Exercises for Lecture Course on Numerical Optimization (NUMOPT)
Albert-Ludwigs-Universität Freiburg – Winter Term 2019-2020

Exercise 5: Exam Type Questions, Inequality Constrained Optimization

Prof. Dr. Moritz Diehl, Dimitris Kouzoupis, Andrea Zanelli and Florian Messerer

Goal of this sheet is to offer some exposure to exam-type questions. Further you will get more familiar with inequality constrained optimization and get to know more of CasADi's functionalities.

Exercise Tasks

1. **A sample exam question.** *(your score in this question will be divided by 3)*

   Regard the following minimization problem.

$$\boxed{\min_{x\in\mathbb{R}^2} \quad x_2^4+(x_1+2)^4 \quad \text{subject to} \quad \left\{\begin{array}{rcl} x_1^2+x_2^2 & \leq & 8 \\ x_1-x_2 & = & 0 \end{array}\right.}$$

   (a) How many scalar decision variables, how many equality, and how many inequality constraints does this problem have?

2	

   two scalar decision variables, 1 equality constraint, 1 inequality constraint

   (b) Sketch the feasible set $\Omega \in \mathbb{R}^2$ of this problem.

3	

   (c) Bring this problem into the NLP standard form:

$$\min_{x\in\mathbb{R}^n} f(x) \quad \text{subject to} \quad \left\{\begin{array}{rcl} g(x) & = & 0 \\ h(x) & \geq & 0 \end{array}\right.$$

   by defining the functions f, g, h appropriately.

$$f(x) = x_2^4 + (x_1+2)^4$$
$$g(x) = x_1 - x_2$$
$$h(x) = 8 - x_1^2 - x_2^2$$

3	

FROM NOW ON UNTIL THE END TREAT THE PROBLEM IN THIS STANDARD FORM.

(d) Is this optimization problem convex? Justify. $f(x)$ is convex, $g(x)$ is affine, $h(x)$ is concave $\Rightarrow$ the problem is convex

2	

(e) Write down the Lagrangian function of this optimization problem.

$$\begin{aligned}\mathcal{L}(x,\lambda,\mu) &= f(x) - \lambda^\top g(x) - \mu^\top h(x)\\ &= x_2^4 + (x_1+2)^4 - \lambda(x_1 - x_2) - \mu(8 - x_1^2 - x_2^2)\end{aligned}$$

where $\lambda, \mu \in \mathbb{R}$.

2	

(f) A feasible solution of the problem is $\bar{x} = (2,2)^T$. What is the active set $\mathcal{A}(\bar{x})$ at this point? $h(\bar{x}) = 8 - 2^2 - 2^2 = 0 \Rightarrow$ the constraint is active, $\mathcal{A}(\bar{x}) = \{1\}$ (This notation interprets $h(x)$ as vector valued function with only one dimension, i.e. a "scalar vector")

2	

(g) Is the *linear independence constraint qualification (LICQ)* satisfied at $\bar{x}$? Justify.
Check linear independence of $\nabla g(\bar{x})$ and $\nabla h_i(\bar{x})$, $i \in \mathcal{A}$ or whether $\begin{bmatrix}\nabla g(\bar{x}) & \nabla h_1(\bar{x})\end{bmatrix}$ is full rank.

$$\nabla g(x) = \begin{bmatrix}1\\-1\end{bmatrix} = \nabla g(\bar{x}) \qquad \nabla h_1(x) = \begin{bmatrix}-2x_1\\-2x_2\end{bmatrix}, \ \nabla h_1(\bar{x}) = \begin{bmatrix}-4\\4\end{bmatrix}$$

$$\det \begin{bmatrix}\nabla g(\bar{x}) & \nabla h_1(\bar{x})\end{bmatrix} = \det \begin{bmatrix}1 & -4\\-1 & -4\end{bmatrix} = 6 > 0 \Rightarrow \text{full rank} \Rightarrow \text{LICQ satisfied}$$

3	

(h) An optimal solution of the problem is $x^* = (-1,-1)^T$. What is the active set $\mathcal{A}(x^*)$ at this point? $h(x^*) = 6 > 0 \Rightarrow \mathcal{A}(x^*) = \{\}$ (no active inequality constraints)

1	

(i) Is the *linear independence constraint qualification (LICQ)* satisfied at x^* ? Justify.

$$\nabla g(x^*) = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \Rightarrow \text{full rank} \Rightarrow \text{LICQ satisfied}$$

2	

(j) Describe the tangent cone $T_\Omega(x^*)$ (the set of feasible directions) to the feasible set at this point x^*, by a set definition formula with explicitly computed numbers.

LICQ holds at x^*, so the tangent cone and the linearized feasible cone coincide:

$$T_\Omega(x^*) = \mathcal{F}(x^*) = \{p \in \mathbb{R}^n \mid \nabla g_i(x^*)^\top p = 0, i = 1, \ldots, m \;\&\; \nabla h_i(x^*)^\top p = 0, i \in \mathcal{A}(x^*)\}$$

Here:

$$\mathcal{F}(x^*) = \{p \in \mathbb{R}^2 \mid \nabla g(x^*)^\top p = 0\} = \{p \in \mathbb{R}^2 \mid \begin{bmatrix} 1 & -1 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = 0\}$$
$$= \{p \in \mathbb{R}^2 \mid p_1 = p_2\} = \{t \begin{bmatrix} 1 \\ 1 \end{bmatrix} \mid t \in \mathbb{R}\}$$

2	

(k) Compute the Lagrange gradient and find the multiplier vectors λ^*, μ^* so that the above point x^* satisfies the KKT conditions.

general KKT conditions for inequality constraint optimization

$$\begin{aligned} \nabla_x \mathcal{L}(x^*, \lambda^*, \mu^*) = \nabla f(x^*) - \nabla g(x^*)\lambda^* - \nabla h(x^*)\mu &= 0 \\ g(x^*) &= 0 \\ h(x^*) &\geq 0 \\ \mu^* &\geq 0 \\ \mu_i^* h_i(x^*) &= 0, \quad i = 1, \ldots, q \end{aligned}$$

Here:

$h(x^*) > 0 \Rightarrow \underline{\mu^* = 0}$

$g(x^*) = 0 \quad \checkmark$

$$\nabla_x \mathcal{L}(x, \lambda, \mu) = \begin{bmatrix} 4(x_1+2)^3 \\ 4x_2^3 \end{bmatrix} - \begin{bmatrix} 1 \\ -1 \end{bmatrix} \lambda - \begin{bmatrix} -2x_1 \\ -2x_2 \end{bmatrix} \mu$$

$$\nabla_x \mathcal{L}(x^*, \lambda^*, \mu^*) = \begin{bmatrix} 4 - \lambda^* \\ -4 + \lambda^* \end{bmatrix} = 0 \Leftrightarrow \underline{\lambda^* = 4}$$

4	

(l) Describe the critical cone $C(x^*, \mu^*)$ at the point (x^*, λ^*, μ^*) in a set definition using explicitly computed numbers

$$\begin{aligned}\mathcal{C}(x^*, \mu^*) = \{p \in \mathbb{R}^n \mid \quad & \nabla g_i(x^*)^\top p = 0, i = 1, \dots, m \\ & \& \; \nabla h_i(x^*)^\top p = 0, i \in \mathcal{A}_+(x^*) \\ & \& \; \nabla h_i(x^*)^\top p \geq 0, i \in \mathcal{A}_0(x^*)\}\end{aligned}$$

Here ($\mathcal{A} = \{\}$):

$$\mathcal{C}(x^*, \mu^*) = \{p \in \mathbb{R}^2 \mid \nabla g(x^*)^\top p = 0\} = \mathcal{F}(x^*) = \{t \begin{bmatrix} 1 \\ -1 \end{bmatrix} \mid t \in \mathbb{R}\}$$

3	

(m) Formulate the second order necessary conditions for optimality (SONC) for this problem and test if they are satisfied at (x^*, λ^*, μ^*). Can you prove that x^* is a local or global minimizer?

SONC: Regard x^* with LICQ. If x^* is a local minimizer of the NLP, then

i. $\exists \lambda^*, \mu^*$ such that KKT conditions hold
ii. $\forall p \in \mathcal{C}(x^*, \mu^*)$ holds $p^\top \nabla_x^2 \mathcal{L}(x^*, \lambda^*, \mu^*) p \geq 0$

Here:

$$\nabla_x^2 \mathcal{L}(x, \lambda, \mu) = \begin{bmatrix} 12(x_1+2)^2 + 2\mu & 0 \\ 0 & 12x_2^2 + 2\mu \end{bmatrix}, \quad \Lambda^* := \nabla_x^2 \mathcal{L}(x^*, \lambda^*, \mu^*) = \begin{bmatrix} 12 & 0 \\ 0 & 12 \end{bmatrix}$$

check SONC

i. holds due to task (1k)
ii. $\Lambda^* \succ 0 \Rightarrow p^\top \Lambda^* p \geq 0 \; \forall p \in \mathbb{R}^n$, therefore this specifically holds also for $\forall p \in \mathcal{C}(x^*, \mu^*)$

$\Rightarrow$ SONC are satisfied
Due to $\Lambda^* \succ 0$ we furthermore have $p^\top \Lambda^* p > 0 \; \forall p \in \mathbb{R}^n \setminus \{0\}$, and therefore specifically $\forall p \in \mathcal{C}(x^*, \mu^*) \setminus \{0\}$. Thus SOSC also holds, and x^* is a local minimizer. Due to convexity of the NLP this is equivalent to x^* being a global minimizer.
Alternative: Theorem 13.6. *For convex NLP and x^* with LICQ holds:*
x^* is a global minimizer $\Leftrightarrow \exists \lambda, \mu$ such that KKT conditions hold.
We know the righthandside to be true, so x^* is a global minimizer.

4	

2. **Hanging chain, the last episode.** Recall the optimization problem of the hanging chain with non-convex inequality constraints and without considering a rest length:

$$\begin{aligned}
\underset{y,z}{\text{minimize}} \quad & \frac{1}{2}\sum_{i=1}^{N-1} D\left((y_i - y_{i+1})^2 + (z_i - z_{i+1})^2\right) + g\sum_{i=1}^{N} m\, z_i && \text{(1a)}\\
\text{subject to} \quad & (y_1, z_1) \;=\; (-2, 1) && \text{(1b)}\\
& (y_N, z_N) = \; (2, 1) && \text{(1c)}\\
& z_i + y_i^2 \;\geq 0, \qquad \text{for } i = 1, \dots, N. && \text{(1d)}
\end{aligned}$$

with parameters $N = 40$, $D = 70$, $g = 9.81$ and $m = 0.1$.

If you define this problem with `opti = casadi.Opti()`, and add decision variables and constraints with `opti.variable()` and `opti.subject_to()` respectively, CasADi will automatically bring it into the CasADi standard form:

$$\begin{aligned}
\underset{x}{\text{minimize}} \quad & f(x) && \text{(2a)}\\
\text{subject to} \quad & g_{\text{lb}} \leq g(x) \leq g_{\text{ub}} && \text{(2b)}
\end{aligned}$$

where for equality constraints the upper and lower bounds are identical. x is a collection of all decision variables, in order of their definition in the script. You can access these (symbolic) expressions and variables via `opti.x`, `opti.f`, `opti.g`, `opti.lbg` and `opti.ubg`. After solving the problem with `sol = opti.solve()`, you can access the value of any expression at the solution via e.g. `fmin = sol.value(opti.f)`. Values obtained like this are still CasADi matrices. Therefore it might be necessary to first change them to a normal Matlab matrix using `full()` (e.g. `fmin = full(fmin)`) before further processing them.

(a) Solve problem (1) with `Opti()` and IPOPT as you did on the first exercise sheet. Find out the standard form into which CasADi has transformed the problem and write it down on paper, i.e. the relation between x and the y_i and z_i as well as the functions $f(x)$, $g(x)$ and the values of g_{lb} and g_{ub}.

$$x = \begin{bmatrix} y \\ z \end{bmatrix}$$

$$g(x) = \begin{bmatrix} y_1 \\ z_1 \\ y_N \\ z_N \\ z + y^2 \end{bmatrix}, \; g_{\text{lb}} = \begin{bmatrix} -2 \\ 1 \\ 2 \\ 1 \\ 0_{40\times 1} \end{bmatrix}, \; g_{\text{ub}} = \begin{bmatrix} -2 \\ 1 \\ 2 \\ 1 \\ \infty_{40\times 1} \end{bmatrix}$$

where the square of a vector is to be understood elementwise and $0_{40\times 1}$ denotes a zero matrix of dimension 40×1, and correspondingly for $\infty_{40\times 1}$ (2 bonus points)

(b) With `opti.callback()` you can pass a function handle `@(i) DOSOMETHING` with iteration index `i` as input. CasADi will call it every iteration. Create a function that makes a plot of the current chain position and pass it to CasADi via a function handle.

Hint: You can access the non-converged solution via `opti.debug.value()`*. To be able to see something, use* `pause()` *after plotting.*

(2 bonus points)

(c) Identify the active set at the solution. Does LICQ hold? Are there any weakly active constraints?

Note: You can consider a constraint $h_i(x) \geq 0$ as active if $h_i(x) \leq 10^{-6}$

(2 bonus points)

(d) Evaluate the Hessian of the Lagrangian at the solution. You may first build a CasADi expression for the Lagrangian. You can then obtain its Hessian via `hessian()`. You can access the Lagrange multipliers as `opti.lam_g`. Note that as opposed to the lecture, the Lagrangian is defined as $\mathcal{L}(x, \lambda) = f(x) + \lambda^\top g(x)$ in CasADi. This also changes the sign of the multipliers.

(2 bonus points)

(e) Calculate the null-space of the Jacobian of the active constraints at the solution using QR decomposition. Use the result to calculate the reduced Hessian. Is the reduced Hessian positive definite? What does this mean?

Hints: Use `qr()`. *If you do not succeed obtaining the null-space via QR decomposition, you can simply proceed by obtaining the null-space from* `null()`.

(2 bonus points)

This sheet gives in total 11 points and 10 bonus points.
NOTE: your score in the exam-type question will be divided by 3.