Block-Structured Adaptive Mesh Refinement

Lecture 2

- Incompressible Navier-Stokes Equations
  - Fractional Step Scheme
- 1-D AMR for “classical” PDE’s
  - hyperbolic
  - elliptic
  - parabolic
- Accuracy considerations
How do we generalize the basic AMR ideas to more general systems?

Incompressible Navier-Stokes equations as a prototype

\[ U_t + U \cdot \nabla U + \nabla p = \epsilon \Delta U \]

\[ \nabla \cdot U = 0 \]

- Advective transport
- Diffusive transport
- Evolution subject to a constraint
Vector field decomposition

Hodge decomposition: Any vector field $V$ can be written as

$$V = U_d + \nabla \phi$$

where $\nabla \cdot U_d = 0$ and $U \cdot n = 0$ on the boundary.

The two components, $U_d$ and $\nabla \phi$ are orthogonal

$$\int U \cdot \nabla \phi \, dx = 0$$

With these properties we can define a projection $P$

$$P = I - \nabla (\Delta^{-1}) \nabla.$$ 

such that

$$U_d = PV$$

with $P^2 = P$ and $\|P\| = 1$. 
Projection form of Navier-Stokes

Incompressible Navier-Stokes equations

\[ U_t + U \cdot \nabla U + \nabla p = \epsilon \Delta U \]
\[ \nabla \cdot U = 0 \]

Applying the projection to the momentum equation recasts the system as an initial value problem

\[ U_t + P(U \cdot \nabla U - \epsilon \Delta U) = 0 \]

Develop a fractional step scheme based on the projection form of equations

Design of the fractional step scheme takes into account issues that will arise in generalizing the methodology to

- More general Low Mach number models
- AMR
Discrete projection

Projection separates vector fields into orthogonal components

\[ V = U_d + \nabla \phi \]

Orthogonality from integration by parts (with \( U \cdot n = 0 \) at boundaries)

\[ \int U \cdot \nabla p \, dx = - \int \nabla \cdot U \, p \, dx = 0 \]

Discretely mimic the summation by parts:

\[ \sum U \cdot GP = - \sum (DU) \, p \]

In matrix form \( D = -G^T \)

Discrete projection

\[ V = U_d + Gp \]

\[ DV = DGp \quad U_d = V - Gp \]

\[ P = I - G(DG)^{-1}D \]
Spatial discretization

Define discrete variables so that \( U, G_p \) defined at the same locations and \( DU, p \) defined at the same locations.

\[
D : V_{space} \rightarrow p_{space} \quad G : p_{space} \rightarrow V_{space}
\]

Candidate variable definitions:
What is the $DG$ stencil corresponding to the different discretization choices

Non-compact stencils $\rightarrow$ decoupling in matrix

Decoupling is not a problem for incompressible Navier-Stokes with homogeneous boundary conditions but it causes difficulties for

- Nontrivial boundary conditions
- Low Mach number generalizations
- AMR

Fully staggered MAC discretization is problematic for AMR

- Proliferation of solvers
- Algorithm and discretization design issues
Approximate projection methods

Based on AMR considerations, we will define velocities at cell-centers

Discrete projection

\[ V = U_d + G_p \]

\[ DV = DG_p \quad U_d = V - G_p \]

\[ P = I - G(DG)^{-1}D \]

Avoid decoupling by replacing inversion of \( DG \) in definition of \( P \) by a standard elliptic discretization.
Approximate projection methods

Analysis of projection options indicates staggered pressure has "best" approximate projection properties in terms of stability and accuracy.

\[
DU_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{u_{i+1,j+1} + u_{i+1,j} - u_{i,j+1} - u_{i,j}}{2\Delta x} \\
+ \frac{v_{i+1,j+1} + v_{i,j+1} - u_{i+1,j} - u_{i,j}}{2\Delta y}
\]

\[
Gp_{ij} = \begin{pmatrix}
\frac{p_{i+1/2,j+1/2} + p_{i+1/2,j-1/2} - p_{i-1/2,j+1/2} - p_{i-1/2,j-1/2}}{2\Delta x} \\
\frac{p_{i+1/2,j+1/2} + p_{i-1/2,j+1/2} - p_{i+1/2,j-1/2} - p_{i-1/2,j-1/2}}{2\Delta y}
\end{pmatrix}
\]

Projection is given by \( P = I - G(L)^{-1}D \)

where \( L \) is given by bilinear finite element basis

\[
(\nabla p, \nabla \chi) = (V, \nabla \chi)
\]

Nine point discretization
2nd Order Fractional Step Scheme

First Step:
Construct an intermediate velocity field $U^*$:

$$\frac{U^* - U^n}{\Delta t} = -[U^{ADV} \cdot \nabla U]^{n+\frac{1}{2}} - \nabla p^{n-\frac{1}{2}} + \epsilon \Delta \frac{U^n + U^*}{2}$$

Second Step:
Project $U^*$ onto constraint and update $p$. Form

$$V = \frac{U^*}{\Delta t} + Gp^{n-\frac{1}{2}}$$

Solve

$$Lp^{n+\frac{1}{2}} = DV$$

Set

$$U^{n+1} = \Delta t(V - Gp^{n+\frac{1}{2}})$$
Computation of Advective Derivatives

- Start with $U^n$ at cell centers
- Predict normal velocities at cell edges using variation of second-order Godunov methodology $\Rightarrow u_{i+1/2,j}^{n+1/2}, v_{i,j+1/2}^{n+1/2}$
- MAC-project the edge-based normal velocities, i.e. solve $D^{MAC}(G^{MAC} \psi) = D^{MAC} U^{n+1/2}$

and define normal advection velocities

\[
\begin{align*}
    u_{i+1/2,j}^{ADV} &= u_{i+1/2,j}^{n+1/2} - G^{x} \psi, \\
    v_{i,j+1/2}^{ADV} &= v_{i,j+1/2}^{n+1/2} - G^{y} \psi
\end{align*}
\]

- Use these advection velocities to define $[U^{ADV} \cdot \nabla U]^{n+1/2}$. 

\[
\begin{array}{ccc}
    \times & \square & \bullet \\
    \times & \times & \times \\
    \bullet & \square & \bullet
\end{array}
\]

$\times$ : $u$
$\square$ : $v$
$\bullet$ : $\psi$
Second-order projection algorithm

Properties
- Second-order in space and time
- Robust discretization of advection terms using modern upwind methodology
- Approximate projection formulation

Algorithm components
- Explicit advection
- Semi-implicit diffusion
- Elliptic projections
  - 5-point cell-centered
  - 9-point node-centered

How do we generalize AMR to work for projection algorithm?

Look at discretization details in one dimension
- Revisit hyperbolic
- Elliptic
- Parabolic

Spatial discretizations
Hyperbolic–1d

Consider \( U_t + F_x = 0 \) discretized with an explicit finite difference scheme:

\[
\frac{U_i^{n+1} - U_i^n}{\Delta t} = \frac{F_{i-1/2}^n - F_{i+1/2}^n}{\Delta x}
\]

In order to advance the composite solution we must specify how to compute the fluxes:

- Away from coarse/fine interface the coarse grid sees the average of fine grid values onto the coarse grid
- Fine grid uses interpolated coarse grid data
- The fine flux is used at the coarse/fine interface
One can advance the coarse grid

\[ \Delta t^f \]

\( (J-1) \quad J \quad J+1 \)

then advance the fine grid

\[ \Delta t^f \]

\( j-1 \quad j \quad (j+1) \)

using “ghost cell data” at the fine level interpolated from the coarse grid data.

This results in a flux mismatch at the coarse/fine interface, which creates an error in \( U_{J}^{n+1} \). The error can be corrected by refluxing, i.e. setting

\[
\Delta x_c U_{J}^{n+1} := \Delta x_c U_{J}^{n+1} - \Delta t^f F_{J-1/2}^c + \Delta t^f F_{j+1/2}^f
\]

Before the next step one must average the fine grid solution onto the coarse grid.
Hyperbolic–subcycling

To subcycle in time we advance the coarse grid with $\Delta t^c$

and advance the fine grid multiple times with $\Delta t^f$.

The refluxing correction now must be summed over the fine grid time steps:

$$\Delta x_c U_j^{n+1} := \Delta x_c U_j^{n+1} - \Delta t^c F_{j-1/2}^c + \sum \Delta t^f F_{j+1/2}^f$$
AMR Discretization algorithms

Design Principles:

- Define what is meant by the solution on the grid hierarchy.
- Identify the errors that result from solving the equations on each level of the hierarchy “independently” (motivated by subcycling in time).
- Solve correction equation(s) to “fix” the solution.
- For subcycling, average the correction in time.

Coarse grid supplies Dirichlet data as boundary conditions for the fine grids.

Errors take the form of flux mismatches at the coarse/fine interface.
Consider \( -\phi_{xx} = \rho \) on a locally refined grid:

\[
\begin{array}{cccccccc}
\Delta x_f & | & | & | & | & | & \Delta x_c \\
\hline
j - 1 & j & J & & J + 1 \\
\end{array}
\]

We discretize with standard centered differences except at \( j \) and \( J \). We then define a flux, \( \phi_{x}^{c-f} \), at the coarse / fine boundary in terms of \( \phi_{x}^{f} \) and \( \phi_{x}^{c} \) and discretize in flux form with

\[
- \frac{1}{\Delta x_f} \left( \phi_{x}^{c-f} - \frac{(\phi_{j}^{f} - \phi_{j-1}^{f})}{\Delta x_f} \right) = \rho_{j}
\]

at \( i = j \) and

\[
- \frac{1}{\Delta x_c} \left( \frac{(\phi_{J+1}^{c} - \phi_{J}^{c})}{\Delta x_c} - \phi_{x}^{c-f} \right) = \rho_{J}
\]

at \( I = J \).

This defines a perfectly reasonable linear system but ...
Elliptic – composite

Suppose we solve

$$-rac{1}{\Delta x_c} \left( \frac{\bar{\phi}_{I+1} - \bar{\phi}_I}{\Delta x_c} - \frac{\bar{\phi}_I - \bar{\phi}_{I-1}}{\Delta x_c} \right) = \rho_I$$

at all coarse grid points $I$ and then solve

$$-rac{1}{\Delta x_f} \left( \frac{\bar{\phi}_{i+1} - \bar{\phi}_i}{\Delta x_f} - \frac{\bar{\phi}_i - \bar{\phi}_{i-1}}{\Delta x_f} \right) = \rho_i$$

at all fine grid points $i \neq j$ and use the “correct” stencil at $i = j$, holding the coarse grid values fixed.
Elliptic – synchronization

The composite solution defined by $\overline{\phi}^c$ and $\overline{\phi}^f$ satisfies the composite equations everywhere except at $J$.

The error is manifest in the difference between $\phi^c_x$ and $\phi^f_x$ and $\frac{(\overline{\phi}_J - \overline{\phi}_{J-1})}{\Delta x_c}$.

Let $e = \phi - \overline{\phi}$. Then $-\Delta^h e = 0$ except at $I = J$ where

$$-\Delta^h e = \frac{1}{\Delta x_c} \left( \frac{(\overline{\phi}_J - \overline{\phi}_{J-1})}{\Delta x_c} - \phi^c_x \right)$$

Solve the composite for $e$ and correct

- $\phi^c = \overline{\phi}^c + e^c$
- $\phi^f = \overline{\phi}^f + e^f$

The resulting solution is the same as solving the composite operator
Consider $u_t + f_x = \varepsilon u_{xx}$ and the semi-implicit time-advance algorithm:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + \frac{f_{i+1/2}^{n+1/2} - f_{i-1/2}^{n+1/2}}{\Delta x} = \frac{\varepsilon}{2} \left((\Delta_u h u_i^{n+1}) + (\Delta h u_i^n)\right)$$

Again if one advances the coarse and fine levels separately, a mismatch in the flux at the coarse-fine interface results.

Let $\bar{u}_c-f$ be the initial solution from separate evolution.
Parabolic – synchronization

The difference $e^{n+1}$ between the exact composite solution $u^{n+1}$ and the solution $\bar{u}^{n+1}$ found by advancing each level separately satisfies

$$(I - \frac{\varepsilon \Delta t}{2} \Delta^h) e^{n+1} = \frac{\Delta t}{\Delta x_c} (\delta f + \delta D)$$

$$\Delta t \delta f = \Delta t (-\bar{f}_{J-1/2} + f_{j+1/2})$$

$$\Delta t \delta D = \frac{\varepsilon \Delta t}{2} \left( \frac{u_{x,J-1/2}^c + u_{x,J-1/2}^{n+1}}{2} - (u_{x,J-1/2}^{c-f,n} + u_{x,J-1/2}^{c-f,n+1}) \right)$$

Source term is localized to coarse cell at coarse / fine boundary

Updating $u^{n+1} = \bar{u}^{n+1} + e$ again recovers the exact composite solution
Parabolic – subcycling

Advance coarse grid

\[ \Delta t^c \]

\[(J-1) \quad J \quad J+1 \]

Advance fine grid \( r \) times

\[ \Delta t^f \]

\[ \Delta t^f \]

\[ \Delta t^f \]

\[ \Delta t^f \]

The refluxing correction now must be summed over the fine grid time steps:

\[
(I - \frac{\varepsilon \Delta t^c}{2} \Delta h) e^{n+1} = \frac{\Delta t^c}{\Delta x_c} (\delta f + \delta D)
\]

\[
\Delta t^c \delta f = -\Delta t^c \bar{f}_{J-1/2}^c + \sum \Delta t^f f_{j+1/2}^f
\]

\[
\Delta t^c \delta D = \frac{\varepsilon \Delta t^c}{2} (\bar{u}_{x,J-1/2}^c + \bar{u}_{x,J-1/2}^{c,n+1})
\]

\[
- \sum \frac{\varepsilon \Delta t^f}{2} (u_{x,J-1/2}^{c,f,n} + u_{x,J-1/2}^{c,f,n+1})
\]
Spatial accuracy – cell-centered

Modified equation gives

\[ \psi^{\text{comp}} = \psi^{\text{exact}} + \Delta^{-1} \tau^{\text{comp}} \]

where \( \tau \) is a local function of the solution derivatives.

Simple interpolation formulae are not sufficiently accurate for second-order operators.
Convergence results

Local Truncation Error

| D | Norm | $\Delta x$ | $||L(U_e) - \rho||_h$ | $||L(U_e) - \rho||_{2h}$ | $R$ | $P$ |
|---|------|-------------|-----------------|-----------------|-----|-----|
| 2 | $L_\infty$ | 1/32 | 1.57048e-02 | 2.80285e-02 | 1.78 | 0.84 |
| 2 | $L_\infty$ | 1/64 | 8.08953e-03 | 1.57048e-02 | 1.94 | 0.96 |
| 3 | $L_\infty$ | 1/16 | 2.72830e-02 | 5.60392e-02 | 2.05 | 1.04 |
| 3 | $L_\infty$ | 1/32 | 1.35965e-02 | 2.72830e-02 | 2.00 | 1.00 |
| 3 | $L_1$ | 1/32 | 8.35122e-05 | 3.93200e-04 | 4.70 | 2.23 |

Solution Error

| D | Norm | $\Delta x$ | $||U_h - U_e||$ | $||U_{2h} - U_e||$ | $R$ | $P$ |
|---|------|-------------|----------------|----------------|-----|-----|
| 2 | $L_\infty$ | 1/32 | 5.13610e-06 | 1.94903e-05 | 3.79 | 1.92 |
| 2 | $L_\infty$ | 1/64 | 1.28449e-06 | 5.13610e-06 | 3.99 | 2.00 |
| 3 | $L_\infty$ | 1/16 | 3.53146e-05 | 1.37142e-04 | 3.88 | 1.96 |
| 3 | $L_\infty$ | 1/32 | 8.88339e-06 | 3.53146e-05 | 3.97 | 1.99 |

\[
\psi_{\text{computed}} = \psi_{\text{exact}} + L^{-1}f
\]

Solution operator smooths the error
Spatial accuracy – nodal

Node-based solvers:

- Symmetric self-adjoint matrix
- Accuracy properties given by approximation theory
Recap

Solving coarse grid then solving fine grid with interpolated Dirichlet boundary conditions leads to a flux mismatch at boundary

Synchronization corrects mismatch in fluxes at coarse / fine boundaries.

Correction equations match the structure of the process they are correcting.

- For explicit discretizations of hyperbolic PDE’s the correction is an explicit flux correction localized at the coarse/fine interface.

- For an elliptic equation (e.g., the projection) the source is localized on the coarse/fine interface but an elliptic equation is solved to distribute the correction through the domain. Discrete analog of a layer potential problem.

- For Crank-Nicolson discretization of parabolic PDE’s, the correction source is localized on the coarse/fine interface but the correction equation diffuses the correction throughout the domain.
Efficiency considerations

For the elliptic solves, we can substitute the following for a full composite solve with no loss of accuracy

- Solve $\Delta \psi^c = g^c$ on coarse grid
- Solve $\Delta \psi^f = g^f$ on fine grid using interpolated Dirichlet boundary conditions
- Evaluate composite residual on the coarse cells adjacent to the fine grids
- Solve for correction to coarse and fine solutions on the composite hierarchy

Because of the smoothing properties of the elliptic operator, we can, in some cases, substitute either a two-level solve or a coarse level solve for the full composite operator to compute the \textit{correction} to the solution.

- Source is localized at coarse cells at coarse / fine boundary
- Solution is a discrete harmonic function in interior of fine grid
- This correction is exact in 1-D