

## Programmierpraktikum Nr. 4 zur Vorlesung Einführung in die Numerik, Winter 2017/18

In dieser Übung vergleichen wir iterative Lösungsverfahren zur Berechnung von Nullstellen  $f(x) = 0$  eindimensionaler Funktionen  $f : \mathbb{R} \rightarrow \mathbb{R}$ . Dazu werden wir im Folgenden die Verfahren Bisektion, Fixpunktiteration, sowie das Newton-Verfahren verwenden.

- **Bisektion:** Sei  $f : \mathbb{R} \rightarrow \mathbb{R}$  eine im Intervall  $[a, b]$  stetige und streng monotone Funktion. Das Intervallhalbierungsverfahren (Bisektion) sucht eine Nullstelle  $z \in [a, b]$  von  $f$ , indem es sich die Eigenschaft zu Nutze macht, dass  $f(a) < 0$  und  $f(b) > 0$  (bzw.  $f(a) > 0$  und  $f(b) < 0$ ), falls eine solche Nullstelle existiert.

Die Bisektion ist durch folgenden Algorithmus definiert:

- 1: Wähle ein Startintervall  $[a, b]$
  - 2: Teste, ob  $f$  im Startintervall eine Nullstelle hat
  - 3: Berechne den Intervallmittelpunkt  $c = \frac{a+b}{2}$
  - 4: Falls  $f(c) = 0$  oder  $|c - a| < TOL$ , akzeptiere  $c$  als Lösung
  - 5: Falls nicht, vergleiche die Vorzeichen von  $f(c)$  und  $f(a)$  und wähle neues Startintervall durch setzen von  $a = c$  oder  $b = c$
  - 6: Gehe zu Schritt 3
- **Fixpunktiteration:** Durch die Vorschrift

$$x_{n+1} = g(x_n)$$

ist ein Iterationsverfahren definiert, das zu gegebener Kontraktion  $g : M \rightarrow M$  und Startwert  $x_0 \in M$  einen Fixpunkt  $z \in M$  der Gleichung

$$g(x) = x$$

approximiert. Dieses Verfahren wird zum Beispiel im Beweis des Banach'schen Fixpunktsatzes benutzt.

- **Newton-Verfahren:** Sei  $f : \mathbb{R} \rightarrow \mathbb{R}$  eine wenigstens einmal stetig differenzierbare Funktion mit regulärem  $f'(x)$  für alle  $x \in G \subset \mathbb{R}$  mit  $G$  geeignet gewählt. Wähle einen Startpunkt  $x_0 \in G$ . Das Newton-Raphson-Verfahren in  $\mathbb{R}^1$  ist durch folgende Iterationsvorschrift definiert

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

### Aufgabe P4.1:

- (a) Schreiben Sie drei Programme, die zu beliebig gewählter Funktion  $f$  mittels Bisektion, Fixpunktiteration, sowie Newton-Verfahren eine Nullstelle von  $f$  bestimmen.

Die Übergabeparameter an die Programme sollen sein:

- Bisektion: Funktion  $f$  und Startintervall  $[a, b]$
- Fixpunktiteration: Geeignet gewählte Funktion  $g = g(f)$  und Startwert  $x_0$
- Newton-Verfahren:  $f$ , Ableitung  $J = f'$  und  $x_0$

Implementieren Sie für die Fixpunktiteration und das Newton-Verfahren die zwei Abbruchkriterien:

1. Kontrollieren Sie das Residuum durch eine vorgegebene Toleranz:  $|g(x) - x| < TOL_1$  (Fixpunktiteration) bzw.  $|f(x)| < TOL_1$  (Newton-Verfahren).
2. Der Betrag der Differenz zweier aufeinander folgender Folgenglieder soll kleiner als eine vorgegebene zweite Toleranz sein:  $|x_{n+1} - x_n| < TOL_2$ .

Lassen Sie sich zu jedem der drei Verfahren die Lösung, die Gesamtanzahl an Iterationen, sowie den geschätzten Fehler (Abbruchkriterium) ausgeben.

Testen Sie Ihre Programme zunächst mit der Funktion  $f(x) = x^2 - 0.5$  zum Startintervall  $[a, b] = [0, 1]$  bzw. Startwert  $x_0 = 0.5$  und setzen Sie  $TOL = 10^{-3}$ .

**Hinweise:** • Das Vorzeichen einer Zahl kann mit der Octave-Funktion "sign" ermittelt werden. • Einfache Funktionen lassen sich in Octave mit einem sogenannten "function-handle" programmieren: z.B. lässt sich die Funktion  $f(x) = 2x + 1$  programmieren durch  $f = @(x)2 * x + 1$ .

(b) Visualisierung: Schreiben Sie ein eigenständiges Programm, das die Folge der Näherungen  $x_n$  zusammen mit der Funktion  $f$  in einem Plot visualisiert. Lassen Sie sich hierzu alle berechneten Näherungen  $x_n$  als Vektor ausgeben und übergeben diesen an die Visualisierungsfunktion. Stellen Sie die Funktion  $f$  als blaue Linie und die Folgenglieder  $x_n$  als rote Kreuze dar. Betiteln Sie außerdem den Plot mit dem Namen des benutzten Verfahrens.

(c) Konvergenzverhalten: Untersuchen Sie die Konvergenzgeschwindigkeit der vorgestellten Verfahren. Betrachten Sie dazu

$$|z - x_{n+1}| = c|z - x_n|^\alpha, \quad |z - x_{n+2}| = c|z - x_{n+1}|^\alpha,$$

( $z$  exakte Lösung des betrachteten Problems,  $(x_n)_{n \in \mathbb{N}}$  Folge der Iterierten) und berechnen Sie daraus  $c$  und  $\alpha$  für drei aufeinanderfolgende Iterierte.

Schreiben Sie ein Programm, das die Konvergenzordnungen  $\alpha$ , sowie die Konstanten  $c$  für jedes  $n$  ausgibt.

**Hinweis:** Für den Fall, dass keine exakte Lösung  $z$  bekannt ist, berechnen Sie eine möglichst genau Approximation  $\tilde{z}$  von  $z$  und nehmen  $z = \tilde{z}$  an.

(d) Benutzen Sie die in den Aufgabenteilen (a)-(c) geschriebenen Programme, um die Nullstellen der Funktionen in der nachfolgenden Tabelle zu berechnen.

Schreiben Sie hierfür ein Hauptprogramm, in welchem Sie die Funktionen  $f$ ,  $J$  und  $g$  definieren, sowie den Startwert  $x_0$  und das Startintervall  $[a, b]$ . Rufen Sie anschließend die Programme zur iterativen Berechnung der Nullstelle von  $f$  auf, zusammen mit der Konvergenzordnungs-Berechnung und der Visualisierung.

$f(x)$	$x_0$	$[a, b]$
$x^2 - 0.5$	0.5, -0.5, 0, 2	$[x_0 - 1, x_0 + 1]$
$\exp(x) - 2$	0, $\log(2) + 10^{-8}$ , 10	''
$\frac{(1+x)^2 - (1-x)^2}{(1+x)^2 + (1-x)^2}$	0.5, 1, $1 - 10^{-8}$	''

Testen Sie nun nacheinander mit den angegebenen Funktionen und Startwerten. Setzen Sie dabei die Toleranzen auf  $10^{-8}$ .

Welche Beobachtungen machen Sie? Welche Vor-/Nachteile haben die Verfahren? Beurteilen Sie Ihre Ergebnisse auf Grundlage der Ausgaben bzgl. erreichter Lösung, geschätztem Fehler, Anzahl an Iterationen und Konvergenzverhalten, sowie der grafischen Ausgabe. Beurteilen Sie auch auf Grundlage der Theorie!