LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-JRNL-746123

# Block preconditioning for contact mechanics saddle point problems

A. Franceschini, N. Castelletto, M. Ferronato

February 9, 2018

Computer Methods in Applied Mechanics and Engineering

## Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 344 (2019) 376–401
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

# Block preconditioning for fault/fracture mechanics saddle-point problems

Andrea Franceschini[a,*], Nicola Castelletto[b,c], Massimiliano Ferronato[a]

[a] Department of Civil, Environmental and Architectural Engineering, University of Padova, Italy
[b] Energy Resources Engineering, Stanford University, United States
[c] Atmospheric, Earth and Energy Division, Lawrence Livermore National Laboratory, United States

Received 12 February 2018; received in revised form 14 July 2018; accepted 24 September 2018
Available online 25 October 2018

**Abstract**

The efficient simulation of fault and fracture mechanics is a key issue in several applications and is attracting a growing interest by the scientific community. Using a formulation based on Lagrange multipliers, the Jacobian matrix resulting from the Finite Element discretization of the governing equations has a non-symmetric generalized saddle-point structure. In this work, we propose a family of block preconditioners to accelerate the convergence of Krylov methods for such problems. We critically review possible advantages and difficulties of using various Schur complement approximations, based on both physical and algebraic considerations. The proposed approaches are tested in a number of real-world applications, showing their robustness and efficiency also in large-size and ill-conditioned problems.
© 2018 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

*Keywords:* Fault mechanics; Lagrange multipliers; Preconditioners; Saddle point problems; Iterative methods

## 1. Introduction

Accurate simulation of fault and fracture mechanics is a key component in a wide number of subsurface engineering applications. Faults and fractures are typically treated by modelers as discontinuity surfaces embedded in three dimensional (3D) continuous media. From a mathematical standpoint, they are described as internal boundaries whose behavior is governed by the displacement and stress fields acting on the surrounding continuum and the traction constitutive relationship defined on the surface, which leads to strongly coupled non-linear problems. Here, we focus on efficient preconditioning techniques for the linear systems arising from the discretization and linearization of the governing equations that describe the contact mechanics of faulted and fractured geological media based on a Lagrange multiplier formulation.

* Corresponding author.
*E-mail addresses:* andrea.franceschini@dicea.unipd.it (A. Franceschini), castelletto1@llnl.gov (N. Castelletto), massimiliano.ferronato@unipd.it (M. Ferronato).

https://doi.org/10.1016/j.cma.2018.09.039
0045-7825/© 2018 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Two methods are mainly used to deal with discontinuities in a continuous domain. The first method, referred to as *penalty approach* [1,2], consists of introducing very stiff springs connecting the faces of the discontinuity to enforce locally the geometrical non-penetration condition. When the stress satisfies a specified failure criterion, the springs yield irreversibly and consequently the contact surfaces become free to move. Because the elastic springs deform for any non-zero stress value, the penalty method is mathematically inexact. Furthermore, from a numerical point of view, this method may cause a severe ill-conditioning of the stiffness matrix [3,4] due to the use of penalty coefficients. Nevertheless, such an approach is widely used, mainly for the ease of implementation. Some examples of application of the penalty approach, including fields beyond geomechanics, can be found in [5–14]. Alternatively, the constraint conditions can be imposed by using *Lagrange multipliers* [1,2,15–18], namely in an analytically exact way. The Lagrange multipliers represent a new set of unknowns, thus the problem is numerically enlarged. We observe that from a physical point of view the Lagrange multipliers are contact traction vectors, i.e. strength acting on the discontinuity plane. Lagrange multiplier-based models rely on two governing equations, namely a linear momentum balance equation coupled to an equation enforcing the compatibility conditions of contact over discontinuity surfaces, which need to be solved for the displacement vector and Lagrange multipliers as primary unknowns. As opposed to penalty formulations, the discrete system changes its nature and becomes indefinite with a $2 \times 2$ block structure. Nevertheless, the increase of the computational cost is generally compensated by a more robust convergence in the non-linear steps and a more stable numerical behavior. Other methods, based on the previously discussed algorithms, have been also developed. For example, the augmented Lagrangian method has been introduced as a compromise between the two techniques [1,19]. Nitsche's method [20] is another intermediate approach between penalty and Lagrange techniques. Improvements and applications of Nitsche's method can be found in [21,22].

In this work, we use the Lagrange multiplier-based formulation developed for fault mechanics modeling in [23]. Faults and fractures are simulated as a pair of inner surfaces embedded in a 3D geological formation. The application of the fault model to large-scale problems gives rise to a set of sparse discrete systems of linearized equations with a generalized non-symmetric saddle point structure. The development of efficient algorithms for the iterative solution of this kind of system is the object of the present work. We focus on preconditioning techniques which exploit the native $2 \times 2$ block structure of the Jacobian. Although this topic is well developed in the recent literature, the efficient solution to large-size indefinite saddle-point linear systems is still an issue in several applications, including the one discussed in this work. On the one hand, it is well recognized that the most effective paradigm for building an optimal preconditioner for this class of problems relies on computing the Schur complement of the native matrix [24–26]. On the other hand, defining a good approximation of the Schur complement and its inverse, which generally cannot be computed explicitly, is strongly problem-dependent and no general rules are available for such a choice [26]. Our work is a contribution aimed to address this issue in the context of the saddle-point problems arising from the treatment of contact mechanics by Lagrange multipliers.

The quality and performance of this class of preconditioners rely on two steps: (i) the preconditioning of the leading (1,1) block and (ii) the Schur complement computation. In this paper we concentrate on the second step and propose and compare various Schur complement preconditioners. The computation can be founded on an explicit algebraic approximation of the leading block inverse or on a physics-based block diagonal block algorithm. As the inverse of the Schur complement must be applied, other possibilities come in. The approximate Schur complement can be inverted inexactly by using another preconditioner, but also a nested direct solver can be efficiently used. Another possibility relies on approximating directly the inverse of the Schur complement by a novel application of the least-square commutator originally introduced for Navier–Stokes problems [27].

The paper is organized as follows. The model problem is first derived, describing the properties of the resulting discrete equations and the associated block Jacobian matrix. Then, a general framework for the preconditioning algorithm is introduced, based on a block triangular approach. At this point, different approximations for the resulting Schur complement are investigated and compared. An extensive numerical experimentation highlights advantages and drawbacks of the different approaches in a set of examples taken from real-world applications of faulted and fractured media. Finally, a conclusive discussion closes the presentation.

## 2. Model problem

### 2.1. Governing equations

Consider a linear elastic domain $\Omega \subset \mathbb{R}^3$ delimited by boundary $\Gamma = \Gamma_{\boldsymbol{u}} \cup \Gamma_{\boldsymbol{\sigma}}$ such that $\Gamma_{\boldsymbol{u}} \cap \Gamma_{\boldsymbol{\sigma}} = \emptyset$, with $\boldsymbol{n}$ denoting its outer unit normal vector (Fig. 1a). Let assume a well-defined internal boundary $\Gamma_f$ embedded in $\Omega$ that

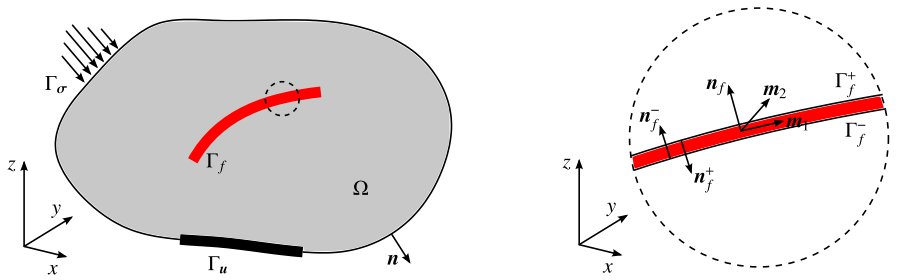



**Fig. 1.** Physical domain for fault modeling (a), and local fault reference (b).

is initially represented by two overlapping surfaces $\Gamma_f^-$ and $\Gamma_f^+$ – hence, pairs of corresponding points on each side of $\Gamma_f$ are readily identified – with unit normal vectors oriented as shown in Fig. 1b. Note that $\boldsymbol{n}_f$ coincides with $\boldsymbol{n}_f^-$. Let $\mathbb{I} = (0, t_{\max})$ denote the time domain of interest. Assuming an infinitesimal strain framework, we focus on the following quasi-static contact model problem [15,17,18].

Given $\boldsymbol{b} : \Omega \times \mathbb{I} \to \mathbb{R}^3$, $\bar{\boldsymbol{u}} : \Gamma_{\boldsymbol{u}} \times \mathbb{I} \to \mathbb{R}^3$, $\bar{\boldsymbol{t}}_{\boldsymbol{\sigma}} : \Gamma_{\boldsymbol{\sigma}} \times \mathbb{I} \to \mathbb{R}^3$, and $\boldsymbol{u}_0 : \overline{\Omega} \to \mathbb{R}^3$, find $\boldsymbol{u} : \overline{\Omega} \times \mathbb{I} \to \mathbb{R}^3$ such that

$$-\nabla \cdot \boldsymbol{\sigma} = \boldsymbol{b} \quad \text{in } \Omega \times \mathbb{I} \quad \text{(linear momentum balance)}, \tag{1a}$$

$$\boldsymbol{u} = \bar{\boldsymbol{u}} \quad \text{on } \Gamma_{\boldsymbol{u}} \times \mathbb{I} \quad \text{(prescribed boundary displacement)}, \tag{1b}$$

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \bar{\boldsymbol{t}} \quad \text{on } \Gamma_{\boldsymbol{\sigma}} \times \mathbb{I} \quad \text{(prescribed boundary traction)}, \tag{1c}$$

$$\boldsymbol{u}|_{t=0} = \boldsymbol{u}_0 \quad \text{in } \overline{\Omega} \quad \text{(initial displacement)}, \tag{1d}$$

subject to the constraints

$$t_N = \boldsymbol{t} \cdot \boldsymbol{n}_f \le 0 \quad \text{on } \Gamma_f \times \mathbb{I} \quad \text{(normal contact conditions)}, \tag{1e}$$

$$g_N = [\![\boldsymbol{u}]\!] \cdot \boldsymbol{n}_f \ge 0 \quad \text{on } \Gamma_f \times \mathbb{I}, \tag{1f}$$

$$t_N g_N = 0 \quad \text{on } \Gamma_f \times \mathbb{I}, \tag{1g}$$

$$\Phi = \|\boldsymbol{t}_T\|_2 - (c - t_N \tan\varphi) \le 0 \quad \text{on } \Gamma_f \times \mathbb{I} \quad \text{(Coulomb frictional contact conditions)}, \tag{1h}$$

$$\dot{\boldsymbol{g}}_T - \alpha \frac{\boldsymbol{t}_T}{\|\boldsymbol{t}_T\|_2} = \boldsymbol{0} \quad \text{on } \Gamma_f \times \mathbb{I}, \tag{1i}$$

$$\alpha \ge 0 \quad \text{on } \Gamma_f \times \mathbb{I}, \tag{1j}$$

$$\Phi\alpha = 0 \quad \text{on } \Gamma_f \times \mathbb{I}. \tag{1k}$$

Here, $\boldsymbol{u}$ is the displacement vector; $\boldsymbol{\sigma}$ is the Cauchy stress tensor, which is related to the displacement field in terms of the fourth-order elasticity tensor $\mathbf{C}$ such that $\boldsymbol{\sigma} = \mathbf{C} : \nabla^s \boldsymbol{u}$, with $\nabla\cdot$ and $\nabla^s$ the divergence and symmetric gradient operator, respectively, and (:) denoting a tensor contraction; $\boldsymbol{b}$ is the body force; $\boldsymbol{t} = \boldsymbol{\sigma} \cdot \boldsymbol{n}_f^- = -\boldsymbol{\sigma} \cdot \boldsymbol{n}_f^+ = (t_N \boldsymbol{n}_f + \boldsymbol{t}_T)$ is the traction vector over $\Gamma_f$, with $t_N$ and $\boldsymbol{t}_T = (t_{m_1}\boldsymbol{m}_1 + t_{m_2}\boldsymbol{m}_2)$ its normal and tangential component, respectively, relative to the local reference system shown in Fig. 1b; $c$ and $\varphi$ are the cohesion and friction angle, respectively, which define the limit value $\tau_{\max} = \tau_{\max}(t_N) = (c - t_N \tan\varphi)$ for the modulus of $\boldsymbol{t}_T$ according to Coulomb's criterion; $[\![\cdot]\!]$ denotes the jump of a quantity across $\Gamma_f$, namely $[\![\boldsymbol{u}]\!] = (\boldsymbol{u}|_{\Gamma_f^+} - \boldsymbol{u}|_{\Gamma_f^-}) = (g_N \boldsymbol{n}_f + \boldsymbol{g}_T)$ is the relative displacement across $\Gamma_f$ having $g_N$ and $\boldsymbol{g}_T$ as normal and tangential components, respectively, with $\boldsymbol{u}|_{\Gamma_f^+}$ and $\boldsymbol{u}|_{\Gamma_f^-}$ the restriction of $\boldsymbol{u}$ on $\Gamma_f^+$ and $\Gamma_f^-$; and $\alpha$ denotes a consistency parameter indicating the magnitude of the rate of the tangential relative motion according to (1i), which is based on a general rate form. For additional details pertaining the derivation of the equations governing the contact initial boundary value problem (IBVP) we refer the reader to [15,17,18].

Generally speaking, $\Gamma_f$ encompasses the whole region over which potential contact events may take place at any $t \in \mathbb{I}$ [17]. Three operating modes are considered for the discontinuity surface, which provide the following partition of $\Gamma_f = \Gamma_f^{\text{stick}} \cup \Gamma_f^{\text{slip}} \cup \Gamma_f^{\text{open}}$, namely:

- **Stick mode** on $\Gamma_f^{\text{stick}}$: the discontinuity is fully closed and compressed with the Coulomb criterion satisfied, i.e. $t_N < 0$ (Eq. (1e)) and $\Phi < 0$ (Eq. (1h)). The three components of the traction are unknown and such that no relative movement is allowed between $\Gamma_f^+$ and $\Gamma_f^-$;
- **Slip mode** on $\Gamma_f^{\text{slip}}$: the fault is compressed, but the inequality (1h) becomes an equality on the slip surface $\Phi$, i.e. the analogue of the yield surface in theories of plasticity [17]. A slip displacement $\boldsymbol{g}_T$ between $\Gamma_f^+$ and $\Gamma_f^-$ is allowed for. Only the normal traction component $t_N$ is unknown. The tangential traction is known, having magnitude $\|\boldsymbol{t}_T^*\|_2 = \tau_{\max}$ and direction collinear with $\boldsymbol{g}_T$;
- **Open mode** on $\Gamma_f^{\text{open}}$: $\Gamma_f^+$ and $\Gamma_f^-$ are not in contact and a free relative displacement $[\![\boldsymbol{u}]\!]$ is allowed. The traction is known and equal to the zero vector in $\mathbb{R}^3$.

It is important to emphasize that, in this work, $\Gamma_f$ is considered to be fixed and well defined during the entire time domain of interest, just its partitioning into stick, slip, and open patches evolves.

**Remark 1.** On $\Gamma_f^{\text{slip}}$ tangential traction and relative displacement can be expressed as:

$$\boldsymbol{t}_T^* = \boldsymbol{t}_T^*(\boldsymbol{u}, \boldsymbol{t}) = \tau_{\max} \frac{\dot{\boldsymbol{g}}_T}{\|\dot{\boldsymbol{g}}_T\|_2}, \tag{2a}$$

$$\boldsymbol{g}_T = \boldsymbol{g}_T(\boldsymbol{u}) = (\mathbf{1} - \boldsymbol{n}_f \otimes \boldsymbol{n}_f) \cdot [\![\boldsymbol{u}]\!], \tag{2b}$$

with $\mathbf{1}$ the second-order identity tensor, and $\otimes$ denoting the dyadic product. Since a dilation-free model is considered, note that $g_N$ is always zero on $\Gamma_f^{\text{slip}}$, hence $\boldsymbol{g}_T = [\![\boldsymbol{u}]\!] - g_N \boldsymbol{n}_f = [\![\boldsymbol{u}]\!]$.

**Remark 2.** Relationships (1e)–(1g) are the Karush–Kuhn– Tucker (KKT) complementary conditions [16] for normal contact. They state that: (i) the normal traction $t_N \boldsymbol{n}_F$ must be compressive (Eq. (1e)); (ii) no penetration is allowed for between the two sides of the discontinuity surface $\Gamma_f$ (Eq. (1f)); and (iii) the compressive normal traction is nonzero only if the discontinuity is in contact mode (Eq. (1g)). Conversely, Eqs. (1h)–(1k) are the KKT conditions for frictional contact and describe if $\Gamma_f$ is in stick or slip mode, namely: (i) an upper bound for the magnitude of the tangential traction vector is established based on the classic Mohr–Coulomb criterion (Eqs. (1h)); (ii) the tangential relative displacement $\boldsymbol{g}_T$ (slip displacement) and $\boldsymbol{t}_T$ are collinear vectors (Eqs. (1i)–(1j)); and (iii) a slip displacement is allowed for only if $\left\|\boldsymbol{t}_T^*\right\| = \tau_{\max}$ (Eqs. (1k)).

### 2.2. Variational form

To derive an appropriate variational principle for problems of constrained evolution, such the contact IBVP (1k), two approaches may be considered [15,17,18]: (i) the variational inequality, and (ii) the variational equality. The first approach requires the definition of a space $\mathcal{K}$ of constrained trial functions, i.e. its members satisfy both the prescribed displacement boundary condition on $\Gamma_{\boldsymbol{u}}$ and the impenetrability condition (1f) at any $t \in \mathbb{I}$, that is also used to express the trial functions. Then, a variational inequality involving the virtual work due to internal stresses and applied loadings along with a frictional functional associated with $\Gamma_f$ must be enforced over $\mathcal{K}$. The second approach circumvents the need for the variational inequality by introducing either penalty regularizations or Lagrange multipliers, so that the problem reduces to a variational equality that is amenable for traditional finite element methods. For an exhaustive review of available methods and a comprehensive mathematical analysis the reader is referred to [15]. Here, we follow the second approach based on a Lagrange multiplier formulation.

Let us first define the spaces of (unconstrained) trial ($\mathcal{U}$) and test ($\mathcal{V}$) functions for the displacement

$$\mathcal{U} := \left\{ \boldsymbol{u} \,\middle|\, \boldsymbol{u} \in [H^1(\Omega)]^3, \boldsymbol{u} = \bar{\boldsymbol{u}} \text{ on } \Gamma_{\boldsymbol{u}} \right\}, \tag{3}$$

$$\mathcal{V} := \left\{ \boldsymbol{v} \,\middle|\, \boldsymbol{v} \in [H^1(\Omega)]^3, \boldsymbol{v} = \mathbf{0} \text{ on } \Gamma_{\boldsymbol{u}} \right\}, \tag{4}$$

with $H^1(\Omega)$ the Sobolev space of square integrable functions with square integrable gradients. Note that $\mathcal{U}$ generally depends on time due to the possible temporal variation of $\bar{\boldsymbol{u}}$ over $\Gamma_{\boldsymbol{u}}$. Conversely, $\mathcal{V}$ is constant in time. The traction vector $\boldsymbol{t}$ acting on $\Gamma_f$ is introduced as an additional primary variable serving as Lagrange multiplier to enforce the

constraint conditions (1e)–(1k). We define a vector value Lagrange multiplier space $\boldsymbol{\mathcal{M}}$ that is the dual space of the trace space $\boldsymbol{\mathcal{W}}$ of $\boldsymbol{\mathcal{V}}$ restricted to $\Gamma_f$. The following subspace of $\boldsymbol{\mathcal{M}}$ is then defined [15,28]:

$$\boldsymbol{\mathcal{M}}(t) := \left\{ \boldsymbol{\mu} \,|\, \boldsymbol{\mu} \in \boldsymbol{\mathcal{M}},\, \mu_N \le 0,\ \int_{\Gamma_f} \boldsymbol{\mu}\cdot\boldsymbol{w}\,\mathrm{d}\Gamma \le \int_{\Gamma_f} \tau_{\max}(t_N)\,\|\boldsymbol{w}_T\|_2\,\mathrm{d}\Gamma,\, \boldsymbol{w}\in\boldsymbol{\mathcal{W}} \text{ with } w_N \ge 0 \right\} \tag{5}$$

where $\mu_N = \boldsymbol{\mu}\cdot\boldsymbol{n}_f$, $w_N = \boldsymbol{w}\cdot\boldsymbol{n}_f$, and $\boldsymbol{w}_T = (\mathbf{1}-\boldsymbol{n}_f\otimes\boldsymbol{n}_f)\cdot\boldsymbol{w}$. Then, the weak mixed formulation for the contact IBVP (1k) reads as follows.

Find $\{\boldsymbol{u},\boldsymbol{t}\}\in\boldsymbol{\mathcal{U}}\times\boldsymbol{\mathcal{M}}(t)$ such that for all $t\in\mathbb{I}$:

$$\mathcal{R}_{\boldsymbol{u}} = \int_\Omega \nabla^s\boldsymbol{v} : \boldsymbol{\sigma}\,\mathrm{d}\Omega - \int_\Omega \boldsymbol{v}\cdot\boldsymbol{b}\,\mathrm{d}\Omega - \int_{\Gamma_{\boldsymbol{\sigma}}} \boldsymbol{v}\cdot\bar{\boldsymbol{t}}\,\mathrm{d}\Gamma + \int_{\Gamma_f} [\![\boldsymbol{v}]\!]\cdot\boldsymbol{t}\,\mathrm{d}\Gamma = 0, \qquad \forall\boldsymbol{v}\in\boldsymbol{\mathcal{V}}, \tag{6a}$$

$$\mathcal{R}_{\boldsymbol{t}} = \int_{\Gamma_f} (t_N-\mu_N)g_N\,\mathrm{d}\Gamma + \int_{\Gamma_f} (\boldsymbol{t}_T-\boldsymbol{\mu}_T)\cdot\dot{\boldsymbol{g}}_T\,\mathrm{d}\Gamma \ge 0, \qquad \forall\boldsymbol{\mu}\in\boldsymbol{\mathcal{M}}(t). \tag{6b}$$

In (6a), the first three integrals represent the virtual work due to internal stresses and applied loadings, whereas the last integral denotes the virtual work due to the contact forces, i.e.

$$\int_{\Gamma_f} [\![\boldsymbol{v}]\!]\cdot\boldsymbol{t}\,\mathrm{d}\Gamma = \int_{\Gamma_f^+} \boldsymbol{v}\cdot\boldsymbol{t}\,\mathrm{d}\Gamma - \int_{\Gamma_f^-} \boldsymbol{v}\cdot\boldsymbol{t}\,\mathrm{d}\Gamma = -\int_{\Gamma_f^+} \boldsymbol{v}\cdot\boldsymbol{\sigma}\cdot\boldsymbol{n}_f^+\,\mathrm{d}\Gamma - \int_{\Gamma_f^-} \boldsymbol{v}\cdot\boldsymbol{\sigma}\cdot\boldsymbol{n}_f^-\,\mathrm{d}\Gamma. \tag{7}$$

Also, note that the same decomposition into normal and tangential components is used for both $\boldsymbol{t}$ and $\boldsymbol{\mu}$ in (6b).

**Remark 3.** It may be noted an apparent dimensional inconsistency in Eq. (6b), the argument of the first and second integral having units of traction times displacement and traction times velocity, respectively. Actually, the rate of the tangential displacement should be regarded as a pseudo-time dependent quantity. To avoid any confusion, $\dot{\boldsymbol{g}}_T$ will be replaced by $\Delta\boldsymbol{g}_T$ from now on. As in [17], the notation $\Delta(\bullet)$ is used to denote the algorithmic approximation to the change of a quantity $(\bullet)$ during a time step $\delta t$, namely $\Delta(\bullet) = (\bullet)_{t+\delta t} - (\bullet)_t \approx \int_t^{t+\delta t}(\bullet)\,\mathrm{d}t$. In other words, the collinearity condition (1i) will be enforced as $\Delta\boldsymbol{g}_T = \Delta\alpha\,\frac{\boldsymbol{t}_T}{\|\boldsymbol{t}\|_2}$, hence (2a) becomes $\boldsymbol{t}^* = \tau_{\max}\frac{\Delta\boldsymbol{g}_T}{\|\Delta\boldsymbol{g}_T\|_2}$, resolving then the dimensional inconsistency.

### 2.3. Discrete form

A fundamental challenge of the Lagrange multiplier formulation is associated with the identification of which portions of $\Gamma_f$ are in contact, slip and open mode, respectively. Knowing stick and slip regions of $\Gamma_f$ at a given $t\in\mathbb{I}$ allows for writing the weak form in terms of equality relationships [18]. Indeed, only portions of $\Gamma_f$ where the contact is active, precisely $\Gamma_f^{\text{stick}}$ and $\Gamma_f^{\text{slip}}$, contribute to the virtual work due to contact forces. For any point belonging to $\Gamma_f^{\text{stick}}$, the three components of $\boldsymbol{t}$ are unknown since $t_N<0$ and $\|\boldsymbol{t}_T\|_2 < \tau_{\max}(t_N)$, hence three kinematic equations are needed to enforce the constraints $g_N=0$ and $\dot{\boldsymbol{g}}_T=\mathbf{0}$. Conversely, for a point over $\Gamma_f^{\text{slip}}$ only $t_N$ is unknown, which requires a single kinematic equation to ensure $g_N=0$, with $\boldsymbol{t}_T=\boldsymbol{t}_T^*$. Unfortunately, the set of active constraints is not in general known a priori. To enable an incremental solution step procedure for the nonlinear contact problem by a Newton loop, iterative techniques that identify regions $\Gamma_f^{\text{stick}}$ and $\Gamma_f^{\text{slip}}$ are needed—e.g., an active set strategy (see [15,17,18] and references therein to the optimization literature). Since the central focus of this work is on preconditioning for the linear solve required at each Newton's step, the details of the developed contact formulation are not discussed. The interested reader is referred to [23,29].

For the remainder of this section we will assume that at a given time $t\in\mathbb{I}$ the active contact regions $\Gamma_f^{\text{stick}}$ and $\Gamma_f^{\text{slip}}$ are known. Hence, the integral over $\Gamma_f$ in Eq. (6a) can be rewritten as follows

$$\begin{aligned}\int_{\Gamma_f} [\![\boldsymbol{v}]\!]\cdot\boldsymbol{t}\,\mathrm{d}\Gamma = &\int_{\Gamma_f^{\text{stick}}\cup\Gamma_f^{\text{slip}}} [\![\boldsymbol{v}]\!]\cdot\underbrace{(\boldsymbol{n}_f\otimes\boldsymbol{n}_f)\cdot\boldsymbol{t}}_{=t_N\boldsymbol{n}_f}\,\mathrm{d}\Gamma + \int_{\Gamma_f^{\text{stick}}} [\![\boldsymbol{v}]\!]\cdot\underbrace{(\mathbf{1}-\boldsymbol{n}_f\otimes\boldsymbol{n}_f)\cdot\boldsymbol{t}}_{=\boldsymbol{t}_T}\,\mathrm{d}\Gamma \\ &+ \int_{\Gamma_f^{\text{slip}}} [\![\boldsymbol{v}]\!]\cdot\underbrace{(\mathbf{1}-\boldsymbol{n}_f\otimes\boldsymbol{n}_f)\cdot\boldsymbol{t}}_{=\boldsymbol{t}_T^*}\,\mathrm{d}\Gamma,\end{aligned} \tag{8}$$

and (6b) simplifies to the variational equality

$$\mathcal{R}_t = \int_{\Gamma_f^{\text{stick}} \cup \Gamma_f^{\text{slip}}} \boldsymbol{\mu} \cdot \underbrace{(\boldsymbol{n}_f \otimes \boldsymbol{n}_f) \cdot [\![\boldsymbol{u}]\!]}_{=g_N \boldsymbol{n}_f} \, \mathrm{d}\Gamma + \int_{\Gamma_f^{\text{stick}}} \boldsymbol{\mu} \cdot \underbrace{(\mathbf{1} - \boldsymbol{n}_f \otimes \boldsymbol{n}_f) \cdot [\![\Delta \boldsymbol{u}]\!]}_{=\Delta \boldsymbol{g}_T} \, \mathrm{d}\Gamma = 0, \qquad \forall \boldsymbol{\mu} \in \boldsymbol{\mathcal{M}}(t), \tag{9}$$

enforcing the impenetrability constraint $g_N = 0$ on $\Gamma_f^{\text{stick}} \cup \Gamma_f^{\text{slip}}$ and the no-slip condition $\Delta \boldsymbol{g}_T = \mathbf{0}$ on $\Gamma_f^{\text{stick}}$. We emphasize that in Eq. (8), $\boldsymbol{t}_T^*$ does not represent a primary unknown and depends on $\boldsymbol{u}$ and $\boldsymbol{t}$ – precisely, the normal component of the traction acting on $\Gamma_f$ – according to Eq. (2a).

Let us consider a 3D computational grid consisting of $n_e$ elements that conform to the discontinuity surfaces $\Gamma_f^+$ and $\Gamma_f^-$ such that $\Omega \approx \Omega^h = \bigcup_{i=1}^{n_e} \Omega_i^e$, with $\Omega_i^e$ indicating nonoverlapping tetrahedral or hexahedral elements. The discrete domain boundary and discontinuity surface are denoted by $\Gamma^h = \Gamma_{\boldsymbol{u}}^h \cup \Gamma_{\boldsymbol{\sigma}}^h$ and $\Gamma_f^h$, respectively. The discrete displacement ($\boldsymbol{u}^h$) and traction vector ($\boldsymbol{t}^h$) fields may be expressed as:

$$\boldsymbol{u} \approx \boldsymbol{u}^h = \mathring{\boldsymbol{u}}^h + \bar{\boldsymbol{u}}^h = \sum_{i=1}^{n_{\boldsymbol{u}}} \boldsymbol{N}_i^{\boldsymbol{u}} d_i + \sum_{i=1}^{n_{\bar{\boldsymbol{u}}}} \boldsymbol{N}_{(n_{\boldsymbol{u}}+i)}^{\boldsymbol{u}} \bar{d}_i, \tag{10}$$

$$\boldsymbol{t} \approx \boldsymbol{t}^h = \mathring{\boldsymbol{t}}^h + \bar{\boldsymbol{t}}^h = \sum_{i=1}^{n_{\boldsymbol{t}}} \boldsymbol{N}_i^{\boldsymbol{t}} \lambda_i + \sum_{i=1}^{n_{\bar{\boldsymbol{t}}}} \boldsymbol{N}_{(n_{\boldsymbol{t}}+i)}^{\boldsymbol{t}} \bar{\lambda}_i. \tag{11}$$

In (10), $\boldsymbol{u}^h$ is given by superposition of function $\mathring{\boldsymbol{u}}^h$, which honors homogeneous Dirichlet conditions on $\Gamma_{\boldsymbol{u}}^h$, and function $\bar{\boldsymbol{u}}^h$, which provides a *lifting* [30] of an approximation of the boundary datum given in Eq. (1b). Similarly, $\boldsymbol{t}^h$ is split into sum of function $\mathring{\boldsymbol{t}}^h$, which vanishes where traction components are known, and function $\bar{\boldsymbol{t}}^h$, which interpolates such known components — namely, normal and tangential components on $\Gamma_f^{h,\text{open}}$ and $\Gamma_f^{h,\text{open}} \cup \Gamma_f^{h,\text{slip}}$, respectively. The unknown discrete displacement ($d_i$) and traction ($\lambda_i$) degrees of freedom (DOFs) are collected in vectors $\boldsymbol{d}$ and $\boldsymbol{\lambda}$, respectively. Known discrete displacement ($\bar{d}_i$) and traction DOFs ($\bar{\lambda}_i$) are collected in vector $\bar{\boldsymbol{d}}$ and $\bar{\boldsymbol{\lambda}}$. Vector functions $\boldsymbol{N}_i^{\boldsymbol{u}}$ and $\boldsymbol{N}_i^{\boldsymbol{t}}$ are bases for discrete function spaces $\boldsymbol{\mathcal{U}}^h$, $\boldsymbol{\mathcal{V}}^h$, and $\boldsymbol{\mathcal{M}}^h(\boldsymbol{t}^h)$ that approximate $\boldsymbol{\mathcal{U}}$, $\boldsymbol{\mathcal{V}}$, and $\boldsymbol{\mathcal{M}}(\boldsymbol{t})$, respectively. Clearly, we have $(n_{\boldsymbol{u}} + n_{\bar{\boldsymbol{u}}}) = \dim(\boldsymbol{\mathcal{U}}^h)$, $n_{\boldsymbol{u}} = \dim(\boldsymbol{\mathcal{V}}^h)$, and $(n_{\boldsymbol{t}} + n_{\bar{\boldsymbol{t}}}) = \dim(\boldsymbol{\mathcal{M}}(\boldsymbol{t}))$.

**Remark 4.** Displacement DOFs are associated to mesh nodes and expressed in terms of the global reference frame, i.e. three DOFs are linked to each mesh node and correspond to displacement components in $x$-, $y$-, and $z$-direction, respectively. By distinction, traction DOFs are associated to mesh nodes lying on $\Gamma_f^h$ and are more conveniently expressed in a local reference frame [18]. For each mesh node lying on $\Gamma_f$ three Lagrange multiplier components are defined that correspond to a traction component in the normal direction ($\boldsymbol{n}_f^h$) and two traction components in the tangential directions ($\boldsymbol{m}_1^h, \boldsymbol{m}_2^h$).

The discretization of the two residual equations (6a) and (9) is based on approximations (10) and (11) and the corresponding discrete function spaces. Time integration is carried out by partitioning the time interval $\mathbb{I}$ into $n_{\delta t}$ subintervals $\mathbb{I}_n = (t_{n-1}, t_n]$, $n = 1, \ldots, n_{\delta t}$. A backward Euler integrator is adopted. Let the vector $\boldsymbol{x}_n^T = \{\boldsymbol{d}_n^T, \boldsymbol{\lambda}_n^T\}$ represent the discrete solution at time $t_n$. The fully-discrete system of equations is provided by the global residual vector $\boldsymbol{r}$

$$\boldsymbol{r}(\boldsymbol{d}_n, \boldsymbol{\lambda}_n) = \begin{Bmatrix} \boldsymbol{r}_{\boldsymbol{u}} \\ \boldsymbol{r}_{\boldsymbol{t}} \end{Bmatrix} = \mathbf{0}, \tag{12}$$

where $\boldsymbol{r}_{\boldsymbol{u}}$ and $\boldsymbol{r}_{\boldsymbol{t}}$ are assembled as sum of element contributions. For detailed expressions of $\boldsymbol{r}_{\boldsymbol{u}}$ and $\boldsymbol{r}_{\boldsymbol{t}}$ see Appendix A. The system of residual equations (12) is nonlinear due to the constraint conditions in (1k). To drive $\boldsymbol{r}$ to zero a full Newton iteration is performed. If $k$ denotes the iteration count, with $\boldsymbol{x}_n^k$ the $k$-estimate of the solution at $t_n$, the updated solution vector is determined by: (i) solving: $\mathcal{J}_n^k \delta \boldsymbol{x} = -\boldsymbol{r}_n^k$, and (ii) updating: $\boldsymbol{x}_n^{k+1} = \boldsymbol{x}_n^k + \delta \boldsymbol{x}$, with $\mathcal{J}_n^k = (\partial \boldsymbol{r} / \partial \boldsymbol{x})_n^k$ the Jacobian matrix evaluated at the current configuration $k$.

Since the residual vector $\boldsymbol{r}$ consists of two blocks, a $2 \times 2$ block structure is reflected in the Jacobian system that needs to be solved in each Newton update, namely [23]

$$\begin{bmatrix} A & B_1 \\ B_2 & \end{bmatrix}_n^k \begin{Bmatrix} \delta \boldsymbol{d} \\ \delta \boldsymbol{\lambda} \end{Bmatrix} = - \begin{Bmatrix} \boldsymbol{r}_{\boldsymbol{u}} \\ \boldsymbol{r}_{\boldsymbol{t}} \end{Bmatrix}_n^k \tag{13}$$

with

$$A = K + E, \tag{14a}$$
$$B_1 = C - F, \tag{14b}$$
$$B_2 = C^T = (B_1 + F)^T. \tag{14c}$$

Here, $K$ is the elasticity stiffness matrix, $C$ is the rectangular block coupling the displacement variables with the Lagrange multipliers, and $E$ and $F$ are low-rank matrices arising from the linearization of the tangential traction $\boldsymbol{t}_T^*$ acting on $\Gamma_f^{h,\text{slip}}$. Obviously, if the discontinuity is entirely in contact mode, i.e. $\Gamma_f^{h,\text{stick}} = \Gamma_f^h$, both $E$ and $F$ vanish, therefore $B_2 = B_1^T$. Detailed expressions for each sub-matrix are given in Appendix A. Matrix $K$ is symmetric if $\mathbf{C}$ is so, which is the case for linear elastic material behavior. Note that this is also guaranteed in the framework of elasto-plasticity if the plastic flow rule is associative and the kinematic hardening is symmetric [1]. In addition, $K$ is positive definite. As to matrices $E$ and $F$, the following propositions hold true.

**Proposition 2.1.** *The matrix $E = E(\boldsymbol{d}, \boldsymbol{\lambda})$ is symmetric positive semidefinite.*

**Proposition 2.2.** *Let $\|\Delta \boldsymbol{g}_T\|_2 \to 0$ and $\|\boldsymbol{u}^h\|_2 > 0$. Then for any compatible matrix norm $\|\cdot\|_p$ we have: (i) $\|E\|_p \to +\infty$, and (ii) $\|F\|_p \to 0$.*

Proposition 2.1 ensures that matrix $A$, i.e. the (1,1)-block of the Jacobian matrix, is symmetric positive definite (SPD). The proof is given in Appendix B. Proposition 2.2 tells us that at *incipient sliding condition* –namely, when the shear stress magnitude exceeds the limiting value $\tau_{\max}$ but $\|\Delta \boldsymbol{g}_T\|_2 \to 0$ – the coefficients of $E$ can become numerically unstable. Actually, if the limiting shear stress is exceeded with no sliding, contribution to $E$ and $F$ should be zero. In practice, terms in $E$ are computed only if $\Delta \boldsymbol{g}_T$ is "large enough", i.e. numerically larger than a prescribed tolerance. For additional details regarding the implementation see [23,29]. The proof of Proposition 2.2 is given in Appendix C.

## 3. Block triangular preconditioners

The solution at each Newton iteration of the Jacobian system (13) represents the most expensive computational kernel in Lagrange multiplier-based contact mechanics formulations. Such generalized non-symmetric saddle point systems of linearized equations are typically highly ill-conditioned. Because of the large size of realistic 3D models, global iterative solution techniques are the method of choice—typically non-symmetric Krylov solvers, such as the generalized minimal residual (GMRES, [31]) or Bi-Conjugate Gradient Stabilized (Bi-CGStab, [32]) methods. Nevertheless, to achieve fast and robust convergence any iterative method must be coupled with some form of preconditioning, which should be at the same time as cheap and accurate as possible.

To derive the preconditioner, consider first the following block LDU-decomposition of the Jacobian matrix:

$$\mathcal{J} = \begin{bmatrix} I_u & \\ B_2 A^{-1} & I_t \end{bmatrix} \begin{bmatrix} A & \\ & S \end{bmatrix} \begin{bmatrix} I_u & A^{-1} B_1 \\ & I_t \end{bmatrix}, \tag{15}$$

with $I_u$ and $I_t$ identity operators in $\mathbb{R}^{n_u}$ and $\mathbb{R}^{n_t}$, respectively, and $S = -B_2 A^{-1} B_1$ the *Schur complement* of $A$ in $\mathcal{J}$. The exact inverse of $\mathcal{J}$ can be now written as:

$$\mathcal{J}^{-1} = \begin{bmatrix} I_u & -A^{-1} B_1 \\ & I_t \end{bmatrix} \begin{bmatrix} A^{-1} & \\ & S^{-1} \end{bmatrix} \begin{bmatrix} I_u & \\ -B_2 A^{-1} & I_t \end{bmatrix}. \tag{16}$$

Having (16) as starting point for the preconditioner design, we elect to use the following block triangular operator as a right-preconditioner for $\mathcal{J}$:

$$\mathcal{P}^{-1} = \begin{bmatrix} \tilde{A} & B_1 \\ & \tilde{S} \end{bmatrix}^{-1} = \begin{bmatrix} \tilde{A}^{-1} & \\ & I_t \end{bmatrix} \begin{bmatrix} I_u & -B_1 \\ & I_t \end{bmatrix} \begin{bmatrix} I_u & \\ & \tilde{S}^{-1} \end{bmatrix}, \tag{17}$$

where $\tilde{A}^{-1}$ and $\tilde{S}^{-1}$ are suitable preconditioners for $A$ and $S$—i.e., $\tilde{A}^{-1}$ and $\tilde{S}^{-1}$ can be regarded as inexact solvers for linear systems involving $A$ and $S$. As usual, $\mathcal{P}^{-1}$ is treated as an implicit operator since the objective in a preconditioned iterative method is to compute its action on block vectors. Overall, the application of the preconditioner requires one solve for $\tilde{A}$, one solve for $\tilde{S}$, and one matrix–vector product with $B_1$.

The block approach mentioned above is the general preconditioning framework that is typically used for saddle-point problems [25,26]. The motivation behind block triangular preconditioning stems from the fact that use of exact solves in (17)–namely, $\tilde{A}^{-1} = A^{-1}$ and $\tilde{S}^{-1} = S^{-1}$—results in at most two inner iterations for an iterative method. Indeed, the spectrum of the preconditioned matrix consists of the unit eigenvalue only and $\mathcal{J}\mathcal{P}^{-1}$ has minimum polynomial of degree two [25]. Clearly, the availability of efficient, approximate solvers for both the mechanical subproblem involving $A$ and the Schur complement $S$ is the key factor for the performance of block triangular preconditioners. Several effective off-the-shelf SPD algebraic preconditioners $\tilde{A}^{-1}$ are already available, in the field of incomplete factorizations, approximate inverses, domain decomposition and multigrid methods, e.g. [33–37]. Unfortunately, designing robust and at the same time inexpensive preconditioners for the exact Schur complement – which is almost completely dense due to the term $A^{-1}$ – is a challenging task and represents the cornerstone the global preconditioner behavior rests on. The Schur complement approximation can be very problem-dependent and still represents an open issue in the simulation of contact mechanics by Lagrange multipliers, such that this technique is often avoided in large-scale simulations because no well-established iterative solution strategy is available. In the remainder of this section we present three approaches for approximating and inverting $\tilde{S}$.

**Remark 5.** Right-preconditioning strategies are typically adopted for norm-minimizing iterative methods such as GMRES. In case of left-preconditioning, a lower block triangular version of $\mathcal{P}^{-1}$ is usually preferred:

$$\mathcal{P}^{-1}_{\text{left}} = \begin{bmatrix} \tilde{A} & \\ B_2 & \tilde{S} \end{bmatrix}^{-1} = \begin{bmatrix} I_{\boldsymbol{u}} & \\ & \tilde{S}^{-1} \end{bmatrix} \begin{bmatrix} I_{\boldsymbol{u}} & \\ -B_2 & I_{\boldsymbol{t}} \end{bmatrix} \begin{bmatrix} \tilde{A}^{-1} & \\ & I_{\boldsymbol{t}} \end{bmatrix}. \tag{18}$$

Note that with Bi-CGStab the computational experience typically does not show significant differences between the two approaches.

### 3.1. Block diagonal (BD) Schur complement

The block diagonal approximate Schur complement $\tilde{S}_{\text{BD}}$ is built exploiting some physics-based assumptions that are related to the locality of deformation—i.e., a variation of stress on a fault interface element produces a relevant variation of displacements only on the close vicinity of that element. For example, in [38,39] the authors utilized such assumption for computing the increment in Newton–Raphson's algorithm. Here, the same idea is used for preconditioning purposes. A similar strategy was proposed in the field of mixed finite element coupled poromechanics in [40].

The procedure for constructing $\tilde{S}_{\text{BD}}$ consists in clustering the traction unknown DOFs associated with each pair of mesh nodes lying on $\Gamma_f^h$ into *supernodes*—for a definition of supernode see, e.g., [41]. The size $n_{\boldsymbol{t}}^{(k)}$ of the $k$th supernode is three for node pairs belonging to $\Gamma_f^{h,\text{stick}}$, but reduces to one if the pair lies on $\Gamma_f^{h,\text{slip}}$ where only the normal traction component is unknown. For each supernode we introduce the following two linear operators: (i) a restriction operator, $R_{\boldsymbol{t}}^{(k)}$, onto the $k$th supernode unknown traction DOFs, i.e.

$$R_{\boldsymbol{t}}^{(k)} : \mathbb{R}^{n_t} \to \mathbb{R}^{n_{\boldsymbol{t}}^{(k)}}, \boldsymbol{\lambda} \mapsto \boldsymbol{\lambda}^{(k)} = R_{\boldsymbol{t}}^{(k)} \boldsymbol{\lambda}, \tag{19}$$

and (ii) a restriction operator, $R_{\boldsymbol{u}}^{(k)}$, onto the $n_{\boldsymbol{u}}^{(k)}$ unknown displacement DOFs coupled to traction DOFs that are collected in $\boldsymbol{\lambda}^{(k)}$, i.e.

$$R_{\boldsymbol{u}}^{(k)} : \mathbb{R}^{n_u} \to \mathbb{R}^{n_{\boldsymbol{u}}^{(k)}}, \boldsymbol{d} \mapsto \boldsymbol{d}^{(k)} = R_{\boldsymbol{u}}^{(k)} \boldsymbol{d}. \tag{20}$$

Using $R_{\boldsymbol{t}}^{(k)}$ and $R_{\boldsymbol{u}}^{(k)}$, we define the following three sub-matrices associated with the $k$th supernode (Fig. 2):

$$A^{(k)} = R_{\boldsymbol{u}}^{(k)} A R_{\boldsymbol{u}}^{(k),T}, \qquad B_1^{(k)} = R_{\boldsymbol{u}}^{(k)} B_1 R_{\boldsymbol{t}}^{(k),T}, \qquad B_2^{(k)} = R_{\boldsymbol{t}}^{(k)} B_2 R_{\boldsymbol{u}}^{(k),T}. \tag{21}$$

As $A^{(k)}$ is a diagonal block of an SPD matrix, it is non-singular and can be regularly inverted. Finally, the BD approximate Schur complement is defined as:

$$\tilde{S}_{\text{BD}} = -\sum_{k=1}^{n_{\text{SN}}} R_{\boldsymbol{t}}^{(k),T} \left( B_2^{(k)} A^{(k),-1} B_1^{(k)} \right) R_{\boldsymbol{t}}^{(k)}, \tag{22}$$

with $n_{\text{SN}}$ the number of supernodes.

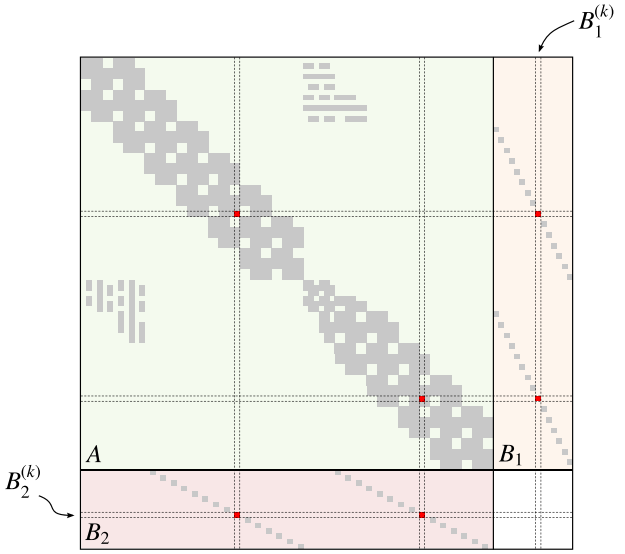



**Fig. 2.** Non-zero entries (in red) of $A$, $B_1$, and $B_2$ required in the computation of the contribution to $\tilde{S}_{\mathrm{DB}}$ according to (22). The pattern of the Jacobian matrix is shown in gray.

The BD approach allows for obtaining either a symmetric or a non-symmetric matrix, according to the fact that $B_1 = B_2^T = C$, or $B_1 = (C - F)$ and $B_2 = C^T$. Given the block structure of $\tilde{S}_{\mathrm{BD}}$, applying exactly its inverse through a direct solver is the optimal choice.

### 3.2. Least square commutator (LSC) Schur complement

The LSC Schur complement approximation was originally introduced in [27] as a "black-box" variant of the so-called *pressure convection–diffusion* (PCD) preconditioner for the incompressible Navier–Stokes equations [42]. The core idea of the standard versions of PCD preconditioning is to circumvent the inversion of the (1,1)-block in the Jacobian replacing the exact Schur complement by a triple product approximation that requires two linear solves for a pressure diffusion- and mass-matrix, respectively, and a matrix–vector product with a matrix explicitly constructed—namely, the discrete convection–diffusion operator on the pressure space. To provide a fully algebraic alternative, the LSC approach mimics PCD methods by constructing the preconditioner based on an algebraic commuting relationship that utilizes Jacobian sub-matrices only.

Let us derive the LSC preconditioner $\tilde{S}_{\mathrm{LSC}}$. The objective is to approximate the exact Schur complement as follows:

$$\tilde{S}_{\mathrm{LSC}} = -B_2 B_1 A_t^{-1}, \tag{23}$$

where $A_t \in \mathbb{R}^{n_t \times n_t}$ is constructed such that $B_1 A_t^{-1} \approx A^{-1} B_1$, i.e. the commutator relationship $AB_1 \approx B_1 A_t$ [27]. This is achieved computing each column of $A_t$ as the solution to the system of the normal equations associated with the least square problem

$$\min \left\| [AB_1]_{*j} - B_1 [A_t]_{*j} \right\|_2, \tag{24}$$

where $[AB_1]_{*j}$ and $[A_t]_{*j}$ denote the $j$th column of matrix $AB_1$ and $A_t$, respectively. Another way to define the operator $A_t$ implies the minimization of the Frobenius norm

$$\| AB_1 - B_1 A_t \|_F . \tag{25}$$

With both definitions, $A_t$ reads:

$$A_t = (B_1^T B_1)^{-1} B_1^T A B_1, \tag{26}$$

hence the LSC Schur complement approximation is

$$\tilde{S}_{\mathtt{LSC}} = -B_2 B_1 (B_1^T A B_1)^{-1} B_1^T B_1, \tag{27}$$

with the related inverse

$$\tilde{S}_{\mathtt{LSC}}^{-1} = -(B_1^T B_1)^{-1} B_1^T A B_1 (B_2 B_1)^{-1}. \tag{28}$$

From (28), it is clear the advantage of the LSC approach that does not require $A^{-1}$ while inverting $\tilde{S}_{\mathtt{LSC}}$. Note that $\tilde{S}_{\mathtt{LSC}}$ is non-symmetric. However, if we allow $B_2 \approx B_1^T = B$ the preconditioner can be symmetrized. Both $(B_2 B_1)$ and $(B_1^T B_1)$ have the structure of a mass matrix for the Lagrange multiplier space. In our fault/fracture formulation [23] a piecewise constant interpolation is utilized for the Lagrange multipliers, which leads to block diagonal $(B_2 B_1)$ and $(B_1^T B_1)$, with block size at most $3 \times 3$. Hence, the computation and application of $(B_2 B_1)^{-1}$ and $(B_1^T B_1)^{-1}$ is totally inexpensive.

**Remark 6.** In the solution of the incompressible Navier–Stokes equations, the inversion of $B_1^T B_1$ can be done only inexactly and the LSC Schur complement is an approximation of the optimal operator [27]. By distinction, in our case we can invert $B_1^T B_1$ exactly and inexpensively. Thus, the available Schur complement is the optimal approximation in the framework of the least square commutator. In particular, both the "static" and "adaptive" techniques presented in [27] provide the same result, being the unique minimum of the function (25).

The following result holds true for the eigenspectrum of the LSC Schur complement preconditioner.

**Theorem 3.1.** *Let $\mathcal{J} \in \mathbb{R}^{(n_u+n_t)\times(n_u+n_t)}$ be the saddle-point matrix in (13), with $A \in \mathbb{R}^{n_u \times n_u}$ SPD, $B_1 = B \in \mathbb{R}^{n_u \times n_t}$ full-rank, $B_2 = B^T$ and $n_t < n_u$. Then, the eigenvalues of $S\tilde{S}_{\mathtt{LSC}}^{-1}$ are real and bounded from below by 1.*

Theorem 3.1 ensures that the eigenspectrum of the preconditioned problem $\mathcal{J}\mathcal{P}^{-1}$ has a theoretical lower bound at 1 whenever $A^{-1}$ is applied exactly. The proof is given in Appendix D.

**Remark 7.** If we set $B^\dagger = (B^T B)^{-1} B^T$, i.e. the pseudo-inverse for the full-rank matrix $B = B_1$ according to the Moore–Penrose definition, the inverse of the symmetrized $\tilde{S}_{\mathtt{LSC}}$ can be formally written as

$$\tilde{S}_{\mathtt{LSC}}^{-1} = -B^\dagger A B^{\dagger,T}. \tag{29}$$

### 3.3. Factorized sparse approximate inverse (FSAI) Schur complement

The FSAI Schur complement approximation is built by exploiting an algebraic FSAI preconditioner of the (1,1)-block in $\mathcal{J}$. The basic idea of FSAI was originally introduced by Kolotilina and Yeremin [43] and consists of building a sparse approximation $G$ of the inverse of the exact lower Cholesky factor $L$ of $A$ by solving the Frobenius norm minimization problem

$$\min_{G \in \mathbb{R}^{n_u \times n_u}} \|I_u - GL\|_F, \tag{30}$$

with $G$ a lower triangular $n_u \times n_u$ matrix with non-zero pattern $\mathcal{S}_G$. One of the main difficulties in the computation of a good approximation is the selection of $\mathcal{S}_G$. The optimal a priori choice of an appropriate nonzero pattern is a difficult task which has been addressed either statically or dynamically—see, e.g., [44–47] and references therein.

Here, we use the adaptive FSAI variant implemented in the FSAIPACK software package [33]. The idea for the pattern detection was extended following [47] so as to select dynamically an optimal lower triangular non-zero pattern of $G$. The advantages of such an approach are manifold: (i) the algorithm building and applying $G$ is very robust, as it does not suffer from the typical numerical instabilities characterizing incomplete factorizations; (ii) the computation of $G$ can be done independently row-by-row, thus being perfectly parallelizable; (iii) the convergence ensured by adaptive FSAI is typically fast with SPD structural stiffness matrices.

Once $G$ is available, it allows for the explicit computation of the Schur complement through matrix–matrix products. In fact, setting $A^{-1} \approx \tilde{A}^{-1} = G^T G$, the approximate Schur complement can be written as

$$\tilde{S}_{\mathtt{FSAI}} = -B_2 G^T G B_1, \tag{31}$$

providing a non-symmetric expression for $\tilde{S}$. Under the assumption $B_2 \approx B_1^T = B^T$, the approximate Schur complement is symmetric and negative definite. Depending on the selected strategy – namely, a non-symmetric or a symmetric one – the inverse of $\tilde{S}_{\texttt{FSAI}}$ is applied through either a direct solver or an appropriate preconditioner.

## 4. Numerical results

In this section three sets of numerical experiments are discussed to highlight advantages and difficulties associated with using the different Schur complement preconditioning strategies presented in Section 3. The first example (Test 1) addresses a synthetic academic problem with a single fracture to illustrate the mesh-dependency of the proposed approach with regards to the convergence rate of the iterative solver. The second set of examples (Test 2) includes two synthetic problems consisting of single- and multiple-fractured media, respectively, that provide evidence of the numerical performance of the proposed class of block triangular preconditioners as compared to a modern sparse direct solver. Finally, the robustness of the algorithms is investigated in the third set of examples (Test 3), which address real-life applications dealing with groundwater withdrawal and underground gas storage operations in fissured/faulted geological formations.

In all tests, to warrant a well-scaled system – i.e. an essential requirement to ensure that both linear momentum balance and constraint equations are satisfied when evaluating the convergence criterion – a block diagonal scaling of the Jacobian matrix $\mathcal{J}$ is applied prior to solving as suggested in [48], namely:

$$\hat{\mathcal{J}} = \begin{bmatrix} D^{-1/2} & \\ & I_t \end{bmatrix} \begin{bmatrix} A & B_1 \\ B_2 & \end{bmatrix} \begin{bmatrix} D^{-1/2} & \\ & I_t \end{bmatrix}, \tag{32}$$

with matrix $D$ defined as follows

$$[D]_{ij} = \begin{cases} [A]_{ij} & \text{if } i\text{th and } j\text{th displacement DOFs are associated with the same grid node,} \\ 0 & \text{otherwise.} \end{cases} \tag{33}$$

Note that, if displacement DOFs are ordered by node, matrix $D$ is block diagonal, with block size at most $3 \times 3$.

The non-symmetric linear system is solved by using GMRES as implemented in routine `mi24` from the HSL collection of FORTRAN linear algebraic solvers for sparse matrices [49]. The right-hand side is computed so that the true solution is the unitary vector. The null vector is used as initial guess $\boldsymbol{x}^{(0)}$. The stopping criterion is based on the reduction of the Euclidean norm of the iterative residual $\boldsymbol{r}^{(k)}$ below a specified tolerance $\tau$, i.e. $\|\boldsymbol{r}^{(k)}\|_2 \leq \tau \|\boldsymbol{r}^{(0)}\|_2$, with $k$ the iteration number.

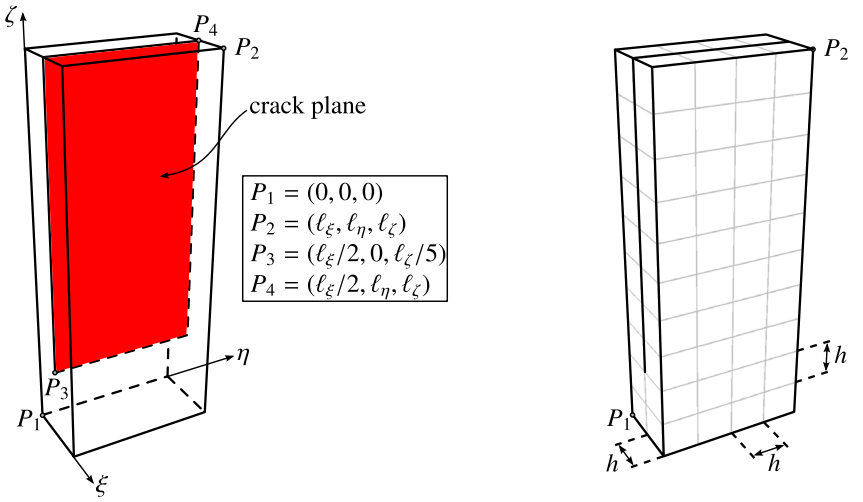
We consider several variants of the block triangular preconditioner $\mathcal{P}^{-1}$ based on different choices of preconditioning operators $\tilde{A}^{-1}$ and $\tilde{S}^{-1}$ for the (1,1)-block and its Schur complement, respectively. When the Schur complement approximation is tackled using either the block diagonal approach, $\tilde{S}^{-1} = \tilde{S}^{-1}_{\texttt{BD}}$, or the least square commutator approach, $\tilde{S}^{-1} = \tilde{S}^{-1}_{\texttt{LSC}}$, the application of $\tilde{S}^{-1}$ is performed exactly, while with the FSAI approach the application of $\tilde{S}^{-1}$ is performed either exactly, $\tilde{S}^{-1}_{\texttt{FSAI_DIR}}$, or inexactly by means of another FSAI approximation, $\tilde{S}^{-1}_{\texttt{FSAI-FSAI}}$. In any case, two options are considered for $\tilde{A}^{-1}$, namely either an incomplete Cholesky (IC) factorization, $\tilde{A}^{-1} = A^{-1}_{\texttt{IC}}$, or a factorized sparse approximate inverse (FSAI), $\tilde{A}^{-1} = A^{-1}_{\texttt{FSAI}}$.

In this work, for the IC factorization we use the variant with a prescribed degree of fill-in, IC($\rho$), proposed in [50], because it allows for an easy control of the memory occupation. The fill-in integer parameter $\rho$ is the number of non-zero entries computed for each row of the IC factor in excess to the number of non-zeros of the same row of the matrix. Moreover, we refer to the adaptive FSAI implementation available in the FSAIPACK software package [33]. We use the notation FSAI($n_{\max}$, $\epsilon$) to emphasize that default parameters are always used except for: (i) $n_{\max}$, i.e. an integer value denoting the maximum number of steps of the inner adaptive procedure for the dynamic construction of the non-zero pattern $\mathcal{S}_G$; and (ii) $\epsilon$, i.e. the tolerance to exit the adaptive procedure before achieving $n_{\max}$ steps. For additional algorithmic details we refer the reader to [33].

The computational performance is evaluated in terms of iteration count $n_{\text{iter}}$ and CPU times $T_p$ and $T_s$ for the preconditioner computation and the solver to converge, respectively, with $T_t = T_p + T_s$ the total time. The overall preconditioner density:

$$\mu = \frac{\texttt{nnz}(\tilde{A}^{-1}) + \texttt{nnz}(B_1) + \texttt{nnz}(\tilde{S}^{-1})}{\texttt{nnz}(A) + \texttt{nnz}(B_1) + \texttt{nnz}(B_2)}, \tag{34}$$

with `nnz` the function providing the number of non-zeros of a sparse matrix, is also evaluated whenever meaningful. The results reported below were obtained on an Intel(R) Xeon(R) CPU E5-2643 processor at 3.3 GHz with 256-GB




(a) Sketch of the problem setup. (b) Computational grid.

**Fig. 3.** Test 1: Single crack plane cutting through a homogeneous elastic block.

of memory. In this work, we are mainly interested in the algorithmic developments, hence we limit our attention to sequential computations only. Nevertheless, we will underline whenever a method can be potentially attractive also for an HPC environment.

### *4.1. Test 1: Single fractured elastic media*

A synthetic case, denoted as Test 1, consisting of a single crack plane embedded in a homogeneous isotropic domain $\Omega$, is first considered to assess the behavior in terms of iteration count needed to achieve convergence by each preconditioning strategy. A sketch of the model is provided in Fig. 3a. The simulation is defined in terms of dimensionless quantities. The block exhibits linear elastic behavior with unit Lamé parameters that correspond to a Young modulus $E = 1.0$ and Poisson ratio $\nu = 0.25$. The dimensions of the computational domain are $\ell_\xi = \ell$, $\ell_\eta = 2\ell$, and $\ell_\zeta = 5\ell$. The crack plane cuts the entire domain in the $y$-directions and spans 80% of the vertical dimension. A roller support condition over domain boundaries lying on planes $\xi$=0 and $\zeta$=0 is assigned. On such boundaries zero-displacements in $\eta$-direction are also enforced over $\eta = \ell_\eta/2$, , while a traction condition is prescribed on the remaining portion of the boundary. In practice, if the domain boundary, $\Gamma$, is decomposed as $\Gamma = \Gamma_{\boldsymbol{u}} \cup \Gamma_{\boldsymbol{\sigma}}$, with $\Gamma_{\boldsymbol{u}} \cap \Gamma_{\boldsymbol{\sigma}} = \emptyset$, homogeneous Dirichlet conditions for $\boldsymbol{u}$ are imposed in a strong way on $\Gamma_{\boldsymbol{u}}$, while traction is assigned on $\Gamma_{\boldsymbol{\sigma}}$ in a weak way. The domain has been discretized using hexahedral elements with characteristic mesh size $h$, as shown in Fig. 3b.

We consider a manufactured exact solution to have both control on the residual and the error. The exact fields for the displacement vector $\boldsymbol{u}$ and the discontinuity traction $\boldsymbol{\lambda}$ are given by:

$$\boldsymbol{u}(\boldsymbol{\xi}) = \frac{\bar{u}}{5\ell}\begin{bmatrix} 5\xi \\ 5\,(\eta - \ell) \\ -\zeta \end{bmatrix}, \qquad \boldsymbol{\lambda}(\boldsymbol{\xi}) = \frac{\bar{u}}{5\ell}\begin{bmatrix} 19 \\ 19 \\ 7 \end{bmatrix}, \tag{35}$$

with $\bar{u}$ an assigned given positive value. The traction condition prescribed on $\Gamma_{\boldsymbol{\sigma}}$ is inferred from (35).

We investigate the quality of the Schur complement approximation in the sequence of refined problems reported in Table 1. The iteration counts to converge of right-preconditioned full GMRES are provided in Table 2 for different options for $\tilde{S}$, namely $\tilde{S}_{\texttt{BD}}$, $\tilde{S}_{\texttt{LSC}}$, and $\tilde{S}_{\texttt{FSAI}}$. In order to provide evidence of the effect of the selected Schur complement approach only, the exact inverse of both $A$ and $\tilde{S}$ is applied via a nested direct or iterative solver. In this way, the results shown in Table 2 can be regarded as the *minimum* number of iterations that can be obtained with each Schur complement approximation. Of course, this approach with nested solvers cannot be competitive in terms of

**Table 1**
Test 1: Size and number of non-zeros of the Jacobian block matrices as a function of the characteristic mesh size $h$.

| $h$ | $n_{\text{total}}$ | $n_{\boldsymbol{u}}$ | $n_t$ | $n_t/n_{\boldsymbol{u}}$ | $\texttt{nnz}(A)$ | $\texttt{nnz}(B_1)$ | $\texttt{nnz}(B_2)$ |
|---|---|---|---|---|---|---|---|
| $\ell/2$ | 735 | 615 | 120 | 0.195 | 28,197 | 720 | 720 |
| $\ell/4$ | 3,699 | 3,267 | 432 | 0.132 | 189,225 | 2,592 | 2,592 |
| $\ell/8$ | 22,083 | 20,451 | 1,632 | 0.080 | 1,376,361 | 9,792 | 9,792 |
| $\ell/16$ | 148,995 | 142,659 | 6,336 | 0.044 | 10,476,873 | 38,016 | 38,016 |
| $\ell/32$ | 1,085,955 | 1,060,995 | 24,960 | 0.024 | 81,710,217 | 149,760 | 149,760 |
| $\ell/64$ | 8,272,899 | 8,173,827 | 99,072 | 0.012 | 645,325,065 | 594,432 | 594,432 |

**Table 2**
Test 1: Number of iteration required by $\mathcal{P}^{-1}$-preconditioned GMRES to achieve convergence as a function of mesh refinement for different strategies used to approximate the Schur complement $S$. The inverses $A^{-1}$ and $\tilde{S}^{-1}$ are applied *exactly* via nested either direct or iterative solvers.

| $h$ | $\mathcal{P}^{-1}\left(A^{-1}, \tilde{S}_{\texttt{BD}}^{-1}\right)$ | $\mathcal{P}^{-1}\left(A^{-1}, \tilde{S}_{\texttt{LSC}}^{-1}\right)$ | $\mathcal{P}^{-1}\left(A^{-1}, \tilde{S}_{\texttt{FSAI}(5,\,0.01)}^{-1}\right)$ | $\mathcal{P}^{-1}\left(A^{-1}, \tilde{S}_{\texttt{FSAI}(20,\,0.01)}^{-1}\right)$ |
|---|---|---|---|---|
| $\ell/2$ | 27 | 22 | 22 | 20 |
| $\ell/4$ | 34 | 27 | 29 | 25 |
| $\ell/8$ | 40 | 32 | 35 | 30 |
| $\ell/16$ | 48 | 39 | 41 | 36 |
| $\ell/32$ | 56 | 46 | 49 | 43 |
| $\ell/64$ | 65 | 54 | 57 | 51 |

**Table 3**
Test 2a and 2b: Size and number of non-zeros of the Jacobian block matrices.

| | $n_{\text{total}}$ | $n_{\boldsymbol{u}}$ | $n_t$ | $n_t/n_{\boldsymbol{u}}$ | $\texttt{nnz}(A)$ | $\texttt{nnz}(B_1)$ | $\texttt{nnz}(B_2)$ |
|---|---|---|---|---|---|---|---|
| Test 2a | 349,131 | 293,052 | 56,079 | 0.191 | 12,432,060 | 336,474 | 336,474 |
| Test 2b | 547,782 | 379,983 | 167,799 | 0.442 | 13,876,785 | 1,006,794 | 1,006,794 |

computational efficiency with an inexact application of $A^{-1}$ and $\tilde{S}^{-1}$. For this reason, we use Test 1 to evaluate only the mesh-dependency on the convergence rate of the proposed approaches.

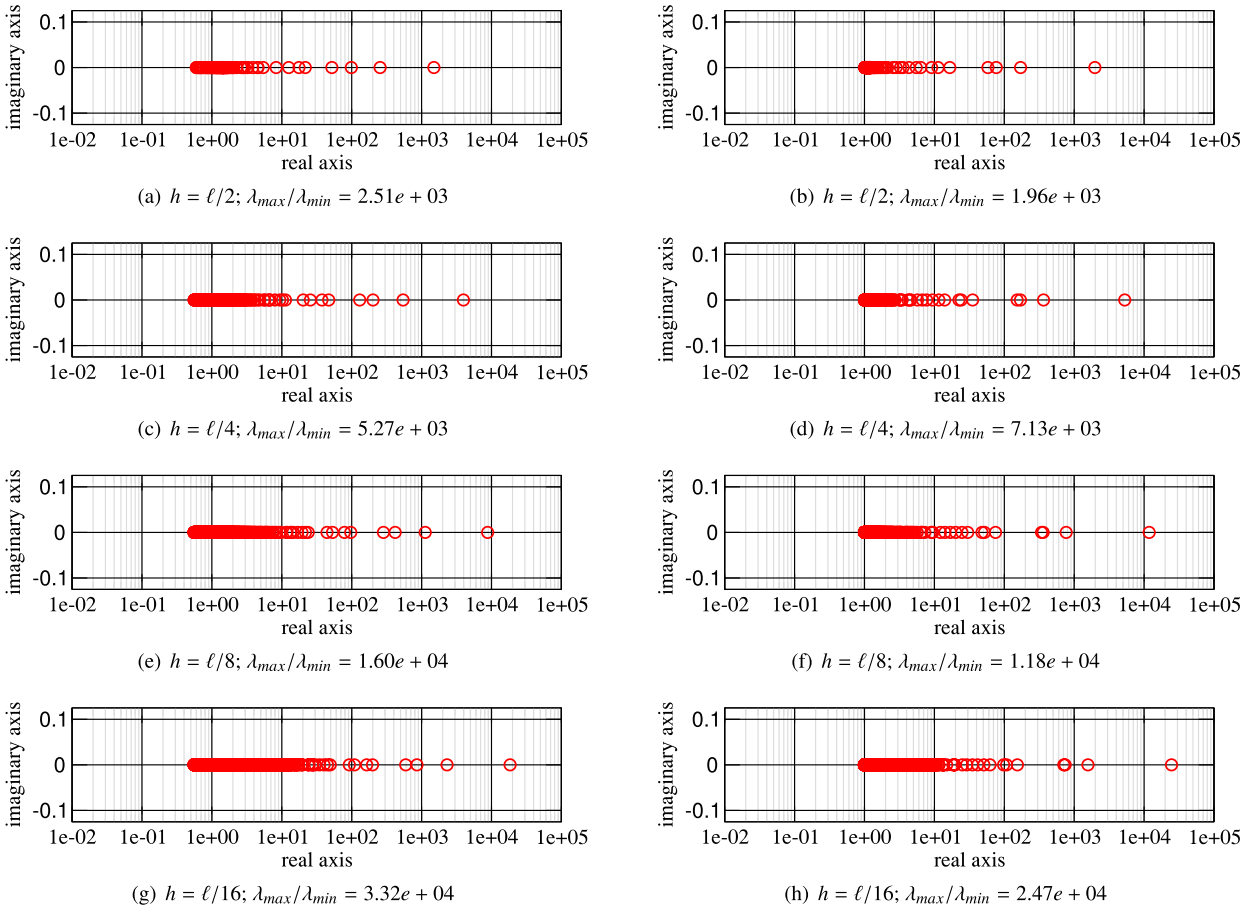
Quite obviously, the quality of $\tilde{S}_{\texttt{FSAI}}$ depends on the fill-in parameters $(n_{\max}, \epsilon)$. Increasing the density of the matrix $G$ in Eq. (31) yields an acceleration to the convergence rate. For reasonable values of $(n_{\max}, \epsilon)$, such as those reported in Table 2, the outcome is comparable to the one obtained with the LSC approximation. The proposed alternatives are not optimal with respect to $h$, however the use of $\tilde{S}_{\texttt{LSC}}$ appears to be slightly less sensitive to progressive grid refinement. In all cases, the BD approximation $\tilde{S}_{\texttt{BD}}$ gives the largest iteration count. It should be remembered, however, that this is also the less expensive Schur complement approach. Table 2 shows that the presented approaches are only mildly mesh-dependent, as the number of iterations grows by a factor about 2.5 when the grid is refined 32 times. Hence, these methods appear to preserve an attractive efficiency for a wide range of discretization sizes.

Figs. 4 and 5 show the eigenspectra of the preconditioned matrices $\hat{\mathcal{J}}\mathcal{P}^{-1}$ for the choices of $\tilde{S}$ of Table 2 and different values of $h$. As expected from Theorem 3.1, with $\tilde{S}_{\texttt{LSC}}$ the eigenvalues are bounded from below by 1. Also with the other approximations the smallest eigenvalue is close to 1, while a few right outliers tend to increase when $h$ decreases, thus accounting for the iteration count growth with $\ell/h$ shown in Table 2.

### *4.2. Test 2: Single and multiple fractured elastic media*

Tests 2a and 2b consider problems where: (i) the discretization of a single fracture plane is much more refined than the surrounding medium (2a), and (ii) several fractures cut the porous medium (2b). The largest examples in Test 1 are characterized by a very small ratio $n_t/n_{\boldsymbol{u}}$. By distinction, the ratio $n_t/n_{\boldsymbol{u}}$ increases here up to about 0.45. These test cases are used to investigate the computational performance of the proposed class of block preconditioners. The size and number of non-zeros of the related Jacobian block matrices are given in Table 3.

In Test 2a, the same domain and boundary conditions as Test 1 are considered (Fig. 3a) with a single crack cutting a homogeneous elastic block. An irregular tetrahedral discretization is used, with the largest element size $h_{\max} = 0.25\ell$ and a mesh refinement on the crack plane with edge length $h_{\min} = h_{\max}/10$ (Fig. 6). The numerical




(a) $h = \ell/2$; $\lambda_{max}/\lambda_{min} = 2.51e+03$

(b) $h = \ell/2$; $\lambda_{max}/\lambda_{min} = 1.96e+03$

(c) $h = \ell/4$; $\lambda_{max}/\lambda_{min} = 5.27e+03$

(d) $h = \ell/4$; $\lambda_{max}/\lambda_{min} = 7.13e+03$

(e) $h = \ell/8$; $\lambda_{max}/\lambda_{min} = 1.60e+04$

(f) $h = \ell/8$; $\lambda_{max}/\lambda_{min} = 1.18e+04$

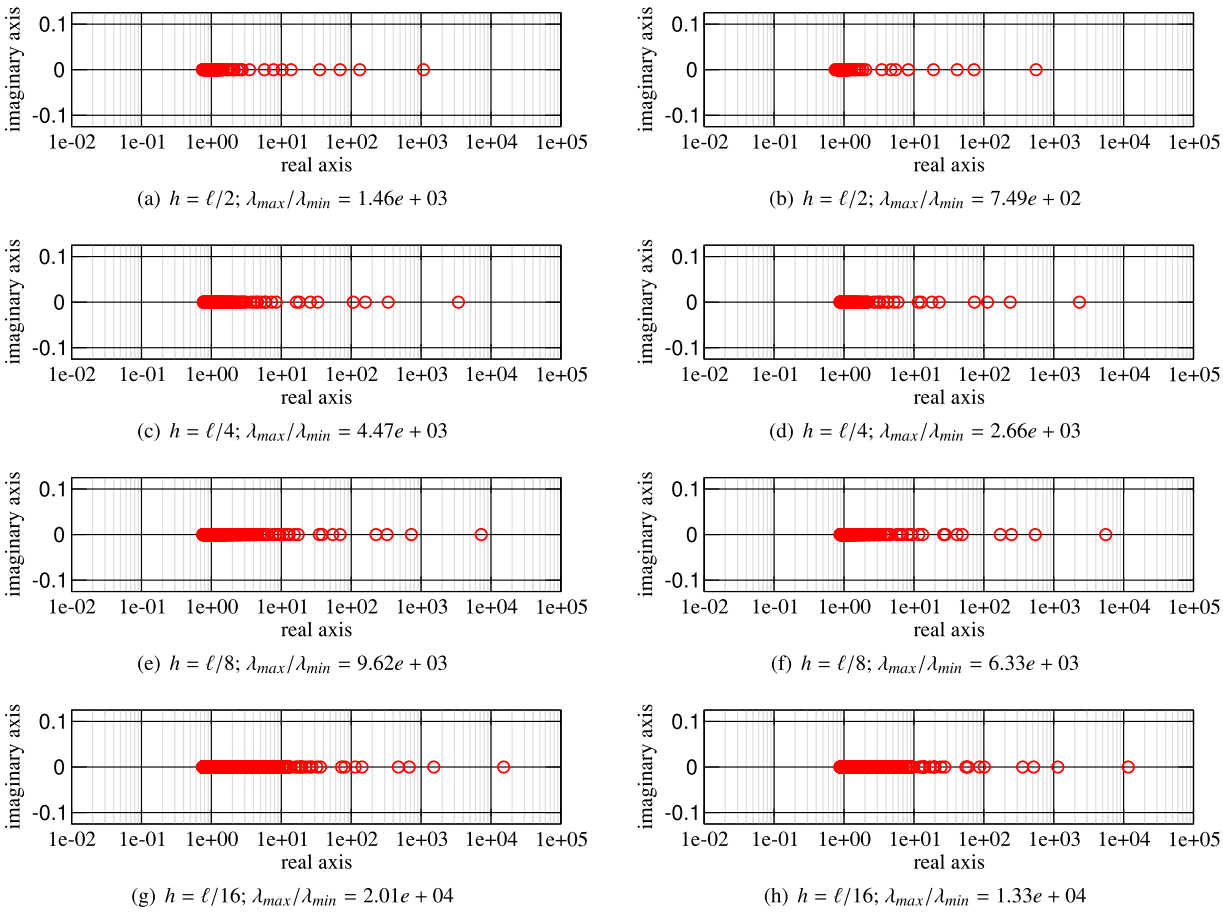
(g) $h = \ell/16$; $\lambda_{max}/\lambda_{min} = 3.32e+04$

(h) $h = \ell/16$; $\lambda_{max}/\lambda_{min} = 2.47e+04$

**Fig. 4.** Test 1: eigenvalue distribution of $\hat{\mathcal{J}}\mathcal{P}^{-1}$ with $\tilde{A}^{-1} = A^{-1}$ and $\tilde{S}_{\mathtt{BD}}$ (left panels) or $\tilde{S}_{\mathtt{LSC}}$ (right panels) for different values of the mesh size $h$.

performance obtained for different choices of $\tilde{S}$ and $\tilde{A}$ is provided in Table 4. We denote by $\tilde{S}_{\mathtt{FSAI-FSAI}}$ and $\tilde{S}_{\mathtt{FSAI-DIR}}$ the approximation (31) for the Schur complement with the inverse applied inexactly through a FSAI preconditioner of $\tilde{S}_{\mathtt{FSAI}}$ and a nested direct solver, respectively. The pair $(n_{\max}, \epsilon) = (50, 0.01)$ is used to set-up both $G$ in the computation of $\tilde{S}_{\mathtt{FSAI}}$ and the FSAI approximation of $\tilde{S}^{-1}_{\mathtt{FSAI}}$. For each option, the number of iterations obtained by using $\tilde{A}^{-1} = A^{-1}$ through a nested direct solver is also reported as a reference. Inspection of Table 4 reveals that the BD and LSC approaches can provide cheap and effective alternatives for the solution of faulted problems. Notice that the largest cost for the preconditioner set-up, which can undermine the overall performance, is mainly due to the FSAI computation for the (1,1) block $A$. This cost cannot be avoided with $\tilde{S}_{\mathtt{FSAI}}$. Such an approach, however, can be of interest when using a massively parallel implementation because of the almost ideal parallel potential of FSAI. It goes without saying that any other choice for $\tilde{A}^{-1}$, e.g., Algebraic Multigrid (AMG) methods, would be possible.

Test 2b addresses the multiple fractured elastic medium shown in Fig. 7. Fifteen horizontal fracture planes are considered in a $\ell_\xi \times \ell_\eta \times \ell_\zeta$ box, with $\ell_\xi = \ell_\eta = 2\ell$ and $\ell_\zeta = \ell$. The elastic properties of the medium are $E = 20$ GPa and $\nu = 0.3$. The fractured body has fixed zero displacements on the bottom boundary, is traction-free on the outer surfaces and is loaded on top. This problem shows an extreme case where the ratio $n_t/n_u$ grows up to 0.442. The numerical performance provided in Table 5 shows that this test case is more challenging than the ones already discussed. Nevertheless, all the proposed approaches are able to achieve convergence, with the exception of the LSC Schur complement and a FSAI approximation of the (1,1) block $A$. However, convergence is soon achieved also in this case by improving the quality of $\tilde{A}^{-1}$ as an approximation of $A^{-1}$ by either increasing the FSAI density or using an IC factorization. In Test 2b, the best performance is obtained by using the BD Schur complement.




(a) $h = \ell/2$; $\lambda_{max}/\lambda_{min} = 1.46e+03$

(b) $h = \ell/2$; $\lambda_{max}/\lambda_{min} = 7.49e+02$

(c) $h = \ell/4$; $\lambda_{max}/\lambda_{min} = 4.47e+03$

(d) $h = \ell/4$; $\lambda_{max}/\lambda_{min} = 2.66e+03$

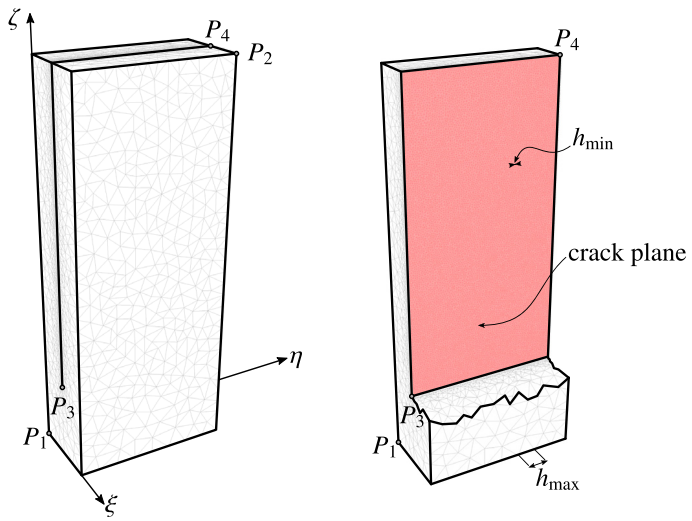
(e) $h = \ell/8$; $\lambda_{max}/\lambda_{min} = 9.62e+03$

(f) $h = \ell/8$; $\lambda_{max}/\lambda_{min} = 6.33e+03$

(g) $h = \ell/16$; $\lambda_{max}/\lambda_{min} = 2.01e+04$

(h) $h = \ell/16$; $\lambda_{max}/\lambda_{min} = 1.33e+04$

**Fig. 5.** Test 1: the same as Fig. 4 with $\tilde{S}_{\texttt{FSAI}(5,0.01)}$ (left panels) and $\tilde{S}_{\texttt{FSAI}(20,0.01)}$ (right panels).



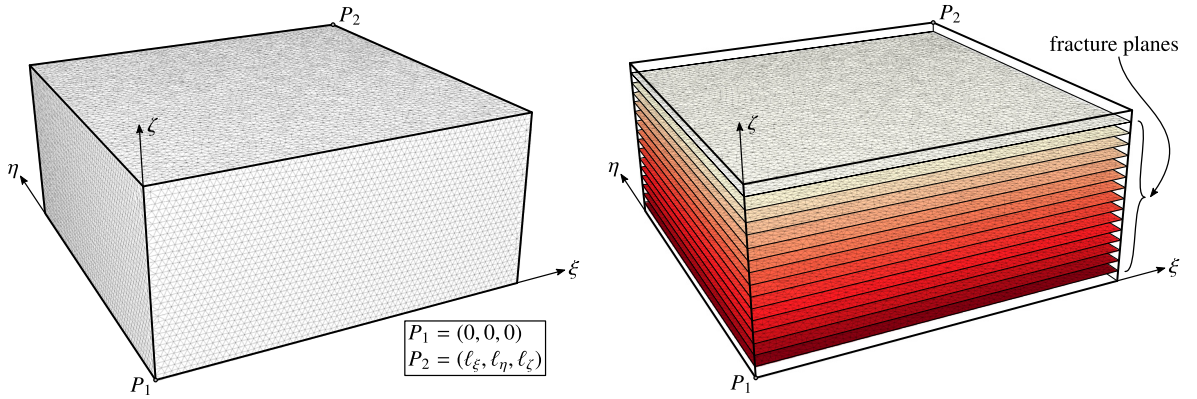

(a) Tetrahedral grid.

(b) Crack plane detail.

**Fig. 6.** Test 2a: Single crack plane cutting through a homogeneous elastic block discretized using a tetrahedral grid obtained with TetGen [51].

**Table 4**
Test 2a: Numerical performance of right-preconditioned GMRES(100), with exit tolerance $\tau = 10^{-8}$. The fill-in parameters $(n_{\max}, \epsilon)$ and $\rho$ for the FSAI, $A^{-1}_{\texttt{FSAI}}$, and IC, $A^{-1}_{\texttt{IC}}$, approximation of $A$ are also reported, respectively. Densities and CPU times are not reported when $\tilde{A}^{-1} = A^{-1}$.

| | $\tilde{S}_{\texttt{BD}}$ | | | | | $\tilde{S}_{\texttt{LSC}}$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $\mu$ | $n_{\text{iter}}$ | $T_p$ [s] | $T_s$ [s] | $T_t$ [s] | $\mu$ | $n_{\text{iter}}$ | $T_p$ [s] | $T_s$ [s] | $T_t$ [s] |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{FSAI}}(50,0.010)$ | 0.693 | 340 | 13.5 | 22.1 | 35.6 | 0.786 | 276 | 13.5 | 23.7 | 37.2 |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{FSAI}}(50,0.005)$ | 0.956 | 315 | 28.3 | 21.9 | 50.2 | 1.049 | 244 | 28.3 | 22.0 | 50.3 |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{IC}}(0)$ | 0.535 | 238 | 1.6 | 20.7 | 22.3 | 0.628 | 153 | 2.0 | 14.8 | 16.8 |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{IC}}(20)$ | 0.970 | 206 | 3.5 | 19.2 | 22.8 | 1.063 | 114 | 3.8 | 13.2 | 17.0 |
| $\tilde{A}^{-1} = A^{-1}$ | – | 21 | – | – | – | – | 14 | – | – | – |
| | $\tilde{S}_{\texttt{FSAI-FSAI}}$ | | | | | $\tilde{S}_{\texttt{FSAI-DIR}}$ | | | | |
| | $\mu$ | $n_{\text{iter}}$ | $T_p$ [s] | $T_s$ [s] | $T_t$ [s] | $\mu$ | $n_{\text{iter}}$ | $T_p$ [s] | $T_s$ [s] | $T_t$ [s] |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{FSAI}}(50,0.010)$ | 0.739 | 264 | 15.0 | 17.3 | 32.3 | 2.038 | 259 | 16.3 | 36.9 | 53.2 |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{FSAI}}(50,0.005)$ | 1.002 | 229 | 29.4 | 15.7 | 45.1 | 2.779 | 223 | 32.1 | 26.5 | 58.6 |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{IC}}(0)$ | 0.582 | 125 | 17.1 | 10.5 | 27.7 | 1.880 | 124 | 18.9 | 19.9 | 38.8 |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{IC}}(20)$ | 1.017 | 88 | 18.7 | 8.7 | 27.4 | 2.315 | 89 | 21.5 | 12.3 | 33.8 |
| $\tilde{A}^{-1} = A^{-1}$ | – | 12 | – | – | – | – | 12 | – | – | – |




(a) Tetrahedral grid. (b) Fracture planes.

**Fig. 7.** Test 2b: Multiple fractured medium cut by 15 horizontal cracks.

To get a sense of the effectiveness of the proposed approaches, we compare the performance reported in Tables 4 and 5 with that of a modern sparse direct solver such as PARDISO [52]. Using a state-of-the-art direct solution technique is the most robust way to address the problem of interest. CPU times and memory requirements obtained with PARDISO in Tests 2a and 2b are provided in Table 6. These results show that using a preconditioned iterative method might become mandatory when dealing with real-world applications with hundreds of thousands unknowns.

### *4.3. Test 3: Real faulted formations*

The performance, effectiveness and robustness of the proposed approaches are investigated in three applications referring to real-world problems dealing with fractured and faulted formations. The problems are denoted as Test 3a, 3b and 3c as follows.

- Test 3a (Fig. 8): the Wuxi area, Jiangsu Province, China, was subjected to a significant groundwater withdrawal from a shallow fresh aquifer system, which has caused an extensive land subsidence and the occurrence of important ground fissures [53,54]. The geological setting is characterized by a shallow rocky paleo-basement covered by compressible sedimentary deposits from the Yangtze River. A rock ridge, buried below the sedimentary sequence, triggers the generation of ground fissures in the permeable sediments along the plane

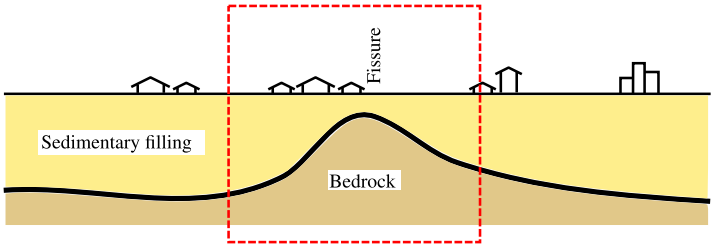
**Table 5**
Test 2b: the same as Table 4. The acronym 'nc' ('not converged') means that convergence was not achieved after 1000 iterations.

| | $\tilde{S}_{\texttt{BD}}$ | | | | | $\tilde{S}_{\texttt{LSC}}$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $\mu$ | $n_{\text{iter}}$ | $T_p$ [s] | $T_s$ [s] | $T_t$ [s] | $\mu$ | $n_{\text{iter}}$ | $T_p$ [s] | $T_s$ [s] | $T_t$ [s] |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{FSAI}}(50,0.010)$ | 0.662 | 482 | 13.7 | 54.4 | 68.1 | 1.137 | nc | *** | *** | *** |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{FSAI}}(50,0.005)$ | 0.862 | 491 | 27.7 | 62.0 | 89.7 | 1.337 | 866 | 26.6 | 135.0 | 161.6 |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{IC}}(0)$ | 0.546 | 701 | 2.6 | 84.0 | 86.6 | 1.021 | 603 | 2.6 | 98.7 | 98.3 |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{IC}}(20)$ | 0.947 | 574 | 3.9 | 76.7 | 80.6 | 1.422 | 407 | 4.3 | 73.5 | 77.8 |
| $\tilde{A}^{-1} = A^{-1}$ | – | 268 | – | – | – | – | 99 | – | – | – |
| | $\tilde{S}_{\texttt{FSAI-FSAI}}$ | | | | | $\tilde{S}_{\texttt{FSAI-DIR}}(*)$ | | | | |
| | $\mu$ | $n_{\text{iter}}$ | $T_p$ [s] | $T_s$ [s] | $T_t$ [s] | $\mu$ | $n_{\text{iter}}$ | $T_p$ [s] | $T_s$ [s] | $T_t$ [s] |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{FSAI}}(50,0.010)$ | 0.837 | 428 | 27.9 | 52.8 | 80.7 | 3.382 | 409 | 24.7 | 71.8 | 96.5 |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{FSAI}}(50,0.005)$ | 1.033 | 385 | 46.3 | 45.4 | 91.7 | 4.758 | 379 | 45.2 | 71.6 | 116.8 |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{IC}}(0)$ | 0.721 | 309 | 33.5 | 38.9 | 72.4 | 3.267 | 309 | 27.9 | 54.7 | 82.5 |
| $\tilde{A}^{-1} = A^{-1}_{\texttt{IC}}(20)$ | 1.123 | 227 | 35.5 | 32.4 | 67.9 | 3.668 | 229 | 29.4 | 44.1 | 73.5 |
| $\tilde{A}^{-1} = A^{-1}$ | – | 92 | – | – | – | – | 100 | – | – | – |

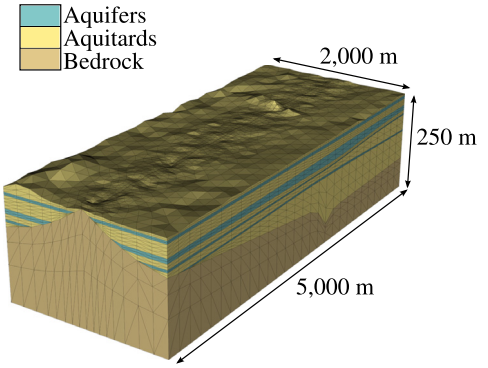
(*) Because of the large size of the Schur complement, the direct solver is replaced by 3 Conjugate Gradient iterations preconditioned by IC(0).

**Table 6**
Test 2a and 2b: CPU times and memory requirements for PARDISO.

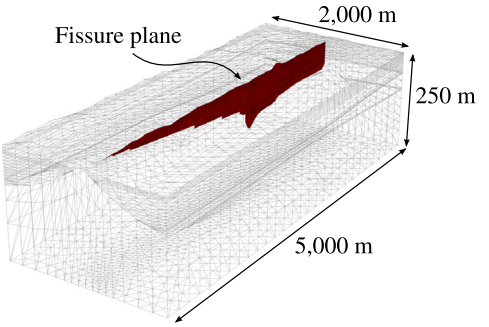
| | $T_t$ [s] | # non-zeros factorization | Memory occupation [GB] |
|---|---|---|---|
| Test 2a | 81.1 | 537,602,245 | 4.11 |
| Test 2b | 612.0 | 1,576,783,074 | 11.89 |




(a) Sketch of a typical geological setting fissure ground rupture (modified after [53]). The simulated configuration is highlighted by the red box.

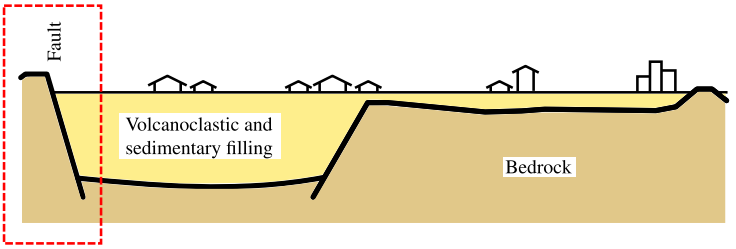



(b) Computational grid.

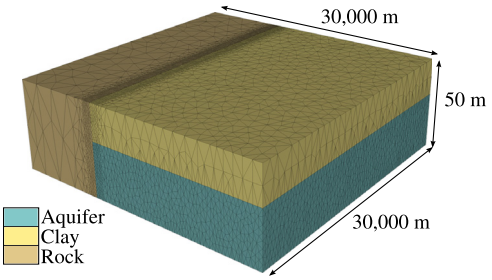



(c) Fissure plane trace.

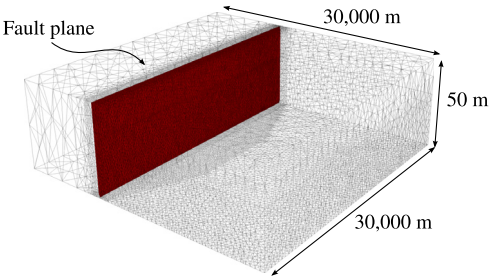
**Fig. 8.** Test 3a: Conceptual scheme, computational grid and fracture plane.




(a) Sketch of a typical geological setting fault ground rupture (modified after [53]). The simulated configuration is highlighted by the red box.




(b) Computational grid.



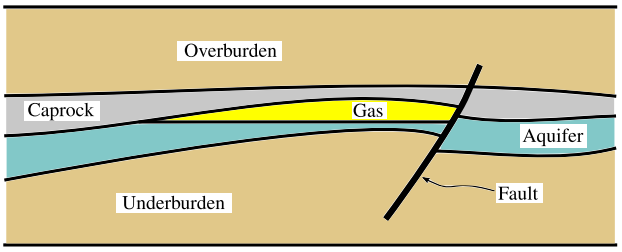

(c) Fault plane trace.

**Fig. 9.** Test 3b: Conceptual scheme, computational grid and fracture plane.

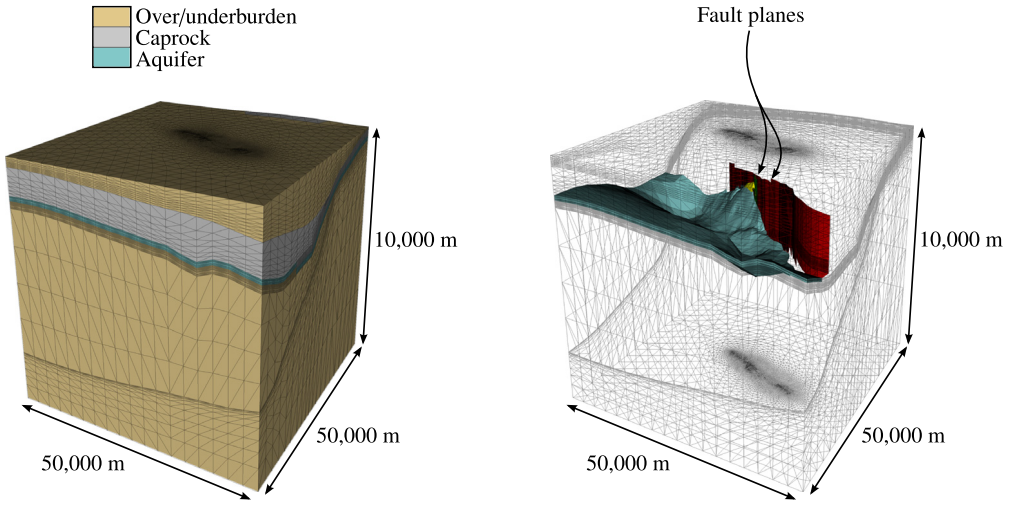
highlighted in Fig. 8c. A computational model is developed to simulate the generation and propagation of the ground rupture, using the formulation developed in Section 2. The domain extends for $2 \times 5$ km$^2$ in the horizontal plane and from the land surface down to 250 m depth in the vertical direction. Zero displacements are prescribed on the bottom and outer boundaries, while the top is traction-free. Further details on the model properties are provided in [55].

- Test 3b (Fig. 9): the development of ground fissures due to shallow aquifer exploitation is a well-known occurrence in Queretaro City, Mexico. The ruptures are typically generated by the differential compaction occurring in the soft fluvio-lacustrine sediments with respect to stiff rock outcrops bounding the pumped formations. The simplified configuration sketched in Fig. 9a, with a rock structure confining a shallow aquifer system, is used to understand the mechanics of this occurrence. The computational model covers a 30 km-side square area and extends vertically from the ground surface down to 50 m. Zero displacements on bottom and outer boundaries, along with a traction-free top surface are prescribed. The geomechanical parameters along with the full model details are provided in [55] and [23].
- Test 3c (Fig. 10): underground gas storage (UGS) in depleted hydrocarbon reservoirs is a strategic practice occurring in many sites in Europe and North America. Modeling the geomechanical effects of the cyclic loading due to summer injection and winter withdrawal is the object of several recent works, e.g., [56–59], and is of paramount importance for an optimal management and schedule of the storage activities. One of the main issues is the seismicity that has been recognized to be possibly induced or triggered by fault activation during the UGS operations [60–63]. Fig. 10 shows the geomechanical model of a faulted UGS site in Italy. The computational grid has a regional size with a $50 \times 50$ km$^2$ areal extent and extends down to 10-km depth. The porous medium is highly heterogeneous and a severely distorted mesh is necessary to reproduce the geometry of the geological structures of interest. Zero displacements are prescribed on the bottom and outer boundaries, with a traction-free top surface representing the ground. Similar models are discussed, for instance, in [59,64].

The size and number of non-zeros of the Jacobian matrices resulting in Test 3a, 3b and 3c are summarized in Table 7. The problems have quite different sizes, varying from about 70,000 to 1,200,000 DOFs, with $n_t$ smaller than 10% $n_u$. In these real-world examples, the portions of the faults that are activated during the simulation is pretty




(a) Sketch of a typical hydrocarbon faulted structural trap.




(b) Computational grid. (c) Fault plane traces confining the gas reservoir.

**Fig. 10.** Test 3c: Conceptual scheme, computational grid and fault plane traces.

**Table 7**
Test 3: Size and number of non-zeros of the test matrices.

| | $n_{\text{total}}$ | $n_u$ | $n_t$ | $n_t/n_u$ | $\texttt{nnz}(A)$ | $\texttt{nnz}(B_1)$ | $\texttt{nnz}(B_2)$ |
|---|---|---|---|---|---|---|---|
| Test 3a | 72,666 | 69,909 | 2,757 | 0.039 | 2,946,915 | 16,542 | 16,542 |
| Test 3b | 183,177 | 171,150 | 12,027 | 0.070 | 7,313,814 | 72,162 | 72,162 |
| Test 3c | 1,180,764 | 1,142,655 | 38,109 | 0.033 | 49,858,749 | 228,654 | 228,654 |

small, hence the contribution provided by matrix $F$ (Eq. (14c)) is usually negligible with respect to $C$. Therefore, for preconditioning purposes, we assume $B_1 = B_2^T = C$, thus obtaining a symmetric Schur complement approximation.

Table 8 provides the numerical performance of the block triangular preconditioners in Tests 3. For each combination of $\tilde{A}$ and $\tilde{S}$ the lowest total CPU time $T_t$ is reported, along with the corresponding iteration count $n_{\text{iter}}$ to reduce the 2-norm of the initial residual by a factor $\tau = 10^{-8}$. GMRES(100) is used as non-symmetric solver. For the sake of completeness, the number of iterations obtained by using a nested direct solver for applying $A^{-1}$ is also given.

As already mentioned before, the number of iterations to converge for $\tilde{A}^{-1} = A^{-1}$ can be regarded as the limiting $n_{\text{iter}}$ value when progressively increasing the quality of the (1,1) block approximation. Table 8 reveals that the use of a good local preconditioner for $A$ might yield convergence for any choice of $\tilde{S}$. The BD approximation, however, appears to be the less robust alternative, with no convergence achieved in less than 1000 iterations in Test 3c both with an FSAI and IC preconditioner for $A$. By distinction, the LSC approach provides the best Schur complement approximation and the most efficient outcome. If an FSAI preconditioner is used for the (1,1) block, as it can be desirable when working in a parallel computational environment, then the $\tilde{S}_{\texttt{FSAI}}$ approach is usually preferable. The exact application of $\tilde{S}_{\texttt{FSAI}}^{-1}$ is a convenient option whenever the ratio $n_t/n_u$ is small enough, i.e., $n_t/n_u < 0.05$ in the present test cases.

**Table 8**
Test 3: Lowest total CPU time for each combination of $\tilde{A}$ and $\tilde{S}$. The best performance for each row is highlighted. The acronym 'nc' ('not converged') means that convergence was not achieved after 1000 iterations.

| | $\tilde{A}^{-1}$ | $\tilde{S}_{\texttt{BD}}$ | | $\tilde{S}_{\texttt{LSC}}$ | | $\tilde{S}_{\texttt{FSAI-FSAI}}$ | | $\tilde{S}_{\texttt{FSAI-DIR}}$ | |
|---|---|---|---|---|---|---|---|---|---|
| | | $n_{\text{iter}}$ | $T_t$ [s] | $n_{\text{iter}}$ | $T_t$ [s] | $n_{\text{iter}}$ | $T_t$ [s] | $n_{\text{iter}}$ | $T_t$ [s] |
| Test 3a | $A^{-1}_{\texttt{FSAI}}$ | 371 | 5.5 | 77 | 2.6 | 73 | 2.0 | **72** | **1.7** |
| | $A^{-1}_{\texttt{IC}}$ | 189 | 3.0 | **24** | **0.8** | 34 | 1.8 | 34 | 1.6 |
| | $A^{-1}$ | 129 | – | 12 | – | 24 | – | 21 | – |
| Test 3b | $A^{-1}_{\texttt{FSAI}}$ | 255 | 13.0 | 175 | 12.3 | **126** | **9.1** | 123 | 9.6 |
| | $A^{-1}_{\texttt{IC}}$ | 240 | 12.2 | **59** | **6.8** | 97 | 10.1 | 98 | 10.3 |
| | $A^{-1}$ | 126 | – | 23 | – | 34 | – | 35 | – |
| Test 3c | $A^{-1}_{\texttt{FSAI}}$ | nc | *** | 768 | 287.1 | 623 | 172.7 | **496** | **156.5** |
| | $A^{-1}_{\texttt{IC}}$ | nc | *** | **125** | **73.2** | 195 | 128.3 | 194 | 116.1 |
| | $A^{-1}$ | 276 | – | 35 | – | 50 | – | 50 | – |

## 5. Discussion and conclusions

The accurate and efficient simulation of fault and fracture mechanics is a topic attracting a growing interest in the scientific community. In this work, we focussed on the formulation proposed in [23] based on the use of Lagrange multipliers. The Jacobian matrix $\mathcal{J}$, resulting from the Finite Element discretization of the governing equations (1a)–(1d) subject to the KKT complementary conditions (1e)–(1g) and (1h)–(1k) for normal and tangential contact, respectively, is characterized by a non-symmetric generalized saddle-point structure (Eq. (13)). A family of block triangular preconditioners was presented and discussed to accelerate the convergence of Krylov subspace methods.

The key ingredients for the preconditioner set-up and application are: (i) the selection of a local preconditioner $\tilde{A}^{-1}$ for the (1,1) block $A$, and (ii) the approximation $\tilde{S}$ of the Schur complement $S = -B_2 A^{-1} B_1$ and its inverse. Since for point (i) several well-established techniques are already available, we concentrated on point (ii) by introducing three different approaches for $\tilde{S}$:

1. the BD Schur complement $\tilde{S}_{\texttt{BD}}$ is built by exploiting physics-based considerations on the locality of stress and deformation along a fracture. The matrix $\tilde{S}_{\texttt{BD}}$ is block-diagonal, with at most $3 \times 3$ diagonal blocks associated to the supernodes detected on the fracture planes, and can be inverted exactly at a negligible cost;
2. the LSC Schur complement $\tilde{S}_{\texttt{LSC}}$ is derived as an extension of the PCD preconditioner originally introduced for Navier–Stokes equations. The advantage of this approach lies on the fact that $\tilde{S}^{-1}_{\texttt{LSC}}$ is known explicitly and can be directly applied by a sequence of matrix-by-vector products involving sparse and block-diagonal matrices;
3. the FSAI Schur complement $\tilde{S}_{\texttt{FSAI}}$ is built in a fully algebraic way by replacing $A^{-1}$ in the $S$ definition by an explicit sparse approximate inverse. Generally, the resulting matrix $\tilde{S}_{\texttt{FSAI}}$ has no specific structure, so other possibilities come into play for applying its inverse. This can be done either inexactly, e.g., by means of another FSAI approximation of $\tilde{S}^{-1}_{\texttt{FSAI}}$, or exactly, with the aid of a nested direct solver.

The theoretical properties and computational performance of these approaches were investigated in three sets of numerical examples, reproducing both synthetic and real-world test cases.

The analysis of the numerical results reveals that all the proposed approaches are effective for solving the target problem, provided that a sufficiently good preconditioner is used for $\tilde{A}^{-1}$. There is not a clear winner among BD, LSC and FSAI, with the most efficient choice being generally problem- and user-dependent. The main points arising from the theoretical and numerical evidences obtained in this work are as follows.

- The BD Schur complement is less robust than LSC and FSAI, requiring more iterations to converge and exceeding the limiting value of 1000 iterations in the largest test case. This should not be a surprise, as $\tilde{S}_{\texttt{BD}}$ is a very sparse approximation of $S$ that can be too poor whenever the assumption of the locality of deformation is far from being verified. Nevertheless, due to the ease of implementation and the very low memory demand, the BD approach can become interesting in problems where the ratio $n_t/n_u$ is relatively large, i.e., the size of $S$ is a significant fraction of $A$, such as applications characterized by multiple-fractured media.

- The LSC approach provides the best approximation of the Schur complement, requiring the lowest iteration count when $A^{-1}$ is applied exactly. This result is theoretically founded, because we proved that the eigenvalues of the preconditioned matrix are bounded from below by 1 whereas with the other approximations a similar bound is not available. This generally leads to the most efficient computational performance for LSC obtained with $\tilde{A}^{-1} = A_{\mathtt{IC}}^{-1}$, i.e., in sequential implementations.
- The FSAI approach requires the computation of an explicit approximate inverse of the (1,1) block for obtaining $\tilde{S}$, hence it is quite natural to use the same matrix as $\tilde{A}^{-1}$. It is well-known that sparse approximate inverses have generally a lower quality as preconditioners with respect to other techniques, such as for instance incomplete factorizations, and this can penalize the performance of the overall algorithm. Computing two different approximations of $A^{-1}$, i.e., one as $\tilde{A}^{-1}$ and one for building $\tilde{S}_{\mathtt{FSAI}}$, increases the set-up costs and so often is not convenient.
- All the different alternatives proposed for $\tilde{S}$ can be efficiently implemented in a parallel computational environment. Hence, the key factor for the overall preconditioner suitability to High Performance Computing (HPC) relies on the parallel degree of $\tilde{A}^{-1}$. An efficient HPC option is $A_{\mathtt{FSAI}}^{-1}$, which is known to show a nearly ideal parallel behavior [33]. In this context, the most natural and attractive option appears to be using the FSAI Schur complement, because it exploits $A_{\mathtt{FSAI}}^{-1}$ also in the computation of $\tilde{S}_{\mathtt{FSAI}}$. With this choice usually the FSAI approach exhibits a faster convergence than BD and LSC.
- The FSAI approach is also the most flexible one, as it gives the user the chance to select several other options: (i) fill-in parameters for $A_{\mathtt{FSAI}}^{-1}$, (ii) fill-in parameters for $G$ in the computation of $\tilde{S}_{\mathtt{FSAI}}$ (Eq. (31)), (iii) if the approximate Schur complement is applied inexactly, another local preconditioner for $\tilde{S}_{\mathtt{FSAI}}$ must be selected with its own fill-in parameters. It can be easily guessed that, depending on the choice for the preconditioner of $\tilde{S}$, the number of user-specified parameters can grow significantly.
- By distinction with the FSAI approach, the BD and LSC Schur complement allow the user to specify the choice of $\tilde{A}^{-1}$ only. This may result in a lower control of the overall preconditioner quality, but at the same time the final algorithm is much simpler to use and optimize also by a non-expert practitioner.

## Acknowledgments

Funding for this research was provided by the University of Padova project "Stable and efficient discretizations of the mechanics of faults" (to A.F. and M.F.). N.C. gratefully acknowledges the financial support provided by the Reservoir Simulation Industrial Affiliates Consortium at Stanford University (SUPRI-B) and Total S.A. through the Stanford Total Enhanced Modeling of Source rock (STEMS) project. Portions of this work were performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. The authors are grateful to Joshua White, and three anonymous reviewers for helpful comments.

## Appendix A. Finite element vectors and matrices

The vectors and matrices introduced in Section 2.3 are assembled in the standard way from element contributions. The global expressions for the residual block vectors (12) read:

$$
\begin{aligned}
\{\boldsymbol{r}_{\boldsymbol{u}}\}_i = &\int_{\Omega^h} \nabla^s \boldsymbol{N}_i^{\boldsymbol{u}} : \boldsymbol{\sigma}_n \,\mathrm{d}\Omega - \int_{\Omega^h} \boldsymbol{N}_i^{\boldsymbol{u}} \cdot \boldsymbol{b}_n \,\mathrm{d}\Omega - \int_{\Gamma_{\boldsymbol{\sigma}}^h} \boldsymbol{N}_i^{\boldsymbol{u}} \cdot \bar{\boldsymbol{t}}_n \,\mathrm{d}\Gamma \\
&+ \int_{\Gamma_f^{h,\mathrm{stick}} \cup \Gamma_f^{h,\mathrm{slip}}} [\![\boldsymbol{N}_i^{\boldsymbol{u}}]\!] \cdot \underbrace{(\boldsymbol{n}_f \otimes \boldsymbol{n}_f) \cdot \boldsymbol{t}_n^h}_{=t_{N,n}^h \boldsymbol{n}_f} \,\mathrm{d}\Gamma \\
&+ \int_{\Gamma_f^{h,\mathrm{stick}}} [\![\boldsymbol{N}_i^{\boldsymbol{u}}]\!] \cdot \underbrace{(\mathbf{1} - \boldsymbol{n}_f \otimes \boldsymbol{n}_f) \cdot \boldsymbol{t}_n^h}_{=\boldsymbol{t}_{T,n}^h} \,\mathrm{d}\Gamma \\
&+ \int_{\Gamma_f^{h,\mathrm{slip}}} [\![\boldsymbol{N}_i^{\boldsymbol{u}}]\!] \cdot \underbrace{(\mathbf{1} - \boldsymbol{n}_f \otimes \boldsymbol{n}_f) \cdot \boldsymbol{t}_n^h}_{=\boldsymbol{t}_{T,n}^*(\Delta \boldsymbol{g}_{T,n}^h, t_{N,n}^h)} \,\mathrm{d}\Gamma \qquad \forall i \in \{1, 2, \ldots, n_{\boldsymbol{u}}\},
\end{aligned}
\tag{A.1a}
$$

$$\{\boldsymbol{r}_t\}_i = \int_{\Gamma_f^{h,\text{stick}}\cup\Gamma_f^{h,\text{slip}}} \boldsymbol{N}_i^t \cdot \underbrace{(\boldsymbol{n}_f\otimes\boldsymbol{n}_f)\cdot[\![\boldsymbol{u}_n^h]\!]}_{=g_{N,n}^h\boldsymbol{n}_f}\,\mathrm{d}\Gamma + \int_{\Gamma_f^{h,\text{stick}}} \boldsymbol{N}_i^t \cdot \underbrace{(\mathbf{1}-\boldsymbol{n}_f\otimes\boldsymbol{n}_f)\cdot[\![\Delta\boldsymbol{u}^h]\!]}_{=\Delta g_{T,n}^h}\,\mathrm{d}\Gamma \qquad \forall i\in\{1,2,\dots,n_t\}. \tag{A.1b}$$

with $[\![\Delta\boldsymbol{u}^h]\!] = [\![\boldsymbol{u}_n^h]\!] - [\![\boldsymbol{u}_{n-1}^h]\!]$. Recalling the definitions (14a)–(14c), the global expressions for the sub-matrices appearing in the global Jacobian matrix (13) read:

$$[K]_{ij} = \int_{\Omega^h} \nabla^s\boldsymbol{N}_i^{\boldsymbol{u}} : \left(\frac{\partial\boldsymbol{\sigma}}{\partial\boldsymbol{\epsilon}}\bigg|_n^k\right) : \nabla^s\boldsymbol{N}_j^{\boldsymbol{u}}\,\mathrm{d}\Omega \qquad \forall(i,j)\in\{1,2,\dots,n_{\boldsymbol{u}}\}\times\{1,2,\dots,n_{\boldsymbol{u}}\}, \tag{A.2a}$$

$$[E]_{ij} = \int_{\Gamma_f^{h,\text{slip}}} [\![\boldsymbol{N}_i^{\boldsymbol{u}}]\!]\cdot\left(\frac{\partial\boldsymbol{t}_T^*}{\partial([\![\boldsymbol{u}]\!])}\bigg|_n^k\right)\cdot[\![\boldsymbol{N}_j^{\boldsymbol{u}}]\!]\,\mathrm{d}\Gamma \qquad \forall(i,j)\in\{1,2,\dots,n_{\boldsymbol{u}}\}\times\{1,2,\dots,n_{\boldsymbol{u}}\}, \tag{A.2b}$$

$$[C]_{ij} = \int_{\Gamma_f^{h,\text{stick}}\cup\Gamma^{h,\text{slip}}} [\![\boldsymbol{N}_i^{\boldsymbol{u}}]\!]\cdot(\boldsymbol{n}_f\otimes\boldsymbol{n}_f)\cdot\boldsymbol{N}_j^{\boldsymbol{t}}\,\mathrm{d}\Gamma + \int_{\Gamma_f^{h,\text{stick}}} [\![\boldsymbol{N}_i^{\boldsymbol{u}}]\!]\cdot(\mathbf{1}-\boldsymbol{n}_f\otimes\boldsymbol{n}_f)\cdot\boldsymbol{N}_j^{\boldsymbol{t}}\,\mathrm{d}\Gamma \qquad \forall(i,j)\in\{1,2,\dots,n_{\boldsymbol{u}}\}\times\{1,2,\dots,n_t\}, \tag{A.2c}$$

$$[F]_{ij} = -\int_{\Gamma_f^{h,\text{slip}}} [\![\boldsymbol{N}_i^{\boldsymbol{u}}]\!]\cdot\left(\frac{\partial\boldsymbol{t}_T^*}{\partial\boldsymbol{t}}\bigg|_n^k\right)\cdot\boldsymbol{N}_j^{\boldsymbol{t}}\,\mathrm{d}\Gamma \qquad \forall(i,j)\in\{1,2,\dots,n_{\boldsymbol{u}}\}\times\{1,2,\dots,n_t\}, \tag{A.2d}$$

with the partial derivatives expanded as:

$$\frac{\partial\boldsymbol{\sigma}}{\partial\boldsymbol{\epsilon}}\bigg|_n^k = \mathbf{C}, \tag{A.3a}$$

$$\frac{\partial\boldsymbol{t}_T^*}{\partial([\![\boldsymbol{u}]\!])}\bigg|_n^k = \frac{\partial\boldsymbol{t}_T^*}{\partial(\Delta\boldsymbol{g}_T)}\bigg|_n^k\frac{\partial(\Delta\boldsymbol{g}_T)}{\partial[\![\boldsymbol{u}]\!]}\bigg|_n^k = \tau_{\max}\frac{\|\Delta\boldsymbol{g}_T\|_2^2\mathbf{1}-\Delta\boldsymbol{g}_T\otimes\Delta\boldsymbol{g}_T}{\|\Delta\boldsymbol{g}_T\|_2^3}\bigg|_n^k\cdot(\mathbf{1}-\boldsymbol{n}_f\otimes\boldsymbol{n}_f), \tag{A.3b}$$

$$\frac{\partial\boldsymbol{t}_T^*}{\partial\boldsymbol{t}}\bigg|_n^k = -\tan\varphi\frac{\Delta\boldsymbol{g}_T}{\|\Delta\boldsymbol{g}_T\|_2}\otimes\boldsymbol{n}_f\bigg|_n^k. \tag{A.3c}$$

## Appendix B. Proof of Proposition 2.1

**Proof.** By definition [65], a matrix $A\in\mathbb{R}^{n\times n}$ is positive semidefinite if:

$$\boldsymbol{c}^T A\boldsymbol{c}\geq 0, \quad \forall\boldsymbol{c}\in\mathbb{R}^n, \quad \boldsymbol{c}\neq\mathbf{0}. \tag{B.1}$$

Let us introduce $\boldsymbol{v}^h = \sum_{i=1}^{n_{\boldsymbol{u}}}\boldsymbol{N}_i^{\boldsymbol{u}}c_i\in\boldsymbol{\mathcal{V}}^h$ and the vector $\boldsymbol{c}\in\mathbb{R}^{n_{\boldsymbol{u}}}$ containing the $c_i$ coefficients defining $\boldsymbol{v}^h$. For matrix $E(\boldsymbol{d},\boldsymbol{\lambda})$, (B.1) becomes:

$$\begin{aligned}
\boldsymbol{c}^T E\boldsymbol{c} &= \sum_{i=1}^{n_{\boldsymbol{u}}}\sum_{j=1}^{n_{\boldsymbol{u}}} c_i[E]_{ij}c_j \\
&= \sum_{i=1}^{n_{\boldsymbol{u}}}\sum_{j=1}^{n_{\boldsymbol{u}}} c_i\left(\int_{\Gamma_f^{h,\text{slip}}}[\![\boldsymbol{N}_i^{\boldsymbol{u}}]\!]\cdot\left(\frac{\partial\boldsymbol{t}_T^*}{\partial([\![\boldsymbol{u}]\!])}\bigg|_n^k\right)\cdot[\![\boldsymbol{N}_j^{\boldsymbol{u}}]\!]\,\mathrm{d}\Gamma\right)c_j \\
&= \int_{\Gamma_f^{h,\text{slip}}}\left(\sum_{i=1}^{n_{\boldsymbol{u}}}[\![\boldsymbol{N}_i^{\boldsymbol{u}}]\!]c_i\right)\cdot\left(\frac{\partial\boldsymbol{t}_T^*}{\partial([\![\boldsymbol{u}]\!])}\bigg|_n^k\right)\cdot\left(\sum_{j=1}^{n_{\boldsymbol{u}}}[\![\boldsymbol{N}_j^{\boldsymbol{u}}]\!]c_j\right)\mathrm{d}\Gamma \\
&= \int_{\Gamma_f^{h,\text{slip}}}[\![\boldsymbol{v}^h]\!]\cdot\left(\frac{\partial\boldsymbol{t}_T^*}{\partial([\![\boldsymbol{u}]\!])}\bigg|_n^k\right)\cdot[\![\boldsymbol{v}^h]\!]\,\mathrm{d}\Gamma\geq 0, \quad \forall\boldsymbol{v}^h\in\boldsymbol{\mathcal{V}}^h, \quad \boldsymbol{v}^h\neq\mathbf{0}.
\end{aligned} \tag{B.2}$$

To show that (B.2) holds true, it is sufficient to prove that $(\partial \boldsymbol{t}_T^*/\partial(\llbracket \boldsymbol{u} \rrbracket))\big|_n^k$ is symmetric and positive semi-definite. Let $\beta$ and $\boldsymbol{m}_T$ denote magnitude and direction of $\Delta \boldsymbol{g}_T$, i.e. $\beta = \|\Delta \boldsymbol{g}_T\|_2$ and $\boldsymbol{m}_T = \Delta \boldsymbol{g}_T / \|\Delta \boldsymbol{g}_T\|_2$. Since $\boldsymbol{m}_T \cdot \boldsymbol{n}_f = 0$, (A.3b) may be rewritten as:

$$\frac{\partial \boldsymbol{t}_T^*}{\partial(\llbracket \boldsymbol{u} \rrbracket)} = \frac{\tau_{\max}}{\beta}(\mathbf{1} - \boldsymbol{m}_T \otimes \boldsymbol{m}_T - \boldsymbol{n}_f \otimes \boldsymbol{n}_f), \tag{B.3}$$

where indexes $n$ and $k$ have been omitted. In (B.3) both $\tau_{\max}$ and $\beta$ are positive, while $(\mathbf{1} - \boldsymbol{m}_T \otimes \boldsymbol{m}_T - \boldsymbol{n}_f \otimes \boldsymbol{n}_f)$ represents the orthogonal projector onto the orthogonal complement of the subspace $\mathtt{span}(\boldsymbol{m}_T, \boldsymbol{n}_f)$, hence it is self-adjoint [65]—i.e., symmetric—and has two zero eigenvalues corresponding to the unit eigenvectors $\boldsymbol{m}_T$ and $\boldsymbol{n}_f$ and one unit eigenvalue [65]—i.e., positive semidefinite. This completes the proof. □

## Appendix C. Proof of Proposition 2.2

**Proof.** The first statement follows immediately from the definition of matrix $E$. Indeed, if we substitute (B.3) in (A.2b), any entry $[E]_{ij}$ reads

$$[E]_{ij} = \int_{\Gamma^{h,\mathrm{slip}}} \llbracket \boldsymbol{N}_i^{\boldsymbol{u}} \rrbracket \cdot \frac{\tau_{\max}}{\beta}(\mathbf{1} - \boldsymbol{m}_T \otimes \boldsymbol{m}_T - \boldsymbol{n}_f \otimes \boldsymbol{n}_f) \cdot \llbracket \boldsymbol{N}_j^{\boldsymbol{u}} \rrbracket \, \mathrm{d}\Gamma, \tag{C.1}$$

with indexes $n$ and $k$ again omitted. All terms in (C.1) are finite, with the exception of $\beta = \|\Delta \boldsymbol{g}_T\|_2 \to 0$, hence $\lim_{\beta \to 0^+} \|E\|_p = +\infty$, regardless of the behavior of $\boldsymbol{u}^h$.

To prove the second part of the proposition we first introduce two linear operators $\mathcal{R}$ and $\mathcal{I}$. The former maps $\boldsymbol{u}$ into $\Delta \boldsymbol{g}_T$, namely $\mathcal{R} : \mathbb{R}^3 \to \mathbb{R}^3$, $\boldsymbol{u} \mapsto \mathcal{R}(\boldsymbol{u}) = \Delta \boldsymbol{g}_T$. The latter is defined such that for any couple of vector $\boldsymbol{a}$ and $\boldsymbol{b}$ in $\mathbb{R}^3$ the relationship $\mathcal{R}(\boldsymbol{a}) \cdot \mathcal{R}(\boldsymbol{b}) = \boldsymbol{a} \cdot \mathcal{I}[\mathcal{R}(\boldsymbol{b})]$ holds true. Using such operators along with (A.3c), any entry $[F]_{ij}$ of (A.2d) becomes

$$\begin{aligned}
[F]_{ij} &= \int_{\Gamma^{h,\mathrm{slip}}} \llbracket \boldsymbol{N}_i^{\boldsymbol{u}} \rrbracket \cdot \frac{\Delta \boldsymbol{g}_T}{\|\Delta \boldsymbol{g}_T\|_2} \otimes \left(\tan\varphi \, \boldsymbol{n}_f\right) \cdot \boldsymbol{N}_j^{\boldsymbol{t}} \, \mathrm{d}\Gamma \\
&= \int_{\Gamma^{h,\mathrm{slip}}} \boldsymbol{N}_i^{\boldsymbol{u}} \cdot \frac{\mathcal{I}[\mathcal{R}(\boldsymbol{u})]}{\sqrt{\boldsymbol{u} \cdot \mathcal{I}[\mathcal{R}(\boldsymbol{u})]}} \otimes \left(\tan\varphi \, \boldsymbol{n}_f\right) \cdot \boldsymbol{N}_j^{\boldsymbol{t}} \, \mathrm{d}\Gamma \\
&= \int_{\Gamma^{h,\mathrm{slip}}} \boldsymbol{N}_i^{\boldsymbol{u}} \cdot \frac{\beta \boldsymbol{z}}{\sqrt{\beta}\sqrt{\boldsymbol{u} \cdot \boldsymbol{z}}} \otimes \left(\tan\varphi \, \boldsymbol{n}_f\right) \cdot \boldsymbol{N}_j^{\boldsymbol{t}} \, \mathrm{d}\Gamma,
\end{aligned} \tag{C.2}$$

with $\beta \boldsymbol{z} = \mathcal{I}[\mathcal{R}(\boldsymbol{u})]$. If $\beta$ tends to zero, then $[F]_{ij}$ vanishes, thus $\lim_{\beta \to 0^+} \|F\|_p = 0$. This completes the proof. □

## Appendix D. Proof of Theorem 3.1

**Proof.** The matrix $M = S\tilde{S}_{\mathrm{LSC}}^{-1}$ reads:

$$M = \left(B^T A^{-1} B\right)\left(B^T B\right)^{-1}\left(B^T A B\right)\left(B^T B\right)^{-1}. \tag{D.1}$$

Let us write the singular-value decomposition (SVD) of $B$:

$$B = V \Sigma W^T, \tag{D.2}$$

where $V \in \mathbb{R}^{n_{\boldsymbol{u}} \times n_{\boldsymbol{t}}}$ is such that $V^T V = I$, $\Sigma$ is diagonal with $n_{\boldsymbol{t}}$ entries, and $W \in \mathbb{R}^{n_{\boldsymbol{t}} \times n_{\boldsymbol{t}}}$ is orthogonal. The Moore–Penrose left pseudoinverse of $B$, $B^\dagger = (B^T B)^{-1} B^T$, is

$$B^\dagger = \left(B^T B\right)^{-1} B^T = \left(W \Sigma V^T V \Sigma W^T\right)^{-1} W \Sigma V^T = W \Sigma^{-2} W^T W \Sigma V^T = W \Sigma^{-1} V^T. \tag{D.3}$$

Introducing Eqs. (D.2) and (D.3) into (D.1) we obtain

$$M = W \Sigma V^T A^{-1} V V^T A V \Sigma^{-1} W^T = W \Sigma \widetilde{M} \Sigma^{-1} W^T, \tag{D.4}$$

hence $M$ is similar to $\widetilde{M}$. Using the QR factorization of $V$, with $Q \in \mathbb{R}^{n_{\boldsymbol{u}} \times n_{\boldsymbol{u}}}$ and $R \in \mathbb{R}^{n_{\boldsymbol{u}} \times n_{\boldsymbol{t}}}$ in the form:

$$R = \begin{bmatrix} I \\ 0 \end{bmatrix}, \qquad I \in \mathbb{R}^{n_{\boldsymbol{t}} \times n_{\boldsymbol{t}}}, \tag{D.5}$$

the matrix $\widetilde{M}$ reads:

$$\widetilde{M} = \left(R^T Q^T A Q R\right)\left(R^T Q^T A^{-1} Q R\right) = \left(R^T H R\right)\left(R^T H^{-1} R\right), \tag{D.6}$$

having denoted the projection $Q^T A Q$ by $H$. Let us split $H$ into four blocks, according to the dimensions $n_t$ and $(n_u - n_t)$:

$$H = \begin{bmatrix} H_{11} & H_{12} \\ H_{12}^T & H_{22} \end{bmatrix}, \qquad H_{11} \in \mathbb{R}^{n_t \times n_t},\ H_{22} \in \mathbb{R}^{(n_u - n_t)\times(n_u - n_t)}. \tag{D.7}$$

With the block structure (D.7), the inverse of $H$ is

$$H^{-1} = \begin{bmatrix} S_1^{-1} & -S_1^{-1} H_{12} H_{22}^{-1} \\ -H_{22}^{-1} H_{12}^T S_1^{-1} & S_2^{-1} \end{bmatrix}, \tag{D.8}$$

where $S_1 = H_{11} - H_{12} H_{22}^{-1} H_{12}^T$ and $S_2 = H_{22} - H_{12}^T H_{11}^{-1} H_{12}$. Through Eqs. (D.5), (D.7), and (D.8), Eq. (D.6) simply becomes:

$$\widetilde{M} = H_{11} S_1^{-1}. \tag{D.9}$$

Using the Woodbury identity, $S_1^{-1}$ reads:

$$\begin{aligned} S_1^{-1} &= \left(H_{11} - H_{12} H_{22}^{-1} H_{12}^T\right)^{-1} = H_{11}^{-1} + H_{11}^{-1} H_{12}\left(H_{22} - H_{12}^T H_{11}^{-1} H_{12}\right)^{-1} H_{12}^T H_{11}^{-1} \\ &= H_{11}^{-1} + H_{11}^{-1} H_{12} S_2^{-1} H_{12}^T H_{11}^{-1}, \end{aligned} \tag{D.10}$$

and the product (D.9) is

$$\widetilde{M} = I + H_{12} S_2^{-1} H_{12}^T H_{11}^{-1} = I + E, \tag{D.11}$$

where $E = H_{12} S_2^{-1} H_{12}^T H_{11}^{-1}$ is similar to:

$$\widetilde{E} = H_{11}^{-1/2} H_{12} S_2^{-1} H_{12}^T H_{11}^{-1/2} = P^T S_2^{-1} P, \tag{D.12}$$

having set $P = H_{12}^T H_{11}^{-1/2}$. Since $S_2$ is SPD, the eigenvalues of $\widetilde{E}$, hence $E$, are real and positive, thus completing the proof. □

## References


[1] O.C. Zienkiewicz, R.L. Taylor, The Finite Element Method: Solid Mechanics, Vol. 2, Butterworth-Heinemann, 2000.

[2] K.-J. Bathe, Finite Element Procedures, Klaus-Jurgen Bathe, 2006.

[3] M. Ferronato, C. Janna, G. Gambolati, Mixed constraint preconditioning in computational contact mechanics, Comput. Methods Appl. Mech. Engrg. 197 (45) (2008) 3922–3931, http://dx.doi.org/10.1016/j.cma.2008.03.008.

[4] M. Ferronato, C. Janna, G. Pini, Parallel solution to ill-conditioned FE geomechanical problems, Int. J. Numer. Anal. Methods Geomech. 36 (4) (2012) 422–437, http://dx.doi.org/10.1002/nag.1012.

[5] A.R. Khoei, R.W. Lewis, Adaptive finite element remeshing in a large deformation analysis of metal powder forming, Internat. J. Numer. Methods Engrg. 45 (7) (1999) 801–820, http://dx.doi.org/10.1002/(SICI)1097-0207(19990710)45:7<801::AID-NME604>3.0.CO;2-#.

[6] E. Onate, J. Rojek, Combination of discrete element and finite element methods for dynamic analysis of geomechanics problems, Comput. Methods Appl. Mech. Engrg. 193 (27) (2004) 3087–3128, http://dx.doi.org/10.1016/j.cma.2003.12.056.

[7] J. Rutqvist, C.-F. Tsang, Coupled hydromechanical effects of $CO_2$ injection, in: C.-F. Tsang, J.A. Apps (Eds.), Underground Injection Science and Technology, Elsevier, 2005, pp. 649–679, http://dx.doi.org/10.1016/S0167-5648(05)52050-1.

[8] J. Rutqvist, J. Birkholzer, F. Cappa, C.-F. Tsang, Estimating maximum sustainable injection pressure during geological sequestration of $CO_2$ using coupled fluid flow and geomechanical fault-slip analysis, Energy Conv. Manag. 48 (6) (2007) 1798–1807, http://dx.doi.org/10.1016/j.enconman.2007.01.021.

[9] M. Ferronato, G. Gambolati, C. Janna, P. Teatini, Numerical modelling of regional faults in land subsidence prediction above gas/oil reservoirs, Int. J. Numer. Anal. Methods Geomech. 32 (6) (2008) 633–657, http://dx.doi.org/10.1002/nag.640.

[10] A.P. Rinaldi, J. Rutqvist, Modeling of deep fracture zone opening and transient ground surface uplift at KB-502 $CO_2$ injection well, In Salah, Algeria, Int. J. Greenhouse Gas Control 12 (2013) 155–167, http://dx.doi.org/10.1016/j.ijggc.2012.10.017.

[11] E.-H. Benkhira, E.-H. Essoufi, R. Fakhar, On convergence of the penalty method for a static unilateral contact problem with nonlocal friction in electro-elasticity, European J. Appl. Math. 27 (1) (2016) 1–22, http://dx.doi.org/10.1017/S0956792515000248.

[12] H. Sabetamal, M. Nazem, S.W. Sloan, J.P. Carter, Frictionless contact formulation for dynamic analysis of nonlinear saturated porous media based on the mortar method, Int. J. Numer. Anal. Methods Geomech. 40 (1) (2016) 25–61, http://dx.doi.org/10.1002/nag.2386.

[13] E. Burman, A. Ern, A nonlinear consistent penalty method for positivity preservation in the finite element approximation of the transport equation, Comput. Methods Appl. Mech. Engrg. 320 (2017) 122–132, http://dx.doi.org/10.1016/j.cma.2017.03.019.

[14] R.R. Settgast, P. Fu, S.D.C. Walsh, J.A. White, C. Annavarapu, F.J. Ryerson, A fully coupled method for massively parallel simulation of hydraulically driven fractures in 3-dimensions, Int. J. Numer. Anal. Methods Geomech. 41 (5) (2017) 627–653, http://dx.doi.org/10.1002/nag.2557.
[15] N. Kikuchi, J.T. Oden, Contact Problems in Elasticity: A Study of Variational Inequalities and Finite Element Methods, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1988.
[16] J.C. Simo, T.J.R. Hughes, Computational Inelasticity, Springer-Verlag New York, 1998, http://dx.doi.org/10.1007/b98904.
[17] A.J. Wathen, Preconditioning, Acta Numer. 24 (2015) 329–376, http://dx.doi.org/10.1017/S0962492915000021.
[18] H. Elman, V.E. Howle, J. Shadid, R. Shuttleworth, R. Tuminaro, Block preconditioners based on approximate commutators, SIAM J. Sci. Comput. 27 (5) (2006) 1651–1668, http://dx.doi.org/10.1137/040608817.
[19] T. Laursen, Computational Contact and Impact Mechanics: Fundamentals of Modeling Interfacial Phenomena in Nonlinear Finite Element Analysis, Springer-Verlag Berlin Heidelberg, 2003.
[20] P. Wriggers, Computational Contact Mechanics, 2nd, Springer-Verlag Berlin Heidelberg, 2006, http://dx.doi.org/10.1007/978-3-540-32609-0.
[21] J.C. Simo, T.A. Laursen, An augmented Lagrangian treatment of contact problems involving friction, Comput. Struct. 42 (1) (1992) 97–116, http://dx.doi.org/10.1016/0045-7949(92)90540-G.
[22] P. Hansbo, Nitsche's method for interface problems in computa-tional mechanics, GAMM-Mitt. 28 (2) (2005) 183–206, http://dx.doi.org/10.1002/gamm.201490018.
[23] C. Annavarapu, M. Hautefeuille, J.E. Dolbow, A Nitsche stabilized finite element method for frictional sliding on embedded interfaces. Part I: single interface, Comput. Methods Appl. Mech. Engrg. 268 (2014) 417–436, http://dx.doi.org/10.1016/j.cma.2013.09.002.
[24] C. Annavarapu, M. Hautefeuille, J.E. Dolbow, A Nitsche stabilized finite element method for frictional sliding on embedded interfaces. Part II: intersecting interfaces, Comput. Methods Appl. Mech. Engrg. 267 (2013) 318–341, http://dx.doi.org/10.1016/j.cma.2013.08.008.
[25] A. Franceschini, M. Ferronato, C. Janna, P. Teatini, A novel Lagrangian approach for the stable numerical simulation of fault and fracture mechanics, J. Comput. Phys. 314 (2016) 503–521, http://dx.doi.org/10.1016/j.jcp.2016.03.032.
[26] M.F. Murphy, G.H. Golub, A.J. Wathen, A Note on Preconditioning for Indefinite Linear Systems, SIAM J. Sci. Comput. 21 (6) (2000) 1969–1972, http://dx.doi.org/10.1137/S1064827599355153.
[27] M. Benzi, G.H. Golub, J. Liesen, Numerical solution of saddle point problems, Acta Numer. 14 (2005) 1–137, http://dx.doi.org/10.1017/S0962492904000212.
[28] C. Hager, S. Hüeber, B.I. Wohlmuth, A stable energy-conserving approach for frictional contact problems based on quadrature formulas, Internat. J. Numer. Methods Engrg. 73 (2) (2008) 205–225, http://dx.doi.org/10.1002/nme.2069.
[29] A. Franceschini, Numerical models for the large-scale simualtion of fault and fracture mechanics (Phd thesis), University of Padova, Italy, 2018.
[30] A. Quarteroni, Numerical models for differential problems, in: MS&A, Springer Milan, 2014.
[31] Y. Saad, M.H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (3) (1986) 856–869, http://dx.doi.org/10.1137/0907058.
[32] H.A. Van der Vorst, Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 13 (2) (1992) 631–644, http://dx.doi.org/10.1137/0913035.
[33] C. Janna, M. Ferronato, F. Sartoretto, G. Gambolati, FSAIPACK: A software package for high-performance factored sparse approximate inverse preconditioning, ACM Trans. Math. Software 41 (2) (2015) 10, http://dx.doi.org/10.1145/2629475.
[34] Y. Bu, B. Carpentieri, Z. Shen, T. Huang, A hybrid recursive multilevel incomplete factorization preconditioner for solving general linear systems, Appl. Numer. Math. 104 (2016) 141–157, http://dx.doi.org/10.1016/j.apnum.2015.12.007.
[35] Y. Xi, R. Li, Y. Saad, An algebraic multilevel preconditioner with low-rank corrections for sparse symmetric matrices, SIAM J. Matrix Anal. Appl. 37 (2016) 235–259, http://dx.doi.org/10.1137/15M1021830.
[36] J. Xu, L. Zikatanov, Algebraic mulrigrid methods, Acta Numer. 26 (2017) 591–721, http://dx.doi.org/10.1017/S0962492917000083.
[37] A. Franceschini, V. Paludetto Magri, M. Ferronato, C. Janna, A robust multilevel approximate inverse preconditioner for symmetric positive definite matrices, SIAM J. Matrix Anal. Appl. 39 (1) (2018) 123–147, http://dx.doi.org/10.1137/16M1109503.
[38] B.T. Aagaard, M.G. Knepley, C.A. Williams, A domain decomposition approach to implementing fault slip in finite-element models of quasi-static and dynamic crustal deformation, J. Geophys. Res.-Solid Earth 118 (6) (2013) 3059–3079, http://dx.doi.org/10.1002/jgrb.50217.
[39] B. Jha, R. Juanes, Coupled multiphase flow and poromechanics: A computational model of pore pressure effects on fault slip and earthquake triggering, Water Resour. Res. 5 (2014) 3776–3808, http://dx.doi.org/10.1002/2013WR015175.
[40] N. Castelletto, J.A. White, M. Ferronato, Scalable algorithms for three-field mixed finite element coupled poromechanics, J. Comput. Phys. 327 (2016) 894–918, http://dx.doi.org/10.1016/j.jcp.2016.09.063.
[41] C. Janna, M. Ferronato, G. Gambolati, The use of supernodes in factored sparse approximate inverse preconditioning, SIAM J. Sci. Comput. 37 (1) (2015) C72–C94, http://dx.doi.org/10.1137/140956026.
[42] D. Silvester, H. Elman, D. Kay, A. Wathen, Efficient preconditioning of the linearized Navier–Stokes equations for incompressible flow, J. Comput. Appl. Math. 128 (1–2) (2001) 261–279, http://dx.doi.org/10.1016/S0377-0427(00)00515-X.
[43] L.Y. Kolotilina, A.Y. Yeremin, Factorized sparse approximate inverse preconditionings I. Theory, SIAM J. Matrix Anal. Appl. 14 (1) (1993) 45–58, http://dx.doi.org/10.1137/0614004.
[44] T. Huckle, Approximate sparsity patterns for the inverse of a matrix and preconditioning, Appl. Numer. Math. 30 (2–3) (1999) 291–303, http://dx.doi.org/10.1016/S0168-9274(98)00117-2.
[45] E. Chow, A priori sparsity patterns for parallel sparse approximate inverse preconditioners, SIAM J. Sci. Comput. 21 (5) (2000) 1804–1822, http://dx.doi.org/10.1137/S106482759833913X.

[46] T. Huckle, Factorized sparse approximate inverses for preconditioning, J. Supercomput. 25 (2) (2003) 109–117, http://dx.doi.org/10.1023/A:1023988426844.
[47] C. Janna, M. Ferronato, Adaptive pattern research for block FSAI preconditioning, SIAM J. Sci. Comput. 33 (6) (2011) 3357–3380, http://dx.doi.org/10.1137/100810368.
[48] P. Vaněk, M. Brezina, R. Tezaur, Two-grid method for linear elasticity on unstructured meshes, SIAM J. Sci. Comput. 21 (3) (1999) 900–923, http://dx.doi.org/10.1137/S1064827596297112.
[49] HSL, A collection of Fortran codes for large scale scientific computation. http://www.hsl.rl.ac.uk, 2013.
[50] C. Lin, C.C. Moré, Incomplete cholesky factorizations with limited memory, SIAM J. Sci. Comput. 21 (1) (1999) 24–45, http://dx.doi.org/10.1137/S1064827597327334.
[51] H. Si, TetGen, a Delaunay-Based Quality Tetrahedral Mesh Generator, ACM Trans. Math. Software 41 (2) (2015) 11:1–11:36, http://dx.doi.org/10.1145/2629697.
[52] O. Schenk, K. Gärtner, Pardiso, in: D. Padua (Ed.), Encyclopedia of Parallel Computing, Springer US, Boston, MA, 2011, pp. 1458–1464.
[53] X. Shi, Y. Xue, S. Ye, J. Wu, Y. Zhang, Y. Jun, Characterization of land subsidence induced by groundwater withdrawals in Su-Xi-Chang area, China, Environ. Geol. 52 (2007) 27–40, http://dx.doi.org/10.1007/s00254-006-0446-3.
[54] G. Wang, G. You, B. Shi, J. Yu, H. Li, K. Zong, Earth fissures triggered by groundwater withdrawal and coupled by geological structures in Jiangsu Province, China, Environ. Geol. 57 (2009) 1047–1054, http://dx.doi.org/10.1007/s00254-008-1390-1.
[55] A. Franceschini, P. Teatini, C. Janna, M. Ferronato, G. Gambolati, S. Ye, D. Carreon-Freyre, Modeling ground rupture due to groundwater withdrawal: applications to test cases in China and Mexico, in: Proceedings of the International Association of Hydrological Sciences, Vol. 362, 2015, pp. 63–68, http://dx.doi.org/10.5194/piahs-372-63-2015.
[56] P. Teatini, N. Castelletto, M. Ferronato, G. Gambolati, C. Janna, E. Cairo, D. Marzorati, D. Colombo, A. Ferretti, A. Bagliani, F. Bottazzi, Geomechanical response to seasonal gas storage in depleted reservoirs: a case study in the Po river basin, Italy, J. Geophys. Res.-Earth Surf. 116 (F02002) (2011), http://dx.doi.org/10.1029/2010JF001793.
[57] B. Jha, F. Bottazzi, R. Wojcik, M. Coccia, N. Bechor, D. McLaughlin, T. Herring, B.H. Hager, S. Mantica, R. Juanes, Reservoir characterization in an underground gas storage field using joint inversion of flow and geodetic data, Int. J. Numer. Anal. Methods Geomech. 39 (14) (2015) 1619–1638, http://dx.doi.org/10.1002/nag.2427.
[58] P. Fokker, B. Wassing, F. van Leijen, R. Hanssen, D. Nieuwland, Application of an ensemble smoother with multiple data assimilation to the Bergermeer gas field, using PS-InSAR, Geomech. Energy Environ. 5 (2016) 16–28, http://dx.doi.org/10.1016/j.gete.2015.11.003.
[59] C. Zoccarato, D. Baù, M. Ferronato, G. Gambolati, A. Alzraiee, P. Teatini, Data assimilation of surface displacements to improve geomechanical parameters of gas storage reservoirs, J. Geophys. Res.-Solid Earth 121 (2016) 1441–1461, http://dx.doi.org/10.1002/2015JB012090.
[60] E. Priolo, M. Romanelli, M. Plasencia Linares, M. Garbin, L. Peruzza, M. Romano, P. Marotta, P. Bernardi, L. Moratto, D. Zuliani, P. Fabris, Seismic monitoring of an underground natural gas storage facility: The Collalto seismic network, Seismol. Res. Lett. 86 (1) (2015) 109–123, http://dx.doi.org/10.1785/0220140087.
[61] B. Gaite, A. Ugalde, A. Villaseño, E. Blanc, Improving the location of induced earthquakes associated with an underground gas storage in the Gulf of Valencia (Spain), Phys. Earth Planet. Inter. 254 (2016) 46–59, http://dx.doi.org/10.1016/j.pepi.2016.03.006.
[62] R. Schultz, R. Wang, Y. Gu, K. Haug, G. Atkinson, A seismological overview of the induced earthquakes in the Duvernay play near Fox Creek, Alberta, J. Geophys. Res. Solid Earth 122 (2017) 492–505, http://dx.doi.org/10.1002/2016JB013570.
[63] J. van Wees, S. Osinga, K. van Thienen-Visser1, P. Fokker, Reservoir creep and induced seismicity: inferences from geomechanical modeling of gas depletion in the Groningen field, Geophys. J. Int. 212 (2018) 1487–1497, http://dx.doi.org/10.1093/gji/ggx452.
[64] C. Zoccarato, D. Baù, F. Bottazzi, M. Ferronato, G. Gambolati, S. Mantica, P. Teatini, On the importance of the heterogeneity assummption in the characterization of reservoir geomechanical properties, Geophys. J. Int. 207 (2016) 47–58, http://dx.doi.org/10.1093/gji/ggw259.
[65] Y. Saad, Iterative Methods for Sparse Linear Systems, second ed., Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2003.