

## 8. SOLUTION OF NONLINEAR SYSTEMS OF EQUATIONS

We next wish to generalize some of the approaches used to derive methods to approximate solutions of a single nonlinear equation to approximate solutions of a system of nonlinear equations. We consider the problem  $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ , where  $\mathbf{x} = (x_1, \dots, x_N)$  and  $\mathbf{F} = (F_1, \dots, F_N)$ , i.e., we consider the system

$$F_1(x_1, \dots, x_N) = 0, \dots, F_N(x_1, \dots, x_N) = 0.$$

**8.1. Newton's method for nonlinear systems.** Many of the methods for this problem are variations of Newton's method, so we consider this first. For the single equation  $f(x) = 0$ , Newton's method is the iteration scheme:

$$x_{n+1} = x_n - f(x_n)/f'(x_n).$$

To generalize this method, we recall that one derivation of Newton's method was to view  $x_{n+1}$  as the approximation obtained by truncating the Taylor series expansion

$$0 = f(x^*) = f(x_n) + f'(x_n)(x^* - x_n) + O(x^* - x_n)^2,$$

i.e., we define  $x_{n+1}$  as the approximation to  $x^*$  which gives equality when we drop the last term. So  $x_{n+1}$  is the solution of

$$f'(x_n)(x_{n+1} - x_n) = -f(x_n).$$

One way to write this is to define  $x_{n+1} = x_n + s_n$ , where  $s_n$  solves  $f'(x_n)s_n = -f(x_n)$ . This derivation generalizes to systems of nonlinear equations. Applying Taylor series to each  $F_k$ , we get

$$0 = F_k(\mathbf{x}^*) = F_k(\mathbf{x}^n) + \sum_{i=1}^N \frac{\partial F_k}{\partial x_i}(\mathbf{x}^n)(x_i^* - x_i^n) + O\left(\sum_{i=1}^N [(x_i^* - x_i^n)^2]\right), \quad k = 1, \dots, N.$$

To avoid confusion, the iteration number is now a superscript and the components of the vectors are subscripts. As before, we define  $\mathbf{x}^{n+1}$  as the approximation to  $\mathbf{x}^*$  which gives equality when we drop the last term. Let  $J(\mathbf{x}^n)$  denote the Jacobian matrix

$$\begin{pmatrix} \partial F_1/\partial x_1 & \cdots & \partial F_1/\partial x_N \\ \cdots & \cdots & \cdots \\ \partial F_N/\partial x_1 & \cdots & \partial F_N/\partial x_N \end{pmatrix}.$$

Then  $\mathbf{x}^{n+1}$  satisfies

$$J(\mathbf{x}^n)(\mathbf{x}^{n+1} - \mathbf{x}^n) = -\mathbf{F}(\mathbf{x}^n).$$

Hence, as before we can write  $\mathbf{x}^{n+1} = \mathbf{x}^n + \mathbf{s}^n$ , where  $\mathbf{s}^n$  is the solution of the linear system

$$J(\mathbf{x}^n)\mathbf{s}^n = -\mathbf{F}(\mathbf{x}^n).$$

Note that we can also write Newton's method in the form

$$\mathbf{x}^{n+1} = \mathbf{x}^n - [J(\mathbf{x}^n)]^{-1}\mathbf{F}(\mathbf{x}^n).$$

However, it is more efficient to solve the linear system, then to compute the inverse of  $J$  and multiply it times the vector  $\mathbf{F}$ .

A typical local convergence theorem for Newton's method is the following.

**Theorem 26.** Suppose  $\mathbf{F}$  satisfies (i)  $F$  is continuously differentiable in an open convex set  $D$  and (ii) there is an  $\mathbf{x}^*$  in  $D$  such that  $\mathbf{F}(\mathbf{x}^*) = 0$  and  $J(\mathbf{x}^*)$  is nonsingular. Then there is an open set  $S$  that contains  $\mathbf{x}^*$  such that for any  $\mathbf{x}_0 \in S$ , the Newton iterates are well-defined, remain in  $S$ , and converge to  $\mathbf{x}^*$ . Moreover, there is a sequence  $\{\alpha_k\}$  that converges to zero such that

$$\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_2 \leq \alpha_k \|\mathbf{x}^k - \mathbf{x}^*\|_2, \quad k = 0, 1, \dots$$

(i.e., superlinear convergence). If in addition,  $F$  satisfies

$$(iii) \quad \|J(\mathbf{x}) - J(\mathbf{x}^*)\|_2 \leq \kappa \|\mathbf{x} - \mathbf{x}^*\|_2, \quad \mathbf{x} \in D$$

for some constant  $\kappa > 0$ , then there exists a constant  $\beta$  such that

$$\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_2 \leq \beta \|\mathbf{x}^k - \mathbf{x}^*\|_2^2, \quad k = 0, 1, \dots,$$

i.e., we have quadratic convergence. Note that the last condition is satisfied if  $D$  is sufficiently small and  $F$  is twice differentiable at  $\mathbf{x}^*$ .

Some advantages of Newton's method: (i) There is a domain of attraction, i.e., there is a set  $S$  such that if one of the Newton iterates lands in  $S$ , then all future iterates remain in  $S$  and converge to  $\mathbf{x}^*$ , (ii) we get at least superlinear and sometimes quadratic convergence.

Some disadvantages of Newton's method: (i) It is not clear how to get a starting guess that is sufficiently close to the solution  $\mathbf{x}^*$  to ensure convergence of the method. (ii)  $J(\mathbf{x}^k)$  must be computed at each iteration. This involves the computation of  $N^2$  scalar functions at each step. (iii) For some problems, exact expressions for  $\partial F_i / \partial x_j$  may be difficult to obtain (since the user is only supplying  $F$ ).

We next consider some variations of Newton's method designed to overcome some of these disadvantages. To avoid (iii), we could replace  $\partial F_i / \partial x_j(\mathbf{x})$  by a finite difference approximation such as  $[F_i(\mathbf{x} + h_j \mathbf{e}^j) - F_i(\mathbf{x})] / h_j$ , where  $\mathbf{e}^j$  is a unit vector with a one in the  $j$ th position and zeroes elsewhere and  $(h_1, \dots, h_N)$  is a suitable vector with small components.

If  $\mathbf{F}$  satisfies the hypotheses of the previous theorem and if  $\|\mathbf{h}^k\| \leq \gamma \|\mathbf{F}(\mathbf{x}^k)\|$  for some constant  $\gamma$ , then the conclusions of the previous theorem hold for this finite difference version of Newton's method. Note that  $\|\mathbf{h}^k\|$  is reduced at each iteration as  $\|\mathbf{F}(\mathbf{x}^k)\| \rightarrow 0$ .

To avoid (ii), one can hold the Jacobian fixed for a given number of iterations. Not only does this save function evaluations, but since the same  $LU$  decomposition is used, there is also a savings in the number of operations needed to solve the linear systems. Note, however, that this will decrease the rate of convergence, so there is a trade-off.

**8.2. Broyden's method.** Another approach to avoiding the problem of recomputing the Jacobian at each step is to approximate  $J(\mathbf{x}^k)$  by a matrix  $B_k$  in such a way that  $B_{k+1}$  can be computed from  $B_k$  in  $O(n^2)$  arithmetic operations per iteration by evaluating  $\mathbf{F}$  at only  $\mathbf{x}^k$  and  $\mathbf{x}^{k+1}$ . The price paid for such a savings will be a reduction from quadratic to superlinear convergence. We consider a method due to Broyden (1965).

We first observe that if we have sufficient smoothness, then by Taylor series,

$$\mathbf{F}(\mathbf{x}^k) = \mathbf{F}(\mathbf{x}^{k+1}) + J(\mathbf{x}^{k+1})(\mathbf{x}^k - \mathbf{x}^{k+1}) + O(\|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2).$$

If  $\|\mathbf{x}^k - \mathbf{x}^{k+1}\|$  is small, then it is reasonable to ask that the approximation  $B_{k+1}$  to  $J(\mathbf{x}^{k+1})$  satisfy the quasi-Newton equation:

$$\mathbf{F}(\mathbf{x}^k) = \mathbf{F}(\mathbf{x}^{k+1}) + B_{k+1}(\mathbf{x}^k - \mathbf{x}^{k+1}).$$

If  $N = 1$ , this completely determines  $B_{k+1}$ , i.e.,  $B_{k+1} = [\mathbf{F}(\mathbf{x}^k) - \mathbf{F}(\mathbf{x}^{k+1})]/[\mathbf{x}^k - \mathbf{x}^{k+1}]$ , and we get the secant method. When  $N > 1$ , this equation does not specify how  $B_{k+1}$  acts on vectors orthogonal to  $(\mathbf{x}^k - \mathbf{x}^{k+1})$ , so we need to add additional conditions.

Since we want to compute  $B_{k+1}$  easily from  $B_k$ , one way of uniquely determining  $B_{k+1}$  is to require that

$$B_{k+1}\mathbf{z} = B_k\mathbf{z}, \quad \text{if } (\mathbf{x}^k - \mathbf{x}^{k+1})^T\mathbf{z} = 0.$$

One can easily check that the solution is given by:

$$B_{k+1} = B_k + \frac{[\mathbf{F}(\mathbf{x}^k) - \mathbf{F}(\mathbf{x}^{k+1}) - B_k(\mathbf{x}^k - \mathbf{x}^{k+1})][\mathbf{x}^k - \mathbf{x}^{k+1}]^T}{(\mathbf{x}^k - \mathbf{x}^{k+1})^T(\mathbf{x}^k - \mathbf{x}^{k+1})}.$$

Broyden's method then consists of the iteration:  $\mathbf{x}^{n+1} = \mathbf{x}^n + \mathbf{s}^n$ , where  $\mathbf{s}^n$  is the solution of the linear system

$$B_n\mathbf{s}^n = -\mathbf{F}(\mathbf{x}^n).$$

Setting  $\mathbf{y}^n = \mathbf{F}(\mathbf{x}^{n+1}) - \mathbf{F}(\mathbf{x}^n)$ , we then update  $B_n$  using the formula

$$B_{n+1} = B_n + \frac{[\mathbf{y}^n - B_n\mathbf{s}^n][\mathbf{s}^n]^T}{[\mathbf{s}^n]^T\mathbf{s}^n}.$$

There are also variations of Broyden's method in which one can update  $H_n = B_n^{-1}$  directly (thus avoiding the solution of a linear system at each step), but these do not seem to be as stable. Under the conditions that give local convergence for Newton's method, it is known that Broyden's method is locally and superlinearly convergent.

**8.3. Obtaining a good initial approximation.** An important issue in solving nonlinear systems of equations is obtaining a good enough initial approximation so that the iterative method one is using will converge. One such approach is to try embedding techniques, i.e., we embed the problem of solving  $\mathbf{F}(\mathbf{x}) = \mathbf{0}$  in a family of problems, parametrized by  $t$ . This is done in a way so that the solution  $\mathbf{x}_0$  is easily determined for  $t = 0$ . We then consider a sequence of problems corresponding to the choices  $t = t_0, t_1, \dots$  and use the solution obtained for the problem with  $t = t_n$  as the initial guess for the problem with  $t = t_{n+1}$ . If  $t_{n+1}$  is close to  $t_n$ , we expect the corresponding solutions to be close.

For example, defining  $\mathbf{H}(t, \mathbf{x}) = \mathbf{F}(\mathbf{x}) - (1-t)\mathbf{F}(\mathbf{x}_0)$ ,  $0 \leq t \leq 1$ , we note that the problem  $\mathbf{H}(0, \mathbf{x}) = \mathbf{0}$  has solution  $\mathbf{x} = \mathbf{x}_0$  and that the solution of the problem  $\mathbf{H}(1, \mathbf{x}) = \mathbf{0}$  is the solution of  $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ . Another possible embedding is to define  $\mathbf{H}(t, \mathbf{x}) = \mathbf{F}(\mathbf{x}) - e^{-t}\mathbf{F}(\mathbf{x}_0)$ ,  $0 \leq t < \infty$ . Again,  $\mathbf{H}(0, \mathbf{x}) = \mathbf{0}$  has solution  $\mathbf{x} = \mathbf{x}_0$  and as  $t \rightarrow \infty$ , the limiting solution corresponds to the solution of  $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ .