Numerical Mathematics I, 2017/2018, Lab session 5

Deadline for discussion: Monday 28/05/18 at 5 pm

Keywords: rootfinding, nonlinear equations, fixed point iteration, Newton method

Remarks

- Make a new folder called `NM1_LAB_5` for this lab session, save all your functions in this folder.
- Whenever a new `MATLAB` function is introduced, try figuring out yourself what this function does by typing `help <function>` in the command window.
- Make sure that you have done the preparation before starting the lab session. The answers should be worked out either by pen and paper (readable) or with any text processing software (LaTeX, Word, etc.).

1 Preparation

1.1 Fixed point iteration

1. Study (Textbook, Section 2.3, 2.6 & 2.7).
2. For a twice continuously differentiable function $\phi : [a, b] \to [a, b]$ we approximate its fixed point α with the iteration process

$$x^{(k+1)} = \phi(x^{(k)}), \quad k = 0, 1, 2, ... \tag{1}$$

   (a) Prove the following two recurrence relations (2) and (3) for the error $e^{(k)} = x^{(k)} - \alpha$:

$$e^{(k+1)} = \phi'(\xi^{(k)})e^{(k)}, \tag{2}$$

$$e^{(k+1)} = \phi'(\alpha)e^{(k)} + \frac{\phi''(\eta^{(k)})}{2}(e^{(k)})^2, \tag{3}$$

   where $\xi^{(k)}$ and $\eta^{(k)}$ lie in between α and $x^{(k)}$. Hint: consider the Taylor series approximation of $\phi(x)$ centered at $x = \alpha$.

   (b) Let the order of convergence of the fixed point iteration be r. Show that the following holds as $k \to \infty$

$$r = \frac{\log(e^{(k+1)}/e^{(k)})}{\log(e^{(k)}/e^{(k-1)})}. \tag{4}$$

   (c) Prove the following relation

$$e^{(k)} = \frac{x^{(k)} - x^{(k+1)}}{1 - \phi'(\xi^{(k)})}, \tag{5}$$

   where $\xi^{(k)}$ lies in between α and $x^{(k)}$.

3. Consider solving the scalar root-finding problem $f(\alpha) = 0$ using the iteration function

$$\phi(x) = x + cf(x). \tag{6}$$

(a) If the root α is simple (i.e., $f'(\alpha) \neq 0$), show that the "best" convergence is obtained if the constant c is chosen equal to $-1/f'(\alpha)$.

(b) Since $f'(\alpha)$ is generally unknown in advance, we make the alternative choice $c = -1/f'(x^{(0)})$. Determine the corresponding order of convergence and asymptotic convergence factor.

4. Consider the iteration function

$$\phi(x) = x + C(x)f(x).$$

(a) For which function $C(x)$ do we obtain Newton's method?

(b) Assuming a simple root, show that the order of convergence of Newton's method equals two. Also determine the asymptotic convergence factor.

5. Consider the Aitken extrapolation formula (Textbook, Equation 2.31). Suppose we are given some fixed point iteration function ϕ, we define $\phi_\Delta(x)$ as

$$\phi_\Delta(x) = x - \frac{(x - \phi(x))^2}{\phi(\phi(x)) - 2\phi(x) + x}.$$

This is the first extrapolation of ϕ. The convergence order of $\phi_\Delta(x)$ is improved as compared to that of ϕ, as follows from (Textbook, Theorem 2.2). However, $\phi_\Delta(x)$ by itself is again an iteration function. Hence we can recursively define

$$\phi_\Delta^{(i+1)}(x) = x - \frac{(x - \phi_\Delta^{(i)}(x))^2}{\phi_\Delta^{(i)}(\phi_\Delta^{(i)}(x)) - 2\phi_\Delta^{(i)}(x) + x}, \tag{7}$$

with $\phi_\Delta^{(0)}(x) := \phi(x)$, and hence $\phi_\Delta^{(1)}(x) = \phi_\Delta(x)$.

(a) What are, according to the textbook, the orders of convergence of $\phi_\Delta^{(1)}$, $\phi_\Delta^{(2)}$ and $\phi_\Delta^{(3)}$ if ϕ converges linearly to a simple root of f?

(b) What if ϕ converges quadratically to a simple root of f?

(c) How many function evaluations of f are needed per evaluation of $\phi_\Delta^{(n)}$, if $\phi_\Delta^{(0)}$ is the static iteration method?

(d) What if $\phi_\Delta^{(0)}$ is Newton's method (here you should also count the evaluation of f' as a function evaluation)?

2 Lab experiments

2.1 Fixed point iteration

Introduction

In this exercise you will investigate the convergence properties of several fixed point methods. As indicated in the preparation, the idea of Aitken extrapolation leads to a family of iterative methods defined by its "parent" iteration function $\phi = \phi_\Delta^{(0)}$.

For all fixed point methods you should terminate the algorithm based on the error estimate given by (5), where you may assume that $\phi'(\xi^{(k)})$ is small and therefore you should use

$$\delta^{(k)} = x^{(k)} - x^{(k+1)}, \tag{8}$$

as the error estimate.

Static iteration

Write a MATLAB function staticIteration for the static iteration (1) with the iteration function (6). The header of your function should look like

```
% INPUT
% f          function of rootfinding problem
% c          static parameter: xnew = x + c*f(x)
% x0         initial guess
% tol        desired tolerance
% maxIt      maximum number of iterations
% OUTPUT
% root       root of f
% flag       if 0: attained desired tolerance
%            if 1: reached maxIt nr of iterations
% convHist   error estimate per iteration
% rootHist   root approximation per iteration
function [root, flag, convHist, rootHist] = staticIteration(f, c, x0, tol
    , maxIt)
```

Test the function staticIteration on the following rootfinding problem

$$\text{Find } \alpha \text{ such that } f(\alpha) = 0, \text{ where } f(x) = \cos(x) + 1.1x. \tag{9}$$

Use as initial guess $x^{(0)} = 0$, and use tol = 1E-12, maxIt = 25. Let c be such that $\phi'(x^{(0)}) = 0$.

Static iteration with Aitken extrapolation

Write a MATLAB function aitkenIteration for the recursive Aitken extrapolation (7) using the iteration function (6). The header should be the same as that of staticIteration, except that it takes an extra argument called maxDepth which specifies the number of extrapolations n that should be applied. Hence the iteration function $\phi_{\Delta}^{(n)}$ should be used and moreover for maxDepth = 0 your function should do exactly the same as staticIteration.

Since the iteration function $\phi_{\Delta}^{(n)}$ is recursively defined (7), it follows that your implementation should also be recursive.

Test your implementation on (9).

Newton's method with Aitken extrapolation

Finally, extend the functionality of aitkenIteration such that the parameter c can also be a function $c = C(x)$. Note that this should result in a *minor* adjustment to your code*. From Preparation question 1.1.4 it follows that for the right choice of $C(x)$ you have now implemented the Aitken extrapolated Newton method.

Test your implementation of the Newton method on (9).

*You can test whether c is a floating-point number or a function handle using isa(c, 'double') or isa(c, 'function_handle') respectively.

Comparison

For both the extrapolated static iteration as well as the extrapolated Newton iteration, consider the following iteration functions $\phi_\Delta^{(0)}, \phi_\Delta^{(1)}, \phi_\Delta^{(2)}, \phi_\Delta^{(3)}$. Using the test problem (9), compare the resulting eight fixed point methods:

- Plot the error estimates $\delta^{(k)}$ in logarithmic scale for all eight methods in a single plot (make sure the lines are distinguishable). Use the number of function evaluations $l(k)$ (count the evaluations of both f and f') instead of the iteration number k as the horizontal scale, see Preparation questions 1.1.5.c & 1.1.5.d.
- Plot the relative error of the error estimate $|e^{(k)} - \delta^{(k)}|/e^{(k)}$ in logarithmic scale for the static iteration methods $\phi_\Delta^{(0)}$ and $\phi_\Delta^{(1)}$. For computation of the error $e^{(k)}$ you may use the 'exact' root given by $\alpha = -0.69704098638574585$.
- Calculate the convergence orders using (4) (instead of considering the limit $k \to \infty$, use the largest k for which you can still compute the order). Use an intentionally 'bad' initial guess $x^{(0)} = 10$ (modify c accordingly) to be able to better compute the convergence orders.
- Measure the average calculation time per iteration for each of the methods.

3 Discussion

3.1 Fixed point iteration

1. Consider the n-th extrapolation of the static iteration function (6) denoted by $\phi_\Delta^{(n)}$.
   (a) Compare (5) to (8). Is $\delta^{(k)}$ always a good error estimate when $n = 0$? What if $n > 0$? Do your experiments confirm this?
   (b) Consider $\lambda^{(n)}$ defined as (Textbook, Equation 2.30)
   $$\lambda^{(n)} := \frac{\delta^{(k)}}{\delta^{(k-1)}},$$
   for which it holds that $\lim_{k\to\infty} \lambda^{(n)} = \phi'(\alpha)$ (Textbook, Lemma 2.1). How can we use $\lambda^{(n)}$ to obtain an error estimate which also holds for $\phi'(\alpha) \approx 1$?
2. Do the found convergence orders agree with the theory? Why is it difficult to check whether the theoretical orders of convergence actually are attained in your experiments for higher order methods?
3. Assume that the function evaluations of f and f' are the most expensive parts of the fixed point methods. When increasing the number of extrapolations n by one, the average calculation time per iteration should increase by a factor two (see Preparation question 1.1.5.c & 1.1.5.d). Are your experiments in agreement with this?
4. For the rootfinding problem $f(\alpha) = 0$, what are the pros and cons when comparing the Newton method without extrapolation to the once extrapolated static iteration (6)?

5. Consider Newton's method and the static iteration method, both without extrapolation. For both methods, plot $|\phi'(x)|$ logarithmically on the interval $x \in (-10, 10)$ (for the static iteration choose c such that $\phi'(10) = 0$). Also include the line $y = 1$. Explain why Newton's method behaves worse for $x^{(0)} = 10$ as compared to $x^{(0)} = 0$, and why the static iteration does not suffer from this problem.