Journal of Computational Physics 227 (2008) 9885–9897

Contents lists available at ScienceDirect

Journal of Computational Physics

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

Mixed Constraint Preconditioners for the iterative solution of FE coupled consolidation equations

Luca Bergamaschi, Massimiliano Ferronato *, Giuseppe Gambolati

Department of Mathematical Methods and Models for Scientific Applications, University of Padova, Via Trieste 63, 35121 Padova, Italy

ARTICLE INFO

Article history:
Received 12 September 2007
Received in revised form 31 July 2008
Accepted 4 August 2008
Available online 14 August 2008

Keywords:
Preconditioning
Saddle point
Krylov subspace methods
Coupled consolidation

ABSTRACT

The Finite Element (FE) integration of the coupled consolidation equations requires the solution of linear symmetric systems with an indefinite saddle point coefficient matrix. Because of ill-conditioning, the repeated solution in time of the FE equations may be a major computational issue requiring ad hoc preconditioning strategies to guarantee the efficient convergence of Krylov subspace methods. In the present paper a Mixed Constraint Preconditioner (MCP) is developed combining implicit and explicit approximations of the inverse of the structural sub-matrix, with the performance investigated in some representative examples. An upper bound of the eigenvalue distance from unity is theoretically provided in order to give practical indications on how to improve the preconditioner. The MCP is efficiently implemented into a Krylov subspace method with the performance obtained in 2D and 3D examples compared to that of Inexact Constraint Preconditioners and Least Square Logarithm scaled ILUT preconditioners. Two variants of MCP (T-MCP and D-MCP), developed with the aim at reducing the cost of the preconditioner application, are also tested. The results show that the MCP variants constitute a reliable and robust approach for the efficient solution of realistic coupled consolidation FE models, and especially so in severely ill-conditioned problems.

1. Introduction

The time-dependent displacements and fluid pore pressure in porous media are controlled by the consolidation theory. This was first mathematically described by Biot [1], who coupled the elastic equilibrium equations with a continuity or mass balance equation to be solved under appropriate boundary and initial flow and loading conditions.

The coupled consolidation equations are typically solved numerically using Finite Elements (FE) in space, thus giving rise to a system of first-order differential equations the solution to which is addressed by an appropriate time marching scheme. A major computational issue is the repeated solution in time of the resulting discretized indefinite equations, which can be generally written as

$$\mathcal{A}\boldsymbol{x} = \boldsymbol{b}, \text{ where } \mathcal{A} = \begin{bmatrix} K & B^{\mathrm{T}} \\ B & -C \end{bmatrix}. \tag{1}$$

Both the sub-matrices K and C are symmetric positive definite (SPD). Denoting with m the number of FE nodes, $C \in \mathbb{R}^{m\times m}$, $B \in \mathbb{R}^{m\times n}$ and $K \in \mathbb{R}^{n\times n}$, where n is equal to $2m$ or $3m$ according to the spatial dimension of the problem if the same interpolation is used for displacement and pressure variables.

* Corresponding author. Tel.: +39 049 8271332; fax: +39 049 8271333.
E-mail address: ferronat@dmsa.unipd.it (M. Ferronato).
URL: http://www.dmsa.unipd.it/~ferronat (M. Ferronato).

0021-9991/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2008.08.002

The use of iterative solvers is recommended in large size realistic consolidation models. Among them, projection (or conjugate gradient-like) methods based on Krylov subspaces for indefinite systems, such as BiCGStab (Bi-Conjugate Gradient Stabilized [2]), are attracting a growing interest on the grounds of their robustness and efficiency [3–8]. However, the small time integration steps typically required in the early phase of the analysis may yield a severe ill-conditioning [9], and the selection of an efficient preconditioning strategy turns out to be a key issue to guarantee and accelerate the convergence. Note on passing that popular symmetric Krylov solvers, such as MINRES, cannot be generally used for problem (1) because of the indefiniteness of the preconditioners.

Matrix $\mathcal{A}$ in (1) is a classical example of saddle point problem, which is encountered in other fields as well including constrained optimization, least squares and Navier–Stokes equations. The constraint preconditioners for Krylov solvers in the solution of saddle point problems have been studied by a number of authors [10–16]. In most of the above references the preconditioner is obtained from $\mathcal{A}$ with the (1,1) block K well approximated and replaced by its diagonal. In the coupled consolidation problem, however, K is not diagonally dominant and a better approximation is required to ensure convergence. Bergamaschi et al. [17] have developed both an Exact and an Inexact Constraint Preconditioner (ECP and ICP, respectively) with the explicit approximation of K^{-1} provided by the approximate inverse preconditioner AINV [18]. The ICP variant is suggested with the aim at avoiding the need for exactly solving an inner $m \times m$ linear system for each preconditioner application as is required by ECP. In the present paper a Mixed Constraint Preconditioner (MCP) is developed where an implicit and an explicit approximation of K^{-1} are provided by an incomplete Cholesky decomposition ILLT and AINV, respectively. Using the spectral analysis it is shown that most of the eigenvalues of the preconditioned matrix are real positive and, most importantly, clustered around unity, with the value of the few remaining ones carefully kept under control. Two variants of MCP are then considered, based on the block structure of the preconditioner. The former, called Triangular MCP (T-MCP), uses an upper block triangular approximation of MCP, while the latter, denoted as Diagonal MCP (D-MCP) uses the block diagonal part of MCP only.

The paper is organized as follows. After a brief review of FE coupled consolidation equations, ECP and ICP with their main properties are revisited. In particular, a theoretical bound is given for the ICP eigenspectrum which helps give some practical indications as to the implementation of an effective preconditioner. Then, MCP is developed on the basis of the previous theoretical findings and experimented with in realistic medium and large size 2D and 3D problems. The MCP performance is compared to that of more traditional preconditioning techniques, such as ILUT with optimal fill-in degree [19] and a preliminary Least Square Logarithm (LSL) scaling [6], and that of ICP. The possible use of the T-MCP and D-MCP variants is finally discussed with a few remarks closing the paper.

2. Finite element coupled consolidation equations

The system of partial differential equations governing the 3D coupled consolidation process in fully saturated porous media is derived from the classical Biot's formulation [1] and successive modifications as:

$$(\lambda+\mu)\frac{\partial \epsilon}{\partial i}+\mu\nabla^2 u_i=\alpha\frac{\partial p}{\partial i},\quad i=x,\ y,\ z, \tag{2}$$

$$\frac{1}{\gamma}\nabla(k\nabla p)=[\phi\beta+c_{br}(\alpha-\phi)]\frac{\partial p}{\partial t}+\alpha\frac{\partial \epsilon}{\partial t}, \tag{3}$$

where c_{br} and β are the volumetric compressibility of solid grains and water, respectively, ϕ is the porosity, k the medium hydraulic conductivity, ϵ the medium volumetric dilatation, α the Biot coefficient, λ and μ are the Lamé constant and the shear modulus of the porous medium, respectively, γ is the specific weight of water, ∇ the gradient operator, x, y, z are the coordinate directions, t is time, and p and u_i are the incremental pore pressure and the components of incremental displacement along the i-direction, respectively.

Use of FE in space yields a system of first order differential equations which can be integrated by the Crank–Nicolson scheme [9]. The resulting linear system has to be repeatedly solved to obtain the transient displacements and pore pressures. The unsymmetric matrix controlling the solution scheme reads:

$$A=\begin{bmatrix} K/2 & -Q/2 \\ \frac{Q^{\mathrm{T}}}{\Delta t} & H/2+\frac{P}{\Delta t}\end{bmatrix}, \tag{4}$$

where K, H, P and Q are the elastic stiffness, flow stiffness, flow capacity and flow-stress coupling matrices, respectively. Matrix A can be readily symmetrized by multiplying the upper set of equations by 2 and the lower set by $-\Delta t$, thus obtaining the sparse 2×2 block symmetric indefinite matrix (1) where $B=-Q^{\mathrm{T}}$ and $C=\Delta t H/2+P$.

A major difficulty in the repeated solution to system (1) is the likely ill-conditioning of $\mathcal{A}$ caused by the large difference in magnitude between the coefficients of blocks K, B and C. The generic (i,j) element of each matrix is related to the hydro-mechanical properties of the porous medium as follows [9]:

$$K_{ij}\propto E, \tag{5}$$

$$B_{ij}\propto\sqrt{V}, \tag{6}$$

$$C_{ij}\propto\Delta t\frac{k}{\gamma}+\phi\beta V, \tag{7}$$

where E is the Young modulus of the porous medium and V a characteristic elemental size of the FE grid. The symbol $\propto$, meaning proportional to, aims at indicating the basic parameters controlling the K_{ij}, B_{ij} and C_{ij} size, with the unknown proportionality constants in (5)–(7) depending on the FE grid size and distortion. Being C_{ij} related to the time integration step Δt, the ill-conditioning of $\mathcal{A}$ is basically dependent on the Δt size. Ferronato et al. [9] have shown that a critical time step Δt_{crit} exists that can be defined as:

$$\Delta t_{\text{crit}} = \chi(\psi)\frac{V\gamma}{kE}, \tag{8}$$

where $\psi = \phi\beta E$ and χ is a generally unknown dimensionless factor depending on ψ and the element distortion. For $\Delta t \leqslant \Delta t_{\text{crit}}$ the conditioning of $\mathcal{A}$ suddenly degrades with the solution to (1) difficult to get independently of the solver choice. In long-term simulations a small Δt is typically needed in the early stage of the consolidation process, while larger values may be used as the system approaches the steady state. Hence, the initial steps are the most critical ones, with the convergence expected to improve as the simulation proceeds.

3. Exact Constraint Preconditioner

To solve system (1) we elect to use a Krylov method accelerated with the preconditioner $\mathcal{M}^{-1}$ where

$$\mathcal{M} = \begin{bmatrix} G & B^{\mathrm{T}} \\ B & -C \end{bmatrix} \tag{9}$$

and G is a SPD substitute for K. An exhaustive eigenanalysis of the preconditioned matrix can be found for instance in [20]. We only mention here some important results concerning the eigenvalue distribution of $\mathcal{M}^{-1}\mathcal{A}$.

Let us denote with α_K and β_K the smallest and the largest eigenvalue, respectively, of $G^{-1}K$. The eigenvalues of the preconditioned matrix $\mathcal{M}^{-1}\mathcal{A}$ depend on the quality of G^{-1} as an approximation of K^{-1}, as is stated by the following theorem:

Theorem 3.1. *If $\alpha_K < 1 < \beta_K$ then the eigenvalues λ of $\mathcal{M}^{-1}\mathcal{A}$ are either one (with multiplicity at least m) or real positive and bounded by $\alpha_K \leqslant \lambda \leqslant \beta_K$.*

Proof. The thesis follows from the statement of Theorem 1 in [21]. □

If G^{-1} is a preconditioner for K, the hypothesis of Theorem 3.1 ($\alpha_K < 1 < \beta_K$) is very common in practice. In particular, if $G^{-1} = \text{diag}(K)^{-1}$ the above hypothesis can be directly verified [22]. Moreover, as proved e.g. in [21], the classical Preconditioned Conjugate Gradient (PCG) algorithm is theoretically expected to converge with the indefinite matrix (1), provided that the last m components of the initial residual $\boldsymbol{r}_0$ are zero. This holds if the initial guess $\boldsymbol{x}_0$ is set to $\mathcal{M}^{-1}\boldsymbol{b}$.

The application of $\mathcal{M}^{-1}$ in a Krylov method, such as PCG, requires at each iteration the computation of $\boldsymbol{y} = \mathcal{M}^{-1}\boldsymbol{r}$, i.e. the solution to:

$$\begin{bmatrix} G & B^{\mathrm{T}} \\ B & -C \end{bmatrix} \begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{r}_1 \\ \boldsymbol{r}_2 \end{bmatrix}. \tag{10}$$

Vector $\boldsymbol{y}$ can be computed by solving for $\boldsymbol{y}_1$ in the upper set of equations (10):

$$\boldsymbol{y}_1 = G^{-1}(\boldsymbol{r}_1 - B^{\mathrm{T}}\boldsymbol{y}_2) \tag{11}$$

and substituting equation (11) in the lower set:

$$(BG^{-1}B^{\mathrm{T}} + C)\boldsymbol{y}_2 = BG^{-1}\boldsymbol{r}_1 - \boldsymbol{r}_2. \tag{12}$$

Matrix $S = BG^{-1}B^{\mathrm{T}} + C$ is the Schur complement of system (10). Hence, the cost of applying $\mathcal{M}^{-1}$ basically rests on the efficient solution to the linear $m \times m$ SPD system (12). This task can be accomplished, for example, by using PCG preconditioned with the incomplete Cholesky decomposition of S with no fill-in (IC(0)), thus defining an inner iteration cycle.

The Schur complement S can be computed only if G^{-1} is known explicitly. To fulfill such a requirement we may use the approximate inverse AINV [23,24] which is readily available in the factorized form:

$$K^{-1} \simeq G^{-1} = ZZ^{\mathrm{T}}, \tag{13}$$

where Z is upper triangular. In Eq. (13) and those following, the symbol $\simeq$ is used to indicate that the right-hand side is a generally dropped approximation of the left-hand side. A simpler choice for G^{-1}, such as $\text{diag}(K)^{-1}$, has proved ineffective in realistic consolidation problems [17]. The preconditioner $\mathcal{M}^{-1}$ with G^{-1} provided by (13) will be referred to as Exact Constraint Preconditioner (ECP).

4. Inexact Constraint Preconditioner

Although a very accurate solution of (10) is not really needed, solving the inner system (12) at each application of $\mathcal{M}^{-1}$ can represent a significant burden for the overall PCG scheme. To make the $\mathcal{M}^{-1}$ application cheaper, we can solve instead a substitute for (12) using the IC(0) factorization of S:

$$S \simeq \widetilde{S} = L_S L_S^{\mathrm{T}}. \tag{14}$$

In this way the solution to system (12) is replaced by a cost effective forward and backward substitution:

$$L_S L_S^{\mathrm{T}} \boldsymbol{y}_2 = BG^{-1}\boldsymbol{r}_1 - \boldsymbol{r}_2. \tag{15}$$

Obviously Theorem 3.1 no longer holds for the new resulting preconditioner $\widehat{\mathcal{M}}^{-1}$, and therefore the PCG convergence is no longer theoretically guaranteed. Using G^{-1} as in (13), $\widehat{\mathcal{M}}^{-1}$ can be factorized as follows:

$$\begin{aligned}\widehat{\mathcal{M}}^{-1} &= \begin{bmatrix} I_n & -G^{-1}B^{\mathrm{T}} \\ 0 & I_m \end{bmatrix} \begin{bmatrix} G^{-1} & 0 \\ 0 & -(L_S L_S^{\mathrm{T}})^{-1} \end{bmatrix} \begin{bmatrix} I_n & 0 \\ -BG^{-1} & I_m \end{bmatrix} \\ &= \begin{bmatrix} Z & -ZZ^{\mathrm{T}}B^{\mathrm{T}}L_S^{-\mathrm{T}} \\ 0 & L_S^{-\mathrm{T}} \end{bmatrix} \begin{bmatrix} Z^{\mathrm{T}} & 0 \\ L_S^{-1}BZZ^{\mathrm{T}} & -L_S^{-1} \end{bmatrix} \end{aligned} \tag{16}$$

$$= \mathcal{U}\mathcal{L}, \tag{17}$$

where $\mathcal{U}$ and $\mathcal{L}$ are upper and lower triangular matrices, respectively, and I_i is the $i \times i$ identity matrix. The factorized form (17) of $\widehat{\mathcal{M}}^{-1}$ is very well suited to an efficient implementation. The preconditioner $\widehat{\mathcal{M}}^{-1}$ will be referred to as Inexact Constraint Preconditioner (ICP) and is to be used in combination with a nonsymmetric Krylov solver such as Bi-CGStab [2] or QMR [25].

By distinction with $\mathcal{M}^{-1}\mathcal{A}$, the preconditioned matrix $\widehat{\mathcal{M}}^{-1}\mathcal{A}$ may possess complex eigenvalues. Let us define the matrices:

$$E_K = I_n - Z^{\mathrm{T}}KZ, \quad E_S = I_m - L_S^{-1}SL_S^{-\mathrm{T}}, \tag{18}$$

which provide a measure of the quality of the K^{-1} and S^{-1} approximations, respectively. Denoting by $\|\cdot\|$ a generic matrix norm (e.g. the spectral norm), the following theorem provides a bound for the distance of the eigenvalues of $\widehat{\mathcal{M}}^{-1}\mathcal{A}$ from unity.

Theorem 4.1. *Let $R = L_S^{-1}BZ$. Then any eigenvalue λ of $\widehat{\mathcal{M}}^{-1}\mathcal{A}$ satisfies the inequality: $|\lambda - 1| \leqslant (1 + \|R\|)^2\|E_K\| + \|E_S\|$.*

Proof. Recalling (16), $\widehat{\mathcal{M}}^{-1}$ can also be written as:

$$\widehat{\mathcal{M}}^{-1} = \mathcal{L}^{\mathrm{T}}J\mathcal{L} = \begin{bmatrix} Z & ZR^{\mathrm{T}} \\ 0 & -L_S^{-\mathrm{T}} \end{bmatrix} \begin{bmatrix} I_n & 0 \\ 0 & -I_m \end{bmatrix} \begin{bmatrix} Z^{\mathrm{T}} & 0 \\ RZ^{\mathrm{T}} & -L_S^{-1} \end{bmatrix}.$$

It is directly proved that the generalized eigenproblem $\mathcal{A}\boldsymbol{z} = \lambda\widehat{\mathcal{M}}\boldsymbol{z}$ is equivalent to $\mathcal{L}\mathcal{A}\mathcal{L}^{\mathrm{T}}\boldsymbol{w} = \lambda J\boldsymbol{w}$, with $\boldsymbol{w} = \mathcal{L}^{-\mathrm{T}}\boldsymbol{z}$, i.e.:

$$\begin{bmatrix} Z^{\mathrm{T}}KZ & -E_K R^{\mathrm{T}} \\ -RE_K & -RE_K R^{\mathrm{T}} - L_S^{-1}SL_S^{-\mathrm{T}} \end{bmatrix} \begin{pmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{pmatrix} = \lambda \begin{bmatrix} I_n & 0 \\ 0 & -I_m \end{bmatrix} \begin{pmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{pmatrix} \tag{19}$$

or equivalently:

$$\begin{bmatrix} -E_K & -E_K R^{\mathrm{T}} \\ RE_K & RE_K R^{\mathrm{T}} \end{bmatrix} \begin{pmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{pmatrix} - \begin{bmatrix} 0 & 0 \\ 0 & E_S \end{bmatrix} \begin{pmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{pmatrix} = (\lambda - 1) \begin{pmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{pmatrix}. \tag{20}$$

Then taking norms leads to:

$$|\lambda - 1| \leqslant (1 + \|R\|)^2\|E_K\| + \|E_S\|. \qquad \square \tag{21}$$

Some authors as well [12,26] have provided other more refined bounds in similar analyses. However, Theorem 4.1 may prove useful for giving indications on the most appropriate practical selection of G, as will be shown later.

The following theorems give additional information on the eigenvalues of $\widehat{\mathcal{M}}^{-1}\mathcal{A}$. In particular, we prove that at least $n - m$ eigenvalues are real positive and bounded by those of $G^{-1}K$, others are 1 and the imaginary part $\Im(\lambda)$ of the complex eigenvalues has a more restrictive upper bound than the one given in (21):

Theorem 4.2. *The preconditioned matrix $\widehat{\mathcal{M}}^{-1}\mathcal{A}$ has at least $n - m$ real eigenvalues λ bounded by $\alpha_K \leqslant \lambda \leqslant \beta_K$.*

Proof. Recalling equation (16), the eigenvalues λ of $\widehat{\mathcal{M}}^{-1}\mathcal{A}$ must satisfy the following generalized eigenvalue problem:

$$\begin{bmatrix} K & B^{\mathrm{T}} \\ B & -C \end{bmatrix} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} = \lambda \begin{bmatrix} I_n & 0 \\ BG^{-1} & I_m \end{bmatrix} \begin{bmatrix} G & 0 \\ 0 & -\widetilde{S} \end{bmatrix} \begin{bmatrix} I_n & G^{-1}B^{\mathrm{T}} \\ 0 & I_m \end{bmatrix} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix}. \tag{22}$$

Performing the product in the right hand side of (22) yields:

$$\widehat{\mathcal{M}} = \begin{bmatrix} G & B^{\mathrm{T}} \\ B & -C + S - \widetilde{S} \end{bmatrix} = \begin{bmatrix} G & B^{\mathrm{T}} \\ B & -\widehat{C} \end{bmatrix}$$

with $\widehat{C} = C - S + \widetilde{S}$. Hence the generalized eigenvalue problem (22) reads:

$$\begin{bmatrix} K & B^{\mathrm{T}} \\ B & -C \end{bmatrix} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} = \lambda \begin{bmatrix} G & B^{\mathrm{T}} \\ B & -\widehat{C} \end{bmatrix} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix}. \tag{23}$$

There are at least $n - m$ linearly independent eigenvectors satisfying the relationship:

$$\begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} = \begin{pmatrix} \mathcal{W}\boldsymbol{u} \\ 0 \end{pmatrix}$$

with $\mathcal{W}$ the $n \times (n - m)$ matrix whose columns form a basis for the null space of B. With these eigenvectors the generalized problem (23) reduces to:

$$K\boldsymbol{x} = \lambda G\boldsymbol{x}.$$

Therefore, at least $n - m$ eigenvalues of $\widehat{\mathcal{M}}^{-1}\mathcal{A}$ are bounded by $\alpha_K \leqslant \lambda \leqslant \beta_K$. □

Theorem 4.3. *If $BZE_KZ^TB^T + S$ is positive semidefinite then*

$$|\Im(\lambda)| \leqslant \|R\| \cdot \|E_K\|. \tag{24}$$

Proof. The proof follows from the application of Proposition 2.12 in [26] to the matrix pencil (19). □

Theorems 4.1 and 4.3 provide a bound that depends on $\|R\|$, i.e. ultimately on the norms of $\widetilde{S}$, B and K. Recalling equations (5)–(7), it can be argued that $\|\widetilde{S}\|$ is related to the hydro-geomechanical properties of the porous medium as follows:

$$\|\widetilde{S}\| \approx \|S\| \leqslant \|B\|^2\|K\|^{-1} + \|C\| \propto \frac{V}{E} + \Delta t\frac{k}{\gamma} + \phi\beta V. \tag{25}$$

In real porous media, the value of $1/E$ is typically between 2 and 3 orders of magnitude larger than β with ϕ seldom exceeding 0.4, and several orders of magnitude larger than k/γ, so for small Δt the norm of $\widetilde{S}$ is practically controlled by V/E. In essence, the magnitude of $\|R\|$ can be roughly estimated as is shown below:

$$\|R\| \leqslant \|L_S\|^{-1}\|B\|\|Z\| \approx \|\widetilde{S}\|^{-1/2}\|B\|\|K\|^{-1/2} \propto \sqrt{\frac{E}{V}} \cdot \sqrt{V} \cdot \sqrt{\frac{1}{E}} = 1. \tag{26}$$

Therefore, it might be concluded that for a small Δt, i.e. in the most ill-conditioned situations, the norm of R is of the order of 1 irrespective of the actual hydro-geomechanical properties of the porous medium. It turns out from (21) that $\|E_K\|$ roughly weighs four times more than $\|E_S\|$, hence the ICP computational performance appears to be in the first place connected with the quality of G^{-1}. By contrast, as $\Delta t \to \infty$ the norm of R becomes increasingly small, hence the errors $\|E_K\|$ and $\|E_S\|$ play a similar role in controlling the eigenvalue distribution of the overall preconditioned matrix. This seems to indicate that a relatively larger effort should be placed on the selection of a better preconditioner for K rather than for S, and particularly so for small to moderate values of the time integration step.

5. Mixed Constraint Preconditioner

A major drawback of the constraint preconditioning as was previously implemented is the somewhat "poor" AINV approximation of K^{-1} which may require a large number of iterations for Bi-CGStab preconditioned with ICP to converge [17]. However, this is only partially connected with the quality of AINV itself. Rather, it is the need for the explicit construction of the Schur complement matrix that calls for a reduced fill-in of Z and hence indirectly prevents a small $\|E_K\|$ value. We can try to remove this inconvenience by looking for an "implicit" approximation of K^{-1}, as is shown below. A most natural choice is the incomplete Cholesky factorization of K with variable fill-in:

$$K^{-1} \simeq G^{-1} = (L_KL_K^{\mathrm{T}})^{-1}. \tag{27}$$

Let us consider the application of ECP using Eq. (27). The fill-in of L_K can be much increased, thus allowing for a faster convergence from Theorem 3.1. Eq. (10) now reads:

$$\begin{bmatrix} L_KL_K^{\mathrm{T}} & B^{\mathrm{T}} \\ B & -C \end{bmatrix} \begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{r}_1 \\ \boldsymbol{r}_2 \end{bmatrix}. \tag{28}$$

Solving for $\boldsymbol{y}_1$ in the upper set of Eq. (28) yields:

$$\boldsymbol{y}_1 = (L_KL_K^{\mathrm{T}})^{-1}(\boldsymbol{r}_1 - B^{\mathrm{T}}\boldsymbol{y}_2) \tag{29}$$

and substituting Eq. (29) in the lower set gives:

$$[B(L_KL_K^{\mathrm{T}})^{-1}B^{\mathrm{T}} + C]\boldsymbol{y}_2 = B(L_KL_K^{\mathrm{T}})^{-1}\boldsymbol{r}_1 - \boldsymbol{r}_2. \tag{30}$$

The coefficient matrix in (30) is a new Schur complement:

$$S = B(L_K L_K^{\mathrm{T}})^{-1} B^{\mathrm{T}} + C. \tag{31}$$

Unfortunately, the SPD system (30) cannot be easily solved as matrix S is not known explicitly. Although the S implicit form (31) can be used to perform a product between S and a vector, and consequently one may think of solving (30) by a PCG method, a suitable preconditioner for S is not easily available. To avoid again the solution to (30) while retaining the convenient factorization (27) for G^{-1}, we can build explicitly an approximation of S using the AINV of K^{-1}, namely:

$$S = BZZ^{\mathrm{T}}B^{\mathrm{T}} + C \tag{32}$$

and then performing an incomplete Cholesky factorization:

$$S \simeq \widetilde{S} = L_S L_S^{\mathrm{T}}.$$

In this way we expect $\|E_K\|$ to be properly reduced with respect to the ICP implementation, but introduce an additional approximation in the Schur complement due to the use of Eq. (32) instead of (31), thus leading to a possibly larger $\|E_S\|$. As Theorem 4.1 still holds with $R = L_S^{-1} B L_K^{-\mathrm{T}}$ and the estimate (26) in (21) indicates that the distance of the eigenvalues of the preconditioned matrix from 1 depends mildly on $\|E_S\|$, we expect the above G^{-1} improvement to make up for the worsening of S yielding on the overall a better preconditioner. This ICP variant blending two different approximations for K^{-1} in the same scheme is called Mixed Constraint Preconditioner (MCP).

5.1. MCP application

Similarly to ECP and ICP [17], the MCP application requires first the explicit calculation of the matrix $S = BZZ^{\mathrm{T}}B^{\mathrm{T}} + C$ and then its incomplete triangular factor. Forming S may be time and memory consuming being the result of two sparse matrix–matrix products and one sparse sum of matrices. However, it may be noted that the evaluation of $S_0 = BZZ^{\mathrm{T}}B^{\mathrm{T}}$, which involves the main computational burden of S, is independent of the time step Δt, and therefore can be done just once at the beginning of the simulation. The computation of S_0 will be referred to as "preprocessing" in the sequel. Moreover, since matrix S can be much less sparse than C, its storage efficiency may be properly increased by dropping the terms below a user-specified tolerance τ_S relative to the diagonal entry.

Recalling Eq. (16), the MCP can be written as:

$$\begin{aligned}
\widetilde{\mathcal{M}}^{-1} &= \begin{bmatrix} I_n & -L_K^{-\mathrm{T}} L_K^{-1} B^{\mathrm{T}} \\ 0 & I_m \end{bmatrix} \begin{bmatrix} (L_K L_K^{\mathrm{T}})^{-1} & 0 \\ 0 & -(L_S L_S^{\mathrm{T}})^{-1} \end{bmatrix} \begin{bmatrix} I_n & 0 \\ -B L_K^{-\mathrm{T}} L_K^{-1} & I_m \end{bmatrix} \\
&= \begin{bmatrix} L_K^{-\mathrm{T}} & -L_K^{-\mathrm{T}} L_K^{-1} B^{\mathrm{T}} L_S^{-\mathrm{T}} \\ 0 & L_S^{-\mathrm{T}} \end{bmatrix} \begin{bmatrix} L_K^{-1} & 0 \\ L_S^{-1} B L_K^{-\mathrm{T}} L_K^{-1} & -L_S^{-1} \end{bmatrix} \\
&= \widetilde{\mathcal{U}}\,\widetilde{\mathcal{L}}.
\end{aligned} \tag{33}$$

With the factorized form (33), the "split" preconditioning technique can be conveniently implemented within the classical Bi-CGStab algorithm. The computation of the product $\widetilde{\mathcal{L}}\mathcal{A}\widetilde{\mathcal{U}}\boldsymbol{h}$, with $\boldsymbol{h}$ a generic vector, is accomplished by the algorithm of Table 1, where $\boldsymbol{v} = \widetilde{\mathcal{U}}\boldsymbol{h}$, $\boldsymbol{z} = \mathcal{A}\boldsymbol{v}$ and $\boldsymbol{t} = \widetilde{\mathcal{L}}\boldsymbol{z}$. Note that the algorithm in Table 1, involving forward and backward substitutions, is only slightly more expensive than the one with ICP which requires the product of the triangular matrices Z and Z^{T} by a vector.

5.2. Triangular and diagonal MCP variants

The computational cost of the application of MCP can be reduced by dropping either the left or the right factor in the first line of Eq. (33), or both, thus giving rise to two new block preconditioners:

$$\widetilde{\mathcal{M}}_1^{-1} = \begin{bmatrix} (L_K L_K^{\mathrm{T}})^{-1} & L_K^{-\mathrm{T}} L_K^{-1} B^{\mathrm{T}} L_S^{-\mathrm{T}} L_S^{-1} \\ 0 & -(L_S L_S^{\mathrm{T}})^{-1} \end{bmatrix}, \tag{34}$$

$$\widetilde{\mathcal{M}}_2^{-1} = \begin{bmatrix} (L_K L_K^{\mathrm{T}})^{-1} & 0 \\ 0 & -(L_S L_S^{\mathrm{T}})^{-1} \end{bmatrix}. \tag{35}$$

Similar strategies have been successfully applied in the solution of the Stokes problem [27–29]. The preconditioner $\widetilde{\mathcal{M}}_1^{-1}$ is denoted as Triangular MCP (T-MCP), while $\widetilde{\mathcal{M}}_2^{-1}$ is the Diagonal MCP (D-MCP). As they are the outcome of additional approximations, $\widetilde{\mathcal{M}}_1^{-1}$ and $\widetilde{\mathcal{M}}_2^{-1}$ are likely to require more Bi-CGStab iterations to converge than $\widetilde{\mathcal{M}}^{-1}$.

Following the proof of Theorem 4.1 with $R = L_S^{-1} B L_K^{-\mathrm{T}}$, the generalized eigenproblem $\mathcal{A}\boldsymbol{z} = \lambda \widetilde{\mathcal{M}}_1 \boldsymbol{z}$ has the same eigenvalues as:

$$\begin{bmatrix} -E_K & -R^{\mathrm{T}} \\ RE_K & -E_S \end{bmatrix} \begin{pmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{pmatrix} = (\lambda - 1) \begin{pmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{pmatrix}. \tag{36}$$

Table 1
Algorithm 1: application of the MCP preconditioned matrix

1.	solve $L_S^{\mathrm{T}} \mathtt{v}_1 = \mathtt{h}_2$
2.	$\mathtt{v}' = B^{\mathrm{T}} \mathtt{v}_1$
3.	solve $L_K \mathtt{w}' = \mathtt{v}'$
4.	$\mathtt{w} = \mathtt{h}_1 - \mathtt{w}'$
5.	solve $L_K^{\mathrm{T}} \mathtt{v}_2 = \mathtt{w}$
6.	$\mathtt{z}_1 = K\mathtt{v}_1 + B^{\mathrm{T}} \mathtt{v}_2$
7.	$\mathtt{z}_2 = B\mathtt{v}_1 - C\mathtt{v}_2$
8.	solve $L_K \mathtt{t}_1 = \mathtt{z}_1$
9.	solve $L_K^{\mathrm{T}} \mathtt{w} = \mathtt{t}_1$
10.	$\mathtt{w}' = B\mathtt{w} - \mathtt{z}_2$
11.	solve $L_S \mathtt{t}_2 = \mathtt{w}'$

The eigenproblem (36) turns out to be quite similar to (20). After taking norms, we obtain a similar bound as in (21):

$$|\lambda - 1| \leqslant (1 + \|R\|)\|E_K\| + \|R\| + \|E_S\|,$$

which indicates that as $\|E_K\|$ and $\|E_S\|$ tend to zero not all the eigenvalues will necessarily tend to one because of the matrix R. However, for reasonable values of $\|E_K\|, \|E_S\|, \|R\|$ the two bounds suggest that MCP and T-MCP should behave similarly. A thorough spectral analysis of block triangular preconditioners of this form can be found in [30].

By distinction, the generalized eigenproblem $\mathcal{A}\boldsymbol{z} = \lambda \widetilde{\mathcal{M}}_2 \boldsymbol{z}$ is equivalent to

$$\begin{bmatrix} I - E_K & -R^{\mathrm{T}} \\ R & -L_S^{-1} C L_S^{-\mathrm{T}} \end{bmatrix} \begin{pmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{pmatrix} = \lambda \begin{pmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{pmatrix}. \tag{37}$$

For small Δt values the order of magnitude of the terms in the (2,2) block of (37) is about 10^{-3}–10^{-2} regardless of $\|E_K\|$ and $\|E_S\|$. Only for very large Δt the preconditioned matrix becomes diagonally dominant with a more favorable eigenvalue distribution. In particular, if Δt is large, R resembles the null matrix with the MCP, T-MCP and D-MCP preconditioned matrices taking on asymptotically the same form:

$$\widetilde{\mathcal{M}}^{-1}\mathcal{A} \approx \widetilde{\mathcal{M}}_1^{-1}\mathcal{A} \approx \widetilde{\mathcal{M}}_2^{-1}\mathcal{A} \approx \begin{bmatrix} I - E_K & 0 \\ 0 & I - E_S \end{bmatrix}. \tag{38}$$

Therefore, we expect MCP, T-MCP and D-MCP to exhibit a similar performance for large Δt, with MCP and T-MCP to be generally preferred for small Δt.

6. Numerical results

6.1. Test problem

A vertical cross-section of the cylindrical porous volume used as a test problem is shown in Fig. 1. The medium consists of a sequence of alternating sandy and clayey layers, with the hydraulic conductivity $k_{\mathrm{sand}} = 10^{-5}$ m/s and $k_{\mathrm{clay}} = 10^{-8}$ m/s, the porosity $\phi = 0.20$, the Poisson ratio $v = 0.25$, and the Young modulus $E = 833.33$ MPa, corresponding to a uniaxial vertical compressibility $c_M = 10^{-3}$ MPa^{-1}. Standard Dirichlet conditions are prescribed, with fixed outer and bottom boundaries, and

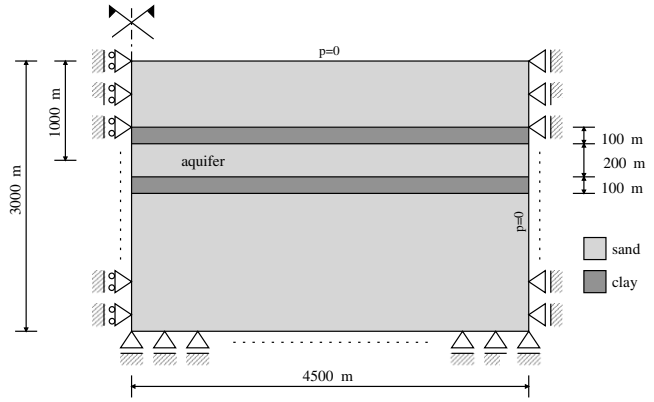

Fig. 1. Schematic representation of a vertical cross-section of the stratified porous medium used as a test problem.

Table 2
Main features of the sample problems

	m	n	N	nnz(K)	nnz(B)	nnz(C)	nnz($\mathcal{A}$)
M3Dsm	3553	10,659	14,212	453,321	151,107	50,369	805,904
M3D	31,775	95,325	127,100	4,177,395	1,392,465	464,155	7,426,480
M2Dsm	5551	11,102	16,653	153,004	76,502	38,251	344,259
M2D	21,901	43,802	65,703	608,404	304,202	152,101	1,368,909

zero pore pressure variation on the top and outer surfaces (see Fig. 1). The upper boundary is a traction-free plane. This sample problem is solved using both fully three-dimensional (M3Dsm and M3D) and axisymmetric (M2Dsm and M2D) grids.

In the M3Dsm test case, the medium is discretized into linear tetrahedral elements by projecting a plane triangulation made of 209 nodes and 400 triangles onto 17 layers located at different depths [31]. The grid M3Dsm totals $m = 3553$ nodes with a global matrix size N equal to 14,212. In the M3D test case, a plane triangulation made of 1025 nodes and 2016 triangles projected onto 31 layers is used. The M3D problem totals $m = 31,775$ nodes with $N = 127,100$.

In the axisymmetric configuration, the porous volume is discretized into annular elements with triangular cross-section and the FE equations are solved on a radial plane. We use two different regular triangulations depending on the values of the radial and vertical spacings Δr and Δz. They will be denoted in the sequel as M2Dsm ($\Delta r = \Delta z = 50$ m, $m = 5551$, $N = 16,653$) and M2D ($\Delta r = \Delta z = 25$ m, $m = 21,901$, $N = 65,703$).

The main features of the matrices arising from the above sample problems are summarized in Table 2 along with the number of nonzeroes nnz of $\mathcal{A}$ and the sub-matrices K, B and C.

6.2. Eigenvalue distribution of the preconditioned matrix

The quality of a preconditioner can be measured by the eigenvalue distribution of the preconditioned matrix. For the sake of simplicity, we discuss the outcome of the smallest problem M3Dsm, the other results being qualitatively similar.

Let us introduce the following symbols:

$$M_\lambda = \max \Re(\lambda), \quad m_\lambda = \min \Re(\lambda), \quad M_I = \max \Im(\lambda), \quad \kappa = \frac{M_\lambda}{m_\lambda},$$

where λ indicates a generic eigenvalue of $\mathcal{M}^{-1}\mathcal{A}$, $\widehat{\mathcal{M}}^{-1}\mathcal{A}$ or $\widetilde{\mathcal{M}}^{-1}\mathcal{A}$ with $\Re(\lambda)$ its real part. The distribution of the eigenvalues is studied for $\Delta t = 1$ s and a dropping tolerance in the AINV computation $\tau_A = 0.1$. We recall on passing that τ_A indicates the fraction of the diagonal term below which an AINV coefficient is dropped, i.e. the larger τ_A the sparser Z. The spectral norm of R, E_K and E_S along with the spectral condition number μ of $G^{-1}K$ is shown in Table 3. Note that with ICP $\|R\|$ is about 1 independently of τ_A, as expected from (26), while with MCP $\|R\|$ increases because of the additional approximation on S (use of Eq. (32) in place of (31)) introduced in the mixed approach.

The eigenvalue distribution of the preconditioned matrix with ECP, ICP, MCP and T-MCP is summarized in Table 4 and Fig. 2. Careful inspection of table and figures reveals that:

- As anticipated from theory, the ECP preconditioned matrix $\mathcal{M}^{-1}\mathcal{A}$ possesses all real and positive eigenvalues. Moreover, the ratio κ is significantly less than $\mu(G^{-1}K)$ thus suggesting that the overall preconditioner is superior to the preconditioner of K;

Table 3
Spectral norms of R, E_K and E_S in the M3Dsm test case with $\Delta t = 1$ s

	$\|R\|$	$\|E_K\|$	$\|E_S\|$	$\mu(G^{-1}K)$
ICP	1.05	1.59	0.64	481.9
MCP	1.95	0.88	3.13	14.5

Table 4
Problem M3Dsm with $\Delta t = 1$ s. Distribution of the eigenvalues of the preconditioned matrix with reference to the distance from unity ($\delta = |\lambda - 1|$)

Prec. (τ_A)	$\delta < 0.01$	$\delta < 0.1$	$\delta < 0.5$	$\delta \geqslant 0.5$	m_λ	M_λ	M_I	κ	# Real
ECP (0.1)	7924 (55.8%)	11,014 (77.5%)	13,875 (97.6%)	337 (2.4%)	0.008	2.365	0	296	14,212
ICP (0.1)	4460 (31.3%)	10,310 (72.5%)	13,861 (97.5%)	351 (2.5%)	0.008	2.364	0.245	296	8040
MCP (0.1)	9801 (69.0%)	12,721 (89.5%)	14,168 (99.7%)	44 (0.3%)	0.183	4.158	0.146	23	11,262
T-MCP (0.1)	8707 (61.3%)	12,605 (88.7%)	14,158 (99.6%)	54 (0.4%)	0.195	4.176	0.644	21	10,446

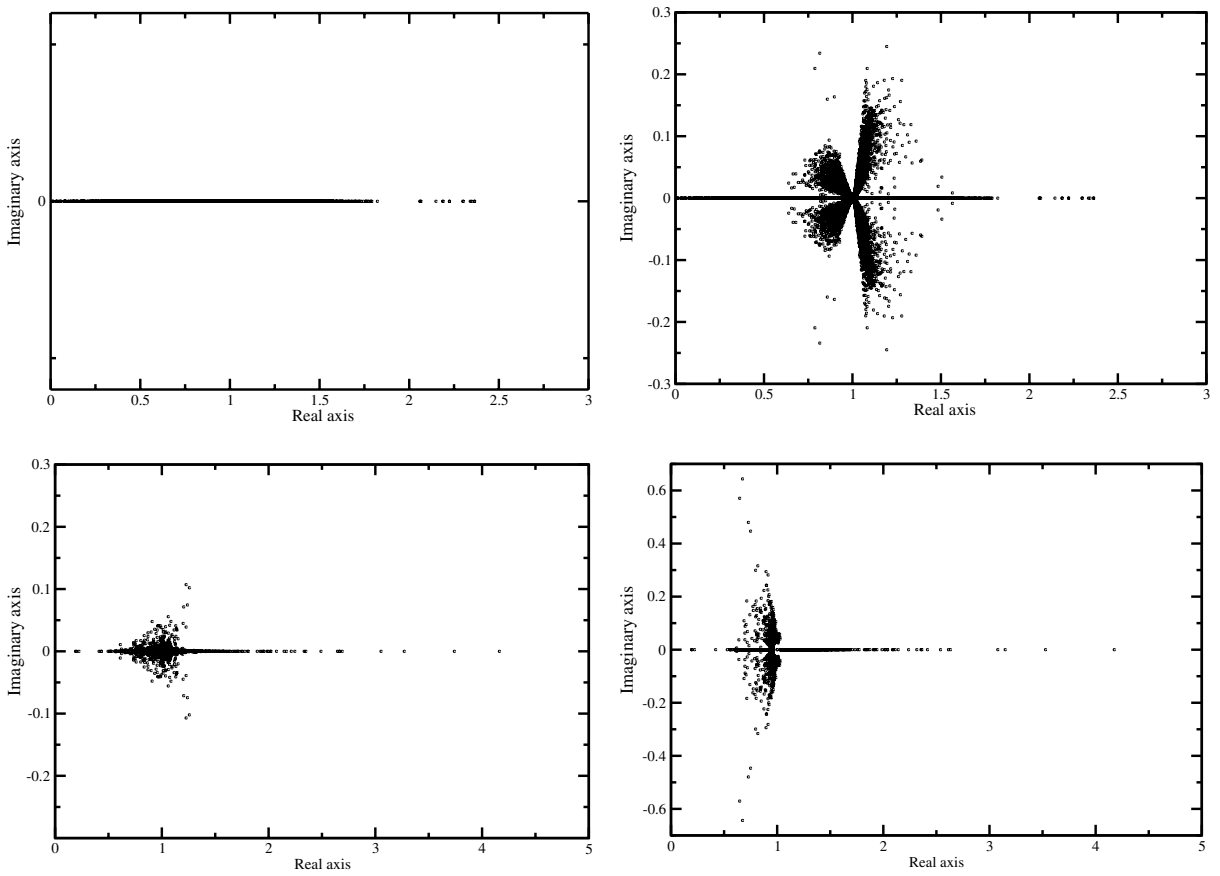

Fig. 2. Problem `M3Dsm` with $\Delta t = 1$ s. Eigenvalues of $\mathcal{A}$ preconditioned with ECP (up left), ICP (up right), MCP (down left) and T-MCP (down right) with $\tau_A = 0.1$.

- With ICP the spectrum of $\widehat{\mathcal{M}}^{-1}\mathcal{A}$ is no longer real. However, it turns out that its real part remains practically unchanged with no significant variations of κ. As is also observed in [26] the eigenvalues equal to one with ECP evolve to complex eigenvalues with a relatively small imaginary part;
- The MCP preconditioned matrix $\widetilde{\mathcal{M}}^{-1}\mathcal{A}$ has a larger number of real eigenvalues than $\widehat{\mathcal{M}}^{-1}\mathcal{A}$ and the ratio κ is more than 10 times smaller. This is due to the better implicit K^{-1} approximation accounted for by a m_λ value closer to 1.
- The main difference between the eigenspectrum of $\widetilde{\mathcal{M}}^{-1}\mathcal{A}$ and $\widetilde{\mathcal{M}}_1^{-1}\mathcal{A}$ rests on the increase of the imaginary part of the complex eigenvalues. The real spectrum, however, is substantially unchanged;
- Consistent with Theorems 4.1, 4.2 and 4.3, the distance of any eigenvalue from unity is smaller than $(1 + \|R\|)^2\|E_K\| + \|E_S\|$, the imaginary part is bounded by $\|R\| \cdot \|E_K\|$ and the number of real eigenvalues is larger than $n - m$.

The above analysis shows that the spectral condition numbers of the preconditioned matrix are similar with both ECP and ICP. Therefore, the inexact implementation is to be generally preferred as it is less expensive, as is also evidenced with other numerical examples in [17]. By distinction, MCP exhibits a more favorable eigenvalue distribution mainly because of a larger m_λ, hence MCP is expected to perform better than ICP. As the real spectrum of the triangular variant T-MCP is practically the same as MCP, we also expect T-MCP to represent a promising alternative to MCP owing to its slightly smaller application cost.

The eigenspectra previously discussed, along with the theoretical findings provided by Theorems 3.1 and 4.2, suggest that the quality of any Constraint Preconditioner is basically bounded by the spectral condition number of $G^{-1}K$. Hence, as the mesh is refined the convergence rate of a solver preconditioned with ECP, ICP and MCP is expected to scale in the same way as using G^{-1} as a preconditioner for K. Recall that K is obtained with the FE discretization of Eq. (2) where the differential operator is very similar to the Laplace operator. It is well-known that the spectral condition number of a Laplace FE matrix scales proportionally to, and hence increases as, $1/\ell^2$, where ℓ is a representative linear measure of the element size, and that an incomplete Cholesky factorization does not mitigate this outcome. It might be shown that this holds true for the structural stiffness matrix K as well. Therefore, as the mesh is refined we expect also the quality of a Constraint Preconditioner to scale approximately as $1/\ell^2$.

6.3. Performance of the Mixed Constraint Preconditioner

The CPU time and the number of iterations to convergence for Bi-CGStab preconditioned with LSL–ILUT [6], ICP and MCP in the M3Dsm, M3D, M2Dsm and M2D test problems are shown below. A measure ρ of the density of the preconditioner factors is defined as:

LSL–ILUT $\rho = \mathrm{nnz}(\mathrm{ILUT})/\mathrm{nnz}(\mathcal{A})$
ICP $\rho = [\mathrm{nnz}(Z) + \mathrm{nnz}(L_S)]/\mathrm{nnz}(\mathcal{A})$
MCP $\rho = [\mathrm{nnz}(L_S) + \mathrm{nnz}(L_K)]/\mathrm{nnz}(\mathcal{A})$

Parameter ρ gives an indication as to the additional core memory needed for computing and storing the preconditioner. T_p denotes the CPU time to evaluate the preconditioner, i.e. in ICP and MCP the overall time to compute S and L_S, and T_s the CPU time required to iterate to convergence, while T_i is the "preprocessing" time needed to compute AINV and S_0. The iterations are completed when the final solution $\boldsymbol{x}$ satisfies the relative error ε:

$$\varepsilon = \frac{\|\boldsymbol{x} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^*\|} \leqslant \mathtt{tol} \tag{39}$$

$\boldsymbol{x}^*$ being a prescribed test solution with all components equal to 1 and tol between 10^{-8} and 10^{-5} depending on problem. All numerical experiments are performed on a Compaq DS20 equipped with an alpha-processor "ev6" at 500 MHz, 1.5 GB of core memory, and 8 MB of secondary cache. The code is written in Fortran 90 and compiled with -O4 -tune=ev6 -arch=ev6 options. The CPU times are given in seconds.

The performance of MCP as compared to ICP and LSL–ILUT in the test cases considered above is shown in Tables 5 (3D problems) and 6 (axisymmetric problems). The factor L_K is computed allowing for 50 new nonzero entries for each row (lfil $_K = 50$) and setting a drop tolerance relative to the average size of the K coefficients $\tau_K = 10^{-4}$.

Inspection of Tables 5 and 6 reveals that MCP yields a faster Bi-CGStab convergence than both LSL–ILUT and ICP. The number of iterations is significantly smaller consistent with the better spectral properties of the preconditioned matrix $\widetilde{\mathcal{M}}^{-1}\mathcal{A}$. Fig. 3 provides the ratio between LSL–ILUT and MCP CPU times, and between ICP and MCP CPU times for different Δt values. It is worth noting that MCP exhibits a speed-up between 1.2 and 4 relative to LSL–ILUT and between 1.7 and 6 relative to ICP for the smallest time steps, i.e. the most difficult problems. As Δt grows, ill-conditioning becomes less severe and all algorithms (LSL–ILUT, ICP and MCP) tend to exhibit a similar performance, with the only exception of problem M2D. Notice that this is also accounted for Theorem 4.1 and Eq. (25), as $\|E_S\|$ turns out to be more important than $\|E_K\|$ for $\Delta t \to +\infty$.

The results obtained with the T-MCP variant are shown in Table 7. As expected, the average number of iterations is slightly larger than with MCP. However, the total CPU time is slightly smaller in a number of cases (printed in italic in Table 7) due to the lower cost per iteration. Therefore, T-MCP appears to be roughly equivalent to MCP, and sometimes even a little better. By contrast, the D-MCP-BiCGstab algorithm did not converge within the maximum number of iterations allowed for in the four test cases and for all time steps $\Delta t \in [1, 10^5]$.

Finally, MCP and T-MCP have been compared with LSL–ILUT on a larger example addressing the consolidation of the Venice lagoon subsurface, Italy, in a pilot project of seawater injection [32]. The corresponding matrix has a size $N = 416,800$ with $\mathrm{nnz}(\mathcal{A}) = 22,322,336$. This new example has been run on an IBM computer equipped with a Power5 dual core processor at 1900 MHz and 16 GB of core memory. The results are shown in Table 8. MCP and T-MCP are again almost equivalent, with T-MCP superior for small time steps. The comparison with LSL–ILUT provides evidence of the better performance of a

Table 5
3D test problems

	Δt	M3Dsm[a]					M3D[b]				
		ρ	Iter.	CPU time (s)			ρ	Iter.	CPU time (s)		
				T_p	T_s	T_t			T_p	T_s	T_t
LSL–ILUT	10^0	1.09	155	8.23	9.23	17.46	1.71	178	73.45	141.28	214.73
	10^2	1.09	162	7.54	9.06	16.60	1.69	163	64.13	114.52	178.65
	10^4	0.99	150	3.87	9.37	13.24	1.43	84	24.62	61.81	86.43
ICP ($\tau_A = 0.05$)	10^0	1.52	70	2.83	6.99	9.82	1.62	175	26.95	165.52	192.47
	10^2	1.45	69	2.16	6.35	8.51	1.60	176	26.77	163.62	190.39
	10^4	1.05	72	1.05	4.96	6.01	1.58	175	23.79	157.45	181.24
MCP ($\tau_A = 0.1$)	10^0	1.11	29	1.15	3.54	4.69	1.20	66	8.66	93.81	102.47
	10^2	1.06	27	0.94	2.95	3.89	1.20	60	8.67	84.61	93.28
	10^4	0.88	30	0.45	3.25	3.70	1.17	55	8.21	77.27	85.48

CPU time (s) for Bi-CGStab preconditioned with optimal LSL–ILUT, ICP and MCP ($T_t = T_p + T_s$). S is computed with $\tau_S = 10^{-4}$.
[a] With ICP: $T_i = 5.28$; with MCP: $T_i = 3.83$.
[b] With ICP: $T_i = 40.20$; with MCP: $T_i = 12.75$.

Table 6
Axisymmetric problems

	Δt	M2Dsm[a]					M2D[b]				
		ρ	Iter.	CPU time (s)			ρ	Iter.	CPU time (s)		
				T_p	T_s	T_t			T_p	T_s	T_t
LSL–ILUT	10^0	3.73	94	1.47	5.49	6.96	3.71	187	5.44	45.90	51.33
	10^2	2.79	98	1.65	4.64	6.29	2.86	166	3.42	33.22	36.64
	10^4	1.86	72	0.80	2.56	3.36	2.82	99	5.04	20.05	25.09
ICP ($\tau_A = 0.05$)	10^0	1.64	165	0.19	8.76	8.95	6.18	162	15.52	128.45	143.97
	10^2	1.63	144	0.18	8.02	8.20	6.15	158	15.44	121.07	136.51
	10^4	1.60	143	0.17	7.13	7.31	6.12	163	9.91	117.34	127.25
MCP ($\tau_A = 0.05$)	10^0	2.74	35	0.19	3.66	3.85	2.81	54	0.84	25.19	26.03
	10^2	2.74	33	0.18	3.19	3.37	2.81	46	0.84	25.04	25.88
	10^4	2.71	35	0.17	3.32	3.61	2.78	41	0.78	19.15	19.93

CPU time (s) for Bi-CGStab preconditioned with optimal LSL–ILUT, ICP and MCP ($T_t = T_s + T_p$). S is computed with $\tau_S = 10^{-4}$.
[a] With ICP: $T_i = 0.75$; with MCP: $T_i = 1.83$.
[b] With ICP: $T_i = 32.21$; with MCP: $T_i = 6.55$.

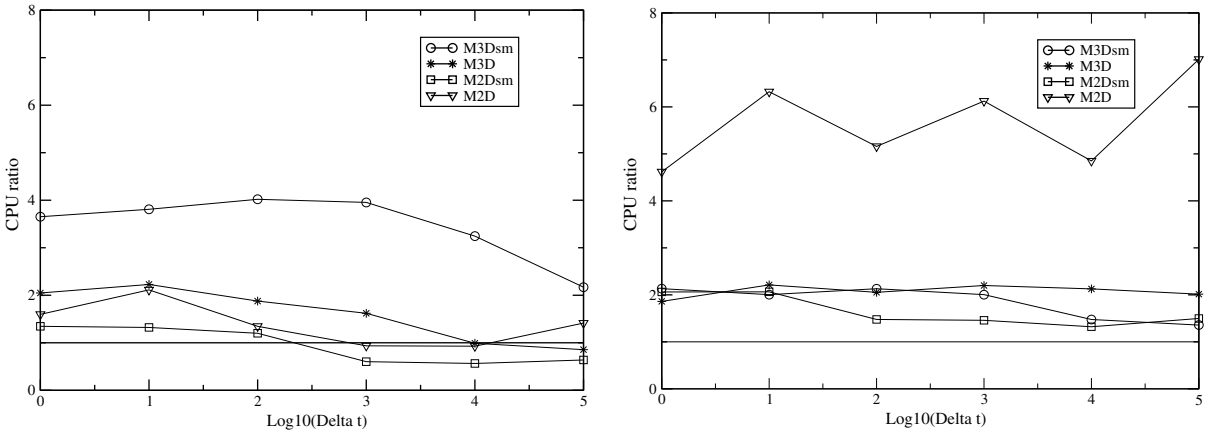

Fig. 3. Ratios between the LSL–ILUT and MCP CPU time (left); ICP and MCP CPU time (right) for different Δt values.

Table 7
Number of iterations and overall CPU time (s) for Bi-CGStab preconditioned with T-MCP

	M3Dsm		M3D		M2Dsm		M2D	
Δt	Iter.	CPU	Iter.	CPU	Iter.	CPU	Iter.	CPU
10^0	37	5.21	66	*85.88*	38	*3.03*	57	*25.34*
10^2	36	4.99	60	*80.95*	37	*3.08*	51	*20.65*
10^4	40	3.96	56	*72.54*	40	*3.24*	49	21.38

Table 8
CPU time (s) for Bi-CGStab preconditioned with LSL–ILUT, MCP and T-MCP in a large size system ($N = 416,800$) addressing a real application

	LSL–ILUT		MCP		T-MCP	
Δt	Iter.	CPU	Iter.	CPU	Iter.	CPU
10^0	66	76.9	38	22.1	32	18.1
10^2	21	40.3	32	18.9	28	15.6
10^4	25	25.2	17	7.7	23	8.7

The examples have been run on a different computer with respect to the previous test cases.

Mixed Constraint approach, with speed-up factors up to 4.2 in the most favorable case. The overall gaining is emphasized considering also the memory occupation, that turns out to be about three times larger using LSL–ILUT than MCP and T-MCP.

7. Conclusions

A Mixed Constraint Preconditioner (MCP) has been developed for the iterative solution to the FE coupled consolidation equations. MCP is an efficient variant of the Inexact Constraint Preconditioner implementing two different approximations (both implicit and explicit) of the inverse of the (1,1) structural block K into the same algorithm. The implicit approximation of K^{-1} is obtained with an incomplete triangular decomposition, while the explicit approximation is provided by the approximate inverse AINV and is used in the Schur complement S. An upper bound is derived for the distance of the eigenvalues of the preconditioned matrix from 1 depending on the errors involved in the approximation of both K^{-1} and S^{-1}. The computational performance of MCP in a number of realistic numerical examples is fully consistent with Theorems 4.1 and 4.3 and the relationship between the norms of the coefficient sub-matrices K, B and C and the hydro-geomechanical properties of the porous medium.

A number of numerical experiments show that in the most ill-conditioned problems MCP typically outperforms both the optimal LSL–ILUT preconditioner and ICP by up to a factor of 7 in the most favorable case. This outcome holds true also for larger size matrices. Similarly to ICP, MCP proves more robust than LSL–ILUT allowing for Bi-CGStab to converge in severely ill-conditioned problems as well. The possibly expensive cost for the computation of MCP is quickly made up for in transient simulations and can be viewed as a preprocessing cost with only a limited impact on the overall solver performance. The Triangular MCP variant proves almost equivalent to MCP and can represent a viable alternative especially for small time steps. By contrast, the Diagonal MCP variant is a too poor approximation of $\mathcal{A}^{-1}$ and may perform satisfactorily only for very large Δt values. Finally, a practical drawback of MCP and T-MCP might be represented by the relatively large number of parameters to be set (dropping tolerances for the computation of S and AINV, and the fill-in parameters for the incomplete decomposition of K and S). However, the analysis carried out on the examples dealt with in the present paper points out that MCP and T-MCP are quite robust in relation to the above parameters and are rather insensitive to the time step size.

References

[1] M.A. Biot, General theory of three-dimensional consolidation, J. Appl. Phys. 12 (1941) 155–164.
[2] 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 (1992) 631–644.
[3] S. Chan, K. Phoon, F. Lee, A modified Jacobi preconditioner for solving ill-conditioned Biot's consolidation equations using symmetric quasi-minimal residual method, Int. J. Numer. Anal. Methods Geomech. 25 (2001) 1001–1025.
[4] G. Gambolati, G. Pini, M. Ferronato, Numerical performance of projection methods in finite element consolidation models, Int. J. Numer. Anal. Methods Geomech. 25 (2001) 1429–1447.
[5] G. Gambolati, G. Pini, M. Ferronato, Direct, partitioned and projected solution to finite element consolidation models, Int. J. Numer. Anal. Methods Geomech. 26 (2002) 1371–1383.
[6] G. Gambolati, G. Pini, M. Ferronato, Scaling improves stability of preconditioned CG-like solvers for FE consolidation equations, Int. J. Numer. Anal. Methods Geomech. 27 (2003) 1043–1056.
[7] K.C. Toh, K.K. Phoon, S.H. Chan, Block preconditioners for symmetric indefinite linear systems, Int. J. Numer. Meth. Eng. 60 (2004) 1361–1381.
[8] X. Chen, K.C. Toh, K.K. Phoon, A modified SSOR preconditioner for sparse symmetric indefinite linear systems of equations, Int. J. Numer. Meth. Eng. 65 (2006) 785–807.
[9] M. Ferronato, G. Gambolati, P. Teatini, Ill-conditioning of finite element poroelasticity equations, Int. J. Solids Struct. 38 (2001) 5995–6014.
[10] L. Lukšan, J. Vlček, Indefinitely preconditioned inexact Newton method for large sparse equality constrained nonlinear programming problems, Numer. Lin. Alg. Appl. 5 (1998) 219–247.
[11] C. Keller, N.I.M. Gould, A.J. Wathen, Constraint preconditioning for indefinite linear systems, SIAM J. Matrix Anal. Appl. 21 (2000) 1300–1317.
[12] I. Perugia, V. Simoncini, Block-diagonal and indefinite symmetric preconditioners for mixed finite elements formulations, Numer. Lin. Alg. Appl. 7 (2000) 585–616.
[13] L. Bergamaschi, J. Gondzio, G. Zilli, Preconditioning indefinite systems in interior point methods for optimization, Comput. Optim. Appl. 28 (2004) 149–171.
[14] M. Benzi, G. Golub, J. Liesen, Numerical solution of saddle point problems, Acta Numer. 14 (2005) 1–137.
[15] H.S. Dollar, N.I.M. Gould, W.H.A. Schilders, A.J. Wathen, Implicit-factorization preconditioning and iterative solvers for regularized saddle-point systems, SIAM J. Matrix Anal. Appl. 28 (2006) 170–189.
[16] L. Bergamaschi, J. Gondzio, M. Venturin, G. Zilli, Inexact constraint preconditioners for linear systems arising in interior point methods, Comput. Optim. Appl. 36 (2007) 136–147.
[17] L. Bergamaschi, M. Ferronato, G. Gambolati, Novel preconditioners for the iterative solution to FE-discretized coupled consolidation equations, Comp. Meth. Appl. Mech. Eng. 196 (2007) 2647–2656.
[18] M. Benzi, M. Tůma, A comparative study of sparse approximate inverse preconditioners, Appl. Numer. Math. 30 (1999) 305–340.
[19] Y. Saad, ILUT: a dual threshold incomplete ILU factorization, Numer. Lin. Alg. Appl. 1 (1994) 387–402.
[20] H.S. Dollar, Constraint-style preconditioners for regularized saddle point problems, SIAM J. Matrix Anal. Appl. 29 (2007) 672–684.
[21] C. Durazzi, V. Ruggiero, Indefinitely preconditioned conjugate gradient method for large sparse equality and inequality constrained quadratic problems, Numer. Lin. Alg. Appl. 10 (2003) 673–688.
[22] M. Rozložník, V. Simoncini, Krylov subspace methods for saddle point problems with indefinite preconditioning, SIAM J. Matrix Anal. Appl. 24 (2002) 368–391.
[23] M. Benzi, C.D. Meyer, M. Tůma, A sparse approximate inverse preconditioner for the conjugate gradient method, SIAM J. Sci. Comput. 17 (1996) 1135–1149.
[24] M. Benzi, J.K. Cullum, M. Tůma, Robust approximate inverse preconditioning for the conjugate gradient method, SIAM J. Sci. Comput. 22 (2000) 1318–1332.
[25] R.W. Freund, N.M. Nachtigal, Software for simplified Lanczos and QMR algorithms, Appl. Numer. Math. 19 (1995) 319–341.
[26] M. Benzi, V. Simoncini, On the eigenvalues of a class of saddle point matrices, Numer. Math. 103 (2006) 173–196.
[27] D. Silvester, A. Wathen, Fast iterative solution of stabilised Stokes systems, Part II: using general block preconditioners, SIAM J. Numer. Anal. 31 (1994) 1352–1367.
[28] D. Silvester, H. Elman, D. Kay, A. Wathen, Efficient preconditioning of the linearized Navier–Stokes equations for incompressible flow, J. Comp. Appl. Math. 128 (2001) 261–279.

[29] H.C. Elman, D.J. Silvester, A.J. Wathen, Performance and analysis of saddle point preconditioners for the discrete steady-state Navier–Stokes equations, Numer. Math. 90 (2002) 665–688.
[30] V. Simoncini, Block triangular preconditioners for symmetric saddle-point problems, Appl. Numer. Math. 49 (2004) 63–80.
[31] G. Gambolati, G. Pini, T. Tucciarelli, A 3-D finite element conjugate gradient model of subsurface flow with automatic mesh generation, Adv. Water Resour. 3 (1986) 34–41.
[32] N. Castelletto, M. Ferronato, G. Gambolati, M. Putti, P. Teatini, Can Venice be raised by pumping water underground? A pilot project to help decide, Water Resour. Res. 44 (2008) W01408, doi:10.1029/2007WR006177.