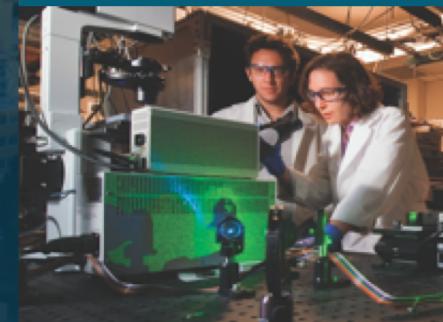


# Guest Lecture Stanford ME469: Splitting and Stabilization Errors



*PRES*ENTED BY

Stefan P. Domino

Computational Thermal and Fluid Mechanics

Sandia National Laboratories SAND2018-4536 PE



Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

## Splitting and Stabilization Errors: Outline



- Block Matrix and Operator Form
- Approximate Factorization
- Splitting Errors
- Stabilization Errors
- Detailed Code Verification
- Conclusions

## Introduction to Block Matrix Form



- Consider the divergence of a vector,  $\mathbf{DF}$  (a scalar)

$$\frac{\partial F_j}{\partial x_j} = 0 \quad \left\{ \begin{array}{l} \int \frac{\partial F_j}{\partial x_j} dV = 0 \\ \int w \frac{\partial F_j}{\partial x_j} d\Omega = 0 \end{array} \right\} \xrightarrow{\text{(Gauss-Divergence)}} \int F_j n_j dS = 0 \quad \xleftarrow{\text{(piecewise-const } w\text{)}}$$

$$-\int F_j \frac{\partial w}{\partial x_j} d\Omega + \int w F_j n_j d\Gamma = 0$$

- Gradient of a scalar,  $\mathbf{Gp}$  (a vector)

$$\frac{\partial p}{\partial x_i} = 0 \quad \left\{ \begin{array}{l} \int \frac{\partial p}{\partial x_i} dV = 0 \\ \int w \frac{\partial p}{\partial x_i} d\Omega = 0 \end{array} \right\} \xrightarrow{\text{(Green-Gauss)}} \int p n_i dS = 0 \quad \xleftarrow{\text{(piecewise-const } w\text{)}}$$

$$-\int p \frac{\partial w}{\partial x_i} d\Omega + \int w p n_i d\Gamma = 0$$

- Laplace operator,  $\mathbf{L}_\lambda T$

$$\frac{\partial q_j}{\partial x_j} = 0 \quad \left\{ \begin{array}{l} \int \frac{\partial q_j}{\partial x_j} dV = 0 \\ \int w \frac{\partial q_j}{\partial x_j} d\Omega = 0 \end{array} \right\} \xrightarrow{\text{(Gauss-Divergence)}} -\int \lambda \frac{\partial T}{\partial x_j} n_j dS = 0 \quad \xleftarrow{\text{(piecewise-const } w\text{)}}$$

$$-\int q_j \frac{\partial w}{\partial x_j} d\Omega + \int w q_j n_j d\Gamma = 0$$

## Introduction to Block Matrix Form for Fluids



- Consider the monolithic, uniform density, low-Mach equation system:

$$\begin{aligned}\frac{\partial u_j}{\partial x_j} &= 0 \\ \frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_j u_i}{\partial x_j} - \frac{\partial \rho \tau_{ij}}{\partial x_j} &= -\frac{\partial p_i}{\partial x_i} + S_i\end{aligned}$$

that can be written in block form as:

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}$$

- We seek to factorize this system via:

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \approx \begin{bmatrix} A & 0 \\ D & B_1 \end{bmatrix} \begin{bmatrix} I & B_2 \\ 0 & I \end{bmatrix} \approx \begin{bmatrix} A & AB_2G \\ D & (B_1 + DB_2G) \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}$$

the exact factorization can be recovered by defining:

$$\left. \begin{array}{l} B_2 = A^{-1} \\ B_1 = -DB_2G \end{array} \right\}$$

- $\mathbf{B}_2$  determines the projection time scale, choose to approximate  $\mathbf{A}^{-1}$
- $\mathbf{B}_1$  controls the projection error, choose to cancel  $\mathbf{B}\mathbf{D}_2\mathbf{G}$

## Introduction to Block Matrix Form



- The approximate factorization

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \approx \begin{bmatrix} A & 0 \\ D & B_1 \end{bmatrix} \begin{bmatrix} I & B_2 \\ 0 & I \end{bmatrix} = \begin{bmatrix} A & AB_2G \\ D & (B_1 + DB_2G) \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}$$

- can be written now as two segregated steps:

Momentum and Continuity:  $\begin{bmatrix} A & 0 \\ D & B_1 \end{bmatrix} \begin{bmatrix} \hat{u} \\ \hat{p} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}$

$$\left. \begin{array}{l} A\hat{u} = f \\ D\hat{u} + B_1\hat{p} = 0 \end{array} \right\}$$

Nodal Projection:  $\begin{bmatrix} I & B_2 \\ 0 & I \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} \hat{u} \\ \hat{p} \end{bmatrix}$

$$\left. \begin{array}{l} u^{n+1} = \hat{u} - B_2p^{n+1} = 0 \\ p^{n+1} = \hat{p} \end{array} \right\}$$

- This approach seems to be straight forward, however, what errors have we introduced by this procedure of splitting the monolithic (fully coupled) system?

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} + \begin{bmatrix} (I - AB_2)Gp^{n+1} \\ -(B_1 + DB_2G)p^{n+1} \end{bmatrix} \quad \text{Exact iff: } \begin{array}{l} B_2 = A^{-1} \\ B_1 = -DB_2G \end{array}$$

- In most cases,  $\mathbf{B}_2$  is approximately  $\mathbf{A}^{-1}$  and a first-order temporal splitting error is noted

## Incremental Pressure-Projection without Pressure Stabilization

6



- Let the inverse of  $\mathbf{A}$ ,  $\mathbf{A}^{-1}$  be approximated by  $\mathbf{B}_2$  as a scalar,  $\tau$
- Let  $\mathbf{B}_1$  be equal to the Laplace operator,  $-\mathbf{L}$

Momentum and Continuity: 
$$\begin{bmatrix} A & 0 \\ D & -\tau L \end{bmatrix} \begin{bmatrix} \hat{u} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} \hat{f} \\ -\tau L p^n \end{bmatrix} \quad \left\{ \begin{array}{l} A\hat{u} = f - Gp^n \\ D\hat{u} - \tau L(p^{n+1} - p^n) = 0 \end{array} \right.$$

Nodal Projection: 
$$\begin{bmatrix} I & \tau G \\ 0 & I \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} \hat{u} \\ \hat{p} \end{bmatrix} + \begin{bmatrix} \tau G p^n \\ 0 \end{bmatrix} \quad \left\{ \begin{array}{l} u^{n+1} = \hat{u} - \tau G(p^{n+1} - p^n) \\ p^{n+1} = \hat{p} \end{array} \right.$$

- The new splitting and stabilization error is given by:

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} + \begin{bmatrix} (I - A\tau)G(p^{n+1} - p^n) \\ \tau(L - DG)(p^{n+1} - p^n) \end{bmatrix}$$

Examples:

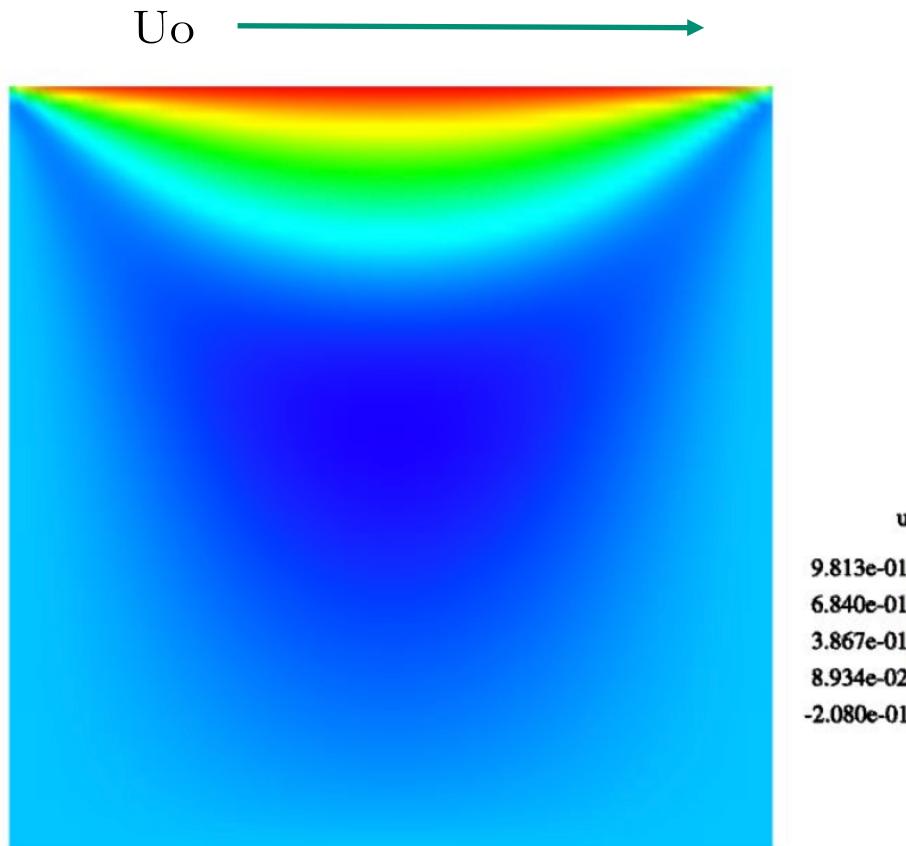
- Dwyer (1990)
- Almgren (2000)

- The above can be shown to demonstrate second-order temporal error (coming)
- A scheme can be designed such that  $\mathbf{L} = \mathbf{D}\mathbf{G}$  (staggered)
- A scheme in which  $\mathbf{L} \neq \mathbf{D}\mathbf{G}$  (collocated or equal-order) can show that  $\mathbf{L}-\mathbf{D}\mathbf{G} \sim 4^{\text{th}}$ -order pressure stabilization

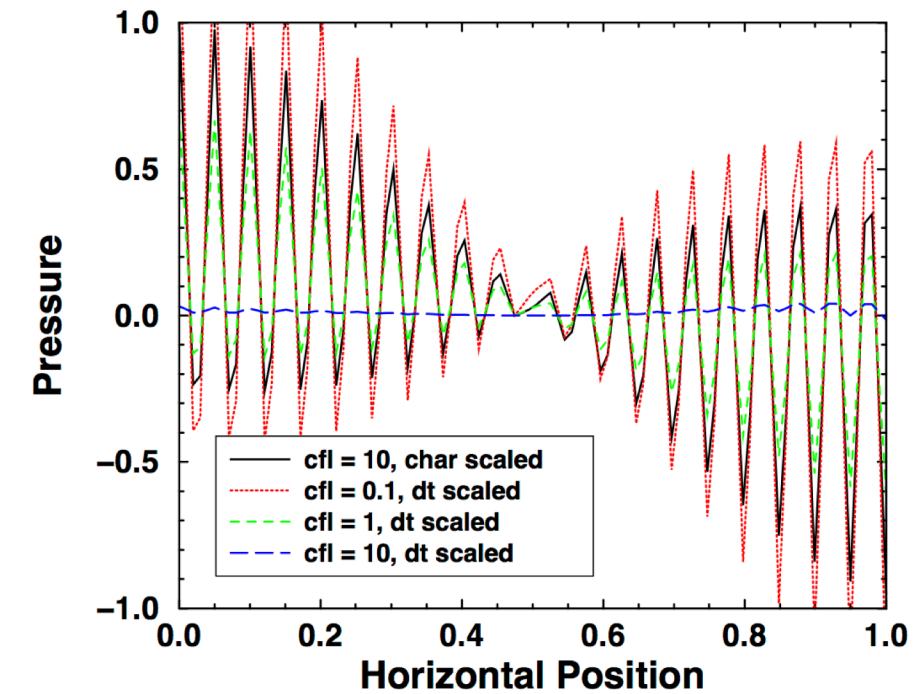
# The Role of Pressure Stabilization ( $L \neq DG$ ) in an Equal-Order Approach



- Consider the classic lid-driven cavity flow with top wall velocity of  $U_0$



lid-driven cavity velocity  
(u-component)



Unsmoothed pressure field at various Courant numbers

# Incremental Approximate Pressure-Projection with Pressure Stabilization Errors

8



- Let the inverse of  $\mathbf{A}$ ,  $\mathbf{A}^{-1}$  be approximated by  $\mathbf{B}_2$  as a scalar,  $\tau$  (which is  $\sim$  time scale)
- Let  $\mathbf{B}_1$  be equal to the Laplace operator,  $-\mathbf{L}$

Momentum and Continuity:

$$\begin{bmatrix} A & 0 \\ D & -\tau L \end{bmatrix} \begin{bmatrix} \hat{u} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} \hat{f} \\ -D\tau G p^n \end{bmatrix} \quad \left\{ \begin{array}{l} A\hat{u} = f - Gp^n \\ D\hat{u} = \tau(Lp^{n+1} - DGp^n) \end{array} \right.$$

Nodal Projection:

$$\begin{bmatrix} I & \tau G \\ 0 & I \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} \hat{u} \\ \hat{p} \end{bmatrix} + \begin{bmatrix} \tau G p^n \\ 0 \end{bmatrix} \quad \left\{ \begin{array}{l} u^{n+1} = \hat{u} - \tau G(p^{n+1} - p^n) \\ p^{n+1} = \hat{p} \end{array} \right.$$

- The new splitting and stabilization error is given by:

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} + \begin{bmatrix} (I - A\tau)G(p^{n+1} - p^n) \\ \tau(L - DG)p^{n+1} \end{bmatrix}$$

Examples:

- Rhie-Chow (1983)
- Peric (1985)

- The above can be shown to hold a second-order temporal error (coming)
- Here, due to equal-order interpolation, i.e., collocation of primitives,  $\mathbf{L} \neq \mathbf{DG}$
- Therefore,  $\mathbf{L} \cdot \mathbf{DG} \sim 4^{\text{th}}$ -order pressure stabilization (pressure oscillations damped)
- Therefore, pressure-stabilization remains

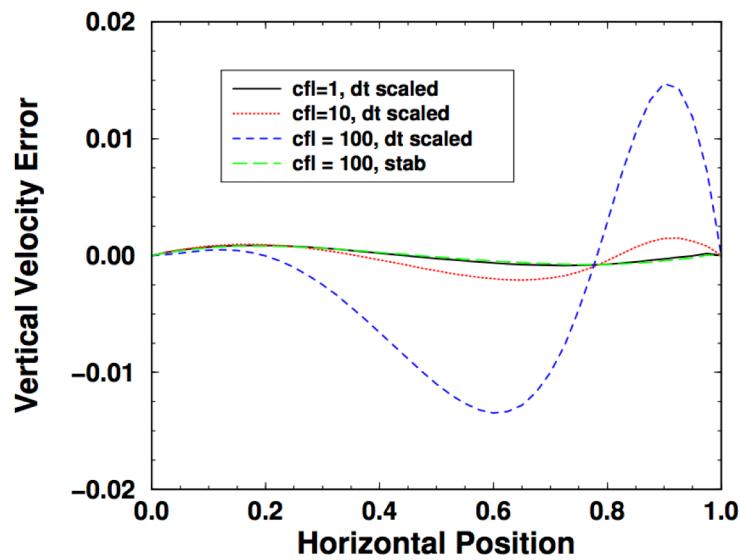
## Sensitivity to chosen $\tau$



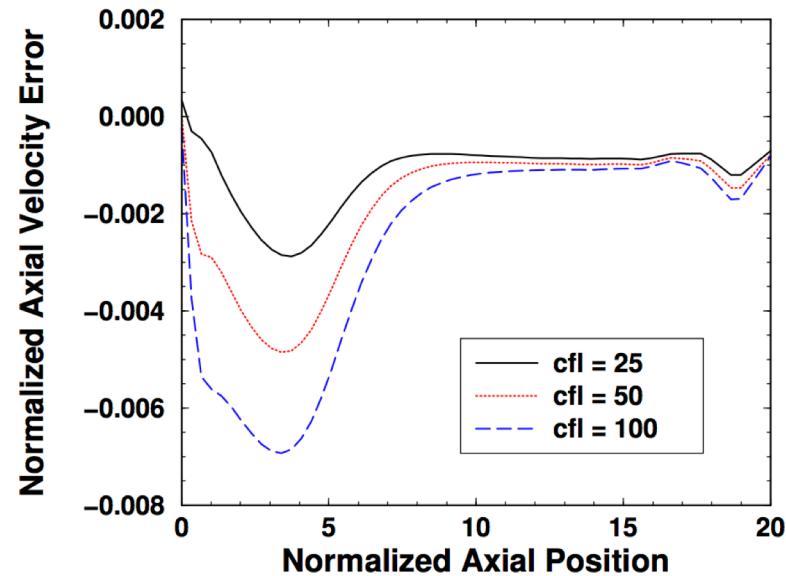
- Recall that the equal-order pressure stabilization error is given by,

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} + \begin{bmatrix} (I - A\tau)G(p^{n+1} - p^n) \\ \tau(L - DG)p^{n+1} \end{bmatrix}$$

- Practical examples of error as a function of  $\tau = \Delta t$  for canonical cavity and jet flow; error relative to either characteristic or stabilized approach of Soto and Lohner



Driven Cavity



Open Jet

## Monolithic Staggered or Equal-order Interpolation (Collocated)



- Note that we need not split the system for a staggered scheme:

$$\begin{bmatrix} A & G \\ D & -\tau L \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ -\tau L p^n \end{bmatrix}$$

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \tau(L - DG)(p^{n+1} - p^n) \end{bmatrix}$$

- or collocated:

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \approx \begin{bmatrix} A & G \\ D & -\tau L \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ -\tau D G p^n \end{bmatrix}$$

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \tau(L - DG)p^{n+1} \end{bmatrix}$$

- Conclusion: Monolithic schemes control splitting error, however, dealing with pressure stabilization is an additional complexity for equal-order methods



## The Choice of the $B_i$ can Vary by The Method..

- In Pressure-Stabilized Petrov-Galerkin methods (Hughes et al, 1985):

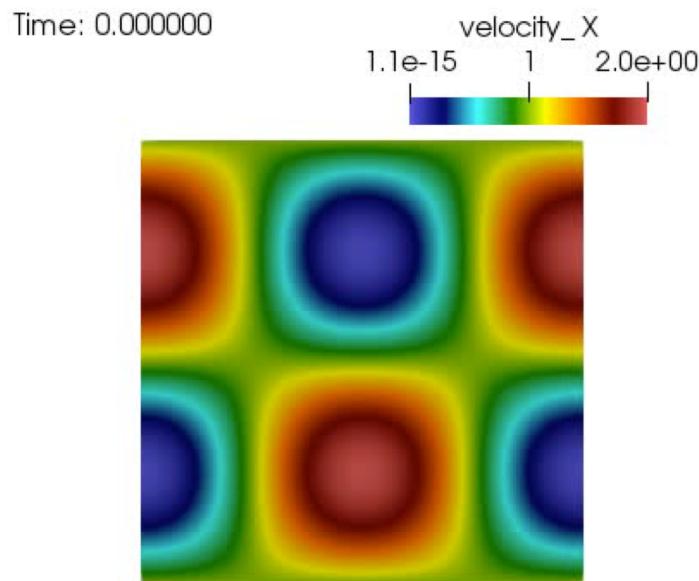
$$\begin{bmatrix} A & G \\ D & -\tau \mathbf{G} \mathbf{w} \cdot \mathbf{M} \end{bmatrix} \begin{bmatrix} \mathbf{u}^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}$$

- Here,  $\mathbf{M}$  is a fine scale momentum residual and  $\mathbf{w}$  is the test function for the Finite Element Method
- Note that  $\mathbf{M}$  contains a local pressure gradient which, thereby, provides the pressure stabilization
- The fine-scale momentum residual is evaluated locally at the quadrature point and with mesh refinement reduces at a design-order rate
- With some algebra, one can show that  $\mathbf{L} \cdot \mathbf{D} \mathbf{G} \sim \mathbf{M}$
- For more references, see Magumdar, Numerical Heat Transfer, 1988
- $\tau$  can be ~:
  - Simulation time step, i.e.,  $\Delta t$
  - Local advection/diffusion time scale,  $\left( \frac{u}{\Delta x} + \frac{v}{\Delta x^2} \right)^{-1}$  - or a more accurate flow-aligned approach
  - Diagonal of  $A$
  - Full inverse of  $A$  (Ozawa)

## Code Verification To Establish Accuracy



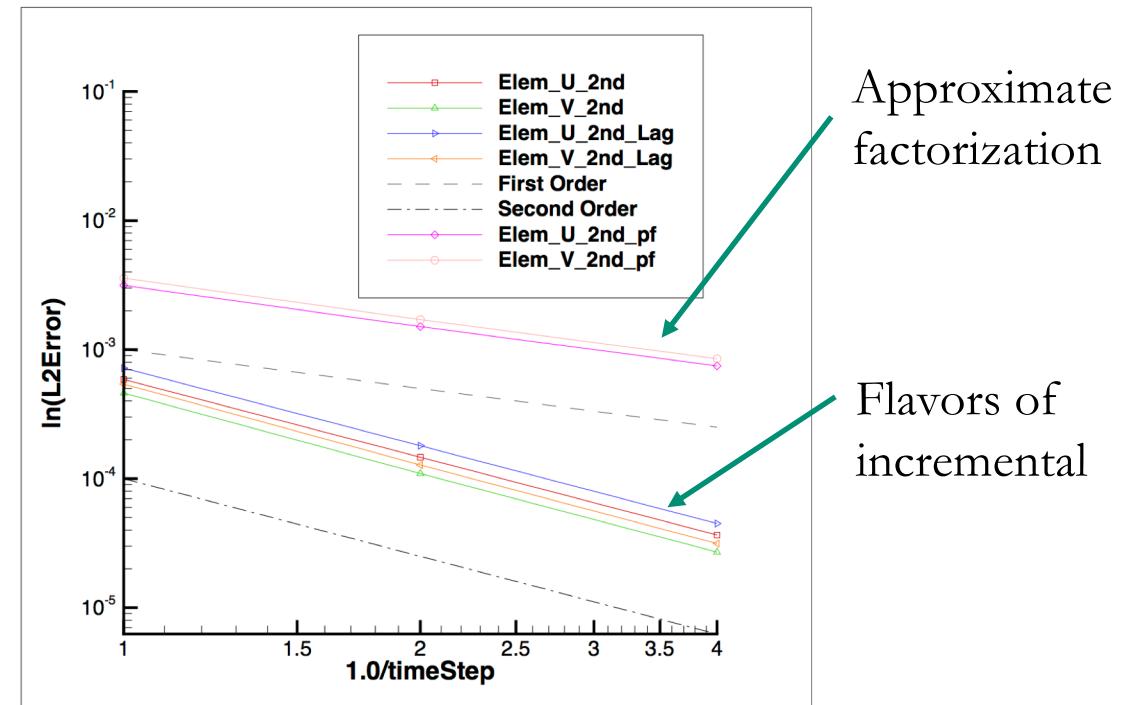
- Consider a two-dimensional transient solution to the incompressible equations of motion:
- This solution is known as the convecting, decaying, Taylor vortex



$$u = u^o - \cos(\pi(x - u^o t)) \sin(\pi(y - v^o t)) e^{-2\omega t}$$

$$v = v^o + \sin(\pi(x - u^o t)) \cos(\pi(y - v^o t)) e^{-2\omega t}$$

$$p = -\frac{p^o}{4} [ (\cos(2\pi(x - u^o t)) + \cos(\pi(y - v^o t))) ] e^{-4\omega t}$$

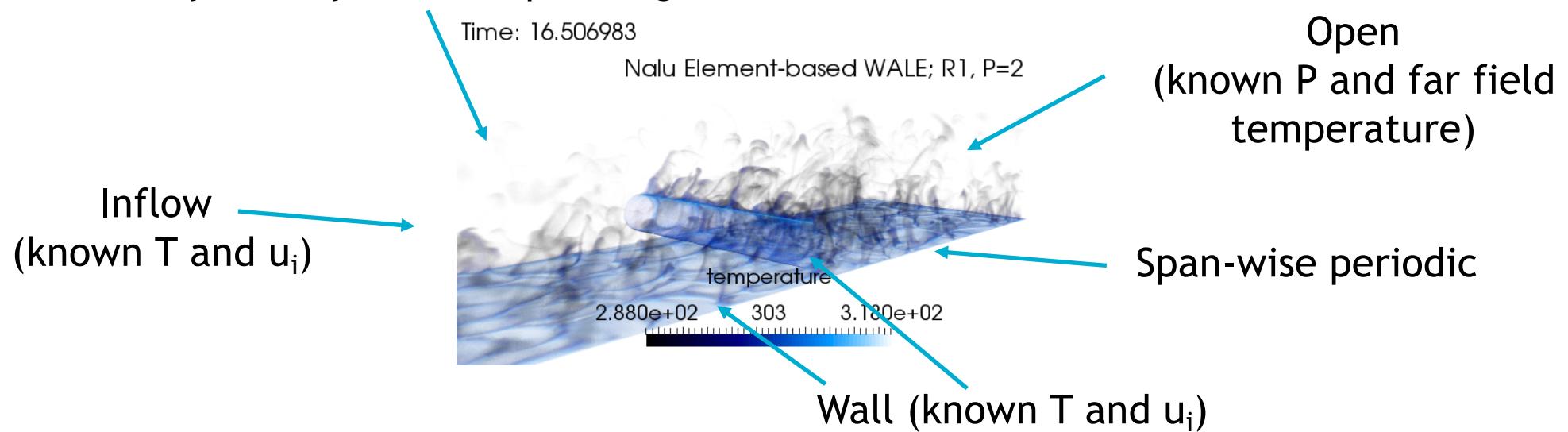


## A note on boundary conditions



- Typical boundaries include the following (mapping to what we know physically in the system):
  - Inflow:
  - Wall: velocity, temperature, scalar flux, etc., are known (“no-slip”)
  - Open: flow leaves or enters the domain with a known “pressure”
  - Symmetry: no-penetration, zero tangential viscous condition
  - Periodic: perfect mapping between nodes at provided surface pairs; provides an infinite domain assumption
  - Non-conformal: static or dynamic/sliding mesh interface in which the surface pair is nonconformal in nature

Symmetry, “flow slips along”

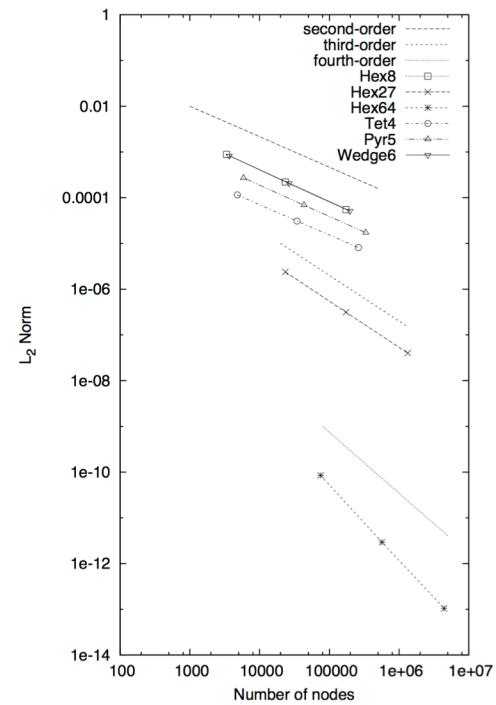
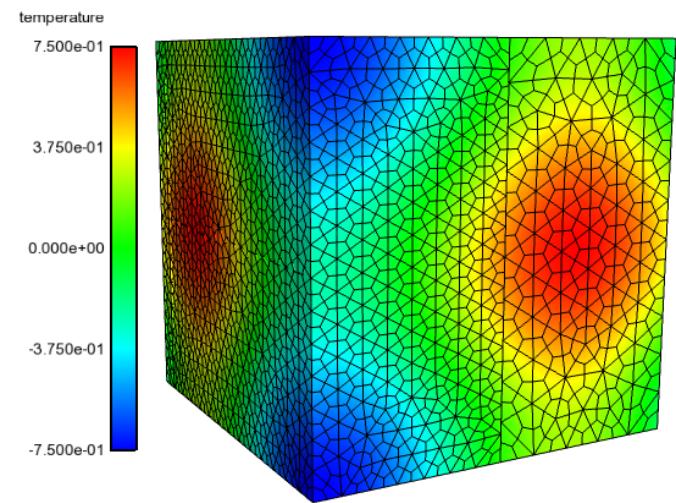


## Weak vs Strong

- For vertex-based schemes, the degree-of-freedom “lives” at the boundary
- In some cases, e.g., wall, inflow and open, boundary condition can be provided by a Dirichlet bc (AKA, strong implementation of a known field).
- A Dirichlet procedure either condenses the row out of the system or zeroes the row with a unity in the diagonal and the specified value on the RHS
- Consider a simple heat conduction solution in which temperature is known at the boundaries
- It can be shown (see Svard and Nordstrom, JCP, 2008) that a stable and accurate weak BC implementation is provided by:

$$q_n = -\lambda \frac{\partial T}{\partial x_j} n_j + \gamma \frac{\lambda}{L} (T - T^{bc})$$

Here,  $\gamma$  is a constant greater than unity



## Splitting and Stabilization Errors: Conclusion



- Block matrix and operator form represents a useful construct to analyze coupling and stabilization
- Approximate Factorization is generally  $O(\Delta t)$
- With very simple modifications, splitting error is mitigated
- Detailed code verification is a critical tool to both test theoretical understandings in addition to establishing a proper code implementation