Mixed Finite Element Formulations of the Poisson Equation

Kazi Ahmed & Brian Cornille

Engineering Physics

April 27, 2017

Mixed Finite Element Methods

- Introduce extra independent variables to the differential equation
- For some poorly conditioned finite element representations, a mixed formulation can be better conditioned
- Simultaneously solves for the extra independent variable, which may be desired

Poisson Equation

$$(D)\begin{cases} -\Delta p = f \\ p\big|_{\partial\Omega} = 0 \end{cases} \xrightarrow{\text{introduce } \vec{u}} (D^*)\begin{cases} \vec{u} = -\nabla p \\ \nabla \cdot \vec{u} = f \\ p\big|_{\partial\Omega} = 0 \end{cases}$$

Introducing $H(\nabla\times)$ and $H(\nabla\cdot)$

$H(\nabla\times)$ for $\Omega \in \mathbb{R}^3$

- $H(\nabla\times;\Omega) := \left\{\vec{w} \;:\; \vec{w} \in \left[L^2(\Omega)\right]^3, \nabla \times \vec{w} \in \left[L^2(\Omega)\right]^3\right\}$
- $\|\vec{w}\|^2_{\nabla\times,\Omega} := |\vec{w}|^2_{0,\Omega} + |\nabla \times \vec{w}|^2_{0,\Omega}$

$H(\nabla\cdot)$ for $\Omega \in \mathbb{R}^n$

- $H(\nabla\cdot;\Omega) := \left\{\vec{w} \;:\; \vec{w} \in \left[L^2(\Omega)\right]^n, \nabla \cdot \vec{w} \in L^2(\Omega)\right\}$
- $\|\vec{w}\|^2_{\nabla\cdot,\Omega} := |\vec{w}|^2_{0,\Omega} + |\nabla \cdot \vec{w}|^2_{0,\Omega}$

Formulations of the Poisson Equation

Standard

$$(D)\begin{cases} -\Delta p = f \\ p|_{\partial\Omega} = 0 \end{cases} \iff (V)\begin{cases} \text{Find } p \in H_0^1 \text{ s.t.} \\ \langle \nabla p, \nabla q\rangle = \langle f, q\rangle \qquad \forall q \in H_0^1 \end{cases}$$

Mixed

$$(D^*)\begin{cases} \vec{u} = -\nabla p \\ \nabla \cdot \vec{u} = f \\ p|_{\partial\Omega} = 0 \end{cases} \iff (V)\begin{cases} \text{Find } \vec{u} \in H(\nabla\cdot),\ p \in L^2 \text{ s.t.} \\ -\langle \vec{u}, \vec{v}\rangle + \langle p, \nabla\cdot\vec{v}\rangle = 0 \quad \forall \vec{v} \in H(\nabla\cdot) \\ \langle \nabla\cdot u, q\rangle = \langle f, q\rangle \qquad \forall q \in L^2 \end{cases}$$

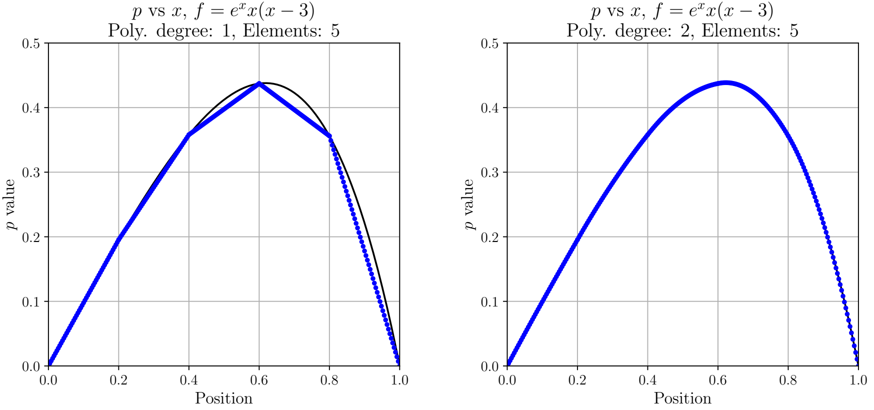
Dual-Mixed

$$(D^*)\begin{cases} \vec{u} = -\nabla p \\ \nabla \cdot \vec{u} = f \\ p|_{\partial\Omega} = 0 \end{cases} \iff (V)\begin{cases} \text{Find } \vec{u} \in H(\nabla\times),\ p \in H_0^1 \text{ s.t.} \\ \langle \vec{u}, \vec{v}\rangle + \langle \nabla p, \vec{v}\rangle = 0 \qquad \forall \vec{v} \in H(\nabla\times) \\ \langle \vec{u}, \nabla q\rangle = -\langle f, q\rangle \qquad \forall q \in H_0^1 \end{cases}$$

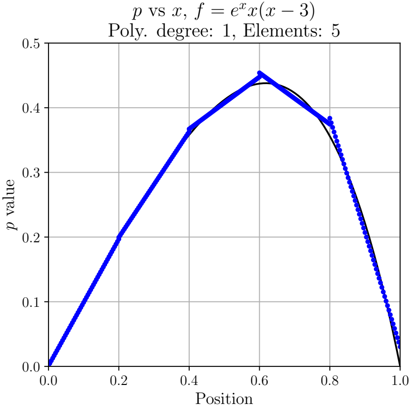
Implementation of the 1D Solver (C++)

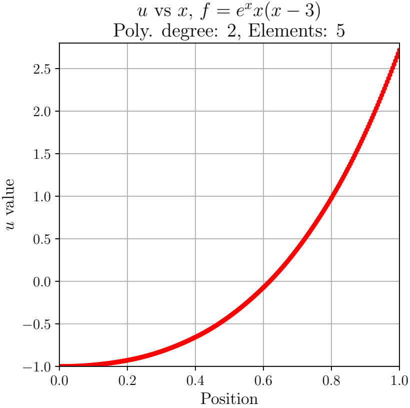
- Evenly spaced elements on $\Omega = [0, 1]$
- Basis functions (arbitrary polynomial order):
  - $H^1(\Omega)$: Lagrange cardinal functions at the Gauss-Legendre-Lobatto quadrature points
  - $L^2(\Omega)$: Lagrange cardinal functions at the Gauss-Legendre quadrature points
- Numerical integration via Gauss-Legendre quadrature
- Linear algebra library: Eigen (`eigen.tuxfamily.org`)
- Elements can be "curved" in 1D by stretching elements with a polynomial transformation
- I/O library: JSON for Modern C++ (`github.com/nlohmann/json`)

Sample 1D Results: Standard

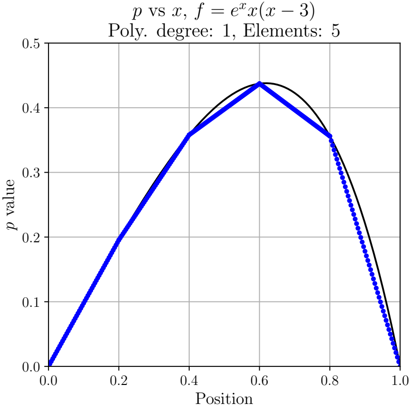


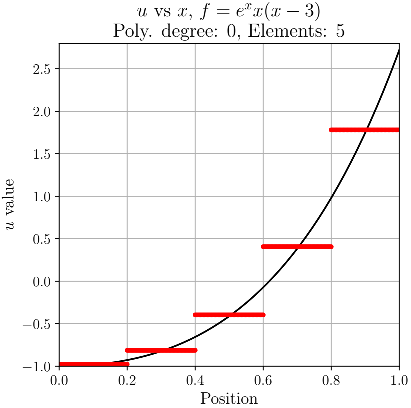
Sample 1D Results: Mixed



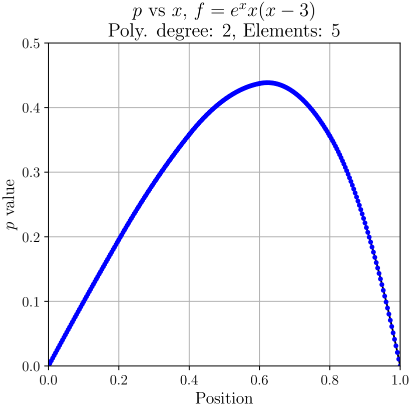


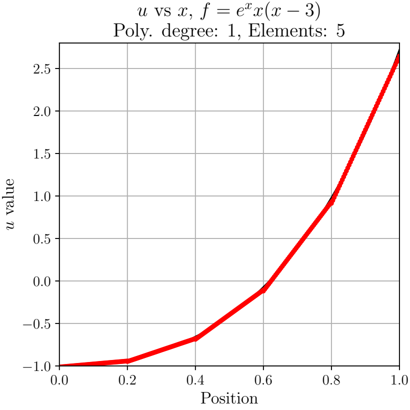
Sample 1D Results: Dual-Mixed



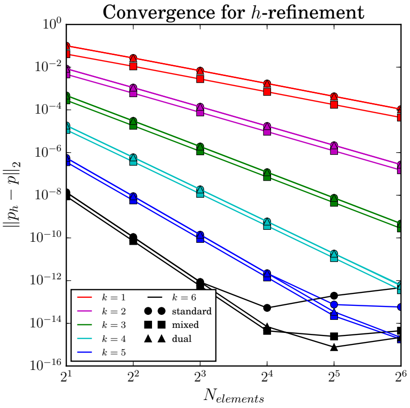


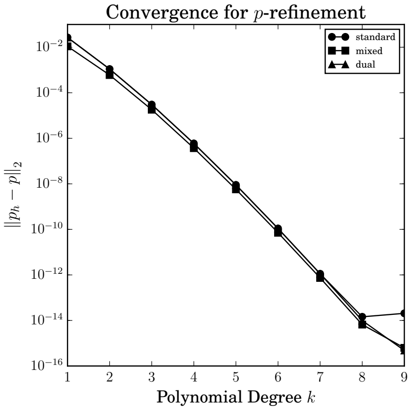
Sample 1D Results: Dual-Mixed





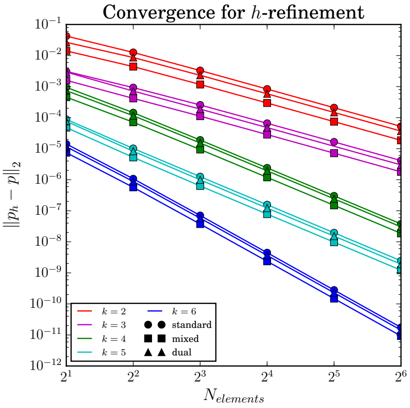
Convergence on a “Straight” Mesh

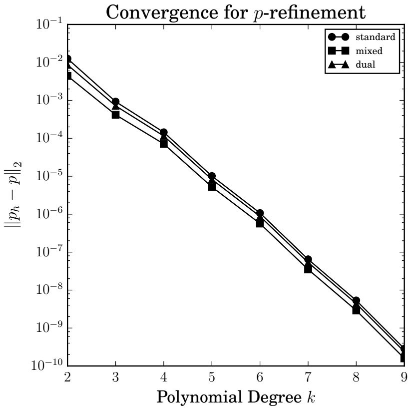




$$N_{elements} = 4$$

Convergence on a "Curved" Mesh





$$N_{elements} = 4$$

Conclusion and Extension

- We have successfully implemented 1D Poisson solver for the formulations we have outlined
- For "straight" elements the Standard and Dual-Mixed formulations are equivalent for p
- This degeneracy is broken for "curved" elements and Dual-Mixed becomes more accurate

- Attempt to recover optimal convergence for a more appropriately "curved" mesh
- If feasible, extend our code to 2D, so we can explore $H(\nabla\cdot)$ and $H(\nabla\times)$ basis implementations