3.2 Higher Order One-Point Iteration Methods

Let us recall the main fixed-point iteration results from last time.

Theorem 3.1. *Let $g \in C^1[a,b]$, such that $g\big([a,b]\big) \subseteq [a,b]$ and*

$$\lambda \;:=\; \max_{x\in[a,b]} \big|g'(x)\big| \;<\; 1. \tag{3.1}$$

Then:

a) *Function g has a unique fixed point $\alpha \in [a,b]$.*

b) *For any initial choice $x_0 \in [a,b]$, the sequence $x_{n+1} \;=\; g(x_n)$ converges to α, as $n \to \infty$.*

c) $|x_n - \alpha| \;\le\; \lambda^n |x_0 - \alpha| \;\le\; \dfrac{\lambda^n}{1-\lambda}|x_1 - x_0|,\; n \ge 1.$

d)

$$\lim_{n\to\infty} \frac{x_{n+1}-\alpha}{x_n - \alpha} \;=\; g'(\alpha), \tag{3.2}$$

so, if $g'(\alpha) \;\neq\; 0$, the iterative method $x_{n+1} \;=\; g(x_n)$ is linearly convergent to the root α with rate of convergence bounded by λ.

Theorem 3.2. *Assume α is a fixed point of g and that g is continuously differentiable in a neighborhood of α, with*

$$|g'(\alpha)| \;<\; 1. \tag{3.3}$$

Then the conclusions of Theorem 3.1 still hold, provided that x_0 is chosen sufficiently close to α.

So far, there isn't much information in the case $g'(\alpha) = 0$, although the convergence is clearly quite good. Moreover, what happens if the derivatives of g of up to some order are all 0 at α? Can we expect a *faster* convergence? The answer is in the following result.

Theorem 3.3. *Assume α is a fixed point of g and that g is p times continuously differentiable for all x near α, for some $p \ge 2$. Furthermore, assume that*

$$g'(\alpha) \;=\; \ldots \;=\; g^{(p-1)}(\alpha) \;=\; 0,\;\; g^{(p)}(\alpha) \;\neq\; 0. \tag{3.4}$$

Then, if the initial value x_0 is chosen sufficiently close to α, the iteration $x_{n+1} \;=\; g(x_n)$ converges to α with order of convergence p, and

$$\lim_{n\to\infty} \frac{x_{n+1}-\alpha}{(x_n-\alpha)^p} \;=\; \frac{1}{p!} g^{(p)}(\alpha). \tag{3.5}$$

Proof. Since $g'(\alpha) = 0$, by Theorem 3.2, it follows that the iterative method $x_{n+1} = g(x_n)$ converges to α, if x_0 is sufficiently close to α.

For the order of convergence, we use the Taylor series expansion of g around α:

$$x_{n+1} = g(x_n) = g(\alpha) + (x_n - \alpha)g'(\alpha) + \cdots + \frac{(x_n-\alpha)^{p-1}}{(p-1)!}g^{(p-1)}(\alpha) + \frac{(x_n-\alpha)^p}{p!}g^{(p)}(\xi_n),$$

for some ξ_n between x_n and α. Using (3.4) and the fact that $g(\alpha) = \alpha$, we get

$$\begin{aligned} x_{n+1} - \alpha &= \frac{(x_n-\alpha)^p}{p!}g^{(p)}(\xi_n), \\ \frac{x_{n+1}-\alpha}{(x_n-\alpha)^p} &= \frac{1}{p!}g^{(p)}(\xi_n). \end{aligned}$$

Letting $n \to \infty$, both $x_n,\ \xi_n \to \alpha$ and, hence, (3.5) follows.

□

Example 3.4. Recall Newton's iterative method

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)},\ n \geq 0. \tag{3.6}$$

Let us analyze it by this new result.

Solution. We have

$$g(x) = x - \frac{f(x)}{f'(x)},$$

for a simple root of f, α, which means $f(\alpha) = 0$ and $f'(\alpha) \neq 0$. We have

$$g'(x) = 1 - \frac{\big(f'(x)\big)^2 - f(x)f''(x)}{\big(f'(x)\big)^2} = \frac{f(x)f''(x)}{\big(f'(x)\big)^2}.$$

We compute the second derivative, but discard the argument x:

$$g'' = \frac{\big(f'f'' + ff'''\big)\big(f'\big)^2 - 2f'f''\cdot ff''}{\big(f'\big)^4}.$$

Then

$$\begin{aligned} g(\alpha) &= \alpha, \\ g'(\alpha) &= 0, \\ g''(\alpha) &= \frac{\big(f'(\alpha)f''(\alpha)\big)\big(f'(\alpha)\big)^2}{\big(f'(\alpha)\big)^4} = \frac{f''(\alpha)}{f'(\alpha)} \end{aligned}$$

and Theorem 3.3 gives the previously found quadratic convergence ($p = 2$) and error estimate

$$\lim_{n\to\infty} \frac{x_{n+1}-\alpha}{(x_n-\alpha)^2} = \frac{1}{2}g''(\alpha) = \frac{f''(\alpha)}{2f'(\alpha)}.$$

■

Example 3.5. Let us revisit the problem from Example 3.7 (Lecture 11): equation $x^2 - 3 = 0$, with $\alpha = \sqrt{3}$.

Solution. Last time we saw several ways of rewriting the equation in the form $g(x) = x$, some "better" than others, from the convergence point of view.

Let us use Newton's iteration. We have

$$f(x) = x^2 - 3, \;\; f'(x) = 2x,$$

so,

$$g(x) = x - \frac{f(x)}{f'(x)} = x - \frac{x^2-3}{2x} = \frac{1}{2}\Big(2x - x + \frac{3}{x}\Big) = \frac{1}{2}\Big(x + \frac{3}{x}\Big), \;\; g(\alpha) = \alpha,$$

which was one of the methods discussed (part **(c)**). Further, we have

$$\begin{aligned} g'(x) &= \frac{1}{2}\Big(1 - \frac{3}{x^2}\Big), \;\; g'(\alpha) = 0, \\ g''(x) &= \frac{3}{x^3}, \;\; g''(\alpha) = \frac{1}{\sqrt{3}} \neq 0. \end{aligned}$$

So, indeed, the iteration $x_{n+1} = \frac{1}{2}\Big(x_n + \frac{3}{x_n}\Big)$ converges quadratically ($p = 2$) to α.

As a side note, rewriting the equation $x^2 - 3 = 0$ as $x = \frac{1}{2}\Big(x + \frac{3}{x}\Big)$ (which leads to faster convergence) was *not* a "lucky guess", it is actually Newton's method.

■

Example 3.6. Consider the equation

$$x = g(x) = cx(1-x), \tag{3.7}$$

with $c \neq 0$. This is called a *logistic equation* and is of great interest in the *mathematical theory of chaos*. This equation has one nonzero solution, denoted by α_c. For what values of c will the iteration $x_{n+1} = g(x_n)$ converge to α_c (provided that x_0 is chosen sufficiently close to α_c)? Determine the convergence order.

Solution. The nonzero solution of equation (3.7) is given by

$$c(1-x) = 1, \;\; x = \alpha_c = 1 - \frac{1}{c} = \frac{c-1}{c}.$$

We have

$$\begin{aligned} g(x) &= c(x - x^2), \;\; g(\alpha_c) = \alpha_c, \\ g'(x) &= c(1-2x), \;\; g'(\alpha_c) = c\Big(1 - 2 + \frac{2}{c}\Big) = 2 - c, \\ g''(x) &= -2c. \end{aligned}$$

In order to have the iteration $x_{n+1} = g(x_n)$ converge to α_c, we impose the condition

$$|g'(\alpha_c)| < 1 \iff |2-c| < 1 \iff c \in (1,3).$$

So, for any $1 < c < 3$, the method converges (at least) linearly, with rate of convergence $|c-2|$. Now, if $g'(\alpha_c) = 0$, i.e., $c = 2$, then the convergence is quadratic ($p = 2$) and

$$\lim_{n\to\infty} \frac{x_{n+1} - \alpha_c}{(x_n - \alpha_c)^2} = \frac{1}{2} g''(\alpha_c) = \frac{-2c}{2} = -c = -2.$$

■

Example 3.7. Does the iteration

$$x_{n+1} = x_n^5 - 10x_n^3 - 20x_n^2 - 15x_n - 5$$

converge to $\alpha = -1$, provided that x_0 is chosen sufficiently close to α? If so, determine the convergence order and bound the error.

Solution. We have $g(x) = x^5 - 10x^3 - 20x^2 - 15x - 5$ and

$$g(-1) = -1 + 10 - 20 + 15 - 5 = -1,$$

so $\alpha = -1$ is a fixed point of g.

For convergence (and the order of convergence), we compute successively

$$\begin{aligned} g'(x) &= 5x^4 - 30x^2 - 40x - 15, \;\; g'(-1) \;=\; 5 - 30 + 40 - 15 \;=\; 0, \\ g''(x) &= 20x^3 - 60x - 40 \;=\; 20(x^3 - 3x - 2), \;\; g''(-1) \;=\; 20(-1 + 3 - 2) \;=\; 0, \\ g'''(x) &= 20(3x^2 - 3) \;=\; 60(x^2 - 1), \;\; g'''(-1) \;=\; 0, \\ g^{(4)}(x) &= 120x, \;\; g^{(4)}(-1) \;=\; -120 \;\neq\; 0. \end{aligned}$$

So, for x_0 sufficiently close to -1, we have

$$\lim_{n\to\infty} \frac{x_{n+1} + 1}{(x_n + 1)^4} \;=\; \frac{1}{4!} g^{(4)}(-1) \;=\; -\frac{1}{24} \cdot 120 \;=\; -5.$$

Then, it follows that

$$|x_{n+1} + 1| \;\leq\; c_1 |x_n + 1|^4 \;\leq\; ... \;\leq\; c|x_0 + 1|^{4^{n+1}}.$$

So, for $|x_0 + 1| < 1$, the method converges with order of convergence $p = 4$ and the error estimate above.

■

4 Numerical Approximation of Multiple Roots

Definition 4.1. *We say that a function f has a* ***root*** *α* ***of multiplicity*** *$m > 1$ if*

$$f(x) \;=\; (x - \alpha)^m \, h(x), \; h \text{ continuous at } x = \alpha \; \text{ and } h(\alpha) \;\neq\; 0. \tag{4.1}$$

We restrict our discussion to the case where m is a positive integer, although some of our considerations are equally valid for non-integer values. If h is smooth enough at $x = \alpha$, then (4.1) is equivalent to

$$f(\alpha) \;=\; f'(\alpha) \;=\; \ldots \;=\; f^{(m-1)}(\alpha) \;=\; 0, \;\; f^{(m)}(\alpha) \;\neq\; 0. \tag{4.2}$$

There are several challenges in approximating multiple roots.

Uncertainty

When finding a root of any function on a computer, there is always an *interval of uncertainty* about the root, due to measuring/rounding/truncation errors, and this is made worse when the root is multiple.

Example 4.2. Consider evaluating the two functions

$$\begin{aligned} f_1(x) &= x^2 - 3, \\ f_2(x) &= x^2(x^2 - 6) + 9. \end{aligned}$$

Notice that $\alpha = \sqrt{3}$ has multiplicity 1 as a root of f_1 and multiplicity 2 as a root of f_2, since

$$f_2'(x) = 4x(x^2 - 3).$$

Using four-digit decimal arithmetic, we have

$$\begin{aligned} f_1(x) &< 0, \text{ for } x \leq 1.731, \\ f_1(1.732) &= 0, \text{ and} \\ f_1(x) &> 0, \text{ for } x > 1.733, \end{aligned}$$

so $\alpha \in (1.731, 1.733)$. But for f_2,

$$f_2(x) = 0, \text{ for } 1.726 \leq x \leq 1.738,$$

implying that $\alpha \in [1.726, 1.738]$, thus limiting the amount of accuracy that can be attained in finding a root of f_2.

Loss of Precision

Another problem with multiple roots is that the earlier rootfinding methods will not perform as well when the root being sought is multiple. Let us investigate this for Newton's method. We consider Newton's method as a fixed-point iteration

$$x_{n+1} = g(x_n), \;\; g(x) = x - \frac{f(x)}{f'(x)}, \; x \neq \alpha, \;\; f(x) = (x - \alpha)^m \, h(x), \; m > 1.$$

We have

$$f'(x) \;=\; m(x-\alpha)^{m-1}\,h(x) + (x-\alpha)^m\,h'(x) \;=\; (x-\alpha)^{m-1}\big[m\,h(x) + (x-\alpha)\,h'(x)\big],$$

so,

$$\begin{aligned} g(x) &= x - \frac{(x-\alpha)^m\,h(x)}{(x-\alpha)^{m-1}\big[m\,h(x) + (x-\alpha)\,h'(x)\big]} \\ &= x - (x-\alpha)\frac{h(x)}{m\,h(x) + (x-\alpha)\,h'(x)} \;=\; x - (x-\alpha)\varphi(x), \end{aligned}$$

where

$$\varphi(x) \;\stackrel{\text{not}}{=}\; \frac{h(x)}{m\,h(x) + (x-\alpha)\,h'(x)}, \;\; \varphi(\alpha) \;=\; \frac{1}{m}.$$

Then,

$$\begin{aligned} g'(x) &= 1 - \varphi(x) - (x-\alpha)\varphi'(x), \\ g'(\alpha) &= 1 - \varphi(\alpha) \;=\; 1 - \frac{1}{m} \;\neq\; 0. \end{aligned}$$

Thus, in this case, Newton's method converges only *linearly*, with rate of convergence $1 - \dfrac{1}{m} \;<\; 1$.

One way to fix this loss of accuracy would be to change the problem into an equivalent one: instead of solving $f(x) = 0$ which has α as a multiple root, consider the equation

$$u(x) \;:=\; \frac{f(x)}{f'(x)} \;=\; 0, \tag{4.3}$$

for which α is a *simple* root. Then Newton's method is defined by

$$x_{n+1} \;=\; x_n - \frac{u(x_n)}{u'(x_n)}. \tag{4.4}$$

We have

$$u' \;=\; \frac{\big(f'\big)^2 - ff''}{\big(f'\big)^2}, \;\; \frac{u}{u'} \;=\; \frac{\dfrac{f}{f'}}{\dfrac{\big(f'\big)^2 - ff''}{\big(f'\big)^2}} \;=\; \frac{ff'}{\big(f'\big)^2 - ff''},$$

so Newton's method is given by

$$x_{n+1} \;=\; x_n - \frac{f(x_n)f'(x_n)}{\big(f'(x_n)\big)^2 - f(x_n)f''(x_n)},\; n \ge 0. \tag{4.5}$$

Although this method restores the order of convergence $p = 2$, it has several disadvantages: it requires the computation of the second derivative f'', it involves more complex computations than the original method, and the denominator in (4.5) can take very small values, as $x_n \to \alpha$.

A better alternative is to modify the *method*, instead of the *function*.

Newton's Method for Multiple Roots

To improve Newton's method, we would like a function g for which $g'(\alpha) = 0$ (as before), even for multiple roots. Consider the following idea: if α is a root of f, then in the vicinity of α, we have

$$f(x) \;=\; (x-\alpha)^m h(x) \;\approx\; (x-\alpha)^m\, c,$$

for some constant c. Then

$$\begin{aligned} f'(x) \;&\approx\; m(x-\alpha)^{m-1}\, c,\;\; \frac{f(x)}{f'(x)} \;\approx\; \frac{x-\alpha}{m}, \\ x-\alpha \;&\approx\; m\,\frac{f(x)}{f'(x)},\;\; \alpha \;\approx\; x - m\,\frac{f(x)}{f'(x)}. \end{aligned}$$

Thus, we define **Newton's method for multiple roots** by

$$x_{n+1} \;=\; x_n - m\,\frac{f(x_n)}{f'(x_n)}. \tag{4.6}$$

Now, indeed, it is easy to check (by our earlier computations) that for $g(x) \;=\; x - m\,\dfrac{f(x)}{f'(x)}$, we have

$$g(\alpha) \;=\; \alpha,\;\; g'(\alpha) \;=\; 1 - m\,\varphi(\alpha) \;=\; 0,$$

so the method converges quadratically again and

$$\lim_{n\to\infty} \frac{x_{n+1}-\alpha}{(x_n-\alpha)^2} \;=\; \frac{1}{2} g''(\alpha).$$

5 Newton's Method for Nonlinear Systems

Many of the methods considered in the previous sections can be generalized to the multidimensional case, i.e. to *systems* of nonlinear equations. These problems are widespread in applications, and they are varied in form. There is a great variety of methods for the solution of such systems. We only consider the two-dimensional case

$$\begin{aligned} f_1(x_1, x_2) &= 0 \\ f_2(x_1, x_2) &= 0, \end{aligned} \tag{5.1}$$

or, in vector notation,

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}, \ \boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \ \boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{bmatrix}. \tag{5.2}$$

The one-point iteration theory discussed in the previous sections still stands, with appropriate adjustments (norm instead of absolute value and *Jacobian matrix* instead of derivative). Newton's method is derived similarly with the one-dimensional case, considering Taylor series expansions of each $f_i, i = 1, 2$ and expanding $f_i(\boldsymbol{\alpha})$ about $\boldsymbol{x_0} = [x_{1,0} \ \ x_{2,0}]^T$. We get

$$\boldsymbol{x_{n+1}} = \boldsymbol{x_n} - \left(\boldsymbol{J_f}(\boldsymbol{x_n})\right)^{-1} \boldsymbol{f}(\boldsymbol{x_n}), \ \ n \geq 0, \tag{5.3}$$

where $\boldsymbol{J_f}(\boldsymbol{x_n})$ is the Jacobian matrix of $\boldsymbol{f}$ at $\boldsymbol{x_n}$

$$\boldsymbol{J_f}(\boldsymbol{x_n}) = \begin{bmatrix} \dfrac{\partial f_1(\boldsymbol{x})}{\partial x_1} & \dfrac{\partial f_1(\boldsymbol{x})}{\partial x_2} \\ \dfrac{\partial f_2(\boldsymbol{x})}{\partial x_1} & \dfrac{\partial f_2(\boldsymbol{x})}{\partial x_2} \end{bmatrix} \tag{5.4}$$

This is **Newton's method for nonlinear systems**.

In actual practice, we *do not invert* $\boldsymbol{J_f}(\boldsymbol{x_n})$, particularly for systems of more than two equations. Instead we solve a linear system for a correction term to $\boldsymbol{x_n}$:

$$\begin{aligned} \boldsymbol{J_f}(\boldsymbol{x_n})\boldsymbol{\delta_{n+1}} &= -\boldsymbol{f}(\boldsymbol{x_n}), \\ \boldsymbol{x_{n+1}} &= \boldsymbol{x_n} + \boldsymbol{\delta_{n+1}}. \end{aligned} \tag{5.5}$$

This is more efficient in computation time, requiring only about one-third as many operations as inverting $\boldsymbol{J_f}(\boldsymbol{x_n})$.

Example 5.1. Solve the system

$$\begin{aligned} f_1(x_1, x_2) &\equiv 4x_1^2 + x_2^2 - 4 = 0 \\ f_2(x_1, x_2) &\equiv x_1 + x_2 - \sin(x_1 - x_2) = 0. \end{aligned}$$

Solution. There are only two roots, one near $(1, 0)$ and its reflection about the origin near $(-1, 0)$ (see Figure 1).

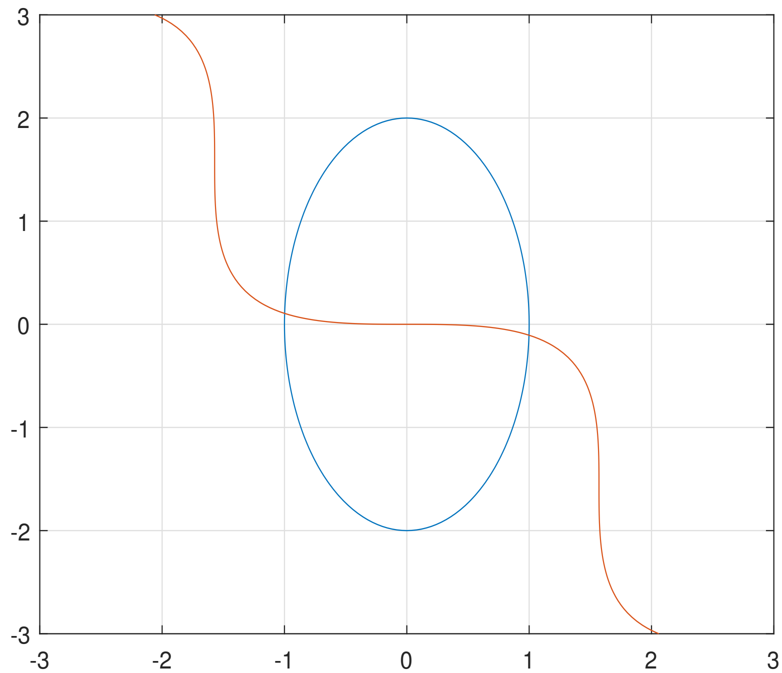

Fig. 1: Example 5.1

Using Newton's method with $\boldsymbol{x_0} = [1 \;\; 0]^T$ we obtain the results in Table 1.

n	$x_{1,n}$	$x_{2,n}$	$f_1(\boldsymbol{x_n})$	$f_2(\boldsymbol{x_n})$
0	1.0	0.0	0.0	$1.59e-1$
1	1.0	-0.1029207154	$1.06e-2$	$4.55e-3$
2	0.9986087598	-0.1055307239	$1.46e-5$	$6.63e-7$
3	0.9986069441	-0.1055304923	$1.32e-11$	$1.87e-12$

Table 1: Newton's Method for Example 5.1

■