

Unit 2: Solving Scalar Equations

Notes prepared by: Amos Ron, Yunpeng Li, Mark Cowlshaw, Steve Wright
Instructor: Steve Wright

1 Introduction

We now discuss algorithms for performing one of the most basic tasks in numerical computing: solving equations in one variable. In general, this problem can be stated as finding x such that $f(x) = 0$, where f is a “smooth” function mapping \mathbb{R} to \mathbb{R} .

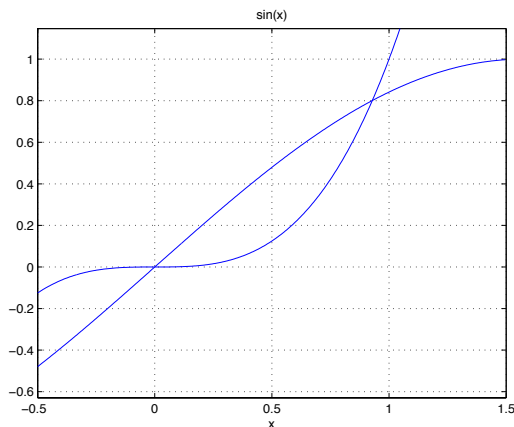


Figure 1: Graphical Solution for $x^3 = \sin x$

Consider the following equation, with solution depicted in Figure 1:

$$x^3 = \sin x, \quad (1)$$

which we can formulate in the general framework $f(x) = 0$ by setting

$$f(x) = x^3 - \sin x. \quad (2)$$

There is no analytical solution to this equation. Instead, we will present an iterative method for finding an approximate solution via a numerical scheme.

1.1 Iterative Method

As in our solution to finding square roots, we would like to find a function g , such that if we input an initial guess at a solution, g will output a better approximation of the actual solution. More

formally, we would like to define a sequence x_0, x_1, x_2, \dots with $x_{i+1} \leftarrow g(x_i)$ such that the sequence $\{x_i\}_{i=0}^\infty$ converges to the solution. We can continue calculating values x_i until we reach a point at which two successive iterates x_j and x_{j+1} are close together.

Clearly, one essential property of g is that $g(r) = r$, where r is the true solution to $f(x) = 0$. A transformation of (1) (e.g. $x = \sin x - x^3 + x$) will yield a $g(x)$ that satisfies this property. Among many possible choices of g , however, the challenge is to choose one for which the sequence $\{x_i\}_{i=0}^\infty$ converges, and converges rapidly. The other important ingredient in setting up the computation is the choice of an initial guess x_0 . Often we have some way of guessing where the root r lies, for example by solving a problem with a simple approximation to f . The use of a good initial guess can improve the robustness of a numerical scheme greatly.

In our discussion and analysis of the general problem $f(x) = 0$, the assumptions we place on f will be very significant. We said about that f should be “smooth,” which is a somewhat nonspecific term. Theory that describes the performance of different methods will make more precise assumptions, such as that f is differentiable, or twice continuously differentiable, or has a bounded second derivative.

2 Fixed Point Iterations

Given an equation of one variable, $f(x) = 0$, we use fixed point iterations as follows:

1. Convert the equation to the form $x = g(x)$, where $g(x)$ is continuous and any r satisfying $r = g(r)$ also satisfies $f(r) = 0$.
2. Start with an initial guess $x_0 \approx r$, where r is the actual solution (root) of the equation.
3. Iterate, using $x_{i+1} := g(x_i)$ for $i = 0, 1, 2, \dots$, stopping when some specified criterion is satisfied.

How well does this process work? We make the following claim:

CLAIM 2.1. *Suppose that g constructed as above is a continuous function. Define $\{x_i\}_{i=1}^\infty$ as in the process above. If this sequence converges, then its limit is a root of $f(x)$.*

To prove the claim, assume that $\{x_i\}_{i=0}^\infty$ converges to some value a . We know from the definition of continuity, that a continuous function evaluated over a convergent sequence also converges. Specifically, since g is a continuous function

$$\lim_{i \rightarrow \infty} x_i = a \Rightarrow \lim_{i \rightarrow \infty} g(x_i) = g(a)$$

Using this fact, we can prove our claim:

$$g(a) = \lim_{i \rightarrow \infty} g(x_i) = \lim_{i \rightarrow \infty} x_{i+1} = a$$

Thus, $g(a) = a$ and, by our assumptions on the construction of g , a is a root of $f(x)$.

The key in setting up a fixed-point iteration is in step 1: finding a transformation of the original equation $f(x) = 0$ to the form $x = g(x)$ so that $\{x_i\}_0^\infty$ converges (rapidly). Using our original example, $f(x) = x^3 - \sin x$, here are some possible choices:

1. $x = \frac{\sin x}{x^2}$
2. $x = \sqrt[3]{\sin x}$
3. $x = \sin^{-1}(x^3)$
4. $x = \frac{\sin x - 1}{x^2 + x + 1} + 1$
5. $x = x - \frac{x^3 - \sin x}{3x^2 - \cos x}$

We can start with $x_0 = 1$, since this is a fairly good approximation to the root, as shown in Figure 1. To choose the best function $g(x)$, we need to determine whether the sequence $\{x_i\}$ converges to a solution, and how fast it converges.

One good way to measure the speed of convergence is to use the ratio of the errors between successive iterations. The error at iteration i can be calculated as:

$$e_i = x_i - r \tag{3}$$

where r is the actual solution. To measure the rate of convergence, we can take the ratio of the error at iteration $i + 1$ to the error at the previous iteration:

$$\nu_{i+1} = \frac{e_{i+1}}{e_i} = \frac{x_{i+1} - r}{x_i - r} \tag{4}$$

However, since we do not know r in advance, we cannot use this measure to control an algorithm (e.g. to decide when to stop). we can however use the following “proxy,” which is a good approximation to ν_i in some circumstances:

$$\mu_{i+1} = \frac{x_{i+1} - x_i}{x_i - x_{i-1}} \tag{5}$$

Note that the magnitude of the error ratio is what is important, so we can safely ignore the sign (sometimes you may see the error or error ratio as an absolute value).

2.1 Order of Convergence

Clearly, we would like the error ratio to be less than 1, or the sequence is not converging. To measure the speed of convergence, we use a concept called the “order of convergence.” The three most interesting orders of convergence are defined here.

Linear Convergence. Linear convergence requires that the error is reduced by at least a constant factor less than 1 at each iteration:

$$|e_{i+1}| \leq c \cdot |e_i| \tag{6}$$

for some fixed constant $c < 1$ and all i sufficiently large. We will study algorithms that converge much more quickly than this, in fact, we have already seen an algorithm (the square root algorithm) that has *quadratic convergence*.

Quadratic Convergence. Quadratic convergence requires that the error at each iteration is proportional to the square of the error on the previous iteration:

$$|e_{i+1}| \leq c \cdot |e_i|^2 \tag{7}$$

for some constant c , which doesn't have to be less than 1. For example, if $c = 10^3$ and $e_i \approx 10^{-4}$, then $e_{i+1} < 10^3 \cdot 10^{-8} = 10^{-5}$, so that significant improvement in the error is still made on this iteration (and further iterations will yield even more rapid convergence toward the solution r).

Superlinear Convergence. Superlinear convergence requires that the ratio of successive errors goes to zero, specifically,

$$\lim_{i \rightarrow \infty} \frac{e_{i+1}}{e_i} = 0.$$

Note that quadratic \Rightarrow superlinear \Rightarrow linear.

It is important to note that equations 6 and 7 provide *bounds* on the convergence rate. It is possible that an algorithm with quadratic convergence will converge more quickly than the bound indicates in certain circumstances, but it will never converge more slowly than the bound indicates. Also note that these bounds are a better indicator of the performance of an algorithm when the errors are small ($e_i \ll 1$).

2.2 Experimental Comparison of Functions for Fixed Point Iterations

Now to return to our problem:

EXAMPLE 2.1. *Solve the equation*

$$x^3 = \sin x \tag{8}$$

How do the functions we considered for $g(x)$ compare? Table 1 shows the results of several iterations using initial value $x_0 = 1$ and four different functions for $g(x)$. Here x_n is the value of x on the n th iteration and μ_n is the error ratio of the n th iteration, as defined in Equation 5.

We can see from the table that when we choose $g(x) = \sqrt[3]{\sin x}$ or $g(x) = x - \frac{\sin x - x^3}{\cos(x) - 3x^2}$ (columns 1 and 4, respectively), the algorithm does converge to 0.92862630873173 with the error ratio μ_n much less than 1. However, when we choose $g(x) = \frac{\sin x}{x^2}$, the error ratio is greater than 1 and the result does not converge. For $g(x) = x + \sin x - x^3$, the error ratio is very close to 1. It appears that the algorithm does converge to the correct value, but very slowly.

Why is there such a disparity in the rate of convergence?

3 Error Analysis of Fixed Point Iterations

For our analysis of the functions g we used for fixed point iterations, we assume that g is differentiable. Note the continued theme: the more we restrict a function, the better our tools for analyzing the function numerically.

We would like to see how g is behaving in the area around the analytic solution, r . For this, we will use the Taylor expansion with remainder. Remember that, for any differentiable function g :

$$g(x) = g(a) + g'(c) \cdot (x - a), \tag{9}$$

where c lies in the (open) interval bounded by a and x . (We don't know what the precise value of c is.)

	$g(x) = \sqrt[3]{\sin x}$	$g(x) = \frac{\sin x}{x^2}$	$g(x) = x + \sin x - x^3$	$g(x) = x - \frac{\sin x - x^3}{\cos x - 3x^2}$
x_1 :	0.94408924124306	0.84147098480790	0.84147098480790	0.93554939065467
μ_1 :	-0.05591075875694	-0.15852901519210	-0.15852901519210	-0.06445060934533
x_2 :	0.93215560685805	1.05303224555943	0.99127188988250	0.92989141894368
μ_2 :	0.21344075183985	-1.33452706115132	-0.94494313796800	0.08778771478600
x_3 :	0.92944074461587	0.78361086350974	0.85395152069647	0.92886679103170
μ_3 :	0.22749668328899	-1.27349109705917	-0.91668584457246	0.18109456255982
x_4 :	0.92881472066057	1.14949345383611	0.98510419085185	0.92867234089417
μ_4 :	0.23059142581182	-1.35803100534498	-0.95508533025939	0.18977634246913
\vdots	\vdots	\vdots	\vdots	\vdots
\vdots	\vdots	\vdots	\vdots	\vdots
x_{26} :	0.92862630873173	-0.00000000000000	0.89462921748990	0.92862630873173
μ_{26} :	0	-1.00000000000000	-0.97525571602895	NaN
x_{27} :	0.92862630873173	0	0.89614104323697	0.92862630873173
μ_{27} :	NaN	-1.00000000000000	-0.97635939022401	NaN

Table 1: Comparison of Functions for Fixed Point Iterations

3.1 Testing for Divergence

Substituting x_n for x and r (the analytic solution) for a , we can use (9) to provide a test for divergence:

$$\begin{aligned}
e_{n+1} &= x_{n+1} - r \\
&= g(x_n) - g(r) \\
&= (x_n - r) \cdot g'(c_n), \quad c_n \in (x_n, r) \\
&= e_n \cdot g'(c_n)
\end{aligned} \tag{10}$$

As x_n approaches r , c_n , since it is between x_n and r is getting closer to r , and, therefore $g'(c_n) \approx g'(r)$. This means that if $|g'(r)| > 1$, the error will get larger, and the sequence x_0, x_1, x_2, \dots will *never* converge.

This suggests a straightforward test to see if a particular function g is a poor choice for fixed point iterations. Simply test the numerical values of g' in the area around the initial approximate solution x_0 (i.e. $g'(x), x \in [x_0 - \delta, x_0 + \delta]$ for some small constant δ). If $|g'(x)| > 1$ on this interval, then the sequence of x_0, x_1, x_2, \dots will not converge and g is a poor choice.

For example, if we look at $g(x) = \frac{\sin x}{x^2}$ for $x \in [.9, 1]$, we see that the absolute value of $g'(x)$ is greater than 1 over the interval. If we look at $g(x) = \sqrt[3]{\sin x}$ on the same interval, its derivative averages around .23. These observations explain why $g(x) = \frac{\sin x}{x^2}$ did not converge on the solution in our experiment and $g(x) = \sqrt[3]{\sin x}$ produced better results.

3.2 Testing for Convergence

We now have a way to test for divergence, but what about testing for convergence? We can use local analysis around the analytic solution r to determine the relevant behavior of g . Define the interval $I = [x_0 - \delta, x_0 + \delta]$ for some small constant δ with the analytic solution $r \in I$. If we can

guarantee that the magnitude of the derivative is less than 1 over I :

$$\forall c \in I \quad |g'(c)| \leq \lambda < 1$$

for some constant λ , then we will have convergence, as long as our values x_i stay in the interval I . We can quantify this with the following two claims:

CLAIM 3.1. *If $\forall c \in I \ 1 > \lambda \geq g'(c) \geq 0$ then the sequence x_0, x_1, x_2, \dots stays in interval I , and will converge to the root r .*

Proof. Consider that, for some $c \in I$, $g(x) = g(r) + (x - r)g'(c)$. If $r < x_0$ then since g has slope that is non-negative and less than 1 and $(x - r) \leq \delta$, $g(x_0) < r + \delta$, which will be in the interval I between x_0 and r . Similarly, if $r > x_0$, $g(x_0) > r - \delta$, which is also in the interval between x_0 and r . The same argument applies to all subsequent values x_i . Since the values all stay in interval I , x_0, x_1, x_2, \dots converges to r . \square

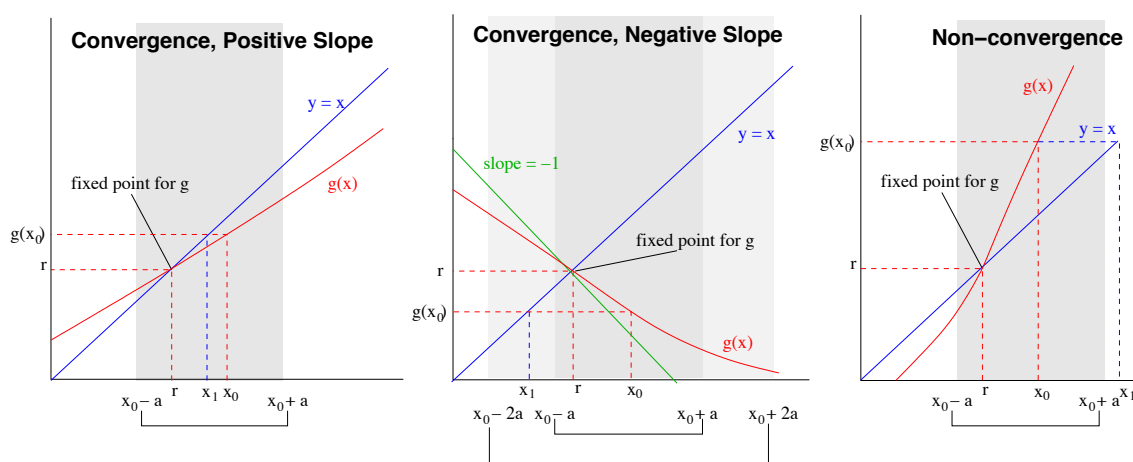


Figure 2: Examples of Convergence for and Divergence for different functions g

CLAIM 3.2. *If $\exists c \in I \ g'(c) < 0$ and $\forall c \in (x_0 - 2\delta, x_0 + 2\delta) \ |f'(c)| \leq \lambda < 1$, then x_0, x_1, x_2, \dots converges to r .*

Proof. Again, if $r < x_0$ then $g(x_0) > r - \delta$, which is in the interval $(x_0 - 2\delta, x_0 + 2\delta)$ since $r \geq x_0 - \delta$ and therefore $g(x_0) > x_0 - 2\delta$. A similar argument works for $r > x_0$. And for subsequent values x_i . Since all values x_i are on the interval $(x_0 - 2\delta, x_0 + 2\delta)$, and since the magnitude of the derivative over this interval is less than 1, the sequence converges to r . \square

Examples of these two situations are depicted in the left and middle diagrams of Figure 2. Furthermore, we can distinguish a bad choice of function g :

CLAIM 3.3 (Non-convergence). *If $|g'(r)| > 1$ then the sequence x_0, x_1, x_2, \dots will never converge to r .*

An example of this possibility is shown in the right-hand diagram of Figure 2

This analysis leads to the following general procedure for using fixed point iterations:

1. Find an interval I that you know contains the solution r to the equation
2. Transform the equation to the form $x = g(x)$. Evaluate $g'(x)$ over the interval I .
3. If $g'(x)$ is small ($\ll 1$) and non-negative over the interval, then use g .
4. If $|g'(x)|$ is small ($\ll 1$) over the interval, but negative somewhere on the interval, define a new interval \tilde{I} that is double the size of the original interval. If $|g'(x)|$ is small ($\ll 1$) over this new interval, then use g .
5. Otherwise, choose a new transformation $x = g(x)$ and begin again.

Note that it is sometimes impossible to identify an interval that contains the solution to the equation. In these cases, you may choose an interval where you know that your function has a small slope, and hope that the sequence x_0, x_1, x_2, \dots converges to a solution. This works in some cases, but there are no guarantees.

Since, ideally, we would like a function g with $\forall_{c \in I} 1 > \lambda \geq |g'(c)|$, what is the ideal value for g' in the vicinity of r ? Intuitively, we would like $g'(r) = 0$. In the next section we will show that this property leads to quadratic convergence.

3.3 Quadratic Convergence of Fixed Point Iterations

Let $x = g(x)$ have solution r with $g'(r) = 0$. Further, assume that g is doubly differentiable on the interval $I = [x_0 - \delta, x_0 + \delta]$, with $r \in I$. Then, from the Taylor expansion of g we know for any $x, a \in I$, there is some c between x and a such that:

$$g(x) = g(a) + g'(a) \cdot (x - a) + g''(c) \frac{(x - a)^2}{2} \quad (11)$$

Using our equation for error and substituting Equation 11 yields:

$$\begin{aligned} e_{n+1} &= x_{n+1} - r \\ &= g(x_n) - g(r) \\ &= g'(r)(x_n - r) + g''(c_n) \overbrace{\frac{(x_n - r)^2}{2}}^{e_n} \quad (\text{some } c_n \in I) \\ &= g''(c_n) \frac{(e_n)^2}{2} \quad (g'(r) = 0) \end{aligned}$$

which indicates *quadratic* convergence. Note that, in order to obtain quadratic convergence, we need two things:

1. An improved algorithm (fixed point iterations with a special function g with the property that $g'(r) = 0$).
2. Improved regularity of g (g must be twice differentiable).

It is a subtle, but important point that these are completely independent restrictions. Constructing a transformation of the original function f to a function g with $g'(r) = 0$ is dependent only on our sophistication in transforming f , it is under our control. However, the regularity of g is something that we have little control over. If g is twice differentiable, it will, almost surely,

inherit this property from function f . If f is a “bad” function (i.e. not differentiable enough), only a miracle will produce a doubly differentiable g . Without such a miracle, g will not be regular enough, and the error analysis which led to quadratic convergence will no longer be valid. In practice, it is very likely that, if the function is not regular enough to analyze in this fashion, quadratic convergence will not occur.

Once again, we see that, even if we use more sophisticated algorithms, we get better performance only on better functions.

4 Newton’s Method

Recall at the end of the last class, we discussed conditions for quadratic convergence of fixed point iterations. The sequence x_0, x_1, x_2, \dots defined using

$$x_{n+1} = g(x_n)$$

converges quadratically to r if $g'(r) = 0$, where r is a fixed point of g (that is, $g(r) = r$). Recall that the error of an iteration e_n is defined as:

$$e_n = x_n - r \tag{12}$$

and quadratic convergence occurs when the error of each iteration is approximately the square of the error in the previous iteration:

$$e_{n+1} \leq \lambda_n e_n^2 \tag{13}$$

where λ_n is defined using the Taylor series remainder:

$$\lambda_n = \frac{g''(c_n)}{2}, \tag{14}$$

and c_n is some value between x_n and r . Note that, unlike the case of linear convergence, λ_n does not have to be small to ensure fast convergence, the square of the previous error will usually dominate.

Recall from unit 1 that we have already seen an example of quadratic convergence in our method for finding numerical solutions to quadratic equations.

EXAMPLE 4.1. *To solve equations of the form*

$$f(x) = x^2 + bx + c = 0, \tag{15}$$

we used the formula

$$x_{k+1} = \frac{x_k^2 - c}{2x_k + b}$$

We can now recognize that this is simply an application of fixed point iterations, where the function f above has been simply transformed into:

$$x = g(x) = \frac{x^2 - c}{2x + b}$$

If we take the derivative of g , we see that:

$$\begin{aligned}
g'(x) &= \frac{(2x+b)(2x) - 2(x^2 - c)}{(2x+b)^2} \\
&= \frac{2x^2 + 2bx + 2c}{(2x+b)^2} \\
&= \frac{2f(x)}{[f'(x)]^2},
\end{aligned}$$

so that $g'(r) = 0$. Without any particular knowledge of the values of the roots of f , g was nevertheless carefully constructed to have derivative 0 at the roots of f .

This function was constructed using a technique called *Newton's Method*. In Newton's method, given a function

$$f(x) = 0$$

We construct the function g as follows:

$$g(x) = x - \frac{f(x)}{f'(x)} \tag{16}$$

For example, remember our method for finding the square root of 5.

EXAMPLE 4.2. *To find the square root of 5, we use the quadratic equation $x^2 = 5$, or:*

$$f(x) = x^2 - 5 = 0$$

Using Newton's method, we construct a function $g(x)$:

$$x = g(x) = x - \frac{f(x)}{f'(x)} = x - \frac{x^2 - 5}{2x} = \frac{x^2 + 5}{2x}$$

Recall that the sequence x_0, x_1, x_2, \dots defined by g converged very quickly to the square root.

4.1 Details of Newton's Method

We must show that Newton's Method produces a valid transformation of $f(x) = 0$ and exhibits quadratic convergence (for most functions) on the solution.

1. The equation $x = g(x)$ defined by Newton's method is equivalent to the original equation $f(x) = 0$.

This is basic algebra, the equation

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

can be easily transformed into $f(x) = 0$, by simply adding x to both sides, then multiplying both sides by $f'(x)$.

2. Newton's Method converges quadratically to the solution r of $f(x) = 0$.

To show this, simply take the derivative

$$\begin{aligned}
 g'(x) &= \frac{d}{dx} \left(x - \frac{f(x)}{f'(x)} \right) \\
 &= 1 - \frac{f'(x) \cdot f'(x) - f(x) \cdot f''(x)}{[f'(x)]^2} \\
 &= 1 - 1 + \frac{f(x) \cdot f''(x)}{[f'(x)]^2} \\
 &= f(x) \cdot \frac{f''(x)}{[f'(x)]^2}
 \end{aligned}$$

and, since $f(r) = 0$, $g'(r) = 0$, which, as we have shown, produces quadratic convergence, provided that $f'(r) \neq 0$.

It is important to note that the above analysis requires the second derivative of g . That is, our error analysis shows that:

$$e_{n+1} = \frac{g''(c_n)}{2} e_n^2 \tag{17}$$

Thus, in order to verify the convergence of x_0, x_1, x_2, \dots to the solution, we must examine the second derivative of g . Since g is constructed using the derivative of f , this analysis requires f to be 3 *times differentiable* in the region near the analytic solution r .

4.2 Geometric Interpretation of Newton's Method

Newton's method uses a simple idea to provide a powerful tool for fixed point analysis. The idea is that we can use tangent lines to approximate the behavior of f near a root. The method goes as follows. We start with a point x_0 , close to a root of f . The line tangent to f at x_0 will intersect the x -axis at some point $(x_1, 0)$. The x -coordinate of this intersection should be closer to the root of f than x_0 was. The process is shown in Figure 4.2.

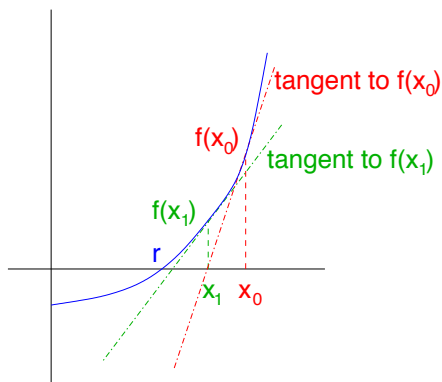


Figure 3: Geometric Interpretation of Newton's Method

This ends our discussion of fixed point iterations. We will now explore some other methods for finding numeric solutions for equations.

5 Bisection

Bisection is a simple method for finding roots that relies on the principle of divide and conquer. The idea is to find an interval containing the root and to split the interval into smaller and smaller sections - in the end, we will have a tiny interval that contains the root, and we may take the midpoint of the interval as our approximation.

To begin, we need to find an interval $[a, b]$ in which the sign of $f(a)$ and $f(b)$ differ. If f is continuous, we know that f must be zero at least once on the interval. An example of this situation is shown in Figure 5.

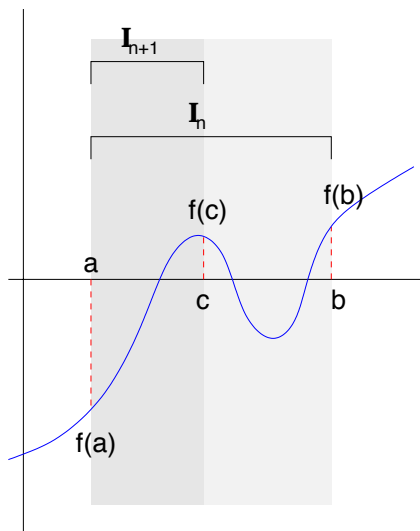


Figure 4: Intervals in the Bisection Method

On each iteration, we calculate the midpoint c of the interval, and examine the sign of $f(c)$. We will then form a new interval with c as an endpoint. If $f(a)$ and $f(c)$ have differing signs, then $[a, c]$ is the new interval, otherwise $[c, b]$ is the new interval. Note that, if $f(c) = 0$, we have just found the root. More formally, the process proceeds as follows:

5.1 The Bisection Process

Given a continuous function f , we will find a root of f ($f(x) = 0$). At the beginning, we need to find an initial interval $I_0 = [a_0, b_0]$ in which $f(a_0)$ and $f(b_0)$ have opposite signs ($f(a) \cdot f(b) < 0$). Then we repeat the following steps for each iteration :

1. Calculate the midpoint c_n :

$$c_n = \frac{a_n + b_n}{2}$$

2. Define the new interval I_{n+1} as:

$$I_{n+1} = \begin{cases} [a_n, c_n] & \text{if } f(a_n) \cdot f(c_n) < 0 \\ [c_n, b_n] & \text{otherwise} \end{cases} \quad (18)$$

When we have iterated to the desired level of accuracy, we take the midpoint of the last interval as our approximation to the root.

5.2 Order of Convergence

Note that, at each iteration, the size of the interval is halved:

$$|I_{n+1}| = \frac{1}{2} \cdot |I_n|$$

Since we will eventually take the midpoint of the interval as our final approximation, the error at any step is at most half the size of the interval, thus, the error is approximately halved at each iteration:

$$e_{n+1} \leq \frac{1}{2} \cdot e_n$$

Thus, bisection has linear convergence. This error measure does not compare precisely to the error measure we used for fixed point iterations, but it is close enough for comparison.

5.3 Comparison with Fixed Point Iterations

Bisection guarantees linear convergence at a rate of $1/2$ for any continuous function and requires only one function evaluation per iteration (we have to evaluate $f(c_n)$, each time, but $f(a_n)$ can be stored). Bisection doesn't even require full function evaluations, it simply requires that we can determine the sign of a function at a particular point.

Why would we use fixed point iterations instead of bisection for a particular function f ?

- If f is triply differentiable near the root, we may be able to use Newton's method, which converges quadratically.
- If f is differentiable near the root, we may be able to find a function $g(x)$ that converges linearly at a rate faster than $1/2$.

Thus, if we are dealing with functions that are not multiply differentiable, or for which we don't have much information, bisection may be a better method for finding roots than fixed point iterations.