The solution of the lambda modes problem using block iterative eigensolvers

A. Carreño[1], A. Vidal-Ferràndiz[1], D. Ginestar[2], and G. Verdú[1]

[1] Instituto Universitario de Seguridad Industrial, Radiofísica y Medioambiental
Universitat Politècnica de València, València, Spain,
amcarsan@iqn.upv.es,anvifer2@upv.es , gverdu@iqn.upv.es
[2] Instituto Universitario de Matemática Multidisciplinar,
Universitat Politècnica de València, València, Spain,
dginesta@mat.upv.es

Abstract. High efficient methods are required for the computation of several lambda modes associated with the neutron diffusion equation. Multiple iterative eigenvalue solvers have been used to solve this problem. In this work, three different block methods are studied to solve this problem. The first method is a procedure based on the modified block Newton method. The second one is a procedure based on subspace iteration and accelerated with Chebyshev polynomials. Finally, a block inverse-free Krylov subspace method is analyzed with different preconditioners. Two benchmark problems are studied illustrating the convergence properties and the effectiveness of the methods proposed.

Keywords: Neutron diffusion equation, Eigenvalue problem, Lambda modes, Block method.

1 Introduction

The neutron transport equation models the behaviour of a nuclear reactor over the reactor domain [14]. However, due to the complexity of this equation, the energy of the neutrons is discretized into two energy groups and the flux is assumed to be isotropic leading to an approximation of the neutron transport equation known as, the two energy groups neutron diffusion equation [14].

The reactor criticality can be forced by dividing the neutron production rate in the neutron diffusion equation by λ obtaining a steady state equation expressed as a generalized eigenvalue problem, known as the λ-modes problem,

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

where

$$\mathcal{L} = \begin{pmatrix} -\boldsymbol{\nabla}(D_1\boldsymbol{\nabla}) + \Sigma_{a_1} + \Sigma_{12} & 0 \\ -\Sigma_{12} & -\boldsymbol{\nabla}(D_2\boldsymbol{\nabla}) + \Sigma_{a_2} \end{pmatrix},$$

is the neutron loss operator and

$$\mathcal{M} = \begin{pmatrix} \nu\Sigma_{f1} & \nu\Sigma_{f2} \\ 0 & 0 \end{pmatrix}, \qquad \phi = \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix}$$

are the neutron production operator and the neutron flux. The rest of coefficient, called macroscopic cross sections, are dependent on the spatial coordinate. The diffusion cross sections are D_1 (for the first energy group) and D_2 (for the second one); Σ_{a1} and Σ_{a2} denote the absorption cross sections; Σ_{12}, the scattering coefficient from group 1 to group 2. The fission cross sections are Σ_{f1} and Σ_{f2}, for the first and second group, respectively. And ν is the average number of neutron produced per fission.

The eigenvalue (mode) with the largest magnitude shows the criticality of the reactor and its corresponding eigenvector describes the steady state neutron distribution in the core. The next sub-critical modes and their associated eigen-functions are useful to develop modal methods to integrate the transient neutron diffusion equation.

For the spatial discretization of the λ-modes problem, a high order continuous Galerkin Finite Element Method (FEM) is used, transforming the problem (1) into an algebraic generalized eigenvalue problem

$$Mx = \lambda Lx, \tag{2}$$

where these matrices are not necessarily symmetric (see more details in [17]). However, with several general conditions, it has been proved, that the dominant eigenvalues of this equation are real positive numbers [8].

Different methods have been successfully used to solve this algebraic gener-alized eigenvalue problem such as the Krylov-Schur method, the classical Arnoldi method, the Implicit Restarted Arnoldi method and the Jacobi-Davidson method [15,16,17]. However, if we want to compute several eigenvalues and they are very clustered, these methods might have problems to find all the eigenvalues. In practical situations of reactor analysis, the dominance ratios corresponding to the dominant eigenvalues are often near unity. By this reason, block methods, which approximate a set of eigenvalues simultaneously are an alternative since their rate of convergence depends only on the spacing of the group of desired eigenvalues from the rest of the spectrum. In this work, three different block methods are studied and compared with the Krylov-Schur method.

The rest of the paper has been structured in the following way. In Section 2, the block iterative methods are presented. In Section 3, numerical results to study the performance of the method for two three dimensional benchmark problems are presented. In the last Section, the main conclusions of the paper are collected.

2 Block iterative methods

This section describes the block methods to obtain the dominant eigenvalues and their associated eigenvectors of a generalized eigenvalue problem of the form

$$MX = LX\Lambda, \tag{3}$$

where $X \in \mathbb{R}^{n \times q}$ has the eigenvectors in their columns and $\Lambda \in \mathbb{R}^{q \times q}$ has the dominant eigenvalues in its diagonal, n denotes the degrees of freedom in the

spatial discretization with the finite element method for the equation (1) and q is the number of desired eigenvalues.

2.1 Modified block Newton method

The original modified block Newton method was proposed by Lösche in [10] for ordinary eigenproblems. This section briefly reviews an extension of this method given by the authors in [4] for generalized eigenvalue problems.

To apply this method to the problem (3), we assume that the eigenvectors can be expressed as

$$X = ZS, \tag{4}$$

where $Z^T Z = I_q$. Then, problem (3) can be rewritten as

$$MX = LX\Lambda \Rightarrow MZS = LZS\Lambda \Rightarrow MZ = LZS\Lambda S^{-1} \Rightarrow MZ = LZK. \tag{5}$$

If we add the biorthogonality condition $W^T Z = I_q$ in order to determine the problem, with W is a matrix of rank q, it is obtained the following system

$$F(Z, \Lambda) := \begin{pmatrix} MZ - LZK \\ W^T Z - I_q \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{6}$$

Applying a Newton's iteration to the problem (6), a new approximation arises from the previous iteration as,

$$Z^{(k+1)} = Z^{(k)} - \Delta Z^{(k)}, \qquad K^{(k+1)} = K^{(k)} - \Delta K^{(k)}, \tag{7}$$

where $\Delta Z^{(k)}$ and $\Delta K^{(k)}$ are solutions of the system that is obtained when the equation (7) is substituted into (6) and it is truncated at the first terms.

The matrix $K^{(k)}$ is not necessarily a diagonal matrix, as a consequence the system is coupled. To avoid this problem, the modified generalized block Newton method (MGBNM) applies previously two steps. The initial step is to apply the modified Gram-Schmidt process to orthogonalize the matrix $Z^{(k)}$. The second step consist on use the Rayleigh-Ritz projection method for the generalized eigenvalue problem [12]. More details of the method can be found in [4].

2.2 Block inverse-free block preconditioned Krylov subspace method

The block inverse-free preconditioned Arnoldi method (BIFPAM) was originally presented and analyzed for L and M symmetric matrices and $L > 0$ (see [7,11]). Nevertheless, this methodology works efficiently to compute the λ-modes.

We start with the problem for one eigenvalue

$$Mx = \lambda Lx, \tag{8}$$

and an initial approximation (λ_0, x_0). We aim at improving this approximation through the Rayleigh-Ritz orthogonal projecting on the m-order Krylov subspace

$$K_m(M - \lambda_0 L, x_0) := \text{span}\{x_0, (M - \lambda_0 L)x_0, (M - \lambda_0 L)^2 x_0, \dots, (M - \lambda_k L)^m x_0\}.$$

Arnoldi method is used to construct the basis K_m. The projection can be carried out as

$$Z^T M Z U = Z^T L Z U \Lambda, \tag{9}$$

where Z is a basis of $K_m(M - \lambda_0 L, x_0)$ and then computing the dominant eigenvalue $\Lambda_{1,1}$ and its eigenvector u_1 to obtain the value of $\lambda_1 = \Lambda_{1,1}$ and its eigenvector $x_1 = Z u_1$. In the same way, we compute the eigenvalues and eigenvectors in the following iterations.

If we are interested on computing q eigenvalues of problem (2), we can accelerate the convergence by using the subspace $\mathcal{K}_m$ with

$$\mathcal{K}_m := \bigcup_{i=1}^{q} K_m^i (M - \lambda_{k,i} L, x_{k,i}),$$

where $\lambda_{k,i}$ denotes the i-th eigenvalue computed in the k-th iteration and $x_{k,i}$ its associated eigenvector. Thus, this method can be dealt with through an iteration with a block of vectors that allows computing several eigenvalues simultaneously.

Furthermore, the BIFAM will be accelerated with an equivalent transformation of the original problem by means of a preconditioner. With an approximate eigenpair $(\lambda_{i,k}, x_{i,k})$, we consider for some matrices $P_{i,k}$, $Q_{i,k}$ the transformed eigenvalue problem

$$(P_{i,k}^{-1} M Q_{i,k}^{-1})x = \lambda (P_{i,k}^{-1} L Q_{i,k}^{-1})x \;\Leftrightarrow\; \hat{M}_{i,k} x = \lambda \hat{L}_{i,k} x, \tag{10}$$

which has the same eigenvalues as the original problem. Applying one step of the block inverse-free Krylov method to the problem (10), the convergence behaviour will be determined by the spectrum of

$$\hat{C}_{i,k} := \hat{M}_{i,k} - \lambda_{i,k} \hat{L}_{i,k} = P_{i,k}^{-1}(M - \lambda_{i,k} L)Q_{i,k}^{-1}. \tag{11}$$

Different preconditioning transformations can be constructed using different factorizations of the matrix $M - \lambda_{i,k} L$. The main goal must be to choose suitably $P_{i,k}$ and $Q_{i,k}$ to obtain a favorable distribution of the eigenvalues of matrix $\hat{C}_{i,k}$.

In this paper, we have considered the classical incomplete LU factorization with level 0 of fill (ILU(0)). We also use constants $P_{i,k} = P_{1,1}$ and $Q_{i,k} = Q_{1,1}$ obtained from a preconditioner for $M - \lambda_{1,1} L$, where $\lambda_{1,1}$ is a first approximation of the first eigenvalue.

2.3 Chebyshev filtered subspace iteration method

Subspace iteration with a Chebyshev polynomial filter (CHEFSI) is a well known algorithm in the literature [12,18]. In this paper, we have studied a version proposed by Berjafa et al. in [5] that iterates over the polynomial filter and the Rayleigh quotient with block structure. This algorithm is implemented for ordinary eigenvalue problems, so the original problem (3) is reformulated as

$$AX = X\Lambda \;\; \text{with} \;\; A = L^{-1}M. \tag{12}$$

The goal of this method is to build an invariant subspace for several eigenvectors using multiplication in block. This subspace is diagonalized using previously a polynomial filter in these vectors to improve the competitiveness of the method.

The basic idea for computing the first dominant eigenvalue is the following: Using the notation introduced in Section 2, it is known that any vector z can be expanded in the eigenbasis as

$$z = \sum_{i=1}^{n} \gamma_i x_i.$$

Applying a polynomial filter $p(x)$ of degree m to A through a matrix-vector product leads to

$$p_m(A)z = p_m(A) \sum_{i=1}^{n} \gamma_i x_i = \sum_{i=1}^{n} p_m(\lambda_i)\gamma_i x_i,$$

where it is assumed that $\gamma_1 \neq 0$, which is almost always true in practice if z is a random vector.

If we want to compute x_1 as fast as possible, then a suitable polynomial would be a $p(x)$ such that $p(\lambda_1)$ dominates $p(\lambda_j)$, when $j \neq 1$. That it means, the filter must separate the desired eigenvalue from the unwanted ones, so that after normalization $p(A)z$ will be mostly parallel to x_1. This leads us to seek a polynomial which takes small values on the discrete set $R = \{\lambda_2, \ldots, \lambda_n\}$, such that $p_m(\lambda_1) = 1$. However, it is not possible to compute this polynomial with the unacknowledged of all eigenvalues of A. The alternative is use a continuous domain in the complex plane containing R but excluding λ_1 instead of the discrete min-max polynomial. In practice, the continuous domain is restricted to an ellipse E containing the unwanted eigenvalues and then theoretically it can be shown that the best min-max polynomial is the polynomial

$$p_m(\lambda) = \frac{C_m((\lambda - c))/e}{C_m((\lambda_1 - c))/e},$$

where C_m is the Chebyshev polynomial of degree m, c is the center of the ellipse E and e is the distance between the center and the focus of E (see more details in [12]).

In our case, where the eigenvalues are positive real numbers, the ellipse E is restricted to an interval $[\alpha, \beta]$, where $\alpha, \beta > 0$. These values are computed following the algorithms proposed in [18].

3 Numerical Results

The competitiveness of the block methods has been tested on two three dimensional problems: the 3D IAEA reactor [13] and the 3D NEACRP reactor [6]. For the spatial discretization of the λ-modes problem, we have used Lagrange polynomials of degree 3 in the finite element method.

In the numerical results, the global residual error has been used, defined as

$$\mathrm{res} = \max_{i=1,\dots,q} \|Lx_i - \lambda_i M x_i\|^2 \ ,$$

where λ_i is the i-th eigenvalue and x_i its associated unitary eigenvector.

As the block methods need an initial approximation of a set of eigenvectors, a multilevel initialization proposed in [3] with two meshes is used to obtain this approximation.

The solutions of linear systems needed to apply the MGBN method and the CHEFSI method have been computed with the GMRES method preconditioned with ILU and a reordering using the Cuthill-McKee method. The dimension of the Krylov subspace for the BIFPAM has been set equal to 8. The degree of the Chebyshev polynomial has been 10.

The methods have been implemented in C++ based on data structures provided by the library Deal.ii [2], PETSc [1] using the definition of the cited papers. For make the computations, we have used a computer that has been an Intel® Core™ i7-4790 @3.60GHz×8 processor with 32Gb of RAM running on Ubuntu 16.04 LTS.

3.1   3D IAEA reactor

The 3D IAEA benchmark reactor is a classical two-group neutron diffusion problem [13]. It has 4579 different assemblies and the coarse mesh used to obtain the initial guess has 1040 cells. The algebraical eigenvalue problems have 263552 and 62558 degrees of freedom, for the fine and the coarse mesh, respectively.

To compare the block methods, the number of iterations for the BIFPAM, the MGBNM and the CHEFSI method and the residual errors are represented in Figure 1(a) in the computation of four eigenvalues. These eigenvalues are 1.02914, 1.01739, 1.01739 and 1.01526. In this Figure, we observe similar slopes in the convergence histories for the BIFPAM and the CHEFSI method and moreover, they are smaller than the convergence history for the MGBNM since this is a second-order method. The computational times (CPU time) and the residual errors (res) obtained for each method are shown in Figure 1(b). In this Figure, in contrast to the previous one, it is observed that the most efficient method in time is the BIFPAM although its CPU times are similar to the CPU times obtained for the MGBNM. This means that in spite of the number of iterations needed to converge the BIFPAM is larger than the MGBNM, the CPU time in each iteration is much smaller than the needed to compute one iteration of the MGBNM. It is due to the BIFPAM does not need to solve linear systems.

3.2   3D NEACRP reactor

The NEACRP benchmark [6] is also chosen to compare the block methodology proposed. The reactor core has a radial dimension of 21.606 cm × 21.606 cm per cell. Axially the reactor is divided into 18 layers with height (from bottom
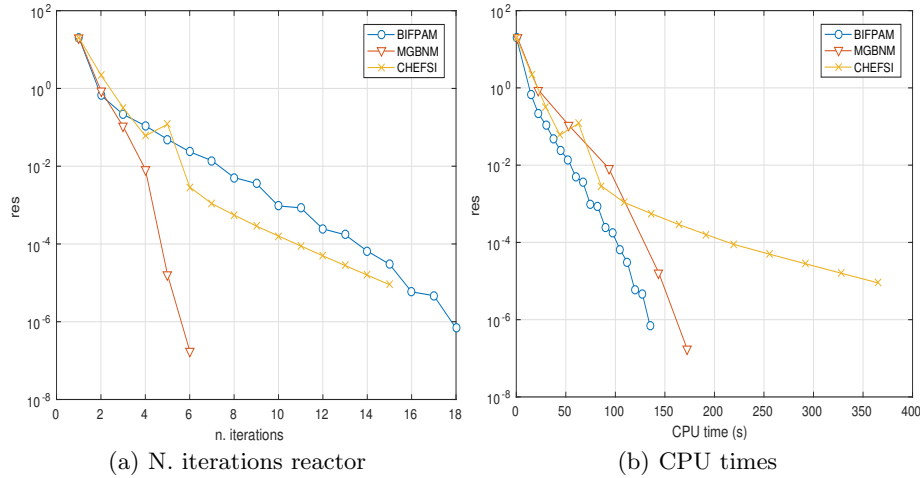
(a) N. iterations reactor   (b) CPU times

Fig. 1. Residual error (res) for the computation of 4 eigenvalues in the IAEA reactor.

to top): 30.0 cm, 7.7 cm, 11.0 cm, 15.0 cm, 30.0 cm (10 layers), 12.8 cm (2 layers), 8.0 cm and 30.0 cm. The boundary condition is zero flux in the outer reflector surface. The fine mesh and the coarse mesh considered have 3978 and 1308 cells, respectively. Using polynomials of degree three the fine mesh has 230120 degrees of freedom. The coarse mesh used to initialize the block methods has 7844 degrees of freedom.

Figure 2(a) shows the convergence histories of the BIFPAM, the MGBNM and the CHEFSI method in terms of the number of iterations in the computation of four eigenvalues. The eigenvalues obtained have been 1.00200, 0.988620, 0.985406 and 0.985406. That it means the spectrum for this problem is very clustered. In this Figure, we observe the similar behaviour between the BIF-PAM and the CHEFSI method being these two methods slower in convergence than the MBNM. Figure 2(b) displays the CPU time and the residual errors obtained for each method. In this Figure, we observe that the quickest method is the BIFPAM by the same reason given in the previous. So, the most efficient block method studied is the BIFPAM.

Finally, these block methods are compared with the Krylov-Schur method implemented in the library SLEPc [9] for the NEACRP reactor. This method is a non-block method, but it is a very competitive method to solve eigenvalue problems. The dimension of the Krylov subspace used in the Krylov-Schur method has been $15 + q$ that is the default value of the library. This method is implemented in the library using a locking strategy, so the history block convergence cannot be displayed and compared with the block method presented in this work. The total computational times obtained for a different number of eigenvalues are displayed in Table 1 to compare the block methods with the Krylov-Schur method. The total CPU time of the block methods includes the time needed to compute
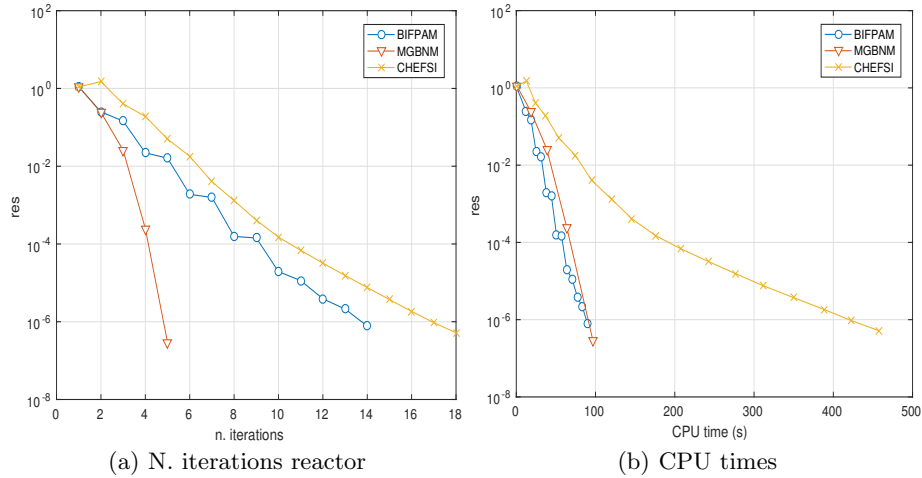
(a) N. iterations reactor

(b) CPU times

Fig. 2. Residual error (res) for the computation of 4 eigenvalues in the NEACRP reactor.

the initial guess. The tolerance set for all methods has been res= 10^{-6}. In this Table, we observe that the BIFPAM and MGBNM methods compute the eigenvalues faster than the Krylov-Schur method from a number of eigenvalues equal to 4, being the fastest the MGBNM. This is also observed when we compute one eigenvalue. For 2 and 3 eigenvalues the CPU times obtained with the Krylov-Schur method are smaller than the CHEFSI method and the BIFPAM, while these values are larger than for the MGBNM. In these cases, it is necessary to use higher subspace dimension than 8 for the BIFPAM to obtain better results. For all cases, it is observed that the CHEFSI method does not improve the times obtained with the other block methods and the Krylov-Schur method.

Table 1. Computational times (s) obtained for the NEACRP reactor using the Krylov-Schur method, the BIFPAM, the MGBNM and the CHEFSI method for different number of eigenvalues

n. eigs (q)	Krylov-Schur	BIFPAM	MGBNM	CHEFSI
1	98	65	76	249
2	134	174	108	390
3	135	207	132	390
4	214	153	149	510
5	237	213	185	630

4   Conclusions

The computation of the λ-modes associated with the neutron diffusion equation is interesting for several applications such as the study of the reactor criticality and the development of modal methods. A high order finite element method is used to discretize the λ-modes problem. Different block methods have been studied and compared to solve the algebraical problem obtained from the discretization. These methods have been tested using two 3D benchmark reactors: the IAEA reactor and the NEACRP reactor.

The main conclusion of this work is that the use of block methods is a good strategy alternative to Krylov methods when we are interested in computing a set of dominant eigenvalues. However, the efficiency depends on the type of method. For generalized eigenvalues problems, the BIFPAM, that does not need to solve linear systems, or the MGBNM, that converges with a short number of iterations, are good choices that improve the computational times obtained with the competitive Krylov-Schur method. With respect to the CHEFSI method, due to their implementation for ordinary eigenvalue problems, it needs to solve many linear systems that makes the method inefficient. In future works, a generalization of this method for generalized eigenvalue problems will be studied.

Acknowledgements