Zur Numerik nichtlinearer Gleichungssysteme (Teil 2)

Werner Vogt
Technische Universität Ilmenau
Institut für Mathematik
Postfach 100565
98684 Ilmenau

Ilmenau, den 10.02.2004

Zusammenfassung Nichtlineare Gleichungssysteme $f(x) = 0$ sind in praktischen Anwendungen oft nicht durch arithmetische Ausdrücke für f verfügbar, sondern selbst das Ergebnis eines aufwendigen Näherungsverfahrens. Betrachtet werden deshalb Varianten des Newton-Verfahren, die die Jacobimatrizen approximieren und mit wenigen Funktionsberechnungen auskommen. Eine Vergrößerung des Einzugsbereiches der Lösungen kann mittels gedämpfter Newton-Verfahren erreicht werden. Hängt das gegebene System von Parametern ab, so bieten effiziente Fortsetzungstechniken gute Approximationen der gesuchten Lösungen.

1 Modifikationen des Newton-Verfahrens

In zahlreichen mathematischen Modellen tritt als zentrales Problem die Lösung nichtlinearer endlichdimensionaler Gleichungssysteme auf. Eine oft anzutreffende Standardaufgabe kann durch das reelle System

$$f(x) \;=\; 0 \qquad \text{mit} \quad f:\; D \subset \mathbb{R}^n \to\; \mathbb{R}^n\,,\; D \text{ offen} \tag{1}$$

mit dem nichtleeren Definitionsbereich D für die n reellen Variablen $x = (x_1, x_2, \ldots, x_n)$ beschrieben werden. Gesucht sind Vektoren $x_* \in D$, für die $f(x_*) = 0$ gilt. Wir wollen stets voraussetzen, daß eine Lösung $x_* \in D$ existiert und regulär ist.

Definition 1 (Reguläre Lösung)
Eine Lösung $x_* \in D$ heißt regulär (isoliert), wenn

- *eine Kugel $S = S(x_*, \delta_*) = \{x \mid ||x - x_*|| \le \delta_*\}$ um x_* mit $S \in int(D)$ existiert,*
- *die Jacobi-Matrix $F(x) = f'(x)$ auf S Lipschitz-stetig ist und*
- *die Jacobi-Matrix $F(x_*)$ an der Lösung regulär ist.*

Die Regularität der Nullstelle stellt eine *Standardvoraussetzung* an das zu lösende Problem dar. Bei singulärer Jacobi-Matrix $F(x_*)$ ist das Verhalten des Newton-Verfahrens überaus kompliziert: Während in wenigen Fällen noch eine lineare Konvergenz eintritt (z.B. bei skalaren Nullstellenproblemen), versagt das Verfahren in höherdimensionalen Systemen häufig.
Eine Lösung x_* ist *geometrisch isoliert*, falls eine Umgebung S existiert, in der keine weitere Lösung $x_+ \neq x_*$ liegt. Die Regularität einer Nullstelle darf deshalb nicht mit deren *geometrischer Isoliertheit* verwechselt werden.[1] Aus der Regularität folgt im übrigen stets die geometrische Isoliertheit einer Lösung.
Nichtlineare Gleichungssysteme in der Standardform (1) bilden oft die zentrale Teilaufgabe einer umfangreichen mathematischen Problematik. Allerdings treten sie in den wenigsten praktischen Anwendungen mit explizit durch arithmetische Ausdrücke gegebenen Funktionen f auf. Meist müssen die Funktionen f durch Näherungsverfahren mit hohem zeitlichen "Funktionsaufwand" berechnet werden. Daß eine analytische Darstellung der Jacobimatrix $f'(x)$ zwar theoretisch verifizierbar, aber praktisch mit vertretbarem Aufwand nicht bestimmbar ist, kommt noch hinzu. Betrachten wir dazu ein typisches Beispiel:

[1] Der häufig anzutreffende Begriff „Isoliertheit" wird deshalb hier nicht benutzt

Beispiel 2 Gegeben ist die nichtlineare Differentialgleichung 2. Ordnung

$$\ddot{x} - \varepsilon(1 - x^2 - \dot{x}^2)\dot{x} + x + b(4x^3 - 3x) \;=\; B\cos 3t \tag{2}$$

mit reellen Parametern $b, B, \varepsilon > 0$, die ein *subharmonisch erregtes elektrisches Netzwerk* beschreibt. Gesucht sind periodische Lösungen der vorgegebenen Periode $T = 2\pi$ für verschiedene Parameterkonstellationen b, B, ε aus dem Arbeitsbereich des elektrischen Netzwerkes. Mit der Periodizitätsbedingung

$$x(0) = x(T), \quad \dot{x}(0) = \dot{x}(T) \tag{3}$$

ist für feste Parameter ein Randwertproblem auf dem Intervall $[0, T]$ zu lösen. Eine naheliegende Idee betrachtet (2) als Anfangswertproblem mit unbekannten Anfangswerten $x(0) = s_1,\ \dot{x}(0) = s_2$ und gewinnt eine - desweiteren als existent vorausgesetzte - Lösung

$$y(t; s) = (x(t; s), \dot{x}(t; s))^T \quad \text{mit dem Vektor der Startwerte} \quad s = (s_1, s_2)^T$$

durch numerische Integration der Differentialgleichung. Da die Periodizitätsbedingung $y(0; s) = y(T; s)$ durch diese Lösung anfangs gewiß nicht erfüllt wird, definieren wir eine Defektfunktion

$$f(s) \;:=\; y(0; s) - y(T; s)\,, \quad f : \mathbb{R}^2 \to \mathbb{R}^2\,,$$

für die nunmehr Nullstellen $s_* \in \mathbb{R}^2$ gesucht sind. Denn die Lösungen des nichtlinearen Gleichungssystems $f(s) = 0$ entsprechen offenbar in eineindeutiger Weise den gesuchten Anfangswerten und damit den $2\pi-$periodischen Lösungen $y(t; s_*)$ der betrachteten Differentialgleichung. Dieser als *Schießverfahren (engl.: shooting method)* bezeichnete Zugang reduziert das Periodizitätsproblem für Differentialgleichungen auf ein Gleichungssystem mit 2 Unbekannten in $\mathbb{R}^2$. ◀

Im Gegensatz zu einfachen Beispielgleichungen ist das nichtlineare System $f(s) = 0$ nun durch folgende Eigenschaften charakterisiert:

1. $f(s)$ ist nicht durch einen exakten arithmetischen Ausdruck definiert, sondern entsteht als Resultat eines numerischen Integrationsverfahrens für Differentialgleichungen. Damit ist auch keine Jacobimatrix $f'(s)$ verfügbar und das Newton-Verfahren für die Aufgabe (1)
$$x_{k+1} \;=\; x_k - [f'(x_k)]^{-1} f(x_k)\,, \quad k = 0, 1, 2, 3, \ldots. \tag{4}$$
nicht anwendbar.

2. Der zeitliche Aufwand zur Berechnung eines einzigen Funktionswertes $f(s)$ beträgt ein Vielfaches des algebraischen Aufwandes zur Lösung des linearen Gleichungssystems pro Newtonschritt (Bei 100 Integrationsschritten des klassischen Runge-Kutta-Verfahrens mit je 4 Berechnungen der Cosinusfunktion pro Schritt in (2) sind wegen des Aufwandes von ca. 25 Gleitpunktoperationen (flops) pro cosinus-Wert bereits insgesamt 10000 Zeiteinheiten erforderlich. Das lineare Gleichungssystem mit 2 Unbekannten benötigt dagegen etwa 10 flops, so daß ein Zeitverhältnis des Funktionswertaufwandes zum algebraischen Aufwand von über 1000 : 1 entsteht). Damit sind überlinear konvergente Verfahren gefragt, die mit sehr geringer Zahl von Iterationsschritten und wenigen f-Werten auskommen.

3. Für viele Parameterwerte in Aufgabe (2) ist die Lage der periodischen Lösungen praktisch nicht bekannt, so daß keine Startlösungen für (s_1, s_2) verfügbar sind, die den Erfolg lokal konvergenter Verfahren garantieren. Ein möglichst großer „Einzugsbereich“ der gesuchten Lösungen s_* ist deshalb unabdingbar.

1.1 Newton-Verfahren mit Differenzenquotienten

Häufig sind die Ableitungen von f nur mit erheblichem Aufwand exakt zu berechnen. Dann kann man das im $\mathbb{R}^1$ bekannte Sekantenverfahren verallgemeinern und approximiert die Jacobi-Matrix $f'(x_k)$ durch eine leichter zu berechnende Matrix $A_k \in \mathbb{R}^{n\times n}$ mittels Differenzenquotienten

$$x_{k+1} = x_k - A_k^{-1} f(x_k) \quad \text{mit} \quad A_k = \nabla f(x_k, h)\,, \quad k = 0, 1, 2, ... \tag{5}$$

und einem geeigneten Schrittweitenvektor $h = (h_1, ..., h_n)^T$, $h_j > 0$. Derartige Verfahren bezeichnet man oft als Newton-ähnlich (engl.: Newton-like), wenn die erzeugten Folgen $\{x_k\}$, $k = 0, 1, 2, \ldots,$ das Verhalten der Newton-Folge aus Verfahren (4) approximieren. Besitzt die Gleichung $f(x) = 0$ die reguläre Lösung $x_* \in D \subset \mathbb{R}^n$, so definiert man genauer:

Definition 3 (Newton-ähnliches Verfahren)
Eine Näherungsfolge $\{x_k\}$, $k = 0, 1, 2, \ldots,$ *mit* $x_k \neq x_*$ *in einer Umgebung* S *von* x_* *heißt Newton-ähnlich, wenn*

$$\lim_{k\to\infty} \frac{\|f(x_k) + f'(x_k)(x_{k+1} - x_k)\|}{\|x_{k+1} - x_k\|} = 0$$

gilt. Ist jede Folge mit $x_0 \in S$ *Newton-ähnlich, so bezeichnet man das Verfahren selbst als Newton-ähnlich.*

Für das Newton-Verfahren selbst trifft die Definition offenbar zu. Man kann unter der Voraussetzung einer regulären Lösung x_* leicht nachweisen, daß ein Verfahren genau dann überlinear konvergiert, wenn es Newton-ähnlich ist.

Betrachten wir nun Verfahren der Gestalt (5) und stellen die Matrix A_k durch Differenzenquotienten dar. Als einfachste Approximation A_k bietet sich die Einpunkt-Approximation mittels der Vorwärts-Differenzenquotienten von $f'(x)$

$$\{\nabla f(x, h)\}_{ij} = \frac{1}{h_j}[f_i(x_1, ..., x_j + h_j, ..., x_n) - f_i(x)]\,, \quad i, j = 1...n$$

an, die mit dem j-ten Einheitsvektor e_j spaltenweise als

$$\nabla f(x, h)e_j = \frac{1}{h_j}[f(x + h_j e_j) - f(x)]\,, \quad j = 1...n \tag{6}$$

notiert werden kann. Zuerst ist zu klären, unter welchen Bedingungen das entstehende Verfahren (5) durchführbar ist und die erzeugte Folge der Näherungen $\{x_k\}$, $k = 0, 1, 2, \ldots$ gegen eine Lösung x_* konvergiert. Da anstelle der einfachen Differenzenquotienten (6) auch andere Approximationen denkbar sind, soll folgende Verallgemeinerung eingeführt werden:

Definition 4 (Streng konsistente Approximation)
$f : D \subset \mathbb{R}^n \to \mathbb{R}^n$ sei auf $D_o \subset int\ D$ differenzierbar. Die Schrittweitenmenge sei

$$H = \{h = (h_1, h_2, \ldots, h_n) \in \mathbb{R}^n \mid h_i \neq 0,\ i = 1, ..., n,\ \|h\| \leq r\}\ .$$

Dann heißt die Abbildung $\nabla f : D_o \times H \to \mathbb{R}^{n\times n}$ streng konsistente Approximation für $f'(x)$ auf D_o mit dem Diskretisierungsbereich $H \subset \mathbb{R}^n$, wenn $0 \in \mathbb{R}^n$ Häufungspunkt von H ist und eine Konstante $C > 0$ existiert, so daß

$$\|\nabla f(x, h) - f'(x)\| \leq C \cdot \|h\| \tag{7}$$

für alle $(x, h) \in D_o \times H$ gilt.

Mit der Voraussetzung $f \in C^2(\mathbb{R}^n)$ erfüllt unsere Differenzenapproximation offenbar die Bedingung (7), d.h. $\lim\limits_{h\to 0} \nabla f(x, h) = f'(x)$ mit Ordnung 1 in h. In [9] wird die Konvergenz des Verfahrens (5) mit folgendem Satz nachgewiesen:

Satz 5 (Konvergenz)
$f : D \subset \mathbb{R}^n \to \mathbb{R}^n$ besitze die reguläre Lösung $x_* \in D$, und $\nabla f : D_0 \times H \to \mathbb{R}^{n\times n}$ sei eine streng konsistente Approximation von f' auf einer Umgebung D_0 von x_*. Dann existieren Konstanten $\delta_0 > 0$ und $\varepsilon > 0$, so daß folgende Behauptungen gelten:

(i) Das Verfahren(5) ist für jedes $x_0 \in S_0 := S(x_*, \delta_0)$ durchführbar, und $x_k \in S_0$ für $k = 0, 1, 2, ...$, wenn in jedem Schritt $h_k \in H$ so gewählt wird, daß $\|h_k\| < \varepsilon$ garantiert ist.

(ii) Die Folge $\{x_k\}$ konvergiert mindestens linear gegen x_*.

(iii) Falls zusätzlich $\lim\limits_{k\to\infty} h_k = 0$ gilt, so ist die Konvergenz der x_k überlinear. □

Damit ist das Verfahren Newton-ähnlich im Sinne der gegebenen Definition, falls die Schrittweitenfolge $\{h_k\}$ gegen Null konvergiert. In praxi können allerdings wegen der begrenzten Stellenzahl der benutzten Gleitpunktzahlen die Schrittweiten h_j der Differenzenapproximation (6) nicht beliebig nahe bei Null gewählt werden, sondern müssen auf jeden Fall größer als die Maschinengenauigkeit ε_M sein. Damit erfüllt die reale Schrittweitenmenge H_{real} nicht die Voraussetzung aus Definition 4 und der Konvergenzsatz ist nur bedingt aussagekräftig.

Beantworten wir deshalb die Frage, wie die Diskretisierungsschrittweiten h_j passend zu wählen sind. Neben dem Diskretisierungsfehler $f'(x) - \nabla f(x, h)$ sind dazu nun auch andere Fehlerquellen, insbesondere Rundungsfehler, zu berücksichtigen. Werde anstelle von $f(x)$ der fehlerbehaftete Funktionswert $\tilde{f}(x) = f(x) + \varepsilon(x)\ ,\ ||\varepsilon(x)|| < \varepsilon_M$ mit der Maschinengenauigkeit ε_M benutzt, so erhält man mit der Voraussetzung $f \in C^2(\mathbb{R}^n)$ für den gesamten Approximationsfehler der Differenzenapproximation die Abschätzung

$$\begin{aligned}
||f'(x)e_j - \nabla\tilde{f}(x, h)e_j|| &= ||f'(x)e_j - \frac{1}{h_j}[\tilde{f}(x + h_j e_j) - \tilde{f}(x)]|| \\
&\leq ||f'(x)e_j - \frac{1}{h_j}[f(x + h_j e_j) - f(x)]|| + \frac{2\max||\varepsilon(x)||}{h_j} \\
&\leq Ch_j + \frac{2\varepsilon_M}{h_j}
\end{aligned}$$

mit einer funktionsabhängigen Konstanten $C > 0$. Minimierung der rechten Seite über h_j ergibt den Wert $h_j = \sqrt{2\varepsilon_M/C}$, womit $h = O(\sqrt{\varepsilon_M})$ gilt. Um für betragsgroße Werte x_j auch entsprechend große Diskretisierungsschrittweiten h_j zu erhalten, hat sich eine kombinierte Absolut-Relativ-Wahl

$$h_j = \sqrt{\varepsilon_M}\,(1 + |x_j|)\,, \quad j = 1(1)n \tag{8}$$

bewährt. Damit wird zugleich garantiert, daß für Werte von x_j nahe Null keine Stellenauslöschung in den Differenzenquotienten erfolgt. Das entstehende Verfahren ist allerdings nun nicht Newton-ähnlich im engeren Sinne der Definition.

Algorithmus 6 (Newton-Verfahren mit Differenzen)

Function newtondiff $(f, x, tolabs, tolrel)$

1. Berechne Toleranz $tol = tolrel \cdot ||f(x)|| + tolabs$
2. Do while $||f(x)|| > tol$
   1. Wähle $h_j = \sqrt{\varepsilon_M}\,(1 + |x_j|)\,,\ j = 1(1)n$
   2. Approximiere die Jacobi-Matrix durch A_k mit
      $$A_k e_j = \frac{1}{h_j}[f(x + h_j e_j) - f(x)],\ j = 1(1)n$$
   3. Löse $A_k \cdot d = -f(x)$ nach d
   4. $x = x + d$
   5. Berechne $f(x)$
3. Return x

Der Algorithmus `newtondiff` erfordert als Input nur die Funktion f, einen Startwert x und die (absolute und relative) Toleranz $tolabs, tolrel$. Als Maschinengenauigkeit ε_M für double-Gleitpunktzahlen liefert MATLAB mit der Standardkonstanten `eps` den Wert $2.2204 \cdot 10^{-16}$. Implementiert man diesen Algorithmus als MATLAB-Funktion, so kann man nun auf den Parameter `Dfname` für die Jacobimatrix verzichten.

Beispiel 7 Wir betrachten das bereits in [11] getestete Gleichungssystem

$$\begin{aligned} f_1(x_1, x_2) &= 2x_1^3 - x_2^2 - 1 &&= 0 \\ f_2(x_1, x_2) &= x_1 x_2^3 - x_2 - 4 &&= 0 \end{aligned}$$

und rufen die MATLAB-Funktion `newtondiff` mit verschiedenen Startpunkten und Genauigkeiten auf:

```
[Loesung,Residuum,Iterationen] =
    newtondiff(@bsp12, [1.2, 1.7]',1e-6,1e-6,10)
[Loesung,Residuum,Iterationen] =
    newtondiff(@bsp12, [30, 20]',1e-12,1e-12,25)
```

```
function [x,res,iter] = newtondiff(fname, x0,
                         tolabs,tolrel,maxit);
% Algorithmus 1.1 : Newton-Verfahren
%                   mit Differenzenquotienten
% ****************************************
%
iter = 0;  x = x0; fx = feval(fname,x);
tol= tolrel * norm(fx) + tolabs;
% Iterationszyklus
while (norm(fx) > tol) & (iter < maxit)
    h =  sqrt(eps)*(abs(x)+1);
    % Approximation der Jacobimatrix
    for i   = 1 : length(x),
        y   = x; y(i) = y(i) +h(i);
        fy  = feval(fname,y);
        A(:,i) = (fy-fx)/h(i);
    end
    d   = - A \ fx;
    x   = x + d;
    fx  = feval(fname,x);
    iter= iter + 1;
end  % while
res = norm(fx);
% ****************************************
```

Das ableitungsfreie Verfahren liefert dieselben Lösungen wie das Newton-Verfahren und benötigt dafür genauso viele Iterationsschritte - allerdings mit geringerem analytischen Aufwand:

```
Loesung =                      Loesung =
    1.234274484114498e+000         1.234274484362242e+000
    1.661526466795909e+000         1.661526467418011e+000
Residuum =                     Residuum =
    3.132478868770743e-013         6.876283586488347e-009
Iterationen =                  Iterationen =
     3                             13
```

1.2 Gedämpfte Newton-Verfahren und globale Konvergenz

Die bisher behandelten Verfahren sind lokal konvergent, da unter geeigneten Voraussetzungen (Glattheit, Regularität) stets eine Konvergenzumgebung S der Nullstelle x_* der Gleichung

$$f(x) \;=\; 0 \;, \quad f : D \subset \mathbb{R}^n \to \mathbb{R}^n \;, \quad D \text{ offen} \tag{9}$$

existiert. Über die Lage und Größe der Umgebung S ist in praxi meist nichts bekannt. Um den semilokalen Konvergenzsatz von L.V.Kantorovics (vgl. [11]) anwenden zu können,

sind schwer nachzuvollziehende Abschätzungen in der Umgebung des Startwertes x_0 durchzuführen, der zudem nahe der Lösung x_* liegen sollte.
Die Menge $S(x_*)$ aller Startwerte x_0, für die das jeweils betrachtete Verfahren gegen die Nullstelle x_* konvergiert bildet den Einzugsbereich von x_*. Um ihn zu vergrößern, kann man zur Bestimmung der Nullstelle x_* von f auch das zugehörige Minimierungsproblem

$$\psi(x) \;=\; \frac{1}{2}\|f(x)\|^2 \quad \Rightarrow \; Min! \quad \text{über} \quad x \in D \subset \mathbb{R}^n \,, \tag{10}$$

mit der Euklidischen Norm $\|x\|$ betrachten. Gesucht ist dann eine Minimalstelle x_* mit $\psi(x_*) = 0$. Ausgehend von einer k-ten Näherung x_k läßt sich der neue Wert $x_{k+1} = x_k + d_k$ mit der Newton-Richtung (bei Minimierungsverfahren auch „Such-Richtung“ genannt) $d_k = -f'(x_k)^{-1} f(x_k)$ ermitteln und anschließend kontrollieren, ob die *einfache Abstiegsbedingung*

$$\psi(x_{k+1}) \;<\; \psi(x_k)$$

für ψ erfüllt ist. Andernfalls kann man den Korrekturvektor d_k mit einem Faktor $\lambda_k \in (0, 1]$ dämpfen und mit dem zurückgesetzten Testpunkt

$$x_{k+1} = x_k + \lambda_k \cdot d_k \quad \text{mit} \quad d_k = -f'(x_k)^{-1} f(x_k) \tag{11}$$

die Abstiegsbedingung erneut überprüfen. Nach eventuell weiteren erforderlichen Reduzierungen $\lambda_k := \lambda_k \cdot \alpha$ mit dem fest gewählten Skalierungsfaktor $\alpha \in (0, 1)$ liefert der erste akzeptierte λ_k-Wert den *Dämpfungsfaktor* des *gedämpften Newton-Verfahrens* (11). Häufig wird eine sukzessive Halbierung mit $\alpha = 0.5$ vorgenommen.

Algorithmus 8 (Gedämpftes Newton Verfahren)

Function newtonarmijo$(f, F, x, tolabs, tolrel)$

1. Berechne Toleranz $tol := tolrel \cdot ||f(x)|| + tolabs$
2. Wähle Konstanten $\delta \in (0, \frac{1}{2}),\ \alpha \in (0, 1)$ und $\lambda := 1$
3. Do while $||f(x)|| > tol$
   1. Wähle $\lambda := \min(\lambda/\alpha, 1)$
   2. Berechne Jacobi-Matrix $F(x)$
   3. Berechne Newton-Richtung d aus $F(x) \cdot d = -f(x)$
   4. Berechne Testpunkt $y := x + \lambda d$
   5. Do while $\|f(y)\|^2 \;>\; (1 - 2\delta\lambda)\|f(x)\|^2$
      1. Reduziere $\lambda := \lambda \cdot \alpha$
      2. Berechne Testpunkt $y := x + \lambda d$
   6. Aktualisiere $x := y$
4. Return $x,\ f(x)$

Bei Minimierungsverfahren hat es sich bewährt, diese einfache Abstiegsbedingung durch eine Bedingung für *hinreichenden Abstieg*

$$\psi(x_{k+1}) \;<\; (1 - 2\delta\lambda_k)\,\psi(x_k)\,, \quad \text{mit einer Konstanten} \quad \delta \in (0, \frac{1}{2}) \tag{12}$$

zu ersetzen. Offenbar ist $1 - 2\delta\lambda_k \in (0, 1)$, wobei Werte $\delta = 10^{-2}, 10^{-4}$ üblich sind. Der erste λ_k-Wert, mit dem diese Bedingung erfüllt ist, liefert ein gedämpftes Newton-Verfahren (11) mit der sogenannten *Armijo–Regel* (12). Eine detaillierte Begründung dieser und weiterer verbesserter Strategien findet man in [4].

Notieren wir den zugehörigen Algorithmus `newtonarmijo`, der die Funktion f, die Jacobimatrix F, einen Startwert x und die (absolute und relative) Toleranz $tolabs, tolrel$ voraussetzt.

Wird der Dämpfungsfaktor λ in jedem Newton-Schritt mit dem Wert $\lambda = 1$ neu initialisiert, so wird das Verfahren wegen ständiger Reduktionen oft über mehrere Iterationen hinweg ineffizient. Besser ist die im Schritt 3.1 des Algorithmus implementierte Strategie, nach jedem gedämpften Newton-Schritt den aktuellen Faktor vorsichtig auf λ/α zu vergrößern, ohne dabei den Wert 1 zu überschreiten.

Offenbar ist das gedämpfte Newton-Verfahren im allgemeinen nur linear konvergent, denn die Iterationsfunktion g lautet nun $g(x) = x - \lambda f'(x)^{-1} f(x)$, womit sich deren Ableitung an der Nullstelle x_* zu $G(x_*) = (1 - \lambda)I$ ergibt. Diese ist für $\lambda < 1$ verschieden von der Nullmatrix. Das Hauptziel, den Einzugsbereich von x_* zu vergrößern, kann mit dem gedämpften Newton-Verfahren jedoch oft erreicht werden. Ist $S \subset \mathbb{R}^n$ ein im allgemeinen großer *vorgegebener* Bereich, so heißt ein Iterationsverfahren *global konvergent auf* S, falls es für jeden Startpunkt $x_0 \in S$ gegen einen Fixpunkt x_* in S konvergiert. Kann die Regularität der Jacobimatrix $f'(x)$ auf einer ganzen Niveaumenge vorausgesetzt werden, so läßt sich die globale Konvergenz des gedämpften Newton-Verfahrens für alle Startwerte aus dieser Menge nachweisen (vgl. [9] zum aufwendigen Beweis des Satzes):

Satz 9 (Globale Konvergenz)
Sei $f : D \subset \mathbb{R}^n \to \mathbb{R}^n$ 2-mal stetig differenzierbar auf der offenen Menge D. Zu einem festem $y \in D$ werde die Niveaumenge

$$S \;=\; \{x \in D \mid \|f(x)\| \;\leq \|f(y)\| \,\}$$

definiert, die als kompakt vorausgesetzt wird. Ist die Jacobimatrix $f'(x)$ für alle $x \in S$ invertierbar, so gelten folgende Behauptungen:

(i) Das gedämpfte Newton- Verfahren (11) mit Armijo-Regel (12) ist für jedes $x_0 \in S$ durchführbar mit $x_k \in S$ und streng monoton fallenden Funktionswerten $\|f(x_k)\|$.

(ii) Die Folge $\{x_k\}$ konvergiert gegen eine Nullstelle x_* von f in der Menge S.

(iii) Falls das Verfahren nicht nach endlich vielen Schritten mit einer Nullstelle endet, so existiert ein $K \in \mathbb{N}$, bei dem das Verfahren für $k > K$ in das gewöhnliche Newton-Verfahren übergeht und Q-quadratische Konvergenz eintritt. □

Die hier eingeführte Dämpfungsstrategie läßt sich nunmehr auch erfolgreich auf das vereinfachte Newton-Verfahren, das Shamanskii-Verfahren in [11] und das Newton-Verfahren mit Differenzenquotienten anwenden und eine globale Konvergenz erreichen. Die komplizierte Konvergenztheorie übersteigt jedoch den Rahmen dieser Darlegung.

1.3 Quasi-Newton-Verfahren

In vielen Anwendungsfällen ist die Berechnung der Funktion $f : D \subset \mathbb{R}^n \to \mathbb{R}^n$ der aufwendigste Verfahrensteil und verursacht außer unvermeidbaren *algebraischen Kosten* hohe *Funktionswertkosten* in den bisher vorgestellten Verfahren. Weder das quadratisch konvergente Newton-Verfahren mit zeitintensiver Berechnung der Jacobimatrix $f'(x_k)$, noch eine Differenzenapproximation A_k mit $n+1$ f-Berechnungen pro Iteration sind dann zu empfehlen. Das vereinfachte Newtonverfahren andererseits ist wegen seiner linearen Konvergenz meist nicht konkurrenzfähig.

Ziel unserer Überlegung ist deshalb ein Verfahrenstyp, der mit geringem Funktionsaufwand eine passable Approximation B_k der Jacobimatrix $f'(x_k)$ liefert und dennoch *überlinear* konvergiert. Das von C.G. Broyden 1965 entwickelte Verfahren benutzt lediglich die im k-ten Iterationsschritt berechneten Verfahrensgrößen, um die Approximationsmatrix B_k geeignet zur Matrix B_{k+1} des folgenden Schrittes aufzudatieren. Betrachten wir dazu 2 aufeinanderfolgende Näherungen x_k und x_{k+1} des Newtonverfahrens

$$x_{k+1} \;=\; x_k - f'(x_k)^{-1} f(x_k)\,. \tag{13}$$

Für den neuen Funktionswert $f(x_{k+1})$ erhält man mit dem Mittelwertsatz

$$f(x_{k+1}) - f(x_k) \;=\; \int_0^1 f'(x_k + t(x_{k+1} - x_k))\,dt\,(x_{k+1} - x_k)\,, \tag{14}$$

dessen Integralterm im nachfolgenden Iterationsschritt

$$x_{k+2} \;=\; x_{k+1} - B_{k+1}^{-1} f(x_{k+1}) \tag{15}$$

durch die Matrix B_{k+1} möglichst genau approximiert werden soll. Eine sinnvolle Bedingung hierfür stellt die sogenannte *Sekantengleichung*

$$B_{k+1}(x_{k+1} - x_k) \;=\; f(x_{k+1}) - f(x_k) \tag{16}$$

dar. Denn im Falle linearer Gleichungssysteme mit $f(x) = Ax - b$ ist $B_k = A$ bereits die exakte Newton-Matrix. Im eindimensionalen Fall liefert Gleichung (16) die Sekantenformel

$$B_{k+1} \;=\; \frac{f(x_{k+1}) - f(x_k)}{x_{k+1} - x_k}$$

und ist mit dem bekannten Sekantenverfahren zur Nullstellenbestimmung identisch.

In mehrdimensionalen Systemen werden jedoch die n^2 unbekannten Koeffizienten von B_{k+1} durch die n Sekantengleichungen nicht eindeutig bestimmt. Dem Ziel des Ansatzes gemäß soll zur Berechnung des B_{k+1} nur auf die im k-ten Schritt berechneten Verfahrenswerte zurückgegriffen werden, also auf die *Newton-Korrektur* $s_k = x_{k+1} - x_k$ und die *Funktionswert-Korrektur* $y_k = f(x_{k+1}) - f(x_k)$. Gesucht ist demzufolge eine *Aufdatierungsformel*

$$B_{k+1} \;:=\; \Phi(B_k, s_k, y_k)\,, \quad \Phi : D \subset \mathbb{R}^{n\times n} \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^{n\times n}\,, \tag{17}$$

die die gegebene Matrix B_k mit Hilfe der Vektoren s_k und y_k in jedem Iterationsschritt modifiziert.

Definition 10 *Jedes zusammengesetzte Iterationsverfahren*

$$\begin{aligned} x_{k+1} &= x_k - B_k^{-1} f(x_k) \\ B_{k+1} &= \Phi(B_k, s_k, y_k)\,, \quad k = 0, 1, 2, \ldots \end{aligned} \tag{18}$$

mit $s_k = x_{k+1} - x_k \quad y_k = f(x_{k+1}) - f(x_k)$ *heißt* Quasi-Newton-Verfahren*, wenn die Sekantengleichung*

$$B_{k+1} s_k = y_k \tag{19}$$

erfüllt ist.

Im mehrdimensionalen Fall lassen sich beliebig viele derartige Verfahren konstruieren, die weitere Eigenschaften der Jacobimatrix $f'(x_k)$ berücksichtigen, wie z.B. Symmetrie, positive Definitheit oder schwache Besetztheit. Wird jedoch keine spezielle Struktur vorausgesetzt, so stellt die Rang-1-Aufdatierung von Broyden das bekannteste Quasi-Newton-Verfahren dar. Ihm liegt die anschauliche Idee zugrunde, daß die beiden aufeinanderfolgenden Verfahrensmatrizen B_k und B_{k+1} dieselbe Wirkung haben sollen, wenn sie auf Vektoren $s \in \mathbb{R}^n$ angewendet werden, die orthogonal zur Newton-Korrektur s_k sind, d.h.

$$B_{k+1} s = B_k s\,, \quad \text{falls} \quad s \perp s_k^T \tag{20}$$

gilt. Hieraus folgt wegen $(B_{k+1} - B_k)s = 0$ für $s_k^T s = 0$, daß die Zeilenvektoren dieser Differenzmatrix orthogonal zu jedem Vektor s mit $s_k^T s = 0$ liegen, also Vielfache der Newton-Korrektur s_k^T sein müssen. Mit einem beliebigen Vektor $v \in \mathbb{R}^n$ läßt sich die Matrixdifferenz deshalb in der Form

$$B_{k+1} - B_k = v \cdot s_k^T \tag{21}$$

darstellen. Multiplikation mit s_k ergibt

$$(B_{k+1} - B_k) s_k = v \cdot s_k^T s_k\,,$$

woraus mit der Sekantengleichung (19)

$$y_k - B_k s_k = v \cdot s_k^T s_k\,,$$

also

$$v = \frac{y_k - B_k s_k}{s_k^T s_k}$$

folgt. Setzen wir diesen Ausdruck in (21) ein, so haben wir die Aufdatierungsformel des *Broyden-Verfahrens* gefunden:

$$\boxed{B_{k+1} = B_k + \frac{(y_k - B_k s_k) s_k^T}{s_k^T s_k}\,.} \tag{22}$$

Bemerkungen 11

1. Da die Aufdatierungsmatrix $v \cdot s_k^T$ für $s_k \neq 0$ den Rang 1 besitzt, liegt bei diesem Verfahren eine Rang-1-Modifikation von B_k vor. Rang-2-Aufdatierungen wurden u.a. von W.Rheinboldt [7] vorgeschlagen; für großdimensionale Systeme werden gegenwärtig auch Rang-(p+1)-Modifikationen entwickelt, in denen Broyden-Aufdatierungen in geeigneten Unterräumen benutzt werden (Broyden subspace iterations).

2. Die Aufdatierungsformel (22) läßt sich auch in der einfachen Form

$$B_{k+1} \;=\; B_k + \frac{f(x_{k+1})s_k^T}{s_k^T s_k} \tag{23}$$

notieren. Sie ist durch die Bedingungen (19) und (20) eindeutig bestimmt.

3. Formel (22) kann alternativ gewonnen werden, denn unter allen reellen Matrizen $B \in \mathbb{R}^{n\times n}$, die der Sekantengleichung genügen, löst B_{k+1} eindeutig die Minimierungsaufgabe

$$\|B - B_k\|_F \;\longrightarrow\; \text{Min !} \;, \quad \text{falls} \quad Bs_k \;=\; y_k$$

mit der Frobeniusnorm $\|A\|_F$. Weiteres findet man in der Literatur (vgl. [4], S.148).

Der Algorithmus `broyden` erfordert als Input lediglich die Funktion f, einen Startwert x und die Toleranzen $tolabs, tolrel$. Die Aufdatierung erfolgt mit der Darstellung (23).

Algorithmus 12 (Broyden-Verfahren)

Function broyden$(f, x, tolabs, tolrel)$

1. Berechne Toleranz $tol = tolrel \cdot ||f(x)|| + tolabs$
2. Berechne eine Approximation B der Jacobi-Matrix $F(x)$
   und $z = f(x)$
3. Do while $||z|| > tol$
   1. Löse $B \cdot s = -z$ nach s
   2. Berechne $x_{neu} = x + s$ und $f_{neu} = f(x_{neu})$
   3. Aktualisiere $B = B + (f_{neu} \cdot s^T)/(s^T s)$
   4. Aktualisiere $x = x_{neu},\ z = f_{neu}$
4. Return $x,\ z$

Stellt man das Broyden-Verfahren wie in Definition 10 durch $x_{k+1} = x_k - B_k^{-1} f(x_k)$ dar, so wird die inverse Matrix B_k^{-1} in jedem Verfahrensschritt benötigt. Hierfür liefert die *Sherman-Morrison-Formel* eine „inverse“ Aufdatierung mit nur $O(n^2)$ algebraischen Operationen

$$B_{k+1}^{-1} \;=\; B_k^{-1} + \frac{(s_k - B_k^{-1}y_k)s_k^T B_k^{-1}}{s_k^T B_k^{-1} y_k} \;. \tag{24}$$

Damit läßt sich der algebraische Aufwand des Algorithmus von bisher $O(n^3)$ Operationen für die LU-Zerlegung von B pro Iteration beträchtlich reduzieren. Dies kann bei großdimensionalen Gleichungssystemen zu beträchtlicher Effizienzsteigerung führen, vorausgesetzt, es existiert eine geeignete inverse Startmatrix B_0^{-1}.

Die Konvergenztheorie des Broyden-Verfahrens ist recht kompliziert. So approximieren zwar die Broyden-Matrizen B_k die exakten Jacobimatrizen $f'(x_k)$ mit einer Schranke für die Abweichung $\|B_k - f'(x_k)\| \leq \varrho$. Im Gegensatz zu den überlinear konvergenten Verfahren mit Differenzenquotienten gilt dagegen nicht allgemein

$$\lim_{k\to\infty} \|B_k - f'(x_k)\| = 0\,.$$

Dennoch läßt sich unter geeigneten Startbedingungen an das Verfahren lokal überlineare Konvergenz nachweisen. Genauer gilt folgender

Satz 13 (Lokale Konvergenz)
$f : D \subset \mathbb{R}^n \to \mathbb{R}^n$ besitze die reguläre Lösung $x_* \in D$. Dann existieren Konstanten $\delta_0 > 0$ und $\varrho_0 > 0$, so daß folgende Behauptungen gelten:

(i) Das Broyden-Verfahren (18), (22) ist für jedes $x_0 \in S_0 := S(x_*, \delta_0)$ und jede Matrix B_0 mit $\|B_0 - f'(x_0)\| \leq \varrho_0$ durchführbar, und $x_k \in S_0$ für $k = 0, 1, 2, \ldots$.

(ii) Entweder das Verfahren endet nach endlich vielen Schritten wegen $f(x_k) = 0$ mit $x_k = x_*$ oder die Folge $\{x_k\}$ konvergiert Q-überlinear gegen x_*. □

Den ausführlichen Beweis dieses Satzes findet man in [9], S.140ff.
In [4] wird die Konvergenz allgemeiner Aufdatierungsverfahren gezeigt und es werden Verfahren für weitere spezielle Klassen (Powells symmetrische Broyden-Formel für symmetrische Jacobimatrizen, Methoden für schwachbesetzte Matrizen) begründet.

1.4 Inexakte Newton–Verfahren

Großdimensionale nichtlineare Gleichungssysteme mit Tausenden von Gleichungen treten direkt bei der Beschreibung großer elektrischer Netzwerke mit nichtlinearen Bauelementen auf, aber auch indirekt nach der Diskretisierung partieller Differentialgleichungen in der Gas- und Hydrodynamik sowie der Elektrodynamik. Setzt man wie in den bisherigen Verfahren in jedem Newtonschritt einen direkten Löser für das lineare Gleichungssystem (LR-Zerlegung, QR-Zerlegung, Cholesky-Zerlegung oder ähnliches) ein, so entstehen trotz ausgefeilter sparse-Matrix-Techniken sehr zeitaufwändige Algorithmen. Der Einsatz eines *Iterationsverfahrens* I liegt deshalb nahe, so dass nun innerhalb der Newton-Iteration N eine weitere "innere" Iteration entsteht, die stets nach endlich vielen Schritten abgebrochen werden muß. Für das so entstehende "inexakte" Newton-Verfahren sind Konvergenz und Konvergenzordnung, aber auch Effizienz und Robustheit zu garantieren.
Das Newton-Verfahren (4) benutzt zur Ermittlung der Lösung x_* die Iterationsvorschrift

$$x_{k+1} \;=\; x_k + s_k \quad \text{mit} \quad f'(x_k)s_k + f(x_k) \;=\; 0\,, \quad k = 0, 1, \ldots, \tag{25}$$

und nimmt die exakte Lösung der linearen Gleichungssysteme an. Dagegen verbleibt bei iterativer Lösung ein inneres Residuum r_k des k-ten Schrittes mit

$$f'(x_k)s_k + f(x_k) \;=\; r_k\,, \quad k = 0, 1, \ldots,$$

das eine Abbruchbedingung nach dem k-ten Iterationsschritt

$$\|r_k\|/\|f(x_k)\| \;\leq\; \eta_k$$

mit einer Konstanten $\eta_k \geq 0$ erfüllen soll. Diese Fehlerabschätzung für die innere Iteration in der Form

$$\|f'(x_k)s_k + f(x_k)\| \;\leq\; \eta_k\|f(x_k)\|\,, \quad k = 0, 1, \ldots, \tag{26}$$

nennt man *inexakte Newton-Bedingung* mit dem *Zwangsterm (forcing term)* η_k. Das so entstehende *inexakte Newton-Verfahren* genügt dann dem allgemeinen Verfahrensschema:

- Gib die Startlösung x_0 und die Folge $\{\eta_k\}$ vor.
- Ermittle für $k = 0, 1, 2, ...$ einen Vektor s_k, der

$$\|f'(x_k)s_k + f(x_k)\| \;\leq\; \eta_k\|f(x_k)\| \tag{27}$$

genügt und setze $x_{k+1} = x_k + s_k$.

Eine Beschränkung $\eta_k \in [0, 1)$ ist sinnvoll, wobei das Newton-Verfahren (25) den Spezialfall $\eta_k = 0,\ k = 0, 1, \ldots$ darstellt. Wir wollen nachfolgend klären,

- unter welchen Voraussetzungen an die Zwangsterme η_k diese Verfahren lokal gegen x_* konvergieren und
- welche Strategien zur Wahl der η_k auf effiziente, also möglichst überlinear konvergente Verfahren führen.

Wir betrachten die Gleichung

$$f(x) \;=\; 0\;, \quad f : D \subset \mathbb{R}^n \to \mathbb{R}^n\,, \quad D \text{ offen} \tag{28}$$

und setzen die Existenz einer regulären Lösung $x_* \in D$ gemäß Definition 1 voraus. Unter dieser Voraussetzung zeigt man leicht folgende Abschätzungen

Lemma 14 *$f : D \subset \mathbb{R}^n \to \mathbb{R}^n$ besitze die reguläre Lösung $x_* \in D$. Dann existiert eine Konstante $\delta_1 > 0$, so dass für allex $x \in S(x_*, \delta_1)$ gilt:*

$$\begin{aligned} ||f(x)|| &\leq 2||f'(x_*)|| \cdot ||x - x_*|| \\ ||f'(x)^{-1}|| &\leq 2||f'(x_*)^{-1}||\,. \end{aligned}$$

Wir betrachten eine mit dem inexakten Newton-Verfahren (27) erzeugte Folge $\{x_k\}$, für deren Fehler $e_k = x_k - x_*$ folgende Abschätzung nachgewiesen werden kann:

Lemma 15 *Die Lösung $x_* \in D$ sei regulär. Dann existieren Konstanten $\delta > 0$ und $K > 0$, so dass für jedes $x_k \in S(x_*, \delta)$ gilt:*

$$||e_{k+1}|| \;\leq\; K(||e_k|| + \eta_k)||e_k||\,. \tag{29}$$

BEWEIS: Sei $S(x_*, \delta_2)$ die nach Voraussetzung garantierte Umgebung, in der das Newton-Verfahren (25) Q-quadratisch konvergiert. Mit $\delta := \min(\delta_1, \delta_2)$ und obigem δ_1 schätzt man mit der inexakten Newton-Bedingung (26) und Lemma 14 ab:

$$\begin{aligned} ||s_k + f'(x_k)^{-1} f(x_k)|| &= ||f'(x_k)^{-1} \left(f'(x_k) s_k + f(x_k)\right)|| \\ &\leq ||f'(x_k)^{-1}|| \cdot \eta_k ||f(x_k)|| \\ &\leq 2\eta_k ||f'(x_*)^{-1}|| \cdot 2||f(x_*)|| \cdot ||e_k|| \\ &\leq 4K_1 \eta_k ||e_k||, \end{aligned} \tag{30}$$

wobei $K_1 > 0$ konstant ist. Für den Fehler e_{k+1} gewinnt man damit die Abschätzung

$$\begin{aligned} ||e_{k+1}|| &= ||x_{k+1} - x_*|| = ||x_k + s_k - x_*|| \\ &= ||x_k + s_k - x_* + f'(x_k)^{-1} f(x_k) - f'(x_k)^{-1} f(x_k)|| \\ &\leq ||x_k - f'(x_k)^{-1} f(x_k) - x_*|| + ||s_k + f'(x_k)^{-1} f(x_k)|| \\ &\leq ||[x_k - f'(x_k)^{-1} f(x_k)] - x_*|| + 4K_1 \eta_k ||e_k||. \end{aligned}$$

Der erste Summand gibt den Fehler der (exakten) Newton-Iterierten des Wertes x_k an, für den wegen der Q-quadratischen Konvergenz eine Konstante $K_2 > 0$ existiert, mit der

$$||e_{k+1}|| \leq K_2 ||e_k||^2 + 4K_1 \eta_k ||e_k|| \leq K(||e_k|| + \eta_k)||e_k||$$

folgt, wobei $K := \max(K_2, 4K_1)$ gewählt werden kann. □

Mit der Fehlerschätzung (29) sind wir nun imstande, die Konvergenz des inexakten Newton-Verfahrens zu begründen.

Satz 16 *Die Lösung $x_* \in D$ sei regulär. Dann existieren Konstanten $\delta > 0$ und $\eta_{max} > 0$, so dass für jedes $x_0 \in S(x_*, \delta)$ und alle Folgen $\{\eta_k\}_{k \in \mathbb{N}}$, $\eta_k \in [0, \eta_{max}]$ folgende Behauptungen gelten:*

(i) Das inexakte Newton-Verfahren (27) ist durchführbar in $S = S(x_*, \delta)$ und konvergiert Q-linear gegen x_*, d.h. $\lim_{k \to \infty} x_k = x_*$.

(ii) Falls $\lim_{k \to \infty} \eta^k = 0$, so ist die Konvergenz Q-überlinear.

(iii) Falls Konstanten $K_\eta > 0$ und $p \in (0, 1]$ existieren, so dass

$$\eta_k \leq K_\eta ||f(x_k)||^p \tag{31}$$

gilt, so konvergiert das Verfahren mit Ordnung $p + 1$.

BEWEIS: Seien δ und K die Konstanten des Lemmas 15. Wenn nötig, so verkleinere man δ und η_{max}, so dass nun

$$\varkappa := K(\delta + \eta_{max}) < 1 \tag{32}$$

erfüllt ist. Sei $x_k \in S(x_*, \delta)$, also $||e_k|| \leq \delta$. Wegen (29) und (32) schätzt man ab

$$||e_{k+1}|| \leq K(||e_k|| + \eta_k)||e_k|| \leq K(\delta + \eta_{max})||e_k|| < \varkappa ||e_k|| < \delta,$$

so dass $x_{k+1} \in S(x_*, \delta)$ folgt. Zudem ist die Q-lineare Konvergenz mit Faktor $\varkappa < 1$ garantiert. Behauptung (ii) ergibt sich wegen

$$\lim_{k\to\infty} \frac{||e_{k+1}||}{||e_k||} \;\leq\; \lim_{k\to\infty} K(||e_k|| + \eta_k) \;=\; 0$$

mit der Definition überlinearer Konvergenz. Ist die Bedingung (31) erfüllt, so lässt sich damit und mit Lemma 14 weiter abschätzen:

$$\begin{aligned} ||e_{k+1}|| \;\leq\; K(||e_k|| + \eta_k)||e_k|| \;&\leq\; K(||e_k|| + K_\eta||f(x_k)||^p)||e_k|| \\ &\leq\; K\left(||e_k|| + K_\eta 2^p||f'(x_*)||^p||e_k||^p\right)||e_k|| \\ &=\; K\left(\delta^{1-p} + K_\eta 2^p||f'(x_*)||^p\right)||e_k||^{p+1}\,, \end{aligned}$$

womit Behauptung (iii) bewiesen ist. □

Um eine praktikable Wahl der Zwangsterme η_k zu treffen, nehmen wir an, daß $f'(x_k) \neq 0$ für alle Iterationswerte ist und Algorithmus (27) nicht abbricht. Eine erste Wahlmöglichkeit für η_k lautet dann $\eta_0 \in [0, 1)$ und

$$\boxed{\eta_k \;:=\; \frac{\|f(x_k) - f(x_{k-1}) - f'(x_{k-1})s_{k-1}\|}{\|f(x_{k-1})\|}\,, \quad k = 1, 2, 3, \ldots} \tag{33}$$

Unter der Regularitätsvoraussetzung 1 an x_* zeigen S. Eisenstat & H. Walker [2] die Durchführbarkeit und überlineare Konvergenz des inexakten Newton-Verfahrens:

Satz 17 *Die Lösung $x_* \in D$ sei regulär. Dann existiert eine Konstante $\delta_* > 0$, so daß für jedes x_0, das hinreichend nahe bei x_* liegt, mit der Wahl (33) für η_k gilt: Das inexakte Newton-Verfahren (27) ist durchführbar in $S = S(x_*, \delta_*)$ und konvergiert gegen x_*. Die Fehler genügen der Abschätzung*

$$||x_{k+1} - x_*|| \;\leq\; C\,||x_k - x_*||\,||x_{k-1} - x_*||\,, \quad k = 1, 2, 3, \ldots \tag{34}$$

mit der von k unabhängigen Konstanten $C > 0$. □

Offenbar ist damit die Konvergenz Q-überlinear. Mit seiner allgemeineren R-Ordnung $(1 + \sqrt{5})/2$ (vgl. [9]) wird das Verfahren vergleichbar mit der vom eindimensionalen Fall her bekannten *Regula falsi* (Sekanten-Verfahren).

Für eine zweite Möglichkeit, den Zwangsterm η_k zu bestimmen, wähle man die Konstanten $\gamma \in [0, 1]$ und $\alpha \in (1, 2]$. Mit dem Startwert $\eta_0 \in [0, 1)$ setze man

$$\boxed{\eta_k \;:=\; \gamma\left(\frac{\|f(x_k)\|}{\|f(x_{k-1})\|}\right)^\alpha\,, \quad k = 1, 2, 3, \ldots} \tag{35}$$

Mit dieser Wahl läßt sich überlineare Konvergenz ebenfalls nachweisen; hier wird der in [2] aufgestellte Satz angegeben:

Satz 18 *Die Lösung $x_* \in D$ sei regulär. Dann existiert eine Konstante $\delta_* > 0$, so daß für jedes x_0, das hinreichend nahe bei x_* liegt, mit der Wahl (35) für η_k gilt: Das inexakte Newton-Verfahren (27) ist durchführbar in $S = S(x_*, \delta_*)$ und konvergiert gegen x_*. Bei $\gamma < 1$ hat die Konvergenz die Q-Ordnung α, bei $\gamma = 1$ lautet die Q-Ordnung p mit $p \in [1, \alpha)$.* □

In praxi sind Konstanten $\alpha = 2$ und $\gamma = 0.9$ üblich, womit theoretisch Q-quadratische Konvergenz erreichbar wird. Zudem wird in beiden Fällen (33) und (35) durch zusätzliche Beschränkungen (engl.: safeguards) vermieden, daß die Zwangsterme zu schnell klein werden und somit überhöhte Genauigkeitsforderungen bereits bei den ersten Newton-Schritten zu unnötigen Iterationen (engl.: oversolving) führen.

2 Parameterabhängige nichtlineare Gleichungssysteme

Wir betrachten in diesem Abschnitt parameterabhängige Gleichungssysteme

$$f(x, \lambda) \;=\; 0\,, \quad f : \mathbb{R}^{n+1} \to \mathbb{R}^n \tag{36}$$

mit einem skalaren Parameter $\lambda \in \Lambda = [a, b] \subset \mathbb{R}$. Die in Elektrotechnik und Maschinenbau auftretenden Systeme enthalten meist zahlreiche Parameter, z.B. Widerstandswerte, Kapazitäten und Induktivitäten, oder auch Elastizitätsmodule, Dichten und externe Kräfte. Vereinfachend wird angenommen, daß alle derartigen Werte bis auf einen ausgewählen reellen Systemparameter λ konstant vorgegeben sind. Numerische Verfahren zur Berechnung einer Lösung $x = x(\lambda)$ von (36) für alle $\lambda \in \Lambda$ bezeichnet man als *Fortsetzungsmethoden.*

Beispiel 19 (Energetische Systeme)

Der nichtlineare parallele Resonanzkreis ist ein oft benutztes Modell der Elektroenergietechnik, bei dem die Kennlinie der Induktivität durch ein Polynom $y_L = ax + bx^n$ mit $a + b = 1$ beschrieben wird. Für Kernmaterialien mit ausgeprägtem Knick in der Magnetisierungskennlinie haben sich z. B. Polynome 9.Grades $y_L = ax + bx^9$ bewährt (vgl. [5]). Nimmt man eine sinusförmige Erregung $u(t) = \lambda \sin(t)$ mit Amplitude $\lambda > 0$ und normierter Periode $T = 2\pi$ an, so ist man an einer Antwort des energetischen Systems der Form

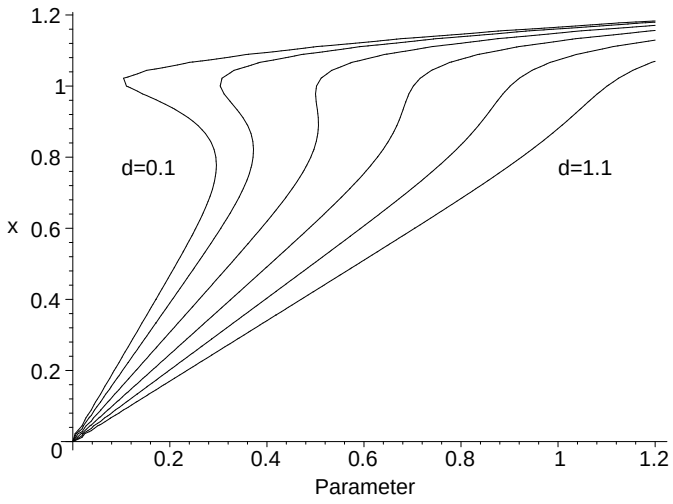

Abbildung 1: Energetisches System

$$z(t) = x \sin(t) \quad \text{mit Amplitude} \quad x > 0$$

interessiert. Die unbekannte Amplitude x genügt gemäß [5] in erster Näherung der Gleichung

$$f(x, \lambda) \;=\; d^2 x^2 + \left((a - \frac{2}{3})x + \frac{126}{256} b x^9 \right)^2 - \lambda^2 \;=\; 0\,, \quad \lambda \in \Lambda = [0, 1.2] \tag{37}$$

mit den Konstanten $a = 0.25$, $b = 0.75$ und dem Dämpfungsparameter $d \in \{0.1, 1.1\}$. Für Dämpfungswerte $d = 0.1, 0.3, 0.5, 0.7, 0.9, 1.1$ sind die Lösungszweige

$$\mathcal{L} \;=\; \{(x,\lambda) \mid f(x,\lambda) = 0, \lambda \in \Lambda\}$$

in Abb. 1 dargestellt. Während die Amplitude $x = x(\lambda)$ bei starker Dämpfung $d = 1.1$ eindeutig als Funktion der Erregungsamplitude λ bestimmt werden kann, treten z.B. für $d = 0.1$ Mehrdeutigkeiten auf. ◀

2.1 Numerische Fortsetzungsmethoden

Zur Vereinfachung der Betrachtung nehmen wir vorerst an, daß die zu ermittelnden Lösungen $x = x(\lambda)$ mit $\lambda \in \Lambda = [a, b]$ regulär sind und treffen dazu folgende

Voraussetzung 20 *Sei $D \subset \mathbb{R}^n$, offen.*
(i) $f \in C^r(D \times \Lambda),\ r \geq 2$
(ii) $f_x(x,\lambda)$ *ist regulär für alle* $(x,\lambda) \in D \times \Lambda$.

Mit diesen Annahmen garantiert der Satz über die implizite Funktion aus Kapitel 3, daß $f(x,\lambda) = 0$ für jedes $\lambda \in \Lambda$ genau eine Lösung $x = x(\lambda) \in D$ besitzt. Diese Funktion $x : \Lambda \to D$ ist zudem differenzierbar mit der Darstellung der Ableitung

$$x'(\lambda) \;=\; -[f_x(x(\lambda),\lambda]^{-1} f_\lambda(x(\lambda),\lambda)\,. \tag{38}$$

Die auch als *Einbettungsgleichung* bezeichnete Differentialgleichung erhält man nach Einsetzen der Funktion $x(\lambda)$ in die Gleichung (36), deren Differentiation nach λ und Beachtung der Voraussetzung 20*(ii)* aus

$$f_x(x(\lambda),\lambda) \cdot x'(\lambda) + f_\lambda(x(\lambda),\lambda) = 0\,.$$

Numerische Fortsetzungsmethoden unterteilen das Parameterintervall $\Lambda = [a, b]$ in N Teilintervalle mit den Teilpunkten $a = \lambda_0 < \lambda_1 < \ldots < \lambda_N = b$ und bestimmen sukzessive die Punkte $(x(\lambda_j), \lambda_j)$ der Lösungskurve

$$\mathcal{L} = \{(x,\lambda) | (x,\lambda) \in D \times \Lambda \;\text{ mit }\; f(x,\lambda) = 0\}, \tag{39}$$

beginnend mit $(x(a), a) = (x_0, \lambda_0)$ in der Reihenfolge $(x_0,\lambda_0),\ (x_1,\lambda_1),\ \ldots,(x_N,\lambda_N)$. Dabei wird abkürzend $x_j = x(\lambda_j)$ notiert. Ein *Fortsetzungsschritt* von λ_{j-1} bis λ_j ist folgendermaßen aufgebaut (vgl. Abb. 2):

1. Gegeben sei ein Kurvenpunkt (x_{j-1}, λ_{j-1}).
2. Festlegung einer Fortsetzungsschrittweite $h_j > 0$ und des neuen Parameterwertes $\lambda_j = \lambda_{j-1} + h_j$.
3. Vorgabe eines Näherungswertes x^P für den neuen Lösungsvektor $x_j = x(\lambda_j)$.

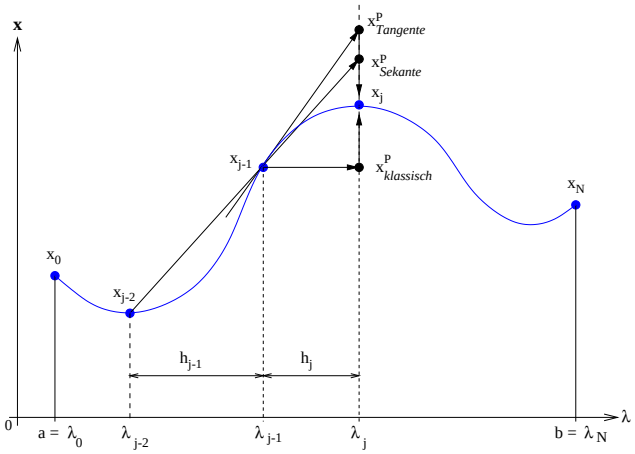

Abbildung 2: Fortsetzungsschritt

4. Iterative Verbesserung des Prädiktorwertes mittels eines Newton-ähnlichen Verfahrens für Gleichung (36) (Korrektorschritt).

Betrachten wir zuerst den Korrektorschritt. Wegen seiner schnellen Konvergenz wird dafür häufig das Newton-Verfahren genutzt. Mit dem gegebenen Punkt (x^P, λ_j), setze man $u_0 := x^P$ und löse für $k = 0, 1, 2, ..., K-1$ das lineare Gleichungssystem

$$\begin{aligned} f_x(u_k, \lambda_j) \cdot \Delta u_k &= -f(u_k, \lambda_j), \\ u_{k+1} &= u_k + \Delta u_k \end{aligned} \tag{40}$$

Der ermittelte Punkt (x_j, λ_j) mit dem Wert $x_j := u_K$ liegt i. allg. nicht auf der Kurve $\mathcal{L}$, da der Abbruch des Verfahrens bereits nach endlich vielen Iterationen erfolgt. Ein geeignetes Abbruchkriterium muß deshalb vorgesehen werden. Um die aufwändigen Berechnungen der Jacobimatrizen zu vermeiden, werden oft Newton-ähnliche Verfahren mit konstanter Matrix $A_0 \sim f_x(x^P, \lambda_j)$ bevorzugt, so dass (40) nun die Form

$$\begin{aligned} A_0 \cdot \Delta u_k &= -f(u_k, \lambda_j), \\ u_{k+1} &= u_k + \Delta u_k \end{aligned} \tag{41}$$

erhält, z. B. mit der Differenzenapproximation der Matrixelemente

$$\{A_0\}_{ij} := \frac{1}{\Delta x_j}\{f_i(x_1^P, ..., x_j^P + \Delta x_j, ...) - f_i(x^P)\}, \quad i, j = 1(1)n. \tag{42}$$

Für den Prädiktorschritt nutzt man Informationen über zuvor bestimmte Kurvenpunkte.

1. Der *klassische Prädiktor* nimmt den vorhergehenden Kurvenpunkt als Startwert für den folgenden Korrektorschritt:

$$x^P := x_{j-1} = x(\lambda_{j-1}) \tag{43}$$

2. Der *Tangenten-Prädiktor* (auch EULER-Prädiktor genannt) ersetzt die Kurve lokal durch ihre Tangente im vorhergehenden Kurvenpunkt

$$x(\lambda) = x(\lambda_{j-1}) + (\lambda - \lambda_{j-1}) \cdot x'(\lambda_{j-1}) \,,$$

wobei $x'(\lambda_{j-1})$ mit der Einbettungsgleichung (38) gewonnen werden kann. Wegen $x_{j-1} := x(\lambda_{j-1})$ ergibt sich die Prädiktorformel

$$x^P := x_{j-1} - h_j \cdot [f_x(x_{j-1}, \lambda_{j-1})]^{-1} f_\lambda(x_{j-1}, \lambda_{j-1}) \,. \tag{44}$$

Sie besitzt i. allg. eine höhere Genauigkeit als (43), erfordert allerdings die Ableitungen f_x und f_λ sowie die Lösung eines linearen Gleichungssystems.

3. Legt man durch die 2 Stützstellen (x_{j-2}, λ_{j-2}) und (x_{j-1}, λ_{j-1}) die Sekante und extrapoliert zum Parameterwert λ_j, so liefert dieser *Sekanten-Prädiktor* den Wert

$$x^P := x_{j-1} + \frac{\lambda_j - \lambda_{j-1}}{\lambda_{j-1} - \lambda_{j-2}} \cdot [x_{j-1} - x_{j-2}] \,. \tag{45}$$

Er benötigt für x^P bereits an der Stelle λ_1 zwei Startwerte, die man sich z.B. mittels des klassischen Prädiktors verschaffen muß. Dieser Ansatz stellt einen guten Kompromiß zwischen den Formeln (43) und (44) dar. Durch die Benutzung weiterer zurückliegender Kurvenpunkte und deren Ableitung $x'(\lambda_{j-2})$ lassen sich mit HERMITE-Interpolation noch genauere Prädiktorwerte bestimmen.

2.2 Schrittweitensteuerung

Die Wahl der Fortsetzungsschrittweite $h_j = \lambda_j - \lambda_{j-1}$ ist oft entscheidend für den Erfolg der Fortsetzungsmethode. Denn bei zu großem Wert kann der Prädiktorpunkt außerhalb des Einzugsbereiches der zu bestimmenden Lösung liegen. Dies führt meist zur Divergenz des Korrektors, mitunter auch zur Konvergenz gegen eine andere unerwünschte Lösung. Durchgehend kleine Schrittweiten h_j erhöhen andererseits die Zahl N der Fortsetzungsschritte und so den Gesamtaufwand über alle Maßen.

Beispiel 21 Gegeben sei $p \in \mathbb{R}, p > 0$; z. B. $p = 10^{-6}$. Die Gleichung lautet

$$f(x,\lambda) \;=\; x^2 - (\lambda - \frac{1}{2})^2 - p^2 \;=\; 0\,, \quad \lambda \in [0,1] \tag{46}$$

Die beiden Lösungen liegen symmetrisch zur $\lambda-$Achse

$$x_{1,2}(\lambda) = \pm\sqrt{(\lambda - \frac{1}{2})^2 + p^2}\,,$$

die damit die Einzugsbereiche trennt (vgl. Abb. 3). Verfolgt man z.B. die positive Lösung $x_1(\lambda)$, beginnend am Parameterwert $\lambda = 0$ und benutzt den Tangenten-Prädiktor, so muß die Fortsetzungsschrittweite h_j stets so klein gewählt werden, daß kein Prädiktorpunkt unterhalb der $\lambda-$Achse zu liegen kommt. Man zeigt leicht: Ist $h_j < 2p = 2 \cdot 10^{-6} \quad \forall_j$, so verbleiben alle Prädiktorwerte $\hat{x}_j$ im Einzugsgebiet des Newtonverfahrens für $x_1(\lambda)$; andernfalls (vgl. Punkt x^Q) kann ein unerwünschter Zweigwechsel erfolgen. Bei konstanter Fortsetzungsschrittweite wären deshalb $N \geq 1/(2p) = 500000$ Schritte auszuführen. ◀

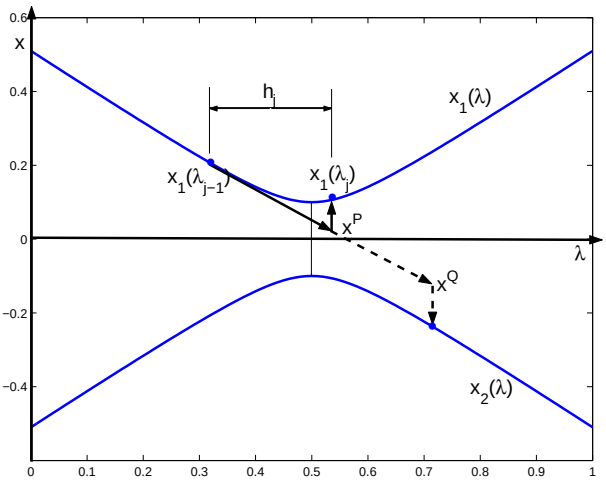

Abbildung 3: Lösungszweige $\mathcal{L}_1$, $\mathcal{L}_2$

Theoretisch aufwendige Strategien zur Schrittweitenwahl findet man in der Literatur [1]. Eine einfache aber zuverlässige Strategie soll solche λ-Schrittweiten $h = h_j > 0$ wählen, die die Konvergenz des Korrektors (gegen die verfolgte Lösung) garantieren und die Anzahl der Rechenoperationen minimieren. Heuristisch läßt sich der Rechenaufwand $A(h)$, bezogen auf einen Einheits-Fortsetzungsschritt, durch

$$A(h) := \frac{k(h)}{h}$$

mit $k(h)$ Korrektorschritten ansetzen. Werde mit "optimaler" Schrittweite $h_{opt} = \varrho \cdot h$ der minimale Aufwand

$$A(h_{opt}) := \frac{k(h_{opt})}{h_{opt}}$$

erreicht, so liefert Division mit den Abkürzungen $k := k(h),\ k_{opt} := k(h_{opt})$

$$\frac{A(h)}{A(h_{opt})} \;=\; \frac{k \cdot h_{opt}}{h \cdot k_{opt}} \;=\; \frac{k}{k_{opt}} \cdot \varrho \;\geq\; 1\,.$$

Also ist $\varrho \geq k_{opt}/k$ sinnvoll. Gibt man eine optimale Anzahl von Korrektoriterationen $k_{opt} =: \kappa$ (z.B. $k_{opt} = 5$) vor, so setze man heuristisch

$$\rho := k_{opt}/k\,. \tag{47}$$

Eine zusätzliche Begrenzung $\frac{1}{2} \leq \varrho \leq 2$ empfiehlt sich, um die Sicherheit der Wahl von $h_{opt} = \varrho \cdot h$ zu erhöhen.

Algorithmus 22 (Fortsetzungs-Verfahren)

Function continuation $(f, Df x, a, b, x, tol)$

1. Setze $j := 0,\ \lambda_0 := a,\ x^P := x$
   Wähle Schrittweite h sowie Iterationszahl k_{opt}.
2. *(Startrechnung)*
   Bestimme $x_0 = x(\lambda_0)$ mit Verfahren (40) oder (41).
3. Do while $\lambda_j \leq b$
   1. Setze $j := j + 1$.
   2. Setze $\lambda_j := \lambda_{j-1} + h$. Falls $\lambda_j > b$, so $\lambda_j := b$.
   3. *(Prädiktor)*
      Bestimme x^P mit Verfahren (43), (44) oder (45).
   4. *(Korrektor)*
      Bestimme $x_j = x(\lambda_j)$ mit Verfahren (40) oder (41)
      in k Schritten bis auf eine Genauigkeit tol .
   5. *(Schrittweiten-Bestimmung)*
      (a) Bestimme $\rho := k_{opt}/k$.
      (b) Beschränke $\rho := \max\{\min(\rho, 2), \frac{1}{2}\}$.
      (c) Setze $h := h * \rho$.
      (d) Falls $h < \varepsilon_{Maschine}$, so STOP.
4. Return $N := j,\ \lambda_j,\ x_j,\ j = 0(1)N$

Wir erhalten den Algorithmus 22 für das Prädiktor-Korrektor-Verfahren zur numerischen Lösungsfortsetzung. Er paßt die aktuelle Fortsetzungsschrittweite h_j in jedem Schritt so an, daß eine "optimale" Zahl k_{opt} von Korrektorschritten ausgeführt wird. Für das Newton-Verfahren hat sich $k_{opt} = 5$ bewährt. Bei zu geringer Fortsetzungsschrittweite h, insbesondere

bei Unterschreiten der Maschinengenauigkeit $\varepsilon_{Maschine}$, sollte der Algorithmus definiert abgebrochen werden. Die Jacobimatrix $Dfx := f_x(x,\lambda)$ wird bei einer Differenzenapproximation (42) nicht benötigt.

2.3 Kurvenverfolgung

Betrachtet man die Lösungszweige $\mathcal{L}$ von $f(x,\lambda) = 0$ als Kurven in $\mathbb{R}^{n+1}$, so versagen die behandelten Fortsetzungstechniken an den Stellen (x_0,λ_0), an denen die Jacobimatrix $f_x(x_0,\lambda_0)$ singulär wird.

Beispiel 23

$$f(x,\lambda) \;=\; (x^3 - x - \lambda)(x - \sin\lambda) \;=\; 0\,, \quad \lambda \in \Lambda = [-2,2] \tag{48}$$

Die Lösungsmenge $\mathcal{L} = \{(x,\lambda) \mid (x,\lambda) \in \mathbb{R}\times\Lambda,\ f(x,\lambda) = 0\}$ setzt sich aus den beiden Lösungszweigen

$$\mathcal{L}_0 \;=\; \{(x,\lambda) \mid x = \sin\lambda\,,\ \lambda\in\Lambda\} \quad \text{und} \quad \mathcal{L}_1 \;=\; \{(x,\lambda) \mid x^3 - x = \lambda\,,\ \lambda\in\Lambda\}$$

zusammen. Anhand der Darstellung von $\mathcal{L} = \mathcal{L}_0 \cup \mathcal{L}_1$ als Kurve in $\mathbb{R}^2$ in Abb. 4 ergibt sich die Anzahl der Lösungen zu

2	$f\ddot{u}r$	$\lambda \in [-2,\lambda_1)$
3	$f\ddot{u}r$	$\lambda = \lambda_1$
4	$f\ddot{u}r$	$\lambda \in (\lambda_1,\lambda_0)$
3	$f\ddot{u}r$	$\lambda = \lambda_0$
4	$f\ddot{u}r$	$\lambda \in (\lambda_0,\lambda_2)$
3	$f\ddot{u}r$	$\lambda = \lambda_2$
2	$f\ddot{u}r$	$\lambda \in (\lambda_2,2]$.

An den 3 Parameterwerten $\lambda_0,\lambda_1,\lambda_2$ ändert sich die Anzahl der Lösungen und damit die Struktur der Lösungsmenge. Die betreffenden 3 singulären Lösungen x_{*0}, x_{*1}, x_{*2} erhält man mit den Bestimmungsgleichungen

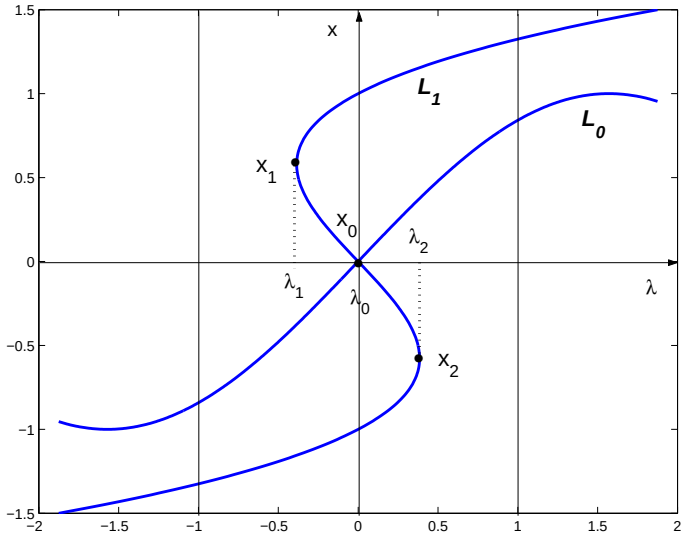

Abbildung 4: Lösungszweige $\mathcal{L}_0$ und $\mathcal{L}_1$

$$\begin{aligned} f(x,\lambda) \;&=\; (x^3 - x - \lambda)(x - \sin\lambda) \qquad\qquad = \;0 \quad \text{und} \\ f_x(x,\lambda) \;&=\; (3x^2 - 1)(x - \sin\lambda) + (x^3 - x - \lambda) \;=\; 0 \quad \text{mit} \quad f_x(x,\lambda) \;=\; \frac{\partial f}{\partial x}(x,\lambda)\,. \end{aligned}$$

Außer dem *Verzweigungspunkt* $y_0 = (0,0)$ bei $\lambda_0 = 0$ existieren die zwei *Umkehrpunkte* $y_1 = (\frac{1}{3}\sqrt{3}, -\frac{2}{9}\sqrt{3})$ und $y_2 = (-\frac{1}{3}\sqrt{3}, \frac{2}{9}\sqrt{3})$. Alle anderen Lösungen x_* mit $(x_*,\lambda) \in \mathcal{L}$ sind hingegen regulär. ◀

Wir setzen $y = (x_1 ..., x_n, \lambda)^\top = (x,\lambda)^\top \in \mathbb{R}^{n+1}$ und betrachten das unterbestimmte Gleichungssystem

$$f(y) = 0\,, \quad f : D \subset \mathbb{R}^{n+1} \to \mathbb{R}^n\,. \tag{49}$$

Voraussetzung 24 *Sei $D \subset \mathbb{R}^{n+1}$, offen.*

(i) $f \in C^r(D),\ r \geq 2$

(ii) Es existiert ein $y_0 \in D \subset \mathbb{R}^{n+1}$ *mit* $f(y_0) = 0$.

(iii) Die erweiterte Jacobimatrix $f'(y_0) = (f_x(y_0), f_\lambda(y_0))$ *hat maximalen Rang* n.

Insbesondere bedeutet *(iii)* eine Abschwächung der bisherigen Regularitätsvoraussetzung. Der Satz über die implizite Funktion liefert dann die

Folgerung 25

Unter den Voraussetzungen 24 existiert eine glatte Kurve $\mathcal{L}$ *in* $\mathbb{R}^{n+1}$ *mit Parameterdarstellung* $y = y(s),\ s \in J = (s_0 - \delta, s_0 + \delta),\ \delta > 0,\ y : J \longrightarrow \mathbb{R}^{n+1}$, *so daß für alle* $s \in J$ *gilt: (i)* $y(s_0) = y_0$, *(ii)* $f(y(s)) = 0$, *(iii)* $rank\ f'(y(s)) = n$, *(iv)* $y'(s) \neq 0$.

$\mathcal{L}$ heißt *Lösungspfad* mit der *Parametrisierung* $y = y(s)$. Einsetzen der Parametrisierung in (49) und Differentiation von *(ii)* nach s liefert

$$f'(y(s)) \cdot y'(s) = 0\,. \tag{50}$$

Also liegt der Vektor $y'(s)$ im Nullraum $N(f'(y(s)))$ und ist orthogonal zu allen n Zeilenvektoren dieser Matrix. Wegen der Rangaussage *(iii)* ist folglich die um eine Zeile erweiterte Jacobimatrix

$$H'(y(s)) := \begin{pmatrix} f'(y(s)) \\ y'(s)^\top \end{pmatrix} \tag{51}$$

regulär auf dem Pfad $\mathcal{L}$ für $s \in J$. Ein Lösungspfad $\mathcal{L} = \{y(s) \mid f(y(s)) = 0\ \ \forall s \in J\}$ mit der Eigenschaft $rank\ f'(y(s)) = n$ für alle $s \in J$ wird als *regulär* bezeichnet. Zur Klassifikation der Punkte $y \in \mathcal{L}$ gilt folgender

Satz 26

Sei $f'(y) = (f_x(x,s), f_s(x,s))$. *Die Bedingung* $rank\, f'(y) = n$ *ist genau dann erfüllt, wenn (i)* $f_x(x,s)$ *regulär ist oder (ii)* $dim\, N(f_x(x,s)) = 1$ *und* $f_s \notin R(f_x(x,s))$ *gilt.*

Reguläre Lösungen $y(s_0) = (x(s_0), \lambda(s_0)) \in \mathcal{L}$ erfüllen offenbar Bedingung *(i)*, wogegen eine Lösung im Falle *(ii)* als *Umkehrpunkt (einfacher Grenzpunkt, simple limit point, turning point)* bezeichnet wird. Unter den Voraussetzungen 24 besteht jeder reguläre Lösungspfad deshalb nur aus regulären Punkten und einfachen Grenzpunkten.

Beispiel 27 Beispiel 23 besitzt die Darstellung

$$f(y) \;=\; (y_1^3 - y_1 - y_2)(y_1 - \sin y_2) \;=\; 0\,, \quad y = (x, \lambda) = (y_1, y_2)\,. \tag{52}$$

(a) Für den Kurvenpunkt $y_3 = (1, 0)$ sind die Voraussetzungen 24 mit $f'(y_3) = (2, -1) \Longrightarrow rank\ f'(y_3) = 1$ erfüllt. Wegen $rank\ f_x(y_3) = 1$, ist y_3 regulärer Kurvenpunkt.

(b) Betrachten wir den Punkt $y_1 = (x_{*1}, \lambda_1) = (\frac{1}{3}\sqrt{3}, -\frac{2}{9}\sqrt{3})$. Man überprüft leicht die Voraussetzungen 24, insbesondere $f'(y_1) = (0, -\frac{1}{3}\sqrt{3} - \sin(\frac{2}{9}\sqrt{3}))$, also $rank\ f'(y_1) = 1$. Allerdings ist nun $rank\ f_x(y_1) = 0$, jedoch $f_\lambda(y_1) \notin R(f_x(y_1))$. Nach Lemma 26 ist folglich y_1 Umkehrpunkt.

(c) Der Punkt $y_0 = (x_{*0}, \lambda_0) = (0, 0)$ erfüllt wegen $f'(y_0) = (0, 0) \Longrightarrow rank\, f'(y_0) = 0 < n$ nicht die Voraussetzungen 24 und ist somit weder regulärer Punkt noch Umkehrpunkt. Eine lokale Parametrisierung $y(s)$ ist hier nicht angebbar.

Für den nach Folgerung 25 garantierten glatten regulären Lösungspfad $\mathcal{L}$ kann als natürlicher Parameter s die Bogenlänge der Kurve $\mathcal{L} \in \mathbb{R}^{n+1}$ benutzt werden, die mit der Euklidischen Norm stets die Bedingung $\|y'(s)\|_2 = 1$ erfüllt.

Wir übertragen nun die 4 Punkte eines Fortsetzungsschrittes von s_{j-1} bis s_j (vgl. Abb. 5):

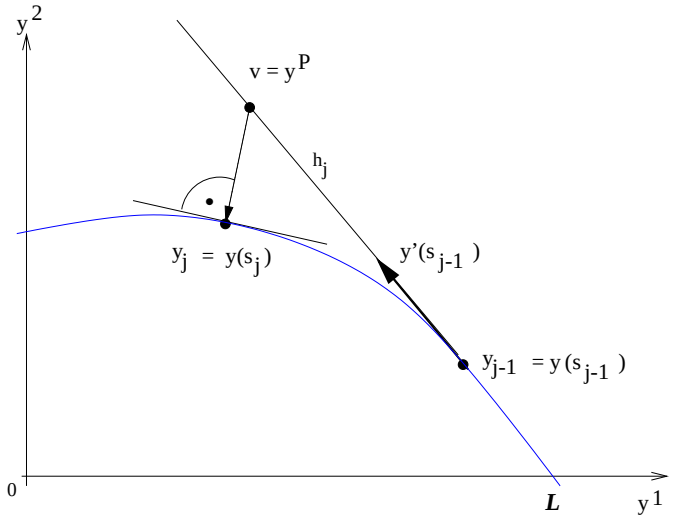

Abbildung 5: Kurvenverfolgung

1. Gegeben sei ein Kurvenpunkt $y_{j-1} = y(s_{j-1})$.

2. Festlegung einer geeigneten Fortsetzungsschrittweite $h_j > 0$ und des neuen Parameterwertes $s_j = s_{j-1} + h_j$.

3. Vorgabe eines Prädiktorpunktes $v = y^P$ für den neuen Kurvenpunkt y_j , z.B. mit dem Tangentenprädiktor

$$y^P := y_{j-1} + h_j \cdot y'(s_{j-1}) \,, \quad \|y'(s_{j-1})\|_2 = 1 \tag{53}$$

oder dem Sekantenprädiktor

$$y^P := y_{j-1} + \frac{h_j}{h_{j-1}} \cdot (y_{j-1} - y_{j-2}) \,. \tag{54}$$

4. Bestimmung eines Kurvenpunktes $y_j \in \mathcal{L}$ mit minimalem Abstand von $v = y^P$, d.h. iterative Lösung der Aufgabe

$$\|y_j - v\|_2 \;=\; \min_{f(y)=0} \|y - v\|_2 \tag{55}$$

im Korrektorschritt (vgl. Abb. 5).

Um dieses Extremwertproblem mit Nebenbedingung zu lösen, gibt man mit den Lagrange-Multiplikatoren $\mu = (\mu_1, ..., \mu_n)^\top$ die zugehörige Lagrange-Funktion

$$L(y, \mu) \;=\; \frac{1}{2}\|y - v\|_2^2 - \mu^\top f(y) \;=\; \frac{1}{2}\sum_{i=1}^{n+1}(y_i - v_i)^2 - \sum_{i=1}^{n} \mu_i f_i(y) \tag{56}$$

an. Durch Differentiation gewinnt man die notwendige Bedingung für ein lokales Minimum an y_j in der Komponentenform

$$\begin{array}{lllllll} (i) & \frac{\partial L}{\partial \mu_k} & = & f_k(y) & = & 0 \,, & k = 1(1)n, \\ (ii) & \frac{\partial L}{\partial y_l} & = & (y_l - v_l) - \sum_{i=1}^{n} \mu_i \frac{\partial f_i}{\partial y_l} & = & 0 \,, & l = 1(1)n+1 \end{array}$$

bzw. in Vektorschreibweise

$$\begin{array}{lll} (i) & f(y) & = \ 0 \\ (ii) & y - v & = \ [f'(y)]^\top \cdot \mu, \quad \mu \in \mathbb{R}^n \, . \end{array} \tag{57}$$

Multiplikation von Bedingung *(ii)* an der Lösung y_j mit dem Tangentenvektor $y'(s_j)$ ergibt

$$y'(s_j)^\top \cdot (y_j - v) \ = \ [f'(y(s_j)) \cdot y'(s_j)]^\top \mu \ = \ 0$$

wegen Voraussetzung 24, womit das bestimmende System für $y_j \in \mathbb{R}^{n+1}$ nun

$$\begin{array}{lll} (i) & f(y) & = \ 0 \\ (ii) & y'(s_j)^\top \cdot (y - v) & = \ 0 \end{array} \tag{58}$$

lautet. Linearisierung dieses $(n+1)-$dimensionalen Systems an der Näherungslösung $v \in \mathbb{R}^{n+1}$ mit zugehörigem Tangentenvektor $v'(s_j)$ liefert das lineare Gleichungssystem für den Wert v_{neu}:

$$\begin{array}{ll} f'(v) \cdot (v_{neu} - v) & = \ -f(v) \\ v'(s_j)^\top \cdot (v_{neu} - v) & = \ \ 0 \, . \end{array} \tag{59}$$

Geometrisch bedeutet dies, eine Lösung $x := v_{neu} - v$ des unterbestimmten linearen Gleichungssystems $Ax = b$ mit $A := f'(v) \, , b := -f(v)$ zu finden, die die Minimierungsaufgabe

$$\|x_*\|_2 \ = \ \min_{Ax=b} \|x\|_2 \tag{60}$$

löst. Derartige minimale Quadratmittel-Lösungen kann man mit der Pseudoinversen (Moore–Penrose–Inversen) A^+ von A formal in der Form

$$x_* = A^+ b \tag{61}$$

darstellen. Die Berechnung von A^+ im allgemeinen Fall einer $(m,n)-$Matrix ist bekanntlich aufwendig und bei Fortsetzungsverfahren nicht praktikabel. Im Falle $A \in \mathbb{R}^{n \times (n+1)}$ mit Voraussetzungen 24 gilt jedoch das folgende

Lemma 28
Besitzt $A \in \mathbb{R}^{n \times (n+1)}$ den vollen Rang n (zeilenregulärer Fall), so kann A^+ durch

$$A^+ = A^\top (AA^\top)^{-1} \tag{62}$$

berechnet werden. In diesem Falle wird A^+ als Rechtsinverse von A bezeichnet (denn $AA^+ = AA^\top (AA^\top)^{-1} = I$).

Anwendung der Pseudoinversen von $A := f'(v)$ auf (59) ergibt den Newtonschritt

$$v_{neu} = v - A^+ f(v) \quad \text{mit} \quad A^+ = A^\top (AA^\top)^{-1} \, ,$$

dessen iterative Wiederholung das *Gauß-Newton-Verfahren für unterbestimmte nichtlineare Gleichungssysteme* liefert:

$$\boxed{v_{k+1} = v_k - A^\top (AA^\top)^{-1} f(v_k) \quad \text{mit} \quad A = f'(v_k), \ k = 0, 1, 2...} \tag{63}$$

Ist der Startpunkt (Prädiktorwert) $v_0 = v = y^P$ hinreichend nahe dem Lösungspfad $\mathcal{L}$, so kann die quadratische Konvergenz des unterbestimmten Gauß-Newton-Verfahrens (63) nachgewiesen werden. Auf den umfangreichen Beweis wird hier verzichtet (vgl. [1], S. 23-27).

Algorithmus 29 (Euler-Newton-Verfahren)

Function eulernewton $(f, Df, a, b, x, smax, tol)$

1. Setze $j := 0,\ \lambda_0 := a,\ x^P := x,\ s_0 := 0$
   Wähle Schrittweite h sowie Iterationszahl k_{opt}.
2. *(Startrechnung)*
   Bestimme $x_0 = x(\lambda_0)$ mit Verfahren (40) oder (41).
   Setze $y_0 = (x_0, \lambda_0)$.
3. Do while ($a \leq \lambda_j \leq b$ and $s_j < smax$)
   - 3.1. Setze $j := j + 1$ und $s_j := s_{j-1} + h$.
   - 3.2. *(Euler-Prädiktor)*
     Bestimme $v_0 := y^P$ mit Verfahren (53).
   - 3.3. *(Gauß-Newton-Korrektor)*
     - (a) Setze $k := 0$.
     - (b) $A := Df(v_k) = f'(v_k)$
     - (c) Löse das reguläre lineare System
       $$(AA^\top) \cdot b = f(v_k),\ b \in \mathbb{R}^n$$
     - (d) $v_{k+1} := v_k - A^\top b,\ k := k + 1$
     - (e) Falls $\|A^\top b\|_2 > tol$, so gehe zu Schritt (b).
     - (f) Setze $y_j := v_k$ und bestimme den Tangentenvektor $y'(s_j)$.
   - 3.4. *(Schrittweiten-Anpassung)*
     - (a) Bestimme $\rho := k_{opt}/k$.
     - (b) Beschränke $\rho := \max\{\min(\rho, 2), \frac{1}{2}\}$.
     - (c) Setze $h := h * \rho$.
     - (d) Falls $h < \varepsilon_{Maschine}$, so STOP.
4. Return $N := j,\ y_j = (x_j, \lambda_j)\ ,\ j = 0(1)N$

Satz 30 *Für $f : D \subset \mathbb{R}^{n+1} \to \mathbb{R}^n$ seien die Voraussetzungen 24 erfüllt. Dann existiert eine offene Umgebung $U(\mathcal{L}) \subset \mathcal{D}$ des Lösungspfades, mit der folgende Behauptungen gelten:*

(i) Zu jedem $v \in U(\mathcal{L})$ existiert genau ein $y_* \in \mathcal{L}$ mit

$$\|y_* - v\|_2 = \min_{f(y)=0} \|y - v\|_2.$$

Die dadurch definiert Abbildung $S : U(\mathcal{L}) \to \mathcal{L}$ ist glatt.

(ii) Ist $v_0 \in U(\mathcal{L})$, so auch $v_1 \in U(\mathcal{L})$.

(iii) Zu jedem $v_0 \in U(\mathcal{L})$ konvergiert die Folge $\{v_k\}$, $k = 0, 1, 2, ...$ der Iterierten gemäß (63) gegen ein $v_\infty \in \mathcal{L}$.

(iv) Für $v_0 \in U(\mathcal{L})$ gelten folgende Abschätzungen gleichmäßig für $v_0 \in U(\mathcal{L})$:

$$\begin{aligned} \|v_2 - v_1\| &\leq C_1\|v_1 - v_0\|^2 \\ \|v_\infty - v_1\| &\leq C_2\|v_\infty - v_0\|^2 \\ \|v_1 - S(v_0)\| &\leq C_3\|v_0 - S(v_0)\|^2 \\ \|v_\infty - S(v_0)\| &\leq C_4\|v_0 - S(v_0)\|^2 \end{aligned} \tag{64}$$

Dieser Satz garantiert zwar die Konvergenz der Newton-Iterierten v_k mit Startwert v_0 gegen einen Kurvenpunkt $v_\infty \in \mathcal{L}$, der jedoch nicht mit $y_* = S(v_0)$ zusammenfallen muß. Zur praktischen Umsetzung des Fortsetzungsverfahrens sind zusätzlich zu den Parametern des Algorithmus 22 die vollständige Jacobimatrix $Df = f'(y)$ und eine obere Schranke s_{max} des Fortsetzungsparameters s vorzugeben. Zusammen mit dem Euler-Prädiktor ergibt der Gauß-Newton-Korrektor das Verfahren 29.

Literatur

[1] Allgower, E. L.; Georg, K.: *Numerical Continuation Methods.* Springer – Verlag, Berlin 1990.

[2] Eisenstat, S.C.; Walker, H.F.: *Choosing the Forcing Terms in an Inexact Newton Method*, SIAM J. Scientific Comput. 17, No.1, pp.16-32

[3] Kelley, C.T.: *Iterative Methods for Linear and Nonlinear Equations.* SIAM, Philadelphia 1995.

[4] Kosmol, P.: *Methoden zur numerischen Behandlung nichtlinearer Gleichungen und Optimierungsaufgaben.* B.G.Teubner, Stuttgart 1993.

[5] Philippow, E.S.; Büntig, W.G.: *Analyse nichtlinearer dynamischer Systeme der Elektrotechnik.* Carl Hanser Verlag, München 1992

[6] Quarteroni, A.; Sacco, R.; Saleri, F.: *Numerische Mathematik. Band 1 und 2*, Springer Verlag, Berlin 2002

[7] Rheinboldt, W.C.: *Methods for Solving Systems of Nonlinear Equations.* 4th ed., SIAM Publications, Philadelphia 1994.

[8] Saad, Y.: *Iterative Methods for Sparse Linear Systems.* PWS Publishing Company, Boston 1995.

[9] Schwetlick, H.: *Numerische Lösung nichtlinearer Gleichungen.* Deutscher Verlag der Wissenschaften, Berlin 1979.

[10] Trefethen, L.N.; Bau, D.: *Numerical Linear Algebra.* SIAM, Philadelphia 1997.

[11] Vogt, W.: *Zur Numerik nichtlinearer Gleichungssysteme (Teil 1).* Preprint No. M 12/01, IfMath TU Ilmenau, 2001.