# Nichtlineare Optimierung
# 3. Rechnerübungsblatt

TECHNISCHE UNIVERSITÄT DARMSTADT

**Fachbereich Mathematik**
**Prof. Dr. Irwin Yousept**
Hannes Meinlschmidt

**WS 2012/13**
**13.12.2012**

## Rechnerübung

**Aufgabe R1** (Inexaktes CG-Newton-Verfahren)

(a) Implementieren Sie das allgemeine Abstiegsverfahren in der Version von Übungsblatt 7, Aufgabe G2 in `matlab`, bestimmen Sie also die Suchrichtung mit dem inexakten Newton-CG-Verfahren und die Schrittweite nach der Armijo-Regel. Verwenden Sie als Konstanten

$$\alpha = 10^{-3} \quad \text{und} \quad \nu = 10^{-2}.$$

Nutzen Sie zur Bestimmung der Schrittweiten Ihre bereits programmierte Funktion `armijo` von der ersten Rechnerübung. Beachten Sie hierbei, dass die Schrittweiten-Bestimmung nach Armijo auch hier mit $\gamma \in (0, 0.5)$ statt $\gamma \in (0, 1)$ aufgerufen werden soll. Der Funktionskopf des Verfahrens sollte folgendermaßen aussehen:

```
function [xn] = cgnewt(x0,fgH,tol,maxit),
```

wobei wie immer `x0` der Startpunkt, `fgH` eine Funktion, die Funktionswert, Gradient und Hessematrix zurückliefert, `tol` eine Abbruchtoleranz und `maxit` die maximale Anzahl durchzuführender Iterationen sein soll.

In jeder Iteration des Abstiegsverfahrens sollte die aktuelle Iteration, der Funktionswert, die Norm des Gradienten und die Schrittweite ausgegeben werden. Beim CG-Verfahren zur Bestimmung der Suchrichtung soll die Anzahl der benötigten Iterationen und das Restresiduum der Newtongleichung ausgegeben werden.

(b) Testen Sie Ihr Programm an den bereits bekannten Funktionen

- $f_1(x_1, x_2) = x_1^2 + \alpha x_2^2$, mit verschiedenen $\alpha \geq 1$ (z.B. mit $\alpha = 10$ und Startwert $x^0 = (10, 20)$),
- der Rosenbrock-Funktion $f_2(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$ und verschiedenen Startwerten $x^0 \neq (1, 1)$,

sowie der neuen Testfunktion

- $f_5(x_1, x_2) = x_1^4 - 2x_2^2 + x_2^4$ mit Startwert $x^0 = (0.5, 0.5)$.

Als weiterer Test ist zudem ein etwas komplizierteres Problem verfügbar, das *Minimalflächenproblem*:

$$\min_z \int_\Omega \sqrt{1 + \|\nabla z(x)\|^2}\, dx, \quad z = z_b \text{ auf } \partial\Omega.$$

Hier wird eine Funktion $z : \Omega \to \mathbb{R}$ gesucht, die auf dem Rand $\partial\Omega$ eines Gebietes $\Omega$ mit einer gegebenen Funktion $z_b$ übereinstimmt.

Dieses Problem wird diskretisiert auf einem $100 \times 100$-Gitter berechnet. Um eine Lösung berechnen zu lassen, laden Sie die Datei `minsurf.zip` von der Veranstaltungshomepage herunter und kopieren Sie die Dateien `runminsurfcgnewt.m` und `Fminsurf.m` in das Verzeichnis, in der auch die Funktion `cgnewt` liegt. Rufen Sie dort dann `runminsurfcgnewt` in `matlab` auf.

Zum Vergleich des CG- mit dem üblichen Newton-Verfahren, wie in der zweiten Rechnerübung implementiert, passen Sie den Funktionskopf Ihres dort programmierten Newton-Verfahrens auf folgendes an:

```
function [xn] = newt(x,fgH,tol,maxit).
```

Entpacken Sie auch die Datei `runminsurfnewt.m` aus dem Archiv `minsurf.zip` ins selbe Verzeichnis wie Ihr Newton-Verfahren, und rufen Sie die Funktion in Matlab auf.

(c) Vergleichen Sie die Ergebnisse, die Anzahl der Iterationen und die Laufzeit des Newton-CG-Verfahrens mit dem globalisierten Newton-Verfahren vom zweiten Rechnerübungsblatt.

*Hinweis:* Verwenden Sie hier die `matlab`-Befehle `tic`,`toc` und `cputime`.

**Algorithmus 1** : Globalisiertes BFGS-Verfahren

Wähle $\gamma \in (0, 0.5)$ und $\eta \in (\gamma, 1)$. Wähle einen Startpunkt $x^0 \in \mathbb{R}^n$ und eine symmetrische, positiv definite Matrix $B_0 = H_0^{-1} \in \mathbb{R}^{n \times n}$;
**for** $k = 0, 1, \ldots$ : **do**
  **if** $\nabla f(x^k) = 0$ **then**
    STOP mit Ergebnis $x^k$;
  **end**
  Berechne $s^k = -B_k \nabla f(x^k)$;
  Bestimme eine Schrittweite $\sigma_k > 0$ nach der Powell-Wolfe-Regel;
  Setze $x^{k+1} = x^k + \sigma_k s^k$;
  Berechne $B_{k+1} = H_{k+1}^{-1}$ nach dem inversen BFGS-Update

$$B_{k+1} = B_k + \frac{(d^k - B_k y^k){d^k}^T + d^k(d^k - B_k y^k)^T}{{y^k}^T d^k} - \frac{(d^k - B_k y^k)^T y^k}{({y^k}^T d^k)^2} d^k {d^k}^T,$$

  mit $d^k = x^{k+1} - x^k$ und $y^k = \nabla f(x^{k+1}) - \nabla f(x^k)$, vgl. Vorlesung;
**end**

**Aufgabe R2** (Globalisiertes BFGS-Verfahren)
Implementieren Sie das globalisierte BFGS-Verfahren (Algorithmus 1) in `matlab`.
Wählen Sie

$$\gamma = 0.001, \quad \theta = 0.9 \quad \text{und} \quad H_0^{-1} = I,$$

und verwenden Sie im Algorithmus, wie in den vorigen Rechnerübungen gelernt, eine relaxierte Abbruchbedingung an die Norm des Gradienten. Verwenden Sie die Funktion zur Bestimmung der Powell-Wolfe-Schrittweite vom zweiten Rechnerübungsblatt.
Testen Sie Ihr Verfahren an den Funktionen $f_1$ und $f_2$ aus der vorigen Aufgabe. Vergleichen Sie auch die Anzahl der Iterationen und die Laufzeit des BFGS-Algorithmus mit dem globalisierten- und dem CG-Newton-Verfahren für diese Funktionen.

## Hausübung

**Aufgabe H1** (Trust-Region-Modelle) (4 Punkte)
Sei $f : \mathbb{R}^2 \to \mathbb{R}$ gegeben durch einen kleinen Bruder der Rosenbrock-Funktion, nämlich

$$f(x_1, x_2) = 10(x_2 - x_1^2)^2 + (1 - x_1)^2.$$

Das quadratische Modell dieser Funktion ist gegeben durch

$$q(s) := f(x) + \nabla f(x)^T s + \frac{1}{2} s^T \nabla^2 f(x) s.$$

Zeichnen Sie jeweils die Höhenlinien des quadratischen Modells im Punkt $x_1 = (0, -1)^T$ und $x_2 = (0, 0.5)^T$ (z.B. mit `matlab`) und fügen Sie die Trust Regions, die durch $\Delta \in \{0.5, 1, 1.5, 2\}$ gegeben sind, ein. Geben Sie anhand der Bilder an, wo sich die Lösung des jeweiligen TR-Problems befindet. Vergleichen Sie auch die tatsächlichen Höhenlinien von $f$ und die Trust Regions um $x_2$.

**Aufgabe H2** (Cauchy-Punkt) (5 Punkte)
In der Vorlesung wurde der *Cauchy-Punkt* als Näherungslösung für das Trust-Region-Problem eingeführt. Der Cauchy-Punkt $s^c$ im Punkt $x \in \mathbb{R}^n$ soll dabei das folgende Minimierungsproblem lösen:

$$\min q(s) := \nabla f(x)^T s + \frac{1}{2} s^T H s \quad \text{s.t.} \quad s = -t \frac{\nabla f(x)}{\|\nabla f(x)\|}, \quad t \in [0, \Delta],$$

mit $\Delta > 0$, $f : \mathbb{R}^n \to \mathbb{R}$ zwei mal stetig differenzierbar und $H \in \mathbb{R}^{n \times n}$ symmetrisch.

(a) Wir betrachten zunächst die Funktion

$$\phi : \mathbb{R} \to \mathbb{R} : t \mapsto \alpha t + \beta t^2 \quad \text{mit } \alpha, \beta \in \mathbb{R} \text{ und } \alpha < 0.$$

Zeigen Sie, dass das Problem

$$\min \phi(t) \quad \text{s.t.} \quad 0 \le t \le \tau$$

für jedes $\tau > 0$ genau eine Lösung $t^*$ besitzt, und dass folgende Abschätzung gilt:

$$\phi(t^*) \le \frac{\alpha}{2} \min \left\{ \frac{|\alpha|}{2|\beta|}, \tau \right\} \tag{1}$$

(dabei ist für $\beta = 0$ der erste Teil im Minimum als $+\infty$ zu lesen).

(b) Wenden Sie nun Aufgabenteil (a) nun mit einer geeigneten Funktion $\phi$ und geeignetem $\tau > 0$ an, um den Cauchy-Punkt zu berechnen. Zeigen Sie zudem:

$$q(s^c) \le -\frac{\|\nabla f(x)\|}{2} \min \left\{ \frac{\|\nabla f(x)\|}{\|H\|}, \Delta \right\}.$$

**Aufgabe H3** (Lösen des Trust-Region-Problems) (6 Punkte)
Sei $H \in \mathbb{R}^{n \times n}$ symmetrisch und $c \in \mathbb{R}^n$. Für die quadratische Funktion $q : \mathbb{R}^n \to \mathbb{R}$,

$$q(s) = c^T s + \frac{1}{2} s^T H s,$$

betrachten wir für $\Delta > 0$ das Trust-Region-Problem

$$\min q(s) \quad \text{s.t. } \|s\| \le \Delta. \tag{TP}$$

In der nächsten Vorlesung werden wir folgende Charakterisierung der Lösung des Trust-Region-Problems beweisen:
**Satz.** *Der Vektor $\bar{s}$ ist* genau dann *eine Lösung des Trust-Region-Problems* (TP)*, falls ein $\lambda \ge 0$ existiert, so dass die Bedingungen*

*(a)* $\|\bar{s}\| \le \Delta$,

*(b)* $(H + \lambda I)\bar{s} = -c$ ,

*(c)* $(H + \lambda I)$ *positiv semidefinit, und*

*(d)* $(\|\bar{s}\| - \Delta)\lambda = 0$, *d.h. ist $\lambda > 0$, so muss $\|\bar{s}\| = \Delta$ gelten bzw. ist $\|\bar{s}\| < \Delta$, dann ist schon $\lambda = 0$*

*erfüllt sind.*
Wir betrachten nun Problem TP mit den gegebenen Werten $\Delta = \sqrt{2}$,

$$c = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \text{und} \quad H = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

(a) Geben Sie die notwendigen und hinreichenden Optimalitätsbedingungen aus dem Satz konkret an.

(b) Berechnen Sie $s(\lambda) := -(H + \lambda I)^{-1} c$, wobei $\lambda > \max\{0, -\lambda_{\min}(H)\}$ und $p(\lambda) := \|s(\lambda)\|$.

(c) Bestimmen Sie die Optimallösung des Trust-Region-Problems. Zeichen Sie die Funktionen $p(\lambda) - \Delta$ und $\frac{1}{p(\lambda)} - \frac{1}{\Delta}$ als Funktionen in $\lambda$ auf einem geeigneten Intervall. Was fällt Ihnen auf?