Outline

• Recap from Wednesday
• Root Finding
  – Bisection
  – Newton-Raphson method
  – Brent’s method
• Optimization
  – Golden Section Search
  – Brent’s Method
  – Downhill Simplex
  – Conjugate gradient methods
  – Multiple level, single linkage (MLSL)
Recap from Wednesday

• Solve linear algebra problems $A^{-1}$ and $A \cdot x = b$
• Gauss-Jordan method ($A' \cdot x = b'$)
• Gaussian Elimination ($U \cdot x = b'$)
• LU Decomposition ($A = L \cdot U$)
• Singular Value Decomposition ($A = U \cdot W \cdot V^T$)
• Sparse Matrices
• Iterative improvement (subtract $A^{-1}(b' - b)$ from $x'$)
• QR Decomposition ($A = \prod Q_i \cdot R$)
Finding Zeros

- Relevance in micro & nano research
- Key concept: bracketing
- Bisection – continuously halve intervals
- Newton-Raphson method – uses tangent
- Laguerre’s method – for polynomials
- Brent’s method – adds inverse quadratic interpolation
Importance of Bracketing

- Critically important for both root finding and optimization
- Can always guarantee at least one solution for continuous functions with sign change in 1D
- If more than one solution present, may not be able to guarantee which one is reached – method-dependent
Bisection

• Most stable and reliable approach

• Algorithm:
  – Choose point $x_3$ in the middle of the bracket with sign change: $[x_1, x_2]$
  – Check sign of $f(x_3)$
  – If non-zero, construct new bracket from midpoint and original point with opposite sign
  – Repeat previous steps
Newton-Raphson Method

• Key assumption: system is nearly linear in region between starting point and root
• When sufficiently close, converge quadratically on correct value (from Taylor expansion)
NR Method Failures

- Getting stuck in a limit cycle is possible
- Can even get worse – certain locally flat curves can send you into outer space!
Laguerre’s Method

- Specifically for polynomials
- Algorithm
  - Calculate quantities $G$ and $H$
  - Assume far roots a distance $b$; one root is a distance $a$ away
  - Iterate solution as $a \to 0$

\[
P_n(x) = \prod_{i}(x - x_i)
\]

\[
G = \frac{d \ln|P_n(x)|}{dx}
\]

\[
H = -\frac{d^2 \ln|P_n(x)|}{dx^2}
\]

\[
a = \frac{n}{G \pm \sqrt{(n - 1)(nH - G^2)}}
\]
Brent’s Method: Finding Roots

• Combines bracketing, bisection, and inverse quadratic interpolation
• Guaranteed to converge, but speed can vary with function and quality of initial guess
• Algorithm:
  – Calculate $f(a)$, $f(b)$, $f(c)$
  – Calculate $R$, $S$, $T$, $P$, $Q$
  – Let $b \rightarrow b + P/Q$
  – Repeat as $f(b) \rightarrow 0$

\[
P = S[T(R - T)(c - b) - (1 - R)(b - a)] \]
\[
Q = (T - 1)(R - 1)(S - 1)\]
Optimization

- Relevance in micro & nano research
- Convexity
- Search classifications
- Techniques:
  - Brent’s Method
  - Golden Section Search
  - Downhill Simplex
  - Conjugate gradient methods
  - Multiple level, single linkage (MLSL)

These and further images from “Numerical Recipes,” by WH Press et al.
Convexity

• Convex functions have certain properties that aid in finding an optimum:
  – Precisely one optimum in an open set of values
  – Continuous and at least twice differentiable
  – Midpoints always lower than edges – i.e.,
    \[ f[\delta x_1 + (1 - \delta) \delta x_2] < \delta f(x_1) + (1 - \delta) f(x_2) \]
• Examples include \( x^2 \), \( \sinh(x) \)
Search Types

• Local – assumes convex/concave problem
• Global – uses heuristics to deal with multiple optima
• Non-derivative based – no specific assumptions about best search direction
• Derivative based – incorporates derivatives to determine search direction
Brent’s Method: Finding Optima

• Assumes a concave function
• Algorithm:
  – Evaluate function at bracket endpoints & center
  – Fit parabola
  – Find $x_{min}$ & $f(x_{min})$
  – Keep two closest points for bracket and repeat until bracket is around $\sqrt{\varepsilon}$
• Infer optimum based
Golden Section Search

- Closely related to bisection approach to finding roots

- Algorithm
  - Taking a downhill step
  - Bracket lowest point with higher values on each side
  - Keep repeating until interval is around $\sqrt{\varepsilon}$
Downhill Simplex Search

- Simplex is a triangle (2D), tetrahedron (3D), etc.

- Algorithm:
  - Create an N-dimensional simplex: $P_i = P_o + \lambda_i \hat{e}_i$
  - Perform one of 4 steps shown on right
  - Repeat until tolerances reached (e.g., for change in simplex end-points, or function values)
Conjugate Gradient Method

- Assumes convex multidimensional function
- Uses derivative information
- Algorithm:
  - Start with initial $g_0 = h_0$
  - Calculate scalars $\lambda_i$, $\gamma_i$
  - Construct new vectors $g_{i+1}$ and $h_{i+1}$, satisfying orthogonality & conjugacy conditions
  - Repeat until tolerance reached
- Note that no \textit{a priori} knowledge of Hessian matrix $A$ is required!

\[
\lambda_i = \frac{g_i \cdot g_i}{h_i \cdot A \cdot h_i} = \frac{g_i \cdot h_i}{h_i \cdot A \cdot h_i}
\]

\[
\gamma_i = \frac{(g_{i+1} - g_i) \cdot g_{i+1}}{g_i \cdot g_i}
\]

\[
g_{i+1} = g_i - \lambda_i A \cdot h_i
\]

\[
h_{i+1} = g_{i+1} + \gamma_i h_i
\]

\[
g_i \cdot g_j = 0
\]

\[
h_i \cdot A \cdot h_j = 0
\]

\[
g_j \cdot h_j = 0
\]
Multiple Level Single Linkage

• Global search

• Algorithm:
  – Quasi-random sequence of starting points
  – Local optimization (e.g., conjugate gradient)
  – Heuristic tracks basins of convergence

Next Class

• Is on Wednesday, Jan. 23 (because of Martin Luther King, Jr. Day)
• Discussion of eigenproblems
• Please read Chapter 11 of “Numerical Recipes” by W.H. Press \textit{et al.}