Poroelasticity Modules in DarcyLite *

Jiangguo Liu[1] and Zhuoran Wang[2]

[1] Department of Mathematics, Colorado State University, Fort Collins, CO 80523, USA, liu@math.colostate.edu
[2] School of Mathematics (Zhuhai), Sun Yat-sen University, Zhuhai, Guangdong 519082, China, wangzhr25@mail.sysu.edu.cn

Abstract. This paper elaborates on design and implementation of code modules for finite element solvers for poroelasticity in our Matlab package DarcyLite [15]. The Biot's model is adopted. Both linear and nonlinear cases are discussed. Numerical experiments are presented to demonstrate the accuracy and efficiency of these solvers.

Keywords: Biot's model · Enriched Lagrangian finite elements · poroelasticity · quadrilateral meshes · weak Galerkin finite element methods

1 Introduction

Poroelasticity problems exist widely in the real world, e.g., drug delivery, food processing, petroleum reservoirs, and tissue engineering. These problems involve fluid flow in porous media that are elastic and deform due to fluid pressure. The Biot's model for linear and nonlinear poroelasticity has been well accepted [4,7,12,20,30]. It couples solid displacement $\mathbf{u}$ and fluid pressure p through the following partial differential equations (PDEs) on a bounded domain Ω for a time period $[0, T]$:

$$\begin{cases} -\nabla \cdot (2\mu\varepsilon(\mathbf{u}) + \lambda(\nabla \cdot \mathbf{u})\mathbf{I}) + \alpha\nabla p = \mathbf{f}, \\ \partial_t (\alpha\nabla \cdot \mathbf{u} + c_0 p) + \nabla \cdot (-\mathbf{K}(\mathbf{u})\nabla p) = s, \end{cases} \tag{1}$$

where $\varepsilon(\mathbf{u}) = \frac{1}{2}\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)$ is the strain tensor, $\sigma(\mathbf{u}) = 2\mu\,\varepsilon(\mathbf{u}) + \lambda(\nabla \cdot \mathbf{u})\mathbf{I}$ the stress tensor, $\lambda > 0, \mu > 0$ the Lamé constants, $\mathbf{f}$ a given body force, $\mathbf{K}$ a conductivity/permeability tensor, s the fluid source, α (≈ 1) the Biot-Williams constant, and $c_0 \geq 0$ the constrained storage capacity. Furthermore, the total stress is defined as

$$\widetilde{\sigma}(\mathbf{u}, p) = \sigma - \alpha\, p\, \mathbf{I}. \tag{2}$$

Dirichlet and Neumann boundary conditions for solid are posed as

$$\mathbf{u}|_{\Gamma_D^{\mathcal{E}}} = \mathbf{u}_D, \qquad (\widetilde{\sigma}\mathbf{n})|_{\Gamma_N^{\mathcal{E}}} = \mathbf{t}_N, \tag{3}$$

* Liu was partially supported by US National Science Foundation grant DMS-1819252. Wang was partially supported by grant 74120-18841215 from Sun Yat-sen University.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

whereas Dirichlet and Neumann boundary conditions for fluid are posed as

$$p|_{\Gamma_D^{\mathcal{D}}} = p_D, \qquad (-\mathbf{K}\nabla p)\cdot\mathbf{n}|_{\Gamma_N^{\mathcal{D}}} = u_N, \tag{4}$$

where $\mathbf{n}$ is the outward unit normal vector to $\partial\Omega$, which has a non-overlapping decomposition $\partial\Omega = \Gamma_D^{\mathcal{E}} \cup \Gamma_N^{\mathcal{E}}$ for solid and another non-overlapping decomposition $\partial\Omega = \Gamma_D^{\mathcal{D}} \cup \Gamma_N^{\mathcal{D}}$ for fluid. As for initial conditions, we have

$$p(\mathbf{x}, 0) = p_0, \qquad \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0. \tag{5}$$

Usually, $\mathbf{u}_0 = \mathbf{0}$, i.e., there is no deformation at the beginning of the simulation.

Finite element methods (FEMs) are common tools for solving the Biot's model. Depending on the unknown quantities to be solved, poroelasticity solvers are usually grouped into 3 types:

- *2-field*: Solid displacement and fluid pressure are to be solved;
- *3-field*: Solid displacement, fluid pressure and velocity are to be solved;
- *4-field*: Solid stress & displacement, fluid pressure & velocity are to be solved.

A major issue in numerical solvers for poroelasticity is the poroelasticity locking, which usually appears as nonphysical pressure oscillations or deteriorating convergence rates in displacement errors. This happens when the porous media are low-permeable or nearly incompressible ($\lambda \to \infty$) [7,21,30].

Early on, the continuous Galerkin (CG) FEMs were applied respectively to solve for displacement and pressure. But it was soon recognized that such solvers were subject to poroelasticity locking and the 2-field approach was nearly abandoned. The mixed finite element methods can be used to solve for pressure and velocity simultaneously and meanwhile coupled with a FEM for linear elasticity that is free of Poisson-locking. Therefore, the 3-field approach has been the main stream [4,18,19,20,27,28]. The 4-field approach is certainly worth of investigation, but it may involve too many unknowns (degrees of freedom) [29].

The weak Galerkin (WG) finite element methods [23] have emerged as a new class of numerical methods with nice features that can be applied to a wide variety of problems including Darcy flow and linear elasticity [10,13]. Certainly, WG solvers can be developed for linear poroelasticity [12], they are free of poroelasticity locking but may involve more degrees of freedom. There have also been efforts on developing HDG methods for the Biot's model [6].

This paper elaborates on code modules for poroelasticity recently added to our `DarcyLite` package [15], which has gained some popularity. We shall discuss five solvers with a variety of discretization schemes for linear elasticity and Darcy flow. A poroelasticity problem may be solved as a monolithic systems (MS) or through operator splitting (OS). This paper explains the mathematical ideas behind these solvers, and their implementation with consideration of modularity and code re-usability. Numerical experiments are presented to demonstrate their use, accuracy, and efficiency.

We focus on 2-dim problems with a quasi-uniform triangular mesh $\mathcal{T}_h$ or convex quadrilateral mesh $\mathcal{E}_h$. For ease of presentation, we consider a uniform temporal partition of $[0, T]$ with $\Delta t = T/N$ and $t_n = n\Delta t$ for $0 \le n \le N$.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

2 Solver I: A 2-field Penalty-free Weak Galerkin Finite Element Solver for Quadrilateral Meshes

Now we consider linear poroelasticity (1) in which $\mathbf{K}$ actually does not depend on $\mathbf{u}$. Then the variational form reads as

$$\begin{cases} 2\mu(\varepsilon(\mathbf{u}),\varepsilon(\mathbf{v})) + \lambda(\nabla\cdot\mathbf{u},\nabla\cdot\mathbf{v}) - \alpha(p,\nabla\cdot\mathbf{v}) = (\mathbf{f},\mathbf{v}) + \langle \mathbf{t}_N,\mathbf{v}\rangle_{\Gamma_N^{\mathcal{E}}}, \\ \alpha(\partial_t\nabla\cdot\mathbf{u},q) + c_0(\partial_t p,q) + (\mathbf{K}\nabla p,\nabla q) = (s,q) - \langle u_N,q\rangle_{\Gamma_N^{\mathcal{D}}} \end{cases} \tag{6}$$

with incorporation of boundary and initial conditions.

We consider WG finite element discretization for both linear elasticity and Darcy flow. $\mathrm{WG}(P_0,P_0;AC_0)$ for Darcy flow has been investigated in [17]. Here we briefly discuss $\mathrm{WG}(P_0^2,P_0^2;AC_0^2)$ finite elements for linear elasticity.

The classical Raviart-Thomas spaces $RT_{[k]}(k\geq 0)$ for rectangles have some limitations. The recently developed Arbogast-Correa mixed finite elements are designed for more general convex quadrilaterals [3]. We shall use the lowest-order AC_0 space, which has a local basis [17] as shown below

$$\begin{bmatrix}1\\0\end{bmatrix},\quad \begin{bmatrix}0\\1\end{bmatrix},\quad \begin{bmatrix}X\\Y\end{bmatrix},\quad \mathcal{P}_E\begin{bmatrix}\hat{x}\\-\hat{y}\end{bmatrix},$$

where $X=x-x_c, Y=y-y_c$, (x_c,y_c) is the element center, $(\hat{x},\hat{y})$ are the coordinates in the reference element $[0,1]^2$, and $\mathcal{P}_E$ is the Piola transformation.

Let E be a convex quadrilateral. and $AC_0^2(E)$ be the space of order-2 matrices whose row vectors are in $AC_0(E)$. We consider a typical discrete weak function $\mathbf{v}=\{\mathbf{v}^\circ,\mathbf{v}^\partial\}\in WG(P_0^2,P_0^2)$. Its ***discrete weak gradient*** $\nabla_w\mathbf{v}$ is reconstructed in $AC_0^2(E)$ via integration by parts

$$(\nabla_w\mathbf{v},\tau) = \langle \mathbf{v}^\partial,\tau\mathbf{n}\rangle_{E^\partial} - (\mathbf{v}^\circ,\nabla\cdot\tau)_{E^\circ},\qquad \forall\tau\in AC_0^2(E). \tag{7}$$

Its ***discrete weak divergence*** $\nabla_w\cdot\mathbf{v}$ is reconstructed in $P_0(E)$ as

$$(\nabla_w\cdot\mathbf{v},\phi) = \langle \mathbf{v}^\partial,\phi\mathbf{n}\rangle_{E^\partial} - (\mathbf{v}^\circ,\nabla\phi)_{E^\circ},\qquad \forall\phi\in P_0(E). \tag{8}$$

Solver I as a time-marching 2-field finite element scheme for (6) reads as

$$\begin{cases} \mathcal{A}_h^{\mathcal{E}}(\mathbf{u}_h^{(n)},\mathbf{v}) - \mathcal{B}_h(p_h^{(n)},\mathbf{v}) = \mathcal{F}_h^{\mathcal{E}}(\mathbf{v}), \\ \mathcal{B}_h(\mathbf{u}_h^{(n)},q) + \mathcal{A}_h^{\mathcal{D}}(p_h^{(n)},q) = \mathcal{F}_h^{\mathcal{D}}(q), \end{cases} \tag{9}$$

where the bilinear forms on the left-hand sides are defined as

$$\begin{cases} \mathcal{A}_h^{\mathcal{E}}(\mathbf{u}_h^{(n)},\mathbf{v}) = \displaystyle\sum_{E\in\mathcal{E}_h} 2\mu(\varepsilon_w(\mathbf{u}_h^{(n)}),\varepsilon_w(\mathbf{v}))_E + \lambda(\nabla_w\cdot\mathbf{u}_h^{(n)},\nabla_w\cdot\mathbf{v})_E, \\ \mathcal{A}_h^{\mathcal{D}}(p_h^{(n)},q) = \displaystyle\sum_{E\in\mathcal{E}_h} c_0(p_h^{(n),\circ},q^\circ)_{E^\circ} + \Delta t(\mathbf{K}\nabla_w p_h^{(n)},\nabla_w q)_E, \\ \mathcal{B}_h(\mathbf{u}_h^{(n)},q) = \displaystyle\sum_{E\in\mathcal{E}_h} \alpha(\nabla_w\cdot\mathbf{u}_h^{(n)},q^\circ)_{E^\circ}. \end{cases} \tag{10}$$

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

The linear forms on the right-hand sides are defined as

$$\begin{cases} \mathcal{F}_h^{\mathcal{E}}(\mathbf{v}) = \sum_{E\in\mathcal{E}_h} (\mathbf{f}^{(n)}, \mathbf{v}^{\circ})_{E^\circ} + \sum_{e\in\Gamma_N^{\mathcal{E}}} \langle \mathbf{t}_N, \mathbf{v}^{\partial}\rangle_e, \\ \mathcal{F}_h^{\mathcal{D}}(q) = \sum_{E\in\mathcal{E}_h} \Delta t(s^{(n)}, q^{\circ}) + c_0(p_h^{(n-1),\circ}, q^{\circ})_{E^\circ} + \alpha(\nabla_w \cdot \mathbf{u}_h^{(n-1)}, q^{\circ})_{E^\circ} \\ \qquad - \sum_{e\in\Gamma_N^{\mathcal{D}}} \Delta t \, \langle u_N, q^{\partial}\rangle_e. \end{cases} \tag{11}$$

Unlike the methods in [12], this WG solver does not need stabilization for either elasticity or Darcy flow. The degrees of freedom (DOFs) at each time step are

$$3\,\#\text{Elements} + 3\,\#\text{Edges}.$$

3 Solver II: A 3-field CG+MFEM Solver for Triangular Meshes

In the 3-field approach, Darcy velocity $\mathbf{q}$ is used. The PDEs take the form

$$\begin{cases} -\nabla\cdot(2\mu\varepsilon(\mathbf{u}) + \lambda(\nabla\cdot\mathbf{u})\mathbf{I}) + \alpha\nabla p = \mathbf{f}, \\ \mathbf{K}^{-1}\mathbf{q} + \nabla p = \mathbf{0}, \\ \partial_t(\alpha\nabla\cdot\mathbf{u} + c_0 p) + \nabla\cdot\mathbf{q} = s. \end{cases} \tag{12}$$

We shall need spaces $\mathbf{V} = \mathbf{H}^1(\Omega)$, $\mathbf{V}^0 = \mathbf{H}_0^1(\Omega)$, $\mathbf{W} = H(\mathrm{div}, \Omega)$, $\mathbf{W}^0 = H_0(\mathrm{div}, \Omega)$, $S = L_0^2(\Omega)$. The variational problem seeks solutions $\mathbf{u} \in \mathbf{V}$, $\mathbf{q} \in \mathbf{W}$, $p \in S$ such that for any $\mathbf{v} \in \mathbf{V}^0$, $\mathbf{w} \in \mathbf{W}^0$ and $q \in S$, there holds

$$\begin{cases} 2\mu(\varepsilon(\mathbf{u}), \varepsilon(\mathbf{v})) + \lambda(\nabla\cdot\mathbf{u}, \nabla\cdot\mathbf{v}) - \alpha(p, \nabla\cdot\mathbf{v}) = (\mathbf{f}, \mathbf{v}) + \langle \mathbf{t}_N, \mathbf{v}\rangle, \\ (\mathbf{K}^{-1}\mathbf{q}, \mathbf{w}) - (p, \nabla\cdot\mathbf{w}) = -\langle p_D, \mathbf{w}\cdot\mathbf{n}\rangle, \\ \alpha(\nabla\cdot(\partial_t\mathbf{u}), q) + (\nabla\cdot\mathbf{q}, q) + c_0(\partial_t p, q) = (s, q). \end{cases} \tag{13}$$

Again initial conditions are omitted for ease of presentation.

As presented in [30], one considers a triangular mesh $\mathcal{T}_h$ for spatial discretization and the implicit Euler for temporal discretization. One utilizes the 1st order Bernardi-Raugel element space $\mathbf{V}_h$ for displacement discretization. The mixed FE pair (RT_0, P_0) is used for discretization of Darcy flow. The velocity/flux FE spaces are denoted as $\mathbf{W}_h$ and $\mathbf{W}_h^0$, for which the edge-based basis functions are used [2,14]. This is especially convenient for handling the Neumann boundary conditions.

Let $\mathbf{u}_h^{(n)}, \mathbf{u}_h^{(n-1)} \in \mathbf{V}_h$ be the approximations to solid displacement at time moments t_n and t_{n-1}, respectively. Similarly, Let $\mathbf{q}_h^{(n)}, \mathbf{q}_h^{(n-1)} \in \mathbf{W}_h$ be the approximations to Darcy velocity. Let $p_h^{(n)}, p_h^{(n-1)} \in S_h$ be the approximations to fluid pressure at time moments t_n and t_{n-1}.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

Combined with the implicit Euler discretization, one establishes the following time-marching scheme, for any $\mathbf{v} \in \mathbf{V}_h^0$, $\mathbf{w} \in \mathbf{W}_h^0$, $q \in S_h^0$,

$$\begin{cases} \mathcal{A}_h^{\mathcal{E}}(\mathbf{u}_h^{(n)}, \mathbf{v}) & - \mathcal{B}_h(p_h^{(n)}, \mathbf{v}) = \mathcal{F}_h^{\mathcal{E}}(\mathbf{v}), \\ & \mathcal{A}_h^{\mathcal{D}}(\mathbf{q}_h^{(n)}, \mathbf{w}) - \mathcal{B}_h^{\mathcal{D}}(p_h^{(n)}, \mathbf{w}) = \mathcal{F}_h^{\mathcal{D},1}(\mathbf{w}), \\ \mathcal{B}_h(\mathbf{u}_h^{(n)}, q) + \mathcal{B}_h^{\mathcal{D}}(\mathbf{q}_h^{(n)}, q) & + \mathcal{C}_h^{\mathcal{D}}(p_h^{(n)}, q) = \mathcal{F}_h^{\mathcal{D},2}(q), \end{cases} \tag{14}$$

where

$$\begin{cases} \mathcal{A}_h^{\mathcal{E}}(\mathbf{u}_h^{(n)}, \mathbf{v}) = 2\mu(\varepsilon(\mathbf{u}_h^{(n)}), \varepsilon(\mathbf{v})) + \lambda(\overline{\nabla \cdot \mathbf{u}_h^{(n)}}, \overline{\nabla \cdot \mathbf{v}}), \\ \mathcal{B}_h(p_h^{(n)}, \mathbf{v}) = \alpha(p_h^{(n)}, \overline{\nabla \cdot \mathbf{v}}), \end{cases} \tag{15}$$

and

$$\begin{cases} \mathcal{A}_h^{\mathcal{D}}(\mathbf{q}_h^{(n)}, \mathbf{w}) = \Delta t\, (\mathbf{K}^{-1}\mathbf{q}_h^{(n)}, \mathbf{w}), \\ \mathcal{B}_h^{\mathcal{D}}(\mathbf{q}_h^{(n)}, q) = \Delta t\, (\nabla \cdot \mathbf{q}_h^{(n)}, q), \\ \mathcal{C}_h^{\mathcal{D}}(p_h^{(n)}, q) = c_0(p_h^{(n)}, q). \end{cases} \tag{16}$$

Additionally,

$$\begin{cases} \mathcal{F}_h^{\mathcal{E}}(\mathbf{v}) = \sum_{T \in \mathcal{T}_h} (\mathbf{f}^{(n)}, \mathbf{v})_T + \sum_{e \in \Gamma_N^{\mathcal{E}}} \langle \mathbf{t}_N, \mathbf{v} \rangle_e, \\ \mathcal{F}_h^{\mathcal{D},1}(\mathbf{w}) = - \sum_{e \in \Gamma_D^{\mathcal{D}}} \Delta t\, \langle p_D, \mathbf{w} \cdot \mathbf{n} \rangle_e, \\ \mathcal{F}_h^{\mathcal{D},2}(q) = \sum_{T \in \mathcal{T}_h} \Delta t (s^{(n)}, q)_T + c_0(p_h^{(n-1)}, q)_T + \alpha(\overline{\nabla \cdot \mathbf{u}_h^{(n-1)}}, q)_T. \end{cases} \tag{17}$$

Note $\overline{\nabla \cdot \mathbf{v}}$ is the elementwise average that represents the reduced integration technique. The above two equations are further augmented with appropriate boundary and initial conditions. This results in a large monolithic system at each time step. The DOFs at each time step are

$$2\,\#\text{Nodes} + \#\text{Elements} + 2\,\#\text{Edges}.$$

4 Solver III & IV: 2-field CG+WG Solvers for Triangular and Quadrilateral Meshes

The MFEM(RT_0, P_0) discretization for Darcy flow in Solver II can be replaced by WG$(P_0, P_0; RT_0)$ discretization. This results in a new 2-field solver, which is easier in implementation, based on our experience. This is labeled as Solver III in this series. Here we provide a brief description of the scheme.

Let $\mathcal{T}_h$ be a quasi-uniform triangular mesh. We use BR_1 for displacement discretization (in linear elasticity), as done in [30]. However, we use WG$(P_0, P_0; RT_0)$ for pressure discretization (in Darcy flow) [13,16]. Let $\mathbf{V}_h$ be the space of BR_1

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

shape functions on $\mathcal{T}_h$ and $\mathbf{V}_h^0$ be its subspace with vanishing values on solid Dirichlet boundary. Similarly, S_h denotes the space of $\mathrm{WG}(P_0, P_0)$ shape functions on $\mathcal{T}_h$ and S_h^0 be its subspace with vanishing values on fluid Dirichlet boundary. Treatment of initial and boundary conditions involve appropriate interpolation and/or projection operators into respective finite element spaces [26]:

$$\mathbf{u}_h^{(0)} = \mathbf{P}_h \mathbf{u}_0, \qquad p_h^{(0)} = Q_h p_0,$$

and

$$\mathbf{u}_h^{(n)}|_{\Gamma_D^{\mathcal{E}}} = \mathbf{P}_h \mathbf{u}_D, \qquad p_h^{(n,\partial)}|_{\Gamma_D^{\mathcal{D}}} = Q_h^{\partial}(p_D).$$

Solver III as a time-marching finite element scheme is formulated as

$$\begin{cases} \mathcal{A}_h^{\mathcal{E}}(\mathbf{u}_h^{(n)}, \mathbf{v}) - \mathcal{B}_h(p_h^{(n)}, \mathbf{v}) = \mathcal{F}_h^{\mathcal{E}}(\mathbf{v}), \\ \mathcal{B}_h(\mathbf{u}_h^{(n)}, q) + \mathcal{A}_h^{\mathcal{D}}(p_h^{(n)}, q) = \mathcal{F}_h^{\mathcal{D}}(q), \end{cases} \tag{18}$$

for any $\mathbf{v} \in \mathbf{V}_h^0$ and any $q \in S_h^0$. The bilinear forms are defined as

$$\begin{cases} \mathcal{A}_h^{\mathcal{E}}(\mathbf{u}_h^{(n)}, \mathbf{v}) = \sum\limits_{T \in \mathcal{T}_h} 2\mu(\varepsilon(\mathbf{u}_h^{(n)}), \varepsilon(\mathbf{v}))_T + \lambda(\overline{\nabla \cdot \mathbf{u}_h^{(n)}}, \overline{\nabla \cdot \mathbf{v}})_T, \\ \mathcal{A}_h^{\mathcal{D}}(p_h^{(n)}, q) = \sum\limits_{T \in \mathcal{T}_h} \Delta t \left(\mathbf{K} \nabla_w p_h^{(n)}, \nabla_w q \right)_T + c_0 (p_h^{(n),\circ}, q^{\circ})_{T^\circ}, \\ \mathcal{B}_h(\mathbf{u}_h^{(n)}, q) = \sum\limits_{T \in \mathcal{T}_h} \alpha (\nabla \cdot \mathbf{u}_h^{(n)}, q^{\circ})_{T^\circ}. \end{cases} \tag{19}$$

The linear forms are defined as

$$\begin{cases} \mathcal{F}_h^{\mathcal{E}}(\mathbf{v}) = \sum\limits_{T \in \mathcal{T}_h} (\mathbf{f}^{(n)}, \mathbf{v})_T + \sum\limits_{e \in \Gamma_N^{\mathcal{E}}} \langle \mathbf{t}_N, \mathbf{v} \rangle_e, \\ \mathcal{F}_h^{\mathcal{D}}(q) = \sum\limits_{T \in \mathcal{T}_h} \Delta t (s^{(n)}, q^{\circ})_{T^\circ} + c_0 (p_h^{(n-1),\circ}, q^{\circ})_{T^\circ} + \alpha (\nabla \cdot \mathbf{u}_h^{(n-1)}, q^{\circ})_{T^\circ} \\ \qquad\quad - \sum\limits_{e \in \Gamma_N^{\mathcal{D}}} \Delta t \langle u_N, q^{\partial} \rangle_e. \end{cases} \tag{20}$$

It is interesting to see that for each time moment t_n, the discrete linear system (18) has the same size as the discrete linear system (14).

Solver IV for quadrilateral meshes is similar to Solver III (for triangular meshes). But quadrilateral meshes are equally versatile as triangular meshes in accommodation of complicated domain geometry but may need less degrees of freedom for discretization. In certain cases, quadrilateral meshes may hold advantages in alignment with geometric and physical features in the problems to be solved [9]. With these considerations, Solver IV in two versions for poroelasticity has already been developed in [8,26].

The later version of Solver IV in [26] applies to general convex quadrilateral meshes. It uses the newly developed AC_0 space in [3] for Darcy flow discretization, which includes the rectangular RT_0 as a special case. For discretization of linear elasticity on quadrilaterals, the BR_1 or enriched Lagrangian elements EQ_1 are used. Therefore, for Solver IV, one just needs slight modification in Equation (18,19,20):

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

- Replace the triangular mesh $\mathcal{T}_h$ by a quadrilateral mesh $\mathcal{E}_h$;
- Replace the triangular BR_1 elements by quadrilateral BR_1 elements (the enriched Lagrangian elements EQ_1);
- Replace triangular $\mathrm{WG}(P_0, P_0; RT_0)$ by $\mathrm{WG}(P_0, P_0; AC_0)$ for quadrilaterals.

The DOFs for Solver IV is also

$$2\#\text{Nodes} + \#\text{Elements} + 2\#\text{Edges},$$

but there are less elements and edges in a quadrilateral mesh.

5 Solver V Based on Operator-splitting for Problems with Dilation-dependent Permeability

Solver V is developed on top of Solver IV but aims at nonlinear poroelasticity in which permeability may depend on dilation. We adopt the approach of operator splitting (OS), namely, linear elasticity and Darcy problems are solved separately within Gauss-Seidel iterations.

For ease of presentation, we consider a convex quadrilateral mesh $\mathcal{E}_h$. We use EQ_1 or BR_1 finite elements for elasticity discretization [11] and $\mathrm{WG}(P_0, P_0; AC_0)$ for discretization of Darcy flow [17], as in Solver IV. Treatment of initial and boundary conditions is similar to that in Solver III or IV.

Solver V as a time-marching finite element scheme is formulated as

$$\begin{cases} \mathcal{A}_h^{\mathcal{E}}(\mathbf{u}_h^{(n)}, \mathbf{v}) - \mathcal{B}_h(p_h^{(n)}, \mathbf{v}) & = \mathcal{F}_h^{\mathcal{E}}(\mathbf{v}), \\ \mathcal{B}_h(\mathbf{u}_h^{(n)}, q) + \mathcal{A}_h^{\mathcal{D}}(p_h^{(n)}, q; \mathbf{u}_h^{(n)}) & = \mathcal{F}_h^{\mathcal{D}}(q), \end{cases} \tag{21}$$

for any $\mathbf{v} \in \mathbf{V}_h^0$ and any $q \in S_h^0$, where the FE spaces have definitions similar to those for Solver IV. The bilinear forms $\mathcal{A}_h^{\mathcal{E}}(\mathbf{u}_h^{(n)}, \mathbf{v})$ and $\mathcal{B}_h(\mathbf{u}_h^{(n)}, q)$ have similar definitions as in Equation (19). But the bilinear form $\mathcal{A}_h^{\mathcal{D}}(p_h^{(n)}, q; \mathbf{u}_h^{(n)})$ depends on the numerical displacement as shown below

$$\mathcal{A}_h^{\mathcal{D}}(p_h^{(n)}, q; \mathbf{u}_h^{(n)}) = \sum_{E \in \mathcal{E}_h} \Delta t \left(\mathbf{K}(\mathbf{u}_h^{(n)}) \nabla_w p_h^{(n)}, \nabla_w q \right)_E + c_0 (p_h^{(n),\circ}, q^\circ)_{E^\circ}, \tag{22}$$

The linear forms $\mathcal{F}_h^{\mathcal{E}}(\mathbf{v})$ and $\mathcal{F}_h^{\mathcal{D}}(q)$ have definitions similar to those in (19).

However, (21) is a nonlinear discrete system about $\mathbf{u}_h^{(n)}, p_h^{(n)}$. This will be solved via operator-splitting or a Gauss-Seidel type iterative procedure as shown below

$$\begin{cases} \mathcal{A}_h^{\mathcal{E}}(\mathbf{u}_h^{(n,k)}, \mathbf{v}) = \mathcal{F}_h^{\mathcal{E}}(\mathbf{v}) + \mathcal{B}_h(p_h^{(n,k-1)}, \mathbf{v}), \\ \mathcal{A}_h^{\mathcal{D}}(p_h^{(n,k)}, q; \mathbf{u}_h^{(n,k)}) = \mathcal{F}_h^{\mathcal{D}}(q) - \mathcal{B}_h(\mathbf{u}_h^{(n,k)}, q). \end{cases} \tag{23}$$

As shown later in Section 6, a typical nonlinear case is a dilation-dependent permeability, e.g.,

$$\mathbf{K}(\mathbf{u}) = (1 + a \nabla \cdot \mathbf{u}) \mathbf{K}_0, \tag{24}$$

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

where a is a small constant and $\mathbf{K}_0$ is a reference permeability. This requires calculation of elementwise averages of dilation (divergence of displacement). It is clear that for Solver IV and hence Solver V, such quantities are readily available.

6 `Matlab` Implementation of Poroelasticity Solvers

For `Matlab` implementation of the poroelasticity solvers discussed in this paper and other similar solvers, we emphasize code modularity. This has implication in two aspects.

(i) Each module has its entirety and a well-designed interface to the calling module. Each module fulfills a well-defined scientific computing task that is clearly separated from other tasks.
(ii) Code modules for similar solvers share uniformity and common features. Some code segments are conveniently portable or can be re-used after simple modification.

Besides mesh preparation and presentation of results (physical quantities of interest and errors when exact solutions are known, etc.), a typical finite element solver usually involves

- Element-wise or edge-wise integration and even node-wise evaluation;
- Assembly of element-wise stiffness matrices and source-type vectors;
- Incorporation and enforcement of boundary conditions;
- Modification of (non-)linear systems due to boundary conditions;
- Solvers for linear or nonlinear systems.

For instance,

- Solver I, IV, V share the same modules for mesh preparation and presentation of quantities of interest;
- Solver I, IV, V share many common modules of $\mathrm{WG}(P_0, P_0; AC_0)$ for pressure discretization and Darcy velocity computation;
- Solver II, III share common modules for triangle Bernardi-Raugel elements;
- The modules for triangular and quadrilateral BR_1 elements share the same structure and many common features.

Many modules previously developed in our `DarcyLite` package for weak Galerkin FEMs for Darcy flow and linear elasticity are also re-used.

7 Numerical Experiments

This section presents numerical examples to demonstrate the accuracy and robustness of the finite element solvers for poroelasticity studied in this paper. We shall focus on Solver I, III, and V. Solver II is essentially equivalent to Solver

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

III but the latter seems to have some convenience in implementation. Some numerical experiments on Solver IV can be found in [26].

Example 1 (Locking-free). Here $\Omega = (0,1)^2$. Analytical solutions for displacement and pressure are given as

$$\mathbf{u} = \sin\left(\frac{\pi}{2}t\right)\left(\begin{bmatrix} \sin^2(\pi x)\sin(2\pi y) \\ -\sin^2(\pi y)\sin(2\pi x) \end{bmatrix} + \frac{1}{1+\lambda}\begin{bmatrix} \sin(\pi x)\sin(\pi y) \\ \sin(\pi x)\sin(\pi y) \end{bmatrix}\right),$$

Table 1. Ex.1 ($\lambda = 1$): Errors and convergence rates of numerical solutions obtained from Solver I (WG+WG) on rectangular meshes

$1/h$	$1/\Delta t$	$\|\mathbf{u}-\mathbf{u}_h\|_{L_2(L_2)}$	Rate	$\|p-p_h^\circ\|_{L_2(L_2)}$	Rate	$\|\mathbf{q}-\mathbf{q}_h\|_{L_2(L_2)}$	Rate
8	8	1.2757E-1	–	1.3289E-1	–	4.2093E-1	–
16	16	6.1993E-2	1.04	6.4829E-2	1.03	2.0427E-1	1.04
32	32	3.0529E-2	1.02	3.1964E-2	1.02	1.0056E-1	1.02
64	64	1.5147E-2	1.01	1.5863E-2	1.01	4.9881E-2	1.01

Table 2. Ex.1 ($\lambda = 10^6$): Errors and convergence rates of numerical solutions obtained from Solver I (WG+WG) on rectangular meshes

$1/h$	$1/\Delta t$	$\|\mathbf{u}-\mathbf{u}_h\|_{L_2(L_2)}$	Rate	$\|p-p_h^\circ\|_{L_2(L_2)}$	Rate	$\|\mathbf{q}-\mathbf{q}_h\|_{L_2(L_2)}$	Rate
8	8	1.2042e-01	–	2.6577e-07	–	8.4154E-7	–
16	16	5.8469e-02	1.04	1.2965e-07	1.03	4.0848E-7	1.04
32	32	2.8786e-02	1.02	6.3926e-08	1.02	2.0110E-7	1.02
64	64	1.4281e-02	1.01	3.1727e-08	1.01	9.9761E-8	1.01

Table 3. Ex.1 ($\lambda = 1$): Numerical results of Solver III (CG+WG) on triangular meshes

$h = \Delta t$	$\|\mathbf{u}-\mathbf{u}_h\|_{L_2(L_2)}$	$\|\mathbf{u}-\mathbf{u}_h\|_{L_\infty(H^1)}$	$\|\sigma-\sigma_h\|_{L_2(L_2)}$	$\|p-p_h^\circ\|_{L_2(L_2)}$	$\|\mathbf{q}-\mathbf{q}_h\|_{L_2(L_2)}$
1/4	6.533E-2	1.401E+0	1.775E+0	1.613E-1	8.760E-1
1/8	1.485E-2	6.648E-1	8.230E-1	7.696E-2	4.186E-1
1/16	3.551E-3	3.265E-1	3.964E-1	3.745E-2	2.038E-1
1/32	8.727E-4	1.624E-1	1.948E-1	1.845E-2	1.004E-1
1/64	2.177E-4	8.113E-2	9.662E-2	9.160E-3	4.986E-2
Conv.rate	2.05	1.02	1.04	1.03	1.03

Table 4. Ex.1 ($\lambda = 10^6$): Numer. results of Solver III (CG+WG) on triangular meshes

$h = \Delta t$	$\|\mathbf{u}-\mathbf{u}_h\|_{L_2(L_2)}$	$\|\mathbf{u}-\mathbf{u}_h\|_{L_\infty(H^1)}$	$\|\sigma-\sigma_h\|_{L_2(L_2)}$	$\|p-p_h^\circ\|_{L_2(L_2)}$	$\|\mathbf{q}-\mathbf{q}_h\|_{L_2(L_2)}$
1/4	6.502E-2	1.395E+0	1.814E+0	3.226E-7	1.752E-6
1/8	1.485E-2	6.631E-1	8.537E-1	1.539E-7	8.372E-7
1/16	3.557E-3	3.259E-1	4.132E-1	7.491E-8	4.076E-7
1/32	8.727E-4	1.622E-1	2.033E-1	3.691E-8	2.009E-7
1/64	2.163E-4	8.102E-2	1.008E-1	1.832E-8	9.972E-8
Conv.rate	2.05	1.02	1.03	1.03	1.03

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

$$p = \sin\left(\frac{\pi}{2}t\right)\frac{\pi}{1+\lambda}\sin\left(\pi(x+y)\right).$$

It is interesting to see that $\nabla\cdot\mathbf{u} = p$ and $\nabla\cdot\mathbf{u}\to 0$ as $\lambda\to\infty$. Dirichlet boundary conditions for both displacement and pressure are specified on the whole boundary using the exact solutions. Furthermore, $\mathbf{K} = \kappa\mathbf{I}$. Direct calculations show that

$$\begin{aligned}\mathbf{f} = -\nabla\cdot\widetilde{\sigma} = -\sin\left(\frac{\pi}{2}t\right)\Bigg(&2\mu\pi^2\begin{bmatrix}(1-4\sin^2(\pi x))\sin(2\pi y)\\ -(1-4\sin^2(\pi y))\sin(2\pi x)\end{bmatrix}\\ &-\frac{2\mu}{1+\lambda}\pi^2\begin{bmatrix}\sin(\pi x)\sin(\pi y)\\ \sin(\pi x)\sin(\pi y)\end{bmatrix}+\frac{\lambda+\mu-\alpha}{1+\lambda}\pi^2\begin{bmatrix}\cos(\pi(x+y))\\ \cos(\pi(x+y))\end{bmatrix}\Bigg)\end{aligned} \tag{25}$$

and

$$s = \left((\alpha+c_0)\cos\left(\frac{\pi}{2}t\right)\frac{\pi}{2}+\sin\left(\frac{\pi}{2}t\right)\kappa(2\pi^2)\right)\frac{\pi}{1+\lambda}\sin(\pi(x+y)). \tag{26}$$

For numerical simulations, we set $\kappa = 1$, $\mu = 1$, $\alpha = 1$, $c_0 = 0$, and $T = 1$. To examine the locking-free property of these solvers, we consider $\lambda = 1$ and $\lambda = 10^6$, respectively.

We examine errors in displacement $(\mathbf{u}-\mathbf{u}_h)$, stress $(\sigma-\sigma_h)$, pressure $(p-p_h^\circ)$, and Darcy velocity $(\mathbf{q}-\mathbf{q}_h)$. For Solver I, see results in Table 1 and 2. For Solver III, see results in Table 3 and 4. Clearly, the convergence rates do not deteriorate as λ is increased from 1 to 10^6. In other words, these new 2-field solvers based on the weak Galerkin methodology are locking-free.

Example 2 (Model's problem). This is a frequently tested benchmark that has known analytical solutions. See [1,5,12,18,22]. The problem involves a poroelastic rectangular slab with extent $2a$ in the x-direction and extent $2b$ in the y-direction being sandwiched by two rigid plates at the top and the bottom. Two forces of magnitude $2F$, pointing to the slab, are applied at the top and bottom plates, respectively. Due to the rigidity of the plates, the slab remains in contact with the two plates. Thus the vertical displacement at the top and bottom are uniform. The initial condition for displacement is $\mathbf{u}(x,y,0) = \mathbf{0}$. Based on symmetry in the problem, we choose the center of the slab as the origin and consider the upper-right quadrant. The Mandel's problem is thus posed for the domain $\Omega = (0,a)\times(0,b)$ for a time period $[0,T]$.

The boundary conditions for the solid and fluid are, see Figure 1(a),

(i) Symmetry or partial Dirichlet: $u_1 = 0$ for $x = 0$; $u_2 = 0$ for $y = 0$;
(ii) Neumann or traction-free: $\tilde{\sigma}\mathbf{n} = \mathbf{0}$ for $x = a$;
(iii) Specially, for $y = b$ (the top side), it is subject to the traction condition $\tilde{\sigma}\mathbf{n} = [0,-2F]$ along with the "rigid plate" constraint, which requires u_2 stays the same for the whole top side;
(iv) Dirichlet: $p = 0$ for $x = a$ (drained);
(v) Neumann or no-flow: $(-\mathbf{K}\nabla p)\cdot\mathbf{n} = 0$ for $x = 0, y = 0, y = b$.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

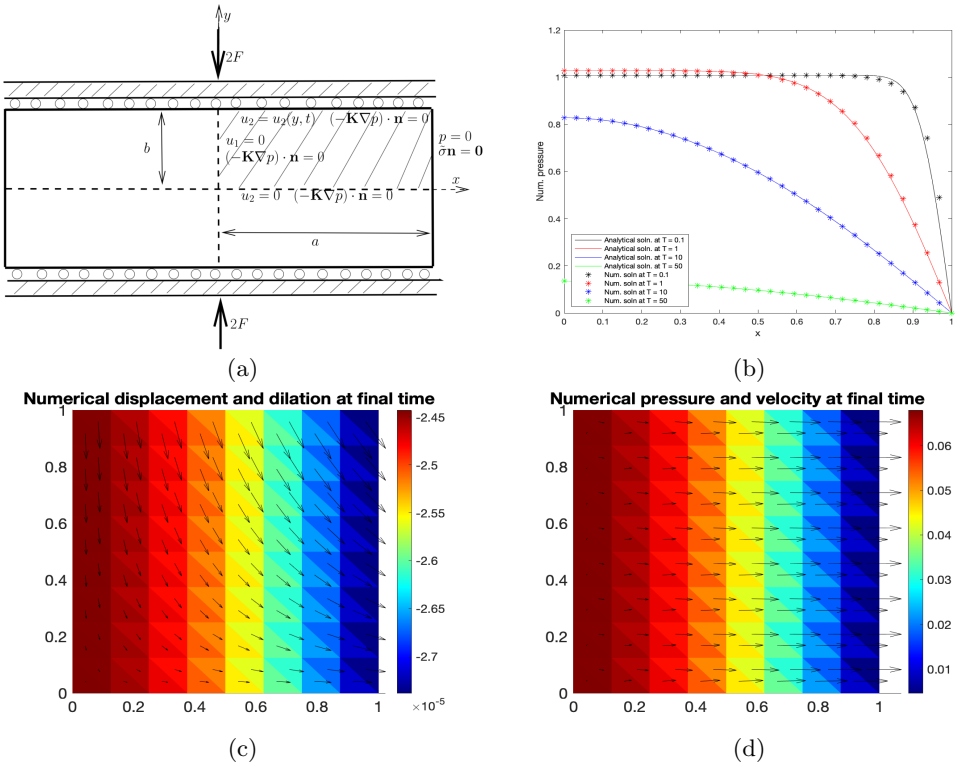

Fig. 1. Mandel's problem ($\nu = 0.4$) solved by Solver III on uniform triangular meshes. **(a)** Problem illustration; **(b)** Numerical pressure on the center line ($y = 0$) at final time $T = 50$ with $h = 1/32$; **(c)** Numerical displacement and dilation with $h = 1/8$; **(d)** Numerical pressure and velocity with $h = 1/8$.

An easier but equivalent treatment for (iii) is to impose a partial Dirichlet boundary condition for u_2 using the known exact solution for displacement [5,22,26].

Example 3 (A nonlinear problem). Here we consider an example in which the permeability depends on the dilation. In particular,

$$\mathbf{K}(\mathbf{u}) = (1 + a\nabla\cdot\mathbf{u})\,\kappa\,\mathbf{I}, \tag{27}$$

where a is a small constant and κ is a reference permeability. Furthermore, we consider a case with known analytical solutions for displacement and pressure:

$$\mathbf{u} = \sin\left(\frac{\pi}{2}t\right)\begin{bmatrix}\sin(\pi x)\sin(\pi y)\\ \sin(\pi x)\sin(\pi y)\end{bmatrix}, \quad p = \sin\left(\frac{\pi}{2}t\right)(1+\cos(\pi y))\,. \tag{28}$$

This allows examination of accuracy and efficiency of Solver V. Furthermore, $\Omega = (0,1)^2$, $T = 1$, $\lambda = \mu = 1$, $\kappa = 1$, $a = 0.1$, $c_0 = 0$, and $\alpha = 1$. Dirichlet boundary conditions are posed for both displacement and pressure.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

Table 5. Ex.3 (Solver V): Convergence rates of errors of numerical solutions obtained from combining EQ_1 and $\mathrm{WG}(P_0, P_0; AC_0)$ on rectangular meshes with $\Delta t = h$

$1/h$	$\|p - p_h^\circ\|$	Rate	$\|\mathbf{q} - \mathbf{q}_h\|$	Rate	$\|\mathbf{u} - \mathbf{u}_h\|$	Rate	$\|\sigma - \sigma_h\|$	Rate	Runtime
4	5.119E-1	–	1.366E+0	–	1.260E-1	–	1.503E-1	–	0.59s
8	2.528E-1	1.01	6.527E-1	1.06	6.023E-2	1.06	6.398E-2	1.23	1.26s
16	1.260E-1	1.00	3.177E-1	1.03	2.931E-2	1.03	2.944E-2	1.11	4.12s
32	6.297E-2	1.00	1.565E-1	1.02	1.444E-2	1.02	1.413E-2	1.05	26.57s
64	3.148E-2	1.00	7.770E-2	1.01	7.169E-3	1.01	6.931E-3	1.02	333.46s

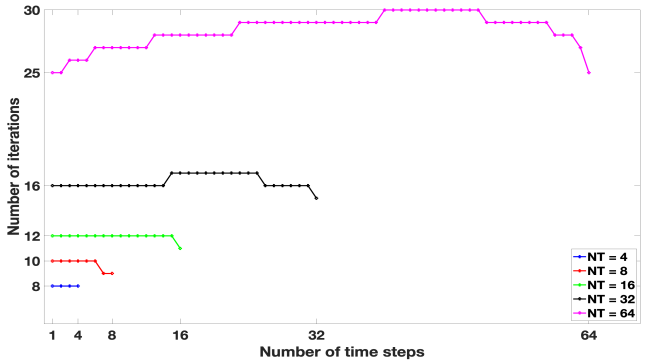

Fig. 2. Example 3: Numbers of Gauss-Seidel iterations during time-marching

We test Solver V on rectangular meshes with $\Delta t = h$. Following the common practices, we examine the discrepancy (difference) of two successive approximate solutions within the Gauss-Seidel iteration. We set discrepancy threshold as $\delta = 10^{-12}$ and check whether the following conditions are satisfied:

$$\begin{aligned} \|\mathbf{u}_h^{(n,k)} - \mathbf{u}_h^{(n,k-1)}\|_{L_2} < \delta, \\ \|p_h^{(n,k)} - p_h^{(n,k-1)}\|_{L_2} < \delta \end{aligned} \tag{29}$$

Table 5 demonstrates good performance of Solver V on this nonlinear poroelasticity problem. For these particular choices of parameters, the numbers of Gauss-Seidel iterations during the time-marching are reported in Figure 2.

8 Concluding Remarks

In this paper, we have discussed five different finite element solvers for linear and nonlinear poroelasticity problems, along with `Matlab` implementation of these solvers. It is demonstrated that weak Galerkin finite element methods can be well integrated with other types of finite element methods. This is also reflected in the modularity of our code development. Under the guidelines discussed in

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

this paper, more modules for finite element solvers for poroelasticity can be integrated into our code package DarcyLite [15].

These poroelasticity modules can also provide computed physical quantities that are needed in finite element solvers for other physical processes. It is particularly interesting to see integration of these poroelasticity solvers with (mass, positivity) property-preserving transport solvers in development of numerical simulators for transport in poroelastic media. This is currently under our investigation and will be reported in our future work.

The work in this paper emphasizes easy access of poroelasticity solvers on the platform offered by Matlab. It echoes our efforts in [24,25] for efficient implementation of WG solvers in deal.II and C++ for large-scale computing tasks. More results from such efforts will be reported in our future work.

References

1. Abousleiman, Y., Cheng, A.D., Cui, L., Detournay, E., Roegiers, J.: Mandel's problem revisted. Geotechnique **46**, 187–195 (1996)
2. Alberty, J., Carstensen, C., Funken, S.: Remarks around 50 lines of Matlab: short finite element implementation. Numer. Algor. **20**, 117–137 (1999)
3. Arbogast, T., Correa, M.: Two families of mixed finite elements on quadrilaterals of minimal dimension. SIAM J. Numer. Anal. **54**, 3332–3356 (2016)
4. Berger, L., Bordas, R., Kay, D., Tavener, S.: Stabilized lowest-order finite element approximation for linear three-field poroelasticity. SIAM J. Sci. Comput. **37**, A2222–A2245 (2015)
5. Correa, M.R., Murad, M.A.: A new sequential method for three-phase immiscible flow in poroelastic media. J. Comput. Phys. **373**, 493–532 (2018)
6. Fu, G.: A high-order HDG method for the Biot's consolidation model. Comput Math. Appl. **77**(1), 237–252 (2019)
7. Haga, J., Osnes, H., Langtangen, H.: On the causes of pressure oscillations in low permeable and low compressible porous media. Int. J. Numer. Anal. Meth. Geomech. **36**, 1507–1522 (2012)
8. Harper, G., Liu, J., Tavener, S., Wang, Z.: A two-field finite element solver for poroelasticity on quadrilateral meshes. Lec. Notes Comput. Sci. **10862**, 76–88 (2018)
9. Harper, G., Liu, J., Tavener, S., Wildey, T.: Coupling arbogast-correa and bernardi-raugel elements to resolve coupled stokes-darcy flow problems. Comput. Meth. Appl. Mech. Engrg. **373**, Article 113469 (2021)
10. Harper, G., Liu, J., Tavener, S., Zheng, B.: Lowest-order weak Galerkin finite element methods for linear elasticity on rectangular and brick meshes. J. Sci. Comput. **78**, 1917–1941 (2019)
11. Harper, G., Wang, R., Liu, J., Tavener, S., Zhang, R.: A locking-free solver for linear elasticity on quadrilateral and hexahedral meshes based on enrichment of Lagrangian elements. Comput. Math. Appl. **80**, 1578–1595 (2020)
12. Hu, X., Mu, L., Ye, X.: Weak Galerkin method for the Biot's consolidation model. Comput. Math. Appl. **75**, 2017–2030 (2018)
13. Lin, G., Liu, J., Mu, L., Ye, X.: Weak Galerkin finite element methdos for Darcy flow: Anistropy and heterogeneity. J. Comput. Phys. **276**, 422–437 (2014)

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15

14. Lin, G., Liu, J., Sadre-Marandi, F.: A comparative study on the weak Galerkin, discontinuous Galerkin, and mixed finite element methods. J. Comput. Appl. Math. **273**, 346–362 (2015)
15. Liu, J., Sadre-Marandi, F., Wang, Z.: `DarcyLite`: A Matlab toolbox for Darcy flow computation. Proc. Comput. Sci. **80**, 1301–1312 (2016)
16. Liu, J., Tavener, S., Wang, Z.: The lowest-order weak Galerkin finite element method for the Darcy equation on quadrilateral and hybrid meshes. J. Comput. Phys. **359**, 312–330 (2018)
17. Liu, J., Tavener, S., Wang, Z.: Penalty-free any-order weak galerkin fems for elliptic problems on quadrilateral meshes. J. Sci. Comput. **83**, Article 47 (2020)
18. Phillips, P., Wheeler, M.: A coupling of mixed with continuous Galerkin finite element methods for poroelasticity I: the continuous in time case. Comput. Geosci. **11**, 131–144 (2007)
19. Phillips, P., Wheeler, M.: A coupling of mixed with continuous Galerkin finite element methods for poroelasticity II: the-discrete-in-time case. Comput. Geosci. **11**, 145–158 (2007)
20. Phillips, P., Wheeler, M.: A coupling of mixed with discontinuous Galerkin finite element methods for poroelasticity. Comput. Geosci. **12**, 417–435 (2008)
21. Phillips, P.J., Wheeler, M.F.: Overcoming the problem of locking in linear elasticity and poroelasticity: an heuristic approach. Comput. Geosci. **13**, 5–12 (2009)
22. Rodrigo, C., Gaspar, F., Hu, X., Zikatanov, L.: Stability and monotonicity for some discretizations of the Biot's consolidation model. Comput. Meth. Appl. Mech. Engrg. **298**, 183–204 (2016)
23. Wang, J., Ye, X.: A weak Galerkin finite element method for second order elliptic problems. J. Comput. Appl. Math. **241**, 103–115 (2013)
24. Wang, Z., Harper, G., O'Leary, P., Liu, J., Tavener, S.: `deal.II` implementation of a weak Galerkin finite element solver for Darcy flow. Lec. Notes Comput. Sci. **11539**, 495–509 (2019)
25. Wang, Z., Liu, J.: `deal.II` implementation of a two-field finite element solver for poroelasticity. Lec. Notes Comput. Sci. **12143**, 88–101 (2020)
26. Wang, Z., Tavener, S., Liu, J.: Analysis of a 2-field finite element solver for poroelasticity on quadrilateral meshes. J. Comput. Appl. Math. **393**, Article 113539 (2021)
27. Wheeler, M., Xue, G., Yotov, I.: Coupling multipoint flux mixed finite element methods with continuous Galerkin methods for poroelasticity. Comput. Geosci. **18**, 57–75 (2014)
28. Yi, S.Y.: A coupling of nonconforming and mixed finite element methods for Biot's consolidation model. Numer. Meth. PDEs **29**, 1749–1777 (2013)
29. Yi, S.Y.: Convergence analysis of a new mixed finite element method for Biot's consolidation model. Numer. Meth. PDEs **30**, 1189–1210 (2014)
30. Yi, S.Y.: A study of two modes of locking in poroelasticity. SIAM J. Numer. Anal. **55**, 1915–1936 (2017)

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_15