Assignment: Iterative Process

Andy Yan

September 2022

1 Question

The equation $x^2 - x - 2 = 0$ may be solved by using iterative process if you write it as $x_{n+1} = x_n^2 - 2$ When you reach a point where $x_n + 1 = x_n$, you have solved the equation. However, this process may not work because it is very sensitive to the initial value.

In this assignment, we will explore this sensitivity on initial value of the nonlinear system. Do the following:

1. Try the initial value $x_1 = 0$. See if this leads to a solution
2. Try the initial value $x_1 = 1$. See if this leads to a solution
3. Try the initial value $x_1 = 0.5$. See if this leads to a solution
4. Try the initial value $x_1 = 0.2$. See if this leads to a solution

Note: you need to come at least x10 to assume if it leads to a solution or not.

2 Problem Approach

In order to solve an equation using the iterative process, we begin with an initial value x_1.

$$x_2 = f(x_1)$$

$$x_3 = f(x_2) = f(f(x_1))$$

$$x_4 = f(x_3) = f(f(x_2)) = f(f(f(x_1)))$$

$$x_{n+1} = f(x_n) = f(f(x_{n-1})) = f(f(f(...$$

To compute the value of x_{n+1}, we must compute all the values before it, since each value is calculated based on the last value. At an unknown value of x_n , x_{n+1} will provide us with a solution of the equation. However, this process is tedious and isn't guaranteed to work. So let us use a program to do it for us.

3 Programming

Lets declare the major variables that we will be needing to use:

- **init** : double (initial value x_1)
- **seq** : double[] (a list that will store values x_1 to x_{n+1}

We will be using the double variable type to store decimals, and a double array so we can store the term of the value in the sequence. So,

$$init = x_1$$

$$seq[] = x_1, x_2, ..., x_{n+1}$$

The value of **init** will be inputted and **seq**[n] will return the x_n or *nth* term in the sequence. Using a for-loop, the program will return the value of **seq**[i] or x_i until the 10th term is reached.

Here is the C++ program.

```cpp
#include <bits/stdc++.h>
using namespace std;
int main() {
    double init, seq[11];
    cin>>init;
    seq[1]=init;
    for(int i = 2;i<=10;i++){
        seq[i]=seq[i-1] * seq[i-1] - 2;
        cout<<"n = "<<i<<" "<<seq[i]<<"\n";
    }
}
```

4 Data

Using the program above we can easily compute the first 10 terms of the sequence given any initial value x_1.

	$x_1 = 0$	$x_1 = 1$	$x_1 = 0.5$	$x_1 = 0.2$
$n = 2$	-2	-1	-1.75	-1.96
$n = 3$	2	-1	1.0625	1.8416
$n = 4$	2	-1	-0.871094	1.39149
$n = 5$	2	-1	-1.2412	-0.063754
$n = 6$	2	-1	-0.459433	-1.99594
$n = 7$	2	-1	-1.78892	1.98376
$n = 8$	2	-1	1.20024	1.9353
$n = 9$	2	-1	-0.559427	1.74537
$n = 10$	2	-1	-1.68704	1.04633

5 Analysis

Now we are given a number of values that could potentially be a solution to the equation $x^2 - x - 2 = 0$. Let us start with our initial values:

$$(0)^2 - (0) - 2 = -2 \neq 0$$

$$(1)^2 - (1) - 2 = -2 \neq 0$$

$$(0.5)^2 - (0.5) - 2 = -2.25 \neq 0$$

$$(0.2)^2 - (0.2) - 2 = -2.16 \neq 0$$

As we can observe, none of these are solutions. We can try our values when $n = 2$.

$$(2)^2 - (2) - 2 = 0$$

$$(-1)^2 - (-1) - 2 = 0$$

$$(0.5)^2 - (0.5) - 2 \neq 0$$

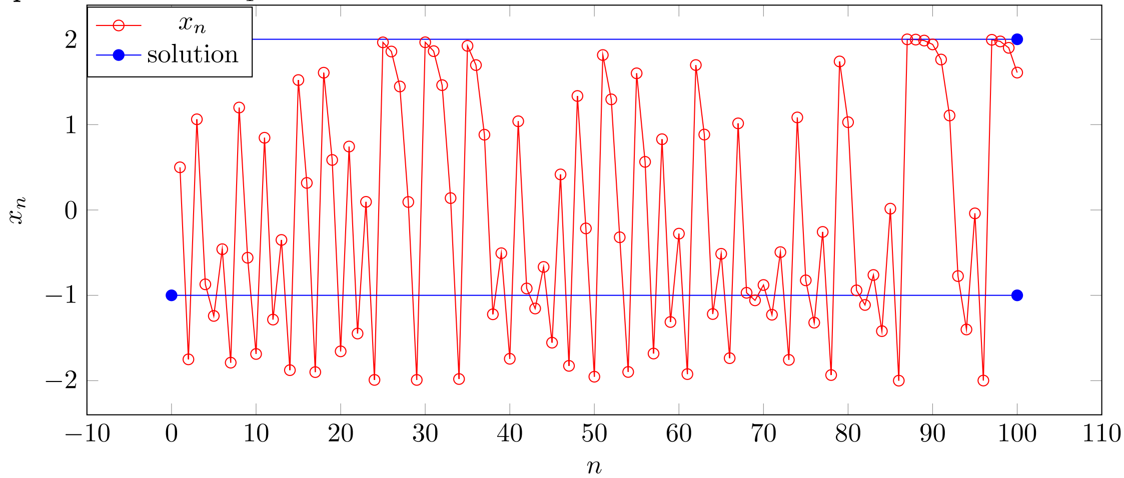
$$(0.2)^2 - (0.2) - 2 \neq 0$$

Now we have found our two solutions 2 and -1 to our quadratic equation. The initial values $x_1 = 0, 1$ do converge to a solution. However, our other two initial values do not converge to 2 or -1 within the 10 terms.

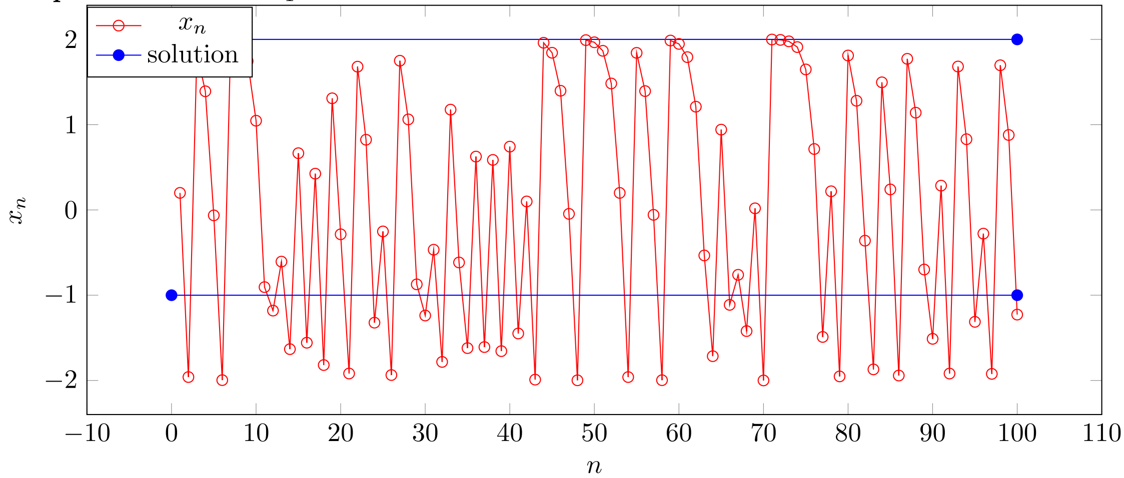
6 Graphs

For our graphs, we can alter our program to compute more values of n to provide a better image. We will represent the x_n value on the y-axis and the n value on the x-axis.

Graph 1: inital value $x_1 = 0.5$



Graph 2: inital value $x_1 = 0.2$



We still observe that the initial values $x_1 = 0.5, 0.2$ don't converge to 2 or -1. We can also observe oscillating behaviors for initial values $x_1 = 0.5, 0.2$, and we know that oscillating functions don't converge to a value when approaching infinity. Hence, we can conclude that initial values $x_1 = 0.5, 0.2$ **don't converge to a solution.**

7 Optional

Lets try to compute up to x_{1000} with the initial value 0.3.

Graph 3: inital value $x_1 = 0.3$