CS 450: Numerical Anlaysis

Lecture 13

Chapter 5 – Nonlinear Equations

Existence, Conditioning, and 1D Methods for Nonlinear Equations

Edgar Solomonik

Department of Computer Science
University of Illinois at Urbana-Champaign

March 2, 2018

Solving Nonlinear Equations

$f(x) = b \Rightarrow g(x) = f(x) - b = 0$

- Solving (systems of) nonlinear equations corresponds to root finding:
  - $\boxed{f(x) = 0}$ function is scalar-valued: takes input scalar
  - $f(\boldsymbol{x}) = 0$ scalar valued, but vector input
  - $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}$ vector-valued functions

$f(x^*) = 0$

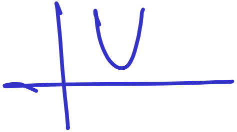
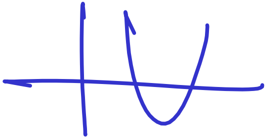
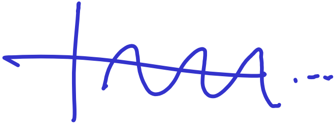
- Root-finding can be reduced to finding a fixed-point $\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{x}$:

$g(x^*) = x^*$

e.g. $g(x) = f(x) + x$ | $g_2(x) = \frac{f(x)}{2} + x$

Nonexistence and Nonuniqueness of Solutions

- Solutions do not generally exist and are not generally unique, even in the univariate case:

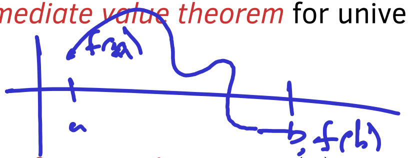
- Solutions in the multivariate case correspond to intersections of hypersurfaces:

every nonlinear equation defines hypersurface

$$f(x) = 0 = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \end{bmatrix}$$

Conditions under which Solutions Exist

- *Intermediate value theorem* for univerate problems:



requires continuity

bracket (interval $[a, b]$) that contains a root

- *Inverse function theorem* $\boldsymbol{J_f}(\boldsymbol{x})$ is nonsingular at $\boldsymbol{x}$ if $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}$:

$f'(x^*) \neq 0$

$J = J_f(x)$

$J_{ij} = \frac{\partial f_i}{\partial x_j}(x^*)$

- If a function has a unique fixed point in a given closed domain if it is is *contractive* and contained in that domain,

$$||\boldsymbol{g}(\boldsymbol{x}) - \boldsymbol{g}(\boldsymbol{z})|| \leq \gamma ||\boldsymbol{x} - \boldsymbol{z}||$$

unique fixed point

$\forall x \in S \quad g(x) \in S$

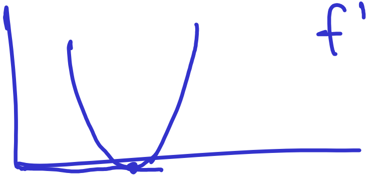
Multiple Roots and Degeneracy

- If x is a root of f with multiplicity m,
  $f(x) = f'(x) = f''(x) = \cdots = f^{(m-1)}(x) = 0$:

$$f(x) = (x-3)^2 \cdot g(x)$$

$$f'(x) = 2(x-3)\,g(x) + (x-3)^2 \cdot g'(x)$$

- Increased multiplicity affects conditioning and convergence:



$$f'(x^*) = 0$$

Conditioning of Nonlinear Equations

- Generally, we take interest in the absolute rather than relative conditioning of solving $\boldsymbol{f}(\boldsymbol{x}) = \mathbf{0}$:

residual $|f(\hat{x})|$

computed solution

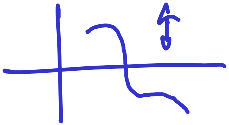
$\kappa_{abs}(f) = f'(x)$ at x

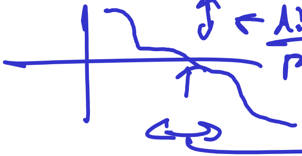
$\boxed{\kappa_{rel} = \frac{f'(x)\,x}{f(x)}}$ 0

- The condition number of solving f with respect to solution x is $1/|f'(x)|$ or $||\boldsymbol{J}_{\boldsymbol{f}}^{-1}(\boldsymbol{x})||$ for $\boldsymbol{f}$ at $\boldsymbol{x}$:

better conditioned

reciprocal relationship to function evaluation



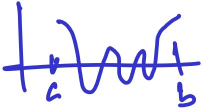


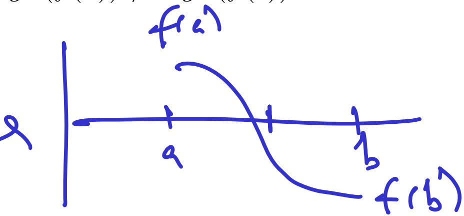
$\frac{\text{dilation}}{\text{perturbation in input}}$

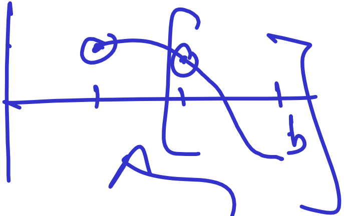
how much does root change

Bisection Algorithm

- Assume we know the desired root exists in a bracket $[a, b]$ and $sign(f(a)) \neq sign(f(b))$:







- Bisection subdivides the interval by a factor of two at each step by considering $f((b-a)/2)$:

$$sign\left(f\left(\frac{a+b}{2}\right)\right) = \begin{cases} sign(f(a)) : & \left[\frac{a+b}{2}, b\right] \\ sign(f(b)) : & \left[a, \frac{a+b}{2}\right] \end{cases}$$

Rates of Convergence

- Let x_k be the kth iterate and $e_k = x_k = x^*$ be the error, bisection obtains *linear convergence*, $\lim_{k\to\infty} ||e_k||/||e_{k-1}|| \leq C$:

$e_k = x_k - x^*$

next

step

$\frac{1}{2}$

- rth order convergence implies that $||e_k||/||e_{k-1}||^r \leq C$

linear $r = 1$

superlinear $r > 1$

quadratic is $r = 2$

Convergence of Fixed Point Iteration

- Fixed point iteration: $x_{k+1} = g(x_k)$ is locally linearly convergent if for $x^* = g(x^*)$, we have $|g'(x^*)| < 1$:

local convergence $\exists x_0$ close $\neq x$, $g(g(x_0)) \to x^*$

$$e_k = g(x_k) - x^* = g(x_k) - g(x^*)$$

$$= g'(\hat{x})(\underbrace{x_k}_{g(x_{k-1})} - x) = g'(\hat{x})\, e_{k-1}$$

$\exists \hat{x} \in [x_k, x^*]$

$g'(\hat{x})(x_k - x) = g(x_k) - g(x)$

- It is quadratically convergent if $g'(x^*) = 0$:

$\bar{x} \in [x_k, x^*]$

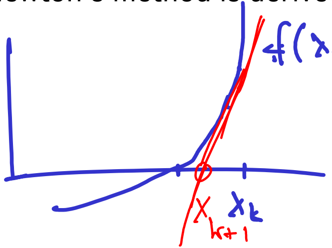
$$e_k = g''(\bar{x})\underbrace{(x_k - x)^2}_{e_{k-1}}/2$$

$$\frac{e_k}{e_{k-1}^2} = \frac{g''(x)}{2} \le C$$

Newton's Method

$$x_{k+1} = x_k - f(x_k)/f'(x_k)$$

- Newton's method is derived from a *Taylor series* expansion of f at x_k:



$$f(x) = f(x_k) + f'(x_k)(x-x_k) + \underbrace{O((x-x_k)^2)}_{\text{makes sense if } x-x_k \ll 1}$$

$$\boxed{-f(x_k) = f'(x_k)\cdot(x-x_k)}$$

- Newton's method is *quadratically convergent* if started sufficiently close to x^* so long as $f'(x^*) \neq 0$:

$$f(x) \approx f(x_k) + f'(x_k)(x-x_k) = 0$$

$$-f(x_k) = f'(x_k)\cdot(x-x_k)$$

$$x = x_k - f(x_k)/f'(x_k)$$

$$g(x) = x - f(x)/f'(x)$$

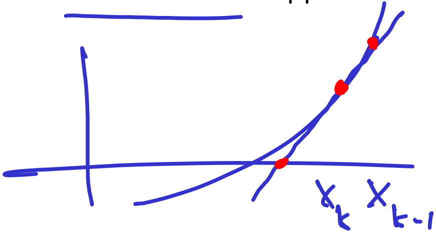
$$g'(x^*) = 1 - \underbrace{f'(x^*)/f'(x^*)}_{} - \frac{\overset{=0}{f(x^*)}\, f''(x^*)}{(f'(x^*))^2}$$

$$g'(x^*) = 0$$

Secant Method

$x_{k+1} = x_k - f(x_k)/f'(x_k)$ $\qquad \hat{f}'(x)$

- The Secant method approximates $f'(x_k) \approx \frac{f(x_k)-f(x_{k-1})}{x_k - x_{k-1}}$: $=$



instead approximate $f'(x_k)$ based on $f(x_k+h)$ and $f(x_k-h)$ for small h

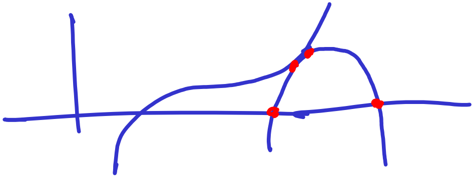
- The convergence is *superlinear* but not quadratic:

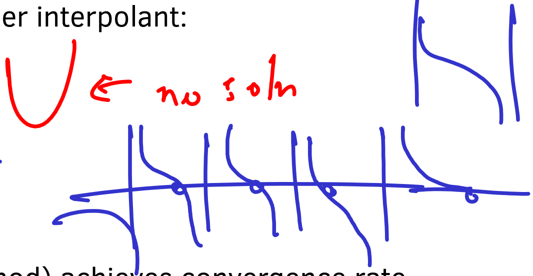
$\frac{e_{k+1}}{e_{k-1}e_k} \le c$ by comp. Newton $\frac{e_{k+1}}{e_k^2} \le c$

$\Longrightarrow r = (1+\sqrt{5})/2$

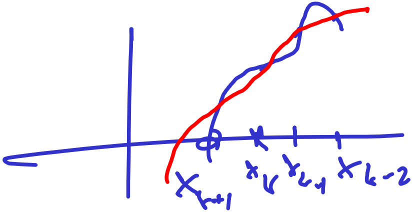
Nonlinear Tangential Interpolants

- Secant method uses a linear interpolant based on points $f(x_k)$, $f(x_{k-1})$, could use more points and higher-order interpolant:





- Quadratic interpolation (Muller's method) achieves convergence rate $r \approx 1.84$:



Achieving Global Convergence

- Hybrid bisection/Newton methods:

try Newton

make progress?

if not, try bisection

- Bounded (damped) step-size:

if not $\quad x_{k+1} = x_k - \alpha f(x_k)/f'(x_k)$

$\alpha < 1$