# 22 Lecture #15: Tuesday, April 7th, 2026

## 22.1 The fixed-point method

Once again, we face algebraic equations that cannot be solved analytically. For this reason, we turn again to Numerical Analysis. Up to this point, our approach has been to rewrite an equation such as $\cos(x) = x$ by defining the function $f(x) = \cos(x) - x$, and then to approximate a root of $f$. We will now change our point of view.

When we look at the equation $\cos(x) = x$, we are seeking a solution of an equation of the form $\varphi(x) = x$. In other words, we are looking for a point $x$ such that the function $\varphi$ maps $x$ to itself; that is, $x$ remains fixed under $\varphi$. We now introduce the formal definition. See Figure 100 for a possible geometric representation of what a fixed point means.

**Definition 22.1.** Let $\varphi$ be a function. A **fixed point** of $\varphi$ is any number $x_f$ satisfying $\varphi(x_f) = x_f$.

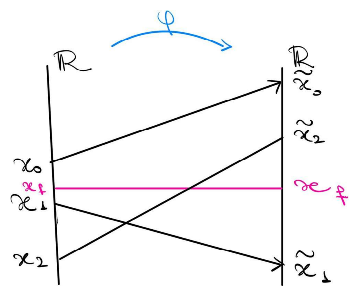


Figure 100: A possible interpretation of a fixed point.

**Example 22.2.** Consider the function $\varphi(x) = x^2$. Notice, for instance, that $\varphi(2) = \varphi(-2) = 4$, as shown in Figure 101. Are there any numbers such that $\varphi(x) = x$? Yes, of course. One of them is 0 and the other is 1. That is, $\varphi(0) = 0$ and $\varphi(1) = 1$, so $\varphi$ has two fixed points.

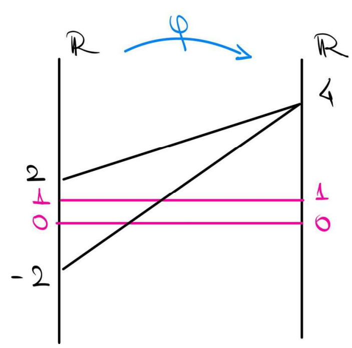


Figure 101: The function $\varphi(x) = x^2$.

There is another function $\varphi$ that has the same fixed points, as we see below.

**Example 22.3.** Consider now $\varphi(x) = \sqrt{x}$. In this case, the domain of $\varphi$ is different, but once again we still have two fixed points, namely 0 and 1. See Figure 102.

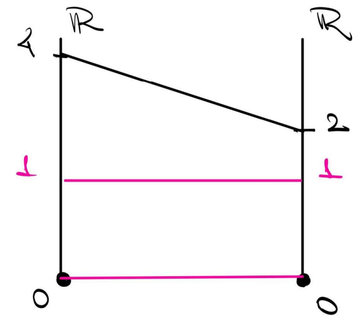


Figure 102: The function $\varphi(x) = \sqrt{x}$.

Let us now look at a rather different example.

**Example 22.4.** Consider the operator that takes a function $f$ and maps it to its derivative. That is, $\varphi[f] = f'$. Here we take the domain of $\varphi$ to be $\mathcal{C}^\infty(I)$, the set of all functions that are infinitely differentiable on an interval $I$. Are there any functions that remain unchanged when differentiated? Yes, there are. For instance, the function $e^x$ and the function identically equal to 0 are examples.

Of course, there are also functions $\varphi$ that do not have any fixed points. For instance, consider the function $\varphi(x) = x + 13$. Indeed, if there were, then there would be $x$ such that $\varphi(x) = x$ and this would mean that $x + 13 = x$, which is impossible as this would imply that $0 = 13$.

Let us now state the following fact which gives us several situations where we can guaratee the existence of fixed points.

**Fact 22.5.** If a function $\varphi$ is continuous on some closed bounded interval $I$ and $\varphi[I] \subseteq I$, then $\varphi$ has a fixed point $x_f \in I$.

Before proving this result, let us give a geometric interpretation of a fixed point. See Figure 103. A fixed point is precisely the intersection between the graph of the function $\varphi$ and the diagonal line $y = x$, as shown in the figure.

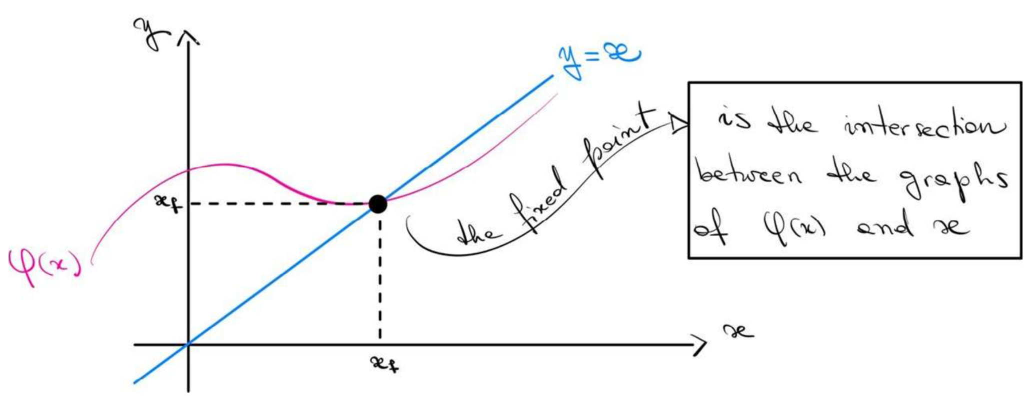


Figure 103: Geometric interpretation of a fixed point.

*Proof of Fact 22.5.* Figure 104 illustrates the geometric idea behind the result. We encourage the reader to examine it and observe that the conclusion of this result follows naturally from the picture. For the analytic proof, let us define the function $h(x) := \varphi(x) - x$ which is continuous on $[a, b]$, where $a, b \in \mathbb{R}$. Since the image of $\varphi$ is contained in $[a, b]$, we have $\varphi(a) \geqslant a$. Therefore, $h(a) = \varphi(a) - a \geqslant 0$. Similarly, since $\varphi(b) \leqslant b$, we obtain $h(b) = \varphi(b) - b \leqslant 0$. Thus, $h$ is continuous on $[a, b]$ and changes sign on this interval. By the Intermediate Value Theorem, there exists $x_f \in [a, b]$ such that $h(x_f) = 0$. This means that $\varphi(x_f) = x_f$. Hence, $\varphi$ has a fixed point.

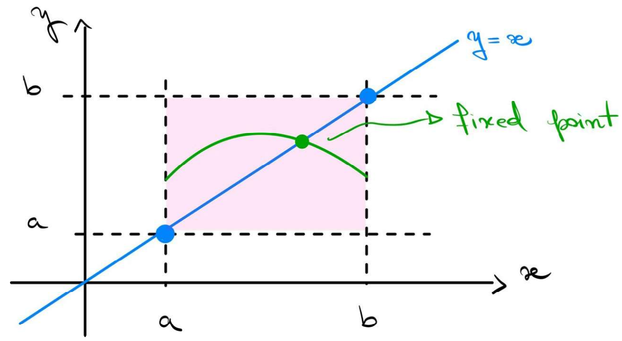


Figure 104: Geometric interpretation of Fact 22.5.

□

The previous fact shows that in many situations fixed points do exist. However, how can we actually find them? Let us consider the following idea. Start with an initial guess $x_0$. Evaluate it under $\varphi$ to obtain $x_1 = \varphi(x_0)$. If $x_1$ is a fixed point, then we are done. If not, we continue and define

$$x_2 = \varphi(x_1) = \varphi(\varphi(x_0)),$$

and so on. Let us return now to Examples 22.2 and 22.3. Suppose that we have $\varphi(x) = x^2$ and our initial guess is $x_0 = \frac{1}{2}$. Then, $x_1 = \varphi(x_0) = \frac{1}{4}$. We apply $\varphi$ again and find $x_2 = \frac{1}{16}$. This gives the sequence

$$\frac{1}{2}, \frac{1}{4}, \frac{1}{16}, \dots$$

which converges to 0, a fixed point of $\varphi$.

Now, if $\varphi(x) = \sqrt{x}$ and the initial guess is $x_0 = \frac{1}{2}$, then we obtain the sequence

$$\frac{1}{2}, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{\sqrt{2}}}, \dots$$

and this sequence converges to 1. Indeed, To prove that this sequence converges to 1, let us write it in a more convenient form. Define $x_n = \left(\frac{1}{2}\right)^{1/2^n}$. We want to show that $\lim_{n\to\infty} x_n = 1$. Since $2^n \to \infty$, it follows that $\frac{1}{2^n} \to 0$. Therefore, $x_n = \left(\frac{1}{2}\right)^{1/2^n} \to \left(\frac{1}{2}\right)^0 = 1$ because the function $a^x$ is continuous for every $a > 0$. Hence, the sequence converges to 1, which is also a fixed point of $\varphi$.

This procedure is called fixed point iteration and the general formula is given by $x_{k+1} = \varphi(x_k)$.

**Algorithm 22.6** (fixed-point iteration)**.** Given a continuous function $\varphi$ and tolerance $\varepsilon$.

**0.** Choose $x_0$.

**1.** Let $x_{k+1} = \varphi(x_k)$.

If $|x_{k+1} - x_k| < \varepsilon$, then the algorithm stops and the output is $x_{k+1}$. Otherwise, increase $k$ by one and go back to step 1.

How about Example 22.4? Let us start the iteration with the function $x^3 + 13$. Then,

$$\varphi[x^3 + 13] = 3x^2, \qquad \varphi[3x^2] = 6x, \qquad \varphi[6x] = 6, \qquad \varphi[6] = 0, \qquad \varphi[0] = 0,$$

and from here one we will keep getting the null function all the time. Therefore, the sequence converges to 0, which is a fixed point of $\varphi$.

**Example 22.7.** Notice that depending on the initial point we consider, we might not get convergence to the fixed point. For instance, if we pick the initial guess $x_0 = 2$ for the function $\varphi(x) = x^2$, then we obtain the sequence $2, 4, 16, 256, \ldots$ which does not converge to a fixed point.

## 22.2 Convergence results

Example 22.7 motivates us to find conditions under which our method converges.

**Theorem 22.8.** Let $\varphi$ be a function, let $x_0 \in \mathbb{R}$, and define $x_{k+1} = \varphi(x_k)$ for $k \in \mathbb{N}$. If $x_k \to x_f$ and $\varphi$ is continuous at $x_f$, then $x_f$ is a fixed point.

*Proof.* Since $x_{k+1} = \varphi(x_k)$, we may take limits on both sides to obtain

$$\lim_{k\to\infty} x_{k+1} = \lim_{k\to\infty} \varphi(x_k).$$

Because $\lim_{k\to\infty} x_{k+1} = \lim_{k\to\infty} x_k$ and $\varphi$ is continuous at $x_f$, it follows that

$$\lim_{k\to\infty} x_k = \varphi\left(\lim_{k\to\infty} x_k\right).$$

Since $\lim_{k\to\infty} x_k = x_f$, we conclude that $x_f = \varphi(x_f)$. Therefore, $x_f$ is a fixed point of $\varphi$, as we wanted to prove. $\square$

Notice that we can apply Theorem 22.8 to the function $\varphi(x) = \cos(x)$. Since $\cos(x)$ is continuous, it follows that if the iteration converges, then its limit must be a fixed point of $\varphi$, that is, a solution of the equation $\cos(x) = x$, which is precisely the equation with which we began this section. In fact, if we use Maple and try the following commands, we can obtain an approximation of this fixed point. Start by defining `x:=1`, and then enter `x:=evalf(cos(x))`, repeating this last command several times. The first time, we obtain $x = 0.5403023059$; the second time, $x = 0.8575532158$; the third time, $x = 0.6542897905$; the fourth time, $x = 0.7934803587$; then, $x = 0.7013687737$. After several iterations, we observe that the first digit, namely 0.7, no longer changes. After more iterations, the first two digits, 0.73, also remain unchanged, and so on. This strongly suggests that the sequence converges, and therefore it provides an approximation to the solution of the equation $\cos(x) = x$.

Let us now introduce another condition that guarantees the convergence of this method.

**Definition 22.9.** Let $\varphi$ be a function defined on an interval $I$. We say that $\varphi$ is **contractive** on $I$, or that it is a **contraction**, if there exists a constant $q < 1$ such that for all $x, y \in I$, we have

$$|\varphi(x) - \varphi(y)| \leqslant q|x - y|.$$

The general idea is that a contractive function brings points closer together. Indeed, if $\varphi$ is a contraction on an interval $I$, then for any two points $x$ and $y$ in that interval, their images $\varphi(x)$ and $\varphi(y)$ are closer to each other than $x$ and $y$ were originally. As we will see, this property is extremely useful for finding fixed points.

**Theorem 22.10** (Banach's fixed-point theorem)**.** Let $\varphi$ be a contractive function on $I = [a, b]$ with contraction constant $q$, and suppose that $\varphi[I] \subseteq I$. Then $\varphi$ has exactly one fixed point $x_f$ in $I$. Moreover, for every choice of $x_0 \in I$, the sequence defined by $x_{k+1} = \varphi(x_k)$ converges to $x_f$ and satisfies

$$|x_f - x_{k+1}| \leqslant q \cdot |x_f - x_k| \quad \text{and} \quad |x_f - x_{k+1}| \leqslant \frac{q}{1-q} \cdot |x_{k+1} - x_k|.$$

Notice in particular that Banach's fixed-point theorem provides an estimate for the error of the method, namely

$$|E_{k+1}| \leqslant q\,|E_k|$$

which shows that the fixed-point method converges linearly; that is, it is a method of order 1. At this stage, the method may seem rather slow in comparison with others we have studied before, but we will later discuss some alternatives that allow us to improve its performance.

Before doing so, let us make one further remark. Again by Banach's fixed-point theorem, in order to guarantee convergence it is enough to show that the functions with which we are working are contractions. The natural question, then, is how to recognize such functions. The next result will help us answer this question.

**Theorem 22.11.** Assume that the function $\varphi$ defined on an interval $I$ has a continuous derivative on the interior $I^\circ$ of $I$. If there is $q < 1$ such that $|\varphi'(t)| \leqslant q$ on $I^\circ$, then $\varphi$ is a contraction on $I$ with coefficient $q$.

Let us apply the previous theorem to the function $\varphi(x) = \cos(x)$. We have $\varphi'(x) = -\sin(x)$ and therefore $|\varphi'(x)| = |\sin(x)|$. Now, although it is true that $|\sin(x)| \leqslant 1$ for every $x \in \mathbb{R}$, this is not enough to apply the theorem on the whole real line, since the theorem requires a constant $q < 1$. However, if we restrict ourselves to a suitable interval $I$ on which $|\sin(x)| \leqslant q < 1$, then the theorem shows that $\varphi$ is a contraction on that interval.

In any case, it is not always easy to determine whether a function is a contraction, since in many situations the function may not be as simple as in the previous example. However, the condition $|\varphi'(t)| \leqslant q < 1$ suggests that the function $\varphi$ must be rather flat. It is also worth noting that not every function with a fixed point is a contraction. In other words, Banach's fixed-point theorem gives a very powerful sufficient condition for the existence, uniqueness, and approximation of fixed points, but it is not a necessary one. There are functions that are not contractive and nevertheless do have fixed points, and in some cases the corresponding fixed-point iteration may still converge. For this reason, if we restrict our attention only to Banach's theorem, we may lose part of the picture and exclude important examples for which fixed points exist even though the contraction condition is not satisfied.

## 22.3 Cobweb pattern and the fixed-point method

Let us now illustrate how the iteration works through a very clear geometric interpretation.

**Example 22.12.** Geometrically, the iteration $x_{k+1} = \varphi(x_k)$ can be understood as a step-by-step construction on the plane using both the graph of $y = \varphi(x)$ and the diagonal line $y = x$. We start with an initial value $x_0$ on the $x$-axis. From the point $(x_0, 0)$, we move vertically until we reach the graph of the function $y = \varphi(x)$. The height of that point is precisely $\varphi(x_0) = x_1$. However, this value is still a $y$-coordinate, and in order to use it as the next input we must transfer it to the horizontal axis. This is done by moving horizontally to the diagonal line $y = x$, because on that line the horizontal and vertical coordinates are equal. In this way, the value $x_1$ becomes the new point on the $x$-axis from which we repeat the process.

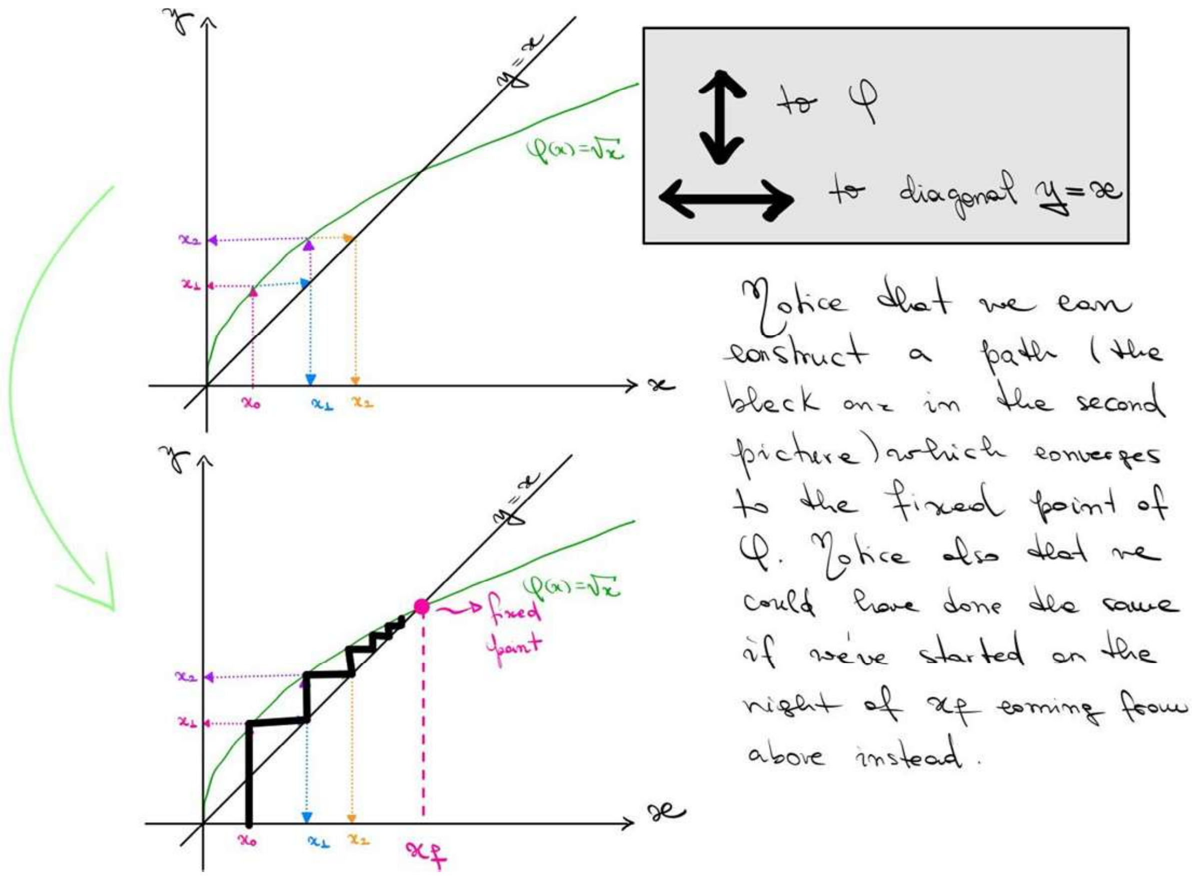


Figure 105: Geometric interpretation of the iterations $x_{k+1} = \varphi(x_k)$ with $\varphi(x) = \sqrt{x}$.

Thus, each iteration consists of two moves: first a vertical move from the diagonal or the $x$-axis up to the graph of $y = \varphi(x)$, and then a horizontal move from the graph to the diagonal $y = x$. Repeating these two motions produces the staircase or cobweb pattern shown in Figure 105.

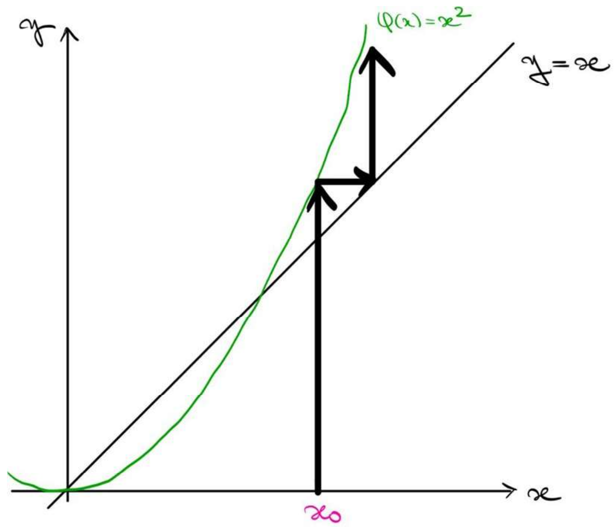


Figure 106: Geometric interpretation of the iterations $x_{k+1} = \varphi(x_k)$ with $\varphi(x) = x^2$.

Notice, however, that if we start at a point $x_0$ where the function is not flat but instead very steep, the iteration may diverge, as shown in Figure 106. On the other hand, if we choose a suitable initial point, then the iteration converges, as illustrated in Figure 107.

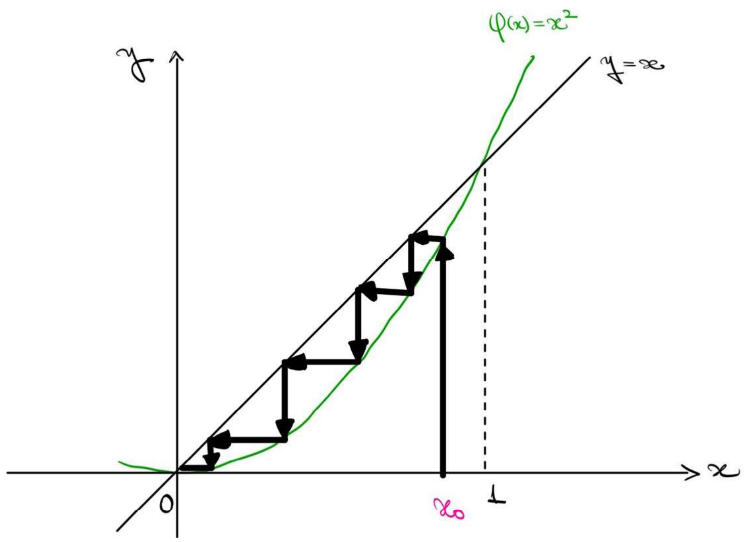


Figure 107: Geometric interpretation of the iterations $x_{k+1} = \varphi(x_k)$ with $\varphi(x) = x^2$.

In Figure 108 the iteration is being represented geometrically for the function $\varphi(x) = \cos(x)$, which in this part of the picture is a decreasing function. This means that as $x$ increases, the value of $\varphi(x)$ decreases, and this is why the graph of $\varphi$ slopes downward. We proceed at the same way as before. Starting from the initial value $x_0$ on the horizontal axis, we move vertically until we meet the graph of $y = \varphi(x)$. The height reached there is $\varphi(x_0) = x_1$. From that point, we move horizontally to the diagonal line $y = x$ in order to transfer that output value into the next input. Once we are on the diagonal, the horizontal coordinate is again equal to the vertical one, so we can repeat the procedure: from there we move vertically to the graph of $\varphi$, then horizontally back to the diagonal, and so on. Because the function is decreasing, the path alternates from one side of the fixed point to the other. In other words, the successive iterates do not approach the fixed point monotonically from just one side; instead, they jump across it. This produces the zigzag pattern shown in the figure.

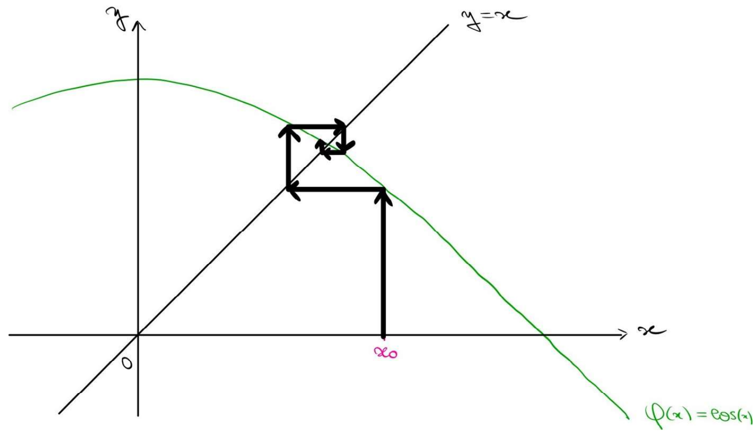


Figure 108: Geometric interpretation of the iterations $x_{k+1} = \varphi(x_k)$ with $\varphi(x) = \cos(x)$.

The last situation is pictured in Figure 109. The geometric iteration is shown for a decreasing function that is *very steep* near the fixed point. As before, the fixed point is the intersection between the graph of $y = \varphi(x)$ and the diagonal line $y = x$. However, in this case the function is not only decreasing, but also very steep. This means that a small change in the input produces a large change in the output. Geometrically, this causes the iteration to overshoot: instead of getting closer to the fixed point, the successive steps jump from one side to the other with larger amplitude. The staircase does not contract toward the intersection point; rather, it is pushed away from it.

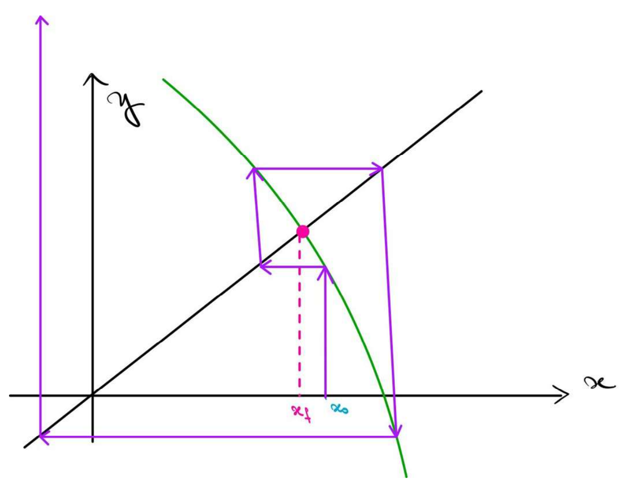


Figure 109: Geometric interpretation of the iterations $x_{k+1} = \varphi(x_k)$ with $\varphi(x)$ a steep function.

## 22.4 Relaxation

Consider once again the equation $\cos(x) = x$ (see Figure 110). Let us compare the fixed point and Newton's method to see what happens on Maple.

```
> givenphi:=cos(x):xstart:=0.0:
  plot({x,givenphi},x=0..1,color=[red,navy],scaling=constrained);
```

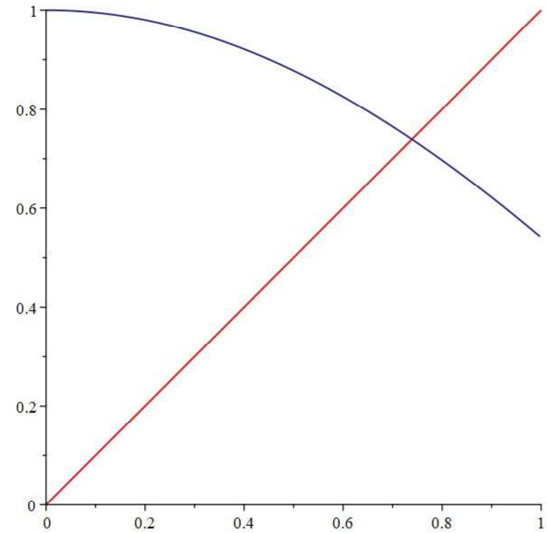


Figure 110: Intersection between the curves $y = \cos(x)$ and $y = x$.

The numerical results in Figure 111 clearly show the difference in performance between the two methods. The fixed-point iteration converges much more slowly, requiring many more steps to reach a similar accuracy, whereas Newton's method approaches the solution in only a few iterations. This behavior is completely consistent with the theory. Indeed, the fixed-point method is a method of order 1, that is, it converges linearly, while Newton's method is a method of order 2, meaning that it converges quadratically.

We now ask whether there is a way to make the fixed-point method faster. This is precisely the idea we explore next. Consider once again the equation $\varphi(x) = x$. Since, by Theorem 22.11, flatter functions are more favorable for convergence, it is natural to try to modify $\varphi$ in such a way that the new iteration becomes less steep. A first idea would be to multiply both sides of the equation by a parameter $\lambda$, thinking of $\lambda$ as a small number. In this way, we obtain

$$\lambda\varphi(x) = \lambda x.$$

However, if we were to apply the fixed-point method directly to the function $\lambda\varphi(x)$, the fixed points would no longer be the same as those of $\varphi$, and so we would lose the original problem. To avoid this difficulty, we add $(1-\lambda)x$ to both sides of the equation. Then we get

$$\lambda\varphi(x) + (1-\lambda)x = \lambda x + (1-\lambda)x = x.$$

Therefore, the original fixed-point equation is equivalent to

$$\varphi_\lambda(x) = x$$

where we define the new function

$$\varphi_\lambda(x) := \lambda\varphi(x) + (1-\lambda)x.$$

```
> Root(givenphi-x,xinit=xstart,method=fixedpoint,tolerance=0.001,digits=7);
k=00  x=  0.0000000   f(x)=  1.0000000  test= 1.0000000
k=01  x=  1.0000000   f(x)= -0.4596977  test= 1.0000000
k=02  x=  0.5403023   f(x)=  0.3172509  test= 0.4596977
k=03  x=  0.8575532   f(x)= -0.2032634  test= 0.3172509
k=04  x=  0.6542898   f(x)=  0.1391906  test= 0.2032634
k=05  x=  0.7934804   f(x)= -0.0921116  test= 0.1391906
k=06  x=  0.7013688   f(x)=  0.0625909  test= 0.0921116
k=07  x=  0.7639597   f(x)= -0.0418573  test= 0.0625909
k=08  x=  0.7221024   f(x)=  0.0283153  test= 0.0418573
k=09  x=  0.7504178   f(x)= -0.0190137  test= 0.0283153
k=10  x=  0.7314040   f(x)=  0.0128333  test= 0.0190137
k=11  x=  0.7442374   f(x)= -0.0086326  test= 0.0128333
k=12  x=  0.7356047   f(x)=  0.0058203  test= 0.0086326
k=13  x=  0.7414251   f(x)= -0.0039182  test= 0.0058203
k=14  x=  0.7375069   f(x)=  0.0026404  test= 0.0039182
k=15  x=  0.7401473   f(x)= -0.0017781  test= 0.0026404
k=16  x=  0.7383692   f(x)=  0.0011980  test= 0.0017781
k=17  x=  0.7395672   f(x)= -0.0008069  test= 0.0011980
k=18  x=  0.7387603   f(x)=  0.0005436  test= 0.0008069

                                            0.7387603199

>
> Root(givenphi-x,xinit=xstart,method=newton,tolerance=0.001,digits=7);
k=00  x=  0.0000000   f(x)=  1.0000000  test= 1.0000000
k=01  x=  1.0000000   f(x)= -0.4596977  test= 1.0000000
k=02  x=  0.7503639   f(x)= -0.0189231  test= 0.2496361
k=03  x=  0.7391129   f(x)= -0.0000465  test= 0.0112510
k=04  x=  0.7390851   f(x)= -0.0000000  test= 0.0000278

                                            0.7390851334
```

Figure 111: Comparison between the fixed point and Newton's method.

This new function is a weighted combination of the old iterate $\varphi(x)$ and the current value $x$. In other words, the parameter $\lambda$ measures how much confidence we place in the transformation $\varphi(x)$, while the weight $(1-\lambda)$ measures how much we prefer to remain close to the current iterate. By adjusting these two weights, we may be able to obtain a new iteration function that is flatter than the original one, and therefore more favorable for convergence. The idea, then, is to play with the weights $\lambda$ and $(1-\lambda)$ in order to see whether a suitable choice can lead to a better and faster approximation of the fixed point. Let us take a look at an example.

Figure 112 illustrates the effect of replacing the original iteration function $\varphi$ by the modified family $\varphi_\lambda(x) = \lambda\varphi(x) + (1-\lambda)x$. All the curves in the figure pass through the same fixed point $x_f$, because if $\varphi(x_f) = x_f$, then

$$\varphi_\lambda(x_f) = \lambda\varphi(x_f) + (1-\lambda)x_f = x_f.$$

Thus, changing $\lambda$ does not change the fixed point; it only changes the shape of the iteration function (recall that we are interested in flat functions).

What the picture shows is precisely how the graph changes as we vary $\lambda$. When $\lambda = 1$, we recover the original function $\varphi$, represented by the blue curve. As $\lambda$ decreases, the new functions become a weighted average of $\varphi(x)$ and the line $y = x$. In this way, the graph is gradually deformed toward the diagonal. Geometrically, this means that we are mixing two pieces of information: the value suggested by the original iteration, namely $\varphi(x)$, and the current value $x$ itself. The purpose of doing this is to make the new iteration flatter near the fixed point. If the graph is too steep, the fixed-point iteration may converge slowly or even diverge (see Example 22.12). By choosing a suitable value of $\lambda$, we may reduce the slope of $\varphi_\lambda$ near $x_f$, and therefore improve the convergence of the method.

```
> xfixed:=0.7:
  l1:=1/5:l2:=2/5:l3:=3/5:l4:=4/5:l5:=1:
  plot([x,(1-l1)*x+l1*givenphi,(1-l2)*x+l2*givenphi,(1-l3)*x+l3*givenphi,(1-l4)*x+l4*givenphi,(1-l5)*x+l5*
  givenphi],x=xfixed-1..xfixed+1,legend=["x","lambda=0.2","lambda=0.4","lambda=0.6","lambda=0.8","lambda=1"],
   xtickmarks=[0.5,1,2,3,4],color=[red,magenta,brown,coral,green,blue],thickness=2,scaling=constrained);
```

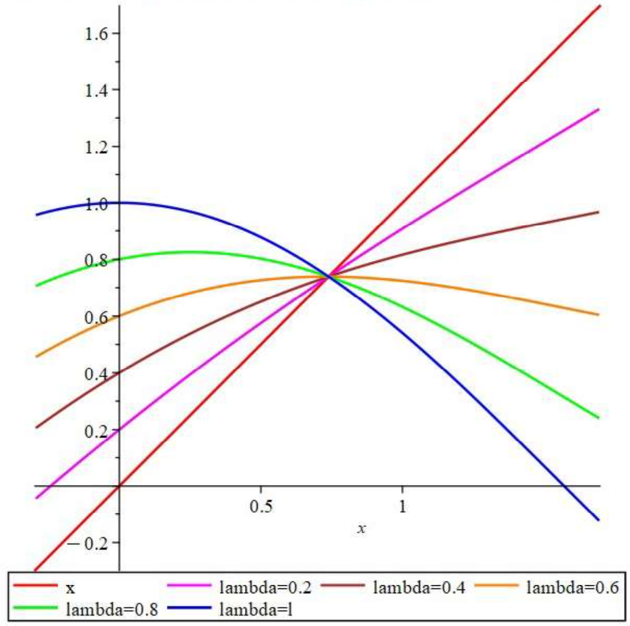


Figure 112: Relaxation procedure.

Comparing the two pictures (see Figure 113 and Figure 114), we see that when $\lambda = 1$ the iteration corresponds to the original function and converges rather slowly, whereas for $\lambda = 0.6$ the modified iteration function is almost flat near the fixed point and the convergence becomes much faster, reaching the same solution in only four steps.

```
> lamb:=1.0:
  myphi:=cos(x):
  Root(givenphi-x,xinit=xstart,method=[fixedpoint,lamb*myphi+(1-lamb)*x],tolerance=0.001,digits=7);
k=00  x=  0.0000000   f(x)=  1.0000000  test= 1.0000000
k=01  x=  1.0000000   f(x)= -0.4596977  test= 1.0000000
k=02  x=  0.5403023   f(x)=  0.3172509  test= 0.4596977
k=03  x=  0.8575532   f(x)= -0.2032634  test= 0.3172509
k=04  x=  0.6542898   f(x)=  0.1391906  test= 0.2032634
k=05  x=  0.7934804   f(x)= -0.0921116  test= 0.1391906
k=06  x=  0.7013688   f(x)=  0.0625909  test= 0.0921116
k=07  x=  0.7639597   f(x)= -0.0418573  test= 0.0625909
k=08  x=  0.7221024   f(x)=  0.0283153  test= 0.0418573
k=09  x=  0.7504178   f(x)= -0.0190137  test= 0.0283153
k=10  x=  0.7314040   f(x)=  0.0128333  test= 0.0190137
k=11  x=  0.7442374   f(x)= -0.0086326  test= 0.0128333
k=12  x=  0.7356047   f(x)=  0.0058203  test= 0.0086326
k=13  x=  0.7414251   f(x)= -0.0039182  test= 0.0058203
k=14  x=  0.7375069   f(x)=  0.0026404  test= 0.0039182
k=15  x=  0.7401473   f(x)= -0.0017781  test= 0.0026404
k=16  x=  0.7383692   f(x)=  0.0011980  test= 0.0017781
k=17  x=  0.7395672   f(x)= -0.0008069  test= 0.0011980
k=18  x=  0.7387603   f(x)=  0.0005436  test= 0.0008069

                                          0.7387603199
:
```

Figure 113: The fixed-point method (relaxation) with $\lambda = 1$.

```
> lamb:=0.6:
  myphi:=cos(x):
  Root(givenphi-x,xinit=xstart,method=[fixedpoint,lamb*myphi+(1-lamb)*x],tolerance=0.001,digits=7);
k=00  x=  0.0000000   f(x)=  1.0000000  test= 1.0000000
k=01  x=  0.6000000   f(x)=  0.2253356  test= 0.6000000
k=02  x=  0.7352014   f(x)=  0.0064943  test= 0.1352014
k=03  x=  0.7390980   f(x)= -0.0000215  test= 0.0038966
k=04  x=  0.7390851   f(x)=  0.0000001  test= 0.0000129

                                        0.7390850797
```

Figure 114: The fixed-point method (relaxation) with $\lambda = 0.6$.

This is called **relaxation** for the fixed-point method.

**Definition 22.13.** Given a continuous function $\varphi$ and a tolerance $\varepsilon > 0$, we choose some **relaxation** parameter $\lambda$ and apply fixed-point iteration to $\varphi_\lambda(x) = \lambda\varphi(x) + (1-\lambda)x$.