

SSY281 MODEL PREDICTIVE CONTROL  
ASSIGNMENT 5 – EXPLICIT MPC AND MINIMUM TIME CONTROL

The purpose of this assignment is to understand how finite time control can be constructed from a family of explicit MPC algorithms.

**Instructions**

The assignments comprise an important part of the examination in this course. Hence, it is important to comply with the following rules and instructions:

- The assignment is pursued and reported individually.
- The findings from each assignment are described in a short report, written by each student independently.
- The report should provide clear and concise answers to the questions, including your motivations, explanations, observations from simulations, etc. Conclusions should be supported by relevant results if applicable; e.g., the system is stable since the eigenvalues,  $[0.5, 0.2 + 0.5j, 0.2 - 0.5j]$ , are inside the unit circle. Figures included in the report should have legends, should be readable, should have proper scaling to illustrate the relevant information, and axes should be labeled. Try to verify your solutions if possible; e.g., plot the inputs and outputs and see whether they respect the constraints.
- Since the assignments are part of the examination in the course, plagiarism is of course not allowed. If we observe that this happens anyway, it will be reported.
- The report should be uploaded to Canvas *before the deadline*. A report uploaded a second or a day after the deadline are penalized equally. Name the report as A5.pdf.
- A MATLAB code should be uploaded which reproduces all numbers and figures in your report. Make sure that one can run your code and see your results without any error. Name the MATLAB script as A5.m.

Table 1: Points per question

Question:	1	2	Total
Points:	6	9	15

## 1. Linear MPC design

Consider the following system

$$x(k+1) = \begin{bmatrix} 1.2 & 1 \\ 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k) \quad (1)$$

The input and state constraints are

$$\mathcal{U} : -1 \leq u(k) \leq 1; \quad (2a)$$

$$\mathcal{X} : \begin{bmatrix} -15 \\ -15 \end{bmatrix} \leq x(k) \leq \begin{bmatrix} 15 \\ 15 \end{bmatrix} \quad (2b)$$

- (a) [2p] Consider  $Q = I_2$ ,  $R = 100$ , and  $N = 4$ . Assume  $\mathcal{X}_f = \mathbf{0}$ , and find a terminal weight  $P_f$  for the constrained receding horizon controller to guarantee asymptotic stability for all  $x_0 \in \mathcal{X}_N$ . Motivate the choice of  $P_f$  in the report and plot  $\mathcal{X}_N$ .
- (b) [2p] Set  $x(0) = [7 \ -4]^T$  and design an MPC controller for  $N = 10, 15, 20$ . Provide three figures in the report where each one is simulated for  $N$  time steps and contains the state predictions at time zero and the actual states when MPC is implemented. Analyze and explain the mismatch between predicted vs closed-loop trajectories (specifically at time  $N$ ) as you increase  $N$  in the MPC design.
- (c) [1p] Assume  $N = 20$ ,  $\mathcal{X}_f = \{x : \|x\|_\infty \leq 0.01\}$ , and  $P_f$  as the Riccati solution. Find the explicit-MPC solution and plot the state-space partitions (you should use MPT to solve this question).
- (d) [1p] Assume  $N = 1$ . Can you choose a new  $\mathcal{X}_f$  so that persistent feasibility is guaranteed for all  $x_0$  belonging to  $C_\infty$ ? Motivate your answer in the report (you should use MPT to solve this question).

## 2. Finite time control of a DC motor

Consider the model of a DC-servo, depicted below and with parameters listed in the table.

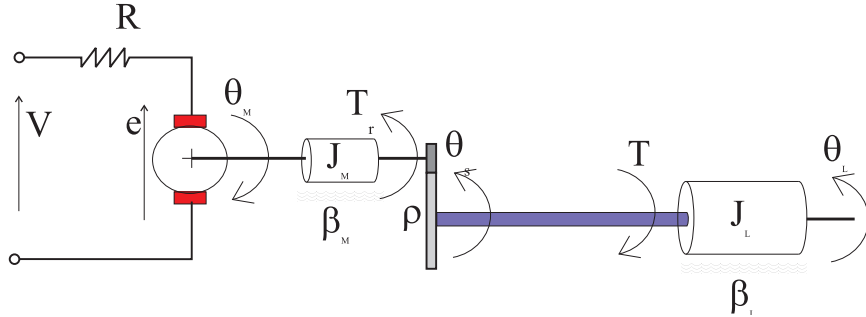


Figure 1: **DC-servomechanism.**

By defining the state  $x = [\theta_L \ \dot{\theta}_L \ \theta_M \ \dot{\theta}_M]^T$ , the model can be described in state-space form

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_\theta}{J_L} & -\frac{\beta_L}{J_L} & \frac{k_\theta}{\rho J_L} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k_\theta}{\rho J_M} & 0 & -\frac{k_\theta}{\rho^2 J_M} & -\frac{\beta_M + k_T^2/R}{J_M} \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{k_T}{R J_M} \end{bmatrix} V \quad (3a)$$

$$T = \begin{bmatrix} k_\theta & 0 & -\frac{k_\theta}{\rho} & 0 \end{bmatrix} x. \quad (3b)$$

System output is the torsional torque  $T$  and input is the DC voltage  $V$ . The DC voltage is constrained within the following range:

$$|V| \leq 200 \text{ V}.$$

Table 2: Model parameters

Symbol	Value (MKS)	Meaning
$L_S$	1.0	shaft length
$d_S$	0.02	shaft diameter
$J_S$	negligible	shaft inertia
$J_M$	0.5	motor inertia
$\beta_M$	0.1	motor viscous friction coefficient
$R$	20	resistance of armature
$k_T$	10	motor constant
$\rho$	20	gear ratio
$k_\theta$	1280.2	torsional rigidity
$J_L$	$50J_M$	nominal load inertia
$\beta_L$	25	load viscous friction coefficient

- (a) [1p] Using sampling interval  $h = 0.1$  s, find the matrices  $A$ ,  $B$ ,  $C$  in the following discrete-time model.

$$\begin{aligned} x(k+1) &= Ax(k) + BV(k), \\ T(k) &= Cx(k). \end{aligned} \tag{4}$$

- (b) [2p] Starting from the initial state  $x(0) = [0 \ 2.5 \ 0 \ 75]^\top$ , design a minimum-time controller that brings the system to the target set  $\dot{\theta}_L = \dot{\theta}_M = 0$ , i.e. it brings the system to standstill. Provide the minimum time and plot the predicted system states and output.

**Note:** You have to design your own minimum time controller, without using a ready solution from the MPT toolbox or elsewhere. You do not need to design an explicit minimum-time controller.

**Note 2:** by "predicted system states and output" it is meant the ones obtained in an open-loop manner, that is, you solve the MPC optimization problem for the whole prediction horizon and the states you get are the predicted ones (it is "open-loop" because you are not just using the first control input you computed, discard the rest and then recompute the whole thing for the next time instant; but rather, you compute control inputs and states for the whole prediction horizon and then you use those)

**Note 3:** if you get some error, when using some MPT functions to simulate the system, about linprog not accepting  $x_0$  as a parameter, add to the system some fictitious cost matrices  $Q = \mathbb{I}_4$ ,  $R = 1$  and  $P_f = \mathbb{I}_4$  (for some reason, the function expects them even if they are not needed and it will therefore not work without them)

- (c) [2p] With the same initial state as in b), design a minimum-time controller to bring the rotating shafts to standstill, such that the torsional torque is constrained within the range

$$|T| \leq 150 \text{ Nm}.$$

Provide the minimum time and plot the predicted system states and output.

- (d) [4p] With the same initial state as in b) and constraint on torsional torque as in c), design a minimum time controller to bring the rotating shafts close to standstill, i.e.,

$$X_{\text{target}} = \{x \in \mathbb{R}^4 : |x_1| \leq 10, |x_2| \leq 0.01, |x_3| \leq 10, |x_4| \leq 0.01\}.$$

To this end, first find the largest control invariant set inside  $X_{\text{target}}$  such that  $|V| \leq 200$  and  $|T| \leq 150$  also hold. Provide the minimum time and plot the predicted system states and output considering the invariant set as your terminal set. Create a closed-loop controller and plot the states and output for at least 2 s. Note that the prediction horizon in the closed

loop shrinks as time proceeds and it becomes one once the states are inside the invariant set, i.e., the system remains in its invariant set afterwards.

**Note:** as opposed to before, here you are asked both to plot the open-loop states and output (like previously) and also the closed-loop ones. By "closed-loop", it is meant the ones obtained in a receding horizon manner