ESE 619 Model Predictive Control
 University of Pennsylvania, Spring 2023
 Student Name: Wei-Cheng (Rory) Huang

Homework 6: Explicit MPC
 updated April 10, 2023
 Penn ID: 55353592

The solutions of this homework are entirely my own. I have discussed these problems with several classmates, they are: Shiming Liang

1. **Using the Multi-parametric Toolbox**

   This exercise will give you an insight into parametric optimization, and explicit and hybrid model predictive control. You will familiarize yourself with two toolboxes for MATLAB, YALMIP and MPT. YALMIP is a toolbox, that simplifies the process of formulating and solving optimization problems of varying form, using various numerical solvers. This includes dealing with optimization problems that are parametric, which means they depend on parameters and we would like to know the solution as a function of those parameters. You will also familiarize yourself with MPT, a toolbox that helps you generate model predictive controllers, explicit MPC controllers and also allows you to visualize and run closed-loop simulations with little effort.

   Details of the installation process can be found in https://www.mpt3.org/Main/Installation.

   Familiarize yourself with the toolbox by invoking one of the demos. Try for example:

   `>>mpt_demo1`

   `>>mpt_demo2`

   `>>mpt_demo_lti1`

   `>>mpt_demo_lti4`

   Refer to MPT wiki: https://www.mpt3.org for more details.
   Done

2. **Explicit MPC**

In this problem we will first derive the parametric solution of the given optimization problem by hand. Then we will verify the solution using the MPT3 toolbox.

Consider the following problem:

$$\begin{aligned} J^*(x_1, x_2) = \min_{z \in \mathbb{R}} \ & \frac{1}{2}z^2 + 2x_1 z + x_2^2 \\ \text{subj. to} \quad & z \le 1 + x_1 \\ & -z \le 1 - x_2 \end{aligned} \tag{mp-QP}$$

where $x_1, x_2 \in \mathbb{R}$ are the parameters.

1. **[4 pt]** Write out the Lagrangian and the KKT conditions for this problem.
Solution:
The Lagrangian of this problem is:

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

The KKT conditions are:

$$\begin{aligned} &\text{Stationary:} && \nabla_z \mathcal{L}(z, \lambda_1, \lambda_2) = 0 \Rightarrow z + 2x_1 + \lambda_1 - \lambda_2 = 0 \\ &\text{Primal Feasibility:} && z - 1 - x_1 \le 0 \\ & && -z - 1 + x_2 \le 0 \\ &\text{Dual Feasibility:} && \lambda_1 \ge 0 \\ & && \lambda_2 \ge 0 \\ &\text{Complementary Slackness:} && \lambda_1 (z - 1 - x_1) = 0 \\ & && \lambda_2 (-z - 1 + x_2) = 0 \end{aligned}$$

2. **[4 pt]** Determine the region of the parameter space for which mpQP has a non-empty feasible set.
Solution:
The feasible set is not empty implies:

$$\mathcal{Z}: \begin{cases} z \le 1 + x_1 \\ -z \le 1 - x_2 \end{cases} \Rightarrow \begin{cases} z \le 1 + x_1 \\ z \ge -1 + x_2 \end{cases} \xRightarrow{\mathcal{Z} \neq \emptyset} -1 + x_2 \le 1 + x_1 \Rightarrow 0 \le x_1 - x_2 + 2$$

Therefore the region of the parameter space $\mathcal{K}^*$ is:

$$\mathcal{K}^* = \left\{ x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2, 0 \le x_1 - x_2 + 2 \right\}$$

3. **[4 pt]** From the primal and dual feasibility conditions, write out the complementary cases that can occur.
4. **[4 pt]** Solve for $z^*(x_1, x_2)$ and $J^*(x_1, x_2)$ for each case.
Solution:(3-4)
There are two constraints in this problem and each of them can be active or inactive. Therefore, there are in totoal 2^2 cases we can discuss. Let $g_1 \triangleq x - 1 - x_1$, $g_2 \triangleq -z - 1 + x2$. The caess are:
(a) g_1, g_2 both are active:

$$\begin{cases} \lambda_1 \ge 0 \\ \lambda_2 \ge 0 \\ z - 1 - x_1 = 0 \\ -z - 1 + x_2 = 0 \\ z + 2x_1 + \lambda_1 - \lambda_2 = 0 \end{cases} \Rightarrow \begin{cases} z^* = 1 + x_1 = -1 + x_2 \\ 1 + 3x_1 + \lambda_1^* - \lambda_2^* = 0 \end{cases}$$

And the optimal objective value is:

$$J^* = \frac{1}{2}z^2 + 2x_1 z + x_2^2 = \frac{1}{2}(1+x_1)^2 + 2x_1(1+x_1) + x_2^2 \quad \text{with} \quad x2 - x1 - 2 = 0$$

(b) g_1 is active, g_2 inactive:

$$\begin{cases} \lambda_1 \geq 0 \\ \lambda_2 = 0 \\ z - 1 - x_1 = 0 \\ -z - 1 + x_2 < 0 \\ z + 2x_1 + \lambda_1 - \lambda_2 = 0 \end{cases} \Rightarrow \begin{cases} z^* = 1 + x_1 \\ 1 + 3x_1 + \lambda_1^* = 0 \\ \lambda_2^* = 0 \end{cases}$$

And the optimal objective value is:

$$J^* = \frac{1}{2}z^2 + 2x_1 z + x_2^2 = \frac{1}{2}(1+x_1)^2 + 2x_1(1+x_1) + x_2^2$$

(c) g_1 is inactive, g_2 active:

$$\begin{cases} \lambda_1 = 0 \\ \lambda_2 \geq 0 \\ z - 1 - x_1 < 0 \\ -z - 1 + x_2 = 0 \\ z + 2x_1 + \lambda_1 - \lambda_2 = 0 \end{cases} \Rightarrow \begin{cases} z^* = -1 + x_2 \\ \lambda_1^* = 0 \\ -1 + x_2 + 2x_1 + \lambda_2^* = 0 \end{cases}$$

And the optimal objective value is:

$$J^* = \frac{1}{2}z^2 + 2x_1 z + x_2^2 = \frac{1}{2}(x_2-1)^2 + 2x_1(x_2-1) + x_2^2$$

(d) g_1 is inactive, g_2 inactive:

$$\begin{cases} \lambda_1 = 0 \\ \lambda_2 = 0 \\ z - 1 - x_1 < 0 \\ -z - 1 + x_2 < 0 \\ z + 2x_1 + \lambda_1 - \lambda_2 = 0 \end{cases} \Rightarrow \begin{cases} z^* = -2x_1 \\ \lambda_1^* = 0 \\ \lambda_2^* = 0 \end{cases}$$

And the optimal objective value is:

$$J^* = \frac{1}{2}z^2 + 2x_1 z + x_2^2 = -2x_1^2 + x_2^2$$

5. **[4 pt]** Draw the critical regions on the parameter space.
Solution:
The critical region is derived by ensuring primal and dual feasibility for different complementary cases (different optimizer and optimal objective). Also note that in the above discussion, the case where g_1, g_2 are both active will shirnk to a point and would not need to be discussed. Therefore, there are three critical regions, as is discussed below:

(a) $\mathcal{CR}_{\{1\}}$:

$$\begin{cases} z^* = 1 + x_1 \\ 1 + 3x_1 + \lambda_1^* = 0 \\ \lambda_2^* = 0 \end{cases} \Rightarrow \begin{array}{lll} \mathcal{P}_p : \left\{ \begin{array}{l} z^* \leq 1 + x_1 \\ -z^* \leq 1 - x_2 \end{array} \right. & \Rightarrow \left\{ \begin{array}{l} 1 + x_1 \leq 1 + x_1 \\ -1 - x_1 \leq 1 - x_2 \end{array} \right. & \Rightarrow \mathcal{K}^* \\ \mathcal{P}_d : \left\{ \begin{array}{l} \lambda_1^* \geq 0 \\ \lambda_2^* \geq 0 \end{array} \right. & \Rightarrow \left\{ \begin{array}{l} -1 - 3x_1 \geq 0 \\ 0 \geq 0 \end{array} \right. & \Rightarrow 3x_1 + 1 \leq 0 \end{array}$$

Therefore:

$$\mathcal{CR}_{\{1\}} = \mathcal{K}^* \cap \mathcal{P}_p \cap \mathcal{P}_d = \left\{ x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2, 0 \leq x_1 - x_2 + 2, 3x_1 + 1 \leq 0 \right\}$$

(b) $\mathcal{CR}_{\{2\}}$:

$$\begin{cases} z^* = -1 + x_2 \\ \lambda_1^* = 0 \\ -1 + x_2 + 2x_1 + \lambda_2^* = 0 \end{cases} \Rightarrow \begin{array}{lll} \mathcal{P}_p : \begin{cases} z^* \leq 1 + x_1 \\ -z^* \leq 1 - x_2 \end{cases} & \Rightarrow \begin{cases} -1 + x_2 \leq 1 + x_1 \\ 1 - x_2 \leq 1 - x_2 \end{cases} & \Rightarrow \mathcal{K}^* \\ \mathcal{P}_d : \begin{cases} \lambda_1^* \geq 0 \\ \lambda_2^* \geq 0 \end{cases} & \Rightarrow \begin{cases} 0 \geq 0 \\ 1 - x_2 - 2x_1 \geq 0 \end{cases} & \Rightarrow 1 - x_2 - 2x_1 \leq 0 \end{array}$$

Therefore:

$$\mathcal{CR}_{\{2\}} = \mathcal{K}^* \cap \mathcal{P}_p \cap \mathcal{P}_d = \left\{ x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2, 0 \leq x_1 - x_2 + 2, 1 - x_2 - 2x_1 \leq 0 \right\}$$

(c) $\mathcal{CR}_{\{\}}$:

$$\begin{cases} z^* = -2x_1 \\ \lambda_1^* = 0 \\ \lambda_2^* = 0 \end{cases} \Rightarrow \begin{array}{lll} \mathcal{P}_p : \begin{cases} z^* \leq 1 + x_1 \\ -z^* \leq 1 - x_2 \end{cases} & \Rightarrow \begin{cases} -2x_1 \leq 1 + x_1 \\ 2x_1 \leq 1 - x_2 \end{cases} & \Rightarrow \begin{cases} 0 \leq 1 + 3x_1 \\ 0 \leq 1 - x_2 - 2x_1 \end{cases} \\ \mathcal{P}_d : \begin{cases} \lambda_1^* \geq 0 \\ \lambda_2^* \geq 0 \end{cases} & \Rightarrow \begin{cases} 0 \geq 0 \\ 0 \geq 0 \end{cases} & \Rightarrow \mathcal{K}^* \end{array}$$

Therefore:

$$\mathcal{CR}_{\{\}} = \mathcal{K}^* \cap \mathcal{P}_p \cap \mathcal{P}_d = \left\{ x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2, 0 \leq x_1 - x_2 + 2, 3x_1 + 1 \leq 0, 0 \leq 1 - x_2 - 2x_1 \right\}$$

The drawing of critical regions is shown below:

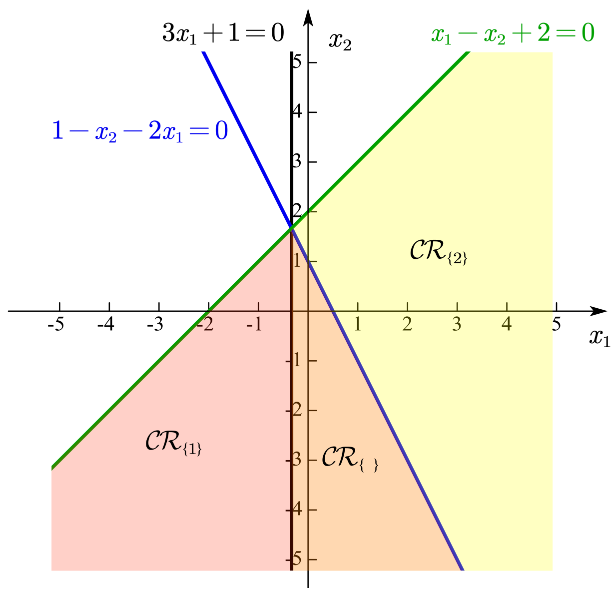

Figure 1: Drawing of critical regions

Now we will use YALMIP to solve mpQP and compare the solution. YALMIP was installed as part of Exercise 1, to get more info type help yalmip in the MATLAB command window and visit the website. To solve mpQP with YALMIP, follow the following steps.

6. [**0 pt**] First, declare the decision variable and parameter variables as symbolic variables:
```
z = sdpvar(1, 1);
x = sdpvar(2, 1);
```
7. [**4 pt**] Use the symbolic variables to define the objective and constraints of mpQP:
```
J = 0.5 * z(1)^2 + ...;
C = [ z(1) <= 1 + x(1), ...];
```
Also include `-5 <= x <= 5` in `C` to restrict the plots to this region.

8. **[0 pt]** Now convert the problem into the format of the MPT3 toolbox and call the routine to solve the multi-parametric program. Type `help Opt` and `help Opt.solve` to get more information on these two commands.

```
mpQP = Opt(C, J, x, z);
solution = mpQP.solve();
```

9. **[0 pt]** To visualize the solution and compare with the analytical results derived above, plot the critical regions, the $z^*(x_1, x_2)$, and $J^*(x_1, x_2)$. Type `help PolyUnion.fplot` and `help PolyUnion.plot` to get more information on these two plotting commands.

```
figure; solution.xopt.plot();
figure; solution.xopt.fplot('primal');
figure; solution.xopt.fplot('obj');
```

Solution:

The coding detials are shown in the live scripts, the required figures are shown below:

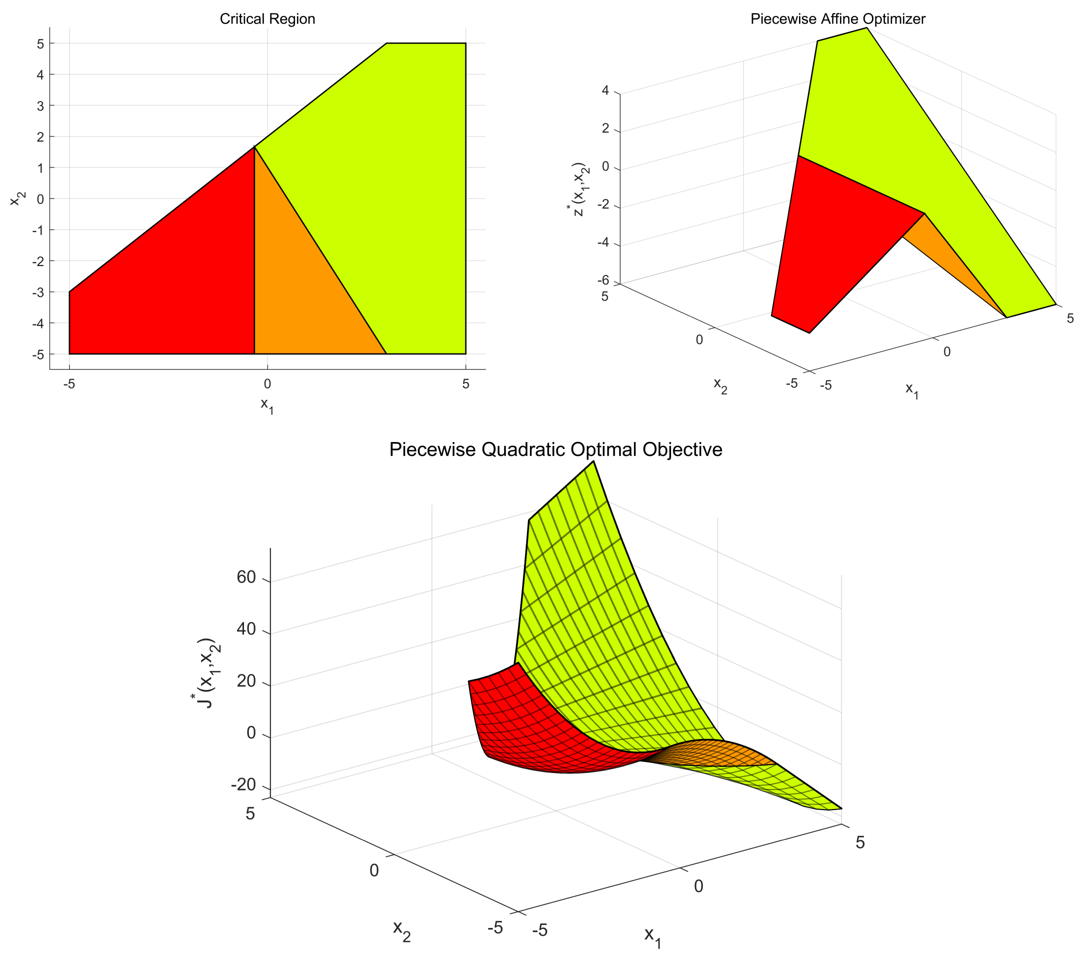

Figure 2: mp-QP results output by the MPT toolbox

10. **[4 pt]** Identify the features of $z^*(x_1, x_2)$, and $J^*(x_1, x_2)$: piece-wise linear/quadratic? Discountinuous, continous or continously differentiable?

Solution:

The optimizer $z^*(x_1, x_2)$ is piece-wise affine, continous but not continously differentiable. The optimal objective $J^*(x_1, x_2)$ is piece-wise quadratic, continous and continously differentiable.

3. **Multiparametric Programming**

A: mpQP and mpLP

[7 pt] Consider a system with the state space representation $x_{k+1} = 0.5x_k + u_k$. For this dynamics, solve the QP (1) and the LP (2) parametrically by taking x_0 as a parameter. Plot the optimal $u_0^*(x_0)$ and the optimal cost $J^*(x_0)$:

$$\begin{aligned} \min_{x,u} \ & \frac{1}{2}\left(x_1^2 + x_2^2 + u_0^2 + u_1^2\right) \\ \text{subj. to} \ & 2.5 \le x_1 \le 5 \\ & -1 \le x_2 \le 1 \\ & -2 \le u_0 \le 2 \\ & -2 \le u_1 \le 2 \end{aligned} \tag{1}$$

$$\begin{aligned} \min_{x,u} \ & |x_1| + 0.5\,|x_2| + 0.5\,|u_0| + |u_1| \\ \text{subj. to} \ & 2.5 \le x_1 \le 5 \\ & -1 \le x_2 \le 1 \\ & -2 \le u_0 \le 2 \\ & -2 \le u_1 \le 2 \end{aligned} \tag{2}$$

Solution:

The coding detials are shown in the live scripts, the required figures are shown below:

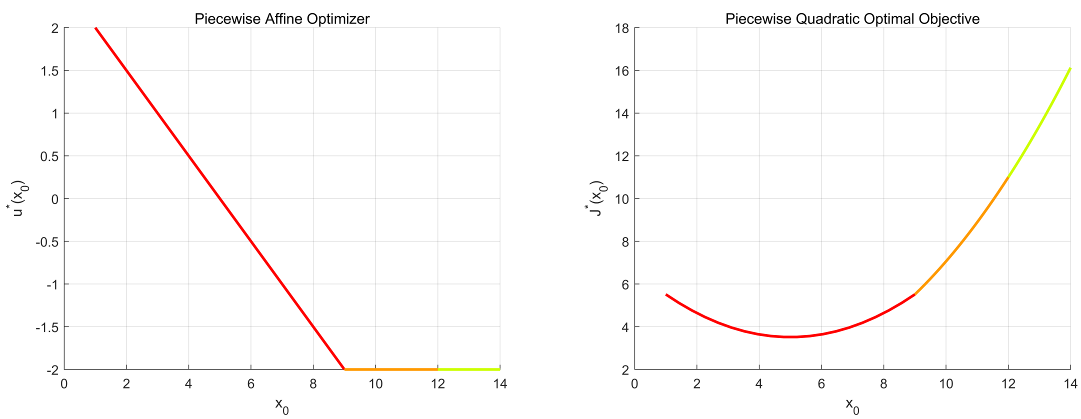

Figure 3: mp-QP solution: optimizer $u_0^*(x_0)$ and opitmal cost $J^*(x_0)$

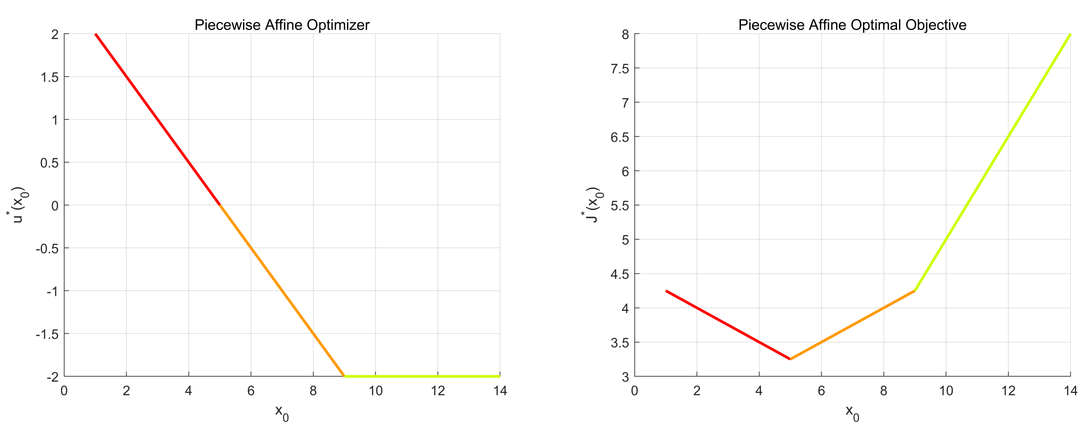

Figure 4: mp-LP solution: optimizer $u_0^*(x_0)$ and opitmal cost $J^*(x_0)$

B: Constrained Optimal Control, Multiparametric Programming and Dynamic Programming

Consider the discrete-time system model:

$$\begin{cases} x_{k+1} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_k \\ y_k = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k \end{cases} \tag{3}$$

Define the following cost function:

$$\left\| \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x_4 \right\|_\infty + \sum_{k=0}^{3} \left(\left\| \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x_k \right\|_\infty + |0.8 u_k| \right) \tag{4}$$

and assume the constraints are

$$\begin{gathered} -1 \leq u_k \leq 1 \quad k = 0, 1, \ldots, 3 \\ \begin{bmatrix} -10 \\ -10 \end{bmatrix} \leq x_k \leq \begin{bmatrix} 10 \\ 10 \end{bmatrix} \quad k = 0, 1, \ldots, 4 \end{gathered} \tag{5}$$

1. **[4 pt]** Compute $u_0^* (x_0)$ for $x_0 = \begin{bmatrix} -5 & 0 \end{bmatrix}^{\mathrm{T}}$ by solving a Linear Program.
   Solution:
   The coding detials are shown in the live scripts, the solution is of the LP is:
   $$u_0^* (x_0) = \begin{bmatrix} 1 & 0.333 & 0 & -0.667 \end{bmatrix}$$
2. **[8 pt]** Compute the state feedback solution $u_0^* (x_0), u_1^* (x_1), \ldots, u_3^* (x_3)$ by using the batch approach and mpLPs. Check the solution obtained at the previous point for the same x_0.
   Solution:
   The coding detials are shown in the live scripts, the state feedback solution is plot below:

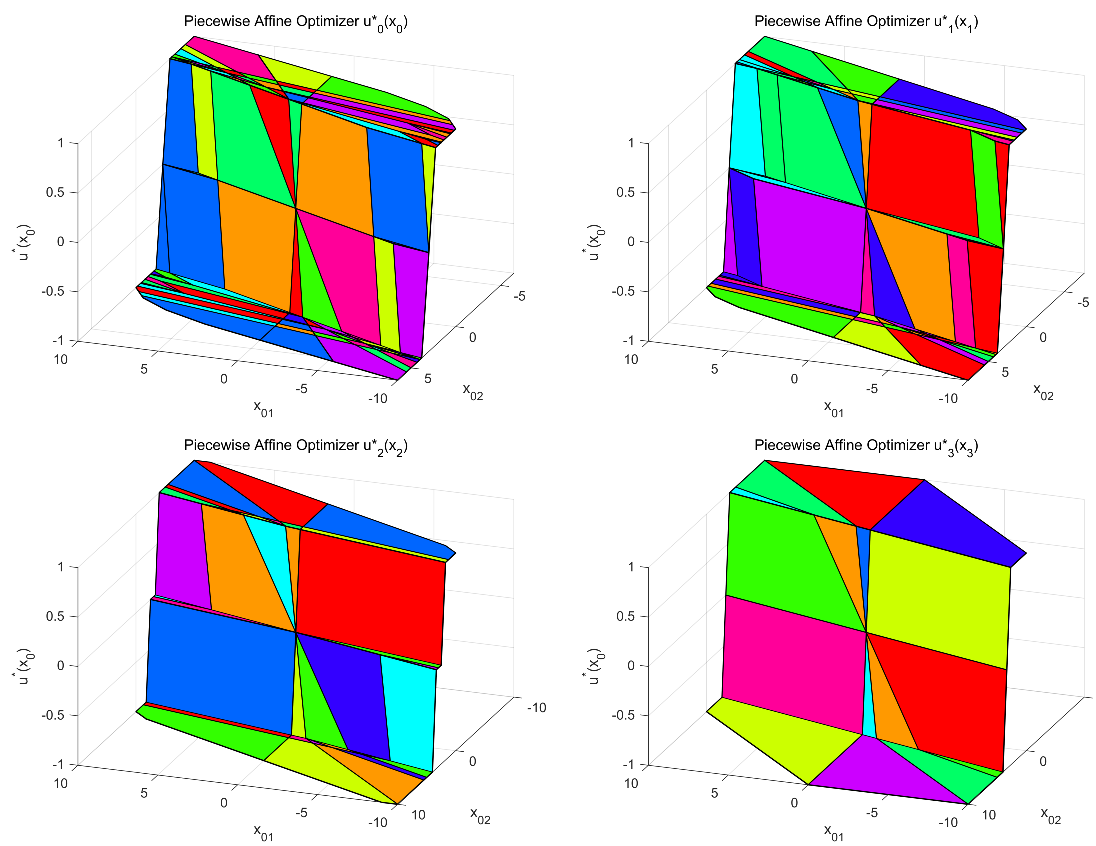

Figure 5: state feedback solution $u_0^* (x_0), u_1^* (x_1), \ldots, u_3^* (x_3)$

The $u_0^*(x_0)$ dervied using mp-LP and assigning $x_0 = \begin{bmatrix}-5 & 0\end{bmatrix}^{\mathrm{T}}$ is $u_0^*(x_0) = \begin{bmatrix}1 & 0.333 & 0 & -0.667\end{bmatrix}$, which is the same as what we get using a single LP

3. **[3 pt]** Describe in words how MPT can be used to compute the state feedback solution $u_0^*(x_0)$, $u_1^*(x_1), \ldots, u_3^*(x_3)$ using dynamic programming and mpLPs.
   Solution:
   In the MPC text book section 11.4.3, Theorem 11.6: The state feedback piecewise affine solution of the CFTOC for 1-norm or ∞-norm case is obtained by solving the optimization problem via N mp-LPs. How to do the recursive approach (dynamic programming) back propagation is also shown in that section.
   Therefore, as is shown in the above example, the MPT toolbox can be used to solve multiple mp-LPs, hence it can be used to generate state feedback solution for the CFTOC for 1-norm or ∞-norm case.
   Moreover, if we are solving CFTOC using receding horizon control, at every timestep, the state of the system is considered as a new initial state. We can solve a mp-LP at each timestep and develops a "look-up map" for the control inputs based on the state at that time. In this sense, it also creates a state feedback solution.