

# physik441 Computerphysik — Vorlesung 4

Marcus Petschlies<sup>1</sup>

<sup>1</sup>Helmholtz-Institut für Strahlen- und Kernphysik

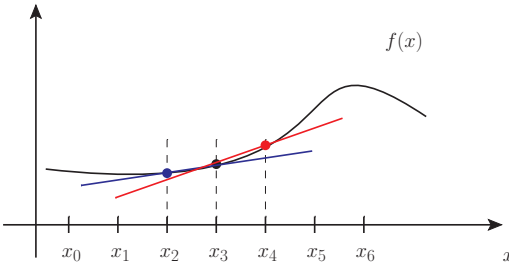
(Dated: May 4, 2023)

Wir fügen weitere Werkzeuge zu unserer numerischen Werkzeugkiste hinzu: die numerische Differentiation und die approximative Bestimmung von Nullstellen einer Funktion. Letzteres ist bereits ein Ausflug in den Bereich des Lösen nicht-linearer Gleichungen. Wir nutzen diese Gelegenheit für einen ersten Blick auf Fixpunkt-Verfahren.

## Numerische Ableitung.—

Das Ziel ist, für eine gegebene, reellwertige Funktion  $f(x)$  die Ableitungen  $f^{(k)}$  an einer Stelle  $x$  numerisch zu bestimmen. Dabei haben wir entweder eine Wertetabelle  $(x_i, f(x_i))$  vorgeben, oder wir können die Funktion  $f$  an selbst gewählten Stützstellen numerisch auswerten. Im folgenden nehmen wir den zweiten Fall, dass wir also äquidistante Stützstellen  $x_i, i = 0, \dots, n$  wählen können mit  $x_{i+1} - x_i = h$  für alle  $i$ .

Die Situation wird also durch den folgenden Plot beschrieben.



Dazu gehen wir davon aus, dass  $f$  hinreichen glatt, also hinreichend oft differenzierbar ist, damit die folgenden Formeln sinnvoll sind.

### 1. Ableitung

Wir nehmen  $x_i = x_3$  (im Plot oben) als Stelle, an der wir die Ableitung schätzen wollen. Für kleine Schrittweite  $h$  bekommen wir dann eine systematische verbesserbare Näherung für  $f$  in der Nähe von  $x_i$ . Nämlich mit Hilfe der Taylor-Entwicklung

$$f(x) = f(x_i) + f^{(1)}(x_i)(x - x_i) + \frac{1}{2} f^{(2)}(x_i)(x - x_i)^2 + \dots \quad (1)$$

Insbesondere können wir Gl. (1) an den Nachbarstellen  $x_{i\pm 1} = x_i \pm h$  auswerten

$$\begin{aligned} f(x_{i+1}) &= f(x_i) + f^{(1)}(x_i)h + \mathcal{O}(h^2), \\ f(x_{i-1}) &= f(x_i) - f^{(1)}(x_i)h + \mathcal{O}(h^2). \end{aligned} \quad (2)$$

Unter Vernachlässigung von Termen der Ordnung  $h^2$  stellen wir Gl. (2) um und erhalten

$$\begin{aligned} f^{(1)}(x_i) &= \frac{1}{h} (f(x_{i+1}) - f(x_i)) + \mathcal{O}(h) \quad \text{Vorwärtsableitung,} \\ f^{(1)}(x_i) &= \frac{1}{h} (f(x_i) - f(x_{i-1})) + \mathcal{O}(h) \quad \text{Rückwärtsableitung,} \end{aligned} \quad (3)$$

mit Richtungsangabe nach dem benutzten Nachbarpunkt.

Das sind natürlich gerade die Ableitungen, die wir durch Interpolation mit einem Polynom 1. Ordnung erhalten, mit Stützstellen  $x_i, x_{i+1}$  oder  $x_i, x_{i-1}$

$$P_1(x) = f(x_i) \pm \frac{f(x_{i\pm 1}) - f(x_i)}{h} (x - x_i). \quad (4)$$

Das können wir in bekannter Weise weiter entwickeln durch Interpolation mit einem Polynom 2. Grades gestützt an den Stellen  $x_{i-1}, x_i, x_{i+1}$ ,

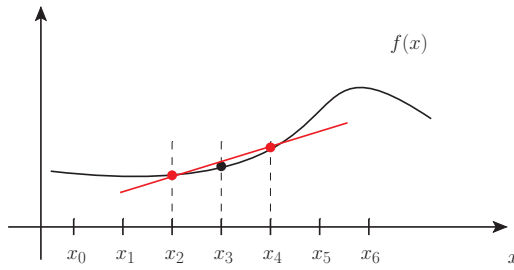
$$\begin{aligned} P_2(x) &= f(x_i) + \frac{f(x_{i+1}) - f(x_{i-1}))}{2h} (x - x_i) \\ &\quad + \frac{1}{2} \frac{f(x_{i+1}) + f(x_{i-1}) - 2f(x_i))}{h^2} (x - x_i)^2. \end{aligned} \quad (5)$$

Wir prüfen mit Gl. (5) leicht nach, dass  $P_2(x_k) = f(x_k)$  für  $k = i, i \pm 1$ .

Daraus erhalten wir die *symmetrische* 1. Ableitung als Differenzenquotienten

$$P_2'(x_i) = \frac{1}{2h} (f(x_{i+1}) - f(x_{i-1})) = f^{(1)}(x_i) + \mathcal{O}(h^2). \quad (6)$$

Die Geometrie dieser Variante ist im folgenden Plot veranschaulicht.



Uns fällt auf, dass wir bei der Symmetrisierung eine Ordnung im Fehlerterm gewinnen, der Fehler ist  $\mathcal{O}(h^2)$ . Bei Auswertung der Gl. (1) bei  $x_{i\pm 1}$

$$\begin{aligned} f(x_{i+1}) &= f(x_i) + f^{(1)}(x_i)h + \frac{1}{2}f^{(2)}(x_i)h^2 + \mathcal{O}(h^3) \\ f(x_{i-1}) &= f(x_i) - f^{(1)}(x_i)h + \frac{1}{2}f^{(2)}(x_i)h^2 + \mathcal{O}(h^3) \end{aligned} \quad (7)$$

hebt sich der Term mit  $f^{(2)}(x_i)h^2$  bei Subtraktion der beiden Gleichungen weg.

Das geht noch weiter, denn die symmetrische 1. Ableitung ist tatsächlich *symmetrisch* in der Schrittweite  $h$ , was aus der Schreibweise  $f(x_{i+1} - f(x_{i-1})) = f(x_i + h) - f(x_i - h)$  deutlich wird. Aufgrund dieser Symmetrieeigenschaft gibt es überhaupt *keine Korrekturterme mit ungerader Ordnung* in  $h$ , das heißt keine Korrekturterme von geraden Ableitungen  $f^{2k}(x_i)$ .

## 2. Ableitung

Aus  $P_2$  in Gl. (5) erhalten wir sofort einen Schätzwert für die 2. Ableitung

$$P_2''(x_i) = \frac{f(x_i + h) + f(x_i - h) - 2f(x_i)}{h^2} = f^{(2)}(x_i) + \mathcal{O}(h^2). \quad (8)$$

Wieder haben wir Fehlerterm erst ab der Ordnung  $h^2$ , weil auch diese 2. Ableitung symmetrisch in  $h$  ist, und damit Terme von ungeraden Ableitungen  $f^{(2k+1)}(x_i)h^{2k+1}/h^2$  nicht beitragen können.

Wir haben gesehen, dass für die Beispiele der symmetrischen 1. (und auch 2.) Ableitung die Entwicklung

$$\frac{f(x+h) - f(x-h)}{2h} = f^{(1)}(x) + \tau_1 h^2 + \tau_2 (h^2)^2 + \dots \quad (9)$$

gilt. Wir können also hier analog zur Romberg-Integration einen Schätzwert für die Ableitung herleiten durch Polynom-Interpolation und Auswertung bei  $h^2 = 0$ . Dazu bietet sich sofort die Neville-Rekursion an.

## Ausgedehnte Ableitungen

Wie können wir systematisch Ableitungsformeln finden, auch solche für höhere Ableitungen oder so, dass Korrekturen erst mit höherer noch Ordnung  $h$  beginnen?

Für den hier betrachteten Fall der äquidistanten Stützstellen geht das einfach ausgehend vom Ansatz

$$f^{(k)}(x) = \dots \alpha_{-2} f(x-2h) + \alpha_{-1} f(x-h) + \alpha_0 f(x) + \alpha_1 f(x+h) + \alpha_2 f(x+2h) + \dots, \quad (10)$$

also der Darstellung von  $f^{(k)}(x)$  als Linearkombination der Funktionswerte an den Stützstellen mit Gewichten  $\alpha_k$ .

Die Gewichte können wir bestimmen durch Taylor-Entwicklung der einzelnen Terme

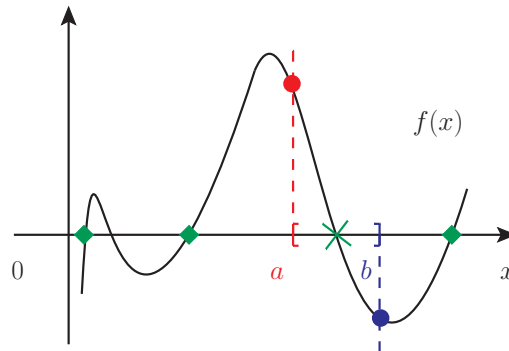
$$f(x+kh) = f(x) + kh f^{(1)}(x) + \frac{k^2}{2!} h^2 f^{(2)}(x) + \dots \quad (11)$$

zu genügend hoher Ordnung. Für ein gegebenes maximales  $k$  vergleichen wir dann die Koeffizienten der Potenzen von  $h$  auf beiden Seiten von Gl. (10), und lösen nach den Gewichten auf.

### Nullstellensuche.—

Die Nullstellensuche ist eigentlich das Lösen linearer und nichtlinearer Gleichungen, also ein sehr weites Feld. Wir beschränken uns auf den Fall einer reellwertigen Funktion  $f: \mathbb{R} \rightarrow \mathbb{R}$ ,  $x \mapsto f(x)$  einer Variablen.

Die Aufgabe ist also: finde  $x^*$ , sodass  $f(x^*) = 0$ . Der folgende Plot illustriert schon ein paar der Probleme, mit denen wir dabei rechnen müssen.



Die Nullstellensuche läuft in zwei Schritten ab: Ich muss zunächst einen Bereich finden, in dem überhaupt eine Nullstelle liegt. Wenn ich dann nah genug an der Nullstelle bin, dann kann ich die Verfahren anwenden, die wir nachfolgend anschauen.

Um ein *Einschließungsintervall*  $[a, b]$  zu finden, brauchen wir entweder weitere Informationen über die Funktion  $f$  (so wie hier, wie sehen den Graphen und wissen, wo in etwa die Nullstelle liegt, die wir approximieren wollen). Oder, wenn wir nichts über den Funktionsverlauf wissen, müssen wir eine grobe Suche durchführen.

Wir gehen hier davon aus, dass wir bereits Startwerte haben und suchen nun Verfahren, mit denen wir möglichst schnell an die Nullstelle herankommen.

### Bisektion

Woher wissen wir überhaupt, dass im Einschließungsintervall  $[a, b]$  eine Nullstelle liegt? Wir brauchen zwei Voraussetzungen:

1. Vorzeichenwechsel mit den Randpunkten, d.h.  $f(a) \cdot f(b) < 0$  (= 0 hieße, einer der Randpunkte ist bereits die Nullstelle);
2.  $f$  ist stetig in  $[a, b]$ ; das garantiert, dass  $f$  den Wert 0 tatsächlich auch annimmt, d.h.  $\exists x^* \in [a, b] : f(x^*) = 0$ .

Das Intervall  $[a, b]$  enthält also eine Nullstelle. Die Idee ist nun, ein neues, kleineres Intervall  $[a_1, b_1]$  zu finden, das immer noch die Nullstelle enthält.

Wir halbieren das Intervall und nehmen den Mittelpunkt  $m = (a + b)/2$ . Dann setzen wir das neue Intervall als

$$[a_1, b_1] = \begin{cases} [a, m] & \text{falls } f(a)f(m) < 0 \\ [m, b] & \text{falls } f(m)f(b) < 0 \end{cases} \quad (12)$$

Diese Vorschrift iterieren, wobei die Intervallbreite  $b_i - a_i = (b - a)/2^i$  mit jedem Schritt um den Faktor 2 kleiner wird.

Als Konvergenzkriterium nehmen wir dann gleich die Breite des Einschließungsintervalls. Für gegebenes  $\epsilon$  können wir die Anzahl der Schritte berechnen,  $N = \log((b - a)/\epsilon)/\log(2)$ .

*Fixpunktverfahren zur Nullstellensuche.*—

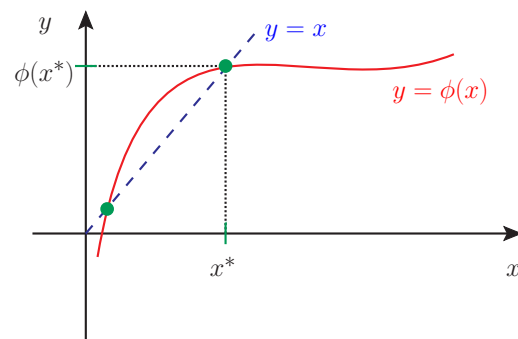
Wir wollen uns zwei weitere Verfahren zum Lösen nichtlinearer Gleichungen anschauen. Die Verfahren unterscheiden sich in der Geschwindigkeit, mit der wir eine vorgegebene Genauigkeit der Lösung erreichen. Die Geschwindigkeit ist für uns sehr wichtig.

Die iterativen Näherungsmethoden sind sogenannte *Fixpunktverfahren*: Mit einer *Iterationsfunktion*  $\phi$  erzeugen wir vom Startwert  $x_0$  eine Folge  $x_{i+1} = \phi(x_i)$ ,  $i = 0, 1, \dots$ . Wir nehmen dann folgendes an:

- $x^*$  ist ein Fixpunkt von  $\phi$ , d.h.  $x^* = \phi(x^*)$ .
- Die Fixpunkte von  $\phi$  sind die Nullstellen von  $f$ .
- $\phi$  soll in einer Umgebung von  $x^*$  stetig sein.

Die Idee ist dann, dass die Folge  $\{x_i\}_{i \geq 0}$  nach  $x^*$  konvergiert,  $x_i \xrightarrow{i \rightarrow \infty} x^*$ .

Die geometrische Interpretation des Fixpunktes zeigt der folgende Plot.



Der Graph von  $\phi$  schneidet im Fixpunkt die Diagonale  $y = x$ .

Aus Sicht des Iterationsverfahrens  $x_{i+1} = \phi(x_i)$  erreicht die Iteration am Fixpunkt das Ende.  $x_i$  läuft in  $x^*$  hinein, und danach gibt es keine Änderung mehr, eben weil  $\phi(\phi(x^*)) = \phi(x^*) = x^*$ .

Wenn wir also eine Funktion  $f$  gegeben haben und eine Nullstelle finden wollen mit Fixpunktiteration, dann fragt sich Woher bekommen wir eine geeignete Iterationsfunktion  $\phi$ ? Konvergiert die Iteration mit  $\phi$  überhaupt zur Lösung? Und wenn ja, wie schnell konvergiert die Folge?

Einen systematischen Ansatz bietet die

### Newton-Raphson-Methode

Wir betrachten also wieder  $f : \mathbb{R} \rightarrow \mathbb{R}$ . Wir setzen weiter voraus, dass  $f$  in einer Umgebung der Nullstelle  $x^*$  hinreichend oft differenzierbar ist (also die Ableitungen, die wir hinschreiben, sollen auch alle existieren). Diese Umgebung bezeichnen wir mit  $U(x^*)$ . Wir denken dabei an ein offenes Intervall, das  $x^*$  enthält.

Unter den genannten Voraussetzungen nutzen wir die Taylor-Entwicklung um eine Stelle  $x_0 \in U(x^*)$ .

$$\begin{aligned} 0 &= f(x^*) \\ &= f(x_0) + f^{(1)}(x_0)(x^* - x_0) + \frac{1}{2}f^{(2)}(x_0)(x^* - x_0)^2 \\ &\quad + \dots + \frac{(x^* - x_0)^{k+1}}{(k+1)!}f^{(k+1)}(x_0 + \nu(x^* - x_0)) \end{aligned} \quad (13)$$

mit  $\nu \in (0, 1)$ .

Wir können also  $f(x^*)$  darstellen also Polynom der Ordnung  $k$  plus den Restterm.

$$0 = f(x^*) = P_k(x^*) + \text{Restterm}. \quad (14)$$

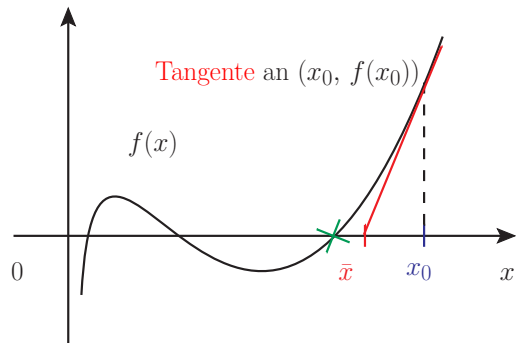
Wenn nun  $P_k$  in der Umgebung der Nullstelle eine Näherung für die Funktion  $f$  ist, dann sollte auch eine Nullstelle von  $P_k$  in der Nähe von  $x^*$  liegen. D.h. wir bestimmen zunächst nicht die Nullstelle von  $f$ , sondern die von  $P_k$ .

### Klassisches Newton-Raphson-Verfahren

Dazu benutzen wir Gl. (14) für  $k = 1$ , d.h. wir linearisieren  $f$  und lösen

$$\begin{aligned} 0 &= f(x_0) + (\bar{x} - x_0)f'(x_0), \\ \bar{x} &= x_0 - \frac{f(x_0)}{f'(x_0)}. \end{aligned} \quad (15)$$

Diese Bestimmung von  $\bar{x}$  ist im folgenden Plot veranschaulicht:



Der Schnittpunkt der Tangente an  $(x_0, f(x_0))$  mit der Ordinatenachse.

Daraus bekommen wir das Iterationsverfahren

$$x_{i+1} = \phi(x_i), \quad \phi(x) = x - \frac{f(x)}{f'(x)}. \quad (16)$$

Wir haben hier noch nichts dazu gesagt, ob und wie diese Iteration wirklich konvergiert. Das folgt noch.

Diese Herstellung einer Iterationsfunktion kann auch für Polynome höheren Grades gemacht werden. Allerdings wissen wir, dass die Nullstellenbestimmung dann aufwendiger wird. Für  $k = 2$  und ein Polynom  $P_2$  zweiten Grades können wir die Nullstellen noch mit  $(p, q)$ -Formel hinschreiben. Für  $k \geq 3$  müssen dann schon wieder numerische Verfahren benutzt werden.

### Zur Konvergenz

Zur Erinnerung, aus der Analysis: Was bedeutet es, dass eine Folge  $\{x_i\}$  gegen  $x$  konvergiert?

Mathematisch hieß das

$$\forall \epsilon > 0 \exists N(\epsilon) \in \mathbb{N} : |x_k - x| < \epsilon \quad \forall k \geq N(\epsilon).$$

Egal wie klein wir eine Umgebung von  $x$  wählen, wenn die Folge einmal eintritt, bleibt sie drin. Für immer gefangen.

Wenn wir Verfahren unterscheiden wollen nach der Geschwindigkeit, mit der sie uns eine Lösung vorgegebener Genauigkeit liefern, dann brauchen wir dazu eine Eigenschaft. Wir benutzen dazu die

**Konvergenzgeschwindigkeit** Ein Verfahren konvergiert mit *Ordnung*  $p \geq 1$  gegen  $x$ , falls es eine Konstante  $C > 0$  gibt und ein  $N$ , sodass für alle  $i \geq N$  gilt

$$|x_{i+1} - x| \leq C |x_i - x|^p. \quad (17)$$

Im Fall  $p = 1$  nennen wir das *lineare* Konvergenz und müssen zusätzlich fordern, dass  $C < 1$  ist, um wirklich eine Verkleinerung des Fehlers zu erreichen. Wir haben dann

$$\begin{aligned} |x_{i+k} - x| &\leq C |x_{i+k-1} - x| \leq C^2 |x_{i+k-2} - x| \\ &\leq \dots \leq C^k |x_i - x|. \end{aligned} \quad (18)$$

$C^k$  konvergiert gegen Null für  $C < 1$  und  $k \rightarrow \infty$ .

Der Fall  $p = 2$  heißt *quadratisch* konvergent. Quadratische Konvergenz ist schnell. Nehmen wir an, dass wir schon  $|x_i - x| \approx 0.1$  haben. Im nächsten Schritt dann also  $|x_{i+1} - x| \leq C |x_i - x|^2 \approx C 0.1^2 = C 0.01$ , und im nächsten  $C 0.0001$ . D.h. die Genauigkeit verdoppelt sich in jedem Schritt! Zum Vergleich, wenn  $C \approx 0.1$  ist, gewinnen wir eine Stelle pro Iterationsschritt.

Wir wenden Gl. (17) auf die Iterationsfunktion  $\phi$  an.

Wir nehmen also  $\phi : \mathbb{R} \rightarrow \mathbb{R}$  an, mit Fixpunkt  $x^*$ . Dazu soll es eine Umgebung  $U(x^*)$  geben, in der Gl. (17) gilt, als

$$|\phi(x) - x^*| \leq C |x - x^*|^p \quad (19)$$

für  $x \in U(x^*)$ .

Dann gilt: Es gibt eine Umgebung  $V(x^*) \subset U(x^*)$ , und wenn ich mit  $x_0 \in V(x^*)$  starte und die Folge  $x_{i+1} = \phi(x_i)$  erzeuge, dann konvergiert  $x_i \rightarrow x^*$  für  $i \rightarrow \infty$  mindestens mit der Ordnung  $p$ .

Im Fall  $p = 1$  zum Beispiel, also  $p = 1$ ,  $0 \leq C \leq 1$  und gegebenes  $\epsilon > 0$  nehmen wir  $V(x^*) = (x^* - \epsilon, x^* + \epsilon)$ . Dann gilt nämlich

$$|x_i - x^*| \leq C^i |x_0 - x^*| \leq C^i \epsilon \leq \epsilon, \quad (20)$$

wenn  $x_0 \in V(x^*)$  ist. Damit haben wir linear Konvergenz.

Nehmen wir den Fall  $p > 1$ . In diesem Fall wählen wir  $\delta \leq \min \left\{ \epsilon, \frac{1}{C^{p-1}} \right\}$  und  $V(x^*) = (x^* - \delta, x^* + \delta)$ .

Es folgt dann

$$\begin{aligned} |x_{i+1} - x^*| &\leq C |x_i - x^*|^p \leq C \delta^p \\ &= C \delta^{p-1} \delta \leq C \left( \frac{1}{C^{p-1}} \right)^{p-1} = \delta. \end{aligned} \quad (21)$$

Das heißt, wenn  $x_0 \in V(x^*)$ , dann bleiben auch alle folgenden  $x_i$  innerhalb der  $\epsilon$ -Umgebung.

Soweit, so gut. Wir wollen also für eine gefundene Iterationsfunktion  $\phi$  eine Abschätzung wie Gl. (17) nachweisen, dann haben wir die Konvergenz mit unterer Schranke für die Ordnung der Konvergenz.

Wie können wir denn nun die Ordnung für gegebenes  $\phi$  bestimmen? Folgender Weg erlaubt das, wenn  $\phi$  genügend oft differenzierbar ist (in einer Umgebung  $U(x^*)$  von  $x^*$ , wie immer).

Dazu nehmen wir an, dass die ersten  $p-1$  Ableitungen von  $\phi$  am Fixpunkt Null sind, also  $\phi^{(k)}(x^*) = 0$  für  $k = 1, 2, \dots, p-1$ . Dann können wir Taylor-entwickeln für  $x \in U(x^*)$

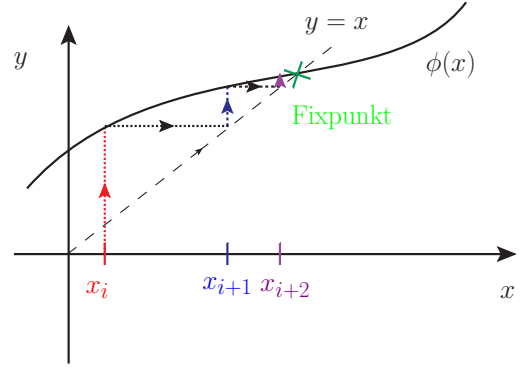
$$\begin{aligned} \phi(x) - x^* &= \phi(x^*) + \phi^{(1)}(x^*) (x - x^*) + \dots \\ &+ \frac{1}{(p-1)!} \phi^{(p-1)}(x^*) (x - x^*)^{p-1} + \frac{1}{p!} \phi^{(p)}(x^*) (x - x^*)^p \\ &+ \dots - \phi(x^*) \\ &= \frac{1}{p!} \phi^{(p)}(x^*) (x - x^*)^p + o(|x - x^*|^p). \end{aligned} \quad (22)$$

In Gl. (22) bleibt nur der Term mit  $\phi^{(p)}$  übrig, weil die Ableitungen vorher alle bei  $x^*$  Null sind. Dazu bleibt der Restterm, den wir mit  $o(|x - x^*|^p)$  angeben, das heißt der Restterm geht schneller gegen Null als die Potenz  $|x - x^*|^p$ , nämlich

$$\lim_{x \rightarrow x^*} \frac{\phi(x) - x^*}{(x - x^*)^p} = \frac{\phi^{(p)}(x^*)}{p!}. \quad (23)$$

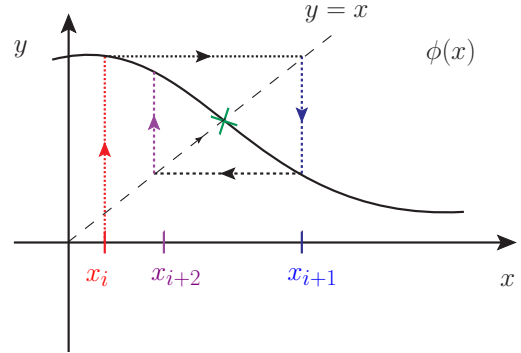
Aus Gl. (23) bekommen wir die Ordnung der Konvergenz  $p$ . Falls  $p = 1$  ist, dann müssen wir zusätzlich  $|\phi^{(1)}(x^*)| < 1$  haben.

*Beispiel lineare Konvergenz* Wir können hier noch unterscheiden nach dem Vorzeichen von  $\phi^{(1)}(x^*)$ . Wenn  $0 < \phi^{(1)}(x^*) < 1$ , dann haben wir einen Iterationsverlauf wie im folgenden Plot angedeutet.



Die  $x_i$  konvergiert *monoton* (in Plot von unten) nach  $x^*$ .

Das Gegenstück liegt vor, wenn  $-1 \leq \phi^{(1)}(x^*) < 0$ . Dann konvergiert die Folge *alternierend* gegen  $x^*$ , angedeutet im folgenden Plot.



### Konvergenz bei Newton-Raphson

Wir wenden Gl. (22) auf die Newton-Raphson Iteration an, also  $\phi(x) = x - \frac{f(x)}{f'(x)}$  mit **einfacher Nullstelle**  $x^*$  von  $f$ , d.h.  $f(x^*) = 0$ , aber  $f'(x^*) \neq 0$ .

Dann ist also

$$\begin{aligned} \phi(x^*) &= x^*, \\ \phi'(x) &= 1 - \frac{f'(x)}{f'(x)} + \frac{f(x) f''(x)}{f'(x)^2} = \frac{f(x) f''(x)}{f'(x)^2}, \\ \phi'(x^*) &= 0. \end{aligned} \quad (24)$$

Dazu prüfen wir, dass  $\phi''(x^*) = \frac{f''(x^*)}{f'(x^*)^2} \neq 0$ .

Es gilt also  $p = 2$  und das Newton-Raphson-Verfahren ist *quadratisch konvergent*.

Zum Vergleich: Beim Bisektionsverfahren hatten wir  $|x_{i+1} - x^*| \leq \frac{1}{2} |x_i - x^*|^1$ , das heißt das Bisektionsverfahren ist linear konvergent, und damit langsamer (wenn nahe bei  $x^*$ ), also das Newton-Raphson. Andererseits ist das Bisektionsverfahren numerisch völlig stabil. Ist ja auch was.

Wir haben vorhin angenommen, dass  $x^*$  eine einfache Nullstelle ist. Sehen wir mal den Unterschied, wenn  $x^*$  eine  $m$ -fache Nullstelle ist, d.h.  $f(x^*) = 0$  und  $f^{(k)}(x^*) = 0$  für  $k = 1, \dots, m-1$ . Es ist dann erst  $f^{(m)}(x^*) \neq 0$ .

Dann können wir  $f$  schreiben als  $f(x) = (x - x^*)^m g(x)$  mit  $g(x^*) \neq 0$ . Die Ableitung ist dann

$$f'(x) = m(x - x^*)^{m-1} g(x) + (x - x^*)^m g'(x) \quad (25)$$

und die Iterationsfunktion  $\phi$  folgt durch Einsetzen in die Newton-Raphson-Vorschrift

$$\begin{aligned} \phi(x) &= x - \frac{f(x)}{f'(x)} = x - \frac{(x - x^*) g(x)}{m g(x) + (x - x^*) g'(x)}, \\ \phi'(x) &= 1 - \frac{g(x)}{m g(x) + (x - x^*) g'(x)} \\ &\quad + (x - x^*) g'(x) + (x - x^*) \cdot \text{irgendwas}. \end{aligned} \quad (26)$$

Wir sehen also, dass im Fall einer  $m$ -fachen Nullstelle  $\phi'(x^*) = 1 - 1/m \neq 0$  und das Newton-Raphson-Verfahren konvergiert nur noch linear, nicht mehr quadratisch.

### Code-Beispiel

Im Programm `nst-comp.c` ist das Newton-Verfahren für die Wurzelfunktion angewendet,  $x = \sqrt{z}$ . Wir

führen das in eine Nullstellengleichung über und nehmen  $f(x) = x^2 - z$  mit  $f'(x) = 2x$ .

$f$  und  $f'$  sind in `func` implementiert. Die Funktion gibt einen zusammengesetzten Datentyp zurück, ein `struct`, das wir in

```
typedef struct {
    double u;
    double v;
} double2;
```

definieren. Eine Variable vom Typ `double2` enthält zwei `double` Variablen, den Wert der Funktion und die Ableitung der Funktion.

Die Iteration ist in der Funktion `newton_raphson` implementiert, insbesondere

```
double const x = x0 - y.u / y.v;
```

wobei `y.u` der Funktionswert ist und `y.v` die Ableitung bei  $x$ .

Vor der Iteration prüfen wir noch

```
if ( fabs( y.v ) < eps )
```

und brechen ab, wenn  $f'(x_0)$  zu klein ist, wir also Gefahr laufen, in der Nähe eines Extremums zu sein.

*Zum Abschluss.—*

Wie immer Dank im Voraus für Hinweise und Kommentare.

Mit [1] und [2] auf dem Weg.

---

[1] R. Freund and R. Hoppe, *Stoer/Bulirsch: Numerische Mathematik 1*, Springer-Lehrbuch (Springer Berlin Heidelberg, 2007).

[2] H. Schwarz and N. Köckler, *Numerische Mathematik* (Vieweg+Teubner Verlag, 2011).