Numerical Computing Notes

B. Math(Hons.) 1st year batch

Contents

Lecture 14
February 21, Part A

Finding a root of a function $f : \mathbb{R} \to \mathbb{R}$, i.e., to say a root of the equation

$$f(x) = 0$$

is a common problem in applied mathematics. We will discuss over how to numerically solve the equation for functions with a closed form, along with some other conditions.

Iteration Methods

All the numerical methods which we will use for approximating the root of the equation will be iteration methods. In iteration methods we compute a sequence of increasingly accurate estimate of the root of the equation $f(x) = 0$.

14.1 Bisection Method

Suppose we have a function $f : \mathbb{R} \to \mathbb{R}$, such that

- $f \in \mathscr{C}[a, b]$, i.e., f in continuous on the closed set $[a, b]$.
- $f(a)f(b) < 0$, i.e., $f(a)$ and $f(b)$ have opposite signs.

Then note that from **Intermediate Value Theorem**[1], we have, there exists at least one $\alpha \in [a, b]$ such that $f(\alpha) = 0$. In this setup we can use **bisection method** to find a root to the equation $f(x) = 0$, to our precise degree of accuracy. We will now design an algorithm for the **bisection method**.

14.1.1 Algorithm for Bisection Method

Our function will take 4 inputs:

- The continuous function f.
- Points a and b in the domain of f such that $a < b$, and we have $f(a)f(b) < 0$.
- An *error tolerance* level $\epsilon > 0$. The *error tolerance* level will indicate our function when to stop the iteration process.

The bisection method will consist of the following steps:

```
Step 1. Define $c = \frac{1}{2}(a + b)$.
Step 2. If $b - c \leq \epsilon$, then terminate the iteration process and return $c$ as the root.
Step 3. If $\text{sgn}(f(b)) \cdot \text{sgn}(f(c)) \leq 0$, then set $a = c$, else set $b = c$, and return to Step 1.
```

[1]**Intermediate Value Theorem:** Let $f : [a, b] \to \mathbb{R}$, be a continuous function, and let η be any real number between $f(a)$ and $f(b)$, then there exists a $c \in [a, b]$ such that $f(c) = \eta$.

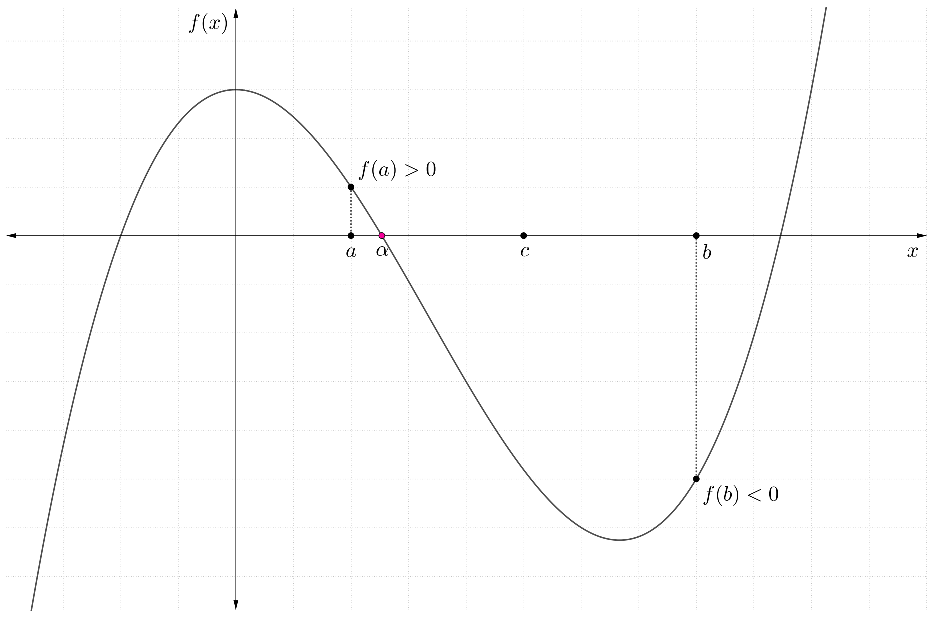

Figure 14.1: Bisection Method

Now, we will show that if f is a continuous function on the set $[a, b]$, where a and b are points such that $f(a)f(b) < 0$, then bisection method is guaranteed to converge to a root of f.

14.1.2 The Bisection Method is guaranteed to converge to a root

Let a_n, b_n and c_n be the n^{th} computed values of a, b and c respectively and we have $a_1 = a, b_1 = b$. And let α be the true root of the function f, i.e.,

$$f(\alpha) = 0$$

Then observe that

$$b_{n+1} - a_{n+1} = \frac{1}{2}(b_n - a_n), \ \forall n \in \mathbb{N} \tag{14.1}$$

and hence from equation (14.1), and using induction we easily get that

$$b_n - a_n = \frac{1}{2^{n-1}}(b - a), \ \forall n \in \mathbb{N} \tag{14.2}$$

Now at the n^{th} iteration, we will have, $c_n = \frac{1}{2}(a_n + b_n)$, and since throughout the process we have either $f(a_n)f(c_n) \leq 0$ or $f(c_n)f(b_n) \leq 0$, thus α either lies in the interval $[a_n, c_n]$ or in the interval $[c_n, b_n]$, in either case, we since

$$c_n - a_n = b_n - c_n = \frac{1}{2}(b_n - a_n)$$

we get that

$$|\alpha - c_n| \leq \frac{1}{2}(b_n - a_n)$$

and then using equation (14.2), we get that

$$|\alpha - c_n| \leq \frac{1}{2^n}(b - a) \tag{14.3}$$

and hence since

$$\lim_{n \to \infty} \frac{1}{2^n}(b - a) = 0$$

we deduce that $c_n \to \alpha$, as $n \to \infty$, hence the **bisection method** guarantees that eventually our estimate will converge to a root of f. Now ofcourse our function can not run for enternity, so we must terminate it at a certain point this is where **Step 2** is necessary. But the next question that arises is how many iterations would we need to reach to our desired root?

14.1.3 How many iterations do we need?

Theorem 14.1.1. Let n be the number of iterations required to a root within our desired error tolerance level $\epsilon > 0$, then we have

$$n \geq \frac{\ln\left(\frac{b-a}{\epsilon}\right)}{\ln 2}$$

where a and b, are the endpoints of our initial interval.

The number iterations required, is equivalent to finding the n such that

$$|\alpha - c_n| \leq \epsilon$$

But from equation (14.3), this is equivalent to finding n such that

$$\frac{1}{2^n}(b-a) \leq \epsilon \Rightarrow \frac{b-a}{\epsilon} \leq 2^n$$

taking logarithm on both sides we get that we must have

$$n \geq \frac{\ln\left(\frac{b-a}{\epsilon}\right)}{\ln 2} \tag{14.4}$$

14.1.4 Pros and Cons of Bisection Method

Pros:

- The number of iterations, i.e., n can be estimated.
- Is guaranteed to converge to a root of the function.

Cons:

- The algorithm converges to a desired root more slowly than other algorithms.
- To we must find a and b such that $\alpha \in [a, b]$, which can be sometimes difficult to find.

Lecture 15
February 21, Part B

15.1 Newton's Method/Newton-Raphson Method

Unlike the Bisection Method, here we don't have to evaluate f to find the appropriate a and b, which is a good thing. However, Newton's Method, as we will see, is not guaranteed to converge, which is a bad thing. We will also see that it depends crucially on the selection of x_0. Moreover, we will need the function to be differentiable in our domain of interest.

15.1.1 Algorithm

First, we make an estimate of the root α of f, which we shall denote by x_0. Consider the equation of the line tangent to the graph of $y = f(x)$ at $(x_0, f(x_0))$

$$p_1(x) = f(x_0) + f'(x_0)(x - x_0).$$

This is just the linear Taylor polynomial for f at x_0.

Define x_1 to be the root of $p_1(x) = 0$. We solve

$$p_1(x_1) = f(x_0) + f'(x_0)(x - x_0) = 0.$$

for x_1 to get

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

Repeating this procedure, we get x_2 from x_1 as

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}.$$

Proceeding inductively, we get a sequence $\{x_n\}$ according to the following recusion formula:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_{n+1})}, n \geq 0 \tag{15.5}$$

The sequence $\{x_n\}$ is a sequence of estimates for α. This procedure of estimating the root α using the recursion formula (15.5) is known as the **Newton-Raphson Method**.

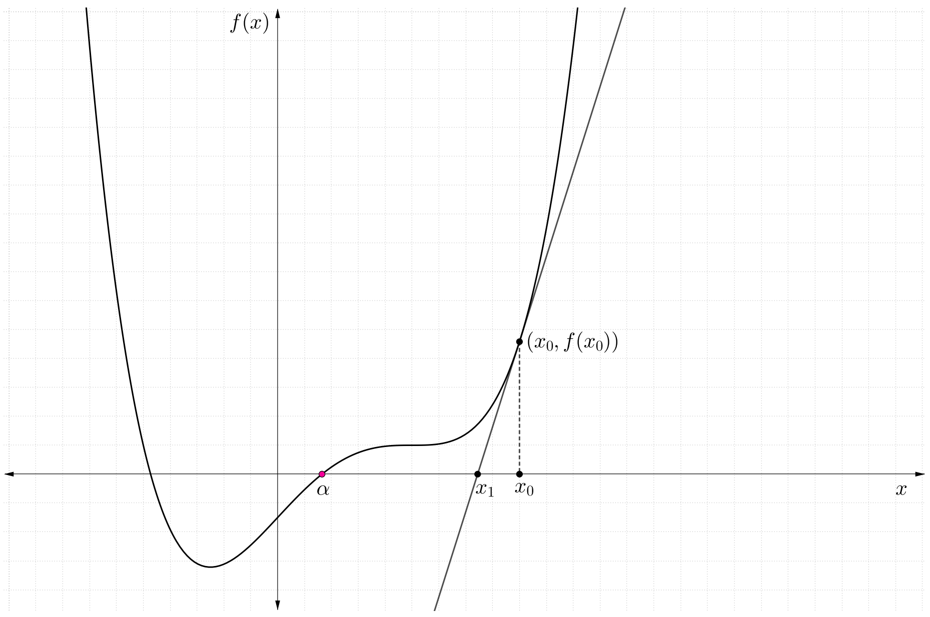

Figure 15.1: Newton Raphson Method

15.1.2 An example computation problem

Now, we take a look at an example computation problem that can be solved using the Newton-Raphson Method.

Given $a, b \in \mathbb{N}$, we want to compute $\frac{a}{b}$ without performing the division operation.

Early computers could support addition, subtraction and multiplication but division needed to be implemented using an algorithm as we are going to discuss. In order to solve this problem, we take the inputs a, b, following which we can get the answer by multiplying a and $\frac{1}{b}$. For this, we need to compute $\frac{1}{b}$, which can be done by solving

$$f(x) = b - \frac{1}{x} = 0 \tag{15.6}$$

Remember that here we are trying to estimate the root $\alpha = \frac{1}{b}$ of the function f defined in (15.6) . We have $f'(x) = \frac{1}{x^2}$. So, by the Newton's Method, the recursion that we get is

$$x_{n+1} = x_n - \frac{b - \frac{1}{x_n}}{\frac{1}{x_n^2}}.$$

which on simplifying gives us

$$x_{n+1} = x_n\left(2 - bx_n\right), n \geq 0 \tag{15.7}$$

Notice that the arithmetic of (15.7) involves only subtraction and multiplication which were supported in the early computers.

Now, we look at the error analysis of this procedure. From $\epsilon_{x_n} = \frac{\alpha - x_n}{\alpha}$, we easily get

$$\epsilon_{x_{n+1}} = \epsilon_{x_n}^2 \implies \epsilon_{x_n} = \left(\epsilon_{x_0}\right)^{2^n} \tag{15.8}$$

According to (15.8), the relative error of x_n,i.e. ϵ_{x_n}, can rapidly decrease to 0, as n increases if we can ensure $|\epsilon_{x_0}| < 1$.

So, we want $|\frac{\alpha - x_0}{\alpha}| < 1$.

- If $x_0 > \alpha$, then $\frac{x_0 - \alpha}{\alpha} < 1 \implies x_0 < 2\alpha = \frac{2}{b}$

- If $x_0 < \alpha$, then $\frac{\alpha - x_0}{\alpha} < 1 \implies x_0 > 0$

This gives us the equivalent condition $0 < x_0 < \frac{2}{b}$ Therefore, the Newton-Raphson Method guarantees convergence to $\alpha = \frac{1}{b}$ if and only if x_0 satisfies the above condition.

At the lowest levels, this procedure is how computation is carried out by some computers even today.

Lecture 16
February 28, Part A

16.1 Error Analysis of the Newton Raphson method

We begin by assuming that we are to use the Newton Raphson method to find a root α of a function f (**which we have a closed form expression for**). We start with an initial guess of α, which we denote by x_0. The iteration used in the Newton Raphson method is

Definition 16.1.1.

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \qquad \forall(n \geq 0)$$

We can calculate $f(x_n)$ and $f'(x_n)$ because we have a closed form expression for f.

Note that if any of the $f'(x_n)$'s are 0, the iteration immediately fails at that stage. This is a flaw present in the Newton Raphson method that the bisection method doesn't suffer from.

In order to carry out the error analysis, we have to make some assumptions which we list below.

- There exists a $\rho > 0$ such that f is continuously differentiable atleast twice in $[\alpha - \rho, \alpha + \rho]$. This assumption must be taken on faith.
- $f'(\alpha) \neq 0$. If this assumption is not true, it is not possible for the iteration to converge to α, as the term $\frac{f(x_n)}{f'(x_n)}$ would blow up to infinity if it did.

Note that the two assumptions listed above imply that $f' \neq 0$ in some neighbourhood of α.

Next, we assume that for some n x_n is sufficiently close to α. Sufficiently close as in sufficiently close for the manipulations that follow. Then, using a Taylor Expansion (we can do that because we have a closed form expression for f), and cutting it off after 3 terms, we can write

$$\begin{aligned}
& f(\alpha) = f(x_n) + (\alpha - x_n)f'(x_n) + \frac{1}{2}(\alpha - x_n)^2 f''(x_n) \\
\implies \quad & 0 \overset{(1)}{=} f(x_n) + (\alpha - x_n)f'(x_n) + \frac{1}{2}(\alpha - x_n)^2 f''(x_n) \\
\implies \quad & 0 \overset{(2)}{=} \frac{f(x_n)}{f'(x_n)} + (\alpha - x_n) + (\alpha - x_n)^2 \frac{f''(x_n)}{2f'(x_n)}
\end{aligned}$$

where equality (1) follows from the fact that α is a root of f, equality (2) is obtained by dividing both sides of the equation by $f'(x_n)$, which we can do since $f'(x_n)$ is close to $f'(\alpha) \neq 0$, which in turns hold because f' is continuous (by assumption) and x_n is sufficiently close to α. Using equation 16.1.1, we obtain

Corollary 16.1.0.1.

$$\alpha - x_{n+1} = (\alpha - x_n)^2 \left[-\frac{f''(x_n)}{2f'(x_n)} \right]$$

Therefore we may say that

$$\text{Error in } x_{n+1} \sim (\text{Error in } x_n)^2$$

where $\sim$ stands for proportionality. (Note that we are assuming that $f'(x_n) \neq 0$). For convergence, $\frac{f''(x_n)}{2f'(x_n)}$ must not be too big.

Next, we run into yet another problem: We cannot compute either side of equation 16.1.0.1, as we don't know the value of α. Which brings us to the next section.

16.2 Bounding some Parameters

Although we don't know the value of α, we are working with an x_n which is "sufficiently close" to it. In that case, since f'' and f' are continuous at $\alpha \in [\alpha - \rho, \alpha + \rho]$, we have

$$M := -\frac{f''(\alpha)}{2f'(\alpha)} \approx -\frac{f''(x_n)}{2f'(x_n)}$$

Rewriting 16.1.0.1, we get

$$\begin{aligned} \alpha - x_{n+1} &\approx (\alpha - x_n)^2 M \\ \implies M(\alpha - x_{n+1}) &\approx [M(\alpha - x_n)]^2 \end{aligned}$$

Now, we carried out this analysis assuming that the iteration converges. If we also assume that iterates do not again stray far from α after they get close to it (this is possible because for convergence only the behaviour in the long run matters), and if we assume that x_0 is sufficiently close to α (previously we'd let x_n be sufficiently close to α for some n whose value was unknown to us), using induction we get

Theorem 16.2.1.

$$M(\alpha - x_n) \approx [M(\alpha - x_0)]^{2^n}$$

Since the iteration converges, it cannot be the case that $|M(\alpha - x_0)| \geq 1$, which implies that

Theorem 16.2.2. Under the conditions required for 16.2.1 to hold,

$$|M(\alpha - x_0)| < 1 \implies |\alpha - x_0| < \frac{1}{|M|}$$

The above theorem implies, among other things, that if $|M|$ is very large, our initial guess x_0 will have to be very small, for the error analysis in theorem 16.2.1 to hold. Intuitively, f should not be "flat" around the root. Newton Raphson will fail often and fail miserably if $f''(\alpha)$ is finite, and $f'(\alpha) = 0$, which happens even when you're dealing with relatively ordinary functions like the trigonometric functions.

The case where $f''(\alpha)$ blows up to infinity doesn't happen as often; for examples look at functions that have exponential-like growth.

16.3 Even more computational problems

We once again ask ourselves - what are the things we can *compute*? A little reflection should reveal to you that we can compute the following and just that:

- We have a closed form expression for f, and hence
- We know the values of $x_0, x_1, \ldots$.
- We know the values of $f(x_0), f(x_1), \ldots$.
- We know the values of $f'(x_0), f'(x_1), \ldots$.
- We know the values of $f''(x_0), f''(x_1), \ldots$.

Now, using the mean value theorem and the fact that $f(\alpha) = 0$, if x_n is sufficiently close to α, we have

$$f(x_n) = f(x_n) - f(\alpha) = f'(\xi_n)(x_n - \alpha)$$

for some $\xi_n \in (\alpha, x_n)$. The last equation can be written as

$$\alpha - x_n = -\frac{f(x_n)}{f'(\xi_n)}$$

If $-\frac{f(x_n)}{f'(\xi_n)} \approx -\frac{f(x_n)}{f'(x_n)}$, then from definition 16.1.1, we obtain

$$\alpha - x_n \approx x_{n+1} - x_n$$

One way to use this is to find such a ξ_n, in which case you can guess where α lies relative to x_n, use the above equation for error analysis, etc.

We end by noting that although the bisection method always works (provided you can bracket the roots), but Newton Raphson can fail. However when it does work, Newton Raphson converges to the root much faster than the bisection method.

16.4 The Secant method

We wish to find a root α of a function f. Start with ***two*** initial guesses of α namely x_0 and x_1. Ideally, the guesses should bracket α. Then, compute subsequent guesses using the following iteration

Definition 16.4.1.

$$x_{n+1} = x_n - f(x_n)\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} \qquad \forall (n \geq 1)$$

Note the similarity between the Newton Raphson method and the Secant method! (The derivative in the Newton Raphson method is replaced with a discrete version of it)

Unfortunately, the Secant method converges more slowly than the Newton Raphson method, and thus finds little practical use.

It is possible for the guesses to alternate between a few values, and not converge.

Also, just because the initial guesses bracket the root, that doesn't mean the subsequent guesses will too.

Normally, when we compute x_{n+1}, we discard x_{n-1}. But you can instead discard x_n if you need to, with the goal of having guesses that bracket the root. Although even an uninterrupted bracket fails when the expressions of the form $f(x_n) - f(x_{n-1})$ are too large, and thus the brackets keep getting larger and larger instead of smaller.

For an example of what can go wrong with the secant method consider the case of a vertical parabola with vertex at 0, and initial guesses ± 1.

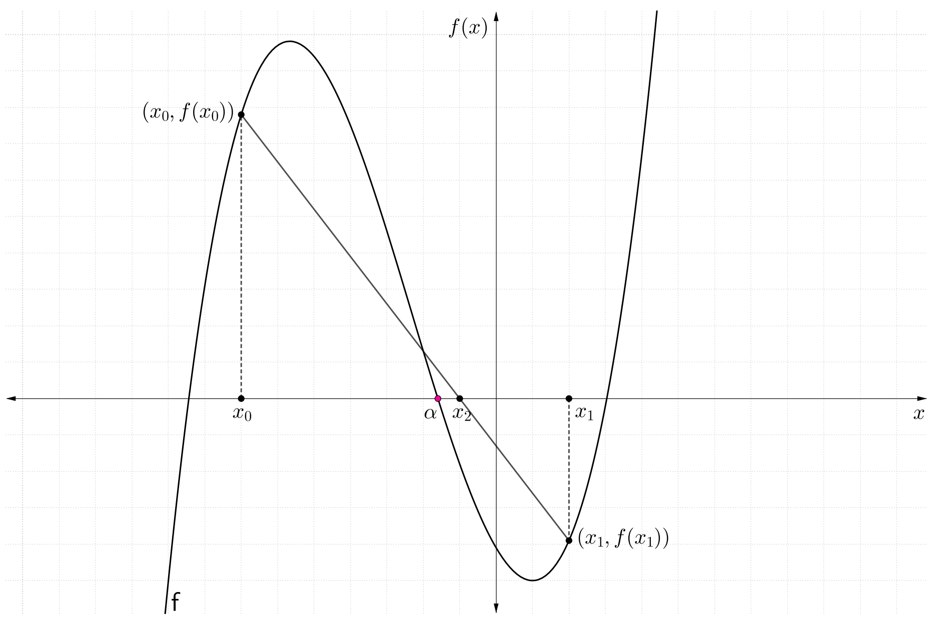

Figure 16.2: Secant Method

It is not clear to me whether the secant method will converge if our initial guesses bracket the root, we always choose our subsequent guesses so that they always bracket the root, and f has different signs at our guesses (including the initial guesses). Someone should probably investigate.

Lecture 17
February 28, Part B

17.1 Fixed Point Method

The **Newton Raphson method** and **secant method** are *one-point* and *two-point* iteration methods respectively (an algorithm is said to be n-point iteration method, if for the iteration to begin we need n initial guessed points). Now we will go over more general theory of *one-point* iteration methods. Conisder the simple equation

$$x^2 - 5 = 0 \tag{17.1}$$

then a root of the equation (17.1), is $\alpha = \sqrt{5}$, and consider the following iterative methods for solving the above equation:

$$x_{n+1} = 5 + x_n - x_n^2 \tag{17.2}$$
$$x_{n+1} = \frac{5}{x_n} \tag{17.3}$$
$$x_{n+1} = 1 + x_n - \frac{1}{5}x_n^2 \tag{17.4}$$
$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{5}{x_n}\right) \tag{17.5}$$

Then its easy to observe that for all the above sequences whenever the sequence $\{x_n\}_{n\in\mathbb{N}}$ converges to some real number α, then $\alpha = \sqrt{5}$. This can be easily seen by assuming $\{x_n\}_{n\in\mathbb{N}}$ converges to α, and then taking limit as n tends to infinity. For example in equation (17.2) (assume that for some initial guess the sequence converges), then we have

$$\alpha = \lim_{n\to\infty} x_{n+1} = \lim_{n\to\infty} \left(5 + x_n - x_n^2\right) = 5 + \alpha - \alpha^2 \Rightarrow \alpha = \sqrt{5}$$

Now observe that all the above iterative equations, have the general form

$$x_{n+1} = g(x_n) \tag{17.6}$$

for some appropriate continuous function on a suitable domain. For example in case of equation (17.2), the function $g(x) = 5 + x - x^2$. And assuming that x_n converges to α, we get that

$$\alpha = \lim_{n\to\infty} x_{n+1} = \lim_{n\to\infty} g(x_n) = g\left(\lim_{n\to\infty} x_n\right) = g(\alpha)$$

Thus α is a solution to the equation $g(x) = x$, and hence we have α is a root of g.

The above idea motivates us to see that the problem of finding a root for the equation $f(x) = 0$, can be converted into an equation $g(x) = x$, where we can take $g(x) = x - f(x)$. Thus if α is a root of f, i.e., $f(\alpha) = 0$, then we have

$$g(\alpha) = \alpha - f(\alpha) = \alpha$$

Now as we will see, if we are given some further informations about the functions we are working with, then **fixed point iteration methods** are faster than **bisection method** and are even guaranteed to converge to a root.

Remarks: *Fixed point iteration method has a lot of applications in Chaos Theory.*

Now let us look at some of the necessary conditions, we may need for the **fixed point method** to work!

The first question that arises naturally is, when does the equation $g(x) = x$ has a solution?

17.1.1 Necessary Condition for Existence of a Fixed Point

Theorem 17.1.1. Let $g : [a, b] \to \mathbb{R}$, be a continuous function, and suppose g satisfies the property

$$a \leq x \leq b \Rightarrow a \leq g(x) \leq b$$

then the equation $x = g(x)$, has at least one solution α in the interval $[a, b]$.

Proof. Define $f(x) = x - g(x)$, then we have f is a continuous function on $[a, b]$, and we further have

$$f(a) = a - g(a) \leq 0 \qquad \text{and} \qquad f(b) = b - g(b) \geq 0$$

and hence by **Intermediate Value Theorem**, we get that there exists a $\alpha \in [a, b]$, such that $f(\alpha) = 0$, but then we get $g(\alpha) = \alpha$. ■

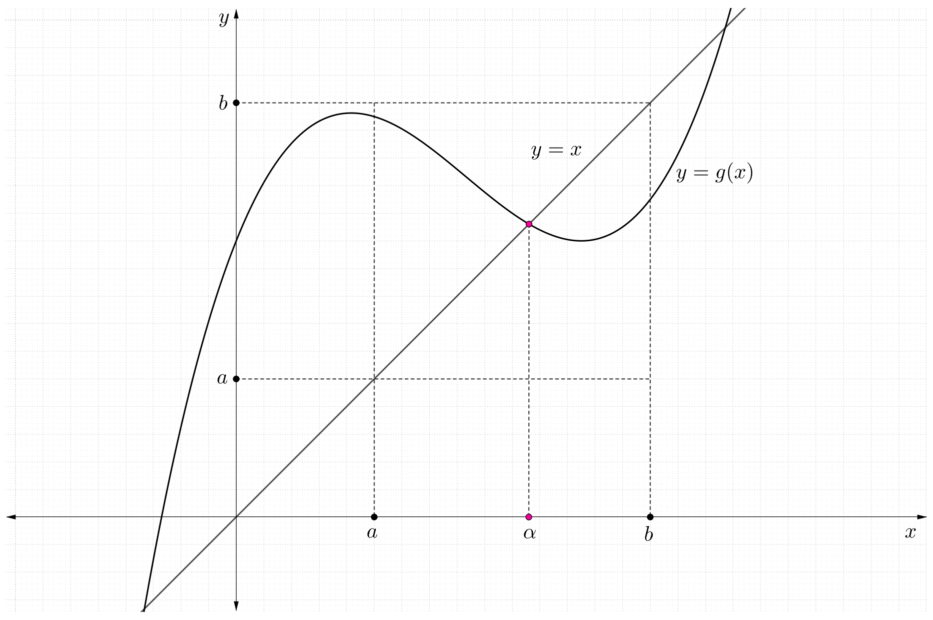

Figure 17.3: geometrical interpretation of **Theorem** 17.1.1

Theorem 17.1.1, can be geometrically interpretated as if we have a function $g : [a, b] \to [a, b]$, i.e., the graph of g is inside the square region $[a, b] \times [a, b]$, then the graph of g must intersect the diagonal of the square, i.e., $y = x$ line at some point, which precisely gives us that there exists a $\alpha \in [a, b]$, such that $g(\alpha) = \alpha$.

Definition 17.1.1. Now consider the following assumptions:
- g is differentiable on $[a, b]$, and further g' is continuous on $[a, b]$.
- $x \in [a, b] \Rightarrow g(x) \in [a, b]$.
- $\lambda := \max\limits_{a \leq x \leq b} |g'(x)| < 1$.

Then with the above assumptions we can guarantee, that there exists an unique root.

17.1.2 Uniqueness of the Fixed Point

Theorem 17.1.2. Assume that g satisfies all the above conditions given in **definition** 17.1.1, then g has an unique fixed point in $[a, b]$.

Proof. Now the fact that g is differentiable on $[a, b]$, tells us that g is continuous on $[a, b]$, and then second condition of our assumptions, along with **theorem** 17.1.1, gives us g has at least one fixed point in $[a, b]$.

Let $w_1, w_2 \in [a, b]$ then from **Mean Value Theorem**[1], we get that there exists a c in between w_1 and w_2 such that

$$g(w_1) - g(w_2) = g'(c)(w_1 - w_2)$$

But then we get that

$$|g(w_1) - g(w_2)| = |g'(c)||w_1 - w_2| \leq \lambda |w_1 - w_2| \tag{17.7}$$

Now suppose there exists $c_1, c_2 \in [a, b]$ such that $g(c_1) = c_1$ and $g(c_2) = c_2$, then we have

$$\begin{aligned} |c_1 - c_2| &= |g(c_1) - g(c_2)| \\ &\overset{(17.7)}{\leq} \lambda |c_1 - c_2| \end{aligned}$$

and hence we get that

$$(1 - \lambda)|c_1 - c_2| \leq 0 \tag{17.8}$$

But note that from the third condition in our assumptions, we have $1 - \lambda > 0$, and hence only way equation (17.8), can hold is

$$|c_1 - c_2| \leq 0 \Rightarrow |c_1 - c_2| = 0 \Rightarrow c_1 = c_2$$

Hence, g has an unique fixed point in $[a.b]$. ■

[1]**Mean Value Theorem:** Let $f : [a, b] \to \mathbb{R}$ be differentiable on (a, b), then there exists a $c \in (a, b)$ such that

$$f(b) - f(a) = f'(c)(b - a)$$

Lecture 18
March 4

18.1 Fixed point methods

Instead of trying to find a root to an equation of the form $f(x) = 0$, we can try to find a root to an equation of the form $x - f(x) = x$. Defining $g(x) := x - f(x)$, we see that the task of finding a root of f is equivalent to finding a **fixed point** of g (a fixed point of a function g is a real α such that $g(\alpha) = \alpha$).

We shall deal mainly with non-linear functions, as finding the roots/fixed points of linear functions can be done by employing the techniques of linear algebra, which we have already covered.

Therefore, we now set for ourselves the task of finding fixed points α of a function g. But first, we recall some theorems that we proved in previous lectures

Theorem 18.1.1.
If the following conditions obtain,
- $g\colon [a,b] \to [a,b]$.
- g is continuous on $[a,b]$.

then g has a (not necessarily unique) fixed point in $[a,b]$. ∎

Theorem 18.1.2.
If the following conditions obtain,
- $g\colon [a,b] \to [a,b]$.
- g is continuous on $[a,b]$.
- g' exists in $[a,b]$.
- g' is continuous on $[a,b]$.
- $\lambda := \max_{x\in[a,b]} |g'(x)| < 1$.

then g has a **unique** fixed point in $[a,b]$. ∎

Note that we have the same problem here as when we did the bisection method; the problem of finding a suitable interval ($[a,b]$ in this case). But let us assume that you have found such an interval, and proceed.

Under the assumptions of theorem 18.1.2, if we start with an initial guess $x_0 \in [a,b]$, and define subsequent guesses using the recursion

Definition 18.1.1.
$$x_{n+1} = g(x_n) \qquad \forall(n \geq 0)$$

then if we denote the unique fixed point of g in $[a,b]$ by α, we have

Theorem 18.1.3.
$$|\alpha - x_n| \le \lambda^n |\alpha - x_0|$$

Proof. Assuming the preconditions of theorem 18.1.2, $g\colon [a,b] \to [a,b]$. Combining that with the fact that $x_0 \in [a,b]$, it is easy to see by induction that $x_n \in [a,b]$ for all $n \ge 0$.

Now, by definition 18.1.1, and the fact that $g(\alpha) = \alpha$, we have

$$\alpha - x_{n+1} = g(\alpha) - g(x_n) = g'(c_n)(\alpha - x_n)$$

for some $c_n \in (\alpha, x_n)$, by the *mean value theorem*. Since by the preconditions of theorem 18.1.2 $\lambda := \max_{x\in[a,b]} |g'(x)| < 1$, we obtain from the above equation

$$|\alpha - x_{n+1}| \le \lambda |\alpha - x_n|$$

By induction, we get

$$|\alpha - x_n| \le \lambda^n |\alpha - x_0|$$

■

Corollary 18.1.3.1. Under the assumptions of theorem 18.1.2,

$$\lim_{n\to\infty} x_n = \alpha$$

Proof. By theorem 18.1.3, $|\alpha - x_n| \le \lambda^n |\alpha - x_0|$. Since $\lambda < 1$, we get

$$\lim_{n\to\infty} |\alpha - x_n| = 0$$

using the squeeze theorem (the L.H.S. of the squeeze is $0 \le |\alpha - x_n|$, which follows from properties of the $|\cdot|$ function). ■

Corollary 18.1.3.2. Under the assumptions of theorem 18.1.2,

$$|\alpha - x_n| \le \frac{\lambda^n}{1-\lambda} |x_0 - x_1|$$

Proof. Note that

$$|\alpha - x_0| \overset{(1)}{\le} |\alpha - x_1| + |x_0 - x_1| \overset{(2)}{\le} \lambda|\alpha - x_0| + |x_0 - x_1|$$

$$\implies (1-\lambda)|\alpha - x_0| \le |x_0 - x_1|$$

$$\overset{(3)}{\implies} |\alpha - x_0| \le \frac{|x_0 - x_1|}{1-\lambda} \tag{4}$$

where inequality (1) is the triangle inequality, and inequality 2 follows from theorem 18.1.3. Implication (3) is justified as $\lambda < 1$ and therefore we are not dividing by 0. But then

$$|\alpha - x_n| \overset{(5)}{\le} \lambda^n |\alpha - x_0| \overset{(6)}{\le} \frac{\lambda^n}{1-\lambda} |x_0 - x_1|$$

where inequality (5) follows from theorem 18.1.3, and inequality (6) follows from inequality (4) above.

■

Now we come to what is probably the most important theorem in this lecture.

Theorem 18.1.4. Under the assumptions of theorem 18.1.2,

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

Proof. Note that by the mean value theorem, for all $n \geq 0$ we have $\alpha - x_{n+1} = g(\alpha) - g(x_n) = g'(c_n)(\alpha - x_n)$ for some $c_n \in (\alpha, x_n)$. Therefore,

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

But since $x_n \to \alpha$, $c_n \to \alpha$ too, and then using the fact that g' is continuous in $[a, b]$, we get

$$\lim_{n\to\infty} g'(c_n) = g'(\alpha)$$

∎

Definition 18.1.2. Suppose we have a sequence $\{y_n\}$ that converges to β. We say that $\{y_n\}$ converges to β **linearly**, if for all n

$$\beta - y_{n+1} \approx c(\beta - y_n)^p$$

with $p = 1$.
If $p > 1$, we say that the sequence $\{y_n\}$ converges to β **super-linearly**.

We are now ready to state the final theorem of this lecture.

Theorem 18.1.5. If we assume the preconditions of 18.1.2, and we additionally assume that $g'(\alpha) \neq 0$, x_n converges to α linearly.

Proof. Theorem 18.1.4 tells us that $\lim_{n\to\infty}\frac{\alpha - x_{n+1}}{\alpha - x_n} = g'(\alpha)$. That implies, as long as $g'(\alpha) \neq 0$ and therefore higher order terms do not dominate, that for large n,

$$\alpha - x_{n+1} \approx g'(\alpha)(\alpha - x_n)$$

(The convergence is guaranteed both by theorem 18.1.1 and by the fact that $g'(\alpha) < 1$, which follows from the preconditions of 18.1.2). ∎

We end by remarking that fixed point iterations are very easy to program owing to the simple nature of their iterations.

[Stopped at 37:36 $\pm$ a few seconds. The remainder of the lecture is about polynomial interpolation.]

Lecture 20
March 7, Part B

4.1 Introduction

In numerical analysis, we are often provided with a table of values of a function f(x) with respect to some argument x. Examples might be log tables, tables for various statistical distributions or non-analytic integrals etc. This approach, although extremely convenient for practical purposes, has its limitations. The most obvious one being that a table can only contain finitely many values of a function, from which we need to estimate the intermediate values. This is where interpolation becomes invaluable.

Interpolation basically deals with approximating a function given its value at some support points. It is used, as the name suggests, to approximate $f(x^*)$, where $x*$ lies between two of the given support points.