Aschaffenburg University of Applied Sciences
Faculty for Engineering
Prof. Dr. Michael Möckel

Email: Michael.Moeckel@th-ab.de
Telephone: +49 (0)6021 / 4206 - 507

Numerical Methods and Simulation
Module MT 03
Winter term 2022/23

https://moodle.th-ab.de/course/view.php?id=624

Problem Set 4

Fixed Point Algorithms for Root Finding

Coursework

Exercise 1: Create a fixed point algorithm lab

Fixed point iteration is a method for solving a nonlinear equation of the form $f(x) = 0$. A suitable algorithm A creates a sequence of iteration points x_i where i is an iteration index by $A(x_i) = x_{i+1}$. The solution of the original problem is represented by a fixed point $x*$ of the algorithm, i.e. at the fixed point the following condition holds:

$$A(x^*) = x^* \tag{1}$$

Note that iterative applications of the algorithm at the fixed point do not change its value:

$$A(A(x^*)) = A(x^*) = x^*$$

(this is why it is called a fixed point).
This exercise aims at building a little lab for comparing different fixed point algorithms and for developing and selecting new ones. The lab should consist of two routines: A Matlab function `[x] = Algorun(A, ` x_0`, n)` which calculates a sequence of n iteration points of the algorithm $A(x)$ and a plotting routine `Algoplot(A, x)`.

(1) Set up the Matlab function `[x] = Algorun(A, ` x_0`, n)` which returns a vector of iteration points x_i. A is an anonymous function. Use a loop to call the algorithm n times, inserting the last result as the new argument.

(2) Define a simple test example $g(x) = x^2$ and calculate $n = 5$ iteration points from an initial value $x_0 = 0.9$. What fixed point do you expect? Hint: think of the equation $g(x) = 0$.

(3) Then set up the function `Algoplot(A, x)` which takes the anonymous function A and the vector of iteration points as an argument. Include a dashed plot of the line $y = x$ using the ':' option in the `plot(x,x,':')` command and a plot of the function $A(x)$ in the maximum range of the data given by the elements of vector x. Hint: use `min(x), max(x)`.

(4) Write a suitable loop to include arrows in the plot which indicate the direction of the evolution under the algorithm. For that, define points `p1 = [x(i) x(i)] p2 = [x(i) x(i+1)]` and their distance `dp = p2-p1` and use vector plotting and labelling commands:

```
quiver(p1(1),p1(2),dp(1),dp(2),0)
text(p1(1),p1(2), sprintf('(%.0f,%.0f)',i,i))
```

(5) Test the plotting routine with the function and values given in (2).

Theorem: Convergence of the fixed point method

The fixed-point iteration method converges if, in the neighbourhood of the fixed point, the derivative of $A(x)$ has an absolute value smaller than one

$$\left|\frac{dA(x)}{dx}\right| < 1$$

Then –in pure mathematics– the function $A(x)$ is called *Lipschitz continuous.*

Exercise 2: Apply the fixed point algorithm lab

We want to use the fixed point algorithm lab from exercise 1 to study the root of the nonlinear function $f(x) = xe^{0.5*x} + 1.2x - 5$ in the interval [0,3]. The goal of this exercise is to re-write the equation in different ways to construct a suitable algorithm for a fixed point iteration and to see what may go wrong.

(1) Plot the function f in Matlab in the iterval [0,3]. Confirm that the function has a root in the interval [1,2].

(2) CASE A1: Rewrite the equation $f(x) = 0$ as $x = (5 - xe^{0.5x})/1.2$. Read off the algorithm $A1(x_i)$ and calculate the derivative $A1'(x) = dA(x)/dx$. Evaluate the derivative at the points $x_l = 1$ and $x_r = 2$.

(3) Use the algorithm lab for calculating and plotting $n = 5$ values using the iterative algorithm A1 with initial value $x_0 = 1.45$. What do you observe? Play with other values of x_0. Is this a suitable fixed point algorithm for finding the root?

(4) CASE A2: Rewrite the equation $f(x) = 0$ as $x = 5/(e^{0.5x} + 1.2)$. Read off the algorithm $A2(x_i)$ and calculate the derivative $A2'(x) = dA(x)/dx$. Evaluate the derivative at the points $x_l = 1$ and $x_r = 2$.

(5) Use the algorithm lab for calculating and plotting $n = 5$ values using the algorithm A2 with initial value $x_0 = 1.45$. What do you observe? Play with other values of x_0. Is this a suitable fixed point algorithm for finding the root? Increase n and see what happens.

(5) Is there another way of creating an iterative algorithm $A3$? Would it be helpful?

Model exam question

Solve the following question using a pocket calculator only (MATLAB won't be available in the exam).

Q	Max	Achieved
1	14	

Model exam question 1: Root finding

Bisection is a standard approach to find a root of a function. Consider the function $f(x) = e^{-x^2} - 1/2$. It has a root around x=0.83.

(a) Under which requirements does the bisection algorithm work?

Does this requirement hold for the current example (Y or N)?

(b) Describe the update mechanism of the bisection algorithm.

(c) Run the algorithm for four iterations by hand and fill in the results in the following tables (precision 2 digits).

Iteration i	x_L	x_R	$f(x_L)$	$f(x_R)$
1	0.0	1.0		
2				
3				
4				

Give a reasonable estimate for the final error of the calculated root: