Numerical Optimization in Robotics
Homework_4 Hints

Task 1.

First, we introduce a notation H representing the real Euclidean space, then recall a fundamental fact. Let a function $f: H \to [-\infty, +\infty]$, the function f is convex, if and only if the following inequality holds

$$f(\alpha x + (1-\alpha)y) \le \alpha f(x) + (1-\alpha) f(y), \forall x, y \in dom f, \alpha \in (0,1)$$

Based on this fact, we can derive some results as follows.

1. The function $\|\cdot\|^2$ is strictly convex.

2. Let $h(x)$ be affine, since the function $\|\cdot\|^2$ is convex, then we state that $\|h(x)\|^2$ is convex.

3. Let $\{g_i\}$ be a family of convex functions from H to $[-\infty, +\infty]$, then $\sup g_i$ is convex.

4. Suppose that the set $H_s \subset H$ is convex and that the functions $f_1, f_2, \ldots, f_m : H_s \to R$ are convex, and define a function $f: H_s \to R^m$ with components f_i . Suppose further that $f(H_s)$ is convex and that the function $g: f(H_s) \to R$ is convex and not decreasing: any points $y \le z$ in $f(H_s)$ satisfy $g(y) \le g(z)$. We have the composition $g \circ f$ is convex.

Task2.

A C++ version of a solver is provided via https://github.com/ZJU-FAST-Lab/SDQP for low-dimensional strictly convex QP problem. A test example give a semi-definite positive matrix M_Q , the SDQP solver cannot be applied to such problem directly. A "proximal" term is suggested to be included. We have some modified versions for the original problem.

P1. $\min\limits_{x \in R^n} \frac{1}{2} x^T (M_Q + \delta I) x + c_Q^T x, s.t. A_Q x \le b_Q$

P2. $\min\limits_{x \in R^n} \frac{1}{2} x^T M_Q x + c_Q^T x + \rho \|x - \bar{x}\|^2, s.t. A_Q x \le b_Q$

We remark that $\bar{x}$ can be a predetermined quantity. It can also be changed as iterations process. The parameters δ, ρ ensure that P1 and P2 can be solved by SDQP solver.

Task3.

Let

$$A=\begin{bmatrix}7&&&&&&\\&6&&&&&\\&&5&&&&\\&&&4&&&\\&&&&3&&\\&&&&&2&\\&&&&&&1\end{bmatrix},b=\begin{bmatrix}1\\3\\5\\7\\9\\11\\13\end{bmatrix},c=\begin{bmatrix}1\\0\\0\\0\\0\\0\\0\end{bmatrix},f=\begin{bmatrix}1\\2\\3\\4\\5\\6\\7\end{bmatrix}$$

and $d=1$, then the original problem can be rewritten in a canonical form

$$\min_x f^T x, s.t. \|Ax+b\|_2 \le c^T x + d$$

where $x=[a,b,c,d,e,f,g]^T$ is the variable to be optimized.

Let

$$A_i=\begin{bmatrix}c^T\\A\end{bmatrix},b_i=\begin{bmatrix}d\\b\end{bmatrix},u=\begin{bmatrix}u_0\\u_1\end{bmatrix}$$

where $u_0=c^T x+d, u_1=Ax+b$, we have $u=A_i x+b_i$. We next define a second-order cone

$$K=\left\{(u_0,u_1)^T \in R\times R^n \mid \|u_1\|_2 \le u_0\right\}$$

Another form is obtained for the original problem

$$\min_x f^T x, s.t. A_i x+b_i \in K$$

Its augmented Lagrangian is defined as

$$L_\rho(x,\mu)=f^T x+\frac{\rho}{2}\left(\left\|P_K\left(\frac{\mu}{\rho}-A_i x-b_i\right)\right\|^2\right)$$

where $P_K(\cdot)$ is a projection mapping onto the cone K. Therefore, the cone ALM algorithm reads

$$\begin{cases}x\leftarrow \arg\min L_\rho(x,\mu)\\ \mu\leftarrow P_K\left(\mu-\rho(A_i x+b_i)\right)\\ \rho\leftarrow \min[(1+\gamma)\rho,\beta]\end{cases}$$

The slides for chapter 4 give more details on how to implement this algorithm. We recommend you review our PPT.