

# Nichtlineare Optimierung II - Restringierte Optimierung

## 7. Übung

Wintersemester 2011/2012



Bergische Universität Wuppertal

Fachbereich C – Angewandte Mathematik / Optimierung und Approximation  
Prof. Dr. Kathrin Klamroth, Dipl. Math. Markus Kaiser

Besprechung des Übungsblattes: 20.06.2012, 14-16 Uhr, D 13.15

Bitte beachten Sie für die Programmieraufgabe:  
Senden Sie Ihr lauffähiges Programm bis spätestens 18 Uhr am Dienstag den 19.06.2012 an [kaiser@math.uni-wuppertal.de](mailto:kaiser@math.uni-wuppertal.de).  
Die Programmieraufgaben dürfen in Zweiergruppen abgegeben werden.  
Achten Sie bitte darauf, genau die angegebenen Ein- und Ausgabeparameter zu verwenden und ersetzen Sie im Funktionsnamen "NACHNAME" durch Ihre(n) Nachnamen.

### Aufgabe 23:

Implementieren Sie eine Funktion

```
function [xopt] = NACHNAME_actset(Q,c,gamma,B,beta,A,alpha),
```

um das quadratische Optimierungsproblem

$$(QP) \quad \begin{array}{ll} \min & \frac{1}{2}x^T Qx + c^T x + \gamma \\ \text{s.t.} & b_j^T = \beta_j \quad j = 1, \dots, p \\ & a_i^T \leq \alpha_i \quad i = 1, \dots, m \end{array}$$

mit einer symmetrischen Matrix  $Q := Q \in \mathbb{R}^{n \times n}$ ,  $c := c \in \mathbb{R}^n$ ,  $\gamma := \gamma \in \mathbb{R}$ ,

$$B := \begin{pmatrix} b_1^T \\ \vdots \\ b_p^T \end{pmatrix} \in \mathbb{R}^{p \times n}, \quad A := \begin{pmatrix} a_1^T \\ \vdots \\ a_m^T \end{pmatrix} \in \mathbb{R}^{m \times n}, \quad \text{beta} := \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} \in \mathbb{R}^p \text{ und}$$

$\alpha := \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_m \end{pmatrix} \in \mathbb{R}^m$  mit Hilfe der Strategie der aktiven Menge (Algorithmus 13.3 aus der Vorlesung) zu lösen.

*Hinweise: Verwenden Sie die erste Phase des Simplex-Verfahrens, um eine zulässige Startlösung in Schritt 0 zu bestimmen und das CG-Verfahren aus Aufgabe 4 des ersten Übungsblattes um die linearen Gleichungssysteme in Schritt 2 zu lösen.*

*Sollten Sie keine Implementierung des Simplexverfahrens aus der Vorlesung zur linearen Optimierung besitzen, können Sie z.B. die Matlab Funktion `linprog` zum Lösen des in Phase 1 auftretenden linearen Optimierungsproblems verwenden.*

*Auf der Vorlesungshomepage finden Sie Beispiele, um Ihre Implementierung zu testen.*



*Bemerkung: Aktuelle Informationen zur Vorlesung und zu den Übungen finden Sie im Internet unter:*

<http://www2.math.uni-wuppertal.de/opt/NLO/>