

Numerical solutions of nonlinear systems of equations

Tsung-Ming Huang

Department of Mathematics
National Taiwan Normal University, Taiwan
E-mail: min@math.ntnu.edu.tw

August 28, 2011



Outline

- 1 Fixed points for functions of several variables
- 2 Newton's method
- 3 Quasi-Newton methods
- 4 Steepest Descent Techniques



Fixed points for functions of several variables

Theorem 1

Let $f : D \subset \mathbb{R}^n \rightarrow \mathbb{R}$ be a function and $x_0 \in D$. If all the partial derivatives of f exist and $\exists \delta > 0$ and $\alpha > 0$ such that $\forall \|x - x_0\| < \delta$ and $x \in D$, we have

$$\left| \frac{\partial f(x)}{\partial x_j} \right| \leq \alpha, \quad \forall j = 1, 2, \dots, n,$$

then f is continuous at x_0 .

Definition 2 (Fixed Point)

A function G from $D \subset \mathbb{R}^n$ into \mathbb{R}^n has a fixed point at $p \in D$ if $G(p) = p$.

Theorem 3 (Contraction Mapping Theorem)

Let $D = \{(x_1, \dots, x_n)^T; a_i \leq x_i \leq b_i, \forall i = 1, \dots, n\} \subset \mathbb{R}^n$.

Suppose $G : D \rightarrow \mathbb{R}^n$ is a continuous function with $G(x) \in D$ whenever $x \in D$. Then G has a fixed point in D .

Suppose, in addition, G has continuous partial derivatives and a constant $\alpha < 1$ exists with

$$\left| \frac{\partial g_i(x)}{\partial x_j} \right| \leq \frac{\alpha}{n}, \quad \text{whenever } x \in D,$$

for $j = 1, \dots, n$ and $i = 1, \dots, n$. Then, for any $x^{(0)} \in D$,

$$x^{(k)} = G(x^{(k-1)}), \quad \text{for each } k \geq 1$$

converges to the unique fixed point $p \in D$ and

$$\|x^{(k)} - p\|_\infty \leq \frac{\alpha^k}{1 - \alpha} \|x^{(1)} - x^{(0)}\|_\infty.$$



Example 4

Consider the nonlinear system

$$\begin{aligned} 3x_1 - \cos(x_2x_3) - \frac{1}{2} &= 0, \\ x_1^2 - 81(x_2 + 0.1)^2 + \sin x_3 + 1.06 &= 0, \\ e^{-x_1x_2} + 20x_3 + \frac{10\pi - 3}{3} &= 0. \end{aligned}$$

- Fixed-point problem:

Change the system into the fixed-point problem:

$$\begin{aligned} x_1 &= \frac{1}{3} \cos(x_2x_3) + \frac{1}{6} \equiv g_1(x_1, x_2, x_3), \\ x_2 &= \frac{1}{9} \sqrt{x_1^2 + \sin x_3 + 1.06} - 0.1 \equiv g_2(x_1, x_2, x_3), \\ x_3 &= -\frac{1}{20} e^{-x_1x_2} - \frac{10\pi - 3}{60} \equiv g_3(x_1, x_2, x_3). \end{aligned}$$

Let $G : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ be defined by $G(x) = [g_1(x), g_2(x), g_3(x)]^T$.



- G has a unique point in $D \equiv [-1, 1] \times [-1, 1] \times [-1, 1]$:

- Existence: $\forall x \in D$,

$$|g_1(x)| \leq \frac{1}{3} |\cos(x_2 x_3)| + \frac{1}{6} \leq 0.5,$$

$$|g_2(x)| = \left| \frac{1}{9} \sqrt{x_1^2 + \sin x_3 + 1.06} - 0.1 \right| \leq \frac{1}{9} \sqrt{1 + \sin 1 + 1.06} - 0.1 < 0$$

$$|g_3(x)| = \frac{1}{20} e^{-x_1 x_2} + \frac{10\pi - 3}{60} \leq \frac{1}{20} e + \frac{10\pi - 3}{60} < 0.61,$$

it implies that $G(x) \in D$ whenever $x \in D$.

- Uniqueness:

$$\left| \frac{\partial g_1}{\partial x_1} \right| = 0, \quad \left| \frac{\partial g_2}{\partial x_2} \right| = 0 \quad \text{and} \quad \left| \frac{\partial g_3}{\partial x_3} \right| = 0,$$

as well as

$$\left| \frac{\partial g_1}{\partial x_2} \right| \leq \frac{1}{3} |x_3| \cdot |\sin(x_2 x_3)| \leq \frac{1}{3} \sin 1 < 0.281,$$



$$\begin{aligned} \left| \frac{\partial g_1}{\partial x_3} \right| &\leq \frac{1}{3} |x_2| \cdot |\sin(x_2 x_3)| \leq \frac{1}{3} \sin 1 < 0.281, \\ \left| \frac{\partial g_2}{\partial x_1} \right| &= \frac{|x_1|}{9\sqrt{x_1^2 + \sin x_3 + 1.06}} < \frac{1}{9\sqrt{0.218}} < 0.238, \\ \left| \frac{\partial g_2}{\partial x_3} \right| &= \frac{|\cos x_3|}{18\sqrt{x_1^2 + \sin x_3 + 1.06}} < \frac{1}{18\sqrt{0.218}} < 0.119, \\ \left| \frac{\partial g_3}{\partial x_1} \right| &= \frac{|x_2|}{20} e^{-x_1 x_2} \leq \frac{1}{20} e < 0.14, \\ \left| \frac{\partial g_3}{\partial x_2} \right| &= \frac{|x_1|}{20} e^{-x_1 x_2} \leq \frac{1}{20} e < 0.14. \end{aligned}$$

These imply that g_1 , g_2 and g_3 are continuous on D and $\forall x \in D$,

$$\left| \frac{\partial g_i}{\partial x_j} \right| \leq 0.281, \quad \forall i, j.$$

Similarly, $\partial g_i / \partial x_j$ are continuous on D for all i and j . Consequently, G has a unique fixed point in D .



- Approximated solution:

- Fixed-point iteration (I):

Choosing $x^{(0)} = [0.1, 0.1, -0.1]^T$, $\{x^{(k)}\}$ is generated by

$$x_1^{(k)} = \frac{1}{3} \cos x_2^{(k-1)} x_3^{(k-1)} + \frac{1}{6},$$

$$x_2^{(k)} = \frac{1}{9} \sqrt{\left(x_1^{(k-1)}\right)^2 + \sin x_3^{(k-1)} + 1.06} - 0.1,$$

$$x_3^{(k)} = -\frac{1}{20} e^{-x_1^{(k-1)} x_2^{(k-1)}} - \frac{10\pi - 3}{60}.$$

- Result:

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$\ x^{(k)} - x^{(k-1)}\ _\infty$
0	0.10000000	0.10000000	-0.10000000	
1	0.49998333	0.00944115	-0.52310127	0.423
2	0.49999593	0.00002557	-0.52336331	9.4×10^{-3}
3	0.50000000	0.00001234	-0.52359814	2.3×10^{-4}
4	0.50000000	0.00000003	-0.52359847	1.2×10^{-5}
5	0.50000000	0.00000002	-0.52359877	3.1×10^{-7}

- Approximated solution (cont.):
 - Accelerate convergence of the fixed-point iteration:

$$x_1^{(k)} = \frac{1}{3} \cos x_2^{(k-1)} x_3^{(k-1)} + \frac{1}{6},$$

$$x_2^{(k)} = \frac{1}{9} \sqrt{\left(x_1^{(k)}\right)^2 + \sin x_3^{(k-1)} + 1.06} - 0.1,$$

$$x_3^{(k)} = -\frac{1}{20} e^{-x_1^{(k)} x_2^{(k)}} - \frac{10\pi - 3}{60},$$

as in the Gauss-Seidel method for linear systems.

- Result:

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$\ x^{(k)} - x^{(k-1)}\ _\infty$
0	0.10000000	0.10000000	-0.10000000	
1	0.49998333	0.02222979	-0.52304613	0.423
2	0.49997747	0.00002815	-0.52359807	2.2×10^{-2}
3	0.50000000	0.00000004	-0.52359877	2.8×10^{-5}
4	0.50000000	0.00000000	-0.52359877	3.8×10^{-8}

Newton's method

First consider solving the following system of nonlinear eqs.:

$$\begin{cases} f_1(x_1, x_2) = 0, \\ f_2(x_1, x_2) = 0. \end{cases}$$

Suppose $(x_1^{(k)}, x_2^{(k)})$ is an approximation to the solution of the system above, and we try to compute $h_1^{(k)}$ and $h_2^{(k)}$ such that $(x_1^{(k)} + h_1^{(k)}, x_2^{(k)} + h_2^{(k)})$ satisfies the system. By the Taylor's theorem for two variables,

$$\begin{aligned} 0 &= f_1(x_1^{(k)} + h_1^{(k)}, x_2^{(k)} + h_2^{(k)}) \\ &\approx f_1(x_1^{(k)}, x_2^{(k)}) + h_1^{(k)} \frac{\partial f_1}{\partial x_1}(x_1^{(k)}, x_2^{(k)}) + h_2^{(k)} \frac{\partial f_1}{\partial x_2}(x_1^{(k)}, x_2^{(k)}) \end{aligned}$$

$$\begin{aligned} 0 &= f_2(x_1^{(k)} + h_1^{(k)}, x_2^{(k)} + h_2^{(k)}) \\ &\approx f_2(x_1^{(k)}, x_2^{(k)}) + h_1^{(k)} \frac{\partial f_2}{\partial x_1}(x_1^{(k)}, x_2^{(k)}) + h_2^{(k)} \frac{\partial f_2}{\partial x_2}(x_1^{(k)}, x_2^{(k)}) \end{aligned}$$



Put this in matrix form

$$\begin{bmatrix} \frac{\partial f_1}{\partial x_1}(x_1^{(k)}, x_2^{(k)}) & \frac{\partial f_1}{\partial x_2}(x_1^{(k)}, x_2^{(k)}) \\ \frac{\partial f_2}{\partial x_1}(x_1^{(k)}, x_2^{(k)}) & \frac{\partial f_2}{\partial x_2}(x_1^{(k)}, x_2^{(k)}) \end{bmatrix} \begin{bmatrix} h_1^{(k)} \\ h_2^{(k)} \end{bmatrix} + \begin{bmatrix} f_1(x_1^{(k)}, x_2^{(k)}) \\ f_2(x_1^{(k)}, x_2^{(k)}) \end{bmatrix} \approx \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

The matrix

$$J(x_1^{(k)}, x_2^{(k)}) \equiv \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(x_1^{(k)}, x_2^{(k)}) & \frac{\partial f_1}{\partial x_2}(x_1^{(k)}, x_2^{(k)}) \\ \frac{\partial f_2}{\partial x_1}(x_1^{(k)}, x_2^{(k)}) & \frac{\partial f_2}{\partial x_2}(x_1^{(k)}, x_2^{(k)}) \end{bmatrix}$$

is called the **Jacobian matrix**. Set $h_1^{(k)}$ and $h_2^{(k)}$ be the solution of the linear system

$$J(x_1^{(k)}, x_2^{(k)}) \begin{bmatrix} h_1^{(k)} \\ h_2^{(k)} \end{bmatrix} = - \begin{bmatrix} f_1(x_1^{(k)}, x_2^{(k)}) \\ f_2(x_1^{(k)}, x_2^{(k)}) \end{bmatrix},$$

then

$$\begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \end{bmatrix} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \end{bmatrix} + \begin{bmatrix} h_1^{(k)} \\ h_2^{(k)} \end{bmatrix}$$

is expected to be a better approximation.



In general, we solve the system of n nonlinear equations $f_i(x_1, \dots, x_n) = 0, i = 1, \dots, n$. Let

$$x = [x_1 \quad x_2 \quad \cdots \quad x_n]^T$$

and

$$F(x) = [f_1(x) \quad f_2(x) \quad \cdots \quad f_n(x)]^T.$$

The problem can be formulated as solving

$$F(x) = 0, \quad F : \mathbb{R}^n \rightarrow \mathbb{R}^n.$$

Let $J(x)$, where the (i, j) entry is $\frac{\partial f_i}{\partial x_j}(x)$, be the $n \times n$ **Jacobian matrix**. Then the Newton's iteration is defined as

$$x^{(k+1)} = x^{(k)} + h^{(k)},$$

where $h^{(k)} \in \mathbb{R}^n$ is the solution of the linear system

$$J(x^{(k)})h^{(k)} = -F(x^{(k)}).$$



Algorithm 1 (Newton's Method for Systems)

Given a function $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$, an initial guess $x^{(0)}$ to the zero of F , and stop criteria M , δ , and ε , this algorithm performs the Newton's iteration to approximate one root of F .

Set $k = 0$ and $h^{(-1)} = e_1$.

While ($k < M$) and ($\|h^{(k-1)}\| \geq \delta$) and ($\|F(x^{(k)})\| \geq \varepsilon$)

 Calculate $J(x^{(k)}) = [\partial F_i(x^{(k)})/\partial x_j]$.

 Solve the $n \times n$ linear system $J(x^{(k)})h^{(k)} = -F(x^{(k)})$.

 Set $x^{(k+1)} = x^{(k)} + h^{(k)}$ and $k = k + 1$.

End while

Output ("Convergent $x^{(k)}$ ") or

 ("Maximum number of iterations exceeded")



Theorem 5

Let x^* be a solution of $G(x) = x$. Suppose $\exists \delta > 0$ with

- (i) $\partial g_i / \partial x_j$ is continuous on $N_\delta = \{x; \|x - x^*\| < \delta\}$ for all i and j .
- (ii) $\partial^2 g_i(x) / (\partial x_j \partial x_k)$ is continuous and

$$\left| \frac{\partial^2 g_i(x)}{\partial x_j \partial x_k} \right| \leq M$$

for some M whenever $x \in N_\delta$ for each i, j and k .

- (iii) $\partial g_i(x^*) / \partial x_k = 0$ for each i and k .

Then $\exists \hat{\delta} < \delta$ such that the sequence $\{x^{(k)}\}$ generated by

$$x^{(k)} = G(x^{(k-1)})$$

converges **quadratically** to x^* for any $x^{(0)}$ satisfying $\|x^{(0)} - x^*\|_\infty < \hat{\delta}$.
Moreover,

$$\|x^{(k)} - x^*\|_\infty \leq \frac{n^2 M}{2} \|x^{(k-1)} - x^*\|_\infty^2, \forall k \geq 1.$$



Example 6

Consider the nonlinear system

$$\begin{aligned}3x_1 - \cos(x_2x_3) - \frac{1}{2} &= 0, \\x_1^2 - 81(x_2 + 0.1)^2 + \sin x_3 + 1.06 &= 0, \\e^{-x_1x_2} + 20x_3 + \frac{10\pi - 3}{3} &= 0.\end{aligned}$$

- Nonlinear functions: Let

$$F(x_1, x_2, x_3) = [f_1(x_1, x_2, x_3), f_2(x_1, x_2, x_3), f_3(x_1, x_2, x_3)]^T,$$

where

$$\begin{aligned}f_1(x_1, x_2, x_3) &= 3x_1 - \cos(x_2x_3) - \frac{1}{2}, \\f_2(x_1, x_2, x_3) &= x_1^2 - 81(x_2 + 0.1)^2 + \sin x_3 + 1.06, \\f_3(x_1, x_2, x_3) &= e^{-x_1x_2} + 20x_3 + \frac{10\pi - 3}{3}.\end{aligned}$$



- Nonlinear functions (cont.):

The Jacobian matrix $J(x)$ for this system is

$$J(x_1, x_2, x_3) = \begin{bmatrix} 3 & x_3 \sin x_2 x_3 & x_2 \sin x_2 x_3 \\ 2x_1 & -162(x_2 + 0.1) & \cos x_3 \\ -x_2 e^{-x_1 x_2} & -x_1 e^{-x_1 x_2} & 20 \end{bmatrix}.$$

- Newton's iteration with initial $x^{(0)} = [0.1, 0.1, -0.1]^T$:

$$\begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ x_3^{(k)} \end{bmatrix} = \begin{bmatrix} x_1^{(k-1)} \\ x_2^{(k-1)} \\ x_3^{(k-1)} \end{bmatrix} - \begin{bmatrix} h_1^{(k-1)} \\ h_2^{(k-1)} \\ h_3^{(k-1)} \end{bmatrix},$$

where

$$\begin{bmatrix} h_1^{(k-1)} \\ h_2^{(k-1)} \\ h_3^{(k-1)} \end{bmatrix} = \left(J(x_1^{(k-1)}, x_2^{(k-1)}, x_3^{(k-1)}) \right)^{-1} F(x_1^{(k-1)}, x_2^{(k-1)}, x_3^{(k-1)}).$$



- Result:

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$\ x^{(k)} - x^{(k-1)}\ _\infty$
0	0.10000000	0.10000000	-0.10000000	
1	0.50003702	0.01946686	-0.52152047	0.422
2	0.50004593	0.00158859	-0.52355711	1.79×10^{-2}
3	0.50000034	0.00001244	-0.52359845	1.58×10^{-3}
4	0.50000000	0.00000000	-0.52359877	1.24×10^{-5}
5	0.50000000	0.00000000	-0.52359877	0



Quasi-Newton methods

- Newton's Methods

- Advantage: **quadratic** convergence
- Disadvantage: For each iteration, it requires $O(n^3) + O(n^2) + O(n)$ arithmetic operations:
 - n^2 partial derivatives for Jacobian matrix – in most situations, the exact evaluation of the partial derivatives is inconvenient.
 - n scalar functional evaluations of F
 - $O(n^3)$ arithmetic operations to solve linear system.

- quasi-Newton methods

- Advantage: it requires only n scalar functional evaluations per iteration and $O(n^2)$ arithmetic operations
- Disadvantage: **superlinear** convergence

Recall that in one dimensional case, one uses the **linear** model

$$\ell_k(x) = f(x_k) + a_k(x - x_k)$$

to **approximate** the function $f(x)$ at x_k . That is, $\ell_k(x_k) = f(x_k)$ for any $a_k \in \mathbb{R}$. If we further require that $\ell'(x_k) = f'(x_k)$, then



The zero of $\ell_k(x)$ is used to give a new approximate for the zero of $f(x)$, that is,

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

which yields **Newton's** method.

If $f'(x_k)$ is **not available**, one instead asks the linear model to satisfy

$$\ell_k(x_k) = f(x_k) \quad \text{and} \quad \ell_k(x_{k-1}) = f(x_{k-1}).$$

In doing this, the identity

$$f(x_{k-1}) = \ell_k(x_{k-1}) = f(x_k) + a_k(x_{k-1} - x_k)$$

gives

$$a_k = \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}.$$

Solving $\ell_k(x) = 0$ yields the **secant** iteration

$$x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} f(x_k).$$



In multiple dimension, the analogue **affine model** becomes

$$M_k(x) = F(x_k) + A_k(x - x_k),$$

where $x, x_k \in \mathbb{R}^n$ and $A_k \in \mathbb{R}^{n \times n}$, and satisfies

$$M_k(x_k) = F(x_k),$$

for any A_k . The zero of $M_k(x)$ is then used to give a new approximate for the zero of $F(x)$, that is,

$$x_{k+1} = x_k - A_k^{-1} F(x_k).$$

The **Newton's** method chooses

$$A_k = F'(x_k) \equiv J(x_k) = \text{the Jacobian matrix}$$

and yields the iteration

$$x_{k+1} = x_k - (F'(x_k))^{-1} F(x_k).$$



When the Jacobian matrix $J(x_k) \equiv F'(x_k)$ is not available, one can require

$$M_k(x_{k-1}) = F(x_{k-1}).$$

Then

$$F(x_{k-1}) = M_k(x_{k-1}) = F(x_k) + A_k(x_{k-1} - x_k),$$

which gives

$$A_k(x_k - x_{k-1}) = F(x_k) - F(x_{k-1})$$

and this is the so-called secant equation. Let

$$h_k = x_k - x_{k-1} \quad \text{and} \quad y_k = F(x_k) - F(x_{k-1}).$$

The secant equation becomes

$$A_k h_k = y_k.$$



However, this secant equation can not uniquely determine A_k . One way of choosing A_k is to minimize $M_k - M_{k-1}$ subject to the secant equation. Note

$$\begin{aligned}
 M_k(x) - M_{k-1}(x) &= F(x_k) + A_k(x - x_k) - F(x_{k-1}) - A_{k-1}(x - x_{k-1}) \\
 &= (F(x_k) - F(x_{k-1})) + A_k(x - x_k) - A_{k-1}(x - x_{k-1}) \\
 &= A_k(x_k - x_{k-1}) + A_k(x - x_k) - A_{k-1}(x - x_{k-1}) \\
 &= A_k(x - x_{k-1}) - A_{k-1}(x - x_{k-1}) \\
 &= (A_k - A_{k-1})(x - x_{k-1}).
 \end{aligned}$$

For any $x \in \mathbb{R}^n$, we express

$$x - x_{k-1} = \alpha h_k + t_k,$$

for some $\alpha \in \mathbb{R}$, $t_k \in \mathbb{R}^n$, and $h_k^T t_k = 0$. Then

$$M_k - M_{k-1} = (A_k - A_{k-1})(\alpha h_k + t_k) = \alpha(A_k - A_{k-1})h_k + (A_k - A_{k-1})t_k.$$



Since

$$(A_k - A_{k-1})h_k = A_k h_k - A_{k-1} h_k = y_k - A_{k-1} h_k,$$

both y_k and $A_{k-1} h_k$ are old values, we have no control over the first part $(A_k - A_{k-1})h_k$. In order to minimize $M_k(x) - M_{k-1}(x)$, we try to choose A_k so that

$$(A_k - A_{k-1})t_k = 0$$

for all $t_k \in \mathbb{R}^n$, $h_k^T t_k = 0$. This requires that $A_k - A_{k-1}$ to be a rank-one matrix of the form

$$A_k - A_{k-1} = u_k h_k^T$$

for some $u_k \in \mathbb{R}^n$. Then

$$u_k h_k^T h_k = (A_k - A_{k-1})h_k = y_k - A_{k-1} h_k$$



which gives

$$u_k = \frac{y_k - A_{k-1}h_k}{h_k^T h_k}.$$

Therefore,

$$A_k = A_{k-1} + \frac{(y_k - A_{k-1}h_k)h_k^T}{h_k^T h_k}. \quad (1)$$

After A_k is determined, the new iterate x_{k+1} is derived from solving $M_k(x) = 0$. It can be done by first noting that

$$h_{k+1} = x_{k+1} - x_k \quad \implies \quad x_{k+1} = x_k + h_{k+1}$$

and

$$M_k(x_{k+1}) = 0 \implies F(x_k) + A_k(x_{k+1} - x_k) = 0 \implies A_k h_{k+1} = -F(x_k)$$

These formulations give the **Broyden's** method.



Algorithm 2 (Broyden's Method)

Given a n -variable nonlinear function $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$, an initial iterate x_0 and initial Jacobian matrix $A_0 \in \mathbb{R}^{n \times n}$ (e.g., $A_0 = I$), this algorithm finds the solution for $F(x) = 0$.

Given x_0 , tolerance TOL , maximum number of iteration M .

Set $k = 1$.

While $k \leq M$ and $\|x_k - x_{k-1}\|_2 \geq TOL$

Solve $A_k h_{k+1} = -F(x_k)$ for h_{k+1}

Update $x_{k+1} = x_k + h_{k+1}$

Compute $y_{k+1} = F(x_{k+1}) - F(x_k)$

Update

$$A_{k+1} = A_k + \frac{(y_{k+1} - A_k h_{k+1})h_{k+1}^T}{h_{k+1}^T h_{k+1}} = A_k + \frac{(y_{k+1} + F(x_k))h_{k+1}^T}{h_{k+1}^T h_{k+1}}$$

$k = k + 1$

End While



Solve the linear system $A_k h_{k+1} = -F(x_k)$ for h_{k+1} :

- LU -factorization: cost $\frac{2}{3}n^3 + O(n^2)$ floating-point operations.
- Applying the Sherman-Morrison-Woodbury formula

$$(B + UV^T)^{-1} = B^{-1} - B^{-1}U(I + V^T B^{-1}U)^{-1}V^T B^{-1}$$

to (1), we have

$$\begin{aligned} & A_k^{-1} \\ &= \left[A_{k-1} + \frac{(y_k - A_{k-1}h_k)h_k^T}{h_k^T h_k} \right]^{-1} \\ &= A_{k-1}^{-1} - A_{k-1}^{-1} \frac{y_k - A_{k-1}h_k}{h_k^T h_k} \left(1 + h_k^T A_{k-1}^{-1} \frac{y_k - A_{k-1}h_k}{h_k^T h_k} \right)^{-1} h_k^T \\ &= A_{k-1}^{-1} + \frac{(h_k - A_{k-1}^{-1}y_k)h_k^T A_{k-1}^{-1}}{h_k^T A_{k-1}^{-1}y_k}. \end{aligned}$$



- Newton-based methods
 - Advantage: high speed of convergence once a sufficiently accurate approximation
 - Weakness: an accurate initial approximation to the solution is needed to ensure convergence.
- Steepest Descent method converges only linearly to the sol., but it will usually converge even for poor initial approximations.
- “Find sufficiently accurate starting approximate solution by using Steepest Descent method” + “Compute convergent solution by using Newton-based methods”
- The method of Steepest Descent determines a local minimum for a multivariable function of $g : \mathbb{R}^n \rightarrow \mathbb{R}$.
- A system of the form $f_i(x_1, \dots, x_n) = 0$, $i = 1, 2, \dots, n$ has a solution at x iff the function g defined by

$$g(x_1, \dots, x_n) = \sum_{i=1}^n [f_i(x_1, \dots, x_n)]^2$$

has the minimal value zero



Basic idea of steepest descent method:

- (i) Evaluate g at an initial approximation $x^{(0)}$;
- (ii) Determine a direction from $x^{(0)}$ that results in a decrease in the value of g ;
- (iii) Move an appropriate distance in this direction and call the new vector $x^{(1)}$;
- (iv) Repeat steps (i) through (iii) with $x^{(0)}$ replaced by $x^{(1)}$.

Definition 7 (Gradient)

If $g : \mathbb{R}^n \rightarrow \mathbb{R}$, the gradient, $\nabla g(x)$, at x is defined by

$$\nabla g(x) = \left(\frac{\partial g}{\partial x_1}(x), \dots, \frac{\partial g}{\partial x_n}(x) \right).$$

Definition 8 (Directional Derivative)

The directional derivative of g at x in the direction of v with $\|v\|_2 = 1$ is defined by

$$D_v g(x) = \lim_{h \rightarrow 0} \frac{g(x + hv) - g(x)}{h} = v^T \nabla g(x).$$



Theorem 9

The direction of the greatest decrease in the value of g at x is the direction given by $-\nabla g(x)$.

- Object: reduce $g(x)$ to its minimal value zero.
 \Rightarrow for an initial approximation $x^{(0)}$, an appropriate choice for new vector $x^{(1)}$ is

$$x^{(1)} = x^{(0)} - \alpha \nabla g(x^{(0)}), \quad \text{for some constant } \alpha > 0.$$

- Choose $\alpha > 0$ such that $g(x^{(1)}) < g(x^{(0)})$: define

$$h(\alpha) = g(x^{(0)} - \alpha \nabla g(x^{(0)})),$$

then find α^* such that

$$h(\alpha^*) = \min_{\alpha} h(\alpha).$$



- How to find α^* ?

- Solve a root-finding problem $h'(\alpha) = 0 \Rightarrow$ Too costly, in general.
- Choose three number $\alpha_1 < \alpha_2 < \alpha_3$, construct quadratic polynomial $P(x)$ that interpolates h at α_1, α_2 and α_3 , i.e.,

$$P(\alpha_1) = h(\alpha_1), P(\alpha_2) = h(\alpha_2), P(\alpha_3) = h(\alpha_3),$$

to approximate h . Use the minimum value $P(\hat{\alpha})$ in $[\alpha_1, \alpha_3]$ to approximate $h(\alpha^*)$. The new iteration is

$$x^{(1)} = x^{(0)} - \hat{\alpha} \nabla g(x^{(0)}).$$

- Set $\alpha_1 = 0$ to minimize the computation
- α_3 is found with $h(\alpha_3) < h(\alpha_1)$.
- Choose $\alpha_2 = \alpha_3/2$.



Example 10

Use the Steepest Descent method with $x^{(0)} = (0, 0, 0)^T$ to find a reasonable starting approximation to the solution of the nonlinear system

$$f_1(x_1, x_2, x_3) = 3x_1 - \cos(x_2x_3) - \frac{1}{2} = 0,$$

$$f_2(x_1, x_2, x_3) = x_1^2 - 81(x_2 + 0.1)^2 + \sin x_3 + 1.06 = 0,$$

$$f_3(x_1, x_2, x_3) = e^{-x_1x_2} + 20x_3 + \frac{10\pi - 3}{3} = 0.$$

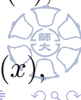
Let $g(x_1, x_2, x_3) =$

$[f_1(x_1, x_2, x_3)]^2 + [f_2(x_1, x_2, x_3)]^2 + [f_3(x_1, x_2, x_3)]^2$. Then

$$\nabla g(x_1, x_2, x_3) \equiv \nabla g(x)$$

$$= \left(2f_1(x) \frac{\partial f_1}{\partial x_1}(x) + 2f_2(x) \frac{\partial f_2}{\partial x_1}(x) + 2f_3(x) \frac{\partial f_3}{\partial x_1}(x), \right.$$

$$\left. 2f_1(x) \frac{\partial f_1}{\partial x_2}(x) + 2f_2(x) \frac{\partial f_2}{\partial x_2}(x) + 2f_3(x) \frac{\partial f_3}{\partial x_2}(x), \right.$$



For $x^{(0)} = [0, 0, 0]^T$, we have

$$g(x^{(0)}) = 111.975 \quad \text{and} \quad z_0 = \|\nabla g(x^{(0)})\|_2 = 419.554.$$

Let

$$z = \frac{1}{z_0} \nabla g(x^{(0)}) = [-0.0214514, -0.0193062, 0.999583]^T.$$

With $\alpha_1 = 0$, we have

$$g_1 = g(x^{(0)} - \alpha_1 z) = g(x^{(0)}) = 111.975.$$

Let $\alpha_3 = 1$ so that

$$g_3 = g(x^{(0)} - \alpha_3 z) = 93.5649 < g_1.$$

Set $\alpha_2 = \alpha_3/2 = 0.5$. Thus

$$g_2 = g(x^{(0)} - \alpha_2 z) = 2.53557.$$



Form quadratic polynomial $P(\alpha)$ defined as

$$P(\alpha) = g_1 + h_1\alpha + h_3\alpha(\alpha - \alpha_2)$$

that interpolates $g(x^{(0)} - \alpha z)$ at $\alpha_1 = 0, \alpha_2 = 0.5$ and $\alpha_3 = 1$ as follows

$$g_2 = P(\alpha_2) = g_1 + h_1\alpha_2 \Rightarrow h_1 = \frac{g_2 - g_1}{\alpha_2} = -218.878,$$

$$g_3 = P(\alpha_3) = g_1 + h_1\alpha_3 + h_3\alpha_3(\alpha_3 - \alpha_2) \Rightarrow h_3 = 400.937.$$

Thus

$$P(\alpha) = 111.975 - 218.878\alpha + 400.937\alpha(\alpha - 0.5)$$

so that

$$0 = P'(\alpha_0) = -419.346 + 801.872\alpha_0 \Rightarrow \alpha_0 = 0.522959$$

Since

$$g_0 = g(x^{(0)} - \alpha_0 z) = 2.32762 < \min\{g_1, g_3\},$$

we set

$$x^{(1)} = x^{(0)} - \alpha_0 z = [0.0112182, 0.0100964, -0.522741]^T.$$

