Prof. Dr. Dirk Lebiedz
Dipl.-Math. oec. Klaus Stolle
Institut für Numerische Mathematik
Universität Ulm

Angewandte Numerik 1
SoSe 2020
19.06.2020

Übungsblatt 09

Besprechung in den Tutorien in der Woche vom 29.06.2020 bis 03.07.2020

Für dieses Übungsblatt gibt es 12 Theorie- und 7 Matlab-Punkte, sowie 20 Theorie- und 13 Matlab-Zusatzpunkte. Punkte, die mit einem * gekennzeichnet sind, sind Zusatzpunkte.
Die 70-Prozent-Grenzen liegen aktuell (inklusive Blatt 09) bei 140,7 Theorie- und 86,8 Matlabpunkten.

Aufgabe 36 *(Programmieraufgabe: Fixpunktiterationen)* (4T*+2T*+3M+4M+2T+3M*+3M* Punkte)
Zur Bestimmung von $\sqrt{5}$ soll die positive Nullstelle der Funktion

$$f(x) = x^2 - 5 \tag{1}$$

berechnet werden. Wir untersuchen dazu die folgenden Fixpunktiterationen

$$\begin{array}{llll}
\text{Fixpunktiteration 1:} & x_{k+1} = \Phi_1(x_k) & \text{mit} & \Phi_1(x) = 5 + x - x^2 \\
\text{Fixpunktiteration 2:} & x_{k+1} = \Phi_2(x_k) & \text{mit} & \Phi_2(x) = \frac{5}{x} \\
\text{Fixpunktiteration 3:} & x_{k+1} = \Phi_3(x_k) & \text{mit} & \Phi_3(x) = 1 + x - \frac{1}{5}x^2 \\
\text{Fixpunktiteration 4:} & x_{k+1} = \Phi_4(x_k) & \text{mit} & \Phi_4(x) = \frac{1}{2}\left(x + \frac{5}{x}\right)
\end{array}$$

a) Zeigen Sie, dass für die Funktionen Φ_i, $(i = 1, 2, 3, 4)$ gilt

$$\Phi_i(x^*) = x^* \iff f(x^*) = 0.$$

b) In der Vorlesung wurde bereits mehrfach das Newton-Verfahren zur näherungsweisen Berechnung einer Nullstelle einer Funktion f erwähnt. Die Iterationsvorschrift des Newton-Verfahrens für eine eindimensionale Funktion f, also $f : \mathbb{R} \to \mathbb{R}$, lautet

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \qquad \text{für } k = 0, 1, 2, \ldots$$

Entspricht die Iterationsvorschrift des auf die Funktion f aus Gleichung (1) angewendeten Newton-Verfahrens einer der obigen Fixpunktiterationen?

c) Schreiben Sie eine MATLAB-Funktion `xks = fixPktIt(phi, anzIt, x0)`, die die ersten `anzIt` Iterierten der Iterationsfunktion `phi` beginnend beim Startwert `x0` berechnet. Ihre MATLAB-Funktion `fixPktIt` soll einen Vektor `xks` zurück geben, der den Startwert `x0` und diese ersten `anzIt` Iterierten enthält.

d) Schreiben Sie ein MATLAB-Skript `main36`, das für die vier Iterationsfunktionen Φ_i, $i = 1, \ldots, 4$, jeweils die ersten 10 Iterierten beginnend beim Startwert $x_0 = 5/2$ berechnet und diese übersichtlich in einer Tabelle ausgibt. Jede Tabellenzeile soll dabei den Index k $(k = 0, \ldots, 10)$ der Iteration und die vier Iterationswerte $\Phi_i(x_k)$ enthalten.

Hinweis: Zur Definition der vier Iterationsfunktionen Φ_i können Sie *anonyme Funktionen* verwenden: `<Funktionsname> = @(<Argumentliste>) Funktionsbeschreibung` legt eine Variable vom Typ `function handle` an, die dann als Parameter an eine andere Funktion, beispielsweise Ihre Funktion `fixPktIt`, übergeben werden kann. Für die erste Fixpunktiteration $\Phi_1(x) = 5 + x - x^2$ legt `phi1 = @(x) 5 + x - x.^2` die Variable `phi1` an, die Sie mit `xks = fixPktIt(phi1, 10, 5/2)` an Ihre MATLAB-Funktion übergeben können.

e) Interpretieren Sie Ihre Ergebnisse. Was lässt sich über das Konvergenzverhalten der vier Iterationen vermuten?

f) Erweitern Sie Ihr MATLAB-Skript `main36` so, dass Sie für jede der vier Iterationsfunktionen Φ_i, $i = 1, \ldots, 4$, ein Schaubild erzeugen, in das Sie jeweils die Funktion Φ_i und die Winkelhalbierende analog zu den Schaubildern der Folien 151 und 152 einzeichnen.

g) Zeichnen Sie in die Schaubilder jeweils auch den Verlauf der ersten drei Iterationen ein.

Aufgabe 37 *(Zyklisches Iterationsverhalten)* (2T*+1T*+4T* Punkte)

In dieser Aufgabe betrachten wir die Fixpunktiteration $\Phi(x) = r\,x\,(1-x)$, wobei r ein reeller Parameter mit $r \in (3, 4)$ ist.

a) Berechnen Sie die beiden Fixpunkte x_1^* und x_2^* von Φ.

b) Machen Sie sich klar, dass jeder Fixpunkt x^* von Φ auch ein Fixpunkt von $\Phi \circ \Phi$ ist.

Zeigen Sie also: $\Phi(x^*) = x^* \quad \Longrightarrow \quad \Phi(\Phi(x^*)) = x^*$.

c) Zeigen Sie (durch Berechnung von $\hat{x}$ in Anhängigkeit vom Parameter r): Es existiert ein $\hat{x} \in \mathbb{R}$ mit $\Phi(\Phi(\hat{x})) = \hat{x}$ und $\Phi(\hat{x}) \neq \hat{x}$. $\hat{x}$ ist also ein Fixpunkt von $\Phi \circ \Phi$, aber kein Fixpunkt von Φ.

Anmerkung: Das bedeutet, dass die Iteration $x_{k+1} = \Phi(x_k)$ mit dem Startwert $\hat{x}$ sich nach jedem zweiten Iterationsschritt zyklisch wiederholt.

Aufgabe 38 *(Banachscher Fixpunktsatz)* (8T+2T+2T* Punkte)

Gegeben sei nun die Funktion $f : I \to \mathbb{R}$, $I := [0, 1]$, mit $f(x) = x^3 + x - 1$. Gesucht ist ein iteratives Verfahren, um das Nullstellenproblem $f(x) = 0$ zu lösen.

a) Zeigen Sie, dass das Nullstellenproblem durch das Iterationsverfahren

$$x_{k+1} = \Phi(x_k) = \big(1 + (x_k)^2\big)^{-1}$$

gelöst werden kann, und weisen Sie für jeden Startwert $x_0 \in I$ die Konvergenz des Verfahrens nach.

Hinweis: Sie können Folgerung 5.10 verwenden, um die Voraussetzungen des Banachschen Fixpunktsatzes zu zeigen.

b) Wie viele Iteration benötigt dieses Iterationsverfahren höchstens, um mit dem Startwert $x_0 = 0$ eine Genauigkeit von 10^{-10} zu erreichen?

c) Geben Sie eine weitere Iterationsvorschrift an, die das Nullstellenproblem lösen könnte. (Sie brauchen keine Aussage zur Konvergenz dieser Iterationsvorschrift zu machen.)

Aufgabe 39 *(Programmieraufgabe: Bisektionsverfahren)* (4M*+3M* Punkte)

a) Schreiben Sie eine MATLAB-Funktion `[a, b] = bisektion(f, a, b)`, die die Lösung einer Gleichung $f(x) = 0$ im Intervall $[a, b]$ mit dem Bisektionsverfahren auf zwei Nachkommastellen genau bestimmt. Rückgabewert soll das auf zwei Nachkommastellen genau bestimmte Intervall sein. Protokollieren Sie den Iterationsverlauf dadurch, dass Sie in jedem Schritt die Intervallgrenzen ausgeben.

b) Testen Sie Ihr Programm an den folgenden Gleichungen und Startwerten:

$$\begin{aligned} x^2 - 2 = 0, &\qquad a = 1,\ b = 3, \\ \sin(x) - \cos(2x) = 0, &\qquad a = 0,\ b = 1, \\ x^3 - 7x^2 + 11 = 5, &\qquad a = 2.7,\ b = 6.5. \end{aligned}$$

Interpretieren Sie den Iterationsverlauf und die berechneten Ergebnisse.

Aufgabe 40 *(Intervallschachtelung)* (5T* Punkte)

Das Bisektionsverfahren beruht auf dem Prinzip der Intervallschachtelung:
Sei $I_0 = [a_0, b_0]$ ein Anfangsintervall. Eine Annäherung an einen Punkt $x^* \in I_0$ kann man mit Hilfe der Intervallschachtelung erreichen. Dazu definiert man iterativ $I_n = [a_n, b_n]$ mit

$$a_{n+1} = \begin{cases} a_n\,, & \text{falls } x^* \leq \frac{b_n+a_n}{2} \\ \frac{b_n+a_n}{2}\,, & \text{falls } x^* > \frac{b_n+a_n}{2} \end{cases} \qquad \text{sowie} \qquad b_{n+1} = \begin{cases} \frac{b_n+a_n}{2}\,, & \text{falls } x^* \leq \frac{b_n+a_n}{2} \\ b_n\,, & \text{falls } x^* > \frac{b_n+a_n}{2}. \end{cases}$$

Wie viele Schritte benötigt das Bisektionsverfahren für das zweite Beispiel aus Aufgabe 39 (also $I_0 = [0, 1]$), um eine Genauigkeit von mindestens 10^{-8} zu erreichen? Bestimmen Sie $n_0 \in \mathbb{N}$, sodass $|x^* - y| < 10^{-8}$ für alle $y \in I_{n_0}$ gilt.

Hinweise:

Die Lösungen der Theorieaufgaben und Ihren Text zu den Programmieraufgaben können Sie in LaTeX erstellen oder handschriftlich aufschreiben und einscannen. Die Programmieraufgaben sind in Matlab zu lösen. Der Source Code muss strukturiert und dokumentiert sein.
Falls Sie die Aufgaben im Team lösen, geben Sie bitte auf allen Lösungen alle an der Aufgabe beteiligten Teammitglieder an. Jedes Teammitglied muss eine Lösung abgeben.
Speichern Sie Ihre Lösungen und Ihre Ergebnisse in einem Directory mit dem Namen **Blatt09_*Vorname*_*Nachname*** und verpacken Sie dieses in eine `.zip`-Datei. Laden Sie **spätestens 48 Stunden vor Ihrem Tutorium** diese `.zip`-Datei in Moodle hoch.