DARTS open-source reservoir simulation framework

Saifullin, I.; Voskov, D.; Novikov, A.; Wapperom, M.; Khait, M.; Tian, X.; Lyu, X.; De Hoop, S.; Orozco, L.; Palha, A.

DOI
10.3997/2214-4609.202437086

Publication date
2024

Document Version
Final published version

Published in
European Conference on the Mathematics of Geological Reservoirs, ECMOR 2024

ECMOR 2024
European Conference on the Mathematics of Geological Reservoirs
EAGE

DARTS open-source reservoir simulation framework

I. Saifullin[1], D. Voskov[1,2], A. Novikov[1], M. Wapperom[1], M. Khait[4], X. Tian[5], X. Lyu[6], S. De Hoop[7], L. Orozco[3], A. Palha[1,3], Y. Chen[1]

[1] TU Delft; [2] Stanford University; [3] Netherlands eScience Center; [4] Stone Ridge Technology S.R.L.; [5] Guangzhou Institute of Energy Conversion; [6] China University of Petroleum; [7] Alten

Summary

The open Delft Advanced Research Terra Simulator (open-DARTS) framework is an open-source reservoir simulation software. The open-DARTS focused on energy transition applications, such as geothermal energy production and carbon sequestration. It enables the modeling of compositional thermal flow, coupled with a geomechanical solver based on the Finite Volume discretization and adjoints method for inverse modeling. The open-DARTS supports different grid types (structured, corner-point geometry, and unstructured), discrete fracture networks, contact mechanics, and various thermal-chemical interactions. The recently proposed generic nonlinear formulation supports the most general nonlinear PDEs designed for various energy transition applications.

The open-DARTS has been implemented in C++ and Python to optimize hardware utilization while ensuring flexibility. The most computationally expensive part is written in C++ and compiled into libraries, which are subsequently exposed to Python using pybind11. This allows the extension and overriding of C++ functions by user-defined Python code. For example, using only a Python interface, one can adjust a timestep strategy, nonlinear solver, or properties output. Besides, the Python interface of open-DARTS provides straightforward coupling with other Python-based numerical modeling packages, including the meshing, file storage, caching, and visualization modules. The open-DARTS core uses the advantages of C++ language, such as efficient low-level memory management, object-oriented programming, compile-time polymorphism, and parallelization with OpenMP. One of the advantages of open-DARTS is the Operator-Based Linearization (OBL) technique, which can resolve challenges associated with complex physics and reduce the computation time, especially for ensemble-based simulations.

We would also like to share our experience on the project, repository, and the development workflow configuration using gitlab.com, including the build system (cmake), handling merge requests, automated testing in CI/CD pipelines, documentation management (gitlab.io), wiki utilization, and release publishing. Additionally, Python's integration into open-DARTS offers the advantage of straightforward installation via PyPI and simplifies defining requirements for users who prefer to avoid compiling code from source files.

Introduction

The transition toward sustainable energy sources has increased the interest in geothermal projects worldwide. Another direction to reduce the human impact is carbon capture and sequestration to the depleted gas fields or aquifers. The optimization and risk assessment of operational management of subsurface reservoirs for these and other applications can be done using numerical simulation. Various modeling features might be required depending on the reservoir properties, such as the Discrete Fracture Model (DFM) for fractured reservoirs, geomechanical modeling for induced seismicity risk assessment, rock dissolution for karstified reservoirs, etc.

In this study, we present an open-source reservoir simulation framework Delft Advanced Research Terra Simulator (open-DARTS). It follows the original DARTS development performed by Khait and Voskov (2017, 2018) which has been dedicated to modeling energy transition applications. Later, it was further extended for advanced CCUS applications (Lyu et al., 2021a,c), modeling of fractured reservoirs (de Hoop et al., 2022; Wang, 2021) and complex geomechanics with induced seismicity treatment (Novikov et al., 2022; Novikov, 2024).

The open-DARTS enables thermal compositional hydrodynamic and fully-coupled geomechanical modeling based on Finite Volume (FV) discretization, contact mechanics, and various thermal-chemical interactions. The adjoints gradients method implemented in the open-DARTS has proven its efficiency for inverse modeling (Tian and Voskov, 2022; Tian et al., 2024). The open-DARTS supports different grid types (structured, Corner Point Geometry, and fully unstructured), complemented with two-point and multi-point flux approximations. Since in the DFM, the meshes aligned to the fracture network might be challenging to use in simulation, the open-DARTS includes a fracture pre-processing module (de Hoop et al., 2022). It identifies the problematic fracture intersection points and adjusts the fracture network with different fracture and mesh resolution tuning parameters.

The reservoir development modeling requires the solution of governing equations describing the conservation of mass and energy. The nonlinear formulation for generic conservation equations includes the conservation of mass, energy, and momentum and supports the most general nonlinear PDE designed for various energy transition applications.

The simulation performance can be crucial in large ensembles of numerical models for uncertainty analysis (Wang et al., 2023; Major et al., 2023). The Operator-Based Linearization (OBL) technique (Voskov, 2017; Khait and Voskov, 2017) implemented in the open-DARTS, can significantly reduce the computation time and resolve challenges associated with complex physics, especially for ensemble-based simulations. The open-DARTS is written using C++ and Python languages to use hardware efficiently and at the same time provide flexibility. The open-DARTS core uses the advantages of C++ language, such as object-oriented programming, compile-time polymorphism, and parallelization with OpenMP. The Python interface of the open-DARTS makes it an easily extendable framework compared to black box reservoir simulators, which is especially beneficial for teaching purposes and developing research projects.

Governing Equations

In this section, we formulate generic governing equations including various physics required for modeling energy transition applications. We assume the investigated domain is split by control volumes Ω, bounded by surface Γ. The conservation equations can be expressed for this domain in a uniformly integral way, as

$$\frac{\partial}{\partial t}\int_{\Omega} M^c d\Omega + \int_{\Gamma} \boldsymbol{F}^c \cdot \boldsymbol{n} d\Gamma = \int_{\Omega} Q^c d\Omega. \tag{1}$$

Here, M^c denotes the accumulation term for the c^{th} component, $\boldsymbol{F}_c$ refers to the flux term of the c^{th} component, $\boldsymbol{n}$ refers to the unit normal pointing outward to the domain boundary, and Q_c denotes the source/sink term of the c^{th} component.

Accumulation term

Eq. 1 represents the conservation of mass, energy and momentum for a description of reservoir dynamics. Mass conservation is defined for the first n_c components as

$$M^c = \phi \sum_{j=1}^{n_p} x_{cj}\rho_j s_j, \quad c = 1,\dots,n_c, \tag{2}$$

where ϕ is porosity, s_j is phase saturation, ρ_j is phase density [kmol/m^3] and x_{cj} is molar fraction of c component in j phase. The energy accumulation term contains the internal energy of fluid and rock,

$$M^{n_c+1} = \phi \sum_{j=1}^{n_p} \rho_j s_j U_j + (1-\phi)\rho_r U_r, \tag{3}$$

where U indicates specific internal energy in [kJ/kmol]. The porosity is defined here as

$$\phi - \phi_0 = \frac{(\psi-\phi_0)(1-\psi)}{\mathrm{K}_s}(p-p_0) + \mathbf{B} : \nabla^s(\mathbf{u}-\mathbf{u}_0) + \alpha_\phi (T-T_0), \tag{4}$$

where p is pore pressure, $\nabla^s\mathbf{u} = (\nabla\mathbf{u} + (\nabla\mathbf{u})^T)/2$ is the matrix of symmetric gradients of displacements, $\mathbf{B}$ is the rank-two tensor of Biot's coefficients, $\psi = I_1(\mathbf{B})/3$ is one-third of the first invariant of $\mathbf{B}$, K_s is rock matrix drained bulk modulus, α_ϕ is the volumetric coefficient of rock matrix thermal dilation.

The accumulation term for momentum balance is equal to zero $\boldsymbol{M}^{n_c+2} = \mathbf{0}$ since we consider quasi-static approximation for geomechanics.

Flux term

The mass flux of each component is represented by the summation over n_p fluid phases,

$$F^c = \sum_{j=1}^{n_p} \left(x_{cj}\rho_j \boldsymbol{v_j} + s_j \mathbf{J}_{cj} \right), \quad c = 1,\dots,n_c. \tag{5}$$

Here the velocity $\boldsymbol{v_j}$ follows the extension of Darcy's law to multiphase flow,

$$\boldsymbol{v_j} = -\mathbf{K}\frac{k_{rj}}{\mu_j}(\nabla p_j - \gamma_j \nabla z), \tag{6}$$

where $\mathbf{K}$ is the permeability tensor [mD], k_{rj} is the relative permeability of phase j, μ_j is the viscosity of phase j [mPa$\cdot$s], p_j is the pressure of phase j [bars], $\gamma_j = \rho_j g$ is the specific weight of phase j [N/m^3] and z is depth [m], g is the gravitational acceleration [m/s^2]. The $\mathbf{J}_{cj}$ is the diffusion flux of component c in phase j, which is described by Fick's law as

$$\mathbf{J}_{cj} = -\phi \mathbf{D}_{cj} \nabla(\rho_j x_{cj}), \tag{7}$$

where $\mathbf{D}_{cj}$ is the diffusion coefficient [m^2/day].

The energy flux includes the thermal convection and conduction terms,

$$\boldsymbol{F}^{n_c+1} = \sum_{j=1}^{n_p} \left(h_j \rho_j \boldsymbol{v_j} + \phi \kappa_j s_j \nabla T \right) + (1-\phi)\kappa_r \nabla T, \tag{8}$$

where h_j is phase enthalpy [kJ/kmol] and κ_j is thermal conductivity for phase j [kJ/m/day/K] (including solid).

The vector of momentum fluxes is defined as

$$\boldsymbol{F}^{n_c+2} = -\boldsymbol{\sigma} \cdot \mathbf{n}, \tag{9}$$

where the rank-two Cauchy stress tensor $\boldsymbol{\sigma}$ [bar] defines the mechanical response of the porous rocks. We consider a linear thermoporoelastic material rheology. Under the assumption of infinitesimal strains, the rheology imposes the following relationship on the stress tensor

$$\boldsymbol{\sigma} - \boldsymbol{\sigma}_0 = \mathbb{C} : \nabla^s (\mathbf{u} - \mathbf{u}_0) - (p - p_0)\mathbf{B} + (T - T_0)\mathbf{A}, \tag{10}$$

where $\mathbb{C}$ is the rank-four drained stiffness tensor [bar], $\nabla^s \mathbf{u} = (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)/2$ is the matrix of symmetric gradients of displacements, $\mathbf{A}$ is the rank-two rock matrix thermal dilation tensor [bar/K], $\mathbf{u}$ is a vector of displacements [m].

Source terms

The source term in mass conservation equations can be present in the following form

$$Q^c = \sum_{k=1}^{n_k} v_{ck} r_k, \quad c = 1, \ldots, n_c, \tag{11}$$

where q_j is the phase source/sink term from the well, v_{ck} is the stoichiometric coefficient associated with chemical reaction k for the component c and r_k is the rate for the reaction. Similarly, the source term in the energy balance equation can be written as

$$Q^{n_c+1} = \sum_{k=1}^{n_k} v_{ek} r_{ek}. \tag{12}$$

Here v_{ek} is the stoichiometric coefficient associated with kinetic reaction k for the energy and r_{ek} is the energy rate for kinetic reaction.

The volumetric (gravitational) forces contributed to the momentum balance read

$$\boldsymbol{Q}^{n_c+2} = -\left((1-\phi)\gamma_s + \phi \sum_j^{n_p} s_j \gamma_j \right) \nabla z, \tag{13}$$

where $\gamma_s = \rho_s g$ is the specific weight of rock matrix [N/m^3], ρ_s is the density of rock matrix.

Method of solution

The nonlinear Eqs. 1 are discretized using finite volume space discretization with a either two-point or multi-point flux approximation (MPFA) for flow problem and MPFA-MPSA ('S' stands for stress) for geomechanical problem Novikov (2024) and upstream weighting for transport terms. For approximation in time, a backward Euler approximation yields an implicit approximation for the flux and source terms. The discretized residual form for a reservoir block i reads as

$$g_i^c = V_i \left(M_i^c(\omega_i) - M_i^c(\omega_i^n) \right) - \Delta t \left(\sum_l A_l F_l^c(\omega) + V_i Q_i^c(\omega) \right) = 0, \quad c = 1, \ldots, n_c + 1, \tag{14}$$

where V_i is the volume of i-th grid block.

The fully implicit time approximation requires the flux term to be defined based on the nonlinear unknowns at the new timestep $(n+1)$, which introduces nonlinearity to the system of governing equations. The nonlinear solution is based on the extension of molar formulation Collins et al. (1992) with temperature and displacements $\mathbf{u}$ used as additional nonlinear variables

$$\boldsymbol{\omega} = \{p, \boldsymbol{z}, T, \mathbf{u}\}. \tag{15}$$

Next, the Newton-Raphson approach is used for teh solution of nonlinear governing equations. the Jacobian and the residual are constructed at each nonlinear iteration and the following linear system of equations is solved:

$$\frac{\partial \boldsymbol{g}(\boldsymbol{\omega}^k)}{\partial \boldsymbol{\omega}^k}(\boldsymbol{\omega}^{k+1} - \boldsymbol{\omega}^k) = -\boldsymbol{g}(\boldsymbol{\omega}^k). \tag{16}$$

The conventional nonlinear solution approach involves the evaluation and storage of all properties and their derivatives with respect to the nonlinear unknowns $\boldsymbol{\omega}$. In the open-DARTS, we use the Operator-Based Linearization (OBL) approach Khait and Voskov (2017) which significantly simplifies the linearization procedure.

In the OBL approach, the Eqs. 14 can be present as a product of the most nonlinear terms only dependent on unknowns $\boldsymbol{\omega}$ and another, almost linear terms, changing in space (like porosity, permeability, etc.) The main advantage of the OBL approach is the simplicity of the Jacobian assembly for complex physical models since it can be represented following a chain rule as a sum of products of linear operators representing partial derivatives of state-dependent terms and the rest, less nonlinear, terms. The Jacobian and residual construction can also be adjusted to modern computational architectures which improves the performance of linearization by almost two orders of magnitude Khait and Voskov (2021). A unique feature of the approach is an opportunity to integrate any nonlinear relationship as a black box without any derivative information. More details and validation of the OBL approach can be found in Wang et al. (2020); Lyu et al. (2021a); Ahusborde et al. (2024).

Governing physics

In target reservoirs for subsurface energy and carbon sequestration applications, highly saline brines are commonly present, along with a range of dissolved gases. In depleted hydrocarbon reservoirs, the interaction between CO_2 light to heavier hydrocarbon fractions and impurities introduces complex physical behavior. Potentially, injected CO_2 forms a separate phase and thermal effects related to phase transitions, mixing and adiabatic expansion may become significant. Furthermore, the potential occurrence of solid phases, such as precipitated salt, gas hydrates or ice, can have pronounced effects on near-wellbore flow patterns and should be considered in the physical modeling approach.

Thermodynamic modelling

The robustness and accuracy of the compositional simulation are determined by the thermodynamic modeling routines. Often, instantaneous thermodynamic equilibrium between reservoir fluids in a control volume is assumed. This implies that at each nonlinear iteration, a solution of thermodynamic equilibrium is required in each grid block. Compositional simulation based on PT- and PH-formulations demands thermodynamic modeling procedures for the different state specifications (Gibbs free energy minimization and entropy maximization, respectively). In addition, kinetic reactions can be modeled by means of the kinetic operators (11)-(12).

The open-DARTS framework provides large flexibility for implementing thermodynamic modeling procedures. Taking advantage of the OBL approach, the user is not required to provide derivatives of thermodynamic and physical properties for assembly of the Jacobian, as they are obtained automatically upon interpolation between supporting points in the OBL grid. Physical properties are defined and evaluated in a Python class called *PropertyContainer*, where users can easily adjust methods for thermodynamic and physical modeling by overriding the default implementations. Default implementations exist for the single-component PH-based *Geothermal* engine and for the more generic *Compositional* engine, based on a multicomponent-multiphase PT-formulation.

For the *Geothermal* engine, Wang (2021) defined a wrapper that employs the open-source industrial standard IAPWS97 package in Python to compute the properties of water and steam. An ensemble of benchmark studies and comparisons with the known reservoir simulators, i.e. Automatic Differentiation General Purpose Research Simulator (AD-GPRS) and TOUGH2 (Battistelli et al., 1997; Croucher and O'Sullivan, 2008; Wong, 2015) demonstrate that the simulation results from open-DARTS have a good

match for both low- and high-enthalpy conditions comparing to the ones from other simulators (Wang et al., 2020).

In parallel to open-DARTS, the development of the DARTS-flash thermodynamic package designed particularly for energy transition applications, extends the capability for open-DARTS to accurately simulate multiphase flow and transport in CO_2-sequestration and geothermal-related technologies. Similar to the open-DARTS architecture, computationally expensive core procedures of DARTS-flash are written in C++ and exposed to Python using pybind11 for use in open-DARTS. The library provides implementations of a range of thermodynamic models for different phase types: equations of state (EoS) such as PR (Peng and Robinson, 1976) and SRK (Soave, 1972) EoS, fugacity-activity models for brine with dissolved gases and ions (Ziabakhsh-Ganji and Kooi, 2012; Jager et al., 2003), and various hydrate and solid EoS (Van der Waals and Platteeuw, 1958; Ballard and Sloan, Jr., 2002).

In addition, the open-DARTS simulator provides possibilities for CO_2-foam-assisted storage processes modeling. To introduce the capability of CO_2-foam-related simulations into the open-DARTS, we first modify the conservation equations based on the physical models. Considering the capability and complexity of the population-balance (PB) model, an implicit-texture (IT) model is implemented to modify the gas relative permeability (i.e., changing operators) in the presence of foam (Lyu et al., 2021b,c).

Geomechanical capabilities

The assessment of risks associated with the development of subsurface reservoirs necessitates geomechanical modeling. Issues such as caprock integrity, land subsidence, induced seismicity, wellbore stability, and fault stability may threaten geo-energy utilization. Numerical modeling of the coupled thermo-hydro-mechanical-compositional (THMC) processes can provide a comprehensive analysis of these issues, helping to understand and minimize undesirable outcomes.

Open-DARTS employs a fully implicit approach based on the cell-centered collocated numerical scheme of the Finite Volume Method (FVM) to solve the coupled system of PDEs representing mass, momentum and energy conservation laws. The employed FVM benefits from the unified treatment of fluid mass, heat and momentum fluxes on an arbitrary star-shaped cell topology, facilitating THMC modeling at both core and reservoir scales. The geomechanical formulation includes linear thermo-poroelasticity for continuum media and frictional non-opening contact for faulted reservoirs. These modeling capabilities have been validated against several analytical (Novikov et al., 2024) and numerical (Novikov, 2024) references, some of which are presented below. Additionally, the coupled THMC modeling is enhanced by an iterative preconditioning strategy which enables detailed geomechanical analysis.

Discretization

The discretization module is a part of the solution process of PDEs described above using the FVM. This module is responsible for computing cell volumes, centroids and connection list with transmissibilities, filtering cells, and exporting the data for visualisation. Although this step is typically evaluated only once, it can be crucial for achieving good performance in highly-resolved models with millions of cells. Note that in open-DARTS, the mesh is static and currently cannot be changed during the simulation. Open-DARTS supports a few mesh types: structured grid, Corner Point Geometry (CPG), and unstructured mesh. Initially, the modules for structured and unstructured meshes were implemented in Python. To overcome the scalability bound due to slow loop processing in Python, the mesh processing for CPG and unstructured meshes was implemented in C++ with a Python interface.

Corner Point Geometry

Since a structured grid can be represented by a CPG grid, its separate C++ implementation was unnecessary. Apart from the structured grid, a CPG grid can have non-neighbor connections in fault and pinch-out zones while retaining IJK cell indices. Besides, the following capabilities, which are of high practical use, are supported in open-DARTS: over- and under-burden layers generation, fault transmis-

sibility multiplier, and setting the volume multiplier at boundaries for grids with active cells mask. Only two-point flow discretization (TPFA) is supported for this grid type.

Unstructured mesh

Open-DARTS includes modules with both Python and C++ implementations for unstructured mesh processing. Both implementations consider the Gmsh format as an input. In addition to TPFA, multi-point flux approximation (MPFA) for flow models and MPFA-MPSA ('S' stands for stresses) for geomechanical models are supported for this mesh type, but only in C++ implementation. In this method, the neighbors of a cell's neighbors also contribute to the flux computation at the cell's interfaces.

Discrete fracture modelling

Implementation in open-DARTS

Discrete Fracture Network (DFN) modeling in the open-DARTS is possible due to unstructured meshing capabilities and an implementation of the Discrete Fracture Model (DFM) method (Karimi-Fard et al., 2004). In the DFM approach, fractures are modeled as lower-dimensional features within the gridding domain, represented by lines in 2D and planes in 3D; however, during discretization, each fracture segment (which is positioned on the edge of a matrix cell) is assigned an aperture and hence becomes a 3-dimensional volume in the computational domain. The advantage is that the same equations can be solved in the fracture and matrix cells and no coupling between different domains is necessary. Intermediate fracture-fracture intersections that result in small volumes are resolved using the star-delta transformation. An example of a DFM model in the open-DARTS can be found in the repository: DFM model in open-DARTS.

Complex natural fracture networks

Natural fracture networks are prevalent in subsurface applications, such as geothermal energy production. These fractured reservoirs often exhibit highly complex structures, which makes modeling flow and transport both slow and unstable, negatively impacting the simulator's convergence. As a result, this complexity limits our ability to perform uncertainty quantification, increasing development costs and can pose environmental risks. Within the open-DARTS project, a preprocessing module has been developed that attempts to overcome this limitation (de Hoop et al., 2022). The preprocessing module generates a fully conformal, uniformly distributed grid for realistic 2D fracture networks with the desired level of precision. The simplified topology directly enhances meshing outcomes, thereby improving the accuracy and efficiency of numerical simulations.

Reactive transport in complex fracture networks

Besides applying two-phase geothermal flow to complex fracture networks at different scales, within the open-DARTS, there also exists the capability of modeling the rock dissolution (i.e., kinetic and equilibrium reactions) in complex fracture networks. The open-DARTS considers that changes in porosity (i.e., degree of dissolution or precipitation) induce changes in permeability and, therefore, changes in pressure and fluid pathways. The main application in which this was tested and developed was the study of cave formation de Hoop et al. (2020); de Hoop (2022). Increasing fracture aperture results in localized dissolution, while decreasing aperture negates the effects of the fractures altogether.

Furthermore, the effect of dispersion on dissolution and flow in fracture networks was investigated. It was evident that significant effective dispersion nearly eliminates the influence of the fractures. Conversely, too little effective dispersion leads to the complete bypass of dissolution within the fracture network, preventing fluid from penetrating the matrix surrounding the fracture walls and resulting in no dissolution. Between these extremes, there are two regions: a highly dispersed giant tunnel with widespread dissolution occurring closer to the source; and very regular dissolution patterns around the fracture walls resulting in a maze-like structure.

Adjoint gradients

In history matching and inverse modeling, it is essential to evaluate gradients efficiently; otherwise, the process becomes exceedingly time-consuming due to the multiple iterations required. To increase the efficiency of gradient evaluation, the adjoint method is implemented in the open-DARTS framework. The adjoint method introduces a Lagrange multiplier λ to combine the objective function E with the governing equation $\boldsymbol{g}$ of the forward simulation. As a result, the new augmented objective function $\bar{\mathscr{E}}$ can be expressed as:

$$\bar{\mathscr{E}}(\boldsymbol{\omega},\boldsymbol{u},\boldsymbol{\lambda}) = E(\boldsymbol{\omega},\boldsymbol{u}) + \boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{g}(\boldsymbol{\omega},\boldsymbol{u}), \tag{17}$$

Where $\boldsymbol{\omega}$ is the state variable and $\boldsymbol{u}$ is the control variables. The term $\boldsymbol{\lambda}^T$ refers to the transposed form of the Lagrange multiplier. Specifically, the objective function of history matching E is defined as (with the state variable $\boldsymbol{\omega}$ omitted):

$$E(\boldsymbol{u}) = ||G(\boldsymbol{u}) - \mathbf{d}_{\mathrm{obs}}||_2^2, \tag{18}$$

where $G(\boldsymbol{u})$ is the model response and $\mathbf{d}_{\mathrm{obs}}$ is the observation data. In the open-DARTS, we take the transmissibility of each interface $\boldsymbol{u}$ as the control variables. Following Tian and Voskov (2022) and Tian et al. (2024), the discretized adjoint method can then be written as:

$$\boldsymbol{\lambda}_{k+1}^{\mathrm{T}}\frac{\partial \boldsymbol{g}_{k+1}}{\partial \boldsymbol{\omega}_k} + \boldsymbol{\lambda}_k^{\mathrm{T}}\frac{\partial \boldsymbol{g}_k}{\partial \boldsymbol{\omega}_k} + \frac{\partial \boldsymbol{j}_k}{\partial \boldsymbol{\omega}_k} = \mathbf{0}, \tag{19}$$

$$\boldsymbol{\lambda}_K^{\mathrm{T}}\frac{\partial \boldsymbol{g}_K}{\partial \boldsymbol{\omega}_K} + \frac{\partial \boldsymbol{j}_K}{\partial \boldsymbol{\omega}_K} = \mathbf{0}, \tag{20}$$

$$\frac{\partial \bar{\mathscr{E}}}{\partial \boldsymbol{u}} = \sum_{k=1}^{K}\left(\boldsymbol{\lambda}_k^{\mathrm{T}}\frac{\partial \boldsymbol{g}_k}{\partial \boldsymbol{u}_k} + \frac{\partial \boldsymbol{j}_k}{\partial \boldsymbol{u}_k}\right), \tag{21}$$

where K is the total number of the simulation time steps, $\boldsymbol{j}_k$ is the misfit between the model response and the observation data at the given simulation time step k. Therefore $E(\boldsymbol{\omega},\boldsymbol{u})$ can also be defined as $E(\boldsymbol{\omega},\boldsymbol{u}) = \sum_{k=1}^{\mathrm{K}} \boldsymbol{j}_k$. The gradient $\frac{\partial \bar{\mathscr{E}}}{\partial \boldsymbol{u}}$ can be determined using Eq. (21), where $\boldsymbol{\lambda}$ can be solved from Eq. (20) and Eq. (19) recursively in a backward manner.

Benchmarks

In this section, we first demonstrate open-DARTS scalability using geothermal model. Next, we present the results of geomechanical models and their consistency with analytical solutions.

Scalability test with a geothermal model

The geological model (Reinhard, 2019) utilized in the geothermal simulation study was created in the past using data collected from the existing geothermal wells located in the West Netherlands Basin (WNB). The model represents a typical fluvial system with 9 major faults. Originally, 90 porosity realizations were generated based on the facies interpretation with the same geological geometry. The porosity-permeability correlation from Willems et al. (2020) is applied to derive the permeability field. These 90 realizations consist of circa 38 million grid cells to properly characterize the heterogeneity of the reservoir. One reference base realization is chosen to investigate the scalability of open-DARTS with respect to different numbers of active grid cells. Following Fig. 1 demonstrates the geometry of the full geological model. Different grid resolutions were used to benchmark the scalability of the open-

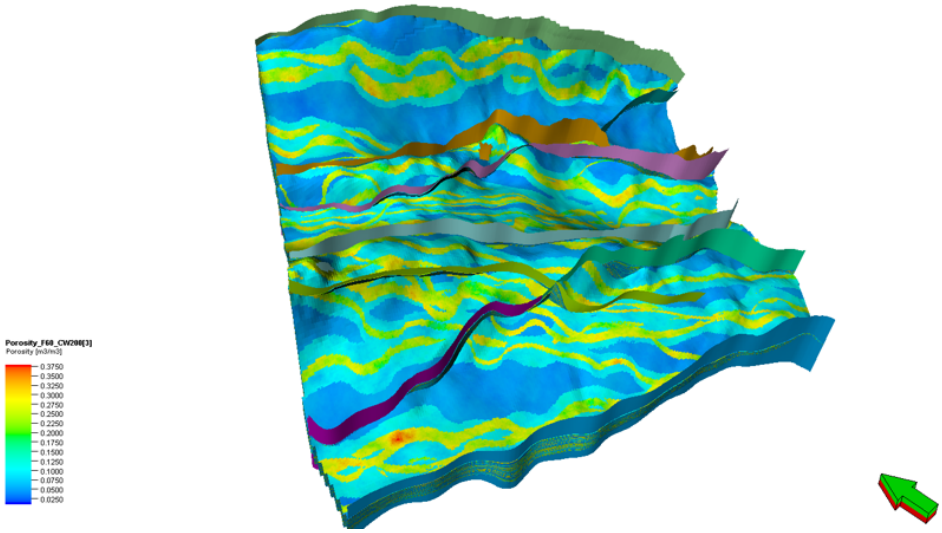


Figure 1: The porosity and faults distribution of the reference base model

DARTS. The grid refinement is carried out by adjusting the number of reservoir layers of the base model while keeping the thickness constant. At the same time, the horizontal resolution and the structure of the faults are preserved. The grid has an active cell mask on the boundaries, and the number of active cells is around half of the total cell number. The cells that correspond to shale facie, the number of which reaches 50% of the total active cell number in the reservoir part, were kept active in the simulation as they serve as thermal recharge sources. We assigned a large effective volume to the active cells located at the lateral boundaries which simulate an aquifer.

The model evaluates the thermal front propagation in geothermal reservoir with six vertical wells during 30 years. The maximal timestep is set to 365.25 days and nonlinear and linear tolerances are set to 1e-2 and 1e-4 correspondingly. The open-DARTS version used is ae5ac63 and was compiled with GCC-13 and default options (-O3). The benchmark was performed on a Linux machine with OS RHEL 8.10 and CPU AMD EPYC 7742 which had almost its maximal frequency 3.4 GHz during the testing. Thread pinning to CPU cores using the environment variable GOMP_CPU_AFFINITY was applied. The grid cells count, simulation time, Newton iteration number (NI) and linear iteration number (LI) for the run using 16 OpenMP threads are listed in Table 2. The number of layers specified there excludes under and overburden layers, whereas the number of active cells contains all the layers. Although the CPU has a larger number of cores, we didn't run the simulation with more than 16 threads as the parallel scalability of our simulator is limited by sequential parts of code and also sparse matrix operations are memory-bound problems as outlined in (Khait et al., 2020). The memory consumption for all grids used in this model is around 4.2 Kb per cell. Note that for this setup we used the Geothermal physics type, which has two variables per cell - pressure and enthalpy. The temperature at 30 years for the grid case with 100 layers is illustrated in Figure 2.

Simulation time with different mesh resolutions using 16 threads (T_{16}) and 1 thread (T_1) are shown in Figure 3, where both runs demonstrate a linear scalability. The speedup achieved (T_1/T_{16}) is 3.3-3.5 for all grids used and it is limited by sequential parts: initialization and precondition step ILU(0) in the linear solver.

Table 1: Parameter settings for reservoir model

Parameter	Value
Grid dimensions	$268 \times 259 \times (20-500)$
Average grid cell size, m	$50 \times 50 \times (0.5-5)$
Permeability, mD	(0.004–1113)
Porosity, -	(0.01–0.264)
Sandstone heat capacity, $kJ/m^3/K$	2450
Sandstone heat conductivity, $kJ/m/day/K$	259.2
Shale heat capacity, $kJ/m^3/K$	2300
Shale heat conductivity, $kJ/m/day/K$	190.08
Initial temperature, K	350
Initial pressure, bars	200
Injection temperature, K	298.15
Rate (4 doublets), m^3/day	[4500, 5000, 3200, 2600]

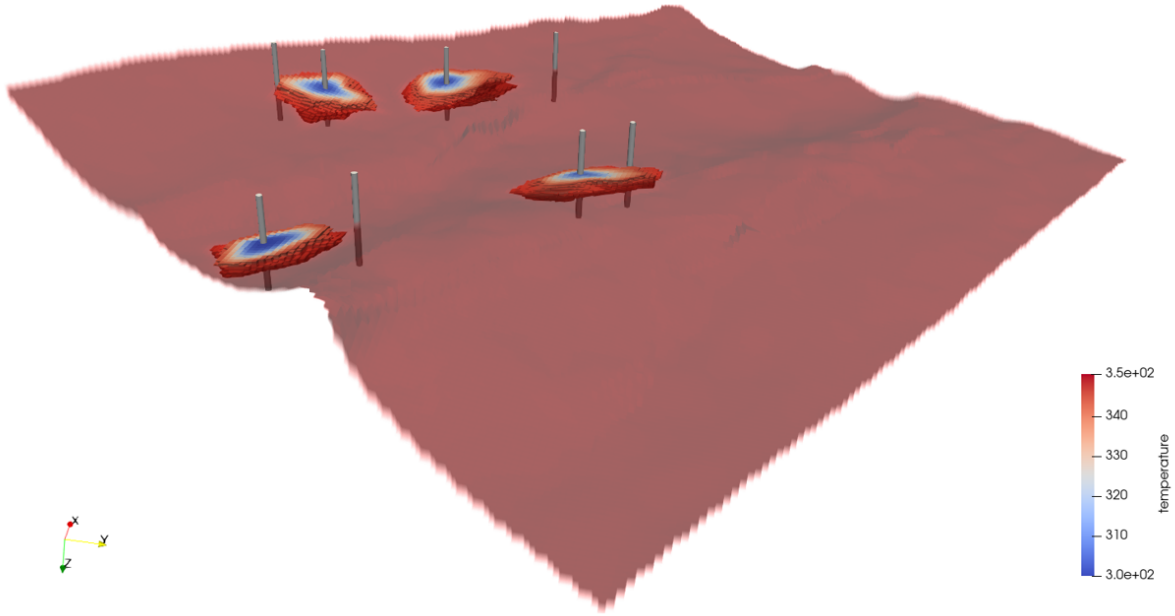


Figure 2: The temperature distribution after 30 years of geothermal energy production in WNB for the grid case with 100 layers.

Table 2: Scalability test using the geothermal model in WNB, the run using 16 OpenMP threads

Number of layers	Active cells, mln.	Simulation time, min	Time steps	NI	LI
20	0.78	1.38	37	76	740
50	1.84	3.33	37	80	793
100	3.62	7.20	37	85	890
200	7.17	15.9	37	90	1021
250	8.94	20.6	37	93	1076
500	17.74	46.8	37	105	1252

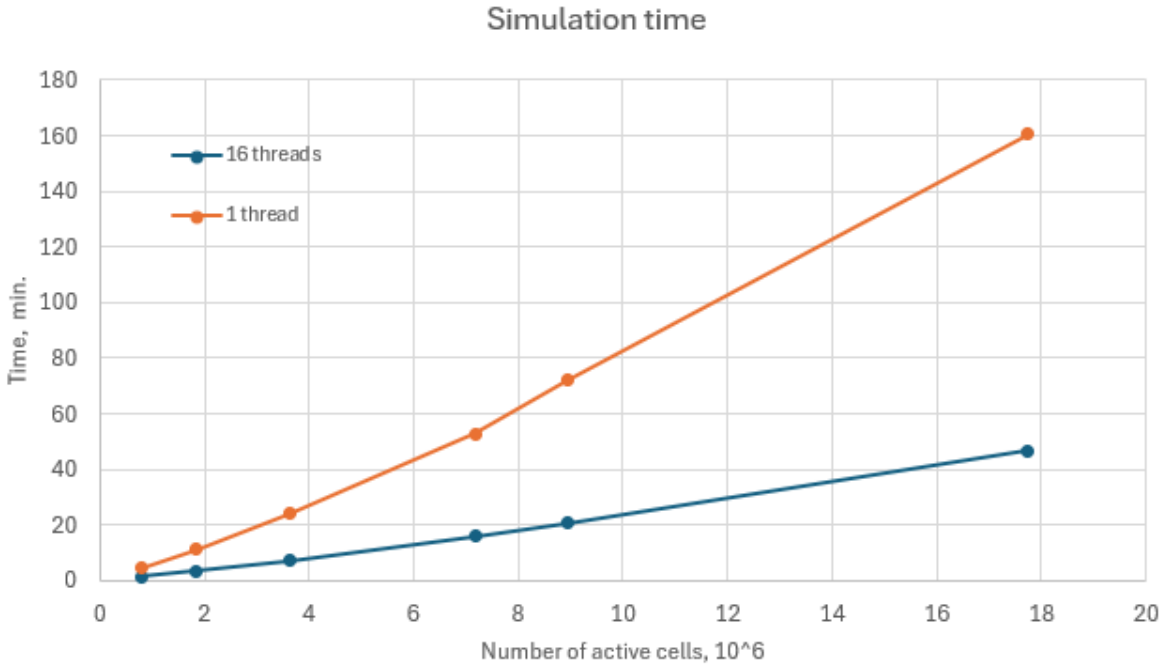


Figure 3: Simulation time of geothermal model with different mesh resolution using 1 and 16 threads.

Table 3: The properties of two layers in the two-layer Terzaghi setup.

Layer	h, m	E, GPa	ν	b	k, mD	ϕ_0	μ_f, cP	β_f, 1/bar
1	25	1	0.35	0.9	1	0.15	1	10^{-10}
2	75	5	0.0053	0.01	0.001	0.15		

Uniaxial poroelastic consolidation (Terzaghi's problem)

We further validate the numerical scheme against the analytical solution for the unidimensional consolidation problem, also known as Terzaghi's problem. As the analytical solution in the presence of the two heterogeneous layers in Terzaghi's problem remains feasible (Verruijt, 2016), we consider a more general setup that includes two distinct layers. The poroelastic domain shown in Fig. 4 of vertical extent $h = 100\,\text{m}$ is comprised by two layers of distinct properties with $h_1 = 25\,\text{m}$, $h_2 = 75\,\text{m}$ respectively. The first layer is adjacent to the right boundary which is subjected to constant normal load $F = 10\,\text{MPa}$ and constant initial pressure $p_0 = 0\,\text{Pa}$. All other sides of the domain are impermeable to fluid and subjected to the roller boundary condition (normal displacement and tangential traction are equal to zero). The domain's permeabilities $\mathbf{K} = k\mathbf{I}$ and Biot's tensors $\mathbf{B} = b\mathbf{I}$ are defined by their scalar counterparts k and b, respectively, while the stiffness tensors are determined by Young's moduli E and Poisson's ratios ν. The properties of porous matrix and fluid including initial porosities ϕ_0, fluid viscosity μ_f and fluid compressibility β_f are listed in Tab. 3.

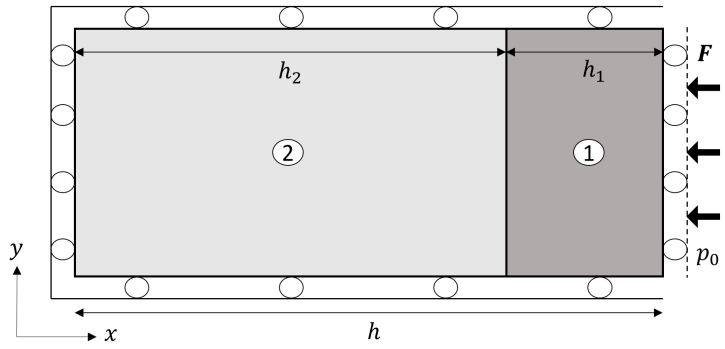


Figure 4: Two-layer Terzaghi setup.

The comparison of results is presented in Fig. 5. We use a uniform mesh comprised of 40 cells so that the top left subfigure (Fig. 5a) demonstrates pressure over time evaluated at $x = 1.25\,\text{m}$ and the bottom left subfigure (Fig. 5c) shows the dynamics of horizontal displacement u_x taken at $x = 98.75\,\text{m}$. The top and bottom right subfigures (Fig. 5b, Fig. 5d) present pressure and horizontal displacement profiles over the domain respectively. Numerical results remain in good accordance with the analytical solution. For the simplicity of the analytical solution, we assume the Skempton's coefficient is equal in both layers. As a result, the compression of the poroelastic domain causes an instant pressure build-up, uniform throughout the whole domain. The thin boundary layer originates at the right side of the domain subjected to constant initial pressure. Numerically it produces spurious oscillations that disappear over time. They can be observed in Fig. 5b.

Biaxial poroelastic consolidation (Mandel's problem)

Consider the same domain as in the previous section with homogeneous properties and different boundary conditions illustrated in Fig. 6. Now roller boundary conditions are applied only to the left and bottom boundaries of the domain. The right boundary is free of both normal and tangential forces while a normal load is applied from the top. Note that this load is applied through the stiff bulk in a way that produces uniform vertical displacement. Therefore, it could be more convenient to specify time-dependent normal displacement at the top estimated from analytical expressions. No-flow conditions are specified for all boundaries except for the right one subjected to the Dirichlet condition $p_0 = 0\,\text{Pa}$.

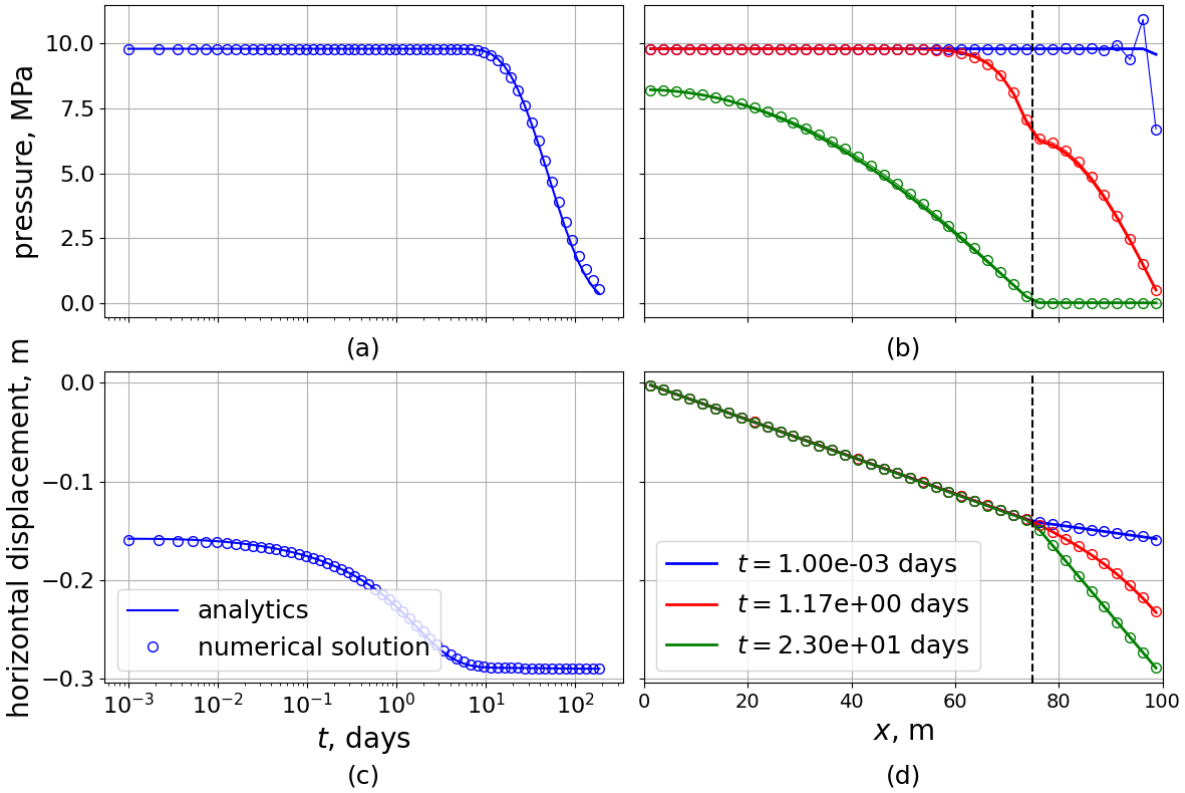


Figure 5: Comparison of the analytical and numerical solutions for the two-layer Terzaghi's problem. In the left column, pressure (a) and displacement (c) at the center of the most left cell are shown over time. In the right column, pressure (b) and displacement (d) profiles over the whole domain are depicted at three moments of time.

This setup is the so-called Mandel's problem which is often used as an example of non-monotonic pressure behavior following undrained loading.

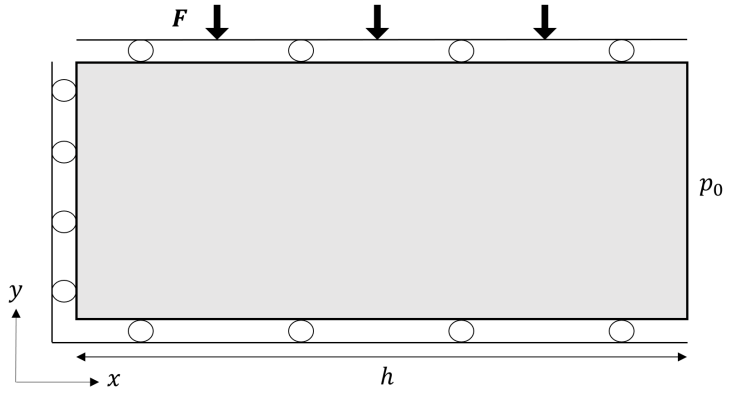


Figure 6: Mandel setup.

A porous homogeneous domain is characterized by Young's modulus E = 1 GPa, Poisson's ratio $\nu = 0.25$, a diagonal permeability tensor $\mathbf{K} = k\mathbf{I}, k = 1$ mD, saturated with a single-phase compressible fluid with compressibility $\beta_f = 10^{-5}1/\text{bar}$, viscosity $\mu_f = 1$ cP, and with a Biot modulus $M = 10^{-5}1/\text{bar}$ and a diagonal Biot tensor $\mathbf{B} = b\mathbf{I}, b = 0.9$.

For the numerical solution, we use 30×30 square mesh. Fig. 7 depicts a comparison between the numerical solution and analytics (Verruijt, 2016). As in the previous section, the top left subfigure (Fig. 7a) shows the pressure dynamics evaluated at $x = 1.66$ m, the bottom left subfigure (Fig. 7c) shows

the dynamics of horizontal displacement u_x at $x = 98.33\,\text{m}$, the right top (Fig. 7b) and right bottom (Fig. 7d) subfigures illustrate the profiles of pressure and vertical displacement correspondingly over horizontal centerline at three moments of time. The numerical solution matches analytics quite well. As in the previous example, spurious oscillations arise around the right side of the domain at the very beginning of the simulation. They can be seen in the top right subfigure (Fig. 7b).

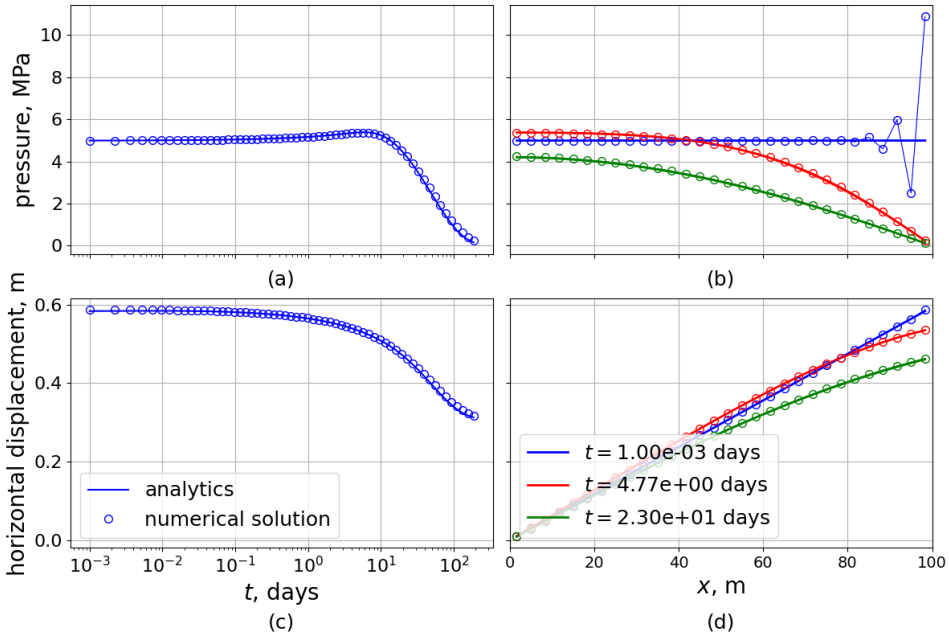


Figure 7: Comparison of the analytical and numerical solutions for Mandel's problem. In the left column, pressure (a) and displacement (c) at the center of the most left cell are shown over time. In the right column, pressure (b) and displacement (d) profiles over the whole domain are depicted at three moments of time.

Uniaxial thermoporoelastic consolidation

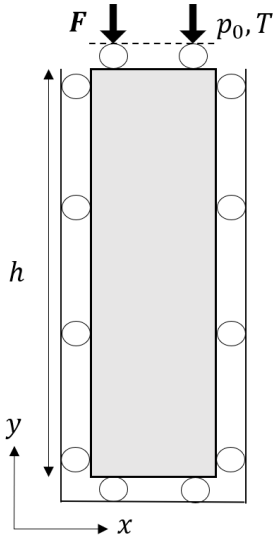
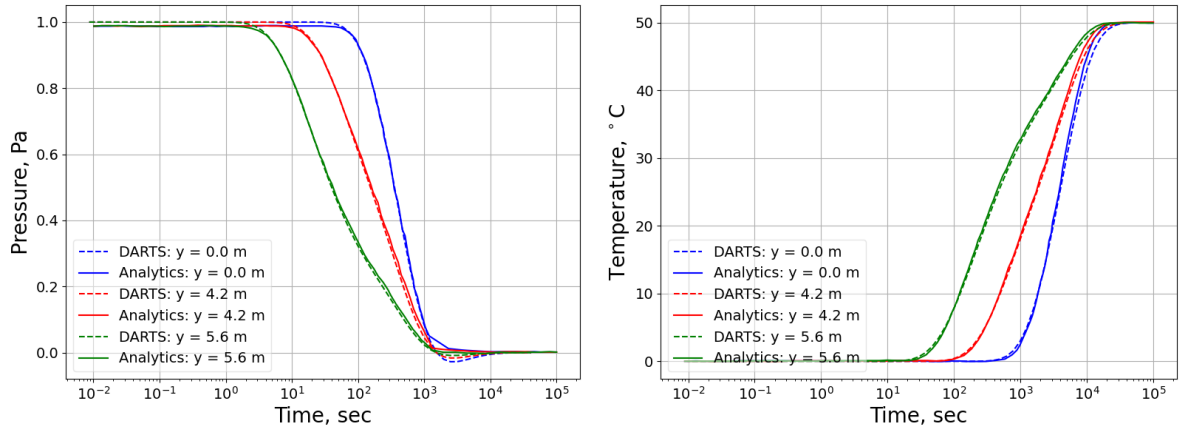


Figure 8: The setup for uniaxial thermoporoelastic consolidation test (Gao and Ghassemi, 2020).

The semi-analytical solution for a coupled problem of fluid mass, energy, and momentum balance in the uniaxial consolidation test (Bai, 2005) can be employed to benchmark the developed scheme. Thus, we consider the vertical column of 7 m height subjected to instant vertical loading with $F = 1\,\text{Pa}$ at the top boundary (Gao and Ghassemi, 2020). The constant pressure $p = 0\,\text{Pa}$ and temperature $T = 50\,^{\circ}\text{C}$

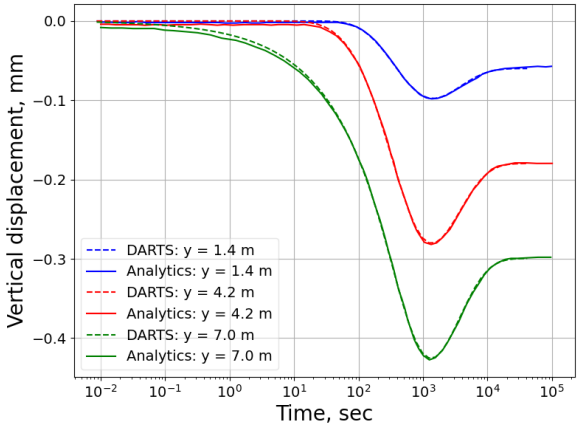
is maintained at the top boundary while the initial pressure and temperature are equal to $p_0 = 0\,\text{Pa}$ and $T_0 = 0\,^\circ\text{C}$, respectively. All other sides of the domain are impermeable to fluid and heat and subjected to the roller boundary condition. The domain is shown in Fig. 8.

Furthermore, the stiffness tensor is defined by Young's modulus $\text{E} = 6\,\text{kPa}$ and Poisson's ratio $\nu = 0.4$. Isotropic permeability $\mathbf{K} = k\mathbf{I}$, Biot's $\mathbf{B} = b\mathbf{I}$, thermal dilation $\mathbf{A} = a\mathbf{I}$ and total heat conduction $\mathbf{\Lambda} = \lambda\mathbf{I}$ tensors are defined by the corresponding scalar values $k = 4 \times 10^{-9}\,\text{m}^2$, $b = 1.0$, $a = 9 \times 10^{-7}\,1/^\circ\text{C}$ and $\lambda = 836\,\text{J}/(\text{m}\,\text{s}\,^\circ\text{C})$, respectively. The total volumetric heat capacity is equal to $\text{c} = 167.2\,\text{kJ}/(\text{m}^3\,^\circ\text{C})$. Fluid is maintained incompressible with fluid viscosity $\mu = 1\,\text{cP}$.



(a) pressure

(b) temperature



(c) vertical displacement

Figure 9: The dynamics of pressure (a), temperature (b) and vertical displacement (c) over time estimated in three points in space with semi-analytical method and open-DARTS.

Fig. 9 illustrates the comparison of pressure, temperature, and vertical displacement to the analytical solution. Pressure, temperature, and vertical displacement evaluated at three points in space are plotted against time. The numerical solution obtained with the proposed scheme (open-DARTS) demonstrates a good match to the analytical solution (Analytics).

An instant loading causes an instant compression followed by further consolidation due to fluid discharge as in the uniaxial poroelastic consolidation test. However, heat conduction propagates energy from the top boundary, which is maintained under high temperature $T = 50\,^\circ\text{C}$, throughout the whole domain. Temperature increase causes thermal expansion, competing with consolidation.

Software project and implementation features

The open-DARTS software project is publicly available in gitlab.com/open-DARTS under Apache 2.0 license and registered in the Research software directory. Users and developers are encouraged to check both the documentation and the wiki.

Build system and third-parties

The building system of the open-DARTS is implemented in CMake, allowing for compilation in different systems (Windows, Linux and MacOS tested) supporting different platforms such as Makefiles, Visual Studio solutions, Ninja, etc. External dependencies are handled as git sub-modules in *thirdparty*. Among the benefits of this approach are:

- Easier linking to dependencies in CMake,
- Allows to set the versions of the dependencies safeguarding reproducibility,
- Reduce the size of the repository and the code to maintain,
- From a copyright perspective: git sub-modules makes explicit the use of external dependencies.

Python interface

The most computationally expensive part of the open-DARTS is written in C++ and it is compiled into libraries, which are then exposed to Python using pybind11. This allows the extension and overriding of C++ functions by user-defined Python code. For example, one can adjust a time-step strategy, nonlinear solver, or properties output only using a Python interface. The open-DARTS Python interface allows to evaluation of dynamic properties for solving PDEs using user-written Python code, which makes it easier to extend the capabilities. For example, the state-dependent water properties in the open-DARTS geothermal model are evaluated using a Python package IAPWS97, and the open-DARTS compositional model can be linked with the open-source Flash calculations library implemented in C++, which has a Python interface as well. In combination with the dynamic interpolation using OBL, this brings a good balance between performance and flexibility. Furthermore, various freely available external python modules are used in the Python part of the open-DARTS code: array operations (numpy), mesh importing (gmsh), storing the results (vtk, pickle), data processing (pandas, scipy) and many others.

CI/CD

In the open-DARTS project, we exploit the Continuous Integration and Continuous Delivery (CI/CD) practices to automate the code check, the documentation update and the release publishing processes. The project repository contains a set of models that covers the most used simulation options with different physics, meshes and other features. These models serve as a validation benchmark for the regressions testing and they are updated continuously in case of the open-DARTS Python API modification. Following the merge request practices, with the test model set we ensure the open-DARTS code with changes introduced stays valid. The open-DARTS binaries for each Python version require compilation and we keep testing only one Python version to reduce the testing time. The complete set of wheels for all Python versions supported is compiled only for the releases. When making a release the Python wheels of the project are generated an published to PyPI. This allows for an installation of open-DARTS via pip command without having to compile the entire project. Since the open-DARTS is used in education, where Jupyter Notebooks are widely used, it is also important to check them. We found "nbconvert" Python module helpful to run Notebooks within a Python script.

Conclusion

In this paper, we described an open source framework for a fully coupled modeling of thermo-hydro-mechanical-compositional processes in the subsurface. We demonstrate the efficiency of our open-

DARTS reservoir simulation framework in addressing modern energy transition subsurface applications with highly nonlinear physics. A generic Python interface makes the framework flexible and extendable while operator-based linearization guarantees the performance. The advanced thermo-poroelastic formulation allows considering complex applications from the energy transition portfolio. We also present a powerful pre-processing module to address discrete fractured networks in the THMC simulation. Inverse modeling capabilities, such as adjoint gradients, complement the open-DARTS framework with powerful data assimilation capabilities.

The open-DARTS was extensively benchmarked against several analytic models and other research and commercial simulators. This study provided additional benchmarks with grids up to 20M cells showing linear scalability of computations within a single computational device. OpenMP parallelization provides additional capabilities for scaling computations using multi-core CPUs to reduce the computational time. The ongoing work includes proper benchmarking of the GPU version of open-DARTS.

Acknowledgements

The authors would like to acknowledge the Netherlands eScience Center for the funding provided under grant number NLESC.OEC.2021.026.

References

Ahusborde, E., Amaziane, B., de Hoop, S., El Ossmani, M., Flauraud, E., Hamon, F.P., Kern, M., Socié, A., Su, D., Mayer, K.U., Tóth, M. and Voskov, D. [2024] A benchmark study on reactive two-phase flow in porous media: Part II - results and discussion. *Computational Geosciences*.

Bai, B. [2005] One-dimensional thermal consolidation characteristics of geotechnical media under non-isothermal condition. *Gongcheng Lixue/Engineering Mechanics*, **22**, 186–191.

Ballard, A.L. and Sloan, Jr., E.D. [2002] The next generation of hydrate prediction I. Hydrate standard states and incorporation of spectroscopy. *Fluid Phase Equilibria*, **194-197**, 371–383.

Battistelli, A., Calore, C. and Pruess, K. [1997] The simulator TOUGH2/EWASG for modelling geothermal reservoirs with brines and non-condensible gas. *Geothermics*, **26**(4), 437–464.

Collins, D., Nghiem, L., Li, Y.K. and Grabenstetter, J. [1992] An efficient approach to adaptive-implicit compositional simulation with an equation of state. *SPE Reservoir Engineering*, **7**, 259–264.

Croucher, A.E. and O'Sullivan, M.J. [2008] Application of the computer code TOUGH2 to the simulation of supercritical conditions in geothermal systems. *Geothermics*, **37**(6), 622–634.

Gao, Q. and Ghassemi, A. [2020] Three-Dimensional Thermo-Poroelastic Modeling and Analysis of Flow, Heat Transport and Deformation in Fractured Rock with Applications to a Lab-Scale Geothermal System. *Rock Mechanics and Rock Engineering*, **53**, 1–22.

de Hoop, S. [2022] *Towards the Uncertainty Quantification of Fractured Karst Systems: Reactive Transport and Fracture Networks: Where Numerical Modeling Meets Outcrop Observations*. Phd thesis, Delft University of Technology.

de Hoop, S., Voskov, D. and Bertotti, G. [2020] Studying the effects of heterogeneity on dissolution processes using operator based linearization and high-resolution lidar data. In: *ECMOR XVII*, 2020. European Association of Geoscientists & Engineers, 1–13.

de Hoop, S., Voskov, D., Bertotti, G. and Barnhoorn, A. [2022] An advanced discrete fracture methodology for fast, robust, and accurate simulation of energy production from complex fracture networks. *Water Resources Research*, **58**(5), e2021WR030743.

Hoop, S.d., Jones, E. and Voskov, D. [2021] Accurate geothermal and chemical dissolution simulation using adaptive mesh refinement on generic unstructured grids. *Advances in Water Resources*, **154**.

Jager, M.D., Ballard, A.L. and Sloan, Jr, E.D. [2003] The next generation of hydrate prediction II. Dedicated aqueous phase fugacity model for hydrate prediction. *Fluid Phase Equilibria*, **211**, 85–107.

Karimi-Fard, M., Durlofsky, L.J. and Aziz, K. [2004] An efficient discrete-fracture model applicable for general-purpose reservoir simulators. *SPE journal*, **9**(02), 227–236.

Khait, M. and Voskov, D. [2018] Operator-based linearization for efficient modeling of geothermal processes. *Geothermics*, **74**, 7 – 18.

Khait, M. and Voskov, D. [2021] A GPU-Based Integrated Simulation Framework for Modelling of Complex Subsurface Applications.

Khait, M., Voskov, D. and Zaydullin, R. [2020] High performance framework for modelling of complex subsurface flow and transport applications. All Open Access, Green Open Access.

Khait, M. and Voskov, D.V. [2017] Operator-based linearization for general purpose reservoir simulation. *Journal of Petroleum Science and Engineering*, **157**, 990 – 998.

Lyu, X., Khait, M. and Voskov, D. [2021a] Operator-based linearization approach for modeling of multiphase flow with buoyancy and capillarity. *SPE Journal*, **26**(4), 1858 – 1878.

Lyu, X. and Voskov, D. [2023] Advanced modeling of enhanced CO2 dissolution trapping in saline aquifers. *International Journal of Greenhouse Gas Control*, **127**.

Lyu, X., Voskov, D. and Rossen, W.R. [2021b] Numerical investigations of foam-assisted CO2 storage in saline aquifers. *International Journal of Greenhouse Gas Control*, **108**, 103314.

Lyu, X., Voskov, D., Tang, J. and Rossen, W.R. [2021c] Simulation of Foam Enhanced-Oil-Recovery Processes Using Operator-Based Linearization Approach. *SPE Journal*, 1–18.

Major, M., Daniilidis, A., Hansen, T.M., Khait, M. and Voskov, D. [2023] Influence of process-based, stochastic and deterministic methods for representing heterogeneity in fluvial geothermal systems. *Geothermics*, **109**.

Novikov, A. [2024] *A Finite Volume Framework for Accurate Modeling of Fault Reactivation in Poroelastic Rocks*. Dissertation (tu delft), Delft University of Technology.

Novikov, A., Behbahani, S.S., Voskov, D., Hajibeygi, H. and Jansen, J.D. [2024] Benchmarking Analytical and Numerical Simulation of Induced Fault Slip. *Under Review in Geomechanics and Geophysics for Geo-Energy and Geo-Resources*.

Novikov, A., Voskov, D., Khait, M., Hajibeygi, H. and Jansen, J.D. [2022] A scalable collocated finite volume scheme for simulation of induced fault slip. *Journal of Computational Physics*, **469**, 111598.

Peng, D.Y. and Robinson, D.B. [1976] A new two-constant equation of state. *Ind. Eng. Chem., Fundam.*, **15**(1), 59–64.

Reinhard, P. [2019] *Pressure and Temperature Interference for Geothermal Projects in Dense Production Areas A Case Study for the Delft Area*. Master's thesis, Delft University of Technology.

Soave, G. [1972] Equilibrium constants from a modified Redlich-Kwong equation of state. *Chemical Engineering Science*, **27**, 1197–1203.

Tian, X., Volkov, O. and Voskov, D. [2024] An advanced inverse modeling framework for efficient and flexible adjoint-based history matching of geothermal fields. *Geothermics*, **116**.

Tian, X. and Voskov, D. [2022] Efficient application of stochastic Discrete Well Affinity (DiWA) proxy model with adjoint gradients for production forecast. *Journal of Petroleum Science and Engineering*, **210**.

Verruijt, A. [2016] *Theory and problems of poroelasticity*. Delft University Press.

Voskov, D.V. [2017] Operator-based linearization approach for modeling of multiphase multi-component flow in porous media. *Journal of Computational Physics*, **337**, 275 – 288.

Van der Waals, J.H. and Platteeuw, J.C. [1958] Clathrate solutions. *Advances in Chemical Physics*, **2**.

Wang, Y. [2021] *The impact of heterogeneity on geothermal production: simulation benchmarks and applications*. Phd thesis, Delft University of Technology.

Wang, Y., Voskov, D., Daniilidis, A., Khait, M., Saeid, S. and Bruhn, D. [2023] Uncertainty quantification in a heterogeneous fluvial sandstone reservoir using GPU-based Monte Carlo simulation. *Geothermics*, **114**.

Wang, Y., Voskov, D., Khait, M. and Bruhn, D. [2020] An efficient numerical simulator for geothermal simulation: A benchmark study. *Applied Energy*, **264**.

Wapperom, M., Lyu, X., Nichita, D. and Voskov, D. [2023] A Unified Thermal-Reactive Compositional Simulation Framework for Modeling CO2 Sequestration at Various Scales.

Willems, C., Vondrak, A., Mijnlieff, H., Donselaar, M. and van Kempen, B. [2020] Geology of the Upper Jurassic to Lower Cretaceous geothermal aquifers in the West Netherlands Basin – an overview. *Netherlands Journal of Geosciences*, **99**, 1–13.

Wong, Z.Y. [2015] *A Geothermal Reservoir Simulator in AD-GPRS*. Phd thesis, Stanford University.

Ziabakhsh-Ganji, Z. and Kooi, H. [2012] An Equation of State for thermodynamic equilibrium of gas mixtures and brines to allow simulation of the effects of impurities in subsurface CO_2 storage. *International Journal of Greenhouse Gas Control*, **11**, S21–S34.