REVIEW Lecture 10:

• Direct Methods for solving (linear) algebraic equations
  – Gauss Elimination
  – LU decomposition/factorization
  – Error Analysis for Linear Systems and Condition Numbers
  – Special Matrices (Tri-diagonal, banded, sparse, positive-definite, etc)

• Iterative Methods:
  \[ x^{k+1} = B x^k + c \]
  \[ k = 0, 1, 2, ... \]

  – Jacobi’s method
  \[ x^{k+1} = -D^{-1} (L + U) x^k + D^{-1} b \]

  – Gauss-Seidel iteration
  \[ x^{k+1} = -(D + L)^{-1} U x^k + (D + L)^{-1} b \]
REVIEW Lecture 10, Iterative Methods Cont’d:

- **Convergence:**
  \[ \rho(B) = \max_{i=1...n} |\lambda_i| < 1, \text{ where } \lambda_i = \text{eigenvalue}(B_{n \times n}) \]  
  (ensures \( ||B|| < 1 \))

  - Jacobi’s method
  - Gauss-Seidel iteration
    \[ i \leq n_{\text{max}} \]
    \[ \|x_i - x_{i-1}\| \leq \varepsilon \]
    \[ \|r_i - r_{i-1}\| \leq \varepsilon, \text{ where } r_i = Ax_i - b \]

- **Stop Criteria:**
  \[ \|r_i\| \leq \varepsilon \]

- **Example**

- **Successive Over-Relaxation Methods:** (decrease \( \rho(B) \) for faster convergence)
  \[ x_{i+1} = (D + \omega L)^{-1} \left[ -\omega U + (1 - \omega)D \right] x_i + \omega (D + \omega L)^{-1} b \]

- **Gradient Methods**
  \[ x_{i+1} = x_i + \alpha_{i+1} v_{i+1} \]
  - Steepest decent
  \[ x_{i+1} = x_i + \left( \begin{array}{c} r_i^T r_i \\ r_i^T Ar_i \end{array} \right) r_i \]
  \[ \frac{dQ(x)}{dx} = Ax - b = -r \]
  \[ r_i = b - Ax_i \text{ (residual at iteration } i) \]
  - Conjugate gradient
TODAY (Lecture 11)

• End of (Linear) Algebraic Systems
  – Gradient Methods and Krylov Subspace Methods
  – Preconditioning of $Ax=b$

• FINITE DIFFERENCES
  – Classification of Partial Differential Equations (PDEs) and examples with finite difference discretizations
  – Error Types and Discretization Properties
    • Consistency, Truncation error, Error equation, Stability, Convergence
  – Finite Differences based on Taylor Series Expansions
    • Higher Order Accuracy Differences, with Example
    • Taylor Tables or Method of Undetermined Coefficients
  – Polynomial approximations
    • Newton’s formulas, Lagrange/Hermite Polynomials, Compact schemes
Gradient Methods

- Applicable to physically important matrices: “symmetric and positive definite” ones
- Construct the equivalent optimization problem

\[ Q(x) = \frac{1}{2} x^T A x - x^T b \]

\[ \frac{dQ(x)}{dx} = A x - b \]

\[ \frac{dQ(x_{opt})}{dx} = 0 \Rightarrow x_{opt} = x_e, \text{ where } A x_e = b \]

- Propose step rule

\[ x_{i+1} = x_i + \alpha_{i+1} v_{i+1} \]

- Common methods
  - Steepest descent
  - Conjugate gradient
Steepest Descent Method

• Move exactly in the negative direction of Gradient

\[ \frac{dQ(x)}{dx} = Ax - b = -(b - Ax) = -r \]

\( r \): residual, \( r_i = b - Ax_i \)

• Step rule

\[ x_{i+1} = x_i + \frac{r_i^T r_i}{r_i^T A r_i} r_i \]

• \( Q(x) \) reduces in each step, but not as effective as conjugate gradient method

Graph showing the steepest descent method.
Conjugate Gradient Method

- Definition: “\(A\)-conjugate vectors” or “Orthogonality with respect to a matrix (metric)”: if \(A\) is symmetric & positive definite,
  
  For \(i \neq j\) we say \(v_i, v_j\) are orthogonal with respect to \(A\), if \(v_i^T A v_j = 0\)

- Proposed in 1952 (Hestenes/Stiefel) so that directions \(v_i\) are generated by the orthogonalization of residuum vectors (search directions are \(A\)-conjugate)
  - Choose new descent direction as different as possible from old ones, within \(A\)-metric

- Algorithm:

\[
\begin{align*}
v_0 &= r_0 = b - Ax_0 \\
\text{do} & \quad \text{Step length} \\
\alpha_i &= (v_i^T r_i)/(v_i^T Av_i) & \text{Approximate solution} \\
x_{i+1} &= x_i + \alpha_i v_i & \text{New Residual} \\
r_{i+1} &= r_i - \alpha_i Av_i & \text{Improved step length} \\
\beta_i &= -(v_i^T Ar_{i+1})/(v_i^T Av_i) & \text{new search direction} \\
v_{i+1} &= r_{i+1} + \beta_i v_i & \\
\text{until a stop criterion holds}
\end{align*}
\]

Figure indicates solution obtained using Conjugate gradient method (red) and steepest descent method (green).
Conjugate Gradient (CG) Method and Krylov Subspace Methods

• Conjugate Gradient Properties
  – Accurate solution with “n” iterations, but decent accuracy with much fewer number of iterations
  – Only matrix or vector products
  – Is a special case of Krylov subspace algorithms for symmetric PD matrices

• Krylov Subspaces for $Ax=b$: Definitions and Properties
  – Krylov sequence: the set of vectors $b, Ab, A^2 b, \cdots$
  – Krylov subspace of size $n$ is: $K_n = \text{span}\{b, Ab, \cdots, A^{n-1} b\}$
  – The sequence converges towards the eigenvector with the largest eigenvalue
  – Vectors become more and more linearly dependent
  – Hence, if one extracts an orthogonal basis for the subspace, one would likely get good approximations of the top eigenvectors with the $n$ largest eigenvalues
  – An iteration to do this is the “Arnoldi’s iteration” which is a stabilized Gram-Schmidt procedure (e.g. see Trefethen and Bau, 1997)
Conjugate Gradient (CG) Method and Krylov Subspace Methods

• CG method is a Krylov Subspace method for PD matrices:
  – The search/residual vectors of CG span the Krylov subspace
  – Hence, intermediate solutions of CG method $x_n$ are in $K_n = \text{span} \left\{ b, Ab, \ldots, A^{n-1}b \right\}$

• Krylov Subspace methods
  – Based on the idea of projecting the “$Ax=b$ problem” into the Krylov subspace of smaller dimension $n$
  – Provide variations of CG for non-symmetric non-singular matrices
    • Generalized Minimal Residual (GMRES) or MINRES (for sym. but non P.D. $A$)
      – Approximates the solution $Ax=b$ by the vector $x_n \in K_n$ that minimizes the norm of the residual $Ax_n - b$
    • (Stabilized) bi-conjugate gradients (BiCGstab)
    • Quasi-minimal residual
  – See Trefethen and Bau, 1997, Asher and Grief, 2011; and other refs
    • This field is full of acronyms!
Preconditioning of $A \, x = b$

• Pre-conditioner approximately solves $A \, x = b$.

Pre-multiply by the inverse of a non-singular matrix $M$, and solve instead:

$$M^{-1} A \, x = M^{-1} \, b \quad \text{or} \quad A \, M^{-1} (M \, x) = b$$

– Convergence properties based on $M^{-1} A$ or $A \, M^{-1}$ instead of $A$!
– Can accelerate subsequent application of iterative schemes
– Can improve conditioning of subsequent use of non-iterative schemes: GE, LU, etc

• Jacobi preconditioning:
  – Apply Jacobi a few steps, usually not efficient

• Other iterative methods (Gauss-Seidel, SOR, SSOR, etc):
  – Usually better, sometimes applied only once

• Incomplete factorization (incomplete LU)

• Coarse-Grid Approximations and Multigrid Methods:
  – Solve $A \, x = b$ on a coarse grid (or successions of coarse grids)
  – Interpolate back to finer grid(s)
Example of Convergence Studies for Linear Solvers

Fig 7.5: Example 7.10, with N=3: convergence behavior of various iterative schemes for the discretized Poisson equation.

Fig 7.7: Iteration progress for CG, PCG with the IC(0) preconditioner and PCG with the IC preconditioner using drop tolerance tol=0.01

Ascher and Greif, Siam-2011

Courtesy of Society for Industrial and Applied Mathematics (SIAM). Used with permission.
Useful reference tables for this material:
FINITE DIFFERENCES - Outline

- Classification of Partial Differential Equations (PDEs) and examples with finite difference discretizations
  - Elliptic PDEs
  - Parabolic PDEs
  - Hyperbolic PDEs
- Error Types and Discretization Properties
  - Consistency, Truncation error, Error equation, Stability, Convergence
- Finite Differences based on Taylor Series Expansions
- Polynomial approximations
  - Equally spaced differences
    - Richardson extrapolation (or uniformly reduced spacing)
    - Iterative improvements using Roomberg’s algorithm
  - Lagrange polynomial and un-equally spaced differences
  - Compact Difference schemes
References and Reading Assignments


From Mathematical Models to Numerical Simulations

Continuum Model

\[ \frac{\partial w}{\partial t} + c \frac{\partial w}{\partial x} = 0 \]

Sommerfeld Wave Equation (c= wave speed). This radiation condition is sometimes used at open boundaries of ocean models.

Discrete Model

\[ t_m = t_0 + m \Delta t, \quad m = 0, 1, \ldots M - 1 \]
\[ x_n = x_0 + n \Delta x, \quad n = 0, 1, \ldots N - 1 \]

\[ \frac{dw}{dx} \approx \frac{\Delta w}{\Delta x}, \quad \frac{dw}{dt} \approx \frac{\Delta w}{\Delta t} \]

Consistency/Accuracy and Stability => Convergence

(Lax equivalence theorem for well-posed linear problems)

Differential Equation

\[ L(p, w, x, t) = 0 \]

“Differentiation”

“Integration”

Difference Equation

\[ L_{mn}(p_{mn}, w_{mn}, x_n, t_m) = 0 \]

System of Equations

\[ \sum_{j=0}^{N-1} F_i(w_j) = B_i \]

Linear System of Equations

\[ \sum_{j=0}^{N-1} A_{ij} w_j = B_i \]

“Solving linear equations”

Eigenvalue Problems

\[ \overline{A} u = \lambda u \iff (\overline{A} - \lambda \overline{I}) u = 0 \]

Non-trivial Solutions

\[ \det(\overline{A} - \lambda \overline{I}) = 0 \]

“Root finding”

2.29 Numerical Fluid Mechanics
Classification of Partial Differential Equations
(2D case, 2\textsuperscript{nd} order)

Quasi-linear PDE for $\phi(x, y)$

$$A\phi_{xx} + B\phi_{xy} + C\phi_{yy} = F(x, y, \phi, \phi_x, \phi_y)$$

A, B and C Constants

$B^2 - 4AC > 0$ Hyperbolic

$B^2 - 4AC = 0$ Parabolic

$B^2 - 4AC < 0$ Elliptic

(Only valid for two independent variables: x,y)

- In general: $A$, $B$ and $C$ are function of: $x, y, \phi, \phi_x, \phi_y$
- Equations may change of type from point to point if $A$, $B$ and $C$ vary with $x$, $y$, . etc
- Navier-Stokes, incomp., const. viscosity:

$$\frac{Du}{Dt} = \frac{\partial u}{\partial t} + (u \cdot \nabla) u = -\frac{1}{\rho} \nabla p + \nu \nabla^2 u + g$$
Classification of Partial Differential Equations
(2D case, 2nd order)

Meaning of Hyperbolic, Parabolic and Elliptic

• The general 2nd order PDE in 2D:
  \[ A\phi_{xx} + B\phi_{xy} + C\phi_{yy} = F \]

  is analogous to the equation for a conic section:
  \[ Ax^2 + Bxy + Cy^2 = F \]

• Conic section:
  - Is the intersection of a right circular cone and a plane, which generates a group of plane curves, including the circle, ellipse, hyperbola, and parabola
  - One characterizes the type of conic sections using the discriminant \( B^2 - 4AC \)

• PDE:
  - \( B^2 - 4AC > 0 \) (Hyperbolic)
  - \( B^2 - 4AC = 0 \) (Parabolic)
  - \( B^2 - 4AC < 0 \) (Elliptic)

Images courtesy of puk on Wikipedia. License: CC-BY.
Partial Differential Equations

Parabolic PDE: \[ B^2 - 4AC = 0 \]

Examples

\[
\frac{\partial T}{\partial t} = \frac{\kappa}{\sigma \rho} \nabla^2 T + f, \quad (\alpha = \frac{\kappa}{\sigma \rho})
\]

\[
\frac{\partial \mathbf{u}}{\partial t} = \nu \nabla^2 \mathbf{u} + \mathbf{g}
\]

- Usually smooth solutions (“diffusion effect” present)
- “Propagation” problems
- Domain of dependence of \( u \) is domain \( D \) (\( x, y, 0 < t < \infty \)):

- Finite Differences/Volumes, Finite Elements

Heat conduction equation, forced or not (dominant in 1D)

Unsteady, diffusive, small amplitude flows or perturbations (e.g. Stokes Flow)
Partial Differential Equations
Parabolic PDE

Heat Flow Equation
\[ \kappa T_{xx}(x,t) = \sigma \rho T_t(x,t), \quad 0 < x < L, 0 < t < \infty \]

Initial Condition
\[ T(x,0) = f(x), \quad 0 \leq x \leq L \]

Boundary Conditions
\[ T(0,t) = c_1, \quad 0 < t < \infty \]
\[ T(L,t) = c_2, \quad 0 < t < \infty \]

\( \kappa \) Thermal conductivity
\( \sigma \) Specific heat
\( \rho \) Density
\( T \) Temperature

IVP in one dimension \((t)\), BVP in the other \((x)\)
Time Marching, Explicit or Implicit Schemes

IVP: Initial Value Problem
BVP: Boundary Value Problem
Partial Differential Equations
Parabolic PDE

Heat Flow Equation

\[ T_t(x,t) = c^2 T_{xx}(x,t), \quad 0 < x < L, \quad 0 < t < \infty \]

\[ c = \sqrt{\frac{\kappa}{\rho \sigma}} \]

Initial Condition

\[ T(x,0) = f(x), \quad 0 \leq x \leq L \]

Boundary Conditions

\[ T(0,t) = g_1(t), \quad 0 < t < \infty \]

\[ T(L,t) = g_2(t), \quad 0 < t < \infty \]
Partial Differential Equations
Parabolic PDE

Equidistant Sampling

\[
\begin{align*}
    h &= \frac{L}{n} \\
    k &= \frac{T}{m}
\end{align*}
\]

Discretization

\[
\begin{align*}
    x_i &= (i-1)h, \quad i = 2, \ldots, n-1 \\
    t_j &= (j-1)k, \quad j = 1, \ldots, m
\end{align*}
\]

Forward (Euler) Finite Difference

\[
T_t(x,t) = \frac{T(x_{i+1},t_j) - T(x_i,t_j)}{k} + O(k)
\]

Centered Finite Difference

\[
T_{xx}(x,t) = \frac{T(x_{i-1},t_j) - 2T(x_i,t_j) + T(x_{i+1},t_j)}{h^2} + O(h^2)
\]

\[
T_{i,j} = T(x_i,t_j)
\]

Finite Difference Equation

\[
\frac{T_{i+1,j} - T_{i,j}}{k} = c^2 \frac{T_{i-1,j} - 2T_{i,j} + T_{i+1,j}}{h^2}
\]
Partial Differential Equations

ELLiptic: $B^2 - 4AC < 0$

Quasi-linear PDE

$$A\phi_{xx} + B\phi_{xy} + C\phi_{yy} = F(x, y, \phi, \phi_x, \phi_y)$$

A, B and C Constants

- $B^2 - 4AC > 0$ Hyperbolic
- $B^2 - 4AC = 0$ Parabolic
- $B^2 - 4AC < 0$ Elliptic
2.29 Numerical Fluid Mechanics
Fall 2011

For information about citing these materials or our Terms of Use, visit: http://ocw.mit.edu/terms.