SSY281 - Model predictive control
Assignment 4

Nicholas Granlund

February 27, 2023

Question 1: Constrained optimization

(a) Find the V- and H-representation of the polyhedron. Explain the differences. The V- and H-representation for the polyhedron can be seen in Figure 1 below. As expected they are the same. They are defined by expression (1).

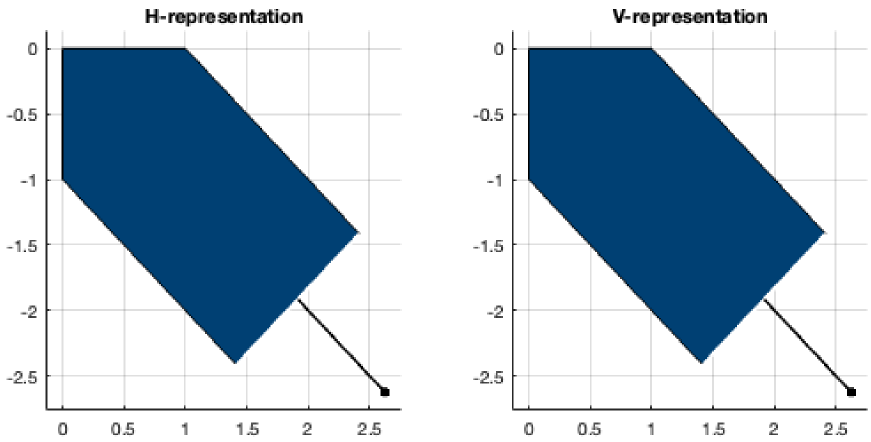


Figure 1: H-representation and V-represetation of the polyhedron

$$V = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad R = \begin{bmatrix} 1 & -1 \end{bmatrix} \qquad H = \begin{bmatrix} A & b \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ -1 & -1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{1}$$

The V-representation consists of the matrix V where each row is a vertices. In the case of an unbounded polyhedron, a ray R is introduced that shows the direction in which the polyhedron is unbounded. For the H-representation, each row instead represents a half-plane.

Both representations define the same polyhedron. The difference is that the H-representation defines the set as the intersection of a finite number of hyperplanes. The V-representation is instead defined as a convex union of finite number of verticies.

(b) Plot the different polyhedrons
The different polyhedrons can be seen in Figure 2. The observations are as follows:

Polyhedron P
Just P.

Polyhedron Q
Just Q.

Polyhedron $P + Q$
The Minkowski sum of P and Q.

Polyhedron $P - Q$
The Pontrygian difference between P and Q.

Polyhedron $(P - Q) + Q$
The Minkowski sum of the polyhedron $(P - Q)$, and Q.

Polyhedron $(P + Q) - Q$
The Pontryagian difference between the polyhedron $(P + Q)$, and Q.

Polyhedron $(Q - P) + P$
The Minkowski sum of the polyhedron $(Q - P)$, and P. Here something interesting occurs. We have that $Q - P = \{\}$ and we know that $\{\} \bigoplus P = \{\}$. However MPT seems to incorrectly return the polyhedron P. The plot is therefore false and should instead be empty...

Polyhedron $(Q + P) - P$
The Pontryagian difference between the polyhedron $(Q + P)$, and P.

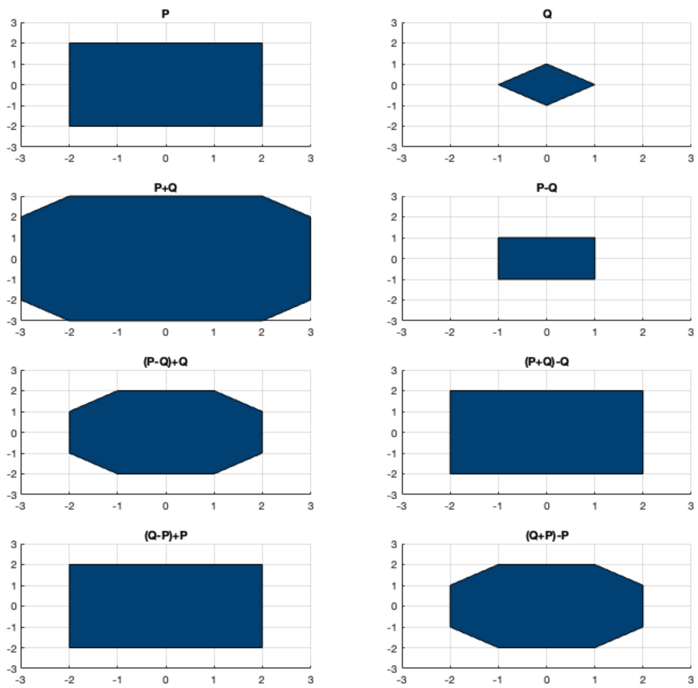


Figure 2: The different Polyhedrons demonstrating Minkowski sum and Pontryagian difference operations.

Question 2: Forward and backward reachability

(a) Show that the set S is positively invariant for the system.
We know that the set S is positively invariant if for all x in the set S, all x in the next iteration also belongs to the set S. i.e $\forall x \in S \rightarrow x^+ \in S$. We can construct the inequalities (hyperplanes) which defines this reachable set. $x^+ = Ax \rightarrow A^{-1}x^+ = x$ which gives the reachable set:

$$\text{reach}(S) = \{x : A_{in}A^{-1}x \leq b_{in}, x \in \mathbb{R}^2\}$$

And since we know that the system $x^+ = Ax$ is stable (eig(A) is in unit disc), we know that the reachable set reach(S) $\subseteq S$, therefore it is positively invariant. The sets can be seen in Figure 3.

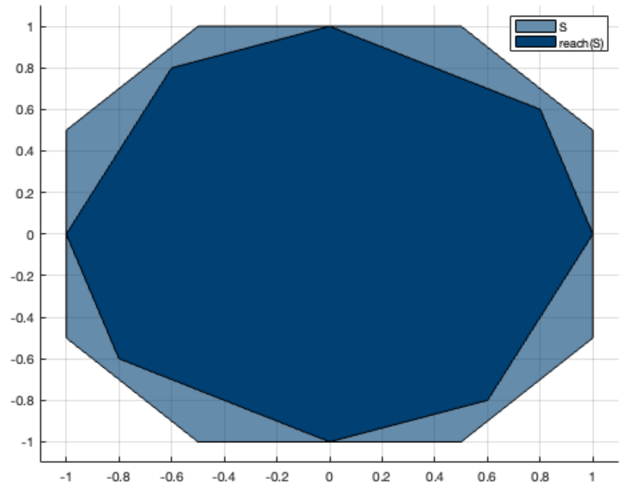


Figure 3: The set S and reach(S). Illustrates how reach(S) $\subseteq S$.

(b) Calculate the one step reachable set from S given the new system.
Firstly we can define the constraint on u as a set. With the constraint $-1 \leq u \leq 1$ we can define set U as:

$$U = \{u : A_u u \leq b_u, u \in \mathbb{R}\} \qquad A_u = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, b_u = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Now we can continue to perform similar manipulations as the previous question. We have that $x^+ = Ax + Bu$ which gives $x = A^{-1}x^+ - A^{-1}Bu$. Using this and inserting it to the inequlity that defines set S we get the total inequality as:

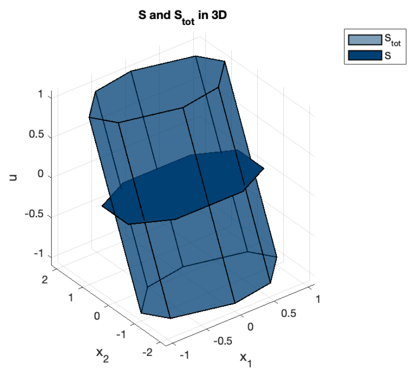
$$A_{in}A^{-1}x^+ - A_{in}A^{-1}Bu \leq b_{in}$$

Now since this inequality depends on both x and u, we have to combine these. If we combine this inequality with the definition of the set U. We can obtain the 3D set for x^+ as:

$$\underbrace{\begin{bmatrix} A_{in}A^{-1} & -A_{in}A^{-1}B \\ 0_{2\times 2} & A_u \end{bmatrix}}_{A_{tot}} \begin{bmatrix} x \\ u \end{bmatrix} \leq \underbrace{\begin{bmatrix} b_{in} \\ b_u \end{bmatrix}}_{b_{tot}}$$

$$S_{tot} = \{x, u : A_{tot} \begin{bmatrix} x \\ u \end{bmatrix} \leq b_{tot},\ x \in \mathbb{R}^2,\ u \in \mathbb{R}\}$$

The set S_{tot} can be seen in Figure 4 to the left. And clearly, the reachable set is then the set of x_1 and x_2 which are contained in this set. Simply it is the projection of this set onto the x_1x_2 plane. The projection can be seen in Figure 4 to the right. The reachable set are in this case larger than the previous set and we can set that $S \subseteq \text{reach}(S)$.



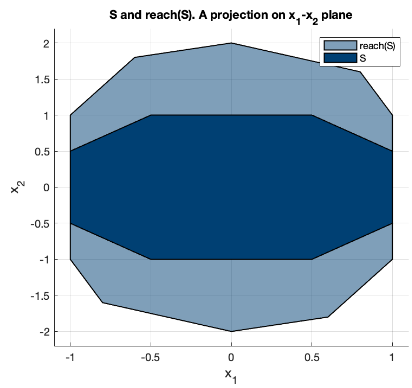


Figure 4: The set S and S_{tot} (left). And the set S and reach(S) as projection (right).

(c) Calculate one step "Pre" for S.

This is done in a similar fashion as the previous question. We still have the same set U, and we have that $x^+ = Ax + Bu$. Since we want to find the predecessors x for which x^+ can be reached, we simply get:

$$A_{in}(Ax + Bu) \leq b_{in}$$

Combining this with the set U we get that

$$\underbrace{\begin{bmatrix} A_{in}A & A_{in}B \\ 0_{2\times 2} & A_u \end{bmatrix}}_{A_{tot}} \begin{bmatrix} x \\ u \end{bmatrix} \leq \underbrace{\begin{bmatrix} b_{in} \\ b_u \end{bmatrix}}_{b_{tot}}$$

$$S_{tot} = \{x, u : A_{tot} \begin{bmatrix} x \\ u \end{bmatrix} \leq b_{tot},\ x \in \mathbb{R}^2,\ u \in \mathbb{R}\}$$

The set S_{tot} can be seen in Figure 5 to the left. Now similar to the previous set. If we want to obtain the Pre(S), we have to look at the projection of the set S_{tot} on the x_1x_2 plane. The predecessor set Pre(S) can be seen in Figure 5 to the right. We can see that $S \subseteq \text{Pre}(S)$

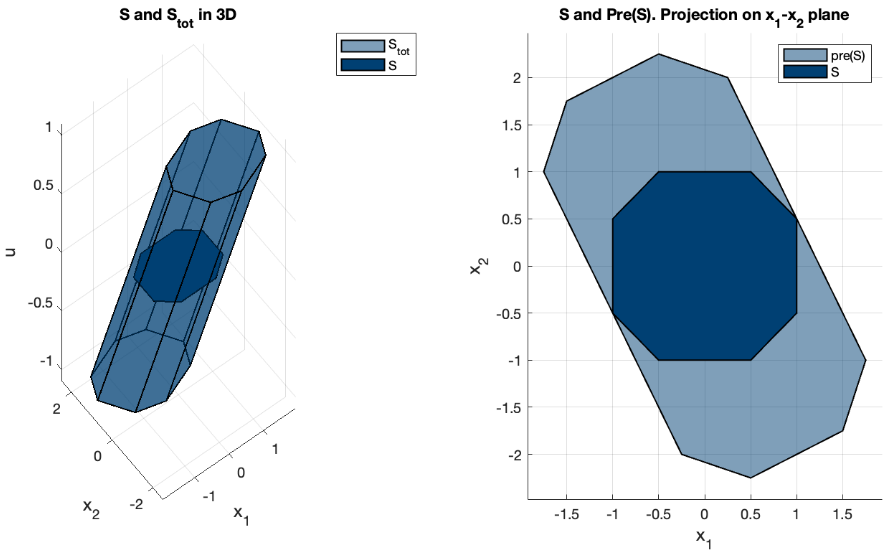


Figure 5: The set S and S_{tot} (left). And the set S and Pre(S) as projection. (right)

Question 3: Persistent feasibility

(a) Find shortest N such that the controller is feasible.
The N is found using a variety of the functions availablie in the MPT toolbox. Please see the matlab code under section (3a). It was calculated that the controller is feasible first when the horizon $N = 26$. Its trajectory can be seen in Figure 6 below. To clarify, the initial set and terminal set are specific points. their size has been exagurated to be visible in the plot. The set of reachable states for each time instant k is also plotted and can be seen in Figure 7. Please see Matlab code for cinematic illustration.

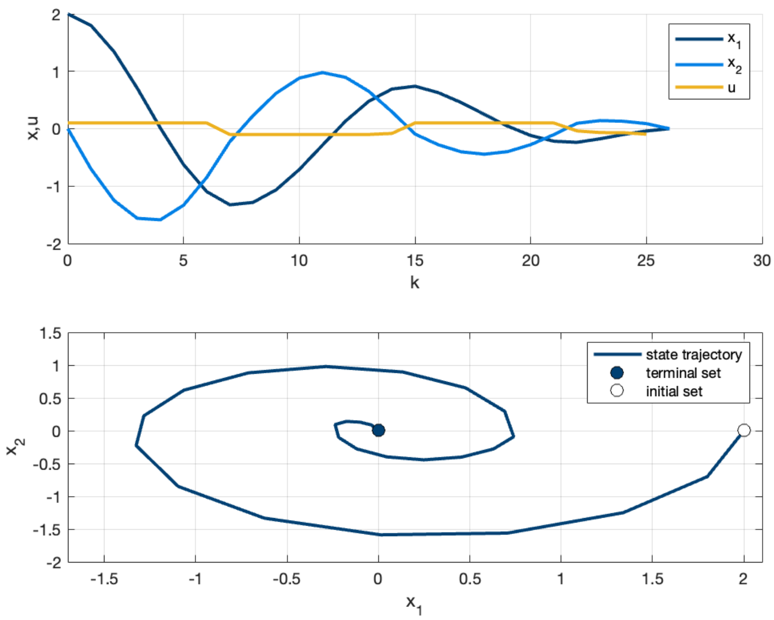


Figure 6: System states for $N = 26$. terminal set (0,0) has been reached.

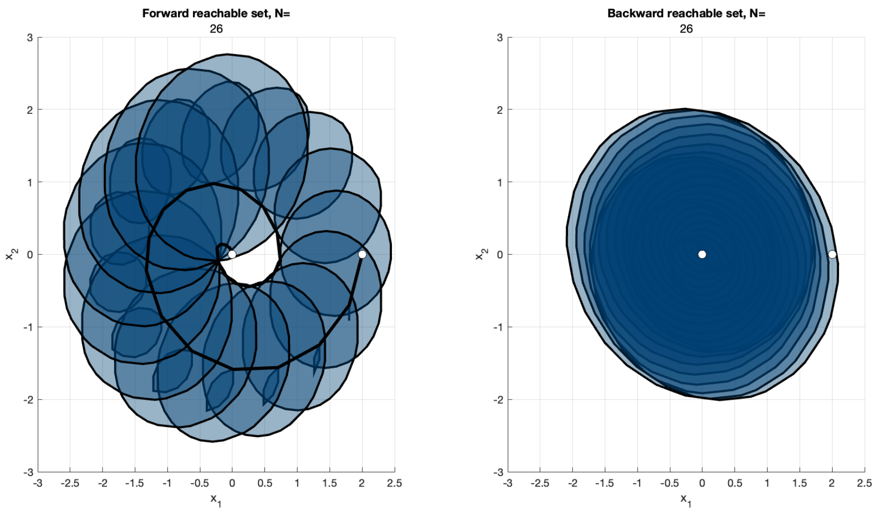


Figure 7: Sets of forward/backward reachable states for each given instant k. It is clear how the states always are at the outermost point of each set (closest to the setpoint (0,0) to minimize state error penalty).

(b) Set $N = 2$ and choose X_f as the maximum control invariant set. Is the controller still feasible?

By doing this we still get that the controller is feasible. We do this because the initial state is a part of the control invariant set $x_0 \in X_f = C_\infty$. This can be seen in Figure 8 below. We can also see that the system will never leave X_f and is thus feasible all the way to the origin.

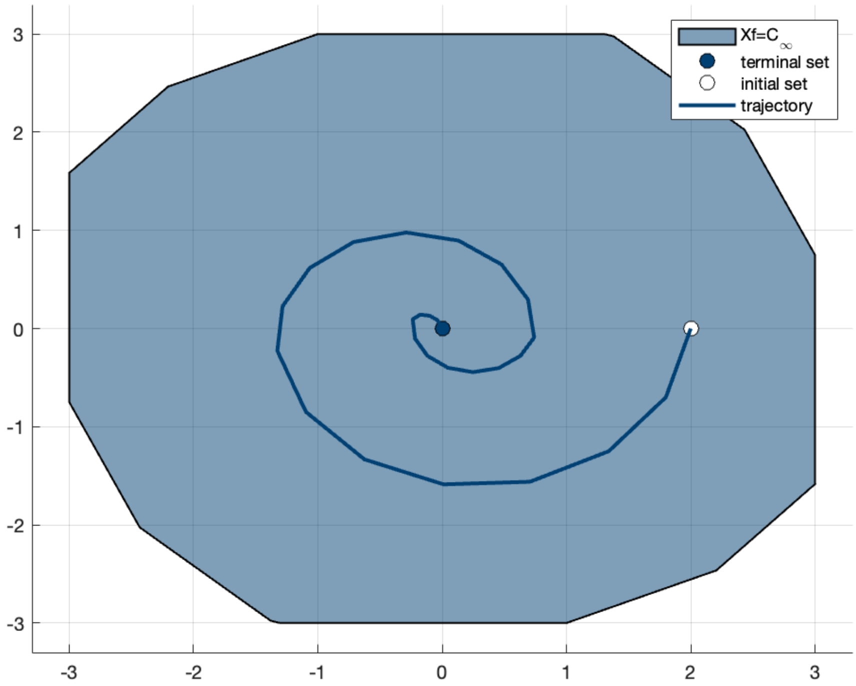


Figure 8: Control invariant set.

(c) Plot the set of feasible initial states for the two controllers...
The set of feasible initial states can be seen in Figure 9 below. The smaller set X_{N1} (darker blue) is the same set which can be seen in Figure 7 to the right, which is the backwards reachable set for the first controller.

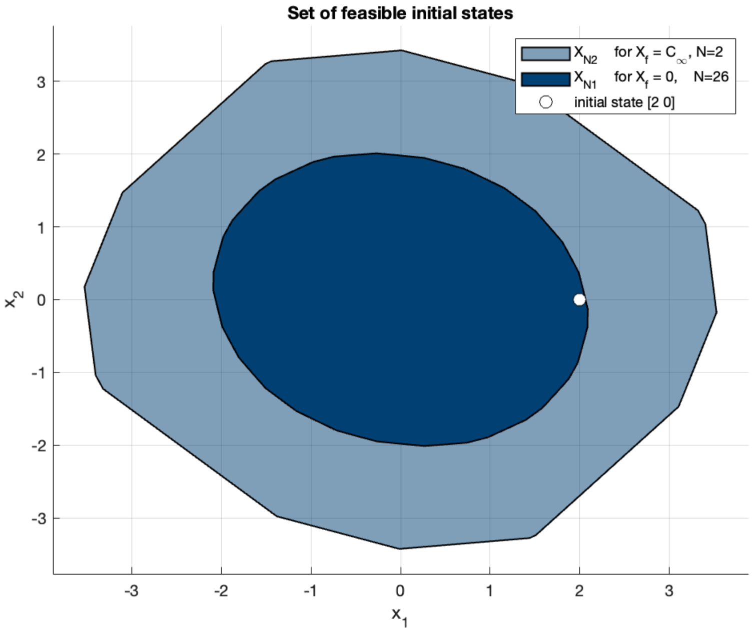


Figure 9: Set of feasible initial states for the two controllers.

The result is not surprising. The first controller (dark blue) has a smaller feasible set X_{N1} than the second controller (light blue) X_{N2}. The feasible set X_{N1} is smaller because the controller has a much smaller terminal state constraint set X_f, which is only the origin (0,0). And with a small terminal set constraints, follows a small feasibility set.

The second controller has a much larger terminal state constraint set X_f and thus a larger feasible set X_{N2}.

The size of the feasible set is also dictated by how long the prediction horizon N is. A longer horizon gives a larger feasible set. However, even though the second controller has a significantly shorter prediction horizon N, the larger terminal state constraint set X_f makes up for it and still gives a larger feasible set compared to the first controller.

Both controllers works on optimizing two states (and one control input), so they both have the same amount of optimization variables. However the first controllers feasible set is defined by 52 inequalities courtesy of the longer horizon $N = 26$, whereas the second controllers feasible set only consists of 16 inequalities. This means that optimizing the states for the first controller is much more computational expensive since the states has to comply to 52 constraints...

/ Nicholas Granlund