

Set 13: Line Search and Quasi-Newton for General Nonlinear Systems

Kyle A. Gallivan

Department of Mathematics

Florida State University

Foundations of Computational Math 1

Fall 2010

Other Methods for Nonlinear Systems

- General line search and Quasi-Newton methods have been discussed for the nonlinear systems that arise in unconstrained optimization.
- We have only considered the line search method (Inexact) Newton for general nonlinear systems but did not discuss stepsizes other than $\alpha_k = 1$.
- Both general line search methods and in particular Quasi-Newton methods can be used for general nonlinear systems.

Other Methods for Nonlinear Systems

- For general nonlinear systems, we do not have the structure of the optimization problem, e.g., symmetry, nor do we have a cost function defining the problem that can be used to set the stepsize.
- We consider line search methods for general nonlinear systems including:
 - a secant method (quasi-Newton) called Broyden's method,
 - stepsize selection,
 - steepest descent.

Broyden's Method

- Quasi-Newton methods (secant methods, modification methods) attempt to approximate the action of the Jacobian not the Jacobian itself.
- the approximation is assumed inexpensive to produce on each step
- generalization of the idea of the secant method
- good example is Broyden's method
- like Newton it is often run with a stepsize added, i.e.,
$$x^{(k+1)} = x^{(k)} + \alpha_k p_k$$

Secant Condition for Scalar Equations

secant method for a scalar equation:

$$\ell_k(x) = f_k + q_k(x - x_k) \rightarrow q_k(x_{k+1} - x_k) = q_k s_k = -f_k$$

uses slope of line connecting (x_{k-1}, f_{k-1}) and (x_k, f_k)

$$q_k = \frac{f_k - f_{k-1}}{x_k - x_{k-1}} \rightarrow q_k s_{k-1} = y_{k-1} \quad \text{1-D secant condition}$$

note that $q_k s_k \neq y_k$

$$\begin{aligned} \ell_{k+1}(x) &= f_{k+1} + q_{k+1}(x - x_{k+1}) \rightarrow q_{k+1}(x_{k+2} - x_{k+1}) \\ &= q_{k+1} s_{k+1} = -f_{k+1} \end{aligned}$$

$$q_{k+1} = \frac{f_{k+1} - f_k}{x_{k+1} - x_k} \rightarrow q_{k+1} s_k = y_k \quad \text{1-D secant condition}$$

note that $q_{k+1} s_{k+1} \neq y_{k+1}$

Secant Condition for Systems

- local model for systems $M_k(x) = F(x^{(k)}) + B_k(x - x^{(k)})$
- $x^{(k+1)} = x^{(k)} + \alpha_k p_k$ where $B_k p_k = -F(x^{(k)})$ gives the change to get to the root of $M_k(x)$ and α_k is a stepsize.
- Let $s_k = x^{(k+1)} - x^{(k)}$ and $y_k = F(x^{(k+1)}) - F(x^{(k)})$
- As with $q_k s_{k-1} = y_{k-1}$ for scalars we want $B_k s_{k-1} = y_{k-1}$.
- We know $B_k s_k \neq y_k$ since we can take $\alpha_k = 1$ and get

$$B_k s_k = B_k p_k = y_k = F(x^{(k+1)}) - F(x^{(k)}) = F(x^{(k+1)}) - B_k p_k$$
$$\therefore F(x^{(k+1)}) = 0$$

- On the next step we want $B_{k+1} s_k = y_k$ etc.

Secant Condition for Systems

Secant condition:

$$B_{k+1} s_k = y_k$$

- Note that this is underdetermined with respect to the n^2 degrees of freedom in B_{k+1} .
- Many possible choices of B_{k+1} at each step.
- Suppose we look for a modification to B_k that makes B_{k+1} satisfy the secant condition, i.e., $B_{k+1} = B_k + E$

Secant Condition for Systems

- Broyden's method chooses:

$$B_{k+1} = \operatorname{argmin}_{Bs_k=y_k} \|B - B_k\|_2$$

$$B_{k+1} = B_k + \frac{1}{\|s_k\|_2^2} (y_k - B_k s_k) s_k^T$$

- essentially a backward error, look for a perturbed matrix that satisfies an equation.

Broyden's Method

Broyden's Method:

Choose $B_0, x^{(0)}$

loop over k until convergence

Solve $B_k p_k = -F(x^{(k)})$

Choose α_k

$$x^{(k+1)} = x^{(k)} + \alpha_k p_k$$

$$s_k = x^{(k+1)} - x^{(k)}$$

$$y_k = F(x^{(k+1)}) - F(x^{(k)})$$

$$B_{k+1} = B_k + \frac{1}{\|s_k\|_2^2} (y_k - B_k s_k) s_k^T$$

end

Broyden's Method

$$F(x) = \begin{pmatrix} (\xi_1 + 3)(\xi_2^3 - 7) + 18 \\ \sin(\xi_2 e^{\xi_1} - 1) \end{pmatrix} \rightarrow x^* = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad \alpha_k = 1$$

k	$\xi_1^{(k)}$	$\xi_2^{(k)}$	$\ f(x^{(k)})\ _2$
0	-0.500000000000000	1.400000000000000	7.3615341974672
1	-0.0553151357177	1.0280665838357	0.5874890107585
2	0.0005099530701	1.0001236434919	0.0020471789647
3	-0.0002338478636	1.0000765608693	0.0020980374394
4	-0.0000408262210	1.0000135978908	0.0003683460658
5	-0.0000001327509	1.0000000453384	0.0000012077182
6	-0.0000000005391	1.0000000001807	0.0000000048739

Broyden's Method

- If $x^{(0)}$ is far from the root the behavior of Broyden like Newton can be erratic. (Damped Newton, i.e., add scale α_k)
- nonsingularity not guaranteed, often a term μI added.
- local convergence, if $\|x^{(0)} - x^*\| \leq \delta$ and $\|B_0 - J_F(x^*)\| \leq \epsilon$ then superlinear convergence to x^* .
- second requirement is difficult to guarantee in practice and is often important to convergence.
- $B_0 = J_F(x^{(0)})$ is often used.

Optimization Methods for Nonlinear Equations

- line search ideas are used extensively in unconstrained optimization
- optimization methods for solving nonlinear equations use some merit function $g(x)$ to find α_k and $x^{(k+1)}$ as the basis for the method
- a search is typically used to try and “minimize” $g(x)$ along the direction p_k .
- All have the flavor of the damped Newton and damped Broyden
- many others are possible, e.g., steepest descent, nonlinear CG etc.

Damped Newton and Broyden Methods

- Newton, Inexact Newton, and Broyden all have local and sometimes rapid convergence
- introducing a stepsize $x^{(k+1)} = x^{(k)} + \alpha_k p_k$ can give global convergence
- α_k chosen via a line search

Stepsize

- given p_k choose α_k
- multidimensional linear root has already been used to get p_k
- need a merit function $g(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ and we make it a function of α by taking $x = x^{(k)} + \alpha p_k$.
- most used

$$g(x) = \frac{1}{2} \|F(x)\|_2^2$$
$$\nabla g(x) = J_F(x)^T F(x)$$

Stepsize

- problem: there are often \tilde{x} where $g(\tilde{x})$ is minimal, i.e., $\nabla g(\tilde{x}) = J_F(\tilde{x})^T F(\tilde{x}) = 0$ but $F(\tilde{x}) \neq 0$.
- We do not want to choose α_k in such a way so as to move to those points.
- Such \tilde{x} must have $J_F(\tilde{x})^T$ singular.
- So use of line search to set α_k for solving nonlinear systems must avoid singular Jacobians or avoid such \tilde{x} .
- typically some sort of backtracking is used with initial guess at α_k biased toward 1 for large k .
- Remember: $g(x)$ is a convenient function to choose α_k **we are not solving a minimization problem with our solution x^* where $F(x^*) = 0$.**

Steepest Descent for Solving Nonlinear Systems

We have

$$g(x) = \frac{1}{2} \|F(x)\|_2^2$$

$$\nabla g(x) = J_F(x)^T F(x)$$

- $p_k = -\nabla g(x^{(k)}) / \|\nabla g(x^{(k)})\|$ and $x^{(k+1)} = x^{(k)} + \alpha_k p_k$.
- We do not have an analytical minimizer for α_k . Some sort of line search must be used.
- A simple strategy samples $g(\alpha p_k)$ at three different α values, interpolates with a quadratic polynomial and takes α_k to be the value that minimizes the quadratic.