R. Hiptmair
S. Hosseini
C. Pagliantini

Spring Term 2014

Numerical Analysis I

ETH Zürich
D-MATH

Homework Problem Sheet 10

Introduction. This problem sheet is devoted to iterative methods for non-linear (systems of) equations, in particular Newton's Method. This was covered in Chapter 5 of the course.

Problem 10.1 Newton's Method

Whenever discussing Newton's method for $f : D \subset \mathbb{R} \to \mathbb{R}$ in class, we made the (tacit) assumption that $f'(x) \neq 0$ in a neighborhood of $x^\star$ with $f(x^\star) = 0$. This guarantees that Newton's method is well defined and converges quadratically, once we start close enough to $x^\star$. In this problem we investigate what happens, in case $f'(x^\star) = 0$.

First some useful terminology is introduced:

Definition. For $f \in C^m(I)$, $I \subset \mathbb{R}$, a zero $x^\star \in I$ has *multiplicity* $p \in \mathbb{N}$, $p \leq m$, if $f(x^\star) = f'(x^\star) = \ldots = f^{(p-1)}(x^\star) = 0$ and $f^{(p)}(x^\star) \neq 0$.

(10.1a) Determine the multiplicity of the zero $x^\star = 0$ of $f(x) = x \sin x$.

(10.1b) State the iteration function Φ for a fixed point method that agrees with Newton's method for $f(x) = x \sin x$. Extend the definition of $\Phi(x)$ to $x^\star = 0$ and justify your definition of $\Phi(x^\star)$.

(10.1c) Show that the iteration $x^{(k+1)} = \Phi(x^{(k)})$ based on the iteration function from subproblem (10.1b) converges for $x^{(0)}$ in some neighborhood E of $x^\star = 0$.

HINT: Use Banach's fixed point theorem.

(10.1d) Assume that $x^{(0)} \in E$, where E is as in subproblem (10.1c). Show that the Newton's iteration for $f(x) = x \sin x$ converges linearly to $x^\star = 0$.

(10.1e) Assume that $x^\star$ is a double zero of a function $f : \mathbb{R} \to \mathbb{R}$. Prove that the quadratic convergence in Newton's iteration is restored by switching to the modified iteration

$$x^{(n+1)} = x^{(n)} - 2\frac{f(x^{(n)})}{f'(x^{(n)})} .$$

(10.1f) Using the result in subproblem (10.1e), define a modified version of Newton's method for $f(x) = x \sin x$ that converges locally with order two to $x^\star = 0$.

(10.1g) Implement a MATLAB function

```
x = newtonxsinx(x0,tol,nmax)
```

that computes an approximation of $x^\star$ root of $f(x) = x \sin x$ with initial guess `x0`. The output `x` is a vector containing all the iterates of Newton's method until convergence. As a stopping criterion, use the maximum number of iterations `nmax` and the distance of the current iterate to the root $x^\star$ with tolerance given as `tol`.

Test your code for two different initial values `x0` $= 1$ and `x0` $= 4$, tolerance $0.5 \cdot 10^{-5}$ and `nmax` $= 10$.

Listing 10.1: Test call for subproblem (10.1g)

```
x = newtonxsinx(1, 0.5*10^-5, 10)
x = newtonxsinx(4, 0.5*10^-5, 10)
```

Listing 10.2: Output for Test call for subproblem (10.1g)

```
>> test_call

x =

    1.0000    0.3910    0.1903    0.0946    0.0472    0.0236

    0.0118    0.0059    0.0030    0.0015    0.0007

x =

    4.0000    3.1021    3.1421    3.1416    3.1416
```

Problem 10.2 Fixed Point Iteration and Newton's Method

Let $f(x) = x^2 g(x)$ for $x \in [-1, 1]$, where $g \in C^2([-1, 1])$ and $g(x) \neq 0$ for all $x \in [-1, 1]$. Obviously, this function has a single double zero in $[-1, 1]$. In this problem we look at Newton's method for this function from the perspective of a generic fixed point iteration, see Section 5.2 of the course.

Please note that parts of this problem draw on insights gained in Problem 10.1, which should be done before.

(10.2a) Determine the iteration function $\Phi : [-1, 1] \to \mathbb{R}$ for Newton's method applied to f. Simplify the expression as far as possible.

(10.2b) Show that if $|x^{(0)}| < \varepsilon \leqslant 1$ for some sufficiently small $\varepsilon > 0$, then the sequence determined by $x^{(0)}$ and the recursion $x^{(n+1)} = \Phi(x^{(n)})$ converges to 0 for $n \to \infty$. State one possible value of ε, which will, of course, depend on g.

(10.2c) Which order of convergence do you expect if $x^{(n)} \to 0$ for $n \to \infty$?

Problem 10.3 Julia Sets

Julia sets are famous fractal shapes in the complex plane. They are constructed from the basins of attraction of zeros of complex functions when the Newton method is applied to find them. This problem aims at examining the convergence of the Newton method and of a damped Newton method in order to find the cubic roots of unity.

In the space $\mathbb{C}$ of complex numbers the equation

$$z^3 = 1 \tag{10.3.1}$$

has three solutions: $z_1 = 1$, $z_2 = -\frac{1}{2} + \frac{1}{2}\sqrt{3}i$, $z_3 = -\frac{1}{2} - \frac{1}{2}\sqrt{3}i$, the cubic roots of unity.

(10.3a) As you know from the analysis course, the complex plane $\mathbb{C}$ can be identified with $\mathbb{R}^2$ via the map $\zeta(x, y) = x + iy$, $\zeta : \mathbb{R}^2 \to \mathbb{C}$. Using this identification, reformulate Equation 10.3.1 into a system of equations $\mathbf{F}(x, y) = \mathbf{0}$ for a suitable function $\mathbf{F} : \mathbb{R}^2 \mapsto \mathbb{R}^2$, where $\mathbf{F} = \zeta^{-1} \circ f \circ \zeta$.

(10.3b) Formulate Newton's method for the function $f : \mathbb{C} \to \mathbb{C}$ given by $f(z) = z^3 - 1$.

(10.3c) Formulate the Newton iteration for the non-linear equation $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ with $\mathbf{x} = (x, y)^T$ and $\mathbf{F}$ from subproblem (10.3a).

(10.3d) Denote by $\mathbf{x}^{(k)}$ the iterates produced by the Newton method from the previous subproblem with some initial vector $\mathbf{x}^{(0)} \in \mathbb{R}^2$. Depending on $\mathbf{x}^{(0)}$, the sequence $\mathbf{x}^{(k)}$ will either diverge or converge to one of the three cubic roots of unity.

Analyze the behavior of the Newton iterates by implementing a MATLAB function

```
x = NewtonJuliaSet(Nit, nGridPts, tol)
```

which takes as input the maximum number `Nit` of iterations and

- uses `nGridPts` equally spaced points on the domain $[-2, 2]^2 \subset \mathbb{R}^2$ as starting points of the Newton iterations,
- colors the starting points differently depending on which of the three roots is the limit of the sequence $\mathbf{x}^{(k)}$. The decision about which root is the limit should be based on a nearest distance criterion.
- stops the iteration once the iterate is closer in distance to one of the third roots of unity than `tol`.

The three (non connected) sets of points whose iterations are converging to the different z_i are called Fatou domains, their boundaries are the Julia sets.

HINT: Useful MATLAB commands are `pcolor`, `colormap`, `shading`, `caxis`.

Verify your code by plotting the Julia sets on a mesh of 500×500 points, with tolerance 10^{-4} and `Nit`=20.

(10.3e) In [NMI, Thm. 5.20], you have seen that Newton's method converges quadratically, but only *locally*. The region of convergence can be small and hence the initial guess needs to be close enough to the solution. In order to "enlarge" the region of convergence, one can make use of a *damped* Newton's method, see Section 5.5.3 of the course.

In class we discussed the details of a MATLAB function

```
[x, res] = dampnewton(x,F,DF,abstol,reltol)
```

that realizes the damped Newton iteration with a sophisticated affine-invariant damping strategy. The function takes as input the initial guess `x` and the absolute and relative tolerances for terminations. The minimum damping factor has been fixed to $\lambda_{\text{min}} = 0.001$. An implementation is available from the course webpage as `dampnewton.m` and it should be used in this problem.

Specifically, we study the convergence of the damped Newton method for computing the cubic roots of unity as in subproblem (10.3d) and conduct a numerical experiment similar to that of subproblem (10.3c).

Repeat the computation of the Julia set from subproblem (10.3c), now using the damped Newton's method instead of the plain Newton iteration. Use $\texttt{abstol} = \texttt{reltol} = 10^{-6}$ when invoking `dampnewton`.

Compare the plot of the regions of convergence obtained with the methods from subproblem (10.3d) and subproblem (10.3e). What kind of difference do you notice? In which way does the damped method change the basins of attraction of the cubic roots?

Published on May 6, 2014.

To be submitted on May 13/14, 2014.

MATLAB: Submit all files in the online system. Include the files that generate the plots. Label all your plots. Include commands to run your functions. Comment on your results.

Written exercises: Hand-in the solutions during the exercise class or in the labeled boxes in HG G 53.x.

References

[NMI] Lecture Notes for the course "Numerical Analysis I".

Last modified on May 6, 2014