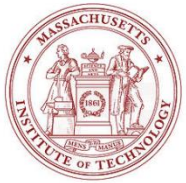
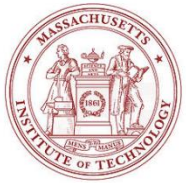


# Deferred-Correction Approaches

- Size of computational molecule affects both storage requirements and effort needed to solve the algebraic system at each time-step
  - Usually, we wish to keep only the nearest neighbors of the center node P in the LHS of equations (leads to tri-diagonal matrix or something close to it) ⇒ easier to solve linear/nonlinear system
  - But, approximations that produce such molecules are often not accurate enough
- Way around this issue?
  - Leave only the terms containing the nearest neighbors in the LHS and bring all other more-remote terms to the RHS
    - This requires that these terms be evaluated with previous or old values, which may lead to divergence of the iterative scheme
- Better approach?



# Deferred-Correction Approaches, Cont'd

- Better Approach
  - Compute the terms that are approximated with a high-order approximation explicitly and put them in the RHS
  - Take a simpler approximation to these terms (that give a small computational molecule). Insert it twice in the equation, with a + and - sign
  - One of these two simpler approximations, keep it in the LHS of the equations (with unknown variables values, i.e. implicit/new). Move the other to the RHS (i.e. computing it explicitly using existing/old values)
  - The RHS now contains the difference between two explicit approximations of the same term, and is likely to be small $\Rightarrow$
    - Likely no convergence problems to an iteration scheme (Jacobi, GS, SOR, etc) or gradient descent (CG, etc)
  - Once the iteration converges, the low order approximation terms (one explicit, the other implicit) drop out and the solution corresponds to the higher-order approximation
- $\Rightarrow$ Using H & L for high & low orders:

$$\mathbf{A}^H \mathbf{x} = \mathbf{b} \quad \rightarrow \mathbf{A}^L \mathbf{x} = \mathbf{b} - \left[ \mathbf{A}^H \mathbf{x} - \mathbf{A}^L \mathbf{x} \right]^{\text{old}}$$

# Deferred-Correction Approaches, Cont'd

- This approach can be very powerful and general
  - Used when treating higher-order approximations, non-orthogonal grids, corrections needed to avoid oscillation effects, etc
  - Since RHS can be viewed as a correction $\Rightarrow$ called deferred-correction
  - Note: both L&H terms could be implicit in time: use L&H explicit starter to get first values and then most recent old values in bracket during iterations (similar to Jacobi vs. Gauss Seidel)
    - Explicit for H (high-order) term, implicit for L (low-order) term

$$\mathbf{A}^H\ \mathbf{x} = \mathbf{b} \quad \rightarrow \mathbf{A}^L\ \mathbf{x}_{\text{implicit}} = \mathbf{b} - \left[\mathbf{A}^H\ \mathbf{x}_{\text{explicit}} - \mathbf{A}^L\ \mathbf{x}_{\text{implicit}}\right]^{\text{old}}$$

    - Implicit for both L and H terms (similar to Gauss-Seidel)

$$\mathbf{A}^H\ \mathbf{x} = \mathbf{b} \quad \rightarrow \mathbf{A}^L\ \mathbf{x}_{\text{implicit}} = \mathbf{b} - \left[\mathbf{A}^H\ \mathbf{x}_{\text{implicit}} - \mathbf{A}^L\ \mathbf{x}_{\text{implicit}}\right]^{\text{old}}$$

# Deferred-Correction Approaches, Cont'd

- Example 1: FD methods with High-order Pade' schemes
  - One can use the PDE itself to express implicit Pade' time derivative $\left(\frac{\partial \phi}{\partial t}\right)_{n+1}$ as a function of $\phi^{n+1}$ (see homework)
  - Or, use deferred-correction (within an iteration scheme of index $r$):
    - In time:
$$\left(\frac{\partial \phi}{\partial t}\right)_n^{r+1} = \left(\frac{\phi_{n+1} - \phi_{n-1}}{2\Delta t}\right)^{r+1} + \left[\left(\frac{\partial \phi}{\partial t}\right)_n^{\text{Pade'}} - \frac{\phi_{n+1} - \phi_{n-1}}{2\Delta t}\right]^r$$
    - In space:
$$\left(\frac{\partial \phi}{\partial x}\right)_i^{r+1} = \left(\frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x}\right)^{r+1} + \left[\left(\frac{\partial \phi}{\partial x}\right)_i^{\text{Pade'}} - \frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x}\right]^r$$
    - The complete 2nd order CDS would be used on the LHS. The RHS would be the bracket term: the difference between the Pade' scheme and the "old" CDS. When the CDS becomes as accurate as Pade', this term in the bracket is zero
    - Note: Forward/Backward DS could have been used instead of CDS, e.g. in time,
$$\left(\frac{\partial \phi}{\partial t}\right)_{n+1}^{r+1} = \left(\frac{\phi_{n+1} - \phi_n}{\Delta t}\right)^{r+1} + \left[\left(\frac{\partial \phi}{\partial t}\right)_{n+1}^{\text{Pade'}} - \frac{\phi_{n+1} - \phi_n}{\Delta t}\right]^r$$

# Deferred-Correction Approaches, Cont'd

- Example 2 with FV methods: Higher-order Flux approximations
  - *Higher-order* flux approximations are computed with "old values" and a lower order approximation is used with "new values" (implicitly) in the linear system solver:

$$F_e = F_e^L + \left[ F_e^H - F_e^L \right]^{\text{old}}$$

where $F_e$ is the flux. For ex., the low order approximation is a UDS or CDS

- Convergence and stability properties are close to those of the low order implicit term since the bracket is often small compared to this implicit term
- In addition, since bracket term is small, the iteration in the algebraic equation solver can converge to the accuracy of higher-order scheme
- Additional numerical effort is explicit with "old values" and thus much smaller than the full implicit treatment of the higher-order terms

  - A factor can be used to produce a mixture of pure low and pure high order. This can be used to remove undesired properties, e.g. oscillations of high-order schemes

$$F_e = \omega F_e^L + (1-\omega)\left[ F_e^H - F_e^L \right]^{\text{old}}$$