Enriched Galerkin discretization for modeling poroelasticity and permeability alteration in heterogeneous porous media

T. Kadeethum [a,b,*], H.M. Nick [a], S. Lee [c], F. Ballarin [d]

[a] *Technical University of Denmark, Denmark*
[b] *Cornell University, New York, USA*
[c] *Florida State University, FL, USA*
[d] *mathLab, Mathematics Area, SISSA, Italy*

A R T I C L E   I N F O

A B S T R A C T

In this paper, we utilize the enriched Galerkin (EG) finite element method for the flow equation in Biot's system, which provides a robust locally conservative flux in heterogeneous porous media. The computational algorithm to solve the coupled system with the permeability alteration is presented with the linearization and Picard's iterative scheme. The block structure is utilized for the linear system in numerical discretization, and the computer code is shared in the open-source platform. In the numerical experiments, we compare the proposed EG method with the classical continuous Galerkin (CG) and discontinuous Galerkin (DG) finite element methods in different scenarios, including the North sea reservoirs setup. While DG and EG methods provide similar approximations for the pressure solutions, the CG method produces spurious oscillations in fluid pressure and volumetric strain solutions near the material interfaces, especially for the soft materials. The difference of flux approximation between EG and DG methods is insignificant; still, the EG method demands approximately two and three times fewer degrees of freedom than the DG method for two- and three-dimensional geometries.

© 2020 Elsevier Inc. All rights reserved.

1. Introduction

The volumetric displacement of a porous medium caused by the changing in pore pressure is relevant for many applications, including groundwater flow, underground heat mining, fossil fuel production, earthquake mechanics, and biomedical engineering [1–7]. The volumetric deformation may impact the hydraulic storability and permeability of porous material, which in turn, influences the fluid flow field. This coupling between fluid flow and solid deformation can be captured through the application of Biot's equation of poroelasticity [8,9]. The Biot's equation is often discretized using the two-field formulation and classical continuous Galerkin (CG) finite element methods because the implementation is straightforward and results in a small number of degrees of freedom (DOF) [10–13]. The CG method, in its classical form, is unable to provide solutions with local mass conservation or fluid pressure oscillation free at material interfaces with high permeability contrast [14–17].

To mitigate fluid pressure oscillations at material interfaces and assuring local mass conservation, the following approaches have been proposed: (1) an element based post-processing technique applied to the classic CG method [18],
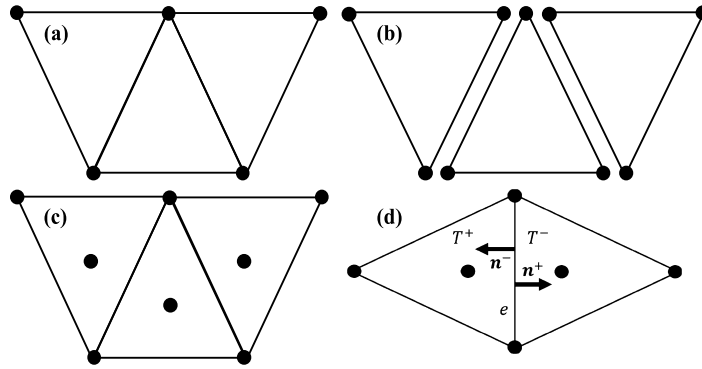

Fig. 1. Comparison of degrees of freedom for linear polynomial case among (**a**) continuous Galerkin (CG), (**b**) discontinuous Galerkin (DG), (**c**) enriched Galerkin (EG) function spaces, and (**d**) EG elements, where T is a triangular element, e is a face of T, and $\boldsymbol{n}$ is a normal unit vector to e.

(2) mixed finite element (MFE), i.e. three- and four-field formulations, developed by adding fluid velocity and solid pressure to primary variables [19,20,10], (3) hybrid-finite-element-finite-volume (FEFV) established by a combination of the CG finite element and cell-centered finite volume discretizations [21,22], (4) a discontinuous Galerkin (DG) method utilizing an interior penalty approach to stabilize the solution and preserve the local mass [23,24], and (5) a weak Galerkin (WG), which employs the first order Bernardi-Raugel elements on quadrilaterals mesh [25,26]. In recent years, physics-informed neural networks (PINN) have also been applied to solve this problem as a meshfree method [27,28].

Although the above six proposed methods have many benefits, there are disadvantages to each technique. The FEFV method creates a dual-mesh, which may result in smearing of mass in heterogeneous reservoirs [29,30]. The MFE method, on the other hand, increases the number of primary variables. Therefore, it may require more computational resources compared to the two-field formulation, especially in a three-dimensional domain. Moreover, it involves the inversion of the permeability tensor, which may lead to an ill-posed problem [31,10]. Even though the DG method is developed based on the two-field formulation, it is a rather computationally expensive method [32].

To overcome some of the disadvantages discussed above, the enriched Galerkin (EG) method [32,33], which is composed by enriching the CG method by a piecewise-constant function at the center of each element, was utilized in [31] to solve the Biot's system. It was shown that the EG method provides local mass conservation without any post-processing operations and delivers smooth pressure solutions across interfaces where material permeability is highly different [34,35]. The EG method has the same interior penalty type bilinear form as the DG method, but the EG method has fewer DOF than the DG method (see Figs. 1a-c for the comparison of DOF among CG, EG, and DG methods). The EG method has been developed to solve general elliptic and parabolic problems with dynamic mesh adaptivity [36–38].

In this study, we extend [31] by considering the following novelties:

1. A numerical model and an algorithm that incorporates fluid flow in heterogeneous poroelastic media and matrix permeability alteration are illustrated. The permeability alteration model, which describes an interaction between solid and fluid, considers the application of underground energy harvesting in oil and gas engineering [39,40]. Picard's iterative scheme with the linearization is utilized to solve the coupled nonlinear system. In the numerical examples, the effect of the permeability alteration from the solid deformation is illustrated with different material properties.
2. The computer code for the EG method by employing the multiphenics framework [41], which is built upon the FEniCS platform [42], is developed. This framework is based on the block structure for the linear system in the numerical discretization and simplifies the implementation of the EG method. The simplified implementation could help researchers to extend any existing DG codes to the EG method easily.
3. The computer codes used in this study are shared with the public, including the numerical examples, and the scripts for creating the EG solver are partially described in this manuscript.
4. Several numerical experiments comparing CG, DG, and EG methods in a single computational platform are presented. The accuracy of different methods with the given material properties in one- to three-dimensional spaces is compared, including the actual North Sea field setup. We observed that the EG method allows using a relatively coarse mesh to maintain accuracy. This benefit is an essential feature of the EG method that makes it more suitable for large scale simulations.

The rest of the paper is organized as follows. The methodology section includes model description, mathematical equations, and their discretizations for CG, EG, and DG methods. Subsequently, the solution strategy, including block formulations and iteration scheme, is illustrated. The results and discussion section begins with an analysis of the error convergence rate between the EG and DG methods to verify the developed block structure. Next, the established numerical schemes, CG, EG, and DG methods, for solving the coupled solid deformation and fluid flow system are validated using Terzaghi's 1D consolidation model [43]. Subsequently, we highlight the positive and negatives aspects of utilizing CG, EG, and DG methods

for four examples: 1D consolidation of a two-layered material, 2D flow in a deformable media with structured heterogeneity, 2D flow in a deformable media with random porosity and permeability, 3D flow in a deformable media with random porosity and permeability, and a North Sea hydrocarbon field. Summary and conclusion are finally provided.

2. Methodology

2.1. Governing equations

We are interested in solving Biot's equation on Ω, which amounts to time-dependent multiphysics problem coupling the solid deformation with fluid flow problem. Let $\Omega \subset \mathbb{R}^d$ be the domain of interest in d-dimensional space where $d = 1$, 2, or 3 bounded by boundary, $\partial\Omega$. $\partial\Omega$ can be decomposed to displacement and traction boundaries, $\partial\Omega_u$ and $\partial\Omega_t$, respectively, for the solid deformation problem. For the fluid flow problem, $\partial\Omega$ is decomposed to pressure and flux boundaries, $\partial\Omega_p$ and $\partial\Omega_q$, respectively. The time domain is denoted by $\mathbb{T} = (0, \tau]$ with $\tau > 0$.

The coupling between the fluid flow and solid deformation can be captured through the application of Biot's equation of poroelasticity, which is composed of linear momentum and mass balance equations [8]. The linear momentum balance equation can be written as follows:

$$\nabla \cdot \boldsymbol{\sigma}(\boldsymbol{u}, p) = \boldsymbol{f}, \tag{1}$$

where $\boldsymbol{u}$ is displacement, p is fluid pressure, $\boldsymbol{f}$ is body force, which is neglected in this study. The bold-face letters or symbols denote tensors and vectors, and the normal letters or symbols denote scalar values. Here, $\boldsymbol{\sigma}$ is total stress, which is defined as:

$$\boldsymbol{\sigma} := \boldsymbol{\sigma}' - \alpha p \boldsymbol{I}, \tag{2}$$

where $\boldsymbol{I}$ is the identity tensor and α is Biot's coefficient defined as [44]:

$$\alpha := 1 - \frac{K}{K_s}, \tag{3}$$

with the bulk modulus of a rock matrix K and the solid grains modulus K_s. In addition, $\boldsymbol{\sigma}'$ is an effective stress written as:

$$\boldsymbol{\sigma}'(\boldsymbol{u}) = \lambda_l \operatorname{tr}(\boldsymbol{\varepsilon}) \boldsymbol{I} + 2\mu_l \boldsymbol{\varepsilon}, \tag{4}$$

where λ_l and μ_l are Lamé constants, $\boldsymbol{\varepsilon}$ is strain, which is defined as:

$$\boldsymbol{\varepsilon}(\boldsymbol{u}) := \frac{1}{2}\left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^{\mathsf{T}}\right), \tag{5}$$

by assuming infinitesimal displacements.

To this end, we can write the linear momentum balance and the boundary conditions as:

$$\nabla \cdot \boldsymbol{\sigma}'(\boldsymbol{u}) - \alpha \nabla \cdot (p\boldsymbol{I}) = \boldsymbol{f} \text{ in } \Omega \times \mathbb{T}, \tag{6}$$

$$\boldsymbol{u} = \boldsymbol{u}_D \text{ on } \partial\Omega_u \times \mathbb{T}, \tag{7}$$

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{\sigma}_{\boldsymbol{D}} \text{ on } \partial\Omega_t \times \mathbb{T}, \tag{8}$$

$$\boldsymbol{u} = \boldsymbol{u}_0 \text{ in } \Omega \text{ at } t = 0, \tag{9}$$

where $\boldsymbol{u}_D$ and $\boldsymbol{\sigma}_{\boldsymbol{D}}$ are prescribed displacement and traction at boundaries, respectively, and t is time.

The second equation is the mass balance equation, which is written as [45]:

$$\rho\left(\phi c_f + \frac{\alpha - \phi}{K_s}\right)\frac{\partial p}{\partial t} + \rho\alpha\frac{\partial(\nabla \cdot \boldsymbol{u})}{\partial t} - \nabla \cdot \boldsymbol{\kappa}(\nabla p - \rho\mathbf{g}) = g \text{ in } \Omega \times \mathbb{T}, \tag{10}$$

$$p = p_D \text{ on } \partial\Omega_p \times \mathbb{T}, \tag{11}$$

$$-\nabla \cdot \boldsymbol{\kappa}(\nabla p - \rho\mathbf{g}) \cdot \boldsymbol{n} = q_D \text{ on } \partial\Omega_q \times \mathbb{T}, \tag{12}$$

$$p = p_0 \text{ in } \Omega \text{ at } t = 0, \tag{13}$$

where ρ is a fluid density, ϕ is an initial porosity and remains constant throughout the simulation (the volumetric deformation is represented by $\frac{\partial(\nabla \cdot \boldsymbol{u})}{\partial t}$), c_f is fluid compressibility, $\mathbf{g}$ is a gravitational vector, which is neglected in this study, g is sink/source, p_D and q_D are specified pressure and flux, respectively, and $\boldsymbol{\kappa}$ is defined as:

$$\boldsymbol{\kappa} := \frac{\rho \boldsymbol{k_m}}{\mu}, \tag{14}$$

k_m is matrix permeability tensor and μ is fluid viscosity. By assuming that the rock volumetric displacement can cause the matrix permeability alteration, k_m is defined as [40,39]:

$$k_m = k_{m0} \frac{\left(1 + \frac{\varepsilon_v}{\phi}\right)^3}{1 + \varepsilon_v}, \tag{15}$$

where k_{m0} represent initial rock matrix permeability. Here, ε_v is the total volumetric strain, which is defined as:

$$\varepsilon_v := \mathrm{tr}(\boldsymbol{\varepsilon}) = \sum_{i=1}^{d} \varepsilon_{ii}. \tag{16}$$

Throughout this paper, we assume ρ and μ are constants; thus, κ varies only as a function of k_m throughout the simulation. Note that the inverse of Biot's modulus ($1/M$) is equivalent to $\phi c_f + \frac{\alpha - \phi}{K_s}$ term of (10) [46].

2.2. Numerical discretization

The domain Ω is partitioned into n_e elements, $\mathcal{T}_h$, which is the family of elements T (triangles in 2D, tetrahedrons in 3D). We will further denote by e a face of T, and $\boldsymbol{n}$ is a normal unit vector to e as illustrated in Fig. 1d. We denote h_T as the diameter of T and define $h := \max(h_T)$, which we may refer as mesh size hereafter. Let $\mathcal{E}_h$ denote the set of all facets, $\mathcal{E}_h^0$ the internal facets, $\mathcal{E}_h^D$ the Dirichlet boundary faces, and $\mathcal{E}_h^N$ denotes the Neumann boundary faces. $\mathcal{E}_h = \mathcal{E}_h^0 \cup \mathcal{E}_h^D \cup \mathcal{E}_h^N$, $\mathcal{E}_h^0 \cap \mathcal{E}_h^D = \emptyset$, $\mathcal{E}_h^0 \cap \mathcal{E}_h^N = \emptyset$, and $\mathcal{E}_h^D \cap \mathcal{E}_h^N = \emptyset$.

This study focuses on three function spaces, arising from CG, EG, and DG discretizations, respectively. We begin with defining the CG function space for displacement, $\boldsymbol{u}$ as

$$\mathcal{U}_h^{\mathrm{CG}_k}(\mathcal{T}_h) := \left\{ \boldsymbol{\psi_u} \in \mathbb{C}^0(\Omega; \mathbb{R}^d) : \boldsymbol{\psi_u}|_T \in \mathbb{P}_k(T; \mathbb{R}^d), \forall T \in \mathcal{T}_h, \boldsymbol{u} = \boldsymbol{u}_D \text{ on } \mathcal{E}_h^D \right\}, \tag{17}$$

where $\mathcal{U}_h^{\mathrm{CG}_k}(\mathcal{T}_h)$ is the space for the CG approximation with kth degree polynomials for the $\boldsymbol{u}$ unknown, $\mathbb{C}^0(\Omega; \mathbb{R}^d)$ denotes the space of vector-valued piecewise continuous polynomials, $\mathbb{P}_k(T; \mathbb{R}^d)$ is the space of polynomials of degree at most k over each element T, and $\boldsymbol{\psi_u}$ denotes a generic function of $\mathcal{U}_h^{\mathrm{CG}_k}(\mathcal{T}_h)$. Subsequently, the CG function space for scalar-valued pressure, p, is written as follows:

$$\mathcal{P}_h^{\mathrm{CG}_k}(\mathcal{T}_h) := \left\{ \psi_p \in \mathbb{C}^0(\Omega) : \psi_p|_T \in \mathbb{P}_k(T), \forall T \in \mathcal{T}_h \right\}. \tag{18}$$

Furthermore, we define the following DG function space for the pressure p:

$$\mathcal{P}_h^{\mathrm{DG}_k}(\mathcal{T}_h) := \left\{ \psi_p \in L^2(\Omega) : \psi_p|_T \in \mathbb{P}_k(T), \forall T \in \mathcal{T}_h \right\}, \tag{19}$$

where $\mathcal{P}_h^{\mathrm{DG}_k}(\mathcal{T}_h)$ is the space for the DG approximation with kth degree polynomials for the p space and $L^2(\Omega)$ is the space of square integrable functions. Finally, we define the EG function space for p as:

$$\mathcal{P}_h^{\mathrm{EG}_k}(\mathcal{T}_h) := \mathcal{P}_h^{\mathrm{CG}_k}(\mathcal{T}_h) + \mathcal{P}_h^{\mathrm{DG}_0}(\mathcal{T}_h), \tag{20}$$

where $\mathcal{P}_h^{\mathrm{EG}_k}(\mathcal{T}_h)$ denotes the space for the EG approximation with kth degree polynomials for the pressure and $\mathcal{P}_h^{\mathrm{DG}_0}(\mathcal{T}_h)$ is the space for the DG approximation with 0th degree polynomials, in other words, a piecewise constant approximation. Note that EG discretization is expected to be beneficial for an accurate approximation of the pressure p near interfaces where high permeability contrast is observed, for which a CG discretization may result in unphysical oscillations. In contrast, the mechanical properties that dictate the displacement field, $\boldsymbol{u}$, are assumed to be homogeneous, and the solid is compressible. Therefore, the CG discretization of the displacement unknown will suffice in the following.

We now seek approximate solutions $p_h \in \mathcal{P}_h^{\mathrm{EG}_k}(\mathcal{T}_h)$ and $\boldsymbol{u_h} \in \mathcal{U}_h^{\mathrm{CG}_k}(\mathcal{T}_h)$ of p and $\boldsymbol{u}$, respectively. The linear momentum balance equation, (6), is discretized using the CG method as follows [31,47,11]:

$$\sum_{T \in \mathcal{T}_h} \int_T \boldsymbol{\sigma}'(\boldsymbol{u}_h) : \nabla^s \boldsymbol{\psi_u} \, dV - \sum_{T \in \mathcal{T}_h} \int_T \alpha p_h^n \boldsymbol{I} : \nabla^s \boldsymbol{\psi_u} \, dV$$
$$= \sum_{T \in \mathcal{T}_h} \int_T \boldsymbol{f} \boldsymbol{\psi_u} \, dV + \sum_{e \in \mathcal{E}_h^N} \int_e \boldsymbol{\sigma_D} \boldsymbol{\psi_u} \, dS, \quad \forall \boldsymbol{\psi_u} \in \mathcal{U}_h^{\mathrm{CG}_k}(\mathcal{T}_h), \tag{21}$$

where $\int_T .dV$ and $\int_e .dS$ refer to volume and surface integrals, respectively. On the contrary, the mass balance equation, (10), is discretized using the EG method with symmetric internal penalty approach [31,48,14], and the space discretization is presented as follows::

$$\sum_{T \in \mathcal{T}_h} \int_T \rho \left(\phi c_f + \frac{\alpha - \phi}{K_s} \right) \frac{\partial p_h}{\partial t} \psi_p \, dV + \sum_{T \in \mathcal{T}_h} \int_T \rho \alpha \frac{\partial \nabla \cdot \boldsymbol{u}_h}{\partial t} \psi_p \, dV$$

$$+ \sum_{T \in \mathcal{T}_h} \int_T \boldsymbol{\kappa} \left(\boldsymbol{u_h} \right) \nabla p_h \cdot \nabla \psi_p \, dV - \sum_{e \in \mathcal{E}_h^0 \cup \mathcal{E}_h^D} \int_e \{ \boldsymbol{\kappa} \left(\boldsymbol{u_h} \right) \nabla p_h \}_{\delta_e} \cdot [\![\psi_p]\!] \, dS$$

$$- \sum_{e \in \mathcal{E}_h^0 \cup \mathcal{E}_h^D} \int_e \{ \boldsymbol{\kappa} \left(\boldsymbol{u_h} \right) \nabla \psi_p \}_{\delta_e} \cdot [\![p_h]\!] \, dS + \sum_{e \in \mathcal{E}_h^0 \cup \mathcal{E}_h^D} \int_e \frac{\beta}{h_e} \kappa \left(\boldsymbol{u_h} \right)_e [\![p_h]\!] \cdot [\![\psi_p]\!] \, dS \tag{22}$$

$$= \sum_{T \in \mathcal{T}_h} \int_T g \psi_p \, dV + \sum_{e \in \mathcal{E}_h^N} \int_e q_D \psi_p \, dS - \sum_{e \in \mathcal{E}_h^D} \int_e \boldsymbol{\kappa} \left(\boldsymbol{u_h} \right) \nabla \psi_p \cdot p_D \boldsymbol{n} \, dS$$

$$+ \sum_{e \in \mathcal{E}_h^D} \int_e \frac{\beta}{h_e} \kappa \left(\boldsymbol{u_h} \right)_e [\![\psi_p]\!] \cdot p_D \boldsymbol{n} \, dS, \quad \forall \psi_p \in \mathcal{P}_h^{\mathrm{EG}_k} \left(\mathcal{T}_h \right).$$

Here, $[\![X]\!]$ is a jump operator defined as:

$$[\![X]\!] = X^+ \boldsymbol{n}^+ + X^- \boldsymbol{n}^-,$$
$$[\![\boldsymbol{X}]\!] = \boldsymbol{X}^+ \cdot \boldsymbol{n}^+ + \boldsymbol{X}^- \cdot \boldsymbol{n}^-, \tag{23}$$

where $\boldsymbol{n}^+$ and $\boldsymbol{n}^-$ are the unit normal vector of T^+ and T^-, respectively (see Fig. 1d). In addition, $\{ X \}_{\delta_e}$ is a weighted average operator defined as follows:

$$\{ X \}_{\delta e} = \delta_e X^+ + (1 - \delta_e) X^-, \tag{24}$$

where δ_e, in this study, is calculated as [33]:

$$\delta_e := \frac{\kappa_e^-}{\kappa_e^+ + \kappa_e^-}, \tag{25}$$

$$\kappa_e^+ := \left(\boldsymbol{n}^+ \right)^T \cdot \boldsymbol{\kappa}^+ \cdot \boldsymbol{n}^+,$$
$$\kappa_e^- := \left(\boldsymbol{n}^- \right)^T \cdot \boldsymbol{\kappa}^- \cdot \boldsymbol{n}^-, \tag{26}$$

κ_e is a harmonic average of κ_e^+ and κ_e^- read as:

$$\kappa_e := \frac{2 \kappa_e^+ \kappa_e^-}{\left(\kappa_e^+ + \kappa_e^- \right)}. \tag{27}$$

The interior penalty parameter, β, is a function of polynomial degree approximation, k. Here, h_e is a characteristic length calculated as:

$$h_e := \frac{\mathrm{meas} \left(T^+ \right) + \mathrm{meas} \left(T^- \right)}{2 \, \mathrm{meas}(e)}, \tag{28}$$

where meas(.) represents a measurement operator, measuring length, area, or volume.

Remark 1. Note that the surface integral terms of interior facets, $e \in \mathcal{E}_h^0$, naturally become zero if p_h were sought in $\mathcal{P}_h^{\mathrm{CG}_k} \left(\mathcal{T}_h \right)$. In other words, if the CG method is used instead of EG discretization of the pressure, $[\![p_h]\!]$ and $[\![\psi_p]\!]$ are equal to zero. Thus, eventually, the CG, EG, and DG methods have the same (bi-)linear forms for both (21) and (22), but a different function space for the pressure is employed.

Remark 2. While the Neumann boundary condition is naturally applied on the boundary faces that belong to the Neumann boundary domain, $e \in \mathcal{E}_h^N$, the Dirichlet boundary condition is strongly enforced in (17) but weakly enforced for (18), (20) and (19) on the Dirichlet boundary faces, $e \in \mathcal{E}_h^D$.

The time domain, $\mathbb{T} = (0, \tau]$, is partitioned into N open subintervals such that, $0 =: t^0 < t^1 < \cdots < t^N := \tau$. The length of the subinterval, Δt^n, is defined as $\Delta t^n = t^n - t^{n-1}$ where n represents the current time step. In this study, implicit first-order time discretization is utilized for a time domain of (22) as shown below for both p_h^n and $\boldsymbol{u}_h^n$:

$$\frac{\partial p_h}{\partial t} \approx \frac{p_h^n - p_h^{n-1}}{\Delta t^n}, \text{ and } \frac{\partial \boldsymbol{u}_h}{\partial t} \approx \frac{\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}}{\Delta t^n}. \tag{29}$$

2.3. Fully discrete form and block formulation

Combining space and time discretizations from the previous section, we propose the fully discrete form and block formulation using displacement, $\boldsymbol{u}$, and pressure, p, as the primary variables. From the linear momentum balance equation (21), we can define the following forms:

$$\mathcal{A}_u\left((\boldsymbol{u}_h^n, p_h^n), \boldsymbol{\psi_u}\right) := a\left(\boldsymbol{u}_h^n, \boldsymbol{\psi_u}\right) + b\left(p_h^n, \boldsymbol{\psi_u}\right), \tag{30}$$

with

$$a\left(\boldsymbol{u}_h^n, \boldsymbol{\psi_u}\right) := \sum_{T \in \mathcal{T}_h} \int_T \boldsymbol{\sigma}'\left(\boldsymbol{u}_h^n\right) : \nabla^s \boldsymbol{\psi_u}\, dV, \tag{31}$$

$$b\left(p_h^n, \boldsymbol{\psi_u}\right) := -\sum_{T \in \mathcal{T}_h} \int_T \alpha p_h^n \boldsymbol{I} : \nabla^s \boldsymbol{\psi_u}\, dV, \tag{32}$$

and

$$\mathcal{L}_u\left(\boldsymbol{\psi_u}\right) := \sum_{T \in \mathcal{T}_h} \int_T \boldsymbol{f}\boldsymbol{\psi_u}\, dV + \sum_{e \in \mathcal{E}_h^N} \int_e \boldsymbol{\sigma_D}\boldsymbol{\psi_u}\, dS. \tag{33}$$

and, equivalently, solve

$$\mathcal{A}_u\left((\boldsymbol{u}_h^n, p_h^n), \boldsymbol{\psi_u}\right) = \mathcal{L}_u\left(\boldsymbol{\psi_u}\right), \quad \forall \boldsymbol{\psi_u} \in \mathcal{U}_h^{\mathrm{CG}_k}\left(\mathcal{T}_h\right) \tag{34}$$

at each time step t^n.

Subsequently, from Equations (22) and (29), we define the following forms:

$$\mathcal{A}_p\left((\boldsymbol{u}_h^n, p_h^n), \psi_p; \boldsymbol{u}_h^n\right) := c\left(\boldsymbol{u}_h^n, \psi_p\right) + d\left(p_h^n, \psi_p; \boldsymbol{u}_h^n\right), \tag{35}$$

with

$$c\left(\boldsymbol{u}_h^n, \psi_p\right) := \sum_{T \in \mathcal{T}_h} \int_T \rho \alpha \nabla \cdot \boldsymbol{u}_h^n \psi_p\, dV, \tag{36}$$

$$\begin{aligned}
d\left(p_h^n, \psi_p; \boldsymbol{u}_h^n\right) := &\sum_{T \in \mathcal{T}_h} \int_T \rho \left(\phi c_f + \frac{\alpha - \phi}{K_s}\right) p_h^n \psi_p\, dV + \Delta t^n \sum_{T \in \mathcal{T}_h} \int_T \kappa\left(\boldsymbol{u_h^n}\right) \nabla p_h^n \cdot \nabla \psi_p\, dV \\
&- \Delta t^n \sum_{e \in \mathcal{E}_h^0 \cup \mathcal{E}_h^D} \int_e \left\{\kappa\left(\boldsymbol{u_h^n}\right) \nabla p_h^n\right\}_{\delta_e} \cdot [\![\psi_p]\!]\, dS \\
&- \Delta t^n \sum_{e \in \mathcal{E}_h^0 \cup \mathcal{E}_h^D} \int_e \left\{\kappa\left(\boldsymbol{u_h^n}\right) \nabla \psi_p\right\}_{\delta_e} \cdot [\![p_h^n]\!]\, dS \\
&+ \Delta t^n \sum_{e \in \mathcal{E}_h^0 \cup \mathcal{E}_h^D} \int_e \frac{\beta}{h_e} \kappa\left(\boldsymbol{u_h^n}\right)_e [\![p_h^n]\!] \cdot [\![\psi_p]\!]\, dS,
\end{aligned} \tag{37}$$

and

$$\begin{aligned}
\mathcal{L}_p\left(\psi_p; \boldsymbol{u}_h^n\right) := &\sum_{T \in \mathcal{T}_h} \int_T \rho \alpha \nabla \cdot \boldsymbol{u}_h^{n-1} \psi_p\, dV + \sum_{T \in \mathcal{T}_h} \int_T \rho \left(\phi c_f + \frac{\alpha - \phi}{K_s}\right) p_h^{n-1} \psi_p\, dV \\
&+ \Delta t^n \sum_{T \in \mathcal{T}_h} \int_T g \psi_p\, dV + \Delta t^n \sum_{e \in \mathcal{E}_h^N} \int_e q_D \psi_p\, dS \\
&- \Delta t^n \sum_{e \in \mathcal{E}_h^D} \int_e \kappa\left(\boldsymbol{u_h^n}\right) \nabla \psi_p \cdot p_D \boldsymbol{n}\, dS + \Delta t^n \sum_{e \in \mathcal{E}_h^D} \int_e \frac{\beta}{h_e} \kappa\left(\boldsymbol{u_h^n}\right)_e [\![\psi_p]\!] \cdot p_D \boldsymbol{n}\, dS.
\end{aligned} \tag{38}$$

and, equivalently, solve

$$\mathcal{A}_p\left((\boldsymbol{u}_h^n, p_h^n), \psi_p; \boldsymbol{u}_h^n\right) = \mathcal{L}_p\left(\psi_p; \boldsymbol{u}_h^n\right), \quad \forall \psi_p \in \mathcal{P}_h^{\mathrm{EG}_k}\left(\mathcal{T}_h\right) \tag{39}$$

at each time step t^n. We remark that (35), (37) and (38) are nonlinear forms in the displacement $\boldsymbol{u}_h^n$ (the nonlinear variable being reported after a semicolon for the sake of clarity), since $\kappa := \kappa(\boldsymbol{u}_h^n)$ depends on the strain $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}(\boldsymbol{u}_h^n)$ (see (14)–(16)). Thus, two linearization strategies are sought in this work:

1. *Pressure-independent k_m model*, which amounts to replacing $\kappa\left(u_h^n\right)$ with $\kappa\left(u_h^0\right)$ in (37) and (38), that is, employing a semi-implicit time-stepping scheme. It preserves the compaction effect, i.e. changing in ε_v, but it neglects the effect of the solid deformation on k_m alteration as illustrated in (15). Therefore, we replace (39) with

$$\mathcal{A}_p\left((u_h^n, p_h^n), \psi_p; u_h^0\right) = \mathcal{L}_p\left(\psi_p; u_h^0\right), \quad \forall \psi_p \in \mathcal{P}_h^{\mathrm{EG}_k}\left(\mathcal{T}_h\right), \tag{40}$$

and solve for (34) and (40). The resulting system of equations is linear, and its algebraic formulation is summarized hereafter in Remark 3. The algorithm used to solve this model is presented in Algorithm 1. A direct solver is used for every time step [49]. For the sake of brevity, in Algorithm 1 (and in the following one) we omit the subscript h as well as superscripts CG_k, DG_k or EG_k, $\mathcal{W}$ represents a mixed function space between $\mathcal{U} \times \mathcal{P}$, and w is a solution vector composed of u and p.

2. *Pressure-dependent k_m model*, which includes the nonlinear effect of the current solid deformation on k_m alteration by performing Picard iterations. k_m is affected by the change in ε_v as reflected in (15). Note that κ is calculated using k_m, ρ, and μ. Given that ρ and μ are constants, κ is only a function of k_m. Thus, we replace (34) and (39) by

$$\begin{cases} \mathcal{A}_u\left((u_h^{n,\iota}, p_h^{n,\iota}), \psi_u\right) = \mathcal{L}_u\left(\psi_u\right), \quad \forall \psi_u \in \mathcal{U}_h^{\mathrm{CG}_k}\left(\mathcal{T}_h\right) \\ \mathcal{A}_p\left((u_h^{n,\iota}, p_h^{n,\iota}), \psi_p; u_h^{n,\iota-1}\right) = \mathcal{L}_p\left(\psi_p; u_h^{n,\iota-1}\right), \quad \forall \psi_p \in \mathcal{P}_h^{\mathrm{EG}_k}\left(\mathcal{T}_h\right), \end{cases} \tag{41}$$

where ι denotes the iteration counter and $(u_h^{n,\iota}, p_h^{n,\iota})$ represents the solution of time step n at Picard iteration ι. Non-linear iterations are initialized with $(u_h^{n,0}, p_h^{n,0}) = (u_h^{n-1}, p_h^{n-1})$. The algorithm utilized to solve this model is presented in Algorithm 2. The iteration tolerance, ξ, is set to be 1×10^{-6}, and a direct solver is used inside each iteration for solving the resulting system of linear equations [49]. The L^2 norm of the difference, ζ, between the current iteration, $w^\iota := (u_h^{n,\iota}, p_h^{n,\iota})$, and the previous iteration, $w^{\iota-1} := (u_h^{n,\iota-1}, p_h^{n,\iota-1})$ is calculated as follows:

$$\zeta = ||w^\iota - w^{\iota-1}||, \tag{42}$$

where $|| \cdot ||$ denotes the L^2 norm on the product space $\mathcal{W}$.

Algorithm 1 Solving algorithm for pressure-independent k_m model. (Syntax here resembles the FEniCS implementation.)

$\mathcal{U}$ = VectorFunctionSpace('CG', k = 2)
$\mathcal{P}$ = FunctionSpace('CG/EG/DG', k = 1)
u = Function($\mathcal{U}$)   ▷ solution in the space $\mathcal{U}$
p = Function($\mathcal{P}$)   ▷ solution in the space $\mathcal{P}$
w = (u, p)
1: Initialize all input parameters   ▷ initial reservoir pressure (p_0) must be provided.
2: Solve the equilibrium state for u_0
3: **while** $t < \tau$ **do**
4:   $t \mathrel{+}= \Delta t^n$
5:   Solve time step n
6:   Post-processing, i.e. σ, residual, etc.
7:   Assign $w^n \to w^{n-1}$ ▷ update time step $n-1$
8:   Output

Remark 3 *(Algebraic formulation)*. We discuss the algebraic formulation of the pressure-independent k_m model for CG, EG, and DG pressure function spaces. A similar formulation can be derived for the pressure-dependent case.

Let $\{\psi_u^{(i)}\}_{i=1}^{\mathcal{N}_u}$ denote the set of basis functions of $\mathcal{U}_h^{\mathrm{CG}_k}(\mathcal{T}_h)$, i.e. $\mathcal{U}_h^{\mathrm{CG}_k}(\mathcal{T}_h) = \mathrm{span}\{\psi_u^{(i)}\}_{i=1}^{\mathcal{N}_u}$, having denoted by $\mathcal{N}_u$ the number of DOF for the CG displacement space. In a similar way, let $\{\psi_{p,\pi}^{(m)}\}_{m=1}^{\mathcal{N}_{p,\pi}}$ be the set of basis functions for the space $\mathcal{P}_h^{\pi_k}(\mathcal{T}_h)$, where π can mean either CG, EG, or DG. Hence, three mixed function spaces, **(1)** $\mathrm{CG}_k \times \mathrm{CG}_k$, **(2)** $\mathrm{CG}_k \times \mathrm{EG}_k$, and **(3)** $\mathrm{CG}_k \times \mathrm{DG}_k$ where k represents the degree of polynomial approximation, are possible, and will be compared in the numerical examples in the next section. The Jacobian matrix corresponding to the left-hand sides of (34) and (40) is assembled composing the following blocks:

$$\begin{aligned} \left[\mathcal{J}_{uu}^{\mathrm{CG}_k \times \mathrm{CG}_k}\right]_{ij} &:= a\left(\psi_u^{(j)}, \psi_u^{(i)}\right), \ i = 1, \ldots, \mathcal{N}_u, j = 1, \ldots, \mathcal{N}_u, \\ \left[\mathcal{J}_{up}^{\mathrm{CG}_k \times \pi_k}\right]_{im} &:= b\left(\psi_{p,\pi}^{(m)}, \psi_u^{(i)}\right), \ i = 1, \ldots, \mathcal{N}_u, m = 1, \ldots, \mathcal{N}_{p,\pi}, \\ \left[\mathcal{J}_{pu}^{\pi_k \times \mathrm{CG}_k}\right]_{lj} &:= c\left(\psi_u^{(j)}, \psi_{p,\pi}^{(l)}\right), \ l = 1, \ldots, \mathcal{N}_{p,\pi}, j = 1, \ldots, \mathcal{N}_u, \\ \left[\mathcal{J}_{pp}^{\pi_k \times \pi_k}\right]_{lm} &:= d\left(\psi_{p,\pi}^{(m)}, \psi_{p,\pi}^{(l)}; u_h^n\right) l = 1, \ldots, \mathcal{N}_{p,\pi}, m = 1, \ldots, \mathcal{N}_{p,\pi}. \end{aligned} \tag{43}$$

Algorithm 2 Solving algorithm for pressure-dependent k_m model. (Syntax here resembles the FEniCS implementation.)

$\mathcal{U}$ = VectorFunctionSpace('CG', k = 2)
$\mathcal{P}$ = FunctionSpace('CG/EG/DG', k = 1)
$\boldsymbol{u}$ = Function($\mathcal{U}$)   ▷ solution in the space $\mathcal{U}$
p = Function($\mathcal{P}$)   ▷ solution in the space $\mathcal{P}$
w = ($\boldsymbol{u}$, p)
 1: Initialize all input parameters   ▷ initial reservoir pressure (p_0) must be provided.
 2: Solve the equilibrium state for $\boldsymbol{u_0}$
 3: Update ε_{v0} and $\boldsymbol{\kappa}_0$ ▷ (16) and (15)
 4: **while** $t < \tau$ **do**
 5:     $t \mathrel{+}= \Delta t^n$
 6:     Solve time step n
 7:     Update ε_v^0 and $\boldsymbol{\kappa}^0$, assign $\iota \to 0$ ▷ iteration zero
 8:     **while** $\zeta > \xi$ **do**
 9:         $\iota \mathrel{+}= 1$
10:         Solve time step n and iteration ι
11:         Update ε_v^ι and $\boldsymbol{\kappa}^\iota$ ▷ iteration ι
12:         **if** $\kappa^\iota < \kappa_r$ **then** $\kappa^\iota = \kappa_r$ ▷ κ_r is a minimum value of κ
13:         Update ζ ▷ (42)
14:         Assign $w^\iota \to w^{\iota-1}$
15:     Assign $\varepsilon_v^\iota \to \varepsilon_v^n$ ▷ time step n
16:     Assign $\boldsymbol{\kappa}^\iota \to \boldsymbol{\kappa}^n$ ▷ time step n
17:     Post-processing, i.e. $\boldsymbol{\sigma}$, residual, etc.
18:     Assign $w^n \to w^{n-1}$ ▷ update time step $n-1$
19:     Output

In a similar way, the right-hand side of (34) and (40) gives rise to a block vector of components

$$\left[\mathcal{L}_u^{CG_k}\right]_i := \mathcal{L}_u\left(\boldsymbol{\psi}_{\boldsymbol{u}}^{(i)}\right), \; i = 1, \ldots, \mathcal{N}_{\boldsymbol{u}},$$
$$\left[\mathcal{L}_p^{\pi_k}\right]_l := \mathcal{L}_p\left(\psi_{p,\pi}^{(l)}; \boldsymbol{u}_h^n\right) \; l = 1, \ldots, \mathcal{N}_{p,\pi}. \tag{44}$$

The resulting block structure is thus

$$\begin{bmatrix} \mathcal{J}_{uu}^{CG_k \times CG_k} & \mathcal{J}_{up}^{CG_k \times \pi_k} \\ \mathcal{J}_{pu}^{\pi_k \times CG_k} & \mathcal{J}_{pp}^{\pi_k \times \pi_k} \end{bmatrix} \left\{ \begin{array}{c} (\boldsymbol{u}_h^n)^{CG_k} \\ (p_h{}^n)^{\pi_k} \end{array} \right\} = \left\{ \begin{array}{c} \mathcal{L}_u^{CG_k} \\ \mathcal{L}_p^{\pi_k} \end{array} \right\}. \tag{45}$$

where $(\boldsymbol{u}_h^n)^{CG_k}$ and $(p_h{}^n)^{\pi_k}$ collect the degrees of freedom for displacement and pressure.

Finally, we remark that (owing to (20)) the case $\pi = \text{EG}$ can be equivalently decomposed into a $CG_k \times (CG_k \times DG_0)$ mixed function space, resulting in:

$$\begin{bmatrix} \mathcal{J}_{uu}^{CG_k \times CG_k} & \mathcal{J}_{up}^{CG_k \times CG_k} & \mathcal{J}_{up}^{CG_k \times DG_0} \\ \mathcal{J}_{pu}^{CG_k \times CG_k} & \mathcal{J}_{pp}^{CG_k \times CG_k} & \mathcal{J}_{pp}^{CG_k \times DG_0} \\ \mathcal{J}_{pu}^{DG_0 \times CG_k} & \mathcal{J}_{pp}^{DG_0 \times CG_k} & \mathcal{J}_{pp}^{DG_0 \times DG_0} \end{bmatrix} \left\{ \begin{array}{c} (\boldsymbol{u}_h^n)^{CG_k} \\ (p_h{}^n)^{CG_k} \\ (p_h{}^n)^{DG_0} \end{array} \right\} = \left\{ \begin{array}{c} \mathcal{L}_u^{CG_k} \\ \mathcal{L}_p^{CG_k} \\ \mathcal{L}_p^{DG_0} \end{array} \right\}. \tag{46}$$

This implementation makes the EG methodology easily implementable in any existing DG codes. For all the computations, matrices and vectors are built using the FEniCS form compiler [42]. The block structure is assembled by using the multiphenics toolbox [41], in which users are only required to provide weak forms associated to each block, as it will be illustrated in the results and discussion section. Solvers are employed from PETSc package [49]. All simulations are computed on XeonE5_2650v4 with a single thread. Random fields, porosity (ϕ) and permeability ($\boldsymbol{k_{m0}}$), are populated using SciPy package [50]. β, penalty parameter, is set at 0.95 and 0.9 for DG and EG methods, respectively. Examples of scripts used in this study are provided in https://github.com/teeratornk/jcp_YJCPH_110030_git.

Remark 4. As discussed above, Picard iteration is employed to solve the pressure-dependent $\boldsymbol{k_m}$ model (Algorithm 2). The Picard iteration is employed in this study because the permeability calculation resulted from (15) can be negative. Therefore, we want to enforce the minimum value of $\boldsymbol{k_m}$ through Picard iteration, as illustrated in Algorithm 2. Throughout this paper, we set κ_r, a minimum value of κ, as 10^{-16}. Note that κ is a function of $\boldsymbol{k_m}$, ρ, and μ as presented in (14). As a future investigation to reduce the number of iterations, the Newton method could be applied to solve this system of nonlinear equations while enforcing the κ_r value.
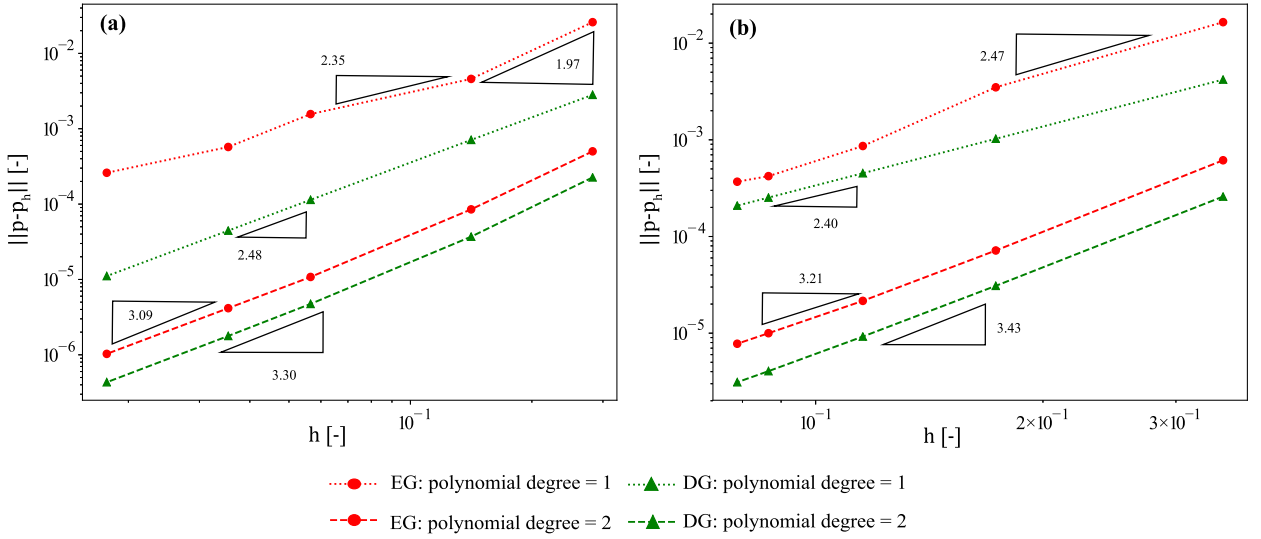
Fig. 2. Error convergence plot between EG and DG methods with polynomial approximation degree 1 and 2 of (**a**) 2-dimension and (**b**) 3-dimension cases for the Poisson equation.

3. Results and discussion

3.1. Error convergence analysis

To verify and illustrate the implementation of the developed block structure utilized to compose the EG function space using multiphenics framework [41], we start by analyzing and comparing the error convergence of the EG method with the DG method for the Poisson equation with a source term as follows:

$$-\nabla^2 p = g, \tag{47}$$

where p is primary variable and g is sink/source term. Note that (47) is a specific case of (10) at the steady-state solution, and $\alpha = \phi = 0$ (uncoupled from (6)). We take $\Omega = [0, 1]^d$, and $d = 2, 3$ and choose the exact solution in Ω as:

$$\begin{cases} p(x, y) = -\cos(x + y), & \text{if } d = 2 \\ p(x, y, z) = -\cos(x + y + z), & \text{if } d = 3 \end{cases} \tag{48}$$

where x, y, and z represent points in x-, y-, z-direction, respectively. Subsequently, g is chosen as:

$$\begin{cases} g(x, y) = -2\cos(x + y), & \text{if } d = 2 \\ g(x, y, z) = -3\cos(x + y + z), & \text{if } d = 3 \end{cases} \tag{49}$$

to satisfy the exact solution. The block structure of the EG method is formed as:

$$\begin{bmatrix} \mathcal{J}_{pp}^{CG_k \times CG_k} & \mathcal{J}_{pp}^{CG_k \times DG_0} \\ \mathcal{J}_{pp}^{DG_0 \times CG_k} & \mathcal{J}_{pp}^{DG_0 \times DG_0} \end{bmatrix} \left\{ \begin{array}{c} (p_h)^{CG_k} \\ (p_h)^{DG_0} \end{array} \right\} = \left\{ \begin{array}{c} \mathcal{L}_p^{CG_k} \\ \mathcal{L}_p^{DG_0} \end{array} \right\}. \tag{50}$$

We calculate L^2 norm of the difference between the exact solution, p, and approximated solution, p_h (2a-b for $d = 2, 3$). These plots show L^2 norm of the difference, $||p - p_h||$, versus the mesh size, h. For both $d = 2, 3$ cases, the EG and DG methods provide the expected convergence rate of two and three for polynomial degree approximation, k, of one and two, respectively [51].

We also want to emphasize the simplicity of the EG implementation using multiphenics framework [41]. This open-source platform is built on top of FEniCS [42], and it takes care of assembling the left-hand matrix and the right-hand side of a block structure. As illustrated in the code snippet below, users are only required to provide weak formulations associated to each block. This simplicity will enable a broader range of users to employ the EG implementation in their research.

Listing 1 Illustration of the EG implementation using multiphenics.

```python
from dolfin import *
import multiphenics as mp

#start initialization
#....
#end initialization

CGk = FunctionSpace(mesh, 'CG', k)
DG0 = FunctionSpace(mesh, 'DG', 0)

W = mp.BlockFunctionSpace([CGk, DG0])

CGkDG0_trial = mp.BlockTrialFunction(W)
(p_CGk, p_DG0) = mp.block_split(CGkDG0_trial)
CGkDG0_test = mp.BlockTestFunction(W)
(psi_CGk, psi_DG0) = mp.block_split(CGkDG0_test)

J11 = inner(grad(p_CGk),grad(psi_CGk))*dx \
      - inner(grad(p_CGk),n)*psi_CGk*ds \
      + beta/h*p_CGk*psi_CGk*ds

J12 = beta/h*p_DG0*psi_CGk*ds

J21 = - inner(avg(grad(p_CGk)),jump(psi_DG0,n))*dS \
      - inner(grad(p_CGk),n)*psi_DG0*ds \
      + beta/h*p_CGk*psi_DG0*ds

J22 = beta/h_e*inner(jump(p_DG0,n),jump(psi_DG0,n))*dS \
      + beta/h*p_DG0*psi_DG0*ds

lhs = [[J11, J12],
       [J21, J22]]

f_CGk = Expression('3.0*cos(x[0]+x[1]+x[2])', element = CGk.ufl_element())
f_DG0 = Expression('3.0*cos(x[0]+x[1]+x[2])', element = DG0.ufl_element())
p_CGk = Expression('cos(x[0]+x[1]+x[2])', element = CGk.ufl_element())
p_DG0 = Expression('cos(x[0]+x[1]+x[2])', element = DG0.ufl_element())

l1 = f_CGk*psi_CGk*dx + beta/h*p_CGk*psi_CGk*ds
l2 = f_DG0*psi_DG0*dx + beta/h*p_DG0*psi_DG0*ds

rhs = [l1, l2]

A = mp.block_assemble(rhs)
F = mp.block_assemble(lhs)

sol = BlockFunction(W)
block_solve(A, sol.block_vector(), F)

#start post-processing
#....
#end post-processing
```

3.2. Terzaghi's 1D consolidation problem

The numerical discretizations, CG, EG, and DG methods, discussed in the methodology section are verified using Terzaghi's 1D consolidation problem [43]. Assuming the reservoir is homogeneous, isotropic, and saturated with a single-phase fluid, the non-dimensional parameters are defined as follows [31]:

$$p^* := \frac{p}{\sigma}, \ z^* := \frac{z}{H}, \ t^* := \frac{c_v}{H^2}t, \tag{51}$$

where p is the fluid pressure, σ is far-field stress or external load, z is the distance from the drainage boundary, H is the domain thickness, and c_v is the coefficient of consolidation, which is calculated by:

$$c_v = 3K\left(\frac{1-\nu}{1+\nu}\right)\frac{k_{m,y}}{\mu}. \tag{52}$$

where $k_{m,y}$ is the matrix permeability in y-direction. Subsequently, the pressure solution of Terzaghi's 1D consolidation is written as follows:
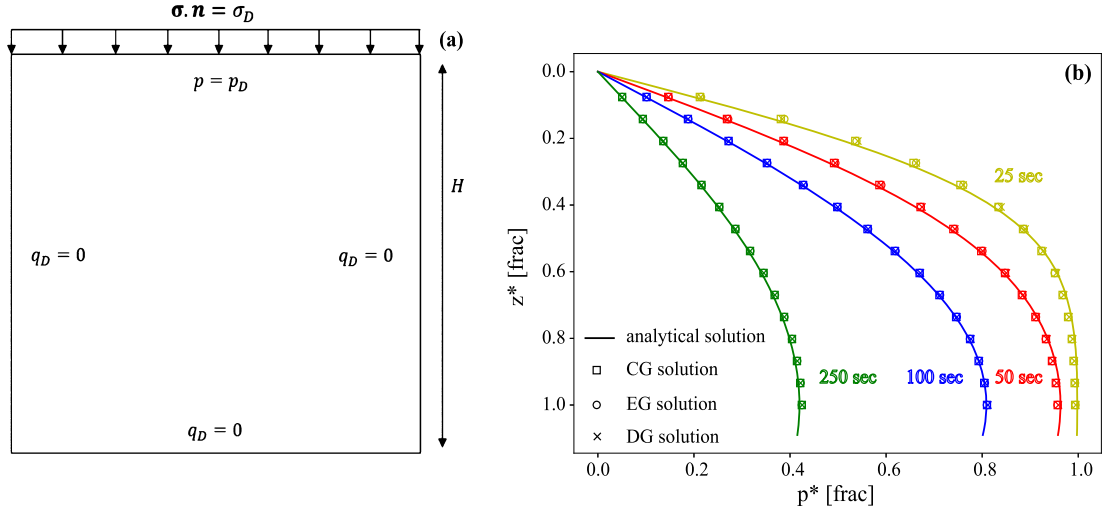
Fig. 3. (**a**) geometry and boundary conditions and (**b**) the verification results of the CG, EG, and DG methods for Terzaghi's 1D consolidation problem.

$$p^*\left(z^*, t^*\right) = \sum_{m=0}^{\infty} \frac{2}{M} \sin\left(Mz^*\right) e^{-M^2 t^*}, \tag{53}$$

where $M = \pi(2m+1)/2$. Geometry and boundary conditions utilized to verify the numerical models are presented in Fig. 3a, and input parameters are $\sigma_D = [0, 1]$ kPa, $p_D = 0$ Pa, $k_m = 10^{-12}I$ m^2, $\mu = 10^{-6}$ kPa.s, $\rho = 1000$ kg/m^3, $K = 1000$ kPa, mesh size(h) = 0.05 m, $K_s \approx \infty$ kPa, which leads to $\alpha \approx 1$, and $\nu = 0.25$, $\Delta t^n = 1.0$ sec, λ_l and μ_l are calculated by the following equations:

$$\lambda_l = \frac{3K\nu}{1+\nu}, \text{ and } \mu_l = \frac{3K(1-2\nu)}{2(1+\nu)}. \tag{54}$$

Results of CG, EG, and DG methods are presented in Fig. 3b illustrating reasonable matches between analytical and numerical solutions for all time steps, 25, 50, 100, and 250 sec. The solutions of CG, EG, and DG methods are approximately similar. More details regarding the convergence rate of the EG solution for Terzaghi's equation can be found in Choo and Lee [31].

3.3. 1D consolidation of a two-layered material

This example is adapted from the previous studies, and it aims to evaluate the pressure, p, and volumetric strain, ε_v, results in two-layered material, which contain a large different permeability, among CG, EG, and DG methods [31]. Fig. 4a presents a geometry and boundary conditions utilized in this investigation. Although most of the input parameters, mesh size, and solving algorithm are similar to the validation case, in this example, $k_m = k_1 I$, if $1 > x > 0.5$ and $k_m = k_2 I$, if $0.5 > x > 0$, and $k_1 = 10^{-12}$ m^2 and $k_2 = 10^{-16}$ m^2.

Fig. 4b shows the pressure solutions among the CG, EG, and DG methods. CG result illustrates unphysical (spurious) pressure oscillations at the interface, which complies with the previous studies [31,10]. The EG and DG methods, on the other hand, provide a smooth pressure solution for all time steps. These results may reflect the fact that the CG method, in its classical form, does not conserve mass locally.

In addition to the pressure oscillation, the CG method also delivers the oscillation in displacement, u, which affects the ε_v calculation as illustrated in Fig. 4c. These oscillations may lead to incorrect fluid flux calculation that may revoke the local mass conservation property of the CG method. Besides, the non-physical oscillation in ε_v may cause incorrect permeability alteration. As expected, there is no ε_v oscillation produced by the EG and DG methods for all time steps.

3.4. 2D flow in a deformable media with structured heterogeneity

The geometry, input properties, mesh, and boundary conditions used in this example are presented in Fig. 5. This example aims to compare the local mass conservative property, flux approximation, $\bar{\kappa}$ and $\bar{\varepsilon}_v$ alteration, number of iterations, and DOF among CG, EG, and DG methods. Note that $\bar{\kappa}$ and $\bar{\varepsilon}_v$ are an arithmetic average of κ and ε_v, respectively. The model input parameters are $\sigma_D = [0, 20]$ MPa, $p_D = 1$ MPa, $p_0 = 10$ MPa, $\alpha = 0.79$, $c_f = 1 \times 10^{-10}$ Pa, $\rho = 1000$ kg/m^3, $\mu = 10^{-6}$ kPa s, $h = 0.025$ m, and $\nu = 0.2$. K_S, $\Delta t^n = 1.0$ sec, λ_l, and μ_l are calculated by (54). k_m is assigned to each subdomain as shown in Fig. 5a, and $k_1 = 10^{-12}I$ m^2 and $k_2 = 10^{-16}I$ m^2. Here, K is varied for three values: $K = 8$, 2, and 1 GPa for case 1, 2, and 3, respectively. The input values of case 1, $K = 8$ GPa, are used according to Jaeger et al. [44]. This setting, three different K values, is proposed to evaluate the interaction between discretization method and saturated rock stiffness (K).
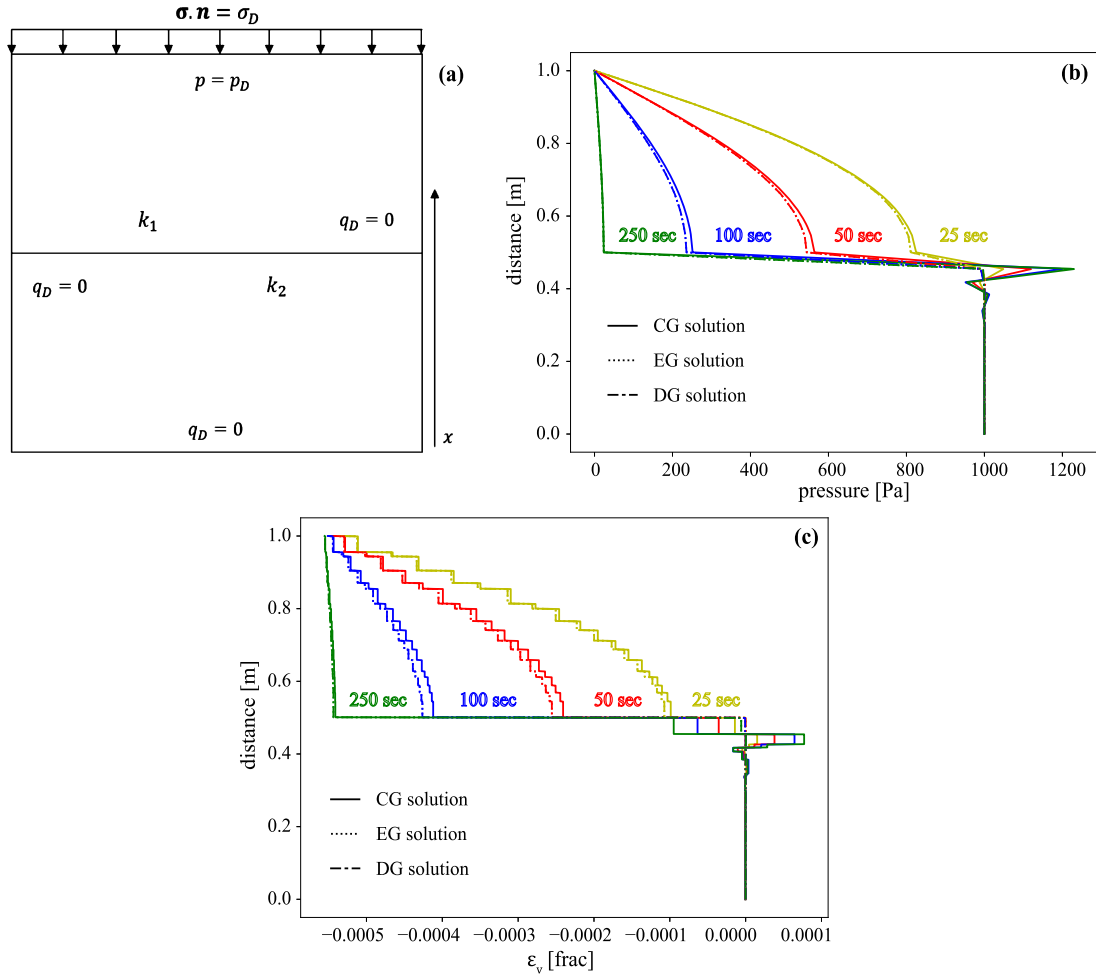
Fig. 4. Two-layered material example:(**a**) geometry and boundary conditions, (**b**) pressure results, and (**c**) ε_v results of the CG, EG, and DG methods. Note that the solutions of EG and DG methods are overlapping.
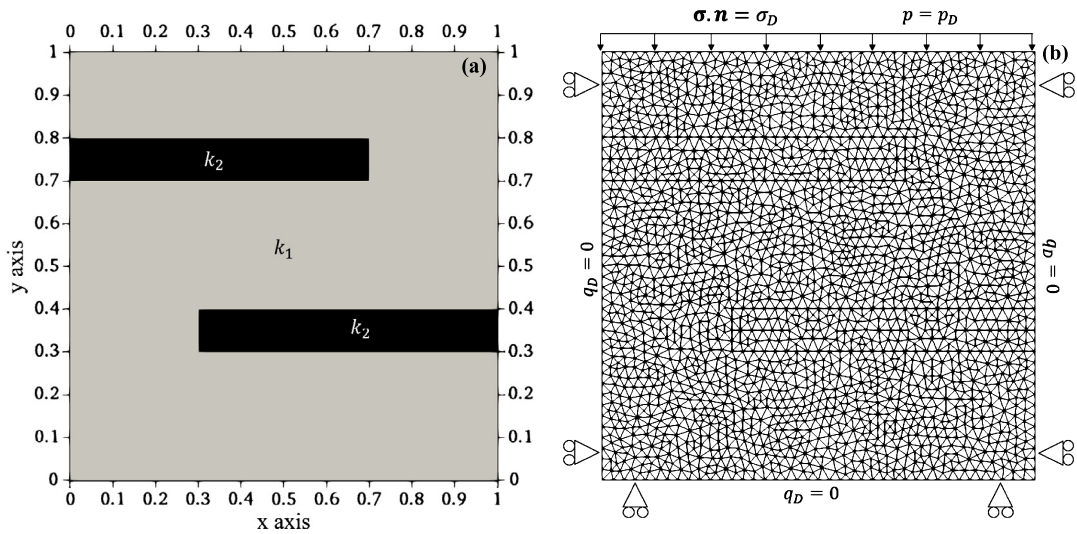

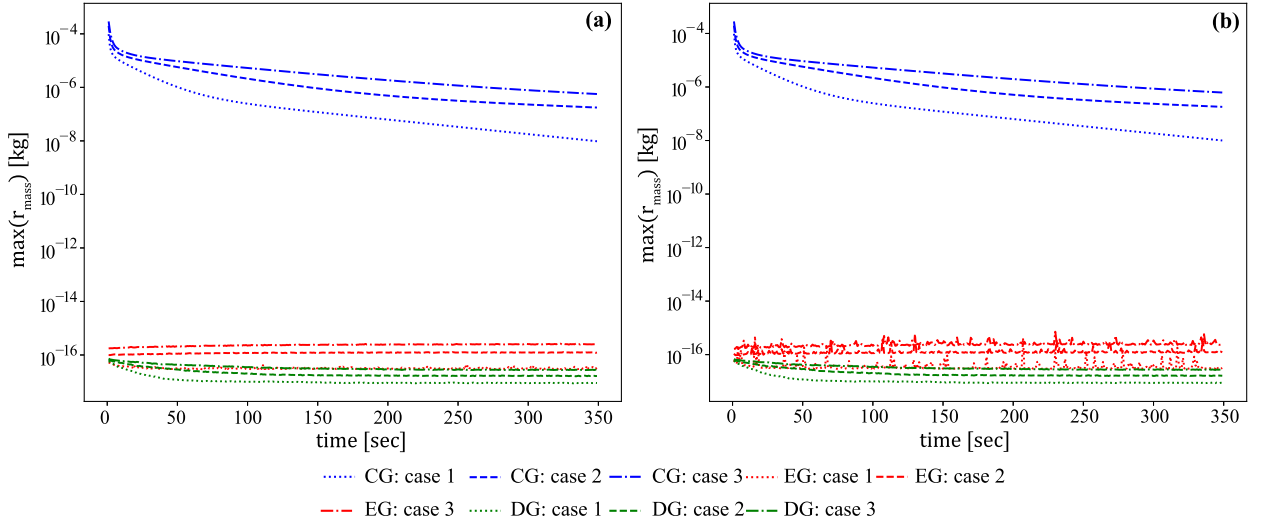**Fig. 5.** Structured heterogeneity example: (**a**) geometry and properties (**b**) mesh and boundary conditions.

Fig. 6. Structured heterogeneity example: the comparison of max(r$_{mass}$) values among the CG (blue line), EG (red line), and DG (green line) methods using (**a**) pressure-independent k_m and (**b**) pressure-dependent k_m models for case 1, $K = 8$ GPa, case 2, $K = 2$ GPa, and case 3, $K = 1$ GPa. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

The local mass conservation of each cell at each time step, r$_{mass}^n$, is calculated as follows:

$$\mathrm{r}^n{}_{mass} := \int_T \rho \left(\phi c_f + \frac{\alpha - \phi}{K_s} \right) \frac{p_h^n - p_h^{n-1}}{\Delta t^n} + \rho \alpha \nabla \cdot \frac{\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}}{\Delta t^n} dV + \sum_{e \in \mathcal{E}_h} \int_e \boldsymbol{v}_h^n \cdot \boldsymbol{n}|_e \, dS, \tag{55}$$

where the numerical flux, $\boldsymbol{v}_h^n \cdot \boldsymbol{n}|_e$, is described as follows [32,37]:

$$\boldsymbol{v}_h^n \cdot \boldsymbol{n}|_e = - \left\{ \kappa \left(\boldsymbol{u_h^n} \right) \nabla p_h^n \cdot \boldsymbol{n} \right\} + \frac{\beta}{h_e} \kappa \left(\boldsymbol{u_h^n} \right) [\![p_h^n]\!] \quad \forall e \in \mathcal{E}_h^0, \tag{56}$$

$$\boldsymbol{v}_h^n \cdot \boldsymbol{n}|_e = q_D \quad \forall e \in \mathcal{E}_h^N, \tag{57}$$

$$\boldsymbol{v}_h^n \cdot \boldsymbol{n}|_e = - \kappa \left(\boldsymbol{u_h^n} \right) \nabla p_h^n \cdot \boldsymbol{n} + \frac{\beta}{h_e} \kappa \left(\boldsymbol{u_h^n} \right) \left(p_h^n - p_D \right) \quad \forall e \in \mathcal{E}_h^D, \tag{58}$$

The local mass conservation results of structured heterogeneity example between pressure-independent k_m and pressure-dependent k_m models are presented in Figs. 6a-b, respectively. The local mass conservation property is well preserved for the EG and DG methods, i.e. the maximum value of r$_{mass}$, max(r$_{mass}$), is close to zero. The CG results, on the contrary, have much higher max(r$_{mass}$) value. The lack of local mass conservation of the CG method is the effect induced by the spurious p and ε_v oscillations that lead to the incorrect flux approximation. Moreover, cases that contain a lower K value have higher max(r$_{mass}$) value because the change in $\boldsymbol{u}$, and subsequently, p are more drastic in the softer material.

Then, we investigate the error in the flux approximation using a recovery factor (RF), which is defined at each time step as follows:

$$\mathrm{RF}^n := \frac{\sum_{t=0}^{t^n} \sum_{e \in \mathcal{E}_h} \int_e \boldsymbol{v}_h^n \cdot \boldsymbol{n}|_e \, dS}{\rho V_0 \phi}, \tag{59}$$

where V_0 is an initial reservoir volume. In this example, $\boldsymbol{v}_h^n \cdot \boldsymbol{n}|_e$ is calculated for all outlet surfaces, i.e. the top surface (Fig. 5b). The results of case 1, case 2, and case 3 are presented in Figs. 7a-c, respectively. The primary axis shows RF values. The secondary axis illustrates the difference between CG and EG solutions and the difference between DG and EG solutions, which are called RF difference hereafter. To be consistent, we define the difference, the magnitude of maximum difference, calculation as follows:

$$\text{difference} (\varphi) = \max \left(\text{abs} \left(\frac{\varphi \, (\text{EG}) - \varphi \, (\text{CG/DG})}{\varphi \, (\text{EG})} \right) \right), \tag{60}$$

where φ is any quantities of interest (e.g., RF difference), $\varphi(\text{EG})$ is the quantity of interest approximating using EG method, and $\varphi(\text{CG/DG})$ is the quantity of interest approximating using CG or DG method. For example, RF difference is calculated as $\max \left(\text{abs} \left(\frac{\text{RF(EG)} - \text{RF(CG/DG)}}{\text{RF(EG)}} \right) \right)$. The following observations can be made from these figures:

Table 1
Structured heterogeneity example: degrees of freedom (DOF) comparison
among CG, EG, and DG methods.

	CG	EG	DG
Displacement	17,258	17,258	17,258
Pressure	2,198	6,432	12,702
Total	19,456	23,690	29,960

1. Even though the results between pressure-independent and -dependent k_m are different, and the difference grows when the value of K is decreased (i.e., more deformable), the differences are still small (maximum difference between pressure-independent and -dependent k_m is approximately 3%.).

2. The maximum difference of RF results between the CG and EG methods is approximately 1.4 %. Besides, the maximum difference of RF results between the DG and EG methods is insignificant (the maximum difference is 0.08%).

3. When the reservoir has a lower K value, i.e. more deformable, RF becomes higher because of the compaction effect. Figs. 7a-c illustrate that the final RF is increased from 2.5×10^{-3} to 1.2×10^{-2} when K is reduced from 8 GPa to 1 GPa. This observation holds even though the k_m reduction is larger in softer media. Therefore, the compaction effect may dominate over k_m reduction effect on reservoir productivity.

Results of $\bar{\kappa}$ and $\bar{\varepsilon}_v$ illustrated in Figs. 8a-b, respectively, supports the second statement. As observed from these figures, the $\bar{\kappa}$ and $\bar{\varepsilon}_v$ among CG, EG, and DG methods are not significantly different. Therefore, the flux approximation, RF, among these methods are almost the same.

Moreover, the results of the number of iterations (Algorithm 2 - Picard iteration) among all discretizations and cases are presented in Fig. 8c. There is not much difference among CG, EG, and DG methods as the number of iterations in all cases range between three and seven. The softer media, however, needs more iteration than the tougher one, i.e. the number of iterations is larger in the material that has lower K value than that of the higher K value material. This is because the κ alteration is more drastic in the softer material than the tougher material; therefore, solutions take more iterations to converge.

The number of DOF among each method is shown in Table 1. Note that all methods use the same CG vector function space for the displacement; hence, the DOF is the same for the displacement domain. For the pressure space, as expected, the CG method has the least DOF, i.e. approximately three times less than that of the EG method. The DG method, on the other hand, has the most DOF, i.e. about two times more than that of the EG method.

3.5. 2D flow in a deformable media with random porosity and permeability

The 2D heterogeneous example presented in Fig. 9 targets to evaluate the local mass conservative property ($\max(r_{mass})$), flux approximation (RF), $\bar{\kappa}$ and $\bar{\varepsilon}_v$ alteration, number of iterations, and DOF among CG, EG, and DG methods. Its geometry, input parameters, and boundary conditions are adapted from the previous study [48,14]. Most of the input parameters have the same values as the structured heterogeneity example. K values are applied similarly to the structured heterogeneity case, i.e. $K = 8$, 2, and 1 GPa for case 1, 2, and 3, respectively. On the other hand, ϕ and κ_0 fields are populated based on normal and log-normal distributions as illustrated in Fig. 9a-b, respectively. The heterogeneous parameters are provided based on the following specifications, $\bar{\phi} = 0.2$, $\text{var}(\phi) = 0.01$, $\phi_{min} = 0.001$, $\phi_{max} = 0.4$, $\bar{\kappa}_0 = 1.2 \times 10^8 I$ s, $\text{var}(\kappa_0) = 1.4 \times 10^{-16} I$ s, $\kappa_{0\,min} = 1.2 \times 10^{-13} I$ s, and $\kappa_{0\,max} = 1.2 \times 10^{-6} I$ s. $\overline{(.)}$ is an arithmetic average, var(.) is the variance, $(.)_{min}$ and $(.)_{max}$ are the minimum and maximum values, respectively. $\bar{\phi}$, ϕ_{min}, ϕ_{max}, $\bar{\kappa}_0$, $\kappa_{0\,min}$, and $\kappa_{0\,max}$ are selected based on the reported values of shales and sandstones from Jaeger et al. [44]. Note that $\bar{\kappa}_0$, $\kappa_{0\,min}$, and $\kappa_{0\,max}$ reflect the k_{m0}, $k_{m0\,min}$, and $k_{m0\,max}$ values of $1.2 \times 10^{-14} I$, $1.2 \times 10^{-19} I$, $1.2 \times 10^{-12} I$, respectively. In the nature, the ϕ and κ_0 fields are normally correlated [52]. In this study, however, these two heterogeneous fields are populated independently, i.e., there is no correlation between ϕ and κ_0. Since the completely random fields are the most challenge setting, i.e., the sharp discontinuity could appear in many places, and the permeability alteration resulted from the solid deformation [39,40] could result in a negative value. Therefore, if the proposed approximation passes this case, it would be able to handle a correlated one.

Figs. 10a-b present $\max(r_{mass})$ results for pressure-independent and -dependent k_m models, respectively. The CG method illustrates a lack of local mass conservation property, and $\max(r_{mass})$ value is significantly high at the beginning (1×10^{-3} kg). On the contrary, the EG and DG methods illustrate preservation of local mass, and their $\max(r_{mass})$ values are almost equivalent to zero. Moreover, the case with a lower K value has higher $\max(r_{mass})$ value because the change in u is more severe in the softer material.

The RF results of case 1, $K = 8$ GPa, case 2, $K = 2$ GPa, and case 3, $K = 1$ GPa are presented in Figs. 11a-c, respectively. The following observations can be drawn from these figures:

1. The pressure-dependent k_m model delivers fluid slower than the pressure-independent k_m model. Since the fluid is withdrawn from the reservoir, k_m is reduced through time. Therefore, the pressure-dependent k_m model has less ability
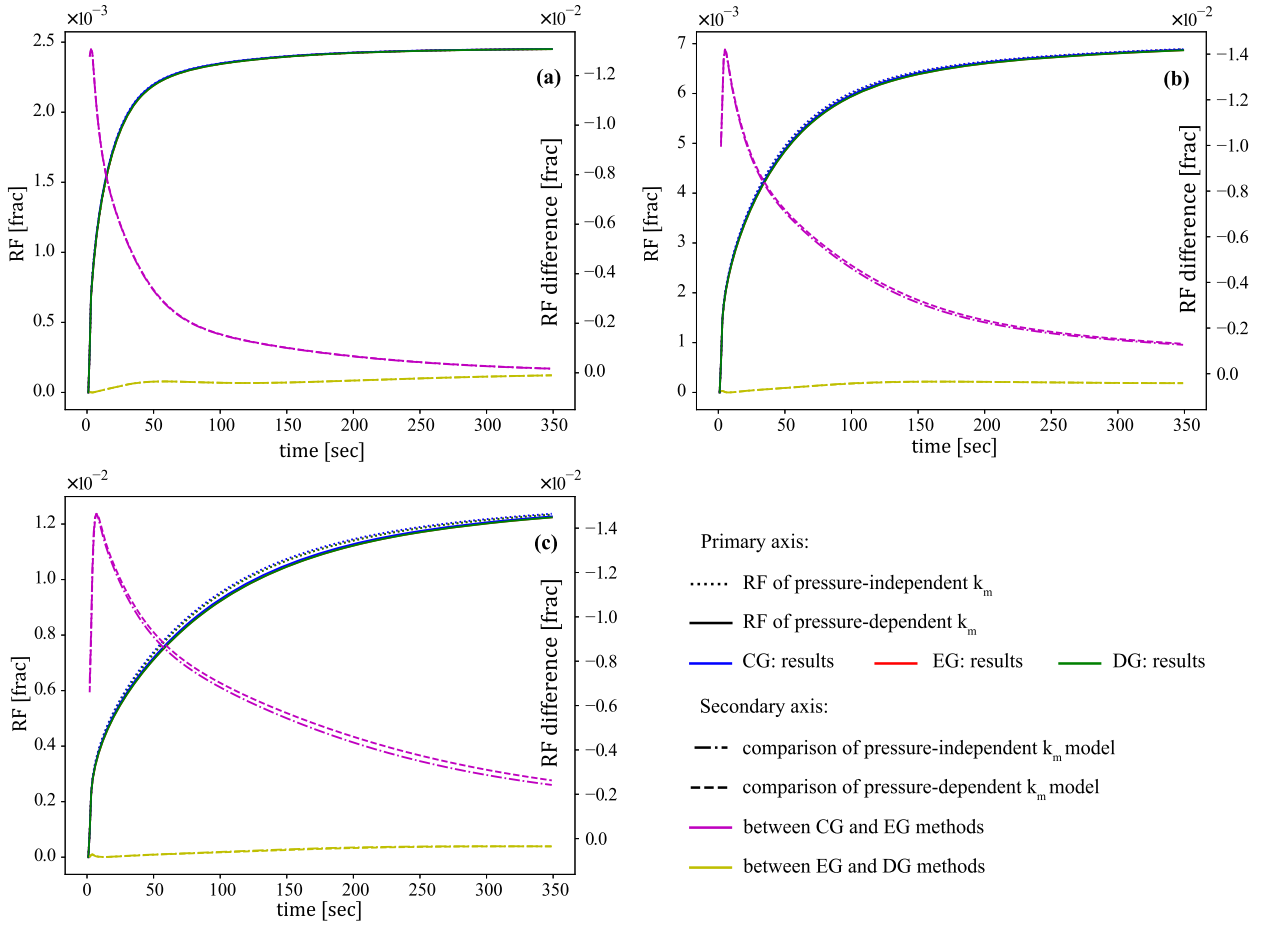
Fig. 7. Structured heterogeneity example: the comparison of RF (primary axis) and the difference of RF (second axis) among the CG (blue line), EG (red line), and DG (green line) methods using pressure-dependent (solid) and pressure-independent (dotted) k_m models for (**a**) case 1, $K = 8$ GPa, (**b**) case 2, $K = 2$ GPa, and (**c**) case 3, $K = 1$ GPa. Note that CG, EG, and DG results overlay on top of each other.

to produce reservoir fluid as the effective stress increases. Moreover, the difference between the two models becomes larger, when the reservoir is more deformable, i.e. K is lower.

2. The difference of RF results between the CG and EG methods is from 14 % to 20 %. The maximum difference of RF results between the DG and EG methods, on the other hand, is much lower. The maximum difference is approximately 0.5 % when $K = 1$ GPa.

3. As also observed from the structured heterogeneity example, the media that has a lower K value, i.e. more deformable, has a higher RF. Figs. 11a-c present that the final RF is increased from 3.5×10^{-3} to 2.0×10^{-1} when K is reduced from 8 GPa to 1 GPa.

The disagreement of RF between the CG and EG/DG methods can be explained using the results of $\bar{\kappa}$ and $\bar{\varepsilon}_v$ presented in Figs. 12a-b, respectively. The $\bar{\kappa}$ and $\bar{\varepsilon}_v$ values of the CG method are significantly different from the other two methods at the beginning, but they converge to the same values at the later time. On the contrary, the $\bar{\kappa}$ and $\bar{\varepsilon}_v$ results of the EG and DG methods are approximately similar.

Besides, we compare the number of iterations of Algorithm 2. We note that the number of iterations results of the EG and DG methods are almost the same, but they are much higher than that of the CG method (see Fig. 12c). This situation is even more pronounced for the softer material. Comparing with the structured heterogeneity example, the number of iterations of this example is also much higher.

Next, the number of DOF among each method is presented in Table 2. As also observed from the structured heterogeneity example, the DOF of EG method is approximately three times higher than that of the CG method, but it is almost two times lower than that of the DG method. Note that CG, EG, and DG methods have the same DOF for displacement space because it is solved on the same CG vector function space with a quadratic approximation.
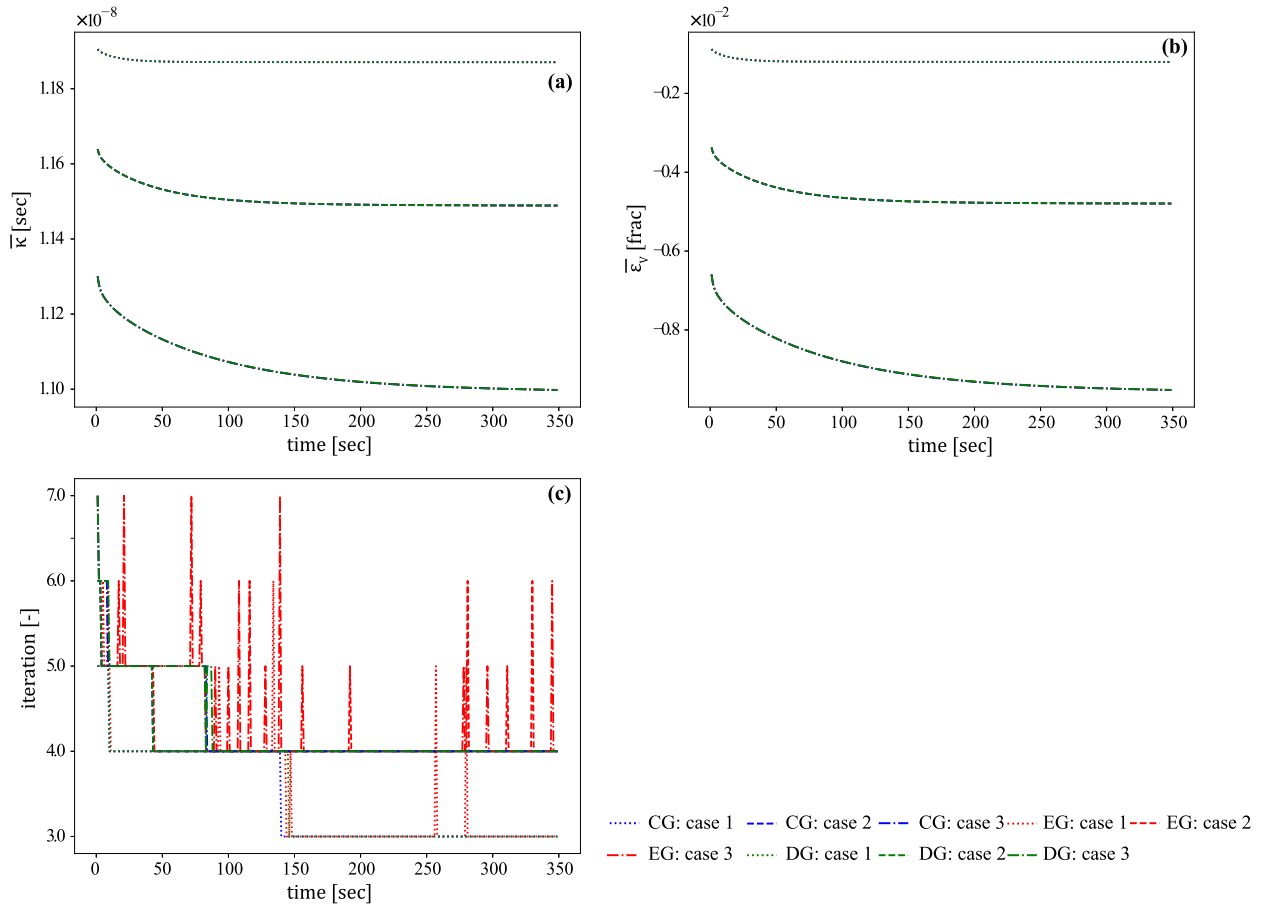
Fig. 8. Structured heterogeneity example: the comparison of (**a**) $\bar{\kappa}$, (**b**) $\bar{\varepsilon}_v$, and (**c**) number of iterations values among the CG (blue line), EG (red line), and DG (green line) methods using pressure-dependent $\boldsymbol{k_m}$ model for case 1, $K = 8$ GPa, case 2, $K = 2$ GPa, and case 3, $K = 1$ GPa. Note that CG, EG, and DG results are overlapped on top of each other in a and b.
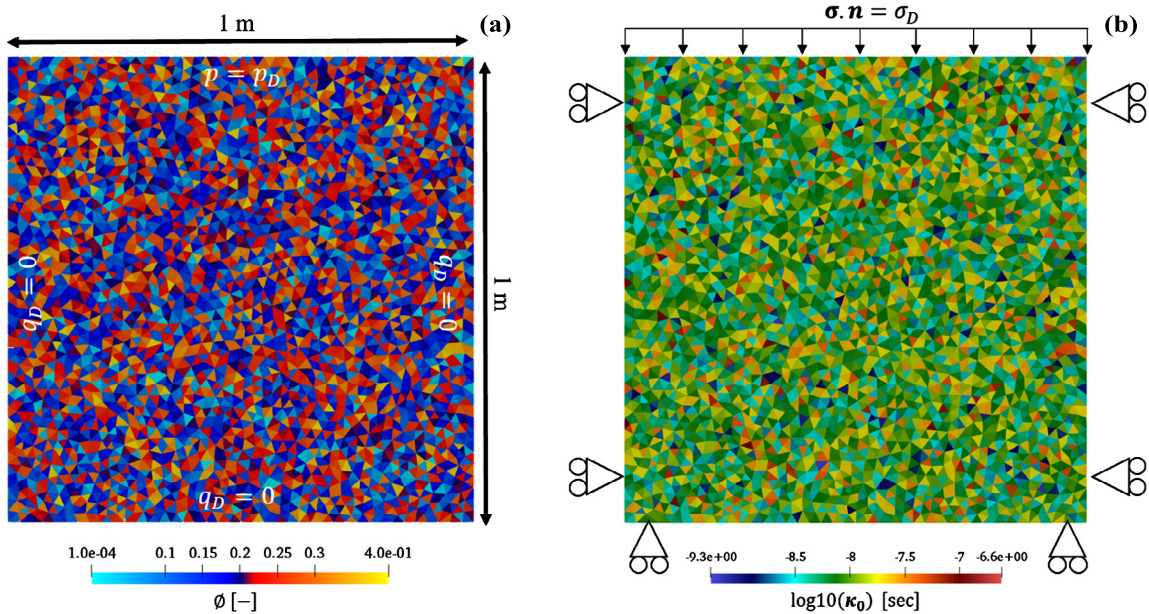

Fig. 9. 2D heterogeneous example: (**a**) ϕ field and mass boundary conditions and (**b**) $\log 10(\boldsymbol{\kappa_0})$ field in x-direction and force boundary conditions.
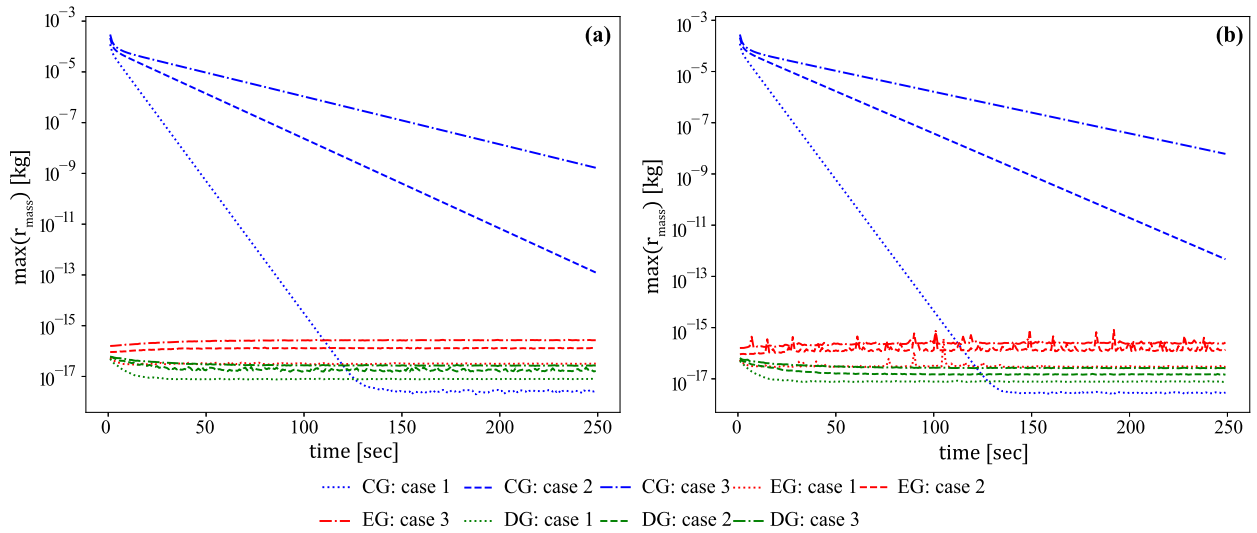
Fig. 10. 2D heterogeneous example: the comparison of max(r_{mass}) values among the CG (blue line), EG (red line), and DG (green line) methods using (**a**) pressure-independent $\boldsymbol{k_m}$ and (**b**) pressure-dependent $\boldsymbol{k_m}$ models for case 1, $K = 8$ GPa, case 2, $K = 2$ GPa, and case 3, $K = 1$ GPa.

Table 2
Degrees of freedom (DOF) comparison among CG, EG, and DG methods for 2D heterogeneous example.

	CG	EG	DG
Displacement	17,370	17,370	17,370
Pressure	2,212	6,474	12,786
Total	19,582	23,844	30,156

3.6. 3D flow in a deformable media with random porosity and permeability

This example aims to extend the investigation from the 2-Dimensional domain, which is presented in the previous section, to a 3-Dimensional domain. The geometry and boundary conditions are presented in Fig. 13a. The input parameters that are applied to the structured heterogeneity and 2D heterogeneous examples are also used in this example. $\boldsymbol{\sigma_D}$ for this case is applied as follows: $\boldsymbol{\sigma_{Dx}} = [15, 0, 0]$, $\boldsymbol{\sigma_{Dy}} = [0, 15, 0]$, and $\boldsymbol{\sigma_{Dz}} = [0, 0, 20]$ MPa. ϕ and $\boldsymbol{\kappa_0}$ fields are populated with the same specifications as 2D heterogeneous example as illustrated in 13c-d, respectively. The illustration of $\boldsymbol{\kappa_0}$ field is presented in Fig. 13b.

Figs. 14a-b present max(r_{mass}) results for pressure-independent and -dependent $\boldsymbol{k_m}$ models, respectively. While the local mass conservation property of the EG and DG methods is well preserved, it is revoked when the CG method is used, i.e. max(r_{mass}) value is as large as 1×10^{-4} kg at the beginning. As expected, the cases with a lower K value have higher max(r_{mass}) value.

The RF results of case 1, $K = 8$ GPa, case 2, $K = 2$ GPa, and case 3, $K = 1$ GPa are presented in Figs. 15a-c, respectively. The observations that are drawn from the 2D heterogeneous example are still valid. The difference of RF results between the CG and EG methods, however, is now more pronounced, and it ranges from 30% to 40%. On the other hand, the maximum difference between the DG and EG methods is still as small as approximately 3%.

The results of $\bar{\kappa}$ and $\bar{\varepsilon}_v$ are presented in Figs. 16a-b, respectively. As also observed from 2D heterogeneous example, the $\bar{\kappa}$ and $\bar{\varepsilon}_v$ values of the CG method are significantly different from the EG and DG methods, in the beginning, then they converge to the same values at the later time since the system reaches an equilibrium state. The results of the number of iterations comply with the 2D heterogeneous example, as shown in Fig. 12c.

The number of DOF among each method is presented in Table 3. Even though the CG, EG, and DG methods have the same DOF for the displacement space, the DOF of pressure space among these three methods are significantly different. The EG method is five times more expensive than the CG method, but it is three times less costly than the DG method. We note that the difference between the number of DOF is more significant in the 3D case than the 2D case.

3.7. North Sea field example

The last example is based on the configuration of a layered hydrocarbon field in the North Sea [53,54]. The reservoir is exploited by several long (2-5 kilometer) parallel horizontal wells. The wells are hydraulically fractured [e.g.12] along the length of the wells. Fig. 17 illustrates the model geometry, including the wells in the middle of each fracture, together with the material properties of each layer and boundary conditions. Note that no-flow boundary condition, i.e. $q_D = 0$, is applied
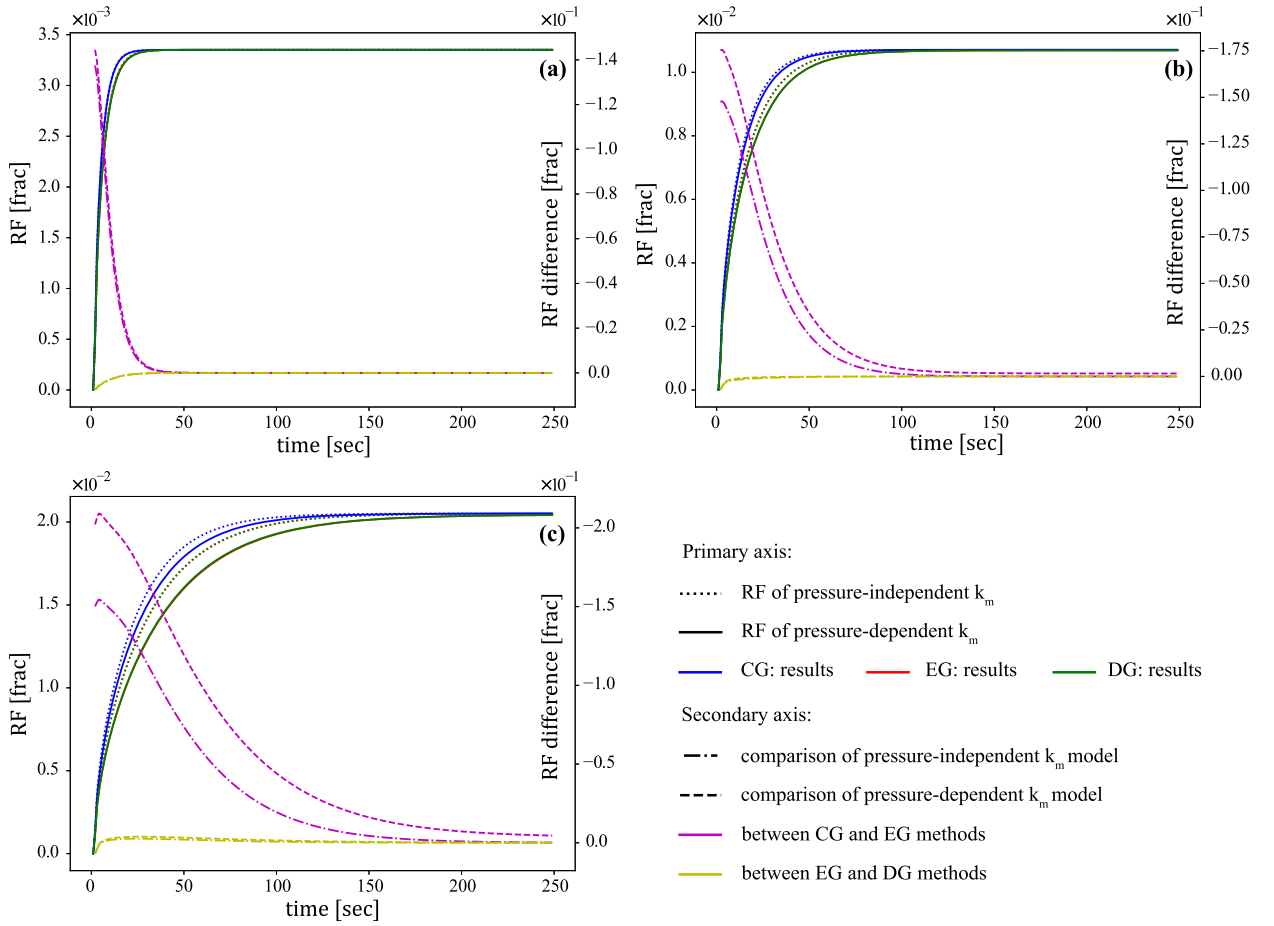
Fig. 11. 2D heterogeneous example: the comparison of RF (primary axis) and the difference of RF (second axis) among the CG (blue line), EG (red line), and DG (green line) methods using pressure-dependent (solid) and pressure-independent (dotted) $\boldsymbol{k_m}$ models for (**a**) case 1, $K = 8$ GPa, (**b**) case 2, $K = 2$ GPa, and (**c**) case 3, $K = 1$ GPa. Note that EG and DG results overlay on each other.

Table 3
Degrees of freedom (DOF) comparison among CG, EG, and DG methods for
3D heterogeneous example.

	CG	EG	DG
Displacement	20,901	20,901	20,901
Pressure	1,068	5,163	16,380
Total	21,969	26,064	37,281

to all external faces. In this example, we set $p_0 = 1.0 \times 10^7$ Pa, $\sigma_D = 30.0 \times 10^6$ Pa, $\rho = 897.0$ kg/m^3, $c_f = 1.0 \times 10^{-9}$ Pa^{-1}, $\Delta t^n = 2.628 \times 10^6$ sec, $\tau = 9.461 \times 10^7$ sec. We assume that the fracture permeability is much higher than the matrix permeability [53,54]; hence, the constant pressure is applied to the fracture, i.e., $p_D = 1.0 \times 10^5$ Pa. We would like to emphasize that to mimic a field model, the robustness of the numerical method to different mesh sizes is one of the vital desire characteristics.

From previous sections, one could see that the effect of permeability alteration resulted from the solid deformation is non-negligible. Therefore, it should be taken into account, and the EG and DG methods could provide comparable results. Hence, in this section, we focus on comparisons between the CG and EG methods using the pressure-dependent $\boldsymbol{k_m}$ model to illustrate that the results of EG method are less sensitive to the mesh size compared to the CG method. The CG and EG results of RF and two different meshes are presented in Fig. 18. Besides, the examples of pressure solution of the coarse mesh (number of elements $\approx 6,000$) are presented in Fig. 19.

The RF results (see Fig. 18a) illustrate that the EG method provides consistent results for both fine and coarse mesh sizes (number of elements $\approx 6,000$ and $85,000$). The EG at the coarse mesh solution already is as precise as the CG on the fine mesh (Fig. 18). While the CG results are mesh dependent and varied significantly among different mesh sizes, the results of
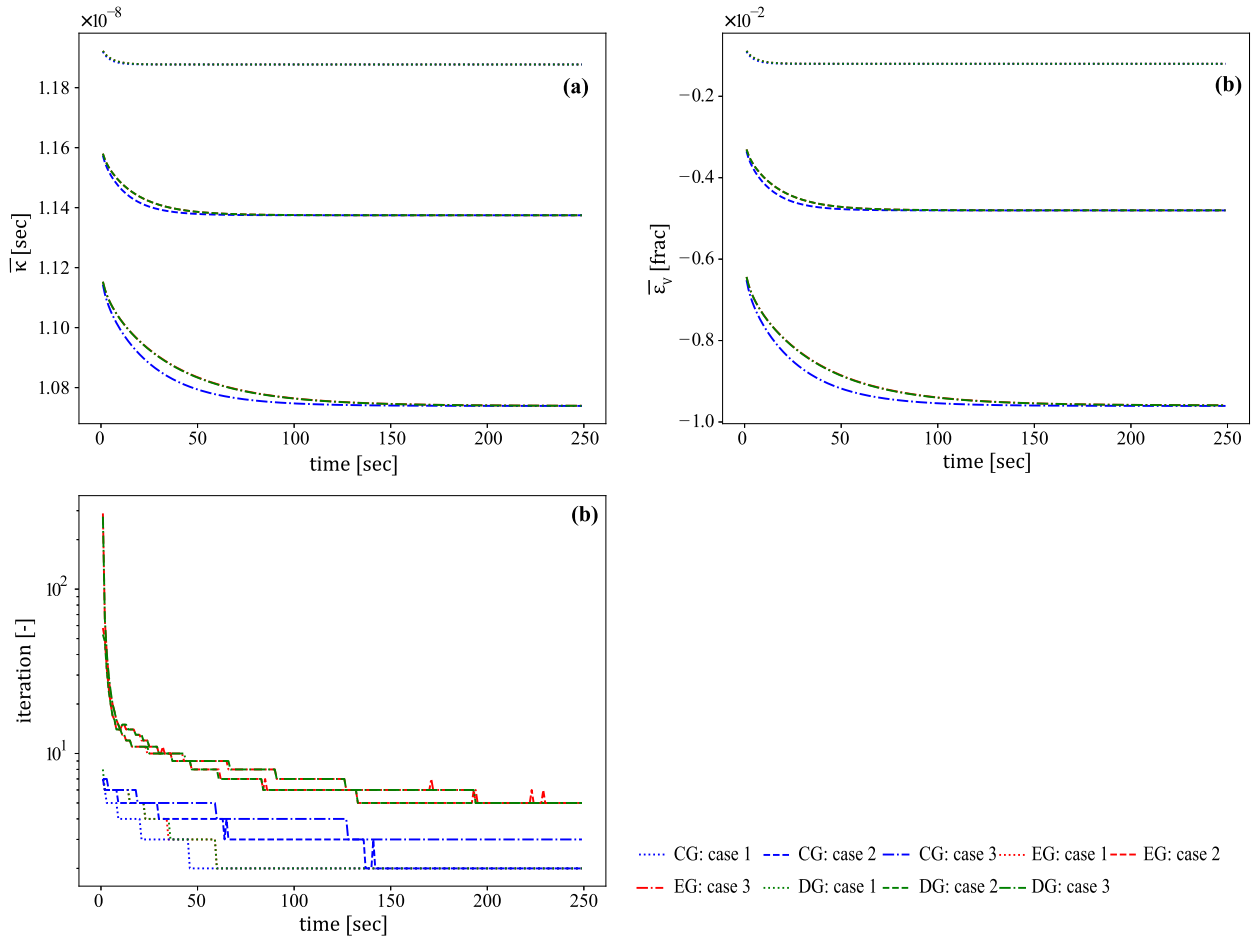
Fig. 12. 2D heterogeneous example: the comparison of (**a**) $\bar{\kappa}$, (**b**) $\bar{\varepsilon}_v$, and (**c**) number of iterations values among the CG (blue line), EG (red line), and DG (green line) methods using pressure-dependent $\boldsymbol{k_m}$ model for case 1, $K = 8$ GPa, case 2, $K = 2$ GPa, and case 3, $K = 1$ GPa. Note that EG and DG results are on top of each other for (**a**) and (**b**).

the EG method are similar regardless of the mesh size. The EG method allows using a relatively coarse mesh to maintain accuracy. This is an essential feature of the EG method that makes it more suitable for large scale simulations.

The DOF comparison of the model used in this example is presented in Table 4. The total DOF for the EG at the coarse mesh is about 11 times less than that for the CG at the fine mesh.

4. Conclusion

This study presents the discretization, block structure, and coupling strategy for solving the single-phase fluid flow in the poroelastic media with permeability alteration using the EG method. Moreover, we compare the positive and negative aspects of the CG (in its classical form), EG, and DG methods. Taking into consideration the limitations and assumptions made in this study, the following conclusions are drawn:

- The CG method suffers from the spurious oscillations in pressure, and volumetric strain at the interfaces, especially with significant permeability differences. The EG and DG methods, on the contrary, deliver smooth solutions since the EG and DG methods can handle the discontinuity of pressure across the interfaces, which also results in the local mass conservation.
- The CG, EG, and DG methods provide approximately the same flux approximation in the deformable media with the structured heterogeneity example. However, flow in the 2D deformable media with random porosity and permeability example illustrates that the CG method's flux approximation is different from the EG and DG method as high as 20%. This difference becomes more substantial in the 3D domain (the maximum difference is 40%) since there are more material interfaces in the 3D than the 2D domain, which leads to the increase of pressure and ε_v oscillation across interfaces. The EG and DG methods deliver approximately the same flux approximation in all examples (the maximum difference is less than 3%).
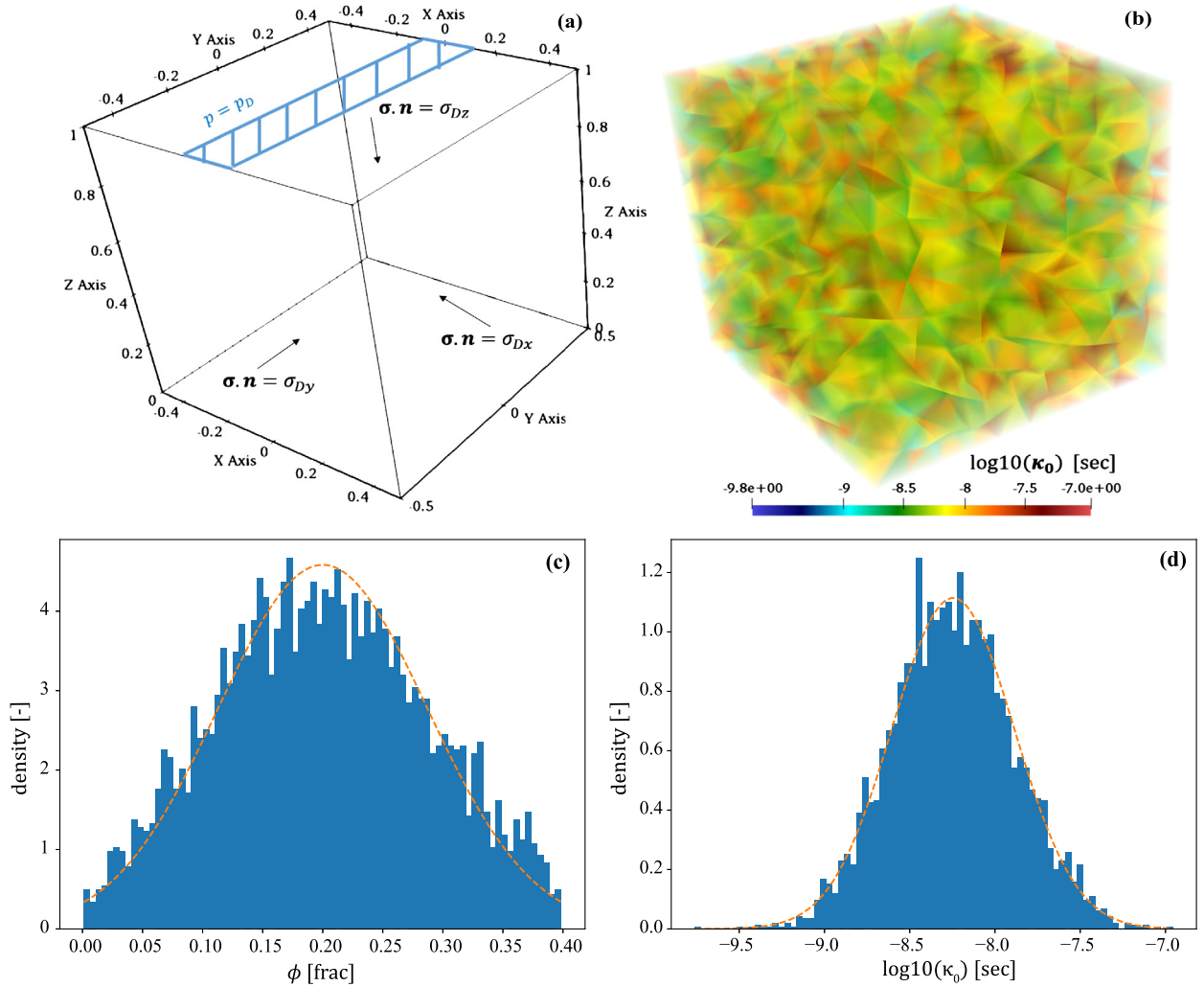
Fig. 13. 3D heterogeneous example: (**a**) geometry and boundary conditions (note that 1. besides the surface that is imposed by the constant pressure boundary, other surfaces are imposed by zero-flux boundary and 2. surfaces that are opposite of the stress boundaries are imposed with zero normal displacement), and (**b**) illustration of $\log 10(\kappa_0)$ field (**c**) ϕ histogram, and (**d**) $\log 10(\kappa_0)$ histogram.

Table 4
Degrees of freedom (DOF) comparison between CG and EG methods using different mesh size for North Sea field example.

	CG		EG	
Number of elements $\approx$	6,000	85,000	6,000	85,000
Displacement	24,486	331,062	24,486	331,062
Pressure	3,096	41,438	9,148	124,094
Total	27,582	372,500	33,634	455,156

- If the lack of local mass conservation is not the primary concern, the CG method could be used to model the poroelastic problem for models with little material discontinuity, e.g., the structured heterogeneity example because it requires the least computational resources. However, the CG method may not be suitable when the porous media are heterogeneous since it may overestimate the flux calculation.

- The EG and DG methods require higher computational resources than the CG method. The number of Picard iteration of EG and DG methods is approximately similar, but it is larger than that of the CG method. Thus, providing or developing the optimal solver for both EG and DG cases is necessary. The number of iterations is increased if K is lower, where the media is more deformable.

- The DOF of EG is approximately three and five times larger than that of the CG method for 2D and 3D domains, respectively. However, it is almost two and three times fewer than that of the DG method for 2D and 3D domains.
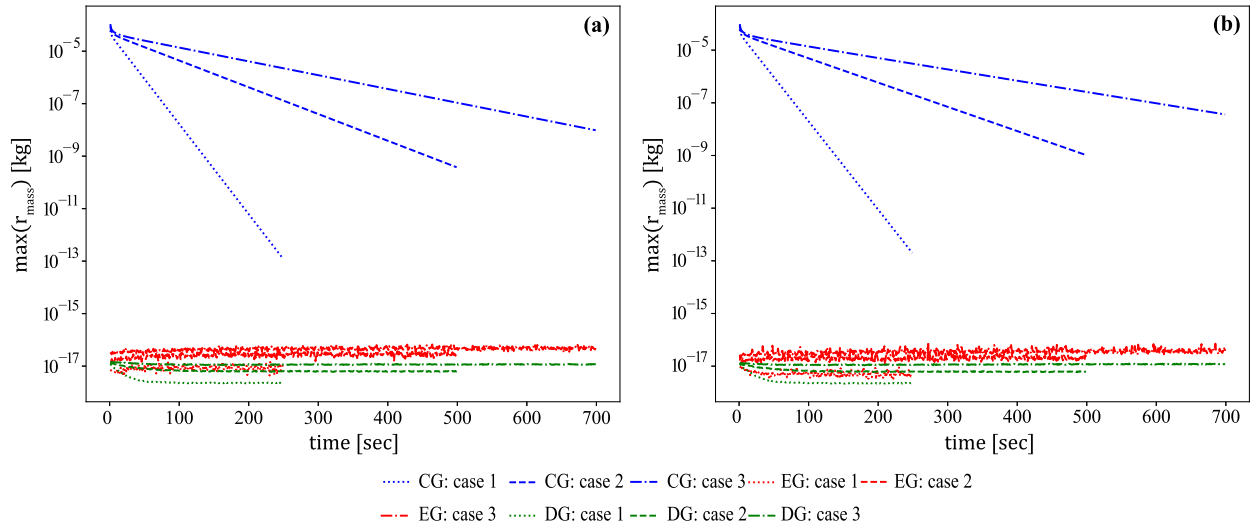
Fig. 14. 3D heterogeneous example: the comparison of max(r_{mass}) values among the CG (blue line), EG (red line), and DG (green line) methods using (**a**) pressure-independent $\boldsymbol{k_m}$ and (**b**) pressure-dependent $\boldsymbol{k_m}$ models for case 1, $K = 8$ GPa, case 2, $K = 2$ GPa, and case 3, $K = 1$ GPa.
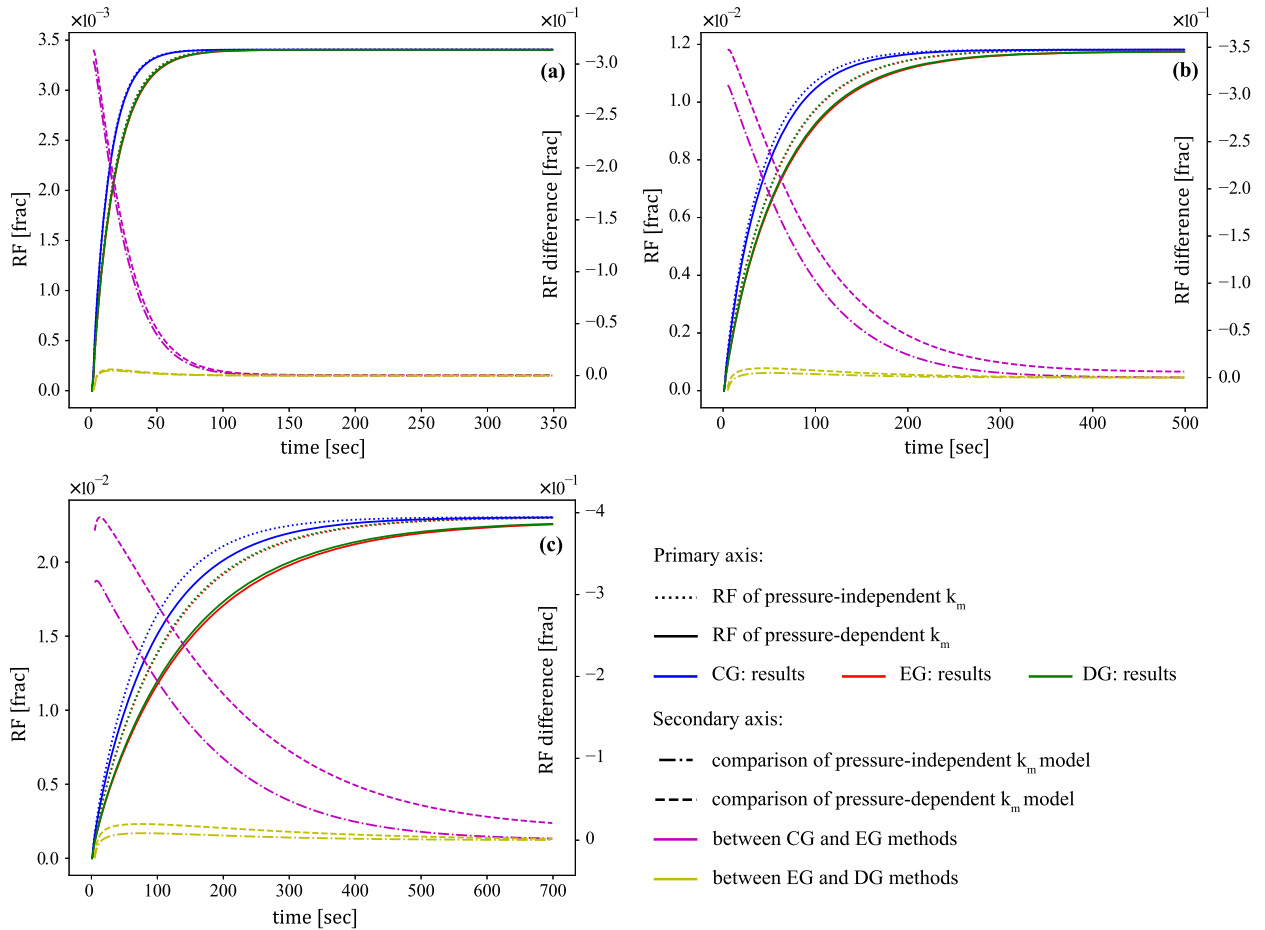

Fig. 15. 3D heterogeneous example: the comparison of RF (primary axis) and the difference of RF (second axis) among the CG (blue line), EG (red line), and DG (green line) methods using pressure-dependent (solid) and pressure-independent (dotted) $\boldsymbol{k_m}$ models for (**a**) case 1, $K = 8$ GPa, (**b**) case 2, $K = 2$ GPa, and (**c**) case 3, $K = 1$ GPa.
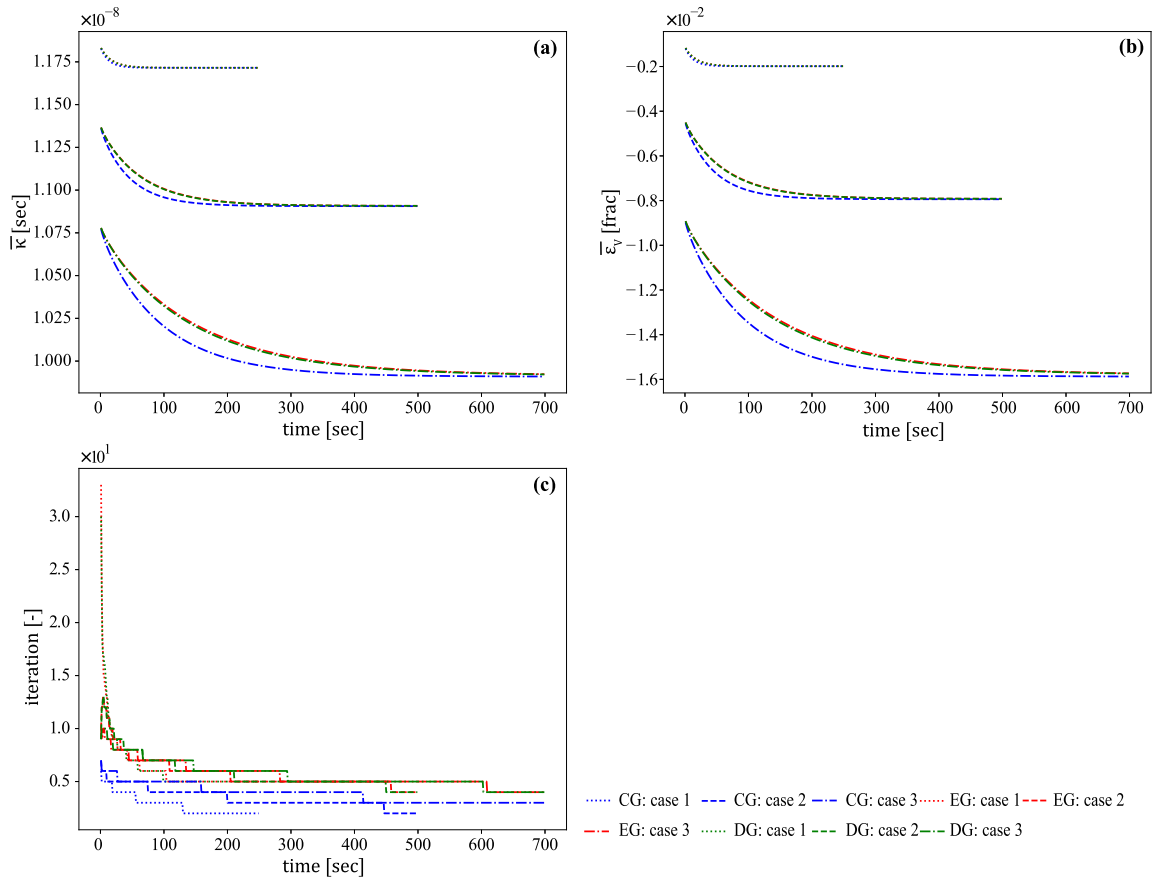
Fig. 16. 3D heterogeneous example: the comparison of (**a**) $\bar{\kappa}$, (**b**) $\bar{\varepsilon}_v$, and (**c**) number of iterations values among the CG (blue line), EG (red line), and DG (green line) methods using pressure-dependent $\boldsymbol{k_m}$ model for case 1, $K = 8$ GPa, case 2, $K = 2$ GPa, and case 3, $K = 1$ GPa.
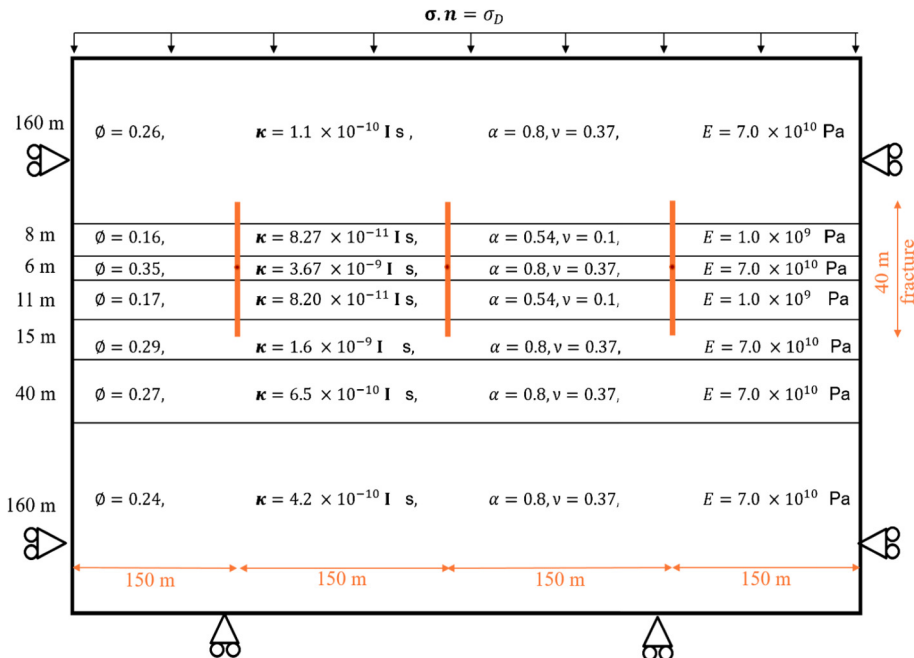

Fig. 17. North Sea field example: the geometry and boundaries conditions. Note that the fractures are shown in orange, and the horizontal wells (red dots) are located in the middle of each fracture. Note that the drawing is not scaled to the actual dimension.
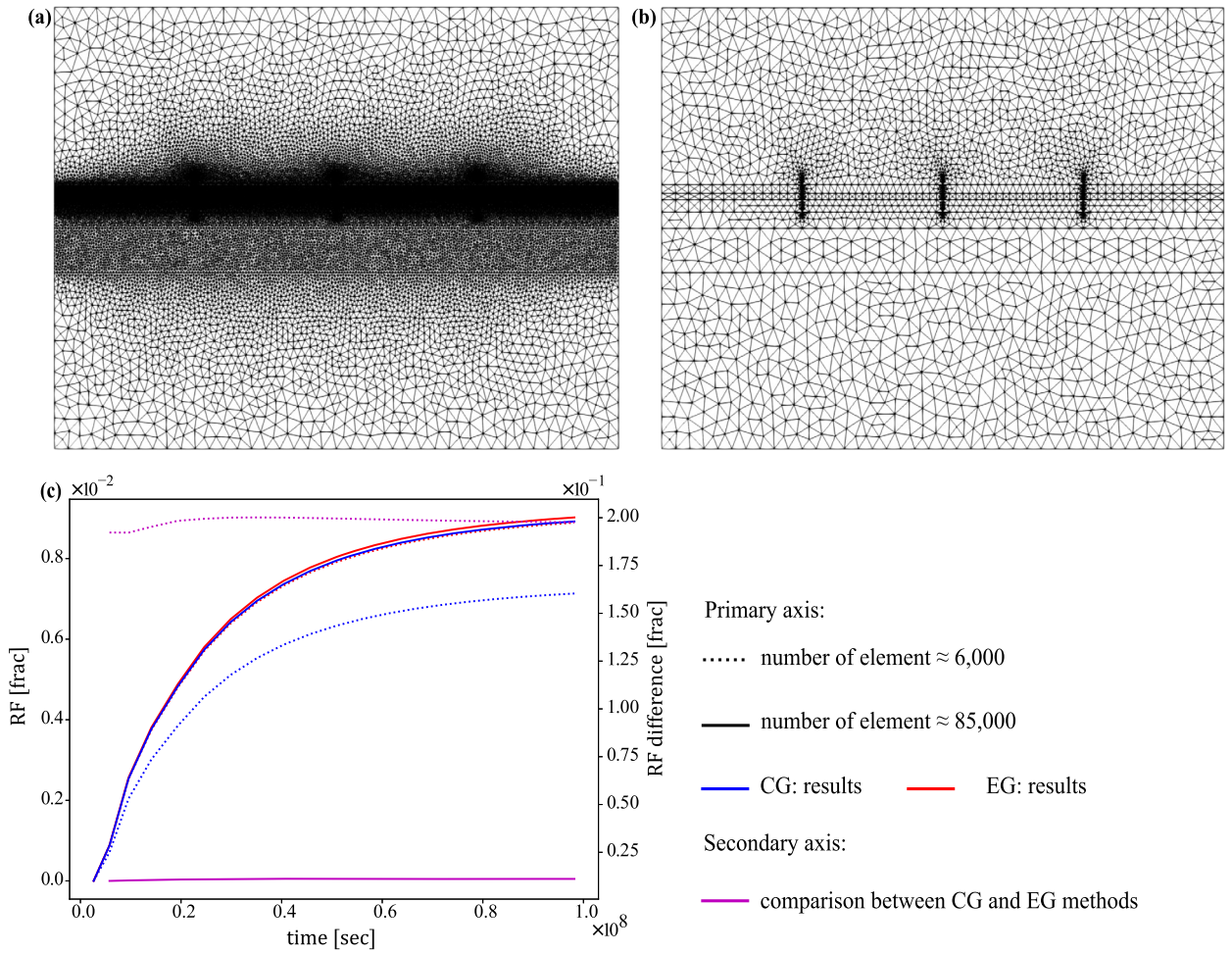
Fig. 18. North Sea field example: the comparison of (**a**) mesh with number of element $\approx 85,000$, (**b**) mesh with number of element $\approx 6,000$, and (**c**) RF (primary axis) and (**a**) the difference of RF (second axis) between the CG (blue line) and EG (red line) methods. Note that for (**c**) the results of EG for all cases and CG with number of elements $\approx 85,000$ are almost overlapped.

- The permeability reduction resulted in the solid deformation is significant, and it should be considered. Moreover, the compaction effect may overcome k_m reduction effect on the reservoir deliverability, i.e., even though the softer reservoir has a higher k_m reduction, its ultimate RF is higher than that of the more hardened media.
- Utilizing the EG method in the field setting illustrates its capability to produce robust results, i.e., the mesh size effect is minimal. This experiment shows the possibility of extending this method to real-field applications.

Finally, in terms of future work, fast solvers of the EG method focusing on multiphysics problems should be investigated and developed. Moreover, an adaptive enrichment, i.e., the piecewise-constant functions only added to the elements where the sharp material discontinuities are observed, can be developed.

CRediT authorship contribution statement

T. Kadeethum: Conceptualization, Formal analysis, Software, Validation, Writing – original draft. **H.M. Nick:** Conceptualization, Funding acquisition, Supervision, Writing – review & editing. **S. Lee:** Conceptualization, Formal analysis, Supervision, Validation, Writing – review & editing. **F. Ballarin:** Conceptualization, Formal analysis, Software, Supervision, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.
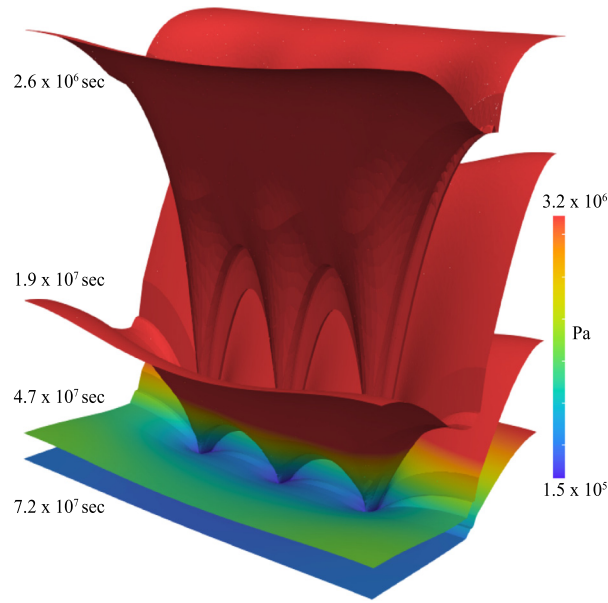
Fig. 19. North Sea field example: the examples of pressure solution using mesh with number of element $\approx 6,000$ with different time.

Computer code availability

The scripts used to produce these results are available at this Git repository. The main dependencies are Numpy ($\geq$ 1.16.5), FEniCS ($\geq$ 2018.1.0) with PETSc ($\geq$ 3.10.5) and petsc4py ($\geq$ 3.10), and multiphenics ($\geq$ 0.2.0).

Acknowledgements

References

[1] H. Nick, A. Raoof, F. Centler, M. Thullner, P. Regnier, Reactive dispersive contaminant transport in coastal aquifers: numerical simulation of a reactive henry problem, J. Contam. Hydrol. 145 (2013) 90–104.

[2] K. Bisdom, G. Bertotti, H. Nick, The impact of in-situ stress and outcrop-based fracture geometry on hydraulic aperture and upscaled permeability in fractured reservoirs, Tectonophysics 690 (2016) 63–75.

[3] S. Lee, M. Wheeler, T. Wick, Pressure and fluid-driven fracture propagation in porous media using an adaptive finite element phase field model, Comput. Methods Appl. Mech. Eng. 305 (2016) 111–132.

[4] R. Juanes, B. Jha, B. Hager, J. Shaw, A. Plesch, L. Astiz, J. Dieterich, C. Frohlich, Were the May 2012 Emilia-Romagna earthquakes induced? A coupled flow-geomechanics modeling assessment, Geophys. Res. Lett. 43 (13) (2016) 6891–6897.

[5] V. Vinje, J. Brucker, M. Rognes, K. Mardal, V. Haughton, Fluid dynamics in syringomyelia cavities: effects of heart rate, CSF velocity, CSF velocity waveform and craniovertebral decompression, Neuroradiol. J. (2018) 1971400918795482.

[6] T. Kadeethum, S. Salimzadeh, H. Nick, Well productivity evaluation in deformable single-fracture media, Geothermics 87 (2020).

[7] T. Kadeethum, H. Nick, S. Lee, F. Ballarin, Flow in porous media with low dimensional fractures by employing enriched Galerkin method, Adv. Water Resour. 142 (2020) 103620.

[8] M. Biot, General theory of three-dimensional consolidation, J. Appl. Phys. 12 (2) (1941) 155–164.

[9] M. Biot, D. Willis, The elastic coefficients of the theory of consolidation, J. Appl. Mech. 15 (1957) 594–601.

[10] J. Haga, H. Osnes, H. Langtangen, On the causes of pressure oscillations in low permeable and low compressible porous media, Int. J. Numer. Anal. Methods Geomech. 36 (12) (2012) 1507–1522.

[11] H. Vik, S. Salimzadeh, H. Nick, Heat recovery from multiple-fracture enhanced geothermal systems: the effect of thermoelastic fracture interactions, Renew. Energy (2018).

[12] S. Salimzadeh, E. Hagerup, T. Kadeethum, H. Nick, The effect of stress distribution on the shape and direction of hydraulic fractures in layered media, Eng. Fract. Mech. 215 (2019) 151–163.

[13] T. Kadeethum, S. Salimzadeh, H. Nick, An investigation of hydromechanical effect on well productivity in fractured porous media using full factorial experimental design, J. Pet. Sci. Eng. 181 (2019) 106233.

[14] T. Kadeethum, S. Lee, H. Nick, Finite element solvers for Biot's poroelasticity equations in porous media, Math. Geosci. (2020) 1–39.

[15] P. Vermeer, A. Verruijt, An accuracy condition for consolidation by finite elements, Int. J. Numer. Anal. Methods Geomech. 5 (1) (1981) 1–14.

[16] M. Murad, A. Loula, On stability and convergence of finite element approximations of Biot's consolidation problem, Int. J. Numer. Methods Biomed. Eng. 37 (4) (1994) 645–667.

[17] G. Scovazzi, M. Wheeler, A. Mikelic, S. Lee, Analytical and variational numerical methods for unstable miscible displacement flows in porous media, J. Comput. Phys. 335 (2017) 444–496.

[18] Q. Deng, V. Ginting, B. McCaskill, P. Torsu, A locally conservative stabilized continuous Galerkin finite element method for two-phase flow in poroelastic subsurfaces, J. Comput. Phys. 347 (2017) 78–98.

[19] P. Phillips, M. Wheeler, A coupling of mixed and continuous Galerkin finite element methods for poroelasticity II: the discrete-in-time case, Comput. Geosci. 11 (2) (2007) 145–158.

[20] M. Wheeler, G. Xue, I. Yotov, Coupling multipoint flux mixed finite element methods with continuous Galerkin methods for poroelasticity, Comput. Geosci. 18 (1) (2014) 57–75.

[21] B. Flemisch, M. Darcis, K. Erbertseder, B. Faigle, A. Lauser, K. Mosthaf, S. Muthing, P. Nuske, A. Tatomir, M. Wolff, et al., DuMux: DUNE for multi-phase, component, scale, physics, flow and transport in porous media, Adv. Water Resour. 34 (9) (2011) 1102–1112.

[22] H. Nick, S. Matthai, Comparison of three FE-FV numerical schemes for single- and two-phase flow simulation of fractured porous media, Transp. Porous Media 90 (2) (2011) 421–444.

[23] P. Phillips, M. Wheeler, A coupling of mixed and discontinuous Galerkin finite-element methods for poroelasticity, Comput. Geosci. 12 (4) (2008) 417–435.

[24] R. Liu, M. Wheeler, C. Dawson, R. Dean, Modeling of convection-dominated thermoporomechanics problems using incomplete interior penalty Galerkin method, Comput. Methods Appl. Mech. Eng. 198 (9–12) (2009) 912–919.

[25] G. Harper, J. Liu, S. Tavener, B. Zheng, Lowest-order weak Galerkin finite element methods for linear elasticity on rectangular and brick meshes, J. Sci. Comput. (2018) 1–25.

[26] J. Liu, S. Tavener, Z. Wang, Lowest-order weak Galerkin finite element method for Darcy flow on convex polygonal meshes, SIAM J. Sci. Comput. 40 (5) (2018) B1229–B1252.

[27] T. Kadeethum, T. Jørgensen, H. Nick, Physics-informed neural networks for solving inverse problems of nonlinear Biot's equations: batch training, in: 54th US Rock Mechanics/Geomechanics Symposium, American Rock Mechanics Association, Golden, CO, USA, 2020.

[28] T. Kadeethum, T. Jørgensen, H. Nick, Physics-informed neural networks for solving nonlinear diffusivity and Biot's equations, PLoS ONE 15 (5) (2020) e0232683.

[29] H. Nick, S. Matthai, A hybrid finite-element finite-volume method with embedded discontinuities for solute transport in heterogeneous media, Vadose Zone J. 10 (1) (2011) 299–312.

[30] P. Salinas, D. Pavlidis, Z. Xie, H. Osman, C. Pain, M. Jackson, A discontinuous control volume finite element method for multi-phase flow in heterogeneous porous media, J. Comput. Phys. 352 (2018) 602–614.

[31] J. Choo, S. Lee, Enriched Galerkin finite elements for coupled poromechanics with local mass conservation, Comput. Methods Appl. Mech. Eng. 341 (2018) 311–332.

[32] S. Sun, J. Liu, A locally conservative finite element method based on piecewise constant enrichment of the continuous Galerkin method, SIAM J. Sci. Comput. 31 (4) (2009) 2528–2548.

[33] S. Lee, Y. Lee, M. Wheeler, A locally conservative enriched Galerkin approximation and efficient solver for elliptic and parabolic problems, SIAM J. Sci. Comput. 38 (3) (2016) A1404–A1429.

[34] J. Choo, Large deformation poromechanics with local mass conservation: an enriched Galerkin finite element framework, Int. J. Numer. Methods Biomed. Eng. 116 (1) (2018) 66–90.

[35] J. Choo, Stabilized mixed continuous/enriched Galerkin formulations for locally mass conservative poromechanics, Comput. Methods Appl. Mech. Eng. 357 (2019) 112568.

[36] S. Lee, M. Wheeler, Adaptive enriched Galerkin methods for miscible displacement problems with entropy residual stabilization, J. Comput. Phys. 331 (2017) 19–37.

[37] S. Lee, M. Wheeler, Enriched Galerkin methods for two-phase flow in porous media with capillary pressure, J. Comput. Phys. 367 (2018) 65–86.

[38] S. Lee, A. Mikelic, M. Wheeler, T. Wick, Phase-field modeling of two phase fluid filled fractures in a poroelastic medium, Multiscale Model. Simul. 16 (4) (2018) 1542–1580.

[39] J. Abou-Kassem, M. Islam, S. Farouq-Ali, Petroleum Reservoir Simulations, Elsevier, 2013.

[40] J. Du, R. Wong, Application of strain-induced permeability model in a coupled geomechanics-reservoir simulator, J. Can. Pet. Technol. 46 (12) (2007) 55–61.

[41] F. Ballarin, G. Rozza, Multiphenics – easy prototyping of multiphysics problems in FEniCS, https://mathlab.sissa.it/multiphenics, 2019.

[42] M. Alnaes, J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. Richardson, J. Ring, M. Rognes, G. Wells, The FEniCS Project Version 1.5, Arch. Numer. Softw. 3 (100) (2015).

[43] K. Terzaghi, Theoretical Soil Mechanics, Chapman And Hall, Limited., London, 1951.

[44] J. Jaeger, N.G. Cook, R. Zimmerman, Fundamentals of Rock Mechanics, John Wiley & Sons, 2009.

[45] O. Coussy, Poromechanics, John Wiley & Sons, 2004.

[46] H. Yoon, J. Kim, Spatial stability for the monolithic and sequential methods with various space discretizations in poroelasticity, Int. J. Numer. Methods Biomed. Eng. 114 (7) (2018) 694–718.

[47] S. Lee, T. Kadeethum, H. Nick, Choice of interior penalty coefficient for interior penalty discontinuous Galerkin method for Biot's system by employing machine learning, preprint, arXiv:2007.10119.

[48] T. Kadeethum, H. Nick, S. Lee, C. Richardson, S. Salimzadeh, F. Ballarin, A novel enriched Galerkin method for modelling coupled flow and mechanical deformation in heterogeneous porous media, in: 53rd US Rock Mechanics/Geomechanics Symposium, American Rock Mechanics Association, New York, NY, USA, 2019.

[49] S. Balay, S. Abhyankar, M. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, A. Dener, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, D. May, L. McInnes, R. Mills, T. Munson, K. Rupp, P. Sanan, B. Smith, S. Zampini, H. Zhang, H. Zhang, PETSc users manual, Tech. Rep. ANL-95/11 - Revision 3.10, Argonne National Laboratory, 2018, http://www.mcs.anl.gov/petsc.

[50] E. Jones, T. Oliphant, P. Peterson, et al., SciPy: open source scientific tools for Python, http://www.scipy.org/, 2001.

[51] I. Babuska, The finite element method with Lagrangian multipliers, Numer. Math. 20 (3) (1973) 179–192.

[52] J. Jensen, L. Lake, P. Corbett, D. Goggin, Statistics for Petroleum Engineers and Geoscientists, vol. 2, Gulf Professional Publishing, 2000.

[53] M. Calvert, L. Vagg, K. Lafond, A. Hoover, K. Ooi, I. Herbert, Insights into sweep efficiency using 4d seismic at Halfdan field in the North Sea, Lead. Edge 33 (2) (2014) 182–187.

[54] A. Calvert, A. Cunningham, M. Motlagh, L. Jensen, T. Fuzeau, K. Ooi, Insights from the first 4d seismic over Valdemar lower cretaceous chalk field, in: 80th EAGE Conference and Exhibition 2018, European Association of Geoscientists & Engineers, 2018, pp. 1–5.