HIGHLY EFFICIENT ENSEMBLE ALGORITHMS FOR COMPUTING THE STOKES-DARCY EQUATIONS

NAN JIANG* AND HUANHUAN YANG †

Abstract. In this report we propose two highly efficient ensemble algorithms incorporating the gPAV and the rotational pressure correction methods for computing Stokes-Darcy flow ensembles. All variables are fully decoupled including the three components of the velocity, leading to smaller linear systems to be solved at each time step. Moreover, all ensemble members share the same constant coefficient matrix for which the fast block CG method can be applied for computing the ensemble at one pass at significantly reduced computational cost. We prove the two new ensemble algorithms are unconditionally stable with respect to the modified energy without any constraints on the uncertain parameters or the time steps. We will provide details of implementation and discuss how to efficiently solve the corresponding linear systems. Numerical examples are presented to show the efficiency and effectiveness of the algorithms.

Key words. Stokes-Darcy equations, gPAV, uncertainty quantification, ensemble algorithm, partitioned method, rotational pressure correction method

1. Introduction. Uncertainty quantification (UQ) usually requires repeated simulations of a flow problem with random inputs of an uncertain parameter. This leads to a high demand on computer resources especially for complex flow problems. There have been significant efforts put into studying efficient numerical methods to reduce the computational cost for UQ, including multilevel Monte Carlo method [4], quasi-Monte Carlo sequences [35], Latin hypercube sampling [19], centroidal Voronoi tessellations [45], and more recently developed stochastic collocation methods [3, 51], non-intrusive polynomial chaos methods [21, 44], and ensemble algorithms [25]. In particular, unlike other methods that aim to reduce the number of simulations or samples required for effective UQ, the ensemble algorithms are specially designed for fast computation of an ensemble of flows in response to different flow parameters and its goal is to be able to compute a large ensemble at significantly reduced computational cost. The ensemble algorithms were initially studied for nonlinear flow problems [25], and the idea is to decompose the nonlinear term into two parts: the mean and the fluctuation. The mean is independent of the ensemble index and will be the same for all realizations, while the fluctuation is different for each realization but is lagged to the previous timesteps so that it goes to the right hand side of the system and does not contribute to the coefficient matrix. Consequently all ensemble members computed will have the same coefficient matrix at each timestep and efficient block linear solvers such as block CG, block GMRES, can be used to compute all realizations at one pass at significantly reduced computational cost. The ensemble algorithms have been extensively tested on various nonlinear flow problems such as the Navier-Stokes flows [16, 17, 18, 24, 26, 31, 32, 48, 49], MHD flows [1, 2, 30, 42], natural convection [13, 14] and fluid-fluid interactions [11], and demonstrated to be very effective in reducing computational cost while maintaining comparable accuracy. The ensemble timestepping idea has also been extended to linear PDE models such as the heat equations [13, 40, 41] and the Stokes-Darcy equations [20, 29, 27], to handle uncertain model parameters.

The Stokes-Darcy equations are a coupled PDE system that models the coupling of a free surface flow and a subsurface porous media flow that appears in many geophysical applications. The main difficulties in designing efficient and long time stable numerical schemes for computing the system include 1) the coupling terms in the discrete schemes usually can not be properly bounded and lead to a time step condition to ensure stability; 2) projection type methods that alleviate the incompressibility constraint in the Stokes equations do not provide intended efficiency as it only decouples the computation of pressure from the velocity but not the three components of the velocity field due to a term $\sum_i \int_I \eta_i (u \cdot \widehat{\tau}_i)(v \cdot \widehat{\tau}_i)\, ds$ from the interface conditions of the Stokes-Darcy model. There have been some recent developments to address the first issue, such as adding a stabilization [28] to remove the time step condition and using the SAV approach to avoid bounding the coupling terms [33, 34]. To our best knowledge, the second issue has not been addressed in the literature. In this report, we propose two ensemble algorithms that have both issues addressed by adopting the generalized positive auxiliary variable (gPAV) idea [52]. The two ensemble algorithms are 1)

*Department of Mathematics, University of Florida, Gainesville, FL 32611, jiangn@ufl.edu. This author was partially supported by the US National Science Foundation grant DMS-2143331.

†Corresponding author. Department of Mathematics, Shantou University, Guangdong, China 515063, huan2yang@stu.edu.cn. This author was supported by the Guangdong Basic and Applied Basic Research Foundation (2023A1515030199).

unconditionally stable without any time step constraints; 2) fully decoupled with the three components of the velocity u_1, u_2, u_3, the pressure p, the hydraulic head ϕ computed separately.

Let D_f denote the surface fluid flow region and D_p the porous media flow region, where $D_f, D_p \subset R^d (d = 2, 3)$ are both open, bounded domains. These two domains lie across an interface, I, from each other and $D_f \cap D_p = \emptyset, \bar{D}_f \cap \bar{D}_p = I$, see Figure 1.1.

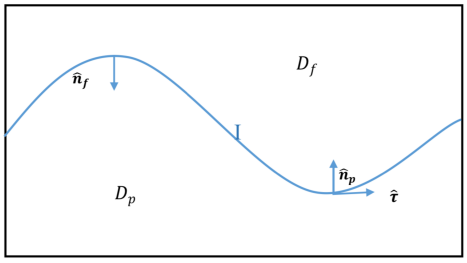


Fig. 1.1: A sketch of the porous median domain D_p, fluid domain D_f, and the interface I.

The linear Stokes-Darcy system [5, 10] that models the coupling of the surface and porous media flows is: find fluid velocity $u(x,t)$, fluid pressure $p(x,t)$, and hydraulic head $\phi(x,t)$ that satisfy

$$\begin{aligned} \partial_t u - \nu\Delta u + \nabla p = f_f(x,t), \nabla\cdot u = 0 \ \text{ in } \ D_f\times(0,T], \\ S_0\partial_t\phi - \nabla\cdot(\mathscr{K}(x)\nabla\phi) = f_p(x,t) \ \text{ in } \ D_p\times(0,T], \\ \phi(x,0) = \phi^0(x) \text{ in } D_p \text{ and } u(x,0) = u^0(x) \text{ in } D_f, \\ \phi(x,t) = b(x,t) \text{ in } \partial D_p\backslash I\times(0,T] \text{ and } u(x,t) = a(x,t) \text{ in } \partial D_f\backslash I\times(0,T], \end{aligned} \tag{1.1}$$

where ν, $\mathscr{K}$, S_0, f_f, f_p, and T are the kinematic viscosity, the hydraulic conductivity tensor, specific mass storativity coefficient (positive), the external body force density, the sink/source term, and the final time, respectively. Let $\widehat{n}_{f/p}$ denote the outward unit normal vector on I associated with $D_{f/p}$, where $\widehat{n}_f = -\widehat{n}_p$. The coupling conditions across I are conservation of mass, balance of forces, and the Beavers-Joseph-Saffman condition on the tangential velocity [6, 46, 22]:

$$\begin{aligned} u\cdot\widehat{n}_f - \mathscr{K}\nabla\phi\cdot\widehat{n}_p = 0 \text{ and } p - \nu\,\widehat{n}_f\cdot\nabla u\cdot\widehat{n}_f = g\phi \text{ on } I\times(0,T], \\ -\nu\widehat{\tau}_i\cdot\nabla u\cdot\widehat{n}_f = \frac{\alpha_{\mathrm{BJS}}}{\sqrt{\widehat{\tau}_i\cdot\mathscr{K}\widehat{\tau}_i}} u\cdot\widehat{\tau}_i \ \text{ on } I\times(0,T], \text{ for any tangential vector } \widehat{\tau}_i \text{ on } I. \end{aligned}$$

Here, g is the gravity constant, α_{BJS} is a dimensionless constant in the Beavers-Joseph-Saffman condition depending only on the structure of the porous medium. The conductivity $\mathscr{K}$ is assumed to be symmetric positive definite (SPD).

We consider computing an ensemble of J Stokes-Darcy systems, corresponding to J different parameter sets $(u_j^0, \phi_j^0, a_j, b_j, f_{f,j}, f_{p,j}, \mathscr{K}_j)$, $j = 1, ..., J$,

$$\begin{aligned} \partial_t u_j - \nu\Delta u_j + \nabla p_j = f_{f,j}(x,t),\ \nabla\cdot u_j = 0, \ \text{ in } \ D_f, \\ S_0\partial_t\phi_j - \nabla\cdot(\mathscr{K}_j(x)\nabla\phi_j) = f_{p,j}(x,t), \ \text{ in } \ D_p, \\ \phi_j(x,0) = \phi_j^0(x), \ \text{ in } D_p \text{ and } u_j(x,0) = u_j^0(x), \ \text{ in } D_f, \\ \phi_j(x,t) = b_j(x,t), \ \text{ in } \partial D_p\backslash I \text{ and } u_j(x,t) = a_j(x,t), \ \text{ in } \partial D_f\backslash I. \end{aligned} \tag{1.2}$$

We assume there are uncertainties in the initial conditions $u^0(x), \phi^0(x)$, Dirichlet boundary conditions $a(x,t), b(x,t)$, forcing terms $f_f(x,t), f_p(x,t)$ and the hydraulic conductivity tensor $\mathscr{K}(x)$, J is the number of total samples, and $(u_j^0, \phi_j^0, a_j, b_j, f_{f,j}, f_{p,j}, \mathscr{K}_j)$ is one of the samples drawn from the respective probabilistic distributions. Let $\bar{\mathscr{K}} = \frac{1}{J}\sum_{j=1}^{J}\mathscr{K}_j$ denote the ensemble mean of the hydraulic conductivity tensor $\mathscr{K}_j$ and let $\eta_{i,j} = \frac{\alpha_{\mathrm{BJS}}}{\sqrt{\widehat{\tau}_i\cdot\mathscr{K}_j\widehat{\tau}_i}}$.

The gPAV methods are first proposed in [52] for dissipative systems and have been studied for different PDE models, e.g., Cahn-Hilliard [43], Navier-Stokes [31, 38], MHD [2]. They provide a general framework to devise unconditionally stable linear numerical schemes for nonlinear systems and guarantee that the computed scalar auxiliary variables are positive. Herein we extend this approach to the linear Stokes-Darcy model and utilize it to design unconditionally stable numerical schemes by avoiding bounding the coupling terms in the model. We are also able to lag the interface term $\sum_i \int_I \eta_i (u\cdot\widehat{\tau}_i)(v\cdot\widehat{\tau}_i)\,ds$ to the previous time steps and have all components of the velocity field decoupled leading to highly efficient ensemble schemes.

We define a shifted energy of the form

$$E_j(t) = E(u_j,\phi_j) = \int_{D_f} \frac{1}{2}|u_j|^2 dx + \int_{D_p} \frac{gS_0}{2}|\phi_j|^2 dx + C_0, \tag{1.3}$$

where $E(u_j,\phi_j)$ is the total kinetic energy of the system, which for physical examples is bounded from below, and C_0 is an arbitrarily small positive constant chosen in such a way that $E_j(t) > 0$ for $0 \le t \le T$. Next, let $\mathscr{F}$ be any one-to-one increasing differentiable function with $\mathscr{F}^{-1} = \mathscr{G}$ such that

$$\begin{cases} \mathscr{F}(\chi) > 0, \quad \chi > 0, & (1.4)\\ \mathscr{G}(\chi) > 0, \quad \chi > 0. & (1.5) \end{cases}$$

The scalar variable $R_j(t)$ is defined by

$$R_j(t) = \mathscr{G}(E_j), \tag{1.6}$$
$$E_j(t) = \mathscr{F}(R_j). \tag{1.7}$$

With E_j as in (1.3), $R_j(t)$ then satisfies

$$\mathscr{F}'(R_j)\frac{dR_j}{dt} = \int_{D_f} u_j\cdot\frac{\partial u_j}{\partial t}dx + \int_{D_p} gS_0\phi_j\cdot\frac{\partial\phi_j}{\partial t}dx. \tag{1.8}$$

Since $\frac{\mathscr{F}(R_j)}{E_j} = 1$ for all j, we may write

$$\begin{aligned}
\mathscr{F}'(R_j)\frac{dR_j}{dt} &= \int_{D_f} u_j\cdot\frac{\partial u_j}{\partial t}dx + \int_{D_p} gS_0\phi_j\cdot\frac{\partial\phi_j}{\partial t}dx\\
&\quad - \left[\frac{\mathscr{F}(R_j)}{E_j} - 1\right]\Big[\nu(\nabla u_j,\nabla u_j)_f - (p_j,\nabla\cdot u_j)_f\\
&\qquad - \int_{\partial D_f\setminus I}\nu(\nabla u_j\cdot\widehat{n}_f)\cdot u_j\,ds + \int_{\partial D_f\setminus I} p_j(u_j\cdot\widehat{n}_f)\,ds - (f_{f,j},u_j)_f\\
&\qquad + g(\mathscr{K}_j\nabla\phi_j,\nabla\phi_j)_p - \int_{\partial D_p\setminus I} g\phi_j\mathscr{K}_j\nabla\phi_j\cdot\widehat{n}_p\,ds - g(f_{p,j},\psi)_p\Big]\\
&\quad - \frac{\mathscr{F}(R_j)}{E_j}\Bigg(\Big[\sum_i\int_I \eta_{i,j}(u_j\cdot\widehat{\tau}_i)(u_j\cdot\widehat{\tau}_i)\;ds + c_I(u_j,\phi_j) - c_I(u_j,\phi_j)\Big]\\
&\qquad - \Big[\sum_i\int_I \eta_{i,j}(u_j\cdot\widehat{\tau}_i)(u_j\cdot\widehat{\tau}_i)\;ds + c_I(u_j,\phi_j) - c_I(u_j,\phi_j)\Big]\Bigg)\\
&\quad + \left[1 - \frac{\mathscr{F}(R_j)}{E_j}\right]\Bigg|\int_{D_f} f_{f,j}\cdot u_j dx + \int_{D_p} gf_{p,j}\cdot\phi_j dx\\
&\qquad + \int_{\partial D_f\setminus I}\nu a_j\cdot(\nabla u_j\cdot\widehat{n}_f)\,ds - \int_{\partial D_f\setminus I} a_j\cdot\widehat{n}_f p_j\,ds + \int_{\partial D_p\setminus I} gb_j\mathscr{K}_j\nabla\phi_j\cdot\widehat{n}_p\,ds\Bigg|\\
&= \int_{D_f} u_j\cdot\frac{\partial u_j}{\partial t}dx + \int_{D_p} gS_0\phi_j\cdot\frac{\partial\phi_j}{\partial t}dx\\
&\quad + \Big[\nu(\nabla u_j,\nabla u_j)_f + \frac{\mathscr{F}(R_j)}{E_j}\sum_i\int_I \eta_{i,j}(u_j\cdot\widehat{\tau}_i)(u_j\cdot\widehat{\tau}_i)\;ds - (p_j,\nabla\cdot u_j)_f
\end{aligned}$$

$$
\begin{aligned}
&+\frac{\mathscr{F}(R_j)}{E_j}c_I(u_j,\phi_j)-\int_{\partial D_f\setminus I}\nu(\nabla u_j\cdot\widehat{n}_f)\cdot u_j\,ds+\int_{\partial D_f\setminus I}p_j(u_j\cdot\widehat{n}_f)\,ds-(f_{f,j},u_j)_f\\
&+g(\mathscr{K}_j\nabla\phi_j,\nabla\phi_j)_p-\frac{\mathscr{F}(R_j)}{E_j}c_I(u_j,\phi_j)-\int_{\partial D_p\setminus I}g\phi_j\mathscr{K}_j\nabla\phi_j\cdot\widehat{n}_p\,ds-g(f_{p,j},\psi)_p\Big]\\
&-\frac{\mathscr{F}(R_j)}{E_j}\Big[\nu(\nabla u_j,\nabla u_j)_f+\sum_i\int_I\eta_{i,j}(u_j\cdot\widehat{\tau}_i)(u_j\cdot\widehat{\tau}_i)\,ds-(p_j,\nabla\cdot u_j)_f\\
&+c_I(u_j,\phi_j)-\int_{\partial D_f\setminus I}\nu(\nabla u_j\cdot\widehat{n}_f)\cdot u_j\,ds+\int_{\partial D_f\setminus I}p_j(u_j\cdot\widehat{n}_f)\,ds-(f_{f,j},u_j)_f\\
&+g(\mathscr{K}_j\nabla\phi_j,\nabla\phi_j)_p-c_I(u_j,\phi_j)-\int_{\partial D_p\setminus I}g\phi_j\mathscr{K}_j\nabla\phi_j\cdot\widehat{n}_p\,ds-g(f_{p,j},\psi)_p\Big]\\
&+\Big[1-\frac{\mathscr{F}(R_j)}{E_j}\Big]\Big|\int_{D_f}f_{f,j}\cdot u_j dx+\int_{D_p}gf_{p,j}\cdot\phi_j dx\\
&+\int_{\partial D_f\setminus I}\nu a_j\cdot(\nabla u_j\cdot\widehat{n}_f)\,ds-\int_{\partial D_f\setminus I}a_j\cdot\widehat{n}_f p_j\,ds+\int_{\partial D_p\setminus I}gb_j\mathscr{K}_j\nabla\phi_j\cdot\widehat{n}_p\,ds\Big|\\
=&\int_{D_f}u_j\cdot\frac{\partial u_j}{\partial t}dx+\int_{D_p}gS_0\phi_j\cdot\frac{\partial\phi_j}{\partial t}dx\\
&+\Big[\nu(\nabla u_j,\nabla u_j)_f+\frac{\mathscr{F}(R_j)}{E_j}\sum_i\int_I\eta_{i,j}(u_j\cdot\widehat{\tau}_i)(u_j\cdot\widehat{\tau}_i)\,ds-(p_j,\nabla\cdot u_j)_f\\
&+\frac{\mathscr{F}(R_j)}{E_j}c_I(u_j,\phi_j)-\int_{\partial D_f\setminus I}\nu(\nabla u_j\cdot\widehat{n}_f)\cdot u_j\,ds+\int_{\partial D_f\setminus I}p_j(u_j\cdot\widehat{n}_f)\,ds-(f_{f,j},u_j)_f\\
&+g(\mathscr{K}_j\nabla\phi_j,\nabla\phi_j)_p-\frac{\mathscr{F}(R_j)}{E_j}c_I(u_j,\phi_j)-\int_{\partial D_p\setminus I}g\phi_j\mathscr{K}_j\nabla\phi_j\cdot\widehat{n}_p\,ds-g(f_{p,j},\psi)_p\Big]\\
&-\frac{\mathscr{F}(R_j)}{E_j}\Big[\nu(\nabla u_j,\nabla u_j)_f+\sum_i\int_I\eta_{i,j}(u_j\cdot\widehat{\tau}_i)(u_j\cdot\widehat{\tau}_i)\,ds-\int_{\partial D_f\setminus I}\nu(\nabla u_j\cdot\widehat{n}_f)\cdot u_j\,ds-(f_{f,j},u_j)_f\\
&+\int_{\partial D_f\setminus I}p_j(u_j\cdot\widehat{n}_f)\,ds+g(\mathscr{K}_j\nabla\phi_j,\nabla\phi_j)_p-\int_{\partial D_p\setminus I}g\phi_j\mathscr{K}_j\nabla\phi_j\cdot\widehat{n}_p\,ds-g(f_{p,j},\psi)_p\Big]\\
&+\Big[1-\frac{\mathscr{F}(R_j)}{E_j}\Big]\Big|\int_{D_f}f_{f,j}\cdot u_j dx+\int_{D_p}gf_{p,j}\cdot\phi_j dx\\
&+\int_{\partial D_f\setminus I}\nu a_j\cdot(\nabla u_j\cdot\widehat{n}_f)\,ds-\int_{\partial D_f\setminus I}a_j\cdot\widehat{n}_f p_j\,ds+\int_{\partial D_p\setminus I}gb_j\mathscr{K}_j\nabla\phi_j\cdot\widehat{n}_p\,ds\Big|.
\end{aligned}
$$

Note that all the additional terms above amount to adding zero to (1.8). We next present two unconditionally stable ensemble methods with shared coefficient matrix across different realizations and time steps for solving the Stokes-Darcy model, based on this reformulation. We will also incorporate the rotational pressure correction method [8, 15, 50] to decouple the the computation of the pressure from the velocity which also makes it possible to decouple all the components of the velocity. The main advantage of the rotational pressure correction projection method for the Navier-Stokes/Stokes equations is that it alleviates the incompressiblity constraint, decouples the pressure from velocity, and decouples all the components of the velocity field. But for the Stokes-Darcy system considered here, the components of the velocity field are difficult to decouple due to the term $\sum_i\int_I\eta_i(u\cdot\widehat{\tau}_i)(v\cdot\widehat{\tau}_i)\,ds$ from the interface condition, [36, 37]. With the help of the gPAV approach, we will lag this term to previous time steps so that all the components of the unknown velocity field are fully decoupled while still being able to construct unconditionally stable ensemble schemes.

Define the function spaces:

$$
\begin{aligned}
&\text{Velocity: } X_f:=\big(H^1(D_f)\big)^d,\\
&\text{Pressure: } Q_f:=L^2(D_f),\quad Y_f^0:=\{q\in H^1(D_f):q=0\text{ on }I\},
\end{aligned}
$$

Hydraulic Head: $X_p := H^1(D_p)$.

The first order pressure correction ensemble scheme based on the Backward Euler time stepping and the gPAV approach we propose is

ALGORITHM 1.1. *Given u_j^n, ϕ_j^n, and p_j^n, find $w_j^{n+1}, u_j^{n+1} \in X_f$, $r_j^{n+1} \in Y_f^0$, $p_j^{n+1} \in Q_f$, and $\phi_j^{n+1} \in X_p$ satisfying*

$$\left(\frac{w_j^{n+1}-u_j^n}{\Delta t}, v\right)_f + \nu(\nabla w_j^{n+1}, \nabla v)_f + \xi_j^{n+1}\sum_i \int_I \eta_{i,j}(u_j^n\cdot\widehat{\tau}_i)(v\cdot\widehat{\tau}_i)\, ds - \left(p_j^n, \nabla\cdot v\right)_f \tag{1.9}$$
$$+ \xi_j^{n+1} c_I(v,\phi_j^n) - \int_{\partial D_f\setminus I} \nu(\nabla w_j^{n+1}\cdot\widehat{n}_f)\cdot v\, ds + \int_{\partial D_f\setminus I} p_j^n v\cdot\widehat{n}_f\, ds = (f_{f,j}^{n+1}, v)_f, \qquad \forall v\in X_f,$$

$$\begin{cases} (\nabla r_j^{n+1}, \nabla q)_f + \frac{1}{\Delta t}(\nabla\cdot w_j^{n+1}, q)_f = 0, \quad r_j^{n+1}|_I = 0, \quad \frac{\partial r_j^{n+1}}{\partial \widehat{n}_f}|_{\partial D_f\setminus I} = 0, \qquad \forall q\in Y_f^0, \\ (u_j^{n+1}, v)_f = (w_j^{n+1}, v)_f - \Delta t(\nabla r_j^{n+1}, v)_f, \qquad \forall v\in X_f, \\ p_j^{n+1} = r_j^{n+1} + p_j^n - \varepsilon\nabla\cdot w_j^{n+1}, \end{cases} \tag{1.10}$$

$$gS_0\left(\frac{\phi_j^{n+1}-\phi_j^n}{\Delta t}, \psi\right)_p + g(\bar{\mathscr{K}}\nabla\phi_j^{n+1}, \nabla\psi)_p + g((\mathscr{K}_j-\bar{\mathscr{K}})\nabla\phi_j^n, \nabla\psi)_p - \xi_j^{n+1}c_I(u_j^n,\psi) \tag{1.11}$$
$$- \int_{\partial D_p\setminus I} g\psi\bar{\mathscr{K}}\nabla\phi_j^{n+1}\cdot\widehat{n}_p\, ds - \int_{\partial D_p\setminus I} g\psi(\mathscr{K}_j-\bar{\mathscr{K}})\nabla\phi_j^n\cdot\widehat{n}_p\, ds = g(f_{p,j}^{n+1},\psi)_p, \qquad \forall\psi\in X_p,$$

$$\xi_j^{n+1} = \frac{\mathscr{F}(R_j^{n+1})}{E(\tilde{\tilde{u}}_j^{n+1}, \tilde{\tilde{\phi}}_j^{n+1})}, \tag{1.12}$$

$$E(\tilde{\tilde{u}}_j^{n+1}, \tilde{\tilde{\phi}}_j^{n+1}) = \frac{1}{2}\|\tilde{\tilde{u}}_j^{n+1}\|^2 + \frac{gS_0}{2}\|\tilde{\tilde{\phi}}_j^{n+1}\|^2 + C_0, \tag{1.13}$$

$$\frac{\mathscr{F}(R_j^{n+1}) - \mathscr{F}(R_j^n)}{\Delta t} = \left(\frac{w_j^{n+1}-u_j^n}{\Delta t}, w_j^{n+1}\right)_f + gS_0\left(\frac{\phi_j^{n+1}-\phi_j^n}{\Delta t}, \phi_j^{n+1}\right)_p \tag{1.14}$$
$$+ \Big[\nu(\nabla w_j^{n+1}, \nabla w_j^{n+1})_f + \xi_j^{n+1}\sum_i\int_I \eta_{i,j}(u_j^n\cdot\widehat{\tau}_i)(w_j^{n+1}\cdot\widehat{\tau}_i)\, ds - \left(p_j^n, \nabla\cdot w_j^{n+1}\right)_f$$
$$+ \xi_j^{n+1}c_I(w_j^{n+1},\phi_j^n) - \int_{\partial D_f\setminus I}\nu(\nabla w_j^{n+1}\cdot\widehat{n}_f)\cdot a_j^{n+1}\, ds + \int_{\partial D_f\setminus I} p_j^n(w_j^{n+1}\cdot\widehat{n}_f)\, ds$$
$$- (f_{f,j}, w_j^{n+1})_f + g(\bar{\mathscr{K}}\nabla\phi_j^{n+1}, \nabla\phi_j^{n+1})_p + g((\mathscr{K}_j-\bar{\mathscr{K}})\nabla\phi_j^n, \nabla\phi_j^{n+1})_p - \xi_j^{n+1}c_I(u_j^n,\phi_j^{n+1})$$
$$- \int_{\partial D_p\setminus I} gb_j^{n+1}\bar{\mathscr{K}}\nabla\phi_j^{n+1}\cdot\widehat{n}_p\, ds - \int_{\partial D_p\setminus I} gb_j^{n+1}(\mathscr{K}_j-\bar{\mathscr{K}})\nabla\phi_j^n\cdot\widehat{n}_p\, ds - g(f_{p,j}^{n+1},\phi_j^{n+1})_p\Big]$$
$$- \xi_j^{n+1}\Big[\nu(\nabla\tilde{\tilde{u}}_j^{n+1}, \nabla\tilde{\tilde{u}}_j^{n+1})_f + \sum_i\int_I \eta_{i,j}(\tilde{\tilde{u}}_j^{n+1}\cdot\widehat{\tau}_i)(\tilde{\tilde{u}}_j^{n+1}\cdot\widehat{\tau}_i)\, ds$$
$$- \int_{\partial D_f\setminus I}\nu(\nabla\tilde{\tilde{u}}_j^{n+1}\cdot\widehat{n}_f)\cdot a_j^{n+1}\, ds + \int_{\partial D_f\setminus I}\tilde{\tilde{p}}_j^{n+1}(a_j^{n+1}\cdot\widehat{n}_f)\, ds - (f_{f,j}^{n+1}, \tilde{\tilde{u}}_j^{n+1})_f$$
$$+ g(\mathscr{K}_j\nabla\tilde{\tilde{\phi}}_j^{n+1}, \nabla\tilde{\tilde{\phi}}_j^{n+1})_p - \int_{\partial D_p\setminus I} gb_j^{n+1}\mathscr{K}_j\nabla\tilde{\tilde{\phi}}_j\cdot\widehat{n}_p\, ds - g(f_{p,j}^{n+1}, \tilde{\tilde{\phi}}_j^{n+1})_p\Big]$$
$$+ \left[1-\xi_j^{n+1}\right]\Big|\int_{D_f} f_{f,j}\cdot\tilde{\tilde{u}}_j^{n+1}dx + \int_{D_p} gf_{p,j}\cdot\tilde{\tilde{\phi}}_j^{n+1}dx$$
$$+ \int_{\partial D_f\setminus I}\nu a_j^{n+1}\cdot(\nabla\tilde{\tilde{u}}_j^{n+1}\cdot\widehat{n}_f)\, ds - \int_{\partial D_f\setminus I} a_j^{n+1}\cdot\widehat{n}_f\tilde{\tilde{p}}_j^{n+1}\, ds + \int_{\partial D_p\setminus I} gb_j^{n+1}\mathscr{K}_j\nabla\tilde{\tilde{\phi}}_j^{n+1}\cdot\widehat{n}_p\, ds\Big|.$$

Here $\tilde{\tilde{u}}_j^{n+1}$, $\tilde{\tilde{p}}_j^{n+1}$, and $\tilde{\tilde{\phi}}_j^{n+1}$ are first order approximations of u_j^{n+1}, p_j^{n+1}, and ϕ_j^{n+1} that will be defined later.

Define

$$\overset{*}{v}^{n+3/2} = \frac{3}{2}v^{n+1} - \frac{1}{2}v^n, \qquad \tilde{v}^{n+1} = 2v^n - v^{n-1}. \tag{1.15}$$

The second order ensemble scheme based on the second order backward differentiation formula (BDF2) time stepping and the gPAV approach we propose is

ALGORITHM 1.2. *Given u_j^{n-1}, u_j^n, ϕ_j^{n-1}, and ϕ_j^n, for $n \geq 1$, find $w_j^{n+1}, u_j^{n+1} \in X_f$, $r_j^{n+1} \in Y_f^0$, $p_j^{n+1} \in Q_f$, and $\phi_j^{n+1} \in X_p$ satisfying*

$$\begin{aligned}
&\left(\frac{3w_j^{n+1} - 4u_j^n + u_j^{n-1}}{2\Delta t}, v\right)_f + \nu(\nabla w_j^{n+1}, \nabla v)_f + \xi_j^{n+1}\sum_i \int_I \eta_{i,j}(\tilde{u}_j^{n+1}\cdot\widehat{\tau}_i)(v\cdot\widehat{\tau}_i)\, ds - \left(\tilde{p}_j^{n+1}, \nabla\cdot v\right)_f \\
&\quad + \xi_j^{n+1} c_I(v, \tilde{\phi}_j^{n+1}) - \int_{\partial D_f\setminus I} \nu(\nabla w_j^{n+1}\cdot\widehat{n}_f)\cdot v\, ds + \int_{\partial D_f\setminus I} \tilde{p}_j^{n+1} v\cdot\widehat{n}_f\, ds = (f_{f,j}^{n+1}, v)_f, \quad \forall v \in X_f,
\end{aligned} \tag{1.16}$$

$$\begin{cases}
(\nabla r_j^{n+1}, \nabla q) + \frac{3}{2\Delta t}(\nabla\cdot w_j^{n+1}, q) = 0, \quad r_j^{n+1}|_I = 0, \quad \frac{\partial r_j^{n+1}}{\partial \widehat{n}_f}|_{\partial D_f\setminus I} = 0, \qquad \forall q \in Y_f^0, \\
(u_j^{n+1}, v)_f = (w_j^{n+1}, v)_f - \frac{2}{3}\Delta t(\nabla r_j^{n+1}, v)_f, \qquad \forall v \in X_f, \\
p_j^{n+1} = r_j^{n+1} + 2p_j^n - p_j^{n-1} - \varepsilon\nabla\cdot w_j^{n+1},
\end{cases} \tag{1.17}$$

$$\begin{aligned}
&gS_0\left(\frac{3\phi_j^{n+1} - 4\phi_j^n + \phi_j^{n-1}}{2\Delta t}, \psi\right)_p + g(\bar{\mathscr{K}}\nabla\phi_j^{n+1}, \nabla\psi)_p + g((\mathscr{K}_j - \bar{\mathscr{K}})\nabla\tilde{\phi}_j^{n+1}, \nabla\psi)_p - \xi_j^{n+1} c_I(\tilde{u}_j^{n+1}, \psi) \\
&\quad - \int_{\partial D_p\setminus I} g\psi\bar{\mathscr{K}}\nabla\phi_j^{n+1}\cdot\widehat{n}_p\, ds - \int_{\partial D_p\setminus I} g\psi(\mathscr{K}_j - \bar{\mathscr{K}})\nabla\tilde{\phi}_j^{n+1}\cdot\widehat{n}_p\, ds = g(f_{p,j}^{n+1}, \psi)_p, \quad \forall \psi \in X_p,
\end{aligned} \tag{1.18}$$

$$\xi_j^{n+1} = \frac{\mathscr{F}(\overset{*}{R}_j^{n+3/2})}{E(\tilde{\tilde{u}}_j^{n+3/2}, \tilde{\tilde{\phi}}_j^{n+3/2})}, \tag{1.19}$$

$$E(\tilde{\tilde{u}}_j^{n+3/2}, \tilde{\tilde{\phi}}_j^{n+3/2}) = \frac{1}{2}\|\tilde{\tilde{u}}_j^{n+3/2}\|^2 + \frac{gS_0}{2}\|\tilde{\tilde{\phi}}_j^{n+3/2}\|^2 + C_0, \tag{1.20}$$

$$\begin{aligned}
\frac{\mathscr{F}(\overset{*}{R}_j^{n+3/2}) - \mathscr{F}(\overset{*}{R}_j^{n+1/2})}{\Delta t} &= \left(\frac{3w_j^{n+1} - 4u_j^n + u_j^{n-1}}{2\Delta t}, w_j^{n+1}\right)_f + gS_0\left(\frac{3\phi_j^{n+1} - 4\phi_j^n + \phi_j^{n-1}}{2\Delta t}, \phi_j^{n+1}\right)_p \\
&\quad + \Big[\nu(\nabla w_j^{n+1}, \nabla w_j^{n+1})_f + \xi_j^{n+1}\sum_i \int_I \eta_{i,j}(\tilde{u}_j^{n+1}\cdot\widehat{\tau}_i)(w_j^{n+1}\cdot\widehat{\tau}_i)\, ds - \left(\tilde{p}_j^{n+1}, \nabla\cdot w_j^{n+1}\right)_f \\
&\qquad + \xi_j^{n+1} c_I(w_j^{n+1}, \tilde{\phi}_j^{n+1}) - \int_{\partial D_f\setminus I} \nu(\nabla u_j^{n+1}\cdot\widehat{n}_f)\cdot a_j^{n+1}\, ds + \int_{\partial D_f\setminus I} \tilde{p}_j^{n+1}(w_j^{n+1}\cdot\widehat{n}_f)\, ds \\
&\qquad - (f_{f,j}, w_j^{n+1})_f + g(\bar{\mathscr{K}_j}\nabla\phi_j^{n+1}, \nabla\phi_j^{n+1})_p + g((\mathscr{K}_j - \bar{\mathscr{K}})\nabla\tilde{\phi}_j^{n+1}, \nabla\phi_j^{n+1})_p - \xi_j^{n+1} c_I(\tilde{u}_j^{n+1}, \phi_j^{n+1}) \\
&\qquad - \int_{\partial D_p\setminus I} g b_j^{n+1}\bar{\mathscr{K}}\nabla\phi_j^{n+1}\cdot\widehat{n}_p\, ds - \int_{\partial D_p\setminus I} g b_j^{n+1}(\mathscr{K}_j - \bar{\mathscr{K}})\nabla\tilde{\phi}_j^{n+1}\cdot\widehat{n}_p\, ds - g(f_{p,j}^{n+1}, \phi_j^{n+1})_p\Big] \\
&\quad - \xi_j^{n+1}\Big[\nu(\nabla\tilde{\tilde{u}}_j^{n+1}, \nabla\tilde{\tilde{u}}_j^{n+1})_f + \sum_i \int_I \eta_{i,j}(\tilde{\tilde{u}}_j^{n+1}\cdot\widehat{\tau}_i)(\tilde{\tilde{u}}_j^{n+1}\cdot\widehat{\tau}_i)\, ds \\
&\qquad - \int_{\partial D_f\setminus I} \nu(\nabla\tilde{\tilde{u}}_j^{n+1}\cdot\widehat{n}_f)\cdot a_j^{n+1}\, ds + \int_{\partial D_f\setminus I} \tilde{\tilde{p}}_j^{n+1}(a_j^{n+1}\cdot\widehat{n}_f)\, ds - (f_{f,j}^{n+1}, \tilde{\tilde{u}}_j^{n+1})_f \\
&\qquad + g(\mathscr{K}_j\nabla\tilde{\tilde{\phi}}_j^{n+1}, \nabla\tilde{\tilde{\phi}}_j^{n+1})_p - \int_{\partial D_p\setminus I} g b_j^{n+1}\mathscr{K}_j\nabla\tilde{\tilde{\phi}}_j\cdot\widehat{n}_p\, ds - g(f_{p,j}^{n+1}, \tilde{\tilde{\phi}}_j^{n+1})_p\Big] \\
&\quad + \left[1 - \xi_j^{n+1}\right]\Big|\int_{D_f} f_{f,j}\cdot\tilde{\tilde{u}}_j^{n+1} dx + \int_{D_p} g f_{p,j}\cdot\tilde{\tilde{\phi}}_j^{n+1} dx \\
&\quad + \int_{\partial D_f\setminus I} \nu a_j^{n+1}\cdot(\nabla\tilde{\tilde{u}}_j^{n+1}\cdot\widehat{n}_f)\, ds - \int_{\partial D_f\setminus I} a_j^{n+1}\cdot\widehat{n}_f\tilde{\tilde{p}}_j^{n+1}\, ds + \int_{\partial D_p\setminus I} g b_j^{n+1}\mathscr{K}_j\nabla\tilde{\tilde{\phi}}_j^{n+1}\cdot\widehat{n}_p\, ds\Big|.
\end{aligned} \tag{1.21}$$

Here $\tilde{\tilde{u}}_j^{n+3/2}$, $\tilde{\tilde{u}}_j^{n+1}$, $\tilde{\tilde{p}}_j^{n+1}$, $\tilde{\tilde{\phi}}_j^{n+3/2}$ *and* $\tilde{\tilde{\phi}}_j^{n+1}$ *are second order approximations of* $u_j^{n+3/2}$, u_j^{n+1}, p_j^{n+1}, $\phi_j^{n+3/2}$, *and* ϕ_j^{n+1} *that will be defined later.*

The rest of the paper is outlined here. In Section 2, we prove the unconditionally long time stability of the proposed algorithms. Section 3 presents an efficient way to implement our numerical algorithms. Section 4 numerically tests the proposed algorithms in terms of effectiveness and efficiency. Conclusion remarks are presented in Section 5.

2. Stability of the ensemble algorithms.

2.1. Backward Euler. THEOREM 2.1. *With homogeneous Dirichlet boundary conditions on* $\partial D_f \backslash I$ *and* $\partial D_p \backslash I$, *and forcing terms equal to zero, Algorithm* (1.1) *is unconditionally stable with respect to the modified energy* $\mathscr{F}(R_j)$.

Proof.
Set v to w_j^{n+1} in (1.9), set ψ to ϕ_j^{n+1} in (1.11), and add these to (1.14). Then one gets

$$\mathscr{F}(R_j^{n+1}) - \mathscr{F}(R_j^n) = -\Delta t \frac{\mathscr{F}(R_j^{n+1})}{E(\tilde{\tilde{u}}_j^{n+1}, \tilde{\tilde{\phi}}_j^{n+1})} \left[\nu \|\nabla \tilde{\tilde{u}}_j^{n+1}\|_f^2 + \sum_i \int_I \eta_{i,j} (\tilde{\tilde{u}}_j^{n+1} \cdot \widehat{\tau}_i)(\tilde{\tilde{u}}_j^{n+1} \cdot \widehat{\tau}_i)\, ds \right. \tag{2.1}$$
$$\left. + g \left(\mathscr{K}_j \nabla \tilde{\tilde{\phi}}_j^{n+1}, \nabla \tilde{\tilde{\phi}}_j^{n+1} \right)_p \right] + \left[1 - \frac{\mathscr{F}(R_j^{n+1})}{E(\tilde{\tilde{u}}_j^{n+1}, \tilde{\tilde{\phi}}_j^{n+1})} \right] |S_j^{n+1}| \Delta t + \frac{\mathscr{F}(R_j^{n+1})}{E(\tilde{\tilde{u}}_j^{n+1}, \tilde{\tilde{\phi}}_j^{n+1})} S_j^{n+1} \Delta t,$$

where

$$S_j^{n+1} = \int_{D_f} f_{f,j}^{n+1} \cdot \tilde{\tilde{u}}_j^{n+1} dx + \int_{D_p} g f_{p,j}^{n+1} \cdot \tilde{\tilde{\phi}}_j^{n+1} dx$$
$$+ \int_{\partial D_f \backslash I} \nu a_j^{n+1} \cdot (\nabla \tilde{\tilde{u}}_j^{n+1} \cdot \widehat{n}_f)\, ds - \int_{\partial D_f \backslash I} a_j^{n+1} \cdot \widehat{n}_f \tilde{\tilde{p}}_j^{n+1}\, ds + \int_{\partial D_p \backslash I} g b_j^{n+1} \mathscr{K}_j \nabla \tilde{\tilde{\phi}}_j^{n+1} \cdot \widehat{n}_p\, ds.$$

Solving for $\mathscr{F}(R_j^{n+1})$ gives

$$\mathscr{F}(R_j^{n+1}) = \frac{\mathscr{F}(R_j^n) + |S_j^{n+1}| \Delta t}{1 + \frac{\Delta t}{E(\tilde{\tilde{u}}_j^{n+1}, \tilde{\tilde{\phi}}_j^{n+1})} \left[H_j^{n+1} + (|S_j^{n+1}| - S_j^{n+1}) \right]}, \tag{2.2}$$

where

$$H_j^{n+1} = \nu \|\nabla \tilde{\tilde{u}}_j^{n+1}\|_f^2 + \sum_i \int_I \eta_{i,j} (\tilde{\tilde{u}}_j^{n+1} \cdot \widehat{\tau}_i)^2\, ds + g \left(\mathscr{K}_j \nabla \tilde{\tilde{\phi}}_j^{n+1}, \nabla \tilde{\tilde{\phi}}_j^{n+1} \right)_p > 0.$$

If $a_j = 0, b_j = 0$, $f_{f,j} = 0$ and $f_{p,j} = 0$, then $S_j^{n+1} = 0$ and

$$\mathscr{F}(R_j^{n+1}) = \frac{\mathscr{F}(R_j^n)}{1 + \frac{\Delta t}{E(\tilde{\tilde{u}}_j^{n+1}, \tilde{\tilde{\phi}}_j^{n+1})} H_j^{n+1}}. \tag{2.3}$$

Note the denominator in (2.3) is greater than or equal to 1. By definition (1.4), if $R_j^0 > 0$, then $\mathscr{F}(R_j^0) > 0$. In fact R_j^0 would be initialized as $\mathscr{G}(E(u_j^0(x), \phi_j^0(x)))$, which by definition (1.5) is guaranteed positive. Then by induction, for any timestep n we have $\mathscr{F}(R_j^{n+1}) > 0$ and

$$0 < \mathscr{F}(R_j^{n+1}) \leq \mathscr{F}(R_j^n), \qquad n \geq 0. \tag{2.4}$$

□

2.2. BDF2. THEOREM 2.2. *With homogeneous Dirichlet boundary conditions on $\partial D_f \backslash I$ and $\partial D_p \backslash I$, and forcing terms equal to zero, Algorithm* (1.2) *is unconditionally stable with respect to the modified energy $\mathscr{F}(R_j)$ as long as the approximations of $R_j(t)$ at timestep $\frac{3}{2}$ are positive.*

Proof.
Set v to w_j^{n+1} in (1.16), set ψ to ϕ_j^{n+1} in (1.18), and add these to (1.21). Then one gets

$$\mathscr{F}(\overset{*}{R}_j^{n+3/2}) - \mathscr{F}(\overset{*}{R}_j^{n+1/2}) = -\Delta t \frac{\mathscr{F}(\overset{*}{R}_j^{n+3/2})}{E(\tilde{\tilde{u}}_j^{n+\frac{3}{2}}, \tilde{\tilde{\phi}}_j^{n+\frac{3}{2}})} \left[\nu \|\nabla \tilde{\tilde{u}}_j^{n+1}\|_f^2 + \sum_i \int_I \eta_{i,j} (\tilde{\tilde{u}}_j^{n+1} \cdot \widehat{\tau}_i)(\tilde{\tilde{u}}_j^{n+1} \cdot \widehat{\tau}_i)\, ds \right. \tag{2.5}$$
$$\left. + g\left(\mathscr{K}_j \nabla \tilde{\tilde{\phi}}_j^{n+1}, \nabla \tilde{\tilde{\phi}}_j^{n+1} \right)_p \right] + \left[1 - \frac{\mathscr{F}(\overset{*}{R}_j^{n+3/2})}{E(\tilde{\tilde{u}}_j^{n+\frac{3}{2}}, \tilde{\tilde{\phi}}_j^{n+\frac{3}{2}})} \right] |S_j^{n+1}| \Delta t + \frac{\mathscr{F}(\overset{*}{R}_j^{n+3/2})}{E(\tilde{\tilde{u}}_j^{n+\frac{3}{2}}, \tilde{\tilde{\phi}}_j^{n+\frac{3}{2}})} S_j^{n+1} \Delta t,$$

where

$$S_j^{n+1} = \int_{D_f} f_{f,j}^{n+1} \cdot \tilde{\tilde{u}}_j^{n+1} dx + \int_{D_p} g f_{p,j}^{n+1} \cdot \tilde{\tilde{\phi}}_j^{n+1} dx$$
$$+ \int_{\partial D_f \backslash I} \nu a_j^{n+1} \cdot (\nabla \tilde{\tilde{u}}_j^{n+1} \cdot \widehat{n}_f)\, ds - \int_{\partial D_f \backslash I} a_j^{n+1} \cdot \widehat{n}_f \tilde{\tilde{p}}_j^{n+1}\, ds + \int_{\partial D_p \backslash I} g b_j^{n+1} \mathscr{K}_j \nabla \tilde{\tilde{\phi}}_j^{n+1} \cdot \widehat{n}_p\, ds.$$

Solving for $\mathscr{F}(\overset{*}{R}_j^{n+3/2})$ gives

$$\mathscr{F}(\overset{*}{R}_j^{n+3/2}) = \frac{\mathscr{F}(\overset{*}{R}_j^{n+1/2}) + |S_j^{n+1}| \Delta t}{1 + \Delta t / E(\tilde{\tilde{u}}_j^{n+\frac{3}{2}}, \tilde{\tilde{\phi}}_j^{n+\frac{3}{2}}) \left[H_j^{n+1} + (|S_j^{n+1}| - S_j^{n+1}) \right]}. \tag{2.6}$$

where

$$H_j^{n+1} = \nu \|\nabla \tilde{\tilde{u}}_j^{n+1}\|_f^2 + \sum_i \int_I \eta_{i,j} (\tilde{\tilde{u}}_j^{n+1} \cdot \widehat{\tau}_i)^2\, ds + g\left(\mathscr{K}_j \nabla \tilde{\tilde{\phi}}_j^{n+1}, \nabla \tilde{\tilde{\phi}}_j^{n+1} \right)_p.$$

If $a_j = 0, b_j = 0$, $f_{f,j} = 0$ and $f_{p,j} = 0$, then $S_j^{n+1} = 0$ and

$$\mathscr{F}(\overset{*}{R}_j^{n+3/2}) = \frac{\mathscr{F}(\overset{*}{R}_j^{n+1/2})}{1 + \Delta t / E(\tilde{\tilde{u}}_j^{n+\frac{3}{2}}, \tilde{\tilde{\phi}}_j^{n+\frac{3}{2}}) H_j^{n+1}}. \tag{2.7}$$

The denominator above is greater than or equal to 1. Now by definition (1.4), if it's ensured the approximation of $R_j(t)$ at timestep 3/2 is positive, i.e. $\overset{*}{R}_j^{3/2} > 0$, then $\mathscr{F}(\overset{*}{R}_j^{3/2}) > 0$. Then by induction for any timestep $n \geq 1$, $\mathscr{F}(\overset{*}{R}_j^{n+3/2}) > 0$ and

$$0 < \mathscr{F}(\overset{*}{R}_j^{n+3/2}) \leq \mathscr{F}(\overset{*}{R}_j^{n+1/2}), \qquad n \geq 1. \tag{2.8}$$

□

Note that for the choice of $\mathscr{F}(\chi) = \chi^2 \geq 0$ for all $\chi \in (-\infty, \infty)$, (2.8) and unconditional stability will hold regardless of whether $\overset{*}{R}_j^{1/2} > 0$.

3. Implementation. Let X_f^h be a finite element space approximating X_f with spatial resolution h, $X_f^{h,0} := \{v \in X_f^h : v = 0 \text{ on } \partial D_f \backslash I\}$. The space Q_f^h, $Y_f^{h,0} \subset Y_f^0$, X_p^h and $X_p^{h,0}$ are defined similarly.

3.1. Backward Euler. To efficiently implement Algorithm (1.1), we proceed in the following manner. Assume

$$w_{j,h}^{n+1} = \hat{w}_{j,h}^{n+1} + \xi_{j,h}^{n+1} \breve{w}_{j,h}^{n+1}, \qquad \phi_{j,h}^{n+1} = \hat{\phi}_{j,h}^{n+1} + \xi_{j,h}^{n+1} \breve{\phi}_{j,h}^{n+1} \tag{3.1}$$

gPAV-BE-Projection: solve the following four subproblems for $\hat{w}_{j,h}^{n+1}$, $\hat{\phi}_{j,h}^{n+1}$, $\breve{w}_{j,h}^{n+1}$,$\breve{\phi}_{j,h}^{n+1}$ respectively.
(BE sub-problem 1): Find $\hat{w}_{j,h}^{n+1} \in X_f^h$ satisfying $\forall\ v_h \in X_f^{h,0}$,

$$\begin{cases} \frac{1}{\Delta t}\left(\hat{w}_{j,h}^{n+1}, v_h\right)_f + \nu(\nabla \hat{w}_{j,h}^{n+1}, \nabla v_h)_f = (f_{f,j}^{n+1}, v_h)_f + \frac{1}{\Delta t}\left(u_{j,h}^n, v_h\right)_f + \left(p_{j,h}^n, \nabla \cdot v_h\right)_f, \\ \hat{w}_{j,h}^{n+1}|_{\partial D_f \setminus I} = a_{j,h}^{n+1}. \end{cases}$$

(BE sub-problem 2): Find $\hat{\phi}_{j,h}^{n+1} \in X_p^h$ satisfying $\forall\ \psi_h \in X_p^{h,0}$,

$$\begin{cases} \frac{gS_0}{\Delta t}\left(\hat{\phi}_{j,h}^{n+1}, \psi_h\right)_p + g(\bar{\mathscr{K}}\nabla \hat{\phi}_{j,h}^{n+1}, \nabla \psi_h)_p \\ \qquad = g(f_{p,j}^{n+1}, \psi_h)_p + \frac{gS_0}{\Delta t}\left(\phi_{j,h}^n, \psi_h\right)_p - g((\mathscr{K}_j - \bar{\mathscr{K}})\nabla(\phi_{j,h}^n), \nabla \psi_h)_p, \\ \hat{\phi}_{j,h}^{n+1}|_{\partial D_p \setminus I} = b_{j,h}^{n+1}. \end{cases}$$

(Be sub-problem 3): Find $\breve{w}_{j,h}^{n+1} \in X_f^h$ satisfying $\forall\ v_h \in X_f^{h,0}$,

$$\begin{cases} \frac{1}{\Delta t}\left(\breve{w}_{j,h}^{n+1}, v_h\right)_f + \nu(\nabla \breve{w}_{j,h}^{n+1}, \nabla v_h)_f = -\sum_i \int_I \eta_{i,j}(u_{j,h}^n \cdot \widehat{\tau}_i)(v_h \cdot \widehat{\tau}_i)\ ds - c_I(v_h, \phi_{j,h}^n), \\ \breve{w}_{j,h}^{n+1}|_{\partial D_f \setminus I} = 0. \end{cases}$$

(BE sub-problem 4): Find $\breve{\phi}_{j,h}^{n+1} \in X_p^h$ satisfying $\forall\ \psi_h \in X_p^{h,0}$,

$$\begin{cases} \frac{gS_0}{\Delta t}\left(\breve{\phi}_{j,h}^{n+1}, \psi_h\right)_p + g(\bar{\mathscr{K}}\nabla \breve{\phi}_{j,h}^{n+1}, \nabla \psi_h)_p = c_I(u_{j,h}^n, \psi_h), \\ \breve{\phi}_{j,h}^{n+1}|_{\partial D_p \setminus I} = 0. \end{cases}$$

We use the following approximations,

$$\tilde{\tilde{v}}_j^{n+1} = \hat{v}_j^{n+1} + \breve{v}_j^{n+1}.$$

then update $\xi_{j,h}^{n+1}$ as

$$\xi_{j,h}^{n+1} = \frac{\mathscr{F}(R_{j,h}^n) + |S_{j,h}^{n+1}|\Delta t}{E(\tilde{\tilde{u}}_{j,h}^{n+1}, \tilde{\tilde{\phi}}_{j,h}^{n+1}) + \Delta t\left[H_{j,h}^{n+1} + (|S_{j,h}^{n+1}| - S_{j,h}^{n+1})\right]}, \tag{3.2}$$

where

$$H_{j,h}^{n+1} = \nu\|\nabla \tilde{\tilde{u}}_{j,h}^{n+1}\|_f^2 + \sum_i \int_I \eta_{i,j}(\tilde{\tilde{u}}_{j,h}^{n+1} \cdot \widehat{\tau}_i)^2\ ds + g\left(\mathscr{K}_j \nabla \tilde{\tilde{\phi}}_{j,h}^{n+1}, \nabla \tilde{\tilde{\phi}}_{j,h}^{n+1}\right)_p > 0.$$

$$\begin{aligned} S_{j,h}^{n+1} &= \int_{D_f} f_{f,j,h}^{n+1} \cdot \tilde{\tilde{u}}_{j,h}^{n+1} dx + \int_{D_p} g f_{p,j,h}^{n+1} \cdot \tilde{\tilde{\phi}}_{j,h}^{n+1} dx \\ &\quad + \int_{\partial D_f \setminus I} \nu a_{j,h}^{n+1} \cdot (\nabla \tilde{\tilde{u}}_{j,h}^{n+1} \cdot \widehat{n}_f)\, ds - \int_{\partial D_f \setminus I} a_{j,h}^{n+1} \cdot \widehat{n}_f \tilde{\tilde{p}}_{j,h}^{n+1}\, ds + \int_{\partial D_p \setminus I} g b_{j,h}^{n+1} \mathscr{K}_j \nabla \tilde{\tilde{\phi}}_{j,h}^{n+1} \cdot \widehat{n}_p\, ds. \end{aligned}$$

Notice $\xi_{j,h}^{n+1}$ is updated via a linear equation and is very direct.

Once we have $\xi_{j,h}^{n+1}$, using the solutions of above four sub-problems we can get $w_{j,h}^{n+1}, \phi_{j,h}^{n+1}$. The final solution $(u_{j,h}^{n+1}, p_{j,h}^{n+1})$ can be obtained by first solving the pressure Poisson equation for $r_{j,h}^{n+1} \in Y_f^{h,0}$

$$\begin{cases} (\nabla r_{j,h}^{n+1}, \nabla q_h) = -\frac{1}{\Delta t}(\nabla \cdot w_{j,h}^{n+1}, q_h),\ \forall q_h \in Y_f^{h,0}, \\ r_{j,h}^{n+1}|_I = 0,\ \frac{\partial r_{j,h}^{n+1}}{\partial \widehat{n}_f}|_{\partial D_f \setminus I} = 0, \end{cases}$$

then updating $u_{j,h}^{n+1}$ and $p_{j,h}^{n+1}$ by

$$(u_{j,h}^{n+1}, v_h)_f = (w_{j,h}^{n+1}, v_h)_f - \Delta t(\nabla r_{j,h}^{n+1}, v_h)_f,\ \forall v_h \in X_f^h,$$

$$p_{j,h}^{n+1} = r_{j,h}^{n+1} + p_{j,h}^{n} - \varepsilon \nabla \cdot w_{j,h}^{n+1}.$$

For the next timestep iteration, we update

$$R_{j,h}^{n+1} = \mathscr{G}\left(\xi_{j,h}^{n+1} E(\tilde{\tilde{u}}_{j,h}^{n+1}, \tilde{\tilde{\phi}}_{j,h}^{n+1})\right) \tag{3.3}$$

and proceed to the next timestep iteration. Since $\xi_{j,h}^{n+1}$ is a ratio of the SAV to itself, we should expect the result to be close to one.

THEOREM 3.1. *The scalar $\xi_{j,h}^{n+1}$ in (3.2) and $R_{j,h}^{n+1}$ in (3.3) are guaranteed to be positive at all timesteps.*

Proof. By definition (1.4), $\mathscr{F}(R_{j,h}^{0}) > 0$ so long as $R_{j,h}^{0} = \mathscr{G}(E(u_{j,h}^{0}, \phi_{j,h}^{0})) > 0$. The energy function $E(u, \phi)$ is always positive. Since $|S_{j,h}^{n+1}| - S_{j,h}^{n+1} \geq 0$, the initially computed $\xi_{j,h}^{n+1}$ is ensured positive. Then by induction, $\xi_{j,h}^{n+1}$ at any timestep is guaranteed positive.

Once it's ensured $\xi_{j,h}^{n+1} > 0$, from the definition (1.5) we can guarantee $R_{j,h}^{n+1}$ in (3.3) is positive.
∎

3.2. BDF2. For Algorithm (1.2), we develop an efficient implementation with the same approach. Note solving Algorithm (1.2) is equivalent the following:

gPAV-BDF2-Projection: solve the following four subproblems for $\hat{w}_{j,h}^{n+1}$, $\hat{\phi}_{j,h}^{n+1}$, $\breve{w}_{j,h}^{n+1}$, $\breve{\phi}_{j,h}^{n+1}$ respectively.

(BDF2 sub-problem 1): Find $\hat{w}_{j,h}^{n+1} \in X_f^h$ satisfying $\forall\ v_h \in X_f^{h,0}$,

$$\begin{cases} \dfrac{3}{2\Delta t}\left(\hat{w}_{j,h}^{n+1}, v_h\right)_f + \nu(\nabla \hat{w}_{j,h}^{n+1}, \nabla v_h)_f \\ \qquad = (f_{f,j}^{n+1}, v_h)_f + \dfrac{2}{\Delta t}\left(u_{j,h}^{n}, v_h\right)_f - \dfrac{1}{2\Delta t}\left(u_{j,h}^{n-1}, v_h\right)_f + \left(\tilde{p}_{j,h}^{n+1}, \nabla \cdot v_h\right), \\ \hat{w}_{j,h}^{n+1}|_{\partial D_f \setminus I} = a_{j,h}^{n+1}. \end{cases}$$

(BDF2 sub-problem 2): Find $\hat{\phi}_{j,h}^{n+1} \in X_p^h$ satisfying $\forall\ \psi_h \in X_p^{h,0}$,

$$\begin{cases} \dfrac{3gS_0}{2\Delta t}\left(\hat{\phi}_{j,h}^{n+1}, \psi_h\right)_p + g(\bar{\mathscr{K}} \nabla \hat{\phi}_{j,h}^{n+1}, \nabla \psi_h)_p = g(f_{p,j}^{n+1}, \psi_h)_p + \dfrac{2gS_0}{\Delta t}\left(\phi_{j,h}^{n}, \psi_h\right)_p - \dfrac{gS_0}{2\Delta t}\left(\phi_{j,h}^{n-1}, \psi_h\right)_p \\ \qquad - g((\mathscr{K}_j - \bar{\mathscr{K}})\nabla(\tilde{\phi}_{j,h}^{n+1}, \nabla \psi_h)_p, \\ \hat{\phi}_{j,h}^{n+1}|_{\partial D_p \setminus I} = b_{j,h}^{n+1}. \end{cases}$$

(BDF2 sub-problem 3): Find $\breve{w}_{j,h}^{n+1} \in X_f^h$ satisfying $\forall\ v_h \in X_f^{h,0}$,

$$\begin{cases} \dfrac{3}{2\Delta t}\left(\breve{w}_{j,h}^{n+1}, v_h\right)_f + \nu(\nabla \breve{w}_{j,h}^{n+1}, \nabla v_h)_f = -\displaystyle\sum_i \int_I \eta_{i,j}(\tilde{u}_{j,h}^{n+1} \cdot \widehat{\tau}_i)(v_h \cdot \widehat{\tau}_i)\ ds - c_I(v_h, \tilde{\phi}_{j,h}^{n+1}), \\ \breve{w}_{j,h}^{n+1}|_{\partial D_f \setminus I} = 0. \end{cases}$$

(BDF2 sub-problem 4): Find $\breve{\phi}_{j,h}^{n+1} \in X_p^h$ satisfying $\forall\ \psi_h \in X_p^{h,0}$,

$$\begin{cases} \dfrac{3gS_0}{2\Delta t}\left(\breve{\phi}_{j,h}^{n+1}, \psi_h\right)_p + g(\bar{\mathscr{K}} \nabla \breve{\phi}_{j,h}^{n+1}, \nabla \psi_h)_p = c_I(\tilde{u}_{j,h}^{n+1}, \psi_h), \\ \breve{\phi}_{j,h}^{n+1}|_{\partial D_f \setminus I} = 0. \end{cases}$$

We use the following approximations,

$$\begin{cases} \tilde{\tilde{v}}_{j,h}^{n+1} = \hat{v}_{j,h}^{n+1} + \breve{v}_{j,h}^{n+1}, & (3.4) \\ \tilde{\tilde{v}}_{j,h}^{n+3/2} = \dfrac{3}{2}\tilde{\tilde{v}}_{j,h}^{n+1} - \dfrac{1}{2} v_{j,h}^{n}. & (3.5) \end{cases}$$

We update $\xi_{j,h}^{n+1}$ as

$$\xi_{j,h}^{n+1} = \frac{\mathscr{F}(\overset{*}{R}_{j,h}^{n+1/2}) + |S_{j,h}^{n+1}|\Delta t}{E(\tilde{\tilde{u}}_{j,h}^{n+\frac{3}{2}}, \tilde{\tilde{\phi}}_{j,h}^{n+\frac{3}{2}}) + \Delta t\left[H_{j,h}^{n+1} + (|S_{j,h}^{n+1}| - S_{j,h}^{n+1})\right]}, \tag{3.6}$$

where

$$H_{j,h}^{n+1} = \nu\|\nabla\tilde{\tilde{u}}_{j,h}^{n+1}\|_f^2 + \sum_i \int_I \eta_{i,j}(\tilde{\tilde{u}}_{j,h}^{n+1}\cdot\widehat{\tau}_i)^2\, ds + g\left(\mathscr{K}_j\nabla\tilde{\tilde{\phi}}_{j,h}^{n+1}, \nabla\tilde{\tilde{\phi}}_{j,h}^{n+1}\right)_p, \tag{3.7}$$

$$\begin{aligned} S_{j,h}^{n+1} &= \int_{D_f} f_{f,j,h}^{n+1}\cdot\tilde{\tilde{u}}_{j,h}^{n+1}dx + \int_{D_p} g f_{p,j,h}^{n+1}\cdot\tilde{\tilde{\phi}}_{j,h}^{n+1}dx \\ &+ \int_{\partial D_f\setminus I} \nu a_{j,h}^{n+1}\cdot(\nabla\tilde{\tilde{u}}_{j,h}^{n+1}\cdot\widehat{n}_f)\, ds - \int_{\partial D_f\setminus I} a_{j,h}^{n+1}\cdot\widehat{n}_f\tilde{\tilde{p}}_{j,h}^{n+1}\, ds + \int_{\partial D_p\setminus I} g b_{j,h}^{n+1}\mathscr{K}_j\nabla\tilde{\tilde{\phi}}_{j,h}^{n+1}\cdot\widehat{n}_p\, ds. \end{aligned} \tag{3.8}$$

Once we have $\xi_{j,h}^{n+1}$, using the solutions of above four sub-problems we can get $w_{j,h}^{n+1}, \phi_{j,h}^{n+1}$. The final solution $(u_{j,h}^{n+1}, p_{j,h}^{n+1})$ can be obtained by first solving the pressure Poisson equation for $r_{j,h}^{n+1} \in Y_f^{h,0}$

$$\begin{cases} (\nabla r_{j,h}^{n+1}, \nabla q_h) = -\frac{3}{2\Delta t}(\nabla\cdot w_{j,h}^{n+1}, q_h),\ \forall q_h \in Y_f^{h,0}, \\ r_{j,h}^{n+1}|_I = 0,\ \frac{\partial r_{j,h}^{n+1}}{\partial\widehat{n}_f}|_{\partial D_f\setminus I} = 0, \end{cases}$$

then updating $u_{j,h}^{n+1}$ and $p_{j,h}^{n+1}$ by

$$\begin{aligned} &(u_{j,h}^{n+1}, v_h)_f = (w_{j,h}^{n+1}, v_h)_f - \frac{2}{3}\Delta t(\nabla r_{j,h}^{n+1}, v_h)_f,\ \forall v_h \in X_f^h, \\ &p_{j,h}^{n+1} = r_{j,h}^{n+1} + 2p_{j,h}^n - p_{j,h}^{n-1} - \varepsilon\nabla\cdot w_{j,h}^{n+1}. \end{aligned}$$

For the next timestep iteration, we update $\overset{*}{R}_{j,h}^{n+3/2}$ as follows:

$$\begin{cases} \overset{*}{R}_{j,h}^{n+3/2} = \mathscr{G}\left(\xi_{j,h}^{n+1}E(\tilde{\tilde{u}}_{j,h}^{n+3/2}, \tilde{\tilde{\phi}}_{j,h}^{n+3/2})\right), & (3.9) \\ R_{j,h}^{n+1} = \frac{2}{3}\overset{*}{R}_{j,h}^{n+3/2} + \frac{1}{3}R_{j,h}^n. & (3.10) \end{cases}$$

THEOREM 3.2. *The scalar $\xi_{j,h}^{n+1}$ in (3.6) and $R_{j,h}^{n+1}$ in (3.10) are guaranteed to be positive at timesteps $n \geq 1$.*

Proof. $\overset{*}{R}_{j,h}^{3/2}$ can be easily initialized as

$$\overset{*}{R}_{j,h}^{3/2} = \mathscr{G}\big(E(\overset{*}{u}_{j,h}^{3/2}, \overset{*}{\phi}_{j,h}^{3/2})\big), \tag{3.11}$$

which by definition (1.5) is guaranteed positive. Again by definition (1.4), $\mathscr{F}(\overset{*}{R}_{j,h}^{3/2}) > 0$. The argument for positivity of $\xi_{j,h}^{n+1}$ proceeds identically to that made in the proof of Theorem (2.2).

Once it's ensured $\xi_{j,h}^{n+1} > 0$, again from definition (1.5) we can guarantee $\overset{*}{R}_{j,h}^{3/2}$ in (3.9) is positive. It's also guaranteed $R_{j,h}^1$ is positive when initialized as $\mathscr{G}\big(E(u_{j,h}^1, \phi_{j,h}^1)\big)$. Thus we conclude by induction that $R_{j,h}^{n+1}$ in (3.10) will be positive for all $n \geq 1$.
□

3.3. Algebraic systems. The proposed gPAV-BDF2-Projection ensemble scheme will be compared to other schemes such as the gPAV-BDF2-Projection nonensemble scheme, the gPAV-BDF2 ensemble and nonensemble schemes (without projection) for computational efficiency check in Sec. 4.2. So we state below the difference among these schemes in terms of algebraic systems after spatial discretization by the finite element method.

Let $\mathbf{M}_u$ and $\mathbf{S}_u$ denote the mass and stiffness matrices for u_1. Coefficient matrices corresponding to different schemes in Sec. 4.2 (efficiency test) are listed below.

1. gPAV-BDF2-Projection ensemble: coefficient matrices for solving $\hat{w}_{j,h}^{n+1}$ and $\breve{w}_{j,h}^{n+1}$ are both

$$\mathbf{A}_1 = \begin{bmatrix} \frac{3}{2\Delta t}\mathbf{M}_u + \nu\mathbf{S}_u & 0 & 0 \\ 0 & \frac{3}{2\Delta t}\mathbf{M}_u + \nu\mathbf{S}_u & 0 \\ 0 & 0 & \frac{3}{2\Delta t}\mathbf{M}_u + \nu\mathbf{S}_u \end{bmatrix},$$

   so the three components of $\hat{w}_{j,h}^{n+1}$ or $\breve{w}_{j,h}^{n+1}$ can be solved independently and simutaneously.
2. gPAV-BDF2-Projection nonensemble: coefficient matrices for solving $\hat{w}_{j,h}^{n+1}$ and $\breve{w}_{j,h}^{n+1}$ are both

$$\mathbf{A}_2^j = \mathbf{A}_1 + \mathbf{B}_j,$$

   where $\mathbf{B}_j$ is the finite element matrix corresponding to the bilinear form

$$a(u,v) = \sum_i \int_I \eta_{i,j}(u \cdot \widehat{\tau}_i)(v \cdot \widehat{\tau}_i).$$

   Since $\mathbf{A}_2^j$ depends on j and is not block-diagonal, the three components of $\hat{w}_{j,h}^{n+1}$ or $\breve{w}_{j,h}^{n+1}$ cannot be solved independently nor simutaneously.
3. gPAV-BDF2 ensemble: coefficient matrices for solving the two subproblems w.r.t. $u_{j,h}^{n+1}$ are both

$$\mathbf{A}_3 = \begin{pmatrix} \mathbf{A}_1 + \overline{\mathbf{B}} & -\mathbf{C} \\ -\mathbf{C}^T & \mathbf{0} \end{pmatrix} \text{ or } \begin{pmatrix} \mathbf{A}_1 & -\mathbf{C} \\ -\mathbf{C}^T & \mathbf{0} \end{pmatrix} \text{ in a more radical way.}$$

   Here $\mathbf{C}$ is the finite element matrix corresponding to the blinear form $b(u,p) = (p, \nabla \cdot u)_f$ and $\overline{\mathbf{B}} = \frac{1}{J}\sum_{j=1}^{J} \mathbf{B}_j$.
4. gPAV-BDF2 nonensemble: coefficient matrices for solving the two subproblems w.r.t. $u_{j,h}^{n+1}$ are both

$$\mathbf{A}_4^j = \begin{pmatrix} \mathbf{A}_1 + \mathbf{B}_j & -\mathbf{C} \\ -\mathbf{C}^T & \mathbf{0} \end{pmatrix}.$$

In summary, the gPAV-BDF2-Projection ensemble scheme results in a common coefficient matrix for all J realizations, thus simultaneous computation for each single component of u can proceed using LU factorization or the block CG iterative solver. In contrast, the other schemes either need to solve u_1, u_2, u_3, and p together or need to handle J realizations one by one. In Sec. 4.2, we will use iterative solvers to observe computational efficiency. Specifically, for matrix $\mathbf{A}_1$ and $\mathbf{A}_2^j$, the (block) CG solver with multigrid preconditioner is enough. For the matrix $\mathbf{A}_3$ and $\mathbf{A}_4^j$, however, the (block) GMRES method needs to be used.

REMARK 3.3. *The SAV-BDF2AC ensemble method proposed in [34] has a coefficient matrix*

$$\mathbf{A}_{\text{uAC}} = \mathbf{A}_1 + \overline{\mathbf{B}} + \frac{1}{3\alpha\Delta t}\mathbf{D},$$

where $\mathbf{D}$ *is the finite element matrix associated with the blinear form* $d(u,v) = (\nabla \cdot u, \nabla \cdot v)_f$*. Note that* $\mathbf{A}_{\text{uAC}}$ *is not block diagonal, so the three components* u_1*,* u_2*,* u_3 *cannot be solved independently.*

4. Numerical examples. This section presents some numerical tests on validating the convergence rate, efficiency, and feasibility in application. We use $F(\chi) = \sqrt{\chi}$ in all of the following tests.

4.1. Convergence test. To validate the convergence rate of the proposed algorithms, we use an analytic solution defined on $D_f = (0,1) \times (1,2)$ and $D_p = (0,1) \times (0,1)$ with interface $I = [0,1] \times \{1\}$. It writes

$$u(x,y,t) = (u_1(x,y,t),\ u_2(x,y,t)),$$
$$u_1(x,y,t) = \left(x^2(y-1)^2 + \exp(y/\sqrt{k_{11}})\right)\cos(t),$$

Table 3.1: Convergence rates of the gPAV-BE-Projection ensemble algorithm for u, p, ϕ with $J = 3$, $\Delta t = h$.

Δt	$\|u_h - u\|_{H^1}^{E,1}$	Rate	$\|u_h - u\|_{H^1}^{E,2}$	Rate	$\|u_h - u\|_{H^1}^{E,3}$	Rate
1/8	1.253×10^{-1}	—	1.348×10^{-1}	—	1.473×10^{-1}	—
1/16	6.083×10^{-2}	1.04	6.587×10^{-2}	1.03	7.239×10^{-2}	1.03
1/32	2.962×10^{-2}	1.04	3.222×10^{-2}	1.03	3.558×10^{-2}	1.02
1/64	1.444×10^{-2}	1.04	1.577×10^{-2}	1.03	1.749×10^{-2}	1.02
1/128	7.065×10^{-3}	1.03	7.746×10^{-3}	1.03	8.618×10^{-3}	1.02
Δt	$\|p_h - p\|_{L^2}^{E,1}$	Rate	$\|p_h - p\|_{L^2}^{E,2}$	Rate	$\|p_h - p\|_{L^2}^{E,3}$	Rate
1/8	1.027×10^{-1}	—	1.089×10^{-1}	—	1.217×10^{-1}	—
1/16	4.769×10^{-2}	1.11	5.135×10^{-2}	1.08	5.849×10^{-2}	1.06
1/32	2.294×10^{-2}	1.06	2.520×10^{-2}	1.03	2.915×10^{-2}	1.00
1/64	1.126×10^{-2}	1.03	1.259×10^{-2}	1.00	1.473×10^{-2}	0.98
1/128	5.598×10^{-3}	1.01	6.338×10^{-3}	0.99	7.472×10^{-3}	0.98
Δt	$\|\phi_h - \phi\|_{H^1}^{E,1}$	Rate	$\|\phi_h - \phi\|_{H^1}^{E,2}$	Rate	$\|\phi_h - \phi\|_{H^1}^{E,3}$	Rate
1/8	8.023×10^{-2}	—	4.955×10^{-2}	—	4.920×10^{-2}	—
1/16	4.005×10^{-2}	1.00	2.474×10^{-2}	1.00	2.461×10^{-2}	1.00
1/32	1.988×10^{-2}	1.01	1.222×10^{-2}	1.02	1.223×10^{-2}	1.01
1/64	9.858×10^{-3}	1.01	6.023×10^{-3}	1.02	6.068×10^{-3}	1.01
1/128	4.890×10^{-3}	1.01	2.969×10^{-3}	1.02	3.014×10^{-3}	1.01

Table 3.2: Convergence rates of the gPAV-BDF2-Projection ensemble algorithm for u, p, ϕ with $J = 3$, $\Delta t = h$.

Δt	$\|u_h - u\|_{H^1}^{E,1}$	Rate	$\|u_h - u\|_{H^1}^{E,2}$	Rate	$\|u_h - u\|_{H^1}^{E,3}$	Rate
1/8	3.651×10^{-3}	—	3.906×10^{-3}	—	4.200×10^{-3}	—
1/16	1.028×10^{-3}	1.83	1.103×10^{-3}	1.83	1.194×10^{-3}	1.82
1/32	2.652×10^{-4}	1.96	2.835×10^{-4}	1.96	3.043×10^{-4}	1.97
1/64	6.442×10^{-5}	2.04	6.795×10^{-5}	2.06	7.113×10^{-5}	2.10
1/128	1.458×10^{-5}	2.14	1.483×10^{-5}	2.20	1.450×10^{-5}	2.29
Δt	$\|p_h - p\|_{L^2}^{E,1}$	Rate	$\|p_h - p\|_{L^2}^{E,2}$	Rate	$\|p_h - p\|_{L^2}^{E,3}$	Rate
1/8	7.350×10^{-3}	—	7.878×10^{-3}	—	8.579×10^{-3}	—
1/16	1.926×10^{-3}	1.93	2.081×10^{-3}	1.92	2.278×10^{-3}	1.91
1/32	4.933×10^{-4}	1.96	5.339×10^{-4}	1.96	5.840×10^{-4}	1.96
1/64	1.238×10^{-4}	1.99	1.334×10^{-4}	2.00	1.448×10^{-4}	2.01
1/128	3.025×10^{-5}	2.03	3.227×10^{-5}	2.05	3.440×10^{-5}	2.07
Δt	$\|\phi_h - \phi\|_{H^1}^{E,1}$	Rate	$\|\phi_h - \phi\|_{H^1}^{E,2}$	Rate	$\|\phi_h - \phi\|_{H^1}^{E,3}$	Rate
1/8	4.009×10^{-3}	—	4.039×10^{-3}	—	4.192×10^{-3}	—
1/16	7.958×10^{-4}	2.33	7.422×10^{-4}	2.44	7.528×10^{-4}	2.48
1/32	1.765×10^{-4}	2.17	1.483×10^{-4}	2.32	1.445×10^{-4}	2.38
1/64	4.102×10^{-5}	2.11	3.114×10^{-5}	2.25	2.890×10^{-5}	2.32
1/128	9.502×10^{-6}	2.11	6.394×10^{-6}	2.28	5.626×10^{-6}	2.36

$$u_2(x,y,t) = \big(\frac{2}{3}x(1-y)^3 + k_{22}(2-\pi\sin(\pi x)\big)\cos(t),$$
$$p(x,y,t) = (2-\pi\sin(\pi x))\sin(0.5\pi y)\cos(t),$$
$$\phi(x,y,t) = (2-\pi\sin(\pi x))(1-y-\cos(\pi y))\cos(t).$$

In this setup, the interface and initial conditions are compatible, and the forcing terms are fomulated according to the exact solutions. For model parameters, we set g, ν, S_0, and $\alpha_{\rm BJS}$ to be one, and the hydraulic conductivity tensor to be $\begin{bmatrix} k_{11} & 0 \\ 0 & k_{22} \end{bmatrix}$ with k_{11} and k_{22} being constants. The parameter ε in pressure projection is 0.8.

For ensemble computation, we assume $J = 3$ samples of k_{11} and k_{22} are seleted from a uniform distribution:

$$k_{11}^j = 1 - 0.1(j-1), \quad k_{22}^j = 1 + 0.1(j-1), \quad j = 1,2,3.$$

Then we get three samples for the initial conditions, boundary conditions, and forcing terms since they all depend on the values of k_{11} and k_{22}. The sample size three was chosen just for computational convenience, while a much larger size such as 1000 is also fine.

The expected computational errors $\|u_h - u\|_{H^1}$, $\|p_h - p\|_{L^2}$, and $\|\phi_h - \phi\|_{H^1}$ are $O(h^2 + \Delta t)$ for gPAV-BE-Projection and $O(h^2 + \Delta t^2)$ for gPAV-BDF2-Projection. In the simulation, we set the mesh size h and time step size Δt to be equal, and they are uniformly refined simutaneously, from the initial time step size $\Delta t = 1/8$ to final size $\Delta t = 1/128$. In this setting, the computational errors should be $O(\Delta t)$ and $O(\Delta t^2)$ for BE and BDF2 respectively. We report the computational errors at the final time $T = 5$ by the gPAV-BE-Projection scheme in Table 3.1 for the fluid velocity u, fluid pressure p, and hydraulic head ϕ, illustrating that the gPAV-BE-Projection algorithm is first order convergent in time. We also report in Table 3.2 the errors computed by the gPAV-BDF2-Projection scheme, from which we can easily validate the expected second order convergence rate.

4.2. Efficiency test. In this experiment, we take a random hydraulic conductivity tensor $\mathscr{K}(x,y,\omega)$ to form a stochastic problem and consider the computation of ensemble flows. Here $\omega \in \Omega$ and $(\Omega, \mathscr{F}, \mathscr{P})$ is a complete probability space. The hydraulic conductivity $\mathscr{K}(x,y,\omega)$ is assumed to be a diagonal stochastic tensor $\mathrm{diag}(k_{11}(x,y,\omega), k_{22}(x,y,\omega))$ with diagonal entries given by the Karhunen-Loève expansion

$$k_{11}(x,y,\omega) = k_{22}(x,y,\omega) = a_0 + \sigma\sqrt{\lambda_0}Y_0(\omega) + \sum_{i=1}^{n_f}\sigma\sqrt{\lambda_i}[Y_i(\omega)\cos(i\pi x) + Y_{n_f+i}(\omega)\sin(i\pi x))], \tag{4.1}$$

where $\lambda_0 = \frac{1}{2}\sqrt{\pi L_c}$, $\lambda_i = \sqrt{\pi}L_c\exp(-\frac{1}{4}(i\pi L_c)^2)$ for $i = 1, \cdots, n_f$, and $Y_0, \cdots, Y_{2n_f}$ are indepdendent and identically uniformly distributed in $[-\sqrt{3}, \sqrt{3}]$ having zero mean and unit variance. In the computation, we set $L_c = 0.25, a_0 = 1, \sigma = 0.15$, and $n_f = 2$, so there are 5 random variables $Y_0, Y_1, \cdots, Y_4$ in total.

The free and porus medias are $D_f = (0,1)\times(1,2)$ and $D_p = (0,1)\times(0,1)$ respectively and their interface is $I = [0,1]\times\{1\}$. Values for the physical parameters are mostly the same as in Sec. 4.1, except that the initial condition, Dirichlet boundary condition, and forcing terms are given according to

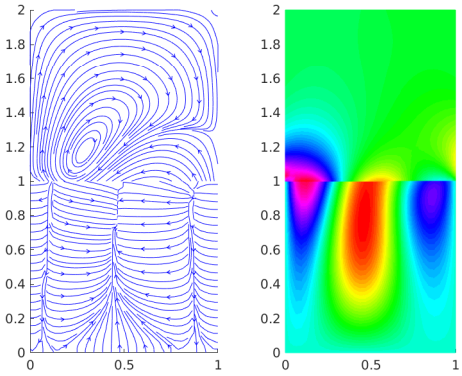
$$u(x,y,t,\omega) = (u_1(x,y,t,\omega),\ u_2(x,y,t,\omega)),$$
$$u_1(x,y,t,\omega) = Y_0(\omega)\big(y^2 - 2y + 1)\big)\cos(t),$$
$$u_2(x,y,t,\omega) = Y_1(\omega)\big(x^2 - x\big)\cos(t),$$
$$\phi(x,y,t,\omega) = Y_2(\omega)y\cos(t),$$
$$f_f = (Y_3(\omega)xy, Y_3(\omega)xy),$$
$$f_p = Y_4(\omega)xy.$$

For the stochastic Stokes-Darcy problem, we solve it by a sparse-grid collocation method utilizing the Smolyak formula. Taking $h = 1/50$, $\Delta t = 1/100$, and $J = 241$ collocation points, we simulate the ensemble flow until $T = 2$ using the gPAV-BDF2-Projection ensemble and nonensemble schemes, gPAV-BDF2 ensemble and nonensemble schemes (no Projection).

Table 4.1: Efficiency performance of the proposed gPAV-BDF2-Projection ensemble scheme versus other schemes, tested using sparse-grid with $J = 241$ collocation points, $h = 1/50, \Delta t = 1/100, T = 2$. Below t_{cpu} denotes the CPU time in seconds.

	gPAV-BDF2-Projection ensemble	gPAV-BDF2-Projection nonensemble	gPAV-BDF2 ensemble	gPAV-BDF2 nonensemble
Average t_{cpu} per time step	10.35 s	0.11×241 s	26.42 s	1.22×241 s
Total CPU time	2070 s	5457 s	5284 s	58883 s

(a) gPAV-BDF2-Projection ensemble



(b) gPAV-BDF2-Projection nonensemble (c) gPAV-BDF2 ensemble (d) gPAV-BDF2 nonensemble

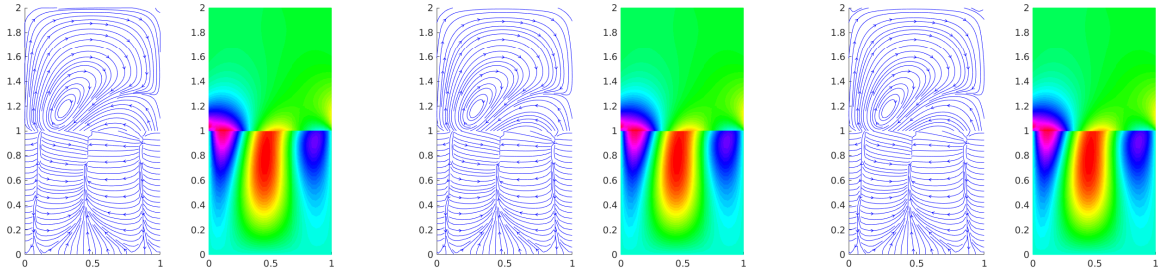


Fig. 4.1: Simulations at $T = 2$ by four different schemes using the sparse-grid method with $J = 241$ collocation points, $h = 1/50, \Delta t = 1/100$. The streamlines of expectations of fluid flow velocity u and porous media flow velocity $v = -\mathscr{K}\nabla\phi$ are plotted in the left of each subfigure; expectations of p and ϕ are plotted in the right of each subfigure.

The CPU times using the mentioned four different schemes are listed in Table 4.1, which shows that the gPAV-BDF2-Projection ensemble scheme outperforms all the other schemes. The win on efficiency thanks to simultaneous computation of 241 solutions in which redundant information due to linear dependence of multiple samples are removed; one should also own to the usage of projection method in combination with gPAV so that the system for solving u_1, u_2, p is fully decoupled into three systems.

The streamlines of the expectations of fluid flow velocity u and porous media flow velocity $v = -\mathscr{K}\nabla\phi$ are plotted in the left of each subfigure of Figure 4.1. The expectations of fluid flow pressure p and hydraulic head ϕ are also plotted in the right of each subfigure. From the figure we can observe that the four numerical schemes in efficiency comparison provide almost identical simulations.

4.3. Simulating the extraction of shale oil. In this experiment, we simulate the extraction of shale oil by a vertical production wellbore. The computational domain is illustrated in Fig. 4.2, which consists of an oil reservoir represented by D_p and a vertical production wellbore denoted by the pipe D_f. The setup here is mainly for validating the feasibility of gPAV-BDF2-Projection. More realistic features can be considered, for instance the pipe can be replaced by a multistage hydraulic fractured vertical wellbore with cased-hole completion [7].

The physical parameters g, ν, $\mathscr{K}$, and α_{BJS} are set to be one. For boundary conditions, we set $\phi = 1$ on $\partial D_p \backslash I$, $u = (0,0)^T$ on the vertical sides of the pipe above the reservoir and $u = (0,1)^T$ on the top of the pipe. We will vary the specific mass storativity coefficient S_0 in simulation. The simulated fluid flow velocity u and porous media flow velocity $v = -\mathscr{K}\nabla\phi$ are then plotted in Figure 4.5 for $t = 0.2, 0.5, 1.0$ from top to bottom. In particular, the left of Figure 4.5 corresponds to $S_0 = 10^{-5}$, while the right corresponds to $S_0 = 1$. From the figure we can see that when S_0 is relatively large, which implies greater oil storage capacity, it takes longer time for the oil extraction process to reach a steady state.

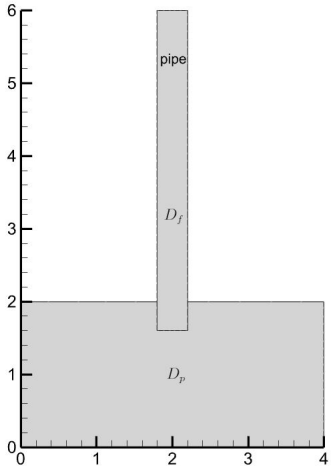


Fig. 4.2: Domains for the simulation of oil extraction.

4.4. Simulating the subsurface flow in a karst aquifer. Inspired by [39], we condier a realistic simulation of the subsurface flow in a karst aquifer. The computational domain is shown in Figure 4.4, where the free flow domain D_f with a curvy boundary $\overline{ABCDEFGH}$ represents a T-shape conduit, and the porous media flow domain D_p simulates the karst aquifer. The two domains together form a unie square and they interact at the the curvy interfaces $\overline{BCD}, \overline{EFG}, \overline{HA}$. Specifically, the computational domain is given by $A = (0, 0.8)$, $B = (0, 0.55)$, $C = (0.55, 0.4)$, $D = (0.7, 0)$, $E = (0.85, 0)$, $F = (0.75, 0.45)$, $G = (1, 0.5)$, and $H = (1, 0.7)$. The physical parameters g, ν, and S_0 are set to be one, and $\alpha_{\mathrm{BJS}} = 0.1$. The source terms in the Stokes-Darcy equations are set to be zero and $\phi = 0$ on $\partial D_p \backslash I$. The hydraulic conductivity $\mathscr{K}(x, y)$ is assumed to be $m\mathbf{I}$, where m is the conductivity magnitude determing how easily the water can flow through the interface. The inflow/outflow boundary condition for u is

$$u = \begin{cases} (s_1, 0) & on \quad \overline{AB} \\ (0, s_2) & on \quad \overline{DE} \\ (s_3, 0) & on \quad \overline{GH} \end{cases},$$

where s_1, s_2 and s_3 are constants.

Simulations are performed for different scenarios using by the gPAV-BDF2-Projection scheme with $h = 0.011$ and $\Delta t = 0.002$. For simulation, we assume the inflow comes from the left boundary $\overline{AB}$, and flows out through the bottom and right outlets $\overline{DE}, \overline{GH}$. The boundary conditions should satisfy $s_1 > 0, s_2 < 0, s_3 > 0$. Two cases are taken into consideration: a balanced case with $s_1 = 2, s_2 = -1, s_3 = 1$ and an imbalanced case with $s_1 = 2, s_2 = -0.1, s_3 = 0.2$. Here "imbalanced" means the infow speed is much higher than the outflow speed. To study the effect of the hydraulic conductivity on the filtration, we also vary the magnitude m among $1, 10^{-2}$, and 10^{-4}.

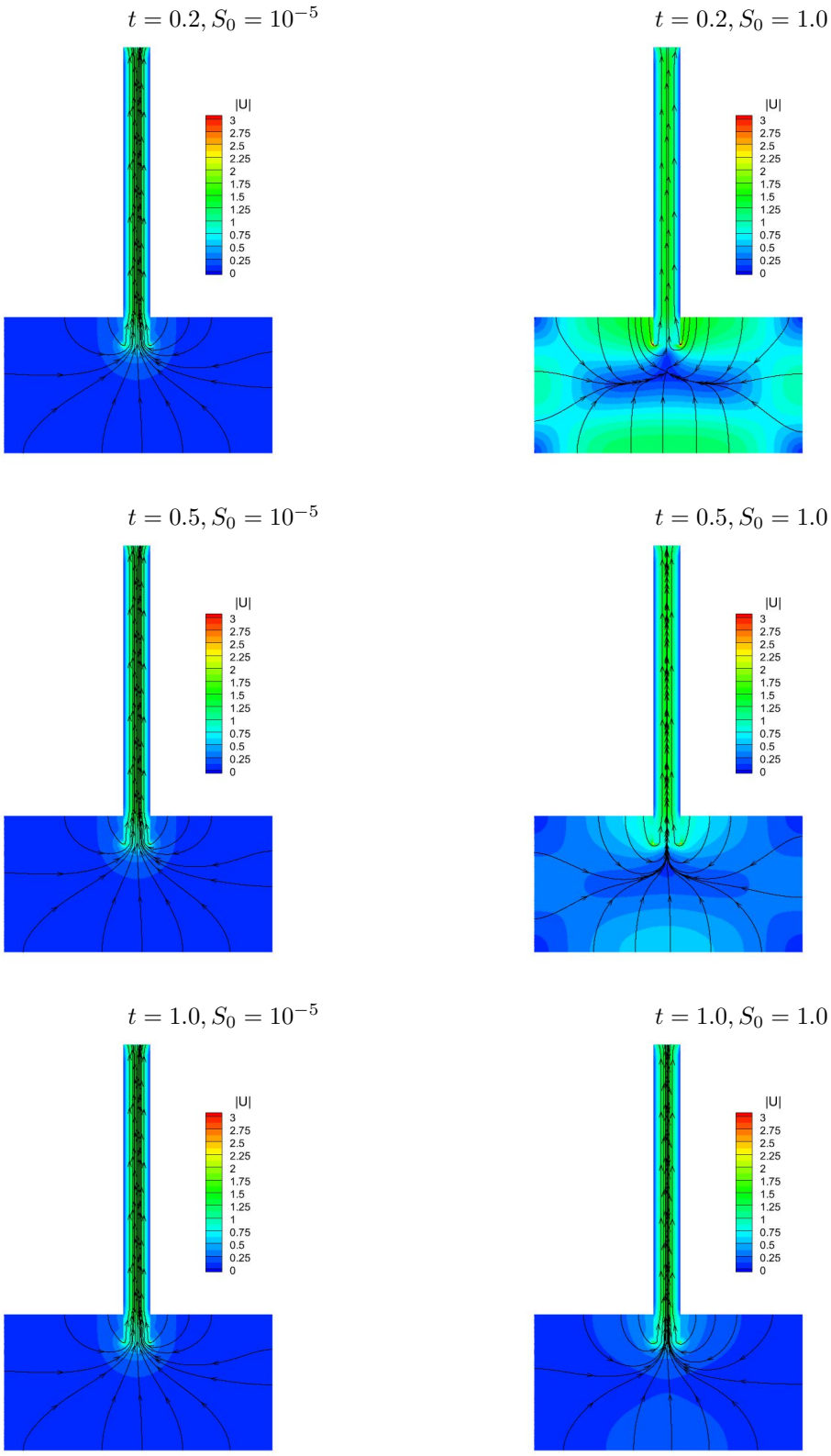


Fig. 4.3: Fluid flow velocity u and porous media flow velocity $v = -\mathscr{K}\nabla\phi$ simulated with different S_0. Left: $S_0 = 10^{-5}$; right: $S_0 = 1$. From top to bottom: $t = 0.2, 0.5, 1.0$.

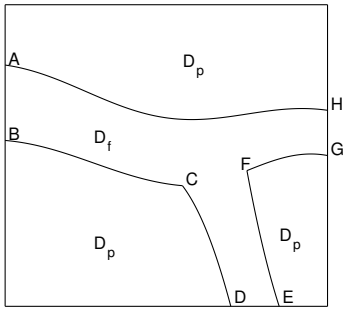


Fig. 4.4: Domains for the simulation of subsurface flow in a karst aquifer.

The simulated fluid flow velocity u and porous media flow velocity $v = -\mathscr{K}\nabla\phi$ are then plotted in Figure 4.5. In particular, the left of the figure corresponds to $s_1 = 2, s_2 = -1, s_3 = 1$, while the right corresponds to $s_1 = 2, s_2 = -0.1, s_3 = 0.2$. From top to bottom of the figure, the conductivity magnitudes are $m = 1,\ 10^{-2},\ 10^{-4}$. From Figure 4.5 we can see that when m decreases, the flow speed in porous media is significantly reduced, and the flow filtrated from the conduit cannot stream too much further if the hydraulic conductivity is too small (see the bottome of Figure 4.5).

We then consider another situation: the inflow comes from both the left boundary $\overline{AB}$ and the right inlet $\overline{GH}$, and steams out through the bottom outlet $\overline{DE}$. Thus the boundary conditions should satisfy $s_1 > 0, s_2 < 0, s_3 < 0$. A representative balanced case will be $s_1 = 1, s_2 = -1, s_3 = -1$ and an imbalanced case be $s_1 = 0.5, s_2 = -1, s_3 = -2$. The simulations are reported in Figure 4.6. Again, when the hydraulic conductivty is small, the stream meanders very slowly from the conduit to the porous media.

5. Conclusions. In this report we proposed two highly efficient ensemble algorithms based on the gPAV approach and the rotational pressure correction method. We proved the proposed algorithms are unconditionally stable with respect to the modified energy. With the adoption of gPAV approach we were also able to make use of the rotational pressure correction method to fully decouple all the variables in the equations including all the components of the velocity field, leading to much smaller linear systems to be solved at each time step. Additionally, all the ensemble members share the same constant coefficient matrix, for which the efficient block CG method can be used to solve all ensemble members together at significantly reduced computational cost using much lest CPU time.

REFERENCES

[1] J. Carter and N. Jiang, *Numerical analysis of a second order ensemble method for evolutionary Magnetohydrodynamics equations at small magnetic Reynolds number*, Numerical Methods for Partial Differential Equations, 38 (2022), 1407-1436.

[2] J. Carter, D. Han and N. Jiang, *Second order, unconditionally stable, linear ensemble algorithms for the Magnetohydrodynamics equations*, Journal of Scientific Computing, 94 (2023), 41.

[3] I. Babuška, F. Nobile and R. Tempone, *A stochastic collocation method for elliptic partial differential equations with random input data*, SIAM Journal on Numerical Analysis, 45 (2007), 1005-1034.

[4] A. Barth and A. Lang, *Multilevel Monte Carlo method with applications to stochastic partial differential equations*, International Journal of Computer Mathematics, 89 (2012), 2479-2498.

[5] J. Bear, *Hydraulics of Groundwater*, McGraw-Hill, New York, 1979.

[6] G. Beavers and D. Joseph, *Boundary Conditions at a Naturally Impermeable Wall*, J. Fluid Mech., 30 (1967), 197-207.

[7] Y. Bi, L. Shan, and H. Zhang, *New decoupled method for the evolutionary dual-porosity-Stokes model with Beavers-Joseph interface conditions*, Applied Numerical Mathematics, 175 (2022), 73-97.

[8] D. Brown, R. Cortez, and M. Minion, *Accurate projection methods for the incompressible Naiver-Stokes equations*, J. Comput. Phys., 168 (2001), 464-499.

[9] H. Calandra, S. Gratton, J. Langou, X. Pinel, X. Vasseur, *Flexible Variants of Block Restarted GMRES Methods with Application to Geophysics*, SIAM Journal on Scientific Computing, vol. 34, no. 2, (2012), 714-736.

[10] Y. Cao, M. Gunzburger, X. He and X. Wang, *Parallel, non-iterative, multi-physics domain decomposition methods for time-dependent Stokes-Darcy systems*, Mathematics of Computation, 83 (2014), 1617-1644.

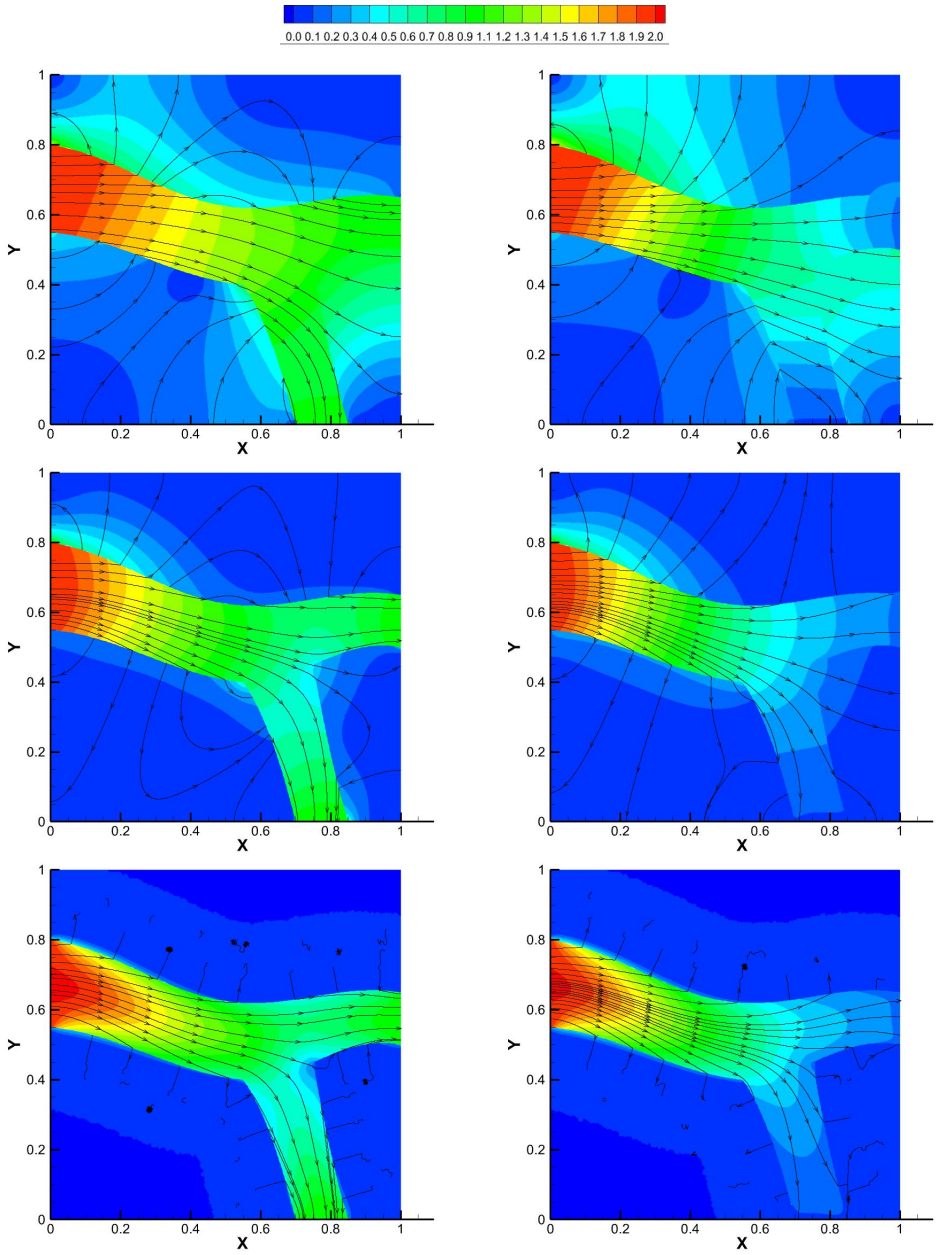


Fig. 4.5: Fluid flow velocity u and porous media flow velocity $v = -\mathscr{K}\nabla\phi$ simulated with different inflow/outflow boundary conditions and conductivity magnitude m. Left: $s_1 = 2, s_2 = -1.0, s_3 = 1.0$; right: simulations with $s_1 = 2, s_2 = -0.1, s_3 = 0.2$. From top to bottom: $m = 1,\ 10^{-2},\ 10^{-4}$.

[11] J. Connors, *An ensemble-based conventional turbulence model for fluid-fluid interactions*, Int. J. Numer. Anal. Model., 15 (2018), 492-519.

[12] J.L. Guermond, P. Minev and J. Shen, *An overview of projection methods for incompressible flows*, Computer Methods in Applied Mechanics and Engineering, 195 (44-47) (2006), pp.6011–6045.

[13] J. Fiordilino, *A second order ensemble timestepping algorithm for natural convection*, SIAM Journal on Numerical Analysis, 56 (2018), 816-837.

[14] J. Fiordilino and S. Khankan, *Ensemble timestepping algorithms for natural convection*, International Journal of Numerical Analysis and Modeling, 15 (2018), 524-551.

[15] J. Guermond and J. Shen, *On the error estimates for the rotational pressure-correction projection methods*, Mathematics of Computation, 73 (2004), 1719–1737.

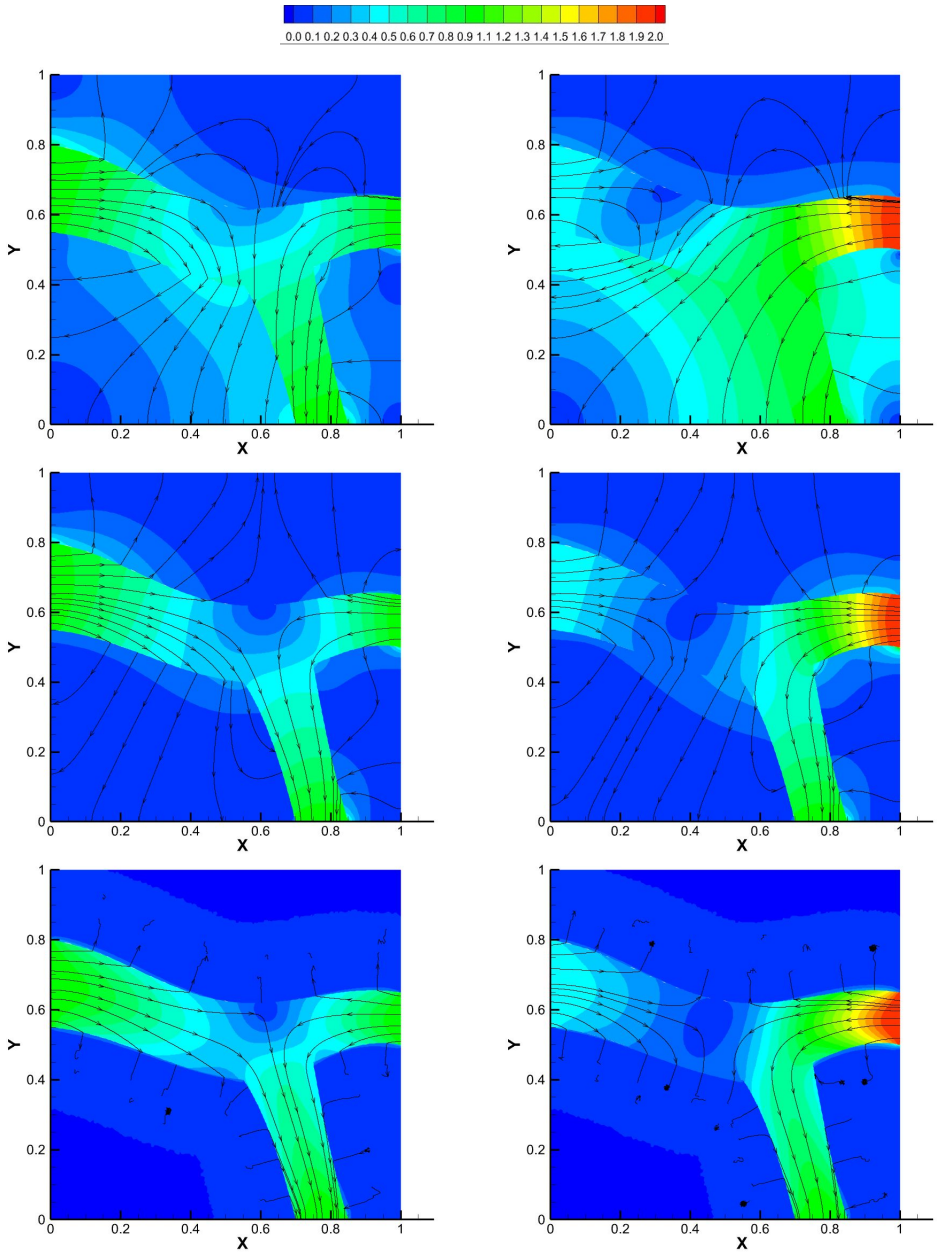


Fig. 4.6: Fluid flow velocity u and porous media flow velocity $v = -\mathscr{K}\nabla\phi$ simulated with different inflow/outflow boundary conditions and conductivity magnitude m. Left: $s_1 = 1.0, s_2 = -1.0, s_3 = -1.0$; right: simulations with $s_1 = 0.5, s_2 = -1.0, s_3 = -2.0$. From top to bottom: $m = 1,\ 10^{-2},\ 10^{-4}$.

[16] M. Gunzburger, T. Iliescu and M. Schneier, *A Leray regularized ensemble-proper orthogonal decomposition method for parameterized convection-dominated flows*, IMA Journal of Numerical Analysis, 40 (2020), 886-913.

[17] M. Gunzburger, N. Jiang and M. Schneier, *An ensemble-proper orthogonal decomposition method for the nonstationary Navier-Stokes equations*, SIAM Journal on Numerical Analysis, 55 (2017), 286-304.

[18] M. Gunzburger, N. Jiang and Z. Wang, *An efficient algorithm for simulating ensembles of parameterized flow problems*, IMA Journal of Numerical Analysis, 39 (2019), 1180-1205.

[19] J.C. Helton and F.J. Davis, *Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems*, Reliability Engineering and System Safety, 81 (2003), 23-69.

[20] X. He, N. Jiang and C. Qiu, *An artificial compressibility ensemble algorithm for a stochastic Stokes-Darcy model with random hydraulic conductivity and interface conditions*, International Journal for Numerical Methods in Engineering,

121 (2020), 712-739.
[21] S. Hosder, R. Walters and R. Perez, *A non-intrusive polynomial chaos method for uncertainty propagation in CFD simulations, AIAA-Paper 2006-891*, 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, January 2006, CD-ROM.
[22] W. Jäger and A. Mikelic, *On the Interface Boundary Condition of Beavers, Joseph, and Saffman*, Siam Journal on Applied Mathematics, 60(4) (2000), 1111-1127.
[23] H. Ji and Y. Li, *A breakdown-free block conjugate gradient method*, BIT Numerical Mathematics, 57(2) (2017), 379-403.
[24] N. Jiang, S. Kaya, and W. Layton, *Analysis of model variance for ensemble based turbulence modeling*, Computational Methods in Applied Mathematics, 15 (2015), 173-188.
[25] N. Jiang and W. Layton, *An algorithm for fast calculation of flow ensembles*, International Journal for Uncertainty Quantification, 4 (2014), 273-301.
[26] N. Jiang and W. Layton, *Numerical analysis of two ensemble eddy viscosity numerical regularizations of fluid motion*, Numerical Methods for Partial Differential Equations, 31 (2015), 630-651.
[27] N. Jiang, Y. Li and H. Yang, *An artificial compressibility Crank-Nicolson leap-frog method for the Stokes-Darcy model and application in ensemble simulations*, SIAM Journal on Numerical Analysis, 59 (2021), 401-428.
[28] N. Jiang, M. Kubacki, W. Layton, M. Moraiti and H. Tran, *A Crank-Nicolson leapfrog stabilization: unconditional stability and two applications*, Journal of Computational and Applied Mathematics, 281 (2015), 263-276.
[29] N. Jiang and C. Qiu, *An efficient ensemble algorithm for numerical approximation of stochastic Stokes-Darcy equations*, Computer Methods in Applied Mechanics and Engineering, 343 (2019), 249-275.
[30] N. Jiang and M. Schneier, *An efficient, partitioned ensemble algorithm for simulating ensembles of evolutionary MHD flows at low magnetic Reynolds number*, Numerical Methods for Partial Differential Equations, 34 (2018), 2129-2152.
[31] N. Jiang, A. Takhirov and J. Waters, *Robust SAV-ensemble algorithms for parametrized flow problems with energy stable open boundary conditions*, Computer Methods in Applied Mechanics and Engineering, 392 (2022), 114709.
[32] N. Jiang and H. Yang, *Stabilized scalar auxiliary variable ensemble algorithms for parameterized flow problems*, SIAM Journal on Scientific Computing, 43 (2021), A2869-A2896.
[33] N. Jiang and H. Yang, *SAV decoupled ensemble algorithms for fast computation of Stokes-Darcy flow ensembles*, Computer Methods in Applied Mechanics and Engineering, 387 (2021), 114150.
[34] N. Jiang and H. Yang, *Fast and accurate artificial compressibility ensemble algorithms for computing parameterized Stokes-Darcy flow ensembles*, Journal of Scientific Computing, 94 (2023), 17.
[35] F. Kuo, C. Schwab and I. Sloan, *Quasi-Monte Carlo finite element methods for a class of elliptic partial differential equations with random coefficients*, SIAM J. Numer. Anal., 50 (2012), 3351-3374.
[36] W. Li, J. Fang, Y. Qin and P. Huang, *Rotational pressure-correction method for the Stokes/Darcy model based on the modular grad-div stabilization*, Applied Numerical Mathematics, 160 (2021), 451-465.
[37] J. Li, M. Yao, M.A. Al Mahbub and H. Zheng, *The efficient rotational pressure-correction schemes for the coupling Stokes/Darcy problem*, Computers & Mathematics with Applications, 79 (2020), 337-353.
[38] L. Lin, X. Liu and S. Dong, *A gPAV-based unconditionally energy-stable scheme for incompressible flows with outflow/open boundaries*, Computer Methods in Applied Mechanics and Engineering, 365 (2020), 112969.
[39] R. Li, J. Li, X. He, and Z. Chen, *A stabilized finite volume element method for a coupled Stokes–Darcy problem*, Applied Numerical Mathematics, 133 (2018), 2-24.
[40] Y. Luo and Z. Wang, *An ensemble algorithm for numerical solutions to deterministic and random parabolic PDEs*, SIAM Journal on Numerical Analysis, 56 (2018), 859-876.
[41] Y. Luo and Z. Wang, *A multilevel Monte Carlo ensemble scheme for random parabolic PDEs*, SIAM Journal on Scientific Computing, 41 (2019), A622-A642.
[42] M. Mohebujjaman and L. Rebholz, *An efficient algorithm for computation of MHD flow ensembles*, Computational Methods in Applied Mathematics, 17 (2017), 121-137.
[43] Y. Qian, Z. Yang, F. Wang and S. Dong, *gPAV-based unconditionally energy-stable schemes for the Cahn-Hilliard equation: stability and error analysis*, Computer Methods in Applied Mechanics and Engineering, 372 (2020), 113444.
[44] M. Reagan, H.N. Najm, R.G. Ghanem and O.M. Knio, *Uncertainty quantification in reacting-flow simulations through non-intrusive spectral projection*, Combustion and Flame, 132 (2003), 545-555.
[45] V. Romero, J. Burkardt, M. Gunzburger and J. Peterson, *Comparison of pure and "Latinized" centroidal Voronoi tessellation against various other statistical sampling methods*, Reliability Engineering and System Safety, 91 (2006), 1266-1280.
[46] P. Saffman, *On the boundary condition at the interface of a porous medium*, Stud. Appl. Math., 1 (1971), 93-101.
[47] L. Shan, H. Zheng and W. Layton, *A decoupling method with different subdomain time steps for the nonstationary Stokes–Darcy model*, Numer. Methods for Partial Differential Eq., 29 (2013), 549-583.
[48] A. Takhirov, M. Neda, and J. Waters, *Time relaxation algorithm for flow ensembles*, Numerical Methods for Partial Differential Equations, 32 (2016), 757-777.
[49] A. Takhirov and J. Waters, *Ensemble algorithm for parametrized flow problems with energy stable open boundary conditions*, Computational Methods in Applied Mathematics, 20 (2020), 531-554.
[50] L. Timmermans, P. Minev and F. Van De Vosse, *An approximate projection scheme for incompressible flow using spectral elements*, International Journal for Numerical Methods in Fluids, 22 (1996), 673-688.
[51] D. Xiu and J.S. Hesthaven, *High-order collocation methods for differential equations with random inputs*, SIAM Journal on Scientific Computing, 27 (2005), 1118-1139.
[52] Z. Yang and S. Dong, *A roadmap for discretely energy-stable schemes for dissipative systems based on a generalized auxiliary variable with guaranteed positivity*, Journal of Computational Physics, 404 (2020), 109121.