Journal of Computer Science & Computational Mathematics, *Volume 12, Issue 4, December 2022*
DOI: 10.20967/jcscm.2022.04.001

Regularized Block Preconditioner for Solving Algebraic Linear Systems Associated With Cluster Dynamic Rate Equations

Achraf Badahmane

Université Paris-Saclay, 91191, Gif-sur-Yvette, France
Corresponding author email: badahmane.achraf@gmail.com

***Abstract*:** Many dynamical systems described by partial differential equations, master equations and chem- ical rate equations are directly simulated using numerical time-stepping methods. These methods require solving algebraic linear systems at each time step. These linear systems are large scale and ill-conditioned, thus they require the development of effective solvers. The aim of this work is the construction and numerical validation of block preconditioners for a set of algebraic linear systems associated with cluster dynamic rate and master equations. We provide a new regularized block preconditioner based on the approximate Schur complement and on a regularization technique. The eigenvalues distributions of the preconditioned matrix are presented and simulation results are run on a computer architecture. We analyze the performance of the iterative solvers in terms of preconditioned GMRES, FGMRES and BICGSTAB iterations and computational time.

Keywords: Preconditioner, MUMPS, LAPACK.

1. Introduction

The goal is to take advantage of the particular structure of the Jacobian and Newton matrices. As aforementioned, the cluster defects, whose number N corresponds to the number of differential equations, can be classified into two subsets, depending on whether they are mobile or immobile. Mobile defects react with all the other defects, while immobile defects react only with the mobile defects. This physical property greatly impacts the form of the linear algebraic problem. Assuming that there are d types of defect clusters that are mobile and s immobile defect clusters $(n+m=N)$, the linear system to solve can be written as follows:

$$\mathcal{A}\tilde{x} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \underbrace{\begin{bmatrix} f \\ g \end{bmatrix}}_{b}, \tag{1}$$

the block-(1,1) of $\mathcal{A}$, denoted by $A \in \mathbb{R}^{n\times n}$ in general is a sparse and nonsymmetric matrix, block-(1,2) denoted by $B \in \mathbb{R}^{n\times m}$ which m«n, $C \in \mathbb{R}^{m\times n}$, $D \in \mathbb{R}^{m\times m}$ is a dense and nonsymmetric matrix, the given vectors $f \in \mathbb{R}^n$ and $g \in \mathbb{R}^m$ are the components of the right-hand side. Sequences of linear systems of the type Eq. (1) appear in rate equation; see Sect. 2. Since the matrices A and B in Eq. (1) are large, solution of Eq. (1) Krylov subspace projection methods are suited to solve linear systems in which the involved matrix is large and sparse. An important number of iterations is however necessary to obtain a reliable approximate of the solution when the involved linear systems being large scale and ill-conditioned, effective linear solvers must be implemented. For ill-conditioned matrices, it is more likely to use ad preconditioner technique incorporated with iterative solver such as GMRES. In practice, the preconditionning matrix $\mathcal{P}$ should be chosen close to the inverse of the Newton matrix, several preconditioners [1–3] have been proposed for (1.1). The rest of the paper is structured as follows. In Sec. 2.1, we formulate the evolution equations for the concentrations of clusters containing self interstitial atoms, vacancies and solute atoms (IVS model). We briefly describe in Sec. 2.2 the numerical time-stepping scheme that is used to integrate the formulated high-dimensional ordinary differential equation. We next introduce the direct approach (Sec. 2.3), iterative approach (Sec. 2.4) and Schur approach (Sec. 2.5) used for solving the linear systems arising within the modified or inexact Newton algorithms. We present simulation results for the Fe-He and Fe-Cu systems obtained using the various solvers in Sec. 3. We finally conclude in Sec. 4 and give some recommendation on which approach to implement depending on the nature of the problem.

2. Problem formulation

As discussed in [4], IVS model describe the evolution of clusters containing interstitial (I), vacancies (V) and solute (S), this so-called "IVS model", are described an ordinary differential equation (ODE).

The evolution equation for the concentration $C_{k,p}$ reads:

$$\begin{aligned}\frac{dC_{k,p}}{dt} = &\sum_{j=-m_v}^{mi}\sum_{q=0}^{ms}\{B_{k-j,p-q,j,q}C_{i,q}C_{k-j,p-q} - B_{k,p,j,q}C_{j,q}C_{k,p}\} \\ &+ \sum_{i=-m_v}^{mi}\sum_{q=0}^{ms}\{A_{k+j,p+q,j,q}C_{k+j,p+q} - A_{k,p,j,q}C_{k,p}\}.\end{aligned} \tag{2}$$

Each cluster is identified by the couple (k,p), where:

- where $|k|$ is the number of interstitials (if $k>0$) or the number of vacancies (if $k<0$), and p ($p \leq 0$) is the number of solutes in this cluster.
- m_i: denote the maximum number of interstitials in mobile species.
- m_v: denote the maximum number of vacancies in mobile species.
- m_s: denote the maximum number of solutes in mobile species.
- $B_{k,p,j,q}$: is the absorption rate of a cluster.

- $A_{k,p,j,q}$: is the emission rate of a cluster.

By using a backward differentiation formula (BDF) [5], to integrate Eq. (2), the linear system with unknown $X = C^{(i+1)}$ to be solved reads

$$C^{(i+1)} - h\gamma F\left(C^{(i+1)}\right) = F\left(C^{(i)}\right). \tag{3}$$

Where:

- i: is the timestep.
- $C^{(i)}$ and $C^{(i+1)}$ represent the vector of cluster concentrations at discretization time t_i and t_{i+1} respectively.
- h: denote the current time step, $h = t_{i+1} - t_i$.
- γ: denote a coefficient depending on the discretization method.
- $F(C)$: denote the functional of jacobian of Eq. (3).

Finding the root of F in Eq. Eq. (3) by means of an exact Newton method requires repeatedly solving linear systems where $\mathcal{A}$ is defined as follow:

$$\mathcal{A} = I - \gamma h \mathcal{J}, \tag{4}$$

The Jacobian matrix $\mathcal{J}$ must be assembled at every iteration.

3. Description and analysis of the preconditioner

Proposition 1: Assume that the block-(1,1) of $\mathcal{A}$ is nonsingular. Then the saddle point matrix $\mathcal{A}$ is nonsingular if and only if $S = -(D + CA^{-1}B)$ is nonsingular. Proof see [6, Proposition 2.1].
We consider the following block structure of the preconditioner:

$$\mathcal{P}_{\alpha,\hat{S}} = \begin{bmatrix} A & B \\ C & \alpha\hat{S} \end{bmatrix}, \tag{5}$$

where α is a given real nonzero parameter and $\hat{S}$ is an approximate Schur complement matrix of $\mathcal{A}$. For the computation of $\hat{S}$ we use Lower-upper approximation based on multifrontal approach [7, 8].

3.1 Implementation of Schur approach

The successive steps for computing the Schur complement $\hat{S}$ are listed below:

1. The block decomposition of $\mathcal{M}$.
2. The lower-diagonal-upper decomposition (LDU) of $\mathcal{M}$.
3. The definition of Schur complement.
4. $A^{-1}B \Leftrightarrow A\mathcal{X} = B$, where $\mathcal{X} \in \mathbb{R}^{n\times m}$, we solve this system with multiple right-hand sides by using Multifrontal approach, for more details, we refer reader to [7, 8].
5. Compute the sum $D - C\mathcal{X}$ by using (LAPACK library) [9].

Note that $\mathcal{P}_{\alpha,\hat{S}}$ is nonsingular under the assumptions of Proposition 1.

In order to apply block preconditioner of the form Eq. (5) within a Krylov subspace method, it is necessary to solve (exactly or inexactly, see below) the following linear system at each step:

$$\begin{bmatrix} A & B \\ C & \alpha\hat{S} \end{bmatrix} \begin{bmatrix} z_k^{(1)} \\ z_k^{(2)} \end{bmatrix} = \begin{bmatrix} r_k^{(1)} \\ r_k^{(2)} \end{bmatrix}. \tag{6}$$

The steps for solving the preconditioner algebraic linear system Eq. (6) using the Schur complement and involve the following additional steps in Fig. 2:

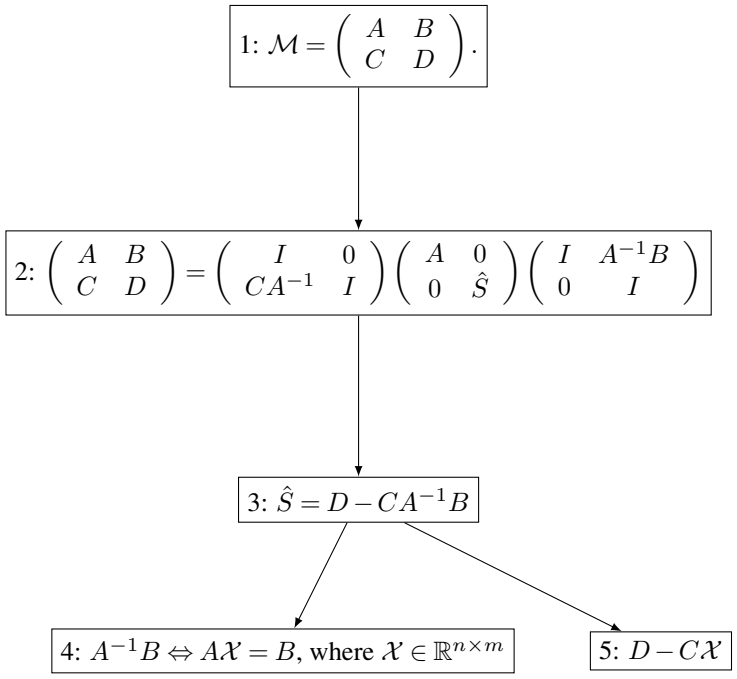

Figure 1. Schematic diagram of Schur complement computation

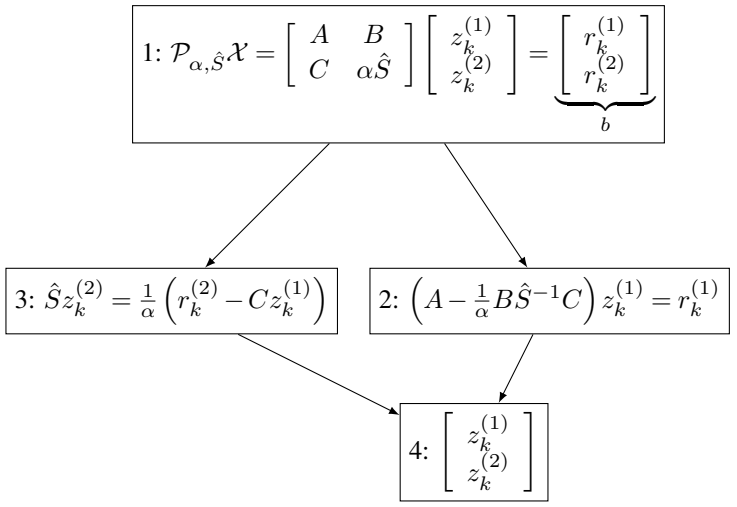

Figure 2. Schematic diagram of Schur approach

1. We consider the solution of linear equation Eq. (6) involving a sparse coefficient matrix having a triangular factorization, thus the main costs at each iteration for computing Eq. (6) are solving two sub-linear systems with coefficient matrices $\hat{S}$ and A respectively.
2. The solution part $z_k^{(2)}$ is computed through backward substitution and the sublinear system involving $\hat{S}$ by a dense solver from (LAPACK library), because $\hat{S}$ is an invertible and dense matrix.
3. The solution part $z_k^{(1)}$ is computed through backward substitution and the sublinear system involving A is solved directly using LU GMRES ($\mathcal{P}$GMRES) inexactly or MUMPS exactly.
4. The last step represent the approximate solution of Eq. (6) at each step in Newton's iteration.

4. Spectral analysis of the preconditioned matrix

The drawback of iterative methods is that they are inexact that's why demand more iterations steps so that the iterate satisfies the tolerance condition. The treatment to this technical disadvantage is called preconditioning. Here, the preconditioner $\mathcal{P}_{\alpha,\hat{S}}$ applies a linear transformation to system Eq. (1) so as to reduce the condition number of the transformed matrix $\mathcal{P}^{-1}_{\alpha,\hat{S}}\mathcal{M}$. The preconditioned linear system to solve writes:

$$\mathcal{P}^{-1}_{\alpha,\hat{S}}X = \mathcal{P}^{-1}_{\alpha,\hat{S}}b. \tag{7}$$

Proposition 2:

$$\begin{aligned}\mathcal{P}_{\alpha,\hat{S}} &= \begin{bmatrix} A & B \\ C & \alpha\hat{S} \end{bmatrix} \\ &= \begin{bmatrix} I & O \\ CA^{-1} & I \end{bmatrix}\begin{bmatrix} A & O \\ O & S_s \end{bmatrix}\begin{bmatrix} I & A^{-1}B \\ O & I \end{bmatrix},\end{aligned} \tag{8}$$

where $S_s = \left(\alpha\hat{S} - CA^{-1}B\right)$.

The inverse of the preconditioner matrix $\mathcal{P}_{\alpha,\hat{S}}$ is given by

$$\mathcal{P}^{-1}_{\alpha,\hat{S}} = \begin{bmatrix} I & -A^{-1}B \\ O & I \end{bmatrix}\begin{bmatrix} A^{-1} & O \\ O & S_s^{-1} \end{bmatrix}\begin{bmatrix} I & O \\ CA^{-1} & I \end{bmatrix}. \tag{9}$$

The eigenpair distribution of the preconditioned matrix illustrate that the iterative solvers converge after a many number of iterations. Therefore, it is significant to illustrate the spectrum of the preconditioned matrix $\mathcal{P}^{-1}_{\alpha,\hat{S}}\mathcal{A}$. The following theorem describes the eigenvalue distribution of the preconditioned matrix

$$\mathcal{P}^{-1}_{\alpha,\hat{S}}\mathcal{A} = \begin{bmatrix} I_n & K_1 \\ 0 & K_2 \end{bmatrix}. \tag{10}$$

Where:

$$S_s = \left(\alpha\hat{S} - CA^{-1}B\right),$$
$$K_1 = \left(I + A^{-1}BS_s^{-1}C\right)A^{-1}B \text{ and } K_2 = -S_s^{-1}CA^{-1}B.$$

Theorem 1: Let the preconditioned matrix be defined as in Eq. (10) and B has a full colmn rank. Then $\mathcal{P}^{-1}_{\alpha,\hat{S}}\mathcal{A}$ has m_1+1 distincts eigenvalues $\{1,\lambda_1,\ldots,\lambda_{m_1}\}$ and $(n+m)$ linearly independent eigenvectors, where $1 \le m_1 \le m$. Furthermore.

- Eigenvalue equal to 1 and its algebraic multiplicity is n.
- If $\frac{\alpha}{2} > Real(\mu)$, then $|\lambda_i| < 1$, for $i = 1,..,m_1$, where μ is the eigenvalue of the matrix $\hat{S}^{-1}CA^{-1}B$.
- If $\frac{\alpha}{2} < Real(\mu)$, then $|\lambda_i| > 1$, for $i = 1,..,m_1$.
- If $\hat{S}^{-1}CA^{-1}B$ is diagonalizable, then $\mathcal{P}^{-1}_{\alpha,\hat{S}}\mathcal{A}$ is diagonalizable.

Proof: Let $\left(\lambda,[x^T,y^T]^T\right)$ be an eigenpair of the preconditioned matrix $\mathcal{P}^{-1}_{\alpha,\hat{S}}\mathcal{A}$ and $[x^T,y^T]^T$. To obtain the spectrum of the matrix , we use following generalized eigenvalue problem:

$$\mathcal{P}^{-1}_{\alpha,\hat{S}}\mathcal{A}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} I_n & K_1 \\ 0 & K_2 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \lambda\begin{bmatrix} x \\ y \end{bmatrix}. \tag{11}$$

Rewriting Eq. (11) equation can be rewritten as follow

$$\begin{cases}(1-\lambda)x = -K_1y, \\ CA^{-1}By = \dfrac{\alpha\lambda}{(\lambda-1)}\hat{S}y.\end{cases} \tag{12}$$

If $\lambda = 1$ satisfies the first equation of Eq. (12), then we get $y = 0$. Then the n eigenvectors $\begin{pmatrix} u^{(i)} \\ 0 \end{pmatrix}$, $(i = 1,..,n)$, corresponding to the eigenvalue equal to 1 are linearly independent, where the arbitrary subset $u^{(i)}$ of will be linearly independent.

If we assume that $\lambda \neq 1$ and $y = 0$, from Eq. (12) we get that $x = 0$. Then we are deriving a contradiction from the initial assumption that $\left(\lambda,[x^T,y^T]^T\right)$ is an eigenpair of the preconditioned matrix $\mathcal{P}^{-1}_{\alpha,\hat{S}}\mathcal{A}$, consequently y cannot equal to zero. We assume that y satisfies the second equation of Eq. (12), then

$$CA^{-1}By = \mu\hat{S}y, \tag{13}$$

where

$$\mu = \frac{\alpha\lambda}{(\lambda-1)}.$$

We deduce that

$$\lambda = \frac{\mu}{\mu-\alpha}. \tag{14}$$

Eq. (14) can be rewritten as follow:

$$\lambda = \frac{a+ib}{a-\alpha+ib}, \tag{15}$$

a and b are the real and imaginary components of the complexe number μ. Hence if $\frac{\alpha}{2} > a$, this by doing some easy compuatation we get the following inequality of λ:

$$|\lambda| < 1.$$

If $\frac{\alpha}{2} < a$, then

$$|\lambda| > 1.$$

$$\hat{S}^{-1}CA^{-1}By = \mu y. \tag{16}$$

As we know that the matrix $\hat{S}^{-1}CA^{-1}B$ is a diagonalizable, thus the m eigenvectors $w^{(i)}$, $(i = 1,..,m)$ of the corresponding matrice are lineary independent. We define the matrices U, V and W by the component $\left(u^{(1)},\ldots,u^{(n)}\right)$, $\left(v^{(1)},\ldots,v^{(m)}\right)$ and $\left(w^{(1)},\ldots,w^{(m)}\right)$, respectively, where the component of V, are defined as follow:

$$v^{(i)} = \frac{K_1}{(\lambda-1)}w^{(i)}. \tag{17}$$

As the matrices U and W are invertibles, we conclude that $\det\begin{pmatrix} U & V \\ O & W \end{pmatrix} = \det(U)\det(W) \neq 0$ and $\mathcal{P}^{-1}_{\alpha,\hat{S}}\mathcal{A}$ is diagonalizable. □

If we assume that the assumptions of the above theorem are satisfied, the preconditioned Krylov subspace methods converge in at most m_1+1 iterations. Furthermore, in Theorem 1, we could remark that the bound depends only on the spectre of the chosen preconditioned matrix $\{1,\lambda_1,...,\lambda_{m1}\}$ and on the matrix of eigenvectors. The preconditioned matrix can be rewritten as follow:

$$\mathcal{P}_{\alpha,\hat{S}} = \mathcal{H} + \mathcal{S}.$$

Where the following matrices are Hermitian matrix

$$\mathcal{H} = \begin{bmatrix} I_n & \frac{K_1}{2} \\ \frac{K_1^T}{2} & \frac{K_2+K_2^T}{2} \end{bmatrix}, \tag{18}$$

and skew-Hermitian matrix

$$\mathcal{S} = \begin{bmatrix} O & \frac{K_1}{2} \\ -\frac{K_1^T}{2} & \frac{K_2-K_2^T}{2} \end{bmatrix}. \tag{19}$$

Lemma 1: Assume that B has a full column rank and $\mathcal{H}$ be defined as in Eq. (18), $D = O$ and $\hat{S} = S$. Then the matrix $\mathcal{H}$ is positive definite, for $\alpha^* = 2$ if and only if:

$$\lambda_{\min} < 3. \tag{20}$$

Where $\lambda_{\min}$ is the minimal eigenvalue of the matrix $B^T A^{-T} A^{-1} B$.

Proof: $\mathcal{H}$ has the block-triangular factorization

$$\mathcal{H} = \begin{bmatrix} I & O \\ \frac{K_1^T}{2} & I \end{bmatrix} \begin{bmatrix} I & O \\ O & S_{\mathcal{H}} \end{bmatrix} \begin{bmatrix} I & \frac{K_1}{2} \\ O & I \end{bmatrix}, \tag{21}$$

where $S_{\mathcal{H}} = \left(\frac{K_2+K_2^T}{2} - \frac{K_1^T K_1}{4} \right)$. From Eq. (21), we infer that $\det(\mathcal{H}) = \det(S_{\mathcal{H}})$, then the matrix $\mathcal{H}$ is positive definite if and only if $S_{\mathcal{H}}$ is positive definite matrix. Now we show that $S_{\mathcal{H}}$ is SPD, by proving that for all vector $y \neq 0$, we have the following inequality:

$$y^T S_{\mathcal{H}} y > 0.$$

Which can be written as

$$y^T \left(\frac{1}{\alpha+1} - \frac{\alpha^2}{4(\alpha+1)^2} B^T A^{-T} A^{-1} B \right) y > 0. \tag{22}$$

Premultiplying Eq. (22) with $\frac{y}{y^T y}$ gives

$$\frac{1}{\alpha+1} > \frac{\alpha^2}{4(\alpha+1)^2} \frac{y^T B^T A^{-T} A^{-1} B y}{y^T y}. \tag{23}$$

Then after some manipulations and by using the fact that $\alpha^* = 2$ is the minimum of the function $g(\alpha) = \frac{\alpha^2}{4(\alpha+1)^2}$, the matrix $\mathcal{H}$ is positive definite, for $\alpha^* = 2$ if and only if:

$$\lambda_{\min} < 3. \tag{24}$$

Where $\lambda_{\min}$ is the minimal eigenvalue of the matrix $B^T A^{-T} A^{-1} B$. Thus, the proof of Lemma 1 is completed. □

Lemma 2: Let $\mathcal{S}$ be defined as in Eq. (19), then $\mathcal{S}$ is purely imaginary.

Proof: Let $X = [x^T, y^T]^T \neq 0$ satisfying the following equality:

$$X^T \mathcal{S} X = \begin{bmatrix} x \\ y \end{bmatrix}^T \begin{bmatrix} O & \frac{K_1}{2} \\ -\frac{K_1^T}{2} & \frac{K_2-K_2^T}{2} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{25}$$

Which can be written as:

$$\begin{aligned} X^T \mathcal{S} X =& y^T \left(\frac{K_2 - K_2^T}{2} \right) y \\ =& i z_2, \end{aligned}$$

where $y^T K_2 y = z_1 + i z_2$. Thus, the proof of Lemma 2 is completed. □

The following theorem readily follows from Lemmas 1-2.

Theorem 2: Let the preconditioned matrix be defined as in Eq. (10) and B has a full colmn rank. Then the real part of the eigenvalues of $\mathcal{P}_{\alpha,\hat{S}}^{-1} \mathcal{A}$ is strictly positive.

5. Numerical experiments

In this section, we carry out the numerical example simulations, to asses the relative efficiencies of the various algebraic numerical methods described that illustrate the convergence behavior of the preconditioned GMRES [10], BICGSTAB [10], and FGMRES [11] methods incorporated with the proper preconditioner. All of the simulation are run with parallel computer architecture. In the meanwhile the feasibility and effectiveness of the approaches, from the point of view of the elapsed Central Process Units time (denoted by CPU) as well as the number of iterations of the preconditioned iterative approach. The norm of residual vector for the linear algebraic system is called Residual, "†" mean that the method fail to converge. Where $\mathcal{P}_{\alpha,\hat{S}}$, $\mathcal{P}_T$ and $\mathcal{P}_D$ are given as follows:

$$\mathcal{P}_{\alpha,\hat{S}} = \begin{bmatrix} \hat{A} & \hat{B} \\ \hat{C} & \alpha\hat{S} \end{bmatrix}, \quad \mathcal{P}_T = \begin{bmatrix} \hat{A} & O \\ \hat{C} & \hat{S} \end{bmatrix} \text{ and } \mathcal{P}_D = \begin{bmatrix} \hat{A} & O \\ O & \hat{S} \end{bmatrix}. \tag{26}$$

The matrices $\hat{A}$, $\hat{B}$, $\hat{C}$ and $\hat{S}$ called the approximation of A, B, C and $D - CA^{-1}B$ respectively. The parameter of the $\mathcal{P}_{\alpha,\hat{S}}$ preconditioner is chosen as to perform the chosen preconditioner dexterously, the challenging problem it is how to compute α to decreases the number of iterations steps and improve the performance of the chosen iterative solver. In Tables 2, 3 and 4, to perform the regularized preconditioner dexterously, we must to compute the α proprely that's why we use the following formula: $\alpha = \left(\|\hat{B}\|_2 \|\hat{C}\|_2 \right) / (\|\hat{A}\|_2 \|\hat{S}\|_2)$, this forula keep the balances of the euclidean norm of chosen matrices . When solving equation Eq. (1) using any of the preconditioned Krylov subspace methods described above, the iterations are stopped as soon as the Euclidean norm of the current residue is lower than a tolerance threshold

$$\frac{\|\mathcal{P}^{-1} b - \mathcal{P}^{-1} \mathcal{A} X^{(k)}\|_2}{\|\mathcal{P}^{-1} b\|_2} < 10^{-12}, \tag{27}$$

where and $X^{(k)} \in \mathbb{R}^{(n+m)}$ denotes the current iterate. We note that for solving the first system of Algorithm above by using any iterative Krylov subspace methods, the preconditioner incoporated with those iteratives methodes is a Lower-upper approximation, performed by using MUMPS package [7, 8]. The threshold value norm less than ($tol = 10^{-6}$).

Table 1. Simulation parameters used for the algebraic linear systems

	Size IVS model			
Test	$\mathcal{M}$-bloc(1,1)	$\mathcal{M}$-bloc(1,2)	$\mathcal{M}$-bloc(2,1)	$\mathcal{M}$-bloc(2,2)
Test 2	1503 ×1503	1503 × 14	14× 1503	14× 14
Test 3	127760×127760	127760× 380	380× 127760	380× 380

Table 2. Execution times associated with the factorization of Schur complement for each simulation tests: the indicated CPU times are in seconds

	$\hat{S}$
Test 1	CPU_{Shur}: 9.76e-03
Test 2	CPU_{Shur}: 1.71e+01

In Table 2, we reported the results of Schur complement approximation in terms of CPU times.

Table 3. Simulation results of diagonal, triangular and regularized preconditioners incorporated with GMRES

Test		$\mathcal{P}_{\alpha,\hat{S}}$	$\mathcal{P}_T$	$\mathcal{P}_D$
Test1	Iter	8	1.70e-03	3.02e-14
	CPU	23	5.65e-03	3.79e-13
	RES	10	2.66e-03	1.72e-12
Test2	Iter	2	1.36	7.94e-22
	CPU	†	†	†
	RES	†	†	†

Table 4. Simulation results of diagonal, triangular and regularized preconditioners incorporated with FGMRES

Test		$\mathcal{P}_{\alpha,\hat{S}}$	$\mathcal{P}_T$	$\mathcal{P}_D$
Test1	Iter	12	2.86e-03	2.20e-24
	CPU	30	6.79e-03	2.00e-11
	RES	29	4.78e-03	2.00e-11
Test2	Iter	2	6.12e-01	5.37e-26
	CPU	†	†	†
	RES	†	†	†

Table 5. Simulation results of diagonal, triangular and regularized preconditioners incorporated with BICGSTAB

Test		$\mathcal{P}_{\alpha,\hat{S}}$	$\mathcal{P}_T$	$\mathcal{P}_D$
Test1	Iter	15	3.68e-03	3.83e-15
	CPU	22	5.59e-03	6.75e-15
	RES	15	3.40e-03	5.40e-15
Test2	Iter	2	5.94e-01	6.27e-28
	CPU	†	†	†
	RES	†	†	†

From simulations results listed above, we remark, $\mathcal{P}_{\alpha,\hat{S}}$ preconditioner Incorporated with GMRES, FGMRES and BICGSTAB methods has a better performance than the than $\mathcal{P}_T$ and $\mathcal{P}_D$ as function of the iterations step and CPU time . The $\mathcal{P}_D$, $\mathcal{P}_T$ preconditioner do not converge in case of Test 2, whereas the $\mathcal{P}_{\alpha,\hat{S}}$ preconditioner converge in case of both Test 1 and Test 2.

6. Conclusion

In the present work, we have developed and studied numerically parallel block preconditioner for a class of linear systems arising from **IVS** model. Parallel block preconditioner have been proposed based on the approximate Schur complement (Block-Schur) and on a regularization technique. Several numerical experiments have been conducted in parallel on a parallel computer architecture to test the efficiently of the iterative solvers in terms of Krylov subspace methods iterations and computational time.

Computing the proper value of α for the chosen preconditioner is very challenging and very complicated problem.

Simulation results obtained in the above Section 5 (Tables 3, 4 and 5) reveal that the chosen parallel block preconditioned iterative solvers with proper parameter better than $\mathcal{P}_D$ and $\mathcal{P}_T$ preconditioned Krylov subspace methods as function of the iterations and CPU times, and illustrate that the regularized parallel block preconditioned Krylov method is a very strong approach for this kind of problem such as Eq. (1).

References

[1] A. Badahmane, "Regularized preconditioned gmres and the regularized iteration method," *Applied Numerical Mathematics*, vol. 152, pp. 159–168, 2020.

[2] M. Benzi and A. J. Wathen, "Some preconditioning techniques for saddle point problems," *Model order reduction: theory, research aspects and applications*, pp. 195–211, 2008.

[3] H. S. Dollar and A. J. Wathen, "Approximate factorization constraint preconditioners for saddle-point matrices," *SIAM Journal on Scientific Computing*, vol. 27, no. 5, pp. 1555–1572, 2006.

[4] A. B. V. Duwig, *New MFVISC Code for Copper and Helium in Iron under Irradiation. Research Report D-P124*. European Project PERFECT, 2004.

[5] J. Cash, "Second derivative extended backward differentiation formulas for the numerical integration of stiff systems," *SIAM Journal on Numerical Analysis*, vol. 18, no. 1, pp. 21–36, 1981.

[6] M. Benzi, G. H. Golub, and J. Liesen, "Numerical solution of saddle point problems," *Acta numerica*, vol. 14, pp. 1–137, 2005.

[7] E. Anderson, Z. Bai, C. Bischof, L. S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney *et al.*, *LAPACK users' guide*. SIAM, 1999.

[8] P. R. Amestoy, I. S. Duff, J.-Y. L'Excellent, and J. Koster, "A fully asynchronous multifrontal solver using distributed dynamic scheduling," *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 1, pp. 15–41, 2001.

[9] E. Anderson, Z. Bai, C. Bischof, L. S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney *et al.*, *LAPACK users' guide*. SIAM, 1999.

[10] Y. Saad, *Iterative methods for sparse linear systems*. SIAM, 2003.

[11] Y. Saad, "A flexible inner-outer preconditioned gmres algorithm," *SIAM Journal on Scientific Computing*, vol. 14, no. 2, pp. 461–469, 1993.