A GRADUATE SEMINOR REPORT
On
Globally Convergent Modification of Newton's Method

Complied By
Girma Kedir

Advisor
Prof. Dr. rer.nat.habil. R. Demulich.

School of Graduate Studies
Addis Ababa University

2003

PREFACE

This seminor paper has two chapters.

The first chapter deals with the study of the solution of nonlinear problems in one variable which provides a basis for the problems we will consider in the next chapter.

Chapter two is devoted to the study of multivariable nonlinear problems, which is the heart of this seminor paper.

ACKNOWLEDGEMENT

Before anything else I would like to thank the Lord God, without who good will this short but tedious journey amounts to no where.

Next I express my gratitude to my dear mother Assegedech Mitiku for her constant encouragement and financial support during my study as well as the long process of compiling this seminor paper. I would also like to record my sincere appreciation to Mr. Tadesse Bekeshe who gave me reference materials and valuable information which helped me a lot for the entrance examination.

Finally my thanks is due to Prof. Dr.nat.rer.nat.habile. R. Deumlich, my Advisor, for the suggestions he gave me on the first manuscript of this seminor paper.

CONTENT

Page

CHAPTER I

1. Newton's Method for a Nonlinear Equation and Unconstrained Minimization in One Variable.

1.1 Solving a Nonlinear Equation in One Variable.

1.1.1 Newton's Method.

Suppose we wish to calculate the square root of three to a reasonable number of places (digits). Then this can be viewed as finding an approximate root x_* of the function

$f(x) = x^2 - 3,\ x \in R_+^+.$, where $R_+ := \{x \in R \mid x \geq 0\}$.

Now, if our current estimate of the answer is $x_c = 2$, we can get a better estimate x_+ by drawing the line that is tangent to f(x) at $(x_c, f(x_c)) = (2, 1)$, and finding the point x_+ where this line crosses the x-axis (See Figure 1.1.1)

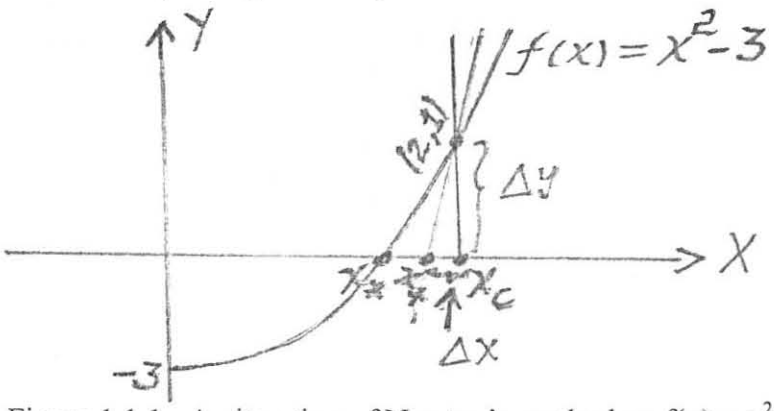


Figure 1.1.1 An iteration of Newton's method on $f(x) = x^2-3$

Since $x_+ = x_c - \Delta x$ and

$$f'(x_c) = \frac{\Delta y}{\Delta x} = \frac{f(x_c)}{\Delta x}$$

we have that

$$f'(x_c))\ \Delta x = \Delta y = f(x_c)$$

or $$x_+ = x_c - \frac{f(x_c)}{f'(x_c)} \qquad (1.1.1)$$

which gives $$x_+ = 2 - \frac{1}{4} = 1.75.$$

Applying the same process from the new current estimate $x_c = 1.75$, (1.1.1) gives $x_+ = 1.732\overline{142857}$, which already has four correct digits of $\sqrt{3}$. One more iteration gives $x_+ \cong 1.7320508$, which has eight correct digits.

The method we have just developed is called the Newton's Method. In developing the above Newton's method what we have done is that; at each iteration we have constructed a local model (which is known as affine model) of our function f(x) and solved for the root of the model.

In the present case, our model is the unique line with function value $f(x_c)$ and slope $f'(x_c)$ at the point x_c given by.

$$M_c(x) = f'(x_c) + f'(x_c)(x-x_c), \qquad (1.1.2)$$

Denotation: In this paper the local model at x_k for finding an approximate root of a nonlinear equation and an approximate local minimum of a nonlinear functional are denoted by M_k and m_k, respectively.

In general in solving nonlinear problem, what Newton's method does is that, it generates a sequence of points, by an iterate process, that we hope come increasingly close to a solution.

x_*

Definition 1.1.1: Let $x_* \in R$, $x_k \in R$, $k = 0,1,2,\ldots..$

a) The sequence $\{x_k\}$ is said to be linearly convergent to x_* if and only if $\{x_k\}$ converges to x_* and there is a constant $c \in [0,1)$ and an integer $k_0 \geq 0$ such that for all $k \geq K_0$,

$$|x_{k+1} - x_*| \leq c|x_k - x_*|, \qquad (1.1.3)$$

b) If for some non negative sequence $\{c_k\}$ of real number that converges to zero,

$$|x_{k+1} - x_*| \leq c|x_k - x_*|, \qquad (1.1.4)$$

then x_k is said to converge super linearly to x_*.

c) If there exist constants $c \geq 0$ and $k_0 \geq 0$ such that $\{x_k\}$ converges to x_* and for $k \geq k_0$,

$$|x_{k+1} - x_*| \leq c|x_k - x_*|^2, \qquad (1.1.5)$$

then $\{x_k\}$ is said to converge to x_* quadratically.

Examples:

1. Let $x_k = 1 + 2^{-k}$, $k = 0, 1,2, \ldots$ and $x_* = 1$

   Then clearly the sequence $\{x_k\}$ converges to x_* linearly with $c = \frac{1}{2}$ and $k_0 = 0$

2. Let $x_k = 1 + 2^{-2^k}$, $k = 0, 1, 2, \ldots$ and $x_* = 1$

   Then the sequence $\{x_k\}$ converges to x_* quadratically with $c = 1$.

Definition 1.1.2: A function $f: D \to R$ is said to be lipschitz continuous with constant Υ in D written $f \in \text{Lip}_\Upsilon(D)$, if for every $x, y \in D$,

$$|f(x) - f(y) \leq \Upsilon|x - y|. \qquad (1.1.6)$$

Lemma 1.1.1 For an open interval D, Let $f: D \to R$ and let $f' \in \text{Lip}_\Upsilon(D)$. Then for any $x,y \in D$,

$$|f(y) - f(x) - f'(x)\,(y-x)| \leq \frac{\gamma(y-x)^2}{2} \qquad (1.1.7)$$

Proof: From basic calculus, $f(y) - f(x) = \int_x^y f'(z)dz$, or equivalently,

$$f(y) - f(x) - f'(x)\,(y-x) = \int_x^y [f'(z) - f'(x)]dz \qquad (1.1.8)$$

Making the change of variables.

$z = x + t(y-x)$, $dz = dt(y-x)$, (1.1.8) becomes

$$f(y) - f(x) - f'(x)(y-x) = \int_0^1 [f'(x+t(y-x)) - f'(x)](y-x)dt,$$

So by the triangular inequality applied to the integral and the Lipschitz continuity of f', we have that

$$|f(y) - f(x) - f'(x)(y-x)| = \left| \int_0^1 [f'(x + t(y-x)) - f'(x)](y-x)dt \right|$$

$$\leq \int_0^1 |f'(x + t(y-x)) - f'(x)| \, |y - x| \, dt$$

$$\leq \int_0^1 \gamma |t(y-x)| \, |y-x| \, dt$$

$$= \int_0^1 \gamma |y-x)|^2 t \, dt = \frac{\gamma \, |y-x|^2}{2}$$

Remark: In the above Lemma the term in the absolute value sign is $f(y) - M_x(y)$ and hence the Lemma shows that if $f' \in Lip_\gamma(D)$, then we can obtain a bound on the error the affine model makes in approximating f at x.

Theorem 1.1.2 Let $f: D \to R$, for an open interval D, let $f' \in Lip_\gamma(D)$ Assume that for some $\rho > D$, $|f'(x)| \geq \rho$ for every $x \in D$. If $f(x) = 0$ has a solution $x_* \in D$, then there is some $\eta > 0$ such that; if $|x_0 - x_*| < \eta$, then the sequence $\{x_k\}$ generated by

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \; k = 0, 1, 2, \ldots,$$

exists and converges to x_*. Furthermore, for $k = 0,1,2,\ldots$

$$|x_{k+1} - x_*| \leq \frac{\gamma}{2\rho} |x_k - x_*|^2 \qquad (1.1.9)$$

Proof:

Let $\tau \in (0, 1)$, let α be the radius of the largest open interval around x_* that is contained in D, and define $\eta = \min \{x, \tau(2\rho/\gamma)\}$

We will show by induction that for $k = 0,1,2,\ldots$ (1.1.9) holds, and

$$|x_{k+1} - x_*| \leq \tau |x_k - x_*| < \eta.$$

The proof simply shows at each iteration that the new error. $|x_{k+1} - x_*|$ is bounded by a constant times the error the affine model makes in approximating f at x_*, which from Lemma 1.1.1 is $\frac{\gamma|x_k - x_*|^2}{2}$

For $k = 0$,

$$x_1 - x_* = x_0 - \frac{f(x_0)}{f'(x_0)} - x_* = x_0 - x_* - \frac{f(x_0) - f(x_*)}{f'(x_0)}$$

$$= \frac{1}{f'(x_0)} [\, f(x_*) - f(x_0) - f'(x_0)(x_* - x_0) \,]$$

The term in brackets is $f(x_*) - M_0(x_*)$, the error at x_* in the local affine model at $x_c = x_0$.

Thus from Lemma 1.1.1, $|x_1 - x_*| = \left| \frac{1}{f'(x_0)} [f(x_*) - f(x_0) - f'(x_0)(x_* - x_0)] \right|$

$$= \left| \frac{1}{f'(x_0)} \right| |f(x_*) - f(x_0) - f'(x_0)(x_* - x_0)| \leq \frac{1}{|f'(x_0)|} \frac{\gamma|x_k - x_*|^2}{2}$$

$$= \frac{\gamma}{2|f'(x_0)|} |x_* - x_0|^2$$

and by the assumption on $f'(x)$ we get

$$|x_1 - x_*| \leq \frac{\gamma}{2\rho} |x_0 - x_*|^2$$

Since $|x_0 - x_*| \leq \eta \leq \tau \cdot 2\rho/\gamma$, we have

$$|x_1 - x_*| \leq \frac{\gamma}{2\rho} \times \frac{\tau . 2\rho}{\gamma} |x_0 - x_*| = \tau|x_0 - x_*| \leq \tau\eta < \eta, \text{ since } \tau \in (0, 1)$$

The proof of the induction sets proceeds identically

Remark: Theorem 1.1.2 guarantees the convergence of Newton's method only from a good starting point x_0, and indeed it is easy to see that Newton's method may not converge at all if $|x_0 - x_*|$ is large. For example, consider the function $f(x) = \arctan x$

(see Figure 1.1.2) For some $x_c \in [1.39, 1.40]$, if $x_0 = x_c$, the Newton's method will produce the cycle $x_1 = -x_c$, $x_2 = x_c$, $x_3 = -x_c$, ...

If $|x_0| < x_c$, Newton's method will converge to $x_* = 0$, but if $|x_0| > x_c$, Newton's method will diverge, i.e the error $|x_k - x_*|$ will increase at each iteration.

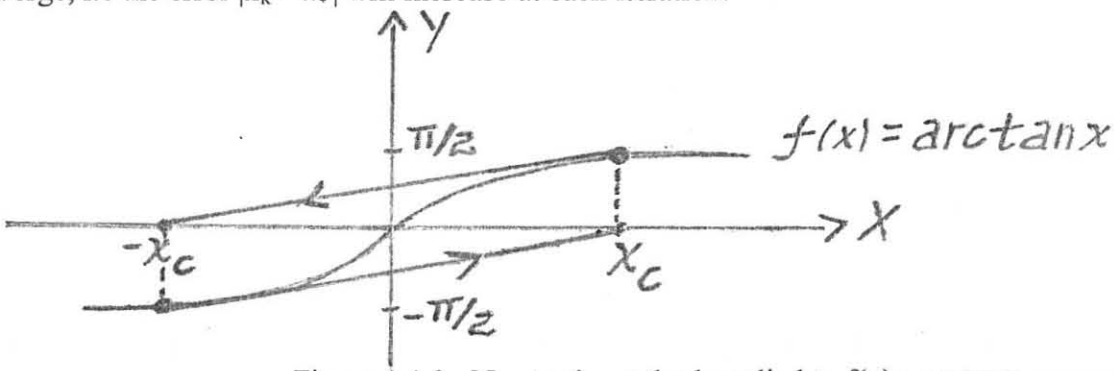


Figure 1.1.2 Newton's method applied to f(x) = arctanx

1.1.2 **Globally Convergent Method.**

In the above example we have seen that Newton's method may not converge at all from bad starting point (i. e when $|x_0 - x_*|$ is large). Such iterative methods are known as local methods on the other hand those iterative methods that guarantee convergence from any starting point are called global methods.

In this paper among the different global methods, we shall discuss the one, which extends to multiple dimension called backtracking. Think of Newton's method as having suggested not only the step $x_N = x_c - f(x_c)/f'(x_c)$, but also the direction in which that step points (assume $f'(x_c) \neq 0$). Although the Newton's step may actually cause an increase in the absolute value of the function, its direction always will be one in which the absolute function value decreases initially (see Figure 1.1.3)

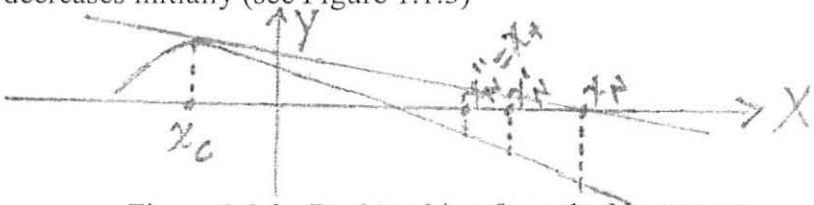


Figure 1.1.3 Backtracking from the Newton step.

Thus if the Newton point x_N doesn't produce a decrease in $|f(x)|$, a reasonable strategy is to backtrack from x_N toward x_c until one finds

x_+ for which $|f(x_+)+ < |f(x_c)|$.

A possible iteration is

$$x_+ = x_c - \frac{f(x_c)}{f'(x_c)}$$

while $|f(x_+)| \geq |f(x_c)|$ do

$$x_+ \leftarrow \frac{x_+ + x_c}{2} \qquad (1.1.10)$$

Iteration (1.1.10) is an example of a hybrid algorith, one that attempts to combine global convergence and fast local convergence by first trying the Newton step at each iteration, but always insisting that the iteration decreases some measure of the closeness to a solution.

Below is the general form of a class of hybrid algorithms for finding a root of nonlinear equation.

ALGORITHM 1.1.1 General hybridquasi – Newton algorithm for solving one nonlinear equation in one unknown:

given $f: R \rightarrow R$, x_0,

for $k = 0, 1, 2, \ldots,$ do

1. decide whether to stop; if not
2. make a local model of f around x_k, and find the point x_N that solves (or comes to solving) the model problem.
3. (a) decide whether to take $x_{k+1} = x_N$, if not

   (b) choose x_{k+1} using a global strategy (make more conservative use of the solution to the model problem).

1.1.3 Methods When The Function is Not Differentiable.

In the preceding section we have been using $f'(x)$ in modeling f near the current solution x_c by the line tangent to f at x_c. But in many practical applications, $f'(x)$ is not available. Hence when $f'(x)$ is unavailable, we replace this model by the secant line that goes through f at x_c and at some near by point $x_c + h_c$ (see Figure 1.1.4).

The slope of this line is

$$a_c = \frac{f(x_c + h_c) - f(x_c)}{h_c} \qquad (1.1.11)$$

and so the model we obtain is the line

$$\hat{M}_c(x) = f(x_c) + a_c(x - x_c)$$

Therefore the quasi – Newton step to the zero of $\hat{M}_c(x)$ then becomes

$$\hat{x}_N = x_c - \frac{f(x_c)}{a_c}.$$

Remark: If h_c is chosen to be a small number, a_c is called a finite difference approximation to $f'(x_c)$.

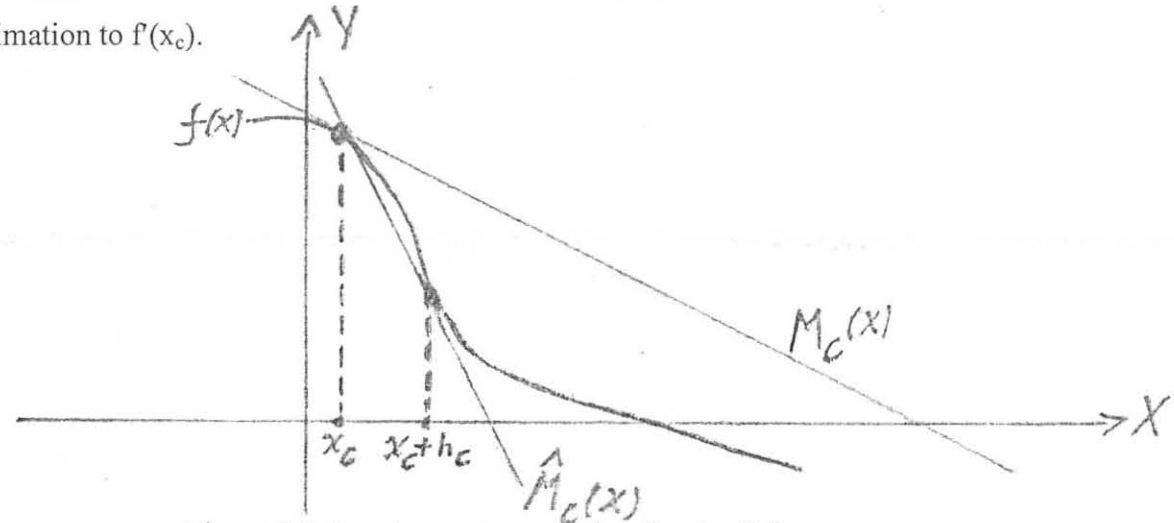


Figure 1.1.4 A secant approximation to f(x)

Example: Finite – difference Newton's method applied to $f(x) = x^2 - 1$ with $h_k = 10^{-7}x_k$ and $x_0 = 2$

x_0	2
x_1	1.2500000266453
x_2	1.025000169057
x_3	1.0003048001120
x_4	1.0000000464701
x_5	1.0

Notice: here $x_{k+1} = x_k - \frac{f(x_k)}{a_k}$

where $a_k = \frac{f(x_k + h_k) - f(x_k)}{h_k}$,

for k = 0, 1, 2,....

If we apply Newton's method to $f(x) = x^2 - 1$ with $x_0 = 2$ we get;

x_0	2
x_1	1.25
x_2	1.025
x_3	1.0003048780489
x_4	1.000000046611
x_5	1.0

In the above example one can see how similar Newton's method and the finite – difference Newton's method are.

In general we have the following corollary which tells us how close, as a function of h_c, the finite-difference approximation (1.1.11) is to $f'(x_c)$.

Corollary 1.1.3 Let $f:D \to R$ for an open interval D and let $f' \in Lip_\gamma(D)$. Let x_c, $x_c + h_c \in D$ and define a_c by (1.1.11). Then

$$|a_c - f'(x_c)| \le \frac{\gamma|h_c|}{2} \qquad (1.1.12)$$

Proof: From basic calculus $f(x_c + h_c) - f(x_c) = \int_{x_c}^{x_c+h_c} f'(z)dz$ or equivalently

$$f(x_c + h_c) - f(x_c) - h_c f'(x_c) - h_c f'(x_c) = \int_{x_c}^{x_c+h_c} [f'(z) - f'(x_c)]dz.$$

Making the change of variables

$$z = x_c + th_c,\ dz = dth_c$$

We get

$$f(x_c + h_c) - f(x_c) - h_c f'(x_c) = \int_0^1 [f'(x_c + th_c) - f'(x_c)]\, h_c dt,$$

and so by the triangle inequality applied to the integral and the Lipschitz continuity of f'

$$|f(x_c + h_c) - f(x_c) - h_c f'(x_c)| \le |h_c| \int_0^1 \gamma|th_c|dt$$

$$= \frac{\gamma|h_c|^2}{2}$$

Dividing both sides by $|h_c|$ where $h_c \neq 0$ gives the desired result,

Corollary 1.1.4 Let $x_+ = \hat{x}_N = x_c - \frac{f(x_c)}{a_c}$. If we define

$e_c = |x_c - x_*|$ and $e_+ = |x_+ - x_*|$, and if $f' \in \mathrm{Lip}_\gamma(D)$, then

$$e_+ \le \frac{\gamma}{2|a_c|}(e_c + |h_c|)\, e_c \qquad (1.1.13a)$$

Proof:

$$x_+ - x_* = x_c - x_* - \frac{f(x_c)}{a_c}$$

$$= a_c^{-1}[f(x_*) - f(x_c) - a_c(x_* - x_c)]$$

$$= a_c^{-1}[f(x_*) - \hat{M}_c(x)]$$

$$= a_c^{-1}\{f(x_*) - f(x_c) - f'(x_c)(x_* - x_c) + [f'(x_c) - a_c](x_* - x_c)\}$$

$$= a_c^{-1}\{\int_{x_c}^{x_*} [f'(z) - f'(x_c)]dz + [f'(x_c) - a_c](x_* - x_c)\}$$

and so by the triangular inequality and by the Lipschitz continuity of f' we get

$$|x_+ - x_*| \le |a_c^{-1}| \left(\int_{x_c}^{x_*} |f'(z) - f'(x_c)| dz + |f'(x_c) - a_c| \, |x_* - x_c| \right)$$

$$\le |a_c^{-1}| \left(\frac{\gamma}{2} |x_c - x_*|^2 + |f'(x_c) - a_c| \, |e_c - x_*| \right)$$

That is, $e_+ \le |a_c^{-1}| \left(\frac{\gamma}{2} e_c^2 + |f'(x_c) - a_c| e_c \right)$ (1.1.13b)

Putting (1.1.12) into (1.1.13b) gives the desired result.

Corollary 1.1.5 If $|f'(x)| \ge \rho > 0$ in a neighborhood of x, for $|h_c|$ sufficiently small and $x_+ \in D$, then

$$e_+ \le \frac{\gamma}{\rho} (e_c + |h_c|) e_c \qquad (1.1.14)$$

Proof: From (1.1.12) for $|h_c|$ sufficiently small $a_c \approx f'(x_c)$. Thus $\left|a_c^{-1}\right| \approx |(f')^{-1}| \le \rho^{-1}$. This implies $|a^{-1}| \le 2\rho^{-1}$. This together with (1.1.13a) gives

$$e_+ \le \frac{\gamma}{2\rho} (e_c + |h_c|) \, e_c.$$

1.2 Minimization of a Nonlinear Functional of One Variable

1.2.1 Newton's Method.

We begin this section by proving two theorems which gives us a necessary and sufficient conditions for finding a minimum point for one variable functions.

Moreover a proof of the first suggests an algorithm.

Theorem 1.2.1 Let $f \in C^1(D)$ for an open interval D, and $z \in D$. If $f'(z) \neq 0$, then for any s with $f'(z).s < 0$, there is a constant $t > 0$ for which $f(z + \lambda s) < f(z)$, for every $\lambda \in (0,t)$.

Proof: Without loss of generality let $f'(z) > 0$. Then from $f'(z)s < 0$ follows that $s < 0$. Now since f' is continuous and $f'(z) > 0$, there is an open interval 0 about z such that

$f'(x) > 0$ for all x in 0. In other words there is $t > 0$ such that $f'(z + \lambda s) > 0$ for all λ in (0,t)

For any such λ, we get

$$f(z + \lambda s) - f(z) = \int_0^{\lambda} f'(z + \alpha s)\, s\, d\alpha < 0$$

Remark: Theorem 1.2.1 suggests that a local minimum of a continuously differentiable function f must com at a point where $f'(x) = 0$. Graphically, this just says that the function cannot initially decrease from such a point. Hence solving $f'(x) = 0$ is necessary for finding a minimizing point for f, but not sufficient.

Theorem 1.2.2. Let $f \in C^2(D)$ for an open interval D and Let $x_* \in D$ for which $f'(x_*) = 0$ and $f''(x_*) > 0$. Then there is some open subinterval $D' \subset D$ for which $x_* \in D'$ and $f(x) > f(x_*)$ for any other $x \in D'$.

Proof: By continuity of f'', there exists an open interval D' about x_* such that $f'' > 0$ on D'.

Then by the extended mean value theorem, for any $x \in D'$, there is an $\bar{x} \in (x_*, x) \subset D'$ for which

$$f(x) - f(x_*) = f'(x_*)(x - x_*) + \frac{1}{2} f''(\bar{x})(x - x_*)^2 > 0$$

Since $f'(x_*) = 0$ and $\bar{x} \in D'$, the assertion follows.

The Hybrid Newton's Method Strategy to compute a Minimum Point

As theorem 1.2.1 suggests solving $f'(x) = 0$ is the necessary condition for finding a minimizing point for f. Hence the easiest way to the class of algorithms we will use is to think of solving $f'(x) = 0$ by applying the hybrid Newton's method strategy of section 1.1.2, only making sure that we find a minimum and not a maximum point by incorporating into the globalizing strategy the requirement that $f(x_k)$ decreases as K increases.

An interation of the hybrid method starts by applying Newton's method, or finite difference method, to $f'(x) = 0$ from the current point x_c. The Newton step is

$$x_+ := x_c - \frac{f'(x_c)}{f''(x_c)} \qquad (1.2.1)$$

Interms of the model problems we have the following as to the meaning of the above step.

Since (1.2.1) is derived by making an affine model of $f'(x)$ around x_c, it is equivalent to having made a quadratic model of $f(x_*)$ around xc, $mc(x) = f(x_c) + f'(x_c)(x - x_c) + \frac{1}{2} f''(x_c)(x - x_c)^2$, and setting x_+ to the critical point of this model.

Remark: i) The above quadratic model can be derived from Taylor's Theorem.
ii) The critical point of the quadratic model, $m_c(x)$, is obtained by solving $m_c'(x)=0$ for x.

Our global strategy for minimization will differ from that in section 1.1.2 in that, rather than deciding to use the Newton point x_N by the condition $|f'(x_N)| < |f'(x_c)|$, which measures progress toward a zero of $f'(x)$, we will want $f(x_N) < f(x_c)$, which indicates progress toward a minimum.

If $f(x_N) \geq f(x_c)$, we consider the following cases.

Case 1: $f'(x_c)(x_N - x_c) < 0$

From the Newton step for x_N given by (1.2.1), we have that $f''(x_c) > 0$, whenever $f'(x_c)(x_N - x_c) < 0$. That is the leading coefficient of the quadratic model m_c, which is used to derive the Newton step x_N, is positive.

Hence the quadratic model has a minimum and not a maximum. Moreover theorem 1.2.1 suggest that f(x) must initially decrease in the direction from x_c towards x_N. (see Figure 1.2.1)

Thus we can find an acceptable next point x_+ by backtracking from x_N toward x_c.

Case 2: $f'(x_c)(x_N - x_c) > 0$

Again from Newton step (1.2.1), we have that $f''(x_c) < 0$, whenever $f'(x_c)(x_N - x_c) > 0$. This yields that the quadratic model m_c, which is used to derive the Newton step x_N, has a maximum and not a minimum. Thus f(x) initially increases going from x_c toward x_N. Therefore we should take a step in the opposite direction. One strategy is to try a step of length $|x_N - x_c|$ and then backtrack, if necessary, until $f(x_+) < f(x_c)$ (see Figure 1.2.2)

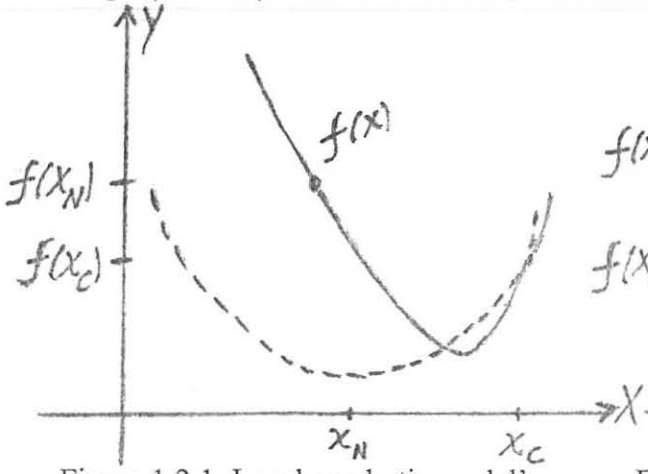


Figure 1.2.1 Local quadratic model's having minimum implies f(x) decreases initially from x_c toward x_N.

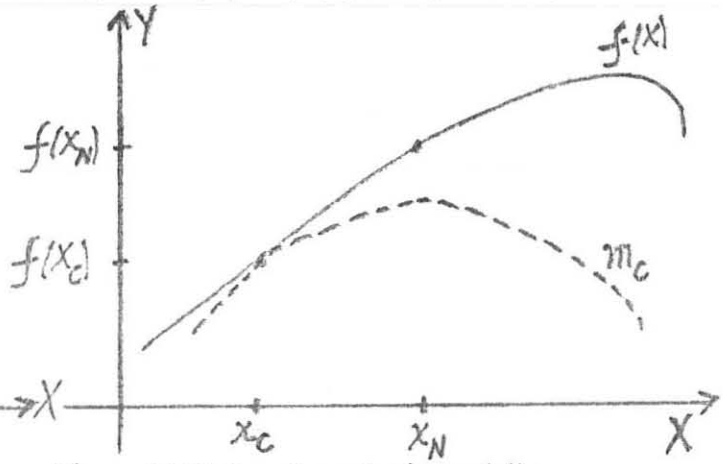


Figure 1.2.2 Local quadratic model's having maximum implies f(x) increases initially from x_c toward x_N.

Remark: Incase m_c of $f'(x)$, $f''(x)$ or both of this derivatives are not available we can approximate them by finite difference, in a similar way as we discussed in section (1.1.3). The corresponding method is known as finite difference Newton's method (for minimization). The detail of this method will be considered in chapter 2.

CHAPTER II

Globally Convergent Modification of Newton's Method.

This chapter is devoted to the major idea for proceedings when the Newton step is unsatisfactory. In section 2.1 we reintroduce Newton's method for unconstrained minimization and moreover we will set the frame work for algorithm we want to consider. In section 2.2 we will discuss the global approach, modern versions of the traditional idea of backtracking along the Newton direction if a full Newton step is unsatisfactory, for unconstrained minimization problem. Finally in section 2.3 we will discuss an application of the global method for solving systems of nonlinear equations whenever a full Newton step is unsatisfactory.

2.1 Introduction

Consider the unconstrained minimization problem:

$$\min_{x\in\Re^n} f:\Re^n \to \Re$$

where f is assumed twice continuously differentiable. Just at in chapter 1, we model f at the current point x_c by the quadratic model

$$m_c(x_c + p) = f(x_c) + \nabla f(x_c)^T p + \frac{1}{2} p^T \nabla^2 f(x_c) p.$$

and solve for the point $x_+ = x_c + s_N$, where $\nabla m_c(x_+) = 0$, a necessary condition for x_+ to be the minimizer of m_c.

This corresponds to the following algorithm.

Algorithm 2.1.1 Newton's Method for unconstrained Minimization

Given $f:\Re^n \to \Re$ twice continuously differentiable, $x_0 \in \Re^n$; at each iteration k,

Find x_k such that $\nabla^2 f(x_k)\, s_k^N = -\nabla f(x_k)$,

$$x_{k+1} := x_k + s_k^N$$

Example: Let $f(x_1, x_2) = (x_1 - 2)^4 + (x_1 - 2)^2 x_2^2 + (x_2 + 1)^2$, which has its minimum at $x_* = (2, -1)^T$.

Algorithm 2.1.1, started from $x_0 = (1, 1)^T$, produces the following sequence of points. (to eight figure on a CDC machine in single precision):

X_k	$f(x_k)$
$x_0 = (1, 1)^T$	6.0
$x_1 = (1, -0.5)^T$	1.5
$x_2 = (1.3913043, -0.69565217)^T$	4.09×10^{-1}
$x_3 = (1.7459441, -0.94879809)^T$	6.49×10^{-2}
$x_4 = (1.9862783, -1.0482081)^T$	2.53×10^{-3}
$x_5 = (1.9987342, -1.0001700)^T$	1.63×10^{-6}
$x_6 = (1.9999996, -1.0000016)^T$	2.75×10^{-12}

Algorithm 2.1.1 is simply the application of Newton's method (Alg. – for system of nonlinear equations) to the system $\nabla f(x) = 0$ of nonlinear equations in n unknowns, because it steps at each iteration to the zero of the affine model of $\nabla f(x)$ defined by

$$M_k(x_k + p) = \nabla [m_k (x_k + p)] = \nabla f(x_k) + \nabla^2 f(x_k)p.$$

As it is evident in the example if x_0 is sufficiently close to a local minimizer x_* of f with $\nabla f(x_*)$nonsingular (and therefore positive positive definite by a corollary on Newton's method), then the sequence $\{x_k\}$ generated by Algorithm 2.1.1 will converge quadratically to x_*. This is the main advantage of Newton's method.

On the other hand even as a local method, Newton's method is not specifically geared to the minimization problem; there is nothing that makes the sequence $\{x_k\}$ generated by Newton's step less likely to proceed toward a maximizer or saddle point of where ∇f is also zero.

In general, even though Newton's method is intended primarily as a local method to be used when the current solution approximation is close enough to a minimizer, we would like to make some modifications, whenever necessary so that the sequence $\{x_k\}$ generated by the Newton step will converge to the solution where x_0 is outside the convergence region of Newton's method.

The basic idea in forming a successful algorithm for unconstrained minimization (as well as for nonlinear equation) is to combine a globally convergent strategy with a fast local strategy in a way that derives the benefits of both. The most important point is to try Newton's method, or some modification of it, first at each iteration insisting that f decreases otherwise fall back on a step dictated by a global method.

The framework for doing this (for both nonlinear equations and unconstrained optimization), is outlined in Algorithm 2.1.2 below. It is a slight expansion of the hybrid algorithm we have seen in chapter 1.

ALGORITHM 2.1.2 Quasi-Newton Algorithm for Nonlinear Equations or Unconstrained Optimization [for optimization, replace $F(x_k)$ by $\nabla f(x_2)$ and J_k by H_k]

Given $F:\Re^n \rightarrow \Re^n$ continuously differentiable, and $x_0 \in \Re^n$. At each interaction k.

1. Compute $F(x_k)$, if it is not already done, and decide whether to stop or continue.
2. Computer J_k to be $J(x_k)$, or an approximation to it.
3. Apply a factorization technique to J_k and estimate its condition number. If J_k is ill – condition, perturb it in an appropriate manner.
4. solve $J_k s_k^N = -F(x_k)$.
5. Decide whether to take a Newton step, $x_{k+1} = x_k + s_k^N$, or to choose x_{k+1} by a global strategy. This step often furnishes $F(x_k)$ to step 1.

Notice that if the global method is chosen and incorporated properly, the algorithm will be globally convergent and moreover it retains the Newton's local

convergence rate close to the solution. The algorithm that takes this approach is called quasi-Newton.

Remark: Step 5 is the topic of this seminar paper. Consequently, henceforth we discuss the same topic and moreover we require, H_k to be positive definite incase of unconstrained minimization problem, and $J(x_k)$ singular for nonlinear equation in Algorithm 2.1.2.

2.2 Globally Convergent Modification of Newton's Method for Unconstrained Minimization

As we have mentioned in the previous section we need to make certain modification whenever Newton's Step is unsatisfactory so that the sequence generate by Algorithm 2.1.2 converges to the solution x_* of the problem. Consequently in this section we discuss the corresponding modification for unconstrained minimization problem.

Now we start our discussion by developing the materials we need later for our global strategy.

2.2.1 Descent Directions for Unconstrained Minimization Problems.

Descent directions are among those materials we will need for our global method. Moreover 'they help one to have a good insight into the basic idea of the global method we are looking for. Hence we discuss them in this subsection.

Definition 2.2.1.1 Let $f: \Re^n \to \Re$ be continuously differentiable or $\Re^n$, and let $x \in \Re^n$. Then the direction p from a point x in which f decreases initially is called a descent direction.

Remark: From a Lemma an Newton's method for unconstrained minimization we have the following and an alternative definition of descent directions. That is p is a descent direction from x if

$$\nabla f(x)^T p < 0, \qquad (2.2.1.1)$$

where $f \in c^{(1)}(\Re^n)$, $p, x \in \Re^n$ and p is an nonzero perturbation.

Now we can introduce the basic idea of a global method for un constrained minimization. Obviously the geometrical interpretation of the idea of global method for unconstrained minimization can be put as: take steps that lead "downhill" for the function f. This can be restated interms of directions as follows. That is one chooses a direction p from the current point x_c in which f decreases initially and a new point x_+ in this direction from x_c such that $f(x_+) < f(x_c)$. Clearly the direction which satisfies this requirement is a descent direction from x_c.

So far, the only direction we have considered for minimization is the Newton direction $s^N = -H_c^{-1}\nabla f(x_c)$, where H_c is either $\nabla^2 f(x_c)$ or an approximation of it. Now one may raise a question as to whether Newton's direction is a descent direction. To see this; Newton's direction is a descent direction if

$$\nabla f(x_c)^T s^N = -\nabla f(x_c)^T H_c^{-1}\nabla f(x_c) < 0, \text{ by } (2.2.1.1)$$

which is true if H_c^{-1} or, equivalently, H_c is positive definite.

This is why we require H_c to be positive definite in Algorithm. 2.1.2

2.2.2 The Method of Line Searches for Unconstrained Minimization Problem

In the previous subsection we have seen that Newton's direction is a descent direction as long as H_c is positive definite. We are interested in Newton's direction in particular, since we use it in our algorithm. In this section we introduce our global strategy which is known as the method of line search.

The idea of our line-search algorithm is that given a descent direction p_k, we take a step in the direction that yields an "acceptable" x_{k+1}, where the word "acceptable" refers to the decreasing in f (i.e. $f(x_{k+1}) < f(x_k)$).

It is summarized in the following;

at each iteration k:

calculate a descent direction p_k,

set $x_{k+1} := x_k + \lambda_k p_k$ for some $\lambda_k > 0$ that makes x_{k+1} acceptable next iterate.

Graphically, this means we select x_{k+1} by considering the half of a one-dimensional cross section of f(x) in which f(x) decreases initially from x_k (see figure 2.2.2.1)

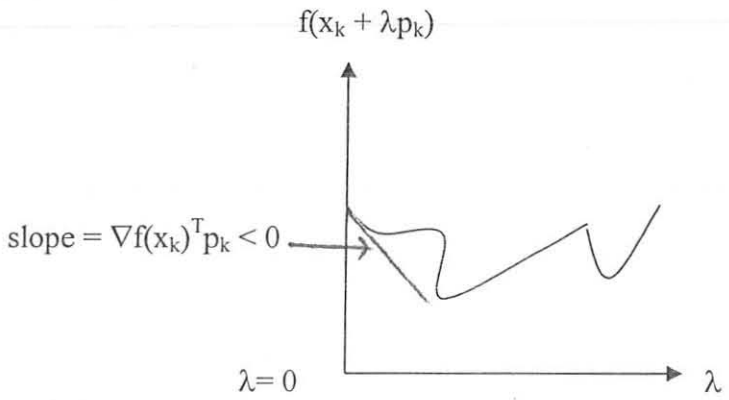


Figure 2.2.2.1A cross section of f(x): $\Re^n \rightarrow \Re$ from x_k in the direction p_k, where $p_k = - H_k^{-1} \nabla f(x_k)$ with H_k positive definite.

The other tool we need for our line search algorithm is λ_k. Actually the term "line search" itself refers to a procedure for choosing λ_k. Next we will give a weak acceptance criteria for $\{\lambda_k\}$ that lead to methods that perform as well in theory and better in practice.

The common procedure now is to try the full quasi – Newton step first and. if $\lambda_k = 1$ fails to satisfy the criterion in use, to backtrack in a systematic way along the directions defined by that step. Notice that, failure to allow $\lambda_k = 1$ in our procedure for choosing λ_k results in loosing the advantage of Newton's method near the solution, which

is its rate of convergence. Hence it is important that our procedure allow $\lambda_k = 1$ near the solution.

In our next discussion of the choice of λ_k in theory and in practice, we consider the search for $x_+ = x_c + \lambda_c$ along the general direction p from the current estimate.

While no step – acceptance rule will always be optimal, it does seem to be common sense to require that

$$f(x_{k+1}) < f(x_k). \qquad (2.2.2.1)$$

Now we consider two simple one-dimensional examples that show two ways that a sequence of iterates can satisfy (2.2.2.1) but tail to converge to a minimize.

Examples 1) Let $f(x) = x^2$, $x_0 = 2$. If we choose $\{p_k\} = \{(-1)^{k+1}\}$, $\{\lambda_k\} = \{2 + 3(2^{-(k+1)})\}$, then $\{x_k\} = \{2, \frac{-3}{2}, \frac{5}{4}, \frac{-9}{8}, \ldots\} = \{(-1)^k (1 + 2^{-k})\}$. Each p_k is a descent direction, one can verify this by substituting x_k and p_k in the term $2x_kp_k$ which yields negative number for each k and finally we have that $f(x_k)$ is monotonically decreasing with

$$\lim_{k\to\infty} f(x_k) = 1$$

See Figure 2.2.2.2(a) Of course this is not a minimum of any sort for f, and furthermore $\{x_k\}$ has limit points ± 1, so it does not converge.

2) Consider the same function with the same initial estimate, and let us take $\{p_k\} = \{-1\}$, $\{\lambda_k\} = \{2^{k+1}\}$. Then $\{x_k\} = \{2, \frac{3}{2}, \frac{5}{4}, \frac{9}{8}, \cdots\} = \{1 + 2^{-k}\}$, each p_k is again a descent direction, $f(x_k)$ decreases monotonically, and $\lim_{k\to\infty} x_k = 1$, which again is not a minimize of f [see Figure 2.2.2.2(b).

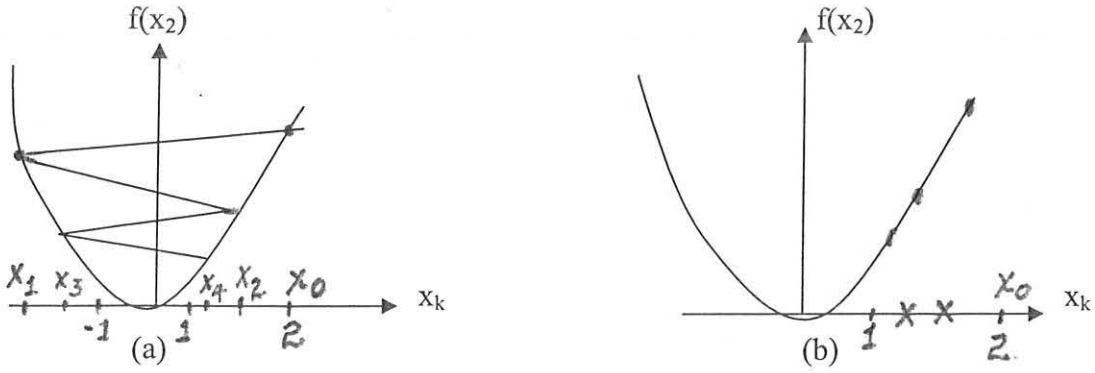


Figure 2.2.2.2 monotonically deceasing sequences of iterates that do not converge to the minimizer.

Not we will discuss separately why the sequences of iterates we have constructed in the above examples fail to converge to a minimizer, when both satisfy (2.2.2.1). Moreover we will see how they can be improved (in each case) so that we get the desired result. That is in each case we set the corresponding step-acceptance criterion, which we will use latter for proving some amazing powerful results, we need.

The problem in Example 1 is that we achieved very small decreases in values of f relative to the lengths of the steps. we can fix this by requiring that the average rate of decrease from $f(x_c)$ to $f(x_+)$ be at least some prescribed fraction of the initial rate of decrease in that direction; that is we pick an $\alpha \in (0, 1)$ and choose λ_c from among those $\lambda > 0$ that satisfy

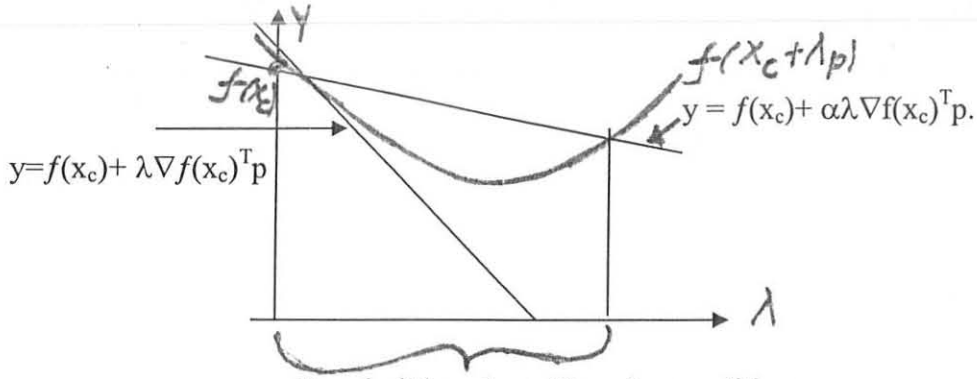
$$f(x_c + \lambda_p) \leq f(x_c) + \alpha\lambda\nabla f(x_c)^T p. \qquad 2.2.2.2.\ (a)$$

(see Figure 2.2.2.3)

Equivalently, λ_c must be chosen so that

$$f(x_+) \leq f(x_c) + \alpha\nabla f(x_c)^T (x_+ - x_c). \qquad (2.2.2.2b)$$

Notice that this precludes the unsuccessful choice of points in case of the first example but the second.



Permissible value of λ under condition

(2.2.2.2) ($\alpha = 0.1$)

Figure 2.2.2.3 Permissible values of λ condition (2.2.2.2.) ($\alpha = 0.1$)

The Problem in Example 2 is that the steps are too small, relative to the initial rate of decrease of f. Hence we need a condition which essures sufficiently large steps. Among various conditions which assure suffiently large steps we will present the one which is appropriate for our purpose. We will require that the rate of decrease of f in the direction p at x_+ be larger than some prescribed fraction of the rate of decrease in the direction p at x_c; that is,

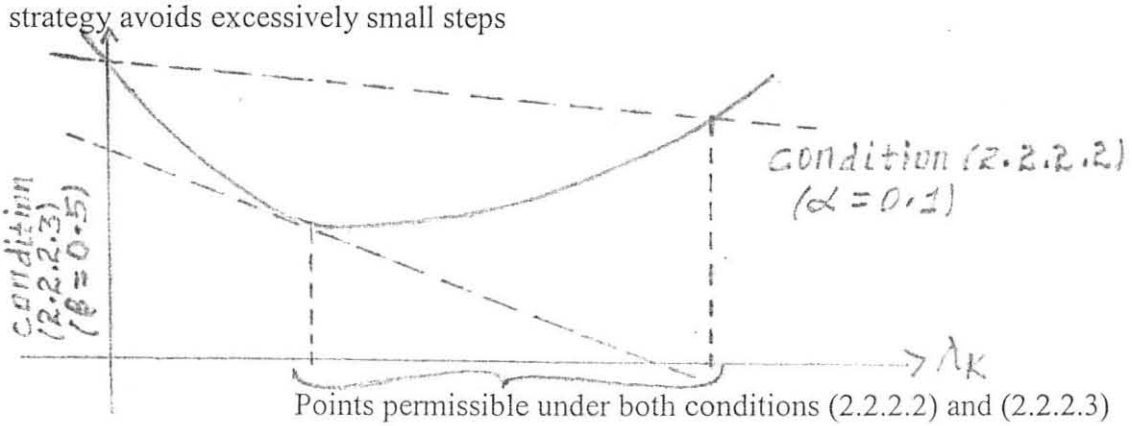
$$\nabla f(x_+)^T p \triangleq \nabla f(x_c + \lambda_c p)^T p \geq \beta\nabla f(x_c)^T p. \qquad (2.2.2.3.a)$$

or equivalently,

$$\nabla f(x_+)^T (x_+ - x_c) \geq \beta\nabla f(x_c)^T (x_+ - x_c) \qquad (2.2.2.3.b)$$

for some fixed constant $\beta\in(\alpha, 1)$ (see Figure 2.2.2.3). The condition $\beta > \alpha$ guarantees that (2.2.2.2) and (2.2.2.3) can be satisfied simultaneously.

In practice we need only (2.2.2.2) for our propose, since the use of a backtracking strategy avoids excessively small steps



Points permissible under both conditions (2.2.2.2) and (2.2.2.3)

Figure 2.2.2.3 The two line search conditions.

We close this subsection by giving an example which demonstrates the effect of conditions (2.2.2.2) and (2.2.2.3\) on a simple functions.

Example 3. Let $f(x_1, x_2) = x_1^4 + x_1^2 + x_2^2$, $x_c = (1, 1)^T$, $p_c = (-3, -1)^T$, and let $\alpha = 0.1$ in (2.2.2.2), $\beta = 0.5$ in (2.2.2.3).

From

$$\nabla f(x) = \begin{bmatrix} 4x_1^3 + 2x_1 \\ 2x_2 \end{bmatrix}, \text{ we get}$$

$$\nabla f(x_c)^T p_c = (6, 2)\,(-3, -1)^T = -20 < 0,$$

and hence p_c is a descent direction for $f(x)$ from x_c.

Now consider $x(\lambda) = x_c + \lambda p_c$. If $x_+ = x(1) = x_c + p_c = (-2, 0)^T$,

$$\nabla f(x_+)^T p_c = (-36, 0)\,(-3, -1)^T = 108 > -10 = \beta \nabla f(x_c)^T p_c,$$

So that x_+ satisfies (2.2.2.3)

on the other hand from,

$$f(x_+) = 20 > 1 = f(x_c) + \alpha \nabla f(x_c)^T p_c,$$

we have that x_+ does not satisfy (2.2.2.2)

Similarly, if $x_+ = x(0.1) = (0.7, 0.9)^T$,

$$f(x_+) = 1.5401 < 2.8 = f(x_c) + \alpha(0.1)\, \nabla f(x_c)^T p_c,$$

So that x_+ satisfies (2.2.2.2), but

$$\nabla f(x_+)^T p_c = (2.772, 1.8)\,(-3, -1)^T = -10.116 < -10 = \beta \nabla f(x_c)^T p_c,$$

So that x_+ does not satisfy (2.2.2.3)

Finally if $x_+ = x(0.5) = (-05, 0.5)$,

$$f(x_+) = 0.5625 < 2 = f(x_c) + \alpha(0.5)\, \nabla f(x_c)^T p_c,$$

So that x_+ satisfies (2.2.2.2) and from

$$\nabla f(x_+)^T p_c = (-1.5, 1)\,(-3, -1)^T = 3.5 > -10 = \beta \nabla f(x_c)^T p_c,$$

we have that x_+ also satisfies (2.2.2.3).

Notice that these three points correspond to the right of, to the left of and in the "permissible" region in Figure (2.2.2.3), respectively.

2.2.3 Convergence Results for Properly Chosen Steps

Using conditions (2.2.2.2) and (2.2.2.3), we can prove some amazing powerful results: that given any direction p_k such that $\nabla f(x_k)^T p_k < 0$, there exist $\lambda_k > 0$ satisfying (2.2.2.2) and (2.2.2.3); that any method that generates a sequence $\{x_k\}$ obeying $\nabla f(x_k)^T(x_{k+1} - x_k) < 0$, (2.2.2.2) and (2.2.2.3) at each iteration is essentially globally convergent; and that close to a minimizer of f where $\nabla^2 f$ is positive definite, Newton steps satisfy (2.2.2.2) and (2.2.2.3).

Theorem 2.2.3.1 Let f: $\Re^n \to \Re$ be continuously differentiable on $\Re^n$. Let $x_k \in \Re^n$, $p_k \in \Re^n$ obey $\nabla f(x_k)^T p_k < 0$, and assume $\{f(x_k + \lambda p_k) \mid \lambda > 0\}$ is bounded below. Then if $0 < \alpha < \beta < 1$, there exist $\lambda_u > \lambda_\ell > 0$ such that $x_{k+1} = x_k + \lambda_k p_k$ satisfies (2.2.2.2) and (2.2.2.3) if $\lambda_k \in (\lambda_\ell, \lambda_u)$.

Proof: By continuously differentiability of f on $\Re^n$ and by the definition of the directional derivative of f at x in the direction of p, we have

$$\lim_{\lambda \to 0} \frac{f(x_k + \lambda p_k) - f(x_k)}{\lambda} = \nabla f(x_c)^T p_k$$

which in turn yields

$$f(x_k + \lambda p_k) - f(x_k) = \lambda \nabla f(x_k)^T p_k, \text{ for all } \lambda > 0 \text{ sufficiently small.}$$

Now for all $\lambda > 0$ sufficiently small, since $0 < \alpha < 1$ and $\nabla f(x_k)^T p_k < 0$ by the hypothesis, we have

$$f(x_k + \lambda p_k) < f(x_k) + \alpha \lambda \nabla f(x_k)^T p_k. \qquad (2.2.3.1)$$

Thus any $x \in (x_k, \hat{x})$ satisfies (2.2.2.2).

On the other hand by the mean value theorem, there exists $\bar{\lambda} \in (0, \hat{\lambda})$ such that

$$\begin{aligned} f(\hat{x}) - f(x_k) &= \nabla f(x_k + \bar{\lambda} p_k)^T (\hat{x} - x_k) \\ &= \nabla f(x_k + \bar{\lambda} p_k)\, \hat{\lambda} p_k, \qquad (2.2.3.2) \end{aligned}$$

and so from (2.2.3.1) and (2.2.3.2),

$$\nabla f(x_k + \bar{\lambda} p_k)^T p_k = \alpha \nabla f(x_k)^T p_k > \beta \nabla f(x_k)^T p_k \qquad (2.2.3.3)$$

Since $\alpha < \beta$ and $\nabla f(x_k)^T p_k < 0$

Finally by the continuity of ∇f, (2.2.3.3). still holds for λ in some interval $(\lambda_\ell, \lambda_u)$ about $\bar{\lambda}$. Therefore, if we restrict $(\lambda_\ell, \lambda_u)$ to be in $(0, \hat{\lambda})$, $x_{k+1} = x_k + \lambda_k p_k$ satisfies (2.2.2.2) and (2.2.2.3) for any $\lambda_k \in (\lambda_l, \lambda_u)$.

Theorem 2.2.3.2 Let $f: \Re^n \to \Re$ be continuously differentiable on $\Re^n$, and assume there exists $\gamma \geq 0$ such that

$$\|\nabla f(z) - \nabla f(x)\|_2 \leq \gamma \|z - x\|_2 \qquad (2.2.3.4)$$

for every $x, z \in \Re^n$. Then, given any $x_0 \in \Re^n$, either f is unbounded below, or there exists a sequence $\{x_k\}$, $k = 0, 1, \ldots$, obeying (2.2.2.2), (2.2.2.3), and either

$$\nabla f(x_k)^T s_k < 0 \qquad (2.2.3.5)$$

or $\quad \nabla f(x_k) = 0$ and $s_k = 0$,

for each $k \geq 0$, where

$$s_k \triangleq x_{k+1} - x_k.$$

Furthermore, for any such sequence, either

(i) $\nabla f(x_k) = 0$ for some $k \geq 0$, or

(ii) $\lim_{k \to \infty} f(x_k) = -\infty$, or

(iii) $\lim_{k \to \infty} \dfrac{\nabla f(x_k)^T s_k}{\|s_k\|_2} = 0.$

Proof: For each k, if $\nabla f(x_k) = 0$, then (i) holds and the sequence is constant subsequently.

If $\nabla f(x_k) \neq 0$, then there exists p_k (for example take $p_k = -\nabla f(x_k)$) such that $\nabla f(x_k)^T p_k < 0$. By Theorem 2.2.3.1, either f is unbounded below, or there exists $\lambda_k > 0$ such that $x_{k+1} = x_k + \lambda_k p_k$ satisfies (2.2.2.2) and (2.2.2.3). In order to simplify notation; let us assume that $\|p_k\|_2 = 1$, so that $\lambda_k = \|s_k\|_2$. (Since $\|x_{k+1} - x_k\|_2 = \|\lambda_k p_k\|_2 = \lambda_k$ and $s_k \triangleq x_{k+1} - x_k$). This constitutes no loss of generality.

So far we have seen that either f is unbounded below, or $\{x_k\}$ exists, and either $\{x_k\}$ satisfies (i) or $s_k \neq 0$ for every k.

Now, if no term of $\{s_k\}$ is zero, then we show either (ii) or (iii) must hold. First we need the following denotation,

$$\sigma_k \triangleq \frac{\nabla f(x_k)^T s_k}{\|s_k\|_2}.$$

By (2.2.2.2) and $\lambda_k\sigma_k < 0$ for every k (notice that $\lambda_k\sigma_k = \nabla f(x_k)^T s_k < 0$), we have that for every $j > 0$,

$$f(x_j) - f(x_0) = \sum_{k=0}^{j-1} (f(x_{k+1}) - f(x_k))$$

$$\leq \sum_{k=0}^{j-1} \alpha \nabla f(x_k)^T s_k$$

$$= \alpha \sum_{k=0}^{j-1} \lambda_k \sigma_k < 0.$$

Hence, either

$$\lim_{j\to\infty} f(x_j) = -\infty \text{ or } \sum_{k=0}^{\infty} \lambda_k \delta_k < \infty \text{ is convergent.}$$

In the first case (ii) is true and we are finished, so we consider the second. In particular we deduce that $\lim_{k\to\infty} \lambda_k\sigma_k = 0$. Now we want to conclude that $\lim_{k\to\infty} \sigma_k = 0$, and so we need to use condition (2.2.2.3), since it was imposed to ensure that the steps do not get too small.

Now by condition (2.2.2.3) we have for each k that

$$\nabla f(x_{k+1})^T s_k \geq \beta \nabla f(x_k)^T s_k$$

and so

$$\nabla f(x_{k+1})^T s_k \geq \nabla f(x_k)^T s_k \geq \beta \nabla f(x_k)^T s_k \geq - \nabla f(x_k)^T s_k$$

or

$$[\nabla f(x_{k+1})^T s_k - \nabla f(x_k)]^T s_k \geq (\beta-1) \nabla f(x_k)^T s_k > 0, \qquad (*)$$

by (2.2.3.5) and $\beta < 1$.

Now applying the Cauchy – Schwarz un equality and (2.2.3.4) to $[\nabla f(x_{k+1}) - \nabla f(x_k)]^T s_k$, we get

$$|[\nabla f(x_{k+1}) - \nabla f(x_k)]^T s_k| \leq \|[\nabla f(x_{k+1}) - \nabla f(x_k)]^T\|_2 \, \|s_k\|_2$$

$$= \|\nabla f(x_{k+1}) - \nabla f(x_k)\|_2 \, \|s_k\|_2$$

$$\leq \gamma \|x_{k+1} - x_k\|_2 \, \|s_k\|_2$$

$$= \gamma \|s_k\|_2^2 = \gamma \lambda_k^2, \qquad (**)$$

Then by using (**) and the definition σ_k, (*) gives us

$$0 < (\beta - 1)\, \lambda_k \sigma_k < \gamma \lambda_k^2,$$

so

$$\lambda_k \geq \frac{\beta - 1}{\gamma} \sigma_k > 0$$

and $$\lambda_k \sigma_k \leq \frac{\beta - 1}{\gamma} \sigma_k^2 < 0.$$

Thus

$$0 = \lim_{k \to \infty} \lambda_k \sigma_k \leq \frac{\beta - 1}{\gamma} \lim_{k \to \infty} \sigma_k^2 \leq 0,$$

which shows that

$$\lim_{k \to \infty} \sigma_k = 0$$

(i.e (iii) is true) and completes the proof.

Note that while Theorem 2.2.3.2 applies readily to any line-search algorithm, it is completely independent of the method for selecting the descent directions or the step lengths. Therefore, this theorem gives sufficient conditions for global convergence, in a weak sense, of any optimization algorithm and hence for our quasi-Newton line-search algorithm. Furthermore, while the Lipschitz condition (2.2.3.4) is assumed on all of $\Re^n$, it is used only in a neighborhood of the solution x_*. Finally, although $\{\sigma_k\} \to 0$ in Theorem 2.23.2 does not necessarily imply $\{\nabla f(x_k)\} \to 0$, it does as long as the angle between $\nabla f(x_k)$ and s_k is bounded away from 90^0. This can easily be achieved in practice. For example, in aquasi-Newton line-search algorithm where $p_k = -H_k^{-1} \nabla f(x_k)$ and H_k is positive definite, all that is needed is that the condition numbers of $\{H_k\}$ are uniformly bounded above. Thus, Theorem 2.2.3.2 can be viewed as implying global convergence

toward f = -∞ or ∇f = 0, although the conditions are too weak to imply that $\{x_k\}$ converges.

The following theorem, due to Dennis and More (1974), shows that our global strategy will permit full quasi-Newton steps $x_{k+1} := x_k - H_k^{-1}\nabla f(x_k)$ close to a minimizer of f as long as $-H_k^{-1}\nabla f(x_k)$ is close to the Newton step.

Theorem 2.2.3.3 Let $f: \Re^n \to \Re$ be twice continuously differentiable in an open convex set D, and assume that $\nabla^2 f \in \text{Lip}_\gamma(D)$. Consider a sequence $\{x_k\}$ generated by $x_{k+1} := x_k + \lambda_k p_k$, where $\nabla f(x_k)^T p_k < 0$ for all k and λ_k is chosen to satisfy (2.2.2.2) with an $\alpha < \frac{1}{2}$, and (2.2.2.3). If $\{x_k\}$ converges to a point $x_* \in D$ at which $\nabla^2 f(x_*)$ is positive definite, and if

$$\lim_{k\to\infty} \frac{\left\|\nabla f(x_k) + \nabla^2 f(x_k) p_k\right\|_2}{\left\|p_k\right\|_2} = 0,$$

then there is an index $k_0 \geq 0$ such that for all $k \geq k_0$, $\lambda_k = 1$ is admissible. Further more $\nabla f(x_*) = 0$, and if $\lambda_k = 1$ for all $k \geq k_0$, then $\{x_k\}$ converges superliearly to x_*.

Proof: The proof is really just a generalization of the easy exercise that if f(x) is a positive definite quadratic and $p_k = -\nabla^2 f(x_k)^{-1}\nabla f(x_k)$, then $\nabla f(x_k)^T p_k < 0$ and $x_{k+1} = x_k + p_k$ satisfies (2.2.2.2) for any $\alpha \leq \frac{1}{2}$, and (2.2.2.3) for any $\beta \geq 0$, and hence the proof is omitted.

Taken together, the conclusion of Theorem 2.2.3.2 and 2.2.3.3 are quite remarkable. They say that if f is bounded below, then the sequence $\{x_k\}$ generated by any algorithm that takes descent steps whose angles with the gradients are bounded away from 90^0, that satisfy conditions (2.2.2.2) and (2.2.2.3), will obey

$$\lim_{k\to\infty} \nabla f(x_k) = 0.$$

Furthermore, if any such algorithm tries a Newton or quasi-Newton step first at each iteration, then $\{x_k\}$ will also converge quadratically or superlinearly to a local minimizer x_* if any x_k is sufficiently close to x_*, and if local convergence assumptions for the Newton or quasi-Newton method are met.

These results are all we will need for the global convergence of the algorithm we discuss in the remainder of this chapter.

2.2.4 Step Selection by backtracking

In this section we specify how our line-search algorithm will choose λ_k. As we have stated in the previous section the modern strategy is to start with $\lambda_k = 1$, and then, if $x_k + p_k$ is not acceptable, "backtrack" (reduce λ_k) until an acceptable $x_k + \lambda_k p_k$ is found. The framework for such an algorithm is given below. Recall that condition (2.2.2.3) is not implemented because the backtracking strategy avoids excessively small steps.

ALGORITHM 2.2.4.1 Backtracking Line-search Framework

Given $\alpha \in (0, \frac{1}{2})$, $0 < \ell < u < 1$

$\lambda_k = 1_j$

While $f(x_k + \lambda_k p_k) > f(x_k) + \alpha\lambda_k \nabla f(x_k)^T p_k$, do

$\lambda_k := \rho\lambda_k$ for some $\rho \in [\ell, u]$;

(*ρ is chosen anew each time by the line search*)

$x_{k+1} := x_k + \lambda_k p_k$;

In practice, α is set quite small, so that hardly more than a decrease in function value is required. Our algorithm uses $\alpha = 10^{-4}$. Finally we discuss the strategy for reducing λ_k (choosing ρ). Let us define

$$\hat{f}(\lambda) \triangleq f(x_k + \lambda p_k),$$

the one-dimensional restriction of f to the line through x_k in the direction p_k. If we need to backtrack, we will use our most current information about $\hat{f}$ to model it, and then take the value of λ that minimizes this model as our next value λ_k in Algorithm 2.2.4.1.

Initially, we have two pieces of information about $\hat{f}(\lambda)$,

$$\hat{f}(0) = f(x_k) \text{ and } \hat{f}'(0) = \nabla f(x_k)^T p_k \qquad (2.2.4.1)$$

After calculating $f(x_k + p_k)$, we also know that

$$\hat{f}(1) = f(x_k + p_k), \qquad (2.2.4.2)$$

So if $f(x_k + p_k)$ doesn't not satisfy (2.2.2.2); that is, $\hat{f}(1) > \hat{f}(0) + \alpha f'(0)$, we model $\hat{f}(\lambda)$ by the one dimensional quadratic satisfying (2.2.4.1) and (2.2.4.2),

$\hat{m}(\lambda) = [\hat{f}(1) - \hat{f}(0) - \hat{f}'(0)]\lambda^2 + \hat{f}'(0)\lambda + \hat{f}(0)$, and calculate the point

$$\hat{\lambda} = \frac{-\hat{f}'(0)}{2[\hat{f}(1) - \hat{f}(0) - \hat{f}'(0)]} \qquad (2.2.4.3)$$

for which $\hat{m}_q'(\hat{\lambda}) = 0$.

Now

$$\hat{m}_q''(\lambda) = 2[\hat{f}(1) - \hat{f}(0) - \hat{f}'(0)] > 0,$$

Since $\hat{f}(1) > \hat{f}(0) + \alpha \hat{f}'(0) > \hat{f}'(0) + \hat{f}(0)$. Thus $\hat{\lambda}$ minimizes $\hat{m}_q(\lambda)$. Also $\hat{\lambda} > 0$, because $f'(0) < 0$. Therefore we take $\hat{\lambda}$ as our new value of λ_k in the Algorithm (see Figure 2.2.4.1). Note that since $\hat{f}(1) > \hat{f}(0) + \alpha \hat{f}'(0)$, we have

$$\hat{\lambda} < \frac{1}{2(1-\alpha)}.$$

In face, if $\hat{f}(1) \geq \hat{f}(0)$, then $\hat{\lambda} \leq \frac{1}{2}$. Thus, (2.2.4.3) gives a useful implicit upper bound of $u \approx \frac{1}{2}$ on the first value of ℓ in Algorithm 2.2.4.1. On the other hand, if $\hat{f}(1)$ is much larger than $\hat{f}(0)$, $\hat{\lambda}$ can be very small. We probably do not want to decrease λ_k too much based on this information, so we impose a lower bound of $\ell = \frac{1}{10}$ in algorithm

2.2.4.1. This means that at the first backtrack at each iteration if $\hat{\lambda}_k \leq \frac{1}{10}$., then we next try $\lambda_k = \frac{1}{10}$

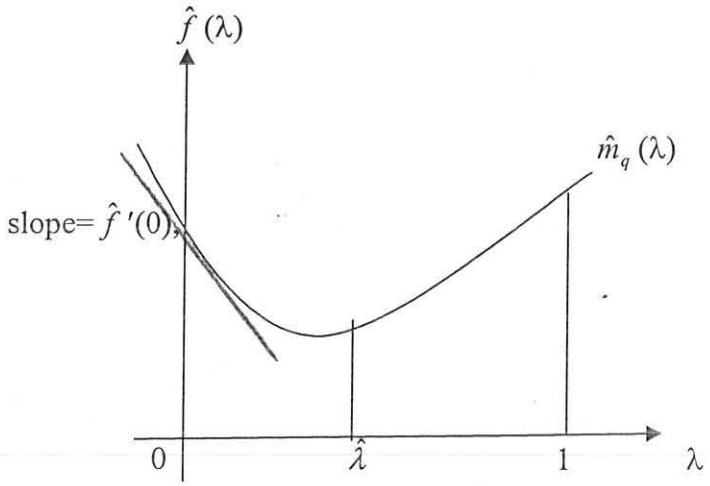


Figure 2.2.4.1 backtracking at the first iteration, using a quadratic model.

Example: Let $f: \Re^n \to \Re$, and x_c and p be given by Example 3 of 2.2.2. since $f(x_c) = 3$ and $f(x_c + p) = 20 > f(x_c)$, $x_c + p$ is not acceptable and a backtrack is needed. Then, from $\hat{f}'(0) = \nabla f(x_c)p_c = -20$, $\hat{f}(1) = f(x_c + p_c) = 20$ and $\hat{f}(0) = f(x_c) = 3$, gives

$$\hat{\lambda} = \frac{20}{2[20-3+20]} = \frac{10}{37} \cong 0.270.$$

Now $x_c + \hat{\lambda} p \cong (0.189, 0.730)^T$ and for $\alpha = 10^{-4}$,

$f(x_c + \hat{\lambda} p) \cong 0.570 \leq 2.99946 = f(x_c) + \alpha\lambda\nabla f(x_c)^T p$, so that $x_c + \hat{\lambda} p$ satisfies condition (2.2.2.2): Therefore $x_+ := x_c + \hat{\lambda} p$.

Now suppose $\hat{f}(\lambda_k) = f(x_k + \lambda_k p_k)$ does not satisfy (2.2.2.2). In this case we need to backtrack again. Although we could use a quadratic model as we did on the first backtrack, we now have four pieces of information about $\hat{f}(\lambda)$. So at this and any subsequent backtrack during the current iteration, we use a cubic model of $\hat{f}$, fit $\hat{m}_{cu}(\lambda)$

to $\hat{f}(0)$, $\hat{f}'(0)$, and the last two value of $\hat{f}(\lambda)$ and, subject to the same sort of upper and lower limits as before. Set λ_k to the value of λ at which $\hat{m}_{cu}(\lambda)$ has its local minmizer (see Figure 2.2.4.2). The reason for using a cubic is that it can more accurately model situations where f has negative curvature, which are likely when (2.2.2.2) has failed for two positive values of λ. Furthermore, such a cubic has a unique minimizer, as illustrated in Figure 2.2.4.2.

The calculation of λ proceeds as follows. Let λprev and λ 2prev be the last two previous values of λ_k. Then the cubic that fits $\hat{f}(0)$ $\hat{f}'(0)$, $\hat{f}'(0)$, $\hat{f}(\lambda\text{prev})$, and $\hat{f}(\lambda 2\text{prev})$ is

$$\hat{m}_{cu}(\lambda) = a\lambda^3 + b\lambda^2 + \hat{f}'(0)\lambda + \hat{f}(0),$$

where

$$\begin{pmatrix} a \\ b \end{pmatrix} = \frac{1}{\lambda prev - \lambda 2 prev} \begin{pmatrix} \frac{1}{\lambda prev^2} & \frac{-1}{\lambda 2 prev^2} \\ \frac{-\lambda 2 prev}{\lambda prev^2} & \frac{\lambda prev}{\lambda 2 prev^2} \end{pmatrix} \begin{pmatrix} \hat{f}(\lambda prev) - \hat{f}(0) - \hat{f}'(0)\lambda prev \\ f(\lambda 2 prev) - \hat{f}(0) - \hat{f}'(0)\lambda 2 prev \end{pmatrix}$$

Its local minimizer $\hat{\lambda}$ is given by,

$$\hat{\lambda} = \frac{-b + \sqrt{b^2 - 3a\hat{f}'(0)}}{3a}.$$

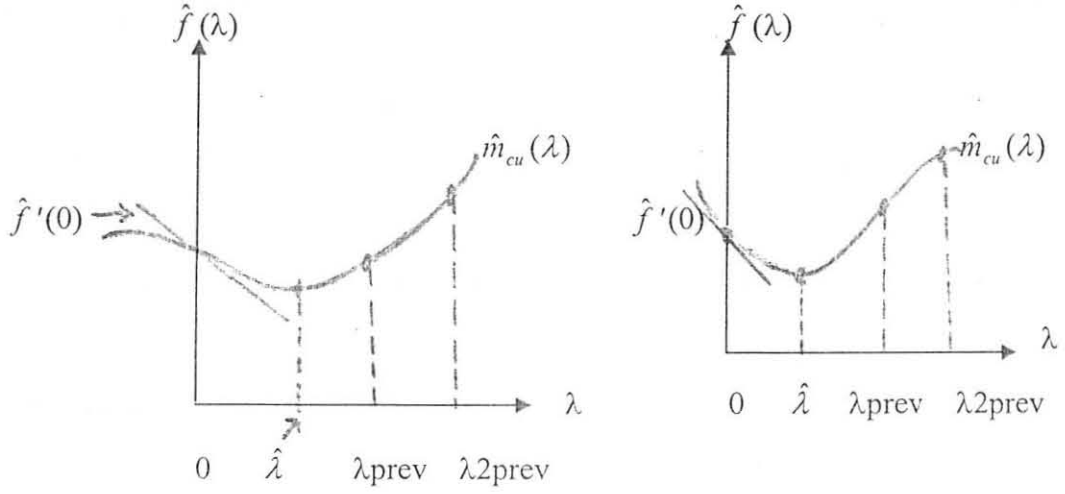


Figure 2.2.4.2 Cubic backtrack – the two possibilities

We close the section by giving the corresponding bounds imposed by u = 0.5 and $\ell = 0.1$

That is if $\hat{\lambda} > \frac{1}{2}\lambda$prev, set $\lambda_k = \frac{1}{2}\lambda$prev and if $\hat{\lambda} < \frac{1}{10}\lambda$prev, set $\lambda_k = \frac{1}{10}\lambda$prev, where λ_k is our new λ.

2.3.1 Global Methods for Systems of Nonlinear Equations.

In chapter one we have discussed that Newton's method may not converge at all from bad starting point (i.e. when the starting guess is not close to any solutions of the one variable nonlinear equation problem) Newton's method shares the same disadvantage in case of multivariable nonlinear equations problem. That is when the current estimate is not close to any solutions of the corresponding nonlinear equations problem we need a tool to decide whether to accept the Newton step as the next iterate. Consequently this section is devoted to the method which essures convergence from a point out side the convergence region of Newton's method. Consider the nonlinear equations problem:

$$\text{given } F: \Re^n \to \Re^n,$$
$$\text{find} \quad x_* \in \Re^n \text{ such that } F(x_*) = 0, \qquad (2.3.1.)$$

where the Newton step is

$$x_+ = x_c - J(x_c)^{-1}F(x_c). \qquad (2.3.2.)$$

Now assume x_c is not close to any solution x_* of (2.3.1.1). How would one decide then whether to accept x_+ as the next iterate? A reasonable answer is that $\|F(x_+)\|$ should be les than $\|F(x_c)\|$ for some norm $\|.\|$, a convenient choice being the ℓ_2 norm $\|F(x)\|^2_2 = F(x)^TF(x)$.

On the other hand requiring that our step result in a decrease of $\|F(x)\|_2$ is the same thing we would require if we were trying to find a minimum of the function $\|F(x)\|_2$. Thus, we have in effect turned our attention to the corresponding minimization problem:

$$\min_{x \in \Re^n} f(x) = \frac{1}{2}F(x)^TF(x), \qquad (2.3.1.3)$$

Where the $\frac{"1"}{2}$ is added for later algebraic convenience. Note that every solution to (2.3.1.1) is a solution to (2.3.1.3), but there may be local minimizer of (2.3.1.3) that are not solutions to (2.3.1.1) (see Figure 2.3.1.1. Hence, although our global strategy for (2.3.1.1) will be based on a global strategy for (2.3.1.3), it is better to use the structure of the original problem to compute the Newton step (2.3.1.2) whenever possible.

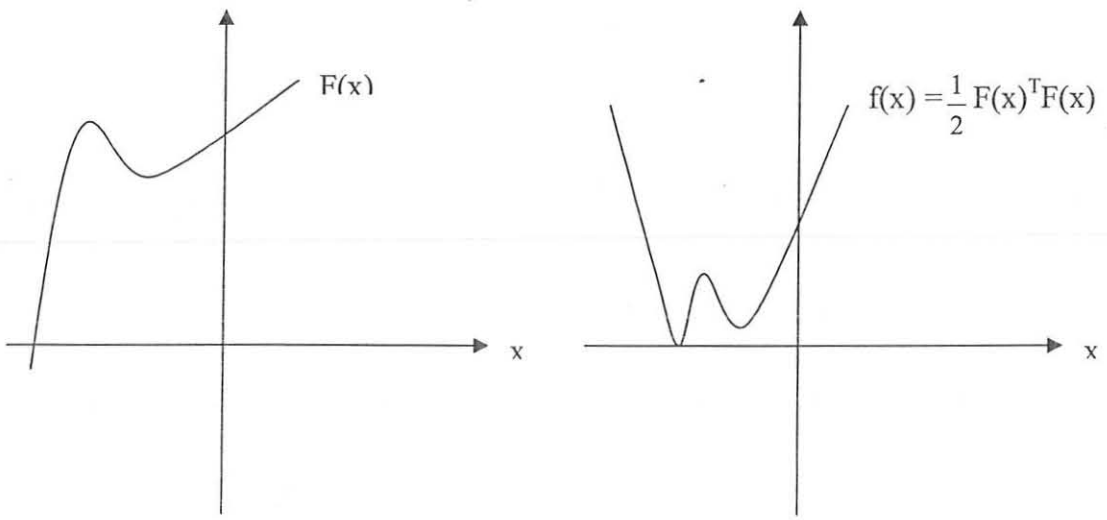


Figure 2.3.1.1 The nonlinear equations and corresponding minimization problem, in one dimension.

2.3.2 Descent Direction for Systememes of Nonlinear Equations

In subsection 2.3.1 saw that our global strategy for nonlinear equations problem will be based on a global strategy for the related minimization problem. consequently in this subsection we will discuss a descent direction for (2.3.1.3)

By Definition 2.2.1.1 we have that descent direction from a current estimate x_c, is a nonzero perturbation $p \in \Re^n$ for which $\nabla^T f(x_c)\, p < 0$ holds, where $f : \Re^n \to \Re$. Thus a descent direction for problem (2.3.1.3) is any direction p for which $\nabla f(x_c)^T p < 0$, where

$$\nabla f(x_c) = \frac{d}{dx}\sum_{i=1}^{n}\frac{1}{2}(f_i(x_c))^2 = \sum_{i=1}^{n} \nabla f_i(x_c)\,.f(x_c) = J(x_c)^T F(x_c).$$

Now we will conform that Newton direction $s^N = -J(x_c)^{-1}F(x_c)$ is a descent direction for (2.3.1.3)

$$\begin{aligned}\nabla f(x_c)^T s^N &= (J(x_c)^T F(x_c))^T(-J(x_c)^{-1}F(x_c)) \\ &= -F(x_c)^T J(x_c)\, J(x_c)^{-1}F(x_c) \\ &= -F(x_c)^T F(x_c) < 0\end{aligned}$$

as long as $F(x_c) \neq 0$. This may seem surprising, but it is geometrically reasonable. Since the Newton step $(x_c + s^N)$ yields a root of

$$M_c(x_c + s) = F(x_c) + J(x_c)\, s,$$

it also goes to a minimum of the quadratic function

$$\hat{m}_c(x_c + s) \triangleq \frac{1}{2} M_c(x_c + s)^T M_c(x_c + s)$$

$$= \frac{1}{2} F(x_c)^T F(x_c) + (J(x_c)^T F(x_c))^T s + \frac{1}{2} s^T (J(x_c)^T J(x_c))s, \qquad (2.3.2.1)$$

because $\hat{m}_c(x_c + s) \geq 0$ for all s and $\hat{m}_c(x_c + s^N) = 0$. Therefore, s^N is a descent direction for $\hat{m}_c$, and since the gradient at x_c of $\hat{m}_c$ and f are the same, it is also a descent direction for f.

2.3.3 The Method of Line Searches for Systems of Nonlinear Equations

From sub section 2.3.1 we have that our global method for (2.3.1.1) will be based on applying our algorithm of section 2.2.4. to the quadratic model $\hat{m}_c(x)$ in (2.3.2.1). Since $\nabla^2 \hat{m}_c(x_c) = J(x_c)^T J(x_c)$, this model is positive definite as long as $J(x_c)$ is nonsingular, which is consistent with the fact that $x_c + s^N$ is the unique root of $M_c(x)$ and thus the unique minimizer of $\hat{m}_c(x)$ in this case. Thus, the model $\hat{m}_c(x)$ has the attractive properties that its minimizer is the Newton point for the original problem, and that all its descent directions are descent directions for f(x) because $\nabla \hat{m}_c(x_c) = \nabla f(x_c)$. Therefore methods based on this model, by going downhill and trying to minimize $\hat{m}_c(x)$, will combine Newton's method for nonlinear equations with global method for an associated

minimization problem. Note that $\hat{m}_c(x)$ is not quite the same as the quadratic model of $f(x) = \frac{1}{2} F(x)^T F(x)$ around x_c,

$$m_c(x + s) = f(x_c) + \nabla f(x_c)^T s + \frac{1}{2} s^T \nabla^2 f(x_c),$$

because $\nabla^2 f(x_c) \neq J(x_c)^T J(x_c)$.

The application of our global method (The method of Line searches) to the nonlinear equations problem is now straightforward. As long as $J(x_c)$ is sufficiently well conditioned, then $J(x_c)^T J(x_c)$ is safely positive definite, and the algorithm (Algorithm 2.2.4.1) apply without change if we define the objective function by $\frac{1}{2}\|F(x)\|^2_2$, the Newton direction by $-J(x)^{-1}F(x)$, and the positive definite quadratic model by (2.3.2.1). Next we consider an example so that we could have a good insight into the application of the method.

Example: Let $F: \Re^2 \to \Re^2$,

$$F(x) = \begin{bmatrix} x_1^2 + x_2^2 - 2 \\ e^{x_1 - 1} + x_2^3 - 2 \end{bmatrix},$$

which has the root $x_* = (1, 1)^T$, and let $x_0 = (2, 0.5)^T$. Define $f(x) = \frac{1}{2} F(x)^T F(x)$. Then

from $J(x) = \begin{bmatrix} 2x_1 & 2x_2 \\ e^{x_1 - 1} & 3x_2^2 \end{bmatrix}$, we get $J(x_0) = \begin{bmatrix} 4 & 1 \\ e & 0.175 \end{bmatrix}$

and together $F(x_0) \cong \begin{bmatrix} 2.25 \\ 0.843 \end{bmatrix}$, gives

$$s_0^N = -J(x_0)^{-1} F(x_0) \cong \begin{bmatrix} -3.00 \\ 9.74 \end{bmatrix}.$$

Our line search algorithm will calculate $x_+ = x_0 + \lambda_0 s_0^N$ starting with $\lambda_0 = 1$, decreasing λ_0 if necessary until $f(x_+) < f(x_0) + \alpha \lambda_0 \nabla f(x_0)^T s_0^N$ where $\alpha = 10^{-4}$.

For $\lambda_0 = x_+ = x_0 + s_0^N \cong \begin{bmatrix} -1.00 \\ 10.24 \end{bmatrix}$, $F(x_+) \cong , \begin{bmatrix} 104 \\ 1071 \end{bmatrix}$,

So that the Newton step clearly is unsatisfactory ($F(x_+)^T F(x_+) > \frac{1}{2} F(x_0)^T F(x_0) + 10^{-4}$ $(J(x_0)^T F(x_0))^T s_0^N$). Therefore we reduce λ_0 by a quadratic backtrack, calculating

$$\lambda_1 = \frac{-\nabla f(x_0)^T s_0^N}{2[f(x_+) - f(x_0) - \nabla f(x_0)^T s_0^N} \qquad (2.3.3.1)$$

In this case, $f(x_+) \cong 5.79 \times 10^5$, $f(x_0) \cong 2.89$, $\nabla f(x_0)^T s_0^N = -F(x_0)^T F(x_0) \cong -5.77$, so that (2.3.3.1) gives $\lambda_1 \cong 4.9 \times 10^{-6}$. Since $\lambda_1 < 0.1$, our algorithm sets $\lambda_1 = 0.5$.

Now $x_+ = x_0 + 0.1 s_0^N \cong \begin{bmatrix} 1.70 \\ 1.47 \end{bmatrix}$; $F(x_+) \cong \begin{bmatrix} 3.06 \\ 3.21 \end{bmatrix}$.

This is still unsatisfactory, and so our algorithm requires us to do a cubic backtrack, where $\lambda_2 = \frac{-b + \sqrt{b^2 - 3a\nabla f(x_0)^T s_0^N}}{3a}$. Notice that $(a, b)^T$ is calculated by the corresponding formula as in case of the unconstrained by the corresponding formula as incase of the unconstrained minimization problem with the Newton direction s_0^N.

One can verify that a backtrack yields $\lambda_2 \cong 0.0659$. Since $\lambda_2 > \frac{1}{2} \lambda_1$, the algorithm sets

$\lambda_2 = \frac{1}{2} \lambda_1 = 0.05$,

$$x_+ = x_0 + 0.05 s_0^N \cong \begin{bmatrix} 1.85 \\ 0.987 \end{bmatrix}, F(x_+) \cong \begin{bmatrix} 2.40 \\ 1.30 \end{bmatrix}.$$

This point is still unsatisfactory, since $f(x_+) \cong 3.71 > f(x_0)$, so the algorithm requires us to do another cubic backtrack, where in this case λprev = 0.1 and λ2prev = 0.05. The corresponding calculation yields $\lambda_3 \cong 0.0116$, which is used since it is in the interval $[\lambda_2/10, \lambda_2/2] = 0.005, 0.025]$.

Now

$$x_+ = x_0 + 0.0116 s_0^N \cong \begin{bmatrix} 1.965 \\ 0.613 \end{bmatrix}, F(x_+) \cong \begin{bmatrix} 2.238 \\ 0.856 \end{bmatrix},$$

This point is satisfactory, since.

$$f(x_+) \cong 2.87 < 2.89 \cong f(x_0) + 10^{-4}\,(0.0116)\,\nabla f(x_0)^T s_0^N ,$$

So we set $x_1 = x_+$ and proceed to the next iteration.

Following in the same fashion (way), first trying the Newton step, the strategy will end up using Newton's method close to the solution.

We conclude the discussion by giving the corresponding modification of our quadratic model $\hat{m}_c(x)$, Whenever J is happened to be nearly singular at the current point x_c. Notice that in case $J(x_c)$ is nearly singular we can not calculate the Newton direction $s^N = -J(x_c)^{-1}F(x_0)$ and the model Hesslan $J(x_c)^T J(x_c)$ is nearly singular.

Thus whenever J is nearly singular at the current point x_c, it is recommended to perturb the quadratic model to

$$\hat{m}_c(x_c + s) = \frac{1}{2}F(x_c)^T F(x_c) + (J(x_c)^T s + \frac{1}{2}s^T H_c s,$$

$(J(x_c)^T F(x_c))^T s$

where.

$$H_c = J(x_c)^T J(x_c) + (n.macheps)^{1/2} \| J(x_c)^T J(x_c)\|_1 . I.$$

REFERENCES

1. Deumlich, R: Optimization and Theory of Approximation textbook, Addis Ababa, 1997.
2. Deumlich, R: Functional Analysis I Textbook, Addis Ababa, 1997.
3. Rudin, W: Functional Analysis Mc-Graw-Hill, New York 1974.
4. Achiezer N: Calculus of Variation 1962.
5. Gelfand /and others/: Calculus 1996.