Fachbereich Mathematik
Prof. Dr. Stefan Ulbrich
M.Sc. Franziska Kartzow
Dipl.-Math. Sebastian Pfaff

TECHNISCHE
UNIVERSITÄT
DARMSTADT

WS 2010/2011
3. Dezember 2010

3. Aufgabenblatt des Rechnerpraktikums zur „Nichtlinearen Optimierung"

Aufgabe P1 (Inexaktes Newton-CG-Verfahren)

(a) Implementieren Sie das allgemeine Abstiegsverfahren (Algorithmus 3) in der Version von Aufgabe H2 vom Übungsblatt 5 in `Matlab`: Bestimmen Sie die Suchrichtung mit dem inexakten Newton-CG-Verfahren und die Schrittweite nach der Armijo-Regel. Verwenden Sie

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

Nutzen Sie zur Bestimmung der Schrittweiten wieder Ihre Funktion `armijo`. Beachten Sie, dass die Schrittweiten-Bestimmung nach Armijo auch hier mit $\gamma \in (0, \frac{1}{2})$ statt $\gamma \in (0, 1)$ aufgerufen werden soll. Der Funktionskopf Ihres Verfahrens sollte folgendermaßen aussehen:

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

Dabei sei `x` der Startpunkt, `fgH` eine Funktion, die Funktionswert, Gradient und Hessematrix zurückliefert, `tol` die Abbruchtoleranz und `maxit` die maximale Anzahl durchzuführender Iterationen.
In jeder Iteration des Abstiegsverfahrens sollte die Iteration, der Funktionswert, die Norm des Gradienten und die Schrittweite ausgegeben werden. Beim CG-Verfahren zur Bestimmung der Suchrichtung, sollte die Anzahl der benötigten Iterationen und das Restresiduum der Newtongleichung ausgegeben werden. Nutzen Sie dazu den Befehl `fprintf`.

(b) Testen Sie Ihr Programm an den Funktionen

- $f_1(x_1, x_2) = x_1^2 + \alpha x_2^2$, mit verschiedenen $\alpha \geq 1$
  – insbesondere mit $\alpha = 10$ und Startwert $x_0 = (10, 20)$,
- $f_2(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$ (globales Minimum bei $(1, 1)$)

jeweils mit verschiedenen Startwerten Ihrer Wahl und

- $f_3(x_1, x_2) = x_1^4 - 2x_2^2 + x_2^4$ mit Startwert $x_0 = (0.5, 0.5)$.

Aufgabe P2 (BFGS-Verfahren)

Implementieren Sie das folgende globalisierte BFGS-Verfahren in `Matlab` :

Algorithmus GLOBALISIERTES BFGS-VERFAHREN

Initialisierung: Wähle $\gamma \in (0, 0.5)$ und $\theta \in (\gamma, 1)$. Wähle einen Startpunkt $x_0 \in \mathbb{R}^n$ und eine symmetrische, positiv definite Matrix $B_0 = H_0^{-1} \in \mathbb{R}^{(n,n)}$.

Für $k = 0, 1, \cdots$:

1. Falls $\nabla f(x_k) = 0$: STOP mit Ergebnis x_k.

2. Berechne $s_k = -B_k \nabla f(x_k)$.

3. Bestimme eine Schrittweite $\sigma_k > 0$ nach der Powell-Wolfe-Regel.

4. Setze $x_{k+1} = x_k + \sigma_k s_k$

5. 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 $d_k = x_{k+1} - x_k, \quad y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$ (siehe auch Skript).

Wählen Sie

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

und verwenden Sie – wie bisher – eine relaxierte Abbruchbedingung an die Norm des Gradienten. Verwenden Sie Ihre `PowellWolfe`- Funktion zur Bestimmung der Powell-Wolfe-Schrittweiten.

Testen Sie ihr Verfahren an den Funktionen f_1, f_2 und f_3 aus der ersten Programmieraufgabe. Vergleichen Sie die Ergebnisse, die Anzahl der Iterationen und die Laufzeit dieses Programms mit denen des inexakten CG-Newton-Verfahrens und des globalisierten Newton-Verfahrens aus der zweiten Rechnerübung.

Hinweis: Verwenden Sie die `Matlab`-Befehle `tic`,`toc` und `cputime`

Hausübungen:

Die Hausübungen für diese Woche finden Sie auf der Homepage zur Veranstaltung (6. Übungsblatt).

Matlabdokumentation im Internet:

http://www-m3.mathematik.tu-muenchen.de/m3/ftp/matlab.ps