An enriched Galerkin method for the Navier-Stokes equations coupled with heat dynamics

Sanjeeb Poudel[1], **Sanghyun Lee**[2], **Lin Mu**[3]

[1] Department of Scientific Computing, Florida State University, Tallahassee, 32306, FL
[2] Department of Mathematics, Florida State University, Tallahassee, 32306, FL
[3] Department of Mathematics, University of Georgia, Athens, 30602, GA

Abstract

The coupling of fluid dynamics and heat flow presents a range of interesting scientific and engineering applications. This coupled system encompasses many challenges, like the inf-sup stability of finite element spaces. This study implements the Enriched Galerkin (EG) finite element method, which provides an inf-sup stable scheme for solving the Navier-Stokes equations while having fewer degrees of freedom. Additionally, various iterative methods to solve the coupled nonlinear system efficiently are discussed.

Introduction

The interplay between fluid and heat dynamics is evident in our everyday experiences, such as the natural convection of fluid in a kettle. Studying this coupled system also offers valuable insights into how fluids are heated in geothermal systems.

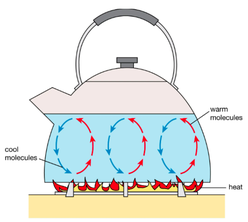


Figure 1: *Kettle*

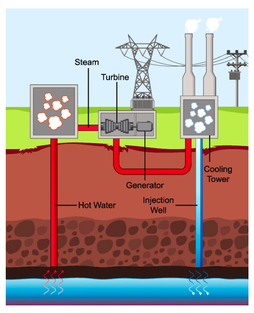


Figure 2: *Geothermal System*

1.1 Governing Equations

The Boussinesq model is one of the most common approaches for simulating the coupled system. For a bounded domain $\Omega \subset \mathbb{R}^d$ for $d = 2, 3$ with simply connected Lipsitz boundary $\partial\Omega$, the goal is to find the fluid velocity $\mathbf{u} : \Omega \to \mathbb{R}^d$, the pressure $p : \Omega \to \mathbb{R}$, and the temperature $T : \Omega \to \mathbb{R}$, such that the following three balance equations are satisfied for time $t = (0, t_f]$.

1.1.1 Momentum Balance

From the conservation of linear momentum, we have

$$\partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \nabla \cdot (2\nu\boldsymbol{\varepsilon}(\mathbf{u})) + \frac{1}{\rho}\nabla p = -\beta T \hat{\mathbf{g}}, \quad \text{in } \Omega \times (0, t_f],$$

where ρ is the constant density, ν is the kinematic viscosity, β is the coefficient of expansion of the fluid, $\hat{\mathbf{g}}$ is the acceleration due to gravity, and $\boldsymbol{\varepsilon}(\mathbf{u}) = \frac{1}{2}(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)$ is the symmetric strain-rate tensor.

1.1.2 Mass Balance

For an incompressible flow, the conservation of mass gives

$$\nabla \cdot \mathbf{u} = 0, \quad \text{in } \Omega \times (0, t_f].$$

1.1.3 Energy Balance

The temperature distribution of the fluid is given by the following advection-diffusion equation:

$$\partial_t T + \mathbf{u} \cdot \nabla T - \nabla \cdot (\kappa \nabla T) = q, \quad \text{in } \Omega \times (0, t_f],$$

where κ is the thermal diffusivity, and q is the source term.

1.1.4 Boundary and Initial Conditions

The system is supplemented by the following boundary and initial conditions:

$$\begin{aligned}
(2\nu\epsilon(\mathbf{u}) - \frac{1}{\rho}p\mathbf{I})\mathbf{n} &= \mathbf{s}, && \text{on } \Gamma_N^u \times (0, t_f],\\
T &= T_D, && \text{on } \Gamma_D^T \times (0, t_f],\\
\frac{\partial T}{\partial t} &= T_N, && \text{on } \Gamma_N^T \times (0, t_f],\\
\mathbf{u}(\cdot, 0) &= \mathbf{u}^0, && \text{in } \Omega,\\
T(\cdot, 0) &= T^0, && \text{in } \Omega.
\end{aligned}$$

Numerical Discretization

1.2 Temporal Discretization

For a given $\mathbf{u}^n$ and T^n, we solve for $\mathbf{u}^{n+1}$, p^{n+1}, and T^{n+1} using the Backward Euler method,

$$\begin{aligned}
\frac{1}{\delta t}\mathbf{u}^{n+1} + (\mathbf{u}^{n+1} \cdot \nabla)\mathbf{u}^{n+1} - \nabla \cdot \left(2\nu\boldsymbol{\varepsilon}(\mathbf{u}^{n+1})\right)\\
+\frac{1}{\rho}\nabla p^{n+1} + \beta T^{n+1}\hat{\mathbf{g}} = \frac{1}{\delta t}\mathbf{u}^n,\\
\nabla \cdot \mathbf{u}^{n+1} = 0,\\
\frac{1}{\delta t}T^{n+1} + \mathbf{u}^{n+1} \cdot \nabla T^{n+1} - \nabla \cdot \left(\kappa \nabla T^{n+1}\right) = \frac{1}{\delta t}T^n + q.
\end{aligned}$$

1.3 Nonlinear Iterative Solvers

1.3.1 Picard Iteration

The Picard method is a fixed-point iteration to find solutions to the equations in the form $\mathbf{x} = g(\mathbf{x})$. Starting with an initial guess $\mathbf{x}_0$, the method generates a sequence of approximations $\{\mathbf{x}_k\}$ by repeatedly applying the function $g(\mathbf{x})$ to the previous iterate,

$$\mathbf{x}_{k+1} = g(\mathbf{x}_k), \tag{3}$$

where k denotes the iteration count. If the method converges, the sequence of iterates approaches the fixed point $\mathbf{x}^*$, where $\mathbf{x}^* = g(\mathbf{x}^*)$. In our case, $\mathbf{x} = [\mathbf{u}, p, T]^\top$, and $g(.)$ provides the solution of the following linear system of equations:

$$\begin{aligned}
\frac{1}{\delta t}\mathbf{u}^{n+1}_{k+1} + (\mathbf{u}^{n+1}_k \cdot \nabla)\mathbf{u}^{n+1}_{k+1} - \nabla \cdot \left(2\nu\boldsymbol{\varepsilon}(\mathbf{u}^{n+1}_{k+1})\right)\\
+\frac{1}{\rho}\nabla p^{n+1}_{k+1} + \beta T^{n+1}_{k+1}\hat{\mathbf{g}} = \frac{1}{\delta t}\mathbf{u}^n, \qquad (4a)\\
\nabla \cdot \mathbf{u}^{n+1}_{k+1} = 0, \qquad (4b)\\
\frac{1}{\delta t}T^{n+1}_{k+1} + \mathbf{u}^{n+1}_k \cdot \nabla T^{n+1}_{k+1} - \nabla \cdot \left(\kappa \nabla T^{n+1}_{k+1}\right) = \frac{1}{\delta t}T^n + q. \qquad (4c)
\end{aligned}$$

1.3.2 Anderson-accelerated Picard Iteration

With Anderson acceleration, the update is given by

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \beta_k \mathbf{w}_{k+1} - (\mathbf{E}_k + \beta_k \mathbf{F}_\mathbf{k})\gamma_\mathbf{k}.$$

- $\mathbf{w}_k := g(\mathbf{x}_{k-1}) - \mathbf{x}_{k-1}$ is the residual
- β_k is the relaxation parameter
- with the depth parameter m_k, the matrix $\mathbf{F}_\mathbf{k} \in \mathbb{R}^{\mathbf{N}\times\mathbf{m_k}}$ is defined as

$$\mathbf{F}_k = [(\mathbf{w}_{k+1} - \mathbf{w}_k)...(\mathbf{w}_{k+1-(m_k-1)} - \mathbf{w}_{k-(m_k-1)})]$$

- The matrix $\mathbf{E}_k \in \mathbb{R}^{N\times m_k}$ is defined as

$$\mathbf{E}_k = [(\mathbf{e}_k)...(\mathbf{e}_{k-(m_k-1)})],$$

where $\mathbf{e}_k := \mathbf{x}_k - \mathbf{x}_{k-1}$ is the difference between iterates.

- Using the method of least square,

$$\gamma_k = \text{argmin}||\mathbf{F}_k\gamma - \mathbf{w}_{k+1}||.$$

1.3.3 Newton Iteration

Newton's method refines the approximation of the root by updating the current guess $\mathbf{x}^k$, according to the relation

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{J}^{-1}(\mathbf{x}^k)g(\mathbf{x}^k),$$

where $\mathbf{J}^{-1}(\mathbf{x}^k)$ denotes the inverse of the Jacobian matrix at the current approximate $\mathbf{x}^k$. After expanding the Jacobian matrix, the iterative equation is given by:

$$\begin{aligned}
\frac{1}{\delta t}\mathbf{u}^{n+1}_{k+1} + (\mathbf{u}^{n+1}_k \cdot \nabla)\mathbf{u}^{n+1}_{k+1} + (\mathbf{u}^{n+1}_{k+1} \cdot \nabla)\mathbf{u}^{n+1}_k - \nabla \cdot \left(2\nu\boldsymbol{\varepsilon}(\mathbf{u}^{n+1}_{k+1})\right)\\
+\frac{1}{\rho}\nabla p^{n+1}_{k+1} + \beta T^{n+1}_{k+1}\hat{\mathbf{g}} = \frac{1}{\delta t}\mathbf{u}^n + (\mathbf{u}^{n+1}_k \cdot \nabla)\mathbf{u}^{n+1}_k,\\
\nabla \cdot \mathbf{u}^{n+1}_{k+1} = 0,\\
\frac{1}{\delta t}T^{n+1}_{k+1} + \mathbf{u}^{n+1}_k \cdot \nabla T^{n+1}_{k+1} + \mathbf{u}^{n+1}_{k+1} \cdot \nabla T^{n+1}_k - \nabla \cdot \left(\kappa \nabla T^{n+1}_{k+1}\right)\\
= \frac{1}{\delta t}T_n + q + \mathbf{u}^{n+1}_k \cdot \nabla T^{n+1}_k.
\end{aligned}$$

1.4 Enriched Galerkin (EG) Finite Element Method

The EG method consists of continuous Lagrange finite elements enriched with some discontinuous functions. We consider a shape regular partition of the computational domain, $\overline{\Omega} = \cup_{K\in\mathcal{K}_h}\overline{K}$, where $K \in \mathcal{K}_h$ are quadrilaterals, and define the finite element space for velocity, pressure, and temperature as follows:

1.4.1 Finite-Element Space for Velocity

The EG finite-element space for the velocity is obtained by extending the continuous space with a discontinuous function.

- $\mathcal{CG}_1 := \{\psi \in [H^1(\Omega)]^d \mid \psi|_K \in [\mathbb{P}_1(K)]^d,\ \forall K \in \mathcal{K}_h\}$,
- $\mathcal{DG}_1 = \{\psi \in [L^2(\Omega)]^d \mid \psi|_K = c_K(\mathbf{x} - \mathbf{x}_K),\ c_K \in \mathcal{R},\ \forall K \in \mathcal{K}_h\}$, where $\mathbf{x}_K$ is the centroid of $K \in \mathcal{K}$
- $\mathcal{V}_h := \mathcal{CG}_1 \oplus \mathcal{DG}_1 \in [L^2(\Omega)]^d$
- The EG method provides fewer degrees of freedom compared to other inf-sup stable methods for the NS equation.
- For a quadrilateral element in 2D, it has 8 CG + 1 DG = 9 DoFs, whereas using the Taylor-Hood element has 16 DoFs.

1.4.2 Finite-Element Space for Pressure

$$\mathcal{W}_h := \begin{cases} \{\psi \in L^2(\Omega) \mid \psi|_K \in \mathbb{P}_0(K), \forall K \in \mathcal{K}_h\} \cap L^2_0(\Omega) & \text{if } |\Gamma_N^u| = 0\\ \{\psi \in L^2(\Omega) \mid \psi|_K \in \mathbb{P}_0(K), \forall K \in \mathcal{K}_h\} & \text{if } |\Gamma_N^u| > 0 \end{cases}$$

1.4.3 Finite-Element Space for Temperature

$$\mathcal{T}_h := \{\psi \in H^1(\Omega),\ \mid \psi|_K \in [\mathbb{P}_1(K)],\ \forall K \in \mathcal{K}_h\}$$

Numerical Experiments

1.4.4 Convergence of smooth solutions

For the domain $\Omega = [0, 1]^2$, the exact solution is given by $\mathbf{u} = \left[\sin(\pi tx)\sin(\pi ty), \cos(\pi tx)\cos(\pi ty)\right]^\top$, $p = \sin(\pi tx)\cos(\pi ty), T = \sin(\pi tx) + \cos(\pi ty)$, with $\nu = \kappa = \rho = q = \hat{\mathbf{g}} = [0, 1]^\top$. The following figure shows the convergence of the solution in L^2 at $t_f = 1$ with $\delta t = 0.001$, using mixed boundary conditions for the velocity field and Dirichlet boundary conditions for the temperature.

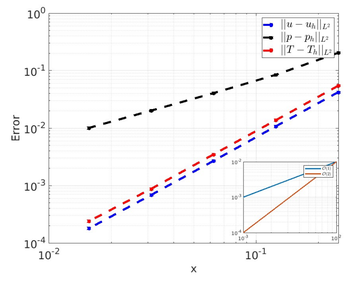
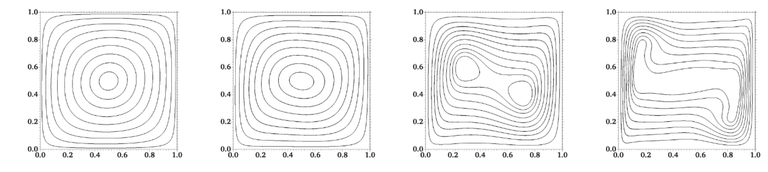


Figure 3: *Convergence of smooth solution*

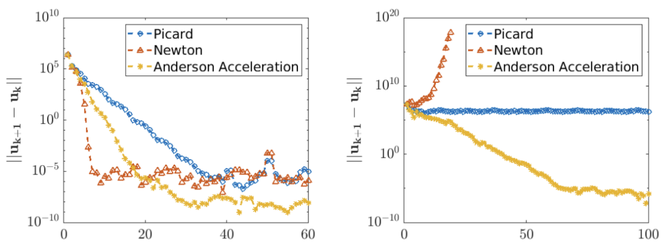
1.4.5 Benchmark: Natural Convection in a Square Cavity

For $\Omega = [0, 1]^2$, we run for various Rayleigh numbers, $Ra = \frac{\beta g \Delta T D^3}{\kappa\nu}$, where β is the expansion coefficient, g is the gravitational acceleration, $\Delta T = T_{hot} - T_{cold}$ is the temperature difference between the two walls, D is the length of the cavity, and κ is the thermal diffusivity.
We use the following boundary conditions: $\mathbf{u} = 0$ on $\Gamma, T = T_{\text{hot}}$ at $x = 0, T = T_{\text{cold}}$ at $x = 1, \partial_n T = 0$ at $y = 0, 1$.



(a) $Ra = 10^3$ (b) $Ra = 10^4$ (c) $Ra = 10^5$ (d) $Ra = 10^6$

Figure 4: *Velocity Streamline*



(a) $Ra = 10^5$ (b) $Ra = 10^6$

Figure 5: *Comparison of various iterative schemes*

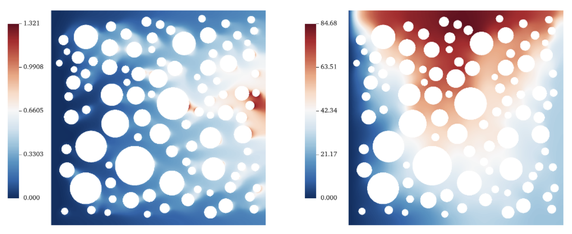
1.4.6 Pore-scale study

The pore geometry and the boundary conditions are shown in the following figure:

$$\mathbf{u} = \begin{bmatrix} 4Uy(0.01 - y) \\ 0 \end{bmatrix}, \quad \begin{matrix} \mathbf{u} = 0, \frac{\partial T}{\partial n} = 0 \\ \frac{\partial T}{\partial n} = 0 \\ \mathbf{u} = 0, \frac{\partial T}{\partial n} = 0 \end{matrix}, \quad (2\nu\epsilon(\mathbf{u}) - \tfrac{1}{\rho}p\mathbf{I})\mathbf{n} = 0, \quad \frac{\partial T}{\partial n} = 0$$

Figure 6: *Gemoetry*

On the solid matrix, we specify a no-slip boundary condition for the velocity, $\mathbf{u} = 0$, and a Neumann boundary condition for the temperature, $\frac{\partial T}{\partial n} = 50$.



(a) U = 4/100 (b) U = 1/25600

Figure 7: *Temperature distribution*

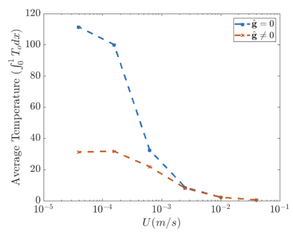
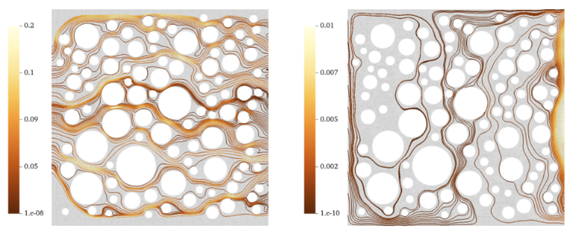


Figure 8: *Average temperature at the outlet*



(a) $U = 4/100$ (b) $U = 1/25600$

Figure 9: *Streamline*

Reference

- Yi, S. Y., Hu, X., Lee, S., & Adler, J. H. (2022). An enriched Galerkin method for the Stokes equations. Computers & Mathematics with Applications, 120, 115-131.
- Hu, X., Lee, S., Mu, L., & Yi, S. Y. (2024). Pressure-robust enriched Galerkin methods for the Stokes equations. Journal of Computational and Applied Mathematics, 436, 115449.