TTK4135 Optimization and Control
Solution to Final Exam — Spring 2015

Department of Engineering Cybernetics

1 Various topics (32 %)

Problem classification

a (8 %) 1.

$$\begin{aligned} \min\ & 3x_1 + x_2 + x_1^2 \\ \text{s.t.}\ & 4x_1 - x_2 \leq 5 \\ & 3x_1 + x_2 \geq 0 \end{aligned}$$

This problem has a quadratic objective function and linear constraint functions, making it a quadratic programming (QP) problem. An active set method is a suitable approach to solve QP problems. The QP problem has positive semi-definite Hessian, therefore it is convex problem.

2.

$$\begin{aligned} \max\ & 15x_1 + 4x_2 \\ \text{s.t.}\ & -4x_1 - 16x_2 \leq 25 \\ & \frac{5}{4}x_1 + \frac{1}{3}x_2 = 1 \end{aligned}$$

This is a linear programming (LP) problem since all the functions are linear. A suitable algorithm for LP problems is the simplex algorithm. LP problems are convex, since both objective function and constraints are linear and convex.

3.

$$\min\ \ 100\left(x_2 - x_1^2\right)^2 + (1 - x_1)^2$$

This is an unconstraint optimization. A suitable algorithm for unconstraint problems where the Hessian is available is the Newton method. The problem is not convex.

4.

$$\begin{aligned} \min\ & (x_1 - x_2)^2 + 2x_1 - x_2 \\ \text{s.t.}\ & 4x_1^2 + x_2^2 = 16 \\ & x_1 + \frac{4}{5}x_2 \geq 1 \end{aligned}$$

This is a nonlinear programming (NLP) problem since the equality constraint function is nonlinear. A suitable algorithm for NLP problems is sequential quadratic programming (SQP). The problem is not convex because the equality constraint is not linear.

Active set method

b (3 %) Active set methods maintain estimates of the active and inactive index sets that are updated at each step of the algorithm. Two examples of active-set methods are the simplex method and active-set method.

c (3 %) The starting point needs to be feasible. In order to obtain the initial point, Phase I problem should be solved, which is as hard as the original problem.

Linear program

d (5 %) Formulate the linear program (LP)

$$
\begin{aligned}
\max \ & 2x_1 - x_2 + 5x_3 \\
\text{s.t. } & x_1 + 2x_2 \geq 5 \\
& 3x_1 + 7x_2 - 2x_3 \leq 25 \\
& 5x_1 + 6x_3 = 40 \\
& x_1, x_2, x_3 \geq 0
\end{aligned}
$$

on standard form (A.6) given in the appendix. Specify A, b and c explicitly.

We first convert the maximization problem into minimization problem by simply negating the objective function. We also convert the inequality constraints to equality constraints by adding or subtracting slack variables to make up the difference between the left- and right- hand sides. Hence, the above problem can be written as

$$
\begin{aligned}
\min \ & -2x_1 + x_2 - 5x_3 \\
\text{s.t. } & x_1 + 2x_2 - x_4 = 5 \\
& 3x_1 + 7x_2 - 2x_3 + x_5 = 25 \\
& 5x_1 + 6x_3 = 40 \\
& x_1, x_2, x_3, x_4, x_5 \geq 0
\end{aligned}
$$

The above problem can also be formulated on standard form (A.6) where

$$
\begin{aligned}
c^T &= \begin{bmatrix} -2 & 1 & -5 & 0 & 0 \end{bmatrix}, \\
A &= \begin{bmatrix} 1 & 2 & 0 & -1 & 0 \\ 3 & 7 & -2 & 0 & 1 \\ 5 & 0 & 6 & 0 & 0 \end{bmatrix}, \\
b &= \begin{bmatrix} 5 \\ 25 \\ 40 \end{bmatrix}.
\end{aligned}
$$

KKT conditions

e (7 %) State the KKT conditions (first-order necessary conditions) for the following quadratic program (QP):

$$
\begin{aligned}
\min_{x,y} \ & \frac{1}{2}(x-y)^T G (x-y) + x^T c + d^T y \\
\text{s.t. } & Ax = b \\
& Ey = h
\end{aligned}
$$

where $G = G^T$ is a symmetric matrix, and where x and y are vectors.

Here, the optimization variables are $\begin{bmatrix} x \\ y \end{bmatrix}$. The Lagrangian function can be written as

$$\mathcal{L}(x, y, \lambda_x, \lambda_y) = \frac{1}{2}(x-y)^T G(x-y) + x^T c + d^T y - \lambda_x^T (Ax - b) - \lambda_y^T (Ey - h),$$

where λ_x and λ_y are Lagrangian multipliers of x and y, respectively. The gradient of the Lagrangian function can be given by

$$\begin{aligned} \nabla_{x,y}\mathcal{L}(x, y, \lambda_x, \lambda_y) &= \begin{bmatrix} \nabla_x \mathcal{L}(x, y, \lambda_x, \lambda_y) \\ \nabla_y \mathcal{L}(x, y, \lambda_x, \lambda_y) \end{bmatrix} \\ &= \begin{bmatrix} Gx - Gy + c - A^T \lambda_x \\ Gy - Gx + d - E^T \lambda_y \end{bmatrix}. \end{aligned}$$

Therefore, the KKT conditions can be written as

$$\begin{aligned} G(x-y) + c - A^T \lambda_x &= 0, \\ -G(x-y) + d - E^T \lambda_y &= 0, \\ Ax &= b, \\ Ey &= h. \end{aligned}$$

Sensitivity

f (6 %) Consider the problem

$$\begin{aligned} \min \quad & -x_1 - x_2 \\ \text{s.t.} \quad & x_1^2 + x_2^2 \leq 2 \\ & x_2 - x_1^2 \geq 0 \end{aligned} \tag{1}$$

with solution $x^* = (1, 1)^T$. Suppose that you could perturb the right-hand side of one of the constraints in (1) appropriately in order to decrease the objective value. Which constraint would you choose? Justify you answer. (Hint: Draw the contours and feasible region of (1).)

We need first to rewrite problem (1) in the standard form:

$$\begin{aligned} \min \quad & f = -x_1 - x_2 \\ \text{s.t.} \quad & c_1 : 2 - x_1^2 - x_2^2 \geq 0 \\ & c_2 : x_2 - x_1^2 \geq 0 \end{aligned} \tag{2}$$

The gradients of the active constraints and the objective function at the solution x^* are as follow

$$\nabla f(x^*) = \begin{bmatrix} -1 \\ -1 \end{bmatrix},$$

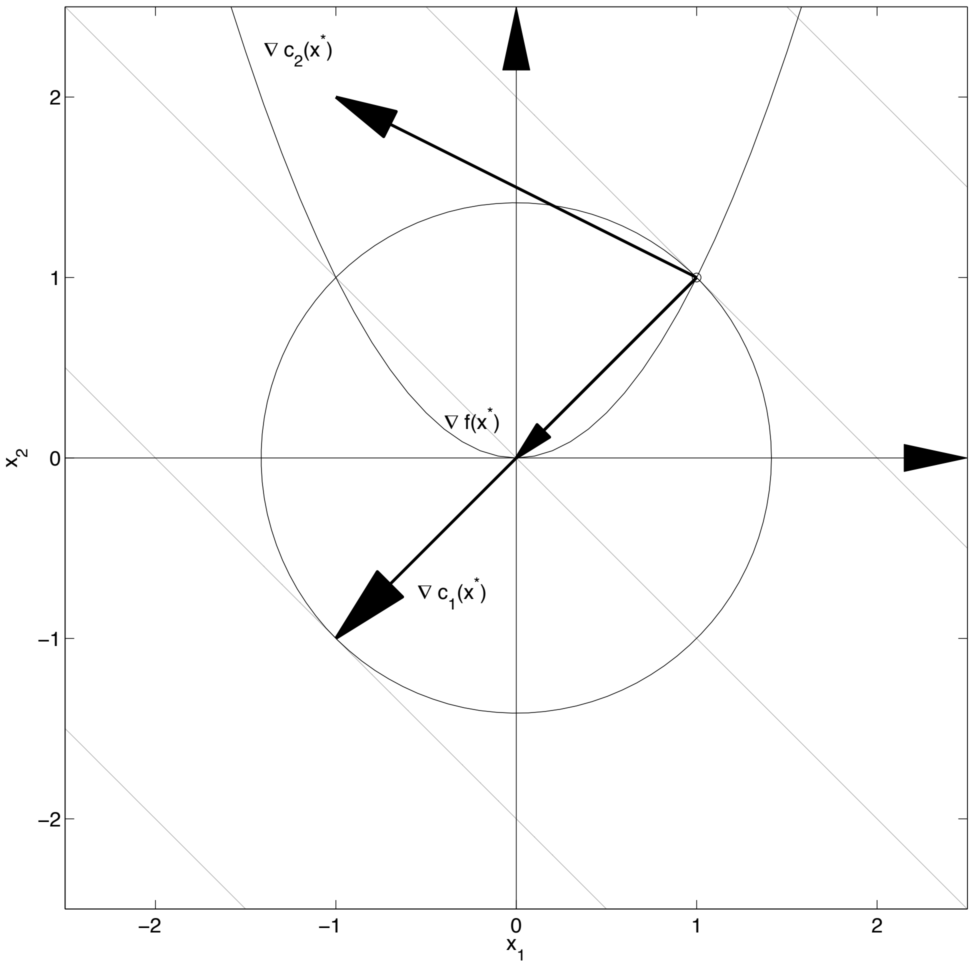
$$\nabla c_1(x^*) = \begin{bmatrix} -2 \\ -2 \end{bmatrix},$$

$$\nabla c_2(x^*) = \begin{bmatrix} -2 \\ 1 \end{bmatrix},$$

The gradients of the active constraints and the objective function at the solution are illustrated in Figure 1. Moreover, from the first KKT condition, we have

$$\nabla f = \lambda_1 \nabla c_1 + \lambda_2 \nabla c_2. \tag{3}$$

Regarding Figure 1 and (3), we can conclude that $\lambda_2 = 0$. If λ_i is exactly zero for some active constraint, small perturbation to c_i in some directions will hardly affect the optimal objective value all; the change is zero, to first order (Page 342 of the book). Therefore, in order to decrease the objective function we choose c_1 which has corresponding nonzero Lagrangian multiplier.

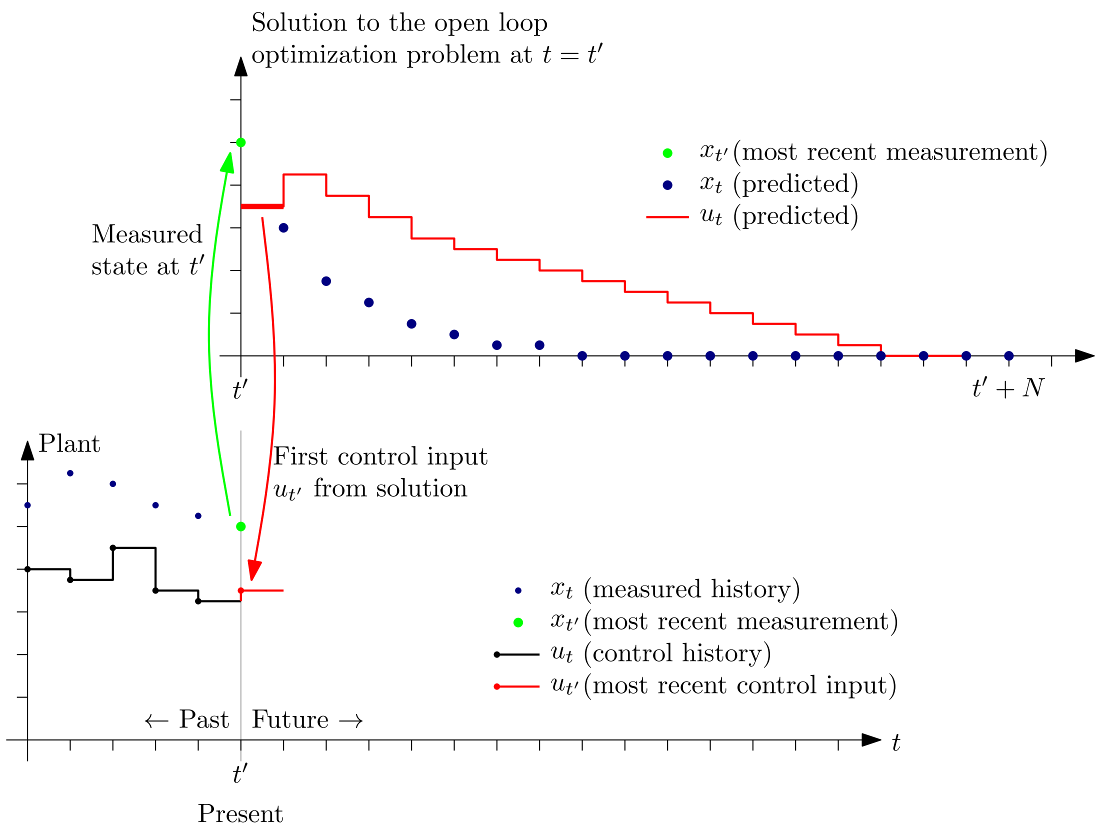


Figur 1: Gradients at the optimal point in Problem (1).

2 MPC and optimal control (32 %)

a (6 %) Explain the principle of model predictive control (MPC). Please include a figure to support your answer.

Model predictive control is a form of control in which the current control action is obtained by solving, at each sampling instant, a finite horizon open-loop optimal control problem, using the current state of the plant as the initial state; the optimization yields an optimal control sequence and the first control in this sequence is applied to the plant.Figure. 2 illustrates the principle of MPC.



Figur 2: MPC principle.

b (6 %) When using the dynamic optimization problem (A.9) in a linear MPC controller, which considerations must be made in the choice of the prediction horizon N?

As a general rule, the prediction horizon must be at least as long as the dominant dynamics (Preferably, 2-3 times larger than the dominant dynamics). The computational time of solving the QP problem must be lower than the control interval. If it is not the case, reducing the prediction horizon results in fewer variables to compute in the QP solved at each control interval, which promotes faster computations.

c (9 %) A common problem when solving problem (A.9) in MPC applications is that it may give an infeasible solution. Explain which of the constraints in (A.9) this concerns, why this often is an issue in MPC, and suggest an approach to address the problem. Present a reformulation of problem (A.9) in this latter case.

We normally have tighter bounds on the state variables than reality since some of the constraints are hard and must always be satisfied. Therefore, the state constraints (A.9d) may be violated for all the time (e.g., if a severe disturbance occurs between time step $t-1$ and t and moves the state beyond the state constraint limit, then no feasible point exists at t). In this case a feasible point may not exist and a control input may not be available. This is an unacceptable situation. To avoid this, we soften the state constraint (A.9d) by using slack variables as below:

$$\begin{aligned}\min_{z\in\mathbb{R}^n} f(z) = \sum_{t=0}^{N-1} \frac{1}{2}x_{t+1}^\top Q_{t+1}x_{t+1} + d_{xt+1}x_{t+1} \\ + \frac{1}{2}u_t^\top R_t u_t + d_{ut}u_t + \rho^T\epsilon + \frac{1}{2}\epsilon^T S\epsilon\end{aligned} \tag{4a}$$

subject to

$$x_{t+1} = A_t x_t + B_t u_t, \qquad t = 0,\dots,N-1 \tag{4b}$$

$$x_0 = \text{given} \tag{4c}$$

$$x^{\text{low}} - \epsilon \le x_t \le x^{\text{high}} + \epsilon, \qquad t = 1,\dots,N \tag{4d}$$

$$u^{\text{low}} \le u_t \le u^{\text{high}}, \qquad t = 0,\dots,N-1 \tag{4e}$$

$$-\Delta u^{\text{high}} \le \Delta u_t \le \Delta u^{\text{high}}, \qquad t = 0,\dots,N-1 \tag{4f}$$

$$Q_t \succeq 0 \qquad t = 1,\dots,N \tag{4g}$$

$$R_t \succeq 0 \qquad t = 0,\dots,N-1 \tag{4h}$$

where

$$\epsilon \in \mathbb{R}^{n_x} \ge 0 \tag{4i}$$

$$\rho \in \mathbb{R}^{n_x} \ge 0 \tag{4j}$$

$$S = \text{diag}\{s_1,\cdots,s_{n_x}\},\ s_i \ge 0,\ i = \{1,2,\cdots,n_x\}. \tag{4k}$$

Two positive terms $\rho^T\epsilon$ and/or $\frac{1}{2}\epsilon^T S\epsilon$ are added to the original QP problem. These are both positive terms, hence there is a desire to derive these terms to zero. More precisely, the slack variables should be nonzero if the corresponding constraints are violated. Adding $\rho^T\epsilon$ is like adding a penalty function and if ρ is chosen big enough, then the solution is exact.

d (6 %) Assume that there are no bounds (inequality constraints) on the state x_t or input u_t, that our system is described by a linear time-invariant (LTI) model, that all states are measured, and that we choose $Q_t := Q \succeq 0$, and $R_t := R \succ 0$ as constant matrices. Furthermore, we consider and infinite horizon ($N = \infty$), in which our optimal control problem reduces to

$$
\begin{aligned}
\min \quad & \sum_{t=0}^{\infty} \frac{1}{2} x_{t+1}^T Q x_{t+1} + \frac{1}{2} u_t^T R u_t \\
\text{s.t.} \quad & x_{t+1} = A x_t + B u_t \\
& x_0 = \text{given}
\end{aligned}
\tag{5}
$$

What type of controller does this optimization problem translate to? Use the fact that $P_t = P_{t+1} = P$ when considering an infinite horizon N, and state the equations for computing the resulting controller. What is the equation for computing P called, and why must we include the additional requirement $P = P^T \succeq 0$?

The controller for this optimization problem is called the Linear Quadratic Regulator (LQR). The solution of (5) is given by

$$u_t = -K x_t, \quad \text{for} \quad 0 \le t \le \infty,$$

where the feedback gain matrix is derived by

$$
\begin{aligned}
K &= R^{-1} B^T P (I + B R^{-1} B^T P)^{-1} A, \\
P &= Q + A^T P (I + B R^{-1} B^T P), \\
P &= P^T \succeq 0.
\end{aligned}
$$

The equation for computing P is called the algebraic Riccati equation, which is a quadratic equation in the unknown matrix P. Thus, there may exist several (particularly two) solutions to this equation. Only one solution, however, will be positive semi-definite and therefore we specify $P = P^T \succeq 0$.

e (6 %) Suppose that some, but not all of the states x_t can be measured, and that a stationary Kalman filter therefore is applied to estimate the states. When using this estimated state $\hat{x}_t$ together with the controller derived from the optimization problem (5) in question **2d**, what is the final control structure called? State the requirements for stability of the closed-loop system.

The control structure is called the Linear Quadratic Gaussian Control (LQG). The closed-loop system given by the optimal solution is asymptotically stable, if (A, B) is stabilizable and (A, D) is detectable, where D is defined by $Q = D^T D$.

3 Nonlinear programming and SQP (36 %)

Consider the following nonlinear program (NLP)

$$\begin{aligned} \min \ & x_2 \\ \text{s.t. } & x_1^3 - 2x_1^2 + x_2 \geq 0 \\ & (1-x_1)^3 - 2(1-x_1)^2 + x_2 \geq 0 \end{aligned} \tag{6}$$

with the global solution $x^* = \left(\frac{1}{2}, \frac{3}{8}\right)$.

a (8 %) State the KKT conditions for (6).

In order to write the KKT condition, we need to define the Lagrangian function:

$$\mathcal{L}(x_1, x_2, \lambda_1, \lambda_2) = x_2 - \lambda_1(x_1^3 - 2x_1^2 + x_2) - \lambda_2((1-x_1)^3 - 2(1-x_1)^2 + x_2)$$

The gradient of the Lagrangian function can be given by

$$\nabla_x \mathcal{L}(x, \lambda) = \begin{bmatrix} -3\lambda_1 x_1^2 + 4\lambda_1 x_1 + 3\lambda_2(1-x_1)^2 - 4\lambda_2(1-x_1) \\ 1 - \lambda_1 - \lambda_2 \end{bmatrix}$$

The KKT conditions can be written as follow

$$\begin{aligned} \nabla_x \mathcal{L}(x^*, \lambda^*) &= 0, \\ (x_1^*)^3 - 2(x_1^*)^2 + x_2^* &\geq 0, \\ (1-x_1^*)^3 - 2(1-x_1^*)^2 + x_2^* &\geq 0, \\ \lambda_1^*, \lambda_2^* &\geq 0, \\ \lambda_1^*\big((x_1^*)^3 - 2(x_1^*)^2 + x_2^*\big) &= 0, \\ \lambda_2^*\big((1-x_1^*)^3 - 2(1-x_1^*)^2 + x_2^*\big) &= 0. \end{aligned}$$

b (7 %) Compute the Hessian $\nabla_{xx}^2 \mathcal{L}(x, \lambda)$. Which problems may this exact Hessian cause if used in the SQP algorithm 18.3 given in the appendix? Suggest a method to circumvent the problem.

$$\nabla_{xx}^2 \mathcal{L} = \begin{bmatrix} 6x_1(-\lambda_1 + \lambda_2) + 4\lambda_1 - 2\lambda_2 & 0 \\ 0 & 0 \end{bmatrix}$$

The Hessian has one zero eigenvalue. Therefore, it is positive semi-definite. As a result, the KKT matrix in calculation of p_k in the SQP algorithm 18.3 is singular. There are some ways to address this issue: (1) Using quasi-Newton method, (2) Adding a multiple of identity matrix, (3) Modified Cholesky factorization

c (8 %) In each iteration of the line-search SQP algorithm, we compute the search direction by the quadratic subproblem

$$\begin{aligned} \min \ & f_k + \nabla f_k^T p + \frac{1}{2} p^T \nabla_{xx}^2 \mathcal{L}_k p \\ \text{s.t. } & \nabla c_i(x_k)^T p + c_i(x_k) = 0 \qquad i \in \mathcal{E} \\ & \nabla c_i(x_k)^T p + c_i(x_k) \geq 0 \qquad i \in \mathcal{I} \end{aligned} \tag{7}$$

Assume that we use a quasi-Newton approximation, in which we replace $\nabla^2_{xx}\mathcal{L}_0$ with $B_0 = I$. Let $x_0 = (\frac{1}{2}, 0)$ be the chosen starting point. Show that the QP subproblem (7) in this case is given by

$$\min \; p_2 + \frac{1}{2}(p_1^2 + p_2^2) \tag{8a}$$

$$\text{s.t.} \; -\frac{5}{4}p_1 + p_2 - \frac{3}{8} \geq 0 \tag{8b}$$

$$\frac{5}{4}p_1 + p_2 - \frac{3}{8} \geq 0 \tag{8c}$$

The objective function can be written as

$$f_0 + \nabla f_0^T p + \frac{1}{2}p^T B_0 p = 0 + p_2 + \frac{1}{2}(p_1^2 + p_2^2).$$

The gradient of the constraints can be written as

$$\begin{aligned}\nabla c_1 &= \begin{bmatrix} 3x_1^2 - 4x_1 & 1 \end{bmatrix},\\ \nabla c_2 &= \begin{bmatrix} -3(1-x_1)^2 + 4(1-x_1) & 1 \end{bmatrix}.\end{aligned}$$

Therefore, the constraints at x_0 can be written as

$$\begin{aligned}c_1(x_0)^T p + c_1(x_0) &= -\frac{5}{4}p_1 + p_2 - \frac{3}{8} &\geq 0,\\ c_2(x_0)^T p + c_2(x_0) &= \frac{5}{4}p_1 + p_2 - \frac{3}{8} &\geq 0.\end{aligned}$$

d (4 %) Draw the feasible region for $p_1 - p_2$ in (8), and the contours of the objective function (8a). By inspecting your plot, what is the solution to (8)? (Do not try to solve the problem by an iterative method.)

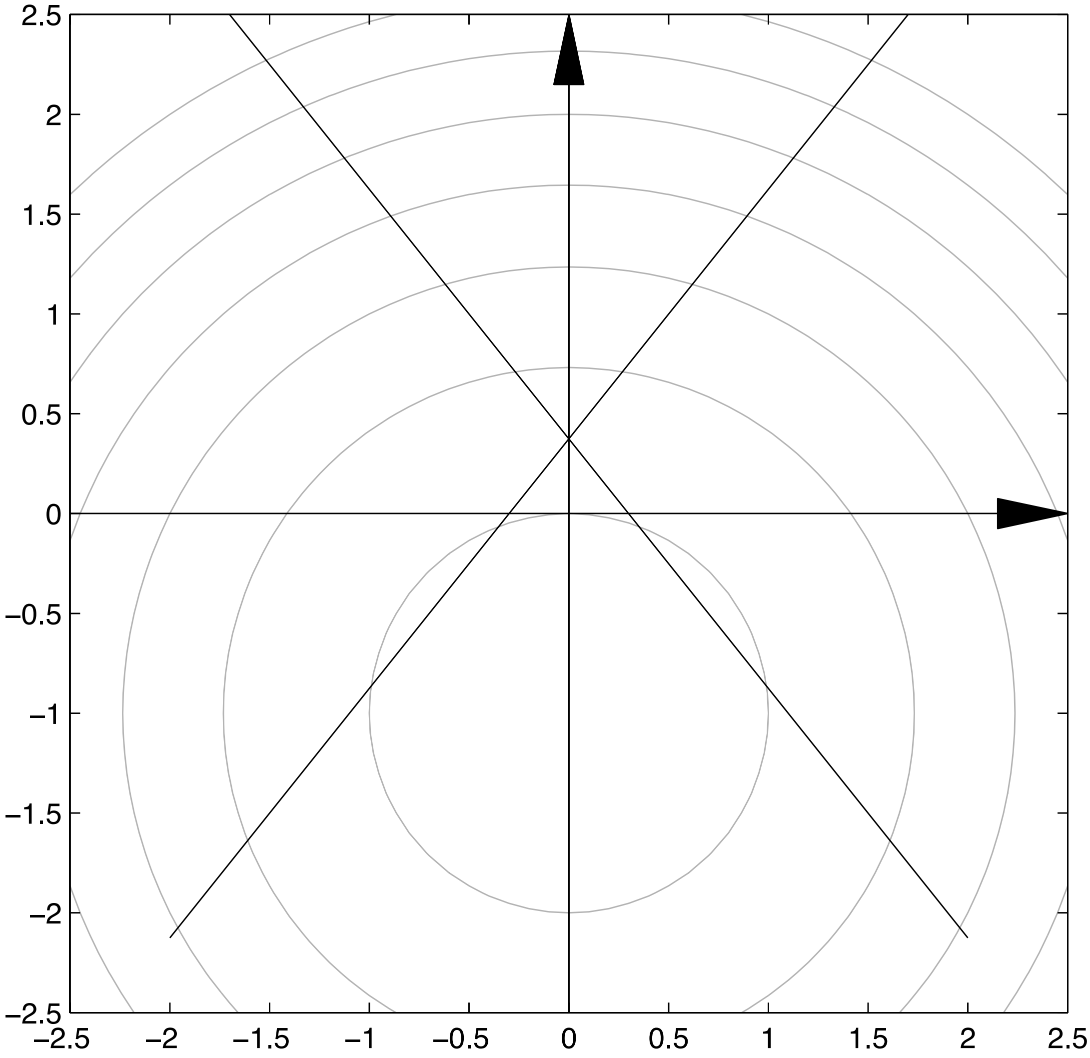
From Figure. 3, we can inspect that the solution of (8) is equal to $p = \begin{bmatrix} 0 \\ \frac{3}{8} \end{bmatrix}$.

e (6 %) Why do we use a merit-function in Algorithm 18.3? Suggest and formulate explicitly a suitable merit-function for the NLP (6).

SQP methods often use a merit function to decide wether a trail step should be accepted. In line search methods, the merit function controls the size of the step. A popular choice of merit function is the l_1 penalty function, and can be written for problem (6) as

$$\phi_1(x;\mu) = x_2 + \mu\big(\max\{0, -(x_1^3 - 2x_1^2 + x_2)\} + \max\{0, -((1-x_1)^3 - 2(1-x_1)^2 + x_2)\}\big)$$

The positive scalar μ is the penalty parameter, which determines the weight that we assigned to constraint satisfaction relative to minimization of the objective.



Figur 3: Feasible region and the contours of the objective function of problem (8).

f (3 %) Let $\alpha_0 = 1$, and show that Algorithm 18.3 converges to x^* in one iteration with the provided starting point $x_0 = (\frac{1}{2}, 0)$ and the direction p computed from the QP (8).

x_1 can be computed by

$$\begin{aligned} x_1 &= x_0 + \alpha_0 * p_0 \\ &= (\frac{1}{2}, 0) + (0, \frac{3}{8}) \\ &= (\frac{1}{2}, \frac{3}{8}). \end{aligned}$$

Appendix

Part 1 Optimization Problems and Optimality Conditions

A general formulation for constrained optimization problems is

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x) && && \text{(A.1a)} \\ \text{s.t.} \quad & c_i(x) = 0, && i \in \mathcal{E} && \text{(A.1b)} \\ & c_i(x) \geq 0, && i \in \mathcal{I} && \text{(A.1c)} \end{aligned}$$

where f and the functions c_i are all smooth, differentiable, real-valued functions on a subset of $\mathbb{R}^n$, and $\mathcal{E}$ and $\mathcal{I}$ are two finite sets of indices.

The Lagrangean function for the general problem (A.1) is

$$\mathcal{L}(x, \lambda) = f(x) - \sum_{i\in\mathcal{E}\cup\mathcal{I}} \lambda_i c_i(x) \tag{A.2}$$

The KKT-conditions for (A.1) are given by:

$$\begin{aligned} \nabla_x \mathcal{L}(x^*, \lambda^*) &= 0 && && \text{(A.3a)} \\ c_i(x^*) &= 0, && i \in \mathcal{E} && \text{(A.3b)} \\ c_i(x^*) &\geq 0, && i \in \mathcal{I} && \text{(A.3c)} \\ \lambda_i^* &\geq 0, && i \in \mathcal{I} && \text{(A.3d)} \\ \lambda_i^* c_i(x^*) &= 0, && i \in \mathcal{E} \cup \mathcal{I} && \text{(A.3e)} \end{aligned}$$

2nd order (sufficient) conditions for (A.1) are given by:

$$w \in \mathcal{C}(x^*, \lambda^*) \Leftrightarrow \begin{cases} \nabla c_i(x^*)^\top w = 0 & \text{for all } i \in \mathcal{E} \\ \nabla c_i(x^*)^\top w = 0 & \text{for all } i \in \mathcal{A}(x^*) \cap \mathcal{I} \text{ with } \lambda_i^* > 0 \\ \nabla c_i(x^*)^\top w \geq 0 & \text{for all } i \in \mathcal{A}(x^*) \cap \mathcal{I} \text{ with } \lambda_i^* = 0 \end{cases} \tag{A.4}$$

Theorem 1: (Second-Order Sufficient Conditions) *Suppose that for some feasible point $x^* \in \mathbb{R}^n$ there is a Lagrange multiplier vector λ^* such that the KKT conditions* (A.3) *are satisfied. Suppose also that*

$$w^\top \nabla^2_{xx} \mathcal{L}(x^*, \lambda^*) w > 0, \qquad \text{for all } w \in \mathcal{C}(x^*, \lambda^*),\ w \neq 0. \tag{A.5}$$

Then x^* is a strict local solution for (A.1).

LP problem in standard form:

$$\min_{x} \quad f(x) = c^\top x \tag{A.6a}$$
$$\text{s.t.} \quad Ax = b \tag{A.6b}$$
$$x \geq 0 \tag{A.6c}$$

where $A \in \mathbb{R}^{m \times n}$ and $\operatorname{rank} A = m$.

QP problem in standard form:

$$\min_{x} \quad f(x) = \frac{1}{2} x^\top G x + x^\top c \tag{A.7a}$$
$$\text{s.t.} \quad a_i^\top x = b_i, \qquad i \in \mathcal{E} \tag{A.7b}$$
$$a_i^\top x \geq b_i, \qquad i \in \mathcal{I} \tag{A.7c}$$

where G is a symmetric $n \times n$ matrix, $\mathcal{E}$ and $\mathcal{I}$ are finite sets of indices and c, x and $\{a_i\}, i \in \mathcal{E} \cup \mathcal{I}$, are vectors in $\mathbb{R}^n$. Alternatively, the equalities can be written $Ax = b$, $A \in \mathbb{R}^{m \times n}$.

Iterative method:

$$x_{k+1} = x_k + \alpha_k p_k \tag{A.8a}$$
$$x_0 \text{ given} \tag{A.8b}$$
$$x_k, p_k \in \mathbb{R}^n, \;\; \alpha_k \in \mathbb{R} \tag{A.8c}$$

p_k is the search direction and α_k is the line search parameter.

Part 2 Optimal Control

A typical open-loop optimal control problem on the time horizon 0 to N is

$$\min_{z\in\mathbb{R}^n} f(z) = \sum_{t=0}^{N-1} \frac{1}{2}x_{t+1}^\top Q_{t+1}x_{t+1} + d_{xt+1}x_{t+1} + \frac{1}{2}u_t^\top R_t u_t + d_{ut}u_t \tag{A.9a}$$

subject to

$$x_{t+1} = A_t x_t + B_t u_t, \qquad t = 0,\dots,N-1 \tag{A.9b}$$
$$x_0 = \text{given} \tag{A.9c}$$
$$x^{\text{low}} \le x_t \le x^{\text{high}}, \qquad t = 1,\dots,N \tag{A.9d}$$
$$u^{\text{low}} \le u_t \le u^{\text{high}}, \qquad t = 0,\dots,N-1 \tag{A.9e}$$
$$-\Delta u^{\text{high}} \le \Delta u_t \le \Delta u^{\text{high}}, \qquad t = 0,\dots,N-1 \tag{A.9f}$$
$$Q_t \succeq 0 \qquad t = 1,\dots,N \tag{A.9g}$$
$$R_t \succeq 0 \qquad t = 0,\dots,N-1 \tag{A.9h}$$

where

$$u_t \in \mathbb{R}^{n_u} \tag{A.9i}$$
$$x_t \in \mathbb{R}^{n_x} \tag{A.9j}$$
$$\Delta u_t = u_t - u_{t-1} \tag{A.9k}$$
$$z^\top = (x_1^\top,\dots,x_N^\top, u_0^\top,\dots,u_{N-1}^\top) \tag{A.9l}$$

The subscript t denotes discrete time sampling instants.

The optimization problem for linear quadratic control of discrete dynamic systems is given by

$$\min_{z\in\mathbb{R}^n} f(z) = \sum_{t=0}^{N-1} \frac{1}{2}x_{t+1}^\top Q_{t+1}x_{t+1} + \frac{1}{2}u_t^\top R_t u_t \tag{A.10a}$$

subject to

$$x_{t+1} = A_t x_t + B_t u_t \tag{A.10b}$$
$$x_0 = \text{given} \tag{A.10c}$$

where

$$u_t \in \mathbb{R}^{n_u} \tag{A.10d}$$
$$x_t \in \mathbb{R}^{n_x} \tag{A.10e}$$
$$z^\top = (x_1^\top,\dots,x_N^\top, u_0^\top \dots, u_{N-1}^\top) \tag{A.10f}$$

Theorem 2: *The solution of (A.10) with $Q_t \succeq 0$ and $R_t \succ 0$ is given by*

$$u_t = -K_t x_t \tag{A.11a}$$

where the feedback gain matrix is derived by

$$K_t = R_t^{-1} B_t^\top P_{t+1}(I + B_t R_t^{-1} B_t^\top P_{t+1})^{-1} A_t, \qquad t = 0, \dots, N-1 \tag{A.11b}$$

$$P_t = Q_t + A_t^\top P_{t+1}(I + B_t R_t^{-1} B_t^\top P_{t+1})^{-1} A_t, \qquad t = 0, \dots, N-1 \tag{A.11c}$$

$$P_N = Q_N \tag{A.11d}$$

Part 3 Sequential quadratic programming (SQP)

Algorithm 18.3 (Line Search SQP Algorithm).

Choose parameters $\eta \in (0, 0.5)$, $\tau \in (0, 1)$, and an initial pair (x_0, λ_0);
Evaluate $f_0, \nabla f_0, c_0, A_0$;
If a quasi-Newton approximation is used, choose an initial $n \times n$ symmetric positive definite Hessian approximation B_0, otherwise compute $\nabla^2_{xx}\mathcal{L}_0$;
repeat until a convergence test is satisfied
  Compute p_k by solving (18.11); let $\hat{\lambda}$ be the corresponding multiplier;
  Set $p_\lambda \leftarrow \hat{\lambda} - \lambda_k$;
  Choose μ_k to satisfy (18.36) with $\sigma = 1$;
  Set $\alpha_k \leftarrow 1$;
  while $\phi_1(x_k + \alpha_k p_k; \mu_k) > \phi_1(x_k; \mu_k) + \eta \alpha_k D_1(\phi(x_k; \mu_k) p_k)$
    Reset $\alpha_k \leftarrow \tau_\alpha \alpha_k$ for some $\tau_\alpha \in (0, \tau]$;
  end (while)
  Set $x_{k+1} \leftarrow x_k + \alpha_k p_k$ and $\lambda_{k+1} \leftarrow \lambda_k + \alpha_k p_\lambda$;
  Evaluate $f_{k+1}, \nabla f_{k+1}, c_{k+1}, A_{k+1}$, (and possibly $\nabla^2_{xx}\mathcal{L}_{k+1}$);
  If a quasi-Newton approximation is used, set
    $s_k \leftarrow \alpha_k p_k$ and $y_k \leftarrow \nabla_x \mathcal{L}(x_{k+1}, \lambda_{k+1}) - \nabla_x \mathcal{L}(x_k, \lambda_{k+1})$,
  and obtain B_{k+1} by updating B_k using a quasi-Newton formula;
end (repeat)