Universität Konstanz — Wintersemester 16/17
Fachbereich Mathematik und Statistik
Prof. Dr. Stefan Volkwein
Jianjie Lu, Sabrina Rogg

Numerische Verfahren der restringierten Optimierung

http://www.math.uni-konstanz.de/numerik/personen/volkwein/teaching/

Program 1 (6 Points)

Submission by E-Mail to sabrina.rogg@uni-konstanz.de:
29.11.2016, 10:00 h

Note:

- Do not forget to write **name** and **email address** of the authors in each file and document your code well!
- Only **running programs** will be considered!
- Stick to the **given function and parameter definitions** as described below! You should not modify them in name or concerning the input and output arguments.

Consider the domain $\Omega = (a_1, a_2)$ and the following Poisson problem

$$\begin{cases} \mathrm{y}_{xx}(x) = \mathrm{b}(x) & \text{in } \Omega, \\ \mathrm{y}(a_1) = \mathrm{g}(a_1), \\ \mathrm{y}(a_2) = \mathrm{g}(a_2). \end{cases}$$

For numerically solving the problem we consider the discrete points $x_i = a_1 + i \cdot h$, $i = 0, \ldots, n+1$, with $h = (a_2 - a_1)/(n+1)$. For the inner points we obtain (by using central differences) the system $Ay = b \in \mathbb{R}^n$, with $b = \big(\mathrm{b}(x_1) + \mathrm{g}(a_1)/h^2, \mathrm{b}(x_2), \ldots, \mathrm{b}(x_{n-1}), \mathrm{b}(x_n) + \mathrm{g}(a_2)/h^2\big)^\top$, $y \approx \big(\mathrm{y}(x_1), \ldots, \mathrm{y}(x_n)\big)^\top$ and

$$A = \frac{1}{h^2} \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix} \in \mathbb{R}^{n \times n}.$$

We consider the constrained optimization problem

$$\min J(y) = \frac{1}{2} y^\top Q y + y^\top d \quad \text{subject to} \quad Ay = b,$$

with $Q \in \mathbb{R}^{n \times n}$ and $d \in \mathbb{R}^n$. Implement the function

```
[y,lambda]=myquadprog(Q,d,A,b,flag)
```

for solving the linear-quadratic problem via direct solve of the KKT-system, where `y` and `lambda` are column vectors and `flag`$\in$`{1,2,3}` should set the system solver according to the following association: `1` for the QR, `2` for LU and `3` (default) backslash (use the appropriate Matlab functions for the matrix decomposition).
Implement a `mymain` file which will define all the necessary matrices, calls `myquadprog` and plots the results, using the following setting:

$$\mathrm{b}(x) = 2\frac{\cos x}{e^x}, \quad \mathrm{g}(x) = \frac{\sin x}{e^x}$$

and $d = \big(\mathrm{d}(x_1), \ldots, \mathrm{d}(x_n)\big)^\top$ with

$$\mathrm{d}(x) = \frac{\sin 0.2}{e^{0.2}} + (x - 0.2)\left(e^{-x}\cos x - e^{-x}\sin x\right).$$

Test your program with $\Omega = (0, 10)$, $Q = I \in \mathbb{R}^{n\times n}$ and solve the quadratic program using the three solvers for $n \in \{1000, 5000, 10000\}$.
Put the plots of the solutions and your observations in the written report.