2 Quadratic Program (QP)

Consider the convex quadratic program (QP) in the form

$$\min_{x} \quad \phi = \frac{1}{2}x'Hx + g'x \tag{2a}$$
$$s.t. \quad l \leq x \leq u, \tag{2b}$$
$$d_l \leq C'x \leq d_u. \tag{2c}$$

1. What is the Lagrangian function of this problem (2)?

2. Write the necessary and sufficient optimality conditions for (2).

3. An important application of constrained convex quadratic programming is linear optimal control. We will consider this as a test problem. The optimal control problem of a linear state-space model in discrete time may be formulated as

$$\min_{\{u_k\}_{k=1}^{N-1}} \quad \phi = \frac{1}{2}\sum_{k=1}^{N} ||z_k - r_k||_Q^2 + \frac{1}{2}\sum_{k=1}^{N-1} ||\Delta u_k||_R^2 \tag{3a}$$
$$s.t. \quad x_0 = \hat{x}_0 \tag{3b}$$
$$x_{k+1} = Ax_k + Bu_k \qquad k = 0, 1, \ldots, N-1 \tag{3c}$$
$$z_k = C_z x_k \qquad k = 0, 1, \ldots, N \tag{3d}$$
$$u_{\min} \leq u_k \leq u_{\max} \qquad k = 1, \ldots, N \tag{3e}$$
$$\Delta u_{\min} \leq \Delta u_k \leq \Delta u_{\max} \qquad k = 1, \ldots, N-1 \tag{3f}$$

The constraints (3c) introduce the system dynamics as a linear model, and constraints (3d) are the outputs of the system. x and u are the states and the inputs, respectively. k indexes the discrete time. The first term in the objective function penalises the error in the set-point tracking of the outputs. The second term penalises the rate of movement of the inputs, that is $\Delta u_k = u_{k+1} - u_k$. The solution of the problem will give the optimal trajectory of the system, by computing the optimal inputs $\{u_k\}_{k=1}^{N-1}$ that minimize the objective function.

The optimal control problem (3) represents the regulation problem in a model predictive control routine. If you want to read more about optimal control and model predictive control, we refer to [1, 2] and the course 02619 Model Predictive Control at DTU Compute.

The problem (3) is a convex QP, and it can be expressed as a constrained QP in the form (2). For this exercise, we consider the 4-tank system [3]. You can find the translated problem to the matrices H, g, C, d_l, d_u, l, u in the Matlab file `QP_Test.mat`. The optimization variable is the vector of stacked inputs

$$U = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} \tag{4}$$

with $u_i \in \mathbb{R}^2 \quad \forall i \in \{1, 2, \dots, N\}$. You need to download the file and read it in Matlab. You may use `loadmat` from `scipy.io` to read a `.mat` file in Python. Similarly, you may use `matread` from the package `MAT` in Julia.

4. Solve the problem (3) using Matlab's `quadprog`. Read `quadprog`'s documentation to make sure you pass the input correctly. You are welcome to (also) solve the problem with other optimization software, e.g. `IPOPT` with `Casadi`, `JuMP` with `Gurobi` in Julia, etc. Document this by providing code in the report. Plot the solution using the provided function `PlotSolutionQP`. Report solution statistics (number of iterations, CPU time, etc).

5. Write pseudo-code for a primal active-set algorithm for solution of the problem (2). Explain the major steps in your algorithm. Explain how you find a feasible initial point.

6. Implement the primal active-set algorithm for (2) and test it. You must provide commented code as well as driver files to test your code, documentation that it works, and performance statistics. Test your software on the optimal control problem (3), and compare the solution appropriately (primal and dual variables) with the one obtained via library software in task 4.

7. Write pseudo-code for a primal-dual interior-point algorithm for solution of the problem (2). Explain the major steps in your algorithm.

8. Implement the primal-dual interior-point algorithm for (2) and test it. You must provide commented code as well as driver files to test your code, documentation that it works, and performance statistics. Test your software on the optimal control problem (3), and compare the solution (primal and dual variables) with the one obtained via library software in subtask 4. Plot the KKT residuals as function of the iterations.

9. (**EXTRA**) Test your algorithms on other relevant problem(s) of your choice. Present the problem(s) and the solution.