TU Chemnitz
C. Helmberg

Chemnitz, 13. April, 2006

Nichtlineare Optimierung
Übung 2

1. (2 Punkte) Sei $C \subseteq \mathbb{R}^n$ konvex. Eine Funktion $f : C \to \mathbb{R}$ heißt *quasikonvex*, falls für alle $x_1, x_2 \in C$ und alle $\alpha \in (0,1)$ gilt, dass $f(\alpha x_1 + (1-\alpha)x_2) \leq \max\{f(x_1), f(x_2)\}$. Zeige: f ist quasikonvex genau dann, wenn alle Niveaumengen $S_f(r) = \{x \in C : f(x) \leq r\}$ konvex sind. Sind alle lokalen Minima auch globale Minima?

2. (3 Punkte) Sei $C \subseteq \mathbb{R}^n$ konvex. Eine quasikonvexe Funktion $f : C \to \mathbb{R}$ heißt *streng quasikonvex*, falls für alle $x_1, x_2 \in C$ mit $f(x_1) \neq f(x_2)$ und alle $\alpha \in (0,1)$ gilt, dass $f(\alpha x_1 + (1-\alpha)x_2) < \max\{f(x_1), f(x_2)\}$. Zeige: Ist f streng quasikonvex, dann ist jedes lokale Minimum auch globales Minimum und die Menge der Minima $\operatorname{Argmin}(f)$ ist konvex. Was geht verloren, wenn man in der Definition von streng quasikonvex nicht fordert, dass f quasikonvex ist?

3. (3 Punkte) Sei $f : [a,b] \to \mathbb{R}$ streng quasikonvex und $a \leq \lambda < \mu \leq b$. Zeige: Falls $f(\lambda) < f(\mu)$, dann ist $f(\alpha) \geq f(\mu)$ für $\alpha \in [\mu, b]$, sonst ist $f(\alpha) \geq f(\lambda)$ für $\alpha \in [a, \lambda]$.

4. (4 Punkte) Direkte Suchverfahren: Sei $f : [a,b] \to \mathbb{R}$ streng quasikonvex und f durch ein Orakel 0-ter Ordnung gegeben.

   (a) Beschreibe einen Algorithmus, der in Iteration $0 \leq k \leq n-1$ das Suchintervall $[a_k, b_k]$ für ein Minimum verkleinert, indem er an den Stellen
   $$\lambda_k = a_k + \frac{F_{n-k-1}}{F_{n-k+1}}(b_k - a_k) \text{ und } \mu_k = a_k + \frac{F_{n-k}}{F_{n-k+1}}(b_k - a_k).$$
   die Funktion auswertet und 3. nutzt. Dabei ist $F_0 = F_1 = 1$ und $F_{n+2} = F_{n+1} + F_n$, $n \in \mathbb{N}_0$, die Fibonacci Folge. Wieviele Auswertungen werden benötigt?

   (b) Verfahren des Goldenen Schnitts: Verwende nun $\theta = \frac{\sqrt{5}-1}{2}$ und
   $$\lambda_k = a_k + (1-\theta)(b_k - a_k), \quad \mu_k = a_k + \theta(b_k - a_k).$$
   Um wieviel verkleinert sich das Suchintervall pro Auswertung?

5. (8 Punkte) Implementieren Sie in matlab eine Linesearch-Funktion der Form

   ```
   [npoint,nval,ngrad]=linesearch(@fun,cpoint,dir,maxa,cval,cgrad,c1,c2);
   ```

   Sie soll für eine stetig differenzierbare Funktion f eine Interpolationssuche auf $f(x_c + \alpha p)$ für $\alpha \in [0, \bar{\alpha}]$ durchführen, so dass die Wolfe-Bedingungen für die Parameter c_1 und c_2 erfüllt sind. Dabei steht `fun` für f, `cpoint` für x_c, `dir` für die Abstiegsrichtung p, `maxa` für $\bar{\alpha}$, `cval` für $f(x_c)$ und `cgrad` für $\nabla f(x_c)$. Rückgabewerte sind der neue Punkt, der neue Funktionswert und Gradient. Alle Vektoren/Punkte sind Spaltenvektoren. Für `fun` soll eine beliebige Funktion (siehe "`help feval`" in matlab) der Form

   ```
   [val,grad]=function_name(point)
   ```

   eingesetzt werden können.