UNIVERSITÄT BERN

Applied Optimization
Exercise 9
Equality Constrained Optimization II

November 23, 2021

Hand-in instructions:

Please hand-in **only one** compressed file named after the following convention: `Exercisen-GroupMemberNames.zip`, where n is the number of the current exercise sheet. This file should contain:

- The complete code folder except for the `build` subfolder. Make sure that your code submission compiles on your machine. Zip the code folder.
- A `readme.txt` file containing a description on how you solved each exercise (use the same numbers and titles) and the encountered problems.
- Other files that are required by your `readme.txt` file. For example, if you mention some screenshot images in `readme.txt`, these images need to be submitted too.
- Submit your solutions to ILIAS before the submission deadline.

Problem 1: Constraints Elimination (2 pts)

Solve the following non-lineal optimization problem in $\mathbb{R}^3$ with lineal equality constraints by constraint elimination:

$$\begin{aligned} &\text{minimize } f(x) = \exp(\mathbf{1}^T x) + \exp(-\mathbf{1}^T x) \\ &\text{subject to: } \begin{pmatrix} 1 & -2 & 0 \\ 2 & 1 & 0 \end{pmatrix} x = \begin{pmatrix} 0 \\ 5 \end{pmatrix} \end{aligned}$$

Hint: use Gaussian elimination or QR decomposition. You are allowed to use a computer algebra system to verify your computations.

Problem 2: Solving the KKT system (2 pts)

Solve the following convex quadratic problem in $\mathbb{R}^3$ with lineal equality constraints

$$\text{minimize } f(x) = \begin{pmatrix} -3 & -2 & 7 \end{pmatrix} x + \frac{1}{2} x^T \begin{pmatrix} 6 & 0 & 2 \\ 0 & 4 & 0 \\ 2 & 0 & 6 \end{pmatrix} x$$
$$\text{subject to: } \begin{pmatrix} 1 & 1 & 2 \\ 0 & 1 & 1 \end{pmatrix} x = \begin{pmatrix} 0 \\ 2 \end{pmatrix}$$

Hint: use gaussian elimination or QR decomposition to solve the resulting KKT system.

Programming

Infeasible Start Newton Method (6 pts)

In last exercise, we implemented the Newton method with linear equality constraints. It works only if we have a feasible start point. Now we will practice the infeasible start newton method which is a generalized version. The idea is to minimize the l_2 norm of the residual $r(y) = \begin{pmatrix} \nabla f(x) + A^T \nu \\ Ax - b \end{pmatrix}$, with $y = \begin{pmatrix} x \\ \nu \end{pmatrix}$. By linearization, we can get $r(y + \Delta y) \approx r(y) + \nabla r(y) \Delta y$ and it turns into a root finding problem:

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \nu \end{bmatrix} = - \begin{bmatrix} \nabla f(x) + A^T \nu \\ Ax - b \end{bmatrix}.$$

By solving this linear system, we can compute a searching direction. Afterwards, perform a backtracking line search on $||r||_2$ to get the step length. Then update the primal variable x and dual variable ν. Iterate the process until convergence. For more details, please refer to the slides.

Write your own code in the function `solve_equality_constrained_with_infeasible_start(...)` in the file `NewtonMethods.hh`. In addition, the function `backtracking_line_search_newton_with_infeasible_start` in the file `LineSearch.hh` also needs to be completed.

You can test your implementation by running the `EqualityConstrainedNewtonInfeasibleStart` executable on random mass spring systems and visualize the results on the usual website. Once again, you will find helpful unit tests alongside the main executable.

Hybrid Newton Method (Bonus 2 pts)

In practice, one often starts with infeasible start newton's method. Once the constraints are satisfied, the standard equality constrained newton is activated. Please implement the hybrid version in the function `solve_equality_constrained_hybrid`.