

Nichtlineare Optimierung

4. Rechnerübungsblatt



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Fachbereich Mathematik
Prof. Dr. Irwin Yousept
Hannes Meinlschmidt

WS 2012/13
7.2.2013

Rechnerübung

Aufgabe R1 (Quadratisches Penalty-Verfahren)

- (a) Programmieren Sie das quadratische Penalty-Verfahren aus Algorithmus 16 in `matlab`, minimieren Sie also die Penalty-Funktion

$$P_\alpha(x) = f(x) + \frac{\alpha}{2} \left(\|(g(x))_+\|_2^2 + \|h(x)\|_2^2 \right).$$

Verwenden Sie dazu Ihre Implementierung des globalen Newton-Verfahrens und des BFGS-Verfahrens aus den vergangenen Rechnerübungen.

Erhöhen Sie den Penalty-Parameter (Schritt 3 des Penalty-Verfahrens) um den Faktor 10, also durch die Vorschrift $\alpha_{k+1} = 10\alpha_k$, und wählen Sie $\alpha_0 = 1$. Verwenden Sie für das äußere Verfahren – also das quadratische Penalty-Verfahren – die Abbruchbedingung $\|(g(x^k))_+\| + \|h(x^k)\| \leq 10^{-4}$. Führen Sie als zusätzliches Abbruchkriterium eine maximale Anzahl an äußeren Iterationen ein. Geben Sie jeweils aus, wieviele Iterationen das globale Newton- bzw. das BFGS-Verfahren in jeder äußeren Iteration laufen musste.

Hinweis: Sie müssen in jeder Iteration des äußeren Verfahrens die quadratische Penaltyfunktion P_α mit neuem Parameter τ an das globale Newton- bzw. BFGS-Verfahren übergeben. Zuweisungen der Form $f(x) := g(x, y)$ für einen festen Wert y können in `matlab` durch den Ausdruck `f=@(x)g(x,y)` realisiert werden. Für die Übergabe mehrerer Funktions-Objekte könnte sich zudem der Datentyp `Cell Array` als nützlich erweisen.

- (b) Testen Sie Ihr Verfahren an dem Problem

$$\min 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \quad \text{s.t.} \quad 3x_1 + x_2 \leq 0 \quad (\text{P1})$$

mit Startwerten $x^0 = (-1, 0.5)^T$ und $x^0 = (4, 5)^T$ (wieso gerade diese?), sowie an dem Problem

$$\begin{aligned} \min \quad & 1000 - x_1^2 - 2x_2^2 - x_3^2 - x_1x_2 - x_1x_3 \\ \text{s.t.} \quad & \begin{cases} x_1^2 + x_2^2 + x_3^2 - 25 = 0 \\ 8x_1 + 14x_2 + 7x_3 - 56 = 0 \end{cases} \end{aligned} \quad (\text{P2})$$

mit Startwert $x^0 = (3, 0.2, 3)^T$.

Aufgabe R2 (SQP-Verfahren und mehr)

- (a) Überlegen Sie sich, welche Veränderungen und Erweiterungen Ihrer (hoffentlich) schon vorhandenen Codes nötig, und welche spezielle Zusatzroutinen nützlich wären, um das globalisierte SQP-Verfahren aus der Vorlesung zu implementieren.
- (b) Auf <http://www.neos-server.org/> finden Sie eine Bibliothek an Optimierungsalgorithmen. Diese können online mit Problemdaten gefüttert werden. Dies geschieht vorzugsweise über die Modellierungssprache AMPL, siehe auch den [AMPL Input] Link neben den jeweiligen Algorithmen und dort das Model File-Feld. Dieses erwartet eine .mod Datei (Zusatzdaten und -befehle können als Data File bzw. Command File übergeben werden, das werden wir aber nicht brauchen). Wählen Sie 3 Solver aus (z.B. `filter` ist ein SQP-Algorithmus) und vergleichen Sie die Ergebnisse, die die jeweiligen Solver ausspucken. AMPL-Dateien finden Sie z.B. auf

<http://netlib.bell-labs.com/netlib/ampl/models/nlmodels/index.html>

oder

http://plato.asu.edu/ftp/ampl_files/

Achten Sie darauf, nicht zu große Probleme an den Solver zu übergeben (z.B. finden sich auf der zweiten angegebenen Seite riesige Probleme, die aus Diskretisierungen von Optimierungsproblemen mit elliptischen bzw. parabolischen partiellen Differentialgleichungen kommen). Generell ist auch die Seite <http://plato.asu.edu> interessant.

- (c) Übersetzen Sie die Probleme (P1) und (P2) aus Aufgabe R1 in AMPL und vergleichen Sie Ihre Implementierung des Penalty-Verfahrens mit den online bereitgestellten Algorithmen.

Hausübung

Aufgabe H1 (Lokales SQP-Verfahren)

(4 Punkte)

Sei folgendes Optimierungsproblem gegeben:

$$\begin{aligned} \min \quad & (x_1 - x_2)^2 + (x_1 - 1)^2 \\ \text{s.t.} \quad & \begin{cases} x_1^2 - 1 + x_2 \leq 0 \\ x_1^2 - 1 - x_2 \leq 0 \end{cases} \end{aligned} \quad (\text{NLP1})$$

Berechnen Sie ausgehend vom Startpunkt $(x^0, \lambda^0) = ((-2, -1)^T, (0, 0)^T)$ einen Schritt des lokalen SQP-Verfahrens (Algorithmus 19.7 im Buch) für (NLP1).

Aufgabe H2 (Probleme beim SQP-Verfahren, Teil 1: Unzulässige Teilprobleme)

(4 Punkte)

Wir betrachten

$$\min -x_1 - x_2 \quad \text{s.t.} \quad x_1^2 + x_2^2 - 1 = 0, \quad x_1, x_2 \geq 0. \quad (\text{NLP2})$$

- (a) Bestimmen Sie die Lösung von (NLP2) mit zugehörigen Lagrangemultiplikatoren.
(b) Zeigen Sie, dass der zulässige Bereich des SQP-Teilproblems im Punkt $x = -(1/2, 1/2)^T$ leer ist.

Bemerkung. Um dieses Problem zu umgehen, kann man z.B. die Nebenbedingungen leicht relaxieren, um sicher zulässige Teilprobleme zu erzeugen, und gleichzeitig die Verletzung gegenüber den tatsächlichen Nebenbedingungen mit in die Zielfunktion aufnehmen, siehe Literatur.

Aufgabe H3 (Probleme beim SQP-Verfahren, Teil 2: Maratos-Effekt)

(? Punkte)

Betrachten Sie das Optimierungsproblem

$$\min 2(x_1^2 + x_2^2 - 1) - x_1 \quad \text{s.t.} \quad x_1^2 + x_2^2 - 1 = 0. \quad (\text{NLP3})$$

- (a) Zeigen Sie, dass $\bar{x} = (1, 0)^T$ das globale Minimum von (NLP3) ist und bestimmen Sie den zugehörigen Lagrangemultiplikator $\bar{\mu}$. Zeigen Sie, dass das lokale SQP-Verfahren für Startpunkte $(x^0, \mu^0) \in B_\delta(\bar{x}, \bar{\mu})$, mit $\delta > 0$ klein genug, Q-superlinear gegen $(\bar{x}, \bar{\mu})$ konvergiert.
(b) Seien nun $x^k \in \{x \in \mathbb{R}^2 : x_1^2 + x_2^2 - 1 = 0\} \setminus \{\pm \bar{x}\}$ und $\mu^k < -1$ beliebig. Zeigen Sie, dass für die Lösung s^k des SQP-Teilproblems (SQP_k) gilt, dass $P_\alpha^1(x^k + s^k) > P_\alpha^1(x^k)$ ist. Zeigen Sie dazu

$$|h(x^k + s^k)| > |h(x^k)| \quad \text{und} \quad f(x^k + s^k) > f(x^k).$$

Damit wird im globalisierten SQP-Verfahren der volle Schritt von der Bewertungsfunktion P_α^1 abgelehnt. Da die Bedingungen $x^k \in Z \setminus \{\bar{x}\}$ und $\mu^k < -1$ für (x^k, μ^k) beliebig nahe bei $(\bar{x}, \bar{\mu})$ gelten, geht das globalisierte SQP-Verfahren hier also *nicht* zu lokal Q-superlinearer Konvergenz über.

Bemerkung. Hier kann man dem Problem beikommen, indem man z.B. eine sogenannten *Second-Order-Correction* (SOC) der Form

$$s_{\text{SOC}}^k = -\nabla h(x^k) \left(\nabla h(x^k)^T \nabla h(x^k) \right)^{-1} h(x^k + s^k)$$

zur ermittelten Suchrichtung s^k addiert. Diese SOC ist nur eine sehr geringe Änderung im Vergleich zu s^k , verbessert die Zulässigkeit des neuen Punktes $x^k + s^k + s_{\text{SOC}}^k$ allerdings immens – siehe auch hier Literatur.