The orthogonalisation process can be performed either analytically or numerically. Analytical orthogonalisation provides exact expression for the calculation of the $\mathbf{B}$ matrix. However, practically, it is complicated to evaluate analytically orthogonal coefficients for arbitrary geometries. Therefore, in the current study, numerical evaluation of the orthogonal coefficients was used.

3 Implementation of the methodology

The incorporation of the orthogonal polynomials into the conventional current coupling collision probability method [12] leads to equations that are more complicated for integration. The equations for the evaluation of the region-to-surface, surface-to-region, region-to-region and surface-to-surface collision and transmission probabilities become more complicated. The probabilities depend now (in contrast to traditional CCCP probabilities) not only on the discretized regions and angles but also on the spatial modes of fluxes. However, since neutron flux is represented now as expansion by the orthogonal polynomials (in contrast to the conventional CCCP method where it is represented by one number), it allows to describe the shape of the flux within calculation region more accurate. Consequently, expansions of the flux using orthogonal polynomials have the potential either to decrease the number of calculation regions significantly, while maintaining the same accuracy, or to improve the calculation accuracy significantly for an identical number of calculation regions.

3.1 Current Coupling Collision Probability Method

As in any kind of the collision probability method, the solution of the neutron transport equation starts from the evaluation of the collision probabilities. In the case of the CCCP method, the collision probabilities are unique for each type of the cell in the assembly (for example fuel cell or absorber cell). There are four types of the collision probabilities which should be evaluated: region-to-region (RR) collision probability, region-to-surface (RS) escape probability, surface-to-region (SR) collision probability and surface-to-surface (SS) transmission probability. In order to account for the angular and spatial dependency of the neutron currents, the directional hemisphere is subdivided into the number of the azimuthal and polar sectors while the cell's sides are subdivided on two or more segments. The example of the angle and spatial discretisation is shown in Figure 1.

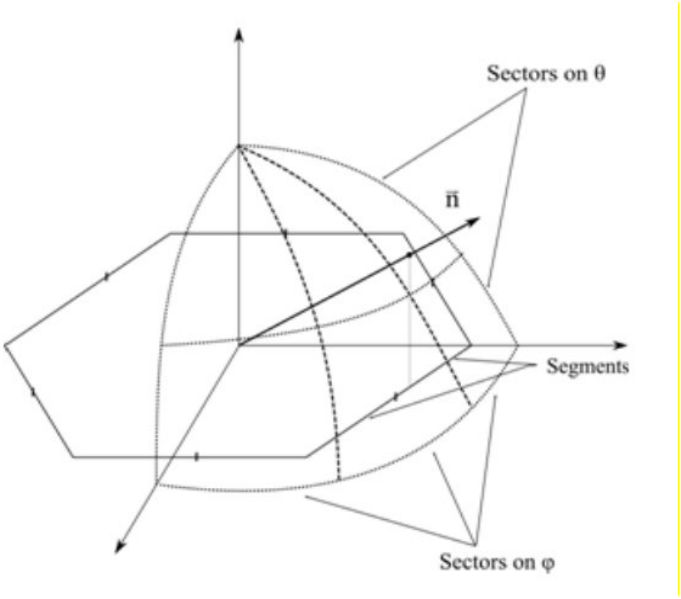

Figure 1. Example of the angular and spatial discretisation

The neutron transport equation in the classical current coupling collision probability method is reduced to a set of linear algebraic equations. Collision probabilities are evaluated by accounting for the angular and spatial discretization of the model [21]. Using notation for the angle, side and region discretization, the equations describing the relations between neutron currents and fluxes for the current CCCP with the standard flat-flux approximation can be written in the following way:

$$v_i \Sigma_i \Phi_i = \sum_{i'} v_{i'} Q_{i'} RR_{i' \to i} + \sum_j \sum_m \sum_l J^{in}_{j,m,l} SR_{j,m,l \to i} \tag{6}$$

$$J^{out}_{j,m,l} = \sum_{i'} v_{i'} Q_{i'} RS_{i' \to j,m,l} + \sum_{j'} \sum_{m'} \sum_l J^{in}_{j',m',l} SS_{j',m',l \to j,m,l}, \tag{7}$$

where $SS_{j',m',l \to j,m,l}$ is the probability for a neutron entering the j'-th segment, m'-th azimuthal sector and l-th polar sector leaving without collision through the j-th segment, m-th azimuthal sector and the l-th polar sector; $SR_{j,m,l \to i}$ is the probability for a neutron entering into the j'-th segment via the m'-th azimuthal sector and the l-th polar sector to have its first collision in the i-th region; $RS_{i' \to j,m,l}$ is the probability for neutron born in the i'-th region to leave without collision through the j-th segment, m-th azimuthal sector and the l-th polar sector; $RR_{i' \to i}$ is the probability for neutron born in the region i' to have its first collision in region i; Φ_i is the neutron flux in the region i; Q_i is the neutron source in the region i; v_i – volume of the region i; Σ_i is the total cross section in the region i; $J^{in}_{j,m,l}$ is the incoming current on the segment j', azimuthal sector m' and polar sector l; $J^{out}_{j,m,l}$ is the outgoing current on the segment j, azimuthal sector m and polar sector l.

3.2 Equations for the CCCP with Orthogonal Flux Expansion

The expansion of the neutron flux using orthogonal polynomials slightly changes the form of Equations (6) and (7). These can now be written in the following form:

$$v_i \Sigma_i \Phi_{i,k} = \sum_{i'} \sum_{k'} v_{i'} Q_{i',k'} RR_{i',k' \to i,k} + \sum_{j} \sum_{m} \sum_{l} J^{in}_{j,m,l} SR_{j,m,l \to i,k} \tag{8}$$

$$J^{out}_{j,m,l} = \sum_{i'} \sum_{k'} v_{i'} Q_{i',k'} RS_{i',k' \to j,m,l} + \sum_{j'} \sum_{m'} \sum_{l} J^{in}_{j',m',l} SS_{j',m',l \to j,m,l'} \tag{9}$$

where $Q_{i',k'}$ is the neutron source in the region i', spatial mode k'; $\Phi_{i,k}$ is the neutron flux in the region i, spatial mode k.

The equations for the collision probabilities are now given in the following way:

$$SS_{j',m',l \to j,m,l} = \frac{1}{\varepsilon_j \chi_m \mu_l} \int_{\varphi_{m-1}}^{\varphi_m} cos\varphi d\varphi \int_{L_{j-1}}^{L_j} Ki_{3,l}(\tau)\, dL \tag{10}$$

$$RS_{i,k \to j,m,l} = \frac{2}{\pi v_i} \int_{\varphi_{m-1}}^{\varphi_m} cos\varphi d\varphi \int_{L_{j-1}}^{L_j} dL \int_{\Delta t_i} Ki_{2,l}(\tau)\, P_k(x,y)\, dt \tag{11}$$

$$RR_{i',k' \to i,k} = \frac{\Sigma_{i'}}{\pi v_{i'}} \int_{0}^{\frac{\pi}{2}} cos\varphi d\varphi \int_{L_{j-1}}^{L_j} dL \int_{0}^{\Delta t_{i'}} P_{k'}(x',y')\, dt' \int_{0}^{\Delta t_i} Ki_1(\tau)\, P_k(x,y)\, dt \tag{12}$$

$$SR_{j,m,l \to i,k} = \frac{1}{\chi_m \mu_l} \frac{4 v_i}{\Sigma_i L_j} RS_{i,k \to j,m,l'} \tag{13}$$

where $RR_{i',k' \to i,k}$ is the probability for a neutron born in region i', harmonic k' to have its first collision in region i, spatial mode k; $SR_{j,m,l \to i,k}$ is the probability for a neutron entering into the segment j' via the azimuthal sector m' and polar sector l to have its first collision in region i, spatial mode k; $RS_{i',k' \to j,m,l}$ is the probability for a neutron born in the region i' and spatial mode k' to leave without collision through the segment j, azimuthal sector m and polar sector l; L_j is the length of j-th segment; $\varepsilon_j, \chi_m, \mu_l$ are normalization indexes for the segment, azimuthal angle and polar angle correspondingly; τ is the optical path; φ, t, t', L are integration variables; $P_k(x,y)$ is the orthogonal polynomial of the k-th order; $Ki_{n,l}$ is the partial Bickley function (for definition see, for example, [12])

The collision probabilities (10) - (13) are integrated using standard ray-tracing methodology. More detailed information about the procedure of the integration using ray-tracing technique can be found in [12, 22]. The surface to region collision probability is not integrated but calculated using Equation (13).

As in each kind of the collision probability method, reciprocity relations exist between different collision probabilities. These relations help the evaluation of different types of probabilities. Several relations for collision probabilities can be written in addition to the Equation (13).

As a particle, entering the cell through one on space element must either collide in one of the regions or escape through one of its segment and sector, the following relation between SS and SR probabilities can be formulated for the zeroth spatial moment.

$$\sum_{j'}\sum_{m'} SS_{j',m',l\to j,m,l} = 1 - \sum_{i} SR_{j,m,l\to i,0} \tag{14}$$

As a particle born in region i must either collide in another (or the same) region or escape through one of cell's side, segment and sector, the following relation between RS and RR can be written for zeroth moment of flux.

$$\sum_{j'}\sum_{m'}\sum_{l} RS_{i,0\to j,m,l} = 1 - \sum_{i'} RR_{i,0\to i',0} \tag{15}$$

Finally, relation between different elements of the RR matrix can be formulated.

$$v_{i'}\Sigma_{i'}RR_{i',k'\to i,k} = v_i\Sigma_i RR_{i,k\to i',k'}, \tag{16}$$

In addition to the reduction in the calculation efforts, Equations (13) – (16) are used for the normalization of the collision probabilities, which in turn increase the stability of the iteration process. The equations (8) and (9) contain the additional summation index k, which reflects the expansion of the flux through the orthogonal polynomials. In contrast to the flat-flux approximation, the flux in each region is now represented by a set of spatial moments of the flux $\Phi_{i,k}$. The meaning of the flux in each location inside the region of the interest can be evaluated with the help of the set $\Phi_{i,k}$ obtained and Equation (1).

3.3 Gram-Schmidt Orthogonalization Procedure

In order to evaluate the coefficients for the orthogonal polynomials, ($b_{i,k}$ in the Equation (1)**Error! Reference source not found.**) for arbitrary calculation regions, the so-called Gram-Schmidt process [23] can be applied. The Gram-Schmidt process is a procedure, which takes a non-orthogonal set of linearly independent functions and constructs an orthogonal (or orthonormal if required) basis over an arbitrary interval (or domain in the case of multidimensional functions). The implementation

of the process for orthogonalization of the base function set will be shown using a set of non-orthogonal functions depending on two variables, x and y:

$$a_1 = a_1(x,y), a_2 = a_2(x,y), a_3 = a_3(x,y) \ldots a_n = a_n(x,y) \tag{17}$$

A set of orthogonal functions $h_n(x,y)$ should be constructed over domain D using functions a as a basis. To solve the problem, the inner product of the functions $f = f(x,y)$ and $g = g(x,y)$ over the domain D can be defined in the following way:

$$(f,g) = \iint_D f(x,y)\, g(x,y)\, dxdy \tag{18}$$

When the inner product of the functions is defined, the projection operator can be introduced in the following way:

$$proj_h a = \frac{(a,h)}{(h,h)} h = \frac{\iint_D a(x,y)\, h(x,y)\, dxdy}{\iint_D h(x,y)\, h(x,y)\, dxdy} h(x,y) \tag{19}$$

Following this operation, the Gram-Schmidt orthogonalization procedure is given by:

$$\begin{gathered} h_1 = a_1 \\ h_2 = a_2 - proj_{h_1} a_2 \\ h_3 = a_3 - proj_{h_1} a_3 - proj_{h_2} a_3 \\ \ldots \\ h_n = a_n - \sum_{j=1}^{n-1} proj_{h_j} a_n \end{gathered} \tag{20}$$

where $h_n(x,y)$ are orthogonal functions.

The orthogonal function $h_n(x,y)$ can be represented as sum of the non-orthogonal functions $a(x,y)$, multiplied be certain coefficients c_{nj}, that is:

$$h_n = \sum_{j=1}^{n} c_{nj} a_j \tag{21}$$

Thus, for the representation of the orthogonal function using Equation (21), the matrix with the coefficients c_{ij} must be defined.

c_{ij} can be evaluated with the help of the Gram-Schmidt procedure. At the beginning of the orthogonalization procedure the following equation can be written for the first function (see Equation (20)):

$$h_1 = c_{11} a_1 \tag{22}$$

With coefficient c_{11} chosen to be equal to unity.

For the second orthogonal function h_2 the following equation can be written as:

$$h_2 = a_2 - \frac{(a_2, h_1)}{(h_1, h_1)} b_1 = a_2 c_{22} - \frac{(a_2, h_1)}{(h_1, h_1)} c_{11} a_1 = c_{22} a_2 + c_{21} a_1 \tag{23}$$

With coefficients c_{21} and c_{22} equal to:

$$\begin{aligned} c_{22} &= 1 \\ c_{21} &= - c_{11} \frac{(a_2, h_1)}{(h_1, h_1)} \end{aligned} \tag{24}$$

The continuation of the procedure leads to the general equation for the coefficients c_{nm}:

$$c_{nm} = \begin{cases} -\sum_{j=n}^{m-1} c_{nj} \frac{(a_m, h_j)}{(h_j, h_j)}, & if\ n < m \\ 1, & if\ n = m \\ 0, & if\ n > m \end{cases} \tag{25}$$

It can be seen from Equation (25) that the matrix containing the coefficients c_{nm} is lower triangular. Finally, it should be noted that the coefficients c_{nm} can be normalized to develop an orthonormal set of functions, using the following expression:

$$c_{ij}^{norm} = \frac{c_{ij}}{(h_j, h_j)} \tag{26}$$

3.3 Evaluation of the Coefficients for Orthogonal Polynomials

The integrals in Equations (25) and (26) can be evaluated analytically. This is trivial in the case of regular hexagonal cells or square cells. However, the analytical evaluation of the coefficients of the

orthogonal polynomials is not a perfect choice from a practical point of view since it causes problems for more complex geometries. Therefore, a more general methodology should be implemented for the evaluation of the orthogonal coefficients. The idea of the methodology which was used in the present study will be demonstrated on an arbitrary convex polygon with inscribed annular regions (see Figure 2).

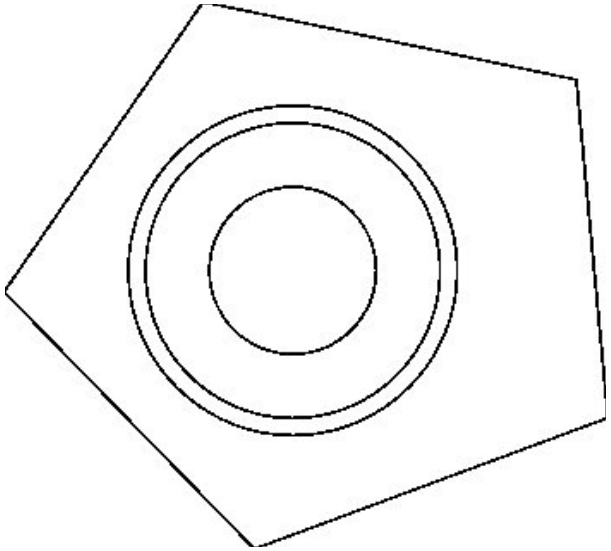

Figure 2. Example of a single cell (an arbitrary outer convex polygon with annular regions inside)

In the current study the order of the polynomials for the flux expansion was limited by the second order of the polynomials. In this case, the set of the monomials (basis functions) functions defined by Equation (2) can be written in the following way:

$$P_0 = 1, P_1 = x, P_2 = y, P_3 = x^2, P_4 = xy, P_5 = y^2 \tag{27}$$

The set of polynomials presented in the (27) should be orthogonalized over each calculation region inside the cell. To achieve this, the following integrals are used (see Equation (25)):

$$\iint_D P_i(x,y)\, P_k(x,y)\, dxdy \tag{28}$$

The polynomials $P_i(x,y)$ and $P_k(x,y)$ in Equation (28) can be both monomials and orthogonal polynomials. Integral over domain D in Equation (28) can be evaluated numerically. In the case of cells with an arbitrary outer polygon and inscribed annular zones (as presented in Figure 2), the evaluation of the integrals for each zone can be separated into two steps. In the first step, the integrals inside the annular zones are evaluated. This can be easily performed after a transformation of the coordinates from a Cartesian to a polar coordinate system. In this case

$$x = r\cos\alpha \quad y = r\sin\alpha. \tag{29}$$

The Jacobian of the transformation from a Cartesian to a polar coordinate system is equal to r. Thus integral (28) for the ring zone will take the form:

$$\iint_{D_{ring}} P_i(x,y)\, P_k(x,y)\, dxdy = \int_0^{2\pi} d\alpha \int_{R_1}^{R_2} rP_i(rcos\alpha,rsin\alpha)\, P_k(rcos\alpha,rsin\alpha)\, dr\,, \qquad (30)$$

where r and α are the variables of integration, R_1 is the inner radius of the ring and R_2 is the outer radius. For the innermost zone the condition $R_1 = 0$ is applied. The right side of Equation (30) can be evaluated numerically using the Gauss quadrature:

$$\int_0^{2\pi} d\alpha \int_{R_1}^{R_2} rP_i(rcos\alpha,rsin\alpha)\, P_k(rcos\alpha,rsin\alpha)\, dr = \frac{\pi(R_2-R_1)}{2} \sum_{k=1}^{N_G} \sum_{l=1}^{N_G} \omega_k \omega_l r_l P_i(r_l cos\alpha_k, r_l sin\alpha_k)\, P_j(r_l cos\alpha_k, r_l sin\alpha_k)\,, \qquad (31)$$

where N_G is the number of Gauss points and α_k and r_l are defined by the following equation:

$$\alpha_k = \pi\xi_k + \pi r_l = \frac{R_2 - R_1}{2}\xi_l + \frac{R_2 + R_1}{2}. \qquad (32)$$

In Equation (32) ξ_k and ξ_l are the nodes of the Gauss quadrature evaluated for the interval $(-1,1)$. After the evaluation of the integrals over the rings, the integral over the last (outermost) calculation region consisting of the arbitrary polygon (outer border) and a circle (inner border) is evaluated. The following technique can be used to evaluate the necessary integrals numerically. The resulting integral over the polygon-circle region can be represented as the difference between the integral over the polygon and the integral over the outermost circle:

$$\iint_{D_{out}} P_i(x,y)\, P_k(x,y)\, dxdy = \iint_{D_{pol}} P_i(x,y)\, P_k(x,y)\, dxdy - \iint_{D_{cir}} P_i(x,y)\, P_k(x,y)\, dxdy\,. \qquad (33)$$

The second integral (over the circle) on the right side of Equation (33) can be evaluated using Equation (31). The evaluation of the integral over the polygon requires a different approach. First, the outer polygon can be subdivided into several triangles as shown in Figure 3.

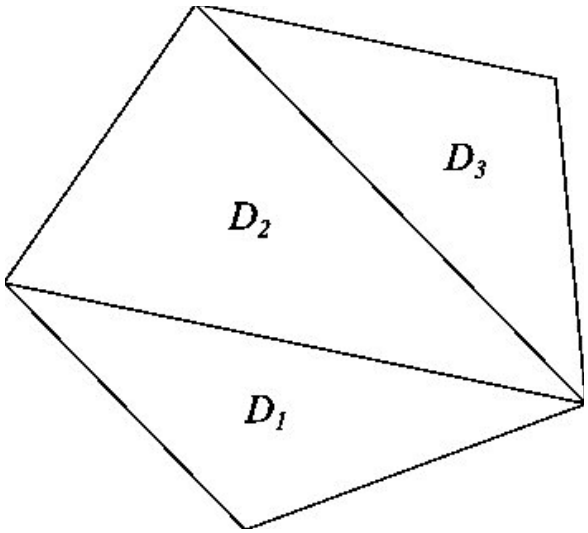

Figure 3. Example of the outer polygon triangulation

The integral over the polygon can be represented as sum of the integrals over the triangles D_1, D_2 and D_3:

$$\begin{aligned}&\iint_{D_{pol}} P_i(x,y)\,P_k(x,y)\,dxdy\\ &\iint_{D_1} P_i(x,y)\,P_k(x,y)\,dxdy+\iint_{D_2} P_i(x,y)\,P_k(x,y)\,dxdy\\ &+\iint_{D_3} P_i(x,y)\,P_k(x,y)\,dxdy\,.\end{aligned} \tag{34}$$

Since the functions $P_i(x,y)$ and $P_k(x,y)$ are polynomials of different orders, the integrals over the triangles can be evaluated numerically using, for example, the method described in [24]. The final integral over the polygon is calculated as a sum of the integrals over each triangle. It should be noted that since triangulation procedure is well understood, and standard algorithms are available, the methodology presented here can be considered as universal for arbitrary convex polygons with inscribed annular regions. In the current version of the program, the Fortran 90 implementation of the Delaunay triangulation available in the GEOMPACK package [25] was used for triangulation of the outer polygon.

4 Verification of the orthogonal coefficients

As it was noted in the previous section, in the current version of the program, orthogonal coefficients $b_{i,j}$ are evaluated numerically for each calculation region. At the same time, the opportunity of the analytical evaluation of the orthogonal coefficients for regular polygons with inscribed circles exists. Therefore, numerically and analytically calculated coefficients can be

compared to evaluate the quality of the numerical algorithms used in the current study.

In order to do that, the orthogonal coefficients were evaluated analytically for the regular square and hexagon with inscribed circles. The geometry of the cells as well as numbering of the regions are presented in Figure 4.

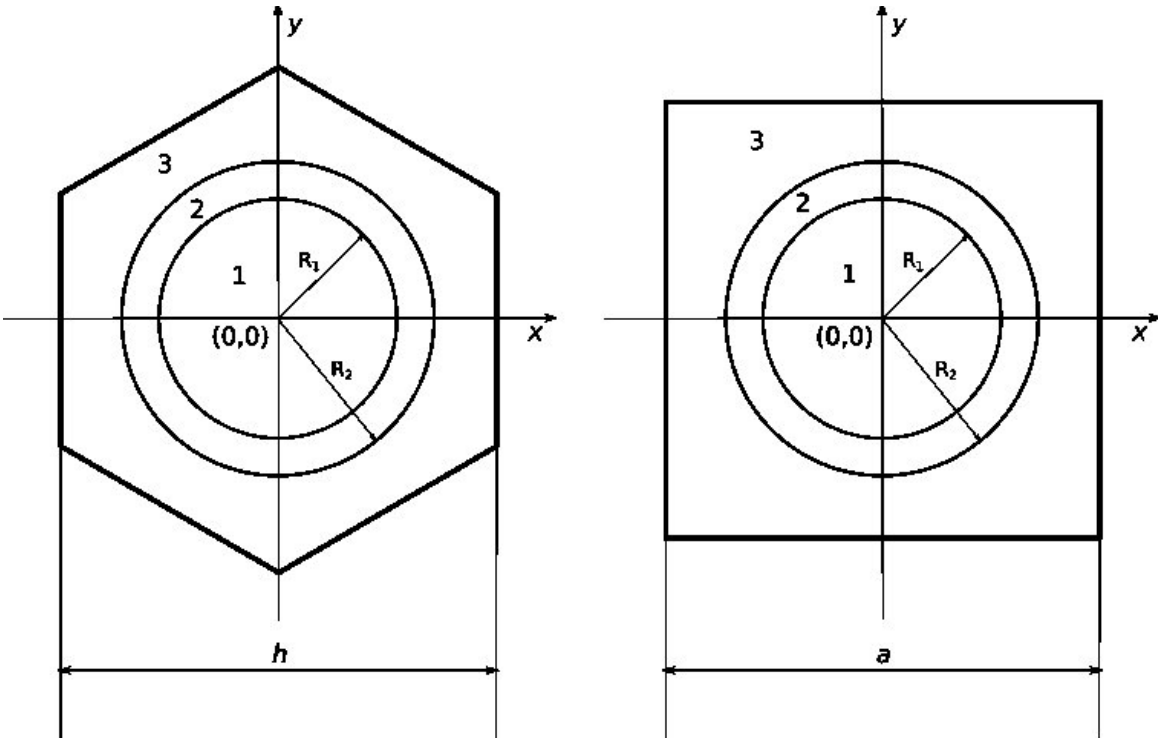

Figure 4. Geometries of the regular cells used for the verification of the orthogonal coefficients

Both square and hexagonal cells consist of three regions. The orthogonal coefficients were evaluated analytically using Gramm-Schmidt process and possibilities of the Sympy library [26] for Python. The evaluation was performed for the six monomials given below:

$$1,x,y,x^2,xy,y^2 \tag{2}$$

Resulting matrix $\mathbf{B}$ of the orthogonal coefficient for any region both for square and hexagonal cells have the following form:

$$\mathbf{B} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ b_{3,0} & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ b_{5,0} & 0 & 0 & b_{5,3} & 0 & 1 \end{pmatrix}. \tag{3}$$

As can be seen from Equation (3), the matrix is lower triangular with the coefficients equal to 0 everywhere except main diagonal and coefficients $b_{3,0}$, $b_{5,0}$ and $b_{5,3}$. The expressions for the analytical evaluations of the non-zero coefficients are identical for the circle regions for square and hexagonal cells and defined only by the radii of the internal and external circles R_1 and R_2

correspondingly.

$$b_{3,0} = -\frac{R_1^2 + R_2^2}{4} \tag{4}$$

$$b_{5,0} = -\frac{1}{3\left(R_1^4 + R_2^4\right)}(R_1^6 + 2R_1^4R_2^2 + 2R_1^2R_2^4 + R_2^6) \tag{5}$$

$$b_{5,3} = \frac{R_1^4 + 4R_1^2R_2^2 + R_2^4}{3R_1^4 + 3R_2^4}. \tag{6}$$

In the case of the innermost region (marked in Figure 4 by number 1) $R_1 = 0$.

For the outermost region (number 3 in Figure 4), analytical expressions for the coefficients will be different for square and hexagonal cells.

For the square cell, coefficients are given by Equations (6 – 8):

$$b_{3,0} = -\frac{3\pi R_2^4 - a^4}{12(\pi R_2^2 - a^2)} \tag{7}$$

$$b_{5,0} = -\frac{45\pi^2 R_2^{10} - 15\pi R_2^6 a^4 - 3\pi R_2^4 a^6 + a^{10}}{3(45\pi^2 R_2^8 - 90\pi R_2^6 a^2 + 30\pi R_2^4 a^4 - 9\pi R_2^2 a^6 + 4a^8} \tag{8}$$

$$b_{5,3} = \frac{5\pi R_2^2(3\pi R_2^6 + 6R_2^4 a^2 - 6R_2^2 a^4 + a^6)}{45\pi^2 R^8 - 90\pi R_2^6 a^2 + 30\pi R_2^4 a^4 - 9\pi R_2^2 a^6 + 4a^8}. \tag{9}$$

For the outermost region of the regular hexagonal cell, coefficients are given by Equations (9 – 11):

$$b_{3,0} = -\frac{3\pi R_2^4 - a^4}{12(\pi R_2^2 - a^2)} \tag{10}$$

$$b_{5,0} = -\frac{A_1}{D_1} \tag{11}$$

$$b_{5,3} = \frac{A_2}{D_2}, \tag{12}$$

where A_1, B_1, A_2 and B_2 are given by the following expressions:

$$\begin{aligned} A_1 \\ = (-36\pi R_2^4 + 5\sqrt{3}h^4)(-2880\pi^5 R_2^{14} + 5760\sqrt{3}\pi^4 R_2^{12}h^2 - 12960\pi^3 R_2^{10}h^4 \\ + 112\sqrt{3}\pi^4 R_2^8 h^6 + 4320\sqrt{3}\pi^2 R_2^8 h^6 - 672\pi^3 R_2^6 h^8 - 1620\pi R_2^6 h^8 \end{aligned} \tag{13}$$

$$+504\sqrt{3}\pi^2R_2^4h^{10}-504\pi R_2^2h^{12}+63\sqrt{3}h^{14})$$

$$D_1$$

$$\begin{aligned}&=9\left(-2\pi R_2^2+\sqrt{3}h^2\right)^2(8640\pi^4R_2^{12}-17280\sqrt{3}\pi^3R_2^{10}h^2+2400\sqrt{3}\pi^3R_2^8h^4\\&+32400\pi^2R_2^8h^4-7200\pi^2R_2^6h^6-672\sqrt{3}\pi^3R_2^6h^6-6480\sqrt{3}\pi R_2^6h^6\\&+1800\sqrt{3}\pi R_2^4h^8+2524\pi^2R_2^4h^8-1012\sqrt{3}\pi R_2^2h^{10}+381h^{12})\end{aligned}\tag{14}$$

$$A_2$$

$$\begin{aligned}&=\left(-8\pi^3R_2^6+12\sqrt{3}\pi^2R_2^4h^2-18\pi R_2^2h^4+3\sqrt{3}h^6\right)\left(720\pi^2R_2^8+720\sqrt{3}\pi R_2^6h^2\right.\\&\left.-600\sqrt{3}\pi R_2^4h^4+56\sqrt{3}\pi R_2^2h^6+41h^8\right)\end{aligned}\tag{15}$$

$$D_2$$

$$\begin{aligned}&=-2\pi R_2^2+\sqrt{3}h^2)(8640\pi^4R_2^{12}-17280\sqrt{3}\pi^3R_2^{10}h^2+2400\sqrt{3}\pi^3R_2^8h^4\\&+32400\pi^2R_2^8h^4-7200\pi^2R_2^6h^6-672\sqrt{3}\pi^3R_2^6h^6-6480\sqrt{3}\pi R_2^6h^6\\&+1800\sqrt{3}\pi R_2^4h^8+2524\pi^2R_2^4h^8-1012\sqrt{3}\pi R_2^2h^{10}+381h^{12}.\end{aligned}\tag{16}$$

Another way to evaluate nonzero coefficients of the matrix is to use numerical techniques for integration as well as for Gramm-Schmidt algorithm as it was shown, for example, in the work [19]. The methodology and algorithms described in mentioned work were applied for numerical evaluation of the coefficients. The results of the comparison are presented in Table 1.

Coefficient	Square cell			Hexagonal cell		
	Numerical	Analytical	Difference	Numerical	Analytical	Difference
Region 1						
$b_{3,0}$	-0.37249E-01	-0.37249E-01	-0.18373E-16	-0.37249E-01	-0.37249E-01	-0.18373E-16
$b_{5,0}$	-0.49665E-01	-0.49665E-01	0.93463E-16	-0.49665E-01	-0.49665E-01	0.93463E-16
$b_{5,3}$	0.33333E+00	0.33333E+00	-0.70314E-15	0.33333E+00	0.33333E+00	-0.70314E-15
Region 2						
$b_{3,0}$	-0.89005E-01	-0.89005E-01	-0.13623E-15	-0.89005E-01	-0.89005E-01	-0.13623E-15
$b_{5,0}$	-0.17494E+00	-0.17494E+00	-0.91625E-15	-0.17494E+00	-0.17494E+00	-0.91625E-15
$b_{5,3}$	0.96549E+00	0.96549E+00	0.43061E-15	0.96549E+00	0.96549E+00	0.43061E-15
Region 3						
$b_{3,0}$	-0.18819E+00	-0.18819E+00	-0.57229E-11	-0.29510E+00	-0.29510E+00	0.11556E-07
$b_{5,0}$	-0.28082E+00	-0.28082E+00	-0.88486E-10	-0.46932E+00	-0.46932E+00	0.46559E-07

$b_{5,3}$	0.49220E+00	0.49220E+00	0.20831E-09	0.59038E+00	0.59038E+00	-0.11070E-06

Table 1. Numerically and analytically evaluated coefficients of the orthogonal polynomials

It should be noted that analytical evaluation of the coefficients can be continued for the monomials higher than 2-nd order. However, due to the complexity of the resulting expressions (see, for example, equations for evaluation of the $b_{5,0}$ coefficient for the hexagonal cell), the number of the base functions was limited by 6 first monomials in this study.

As can be seen from Table 1, numerically and analytically evaluated coefficients agrees very well for regions 1 and 2 (circle regions) both for hexagonal and square cells. Maximal absolute difference between them does not exceed $\sim 0.7 \cdot 10^{-15}$. The situation changes for the outermost region bounded externally by polygon. In this region maximal difference between numerically and analytically evaluated coefficients rises up to the $\sim 0.2 \cdot 10^{-9}$ and $\sim 0.1 \cdot 10^{-6}$ for the square and hexagonal cells correspondingly.

The difference observed for the outermost region can be significant due to the error accumulation in the numerical simulations. Therefore, additional check was performed to evaluate the orthogonality of the polynomials calculated numerically. In order to do that, matrix B was orthonormalised. After that, the scalar product of the numerically orthonormalised polynomials was calculated. The results of the calculations for the outermost region both for square and hexagonal cells are presented in Table 2.

Order of the orthonormalised polynomials		Absolute difference from 0 or 1	
		Square	Hexagon
1	1	0.15543122E-14	0.00000000E+00
2	1	-0.10927739E-14	-0.19405831E-16
2	2	-0.66613381E-15	0.00000000E+00
3	1	-0.10794897E-15	-0.60378707E-16
3	2	-0.95930851E-17	0.86207651E-17
3	3	0.88817842E-15	0.22204460E-15
4	1	-0.33306691E-15	-0.99920072E-15
4	2	0.51012992E-15	0.13349606E-15
4	3	0.48787184E-16	0.11736111E-15
4	4	0.00000000E+00	-0.77715612E-15
5	1	-0.73972069E-17	0.10776896E-16
5	2	-0.80508376E-16	0.23570253E-16
5	3	-0.14147377E-15	0.95045266E-16
5	4	-0.96305105E-17	0.11185046E-16
5	5	0.22204460E-15	-0.22204460E-15
6	1	-0.13322676E-14	-0.21094237E-14
6	2	0.20861162E-14	0.51982296E-15
6	3	-0.16270945E-15	0.19128116E-15
6	4	-0.77715612E-14	0.13322676E-14
6	5	-0.43775304E-16	-0.83811575E-16

6	6	-0.22204460E-14	0.00000000E+00

Table 2. Difference from 0 or 1 for the numerically evaluated orthogonal polynomials (outermost region)

As it can be expected, the error rises with the rise of the polynomial orders. Highest errors are observed for the 6th order of the polynomials both for square and hexagonal cells. Nevertheless, the absolute values of the errors are small ($\sim 10^{-14}$ both for hexagonal and square cells). Therefore, on our view, this level of the accuracy is acceptable for the practical evaluation of the orthogonal coefficients numerically. Despite the fact, that the error of the normalisation stays low for the base functions considered in present work (monomials up to the 2nd order), additional investigations of the methodology should be performed for the monomials of the higher orders. Some initial tests were performed in the current study. Monomials up to the 4th order were checked. The normalisation error stayed on the acceptable level ($\sim 10^{-13}$) for the polynomials up to the 3rd order, but for the 4th order it increased rapidly and reached unacceptable level $\sim 10^{-1}$. This result was predictable due to the well-known unstable nature of the Gramm-Schmidt orthogonalisation process. Therefore, in the further studies, more stable algorithms should be implemented for the orthogonalisation of the polynomials higher than 2nd order. Possible candidates can be: modified Gramm-Schmidt algorithm [23], Householder orthogonalisation algorithm [27], Givens reflections.

5 Results of Test Calculations

The methodology described above was coded and tested. In the current stage of work, the method has been implemented for calculations of the regular and irregular single cells with reflective boundary conditions on the outer borders (infinite lattices). This set of the test calculations was chosen to cover different types of the geometries arising in the real reactor calculations. The results of the calculations were compared with the reference Monte Carlo calculations using OpenMC code [28]. OpenMC is a relatively young Monte Carlo particle transport code developed in the Computational Reactor Physics Group (CRPG) at Massachusetts Institute of Technology (MIT). It possesses several features that were used in current study, such as the possibility to simulate the systems with given cross sections sets and external neutron sources. In the current study, the results of the OpenMC simulations were used as reference for the comparison with the developed transport solver.

4.1 Calculations of the Regular Cells

In the current study, the developed solver was applied for calculation of the neutron fluxes in the regular hexagonal, pentagonal and square cells. Hexagonal and square geometries very often arise within nuclear reactors analysis. Pentagon cell was chosen for the demonstration. The geometries of the cells are presented in Figure 5. Hexagonal and square geometries simulate real reactor unit cells: central zone - fuel, middle zone - cladding, outer zone - moderator. The same is applicable to the pentagonal geometry, except the fact that this geometry is quite unusual for the modern reactor design.

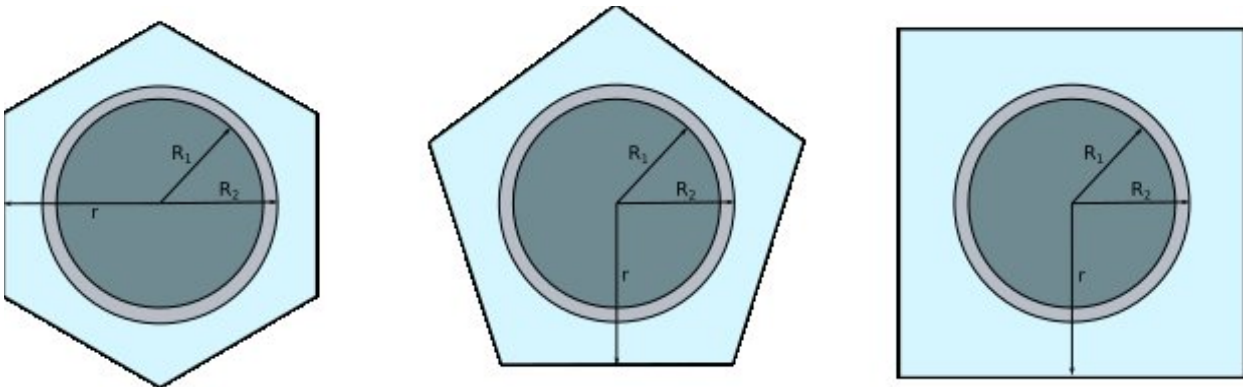

Figure 5. Geometries of hexagonal and square cells used in the calculations

The calculations were carried out using one energy group with the source of the neutrons located in the outermost region of the cells. This is typical for the thermal group where majority of the moderated neutrons are coming from the moderator region.

Two different cross sections sets were chosen to simulate the cells containing fuel and absorbers; two different cross section sets were chosen to simulate normal and low densities of the moderator; three different pitches were chosen to check the cells with different pitches of the outer regular hexagon, pentagon or square. The combinations of the mentioned parameters are presented in Table 3.

Case number	Moderator		Fuel		r_h, cm	r_s, cm	r_p, cm
	Σ_t, cm^{-1}	Σ_s, cm^{-1}	Σ_t, cm^{-1}	Σ_s, cm^{-1}			
1	1.5	1.425	0.63	0.33	0.508	0.504	0.508
2	1.5	1.425	10.0	0.30	0.508	0.504	0.508
3	1.5	1.425	0.63	0.33	0.635	0.630	0.635
4	1.5	1.425	10.0	0.30	0.635	0.630	0.635
5	1.5	1.425	0.63	0.33	0.79375	0.7875	0.79375
6	1.5	1.425	10.0	0.30	0.79375	0.7875	0.79375
7	0.1	0.095	0.63	0.33	0.508	0.504	0.508
8	0.1	0.095	10.0	0.30	0.508	0.504	0.508
9	0.1	0.095	0.63	0.33	0.635	0.630	0.635
10	0.1	0.095	10.0	0.30	0.635	0.630	0.635
11	0.1	0.095	0.63	0.33	0.79375	0.7875	0.79375
12	0.1	0.095	10.0	0.30	0.79375	0.7875	0.79375

Table 3: Input parameters for the square and hexagonal cells

The radii R_1 (radius of the fuel zone) and R_2 (outer radius of the cladding) were 0.386 cm and 0.455 cm respectively for all cases. The total and scattering cross sections for the cladding region were Σ_t=0.276 cm^{-1} and $\Sigma_s = 0.272$ cm^{-1} correspondingly for all cases. The source of the neutrons was located in the moderator region with the density equal 1.0 cm^{-3} for all cases.

The discrepancies in the neutron fluxes for fuel, cladding and moderator regions between OpenMC and developed solver are presented in Table 4.

Case	Difference (hexagonal cell)						Difference (square cell)						Difference (pentagonal cell)					
	Fuel		Cladding		Mod.		Fuel		Cladding		Mod.		Fuel		Cladding		Mod.	
	P_0	P_2	P_0	P_2	P_0	P_2	P_0	P_2	P_0	P_2	P_0	P_2	P_0	P_2	P_0	P_2	P_0	P_2
1	0.0	0.0	-0.1	0.0	-0.3	-0.2	-0.1	-0.1	-0.5	-0.6	0.3	0.4	0.0	0.0	-0.2	-0.2	-0.2	-0.1
2	0.1	0.1	-3.1	-2.8	-3.6	-3.1	0.0	0.0	-3.7	-3.7	-0.3	0.9	0.0	0.0	-3.0	-2.9	-3.0	-2.3
3	0.1	0.0	-0.1	-0.1	-0.1	0.0	-0.1	-0.2	-0.2	-0.4	0.1	0.2	0.0	0.0	-0.1	-0.1	-0.2	-0.1
4	0.4	0.3	-1.4	-1.3	-2.1	-1.3	0.5	0.0	-1.6	-1.8	-1.0	0.3	0.3	0.0	-1.0	-1.0	-2.1	-1.3
5	0.2	0.0	0.2	0.0	-0.3	-0.1	0.3	0.1	0.4	0.1	-0.4	-0.1	0.1	0.0	0.2	0.0	-0.3	-0.1
6	0.6	0.3	0.2	-0.3	-2.6	-1.2	0.7	0.3	0.7	0.0	-3.1	-1.3	0.2	0.0	0.2	-0.3	-2.2	-0.9
7	0.0	0.0	-0.7	-0.6	-0.7	-0.6	0.0	0.0	-0.7	-0.6	-0.4	-0.3	-0.1	-0.1	-0.5	-0.4	-0.5	-0.4
8	0.0	0.0	-8.5	-8.2	-6.8	-6.6	0.0	0.0	-8.3	-7.9	-4.2	-3.8	0.0	0.0	-8.8	-8.3	-7.1	-6.8
9	0.0	0.0	-0.6	-0.5	-0.9	-0.8	-0.1	-0.1	-0.6	-0.5	-0.5	-0.4	0.0	0.0	-0.5	-0.3	-0.6	-0.5
10	0.0	0.0	-7.5	-7.2	-7.2	-6.8	0.5	0.5	-7.0	-6.6	-5.0	-4.6	0.0	0.0	-7.6	-7.2	-7.5	-7.2
11	0.0	0.0	-0.3	-0.2	-0.6	-0.5	0.0	0.0	-0.3	-0.2	-0.5	-0.4	0.0	0.0	-0.4	-0.3	-0.5	-0.4
12	0.0	0.0	-6.5	-6.1	-7.1	-6.7	0.3	0.3	-5.8	-5.4	-6.2	-5.8	0.2	0.2	-6.5	-6.1	-6.5	-6.1

Table 4: Discrepancies in the fluxes between transport solver and OpenMC, %

Zeroth order as well as second order sets of polynomials for the flux expansion have been used in the calculations. The results obtained with the first degree of the polynomials are identical to the flat source approximation for the cells with the reflective boundary conditions. To minimize computational errors, the level of the angle discretisation was high. Each outer boundary of the polygons was subdivided into segments with the size equal to 0.05 cm, the azimuthal angle was discretized into 128 equidistant sectors. The maximal track spacing was 0.005 cm. The extended trapezoidal rule was used for integration of the collision probabilities within the sectors and segment boundaries. The Gaussian quadrature was used for integration of the collision probabilities along the track. Due to the limitations of the current version of the program, there was not any discretisation on the polar angle. The standard rather than partial Bickley functions were used for the evaluation of the collision probabilities.

Table 4 indicates that the results of the calculations obtained with the proposed methodology agree well with results obtained by the Monte-Carlo method. The errors in the fluxes for major part of the cases are below 1%. In the fuel zone, the errors are less than 1% for all the cases. Even for the most critical cases with high absorption cross sections, the difference in the fluxes does not exceed 0.3% in the fuel/absorber regions. However, there are some cases where discrepancies between OpenMC and the developed solver are quite high. The major discrepancies in the fluxes in the cladding/moderator regions are observed for the cases 8, 10 and 12. The cases 8, 10 and 12 are characterized by the low total cross sections of the moderator region and consequently, its low optical thickness. As mentioned above, the current version of the solver uses only one sector for the discretisation on the polar angle. However, for the optically thin mediums it is essential to account for the distribution of the neutrons in the polar direction due to the strong streaming effect. Therefore, it can be expected that the higher discretisation of the polar angle will lead to better performance of the code for the regions with small optical thickness. The same behavior of the discrepancies can be observed for case 2 where optically thin outermost region arise due to the very small distance between cladding and outer edges of the cells.

The results of the calculations presented in Table 4 were obtained using a set of zero order of polynomials (i.e. flat flux approximation) and a set of second order of polynomials (i.e. six spatial modes, up to the second order of polynomials). Where the P_0-approximation was used for the same geometries of the cells, the results are generally worse than those calculated with the P_2-approximation; however, the difference between the results was not as large as it was expected. The main reason can be found in the symmetry of the test problem. In such cases, the flat-flux approximation can provide acceptable results, since no tilted fluxes appear like in a real unit cell.

The main advantages of the proposed methodology are expected to be seen on the level of full assembly calculations (as it has already been shown for regular hexagonal assemblies [1]).

4.2 Calculations of the Arbitrary Cells

For further testing of the developed solver, two sets of the arbitrary cells with the outer polygons with four and six sides were calculated using the proposed methodology. The results were compared with the results of the Monte Carlo simulations. The reference Monte Carlo simulations were performed using OpenMC code. The vertices of the polygon were randomly placed on the circle with the radius $R_{out} = 0.890955$ cm for the arbitrary quadrangles and $R_{out} = 0.733235$ cm for the arbitrary hexagons. In total, 5 different outer polygons were obtained and considered both for hexagonal and quadrangle cells. The geometries of the arbitrary cells are presented in Figure 6 and Figure 7.

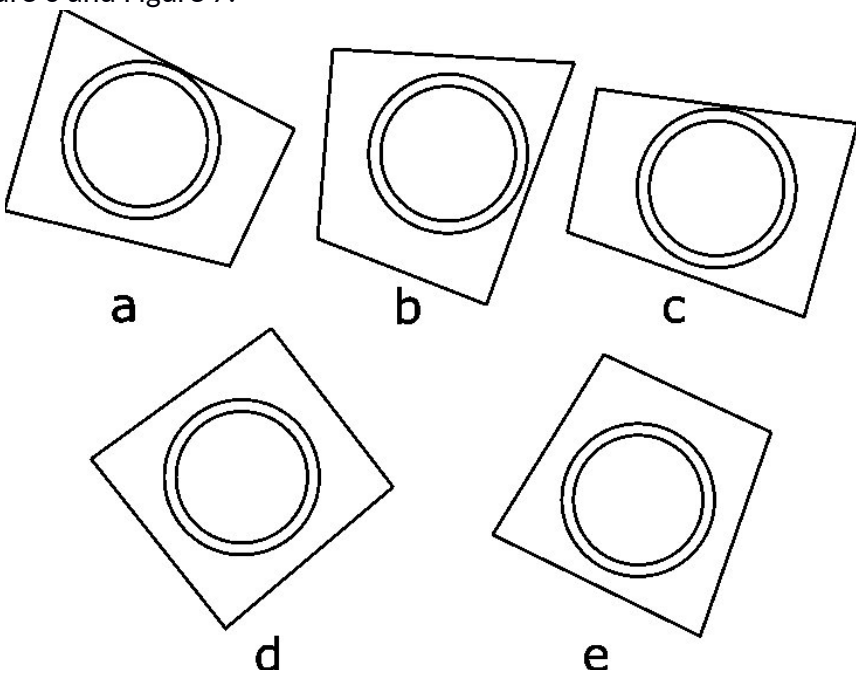

Figure 6. Geometries of the cells with the arbitrary outer quadrangles

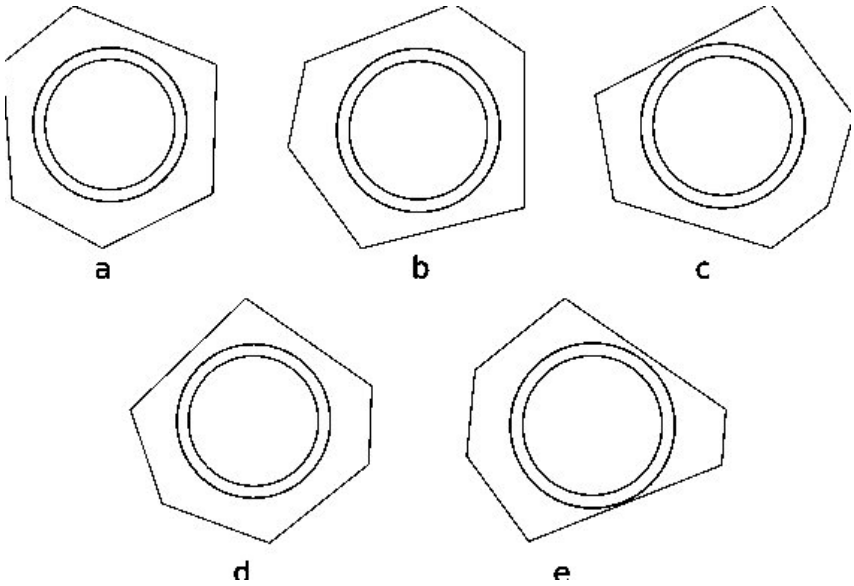

Figure 7. Geometries of the cells with the arbitrary outer hexagons

The coordinates of the vertices of the outer polygons are given in Table 8.

Polygons with 4 sides								
Outer polygon	Vertex 1		Vertex 2		Vertex 3		Vertex 4	
	x	y	x	y	x	y	x	y
a	0.88869	0.06347	-0.45811	0.76410	-0.79246	-0.40720	0.51453	-0.72736
b	0.72395	0.51933	-0.65999	0.59851	-0.74700	-0.48558	0.22029	-0.86329
c	0.81009	0.37088	-0.68460	0.57020	-0.85532	-0.24947	0.50258	-0.73567
d	0.17341	0.87381	-0.88432	0.10856	-0.09422	-0.88596	0.88920	-0.05586
e	0.79371	0.40475	-0.20345	0.86742	-0.86699	-0.20525	0.37076	-0.81015

Polygons with 6 sides												
Outer polygon	Vertex 1		Vertex 2		Vertex 3		Vertex 4		Vertex 5		Vertex 6	
	x	y	x	y	x	y	x	y	x	y	x	y
a	0.57779	0.45143	0.04727	0.73171	-0.60780	0.41014	-0.64082	-0.35634	0.21999	-0.69946	0.64098	-0.35605
b	0.72704	0.09512	0.31728	0.66103	-0.59122	0.43370	-0.58937	-0.43620	-0.19919	-0.70566	0.62976	-0.37554
c	0.71474	0.16363	0.36582	0.63546	-0.70223	0.21098	-0.72677	-0.09717	0.16921	-0.71344	0.66375	-0.31157
d	0.60560	0.41338	-0.27098	0.68132	-0.58377	0.44367	-0.73069	-0.06100	-0.26119	-0.68514	0.71188	-0.17567
e	0.54316	0.49256	-0.09985	0.72640	-0.68663	0.25724	-0.70357	-0.20645	0.04715	-0.73171	0.73054	-0.06276

Table 5: Coordinates of the vertices of the arbitrary polygons

Different cross sections for the region containing the fuel, different total and scattering cross sections for the outermost zone have been chosen for the analyses. Two different cross sections sets were chosen to simulate the cells with fuel and absorbers and two different cross section sets were chosen to simulate normal and low densities of the moderator. The final combination of the cells parameters are given in Table 6.

Case number	Outer polygon type	Fuel		Moderator	
		Σ_t	Σ_s	Σ_t	Σ_s
1	a	0.63	0.33	1.5	1.425
2	a	10.0	0.30	1.5	1.425
3	a	0.63	0.33	0.1	0.095

4	a	10.0	0.30	0.1	0.095
5	b	0.63	0.33	1.5	1.425
6	b	10.0	0.30	1.5	1.425
7	b	0.63	0.33	0.1	0.095
8	b	10.0	0.30	0.1	0.095
9	c	0.63	0.33	1.5	1.425
10	c	10.0	0.30	1.5	1.425
11	c	0.63	0.33	0.1	0.095
12	c	10.0	0.30	0.1	0.095
13	d	0.63	0.33	1.5	1.425
14	d	10.0	0.30	1.5	1.425
15	d	0.63	0.33	0.1	0.095
16	d	10.0	0.30	0.1	0.095
17	e	0.63	0.33	1.5	1.425
18	e	10.0	0.30	1.5	1.425
19	e	0.63	0.33	0.1	0.095
20	e	10.0	0.30	0.1	0.095

Table 6: Input parameters of the arbitrary cells

All 20 cases presented in Table 6 were simulated using both OpenMC code and the developed transport solver both for quadrangle and hexagonal cells. The results of the OpenMC simulations were used as reference. The discrepancies between the transport solver (with and without flux expansion) and Monte Carlo reference results are presented in Table 7.

Case number	Quadrangle cell						Hexagonal cell					
	Fuel		Cladding		Moderator		Fuel		Cladding		Moderator	
	P_0	P_2	P_0	P_2	P_0	P_2	P_0	P_2	P_0	P_2	P_0	P_2
1	0.2	0.0	0.2	-0.1	-0.5	-0.2	0.0	0.0	0.0	0.0	-0.3	-0.1
2	0.0	0.0	-0.1	-0.8	-3.9	-1.7	0.1	0.0	-1.1	-1.1	-2.7	-1.7
3	0.0	0.0	-0.4	-0.3	-0.8	-0.7	0.0	0.0	-0.4	-0.3	-0.7	-0.7
4	0.0	0.0	-7.5	-6.9	-8.1	-7.5	0.0	0.0	-7.5	-7.1	-7.6	-7.3
5	0.0	-0.2	-0.1	-0.3	-0.2	0.1	0.0	0.0	-0.1	-0.1	-0.1	0.1
6	0.0	0.0	-0.8	-1.5	-3.0	-0.2	0.1	0.0	-1.7	-1.7	-1.6	-0.5
7	0.0	0.0	-0.4	-0.4	-0.6	-0.5	0.0	0.0	-0.4	-0.4	-0.5	-0.4
8	0.5	0.5	-7.9	-7.5	-7.5	-7.0	0.2	0.1	-7.9	-7.5	-7.2	-6.8
9	0.2	0.0	0.3	0.0	-0.6	-0.1	0.2	0.1	0.2	0.1	-0.5	-0.3
10	0.0	0.0	-0.2	-0.9	-3.7	-1.3	0.6	0.4	-0.8	-1.0	-4.3	-2.0
11	-0.1	-0.1	-0.4	-0.3	-0.6	-0.4	0.0	0.0	-0.4	-0.3	-0.7	-0.6
12	0.0	0.0	-7.3	-6.8	-6.5	-5.4	0.2	0.2	-7.7	-7.3	-8.0	-7.4
13	0.1	0.0	0.1	0.0	0.4	-0.1	0.1	0.0	0.1	0.0	-0.4	-0.2
14	0.5	0.0	-0.4	-0.6	-3.0	-1.8	0.0	0.0	-1.0	-1.1	-3.6	-1.9
15	0.0	0.0	-0.4	-0.3	-0.8	-0.7	0.0	0.0	-0.4	-0.3	-0.7	-0.6
16	0.0	0.0	-7.0	-6.5	-7.3	-6.8	0.1	0.1	-7.7	-7.2	-8.0	-7.6
17	0.2	0.0	0.2	0.0	-0.3	-0.1	0.1	0.0	0.0	0.0	-0.3	-0.2

18	0.5	0.5	-0.4	-0.8	-3.3	-1.6	0.5	0.4	-1.0	-1.0	-3.0	-1.8
19	-0.1	-0.1	-0.4	-0.3	-0.6	-0.5	-0.1	0.0	-0.4	-0.3	-0.6	-0.5
20	0.0	0.0	-7.3	-6.9	-7.5	-7.1	0.0	0.0	-7.7	-7.3	-8.1	-7.7

Table 7: Discrepancies in the fluxes between transport solver and OpenMC (%) for irregular cells (the rows with the normal density of water are shaded with grey colour)

As can be seen from the table above, the difference in the fluxes do not exceed 0.5% on in the fuel region for all the cases under consideration. For the cases with normal density of water, the discrepancies in the fluxes are slightly higher or less than 2% when the polynomials of the second order are used for the flux expansion. For the case of the low moderator density, the discrepancy in the moderator and cladding regions rises and reaches 8.1% for the case number 4. The reason for this difference is the absence of the discretisation on the polar angle in the current version of the transport solver. Application of the second order of the flux expansion improves the accuracy almost for all the considered cases. In some cases, the accuracy stays on the same level. The highest improvements are observed for the cells with the absorber and normal density of the water. In general, the accuracy of the calculations is acceptable for the practical calculations even without discretisation of the polar angle, when the normal density of the moderator is considered. However, discretisation of the problem on polar angle is required to take into account streaming effects for lower densities of moderator. The effect of the flux expansion for the single cell is not as high as it was expected (in absolute numbers). Flat flux approximation provides reasonable results for the major part of the considered cells. It happens, on our view, due to the reflective boundary conditions applied on the sides of the cells. The effect of the flux expansion becomes more visible when the proposed methodology is applied for the full assembly simulations as it has been demonstrated for the regular hexagonal assemblies [1, 19].

4.3 Influence of the polar angle discretisation

In order to clarify the influence of the polar angle discretisation on the accuracy of the results, the set of the calculations was performed. Current version of the solver for the arbitrary cells is able to simulate the problems using only one polar sector. However, older version of the code developed for the regular hexagonal cells [1] have got the possibility to run simulations with the higher number of the polar sectors. Therefore, the numerical experiments were performed for the regular hexagonal cells only. The cell number 8 from Table 3 was chosen for the tests as it had highest discrepancies in flux. Calculations were performed both for zero and second orders of the polynomials. The number of the polar sectors varied from 1 to 15. The results of the calculations are presented in Figure 8.

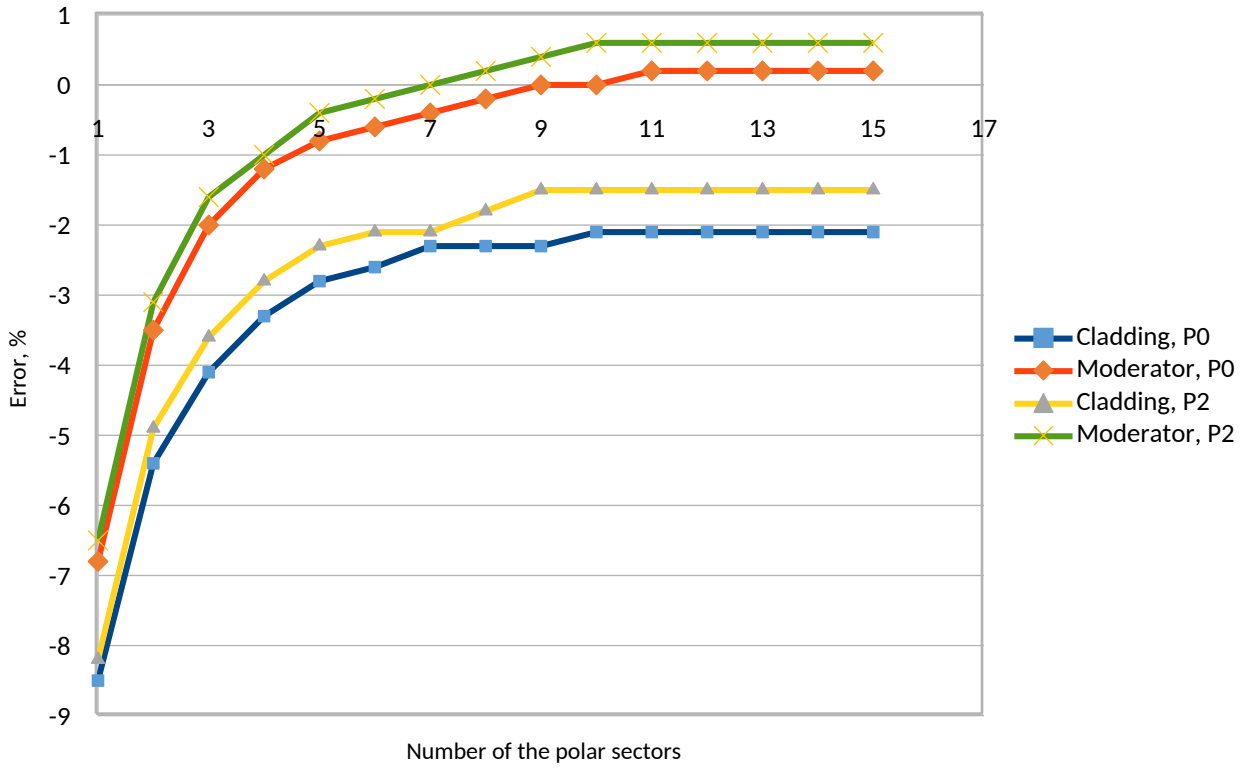

Figure 8. Results of the calculations with different number of polar sectors

It is observed that the error both in moderator and cladding regions rapidly decreases as the number of the polar sectors increases. The errors in the moderator region falls below 1.0% both for the zero and second orders of the polynomials. The errors in the cladding regions decrease from ~8% for the one polar sector to 1.5-2% for 11 polar sectors. The converged behaviour of the error for the number of polar sectors higher than 11 indicates that further improvements of the results should not be expected. It should be noted that the considered case was quite artificial (very dense lattice with the strong absorbers and low density of the moderator) and better results can be achieved for the more realistic cases. The results presented in the current section creates the evidence that quite high discrepancies observed for the cases with the low moderator density can be decreased with the higher discretisation of the polar angle.

4.4 Performance of the CCCP Method with Orthogonal Flux Expansion

To evaluate the efficiency of the proposed methodology, a set of the calculations has been performed. The case number 4 (hexagonal cell) from Table 3 has been chosen for the tests. The moderator region of the cell was subdivided into several concentric subregions (see Figure 9). The calculations of the fluxes were performed using flat flux approximation. The time of the calculations was measured and compared with the calculation time of the same cell using second order of the polynomials for the flux expansion. In the case of the second order of the flux, there were no any subdivisions of the outermost region. The results of the calculations are presented in Table 8.

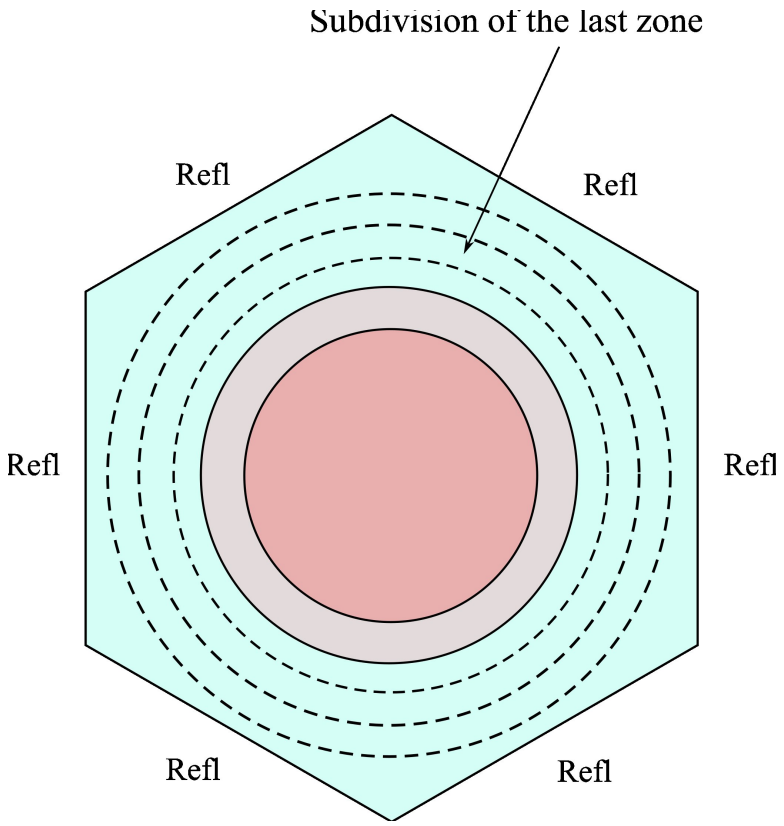

Figure 9. Subdivision of the outermost region with reflective boundary conditions

Number of subregions	$\frac{t_{P_0}}{t_{P_2}}$	Error in average flux, %	
		P_0	P_2
1	0.80	-1.8	-1.1
2	0.85	-1.5	
3	0.97	-1.5	
4	1.10	-1.4	
5	1.31	-1.4	
7	1.70	-1.4	
10	2.72	-1.4	

Table 8: Results of the performance tests

The relationship between calculation time in P_0 and P_2 approximations ($\frac{t_{P_0}}{t_{P_2}}$) is given in the second column of the table, while the error for P_0 approximation is presented in the third column. It can be observed, that the computational time for the P_0-approximation without subdivision of the moderator zone is approximately on 20% less than the computational time for the P_2-approximation but at the cost of reduced accuracy as it is to be expected. However, increasing the number of sub-zones leads to an increased calculation time and, as already seen for the case with four sub-zones,

the P_2-approximation becomes computationally more efficient (the quotient $\frac{t_{P_0}}{t_{P_2}}$ becomes higher than 1) while the accuracy of the P_0 approximation is still worse than the P_2 approximation. The error of the calculations (in comparison with Monte-Carlo calculations), remains higher for the P_0-approximation than for the P_2-approximation (-1.1%) at least until 10 subregions are inserted. The difference between the flat flux approximation and the second order polynomial expansion is not as high as it was expected for the case of the single cell with reflective boundary conditions. Due to the symmetry of the problem, flat flux approximation can provide comparably accurate solution. However, the results for the flat flux approximation are expected to become worse in the case of a tilted flux. Such a flux will appear in the unit cells of a real fuel assembly due to inhomogeneity caused by zone enrichment and the influence of guide tubes for the control rods in western type fuel assemblies, or the different size of the water gaps between assemblies.

4.5 Visualisation

Additional verification of the methodology was performed by producing two-dimensional visualisation of the neutron fluxes within hexagonal cell. Appropriate sets of the pictures were produced for hexagonal cell containing absorber or fuel in the central region. The results of the visualisation are presented in Figure 10 - Figure 15.

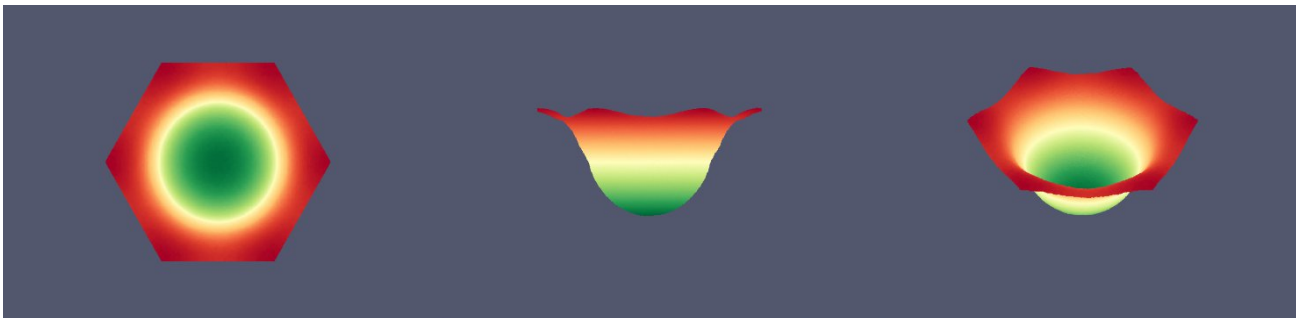

Figure 10. Plane (left), axial (middle) and 3D (right) views of the neutron flux distribution within hexagonal fuel cell calculated using OpenMC

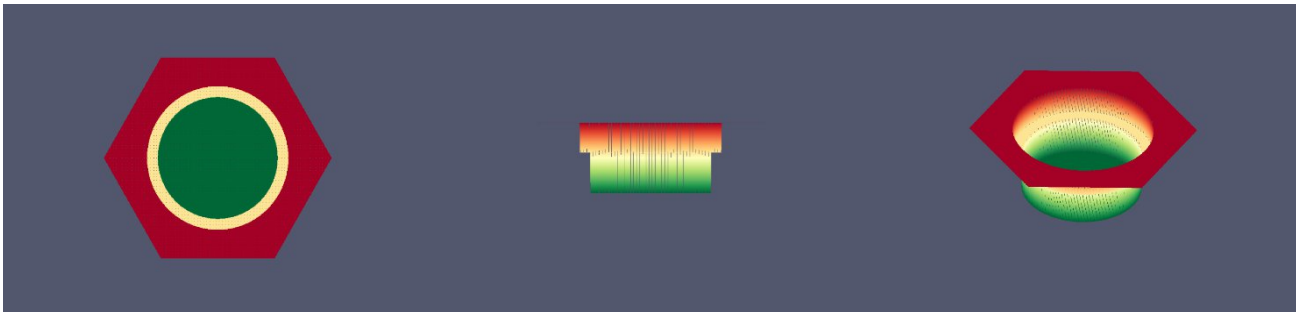

Figure 11. Plane (left), axial (middle) and 3D (right) views of the neutron flux distribution within hexagonal fuel cell calculated with transport solver (P_0 approximation)

Figure 12. Plane (left), axial (middle) and 3D (right) views of the neutron flux distribution within hexagonal fuel cell calculated with transport solver (P_2 approximation)

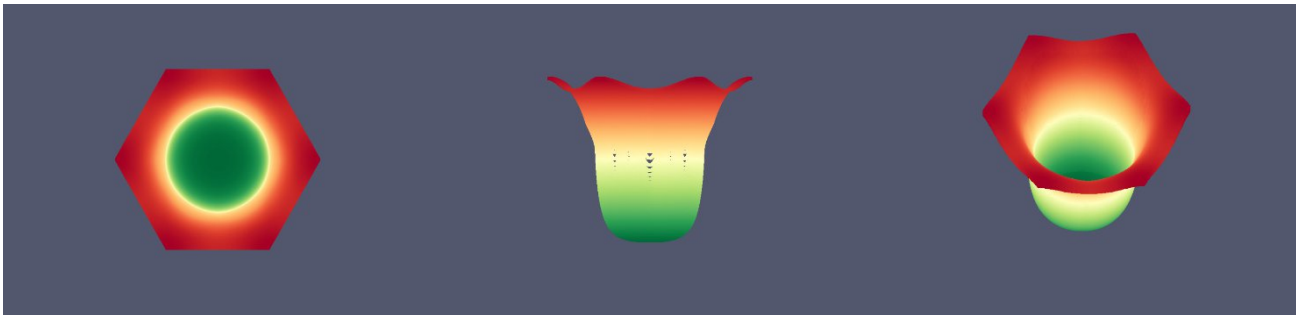

Figure 13. Plane (left), axial (middle) and 3D (right) views of the neutron flux distribution within hexagonal absorber cell calculated with OpenMC

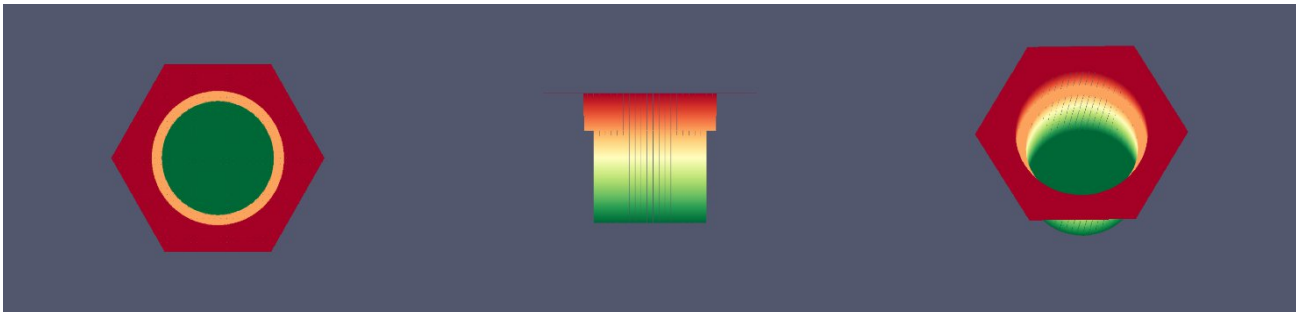

Figure 14. Plane (left), axial (middle) and 3D (right) views of the neutron flux distribution within hexagonal absorber cell calculated with transport solver (P_0 approximation)

Figure 15. Plane (left), axial (middle) and 3D (right) views of the neutron flux distribution within hexagonal fuel cell calculated with transport solver (P_2 approximation)

The above figures represent the flux distributions inside the regions of a typical eastern type

light water reactor unit cell for P_0 and P_2 approximations. The calculations have been performed for fuel and absorber cells and the flux was visualized using the possibilities of the ParaView software [29]. The neutron flux within calculation regions was reconstructed on the basis of Equation and values (such as Φ_k, $b_{k',k}$) obtained during the transport solver run. It is obvious from Figure 11, Figure 14 that the shape of flux for P_0 approximation is flat for each zone of the cells (classical for the flat flux approximation). In the identical cases for the P_2 approximation, the shape of the flux in the different regions is represented by a second order surface. Thus, the flux distribution is much closer to the reference shape of the flux calculated by OpenMC and presented in Figure 12 and Figure 15 (see also [17]). As can be seen from the figures above, the gradient of the flux for the fuel cell is much less than for the absorber cell. It is one of the reasons, why flat-flux approximation provides good results for the fuel cells. For the absorber cell, the gradient of the flux is much higher and the effect of the higher order polynomials is more visible. The visualisation demonstrates significantly better performance of the calculations using the P_2 approximations in comparison with those assuming a flat flux approximation. Visible jumps at the boundary between moderator and cladding and between cladding and fuel/absorber demonstrate that the second order approximation is not powerful enough to reconstruct the complexity of the problem completely. However, from engineering perspective, the second order approximation seems to be a good and reliable choice.

6 Conclusions

The methodology of current coupling collision probability method combined with orthogonal flux expansion has been applied for the numerical solution of the transport equation. In contrast to the previously developed techniques, where orthogonal coefficients evaluated analytically for the limited number of geometries, the process of the orthogonalisation in the current solver is based on a numerical implementation of the Gram-Schmidt procedure. This has extended the proposed methodology to broad range of heterogeneous geometries. Application of the orthogonal polynomials for the flux expansion allows a significant reduction of the number of calculation regions, while maintaining or improving the accuracy of the results. This technique allows to overcome one of the drawbacks of the traditional collision probability method – poor representation of the spatial distribution of the neutron source/flux. It was shown that the application of the higher order of the polynomials for the flux expansion improves the accuracy and the efficiency (in terms of the accuracy/run time ratio) of the calculations. Another benefit of the presented technique is the possibility to relieve the user from determining the correct discretisation of the calculation regions with all the following challenges such as the adoption of ray tracing to the changed geometry to achieve a reliable result. It can be even more important since, for example, in the well-known code HELIOS [12] utilising the CCCP method with the flat flux approximation, the spatial discretisation of the calculation regions is performed manually by user. Therefore, the refinement of the calculation regions requires additional effort and makes the input files more complicated. Application of the proposed methodology would significantly simplify this process for the user. In this case change of the only one input parameter (order of the polynomials for the flux expansion) would improve the accuracy of the computations. Application of the numerical methods for evaluation of the coefficients of orthogonal polynomials increase the flexibility of the method significantly. Previously,

current coupling collision probability method with expansion of the neutron flux by orthogonal polynomials was applicable in practice only for the regular lattices [16, 1]. Now, this technique can be applied for much wider range of the 2D problems including unstructured geometries.

The verification of the developed code has been performed both for regular and irregular cells. The results of the comparison with Monte-Carlo code demonstrate sufficient accuracy for practical calculations. Even though the results obtained in presented study agree well with the results of reference Monte Carlo calculations, further investigations are necessary for a better understanding of the stability and limitations of the proposed methodology. The Gram-Schmidt procedure is known to be unstable. Even though we did not encounter any problems with the stability of the solver during the above tests, we are aware about possible stability problems which can arise for some complicated geometries or for the higher orders of the polynomials. Therefore, some other methods for construction of the orthogonal set of the polynomials need to be investigated (for example, applying the modified Gram-Schmidt algorithm [23]). The numerical scheme for integration over triangle used in present study enables the evaluation of integrals up to a maximum of the 7^{th} order. For higher orders of the polynomials, this scheme of the integration can be unstable. Therefore, other quadrature sets should be investigated.

After an appropriate completion and extensions, the developed transport solver will be incorporated into a nodal code to enable a multi-scale and multimethodological approach for pin-wise 2D calculations. This approach combines a transport solver using the real fuel assembly geometry reproduced on an unstructured mesh with the boundary conditions extracted from the 3D full core nodal solution. The previous investigations performed in this field clearly demonstrate that this kind of coupling can obtain reliable results with a good accuracy on the pin level [2]. The main efforts should be made in the coupling of the available transport code into an available nodal code and the optimisation in the given software environment respecting the requests of a licensing grade software tool.

Acknowledgements

The authors would also like to acknowledge the support of the EPSRC through the funding of the EPSRC grant “Innovative LWR Simulation Tool for the Nuclear Renaissance in the UK” (EPSRC grant EP/R005850/1).

References

[1] D. Litskevich and B. Merk, «SP3 Solution versus Diffusion Solution in Pin-by-Pin Calculations and Conclusions Concerning Advanced Methods,» *Journal of Computational and Theoretical Transport*, vol. 43, pp. 214-239, 2014.

[2] B. S. Collins, *Multiscale Methods for Nuclear Reactor Analysis*, Argonne National Laboratory, 2011.

[3] B. Merk and U. Rohde, «An analytical solution for the consideration of the effect of adjacent fuel elements,» *Annals of Nuclear Energy, vol*. 38, № 11, pp. 2428-2440, 2011.

[4] «The Consortium for Advanced Simulation of Light Water Reactors,» A DOE Energy Innovation Hub, [Online]. Available: https://www.casl.gov/. [Accessed: 14 11 2019].

[5] B. Lindley, D. Allen, J. Lillington, J. Smethurst, P. Smith, D. Bowman, L. Dwyer, K. Lai, A. Levers,

K. Vikhorev, D. Litskevich, B. Merk, E. Patelli, E. Patterson, M. Bankhead, A. Peakman, S. D. Haas, C. Jackson, J. Draup, E. Galenne, G. Parks, E. Shwageraus and M. Eaton, «Modelling and Simulation Activities in Support of the UK Nuclear R&D Programme on Digital Reactor Design,» in *26th International Conference on Nuclear Engineering (ICONE)*, London, 2018.

[6] E. A. Thomas, M. Evans, «A new three-dimensional parallel discrete ordinates code in SCALE,» *Nuclear technology*, vol. 2, pp. 171 - 200, 2010.

[7] J. Leppänen, «Development of a new Monte Carlo reactor physics code,» VTT Technical Research Centre of Finland, 2007.

[8] K. Koebke and M. Wagner, «The determination of the pin-power distribution in a reactor core on the basis of nodal coarse mesh calculations,» *Atomkernenergie*, vol. 30, p. 136, 1977.

[9] A. Johnson, S. Grill and J. Rec, «Nodal imbedded calculation for the retrieval of local power peaking,» в *International Topical Meeting on Advances in Mathematical Methods for the Solution of Nuclear Engineering Problems*, 1981.

[10] A. Jonsson, S. Grill and J. Rec, «Nodal imbedded method to recover local power peaking from coarse-mesh reactor calculations,» in *Trans. Am. Nucl. Soc.*, 1980.

[11] F. Nissen, *Determination of Local Pin Powers in the Framework of Nodal Coarse Mesh Solutions*, Riso National Laboratory, 1982.

[12] E. A. Villarino, R. J. J. Stamm'ler, A. A. Ferri and J. J. Casal, «HELIOS: Angularly Dependent Collision Probabilities,» *Nuclear Science and Engineering*, vol. 112, № 1, pp. 16-31, 1992.

[13] O. Safarzadeh, A. Minuchehr and A. S. Shirani, «Heterogeneous reactor core transport technique using response matrix and collision probability methods,» *Annals of Nuclear Energy*, vol. 62, pp. 137 - 143, 2013.

[14] G. A. Gunow, «Full Core 3D Neutron Transport Simulation Using the Method of Characteristics with Linear Sources,» PhD Thesis. Massachusetts Institute of Technology, June 2018.

[15] R. M. Ferrer and J. Rhodes, «A Linear Source Approximation Scheme for the Method,» *Nuclear Science and Engineering*, vol. 182, pp. 151 - 165, 2015.

[16] I. Rubin and N. Dneprovskaya, «Heterogeneous calculation of hexagonal lattice by transmission probability method,» *Atomnaya energiya*, vol. 87, № 5, pp. 330 - 335, 1999.

[17] B. Merk and R. Koch, «On the influence of spatial discretization in LWR cell- and lattice calculations with HELIOS 1.9,» *Annals of Nuclear Energy*, vol. 35, № 08, pp. 1492-1501, 2008.

[18] E. E. Lewis and W. F. Miller, Computational methods of neutron transport, New York: JOHN WILEY & SONS, 1984.

[19] D. Litskevich, «Development of an advanced neutron transport solver for zooming in DYN3D,» RWTH Aachen, Aachen, 2017.

[20] I. E. Rubin and N. M. Dneprovskaya, «Heterogeneous calculation of hexagonal lattices by the method of passage probabilities,» *Atomic Energy*, vol. 87, pp. 791 - 797, 1999.

[21] R. J. J. Stamm'ler and M. J. Abbate, Methods of Steady-state Reactor Physics in Nuclear Design, Academic Press, 1983.

[22] A. Hebert, Applied Reactor Physics, Presses internationales Polytechnique, 2009.

[23] Å. Björck, «Numerics of Gram-Schmidt orthogonalization,» *Linear Algebra and its*

Applications, vol. 197–198, pp. 297-316, 1994.

[24] T. Moan, «Experiences with orthogonal polynomials and best numerical integration formulas on a triangle with particular reference to finite element approximations,» *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift fuer Angewandte Mathematik und Mechanik*, vol. 54, № 7, pp. 501 - 508, 1974.

[25] B. Joe, «GEOMPACK — a software package for the generation of meshes using geometric algorithms,» *Advances in Engineering Software and Workstations*, vol. 13, № 5-6, pp. 325-331, 1991.

[26] A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry, A. R. Terrel, Š. Roučka, A. Saboo, I. Fernando, S. Kulal, R. Cimrman and A. Scopatz, «SymPy: symbolic computing in Python,» *PeerJ Computer Science*, vol. 3, p. e103, January 2017.

[27] A. S. Householder, «Unitary Triangularization of a Nonsymmetric Matrix,» *Journal of the ACM*, vol. 5, pp. 339 - 342, 1958.

[28] P. K. Romano, N. E. Horelik, B. R. Herman, A. G. Nelson, B. Forget and K. Smith, «OpenMC: A state-of-the-art Monte Carlo code for research and development,» *Annals of Nuclear Energy*, vol. 82, pp. 90 - 97, 2015.

[29] U. Ayachit, The ParaView Guide: A Parallel Visualization Application, Kitware, 2015.

[30] G. B. Arfken and H. J. Weber, Mathematical Methods for Physicists, Sixth Edition, Elsevier Academic Press, 2005.

[31] C. F. Dunkl and Y. Xu, Orthogonal Polynomials of Several Variables, Cambridge: Cambridge University Press, 2014.

[32] G. R. Cowper, «Gaussian quadrature formulas for triangles,» *International Journal for Numerical Methods in Engineering*, vol. 7, № 3, pp. 405 - 408, 1973.