Provide your results, discussion, and Matlab code for the following questions in a single PDF.

1. In this problem you will explore the stability (or instability) of closed-loop MPC with and without stabilizing terminal constraints. The goal is to use MPC to regulate a system to the origin with discrete-time state-space matrices

$$A = \begin{bmatrix} \frac{4}{3} & -\frac{2}{3} \\ 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} -\frac{2}{3} & 1 \end{bmatrix}. \tag{1}$$

   The controller prediction horizon is $N = 5$ steps and the quadratic cost function penalizes the states and inputs with matrices $Q = I$ and $R = I$. Assume there is no terminal cost ($P = 0$).

   (a) First, implement the unconstrained case (no input or state constraints) using Matlab and Yalmip. Use an initial condition of $x(0) = [3\ 3]^T$. Plot the closed-loop trajectories of the system in a single figure with three subplots for a 10 step simulation. Use the top subplot to show the state trajectories over time (provide a legend to label the two states), use the middle subplot to show the output trajectory, and use the bottom subplot to show the input trajectory. Comment of the transient behavior of the closed-loop system including the largest value of the output and the approximate time required to reach the origin.

   (b) Now add the output constraint $y(k) \leq 0.5$ to the controller formulation. Do not apply the output constraint at the first time step in the prediction horizon. Since, the output is only a function of the state, there is no value in constraining the output at first time step since it is based on the measured state. Also, based on the initial condition $x(0) = [3\ 3]^T$, the corresponding initial output is $y(0) = Cx(0) = 1$ which does not satisfy the imposed constraint and would lead to an infeasible optimization problem. Plot the closed-loop trajectories in the same way as the unconstrained case above with the addition of a horizontal dashed line indicating the output constraint on the middle subplot. Is the MPC controller stabilizing this this case? Can you tune the cost function matrices Q and R to regain stability?

   (c) Put the cost function matrices back to $Q = I$ and $R = I$. In an attempt to enforce stability based on what we proved in class, try adding the terminal constraint $x_N = 0$ to the controller formulation. What is the solution to the optimization problem at the first time step? Make sure you monitor the `diagnostics` output of the controller using a command similar to `[solutions,diagnostics] = controller{inputs}`. The `diagnostics` output will contain an error code as defined here `https://yalmip.github.io/command/yalmiperror/`. Try increasing the prediction horizon to see if this problem goes away.

   (d) To gain more insight into this problem, add a slack variable, s, to the output constraint such that $y(k) \leq 0.5 + s$. Constrain the slack variable to be non-negative and add a large cost term to the objective function to heavily penalize non-zero values of s. Play with different values of N to see how the output constraint violation and overall closed-loop behavior changes as a function of prediction horizon. Provide the same three-subplot figure from the steps above showing the state, output, and input trajectories for a value of N of your choosing. What can you say about the combination of the output constraint $y(k) \leq 0.5$ and the terminal constraint $x_N = 0$?

2. In this problem you will continue exploring the stability of closed-loop MPC but with a particular focus on determining the set of initial conditions for which the constrained MPC formulation admits a feasible solution, also known as the region of attraction. The goal is to use MPC to regulate a system to the origin with discrete-time state-space matrices

$$A = \begin{bmatrix} 2 & 1 \\ 0 & 2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{2}$$

The states and inputs are constrained such that $x_k \in \mathcal{X}$ and $u_k \in \mathcal{U}$ where

$$\mathcal{X} = \{x \in \mathbb{R}^2 \mid [1\ 0]x \leq 5\}, \quad \mathcal{U} = \{u \in \mathbb{R}^2 \mid -\mathbf{1} \leq u \leq \mathbf{1}\}. \tag{3}$$

Note that $\mathbf{1} \in \mathbb{R}^2$ denotes a vector of ones. The controller prediction horizon is $N = 3$ steps and the quadratic cost function penalizes the states and inputs with matrices $Q = \alpha I$ and $R = I$. Assume there is no terminal cost ($P = 0$).

(a) Start by implementing the unconstrained MPC formulation (no state or input constraints) for a few different values of α. Assume an initial condition of $x(0) = [-1\ 0.5]^T$ and a simulation time of 10 steps. Determine the effect of α on the closed-loop system and find a value of α such that the closed-loop system is unstable. Provide example figures of a stable closed-loop trajectory and an unstable closed-loop trajectory along with their corresponding values of α. For each figure, include two subplots where the top subplot shows the state trajectories over time and the bottom subplot shows the input trajectories over time. Be sure to include legends to label the state and input trajectories.

(b) Using the unstable value of α, now add the state and input constraints. Is the resulting closed-loop system stable or unstable? Also note the feasibility of the optimization problem at each time step.

(c) Implement the constrained MPC formulation with the additional terminal constraint of $x_N = 0$. Provide a figure of the closed-loop state and input trajectories.

(d) Finally, find the region of attraction for this constrained MPC controller using both the projection and recursive algorithms discussed in class. Verify that both approaches give you the same set and provide a plot of one of these sets. Use this `tic` and `toc` commands in Matlab to determine the computation time for each approach and comment on the similarity or difference.