

RESEARCH ARTICLE

Open Access



Scalable block preconditioners for saturated thermo-hydro-mechanics problems

Ana C. Ordonez¹, Nicolas Tardieu^{1*} , Carola Kruse², Daniel Ruiz² and Sylvie Granet¹

*Correspondence:
nicolas.tardieu@edf.fr

¹EDF/R & D, IMSIA UMR
EDF-CNRS-CEA-ENSTA 9219, Bd
Gaspard Monge, Palaiseau
91120, France

²Cerfacs, 42 Av. Gaspard Coriolis,
Toulouse 31100, France

Abstract

We are interested in the modelling of saturated thermo-hydro-mechanical (THM) problems that describe the behaviour of a soil in which a weakly compressible fluid evolves. It is used for the evaluation of the THM impact of high-level activity radioactive waste exothermicity within a deep geological disposal facility. We shall present the definition of a block preconditioner with nested Krylov solvers for the fully coupled THM equations. Numerical results reflect the good performance of the proposed preconditioners that show to be weakly scalable until more than 2000 cores and more than 1 billion degrees of freedom. Thanks to their performance and robustness, a real waste storage problem on a scale, to our knowledge, unprecedented in the field, can be addressed.

Keywords: Multiphysics, Preconditioning, Finite element, High performance computing

Introduction

Context

The detailed modeling of underground phenomena is of major interest in several industrial fields ranging from oil and gas to nuclear waste storage and civil engineering [1–3]. This is particularly the case for coupled phenomena where several physics come into play and can make it difficult to understand their respective influences. Nuclear waste storage is an illustrative example of a thermo-hydro-mechanics (THM) coupled problem [4]. Nuclear wastes generate heat that increases the fluid and soil temperature, thus changing not only the volume of each phase due to thermal expansion but also the values of the material parameters which are functions of temperature. The numerical simulation of these problems results in the solution of a coupled system of nonlinear partial differential equations (PDE). Due to implicit time discretization, often preferred for its unconditional stability, the solution of large ill-conditioned linear systems turns out to be the heaviest task in terms of computational burden. The design of scalable parallel solvers is thus an essential topic to benefit from the massive computational power of computer architectures. This is the concern of the present work.

Previous work

The modeling of underground phenomena has been initiated by the pioneering work of Terzaghi in his theory of one-dimensional consolidation [5]. The theory was then expanded by Biot, who used the coupling of Darcy's and Hooke's laws together with the Terzaghi's principle [6]. Then, he also included the effect of temperature by using the new concept of "virtual dissipation" [7]. Finally Coussy showed that a general theory of thermomechanics of saturated porous media could be established based on standard thermodynamics principles [8]. We shall follow this well-established framework to model the thermomechanical behavior of a deformable porous media, saturated with an almost incompressible single-phased fluid. It results in a system of three balance equations for the linear momentum, the mass of fluid and the energy of the medium, where the physical phenomena in play are conduction and convection.

When considering the numerical solution of a coupled system of PDE, sequential or monolithic approaches can be used. For the sequential method, also called staggered or operator-splitting method, each balance equation is solved once at a time, thus requiring an update strategy in order to transfer the values of each field from a balance equation to the other. Thanks to an appropriate convergence criterion, the solution of the coupled system is recovered. The main interest of such approaches relies in the use of existing verified and robust simulators, dedicated to a particular problem, with the final goal to reduce coding efforts. However, they obviously require a special coupling algorithm, whose numerical stability and accuracy can be problematic [9, 10] and sometimes also the use of a coupling software, which can penalize the reduction of the programming burden [11]. For the monolithic method, all balance equations are solved simultaneously, requiring processing within the same computer software. In addition to the development of a dedicated software application, monolithic approaches require that the inf-sup condition is met [12], in order to avoid spurious oscillations in the pressure field. This topic benefited special attention in the literature and several strategies were proposed to circumvent the problem. The latest works consider three-field formulations for the Biot's part (aka HM) of the global problem, with different choices of the extra-field (solid pressure [13, 14] or Darcy's velocity [15, 16]) in order to alleviate the non-physical pressure oscillations at the interface between materials with different permeabilities. In the sequel, a monolithic approach is considered and, as mentioned earlier, an efficient and robust solver is essential for the performance of the method.

When systems are of small to medium size, direct solvers are well suited mainly because of their excellent robustness. However, when systems become larger ($> 10^7$ Degree of Freedom (DoF)), time and memory consumption of direct solvers, which grows substantially faster than the number of unknowns, gets unfeasible and the use of iterative methods become mandatory from a performance point of view. This topic enjoys intense research for more than two decades, mainly in the field of HM problems, focused either on multigrid methods or on block preconditioners for Krylov methods.

In the multigrid framework, the choice of the smoother is a key point to ensure the convergence and the performance of the method. Several strategies have been proposed such as Vanka-type smoothers for three-field non-linear poroelasticity, Uzawa-type smoothers obtained by splitting the discrete operators or also parameter dependent smoothers based

on a fixed-stress scheme [17–19]. An efficient distributive smoother for staggered grids is proposed and analyzed in [20,21].

Block preconditioners are a natural solution for linear systems involving unknowns of different kinds. This is obviously the case for saddle point problems, of which the Stokes problem is a canonical example and has motivated intense research (see [22] for extensive bibliography). Based on block factorizations, diagonal or triangular preconditioners have been proposed. Algebraic multigrid (AMG), incomplete factorization and approximate inverses are often used for preconditioning the displacement block [16,23]. The design of a preconditioner for the pressure block implies the approximation of the Schur complement, whose exact evaluation is computationally impossible even for moderate size problems due to its dense nature. When the media is fully saturated by the fluid, several approximations are proposed based on AMG, mass matrices and incomplete factorization [23–25]. When phase change in the fluid are to be considered, the Constrained Pressure Residual (CPR) method is often considered as the preconditioner of choice for real-life problems in the oil reservoir community [26–30]. It must be noticed that all block preconditioners do not rely on block factorization. Recent works based on the choice of physical parameter based norms have been proposed [13,31]. They turn to be block diagonal or triangular preconditioners that show great independence from the material parameters, thus motivating the name of parameter-robust preconditioners. Other approaches are based on particular matrix decompositions which enjoy nice convergence properties when used as preconditioners [32].

Present work

We consider a porous deformable solid material saturated with an almost incompressible fluid with non-isothermal effects. Although the material is considered elastic and the fluid obeys Darcy's law, the energy balance equation is nonlinear and drives ourselves to the use of the Newton's method. We use a monolithic solution approach and focus on the use of block preconditioned Krylov methods in order to solve the linearized system, whose unknowns are the displacement, the pressure and the temperature of the continuum. In the present work, we consider a two field, displacement and pressure, formulation with respective quadratic and linear interpolations. It has indeed shown to be stable on several severe tests [24]. The temperature field has linear interpolation and this choice will be discussed in the sequel.

Outline

We start by presenting the general framework of the THM system in some detail in order to show how the couplings with temperature through diffusion and convection mechanisms are handled. After writing the weak formulation, time and space discretizations are presented followed by the linearization of the residual. The block preconditioning strategy is discussed and its robustness with respect to the number of DoF and to the physical parameters are assessed on a test case. The weak and strong scaling are also evaluated and compared the performance of a commonly used block factorization preconditioner. The solution of a large industrial problem by the proposed framework is discussed. The paper ends with conclusions and outlooks.

Table 1 Parameters

Symbol	Definition	Unit
$\underline{\underline{A}}$	Fourth order Hooke's tensor	Pa
$\underline{\underline{\varepsilon}}$	Strain tensor	–
E	Young's modulus	Pa
ν	Poisson's ratio	–
K_0	Drained bulk modulus of the continuum	Pa
K_f	Bulk modulus of the fluid	Pa
K_s	Bulk modulus of the solid matrix	Pa
K_{int}	Intrinsic permeability	m ²
φ	Porosity	–
μ_f	Fluid dynamic viscosity	Pa s
h_f	Specific enthalpy of the fluid	J kg ^{−1}
h_{f0}	Initial specific enthalpy of the fluid	J kg ^{−1}
p_{atm}	Atmospheric pressure	Pa
C_s	Specific heat of the solid	J kg ^{−1} K ^{−1}
C_f	Specific heat of the fluid	J kg ^{−1} K ^{−1}
C_f^p	Specific heat of the fluid with constant pressure	J kg ^{−1} K ^{−1}
C_ϵ^0	Specific heat of the medium to constant deformation	J K ^{−1} m ^{−3}
C_σ^0	Specific heat of the medium to constant constraint	J K ^{−1} m ^{−3}
ρ_s	Solid density	kg m ^{−3}
ρ_f	Fluid density	kg m ^{−3}
ρ_m	Medium density	kg m ^{−3}
λ_H	Hydraulic conductivity	Pa ^{−1} m ² s ^{−1}
λ_T	Thermal conductivity	W m ^{−1} K
T_0	Temperature of reference	K
α_s	Dilation coefficient of the solid	K ^{−1}
α_f	Dilation coefficient of the fluid	K ^{−1}
α_m	Homogenized dilation coefficient of the medium	K ^{−1}

The THM equations

Due to the large quantity of parameters, all symbols and units used in the article are listed in Table 1.

General framework

An isotropic saturated mono-phased porous medium is considered in the context of small perturbations. According to Biot's theory, it is modeled as a linear elastic solid skeleton with pores containing a freely moving fluid. Due to the presence of the pores, an essential characteristic of the medium is the porosity, named φ in the sequel. It is the ratio between the volume of the void and the total volume of the medium. The latter is considered on a macroscopic scale within the framework of continuum mechanics and we therefore assume that the representative elementary volume includes a sufficient volume of grains and void space to verify this assumption. The above volumes are expressed in the current configuration so that φ is often referred to as the Eulerian porosity.

Another essential parameter of the medium is the Biot's coefficient b . It is the ratio of the volume of fluid gained or lost in a specimen under load to the change in volume of that specimen, when the pore pressure remains constant [33]. Given the solid matrix bulk modulus K_s and the bulk modulus of the drained medium K_0 , it expresses as $b = 1 - \frac{K_0}{K_s}$. We shall suppose that the solid matrix does not undergo significant volume changes,

which is the case for the soft soils we consider here; it results in $b = 1$, that will be used in the sequel.

Besides the aforementioned bulk moduli, the medium is also characterized from a material point of view by the following parameters:

- the hydraulic permeability λ_H . It measures the medium's ability to transmit a given fluid. It is the ratio between the intrinsic permeability named K_{int} and the fluid viscosity μ_l .
- the thermal conductivity named λ_T . It measures the medium's ability to conduct heat.
- the specific enthalpy of the water h_f represents the enthalpy of the fluid per unit mass. It is the sum of the specific internal energy of the fluid and the product of the pressure and the specific volume.

These parameters, some of which appear explicitly therein, are of major importance in the balance equations. They are three in number since the medium is saturated and mono-phased: the linear momentum, the mass of fluid and the energy of the medium.

Balance equations

We introduce the three balance equations of the problem that constitutes the thermohydromechanical model of the ground. This model follows the work of Coussy [8].

- The linear momentum equation:

$$-\operatorname{div}(\underline{\underline{\sigma}}) = \underline{\underline{f}}^e$$

- $\underline{\underline{\sigma}}$ denotes the total Cauchy stress tensor
- $\underline{\underline{f}}^e$ denotes the total external forces

- The water mass conservation:

$$\dot{m}_f + \operatorname{div}(\underline{\underline{\psi}}) = 0$$

- m_f denotes the fluid mass of the continuum
- $\underline{\underline{\psi}}$ is the fluid mass flux

- The energy conservation:

$$h_f \dot{m}_f + \dot{Q}' + \operatorname{div}(h_f \underline{\underline{\psi}}) + \operatorname{div}(\underline{\underline{q}}) = \Theta$$

- h_f denotes the specific fluid enthalpy
- $\underline{\underline{q}}$ denotes the heat flux
- Θ denotes the source/sink of heat
- Q' denotes the heat in the medium that is not convected, the heat input that doesn't come from an outside source.

Let us now detail the above balance equation in order to reveal the couplings between the phenomena involved.

The balance of linear momentum

The mechanics equilibrium equations is applied on the total stress $\underline{\underline{\sigma}}$

$$- \operatorname{div}(\underline{\underline{\sigma}}) = \underline{\underline{f}}^e \quad (1)$$

where $\underline{\underline{f}}^e$ denotes the total volume external forces. Biot's definition of effective stress $\underline{\underline{\sigma}}' = \underline{\underline{\sigma}} + p\underline{\underline{\mathbf{I}}}$ with the tension positive-sign convention is used in this article (we recall that $b = 1$ in the previous equation). After including it in Eq. (1), we get:

$$- \operatorname{div}(\underline{\underline{\sigma}}') + \nabla p = \underline{\underline{f}}^e \quad (2)$$

We shall now use the expression of the constitutive equation while taking into account the thermal expansion of the medium $\underline{\underline{\varepsilon}}^{th}$, which is a function of T :

$$\begin{aligned} \underline{\underline{\sigma}}' &= \underline{\underline{A}} : (\underline{\underline{\varepsilon}}(\underline{\underline{u}}) - \underline{\underline{\varepsilon}}^{th}(T)) \\ &= \underline{\underline{A}} : (\underline{\underline{\varepsilon}}(\underline{\underline{u}}) - \alpha_s(T - T_0)\underline{\underline{\mathbf{I}}}) \\ &= \underline{\underline{A}} : \underline{\underline{\varepsilon}}(\underline{\underline{u}}) - 3K_s\alpha_s(T - T_0)\underline{\underline{\mathbf{I}}} \end{aligned} \quad (3)$$

$\underline{\underline{A}}$ is the fourth order Hooke's tensor (which is a function of E and ν), $K_s = \frac{E}{3(1-2\nu)}$ is the bulk modulus of the solid matrix, α_s is the thermal expansion coefficient and T_0 is the reference temperature (temperature at equilibrium).

If we inject the constitutive law into Eq. (2), we obtain the expression, where all couplings become explicit:

$$- \operatorname{div}(\underline{\underline{A}} : \underline{\underline{\varepsilon}}(\underline{\underline{u}})) + \nabla p + 3K_s\alpha_s\nabla T = \underline{\underline{f}}^e \quad (4)$$

The conservation of water mass

The water mass conservation equation is

$$\dot{m}_f + \operatorname{div}(\underline{\underline{\psi}}) = 0 \quad (5)$$

We shall now inject in the above equation several hypothesis on the fluid behaviour.

First, we call the domain's initial porosity φ^0 and the fluid's initial density ρ_f^0 . Then the total fluid mass expresses with respect to this initial state:

$$m_f = (1 + \operatorname{div}(\underline{\underline{u}}))\rho_f\varphi - \rho_f^0\varphi^0$$

The time derivative of the fluid mass is then given by:

$$\begin{aligned} \dot{m}_f &= \rho_f\varphi \operatorname{div}(\dot{\underline{\underline{u}}}) + (1 + \operatorname{div}(\underline{\underline{u}}))\varphi \dot{\rho}_f + \rho_f(1 + \operatorname{div}(\underline{\underline{u}}))\dot{\varphi} \\ &= \rho_f\varphi \operatorname{div}(\dot{\underline{\underline{u}}}) + \varphi \dot{\rho}_f + \rho_f\dot{\varphi} \end{aligned} \quad (6)$$

where we used $\operatorname{div}(\underline{\underline{u}}) \ll 1$ since we make the assumption of small displacements. Next, we use the definition of the time derivative of the fluid's density [8]:

$$\dot{\rho}_f = \rho_f \left(\frac{1}{K_l} \dot{p} - 3\alpha_l \dot{T} \right) \quad (7)$$

We now turn our attention to the evolution of the porosity. By using the definition of the Eulerian porosity φ related to the Lagrangian porosity by $\phi = (1 + \operatorname{div}(\underline{\underline{u}}))\varphi$, its expression

$\phi = \text{div}(\underline{u}) + (1 - \varphi)\frac{p}{K_s} + 3(1 - \varphi)\alpha_s T$ in the THM context [34] and the incompressibility of the solid matrix, we have:

$$\begin{aligned}\dot{\phi} &= (1 - \varphi)(\text{div}(\underline{\dot{u}}) - 3\alpha_s \dot{T} + \frac{\dot{p}}{K_s}) \\ &= (1 - \varphi)(\text{div}(\underline{\dot{u}}) - 3\alpha_s \dot{T})\end{aligned}\quad (8)$$

The fluid mass supply is then

$$\begin{aligned}\dot{m}_f &= \rho_f(\varphi \text{div}(\underline{\dot{u}}) + \frac{\varphi}{K_l}\dot{p} - 3\alpha_l \varphi \dot{T} + (1 - \varphi)(\text{div}(\underline{\dot{u}}) - 3\alpha_s \dot{T})) \\ &= \rho_f(\text{div}(\underline{\dot{u}}) + \frac{\varphi}{K_l}\dot{p} - 3\alpha_m \dot{T})\end{aligned}$$

where $3\alpha_m = (3\alpha_l \varphi + 3(1 - \varphi)\alpha_s)$.

Finally, we take into consideration Darcy's law where the effect of gravity is neglected in coherence with the targeted application (the theory does not need this assumption which is just a short simplification)

$$\underline{\psi} = -\rho_f \lambda_H \nabla p$$

The final water mass conservation Eq. (5) is then

$$\rho_f(\text{div}(\underline{\dot{u}}) + \frac{\varphi}{K_l}\dot{p} - 3\alpha_m \dot{T}) - \text{div}(\rho_f \lambda_H \nabla p) = 0 \quad (9)$$

The energy conservation

The energy conservation equation is

$$h_f \dot{m}_f + \text{div}(h_f \underline{\psi}) + \text{div}(\underline{q}) + \dot{Q}' = \Theta \quad (10)$$

Θ denotes the total sources of heat and it equals the four terms on the left-hand side: the heat coming from the fluid enthalpy, the energy convected by the fluid, the heat flux and the non-convective heat. We shall start by detailing the latter.

The non-convective heat Q' is the thermal input received by the system excluding the enthalpy contribution of the fluid. It is the sum of three terms of heat input due respectively to the deformation of the solid matrix, to the fluid compression and to temperature variation. It is a non-linear term whose expression is:

$$\dot{Q}' = 3K_0 \alpha_s \text{div}(\underline{\dot{u}})T - 3\alpha_l \dot{p}T + C_\epsilon^0 \dot{T}$$

By developing the specific heat of the medium to constant deformation, we get $C_\epsilon^0 = C_\sigma^0 - 9TK_0 \alpha_s^2$ [35, p78].

$$\dot{Q}' = (3K_0 \alpha_s \text{div}(\underline{\dot{u}}) - 3\alpha_l \dot{p} - 9K_0 \alpha_s^2 \dot{T})T + C_\sigma^0 \dot{T}$$

By replacing \dot{Q}' , \dot{m}_f , $\underline{\psi}$ and using the fact that the heat diffusion follows Fourier's law $\underline{q} = -\lambda_T \nabla T$, Eq. (10) becomes

$$\begin{aligned}\rho_f h_f \left(\text{div}(\underline{\dot{u}}) + \frac{\varphi}{K_l}\dot{p} - 3\alpha_m \dot{T} \right) - \text{div}(\rho_f h_f \lambda_H \nabla p) \\ + (3K_0 \alpha_s \text{div}(\underline{\dot{u}}) - 3\alpha_l \dot{p} - 9K_0 \alpha_s^2 \dot{T})T + C_\sigma^0 \dot{T} - \text{div}(\lambda_T \nabla T) = \Theta\end{aligned}$$

The final system

After detailing each balance equation in order to reveal the detailed coupling between the phenomena involved, the final system is obtained.

Let Ω be a d dimensional domain, $1 \leq d \leq 3$, and t_f the final time of the simulation. The THM model describes the evolution of 3 primal unknowns: the vector displacement field, $\underline{u}(\underline{x}, t)$, the fluid pressure field, $p(\underline{x}, t)$, the temperature field $T(\underline{x}, t)$.

The coupled system consists of, $\forall \underline{x} \in \Omega$ and $\forall t > 0 \in [0, t_f]$:

$$\begin{aligned} -\operatorname{div}(\underline{\underline{A}} : \underline{\underline{\varepsilon}}(\underline{u})) + \nabla p + 3K_s \alpha_s \nabla T &= \underline{f}^e & \text{in } \Omega \times (0, t_f) \\ -\operatorname{div}(\rho_f \lambda_H \nabla p) + \rho_f (\operatorname{div}(\underline{\dot{u}}) + \frac{\varphi}{K_l} \dot{p} - \alpha_m 3\dot{T}) &= 0 & \text{in } \Omega \times (0, t_f) \\ -\operatorname{div}(\lambda_T \nabla T) - \operatorname{div}(\rho_f h_f \lambda_H \nabla p) \\ + \rho_f h_f (\operatorname{div}(\underline{\dot{u}}) + \frac{\varphi}{K_l} \dot{p} - \alpha_m 3\dot{T}) \\ + (3K_0 \alpha_s \operatorname{div}(\underline{\dot{u}}) - 3\alpha_m \dot{p} - 9K_0 \alpha_s^2 \dot{T}) + C_\sigma^0 \dot{T} &= \Theta & \text{in } \Omega \times (0, t_f) \end{aligned}$$

The boundary of Ω is denoted $\partial\Omega$ and six different partitions are needed to define the boundary conditions. For each primal unknown, we may define Dirichlet and Neumann boundary conditions, say the displacement \underline{u} and the stress $\underline{\underline{\sigma}}$, the pressure p and the fluid flux q , the temperature T and the thermal flux Ψ .

We thus have, respectively, the boundary conditions on the displacement unknowns, on the pressure unknowns and on the temperature unknowns such as:

$$\begin{aligned} \partial\Omega &= \partial\Omega^{\underline{u}} \cup \partial\Omega^{\underline{\underline{\sigma}}} \text{ with } \partial\Omega^{\underline{u}} \cap \partial\Omega^{\underline{\underline{\sigma}}} = \emptyset \\ \partial\Omega &= \partial\Omega^p \cup \partial\Omega^q \text{ with } \partial\Omega^p \cap \partial\Omega^q = \emptyset \\ \partial\Omega &= \partial\Omega^T \cup \partial\Omega^\Psi \text{ with } \partial\Omega^T \cap \partial\Omega^\Psi = \emptyset \end{aligned}$$

The boundary and initial conditions are given by:

$$\begin{aligned} \underline{\underline{\sigma}}(\underline{u}) \cdot \underline{n} &= \underline{t}^e & \text{on } \partial\Omega^{\underline{\underline{\sigma}}} \times (0, t_f) \\ -\lambda_H \nabla p \cdot \underline{n} &= q^e & \text{on } \partial\Omega^q \times (0, t_f) \\ -\lambda_T \nabla T \cdot \underline{n} &= \Psi^e & \text{on } \partial\Omega^\Psi \times (0, t_f) \\ \underline{u} &= \underline{u}^e & \text{on } \partial\Omega^{\underline{u}} \times (0, t_f) \\ p &= p^e & \text{on } \partial\Omega^p \times (0, t_f) \\ T &= T^e & \text{on } \partial\Omega^T \times (0, t_f) \\ \underline{u}(\underline{x}, 0) &= \underline{u}_0(\underline{x}) & \text{in } \Omega \\ p(\underline{x}, 0) &= p_0(\underline{x}) & \text{in } \Omega \\ T(\underline{x}, 0) &= T_0(\underline{x}) & \text{in } \Omega \end{aligned}$$

where \underline{n} is the outward normal.

Furthermore, the material parameters' definitions are given in Table 1.

Linearization and discretization

The next step to solve the non-linear time-dependent THM system is to do the time and space discretization, as well as the linearization.

Variational formulation

We define the Sobolev spaces

$$\begin{aligned}\mathcal{U}(\Omega) &= \{\underline{u} \in (H^1(\Omega))^d, \underline{u} = \underline{u}^e \text{ on } \partial\Omega^{\underline{u}}\}, \\ \mathcal{P}(\Omega) &= \{p \in H^1(\Omega), p = p^e \text{ on } \partial\Omega^p\}, \\ \mathcal{T}(\Omega) &= \{T \in H^1(\Omega), T = T^e \text{ on } \partial\Omega^T\},\end{aligned}$$

By considering the appropriate Sobolev spaces defined above and by integration by parts, we have the following weak form:

Find $(\underline{u}, p, T) \in \mathcal{U}(\Omega) \times \mathcal{P}(\Omega) \times \mathcal{T}(\Omega)$ such as for all $(\underline{v}, q, W) \in \mathcal{U}(\Omega) \times \mathcal{P}(\Omega) \times \mathcal{T}(\Omega)$, we have

$$\begin{aligned}\int_{\Omega} \left(-\underline{\underline{A}} : \underline{\underline{\varepsilon}}(\underline{u}) : \underline{\underline{\varepsilon}}(\underline{v}) + p \operatorname{div}(\underline{v}) + 3K_s \alpha_s T \operatorname{div}(\underline{v}) \right) dx &= \int_{\Omega} f^e \underline{v} dx + \int_{\partial\Omega^t} \underline{t}^e \underline{v} ds \\ \int_{\Omega} \rho_f (-\lambda_H \nabla p \nabla q + \operatorname{div}(\underline{u}) q + \frac{\varphi}{K_l} \dot{p} q - \alpha_m 3 \dot{T} q) dx &= \int_{\partial\Omega^q} \rho_f q^e q ds \\ \int_{\Omega} (-\lambda_T \nabla T \nabla W + C_{\sigma}^0 \dot{T} W &+ \rho_f h_f (-\lambda_H \nabla p \nabla W + \operatorname{div}(\underline{u}) W + \frac{\varphi}{K_l} \dot{p} W - \alpha_m 3 \dot{T} W) \\ &+ (T 3K_0 \alpha_s \operatorname{div}(\underline{u}) W - 3\alpha_m \dot{p} W - 9K_0 \alpha_s^2 \dot{T} W) dx = \int_{\Omega} \Theta W dx + \int_{\partial\Omega^{\Psi}} \Psi^e W ds\end{aligned}\quad (11)$$

Time discretization

To solve the THM time-dependent problem, we chose an implicit Euler method to discretize the problem in time and solve a static problem at each time step.

We use the notations $\underline{u}^n(x) := \underline{u}(x, t^n)$, $p^n(x) := p(x, t^n)$ and $T^n(x) := T(x, t^n)$ which denote the displacement field, the pressure field and the temperature field at $t^n = n\Delta t$, where Δt is a given time increment.

To apply Euler's implicit method, $\partial_t \underline{u}$ (equivalently for \dot{p} and \dot{T}) is replaced by:

$$\underline{\dot{u}}(x, t^{n+1}) = \frac{\underline{u}(x, t^{n+1}) - \underline{u}(x, t^n)}{t^{n+1} - t^n} = \frac{\underline{u}^{n+1}(x) - \underline{u}^n(x)}{\Delta t}\quad (12)$$

The THM semi-discrete weak formulation becomes

$$\begin{aligned}\int_{\Omega} \left(\underline{\underline{A}} : \underline{\underline{\varepsilon}}(\underline{u}^{n+1}) : \underline{\underline{\varepsilon}}(\underline{v}) - \operatorname{div}(\underline{v}) p^{n+1} - 3K_s \alpha_s \operatorname{div}(\underline{v}) T^{n+1} \right) dx &= \int_{\Omega} f^e \underline{v} dx + \int_{\partial\Omega^t} \underline{t}^e \underline{v} ds \\ \int_{\Omega} \rho_f \operatorname{div}(\underline{u}^{n+1}) q + \frac{\varphi}{K_l} p^{n+1} q - \alpha_m 3 T^{n+1} q + \Delta t \lambda_H \nabla p^{n+1} \nabla q dx &= \int_{\partial\Omega^q} \rho_f q^e q ds + \int_{\Omega} \rho_f \left(\operatorname{div}(\underline{u}^n) q + \frac{\varphi}{K_l} p^n q - \alpha_m 3 T^n q \right) dx\end{aligned}$$

$$\begin{aligned}
 & \int_{\Omega} -\Delta t \lambda_T \nabla T^{n+1} \nabla W + C_{\sigma}^0 T^{n+1} W \\
 & + \rho_f h_f (-\Delta t \lambda_H \nabla p^{n+1} \nabla W + \operatorname{div}(\underline{u}^{n+1}) W + \frac{\varphi}{K_l} p^{n+1} W - \alpha_m 3 T^{n+1} W) \\
 & + T^{n+1} (3K_0 \alpha_s \operatorname{div}(\underline{u}^{n+1}) W - 3\alpha_m p^{n+1} W - 9K_0 \alpha_s^2 T^{n+1} W) dx \\
 & = \Delta t \int_{\Omega} \Theta W dx + \int_{\partial \Omega^{\Psi}} \Psi^e W d \\
 & + \int_{\Omega} C_{\sigma}^0 T^n W + \rho_f h_f \left(\operatorname{div}(\underline{u}^n) W + \frac{\varphi}{K_l} p^n W - \alpha_m 3 T^n W \right) \\
 & + T^n (3K_0 \alpha_s \operatorname{div}(\underline{u}^n) W - 3\alpha_m p^n W - 9K_0 \alpha_s^2 T^n W) dx
 \end{aligned}$$

Linearization and Newton's method

The system is linearized using Newton's method and requires the partial derivatives of each equation residual with respect to \underline{u}^{n+1} , p^{n+1} and T^{n+1} .

Let's introduce the residual notation for the displacement

$$\begin{aligned}
 R_{\underline{u}} := & \int_{\Omega} (\underline{A} : \underline{\varepsilon}(\underline{u}^{n+1}) : \underline{\varepsilon}(\underline{v}) - \operatorname{div}(\underline{v}) p^{n+1} - 3K_s \alpha_s \operatorname{div}(\underline{v}) T^{n+1}) dx \\
 & - \int_{\Omega} f^e \underline{v} dx + \int_{\partial \Omega^{\ell}} \underline{t}^e \underline{v} ds
 \end{aligned}$$

where $R_{\underline{u}}$ is a function of $((\underline{u}^{n+1}, p^{n+1}, T^{n+1}), (\underline{u}^n, p^n, T^n), \underline{v})$. The same is done for the pressure residual R_p and the temperature residual R_T .

Newton's method requires to find a correction $(\delta_{\underline{u}}, \delta_p, \delta_T)$ solution of

$$\mathbf{J} \begin{bmatrix} \delta_{\underline{u}} \\ \delta_p \\ \delta_T \end{bmatrix} = \begin{bmatrix} \frac{\partial R_{\underline{u}}}{\partial \underline{u}} & \frac{\partial R_{\underline{u}}}{\partial p} & \frac{\partial R_{\underline{u}}}{\partial T} \\ \frac{\partial R_p}{\partial \underline{u}} & \frac{\partial R_p}{\partial p} & \frac{\partial R_p}{\partial T} \\ \frac{\partial R_T}{\partial \underline{u}} & \frac{\partial R_T}{\partial p} & \frac{\partial R_T}{\partial T} \end{bmatrix} \begin{bmatrix} \delta_{\underline{u}} \\ \delta_p \\ \delta_T \end{bmatrix} = - \begin{bmatrix} R_{\underline{u}} \\ R_p \\ R_T \end{bmatrix} \quad (13)$$

where \mathbf{J} is the residual's Jacobian.

The solution is then updated,

$$\underline{u}_k^{n+1} = \underline{u}_{k-1}^{n+1} + \delta_{\underline{u}} \quad (14)$$

$$p_k^{n+1} = p_{k-1}^{n+1} + \delta_p \quad (15)$$

$$T_k^{n+1} = T_{k-1}^{n+1} + \delta_T \quad (16)$$

until the stopping criterion is reached $\frac{\|\underline{r}_k\|}{\|\underline{r}_0\|} < 10^{-6}$ where $\underline{\delta} = \begin{bmatrix} \delta_{\underline{u}} \\ \delta_p \\ \delta_T \end{bmatrix}$ and $\underline{r}_k = \begin{bmatrix} R_{\underline{u}} \\ R_p \\ R_T \end{bmatrix}$ at

the k th iteration.

In order to solve the system (13), we need to find \mathbf{J} by linearizing the three residuals.

Since the first two equations of the system are linear, the first two equations of the linearized system are

$$\begin{aligned}
 & \frac{\partial R_{\underline{u}}}{\partial \underline{u}^{n+1}} \delta_{\underline{u}} + \frac{\partial R_{\underline{u}}}{\partial p^{n+1}} \delta_p + \frac{\partial R_{\underline{u}}}{\partial T^{n+1}} \delta_T \\
 & = \int_{\Omega} \left(\underline{A} : \underline{\varepsilon}(\delta_{\underline{u}}) : \underline{\varepsilon}(\underline{v}) - \operatorname{div}(\underline{v}) \delta_p - 3K_s \alpha_s \operatorname{div}(\underline{v}) \delta_T \right) dx \quad (17)
 \end{aligned}$$

$$\begin{aligned} & \frac{\partial R_p}{\partial \underline{u}^{n+1}} \delta \underline{u} + \frac{\partial R_p}{\partial p^{n+1}} \delta p + \frac{\partial R_p}{\partial T^{n+1}} \delta T \\ &= \int_{\Omega} \rho_f (\operatorname{div}(\delta \underline{u})) q + \frac{\varphi}{K_l} \delta p q - \alpha_m 3 \delta T q + \Delta t \lambda_H \nabla \delta p \nabla q) dx \end{aligned} \quad (18)$$

We linearize each term of the third equation using $\frac{\partial h_f}{\partial p} = \frac{(1-3\alpha_l T)}{\rho_f}$ and $\frac{\partial h_f}{\partial T} = C_f^p$ [35,36].

$$\begin{aligned} \frac{\partial R_T}{\partial \underline{u}^{n+1}} \delta \underline{u} &= \int_{\Omega} (\rho_f h_f \operatorname{div}(\delta \underline{u})) W + T_{k-1}^{n+1} 3K_0 \alpha_s \operatorname{div}(\delta \underline{u}) W) dx \\ \frac{\partial R_T}{\partial p^{n+1}} \delta p &= \int_{\Omega} (1 - 3\alpha_l T_{k-1}^{n+1}) \delta p \left(-\Delta t \lambda_H \nabla p_{k-1}^{n+1} \nabla W + \operatorname{div}(\underline{u}_{k-1}^{n+1}) W \right. \\ &\quad \left. + \frac{\varphi}{K_l} p_{k-1}^{n+1} W - \alpha_m 3 T_{k-1}^{n+1} W \right) \\ &\quad + \rho_f h_f \left(-\Delta t \lambda_H \nabla \delta p \nabla W + \frac{\varphi}{K_l} \delta p W \right) - T_{k-1}^{n+1} 3\alpha_m \delta p W dx \\ \frac{\partial R_T}{\partial T^{n+1}} \delta T &= \int_{\Omega} -\Delta t \lambda_T \nabla \delta T \nabla W + C_{\sigma}^0 \delta T W \\ &\quad + \rho_f C_f^p \delta T \left(-\Delta t \lambda_H \nabla p_{k-1}^{n+1} \nabla W + \operatorname{div}(\underline{u}_{k-1}^{n+1}) W \right. \\ &\quad \left. + \frac{\varphi}{K_l} p_{k-1}^{n+1} W - \alpha_m 3 T_{k-1}^{n+1} W \right) \\ &\quad - \rho_f h_f (\alpha_m 3 \delta T W) + \delta T \left(3K_0 \alpha_s \operatorname{div}(\underline{u}_{k-1}^{n+1}) W \right. \\ &\quad \left. - 3\alpha_m p_{k-1}^{n+1} W - 18K_0 \alpha_s^2 T_{k-1}^{n+1} W \right) dx \end{aligned}$$

Space discretization

The finite element method is used for space discretization and Taylor-Hood P2-P1-P1 finite elements are considered. This translates into using continuous piecewise quadratic polynomials to approximate the displacement and continuous piecewise linear polynomials to approximate the pressure and the temperature. In [37], these elements were studied for poroelasticity and having the polynomial interpolation for the displacement be one degree higher than for the pressure, equilibrates the convergence rate of all terms in the energy norm. Furthermore the convergence is robust with respect to the mesh size. As mentioned in the introduction, this choice has also shown to be stable on several severe tests [24].

The choice of the P1 interpolation of the temperature relies on the fact that this field is directly used in (3) for the evaluation of the thermal expansion of the medium. Since the latter is subtracted to the mechanical strain computed as the symmetric gradient of the displacement, this choice ensures consistency of the interpolations and avoid non-physical artefacts in the case where the temperature is interpolated with the same shape functions as the displacement, as pointed out in [38, p.104].

Let $U_h(\Omega)$ be the discrete Sobolev subspace of $\mathcal{U}(\Omega)$ of dimension N_u with $h > 0$ being a parameter that refers to the mesh size. The same formulation is used for $P_h(\Omega)$ and $\mathcal{T}_h(\Omega)$.

Let $\{\phi_{v_j}\}_{j=1}^{N_u}$ be a basis for the finite element space $U_h(\Omega)$, then for all $v_h \in U_h(\Omega)$ we have

$$v_h = \sum_{j=1}^{N_u} v_h \phi_{v_j}.$$

Let $\{\phi_{q_j}\}_{j=1}^{N_p}$ be a basis for the finite element space $P_h(\Omega)$, then for all $q_h \in U_h(\Omega)$ we have

$$q_h = \sum_{j=1}^{N_p} q_h \phi_{q_j}.$$

Let $\{\phi_{w_j}\}_{j=1}^{N_T}$ be a basis for the finite element space $\mathcal{T}_h(\Omega)$. Then for all $W_h \in \mathcal{T}_h(\Omega)$ we have

$$W_h = \sum_{j=1}^{N_T} W_h \phi_{w_j}.$$

The discrete problem is: find $(\delta_{u_h}, \delta_{p_h}, \delta_{T_h}) \in U_h(\Omega) \times P_h(\Omega) \times \mathcal{T}_h(\Omega)$ such that for all $(v_h, q_h, W_h) \in U_h(\Omega) \times P_h(\Omega) \times \mathcal{T}_h(\Omega)$ holds

$$\mathbf{J} \begin{bmatrix} \delta_{u_h} \\ \delta_{p_h} \\ \delta_{T_h} \end{bmatrix} = - \begin{bmatrix} R_{u_h} \\ R_{p_h} \\ R_{T_h} \end{bmatrix} \quad (19)$$

where the matrix blocks are detailed in Appendix A.

Preconditioning

Iterative solvers often suffer from bad conditioning of the linear system matrix and require preconditioning to achieve satisfactory performance in terms of iteration count and simulation time. This is especially true for THM problems, that are in general ill-conditioned due to the properties of each physical component which are included via parameters into the linear system and the right-hand side. In this section, we first discuss this issue and define a preconditioner tailored for our application.

Multiphysics preconditioners for THM

As explained in the derivation above, the linear system to be solved is of the structure

$$\begin{bmatrix} \mathbf{J}_{uu} & \mathbf{J}_{up} & \mathbf{J}_{uT} \\ \mathbf{J}_{pu} & \mathbf{J}_{pp} & \mathbf{J}_{pT} \\ \mathbf{J}_{Tu} & \mathbf{J}_{Tp} & \mathbf{J}_{TT} \end{bmatrix} \begin{bmatrix} \delta_{u_h} \\ \delta_{p_h} \\ \delta_{T_h} \end{bmatrix} = - \begin{bmatrix} R_{u_h} \\ R_{p_h} \\ R_{T_h} \end{bmatrix}. \quad (20)$$

As it is often the case for monolithic coupled formulations [23], this system is ill-conditioned. Furthermore, there are significant differences in the order of magnitudes of each parameter (see Table 3 in the sequel for illustrative values) which translates into different orders of magnitudes between the 2-norms of each physics-based block in the matrix in (20)

$$\mathbf{S} := \begin{bmatrix} \|\mathbf{J}_{uu}\|_2 & \|\mathbf{J}_{up}\|_2 & \|\mathbf{J}_{uT}\|_2 \\ \|\mathbf{J}_{pu}\|_2 & \|\mathbf{J}_{pp}\|_2 & \|\mathbf{J}_{pT}\|_2 \\ \|\mathbf{J}_{Tu}\|_2 & \|\mathbf{J}_{Tp}\|_2 & \|\mathbf{J}_{TT}\|_2 \end{bmatrix} \approx \begin{bmatrix} 1. \text{e}+13 & 1. \text{e}+01 & 1. \text{e}+06 \\ 1. \text{e}+04 & 1. \text{e}-08 & 1. \text{e}-02 \\ 1. \text{e}+08 & 1. \text{e}-03 & 1. \text{e}+05 \end{bmatrix}. \quad (21)$$

We have a maximal scaling difference of 10^{21} between the displacement and pressure blocks. Solving this system naively could lead to cancellation effects in the solution. Prior scaling or preconditioning of the matrix are thus compulsory.

Matrix scaling

Let us first look into matrix scaling and follow the algorithm given in [39]. The main features of this algorithm are that the scaled matrix becomes diagonally dominant and the scaling becomes block-symmetric. For these kinds of matrices, iterative solvers often converge more easily. The scaling algorithm takes the 3×3 matrix \mathbf{S} in (21) and computes the following block-diagonal matrices

$$\mathbf{D}_r = \begin{bmatrix} 10^{-7} \mathbf{I}_{N_u} & 0 & 0 \\ 0 & 10^{-2} \mathbf{I}_{N_p} & 0 \\ 0 & 0 & 10^{-4} \mathbf{I}_{N_T} \end{bmatrix}, \mathbf{D}_l = \begin{bmatrix} 10^{-8} \mathbf{I}_{N_u} & 0 & 0 \\ 0 & 10^4 \mathbf{I}_{N_p} & 0 \\ 0 & 0 & 10^{-2} \mathbf{I}_{N_T} \end{bmatrix}$$

where \mathbf{I}_{N_u} , \mathbf{I}_{N_p} , \mathbf{I}_{N_T} are the identity matrices of size N_u , N_p , N_T . The THM system is then scaled using \mathbf{D}_r and \mathbf{D}_l by

$$\mathbf{J}^{sc} = \mathbf{D}_r \mathbf{J} \mathbf{D}_l \mathbf{x}_s = \mathbf{b}_s$$

with $\mathbf{x}_s = \mathbf{D}_l^{-1} \mathbf{x}$ and $\mathbf{b}_s = \mathbf{D}_r^{-1} \mathbf{b}$. The entries in the scaled system are now of the following magnitudes

$$\begin{bmatrix} \|\mathbf{J}_{uu}^{sc}\|_2 & \|\mathbf{J}_{up}^{sc}\|_2 & \|\mathbf{J}_{uT}^{sc}\|_2 \\ \|\mathbf{J}_{pu}^{sc}\|_2 & \|\mathbf{J}_{pp}^{sc}\|_2 & \|\mathbf{J}_{pT}^{sc}\|_2 \\ \|\mathbf{J}_{Tu}^{sc}\|_2 & \|\mathbf{J}_{Tp}^{sc}\|_2 & \|\mathbf{J}_{TT}^{sc}\|_2 \end{bmatrix} \approx \begin{bmatrix} 1. \text{e}-01 & 1. \text{e}-01 & 1. \text{e}-03 \\ 1. \text{e}-01 & 1. \text{e}-01 & 1. \text{e}-02 \\ 1. \text{e}-03 & 1. \text{e}-02 & 1. \text{e}-01 \end{bmatrix}.$$

Definition of block preconditioners

Note that the matrices \mathbf{J} and \mathbf{J}^{sc} are non-symmetric. We thus need an iterative solver for non-symmetric systems and choose the *flexible GMRES* (FGMRES) method [40]. We will next define a preconditioner that can be applied to the scaled or unscaled system. The idea behind preconditioning is to construct a matrix \mathbf{P} that is a good enough approximation of \mathbf{J} but that is easily invertible. In our computations, the preconditioner is applied from the right, which means that we solve the system

$$\mathbf{J}\mathbf{P}^{-1}\mathbf{y} = \mathbf{r},$$

with $\mathbf{y} = \mathbf{P}\mathbf{x}$. The solution \mathbf{x} of the system remains the same but if \mathbf{P} is a good approximation of \mathbf{J} , then $\mathbf{J}\mathbf{P}^{-1}$ becomes 'closer' to the identity and the iterative method will converge faster than for the unpreconditioned system.

The linear system in (20) is of block structure, where each diagonal block corresponds to one of the three physical models. It thus seems natural to choose a block preconditioner,

as it has been for example described in the reference [41]. The simplest preconditioner is probably the block Jacobi preconditioner given by

$$\mathbf{P}_{Jac} = \begin{bmatrix} \mathbf{J}_{uu} & 0 & 0 \\ 0 & \mathbf{J}_{pp} & 0 \\ 0 & 0 & \mathbf{J}_{TT} \end{bmatrix}.$$

The application of a standard Jacobi preconditioner is simple, as it is trivial to invert a diagonal matrix. For the block Jacobi preconditioner we need the inverses of the three separate individual physics blocks, i.e. \mathbf{J}_{uu}^{-1} , \mathbf{J}_{pp}^{-1} and \mathbf{J}_{TT}^{-1} . This is costly and thus we search for a good approximation of each block, that can be more easily inverted. Before we discuss this further, we introduce our second and third choice for a preconditioner. These are the lower and upper block Gauss-Seidel preconditioners, denoted by \mathbf{P}_{LGS} and \mathbf{P}_{UGS} , respectively, given by

$$\mathbf{P}_{LGS} = \begin{bmatrix} \mathbf{J}_{uu} & 0 & 0 \\ \mathbf{J}_{pu} & \mathbf{J}_{pp} & 0 \\ \mathbf{J}_{Tu} & \mathbf{J}_{Tp} & \mathbf{J}_{TT} \end{bmatrix}, \quad \mathbf{P}_{UGS} = \begin{bmatrix} \mathbf{J}_{uu} & \mathbf{J}_{up} & \mathbf{J}_{uT} \\ 0 & \mathbf{J}_{pp} & \mathbf{J}_{pT} \\ 0 & 0 & \mathbf{J}_{TT} \end{bmatrix}$$

Even though these preconditioners use the rectangular lower (or upper) triangular blocks of the system, when applying their inverse we still only need to compute the inverses of the three diagonal blocks \mathbf{J}_{uu}^{-1} , \mathbf{J}_{pp}^{-1} and \mathbf{J}_{TT}^{-1} . Let \mathbf{I} be the identity matrix of appropriate size for each block. For ease of notation, we do not add the size in the index. The inverse of the lower Gauss-Seidel preconditioner is given by

$$\mathbf{P}_{LGS}^{-1} = \begin{bmatrix} \mathbf{I} & 0 & 0 \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{J}_{TT}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{I} & 0 & 0 \\ 0 & \mathbf{I} & 0 \\ -\mathbf{J}_{Tu} & -\mathbf{J}_{Tp} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & 0 & 0 \\ 0 & \mathbf{J}_{pp}^{-1} & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix} \\ \begin{bmatrix} \mathbf{I} & 0 & 0 \\ -\mathbf{J}_{pu} & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{J}_{uu}^{-1} & 0 & 0 \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix}$$

and the inverse of \mathbf{P}_{UGS} is given by

$$\mathbf{P}_{UGS}^{-1} = \begin{bmatrix} \mathbf{J}_{uu}^{-1} & 0 & 0 \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & -\mathbf{J}_{up} & -\mathbf{J}_{uT} \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & 0 & 0 \\ 0 & \mathbf{J}_{pp}^{-1} & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix} \\ \begin{bmatrix} \mathbf{I} & 0 & 0 \\ 0 & \mathbf{I} & -\mathbf{J}_{pT} \\ 0 & 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & 0 & 0 \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{J}_{TT}^{-1} \end{bmatrix}$$

As mentioned above, for each one of these three preconditioners, we need, in theory, the inverse matrices of the diagonal blocks. In practice, these inverses are never explicitly computed, as this is too costly. Incomplete factorisations of the original matrices are for example well suited but lack scalability. Furthermore, it is not necessary to have an explicit or precomputed representation of the preconditioner. In iterative methods like for example Conjugate Gradient or FGMRES, it is indeed enough to apply the preconditioner

in form of matrix–vector products. It is for example possible to completely replace a particular inverse approximation by a linear process. In our later numerical experiments, we will use one V-cycle of an algebraic multigrid solver (AMG) as preconditioner for each one of the three diagonal block matrices [42]. Furthermore, the preconditioner can be replaced by another iterative method, for example GMRES itself, but then the preconditioner becomes non-linear. In this case, we need *flexible* versions of the iterative solvers, as FGMRES, that allow a variable preconditioner at each iteration. A nested approach can also be used, where the preconditioner is implemented in form of some iterations of an iterative solver that is preconditioned itself (see [43] for extreme-scale applications of nested Krylov methods). In any case, the choice of each approximation will ultimately come down to the specific characteristics of the block to invert. In the following experiments, we use a nested preconditioning approach, where we apply the block-preconditioners \mathbf{P}_{Jac} , \mathbf{P}_{LGS} , \mathbf{P}_{UGS} by using some iterations of the FGMRES method preconditioned by one V-cycle of AMG for each block \mathbf{J}_{uu} , \mathbf{J}_{pp} , \mathbf{J}_{TT} . This provides the possibility to control the quality of the inverse for each block by defining a stopping tolerance or a fixed number of FGMRES iterations. In this strategy, we have an interplay between the number of outer iterations of FGMRES on the block system with the number of inner FGMRES iterations applied to each diagonal block. A stricter tolerance for the inner FGMRES solvers might lead to a smaller number of outer FGMRES iterations and vice-versa. This choice is guided by the numerical experiments described in the following, where the trade-off between performance and robustness was a primary goal.

The choice of an AMG preconditioner for each inner FGMRES iteration can be explained as follows. The diagonal blocks of the discretized system (see Sect. “Time discretization”) involve elliptic and non-degenerate parabolic operators, for which V-cycle multigrid preconditioners are especially suited [13,41]. Note that in our particular case, we could as well use the GMRES method, since one V-cycle is a constant linear preconditioner and thus does not require a flexible version. This would come with a small memory gain.

Numerical experiments for scaling issues

In this section, we present numerical results for the above defined nested solvers for the scaled and unscaled linear system. The tolerance of the outer FGMRES solver is set to $\epsilon = 10^{-6}$. This rather large tolerance is used since the linear system is the linearized problem in a Newton fixed point iteration. The Newton iterations are required to converge at a tolerance of $\epsilon_N = 10^{-6}$, so that a stricter tolerance ϵ would be more costly than useful. For the nested preconditioner, we use FGMRES preconditioned by one V-cycle of AMG. We have found empirically that using a fixed number of 10 iterations for the displacement block, and 3 iterations for the pressure and temperature blocks gives a good compromise between the cost of inner and outer iterations with respect to the global computation time. We use the algebraic multigrid solver BoomerAMG from the hypre library through PETSc with its default parameters [44].

Since an analytic solution is not available for the test problem, we solve the scaled system as precise as possible by using the sparse direct solver MUMPS [45] and use this solution as reference solution. Before computing the errors, the solution x_s of the scaled system is

brought back to the original scaling x using the formula $x = D_I x_s$. We use the following notations for the different solution strategies:

- xs_0 : direct solution of the scaled system,
- xs : Iterative solver solution of the scaled system,
- x : Iterative solver solution of the initial system.

The relative error with respect to the reference solution $D_I xs_0$ is computed for each physical unknown. With obvious notation, the three errors are computed as follows:

- for x

$$\text{err}_u = \frac{\|D_I xs_0 - x_u\|_2}{\|D_I xs_0\|_2} \quad \text{err}_p = \frac{\|D_I xs_0 - x_p\|_2}{\|D_I xs_0\|_2} \quad \text{err}_T = \frac{\|D_I xs_0 - x_T\|_2}{\|D_I xs_0\|_2}$$

- for xs

$$\text{err}_u = \frac{\|D_I xs_0 - D_I xs_u\|_2}{\|D_I xs_0\|_2} \quad \text{err}_p = \frac{\|D_I xs_0 - D_I xs_p\|_2}{\|D_I xs_0\|_2} \quad \text{err}_T = \frac{\|D_I xs_0 - D_I xs_T\|_2}{\|D_I xs_0\|_2}$$

We present the simulation time, the number of outer FGMRES iterations and the above defined errors for each one of the three preconditioner P_{Jac} , P_{LGS} , P_{UGS} in Table 2. Comparing the effect of the scaling on the solution, there is no clear winner. Indeed, the errors for each unknown are less variable across the preconditioners for the scaled system. In case of the P_{UGS} , scaling is even compulsory. Here, solving the unscaled system does not lead to satisfactory results in the displacement and pressure variables. This can be explained by the difference in order of magnitudes in the entries of the matrix blocks and the order in which these are applied in the solution process. In terms of iteration count, FGMRES shows mesh independent convergence for P_{LGS} and we expect this behavior also for the Jacobi preconditioner once the problem size is further increased. In general, FGMRES needs fewer iterations to reduce the residual below the required tolerance for the scaled system. This leads however to higher errors in (almost) each variable when the matrix is preconditioned by P_{Jac} and P_{LGS} . The lower iteration count thus does not necessarily present an advantage when interested in the actual error and not the residual.

This numerical experiment suggests that the use of P_{Jac} and P_{LGS} as preconditioners on the unscaled system results in a precise and robust solution strategy. The latter is therefore used in the robustness and scalability studies in the following.

Solver performance

The robustness and efficiency of the proposed solver are crucial for industrial applications. We thus present an illustrative test case, challenging the preconditioner's robustness by varying some parameters. The parallel efficiency is also evaluated by weak and strong scalability tests. The method is implemented in code_aster, the massively parallel open source general purpose finite element analysis software developed at EDF R&D [46].

Model problem

The test case needs to be simple enough so that the mesh can be easily refined but complex enough to resemble the industrial problem in consideration.

Table 2 Error analysis

		Errors using the scaled system: xs				
	Size	Time	it	err_u	err_p	err_T
P_J	429	1.40e−01	15	3.71e−05	2.96e−04	6.44e−09
	2437	4.69e−01	15	3.78e−05	2.55e−04	9.82e−09
	117,637	4.55e+01	15	1.34e−04	1.39e−04	7.01e−08
P_{LGS}	429	1.13e−01	9	7.41e−06	3.47e−04	1.08e−09
	2437	3.53e−01	10	1.64e−05	2.19e−04	1.83e−08
	117,637	2.95e+01	10	2.31e−04	1.40e−04	1.83e−08
P_{UGS}	429	1.17e−01	9	1.49e−05	8.85e−05	1.46e−10
	2437	4.28e−01	13	1.51e−06	8.24e−06	1.57e−09
	117,637	3.24e+01	12	4.26e−06	3.40e−05	6.29e−09
		Errors using the initial system: x				
	Size	Time	it	err_u	err_p	err_T
P_J	429	1.66e−01	18	1.19e−03	1.44e−03	1.13e−07
	2437	1.02e+00	34	6.24e−05	4.74e−05	3.79e−08
	117,637	1.15e+02	40	4.36e−05	1.25e−05	9.01e−08
P_{LGS}	429	1.40e−01	12	5.81e−07	2.20e−06	1.36e−10
	2437	5.39e−01	14	3.49e−06	7.14e−07	4.01e−10
	117,37	4.21e+01	14	1.93e−06	4.53e−07	2.59e−10
P_{UGS}	429	1.07e−01	3	8.67e−01	1.08e+00	1.70e−05
	2437	3.29e−01	8	1.78e−01	1.79e−01	6.06e−07
	117,637	2.45e+01	8	3.39e−01	2.95e−01	6.85e−07

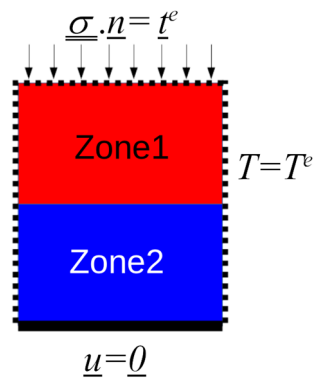


Fig. 1 Test case

For this purpose, a 3D rectangular sample is modelled as seen in Fig. 1, with a 0.1 m length following x , a 0.1 m height following y and 0.05 m large following z . The tetrahedral mesh was generated using Gmsh 4.4.1.

The displacement was set to 0 on the bottom surface ($y = 0$), a mechanical pressure of 5 MPa was applied on the top surface ($y = 0.1$) and a temperature of 80°C was imposed on the whole surface of the sample. No other boundary condition is applied. Regarding the hydraulics, the experiment is conducted under undrained condition. With respect to the initial conditions, the sample is supposed to be initially in a state of zero force.

Depending on the assessment under consideration, the sample consists of a single material or of 2 different materials. For the robustness experiments, it consists of clay only, while for the scalability experiments, it consists of clay and concrete. The 2 different subdomains are illustrated in Fig. 1. We emphasize that the order of magnitude of the

Table 3 The test case parameters

Symbol	Value	Unit
μ_l	10^{-3}	Pa s
K_l	2.10^9	Pa
C_s	1000	$\text{J kg}^{-1} \text{K}^{-1}$
C_f	4180	$\text{J kg}^{-1} \text{K}^{-1}$
C_f^p	4180	$\text{J kg}^{-1} \text{K}^{-1}$
ρ_f	1000	kg m^{-3}
λ_T	1.6	$\text{W m}^{-1} \text{K}$
T_0	273	K
p_{atm}	10^5	Pa
α_s	10^{-5}	K^{-1}
α_l	10^{-4}	K^{-1}
h_{f0}	$\frac{p_{atm}}{\rho_l}$	J kg^{-1}
K_s	$\frac{E}{3(1-2\nu)}$	Pa
K_0	K_s	Pa
λ_H	K_{int}/μ_l	$\text{Pa}^{-1} \text{m}^2 \text{s}^{-1}$
C_σ^0	$C_s \rho_s (1 - \varphi) + C_l \rho_f \varphi$	$\text{J K}^{-1} \text{m}^{-3}$
ρ_s	$(\rho_m - \varphi \rho_f)/(1 - \varphi)$	kg m^{-3}
α_m	$\varphi \alpha_l + (1 - \varphi) \alpha_s$	K^{-1}
Clay		
Symbol	Value	Unit
E	6.10^9	Pa
ν	0.3	–
ρ_m	2410	kg m^{-3}
K_{int}	4.10^{-21}	m^2
φ	0.18	–
Concrete		
Symbol	Value	Unit
E	15.10^9	Pa
ν	0.2	–
ρ_m	2500	kg m^{-3}
K_{int}	10^{-11}	m^2
φ	0.2	–

material parameters are of great importance in the industrial applications. The values of the material parameters, displayed in Table 3, are representative of a typical industrial problem of geological waste disposal [4].

The tests were solved with code_aster using the THM framework presented in the section above, corresponding to an isotropic saturated single-phased THM medium using P2-P1-P1 finite elements.

Robustness

The robustness of the preconditioners is evaluated by varying the values of the Young's modulus E , the intrinsic permeability k_{int} and the thermal conductivity λ_T . These parameters are chosen since they appear respectively in each balance equation and have a major influence therein. The tests are done using the test case of Fig. 1 with both Zones 1 and 2 made of clay, using the material parameters in Table 3.

Table 4 Block Jacobi Parameter Robustness

\mathbf{P}_J					
Parameters			DoF		
E	k_{int}	λ_T	10 000	60 000	200 000
1.0e+09	4.0e−15	4.0e−01	6 (3)	6 (2)	6 (4)
		2.3e+00	5 (3)	5 (2)	4 (2)
	4.0e−18	4.0e−01	8 (3)	8 (3)	9 (3)
		2.3e+00	8 (3)	7 (3)	8 (3)
	4.0e−21	4.0e−01	48 (3)	46 (3)	50 (3)
		2.3e+00	70 (3)	44 (3)	40 (3)
2.5e+10	4.0e−15	4.0e−01	6 (2)	6 (2)	7 (3)
		2.3e+00	6 (2)	6 (2)	4 (2)
	4.0e−18	4.0e−01	7 (2)	6 (2)	8 (2)
		2.3e+00	6 (2)	6 (2)	7 (2)
	4.0e−21	4.0e−01	12 (2)	12 (2)	11 (2)
		2.3e+00	11 (2)	11 (2)	11 (2)
5.0e+10	4.0e−15	4.0e−01	6 (2)	6 (2)	7 (3)
		2.3e+00	6 (2)	6 (2)	5 (2)
	4.0e−18	4.0e−01	6 (2)	6 (2)	8 (2)
		2.3e+00	6 (2)	6 (2)	6 (2)
	4.0e−21	4.0e−01	10 (2)	11 (2)	10 (2)
		2.3e+00	10 (2)	9 (2)	9 (2)

Table 5 Block Gauss-Seidel Parameter Robustness

\mathbf{P}_{LGS}					
Parameters			DoF		
E	k_{int}	λ_T	10 000	60 000	200 000
1.0e+09	4.0e−15	4.0e−01	4 (3)	5 (2)	5 (4)
		2.3e+00	4 (3)	4 (2)	3 (2)
	4.0e−18	4.0e−01	6 (3)	5 (3)	6 (3)
		2.3e+00	5 (3)	5 (3)	6 (3)
	4.0e−21	4.0e−01	21 (3)	22 (3)	20 (3)
		2.3e+00	18 (3)	20 (3)	18 (3)
2.5e+10	4.0e−15	4.0e−01	5 (2)	6 (2)	5 (3)
		2.3e+00	5 (2)	5 (2)	4 (2)
	4.0e−18	4.0e−01	5 (2)	6 (2)	6 (2)
		2.3e+00	5 (2)	5 (2)	6 (2)
	4.0e−21	4.0e−01	8 (2)	8 (2)	8 (2)
		2.3e+00	7 (2)	7 (2)	7 (2)
5.0e+10	4.0e−15	4.0e−01	5 (2)	5 (2)	5 (3)
		2.3e+00	5 (2)	5 (2)	4 (2)
	4.0e−18	4.0e−01	5 (2)	5 (2)	7 (2)
		2.3e+00	5 (2)	5 (2)	7 (2)
	4.0e−21	4.0e−01	7 (2)	7 (2)	7 (2)
		2.3e+00	6 (2)	7 (2)	6 (2)

The results are compiled in Table 4 for \mathbf{P}_J and Table 5 for \mathbf{P}_{LGS} . The maximum number of outer FGMRES iterations during the Newton iterations are displayed first, followed by the total number of Newton iterations in parentheses. The very large range of variation of the mesh size and of each parameter (up to six orders of magnitude) is emphasized.

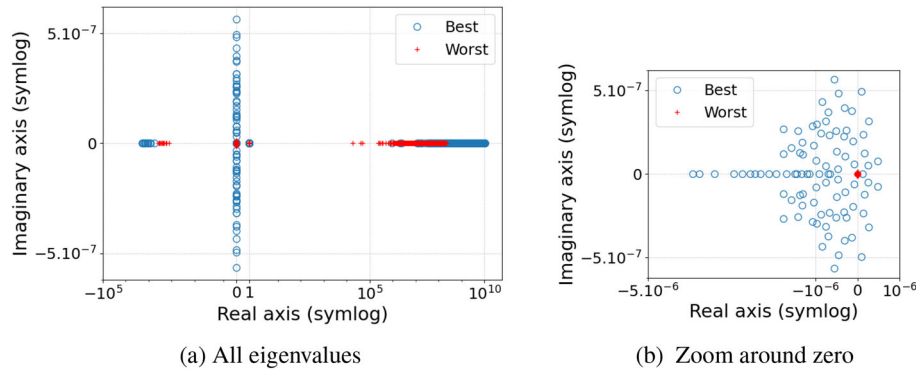


Fig. 2 Eigenvalue distribution for the “best” and “worst” cases

In order to analyse the results in Tables 4 and 5, we propose first a row-wise reading then a column-wise reading.

The row-wise reading provides information on the influence of the mesh size, with the material parameters being fixed. An excellent independence with respect to the mesh size is observed. The outer number of FGMRES iterations remains constant even though the size of the system is multiplied by 20, except for \mathbf{P}_J with the set of parameters ($E=1.e+9$, $k_{int}=4.e-21$, $\lambda_T=2.3$) where the number of iterations varies from 70 to 44 but seems to stabilize by reaching 40 in the biggest mesh.

The column-wise reading provides information on the influence of the material parameters, with the mesh size being fixed. A moderate variation of the outer number of FGMRES iterations is observed, that remains mostly under 12 except for the “worst” set of parameters ($E=1.e+09$, $k_{int}=4.e-21$), where it reaches up to 70 iterations for \mathbf{P}_J and 22 for \mathbf{P}_{LGS} . This particular result tends to show a better robustness of the block Gauss-Seidel variant compared to the Jacobi variant, which is further analyzed in the next section. In spite of this, both preconditioners appear to be very robust as they achieve convergence at each run and the increase in Krylov iterations remains moderate compared to the large variations in material parameters.

Finally, we highlight the excellent robustness with respect to the Newton iterations, which remain between 2 and 4 for every run.

Spectral analysis

In the previous section, a better robustness of the lower block Gauss-Seidel variant \mathbf{P}_{LGS} compared to the Jacobi variant \mathbf{P}_J was observed. In order to further analyze this, let us denote by:

- “best” case, the set of parameters ($E = 5.e+10$ Pa, $k_{int} = 4.e-15$ m², $\lambda_T = 0.4$ W m⁻¹.K)
- “worst” case, the set of parameters ($E = 1.e+9$ Pa, $k_{int} = 4.e-21$ m², $\lambda_T = 2.3$ W m⁻¹.K)

In the “best” case, the proposed Krylov method and preconditioners converge in 6 iterations while in the “worst” case, 70 iterations are needed.

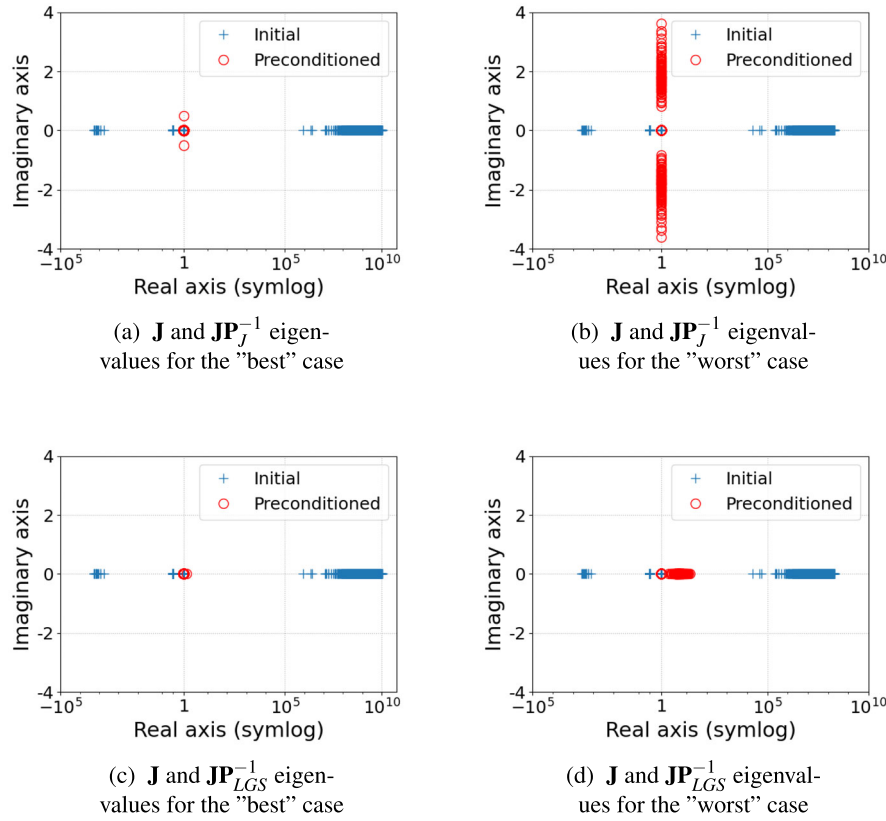


Fig. 3 Eigenvalue distribution

Let us begin by commenting the spectrum of the initial system, displayed in Fig. 2. In both cases, apart from some differences in the magnitude of the extreme values, the real parts of the eigenvalues are organised in four blocks:

- few percents are negative, lying around -10^3
- few percents lie around zero
- few percents lie around one
- most of the values lie around 10^5 and 10^8 (roughly speaking 80%)

In fact, the main difference is found around the origin. As can be seen from the zoom around the origin in Fig. 2, the "worst" case exhibit a clear cluster of almost zero eigenvalues while they are much more scattered in the "best" case.

We shall now evaluate the effect of both preconditioners on the spectrum of the Jacobian matrix. The eigenvalues of both the initial (i.e. not preconditioned) and the preconditioned system are displayed in Fig. 3.

Once a preconditioner is applied, the real part of all eigenvalues is clustered around 1. \mathbf{P}_J completely clusters the real part of the eigenvalues to 1 but distributes the imaginary part to of the eigenvalues between -1 and 1 in the "best" case and between -4 and 4 in the "worst" case. In any case, the eigenvalues belong to three different clusters, that appear to be more scattered in the "worst" case.

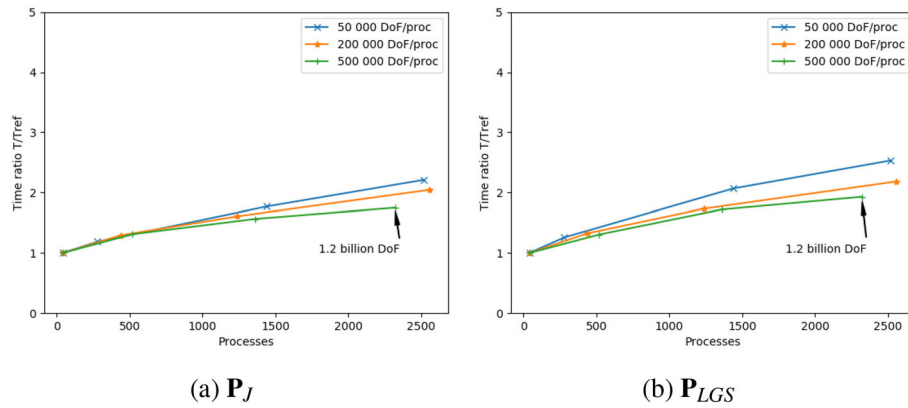


Fig. 4 Weak scalability

P_{LGS} generates almost real eigenvalues, with a tight cluster around 1 for the “best” case and between 1 and 100 in the “worst” case.

The difference in clustering the eigenvalues (three blocks for P_J and a single one for P_{LGS}) might explain the better results of the latter.

Parallel scalability

A good scalability of the proposed preconditioner is essential to keep the resolution time reasonable when switching to bigger systems. Weak and strong scalability tests are considered using the bi-material case from Fig. 1 with Zone 1 made of clay and Zone 2 made of concrete. Realistic parameter values were chosen from Table 3. Both of the scalability tests are run on EDF’s cluster Cronos. It consists in 1272 nodes, equipped with 2 Xeon Platinum 8260 24C 2.4 GHz processors with 24 cores each.

A weak scalability test consists in setting a fixed number of degrees of freedom (DoF) by processor and increasing the size of the problem by increasing the number of processes. In other words, we set the size of a sub-domain and make the problem bigger by increasing the total number of sub-domains. Our goal is to investigate if the solution algorithm needs the same resolution time whether we solve N DoF on 1 process or $1000 \times N$ DoF on 1000 processes. In case of perfect weak scalability, the time should remain constant when increasing the number of processes.

As can be seen in Fig. 4, the number of DoF per process is fixed to 50,000 (blue line), 200,000 (orange line) and 500,000 (green line) and the test case is run from 40 processes to 2500 processes. The ratio between the solution time to the 40 processes time is presented. For small numbers of DoF per process, it remains between 1. and 2.5 for P_{LGS} and between 1. and 2.2 for P_J . Whereas for 500,000 DoF per process, it remains between 1. and 1.9 for P_{LGS} and between 1. and 1.7 for P_J . This sub-optimal behavior for small sub-domains is often due to latency of the cluster’s network. When sub-domains are large and there is more work per process, the computation dominates the cost associated with communication. Even though P_J scales slightly better, the resolution time is higher than with P_{LGS} due to higher number of iterations that range between 8 and 16 for P_J and 7 and 11 for P_{LGS} . We highlight that using P_{LGS} for 500,000 DoF per processor (green line), the size of the linear system ranges from 20 million with a solving time of 465 s to more than 1.2 billion DoF with a solving time of 891 s. The size of the problem is multiplied

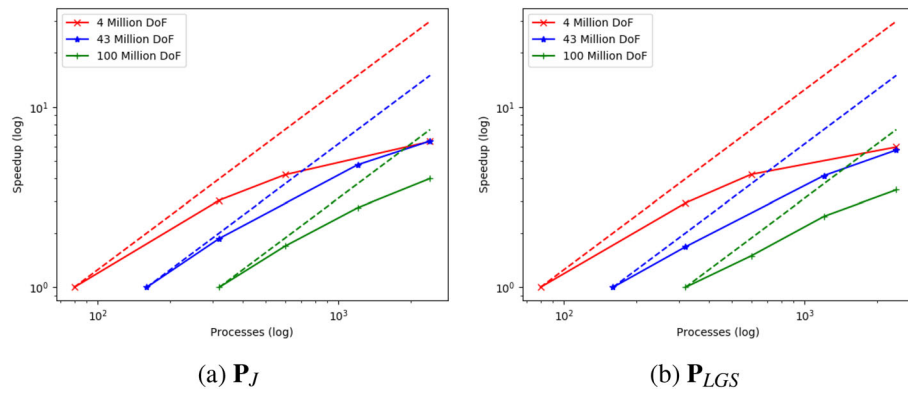


Fig. 5 Strong scalability

by 60 whereas the solving time only increases by 1.9. This is a very good scalability result since the test case is rather complex especially due to the variation of material parameters between clay and concrete.

Let us switch to the strong scalability test, which consists in fixing the size of the problem and increasing the number of processors. The goal is to solve the system faster by adding resources. For example, if we solve a system of a given size using N processes when using $N \times M$ processes the solving time should be divided by M . In case of perfect strong scalability, the solving time decreases proportionally to the increase of the number of processes.

The strong scalability tests are presented in Fig. 5. The size of the problem is fixed to 4 millions DoF (red line), 43 millions DoF (blue line) and 100 millions DoF (green line). The number of processes were increased from 80 to 2400. The dashed line represents the ideal strong scalability. The speedup with respect to the number of processes is presented. For all cases, when increasing the processes from 80 to 320 with 4 million DoF, the strong scalability remains satisfactory with an efficiency of 76% for P_J and of 73% for P_{LGS} . Then the efficiency starts declining until it reaches 2400 processes and is at 20% for P_J and for P_{LGS} . For the 100 million DoF case, from 320 to 600 processes the efficiency for P_J is 90% and for P_{LGS} 79%. When increasing the processes to 2400, the efficiency for P_J is 53% and for P_{LGS} 46%. Similarly to the weak scalability, the difference in efficiency between each problem is often due to latency of the cluster's network, since at 4 millions DoF there is less work per process. P_J scales slightly better than P_{LGS} but the latter remains faster for all the tested cases. Given the complexity of the test case and the fact that we started at 80 processes in order to be able to solve size-wise representative problems, the strong scalability of the proposed preconditioners is satisfactory.

Performance comparison with a Schur complement preconditioner

Definition of the alternative preconditioner

We now provide some comparisons with a commonly used preconditioning strategy. Let us consider a matrix with the following 2x2 block structure:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{00} & \mathbf{A}_{01} \\ \mathbf{A}_{10} & \mathbf{A}_{11} \end{bmatrix}$$

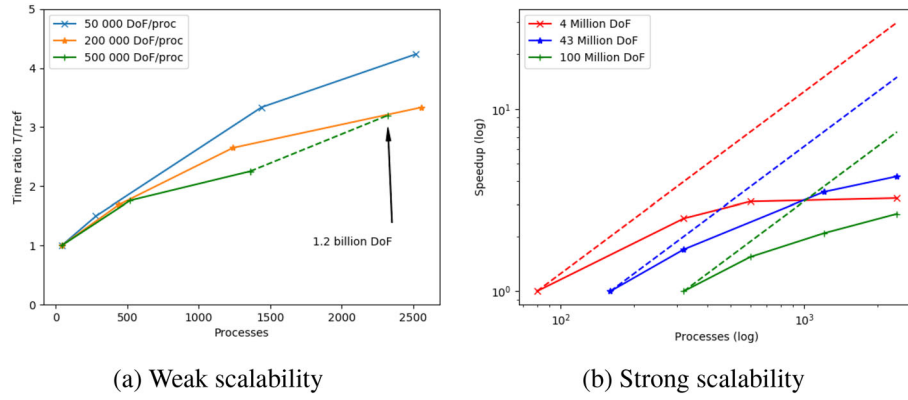


Fig. 6 Weak and strong scalability of \mathbf{P}_{Schur}

It has a LU factorization:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{00} & 0 \\ \mathbf{A}_{10} & \mathbf{S} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{A}_{00}^{-1}\mathbf{A}_{01} \\ 0 & \mathbf{I} \end{bmatrix}$$

where $\mathbf{S} = \mathbf{A}_{11} - \mathbf{A}_{10}\mathbf{A}_{00}^{-1}\mathbf{A}_{01}$ is called the Schur complement. From similarity considerations, it is shown that

$$\mathbf{P} = \begin{bmatrix} \mathbf{A}_{00} & 0 \\ \mathbf{A}_{10} & \mathbf{S} \end{bmatrix}$$

is an optimal preconditioner of \mathbf{A} since $\mathbf{P}^{-1}\mathbf{A}$ has a single eigenvalue of value 1. Unfortunately, as mentioned in the introduction, the exact evaluation of the Schur complement is computationally impossible due to its dense nature and approximations have to be used.

The approximation $\mathbf{S} \approx \hat{\mathbf{S}} = \mathbf{A}_{11} - \mathbf{A}_{10}\text{diag}(\mathbf{A}_{00})^{-1}\mathbf{A}_{01}$ is frequently used in the field of porous media [24,47] as well as in other domains [48,49].

In the sequel, we propose to evaluate the preconditioner:

$$\mathbf{P}_{Schur} = \begin{bmatrix} \mathbf{J}_{uu} & 0 & 0 \\ \mathbf{J}_{p\underline{u}} & \hat{\mathbf{S}} & 0 \\ \mathbf{J}_{T\underline{u}} & \mathbf{J}_{Tp} & \mathbf{J}_{TT} \end{bmatrix} \text{ where } \hat{\mathbf{S}} = \mathbf{J}_{pp} - \mathbf{J}_{p\underline{u}}\text{diag}(\mathbf{J}_{uu})^{-1}\mathbf{J}_{up}$$

The choice of using the Schur complement approach for the unknowns \underline{u} and p comes from [24], because of the strong coupling that exists between these degrees of freedom. The same choice was made in [50] for the case of an incompressible flow with thermal convection. In the latter, the authors deal with the temperature block according to a lower Gauss-Seidel strategy (as used above). They point out that this approach works remarkably well in practice.

Performance comparison

The same scalability tests are conducted on the Schur complement preconditioner \mathbf{P}_{Schur} . The results are shown in Fig. 6. Regarding the weak scalability, it is clear that \mathbf{P}_{Schur} performs worse than \mathbf{P}_{LGS} . For small numbers of DoF per process, the weak scalability remains between 1. and 4.2 for \mathbf{P}_{Schur} whereas it remains between 1. and 2.5 for \mathbf{P}_{LGS} .

For the first three cases of 500 000 DoF per processor, it remains between 1. and 1.7 for \mathbf{P}_{LGS} and between 1. and 2.2 for \mathbf{P}_{Schur} . The fourth point is in dashed line and must be regarded with caution. Indeed, with \mathbf{P}_{Schur} , the nonlinear solver failed to converge on the case that reaches 1.2 billion DoF. We have been obliged to use a lower tolerance for the outer Krylov method to recover the nonlinear convergence, thus implying an increase in the solution time with respect to the previous points.

If we focus on the third point of the green curve, the problem of 680 million DoF is solved on 1400 processes. The solution time using \mathbf{P}_{LGS} is of 809 s and of 1080 s using \mathbf{P}_{Schur} with an increase of 33%.

For the strong scalability, in all cases, when increasing the processes from 80 to 320 with 4 million DoF, the strong scalability remains satisfactory with an efficiency of 62% for \mathbf{P}_{Schur} and of 73% for \mathbf{P}_{LGS} . Then the efficiency starts decreasing when 2 400 processes are used where it drops to 10% for \mathbf{P}_{Schur} and 20% for \mathbf{P}_{LGS} . For the 100 million DoF case, from 320 to 600 processes, the efficiency for \mathbf{P}_{Schur} is 82% and 79% for \mathbf{P}_{LGS} . When increasing to 2400 processes, the efficiency for \mathbf{P}_{Schur} is 35% and 46% for \mathbf{P}_{LGS} .

Finally, in all weak and strong scalability tests, the outer FGMRES iterations for \mathbf{P}_{Schur} range from 27 to 72. For \mathbf{P}_{LGS} they remain between 7 and 10. We conclude that \mathbf{P}_{LGS} shows a better robustness, has a better scalability and is faster than \mathbf{P}_{Schur} for problems bigger than 2 million DoF.

Illustrative numerical results

Industrial problem

In this section, we apply the proposed model and the associated preconditioner to the long-term evolution of the rock surrounding a deep geological repository for radioactive waste. The problem is based on a repository for high-level, long-lived radioactive waste in the Callovo-Oxfordian clay and is fully presented in [4]. The goal of the simulation is to model the excavation of the disposal and the placement of the packaged waste modeled by a representative thermal flow: indeed, due to the remaining radioactivity, heat is emitted and its influence on the surrounding media needs to be evaluated.

The geometry and the associated dimensions of the repository are shown in Fig. 8. It consists of a main access gallery from which multiple storage cells branch off, into which packaged waste are placed. The repository is located at a depth of -560 m. Dimensions taken into account are realistic but do not correspond to a real and precise architecture.

Given the the symmetry of the site, the geometry of the model consists of a section of 18 m of the main gallery that crosses a single storage cell, as shown in Fig. 9. The domain is confined by a layer of argillite and we apply symmetry conditions on the lateral and upper parts of the domain. The material parameters of the clay and of the concrete are given in Table 3. The measured initial pore pressure at this depth is $p_0 = 5.6$ MPa and it varies linearly with depth according to $p(z) = \rho_f \cdot g \cdot (z - 560) + 5.6e6$. Similarly, the initial stress state is $\sigma_{xx} = -12.4$ MPa, $\sigma_{yy} = -16.1$ MPa, $\sigma_{zz} = -12.4$ MPa and it varies linearly with z to the surface. The initial temperature varies linearly with respect to z from 25°C at $z = -560$ m to 22.7°C at $z = -483.75$ m. The initial porosity is 0.18 throughout the argillite.

The methodology to model the excavation follows the classical Convergence-Confinement method [51] (also called CV-CF). It begins with an initial state with no galleries, where the

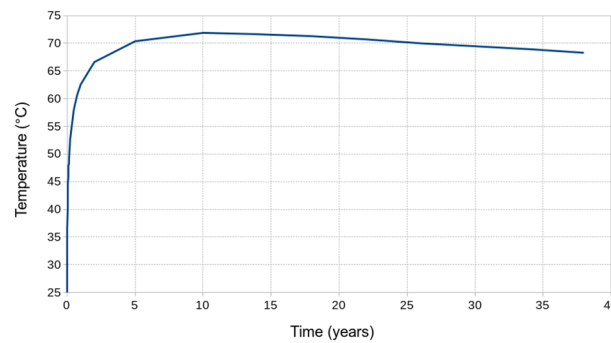


Fig. 7 Applied temperature on the wall of the cell

domain is only submitted to gravity. In a first step, a quasi-instantaneous excavation of the gallery and cell is considered; that is to say that the pore pressure and total radial stress at the galleries walls become zero in one second. In a second step, the concrete lining is introduced. In a third and final step, the radioactivity causes heat to be emitted around the cell in the course of time, and we therefore apply a representative temperature at the cell wall, according to the curve in Fig. 7. Fixed radial displacements are maintained at the cell wall. At the wall between the concrete lining and the gallery, the total radial stress is set to zero. The simulation is run until 40 years after the waste has been placed in the cell.

We turn now our attention to the solution of the simulation. The evolution of pressure over time is shown in Fig. 10. A rapid and significant increase of the hydraulic pressure is observed in the vicinity of the cell. This is due to the differential expansion caused by the thermal load since the thermal dilation of the water is bigger than that of the solid. After reaching a maximum, the pressure decreases steadily due to the diffusion of water and the decrease in temperature. Figure 11 shows that the main gallery undergoes a vertical collapse of about 1.3 cm as a result of its excavation, which is clearly present from the first year. Subsequently, due to the significant thermal expansion and the increase in fluid pressure, the upper part of the domain undergoes a strong upward shift. This leads to a complex and highly three-dimensional stress state as can be seen in Fig. 12. The zoom on the crossing of the gallery and the cell shows an intense traction zone where the signed Von Mises stress reaches 10 MPa. This is due to the fact that this section of the cell does not contain any waste. The complexity of the stress distribution fully justifies the fineness of the mesh.

Solver performance

The geometry and the mesh have been generated using the Salomé Platform 9.8 [52]. Due to the required precision of the analysis, a very fine mesh in the vicinity the cell-gallery crossing is used, with 104,873,938 nodes and 77,190,165 tetrahedral elements. It results in a problem with 341,292,114 DoF. The domain has been divided in 2560 subdomains so that the simulation is run on 64 nodes of the Cronos cluster, each one running 40 MPI processes.

From a practical point of view, the 4 phases of the method consist in 4 non-linear simulations, each one being the initial state of the following. The last one deals with the effect of the waste on 40 years. The size of the time step follows the variation of the heat

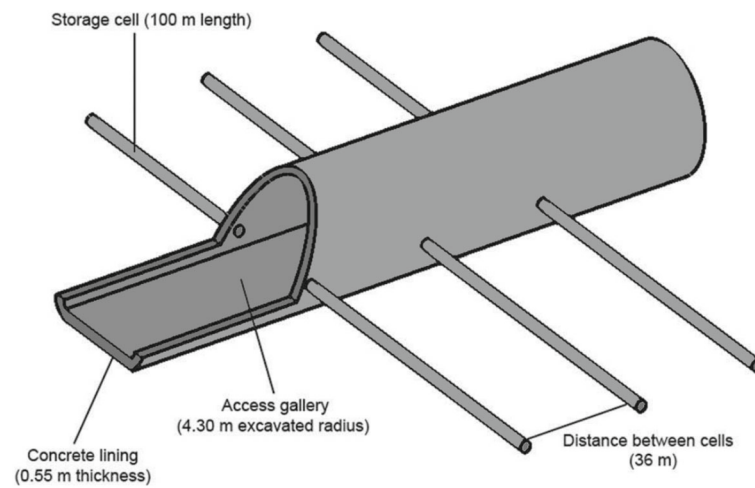


Fig. 8 Dimensions of the storage facility (courtesy from [4])

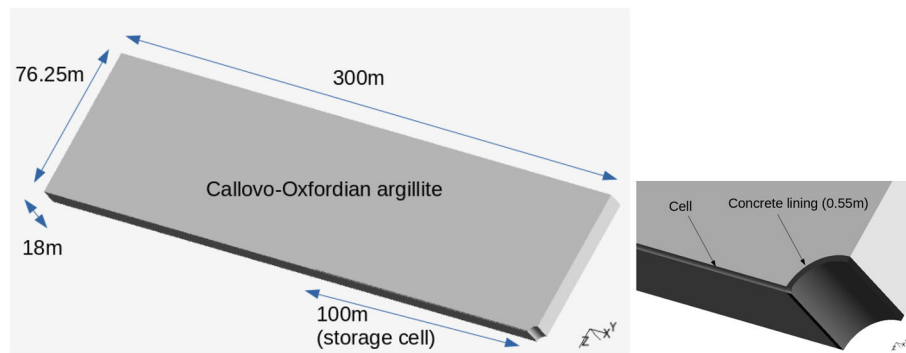


Fig. 9 Geometry of the site and zoom on the cell and concrete lining (courtesy from [4])

induced by the waste, shorter at the beginning when it varies strongly (say the order of 1 day) and longer when it stabilizes (say the order of 1 month).

The non-linear convergence is easily reached in 2 or 3 Newton's iterations per time-step, exhibiting quadratic convergence. Each linear solve requires less than 9 outer iterations and takes roughly 30 s. Given the total number of 130 Newton's iterations, the simulation is achieved in 3800 s. If the Schur preconditioner is used, the total solution time reaches 4700 s, increasing by 24%. We emphasize that the use of the proposed preconditioner allows for a simulation whose size is incomparable to other simulations in the specific field of radioactive waste disposal [4,53,54].

Conclusion and outlook

This paper deals with the assessment of the robustness and the weak and strong scalability of a preconditioner dedicated to coupled THM problems, which relies on the block structure of the Jacobian of the linearized system. A block Jacobi and a block Gauss-Seidel variants are investigated, sharing the same tailored sub-solvers (Krylov methods preconditioned by AMG preconditioners).

Coupled systems often exhibit very bad scaling due to the presence of parameters of different orders of magnitude. It can be addressed by the use of a dedicated scaling algorithm

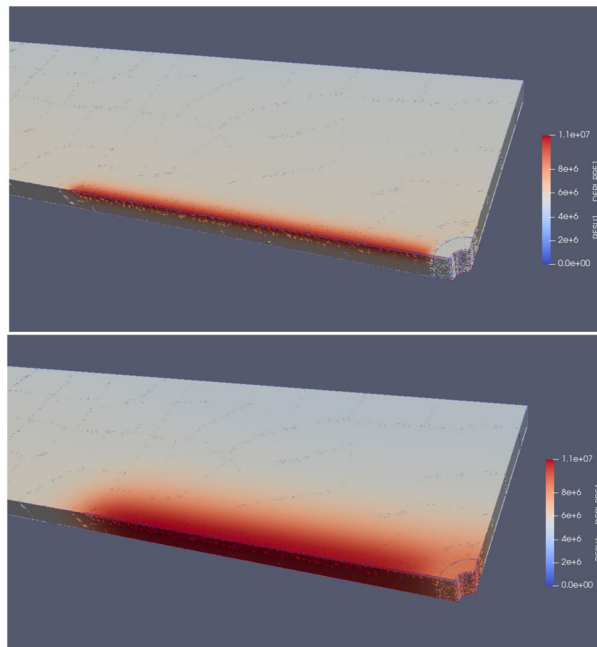


Fig. 10 Pressure distribution after 1 and 10 years

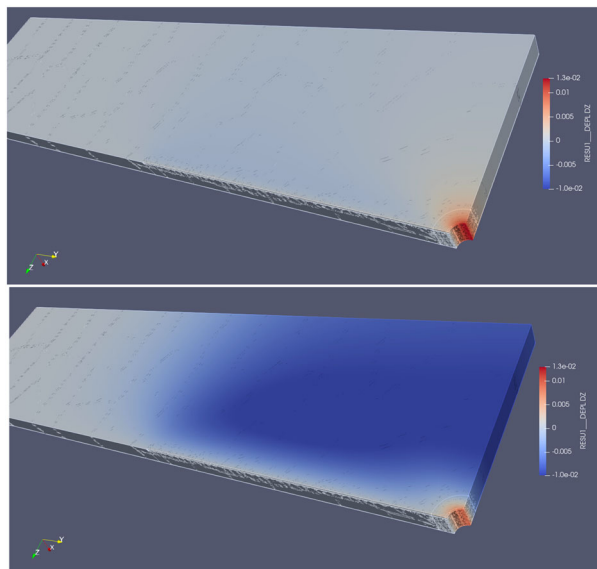


Fig. 11 Vertical displacement distribution after 1 and 10 years

that efficiently re-balances the Jacobian. It is nevertheless not mandatory in our case since it is shown that the proposed block preconditioners can handle naturally the unbalance of the different blocks. For the case of the block Gauss-Seidel variant, special attention is needed to eliminate the unknowns in a well-chosen order. Finally, both variants show excellent mesh size independence, good robustness with respect to parameters variation and good scalability on a simple yet representative test case.

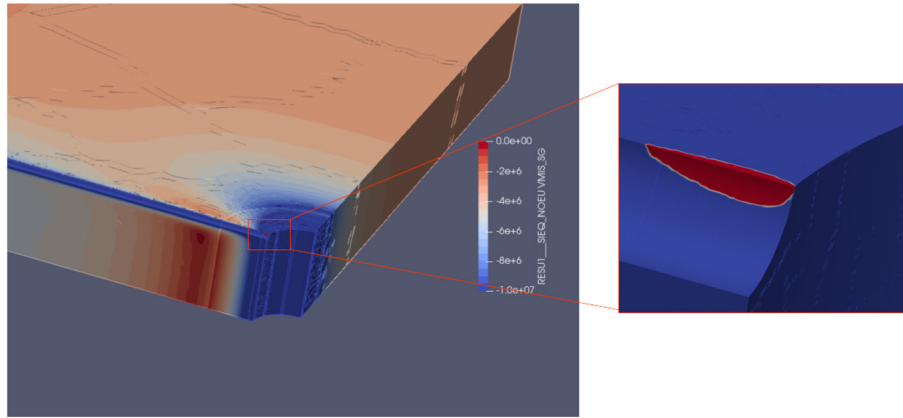


Fig. 12 Signed VonMises stress and zoom after 10 years

Though established in the linear regime, these results are very valuable when considering to move to nonlinear constitutive laws. This point is being investigated and encouraging results have already been obtained.

Appendix A

The matrix blocks of the Jacobian matrix are given by:

$$\begin{aligned}
 (\mathbf{J}_{uu}) &= \int_{\Omega} -\underline{\underline{A}} : \underline{\underline{\varepsilon}}(\phi_{v_i}) : \underline{\underline{\varepsilon}}(\phi_{v_j}) dx, & \forall i, j = 1, N_u \\
 (\mathbf{J}_{up}) &= \int_{\Omega} \phi_{q_j} \operatorname{div}(\phi_{v_i}) dx, & \forall i = 1, N_u, \quad \forall j = 1, N_p \\
 (\mathbf{J}_{uT}) &= \int_{\Omega} 3K_s \alpha_s \phi_{w_j} \operatorname{div}(\phi_{v_i}) dx, & \forall i = 1, N_u, \quad \forall j = 1, N_T \\
 (\mathbf{J}_{pu}) &= \int_{\Omega} \rho_f \phi_{q_i} \operatorname{div}(\phi_{v_j}) dx, & \forall i = 1, N_p, \quad \forall j = 1, N_u \\
 (\mathbf{J}_{pp}) &= \int_{\Omega} \frac{\varphi}{K_l} \rho_f \phi_{q_i} \phi_{q_j} - \rho_f \lambda_H \Delta t \nabla \phi_{q_i} \nabla \phi_{q_j} dx, & \forall i, j = 1, N_p \\
 (\mathbf{J}_{pT}) &= \int_{\Omega} -\rho_f \alpha_m 3 \phi_{q_i} \phi_{w_j} dx, & \forall i = 1, N_p, \quad \forall j = 1, N_T \\
 (\mathbf{J}_{Tu}) &= \int_{\Omega} \rho_f h_f \phi_{v_j} \operatorname{div}(\phi_{w_i}) \\
 &\quad + T_{k_h}^{n+1} 3K_0 \alpha_s \phi_{v_j} \operatorname{div}(\phi_{w_i}) dx, & \forall i = 1, N_T, \quad \forall j = 1, N_u \\
 (\mathbf{J}_{Tp}) &= \int_{\Omega} (1 - 3\alpha_l T_{k_h}^{n+1}) \phi_{q_j} \\
 &\quad - \Delta t \lambda_H \nabla p_{k_h}^{n+1} \nabla \phi_{w_i} + \operatorname{div}(\underline{\underline{u}}_{k_h}^{n+1}) \phi_{w_i} \\
 &\quad + \frac{\varphi}{K_l} p_{k_h}^{n+1} \phi_{w_i} - \alpha_m 3 T_{k_h}^{n+1} \phi_{w_i} \\
 &\quad + \rho_f h_f - \Delta t \lambda_H \nabla \phi_{q_j} \nabla \phi_{w_i} \\
 &\quad + \frac{\varphi}{K_l} \phi_{q_j} \phi_{w_i} - T_{k_h}^{n+1} 3\alpha_m \phi_{q_j} \phi_{w_i} dx, & \forall i = 1, N_T, \quad \forall j = 1, N_p \\
 (\mathbf{J}_{TT}) &= \int_{\Omega} -\Delta t \lambda_T \nabla \phi_{w_i} \nabla \phi_{w_j} + C_{\sigma}^0 \phi_{w_i} \phi_{w_j} \\
 &\quad + \rho_f C_f^p \phi_{w_i} - \Delta t \lambda_H \nabla p_{k_h}^{n+1} \nabla \phi_{w_j} + \operatorname{div}(\underline{\underline{u}}_{k_h}^{n+1}) \phi_{w_j} \\
 &\quad + \frac{\varphi}{K_l} p_{k_h}^{n+1} \phi_{w_j} - \alpha_m 3 T_{k_h}^{n+1} \phi_{w_j} \\
 &\quad - \rho_f h_f \alpha_m 3 \phi_{w_i} \phi_{w_j} \\
 &\quad + \phi_{w_i} 3K_0 \alpha_s \operatorname{div}(\underline{\underline{u}}_{k_h}^{n+1}) \phi_{w_j} - 3\alpha_m p_{k_h}^{n+1} \phi_{w_j} \\
 &\quad - 18K_0 \alpha_s^2 T_{k_h}^{n+1} \phi_{w_j} dx, & \forall i, j = 1, N_T
 \end{aligned}$$

Acknowledgements

The authors acknowledge the support of the french "Association Nationale de la Recherche et de la Technologie" (see the Funding section for further detail).

Author contributions

All tests were run by the first author. The first three authors contributed to the definition of the preconditioners. The fourth author contributed to the scaling study. The fifth author contributed to the THM formulation. All authors have prepared the manuscript. All authors have read and approved the final manuscript.

Funding

Ana Ordóñez was partially supported by a CIFRE fellowship of the french "Association Nationale de la Recherche et de la Technologie" (ANRT) under grant n° 2019/0866 and EDF R&D.

Data availability statement

The source code of code_aster is freely available on the web site www.code-aster.org. The data for the scalability tests can be made available upon reasonable request.

Received: 13 July 2022 Accepted: 4 May 2023

Published online: 26 June 2023

References

- Hetttema MHH, Schutjens PMTM, Verboom BJM, Gussinklo HJ. Production-induced compaction of a sandstone reservoir: the strong influence of stress path. *SPE Reserv Eval Eng*. 2000;3(04):342–7.
- Moridis GJ, Kim J, Reagan MT, Kim S-J. Feasibility of gas production from a gas hydrate accumulation at the ubgh2-6 site of the Ullung basin in the Korean east sea. *J Petrol Sci Eng*. 2013;108:180–210. <https://doi.org/10.1016/j.petrol.2013.03.002>.
- Rutqvist J, Freifeld B, Min K-B, Elsworth D, Tsang Y. Analysis of thermally induced changes in fractured rock permeability during 8 years of heating and cooling at the yucca mountain drift scale test. *Int J Rock Mech Min Sci*. 2008;45:1373–89. <https://doi.org/10.1016/j.jrmms.2008.01.016>.
- Giot R, Granet S, Faivre M, Massoussi N, Huang J. A transversely isotropic thermo-poroelastic model for claystone: parameter identification and application to a 3D underground structure. *Geomech Geoeng*. 2018;13(4):246–63. <https://doi.org/10.1080/17486025.2018.1445874>.
- Terzaghi K, et al. *Erdbaumechanik auf bodenphysikalischer grundlage* 1925.
- Biot MA. General theory of three-dimensional consolidation. *J Appl Phys*. 1941;12(2):155–64. <https://doi.org/10.1063/1.1712886>.
- Biot MA. Variational Lagrangian-thermodynamics of nonisothermal finite strain mechanics of porous solids and thermomolecular diffusion. *Int J Solids Struct*. 1977;13(6):579–97. [https://doi.org/10.1016/0020-7683\(77\)90031-2](https://doi.org/10.1016/0020-7683(77)90031-2).
- Coussy O. *Mechanics and physics of porous solids*. 2010. <https://doi.org/10.1002/9780470710388.ch5>.
- Settari A, Walters D. Advances in coupled geomechanical and reservoir modeling with applications to reservoir compaction. *SPE J*. 2001;6:334–42. <https://doi.org/10.2118/74142-PA>.
- Armero F, Simo JC. A new unconditionally stable fractional step method for non-linear coupled thermomechanical problems. *Int J Numer Methods Eng*. 1992;35(4):737–66. <https://doi.org/10.1002/nme.1620350408>.
- Bungartz H-J, Lindner F, Gatzhammer B, Mehl M, Scheufele K, Shukaev A, Uekermann B. preCICE—a fully parallel library for multi-physics surface coupling. *Comput Fluids*. 2016;141:250–8. <https://doi.org/10.1016/j.compfluid.2016.04.003>.
- Markert B, Heider Y, Ehlers W. Comparison of monolithic and splitting solution schemes for dynamic porous media problems. *Int J Numer Meth Eng*. 2010;82(11):1341–83. <https://doi.org/10.1002/nme.2789>.
- Lee J, Mardal K-A, Winther R. Parameter-robust discretization and preconditioning of biot's consolidation model. *SIAM J Sci Comput*. 2017;39:1–24. <https://doi.org/10.1137/15M1029473>.
- Chen S, Hong Q, Xu J, Yang K. Robust block preconditioners for poroelasticity, 2020.
- Ferronato M, Castelletto N, Gambolati G. A fully coupled 3-d mixed finite element model of biot consolidation. *J Comput Phys*. 2010;229(12):4813–30. <https://doi.org/10.1016/j.jcp.2010.03.018>.
- Castelletto N, White JA, Ferronato M. Scalable algorithms for three-field mixed finite element coupled poromechanics. *J Comput Phys*. 2016;327:894–918. <https://doi.org/10.1016/j.jcp.2016.09.063>.
- Luo P, Rodrigo C, Gaspar FJ, Oosterlee CW. Multigrid method for nonlinear poroelasticity equations. *Comput Visual Sci*. 2015;17(5):255–65. <https://doi.org/10.1007/s00791-016-0260-8>.
- Luo P, Rodrigo C, Gaspar FJ, Oosterlee CW. On an Uzawa smoother in multigrid for poroelasticity equations. *Numer Linear Algebra Appl*. 2017;24(1):2074. <https://doi.org/10.1002/nla.2074.e2074nla.2074>.
- Gaspar FJ, Rodrigo C. On the fixed-stress split scheme as smoother in multigrid methods for coupling flow and geomechanics. *Comput Methods Appl Mech Eng*. 2017;326:526–40. <https://doi.org/10.1016/j.cma.2017.08.025>.
- Wienands R, Gaspar FJ, Lisbona FJ, Oosterlee CW. An efficient multigrid solver based on distributive smoothing for poroelasticity equations. *Computing*. 2004;73(1):99–119. <https://doi.org/10.1007/s00607-004-0078-y>.
- Gaspar FJ, Lisbona FJ, Oosterlee CW, Wienands R. A systematic comparison of coupled and distributive smoothing in multigrid for the poroelasticity system. *Numer Linear Algebra Appl*. 2004;112(2–3):93–113. <https://doi.org/10.1002/nla.372>.
- Benzi M, Golub GH, Liesen J. Numerical solution of saddle point problems. *Acta Numer*. 2005;14:1–137.
- White B, Joshua A, I R. *Computational Geosciences*. 2011. p. 647–59. <https://doi.org/10.1007/s10596-011-9233-7>.
- Haga JB, Osnes H, Langtangen HP. Efficient block preconditioners for the coupled equations of pressure and deformation in highly discontinuous media. *Int J Numer Anal Methods Geomech*. 2011;35(13):1466–82. <https://doi.org/10.1002/nag.973>.
- White JA, Castelletto N, Tchelepi HA. Block-partitioned solvers for coupled poromechanics: a unified framework. *Comput Methods Appl Mech Eng*. 2016;303:55–74. <https://doi.org/10.1016/j.cma.2016.01.008>.
- Cao H, Tchelepi HA, Wallis JR, Yardumian HE. Parallel scalable unstructured CPR-type linear solver for reservoir simulation. *SPE Annual Technical Conference and Exhibition, vol. All Days (2005)*. SPE-96809-MS. <https://doi.org/10.2118/96809-MS>.

- onepetro.org/SPEATCE/proceedings-pdf/05ATCE/All-05ATCE/SPE-96809-MS/1839702/spe-96809-ms.pdf. <https://doi.org/10.2118/96809-MS>.
27. Gries S, Stüben K, Brown GL, Chen D, Collins DA. Preconditioning for efficiently applying algebraic multigrid in fully implicit reservoir simulations. *SPE J*. 2014;19(04):726–36. <https://doi.org/10.2118/163608-PA>.
 28. Tchalepi HA, Jiang Y. Scalable multistage linear solver for coupled systems of multisegment wells and unstructured reservoir models. *SPE Reservoir Simulation Conference*, vol. All Days (2009). <https://doi.org/10.2118/119175-MS>. <https://onepetro.org/spersc/proceedings-pdf/09RSS/All-09RSS/SPE-119175-MS/1784642/spe-119175-ms.pdf>.
 29. Cusini M, Lukyanov AA, Natvig J, Hajibeygi H. Constrained pressure residual multiscale (cpr-ms) method for fully implicit simulation of multiphase flow in porous media. *J Comput Phys*. 2015;299:472–86. <https://doi.org/10.1016/j.jcp.2015.07.019>.
 30. Cremon MA, Castelletto N, White JA. Multi-stage preconditioners for thermal-compositional-reactive flow in porous media. *J Comput Phys*. 2020;418: 109607. <https://doi.org/10.1016/j.jcp.2020.109607>.
 31. Adler J, Gaspar F, Hu X, Rodrigo C, Zikatanov L. Robust block preconditioners for biot's model. 2017.
 32. Kruse C, Darrigrand V, Tardieu N, Arioli M, Rüde U. Application of an iterative Golub-Kahan algorithm to structural mechanics problems with multi-point constraints. *Adv Model Simul Eng Sci*. 2020. <https://doi.org/10.1186/s40323-020-00181-2>.
 33. Biot MA, Willis DG. The elastic coefficients of the theory of consolidation. *J Appl Mech*. 1957;24(4):594–601. <https://doi.org/10.1115/1.4011606>.
 34. Coussy O. Revisiting the constitutive equations of unsaturated porous solids using a lagrangian saturation concept. *Int J Numer Anal Meth Geomech*. 2007;31(15):1675–94. <https://doi.org/10.1002/nag.613>.
 35. Coussy O. 3. Thermodynamics. New Jersey: Wiley; 2003. pp. 37–70. <https://onlinelibrary.wiley.com/doi/pdf/10.1002/0470092718.ch3>. <https://doi.org/10.1002/0470092718.ch3>.
 36. Coussy O. 4. Thermoporoelasticity. New Jersey: Wiley; 2003. pp. 71–112. <https://doi.org/10.1002/0470092718.ch4>. <https://onlinelibrary.wiley.com/doi/abs/10.1002/0470092718.ch4>.
 37. Ern A, Meunier S. A posteriori error analysis of Euler-Galerkin approximations to coupled elliptic-parabolic problems. *ESAIM Math Model Numer Anal*. 2009;43(2):353–75. <https://doi.org/10.1051/m2an:2008048>.
 38. Dhondt G. The finite element method for three-dimensional thermomechanical applications. Chichester, England: Wiley; 2004.
 39. Knight PA, Ruiz D, Uçar B. A symmetry preserving algorithm for matrix scaling. *SIAM J Matrix Anal Appl*. 2014;35(3):931–55. <https://doi.org/10.1137/110825753>.
 40. Saad Y. A flexible inner-outer preconditioned gmres algorithm. *SIAM J Sci Comput*. 1993;14(2):461–9. <https://doi.org/10.1137/0914028>.
 41. Mardal K-A, Winther R. Preconditioning discretizations of systems of partial differential equations. *Numer Linear Algebra Appl*. 2011;18:1–40. <https://doi.org/10.1002/nla.716>.
 42. Briggs W, Henson V, McCormick S. A multigrid tutorial, 2nd ed. 2000.
 43. McInnes LC, Smith B, Zhang H, Mills RT. Hierarchical Krylov and nested Krylov methods for extreme-scale computing. *Parallel Comput*. 2014;40(1):17–31. <https://doi.org/10.1016/j.parco.2013.10.001>.
 44. Falgout R, Jones J, Yang U. The design and implementation of hypre, a library of parallel high performance preconditioners. 2006; 51:267–294.
 45. Amestoy PR, Duff IS, Koster J, L'Excellent J-Y. A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM J Matrix Anal Appl*. 2001;23(1):15–41.
 46. de France E. Finite element *code_aster*, analysis of structures and thermomechanics for studies and research, Year = 1989–2022. Open source on www.code-aster.org.
 47. Joshaghani MS, Chang J, Nakshatrala KB, Knepley MG. Composable block solvers for the four-field double porosity/permeability model. *J Comput Phys*. 2019;386:428–66. <https://doi.org/10.1016/j.jcp.2019.02.020>.
 48. Cahouet J, Chabard J-P. Some fast 3d finite element solvers for the generalized stokes problem. *Int J Numer Methods Fluids*. 1988;8(8):869–95. <https://doi.org/10.1002/flid.1650080802>.
 49. Carriere D, Jeandel P. A3d finite element method for the simulation of thermoconvective flows and its performance on a vector-parallel computer. *Int J Numer Methods Fluids*. 1991.
 50. Howle VE, Kirby RC. Block preconditioners for finite element discretization of incompressible flow with thermal convection. *Numer Linear Algebra Appl*. 2012;19:427–40. <https://doi.org/10.1002/nla.1814>.
 51. de la Fuente M, Taherzadeh R, Sulem J, Nguyen XS, Subrin D. Applicability of the convergence-confinement method to full-face excavation of circular tunnels with stiff support system. *Rock Mech Rock Eng*. 2019;52:2361–76.
 52. Salomé Platform. <https://www.salome-platform.org/>. 2022.
 53. Mountaka S, Minh-Ngoc V, Gilles A. 3d modelling of excavation-induced anisotropic responses of deep drifts at the meuse/haute-marne url. *Rock Mech Rock Eng*. 2022;55:4183–207. <https://doi.org/10.1007/s00603-022-02841-8>.
 54. Xu H, Rutqvist J, Plúa C, Armand G, Birkholzer J. Modeling of thermal pressurization in tight claystone using sequential thm coupling: benchmarking and validation against in-situ heating experiments in cox claystone. *Tunn Undergr Space Technol*. 2020;103: 103428. <https://doi.org/10.1016/j.tust.2020.103428>.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.