# SSY281 - Model predictive control
# Assignment 5

Nicholas Granlund

March 6, 2023

# Question 1: Linear MPC design

### (a) Find a terminal weight $P_f$ such that the system is asymptotically stable $\forall x_0 \in X_N$

Asymptotic stability can be obtained $\forall x_0 \in X_N$ if $P_f$ is chosen as the value function function $V_f(x(k)) = V_\infty^{uc}(x(k)) = x^\top(k)Px(k)$ from the corrresponding unconstrained LQ-problem. where $P$ is the solution to the discrete algebraic Riccati equation. Solving for $P$ with Matlabs idare() we obtain:

$$P = \begin{bmatrix} 16.0929 & 32.9899 \\ 32.9899 & 93.9396 \end{bmatrix}$$

Giving us:

$$V_f(x(k)) = x^\top(k) \begin{bmatrix} 16.0929 & 32.9899 \\ 32.9899 & 93.9396 \end{bmatrix} x(k)$$

According to Theorem 11.13 in the lecture notes, by choosing $P_f = V_f(x(k))$ we can guarantee asymptotic stability $\forall x_0 \in X_N$ since the origin is asymptotically stable with a region of attraction defined by the set $X_N$. The set $X_N$ for $N = 4$ can be seen in Figure 1 below.

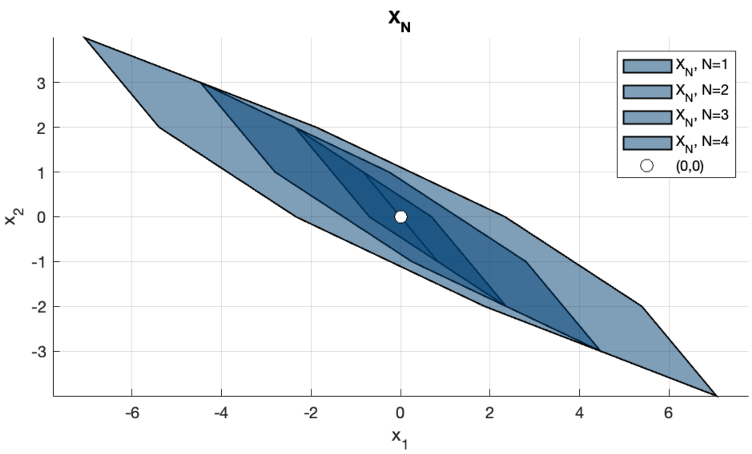


Figure 1: Set $X_N$

**(b) Design an MPC for $N = [10, 15, 20]$. Analyze and explain.**

The three controllers can be seen in Figure 2 below. All three controllers are feasible since $x_0 = [7 \quad -4]^\top$ belongs to the set of feasible initial points which is illustrated by the plots underneath. The controller needs a prediction horizon of $N > 5$ to be feasible. This is because the backwards reachable set from $(0, 0)$ does not contain $x_0$ when $N$ is shorter than 6.

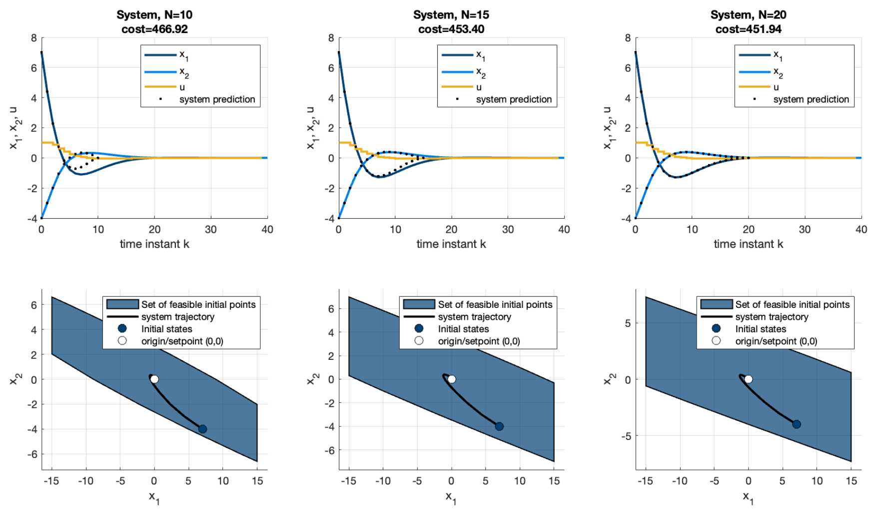


Figure 2: RHC for $N = [10, 15, 20]$. ploted from $k = 0$ to $k = 40$

We can see that, when the controller has a shorter prediction horizon, the predicted system trajectory gets worse. Furthermore, the total cost also gets larger. For the controller with $N = 20$, the prediction is rather similar to the actual trajectory of the system.

The controllers with shorter horizon lengths $N = 10, 15$ are unable to accurately predict the correct trajectory because future states beyond the predicted horizon will affect the cost function, unbeknownst to the controller. This leads to a change in the optimal control input applied in subsequent time steps, resulting in a change in the trajectory.

This is not the case when the controllers prediction horizon is sufficiently long, such that future states beyond the horizon will not affect the total cost. Given such long horizons, the controller is able to commit to the predicted control, without alteration in the subsequent time steps. By testing we find that when $N = 22$, the predicted trajectory is identical to the actual trajectory of the system, and the cost function has reached the global minimum at 451.91. By reaching the global minimum of the cost function, we are ensured that we have the best control policy possible for the system. Horizons longer than $N = 22$ will hence not improve performance of the controller, but instead increase the computational complexity of the optimization...

**(c) Assume $N = 20$, $X_f = \{x : ||x||_\infty \leq 0.01\}$, and $P_f$ as the Riccati solution. Find the explicit-MPC solution and plot the state-space partitions**

Finding the explicit solution is simply done by utilizing the functions in the MPT toolbox. Please be advised that, the code generating the explicit solution, does not work on Matlab R2022a and/or on a MACINTOSH. The solution was thus generated using another windows-based computer using the version R2021b. The state-space partitions can be seen in Figure 3 below.

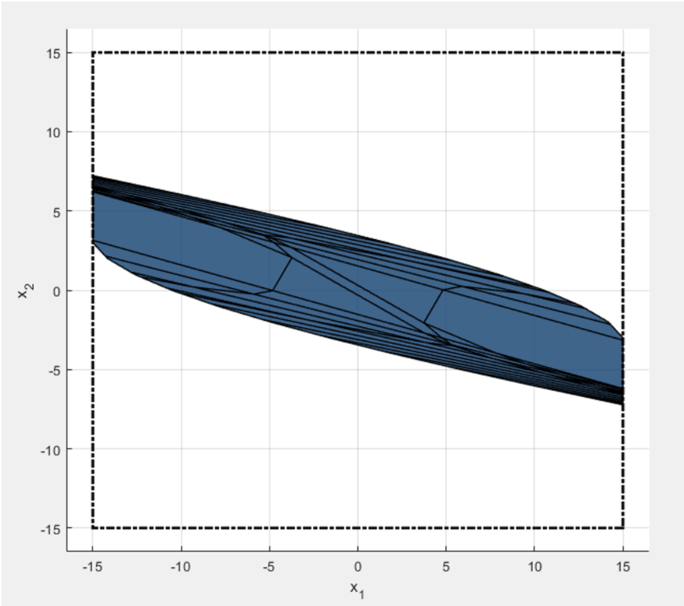


Figure 3: State-space partitions of the explicit solution. There is a total of 71 partitions.

**(d) Assume $N = 1$, can you choose a new $X_f$ so that persistent feasibility is guaranteed $\forall x_0 \in C_\infty$? Motivate your answer.**

Yes this is possible. Firstly, we know that recursive feasibility is obtainable if $X_f$ is a subset of the control invariant set $C_\infty$, i.e $X_f \subseteq C_\infty$. By having $N = 1$ we want the reachable set from $X_f$ to be amongst the control invariant set $C_\infty$. By this logic we should choose the terminal set $X_f$ as the intersection of the control invariant set $C_\infty$ with the set which is reachable from $C_\infty$ in one step. This will guarantee persistent feasibility, hence we get:

$$X_f = \text{reach}^{N=1}(C_\infty) \cap C_\infty$$

The set $X_f$ can be seen in Figure 4 below.

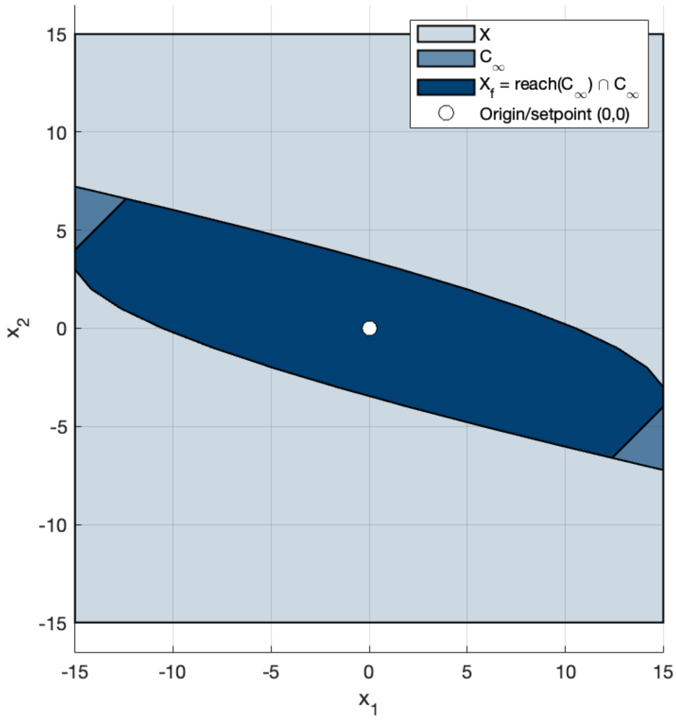


Figure 4: The set $X_f$ which guarantees persistent feasibility.

# Question 2: Finite time control of DC motor

## (a) Find matrices $A$,$B$ and $C$ in the discrete time model.

By constructing the continuous model with the given parameters, it is rather straightforward to discretize. Doing so with the sampling interval $h = 0.1$ we obtain the matrices:

$$A = \begin{bmatrix} 0.7637 & 0.0873 & 0.0118 & 0.0003 \\ -4.4281 & 0.6764 & 0.2214 & 0.0086 \\ 0.4444 & 0.0159 & 0.9778 & 0.0621 \\ 7.1286 & 0.4285 & -0.3564 & 0.3449 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0000 \\ 0.0003 \\ 0.0036 \\ 0.0621 \end{bmatrix}$$

$$C = \begin{bmatrix} 1280.2000 & 0.0000 & -64.0100 & 0.0000 \end{bmatrix}$$

**(b) Start from the initial state $x(0) = [0 \quad 2.5 \quad 0 \quad 75]^\top$. Design a minimum time controller that brings the system to a standstill.**

By bringing the system to a standstill we simply want the angular velocities to reach zero. This means that we want to reach any set of states where $x_2 = 0$ and $x_4 = 0$, i.e there is no target value for $x_1$ and $x_3$. By designing a minimum time controller we get that it is possible to bring the system to a standstill in $t = 0.30$ seconds whilst not violating the constraint $\mid V \mid \leq 200$ V. This is achieved with some rather aggressive control, yielding large torsional torque. The system states, measurements and inputs can be seen in Figure 5 below.

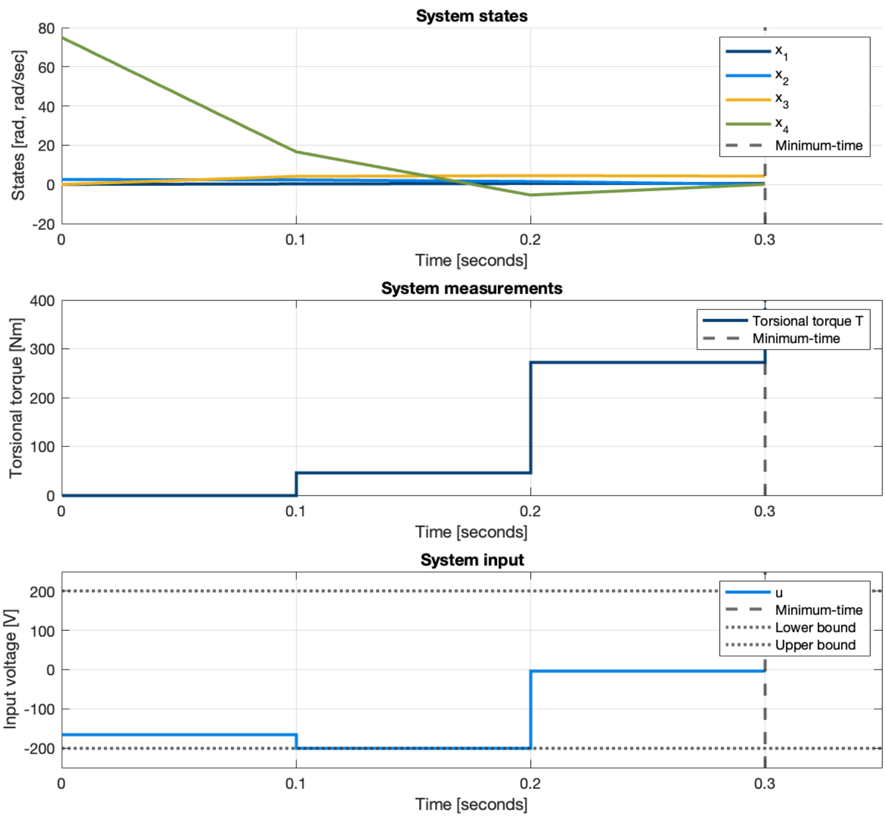


Figure 5: System states, measurements and control input for minimum-time controller. The system is brought to a standstill in $t = 0.30$ seconds

**(c) Start from the initial state $x(0) = [0 \;\; 2.5 \;\; 0 \;\; 75]^\top$. Design a minimum time controller that brings the system to a standstill whilst the torque is constrained by $\mid T \mid \leq 150$ Nm.**

This task is approached in a similar manor as the previous one. However, in this case we have to constrain the maximum torque applied, which is generated by the inertia whilst braking the system. Simply put, we dont want the states of the system to violate $\mid Cx(k) \mid \leq 150$ for any $k$. In this case, the designed minimum-time controller is able to bring the system to a standstill in $t = 0.60$ seconds. The system states, measurements and inputs can be seen in Figure 6 below.

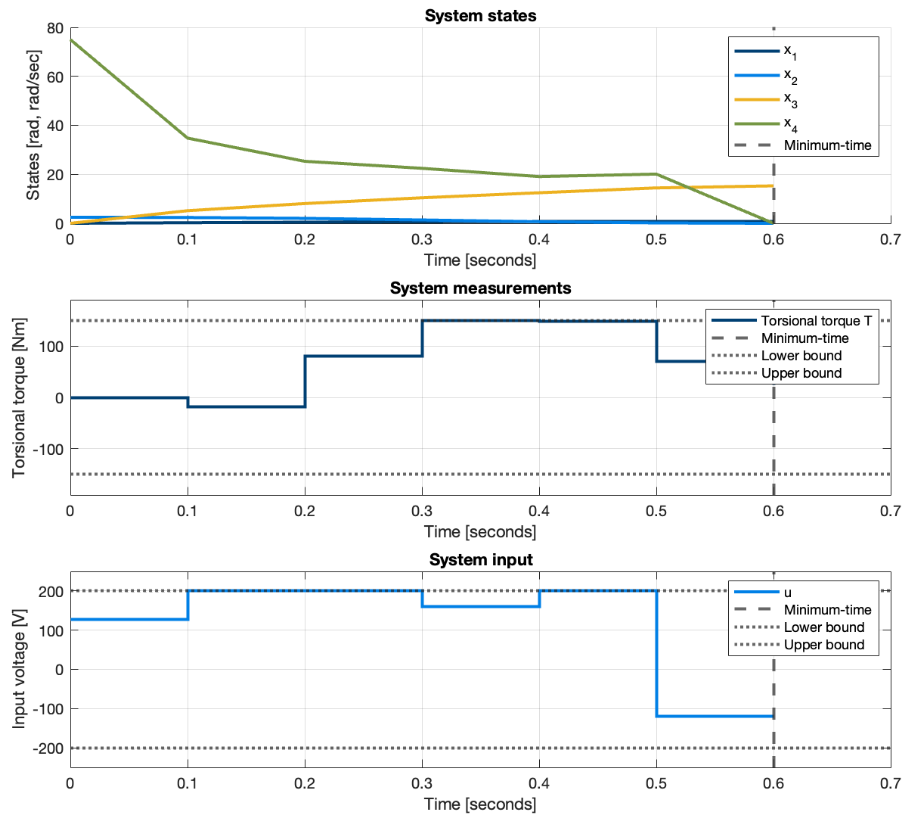


Figure 6: System states, measurements and control input for minimum-time controller. The system is brought to a standstill in $t = 0.60$ seconds

**(d) Start from the initial state $x(0) = [0 \quad 2.5 \quad 0 \quad 75]^\top$. Design a minimum time controller that brings the system to $X_{target}$.**

This task is somewhat more intricate than the previous ones. The important part here is that we shouldn't just drive the system towards the target set $X_{target}$, but we should drive it towards the control invariant part of $X_{target}$. This means that we want the final set to be:

$$X_f = X_{target} \cap C_\infty$$

This implies that when the system is brought to $X_f$, it will remain inside $C_\infty$ regardless of what control $u$ is applied. The designed minimum-time controller is able to bring the system to a standstill in $t = 1.10$ seconds. After this, the controller is turned off ($u = 0$). The system states, measurements and inputs can be seen in Figure 7 below. The system with longer simulation can be seen in Figure 8. That figure demonstrates the decay of torsional torque $T$ and that the system states reamins inside $X_f$ even though no control is applied.

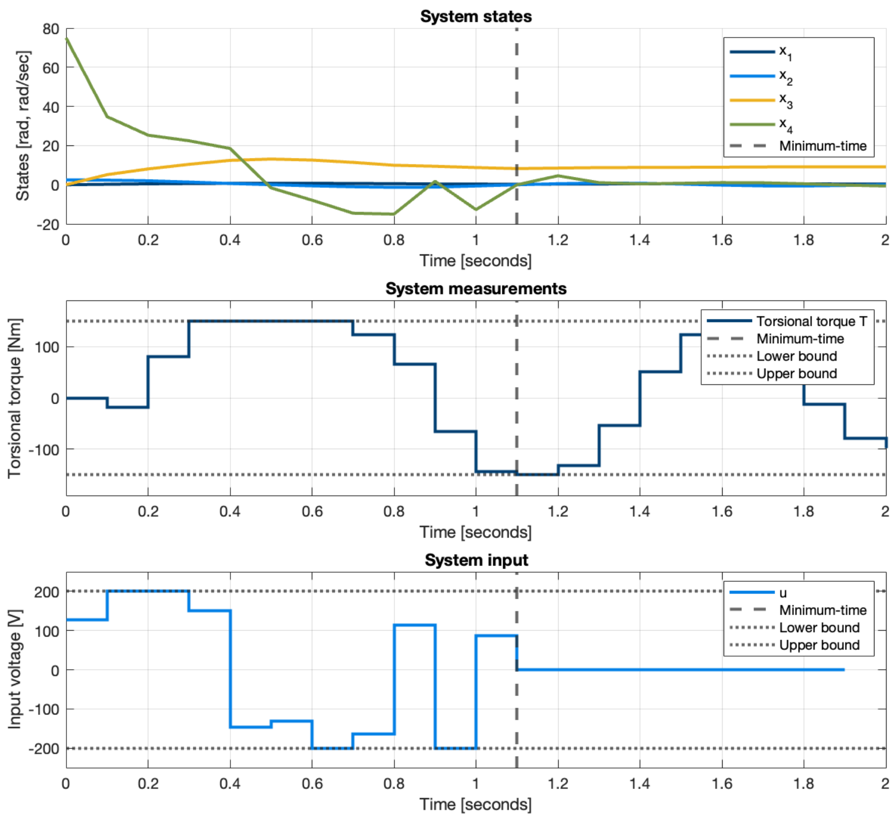


Figure 7: System states, measurements and control input for minimum-time controller. The system is brought to a standstill in $t = 1.10$ seconds

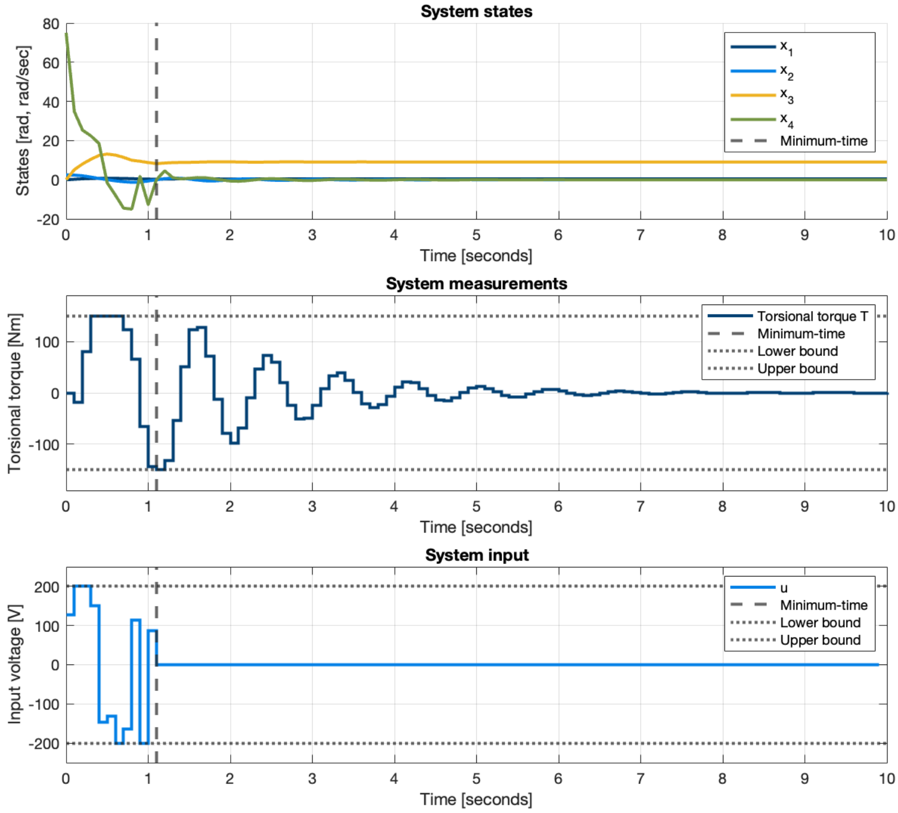


Figure 8: System states, measurements and control input for minimum-time controller. controller is turned off at $t = 1.1$ seconds. Since the system is brought to the control invariant set of states, it will remain there even if no control is applied.

/Nicholas Granlund