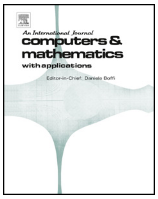
Computers and Mathematics with Applications 161 (2024) 190–201

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa



Splitting schemes for coupled differential equations: Block Schur-based approaches & Partial Jacobi approximation

Roberto Nuca [a], Erlend Storvik [b,c], Florin A. Radu [b], Matteo Icardi [d,*]

[a] CEMSE, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia
[b] Center for Modeling of Coupled Subsurface Dynamics, Department of Mathematics, University of Bergen, Allégaten 44, 5007 Bergen, Norway
[c] Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Svanehaugvegen 1, 6812 Førde, Norway
[d] School of Mathematical Sciences, University of Nottingham, NG7 2RD, Nottingham, UK

ARTICLE INFO

Dataset link: https://codeocean.com/capsule/6571121/tree

Keywords:
Partitioned multi-physics
Splitting schemes
Approximate Schur complement
Partial Jacobi
Block-iterative schemes
Sequential coupling

ABSTRACT

Coupled multi-physics problems are encountered in countless applications and pose significant numerical challenges. In a broad sense, one can categorise the numerical solution strategies for coupled problems into two classes: monolithic approaches and sequential (also known as split, decoupled, partitioned or segregated) approaches. The monolithic approaches treat the entire problem as one, whereas the sequential approaches are iterative decoupling techniques where the different sub-problems are treated separately. Although the monolithic approaches often offer the most robust solution strategies, they tend to require ad-hoc preconditioners and numerical implementations. Sequential methods, on the other hand, offer the possibility to add and remove equations from the model flexibly and rely on existing black-box solvers for each specific equation. Furthermore, when problems are non-linear, inner iterations need to be performed even in monolithic solvers, making the sequential approaches an even more viable alternative. The cost of running inner iterations to recover the multi-physics coupling could, however, easily become prohibitive. Moreover, the sequential approaches might not converge at all. In this work, we present a general formulation of splitting schemes for continuous operators with arbitrary implicit/explicit splitting, like in standard iterative methods for linear systems. By introducing a generic relaxation operator, we find the conditions for the convergence of the iterative schemes. We show how the relaxation operator can be thought of as a preconditioner and constructed based on an approximate Schur complement. We propose a Schur-based Partial Jacobi relaxation operator to stabilise the coupling and show its effectiveness. Although we mainly focus on scalar-scalar linear problems, most results are easily extended to non-linear and higher-dimensional problems. The schemes presented are not explicitly dependent on any particular discretisation methodologies. Numerical tests (1D and 2D) for two PDE systems, namely the Dual-Porosity model and a Quad-Laplacian operator, are carried out to investigate the practical implications of the theoretical results.

1. Introduction

Multi-physics phenomena can involve large systems of linear or non-linear partial differential equations. Important examples of coupled multiphysics problems include bulk coupling (where equations are solved in the same domain), such as Navier-Stokes equations, poromechanics, multi-species reactive transport, and interface couplings such as fluid-structure interaction and conjugate heat transfer, to name a few. Many theoretical and numerical studies have been devoted to these problems, resulting in optimised monolithic solution approaches that often need detailed analysis of the system, ad-hoc coupled stable discretisation techniques and pre-conditioners for the resulting (large) linear systems. This is not always achievable in computational engineering, where new coupled models often arise from the need to couple existing sub-models as black-boxes. In this direction, there have recently been efforts in developing generic frameworks capable of combining existing solvers, see [1]. These provide a layer of abstraction where solvers communicate with proper data mapping.

The code (and data) in this article has been certified as Reproducible by Code Ocean: https://codeocean.com/. More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

[*] Corresponding author.
E-mail address: matteo.icardi@nottingham.ac.uk (M. Icardi).

https://doi.org/10.1016/j.camwa.2024.02.042

The strategies adopted to solve linear systems derived by the discretisation of coupled problems can be divided into monolithic and segregated/partitioned approaches. In monolithic approaches, the whole system is discretised in a single matrix, and all unknowns are solved simultaneously. In contrast, sequential iterative approaches consider each equation/sub-problem sequentially in a sequence with an internal loop to recover the coupled structure of the full problem. We denote here by "splitting scheme", the iterative and partitioned strategy that defines this internal loop. Internal iterations in splitting schemes can be conveniently combined with the non-linear iterations (Picard or Newton) required to solve non-linear problems.

Besides pure numerical aspects, there are software engineering motivations for why segregated/partitioned approaches could be preferable to monolithic approaches. One motivation is related to memory management because only one matrix at a time, derived by one single equation, needs to be stored. A second and probably most relevant motivation is the possibility of using many efficient and robust solvers already available in the scientific computing community to solve each equation appearing in the multiphysics model.

In this work, we focus on coupled scalar-scalar linear model problems, although most of the results are valid or easily extendable to other linear problems. The objective of this work is to develop a new family of splitting schemes based on approximate Schur complements and provide a theoretical framework to analyse them. The new framework generalises standard stabilised decoupling methods such as the fixed-stress splitting scheme [2–5] and the undrained splitting scheme [6] for the Biot equations of poroelasticity. These are based on adding a diagonal relaxation operator to stabilise the sequential method. Recent works have discussed the optimisation of this relaxation term for the fixed-stress splitting scheme in [7]. Here, one of the overarching goals is to provide some insight into how more general operators can be used to stabilise coupled operators. While only linear problems are studied here, the generalisation to non-linear partial differential equations is straightforward as the splitting approaches proposed here can be easily combined with linearisation methods, such as those applied for Richards equation coupled with transport in [8]. Moreover, as with all fixed-point iterations, the methods proposed can be further improved by acceleration methods, such as the Anderson acceleration, which is known to accelerate linearly convergent schemes [9]. This has already been successfully applied in combination with decoupling methods for the Biot equations [10] and phase-field methods for brittle fracture propagation [11].

The paper is organised as follows. In section 2, we introduce two model problems, the dual-porosity problem and the Quad-Laplacian. In the next section, we present a unified nomenclature for decoupling methods, conveniently seen as block-based versions of standard iterative schemes for linear systems. We split the monolithic problem into an implicit and an explicit operator and then introduce a relaxation operator to improve and stabilise the iterations. In section 4, we extend the idea behind the Uzawa algorithm and propose a new general decoupling method for solving iteratively coupled systems of equations based on an approximation of the Schur complement. In section 5, convergence conditions for generic splitting schemes and convergence proofs for several special cases, namely the block-symmetric and block-skew-symmetric cases, are provided for the new method. Eventually, in section 6, numerical results with the Schur-based Partial Jacobi approximation are compared to standard iterative methods.

2. Model problems

In this section, we present two model problems as examples of a system of partial differential equations falling in the generic representation used in section 3, showing the conditions for well-posedness. In the last chapter, we will use these models as examples in numerical tests.

2.1. Dual-porosity Darcy's flow

The first model problem we introduce is the so-called "dual-porosity" Darcy's flow model [12], which is an effective/mixture model to describe the flow through a fractured porous medium. Here, u and v are the matrix and fracture pressures, respectively. The equations read as follows:

$$\begin{aligned} -\nabla \cdot \left(m_u \nabla u\right) &= \beta(v-u) + f_1\,, \\ -\nabla \cdot \left(m_v \nabla v\right) &= \beta(u-v) + f_2\,, \end{aligned} \tag{1}$$

where $m_{(\cdot)}$ is the *mobility* (permeability divided by the fluid viscosity), β is the *transfer coefficient* and $f_{1,2}$ are source terms. For the sake of simplicity, we consider homogeneous Dirichlet boundary conditions, but the methods developed in this paper can be applied to more general boundary conditions as well.

The continuous variational formulation of problem eq. (1) reads:

Find $(u,v) \in H^1_0(\Omega) \times H^1_0(\Omega)$ such that

$$\begin{aligned} a(u,\varphi) + b(u,\varphi) - b(v,\varphi) &= \langle f_1, \varphi\rangle \qquad \forall \varphi \in H^1_0(\Omega)\,, \\ d(v,\vartheta) + b(v,\vartheta) - b(u,\vartheta) &= \langle f_2, \vartheta\rangle \qquad \forall \vartheta \in H^1_0(\Omega)\,, \end{aligned} \tag{2}$$

where

$$a(u,\varphi) := \int_\Omega m_u \nabla u \cdot \nabla \varphi \, dx \qquad d(v,\vartheta) := \int_\Omega m_v \nabla v \cdot \nabla \vartheta \, dx$$
$$b(u,\varphi) := \int_\Omega \beta u \varphi \, dx.$$

We can sum eq. (2) and using the bi-linearity property of the form $b(\cdot,\cdot)$ to get

$$a(u,\varphi) + b(u-v,\varphi-\vartheta) + d(v,\vartheta) = \langle f_1,\varphi\rangle + \langle f_2,\vartheta\rangle$$
$$\forall (\varphi,\vartheta) \in H^1_0(\Omega) \times H^1_0(\Omega).$$

The form

$$\mathcal{A} : W \times W \to \mathbb{R} \qquad W := H^1_0(\Omega) \times H^1_0(\Omega)$$
$$((u,v),(\varphi,\vartheta)) \mapsto a(u,\varphi) + d(v,\vartheta) + b(u-v,\varphi-\vartheta)$$

is continuous and coercive under the following conditions:

$$m_u, m_v > 0\,, \beta \geq 0 \text{ a.e. and } m_u, m_v \in L^\infty(\Omega).$$

Hence, the problem eq. (1) is well-posed thanks to the Lax-Milgram Lemma. We will use this problem as a prototype of a block-symmetric coupled operator.

2.2. Quad-Laplacian

The second problem we study is composed of four Laplace operators, and we name it as Quad-Laplacian. The main motivation for introducing this synthetic model is to study scalar problems with non-trivial off-diagonal "coupling" operators. The strong form of Quad-Laplacian reads

$$\begin{aligned} -\nabla \cdot \left(m_{uu} \nabla u\right) - \nabla \cdot \left(m_{uv} \nabla v\right) &= f_1\,, \\ -\nabla \cdot \left(m_{vu} \nabla u\right) - \nabla \cdot \left(m_{vv} \nabla v\right) &= f_2\,, \end{aligned} \tag{3}$$

where $m_{(\cdot,\cdot)}$ are diffusivity coefficients. Its weak form we get

$$\begin{aligned} a(u,\varphi) + b(v,\varphi) &= \langle f_1,\varphi\rangle \qquad \forall \varphi \in H^1_0(\Omega)\,, \\ c(u,\vartheta) + d(v,\vartheta) &= \langle f_2,\vartheta\rangle \qquad \forall \vartheta \in H^1_0(\Omega)\,, \end{aligned} \tag{4}$$
$$a(u,\varphi) = \int_\Omega m_{uu} \nabla u \cdot \nabla \varphi \, dx, \qquad b(v,\varphi) = \int_\Omega m_{uv} \nabla v \cdot \nabla \varphi \, dx,$$

$$c(u,\vartheta) = \int_\Omega m_{vu} \nabla u \cdot \nabla \vartheta \, dx, \qquad d(v,\vartheta) = \int_\Omega m_{vv} \nabla v \cdot \nabla \vartheta \, dx.$$

Assuming $m_{uv} = -m_{vu}$, i.e., $b(v,u) = -c(u,v)$,

$$\mathcal{A} : W \times W \to \mathbb{R} \qquad W := H_0^1(\Omega) \times H_0^1(\Omega)$$
$$((u,v),(\varphi,\vartheta)) \mapsto a(u,\varphi) + d(v,\vartheta) + b(v,\varphi) + c(u,\vartheta)$$

is continuous and coercive under the following conditions:

$$m_{uu}, m_{vv} > 0 \text{ a.e. and } m_{uu}, m_{vv}, m_{uv} \in L^\infty(\Omega).$$

Hence, the problem eq. (3) is well-posed due to the Lax-Milgram Lemma. In the following, we will assume this condition to be valid, making the system block-skew-symmetric. This structure is typical in many coupled problems arising from mass and momentum conservation and semi-discrete poroelasticity. For the quad-Laplacian case, each operator is also symmetric (i.e., self-adjoint), making the whole system skew-symmetric.

2.3. *Well-posedness conditions and connection with saddle-point problems*

The models presented above are special cases of the following system of equations:

$$\mathcal{A}w = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = f \tag{5}$$

where A, B, C and D represent linear differential operators in the strong form. This system is intentionally left without formal definitions of operators and involved spaces, as such details are discussed in the next section. In this work, we mostly adopt the terminology and concepts from linear algebra theory. Since not all steps are equally applicable to infinite dimensional (e.g. differential) operators, we occasionally interpret and indicate specific assumptions and differences for the case of differential operators separately. System eq. (5) for Hilbert spaces is very general and a simple approach to studying the well-posedness conditions, as done in the previous section for the Dual-Porosity and Quad-Laplacian system, is deriving a bilinear form acting between the Cartesian product of proper spaces and applying the hypotheses of Lax-Milgram lemma. However, one of the most relevant cases of eq. (5) is the case of $D = 0$ and $B = C^\top$ corresponding to the well-known saddle point problem (SPP) and its generalisations. For SPP, a robust and comprehensive theory has been developed [13,14]. A generalisation of SPP where $D \neq 0$ is sometimes referred to as a perturbed saddle point problem like those arising for nearly incompressible materials, recently analysed by [15]. The SPP and perturbed SPP are examples where applying the Lax-Milgram lemma provides an unsatisfactory stability estimate because of the nature of the perturbative term [14, Sec. 4.3, p. 238]. Another generalisation of SPP is where $D = 0$ but B and C are generic. This case has been studied in [16] as a direct generalisation of Brezzi-Babuska Theory. Even non-variational methods can be used to study the well-posedness of eq. (5) provided proper spaces for the monolithic operator $\mathcal{A}$ [17, Ch. 5]. In the context of PDEs, the continuity of the operators is a natural hypothesis, while coercivity and strong monotonicity for variational and non-variational methods, respectively, is the key hypothesis to ensure the injectivity of the operator.

In the continuous setting, eq. (5) makes sense once proper spaces have been defined. Let U and V be Hilbert spaces with $u \in U$ and $v \in V$. We have that

$$A : U \longrightarrow U', \qquad B : V \longrightarrow U', \qquad C : U \longrightarrow V', \qquad D : V \longrightarrow V',$$

hence $f_1 \in U'$ and $f_2 \in V'$. We can use the duality pairing in both equations and sum:

$$\langle Au, \varphi \rangle_* + \langle Bv, \varphi \rangle_* + \langle Cu, \mu \rangle_\star + \langle Dv, \mu \rangle_\star = \langle f_1, \varphi \rangle_* + \langle f_2, \mu \rangle_\star \quad \forall (\varphi, \mu) \in U \times V,$$

i.e.,

$$a(u,\varphi) + b(v,\varphi) + c(u,\mu) + d(v,\mu) = \langle f_1, \varphi \rangle_\star + \langle f_2, \mu \rangle_\star \quad \forall (\varphi, \mu) \in U \times V.$$

This expression defines a bilinear form

$$\mathcal{A} : W \times W \longrightarrow \mathbb{R}$$
$$\mathcal{A}(w, \eta) = \langle f, \eta \rangle \qquad \forall \eta \in U \times V,$$

where $W := U \times V$ and f is the functional defined by the sum of f_1 and f_2. In the following, we will always assume $a(\cdot,\cdot)$ and $d(\cdot,\cdot)$ to be continuous and coercive with coercivity constants α_a and α_d, respectively (i.e., the discrete operator A and D to be positive definite), and the monolithic operator $\mathcal{A}$ to be trivially well-posed by assuming its continuity and coercivity. This can be a consequence of the hypotheses introduced for the dual-porosity or the quad-Laplacian model problems, or, more in general, by assuming that there exists a coercivity constant $\alpha_{\mathcal{A}} > 0$ such that

$$0 < \alpha_{\mathcal{A}} \le \min(\alpha_a, \alpha_d) - \frac{\|b\| + \|c\|}{2}, \tag{6}$$

where α_a and α_d are the coercivity constants of the bilinear forms $a(\cdot,\cdot)$ and $d(\cdot,\cdot)$ respectively. In fact we have that, $\forall u, v \in U \times V$,

$$|b(v,u) + c(u,v)| \le (\|b\| + \|c\|) \|v\| \|u\| \le \frac{\|b\| + \|c\|}{2} \left(\|v\|^2 + \|u\|^2 \right).$$

Throughout the paper, although something else is specified, we use Euclidean norms for vectors and the induced matrix norm. In the continuous case, on any generic normed spaces $(V, ||\cdot||_V), (W, ||\cdot||_W)$, we are using the operator norm $||A|| = sup_{u \neq 0} \frac{||Au||_W}{||u||_V}$. In the following analysis and the convergence theorems below, one can think of discrete norms. For any particular model problem, one would have different (then specified) continuous norms. The choice of $\alpha_{\mathcal{A}}$ gives

$$|b(v,u) + c(u,v)| \le \left(\alpha_a - \alpha_{\mathcal{A}} \right) \|u\|^2 + \left(\alpha_d - \alpha_{\mathcal{A}} \right) \|v\|^2,$$

and, therefore,

$$b(v,u) + c(u,v) \ge \left(\alpha_{\mathcal{A}} - \alpha_a \right) \|u\|^2 + \left(\alpha_{\mathcal{A}} - \alpha_d \right) \|v\|^2,$$

where this last inequality implies the coercivity of $\mathcal{A}$.

3. Block-iterative methods for coupled PDEs

Classic static iterative methods for linear systems can be adapted to block matrices. In our case, the block pattern of the coupled operator $\mathcal{A}$ in eq. (5) is determined by the number of equations and unknowns, while the dimensions and properties of each block are determined by the mesh size and discretisation scheme adopted for that particular equation. Convergence analysis for the particular case of positive definite matrices is treated in [18, p. 47]. The same author provides rates of convergence in the case of the Poisson equation. In the following paragraphs, we recall the classical (static) iterative methods in their block extensions. The superscript index indicates the iterative step. In general, any iterative splitting/decoupling scheme can be represented by a decomposition of the operator $\mathcal{A}$ into an implicit part $\mathcal{A}_i$ and an explicit part $\mathcal{A}_e$:

$$\mathcal{A}_i w + \mathcal{A}_e w = \mathcal{A} w = f. \tag{7}$$

This can be rewritten as a stationary iterative scheme as

$$\mathcal{A}_i w^{k+1} + \mathcal{A}_e w^k = f. \tag{8}$$

While this splitting is often introduced also in monolithic schemes to, for example, derive operator-based preconditioners [19], here, for the resulting system to be “decoupled”, we require $\mathcal{A}_i$ is a block-triangular operator. For clarity, we now introduce the block version of the most straightforward stationary iterative schemes: Jacobi, Gauss-Seidel, and SOR.

Block-Jacobi Block-Jacobi is the simplest iterative method, and its block-version can be formulated as follows:

$$Au^{k+1} = f_1 - Bv^k, \tag{9}$$

$$Dv^{k+1} = f_2 - Cu^k, \tag{10}$$

with

$$\mathcal{A}_i = \begin{bmatrix} A & 0 \\ 0 & D \end{bmatrix}, \qquad \mathcal{A}_e = \begin{bmatrix} 0 & B \\ C & 0 \end{bmatrix}.$$

Block-Gauss-Seidel Analogously to the classical Gauss-Seidel method, the block-Gauss-Seidel method (starting from the first equation) reads

$$Au^{k+1} = f_1 - Bv^k, \tag{11}$$

$$Dv^{k+1} = f_2 - Cu^{k+1}, \tag{12}$$

$$\mathcal{A}_i = \begin{bmatrix} A & 0 \\ C & D \end{bmatrix}, \qquad \mathcal{A}_e = \begin{bmatrix} 0 & B \\ 0 & 0 \end{bmatrix}.$$

This standard approach is implemented in commercial and applied research codes for solving multi-physics problems.

Block-SOR The block version of the SOR method (starting from the first equation) can be written as follows:

$$Au^{k+1} = (1-\omega)Au^k + \omega\left[f_1 - Bv^k\right], \tag{13}$$

$$Dv^{k+1} = (1-\omega)Dv^k + \omega\left[f_2 - Cu^{k+1}\right], \tag{14}$$

or, equivalently,

$$Au^{k+1} + Bv^k + \frac{(1-\omega)}{\omega}A(u^{k+1} - u^k) = f_1, \tag{15}$$

$$Dv^{k+1} + Cu^{k+1} + \frac{(1-\omega)}{\omega}D(v^{k+1} - v^k) = f_2, \tag{16}$$

$$\mathcal{A}_i = \begin{bmatrix} \left(\frac{1}{\omega}\right)A & 0 \\ C & \left(\frac{1}{\omega}\right)D \end{bmatrix}, \qquad \mathcal{A}_e = \begin{bmatrix} \left(\frac{\omega-1}{\omega}\right)A & B \\ 0 & \left(\frac{\omega-1}{\omega}\right)D \end{bmatrix}.$$

3.1. Convergence properties

Theorem 1. *The convergence of the iteration eq.* (8) *is guaranteed if*

$$\alpha_{\mathcal{A}} > 2\|\mathcal{A}_e\|, \tag{17}$$

or

$$\alpha_{\mathcal{A}_i} > \|\mathcal{A}_e\|, \tag{18}$$

where $\alpha_{\mathcal{A}}$ is the coercivity constant of the monolithic operator, $\alpha_{\mathcal{A}_i}$ is the coercivity constant of the implicit part, and $\|\mathcal{A}_e\|$ is the continuity constant of the explicit part of the operator.

Proof. From eq. (8) we can derive the following error equations for $e^k := w - w^k$:

$$\mathcal{A}e^{k+1} = \mathcal{A}_e(e^{k+1} - e^k); \qquad \mathcal{A}_i e^{k+1} = -\mathcal{A}_e e^k \tag{19}$$

If we multiply by e^{k+1} both equations, use the coercivity and the Cauchy-Schwarz inequality we obtain the following two bounds

$$\alpha_{\mathcal{A}}\|e^{k+1}\| \le \|\mathcal{A}_e\|\|e^{k+1} - e^k\|; \qquad \alpha_{\mathcal{A}_i}\|e^{k+1}\| \le \|\mathcal{A}_e\|\|e^k\|,$$

i.e.,

$$\left(\frac{\alpha_{\mathcal{A}}}{\|\mathcal{A}_e\|} - 1\right)\|e^{k+1}\| \le \|e^k\|; \qquad \frac{\alpha_{\mathcal{A}_i}}{\|\mathcal{A}_e\|}\|e^{k+1}\| \le \|e^k\|;$$

The fixed-point iterations converge if the LHS coefficient is larger than one, giving the conditions eqs. (17) and (18). □

3.2. Relaxation schemes for iterative splitting

One might find that the block Gauss-Seidel method eqs. (11) and (12) might not converge (for example, it might not satisfy eq. (17)). A remedy is to stabilise the equations, similar to what is done in the *fixed-stress* [2,4,3,7] or *undrained* [6] splitting methods for poroelasticity equations. We introduced here a generalised stabilised/relaxed iteration as follows: Given v^k, u^k find v^{k+1}, u^{k+1} such that

$$Au^{k+1} + Bv^k + L_u\left(u^{k+1} - u^k\right) = f_1 \tag{20}$$

$$Cu^{k+1} + Dv^{k+1} + L_v\left(v^{k+1} - v^k\right) = f_2, \tag{21}$$

or, equivalently,

$$\left(\mathcal{A}_i + \mathcal{L}\right)w^{k+1} = \left(\mathcal{L} - \mathcal{A}_e\right)w^k + f, \tag{22}$$

where

$$\mathcal{L} = \begin{bmatrix} L_u & 0 \\ 0 & L_v \end{bmatrix}$$

is the relaxation operator and, contrarily to what is typically done in literature, is not necessarily a diagonal operator. It is interesting to notice that there is a strong link between this relaxation and the Block-SOR method where, in the SOR method, the relaxation operators are $\frac{1}{\omega}A$ and/or $\frac{1}{\omega}D$. Although any relaxation operator can be included in the definition of $\mathcal{A}_i$ and $\mathcal{A}_e$, we treat these terms separately to study their influence better.

One of the difficulties with stabilisation is choosing the appropriate operator $\mathcal{L}$, as its choice will significantly influence the scheme's performance. In section 4, the relation between the stabilised Gauss-Seidel method and an approximate Schur complement approach will be discussed, providing guidelines to choose the stabilisation operator $\mathcal{L}$. Moreover, in section 5 convergence proofs for several cases of the stabilised block Gauss-Seidel method eqs. (20) and (21) are provided, which gives a theoretical justification for the method.

ℓ-scheme A simple but effective choice of relaxation for the operators L_u is to be a multiple of the identity and $L_v = 0$:

$$\mathcal{L} = \begin{bmatrix} \ell I & 0 \\ 0 & 0 \end{bmatrix},$$

with $\ell > 0$. This scheme has been used extensively to solve, for example, poromechanics equations, see, e.g. [4]. When applied to poromechanics, it is often called *fixed-stress* splitting, and it is the standard over-relaxation approach implemented in many commercial codes to increase the diagonal dominance of the first equation and stabilise the iteration.

4. Approximate Schur-based methods

The Schur complement can be interpreted as the Gaussian elimination formula for block matrices. One of the Schur complement's most important applications is for the Saddle Point Problem arising, for example, when solving the incompressible Navier-Stokes equations. The well-known Chorin-Temam algorithm and the analogous Yosida method rely on the block-LU decomposition [20], which leads to a Schur complement. These two are special cases of the Uzawa algorithm introduced to solve the SPP [21]. A similar method has been applied for the case of coupled energy equations [22].

To generalise these approaches for generic block systems, we introduce a general iterative algorithm to stabilise the convergence of iterative coupled schemes with approximate Schur complements by mimicking the incomplete block-LU factorisation. Furthermore, in section 4.2, we propose simple diagonal approximations to avoid explicitly computing matrix inverses, although more advanced techniques could also be used [23]. The same strategy has been widely applied to derive optimal preconditioners [24].

It is important to notice that if the exact Schur complement were used, the iterative scheme would converge to the exact solution in one iteration. Therefore, we expect that, by using an approximation, we can stabilise the iterations (ensuring their convergence) and reduce the number of iterations needed for convergence. In this sense, we could also consider these methods as accelerators.

4.1. Approximate Schur complement factorisation

Let us consider the decomposition

$$Au = \mathbb{A}u + (A - \mathbb{A})u \tag{23}$$

where $\mathbb{A}$ is an approximation of A with the property of being computationally easy to invert, for example, the diagonal of A. We introduce the block-Schur matrix factorisation

$$S = \begin{bmatrix} I & 0 \\ -C\mathbb{A}^{-1} & I \end{bmatrix}$$

which, left-multiplied to the LHS and RHS of the original system eq. (5), gives

$$\begin{bmatrix} A & B \\ -C\mathbb{A}^{-1}(A - \mathbb{A}) & D - C\mathbb{A}^{-1}B \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 - C\mathbb{A}^{-1}f_1 \end{bmatrix} \tag{24}$$

which is an approximate Schur-based decomposition. If we apply the same technique to the first equation, by using the splitting $Dv = \mathbb{D}v + (D - \mathbb{D})v$ and

$$\begin{bmatrix} I & -B\mathbb{D}^{-1} \\ 0 & I \end{bmatrix}$$

instead of S, combining the results we obtain

$$\begin{bmatrix} A - B\mathbb{D}^{-1}C & -B\mathbb{D}^{-1}(D - \mathbb{D}) \\ -C\mathbb{A}^{-1}(A - \mathbb{A}) & D - C\mathbb{A}^{-1}B \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} f_1 - B\mathbb{D}^{-1}f_2 \\ f_2 - C\mathbb{A}^{-1}f_1 \end{bmatrix} \tag{25}$$

which we denote as an alternate approximate Schur-based decomposition.

These approximate Schur complements do not provide a decoupled system (i.e., a block-triangular matrix); therefore, the next step is to apply a block-iterative method. If we apply the Gauss-Seidel method, the identity eq. (24) results in the following iterative scheme:

$$(D - C\mathbb{A}^{-1}B)v^{k+1} = f_2 - C\mathbb{A}^{-1}(f_1 - (A - \mathbb{A})u^k), \tag{26}$$

$$Au^{k+1} = f_1 - Bv^{k+1}. \tag{27}$$

Using the fact $Au^k + Bv^k = f_1$ from the previous iteration, eq. (27) and eq. (26) can be rewritten in the following form:

$$Cu^k + Dv^{k+1} - C\mathbb{A}^{-1}B\left(v^{k+1} - v^k\right) = f_2, \tag{28}$$

$$Au^{k+1} + Bv^{k+1} = f_1, \tag{29}$$

where we can identify the relaxation operator $\mathcal{L}$ as

$$\mathcal{L} = \begin{bmatrix} 0 & 0 \\ 0 & -C\mathbb{A}^{-1}B \end{bmatrix}.$$

In this formulation, it is interesting to notice the role of the approximate Schur complement $C\mathbb{A}^{-1}B$ as a stabilisation/acceleration term. Similarly, from eq. (25), one can obtain the iterative scheme for the alternate approximate Schur-based version, i.e.

$$(A - B\mathbb{D}^{-1}C)u^{k+1} = f_1 - B\mathbb{D}^{-1}(f_2 - (D - \mathbb{D})v^k), \tag{30}$$

$$(D - C\mathbb{A}^{-1}B)v^{k+1} = f_2 - C\mathbb{A}^{-1}(f_1 - (A - \mathbb{A})u^{k+1}). \tag{31}$$

As it will be clarified in the next section, it is desirable to reformulate the problem again as a relaxed iteration, i.e.,

$$Bv^k + Au^{k+1} - B\mathbb{D}^{-1}C\left(u^{k+1} - u^k\right) = f_1, \tag{32}$$

$$Cu^{k+1} + Dv^{k+1} - C\mathbb{A}^{-1}B\left(v^{k+1} - v^k\right) = f_2, \tag{33}$$

with a relaxation operator

$$\mathcal{L} = \begin{bmatrix} -B\mathbb{D}^{-1}C & 0 \\ 0 & -C\mathbb{A}^{-1}B \end{bmatrix}.$$

This alternate algorithm eq. (32), however, written in the relaxation form, is no longer exactly equivalent to the form directly obtained from the Schur factorisation eq. (30). Here, we can no longer guarantee that the system is consistent in the coupled sense for each iteration k, i.e., $Au^k + Bv^k \neq f_1$ and $Cu^k + Dv^k \neq f_2$. In the following, we will focus on the second (relaxation-like) form, although our numerical results showed that the two have negligible differences in the convergence properties.

It is essential to notice that the relaxation operator $\mathcal{L}$ constructed as above is not always coercive. As discussed in section 5, this means that $\mathcal{L}$ can act as a stabilisation (e.g., in the Quad-Laplacian problem) or as an acceleration (e.g., in the dual-porosity problem).

4.2. Diagonal approximations of Schur operators

In section 4, we have introduced a general method to solve coupled problems iteratively based on approximated Schur complements. In section 5, we have shown that provided some key assumptions for this relaxing operator, the splitting scheme converges. In this section, we introduce practical strategies to approximate the Schur complement, suitable to be easily implemented in PDE toolboxes. These approaches are tested in section 6.

4.2.1. Schur-based Partial Jacobi

The first case we consider here is when the approximate Schur-based iteration eq. (28) (single) or eq. (30) (alternate) is build approximating A by its diagonal (Partial Jacobi). We denote this approach as Schur-based Partial Jacobi (SPJ). In its "alternate" version (applying the relaxation to both equations), the relaxation operator is:

$$\mathcal{L} = \begin{bmatrix} -B\text{diag}(D)^{-1}C & 0 \\ 0 & -C\text{diag}(A)^{-1}B \end{bmatrix}.$$

The resulting operator is still the product of two potentially full operators and a diagonal. While discretising this operator (or assembling it from existing sparse matrices) in the general case could be not straightforward, the case of the dual-porosity model discussed in section 2, where the operators B and C are diagonal, is particularly simple.

The accuracy of the diagonal approximation is directly linked to the condition number of the matrix and, therefore, to the mesh size. For example, for a Laplacian operator on a uniform mesh, the diagonal approximation has a degenerate spectrum with a single repeated eigenvalue, while the full operator spectrum contains separate eigenvalues (with the ratio between the largest and the smallest eigenvalue being the condition number). This effect will be investigated numerically in section 6.

4.2.2. Schur-based Double Partial Jacobi

The SPJ approach could lead to non-sparse matrices, even if A, B, C and D are sparse. In fact, the products $B\mathbb{D}^{-1}C$ and $C\mathbb{A}^{-1}B$ are not sparse in general. Therefore, we approximate two of the three matrices involved in the two products to keep the sparsity property. By approximating A and C, the system reads

$$\begin{bmatrix} \mathbb{A} & B \\ \mathbb{C} & D \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} f_1 - (A - \mathbb{A})u \\ f_2 - (C - \mathbb{C})u \end{bmatrix}$$

that can then be solved iteratively with block-Gauss-Seidel as

$$(D - \mathbb{C}\mathbb{A}^{-1}B)v^{k+1} = f_2 - \mathbb{C}\mathbb{A}^{-1}(f_1 + (\mathbb{A} - A)u^k) + (\mathbb{C} - C)u^k, \tag{34}$$

$$Au^{k+1} = f_1 - Bv^{k+1}. \tag{35}$$

Analogously to the SPJ case, the system can be rewritten in the form

$$Cu^k + Dv^{k+1} - \mathbb{C}\mathbb{A}^{-1}B(v^{k+1} - v^k) = f_2, \tag{36}$$

$$Au^{k+1} + Bv^{k+1} = f_1 \tag{37}$$

which is equivalent to eq. (28) but with a different stabilisation/acceleration term $\mathbb{C}\mathbb{A}^{-1}B$.

If both approximations are diagonal, the resulting matrices are all sparse, and we denote this approach as the double SPJ method (S2PJ). This can be done only if the coupling operators B, C are square matrices/operators. In its "alternate" version (applying the relaxation to both equations), the S2PJ relaxation operator is:

$$\mathcal{L} = \begin{bmatrix} -\text{diag}(B)\text{diag}(D)^{-1}C & 0 \\ 0 & -\text{diag}(C)\text{diag}(A)^{-1}B \end{bmatrix}.$$

This method is readily applicable to any coupled scalar problem as the resulting system does not involve any product of operators.

When the two operators approximated by a diagonal are of the same type (e.g., the Quad-Laplacian problem, see numerical results in section 6), this method has the advantage of being no longer strongly dependent on the condition number of the matrix A, as the spectrum of BD^{-1} is, in fact, clustered and well approximated by the diagonals.

5. Convergence analysis of relaxed splitting schemes

In this section, we study the convergence properties of the generic splitting scheme with relaxation, eq. (22). We will first propose a more general result based on the properties of the monolithic operators $\mathcal{L}$ and $\mathcal{A}$, following [17,25]. Then we specialise the analysis for the specific cases of interest, namely the block-symmetric case (such as the dual-porosity problem eq. (1)), and the block-skew-symmetric case (such as the quad-Laplacian eq. (3) with $m_{uv} = -m_{vu}$). In the following, we will make an extensive use of the following

Lemma 1. *Let S a self-adjoint linear operator, then the following identity holds*

$$\langle x, S(x-y)\rangle = \frac{1}{2}\left(\langle x, Sx\rangle + \langle x-y, S(x-y)\rangle - \langle y, Sy\rangle\right) \qquad \forall x, y.$$

Proof. It is sufficient to expand both sides of the equation and use the fact that $\langle Sy, x\rangle = \langle Sx, y\rangle$ since $S = S^\top$. □

5.1. Convergence of the monolithic iteration

The generic relaxed splitting scheme can be written as

$$\mathcal{L}w^{k+1} + \mathcal{A}_i w^{k+1} = f - \mathcal{A}_e w^k + \mathcal{L}w^k. \tag{38}$$

Theorem 2. *Under the assumptions of $\mathcal{L}$ symmetric and coercive, and $\mathcal{A}$ coercive, the iteration eq. (38) converges if*

$$\frac{\|\mathcal{A}_e\|^2}{2\alpha_{\mathcal{L}}} \le 2\alpha_{\mathcal{A}} - \varepsilon_{\mathcal{L}}\alpha_{\mathcal{L}} \tag{39}$$

where $\varepsilon_{\mathcal{L}} \ge 0$ is the constant such that $\|\mathcal{L}\| = (1+\varepsilon_{\mathcal{L}})\alpha_{\mathcal{L}}$. Furthermore, the optimal convergence rate, obtained choosing $\alpha_{\mathcal{L}}^{opt} = \frac{\|\mathcal{A}_e\|^2}{2\alpha_{\mathcal{A}}}$ is

$$r(\alpha_{\mathcal{L}}^{opt}) = \sqrt{(1+\varepsilon_{\mathcal{L}})\frac{\|\mathcal{A}_e\|^2}{\|\mathcal{A}_e\|^2 + 2\alpha_{\mathcal{A}}^2}}. \tag{40}$$

Proof. Subtracting the iteration eq. (38) from the exact equation (7), and substituting $\mathcal{A}_i = \mathcal{A} - \mathcal{A}_e$, the error equation reads:

$$\mathcal{L}(e^{k+1} - e^k) + \mathcal{A}e^{k+1} = \mathcal{A}_e(e^{k+1} - e^k), \tag{41}$$

where $e^k := w - w^k$. The multiplication of eq. (41) by e^{k+1} and the application of Lemma 1 leads to

$$\langle e^{k+1} - e^k, \mathcal{L}(e^{k+1} - e^k)\rangle + 2\langle e^{k+1}, \mathcal{A}e^{k+1}\rangle + \langle e^{k+1}, \mathcal{L}e^{k+1}\rangle$$
$$= \langle e^k, \mathcal{L}e^k\rangle + 2\langle e^{k+1}, \mathcal{A}_e(e^{k+1} - e^k)\rangle.$$

Thanks to the coercivity of $\mathcal{A}$ and $\mathcal{L}$ applied to the LHS of the equation and the Cauchy-Schwarz/δ-Cauchy-Schwarz inequality on the last term of the RHS we obtain

$$\alpha_{\mathcal{L}}\|e^{k+1} - e^k\|^2 + 2\alpha_{\mathcal{A}}\|e^{k+1}\|^2 + \langle e^{k+1}, \mathcal{L}e^{k+1}\rangle$$
$$\le \langle e^k, \mathcal{L}e^k\rangle + \delta\|e^{k+1}\|^2 + \frac{\|\mathcal{A}_e\|^2}{2\delta}\|e^{k+1} - e^k\|^2,$$

we can collect the terms and obtain

$$\left(\alpha_{\mathcal{L}} - \frac{\|\mathcal{A}_e\|^2}{2\delta}\right)\|e^{k+1} - e^k\|^2 + \left(2\alpha_{\mathcal{A}} - \delta\right)\|e^{k+1}\|^2 + \langle e^{k+1}, \mathcal{L}e^{k+1}\rangle$$
$$\le \langle e^k, \mathcal{L}e^k\rangle. \tag{42}$$

To make the first term in the LHS of eq. (42) non-negative, we require that

$$\delta \ge \frac{\|\mathcal{A}_e\|^2}{2\alpha_{\mathcal{L}}} \tag{43}$$

and remove the first term from the LHS, leading to

$$\left(2\alpha_{\mathcal{A}} - \delta\right)\|e^{k+1}\|^2 + \langle e^{k+1}, \mathcal{L}e^{k+1}\rangle \le \langle e^k, \mathcal{L}e^k\rangle.$$

To obtain a bound for the convergence rate, we can use the coercivity and continuity of $\mathcal{L}$ to obtain

$$\left(2\alpha_{\mathcal{A}} + \alpha_{\mathcal{L}} - \delta\right)\|e^{k+1}\|^2 \le \|\mathcal{L}\|\|e^k\|^2.$$

Since $\alpha_{\mathcal{L}} \le \|\mathcal{L}\|$, there exists $\varepsilon_{\mathcal{L}} \ge 0$ such that $\|\mathcal{L}\| = (1+\varepsilon_{\mathcal{L}})\alpha_{\mathcal{L}}$, leading to the following expression

$$\left(2\alpha_{\mathcal{A}} + \alpha_{\mathcal{L}} - \delta\right)\|e^{k+1}\|^2 \le (1+\varepsilon_{\mathcal{L}})\alpha_{\mathcal{L}}\|e^k\|^2.$$

We can rewrite this expression to emphasise the rate of convergence as follows:

$$\|e^{k+1}\| \le \sqrt{\frac{(1+\varepsilon_{\mathcal{L}})\alpha_{\mathcal{L}}}{2\alpha_{\mathcal{A}} + \alpha_{\mathcal{L}} - \delta}}\|e^k\|. \tag{44}$$

The convergence is guaranteed if

$$0 < \frac{(1+\varepsilon_{\mathcal{L}})\alpha_{\mathcal{L}}}{2\alpha_{\mathcal{A}} + \alpha_{\mathcal{L}} - \delta} < 1$$

hence $\delta < 2\alpha_{\mathcal{A}} - \varepsilon_{\mathcal{L}}\alpha_{\mathcal{L}}$ and $\delta < 2\alpha_{\mathcal{A}} + \alpha_{\mathcal{L}}$. In conjunction with eq. (43) we can write

$$0 \le \frac{\|\mathcal{A}_e\|^2}{2\alpha_{\mathcal{L}}} \le \delta \le 2\alpha_{\mathcal{A}} - \varepsilon_{\mathcal{L}}\alpha_{\mathcal{L}} \tag{45}$$

which leads to eq. (39). Since we want to minimise the coefficient in front of $\|e^k\|$ we want to pick the smallest possible value of δ, which is $\|\mathcal{A}_e\|^2/2\alpha_{\mathcal{L}}$ according to eq. (45), hence we chose $\alpha_{\mathcal{L}}$ in order to minimise

$$r(\alpha_{\mathcal{L}}) := \sqrt{\frac{2(1+\varepsilon_{\mathcal{L}})\alpha_{\mathcal{L}}^2}{2\alpha_{\mathcal{L}}^2 + 4\alpha_{\mathcal{A}}\alpha_{\mathcal{L}} - \|\mathcal{A}_e\|^2}}.$$

The study of this single variate optimal problem leads to the optimal contraction rate

$$\alpha_{\mathcal{L}}^{opt} = \frac{\|\mathcal{A}_e\|^2}{2\alpha_{\mathcal{A}}},$$

which gives the rate of convergence eq. (40). □

Remark 1. To recover the classical ℓ-scheme stabilisation, where $\mathcal{L} = \ell I$, we can choose $\varepsilon_{\mathcal{L}} = 0$ and $\alpha_{\mathcal{L}} = \|\mathcal{L}\| = \ell > 0$ and the converges condition eq. (45) becomes:

$$\alpha_{\ell} \geq \frac{\|\mathcal{A}_e\|^2}{2\alpha_{\mathcal{A}}}, \tag{46}$$

with the minimiser of the contraction rate being again the equality. This means that we can always guarantee the convergence of the scheme by choosing appropriately $\alpha_{\mathcal{L}}$ for any given block-iterative approach given by $\mathcal{A}_i$ and $\mathcal{A}_e$. The presence of a more complex relaxation operator, with $\varepsilon_{\mathcal{L}} > 0$, seems to penalise the convergence rate. This is because, in this analysis, we have not exploited the operator $\mathcal{L}$ structure but used only its coercivity and continuity. This result, therefore, cannot fully explain the advantages of using more complex relaxation operators such as the Schur-based factorisation. Nevertheless, this allows us to ensure its convergence.

The stabilisation $\mathcal{L}$ could also be included in the definition of $\mathcal{A}_i$ and $\mathcal{A}_e$, and its convergence would be covered by Theorem 1. Nevertheless, this would not allow us to apply the theorem above, and for many practical applications, it is helpful to design a (coercive) extra term to stabilise the splitting. In the case of Schur-based operators $\mathcal{L}$, the coercivity is not always guaranteed (such as in the dual-porosity problem introduced above and further studied below). In a case of non-coercive $\mathcal{L}$, Theorem 1 could be applied rewriting eq. (17) as a condition on the norm of $\mathcal{L}$:

$$\|\mathcal{L}\| \leq \frac{\alpha_{\mathcal{A}}}{2} - \|\mathcal{A}_e\| \tag{47}$$

5.2. Convergence for block-Gauss-Seidel iterations with single relaxation

The proof above has the advantage of being more general, but it also requires conditions for a coercive and symmetric monolithic relaxation operator $\mathcal{L}$, chosen based on the continuity constant $\mathcal{A}_e$ and the coercivity constant of the monolithic operator $\mathcal{A}$. In this section, we consider more in detail the case of the Dual-Porosity model (where $B = C$ or $B = C^{\top}$, and the Quad-Laplacian (when $B = -C^{\top}$) and we focus on the block-Gauss-Seidel iterations, with a generic relaxation (e.g., the Schur-based explained above) applied on one equation only, i.e.,

$$Cu^k + Dv^{k+1} + L(v^k - v^{k+1}) = f_2, \tag{48}$$

$$Au^{k+1} + Bv^{k+1} = f_1, \tag{49}$$

where L is, for example, an approximate Schur complement, such as $L = C\mathbb{A}^{-1}B$.

5.2.1. Block-skew symmetric case $C = -B^{\top}$

We first consider the block-skew symmetric case, which corresponds to the Quad-Laplacian problem section 2.2

Theorem 3. *Let $C = -B^{\top}$, L symmetric and coercive, such that*

$$\alpha_L \geq \frac{\|B\|^4}{\alpha_D \alpha_A^2}, \tag{50}$$

where α_L, α_D and α_A are the coercivity constants of L, D, and A respectively. Then the splitting scheme eqs. (48) and (49) converges.

Proof. Define the error functions $e_v^k := v^k - v$ and $e_u^k = u^k - u$, and subtract the exact equation eq. (5) from eq. (49) and eq. (48). The scalar product against e_u^{k+1} and e_v^{k+1}) leads to:

$$\langle De_v^{k+1}, e_v^{k+1}\rangle - \left\langle L\left(e_v^k - e_v^{k+1}\right), e_v^{k+1}\right\rangle - \left\langle B^{\top} e_u^k, e_v^{k+1}\right\rangle = 0, \tag{51}$$

$$\left\langle Ae_u^{k+1}, e_u^{k+1}\right\rangle + \left\langle Be_v^{k+1}, e_u^{k+1}\right\rangle = 0. \tag{52}$$

Adding the two equations above gives the equality

$$\langle De_v^{k+1}, e_v^{k+1}\rangle + \left\langle Ae_u^{k+1}, e_u^{k+1}\right\rangle + \left\langle L\left(e_v^{k+1} - e_v^k\right), e_v^{k+1}\right\rangle + \langle B^{\top}(e_u^{k+1} - e_u^k), e_v^{k+1}\rangle = 0. \tag{53}$$

By applying Lemma 1 we obtain

$$\begin{aligned}&\langle De_v^{k+1}, e_v^{k+1}\rangle + \left\langle Ae_u^{k+1}, e_u^{k+1}\right\rangle + \frac{1}{2}\left\langle Le_v^{k+1}, e_v^{k+1}\right\rangle + \\ &+ \frac{1}{2}\left\langle L\left(e_v^{k+1} - e_v^k\right), e_v^{k+1} - e_v^k\right\rangle - \frac{1}{2}\left\langle Le_v^k, e_v^k\right\rangle + \langle B^{\top}(e_u^{k+1} - e_u^k), e_v^{k+1}\rangle \\ &= 0.\end{aligned} \tag{54}$$

Now, using Cauchy-Schwarz and Young's inequalities on $\langle B^{\top}(e_u^{k+1} - e_u^k), e_v^{k+1}\rangle$ we obtain

$$\begin{aligned}&\langle De_v^{k+1}, e_v^{k+1}\rangle + \left\langle A\, e_u^{k+1}, e_u^{k+1}\right\rangle + \frac{1}{2}\left\langle Le_v^{k+1}, e_v^{k+1}\right\rangle \\ &+ \frac{1}{2}\left\langle L\left(e_v^{k+1} - e_v^k\right), e_v^{k+1} - e_v^k\right\rangle \\ &\leq \frac{1}{2}\left\langle Le_v^k, e_v^k\right\rangle + \delta\|e_v^{k+1}\|^2 + \frac{1}{4\delta}\left\|B^{\top}\left(e_u^{k+1} - e_u^k\right)\right\|^2\end{aligned} \tag{55}$$

where $\delta > 0$ is free to be chosen. From eq. (49), we have

$$\left\langle A(e_u^{k+1} - e_u^k), e_u^{k+1} - e_u^k\right\rangle = -\left\langle B(e_v^{k+1} - e_v^k), e_u^{k+1} - e_u^k\right\rangle, \tag{56}$$

which by the coercivity of A (with constant α_A) and Cauchy-Schwarz inequality gives

$$\alpha_A\|(e_u^{k+1} - e_u^k)\|^2 \leq \|B\|\|e_v^{k+1} - e_v^k\|\|e_u^{k+1} - e_u^k\| \tag{57}$$

and thereby

$$\|e_u^{k+1} - e_u^k\| \leq \frac{\|B\|}{\alpha_A}\|e_v^{k+1} - e_v^k\|. \tag{58}$$

Using the coercivity of A, D and L (with constants α_A, α_D and α_L respectively) we then get from eq. (56)

$$\begin{aligned}&\alpha_D\|e_v^{k+1}\|^2 + \alpha_A\|e_u^{k+1}\|^2 + \frac{1}{2}\left\langle Le_v^{k+1}, e_v^{k+1}\right\rangle + \frac{\alpha_L}{2}\|e_v^{k+1} - e_v^k\|^2 \\ &\leq \frac{1}{2}\left\langle Le_v^k, e_v^k\right\rangle + \delta\|e_v^{k+1}\|^2 + \frac{1}{4\delta}\|B^{\top}\|^2\frac{\|B\|^2}{\alpha_A^2}\|e_v^{k+1} - e_v^k\|^2.\end{aligned} \tag{59}$$

Collecting terms, we then have

$$\begin{aligned}&(\alpha_D - \delta)\|e_v^{k+1}\|^2 + \alpha_A\|e_u^{k+1}\|^2 + \frac{1}{2}\left\langle Le_v^{k+1}, e_v^{k+1}\right\rangle \\ &+ \left(\frac{\alpha_L}{2} - \frac{1}{4\delta}\frac{\|B\|^4}{\alpha_A^2}\right)\|e_v^{k+1} - e_v^k\|^2 \leq \frac{1}{2}\left\langle Le_v^k, e_v^k\right\rangle.\end{aligned} \tag{60}$$

Choosing $\delta = \frac{\alpha_D}{2}$, and using the assumption eq. (50) we sum the equation eq. (60) from $k = 0$ to $k = n$ to obtain

$$\frac{\alpha_D}{2}\sum_{k=0}^{n}\|e_v^{k+1}\|^2 + \alpha_A\sum_{k=0}^{n}\|e_u^{k+1}\|^2 + \frac{1}{2}\left\langle Le_v^{n+1}, e_v^{n+1}\right\rangle \leq \frac{1}{2}\left\langle Le_v^0, e_v^0\right\rangle, \tag{61}$$

and it follows that as $n \to \infty$ the norms of the errors converge to zero; $\|e_v^{n+1}\|^2 \to 0$, $\|e_u^{n+1}\|^2 \to 0$. □

Remark 2. Although the proof above does not provide a convergence rate in the inherent norms, we can obtain one in the weighted L-norm: $\|x\|_L^2 = \langle Lx, x\rangle$. To see this, we first realise that $\|x\|_L$ is a norm due to the coercivity property of L. Furthermore, we have the bound

$$\|x\|_L^2 \leq \|L\|\|x\|^2, \tag{62}$$

and it follows from equation eq. (60) that

$$\frac{\alpha_D}{2\|L\|}\|e_v^{k+1}\|_L^2 + \alpha_A\|e_u^{k+1}\|^2 + \frac{1}{2}\|e_v^{k+1}\|_L^2 \leq \frac{1}{2}\|e_v^k\|_L^2.$$

Thereby, we have the contraction

$$\left(\frac{\alpha_D}{\|L\|}+1\right)\|e_v^{k+1}\|_L^2 \leq \|e_v^k\|_L^2. \tag{63}$$

5.2.2. Block-symmetric case $C = B$

We study here the case of the dual-porosity and other models with a similar structure, namely with block-symmetric square operators with negative coupling terms. We first study the case of a generic coercive stabilisation and then consider the case arising from the Schur factorisation, which leads to a non-coercive acceleration term L.

Theorem 4. *Under the assumptions of $C = B$, $A + B$ and $D + B$ coercive, $-B$ coercive and bounded, and L symmetric and coercive the solution strategy eqs.* (48) *and* (49)*, i.e.,*

$$Au^{k+1} + Bv^{k+1} = f_1 \tag{64}$$

$$Bu^k + Dv^{k+1} + L\left(v^{k+1} - v^k\right) = f_2, \tag{65}$$

converges.

Proof. Subtracting the exact equation eq. (5) from eq. (65)-eq. (64) we obtain the error equations

$$Ae_u^{k+1} + Be_v^{k+1} = 0 \tag{66}$$

$$Be_u^k + De_v^{k+1} + L\left(e_v^{k+1} - e_v^k\right) = 0. \tag{67}$$

Take the inner product of equation eq. (66) with e_u^{k+1} and eq. (67) with e_v^{k+1}, add the resulting equations and add and subtract $\left\langle Be_v^{k+1}, e_v^{k+1}\right\rangle + \left\langle B^\top e_u^{k+1}, e_v^{k+1} - e_u^{k+1}\right\rangle$

$$\begin{aligned}&-\left\langle B\left(e_v^{k+1} - e_u^{k+1}\right), e_v^{k+1} - e_u^{k+1}\right\rangle - \left\langle B\left(e_u^{k+1} - e_u^k\right), e_v^{k+1}\right\rangle \\ &+\left\langle (A+B)e_u^{k+1}, e_u^{k+1}\right\rangle + \left\langle (D+B)e_v^{k+1}, e_v^{k+1}\right\rangle + \left\langle L\left(e_v^{k+1} - e_v^k\right), e_v^{k+1}\right\rangle \\ &= 0.\end{aligned} \tag{68}$$

Using Lemma 1, the coercivity of $A + B$ and $B + D$, the coercivity and boundedness of $-B$ and the Cauchy-Schwarz inequality we get

$$\begin{aligned}&\alpha_B\left\|e_v^{k+1} - e_u^{k+1}\right\|^2 + \alpha_{A+B}\left\|e_u^{k+1}\right\|^2 + \alpha_{D+B}\left\|e_v^{k+1}\right\|^2 \\ &+\frac{1}{2}\left\langle L\left(e_v^{k+1} - e_v^k\right), e_v^{k+1} - e_v^k\right\rangle + \frac{1}{2}\left\langle Le_v^{k+1}, e_v^{k+1}\right\rangle \\ &\leq \frac{1}{2}\left\langle Le_v^k, e_v^k\right\rangle + \|B\|\left\|e_u^{k+1} - e_u^k\right\|\left\|e_v^{k+1}\right\|.\end{aligned} \tag{69}$$

Subtracting equation eq. (66) at iteration k from the same equation at iteration $k+1$ together with the boundedness of B and coercivity of A gives the inequality

$$\left\|e_u^{k+1} - e_u^k\right\| \leq \frac{\|B\|}{\alpha_A}\left\|e_v^{k+1} - e_v^k\right\|,$$

which together with Young's inequality with constant δ and the coercivity of L can be applied to eq. (69) to obtain

$$\begin{aligned}&\alpha_B\left\|e_v^{k+1} - e_u^{k+1}\right\|^2 + \left(\alpha_{D+B} - \frac{\|B\|^4}{2\alpha_A^2\delta}\right)\left\|e_v^{k+1}\right\|^2 \\ &+\alpha_{A+B}\left\|e_u^{k+1}\right\|^2 + \frac{\alpha_L}{2}\left\|e_v^{k+1} - e_v^k\right\|^2 + \frac{1}{2}\left\langle Le_v^{k+1}, e_v^{k+1}\right\rangle \\ &\leq \frac{1}{2}\left\langle Le_v^k, e_v^k\right\rangle + \frac{\delta}{2}\left\|e_v^{k+1} - e_v^k\right\|^2.\end{aligned}$$

By choosing $\delta = \frac{\|B\|^4}{\alpha_{D+B}\alpha_A^2}$, and $\alpha_L \geq \delta$ the inequality reduces to

$$\begin{aligned}&\alpha_B\left\|e_v^{k+1} - e_u^{k+1}\right\|^2 + \alpha_{A+B}\left\|e_u^{k+1}\right\|^2 + \frac{\alpha_{D+B}}{2}\left\|e_v^{k+1}\right\|^2 + \frac{1}{2}\left\langle Le_v^{k+1}, e_v^{k+1}\right\rangle \\ &\leq \frac{1}{2}\left\langle Le_v^k, e_v^k\right\rangle,\end{aligned}$$

and by the arguments from Remark 2, the solution strategy converges. □

Remark 3. It is important to notice that, although the conditions on $A + B$ and $D + B$ seem particularly strong, it must be noticed that, in the dual-porosity model $A[\cdot] = -\nabla \cdot (m_u \nabla[\cdot]) + \beta[\cdot]$ and $B[\cdot] = -\beta[\cdot]$, therefore $A + B$ is simply the Laplacian operator, and similarly for $D + B$. For more general problems, these conditions are necessary for the coercivity of the operator $\mathcal{A}$.

Theorem 5. *Under the assumptions of $C = B$, $A + B$ and $D + B$ coercive, $-B$ coercive and bounded, and*

$$\alpha_{D+B} + \alpha_{-B} \geq 2\|L\| + \frac{\|B\|^2}{4(\alpha_{A+B} + \alpha_{-B})} + \frac{\|B\|^2}{\alpha_A},$$

the solution strategy eqs. (48) *and* (49)*, i.e.,*

$$Au^{k+1} + Bv^{k+1} = f_1,$$

$$Bu^k + Dv^{k+1} + L\left(v^{k+1} - v^k\right) = f_2,$$

converges. The convergence rate is given by

$$\frac{\|L\| + \frac{\|B\|^2}{4(\alpha_{A+B}+\alpha_{-B})} + \frac{\|B\|^2}{\alpha_A}}{\alpha_{D+B} + \alpha_{-B} - \|L\|}. \tag{70}$$

Proof. The proof follows the same lines as the proof of Theorem 4 until equation (68). From there, we reorganise and apply coercivity and boundedness properties to obtain

$$\begin{aligned}&\alpha_{-B}\left\|e_v^{k+1}\right\|^2 + \alpha_{-B}\left\|e_u^{k+1}\right\|^2 + \alpha_{A+B}\left\|e_u^{k+1}\right\|^2 + \alpha_{D+B}\left\|e_v^{k+1}\right\|^2 \\ &= -\left\langle L\left(e_v^{k+1} - e_v^k\right), e_v^{k+1}\right\rangle - \left\langle Be_v^{k+1}, e_u^{k+1}\right\rangle - \left\langle Be_u^k, e_v^{k+1}\right\rangle.\end{aligned} \tag{71}$$

By the Cauchy-Schwarz inequality, we obtain

$$\begin{aligned}&(\alpha_{A+B} + \alpha_{-B})\left\|e_u^{k+1}\right\|^2 + (\alpha_{D+B} + \alpha_{-B})\left\|e_v^{k+1}\right\|^2 \\ &\leq \|L\|\|e_v^{k+1}\|^2 + \|L\|\|e_v^k\|\|e_v^{k+1}\| + \|B\|\|e_v^{k+1}\|\|e_u^{k+1}\| + \|B\|\|e_u^k\|\|e_v^{k+1}\|.\end{aligned} \tag{72}$$

From equation (66) at iteration k we obtain

$$\|e_v^k\| \leq \frac{\|B\|}{\alpha_A}\|e_v^k\|, \tag{73}$$

which inserted in (72) together with Young's inequality gives

$$\begin{aligned}&(\alpha_{A+B} + \alpha_{-B})\left\|e_u^{k+1}\right\|^2 + (\alpha_{D+B} + \alpha_{-B})\left\|e_v^{k+1}\right\|^2 \\ &\leq \|L\|\|e_v^{k+1}\|^2 + \left(\|L\| + \frac{\|B\|^2}{\alpha_A}\right)\|e_v^k\|\|e_v^{k+1}\| + (\alpha_{A+B} + \alpha_{-B})\|e_u^k\|^2 \\ &+ \frac{\|B\|^2}{4(\alpha_{A+B} + \alpha_{-B})}\|e_v^{k+1}\|^2.\end{aligned} \tag{74}$$

Dividing by $\|e_v^{k+1}\|$ and collecting terms yield

$$\left(\alpha_{D+B} + \alpha_{-B} - \frac{\|B\|^2}{4(\alpha_{A+B} + \alpha_{-B})} - \|L\|\right)\left\|e_v^{k+1}\right\| \leq \left(\|L\| + \frac{\|B\|^2}{\alpha_A}\right)\|e_v^k\|.$$

Hence, we have a contraction provided

$$\alpha_{D+B} + \alpha_{-B} \geq 2\|L\| + \frac{\|B\|^2}{4(\alpha_{A+B} + \alpha_{-B})} + \frac{\|B\|^2}{\alpha_A}. \quad \square$$

Remark 4. We point out that when L is non-coercive, it no longer plays the role of stabilisation, and therefore, it is not strictly necessary, i.e. the convergence is also ensured for $L = 0$. As we will observe in section 6, a properly constructed non-coercive L in the dual-porosity problem will lead to acceleration. In the special case that $L = 0$ and $\alpha_{-B} = \|B\|$ we get the condition

$$\alpha_{D+B} + \alpha_{-B} \geq \frac{\alpha_{-B}^2}{4(\alpha_{A+B} + \alpha_{-B})} + \frac{\alpha_{-B}^2}{\alpha_A}.$$

6. Numerical tests

This section shows illustrative numerical examples of the one- and two-dimensional Dual-Porosity and Quad-Laplacian models. We will test and compare the following schemes, which are labelled as follows: (unrelaxed) Block-Jacobi iterations (BJ), (unrelaxed) Block-Gauss-Seidel (BGS), Shur-based Partial-Jacobi on u ($SPJu$), Shur-based Partial-Jacobi on v ($SPJv$), Shur-based Partial-Jacobi on all equations (alternate, $SPJa$) and Shur-based Double-Partial-Jacobi on u, v and both equations (respectively $S2PJ_u$, $S2PJ_v$ and $S2PJ_a$).

One-dimensional finite volumes solver To test the algorithms in a controlled and simple setup, we implemented a one-dimensional finite volume solver in Python for which we can control each step of the discretisation. The finite-volume formulation has a second-order accuracy for flux reconstruction and boundary conditions. Ghost nodes are used to implement boundary conditions. We adopt the method of manufactured solutions for two one-dimensional coupled problems described in the following sections. Both problems are parameterised by the value β to change the properties of the resulting monolithic system by tuning the coupling terms. The sparse linear systems derived by internal blocks are solved with the `sparse.spsolve` method available in the `scipy` library.

OpenFOAM® solver Together with one-dimensional tests, we provide two-dimensional tests in the CFD-oriented Finite-Volumes platform OpenFOAM®. The same models adopted for one-dimensional tests are tested in the unit square domain with Dirichlet and Neumann boundary conditions.

Both solvers and the corresponding algorithms are available open-source [26].

6.1. Dual-porosity model

Here, we present experiments for the Dual porosity model for both one and two dimensions.

6.1.1. One-dimensional example

Equations eq. (1) are tested in one dimension over the interval $[0,\pi]$ with Dirichlet boundary conditions in a grid of 128 cells. The model in one-dimensional form with forcing terms reads

$$\beta(u-v)-\frac{d}{dx}\left(m_u\frac{du}{dx}\right)=f_1, \qquad x\in(0,\pi),$$
$$\beta(v-u)-\frac{d}{dx}\left(m_v\frac{dv}{dx}\right)=f_2, \qquad x\in(0,\pi).$$

Method of manufactured solutions requires to set the solutions u and v. In addition, we also choose non-trivial coefficients m_u and m_v. β represents the so-called transfer coefficient, but here, it is mainly used to study the efficiency of the proposed algorithms for increasingly coupled and ill-conditioned cases. β, in fact, controls the coupling between the two equations and the dominance of diagonals in the discretised system. For the mentioned functions, we choose

$$\begin{aligned}
u(x) &= \sin(2x),\\
v(x) &= e^{-2x},\\
m_u(x) &= 10^4\left[1+\frac{\sin(2x)}{2}\right],\\
m_v(x) &= 1+\frac{\sin(4x)}{2}.
\end{aligned}$$

The values for Dirichlet boundary conditions are derived by evaluating u and v in $x=0$ and $x=\pi$. The forcing terms f_1 and f_2 are derived by substituting u, v, m_u and m_v in the model. In Fig. 1, we report the number of iterations needed for the various algorithms to converge up to an algebraic tolerance of 10^{-6} or a maximum number of 100 iterations. As seen, the $SPJ_{(\cdot)}$ class of algorithms behave as well as BGS in the worst-case scenario and much better for SPJ_a and SPJ_u. Choosing the equation to apply the relaxation to may be important but not easy. The alternate algorithm has the advantage of being agnostic and applying it to both equations while not compromising its effectiveness. We remark that $SPJ_{(\cdot)}$ and $S2PJ_{(\cdot)}$ coincide because the off diagonal operators are diagonal, hence $\mathbb{B}\equiv B$ or $\mathbb{C}\equiv C$. This case is limited to a very simple mesh to keep it as simple as possible. However, some tests regarding the mesh size are presented for the two-dimensional case.

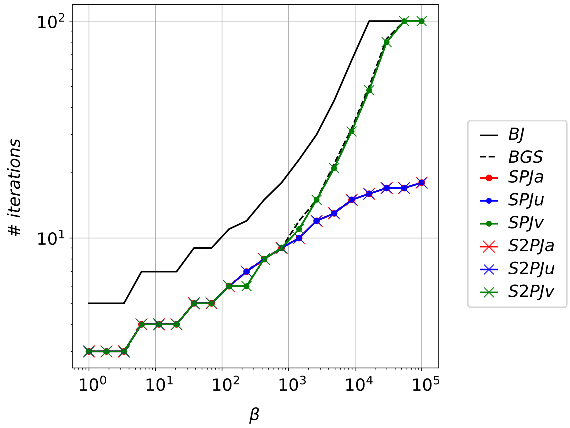


Fig. 1. One-dimensional Dual-porosity problem. Number of iterations required to reach an algebraic residual of 10^{-6} for several algorithms as a function of the parameter β. The plateau at the top of the plot indicates that the maximum number of iterations is reached (100).

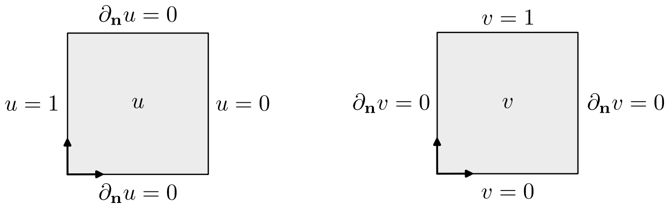

$\partial_{\mathbf{n}}u=0$
$u=1$
u
$u=0$
$\partial_{\mathbf{n}}u=0$
$v=1$
$\partial_{\mathbf{n}}v=0$
v
$\partial_{\mathbf{n}}v=0$
$v=0$

Fig. 2. Two-dimensional Dual-porosity problem. Schematic representations of boundary conditions for u and v in the unit square domain.

6.1.2. Two-dimensional example

In this two-dimensional example, we consider two continuous, highly heterogeneous, strongly anti-correlated fields m_u and β. This represents that fracture flow is dominant where the matrix is negligible and vice versa. Fig. 2 shows the boundary conditions for u and v in the square domain. Fig. 3 shows the plots of the function β, m_u and the solutions u and v, the value of m_v is set equal to one and the source terms f are zero. The mesh is uniform and Cartesian.

Table 1 shows the number of iterations for each algorithm implemented in OpenFOAM® for three different meshes. In this simulation, we set 10^{-6} as the tolerance for residuals of u and v. The results show that SPJ acts as an acceleration term and reduces the number of iterations, with the alternate version showing the best performances. However, this acceleration deteriorates for larger condition numbers of the problem.

6.2. Quad-Laplacian model

This section follows the same testing procedure of the Dual-porosity case for the Quad-Laplacian problem of eq. (3). This model has non-trivial off-diagonal blocks, which are discretisations of Laplace operators. This highlights the effect of the approximations of operators in the Schur complement. To our knowledge, there are no direct

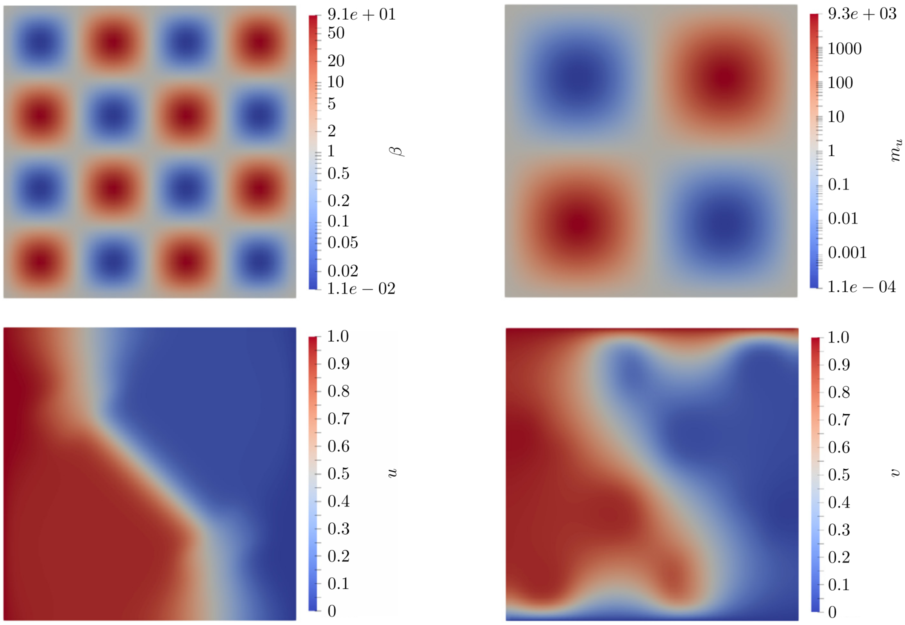


Fig. 3. Two-dimensional Dual-porosity problem. Plots of the function β, m_u and the solutions u and v. We remark that β and m_u are plotted in logarithmic scale.

Table 1
Two-dimensional Dual-porosity problem. Number of iterations to reach convergence condition for several algorithms implemented in OpenFOAM®. Columns indicate the mesh resolutions in x and y directions, respectively, and rows indicate the adopted algorithm.

method	50×50	100×100	200×200
BJ	90	104	105
BGS	59	60	57
SPJ_u	43	54	55
SPJ_v	31	47	53
SPJ_a	28	40	50

physical interpretations of this model. However, we can imagine that in a multi-physics context, equations can get easily coupled through diffusive interactions. Therefore, we adopt the Quad-Laplacian as a suitable toy problem for numerical tests. We present one- and two-dimensional tests in the same fashion as the tests presented for the Dual-porosity model. We consider here a Quad-Laplacian problem in one- and two-dimensions with a skew-block-symmetric structure, namely $m_{uv} = -m_{vu}$, in connection with Theorem 3.

6.2.1. One-dimensional example

Equations eq. (3) are tested in one dimension over the interval $[0, 2\pi]$ with Dirichlet boundary conditions in a grid of 128 cells. For the manufactured solutions and model coefficients, we choose

$$u(x) = e^{\sin x},$$

$$v(x) = -x^2 + x - 1,$$

$$m_{uu} = 1 + \frac{\sin(4x)}{2},$$

$$m_{vv} = \frac{1}{\beta}\left(10^{-2} + \frac{10^{-2}\sin(2x)}{2}\right),$$

$$m_{uv} = -m_{vu} = \beta.$$

The values for Dirichlet boundary conditions are derived by evaluating u and v in $x = 0$ and $x = \pi$. The forcing terms f_1 and f_2 are derived by substituting u, v, m_{uu}, m_{vv}, m_{uv} and m_{vu} in the model. The parameter β appears in the diffusivity coefficients m_{vv}, m_{uv} and m_{vu}, and it plays the same role as in the Dual-porosity problem. In Fig. 4, we report the number of iterations needed for the various algorithms to converge up to an algebraic tolerance of 10^{-6} or a maximum number of 100 iterations. Unlike the Dual-Porosity case, the $S2PJ_{(\cdot)}$ class of schemes behaves much better than $SPJ_{(\cdot)}$.

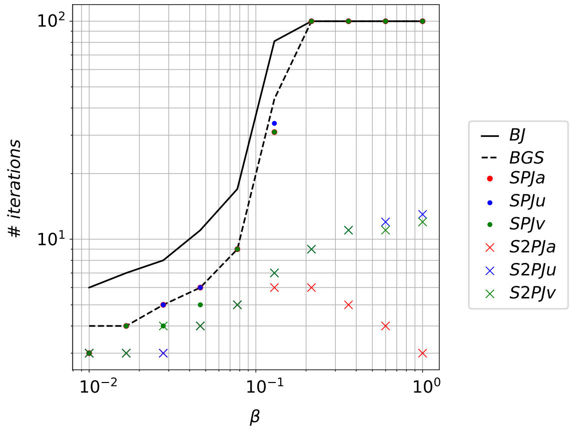


Fig. 4. One-dimensional Quad-Laplacian problem. Number of iterations required to reach an algebraic residual of 10^{-6} for several algorithms as a function of the parameter β. The plateau at the top of the plot indicates that the maximum number of iterations is reached (100).

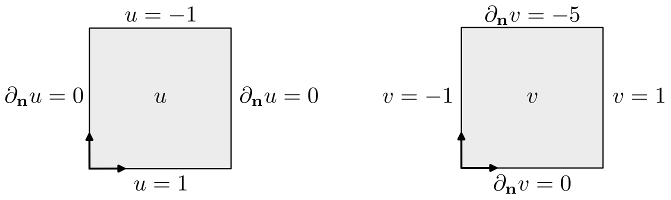

$u = -1$
$\partial_{\mathbf{n}} u = 0$
u
$\partial_{\mathbf{n}} u = 0$
$u = 1$
$\partial_{\mathbf{n}} v = -5$
$v = -1$
v
$v = 1$
$\partial_{\mathbf{n}} v = 0$

Fig. 5. Two-dimensional Quad-Laplacian problem. Schematic representations of boundary conditions for u and v in the unit square domain.

6.2.2. Two-dimensional example

For the two-dimensional case, we consider the domain in Fig. 5, and heterogeneous parameters m_{uu} and m_{vv} as shown in Fig. 6, while $m_{uv} = -m_{vu} = 1$. The mesh is uniform and Cartesian OpenFOAM®.

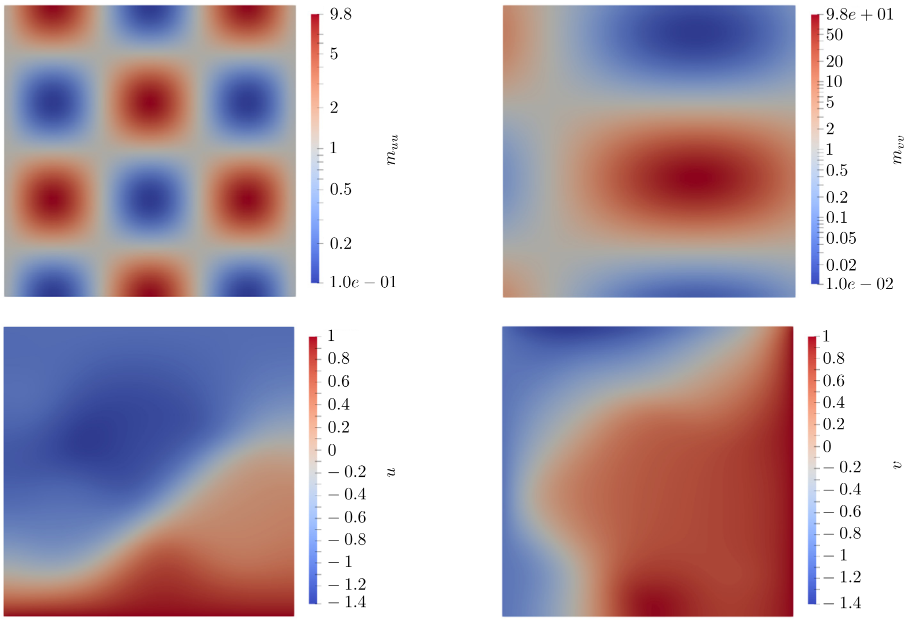


Fig. 6. Two-dimensional Quad-Laplacian problem. Plots of the function m_{uu}, m_{vv} and the solutions u and v. We remark that m_{uu} and m_{vv} are plotted in logarithmic scale.

Table 2
Two-dimensional Quad-Laplacian problem. Number of iterations to reach the desired residue's tolerance of 10^{-6} for several algorithms implemented in OpenFOAM®. Columns indicate the mesh resolutions in x and y directions, respectively, and rows indicate the adopted algorithm. The symbol "×" indicates that the algorithm failed to converge.

method	50×50	100×100	200×200
BJ	×	×	×
BGS	×	×	×
$S2PJ_u$	37	37	38
$S2PJ_v$	26	26	26
$S2PJ_a$	15	13	13

Results for different meshes are shown in Fig. 6 and in Table 2. The method is robust with respect to the mesh size and the conditioning number of the problem. Here, the relaxation operator acts as a stabilisation term, ensuring the convergence of the splitting scheme. Since the high-level interfaces provided by OpenFOAM® do not allow to compute matrix products of sparse matrices required in the SPJ, we can use here the $S2PJ$ variant, thanks to the diagonal approximations of two operators appearing in the double product; only one of the three matrices is sparse, and the other two are simple fields (interpreted as diagonal matrices). However, we also recall that this double approximation makes sense only if the off-diagonal discrete operators are square matrices.

The S2PJ performs consistently well due to the structure of the problem. Being all operators Laplacian, the two diagonal approximations tend to cancel each other, leading to an accurate approximation of the Schur complement. Therefore, while BJ and BGS diverge, $S2PJ$ converges in a few iterations.

7. Conclusions

This work focuses on the development of a unified treatment and theory for iterative splitting schemes for coupled systems of differential equations. We have shown how stationary iterative methods for linear systems can be applied to systems of Partial Differential Equations (PDEs) and demonstrated the need to introduce a relaxation operator to stabilise the iteration and ensure convergence. We have extended the idea of relaxation to include generic operators and shown how these can be built based on approximated Schur complements. Convergence theorems are proposed and define sufficient conditions on the relaxation operators. Numerical tests in one and two dimensions for the Dual-Porosity and Quad-Laplacian problems have been performed using two different open-source codes, one based on the Python library `scipy` and the other on the OpenFOAM®) libraries. The numerical results confirm the theory and show the potential of the Schur-based relaxation operator, either as a stabilisation term to ensure convergence or as an acceleration to reduce the number of iterations, depending on the problem. The approaches and methods presented here can be extended to include the generalisation to $N \times N$ systems of equations and non-linear problems. In future works, we aim to include the extension of the convergence theory to better exploit the specific structure of Schur-based relaxation operators.

Data availability

All data is available on GitHub, Zenodo and CodeOcean.

Link to the Reproducible Capsule

https://codeocean.com/capsule/6571121/tree

Acknowledgements

This work has been funded by the following grants:

- MI has been supported by the European Union's Horizon 2020 research and innovation programme, grant agreement number 764531, "SECURe – Subsurface Evaluation of Carbon capture and storage and Unconventional risks", and by the Nottingham EPSRC Impact Acceleration Award;
- RN has been supported by the Royal Academy of Engineering (TSP2021\100342), with the grants: "GHG-intensive-industry Medium-scale Capture and Utilization Solutions Assessment In Colombia" and "Asphaltene dynamics at the pore-scale and the impact on oil production at the field-scale", and by the Nottingham Interdisciplinary Fund.

References

[1] H. Bungartz, F. Lindner, B. Gatzhammer, M. Mehl, K. Scheufele, A. Shukaev, B. Uekermann, preCICE – a fully parallel library for multi-physics surface coupling, Comput. Fluids 141 (2016), https://doi.org/10.1016/j.compfluid.2016.04.003.

[2] A. Settari, F. Mourits, A coupled reservoir and geomechanical simulation system, SPE J. 3 (03) (1998) 219–226.

[3] J. Kim, H. Tchelepi, R. Juanes, Stability and convergence of sequential methods for coupled flow and geomechanics: fixed-stress and fixed-strain splits, Comput. Methods Appl. Mech. Eng. 200 (13–16) (2011) 1591–1606.

[4] A. Mikelić, M. Wheeler, Convergence of iterative coupling for coupled flow and geomechanics, Comput. Geosci. 17 (3) (2013) 455–461.

[5] J. Both, M. Borregales, J. Nordbotten, K. Kumar, F. Radu, Robust fixed stress splitting for Biot's equations in heterogeneous media, Appl. Math. Lett. 68 (2017) 101–108.

[6] J. Kim, H. Tchelepi, R. Juanes, Stability and convergence of sequential methods for coupled flow and geomechanics: drained and undrained splits, Comput. Methods Appl. Mech. Eng. 200 (23–24) (2011) 2094–2116.

[7] E. Storvik, J. Both, K. Kumar, J. Nordbotten, F. Radu, On the optimization of the fixed-stress splitting for Biot's equations, Int. J. Numer. Methods Eng. 120 (2) (2019) 179–194.

[8] D. Illiano, I. Pop, F. Radu, Iterative schemes for surfactant transport in porous media, Comput. Geosci. 25 (2) (2021) 805–822.

[9] C. Evans, S. Pollock, L. Rebholz, M. Xiao, A proof that Anderson acceleration improves the convergence rate in linearly converging fixed-point methods (but not in those converging quadratically), SIAM J. Numer. Anal. 58 (1) (2020) 788–810.

[10] J. Both, K. Kumar, J. Nordbotten, F. Radu, Anderson accelerated fixed-stress splitting schemes for consolidation of unsaturated porous media, Comput. Math. Appl. 77 (6) (2019) 1479–1502.

[11] E. Storvik, J. Both, J. Sargado, J. Nordbotten, F. Radu, An accelerated staggered scheme for variational phase-field models of brittle fracture, Comput. Methods Appl. Mech. Eng. 381 (2021) 113822.

[12] J. Douglas Jr, T. Arbogast, Dual porosity models for flow in naturally fractured reservoirs, in: Dynamics of Fluids in Hierarchical Porous Media, 1990, pp. 177–221.

[13] M. Fortin, F. Brezzi, Mixed and Hybrid Finite Element Methods, vol. 3, Springer-Verlag, New York, 1991.

[14] D. Boffi, F. Brezzi, M. Fortin, et al., Mixed Finite Element Methods and Applications, vol. 44, Springer, 2013.

[15] Q. Hong, J. Kraus, M. Lymbery, F. Philo, A new practical framework for the stability analysis of perturbed saddle-point problems and applications, Math. Comput. (2022).

[16] R. Nicolaides, Existence, uniqueness and approximation for generalized saddle point problems, SIAM J. Numer. Anal. 19 (2) (1982) 349–357.

[17] W. Han, K. Atkinson, Theoretical Numerical Analysis: A Functional Analysis Framework, Springer, 2009.

[18] W. Hackbusch, Iterative Solution of Large Sparse Systems of Equations, vol. 95, Springer, 2016.

[19] O. Axelsson, A survey of robust preconditioning methods, in: Topics in Numerical Analysis, Springer, 2001, pp. 29–48.

[20] A. Viguerie, M. Xiao, Effective Chorin–Temam algebraic splitting schemes for the steady Navier–Stokes equations, Numer. Methods Partial Differ. Equ. 35 (2) (2019) 805–829.

[21] G. Golub, C. Greif, On solving block-structured indefinite linear systems, SIAM J. Sci. Comput. 24 (6) (2003) 2076–2092.

[22] K. Karki, S. Patankar, Application of the partial elimination algorithm for solving the coupled energy equations in porous media, Numer. Heat Transf., Part A, Appl. 45 (6) (2004) 539–549.

[23] C. Filelis-Papadopoulos, G. Gravvanis, Generic approximate sparse inverse matrix techniques, Int. J. Comput. Methods 11 (06) (2014) 1350084.

[24] O. Axelsson, R. Blaheta, M. Neytcheva, Preconditioning of boundary value problems using elementwise Schur complements, SIAM J. Matrix Anal. Appl. 31 (2) (2009) 767–789.

[25] R. Kirby, From functional analysis to iterative methods, SIAM Rev. 52 (2) (2010) 269–293.

[26] R. Nuca, M. Icardi, splittingschemes v1.0, https://doi.org/10.5281/zenodo.7457786, 2022.