Vidal Ferràndiz, A.; Fayez Moustafa Moawad, R.; Ginestar Peiro, D.; Verdú Martín, GJ. (2014). Solution of the Lambda modes problem of a nuclear power reactor using an h-p finite element method. Annals of Nuclear Energy. 72:338-349. doi:10.1016/j.anucene.2014.05.026.

Solution of the Lambda modes problem of a nuclear power reactor using an *h-p* finite element method

A. Vidal-Ferrandiz[a], R. Fayez[a], D. Ginestar[b], G. Verdú[a,*]

[a] *Instituto de Seguridad Industrial: Radiofísica y Medioambiental, Universitat Politècnica de València, Camino de Vera s/n, 46022, València, Spain*
[b] *Instituto Universitario de Matemática Multidisciplinar, Universitat Politècnica de València, Camino de Vera s/n, 46022, València, Spain*

Abstract

Lambda modes of a nuclear power reactor have interest in reactor physics since they have been used to develop modal methods and to study BWR reactor instabilities. An *h-p*-adaptation finite element method has been implemented to compute the dominant modes the fundamental mode and the next subcritical modes of a nuclear reactor. The performance of this method has been studied in three benchmark problems, a homogeneous 2D reactor, the 2D BIBLIS reactor and the 3D IAEA reactor.

Keywords: Lambda Modes, Finite Element, *h-p*-adaptability, eigenvalue problems.

1. Introduction and motivation

The neutron diffusion equation is an approximation of the neutron transport equation that states that the neutron current is proportional to the gradient of the neutron flux by means of a diffusion coefficient. This approximation is analogous to the Fick's law in species diffusion and to the Fourier law in heat transfer.

For a given configuration of a nuclear reactor core it is always possible to force its criticality dividing the neutron production rate by a positive number, λ, obtaining a neutron balance equation. This equation is known as the Lambda modes problem (Henry, 1975),

$$\mathcal{L}\Phi = \frac{1}{\lambda}\mathcal{M}\Phi\,, \tag{1}$$

where $\mathcal{L}$ is the neutron loss differential operator and $\mathcal{M}$ is the neutron production operator.

Therefore this turns the formulation into a differential generalized eigenvalue problem. The fundamental eigenvalue (the one with the largest magnitude) shows the criticality of the reactor core and its corresponding eigenfunction describes the steady state neutron distribution in the core. Next sub-critical eigenvalues and their corresponding eigenfunctions are interesting because they have been successfully used to develop modal methods to integrate the time dependent neutron diffusion equation (Miró et al., 2002). Also the sub-critical modes have been used to classify BWR instabilities (March-Leuba and Rey, 1993), (Ginestar et al., 2011).

Different methods have been proposed to solve the neutron diffusion equation. Core-level codes traditionally use nodal collocation methods (Verdú et al., 1994). In these methods, the diffusion equation is integrated over large homogenized regions known as nodes to obtain a balance with average surface currents and fluxes as unknowns. Modern nodal methods usually rely in the Nodal Expansion Method (NEM) (Singh et al., 2014) and analytical nodal method (ANM) (Hébert, 1987) to overcome the problem in the recalculation of coupling coefficients.

*Corresponding author
Email addresses: anvifer2@etsid.upv.es (A. Vidal-Ferrandiz), rafamou@upv.es (R. Fayez), dginesta@mat.upv.es (D. Ginestar), gverdu@iqn.upv.es (G. Verdú)

Finite elements methods have been also used in rectangular geometry as the PWR and BWR (Hébert, 2008), and hexagonal geometries (González-Pintor et al., 2009). Adaptivity is one of main advantages in the use of the finite method. *h*-adaptable meshes have been proposed to obtain the static configuration of a nuclear reactor core with the use of triangular finite element (Baker et al., 2013) and rectangular elements (Wang and Ragusa, 2009). Also unstructured grid schemes (Theler, 2013) have been developed to solve the problem in non standard geometries.

In this work, an *h-p* finite element method is used to obtain the dominant lambda modes associated with a configuration of a reactor core. This method allows using heterogeneous meshes, and leads to different refinements such as *h*-refinement and *p*-refinement. In *h*-refinement, the finite elements are subdivided in order to reduce the cell size. *p*-refinement increases the polynomial degree of the basic functions used in the expansions to increase the accuracy of the solution. With the *h*-refinement is possible to solve the neutron diffusion equation with cross sections assembly averaged for the majority of fuel assemblies and pin-cell averaged for particular fuel assembly. To obtain pin by pin fluxes in the hottest assembly is crucial to predict adequately the peak cladding temperature.

The *h-p* finite element method used in this work has been implemented using the open source finite elements library Deal.II (Bangerth et al., 2007). With the help of the library, the code proposed is dimension independent and can manage different cell sizes and different types of finite elements (Bangerth and Kayser-Herold, 2009). In order to solve the resulting algebraic eigenvalue problem from the spatial discretization of the Lambda problem the SLEPc library (Hernandez et al., 2005) is used.

The rest of the paper is organized as follows, in Section 2, the discretization used for the Lambda modes problem is presented, describing the finite element formulation. To test the performance of the method, several benchmarks are analysed in Section 3. Finally, the main conclusions of the paper are summarized in Section 4.

2. Discretization of the problem

The Lambda modes equation in the approximation of two groups of energy is considered. This equation can be expressed as (Henry, 1975),

$$\begin{pmatrix} -\vec{\nabla}(D_1\vec{\nabla}) + \Sigma_{a1} + \Sigma_{12} & 0 \\ -\Sigma_{12} & -\vec{\nabla}(D_2\vec{\nabla}) + \Sigma_{a2} \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \frac{1}{\lambda} \begin{pmatrix} \nu\Sigma_{f1} & \nu\Sigma_{f2} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix}, \tag{2}$$

where D_g, $g = 1, 2$ are the diffusion coefficients, Σ_{ag}, Σ_{fg} and Σ_{12} are the macroscopic cross sections of absorption, fission and scattering, respectively. ϕ_1 and ϕ_2 are the fast and thermal neutron fluxes, respectively. The weak formulation of this equation is obtained by pre-multiplying by a test function $\left(\varphi^T = \begin{pmatrix} \varphi_1, & \varphi_2 \end{pmatrix}\right)$ and integrating over the domain Ω, defining the reactor core,

$$\int_\Omega \begin{pmatrix} \varphi_1 & \varphi_2 \end{pmatrix} \begin{pmatrix} -\vec{\nabla}(D_1\vec{\nabla}) + \Sigma_{a1} + \Sigma_{12} & 0 \\ -\Sigma_{12} & -\vec{\nabla}(D_2\vec{\nabla}) + \Sigma_{a1} + \Sigma_{12} \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} \mathrm{d}V = \frac{1}{\lambda} \int_\Omega \begin{pmatrix} \varphi_1 & \varphi_2 \end{pmatrix} \begin{pmatrix} \nu\Sigma_{f1} & \nu\Sigma_{f2} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} \mathrm{d}V \,. \tag{3}$$

The vectorial identity, $\vec{\nabla} \cdot \left(u\vec{\nabla}v\right) = \left(\vec{\nabla}u\right) \cdot \left(\vec{\nabla}v\right) + u\left(\vec{\nabla} \cdot \vec{\nabla}v\right)$, is applied and expression (3) is rewritten as

$$\begin{aligned} &\int_\Omega \vec{\nabla}\varphi_1 D_1 \vec{\nabla}\phi_1 \mathrm{d}V - \int_\Omega \vec{\nabla} \cdot \left(\varphi_1 D_1 \vec{\nabla}\phi_1\right) \mathrm{d}V + \int_\Omega \varphi_1 \left(\Sigma_{a1} + \Sigma_{12}\right) \phi_1 \mathrm{d}V + \\ &+ \int_\Omega \vec{\nabla}\varphi_2 D_2 \vec{\nabla}\phi_2 \mathrm{d}V - \int_\Omega \vec{\nabla} \cdot \left(\varphi_2 D_2 \vec{\nabla}\phi_2\right) \mathrm{d}V + \int_\Omega \varphi_2 \Sigma_{a2} \phi_2 \mathrm{d}V + \\ &- \int_\Omega \varphi_2 \Sigma_{12} \phi_1 \mathrm{d}V = \frac{1}{\lambda} \left(\int_\Omega \varphi_1 \nu\Sigma_{f1} \phi_1 \mathrm{d}V + \int_\Omega \varphi_1 \nu\Sigma_{f2} \phi_2 \mathrm{d}V \right). \end{aligned} \tag{4}$$

Using Gauss Divergence theorem $\left(\int_\Omega \vec{\nabla}\cdot\vec{F}\mathrm{d}V = \int_\Gamma \vec{F}\mathrm{d}\vec{S}\right)$ to eliminate second order derivatives,

$$\begin{aligned}
&\int_\Omega \vec{\nabla}\varphi_1 D_1 \vec{\nabla}\phi_1 \mathrm{d}V - \int_\Gamma \varphi_1 D_1 \vec{\nabla}\phi_1 \mathrm{d}\vec{S} + \int_\Omega \varphi_1 \left(\Sigma_{a1} + \Sigma_{12}\right)\phi_1 \mathrm{d}V + \\
&+ \int_\Omega \vec{\nabla}\varphi_2 D_2 \vec{\nabla}\phi_2 \mathrm{d}V - \int_\Gamma \varphi_2 D_2 \vec{\nabla}\phi_2 \mathrm{d}\vec{S} + \int_\Omega \varphi_2 \Sigma_{a2}\phi_2 \mathrm{d}V + \\
&- \int_\Omega \varphi_2 \Sigma_{12}\phi_1 \mathrm{d}V = \frac{1}{\lambda}\left(\int_\Omega \varphi_1 \nu\Sigma_{f1}\phi_1 \mathrm{d}V + \int_\Omega \varphi_1 \nu\Sigma_{f2}\phi_2 \mathrm{d}V\right)
\end{aligned} \tag{5}$$

is obtained, where Γ is the boundary of the domain defining the reactor.

Finally, the reactor domain Ω is divided into cell subdomains Ω_e ($e = 1, ..., N_t$) where it is assumed that the nuclear cross sections remain constant. Γ_e is also defined as the corresponding subdomain surface which is part of the reactor frontier Γ. Equation (5) is rewritten as

$$\begin{aligned}
&\sum_{e=1}^{N_t}\left(D_1 \int_{\Omega_e} \vec{\nabla}\varphi_1 \vec{\nabla}\phi_1 \mathrm{d}V - D_1 \int_{\Gamma_e} \varphi_1 \vec{\nabla}\phi_1 \mathrm{d}\vec{S} + \left(\Sigma_{a1} + \Sigma_{12}\right)\int_{\Omega_e} \varphi_1\phi_1 \mathrm{d}V \; + \right. \\
&\quad + D_2 \int_{\Omega_e} \vec{\nabla}\varphi_2 \vec{\nabla}\phi_2 \mathrm{d}V - D_2 \int_{\Gamma_e} \varphi_2 \vec{\nabla}\phi_2 \mathrm{d}\vec{S} + \Sigma_{a2}\int_{\Omega_e} \varphi_2\phi_2 \mathrm{d}V + \\
&\quad \left. - \Sigma_{12}\int_{\Omega_e} \varphi_2\phi_1 \mathrm{d}V\right) = \frac{1}{\lambda}\sum_{e=1}^{N_t}\left(\nu\Sigma_{f1}\int_{\Omega_e} \varphi_1\phi_1 \mathrm{d}V + \nu\Sigma_{f2}\int_{\Omega_e} \varphi_1\phi_2 \mathrm{d}V\right).
\end{aligned} \tag{6}$$

It has to be noted that there are several surface integrals over the boundary of the subdomains (Γ_e) that rely on the boundary conditions and that will be studied below (Section 2.1). The solution ϕ_g is approximated through usual trial solution as sum of shape functions N_{gj} multiplied by their corresponding nodal values $\tilde{\phi}_{gj}$.

$$\phi_g \approx \sum_j N_{gj}\tilde{\phi}_{gj}\,. \tag{7}$$

In the same way, a Galerkin method Zienkiewicz et al. (2005) is used assuming that the test function are a finite set of shape functions (Bangerth et al., 2007). Introducing these expressions in (6) and eliminating redundant coefficients to obtain continuous solutions (see, for example, (González-Pintor et al., 2009) for more details) in terms of global coefficients, the procedure leads to an algebraic eigenvalue problem of the form

$$\mathbf{L}\begin{pmatrix}\tilde{\boldsymbol{\phi}}_\mathbf{1}\\ \tilde{\boldsymbol{\phi}}_\mathbf{2}\end{pmatrix} = \frac{1}{\lambda}\,\mathbf{M}\begin{pmatrix}\tilde{\boldsymbol{\phi}}_\mathbf{1}\\ \tilde{\boldsymbol{\phi}}_\mathbf{2}\end{pmatrix}, \tag{8}$$

where the matrices elements are given by

$$\begin{aligned}
L_{ij} = \sum_{e=1}^{N_t}\Bigg(&D_1 \int_{\Omega_e} \vec{\nabla}N_{1i}\vec{\nabla}N_{1j}\mathrm{d}V - D_1 \int_{\Gamma_e} N_{1i}\vec{\nabla}N_{1j}\mathrm{d}\vec{S} + \left(\Sigma_{a1} + \Sigma_{12}\right)\int_{\Omega_e} N_{1i}N_{1j}\mathrm{d}V \; + \\
&+D_2 \int_{\Omega_e} \vec{\nabla}N_{2i}\vec{\nabla}N_{2j}\mathrm{d}V - D_2 \int_{\Gamma_e} N_{2i}\vec{\nabla}N_{2j}\mathrm{d}\vec{S} + \Sigma_{a2}\int_{\Omega_e} N_{2i}N_{2j}\mathrm{d}V - \Sigma_{12}\int_{\Omega_e} N_{2i}N_{1j}\mathrm{d}V\Bigg),
\end{aligned} \tag{9}$$

$$M_{ij} = \sum_{e=1}^{N_t}\left(\nu\Sigma_{f1}\int_{\Omega_e} N_{1i}N_{1j}\mathrm{d}V + \nu\Sigma_{f2}\int_{\Omega_e} N_{1i}N_{2j}\mathrm{d}V\right). \tag{10}$$

These integrals only have non-zero value when the degrees of freedom i and j collide inside the same cell, therefore highly sparse global matrices are obtained.

2.1. Boundary conditions

Implemented boundary conditions: are zero-flux at the frontier, zero-current boundary condition and Albedo boundary conditions. These last conditions are mixed boundary conditions of the form

$$\vec{n}\vec{\nabla}\phi_g(\vec{x}) + \frac{1}{D_g}\frac{1-\beta}{2(1+\beta)}\phi_g(\vec{x}) = 0\,, \qquad \vec{x} \in \Gamma\,. \tag{11}$$

If there are zero-flux boundary conditions, the shape functions of the corresponding nodes are fixed to zero. Thus, the number of degrees of freedom (DoF) of the problem is reduced because the nodes have their flux restricted. On the other hand, if the boundary conditions are zero-current conditions the integral surface terms are equal to zero and the finite element formulation takes care of these conditions without restrictions in the nodes. Albedo boundary conditions are treated pre-multiplying the condition by the test function and integrating over the surface of the domain,

$$\begin{aligned} \int_\Gamma \varphi_g \left(D_g \vec{\nabla}\phi_g + \frac{1}{2}\frac{1-\beta}{1+\beta}\phi_g \right) \mathrm{d}\vec{S}\,, \\ -D_g \int_\Gamma \varphi_g \vec{\nabla}\phi_g \mathrm{d}\vec{S} = \frac{1}{2}\frac{1-\beta}{1+\beta}\int_\Gamma \varphi_g \phi_g \mathrm{d}\vec{S}\,. \end{aligned} \tag{12}$$

Hence, the surface terms that appear in Equation (8) are substituted by,

$$\sum_{e=1}^{N_t} -D_g \int_{\Gamma_e} N_{gi}\vec{\nabla}N_{gj}\mathrm{d}\vec{S} = \sum_{e=1}^{N_t} \frac{1}{2}\frac{1-\beta}{1+\beta}\int_{\Gamma_e} N_{gi}N_g \mathrm{d}\vec{S}\,. \tag{13}$$

2.2. Reference element

As it has been already mentioned, the whole reactor domain is discretized into cells. In order to define these subdomains always over the same reference cell an affine mapping is used to map each physical element to the reference element. An example for a bidimensional cell is shown in Figure 1.

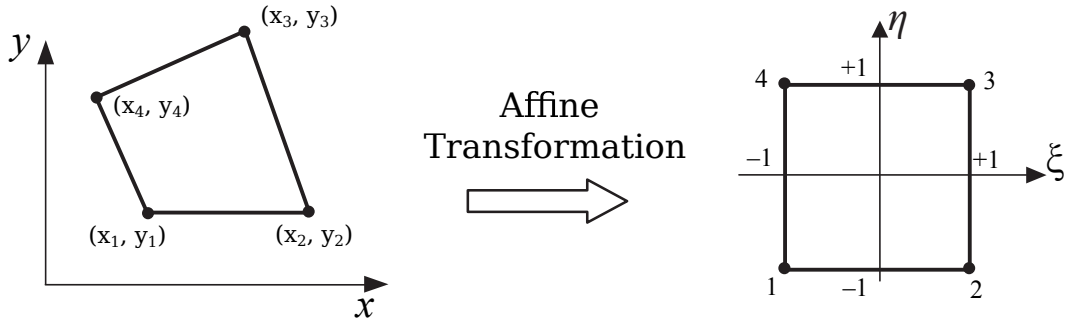


Figure 1: Affine transformation used.

This changes of variables relates physical coordinates (x, y), with the coordinates of the reference domain (ξ, η) and it is given by

$$\begin{aligned} x(\xi,\eta) = \frac{1}{4}\left((1-\xi)(1-\eta)x_1 + (1-\xi)(1+\eta)x_2 + (1+\xi)(1-\eta)x_3 + (1-\xi)(1+\eta)x_4\right)\,, \\ y(\xi,\eta) = \frac{1}{4}\left((1-\xi)(1-\eta)y_1 + (1-\xi)(1+\eta)y_2 + \;.(1+\xi)(1-\eta)y_3 + (1-\xi)(1+\eta)y_4\right)\,. \end{aligned} \tag{14}$$

This affine mapping helps to compute the integrals defining the matrix elements taking into account the Jacobian of the transformation $|\mathbf{J}^\mathbf{e}|$.

$$\mathrm{d}V = \mathrm{d}x\mathrm{d}y \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{vmatrix} \mathrm{d}\xi\mathrm{d}\eta = |\mathbf{J}^\mathbf{e}|\mathrm{d}\xi\mathrm{d}\eta\,. \tag{15}$$

2.3. Lagrange finite elements

For simplicity, Lagrange finite elements (Zienkiewicz et al., 2005) are used. These elements have their nodes distributed forming a regular mesh over the cell. Their shape functions are defined with *Lagrange polynomials* for every dimension. These polynomials have a value of unity at the corresponding nodal point and zero at the other nodes and they satisfy all inter-element continuity conditions. Lagrange polynomials are defined as

$$l_I^p(\xi) = \frac{(\xi-\xi_1)\dots(\xi-\xi_{I-1})(\xi-\xi_{I+1})\dots(\xi-\xi_{p+1})}{(\xi_I-\xi_1)\dots(\xi_I-\xi_{I-1})(\xi_I-\xi_{I+1})\dots(\xi_I-x_{p+1})} = \prod_{\substack{k=0\\k\neq I}}^{p+1} \frac{\xi-\xi_k}{\xi_I-\xi_k}\,, \tag{16}$$

where p is the polynomial degree of the expansion, and ξ_i is the position of every node in the element. Multidimensional versions of these elements are obtained by tensorial product of their elements. Thus, in two coordinates, if the node is labelled by its column and row number I, J.

$$N_{I,J}(\xi,\eta) = l_I^p(\xi) l_J^p(\eta)\,. \tag{17}$$

Figure 2 shows the shape functions of some one-dimensional Lagrange elements and an example of these shape functions in a bidimensional element is displayed in Figure 3. Finally it should be noted that for the integration of

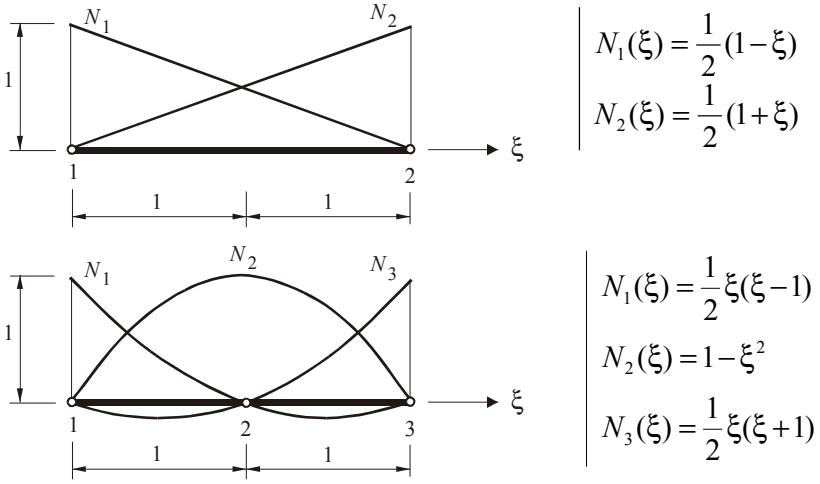


Figure 2: Example of unidimensional shape functions used, linear and quadratic.

the weak formulation a Gauss quadrature is used in each cell with the same number of points as the number of nodes of the element. This ensures an exact integration inside the approximation of polynomial shape functions.

2.4. Eigenvalue solver and postprocess

With the finite element method exposed above the Lambda modes problem is approximated by a generalized algebraic eigenvalue problem of Equation (8) with the following block structure

$$\begin{pmatrix} \mathbf{L_{11}} & 0 \\ -\mathbf{L_{21}} & \mathbf{L_{22}} \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{\phi}}_\mathbf{1} \\ \tilde{\boldsymbol{\phi}}_\mathbf{2} \end{pmatrix} = \frac{1}{\lambda} \begin{pmatrix} \mathbf{M_{11}} & \mathbf{M_{12}} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{\phi}}_\mathbf{1} \\ \tilde{\boldsymbol{\phi}}_\mathbf{2} \end{pmatrix}. \tag{18}$$

To solve this problem a Krylov-Schur method (Stewart, 2002) is used from the library SLEPc Hernandez et al. (2005). First the generalized problem is reduced to an ordinary eigenvalue problem,

$$\mathbf{L_{11}^{-1}} \left(\mathbf{M_{11}} + \mathbf{M_{12}} \mathbf{L_{22}^{-1}} \mathbf{L_{21}} \right) \tilde{\boldsymbol{\phi}}_\mathbf{1} = \lambda \tilde{\boldsymbol{\phi}}_\mathbf{1}\,, \tag{19}$$

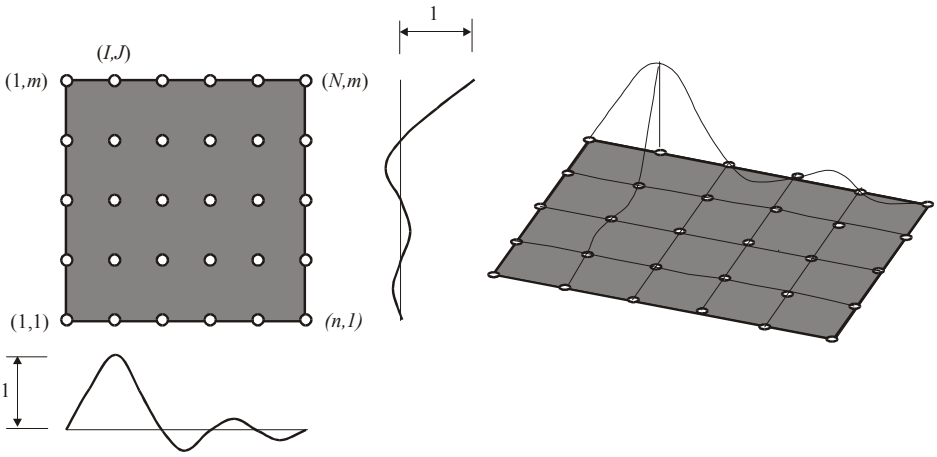


Figure 3: A typical shape function for a 2D Lagrangian element, (I=2, J=5, p=4).

which is solved for the n dominant eigenvalues and their corresponding eigenvectors. In this way, for each matrix-vector product it is necessary to solve two linear systems associated with the L_{11} and L_{22}, to avoid the calculation of their inverse matrices. These systems are solved by means of an iterative scheme as the preconditioned GMRES method (Morgan, 2002). Particularly, a Cuthill-McKee reordering (Cuthill and McKee, 1969) is performed to reduce the bandwidth of the matrices, together with an incomplete LU factorization of the matrices is used for the preconditioning.

Once the fluxes are obtained other practical magnitudes are computed as the neutronic power that is defined as a weighted sum of the neutron fluxes

$$P = \nu\Sigma_{f1}|\phi_1| + \nu\Sigma_{f2}|\phi_2| \, . \tag{20}$$

The eigenvectors should be normalized through some criteria. The most usual one is to fix the mean power productions to 1. It is needed to introduce the absolute value of the fluxes because this definition is extended to the subcritical eigenfunctions.

$$\bar{P} = \frac{1}{V_t} \int_{V_t} \left(\nu\Sigma_{f1}|\phi_1| + \nu\Sigma_{f2}|\phi_2| \right) \mathrm{d}V = 1 \, . \tag{21}$$

2.5. Refinement and error estimator

Once the problem is solved, it is convenient to estimate if the solution is obtained has enough accuracy and if not, to refine the mesh accordingly. In this way, two types of refinements are considered, an uniform refinement, where all cell are refined, and an adaptive refinement, where only part of the cells are refined. To choose which cells are refined a modified versions of the error estimator proposed by Kelly et al.(De S. R. Gago et al., 1983) generalized for a non-constant diffusion coefficients (Wang and Ragusa, 2009) is used,

$$\eta_e^2 = \frac{h}{24}\Sigma_{f1} \int_{\Gamma_i} \left(D_1 \vec{\nabla}\phi_1 \right) \mathrm{d}\vec{S} + \Sigma_{f2} \int_{\Gamma_i} \left(D_2 \vec{\nabla}\phi_2 \right) \mathrm{d}\vec{S} \, , \tag{22}$$

where Γ_i is all interior boundaries of the element and h is the adimensional cell size. In other words, we are using the jump in the net current multiplied by the fission cross sections as an error estimator. Even though, this is an error estimator for the Poisson equation (for example $\nabla^2\varphi = f$), this indicator is widely used as a heuristic refinement indicator and it is considered a good choice in the absence of actual estimators for a particular equation (Bangerth and Kayser-Herold, 2009).

3. Numerical Results

To study the performance of the *h-p* finite element method to determine the Lambda modes of a nuclear reactor, three different benchmark problems have been considered. To compare the performance of the method using different types of meshes, refinement sizes and strategies different errors have been employed:

$$
\begin{aligned}
&\text{the relative power error,} && \varepsilon_i = \frac{|P_i - P_i^*|}{|P_i|} \,, \\
&\text{the mean relative error,} && \bar{\varepsilon} = \frac{1}{V_t} \sum_i \varepsilon_i V_i \,, \\
&\text{the maximum absolute error,} && \varepsilon_{\max} = \max_i |P_i - P_i^*| \,, \\
&\text{the relative power peaking error,} && RPP = \frac{\max_i |P_i| - \max_i |P_i^*|}{\max_i |P_i|} \,, \\
&\text{and the eigenvalue error (pcm),} && \epsilon_{eig} = 10^5 \left(\frac{|\lambda_i - \lambda_i^*|}{\lambda_i} \right) \,,
\end{aligned}
$$

where P_i and P_i^* are the reference power and the obtained power in the i-th cell (cell averages), respectively. V_i is the volume of the cell and V_t is the total volume of the reactor, λ_i is the reference eigenvalue and λ_i^* is the computed eigenvalue. To validate the results of the implemented code first a 2D homogeneous reactor has been studied, since an analytical solution can be found for this problem. Also more realistic reactors, as the BIBLIS 2D reactor and the IAEA 3D reactor have been studied. The code has been written in C++ and executed in a computer with an Intel®i3-3220 @ 3.30GHz processor with 4 Gb of RAM running Ubuntu GNU/Linux 12.10. The number of eigenvalues requested has been set to 4 with a relative tolerance of 10^{-7} in all the examples.

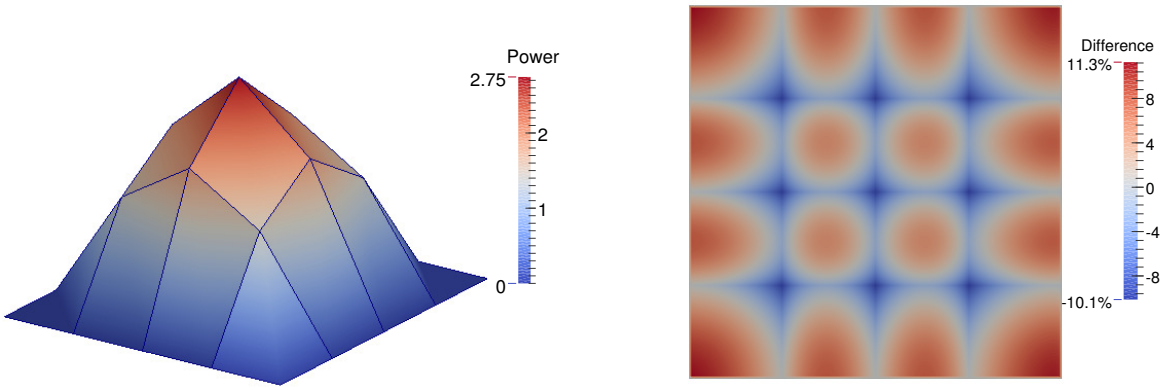
3.1. Homogeneous Reactor

The simplest theoretical reactor is one consisting of a $2D$ rectangular homogeneous material. Even though this problem is completely theoretical, it is relevant because it can be solved analytically for all its eigenvalues. The analytical solution is developed in Appendix A. The material cross sections for the $(L_1 \times L_2)$ rectangular reactor are shown in Table 1.

Table 1: Geometry and cross section values for the homogeneous reactor.

$\mathbf{L_1}$ cm	$\mathbf{L_2}$ cm	$\mathbf{D_1}$ (cm^2)	$\mathbf{D_2}$ (cm^2)	$\boldsymbol{\Sigma_{a1}}$ ($1/cm^2$)	$\boldsymbol{\Sigma_{a2}}$ ($1/cm^2$)	$\boldsymbol{\Sigma_{12}}$ ($1/cm^2$)	$\boldsymbol{\nu\Sigma_{f1}}$ ($1/cm^2$)	$\boldsymbol{\nu\Sigma_{f2}}$ ($1/cm^2$)
40	40	1.32	0.2772	0.0026562	0.071596	0.023106	0.0074527	0.13236

Table 2 shows the eigenvalue results using different number of cells and different degrees of polynomials. The power distribution for the dominant eigenvalue and zero-flux boundary conditions using a very coarse mesh (16 cells $p = 1$) is shown in Figure 4a and the relative power error distribution, ε_i, is shown in Figure 4b. It should be noted that the maximum difference with the analytical solution is up to 11% but the averaged relative error ($\bar{\varepsilon}$) is only about 3.04%. Figure 5 compares the power distribution along the center line $y = 20$ cm for the first two dominant eigenvalues. It is observed a good agreement with the analytical solutions when the number of cells or the polynomial expansion degrees are increased.



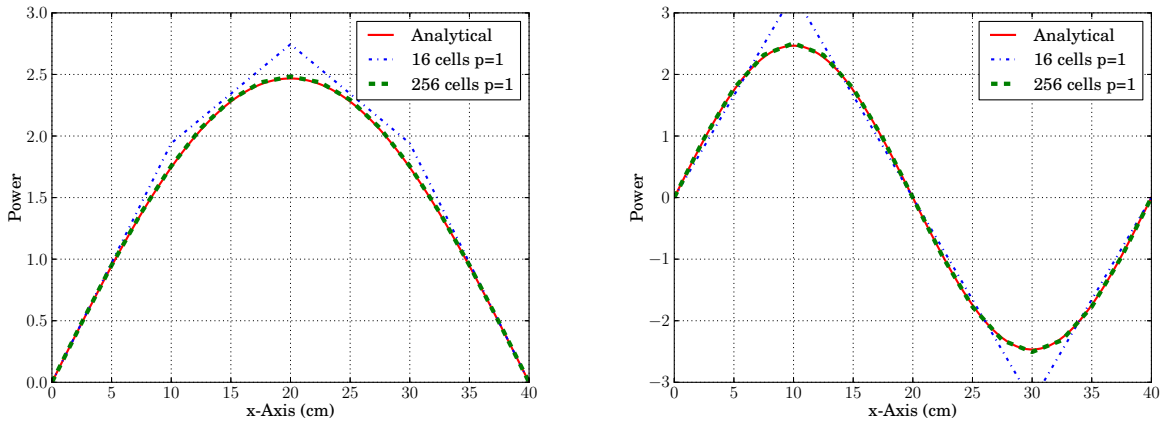
(a) Power distribution.

(b) Error distribution.

Figure 4: Fundamental mode power distribution and its error distribution for homogeneous reactor with zero-flux boundary conditions.

Table 2: Eigenvalue results for the homogeneous reactor using uniform meshes.

Number of of cells	**Degree of FE**	**Number of DoF**	First Mode		Second Mode	
			λ_1	ϵ_{eig} **(pcm)**	$\lambda 2$	ϵ_{eig} **(pcm)**
16	1	25	1.12178	2186	0.60700	11536
256	1	289	1.14528	137	0.68092	764
4096	1	4225	1.14675	8.7	0.68587	42
16	2	81	1.14660	22	0.68322	429
256	2	1089	1.14685	0.3	0.68615	1.7
16	3	169	1.14685	0.2	0.68610	7.7
Analytical			1.14685		0.68616	



(a) 1st Mode.

(b) 2nd Mode.

Figure 5: Power distribution along the line $y = 20\ cm$ for the homogeneous reactor.

3.2. BIBLIS 2D Reactor

A more realistic 2D example is chosen, the BIBLIS 2D benchmark (Müller and Weiss, 1991). This is a classical two-group neutron diffusion problem taken as a benchmark for different numerical codes (Hébert, 1985). It has 257 different assemblies including 64 cells modelling the reflector. The definition of the 8 different materials and their cross sections are defined in Figure 6 and Table 3.

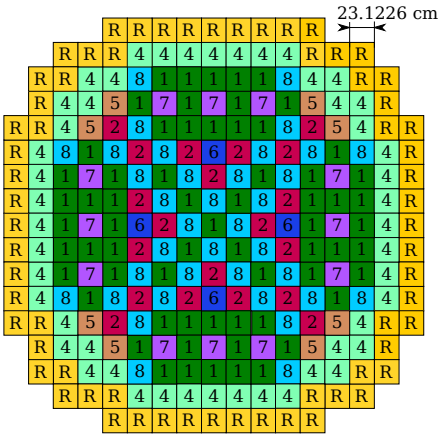


Figure 6: BIBLIS Material definition.

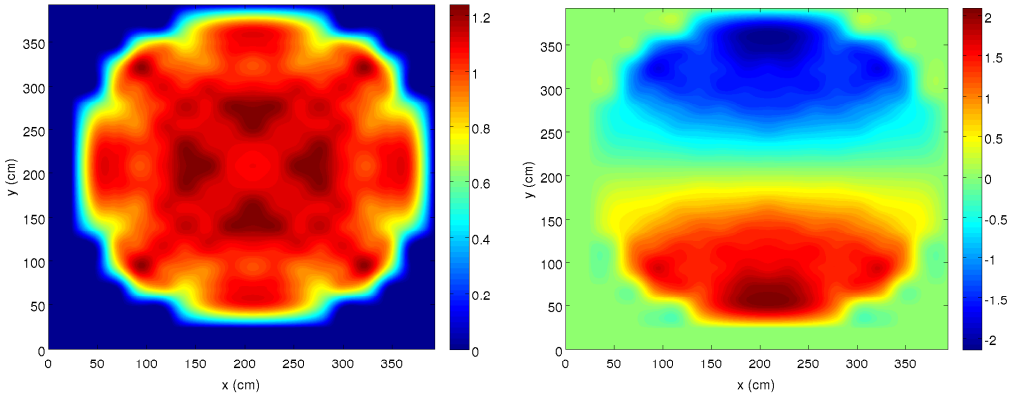
Table 3: Macroscopic cross sections of the BIBLIS reactor.

Material	$\mathbf{D_1}$ (cm^2)	$\mathbf{D_2}$ (cm^2)	$\mathbf{\Sigma_{a1}}$ ($1/cm^2$)	$\mathbf{\Sigma_{a2}}$ ($1/cm^2$)	$\mathbf{\Sigma_{12}}$ ($1/cm^2$)	$\mathbf{\nu\Sigma_{f1}}$ ($1/cm^2$)	$\mathbf{\nu\Sigma_{f2}}$ ($1/cm^2$)
1	1.4360	0.3635	0.0095042	0.075058	0.017754	0.0058708	0.096067
2	1.4366	0.3636	0.0096785	0.078436	0.017621	0.0061908	0.103580
R	1.3200	0.2772	0.0026562	0.071596	0.023106	0.0	0.0
4	1.4389	0.3638	0.0103630	0.091408	0.017101	0.0074527	0.132360
5	1.4381	0.3665	0.0100030	0.084828	0.017290	0.0061908	0.103580
6	1.4385	0.3665	0.0101320	0.087314	0.017192	0.0064285	0.109110
7	1.4389	0.3679	0.0101650	0.088024	0.017125	0.0061908	0.103580
8	1.4393	0.3680	0.0102940	0.090510	0.017027	0.0064285	0.109110

Figure 7 shows the neutronic power distribution for the four dominant modes of this reactor. Tables 4 and 6 display the eigenvalue results for uniform refined meshes and h-refined meshes. Tables 5 and 7 show the power distribution errors obtained using different meshes and different polynomial degrees together with the CPU time needed to compute 4 eigenvalues. The reference values for the first mode are extracted from (Müller and Weiss, 1991). For the second mode the reference values are extracted from the most converged solution (16448 cells with $p = 4$). Figure 8 displays the error distribution (ε_i) for 257 elements with $p = 3$ for the first mode. As example of adaptive refinemen, Figure 9, displays the meshes generated in 6 different iterations of the code using the error estimator (22). It is observed that the code refines the cells with the highest error that are pointed out correctly by the error estimator. Also it can be noted that the algorithm refines the cells near the locations where the material changes, particularly in the last iterations.

In Figures 10 and 11, the mean relative error for the two dominant eigenvalues is displayed as a function of the execution time for different meshes and degrees of finite elements. From these figures, it can be seen that the errors follow a typical exponential convergence with the computation time. Also, it can be concluded that the errors in the power distribution do not depend on which eigenvalue is being calculated. These Figures and Tables show that

the local or uniform h-refinement is not a better strategy than increasing the polynomial uniformly because of the smoothness in the fluxes solutions and the computational cost of the evaluation of the error estimator. For example, a coarse mesh with 257 elements with $p = 3$ gives better results ($\bar{\varepsilon}_1 = 1.29\%$ and $\epsilon_{eig1} = 8$ pcm) than a h-refined mesh with 4280 cells and $p = 1$ ($\bar{\varepsilon}_1 = 1.87\%$ and $\epsilon_{eig1} = 65$ pcm), even though the cubic approximation is faster to be solved (0.22 s against 0.51 s). In Figure 12, a convergence graph is shown for the relative power peaking error Wang and Ragusa (2009) versus the time of execution. Although, this is an error of importance in nuclear engineering the relative power peaking error shows a behaviour which is less smooth than the mean error between meshes and different polynomial degrees of finite elements.



(a) 1st Mode.

(b) 2nd Mode.

(c) 3rd Mode.

(d) 4th Mode.

Figure 7: Power distribution for the four dominant modes of the BIBLIS reactor.

Table 4: Eigenvalue results with uniform refined meshes in BIBLIS Reactor .

Number of of cells	**Degree of FE**	**Number of DoF**	**CPU time (s)**	**First Mode**		**Second Mode**	
				λ_1	ϵ_{eig} **(pcm)**	λ_2	ϵ_{eig} **(pcm)**
257	1	292	0.02	1.021792	322	1.013800	410
4112	1	4249	0.28	1.024903	19.1	1.017800	17.3
65792	1	66337	17.3	1.025087	1.04	1.017969	0.72
257	2	1097	0.08	1.025246	14.6	1.018319	34.0
4112	2	37417	2.04	1.025128	1.56	1.018085	3.52
16448	2	66337	17.4	1.025098	0.16	1.017999	0.35
257	3	2416	0.22	1.025180	8.08	1.018218	23.7
4112	3	37417	8.12	1.025099	0.09	1.017978	0.210
Reference				1.025099		1.017986	

Table 5: Power distribution errors with uniform refined meshes in BIBLIS Reactor.

Number of cells	**Degree of FE**	**Number of DoF**	**CPU time (s)**	**1st Mode**		**2nd Mode**	
				$\bar{\varepsilon}$ **(%)**	$\varepsilon_{\max}$	$\bar{\varepsilon}$ **(%)**	$\varepsilon_{\max}$
257	1	292	0.02	7.03	1.7e-1	6.40	2.5e-1
4112	1	4249	0.28	0.55	5.5e-2	0.56	1.4e-2
65792	1	66337	17.3	0.04	8.4e-4	0.05	4.7e-2
257	2	1097	0.08	1.72	3.7e-2	1.69	5.5e-2
4112	2	37417	2.04	0.16	3.6e-3	0.14	4.0e-3
16448	2	66337	17.4	0.02	3.7e-4	0.01	3.7e-4
257	3	2416	0.22	1.29	3.0e-2	1.18	5.6e-2
4112	3	37417	8.12	0.01	1.8e-4	0.01	1.9e-4

Table 6: Eigenvalue results with *h*-refined meshes in BIBLIS Reactor .

Number of of cells	**Degree of FE**	**Number of DoF**	**CPU time (s)**	**First Mode**		**Second Mode**	
				λ_1	ϵ_{eig} **(pcm)**	λ_2	ϵ_{eig} **(pcm)**
644	1	732	0.06	1.022184	284	1.014401	351
4280	1	4736	0.51	1.024426	65.4	1.017292	67.3
28613	1	30982	8.71	1.024993	10.1	1.017858	11.7
72866	1	68646	34.4	1.025013	8.29	1.017858	11.8
647	2	2861	0.30	1.025125	2.64	1.018116	13.7
4064	2	18139	4.82	1.025098	0.05	1.017999	2.15
10187	2	57881	17.9	1.025095	0.24	1.017980	0.31
Reference				1.025099		1.017986	

Table 7: Power distribution errors with *h*-refined meshes in BIBLIS Reactor.

Number of cells	**Degree of FE**	**Number of DoF**	**CPU time (s)**	**1st Mode**		**2nd Mode**	
				$\bar{\varepsilon}$ **(%)**	$\varepsilon_{\max}$	$\bar{\varepsilon}$ **(%)**	$\varepsilon_{\max}$
644	1	732	0.06	4.56	9.2e-2	4.58	1.3e-1
4280	1	4736	0.51	1.87	9.1e-2	1.65	4.8e-2
28613	1	30982	8.71	0.21	2.2e-3	0.19	1.9e-1
72866	1	68646	34.4	0.05	2.8e-3	0.05	1.5e-3
647	2	2861	0.30	1.09	2.7e-2	1.05	4.1e-2
4064	2	18139	4.82	0.23	8.2e-3	0.19	7.9e-3
10187	2	45301	17.9	0.04	1.4e-3	0.03	3.2e-2

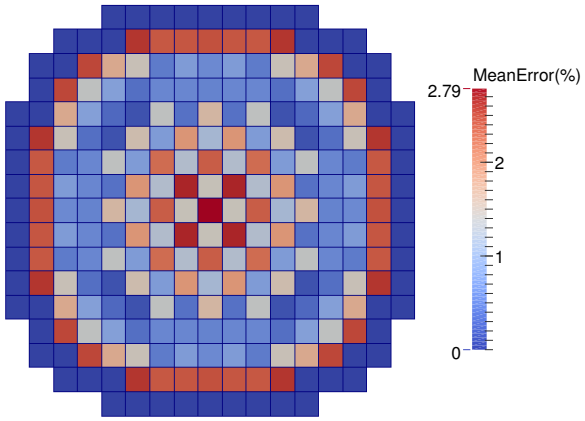


Figure 8: Relative error distribution using a mesh with 257 elements $p = 3$ in 1st mode of BIBLIS Reactor.

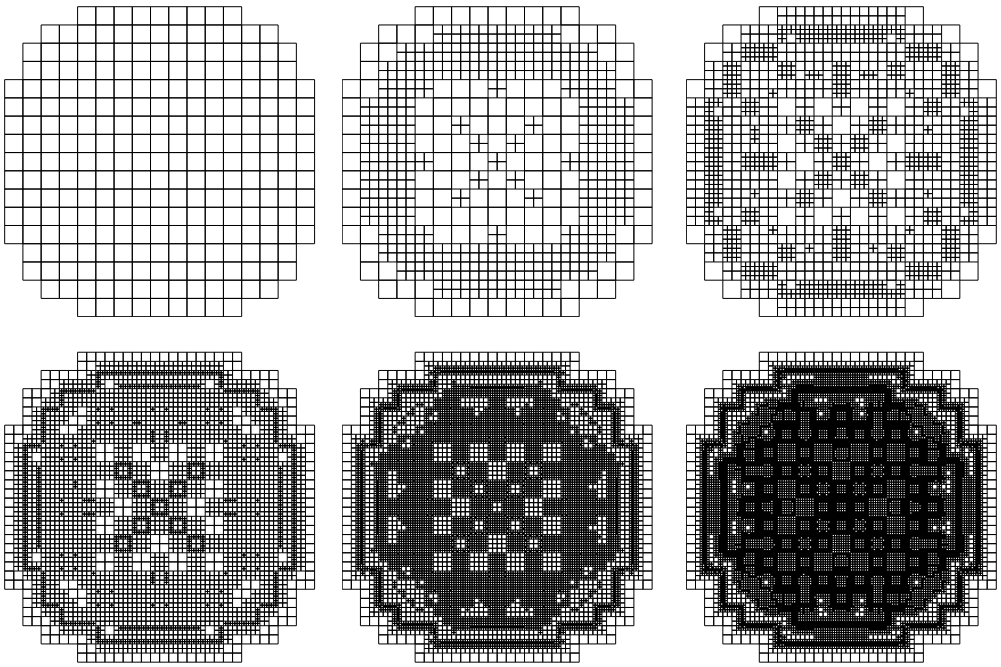


Figure 9: Example of 6 iterations in the refination process, refining for the 1st ode.

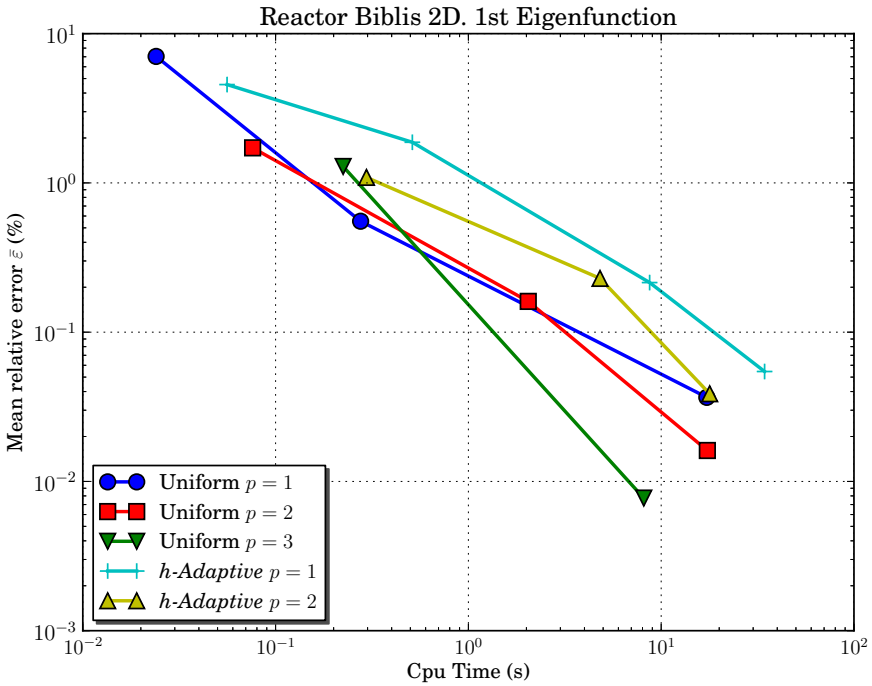


Figure 10: Mean error against CPU time for 1st eigenvalue of the BIBLIS Reactor.

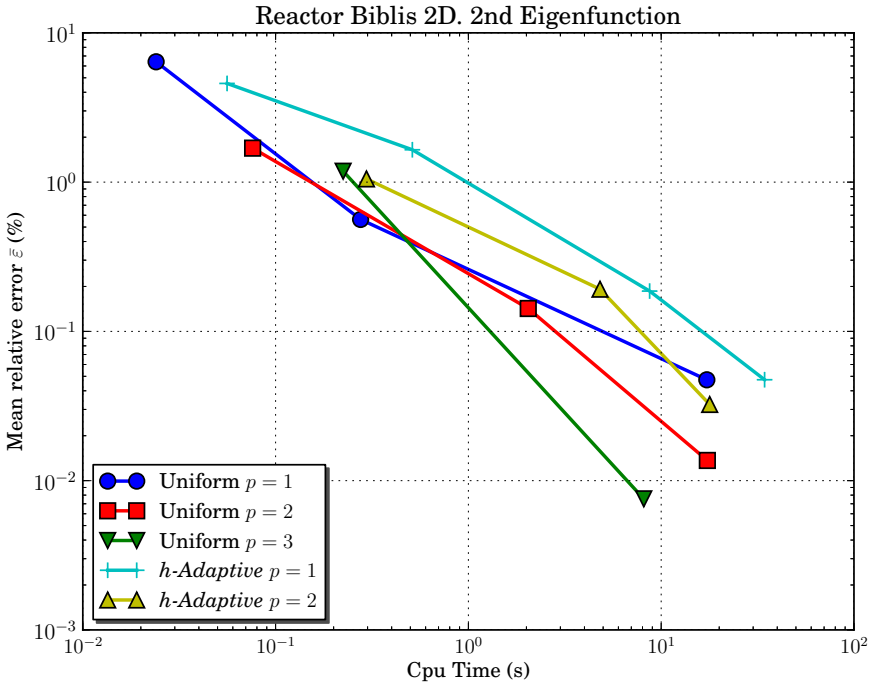


Figure 11: Mean error against CPU time for 2nd eigenvalue of the BIBLIS Reactor.

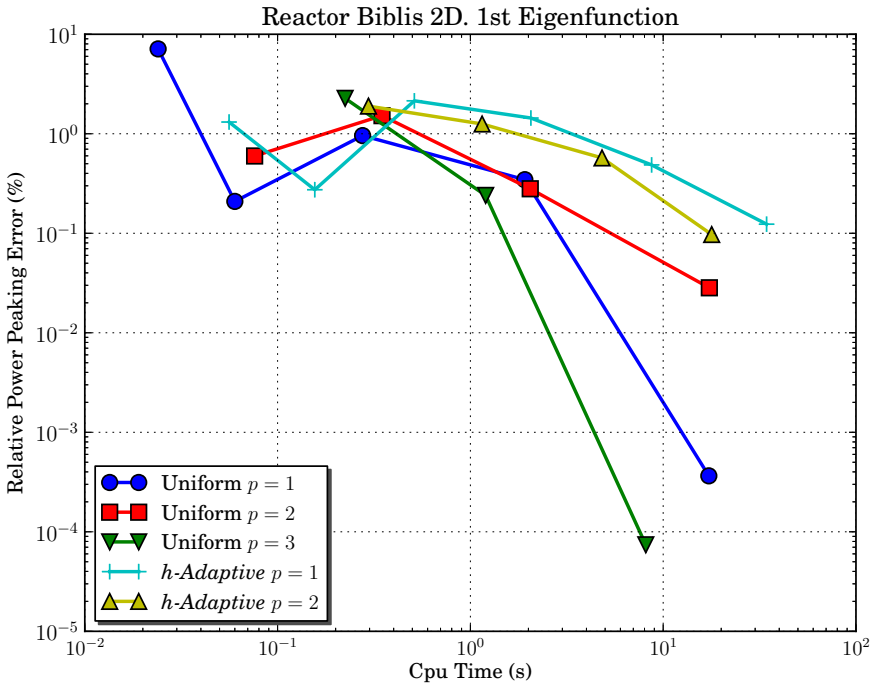


Figure 12: Relative power peaking error against CPU time for 1st eigenvalue of the BIBLIS Reactor.

3.3. IAEA 3D Reactor

The IAEA PWR 3D benchmark (American Nuclear Society, 1977) has been solved as an example of a 3D reactor. The core is composed by 241 rod assemblies including 64 assemblies modelling the reflector. The definition of this reactor is exposed in Figure 13 and the cross sections of the different materials are shown in Table 8. The reference values for the first mode are extracted from (American Nuclear Society, 1977) and for the second mode they are extracted from the most converged solution (36632 elements with $p = 3$).

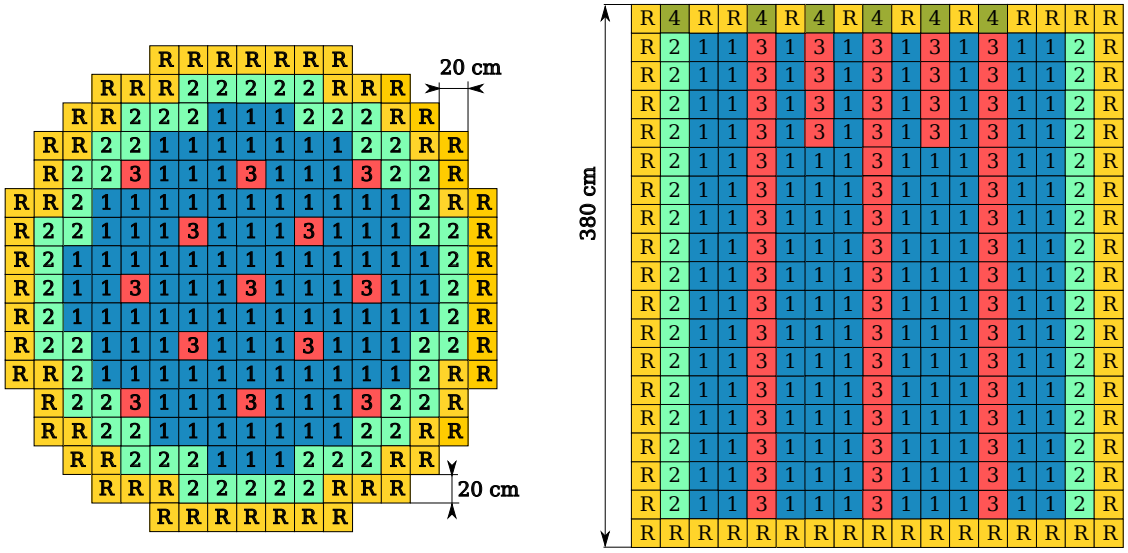


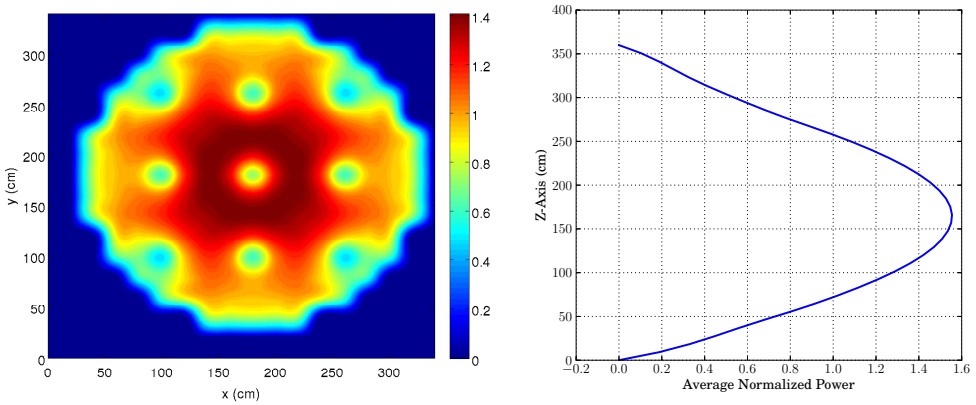
Figure 13: Geometry and material definition of the IAEA 3D Reactor.

Table 8: Macroscopic cross sections of the IAEA 3D reactor.

Material	$\mathbf{D_1}$ (cm^2)	$\mathbf{D_2}$ (cm^2)	$\mathbf{\Sigma_{a1}}$ ($1/cm^2$)	$\mathbf{\Sigma_{a2}}$ ($1/cm^2$)	$\mathbf{\Sigma_{12}}$ ($1/cm^2$)	$\mathbf{\nu\Sigma_{f1}}$ ($1/cm^2$)	$\mathbf{\nu\Sigma_{f2}}$ ($1/cm^2$)
1	1.500	0.400	0.010	0.085	0.020	0.000	0.135
2	1.500	0.400	0.010	0.130	0.020	0.000	0.135
3	1.500	0.400	0.010	0.080	0.020	0.000	0.135
4	2.000	0.300	0.000	0.055	0.040	0.000	0.000
R	2.000	0.300	0.000	0.010	0.040	0.000	0.000

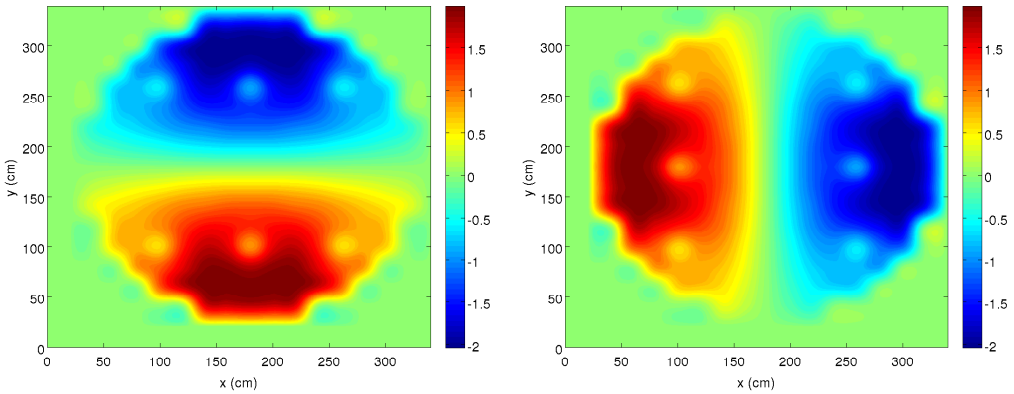
Figure 14 shows the averaged power distribution for the three dominant modes of this reactor. Tables 9 and 11 display the eigenvalue results for uniform refined meshes and h-refined meshes. Also, Tables 9 and 11 show the power distribution errors for different computation parameters. Figures 15 and 16 display the convergence graphs (mean cell error against time of execution) for the two dominant eigenvalues. To make hardware independent comparisons of the algorithms, the mean error against the number of the degrees of freedom (DoF) is represented in Figure 17. Figure 18 shows the computation times against the number of degrees of freedom in order to confirm the relationship. These figures and tables show that also fo this 3D problem the local h-refinement is not a better strategy than increasing the polynomial degree uniformly because of the smoothness in the solutions for the fluxes and the computational cost of evaluating the error estimator. For example, a coarse mesh with 4579 cells with $p = 3$ gives better results ($\bar{\varepsilon} = 0.79\%$ and $\epsilon_{eig} = 8.1$ pcm) than a h-refined mesh with 20609 cells and $p = 2$ ($\bar{\varepsilon} = 1.54\%$ and $\epsilon_{eig1} = 87$ pcm), even though the first one is faster (97.5 s against 108.2 s). Also, it is observed that the errors behaviour for the first eigenvalue and its corresponding eigenvector are similar to the errors for the second eigenvector. It can be concluded that the best

strategy in this case, with a moderate computational cost, is to use finite elements with cubic polynomials in a coarse mesh.



(a) 1st Mode (average plane).

(b) 1st Mode (profile).



(c) 2nd Mode (average plane).

(d) 3r Mode (average plane).

Figure 14: Power distribution for the three dominant modes of the IAEA 3D reactor.

Table 9: Eigenvalue results with uniform refined meshes in IAEA 3D Reactor .

Number of of cells	**Degree of FE**	**Number of DoF**	**CPU time (s)**	**First Mode**		**Second Mode**	
				λ_1	ϵ_{eig} **(pcm)**	λ_2	ϵ_{eig} **(pcm)**
4579	1	5520	1.12	1.050388	2069	1.046274	2866
36632	1	40287	8.76	1.030602	146	1.020614	343
293056	1	307461	107.4	1.029246	14.6	1.017725	60
4579	2	40287	15.00	1.030173	104	1.019336	218
36632	2	307461	169.6	1.029077	1.8	1.017228	11
4579	3	131776	97.48	1.029013	8.1	1.017170	5.2
Reference				1.029096		1.0171168	

Table 10: Power distribution errors with uniform refined meshes in IAEA Reactor.

Number of cells	**Degree of FE**	**Number of DoF**	**CPU time (s)**	**1st Mode**		**2nd Mode**	
				$\bar{\varepsilon}$ **(%)**	$\varepsilon_{\max}$	$\bar{\varepsilon}$ **(%)**	$\varepsilon_{\max}$
4579	1	5520	1.12	71.19	2.1e+0	56.41	3.4e+0
36632	1	40287	8.76	11.43	3.4e-1	10.68	4.6e-1
293056	1	307461	107.4	1.76	5.9e-2	2.01	7.7e-2
4579	2	40287	15.00	5.78	1.8e-1	5.80	2.1e-1
36632	2	307461	169.6	0.70	9.2e-3	0.92	3.7e-2
4579	3	131776	97.48	0.79	2.5e-1	0.79	7.7e-2

Table 11: Eigenvalue results with *h*-refined meshes in IAEA 3D Reactor .

Number of of cells	**Degree of FE**	**Number of DoF**	**CPU time (s)**	**First Mode**		**Second Mode**	
				λ_1	ϵ_{eig} **(pcm)**	λ_2	ϵ_{eig} **(pcm)**
20609	1	24698	5.09	1.029612	51	1.020190	302
103818	1	125214	36.1	1.028816	28	1.017521	35
20609	2	193466	108.2	1.029923	87	1.018781	161
Reference				1.029096		1.017170	

Table 12: Power distribution errors with *h*-refined refined meshes in IAEA 3D Reactor.

Number of cells	**Degree of FE**	**Number of DoF**	**CPU time (s)**	**1st Mode**		**2nd Mode**	
				$\bar{\varepsilon}$ **(%)**	$\varepsilon_{\max}$	$\bar{\varepsilon}$ **(%)**	$\varepsilon_{\max}$
20609	1	24698	5.09	25.2	5.6e-1	23.7	6.7e-1
103818	1	125214	36.1	8.22	1.4e-1	9.26	3.2e-1
20609	2	193466	108.2	1.54	2.9e-2	1.64	1.6e+0

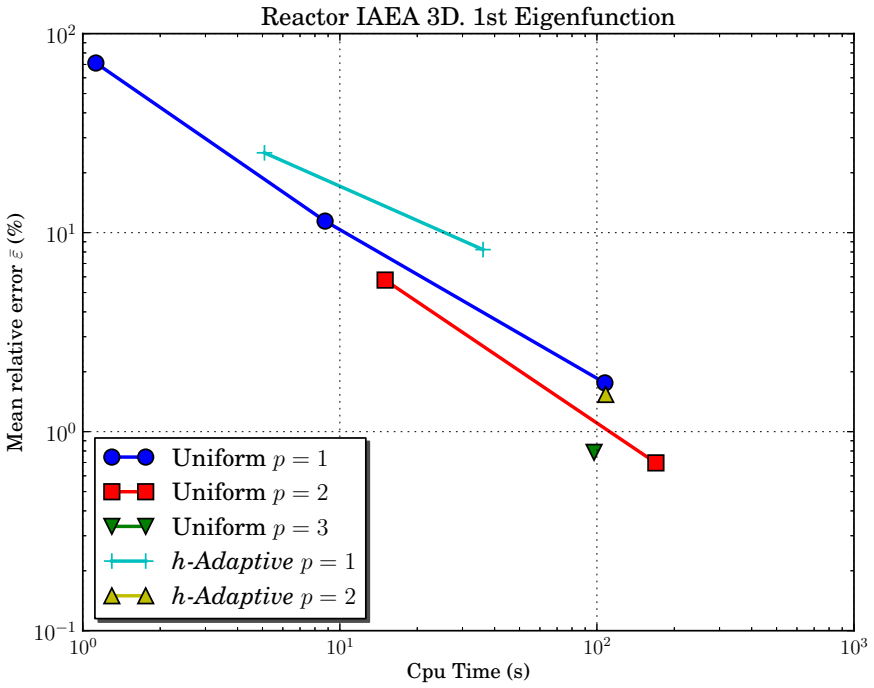


Figure 15: Mean error against CPU time for 1st mode of the IAEA 3D Reactor.

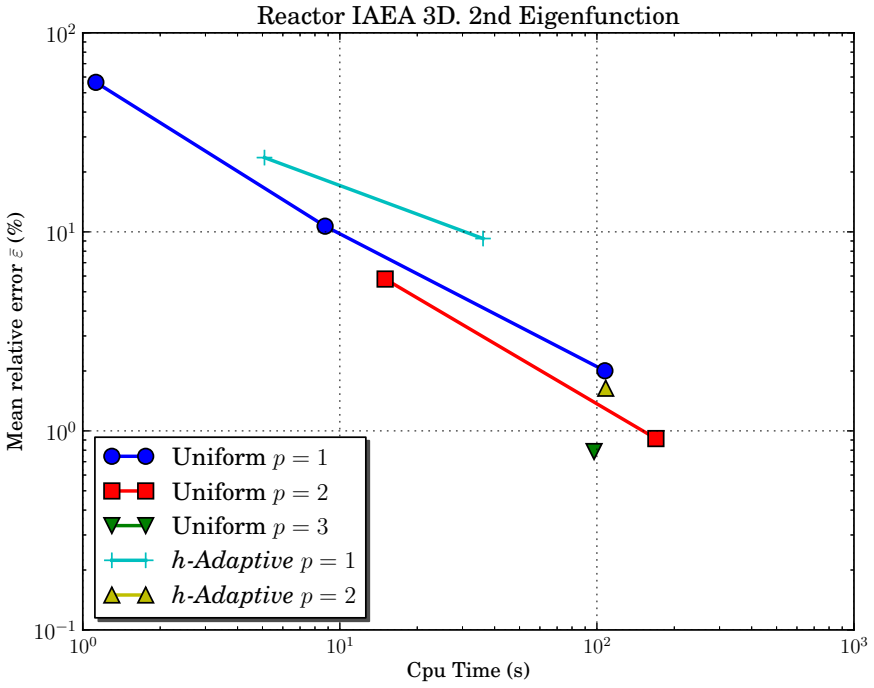


Figure 16: Mean error against CPU time for the 2nd mode of the IAEA 3D Reactor.

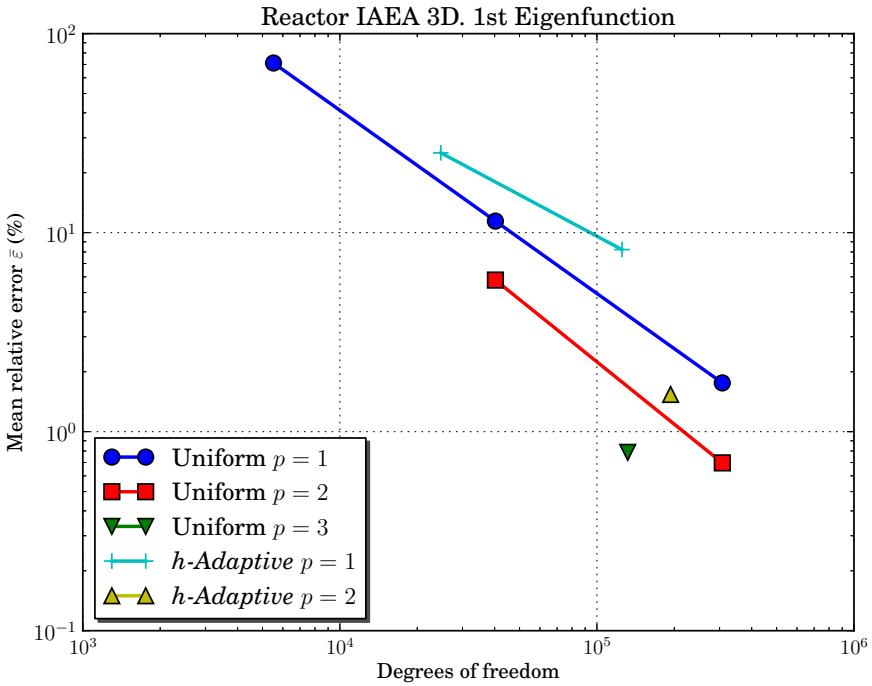


Figure 17: Mean relative error against number of degrees of freedom for the 1st mode of the IAEA 3D Reactor.

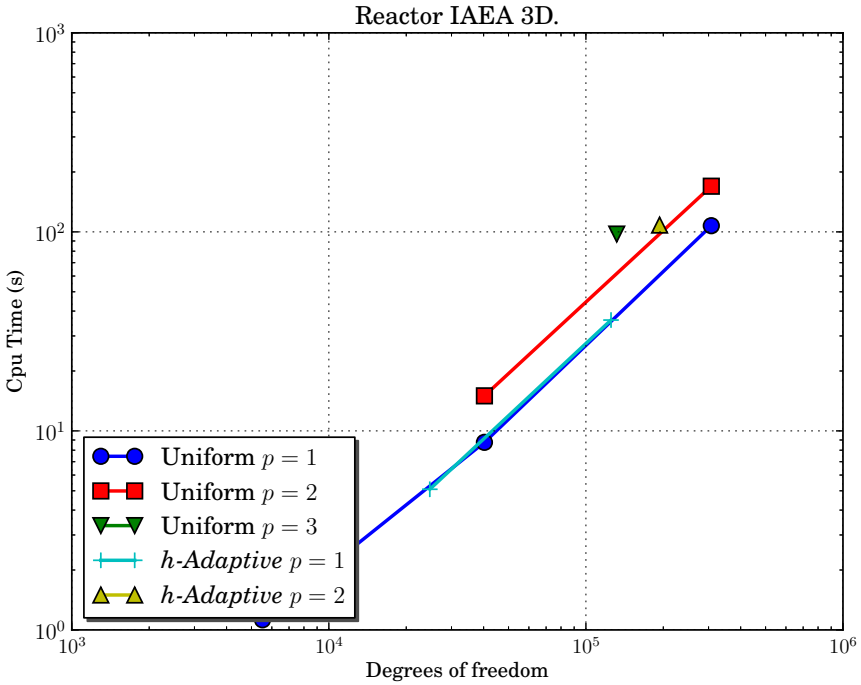


Figure 18: CPU time against degrees of freedom for the IAEA 3D Reactor.

4. Conclusions

In this contribution, we have presented an adaptive finite element algorithm for the Lambda modes problem. This method allows using high order finite elements with heterogeneous meshes. In this way, to increase the accuracy of the solution it is possible both to refine the spatial mesh and to increase the degree of the polynomials in the finite element method. To study the performance of the method to compute the dominant eigenvalues and their corresponding eigenvectors of a nuclear power reactor, different benchmark problems have been analysed, using different meshes and configurations of the computations. From all the analyses performed is concluded that the method converges if the mesh is refined or the degree of the polynomial expansions is increased, being the last strategy the most convenient one to obtain accurate results with a moderate computational cost.

Acknowledgements

This work has been partially supported by the Spanish Ministerio de Ciencia e Innovación under project ENE2011-22823, the Generalitat Valenciana under projects PROMETEO/2010/039 and ACOMP/2013/237, and the Universitat Politècnica de València under project UPPTE/2012/118.

Appendix A. Analytic solution of the homogeneous reactor

A 2D rectangular homogeneous reactor is considered. The Lambda modes problem for a bidimensional domain is defined as

$$\begin{aligned} -\vec{\nabla} D_1 \vec{\nabla} \phi_1(x,y) + (\Sigma_{a1} + \Sigma_{12})\,\phi_1(x,y) &= \frac{1}{\lambda}\left(\nu\Sigma_{f1}\phi_1(x,y) + \nu\Sigma_{f2}\phi_2(x,y)\right) , \\ -\Sigma_{12}\phi_1(x,y) - \vec{\nabla} D_2 \vec{\nabla} \phi_2(x,y) + \Sigma_{a2}\phi_2(x,y) &= 0 , \qquad (x,y) \in [0,L_1]\times[0,L_2] , \end{aligned} \tag{A.1}$$

with the homogeneous boundary conditions

$$\phi_g(0,y) = \phi_g(L_1,y) = 0 , \qquad \phi_g(x,0) = \phi_g(x,L_2) = 0 , \qquad g = 1,2 . \tag{A.2}$$

Using the variables separation method,

$$\phi_g(x,y) = X_g(x)Y_g(y), \tag{A.3}$$

X_g and Y_g are solutions of

$$\frac{\mathrm{d}^2 X_g}{\mathrm{d}x^2}(x) = \mu_x X_g(x), \qquad \frac{\mathrm{d}^2 Y_g}{\mathrm{d}y^2}(y) = \mu_y Y_g(y), \tag{A.4}$$

satisfying,

$$X_g(0) = X_g(L_1) = Y_g(0) = Y_g(L_2) = 0 . \tag{A.5}$$

Thus, these functions have the general form,

$$\begin{aligned} X_g &= A_{g,x}\cos(\mu_x x) + B_{g,x}\sin(\mu_x x) , \\ Y_g &= A_{g,y}\cos(\mu_y y) + B_{g,y}\sin(\mu_y y) . \end{aligned} \tag{A.6}$$

Using the boundary conditions (A.2)

$$X(x) = B_{g,x}\sin\left(\frac{n\pi}{L_1}\right) , \qquad \mu_x = \frac{n\pi}{L_1} , \tag{A.7}$$

$$Y(y) = B_{g,y}\sin\left(\frac{m\pi}{L_2}\right) , \qquad \mu_x = \frac{m\pi}{L_2} , \tag{A.8}$$

and

$$\mu^2 = \mu_x^2 + \mu_y^2 , \tag{A.9}$$

with $n, m \in \mathbb{N}$. Different values of n, m correspond to the different eigenvalues and eigenfunctions of the reactor. The equations are joined as equation (A.3),

$$\phi_g(x, y) = k_g \sin(\mu_x) \sin(\mu_y) \,. \tag{A.10}$$

The Equation (A.1) implies

$$\phi_1(x, y) = \frac{D_2\mu^2 + \Sigma_{a2}}{\Sigma_{12}} \phi_2(x, y) \,. \tag{A.11}$$

Solving for the eigenvalue from the Equation (A.1), it is obtained

$$\lambda = \frac{\nu\Sigma_{f1}\left(D_2\mu^2 + \Sigma_{a2}\right) + \nu\Sigma_{f2}\Sigma_{12}}{\left(D_2\mu^2 + \Sigma_{a2}\right)\left(\Sigma_{a1} + \Sigma_{12} + D_1\mu^2\right)} \tag{A.12}$$

with the eigenfunctions

$$\phi_1(x, y) = k\left(\frac{D_2\mu^2 + \Sigma_{a2}}{\Sigma_{12}}\right) \sin(\mu_x x) \sin(\mu_y y) \,, \tag{A.13}$$

$$\phi_2(x, y) = k \sin(\mu_x x) \sin(\mu_y y) \,. \tag{A.14}$$

As the fluxes are defined up to a multiplicative constant k, these should be normalized with the criterion exposed in Equation (21),

$$1 = \frac{1}{V_t} \int_V \left(\Sigma_{f1}|\phi_1| + \nu\Sigma_{f2}|\phi_2|\right) \mathrm{d}V = \frac{1}{L_1 L_2} \left(\nu\Sigma_{f1} \frac{D_2\mu^2 + \Sigma_{a2}}{\Sigma_{12}} + \nu\Sigma_{f2}\right) \int_0^{L_1} \mathrm{d}x \int_0^{L_2} \mathrm{d}y \, |\phi_2| \,. \tag{A.15}$$

Hence, the normalized magnitudes obtained are

$$\phi_1(x, y) = \left(\frac{D_2\mu^2 + \nu\Sigma_{a2}}{\nu\Sigma_{f1} D_2\mu^2 + \nu\Sigma_{f1}\Sigma_{a2} + \nu\Sigma_{f2}\Sigma_{12}}\right)\left(\frac{\pi^2}{4}\right) \sin\left(\frac{n\pi}{L_1}x\right) \sin\left(\frac{m\pi}{L_2}y\right) \,, \tag{A.16}$$

$$\phi_2(x, y) = \left(\frac{\Sigma_{12}}{\nu\Sigma_{f1} D_2\mu^2 + \nu\Sigma_{f1}\Sigma_{a2} + \nu\Sigma_{f2}\Sigma_{12}}\right)\left(\frac{\pi^2}{4}\right) \sin\left(\frac{n\pi}{L_1}x\right) \sin\left(\frac{m\pi}{L_2}y\right) \,, \tag{A.17}$$

$$P = \left(\nu\Sigma_{f1}\phi_1 + \nu\Sigma_{f2}\phi_2\right) = \frac{\pi^2}{4} \sin\left(\frac{n\pi}{L_1}x\right) \sin\left(\frac{m\pi}{L_2}y\right) \,. \tag{A.18}$$

This proves that the normalized neutron distribution in a homogeneous reactor does not depend on the nuclear properties of the material.

References

American Nuclear Society, Jun 1977. Argonne code center: benchmark problem book, anl-7416(suppl.2). Tech. rep.

Baker, C., Buchan, A., Pain, C., Tollit, B., Goffin, M., Merton, S., Warner, P., 2013. Goal based mesh adaptivity for fixed source radiation transport calculations. Annals of Nuclear Energy 55 (0), 169 – 183.

Bangerth, W., Hartmann, R., Kanschat, G., 2007. deal.II – a general purpose object oriented finite element library. ACM Trans. Math. Softw. 33 (4), 24/1–24/27.

Bangerth, W., Kayser-Herold, O., 2009. Data structures and requirements for hp finite element software. ACM Trans. Math. Softw. 36 (1), 4/1–4/31.

Cuthill, E., McKee, J., 1969. Reducing the bandwidth of sparse symmetric matrices. In: Proceedings of the 1969 24th national conference. ACM '69. ACM, New York, NY, USA, pp. 157–172.

De S. R. Gago, J. P., Kelly, D. W., Zienkiewicz, O. C., Babuska, I., 1983. A posteriori error analysis and adaptive processes in the finite element method: Part II—adaptive mesh refinement. International Journal for Numerical Methods in Engineering 19 (11), 1621–1656.

Ginestar, D., Miró, R., Verdú, G., Barrachina, T., 2011. Modal processing of the local power range monitors signals in BWR NPP. Annals of Nuclear Energy 38 (11), 2441 – 2455.

González-Pintor, S., Ginestar, D., Verdú, G., 2009. High order finite element method for the Lambda modes problem on hexagonal geometry. Annals of Nuclear Energy 36 (9), 1450 – 1462.

Hébert, A., 1985. Application of the Hermite method for finite element reactor caluclations. Nuclear Science and Engineering 91 (1), 34–58.

Hébert, A., 1987. Development of the nodal collocation method for solving the neutron diffusion equation. Annals of Nuclear Energy 14 (10), 527 – 541.

Hébert, A., 2008. A Raviart-Thomas-Schneider solution of the diffusion equation in hexagonal geometry. Annals of Nuclear Energy 35 (3), 363 – 376.

Henry, A., 1975. Nuclear-reactor analysis. MIT Press, Cambridge, MA, USA.

Hernandez, V., Roman, J. E., Vidal, V., 2005. SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems. ACM Trans. Math. Software 31 (3), 351–362.

March-Leuba, J., Rey, J., 1993. Coupled thermohydraulic-neutronic instabilities in boiling water nuclear reactors: a review of the state of the art. Nuclear Engineering and Design 145 (1-2), 97–111.

Miró, R., Ginestar, D., Verdú, G., Hennig, D., 2002. A nodal modal method for the neutron diffusion equation. application to BWR instabilities analysis. Annals of Nuclear Energy 29 (10), 1171 – 1194.

Morgan, R. B., Jan. 2002. GMRES with deflated restarting. SIAM J. Sci. Comput. 24 (1), 20–37.

Müller, E., Weiss, Z., 1991. Benchmarking with the multigroup diffusion high-order response matrix method. Annals of Nuclear Energy 18 (9), 535 – 544.

Singh, T., Mazumdar, T., Pandey, P., 2014. Nemsqr: A 3-d multi group diffusion theory code based on nodal expansion method for square geometry. Annals of Nuclear Energy 64 (0), 230 – 243.

Stewart, G., 2002. A Krylov-Schur algorithm for large eigenproblems. SIAM Journal on Matrix Analysis and Applications 23 (3), 601–614.

Theler, G., 2013. A consistent multidimensional nodal method for transient calculations. Science and Technology of Nuclear Installations 2013 (641863).

Verdú, G., Ginestar, D., Vidal, V., Muñoz-Cobo, J., 1994. 3D Lambda-modes of the neutron-diffusion equation. Annals of Nuclear Energy 21 (7), 405 – 421.

Wang, Y. Bangerth, W., Ragusa, J., 2009. Three-dimensional h-adaptivity for the multigroup neutron diffusion equations. Progr. Nucl. Energy.

Zienkiewicz, O. C., Taylor, R. L., Zhu, J. Z., 2005. The finite element method: its basis and fundamentals. Butterworth-Heinemann.