POROELASTICITY 1D CODE DOCUMENTATION (V.2)

MALGORZATA PESZYNSKA AND NAREN VOHRA

GitHub Link: https://github.com/nvohra0016/Biot1D-MATLAB

1. Introduction

In this document we provide the details of a one dimensional (1D) implementation of Biot's poroelasticity system [3] in MATLAB. We use a 3 field, mixed finite element scheme [6] in which the displacement is approximated in the space of piecewise linear polynomials, pressure in the space of piecewise constants, and fluxes in the lowest order Raviart-Thomas space.

1.1. Credits and Use.

This code is part of the MPower toolbox:
`http://sites.science.oregonstate.edu/~mpesz/mpower/`
The code is publicly available through GitHub: https://github.com/nvohra0016/Biot1D-MATLAB. To view the funding sources, please see the Acknowledgement section (Section 7.1) in this document. For any questions, please feel free to contact the authors via email:

Malgorzata Peszynska: `malgo.peszynska@oregonstate.edu`
Website: https://sites.science.oregonstate.edu/ mpesz/index.html
Naren Vohra: `vohran@oregonstate.edu`
Website: https://nvohra0016.github.io

2. Governing Equations

Let $\Omega = (a, b) \subset \mathbb{R}$ represent a porous medium that is fully saturated with a fluid. For any $x \in \Omega,\ t > 0$, let $u(x,t)$ denote the displacement of the medium and $p(x,t)$ denote the pressure. Then, $\forall x \in \Omega,\ t > 0$ [6]

$$-\frac{\partial}{\partial x}\left[(\lambda + 2\mu)\frac{\partial u}{\partial x}\right] + \alpha\frac{\partial p}{\partial x} = f + \overline{\rho}G, \tag{1a}$$

$$\frac{\partial \eta_l}{\partial t} + \frac{\partial q_l}{\partial x} = g, \tag{1b}$$

$$q_l = -\frac{\kappa}{\mu_l}\left[\frac{\partial p}{\partial x} - \rho_l G\frac{\partial D}{\partial x}\right], \tag{1c}$$

where η_l denotes the total fluid content, given by

$$\eta_l(x,t) = \beta_l \eta p + \alpha\frac{\partial u}{\partial x},\ \forall x \in \Omega, t > 0, \tag{2}$$

Last updated: 29 August 2023.

Variable	Description	Units
u	Displacement	[m]
p	Pressure	[Pa]
q_l	Flux	[m/s]
η_l	Total fluid content	[$-$]
D	Depth	[m]
f	Volumetric body force	[N/m^3]
g	Mass source rate	[1/s]
Parameter	**Description/Units**	**Typical value**
E	Young's modulus [Pa]	Sand (medium/fine): $12 - 20$[MPa] [4](Pg. 407) Silt: $2 - 20$[MPa] [4](Pg. 407) Clay (medium/firm): $15 - 50$[MPa] [4](Pg. 406)
ν	Poisson's ratio [$-$]	Sand (medium/ fine): 0.25 [4](Pg. 407) Silt : $0.30 - 0.35$ [4](Pg. 407) Clay(medium/firm): 0.30 [4](Pg. 406)
λ μ	Lamé parameter [Pa]; $\dfrac{E\nu}{(1+\nu)(1-2\nu)}$ Lamé parameter [Pa]; $\dfrac{E}{2(1+\nu)}$	
ρ_l	Fluid density [kg/m^3]	998.21[kg/m^3] (at 20° C) [5]
ρ_r	Solid particles density [kg/m^3]	Sand: 2650[kg/m^3] [1](Pg. 22) Clay: 2700[kg/m^3] [1](Pg. 22)
η	Porosity [$-$]	Sand(medium/fine): $0.30 - 0.35$ [2](Pg. 74) Silt: $0.4 - 0.5$ [2](Pg. 74) Clay: $0.45 - 0.55$ [2](Pg. 74)
$\overline{\rho}$	Average density of media [kg/m^3]; $\rho_l\eta + \rho_r(1-\eta)$	$-$[kg/m^3]
β_l	Fluid compressibility [1/Pa]; $\dfrac{1}{\rho_l}\dfrac{\partial \rho_l}{\partial p}$	4.58×10^{-10} [5]
κ	Permeability [m^2]	Sand: $10^{-13} - 10^{-11}$[m^2] (saturated) [4](Pg. 373) Silt: $10^{-15} - 10^{-13}$[m^2] (saturated) [4](Pg. 373) Clay: $10^{-18} - 10^{-15}$[m^2] (saturated) [4](Pg. 373)
μ_l	Fluid viscosity [Pa s]	1.0005×10^{-3} (at 20°C) [5]
G	Acceleration due to gravity [m/s^2]	9.8218 (at Anchorage, Alaska) [5]

TABLE 1. Variables and parameters used throughout the article.

q_l is the flux, and D is the depth. The other variables and physical parameters used in (1) are described in Table 1. Mechanical gravitational effects are in included in (1a) using the term $\overline{\rho}G$ and the hydrological gravitational effects are included in Darcy's law (1c) using $\rho_l G \dfrac{\partial D}{\partial x}$. Here, the depth D is a linear function which depends on the physical scenario. For example, if $\Omega = (a, b)$ represents a vertical column of soil with $x = a$ being the top and $x = b$ being the bottom, then

$$D(x) = x - a, \ \forall x \in \Omega. \tag{3}$$

If $x = a$ is the bottom then

$$D(x) = b - x, \ \forall x \in \Omega. \tag{4}$$

For a horizontal domain we have $D = 0$.

2.1. **Boundary and initial conditions.** The two distinct sets of boundary conditions correspond to mechanical deformation [M] and hydrological flow [H]. Mixed boundary conditions corresponding to [M] are

$$
\begin{aligned}
u &= u_D, \quad \text{on } \Gamma_{MD}, && (5a)\\
\tilde{\sigma} n &= \sigma_N, \quad \text{on } \Gamma_{MN}, && (5b)
\end{aligned}
$$

where $\Gamma_{MD}, \Gamma_{MN} \subset \{a, b\}$, $\Gamma_{MD} \cup \Gamma_{MN} = \{a, b\}$, and $\Gamma_{MD} \neq \emptyset$, and $n \in \{-1, 1\}$ is the outward unit normal to Γ_{MN}. The boundary conditions corresponding to [H] are

$$
\begin{aligned}
p &= p_D, \quad \text{on } \Gamma_{HD}, && (6a)\\
q_l n &= q_N, \quad \text{on } \Gamma_{HN}, && (6b)
\end{aligned}
$$

where $\Gamma_{HD}, \Gamma_{HN} \subset \{a, b\}$.

The initial condition is given on the fluid content η_l as

$$
\eta_l(x, 0) = \eta_{l\,init}(x) \; \forall x \in (a, b), \tag{7}
$$

although in practise it is sometimes computed using initial conditions on p and $\frac{\partial u}{\partial x}$.

We test multiple examples with different combinations of the Dirichlet (D) and Neumann (N) boundary conditions. In our notation, we specify the deformation boundary conditions followed by the flow boundary conditions: [MMHH]. An example of our notation is as follows: a case with Dirichlet boundary conditions for [M] and mixed boundary conditions for [H], with $\Gamma_{HD} = \{a\}$ and $\Gamma_{HN} = \{b\}$ is denoted as [DDDN].

3. Discretization

Consider the discretization of $\Omega = (a, b)$ into M cells using a grid $\mathscr{T}_h = (\omega_j)_j$ so that $\Omega = \cup_{j=1}^{M} \omega_j$, where the cell $\omega_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ has center x_j, $1 \leq j \leq M$ and size $|\omega_j| = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}} = h_j$. Consider a uniform time step $\tau > 0$ so that the n^{th} time step is given by $t_n = n\tau$.

3.1. Approximation spaces.

Here we describe the finite dimensional function spaces built on $\mathscr{T}_h$ in which we seek an approximate solution to (1). Let

$$
V_h = \{\phi_h \in C^0(\Omega) \mid \phi_h|_{\omega_j} = a_j x + b_j, \; a_j, b_j \in \mathbb{R}, \; 1 \leq j \leq M\}, \tag{8a}
$$

be the space of continuous, piecewise linear functions on Ω and

$$
V_{h,0} \subset V_h, \; V_{h,0} = \{\phi_h \in V_h \mid \phi_h|_{\Gamma_{MD}} = 0\}, \tag{8b}
$$

be the subspace of V_h containing functions which vanish on Γ_{MD}. Let

$$
M_h = \{\eta_h \mid \eta_h|_{\omega_j} = const, \; \forall \; 1 \leq j \leq M\}, \tag{8c}
$$

be the space of piecewise constants on Ω and

$$
X_h = \{\psi_h \in C^0(\Omega) \mid \psi_h|_{\omega_j} = a_j x + b_j, a_j, b_j \in \mathbb{R}, \; 1 \leq j \leq M\}, \tag{8d}
$$

be the lowest order Raviart-Thomas space of functions with

$$
X_{h,0} \subset X_h, \; X_{h,0} = \{\psi_h \in X_h \mid \psi_h|_{\Gamma_{HN}} = 0\}, \tag{9}
$$

its subspace consisting of functions which vanish on Γ_{HN}. The basis functions of these spaces are described in Section 7.

3.2. **Discrete problem.** Let $\tilde{u_D}^n \in V_h$, be such that

$$\tilde{u_D}^n(x_{i-\frac{1}{2}}) = u_D(x_{i-\frac{1}{2}}, t_n),\ \forall x_{i-\frac{1}{2}} \in \Gamma_{MD},\ n \geq 1, \tag{10a}$$

and $\tilde{q_N}^n \in X_h$ be such that

$$\tilde{q_N}^n(x_{i-\frac{1}{2}}) = q_N(x_{i-\frac{1}{2}}, t_n),\ \forall x_{i-\frac{1}{2}} \in \Gamma_{HN},\ n \geq 1, \tag{10b}$$

whenever $|\Gamma_{HN}| > 0$. The three field discrete problem corresponding to (1) is given by: find $u_h^n \in \tilde{u_D}^n + V_{h,0},\ p_h^n \in M_h,\ q_{l_h}^n \in \tilde{q_N}^n + X_{h,0}$ such that

$$\left((\lambda + 2\mu)\frac{\partial u_h^n}{\partial x}, \frac{\partial \phi_h}{\partial x}\right) - \alpha\left(p_h^n, \frac{\partial \phi_h}{\partial x}\right) = \left(f(\cdot, t_n) + \overline{\rho} G \frac{\partial D}{\partial x}, \phi_h\right) + [\sigma_N \phi_h]\big|_{\Gamma_{MN}}, \tag{11a}$$
$$\forall \phi_h \in V_{h,0},$$

$$(\beta_l \eta p_h^n, \eta_h) + \alpha\left(\frac{\partial u_h^n}{\partial x},\ \eta_h\right) + \tau\left(\frac{\partial q_{l_h}^n}{\partial x}, \eta_h\right) = \frac{\tau}{\rho_l}(g(\cdot, t_n), \eta_h) + (\eta_{l_h}^{n-1}, \eta_h), \tag{11b}$$
$$\forall \eta_h \in M_h,$$

$$\left(\left(\frac{\kappa}{\mu_l}\right)^{-1} q_{l_h}^n, \psi_h\right) - \left(p_h^n, \frac{\partial \psi_h}{\partial x}\right) = -\left[p_D(\cdot, t_n)\psi_h n\right]\big|_{\Gamma_{HN}} + \rho_l G\left(\frac{\partial D}{\partial x}, \psi_h\right), \tag{11c}$$
$$\forall \psi_h \in X_{h,0}.$$

Let $U^n, P^n, {Q_l}^n$ the degrees of freedom of $u_h^n, p_h^n, q_{l_h}^n$ in their bases. Then (11) can be rewritten in the matrix form

$$\begin{bmatrix} A_{uu} & -\alpha A_{pu} & 0 \\ \alpha A_{pu}^T & M_{pp} & \tau A_{q_l p} \\ 0 & -A_{q_l p}^T & M_{q_l q_l} \end{bmatrix} \begin{bmatrix} U^n \\ P^n \\ {Q_l}^n \end{bmatrix} = \begin{bmatrix} \mathscr{F}^n \\ \mathscr{G}^n \\ \mathscr{H}^n \end{bmatrix},\ n \geq 1, \tag{12}$$

where the stiffness and mass matrices $\{A_{uu}, A_{pu}, A_{q_l p}, M_{pp}, M_{q_l q_l}\}$ and vectors $\{\mathscr{F}^n,\ \mathscr{G}, \mathscr{H}^n\}$ are described in Section 7. The linear system (12) can be further reduced to

$$\mathscr{M}\begin{bmatrix} U^n \\ P^n \end{bmatrix} = \begin{bmatrix} \mathscr{F}^n \\ -\mathscr{G}^n + \tau A_{q_l p} M_{q_l q_l}^{-1} \mathscr{H}^n \end{bmatrix}, \tag{13a}$$

$${Q_l}^n = M_{q_l q_l}^{-1}\left(\mathscr{H}^n + A_{q_l p}^T P^n\right), \tag{13b}$$

where

$$\mathscr{M} = \begin{bmatrix} A_{uu} & -\alpha A_{pu} \\ -\alpha A_{pu}^T & -M_{pp} - \tau A_{q_l p} M_{q_l q_l}^{-1} A_{q_l p}^T \end{bmatrix}. \tag{14}$$

4. Algorithm

The code executes the following steps:

(1) The spatial and temporal grid are generated using the spatial domain end points, $\{a, b\}$, and the final time T_{end} inputted by the user.
   (a) The spatial grid can be uniform or non-uniform. In the first case, the grid size is determined using the number of cells M as an input. For a non-uniform grid, the node positions $\{x_{j+\frac{1}{2}}\}_j$ are inputted.
   (b) The temporal grid is uniform, and is generated from the user inputted number time step τ.

(2) The physical parameters, boundary conditions, and the initial condition is specified by the user.
(3) The code computes the matrix $\mathscr{M}$ given by (14) and vectors $\{\mathscr{F}^n, \mathscr{G}^n, \mathscr{H}^n\}$ in (13).
(4) At each $n = 1, 2 \ldots, N$, $\{U^n, P^n\}$ are obtained as

$$\begin{bmatrix} U^n \\ P^n \end{bmatrix} = \mathscr{M}^{-1} \begin{bmatrix} \mathscr{F}^n \\ -\mathscr{G}^n + \tau A_{q_l p} M_{q_l q_l}^{-1} \mathscr{H}^n \end{bmatrix}. \tag{15}$$

Then, q_l^n is obtained from (13b).

4.1. **MATLAB implementation.** The code comprises of two files: Biot1D.m and Biot_data.m. The latter consists of the physical parameters of the medium, and the former contains the code.

The code is run using 7 input parameters:

(1) `Tend`: This is a positive constant that determines the time period $(0, T_{end})$.
(2) `nx`: This may be a scalar or a vector. If scalar, then it is the number of cells M and a uniform spatial grid is created with M cells. If it is a vector, then it contains the element nodes $\{x_{j+\frac{1}{2}}\}_{j=0}^{M}$ (nodal grid), starting from $x_{\frac{1}{2}} = a$ and including $x_{M+\frac{1}{2}} = b$.
(3) `dt`: This is positive constant that denotes the time step τ. A uniform temporal grid.
(4) `bdaryflags`: This is a 4×1 vector with each entry either 0 or 1. The entry 0 corresponds to Dirichlet boundary condition and 1 corresponds to Neumann boundary condition. The order is determined as [MMHH], i.e., the first two entries correspond to mechanical deformation at $x = a$ and $x = b$, and the last two entries are for hydrological flow (pressure or flux) at $x = a$ and $x = b$; see Section 2.1.
(5) `caseflag`: This is an integer between 0 and 6 which toggles between particular inbuilt examples or a custom scenario: `caseflag` $= 1, 2, 3, 4$ or 5 corresponds to the hard-coded examples demonsrated in Section 5 in this document. `caseflag` $= 6$ corresponds to a custom scenario that needs to be set up by the user.
(6) `ifsave`: This is an integer that toggles the saving and debugging features. If `ifsave` > 0, the output is saved to a file at the end of the simulation. If `ifsave` $= 0$, then save file is generated. If `ifsave` $= -1$, then matrix $\mathscr{M}$ and the boundary conditions are outputted for debugging purposes. If `ifsave` $= -2$, then error is computed based on the known analytical expression.
(7) `ifplot`: This is an integer that plots the solution every $n\times$ `ifplot`, $n \in \mathbb{Z}^+$, time steps.

The code returns the following output:

(1) `xfem`: The nodal grid $\{x_{j+\frac{1}{2}}\}_{j=0}^{M}$.
(2) `usol`: The nodal displacement profile at the final time step.
(3) `xplot`: The cell centered spatial grid $\{x_j\}_{j=1}^{M}$.
(4) `psol`: The cell centered pressure profile at the final time step.

The solution is plotted at the final time step.

Example simulation runs:

```
>> [xu, u, xp, p] = Biot1D(1, 10, 0.1, [0 0 0 0], 1, -2, 1);
>> [xu, u, xp, p] = Biot1D(86400, 20, 3600, [1 0 0 1], 4, 0, 1);
```

For a complete demonstration of the in-built examples, see Section 5.

The physical parameters are specified in the file Biot_data.m. The parameters are as listed below.

(1) `COF_c0`: Specific storage coefficient $c_0 = \eta\beta_l$.
(2) `COF_lambda` and `COF_mu`: Elasticity parameters λ and μ.
(3) `COF_alpha`: Biot-Willis coefficient α.
(4) `COF_kappa`: Permeability and viscosity parameter $\frac{\kappa}{\mu_l}$.
(5) `COF_rhol` and `COF_rhor`: Density parameters ρ_l and ρ_r.
(6) `COF_G`: Acceleration due to gravity G.

Initial and boundary conditions. The initial conditions are specified using initial fluid content function `fluid_init`.

The boundary condition flags specified by the user as the input parameters `bdaryflags`. The boundary values are specified in one of two ways. (i) First, if the exact solution is known, then the boundary conditions are calculated using the expression of exact solution. (ii) If the exact solution is not known (such as in physical scenarios), then the boundary conditions are manually coded by the user in the time loop.

5. Numerical Examples

5.1. **Manufactured solution.**

Example 5.1. *Consider* $\Omega = (0,1)[\mathrm{m}]$

$$u(x,t) = -\frac{1}{\pi}\cos(\pi x)\sin\left(\frac{\pi t}{2}\right), \tag{16a}$$

$$p(x,t) = \sin(\pi x)\sin\left(\frac{\pi t}{2}\right), \ \forall x \in (0,1), \ t > 0. \tag{16b}$$

The source terms are

$$f(x,t) = [-(\lambda+2\mu)\pi + \alpha\pi]\cos(\pi x)\sin\left(\frac{\pi t}{2}\right), \tag{17a}$$

$$g(x,t) = \frac{\pi}{2}(\beta_l\eta + \alpha)\sin(\pi x)\cos\left(\frac{\pi t}{2}\right) + \frac{\kappa}{\mu}\pi^2\sin(\pi x)\sin\left(\frac{\pi t}{2}\right). \tag{17b}$$

The physical parameters used in this example are tabulated in Table 2. We neglect gravitational effects.

The simulation is run over $(0,1)[\mathrm{s}]$ using $M = 10$ uniform cells, time step $\tau = 0.1[\mathrm{s}]$, and all Dirichlet boundary conditions [DDDD] using the following command:

```
>> [xu,u,xp,p]=Biot1D(1,10,0.1,[0,0,0,0],1,-2,1);
```

The displacement and pressure profile at the final time step is shown in Figure 1.

Example 5.2. *Consider* $\Omega = (0,1)[\mathrm{m}]$ *with*

$$u(x,t) = \sin\left(\frac{\pi x}{2}\right)e^{-t}, \tag{18a}$$

$$p(x,t) = \cos\left(\frac{\pi x}{2}\right)e^{-t}, \ \forall x \in (0,1), \ t > 0. \tag{18b}$$

Parameter	*Value*
λ	1
μ	1
η	1
β_l	1
κ	1
μ_l	1
ρ_l	1
G	0

TABLE 2. Parameters used in Example 5.1, Example 5.2, and Example 5.3. Units as in Table 1.

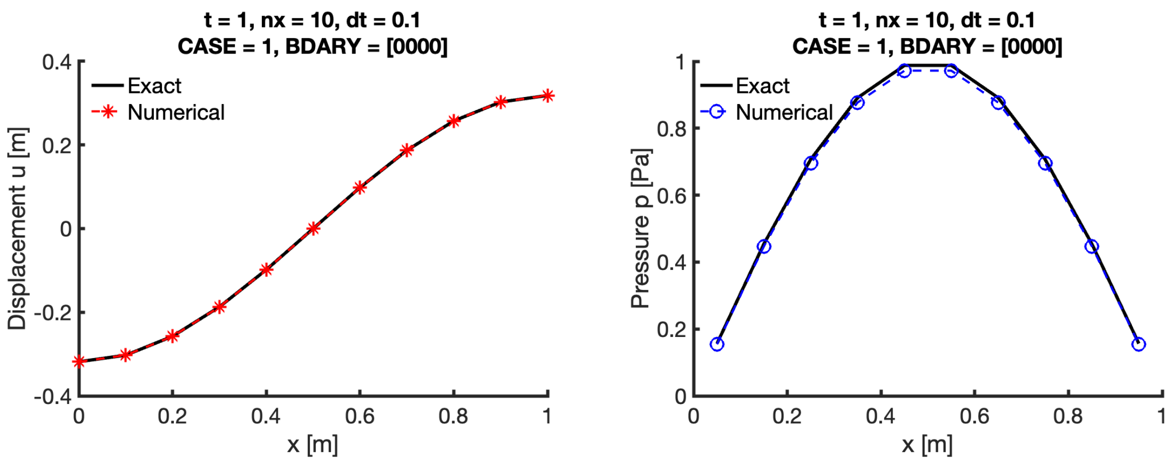

FIGURE 1. Results from Example 5.1 showing the displacement (left) and pressure (right) profile at the final time step.

The source terms are

$$f(x,t) = \left[(\lambda + 2\mu)\frac{\pi^2}{4} - \alpha\frac{\pi}{2}\right] \sin\left(\frac{\pi x}{2}\right) e^{-t}, \tag{19a}$$

$$g(x,t) = \left[-\beta_l \eta - \alpha\frac{\pi}{2} + \frac{\kappa}{\mu_l}\frac{\pi^2}{4}\right] \cos\left(\frac{\pi x}{2}\right) e^{-t}. \tag{19b}$$

We neglect gravitational effects. The physical parameters used in this example are tabulated in Table 2.

The simulation is run over $(0,1)$[s] using a non-uniform grid with $M = 10$ cells, time step $\tau = 0.1$[s], and mixed boundary conditions [NDND] using the following commands:

```
>> nx = [0; 0.05; 0.1; 0.15; 0.2; 0.4; 0.6; 0.8; 0.9; 0.95; 1.0];
>> [xu,u,xp,p]=Biot1D(1,nx,0.1,[1,0,1,0],2,-2,1);
```

The displacement and pressure profile at the final time step is shown in Figure 2.

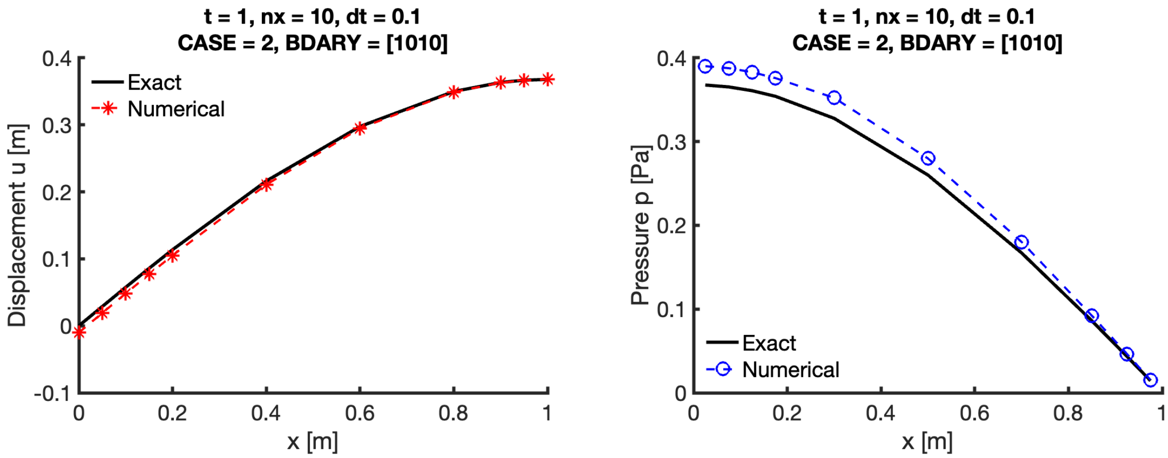

FIGURE 2. Results from Example 5.2 showing the displacement (left) and pressure (right) profile at the final time step.

Example 5.3. *Consider* $\Omega = (0, 1)[\mathrm{m}]$ *with*

$$u(x,t) = 2 - x, \tag{20a}$$
$$p(x,t) = 1 + x, \ \forall x \in (0,1), \ t > 0. \tag{20b}$$

The source terms are

$$f(x,t) = \alpha, \tag{21a}$$
$$g(x,t) = 0. \tag{21b}$$

The physical parameters used in this example are tabulated in Table 2. We neglect gravitational effects.

The simulation is run over $(0, 1)[\mathrm{s}]$ using a uniform grid with $M = 20$ cells (corresponding to $h = 0.05[\mathrm{m}]$), time step $\tau = 0.1[\mathrm{s}]$, and mixed boundary conditions [DNNN] using the following commands:

```
>> [xu,u,xp,p]=Biot1D(1,20,0.1,[0,1,1,1],3,-2,1);
```

The displacement and pressure profile at the final time step is shown in Figure 3.

5.2. **Physical examples.**

Example 5.4. Consolidation test for clay: *This example is inspired from the standard consolidation tests performed at the laboratory scale [4] (Pg. 185). Consider* $\Omega = (0, 0.1)[\mathrm{m}]$, *with* $x = 0[\mathrm{m}]$ *being the top of the soil column. The source terms are*

$$f(x,t) = 0, \tag{22a}$$
$$g(x,t) = 0. \tag{22b}$$

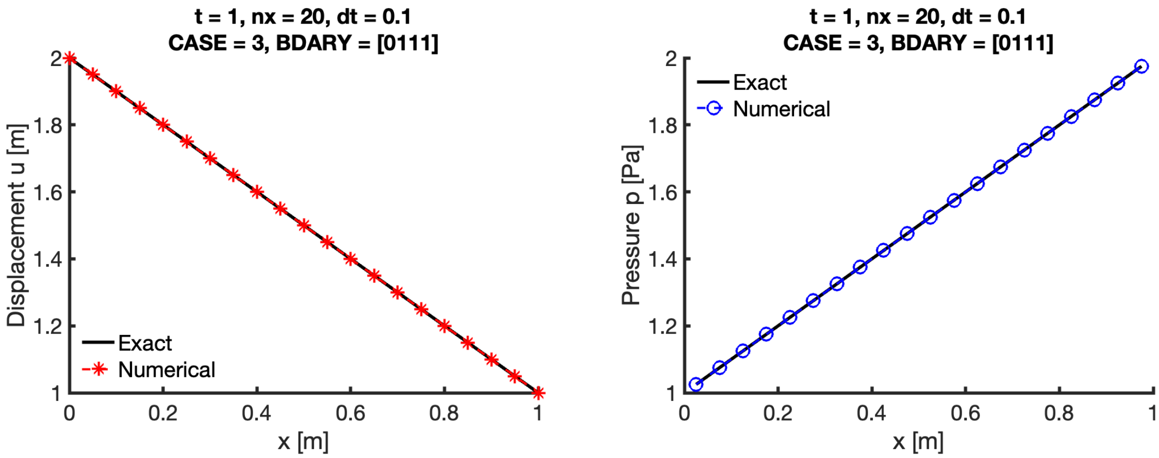

FIGURE 3. Results from Example 5.3 showing the displacement (left) and pressure (right) profile at the final time step.

Parameter	Value	Reference
E	20×10^6	[4](Pg. 406)
ν	0.30	[4](Pg. 406)
η	0.50	[2](Pg. 74)
β_l	4.58×10^{-10}	[5]
κ	10^{-17}	[4](Pg. 373)
μ_l	1.0005×10^{-3}	[5]
ρ_l	998.21	[5]
ρ_r	2700	[1](Pg. 22)
G	9.8218	[5]

TABLE 3. Physical parameters for clay used in Example 5.4. Units as in Table 1.

We consider the mixed boundary conditions

$$\begin{aligned} \tilde{\sigma} n &= 10^5, \ x = 0, && (23a) \\ u &= 0, \ x = 0.1, && (23b) \\ p &= 0, \ x = 0, && (23c) \\ q_l n &= 0, \ x = 0.1. && (23d) \end{aligned}$$

We do not neglect gravitational effects. The physical parameters used in this example are tabulated in Table 3.

The simulation is run over $(0, 24)$[hr] ($= (0, 86400)$ [s]) using a uniform grid with $M = 20$ cells (corresponding to $h = 0.005$[m]), and time step $\tau = 1$[hr] ($= 3600$ [s]). We use mixed boundary conditions [NDDN] using the following commands:

```
>> [xu,u,xp,p]=Biot1D(86400,20,3600,[1,0,0,1],4,0,1);
```

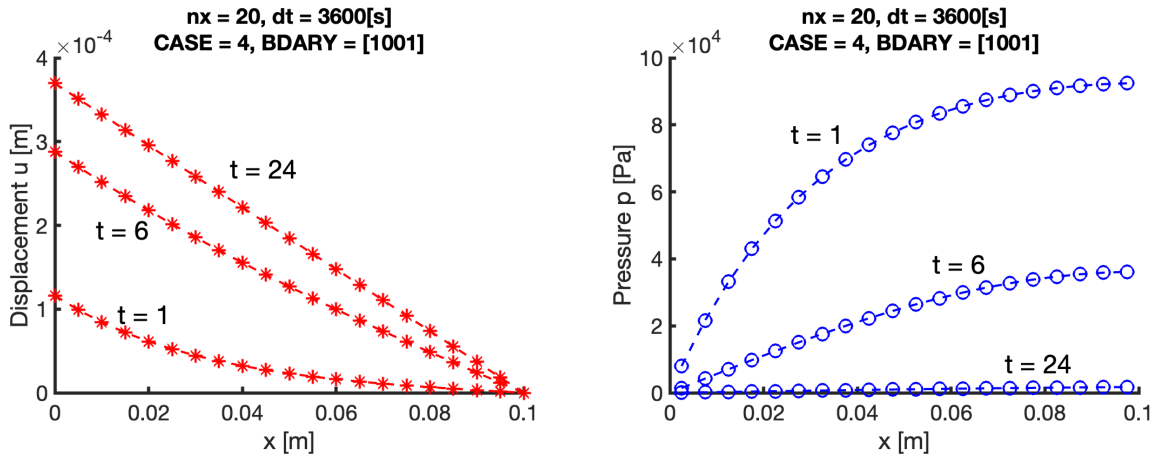

FIGURE 4. Results from Example 5.4 showing the evolution of the displacement (left) and pressure (right) profiles at $t = 1$, $t = 6$, and $t = 24$[hr].

The displacement and pressure profiles at $t = 1.2$, $t = 10.8$ and $t = 24$[hr] are shown in Figure 4. The total settlement $s(t) = u(0, t)$ over the simulation period $t \in (0, 24)$[hr] is also shown.
Comments:The smooth evolution of the displacement and pressure profiles can be observed in Figure 4. We get a settlement of 3.6963×10^{-4}[m] at the end of the simulation.

Example 5.5. Consolidation test for sand and clay: *This example is to demonstrate the robustness of the solver in the presence of heterogeneity. Consider* $\Omega = (0, 1)$[m]*, with* $x = 0$[m] *being the top of the soil column. The source terms are*

$$f(x,t) = 0, \tag{24a}$$
$$g(x,t) = 0. \tag{24b}$$

We consider the mixed boundary conditions

$$\tilde{\sigma} n = 10^5, \; x = 0, \tag{25a}$$
$$u = 0, \; x = 1, \tag{25b}$$
$$p = 0, \; x = 0, \tag{25c}$$
$$q_l n = 0, \; x = 1. \tag{25d}$$

We do not neglect gravitational effects. To introduce heterogeneity, we consider $\Omega = \Omega_{sand} \cup \Omega_{clay}$ *to be occupied by sand in* $\Omega_{sand} = (0, 0.5)$[m] *and clay in* $\Omega_{clay} = (0.5, 1)$[m]*. The physical parameters used in this example are tabulated in Table 4.*

The simulation is run over $(0, 365)$[day] (i.e., 1[year] = 31536000 [s]) using a uniform grid with $M = 20$ cells (corresponding to $h = 0.05$[m]), $\tau = 24$ [hr] (= 86400 [s]), and mixed boundary conditions [NDDN] using the following commands:

```
>>  [xu,u,xp,p]=Biot1D(31536000,20,86400,[1,0,0,1],5,0,1);
```

The displacement and pressure profiles at the end of the simulation are shown in Figure 5.
Comments: No spurious pressure oscillations were observed throughout the simulation. At the end of the simulation we get a settlement of 4.6347×10^{-3} [m].

Parameter	Value	Reference
E	Clay: 20×10^6 Sand: 15×10^6	[4](Pg. 406) [4](Pg. 407)
ν	Clay: 0.30 Sand: 0.25	[4](Pg. 406) [4](Pg. 407)
η	Clay: 0.50 Sand: 0.30	[2](Pg. 74) [2](Pg. 74)
β_l	4.58×10^{-10}	[5]
κ	Clay: 1×10^{-17} Sand: 1×10^{-12}	[4](Pg. 373) [4](Pg. 373)
μ_l	1.0005×10^{-3}	[5]
ρ_l	998.21	[5]
ρ_r	Clay: 2700 Sand: 2650	[1](Pg. 22) [1](Pg. 22)
G	9.8218	[5]

Table 4. Physical parameters for clay and sand used in Example 5.5. Units as in Table 1.

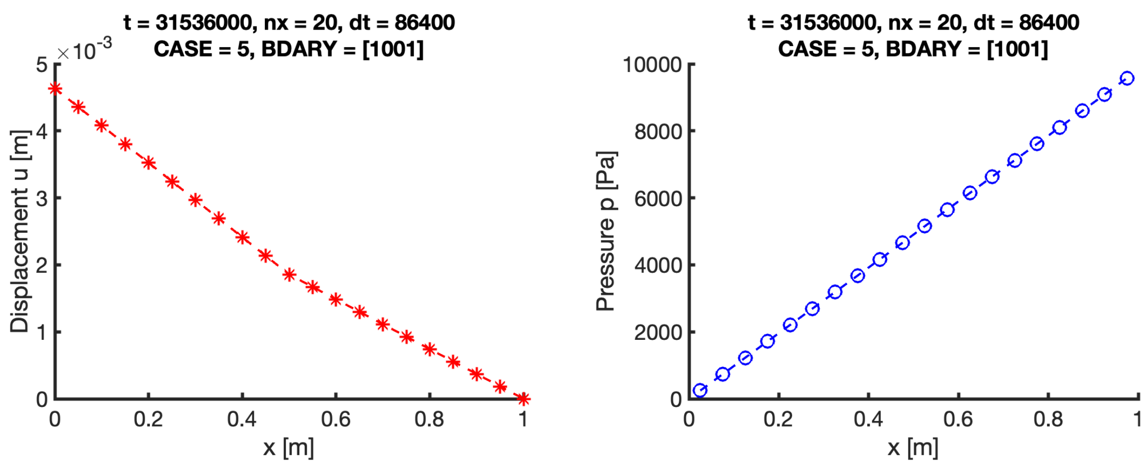

Figure 5. Results from Example 5.5 showing the displacement (left) and pressure (right) profile at $t = 365$[day].

6. Acknowledgement

This research was partially supported by NSF DMS-1912938 "Modeling with Constraints and Phase Transitions in Porous Media" and NSF DMS-1522734 "Phase transitions in porous media across multiple scales"; PI: Dr. Malgorzata Peszynska.

7. Appendix

Here we give the details of the matrices and vectors computed in Section 3.

7.1. **Approximation spaces bases functions.** We define the basis functions $\{\phi_{j+\frac{1}{2}}\}_{j=0}^{M}$, $\{\eta_j\}_{j=1}^{M}$, and $\{\psi_{j+\frac{1}{2}}\}_{j=0}^{M}$ for V_h, M_h, and X_h, respectively, defined in 3.1 as follows: $\forall x \in \Omega$

$$\phi_{j+\frac{1}{2}}(x) = \begin{cases} \frac{1}{h_j}\left(x - x_{j-\frac{1}{2}}\right); & x \in \omega_j \\ \frac{1}{h_{j+1}}\left(x_{j+\frac{3}{2}} - x\right); & x \in \omega_{j+1} \\ 0; & \text{otherwise} \end{cases}, \ \forall\, 1 \le j \le M-1, \tag{26}$$

$$\phi_{\frac{1}{2}}(x) = \begin{cases} \frac{1}{h_1}\left(x_{\frac{3}{2}} - x\right); & x \in \omega_1 \\ 0; & \text{otherwise} \end{cases}, \tag{27}$$

$$\phi_{M+\frac{1}{2}}(x) = \begin{cases} \frac{1}{h_M}\left(x - x_{M-\frac{1}{2}}\right); & x \in \omega_M \\ 0; & \text{otherwise} \end{cases}, \tag{28}$$

$$\eta_j(x) = \begin{cases} 1; & x \in \omega_j \\ 0; & \text{otherwise} \end{cases}, \ \forall\, 1 \le j \le M, \tag{29}$$

$$\psi_{j+\frac{1}{2}} = \phi_{j+\frac{1}{2}}, \ \forall\, 0 \le j \le M, \tag{30}$$

where (30) holds since $H_{div}(I) = H^1(I), \ \forall I \subset \mathbb{R}$.

7.2. **Implementation example using mixed boundary conditions [NDDN].** Consider (1) with the following mixed boundary conditions

$$\tilde{\sigma} n = \sigma_N, \ x = a, \tag{31a}$$
$$u = u_D, \ x = b, \tag{31b}$$
$$p = p_D, \ x = a, \tag{31c}$$
$$q_l n = q_N, \ x = b. \tag{31d}$$

We denote by $U_j^n = u_h^n(x_{j-\frac{1}{2}}, t_n)$, $P_j^n = p_h^n(x_j, t_n)$, and ${Q_l}_j^n = {q_l}_h^n(x_{j-\frac{1}{2}}, t_n)$. Then we can rewrite

$$u_h^n = \sum_{j=1}^{M} U_j^n \phi_{j-\frac{1}{2}} + u_D(b, t_n)\phi_{M+\frac{1}{2}}, \tag{32a}$$

$$p_h^n = \sum_{j=1}^{M} P_j^n \eta_j, \tag{32b}$$

$$ {q_l}_h^n = \sum_{j=1}^{M} {Q_l}_j^n \psi_{j-\frac{1}{2}} + q_N(b, t_n)\psi_{M+\frac{1}{2}}, \tag{32c}$$

Further denote by ${\sigma_N}_{\frac{1}{2}}^n = \sigma_N(a, t_n)$, and ${p_D}_{\frac{1}{2}}^n = p_D(a, t_n)$. The discrete system (11) can be rewritten as

$$\begin{bmatrix} A_{uu} & -\alpha A_{pu} & 0 \\ \alpha A_{pu}^T & M_{pp} & \tau A_{q_l p} \\ 0 & -A_{q_l p}^T & M_{q_l q_l} \end{bmatrix} \begin{bmatrix} U^n \\ P^n \\ {Q_l}^n \end{bmatrix} = \begin{bmatrix} \mathscr{F}^n \\ \mathscr{G}^n \\ \mathscr{H}^n \end{bmatrix}, \ n \ge 1, \tag{33}$$

where the stiffness and mass matrices are

$$
\begin{aligned}
A_{uu} &= \left[\left((\lambda+2\mu)\frac{d\phi_{j-\frac{1}{2}}}{dx}, \frac{d\phi_{i-\frac{1}{2}}}{dx}\right)\right]_{1\le i,j\le M}, && (34a)\\
A_{pu} &= \left[\left(\eta_j, \frac{d\phi_{i-\frac{1}{2}}}{dx}\right)\right]_{1\le i,j\le M}, && (34b)\\
M_{pp} &= [(\eta_j,\eta_i)]_{1\le i,j\le M}, && (34c)\\
A_{q_l p} &= \left[\left(\frac{d\psi_{j-\frac{1}{2}}}{dx}, \eta_i\right)\right]_{1\le i,j\le M}, && (34d)\\
M_{q_l q_l} &= \left[\left(\left(\frac{\kappa}{\mu_l}\right)^{-1}\psi_{j-\frac{1}{2}}, \psi_{i-\frac{1}{2}}\right)_T\right]_{1\le i,j\le M}. && (34e)
\end{aligned}
$$

The subscript T in (34e) is used to denote the use of the trapezoidal rule to evaluate the integral. This reduces $M_{q_l q_l}$ to a diagonal matrix instead of a tri-diagonal system if full integration were used.

The terms on the right hand side in (33), $\mathscr{F}^n, \mathscr{G}^n$, and $\mathscr{H}^n \in \mathbb{R}^M$ are given by $\forall 1 \le i \le M,\ n \ge 1$,

$$
\begin{aligned}
\mathscr{F}_i^n &= \left(\left(f(\cdot,t_n) + \overline{\rho} G \frac{\partial D}{\partial x}\right), \phi_{i-\frac{1}{2}}\right) \\
&+ \begin{cases} \sigma_N(a,t_n) & i=1 \\ 0; & 1<i<M \\ (\lambda(x_M)+2\mu(x_M))\left(\dfrac{u_D(b,t_n)}{h_M}\right); & i=M \end{cases}, && (35a)\\
\mathscr{G}_i^n &= h_i \eta_{l_i}^{n-1}\eta_i - \begin{cases} 0; & 1<i\le M \\ \alpha u_D(b,t_n) + \tau q_N(b,t_n); & i=M \end{cases}, && (35b)\\
\mathscr{H}_i^n &= \left(\rho_l G \frac{\partial D}{\partial x}, \psi_{i-\frac{1}{2}}\right) + \begin{cases} p_D(a,t_n); & i=1 \\ 0; & 1<i\le M \end{cases}, && (35c)
\end{aligned}
$$

where $\forall n \ge 1$,

$$
\begin{aligned}
\eta_{l_i}^n &= \beta_l \eta(x_i) p_h^n(x_i) + \alpha \frac{\partial u_h^n}{\partial x}(x_i) \\
&= \begin{cases} \beta_l \eta(x_i) P_i^n + \alpha \left(\dfrac{U^n_{i+\frac{1}{2}} - U^n_{i-\frac{1}{2}}}{h_j}\right); & 1 \le i < M \\ \beta_l \eta(x_i) P_i^n + \alpha \left(\dfrac{u_D(b,t_n) - U^n_{i-\frac{1}{2}}}{h_j}\right); & i=M \end{cases}. && (36)
\end{aligned}
$$

Note: In (35b), for $n=1$, $\eta_{l_i}^0 = \eta_{l_{init}}(x_i)$ is calculated using the given initial condition (7).

References

[1] Orlando B. Andersland and Branko Ladanyi. *Frozen Ground Engineering*. 2nd ed. Hoboken, NJ : [Reston, Va.]: Wiley; ASCE, 2004. ISBN: 0471615498.

[2] Jacob Bear and Alexander Cheng. *Modeling Groundwater Flow and Contaminant Transport*. Vol. 23. Jan. 2010. ISBN: 978-1-4020-6681-8. DOI: 10.1007/978-1-4020-6682-5.

[3] Maurice A. Biot. "General Theory of Three-Dimensional Consolidation". In: *Journal of Applied Physics* 12.2 (1941), pp. 155–164. DOI: 10.1063/1.1712886.

[4] Jean-Louis Briaud. *Geotechnical Engineering: Unsaturated and Saturated Soils*. Wiley, 2013. ISBN: 9780470948569.

[5] *Engineering Toolbox*. https://www.engineeringtoolbox.com. retrieved in 2023.

[6] Phillip Phillips and Mary Wheeler. "A coupling of mixed and continuous Galerkin finite element methods for poroelasticity I: The continuous-in-time case". In: *Computational Geosciences* 11 (June 2007), pp. 145–158. DOI: 10.1007/s10596-007-9044-z.