Universität Konstanz
Fachbereich Mathematik und Statistik
Prof. Dr. Stefan Volkwein
Tim Seger

25. Juni 2009

Optimierung
5. Übungsblatt

Alle Aufgaben sind zum Vortragen in den Übungen am Dienstag, 07. Juli 2009, vorzubereiten. Die mit ✏ gekenntzeichneten Aufgaben sind auch schriftlich auszuarbeiten und am Donnerstag, 02. Juli 2009, 10 Uhr in den Briefkasten Nr. 16 abzugeben.

Aufgabe 5.1 ✏ Eine einfache Strategie, um das Trust-Region-Hilfsproblem (5.4) der Vorlesung approximativ zu lösen, beruht auf dem Verfahren des steilsten Abstiegs, unter Berücksichtigung des Vertrauensradius Δ_k: Betrachte mit den Bezeichnungen der Vorlesung die Aufgaben

$$(1)\begin{cases}\min\limits_{p\in\mathbb{R}^n} f(x_k)+\nabla f(x_k)^T p\\ \quad\text{u. d. N. } \|p\|\le\Delta_k\end{cases},\quad (2)\begin{cases}\min\limits_{0\le t\le 1} m_k(x_k+tp_k)\\ \text{u. d. N. } \|tp_k\|\le\Delta_k\end{cases}$$

wobei der Vektor p_k in (2) die Lösung von (1) sein soll. Mit der Lösung t_k von (2) definiert man den Cauchy-Punkt durch $x_k^{CP} := x_k + t_k p_k$.

a) Finden Sie[1] die Lösung des Problems (1).

b) Zeigen Sie, dass die Lösung von (2) gegeben ist durch

$$t_k = \begin{cases}1, & \text{falls } \nabla f(x_k)^T H_k \nabla f(x_k)\le 0\\ \min\left(1, \dfrac{\|\nabla f(x_k)\|^3}{\Delta_k \nabla f(x_k)^T H_k \nabla f(x_k)}\right) & \text{sonst.}\end{cases}$$

Aufgabe 5.2 ✏ Ein Verfahren zur approximativen Lösung des Trust-Region-Hilfsproblems (5.4) ist das Dogleg Verfahren. Dabei wird in jeder Iteration anstelle des Problems (5.4) das folgende Problem gelöst:

$$\min_{0\le t\le 2} m_k(x_k(t)) \text{ u. d. N. } \|x_k(t)\|\le\Delta_k \tag{3}$$

mit dem stückweise linearen Pfad

$$x_k(t) = \begin{cases} x_k + t\left(x_k^{CP}-x_k\right) & \text{für } 0\le t\le 1\\ x_k^{CP} + (t-1)(x_k^N - x_k^{CP}) & \text{für } 1\le t\le 2,\end{cases}$$

wobei x_k^{CP} der Cauchy-Punkt aus Aufgabe 5.1 und $x_k^N := x_k - H_k^{-1}\nabla f(x_k)$ der Newton-Schritt ist. Wir gehen hier davon aus, dass die Approximation der Hessematrix H_k positiv definit ist.[2]

a) Zeigen Sie: $\|x_k(t)\|$ ist streng monoton wachsend und $m_k(x_k(t))$ ist streng monoton fallend.

[1] entweder mit elementaren Überlegungen oder auch mit der Methode der Lagrange-Multiplikatoren aus Analysis II

[2] Dies liefert $\langle x_k^N - x_k^{CP}, x_k^{CP} - x_k\rangle > 0$, was für Teil a) hilfreich sein dürfte.

b) Warum sind diese Eigenschaften hilfreich beim Lösen des Problems (3)?

c) Entwerfen Sie einen Algorithmus zur Lösung von (3).

Aufgabe 5.3 Sei $f(x) := 10(x_2 - x_1^2)^2 + (1 - x_1)^2$ die „abgeschwächte" ROSENBROCK-Funktion. Zeichnen Sie bei $x_0 = (0, -1)$ die Höhenlinien des quadratischen Modells

$$m_{x_0}(p) := f(x_0) + \nabla f(x_0)^T p + \frac{1}{2} p^T \nabla^2 f(x_0) p.$$

Zeichnen Sie die Lösungsschar des Problems

$$\min_{p \in \mathbb{R}^n} m_{x0}(p) \quad \text{u. d. N. } \|p\| \leq \Delta_k, \tag{4}$$

wenn der Vertrauenradius Δ_k das Intervall $[0, 2]$ durchläuft.

ALGORITHMUS **(Globalisiertes Newton-Verfahren).**

(S.0) Seien $x_0 \in \mathbb{R}^n$, $\varrho > 0$, $p > 2$, $\beta \in (0, 1)$, $\sigma \in (0, \frac{1}{2})$ und $\varepsilon > 0$ gegeben. Setze $k := 0$.

(S.1) Falls $\|\nabla f(x_k)\| \leq \varepsilon$: STOP.

(S.2) Versuche die NEWTON-Gleichung $\nabla^2 f(x_k) d_k = -\nabla f(x_k)$ zu lösen. Ist diese nicht lösbar, oder ist die Bedingung $\nabla f(x_k)^T d_k \leq -\varrho \|d_k\|^p$ nicht erfüllt, so setze $d_k := -\nabla f(x_k)$.

(S.3) Wähle mit der ARMIJO-Regel eine Schrittweite t_k.

(S.4) Setze $x_{k+1} := x_k + t_k d_k$, $k := k + 1$ und gehe zu (S.1).

Aufgabe 5.4 Programmieraufgabe[3] (in Zweiergruppen zu bearbeiten)
Implementieren Sie das globalisierte NEWTON-Verfahren in Matlab. Erstellen Sie dazu eine Funktion

```
function X = global_newton(fhandle, x0, epsilon, t0, alpha, beta, rho, p).
```

Dabei bezeichnet `x0` den Startpunkt, `epsilon` den Parameter für das Abbruchkriterium, `t0`, `alpha` und `beta` die Parameter für die ARMIJO-Schrittweitensuche und `rho` und `p` die im Algorithmus verwendeten Parameter, um eine gute Abstiegsrichtung zu gewährleisten. Zurückgegeben werden soll eine Matrix `X`, welche den gesamten Iterationsverlauf enthält. Geben Sie ferner die Anzahl der benötigten Iterationen, den letzten Näherungspunkt, die Norm des Gradienten und den Funktionswert am letzten Näherungspunkt auf den Bildschirm aus.

Testen Sie das Verfahren mit `t0 = 1`, `alpha =` 10^{-2}, `beta = 0.5`, `p=3` und `rho=1` an den folgenden Beispielen und visualisieren Sie den Iterationsverlauf im eindimensionalen Fall bzw. zweidimensionalen Fall in einem `plot` bzw. `contourplot` mit Hilfe der zurückgegebenen Matrix `X`.

a) An der Cosinus-Funktion mit `x0 = 1.1656` und `epsilon=` 10^{-3}.

b) An der Funktion von HIMMELBLAU:

$$f(x) := (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2,$$

mit `epsilon=` 10^{-1}, und den Startwerten `x0 = [-0.27;-0.91]`, `x0 = [-0.271;-0.91]`, `x0 = [-0.25;-1.1]`, `x0 = [-0.25;-1]`.

c) An der ROSENBROCK-Funktion mit `x0 = [1;-0.5]`, `epsilon=` 10^{-2}.

---

[3] HINWEISE: Der Programmtext ist angemessen zu kommentieren. Hier ist es günstig, wenn Funktionwert, Gradient und Hessematrix der Zielfunktion zusammen abgerufen werden können. Die Abfrage der Lösbarkeit des auftretenden LGS kann z.B. mit `if rcond(`H_k`) < eps` durchgeführt werden.