Exercise 5

Rendell Cale rendellc@stud.ntnu.no , mttk

Problem 1

a) Stable if all eigenvalues are less than 1 in magnitude.

$$0 = \begin{vmatrix} \lambda & 0 & 0 \\ 0 & \lambda & -1 \\ -0.1 & 0.79 & \lambda - 1.78 \end{vmatrix} = \lambda \begin{vmatrix} \lambda & -1 \\ 0.79 & \lambda - 1.78 \end{vmatrix}$$

$$= \lambda \left[\lambda(\lambda - 1.78) + 0.79 \right]$$

$$= \lambda \left(\lambda^2 - 1.78\lambda + 0.79 \right)$$

$$= \lambda (\lambda - 0.94)(\lambda - 0.84)$$

$$\Rightarrow \lambda_1 = 0, \ \lambda_2 = 0.94, \ \lambda_3 = 0.84$$

All $|\lambda_i| < 1$ so system is open loop stable.

b) x_t is 3×1 $\begin{bmatrix} \cdot \\ \cdot \\ \cdot \end{bmatrix}$

u_t is a scalar

$$f(z) = \sum_{t=0}^{N-1} y_{t+1}^2 + r u_t^2$$

$$= \frac{1}{2} \sum_{t=0}^{N-1} 2 y_{t+1}^T y_{t+1} + u_t^T 2 r u_t$$

$$= \frac{1}{2} \sum_{t=0}^{N-1} \left(\begin{bmatrix} 0 & 0 & 1 \end{bmatrix} x_{t+1} \right)^T 2 \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} x_{t+1} + u_t^T 2r u_t$$

$$= \frac{1}{2} \sum_{t=0}^{N-1} x_{t+1}^T 2 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} x_{t+1} + u_t^T 2r u_t$$

$$= \frac{1}{2} \sum_{t=0}^{N-1} x_{t+1}^T \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 2 \end{bmatrix}}_{Q} x_{t+1} + u_t^T \underbrace{2r}_{R} u_t$$

c) Q,R is positive semidefinite so objective function is convex

Constraints can be written as $Az = b$ which defines a convex set, so set is convex.

$\Rightarrow$ Problem is convex.

Depends on Q,R since if $Q < 0$, then $f(z)$ would not be convex!

d) With $z = [x_1, \dots, x_N, u_0, \dots, u_{N-1}]^T$

we can write $f(z)$ as

$$f(z) = \frac{1}{2}\sum_{t=0}^{N-1} x_{t+1}^T Q x_{t+1} + u_t^T R u_t$$

$$= \frac{1}{2}\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \\ u_0 \\ u_1 \\ \vdots \\ u_{N-1}\end{pmatrix}^T \underbrace{\begin{pmatrix} Q & & & & & & \\ & Q & & & & & \\ & & \ddots & & & & \\ & & & Q & & & \\ & & & & R & & \\ & & & & & R & \\ & & & & & & \ddots \\ & & & & & & & R \end{pmatrix}}_{G} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \\ u_0 \\ u_1 \\ \vdots \\ u_{N-1}\end{pmatrix}$$

$$= \frac{1}{2} z^T G z$$

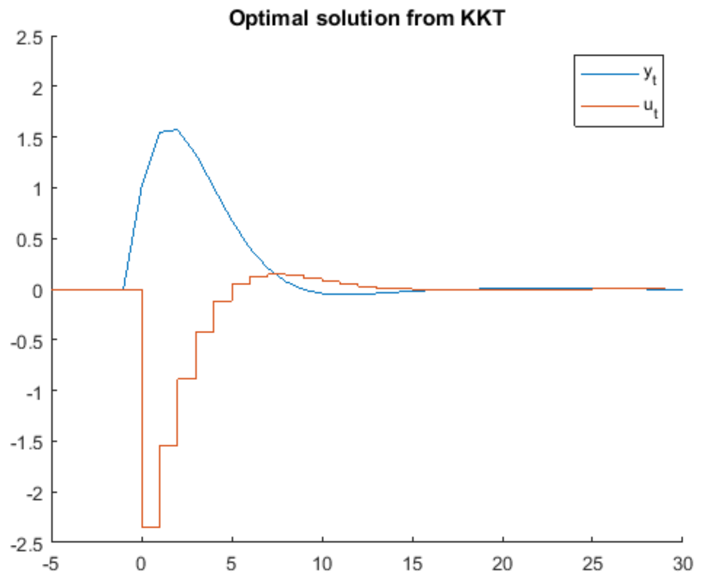
From $x_{t+1} = Ax_t + Bu_t$ we see that

$$\begin{aligned} x_1 \quad\quad\quad - Bu_0 &= Ax_0 \\ -Ax_1 + x_2 \quad\quad\quad - Bu_1 &= 0 \\ -Ax_2 + x_3 \quad\quad\quad - Bu_2 &= 0 \\ &\vdots \end{aligned}$$

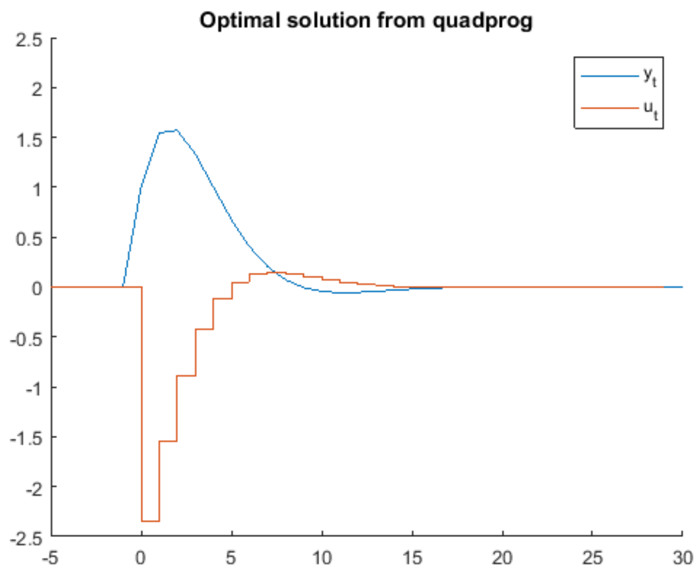
$$\Rightarrow A_{eq} = \begin{bmatrix} I & & & \cdots & -B & & & \\ -A & I & & \cdots & & -B & & \\ & -A & I & \ddots & & & -B & \\ & & & \ddots & & & & \ddots \end{bmatrix}, \quad b_{eq} = \begin{pmatrix} Ax_0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

KKT conditions:

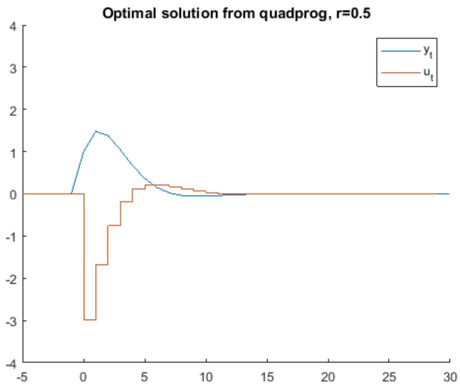
$$\begin{pmatrix} G & -A_{eq}^T \\ A_{eq} & 0 \end{pmatrix} \begin{pmatrix} z^* \\ \lambda^* \end{pmatrix} = \begin{pmatrix} 0 \\ b_{eq} \end{pmatrix}$$

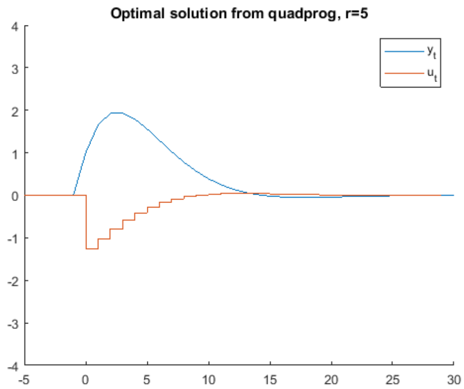


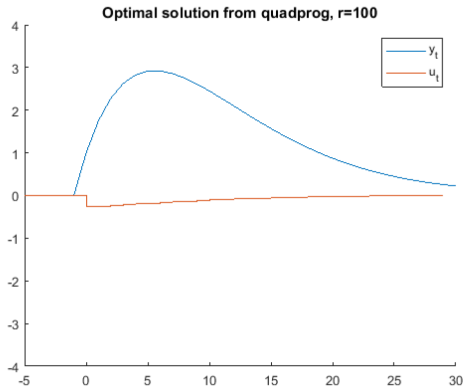
e)



Looks identical to the KKT solution!



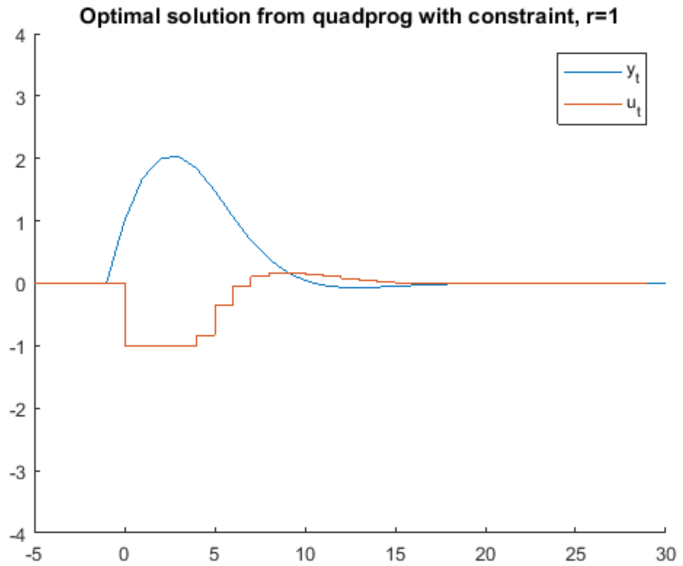




When r increases, we see that the optimal solution requires less use of action u_t.
This leads to slower convergence.

f) Want to formulate $-1 \leq u_t \leq 1$, $t \in [0, N-1]$
as a constraint on z.
Since $z = \begin{pmatrix} x_1^T & \cdots & x_N^T & u_0^T & \cdots & u_{N-1}^T \end{pmatrix}^T$ we require

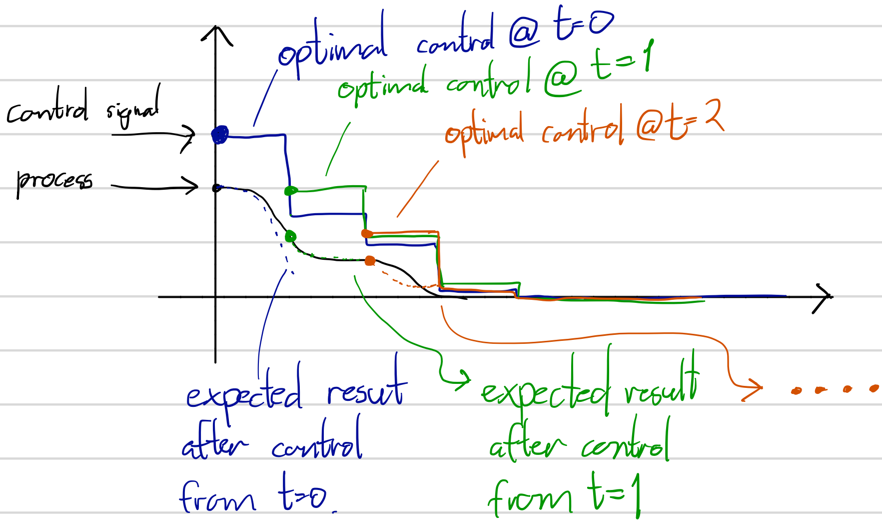
$$\begin{array}{r} N \cdot n_x = 3N \left\{ \begin{array}{c} \\ \\ \\ \end{array} \right. \\ N \cdot n_u = N \left\{ \begin{array}{c} \\ \\ \\ \end{array} \right. \end{array} \begin{pmatrix} -\infty \\ \vdots \\ -\infty \\ -1 \\ \vdots \\ -1 \end{pmatrix} \leq z \leq \begin{pmatrix} \infty \\ \vdots \\ \infty \\ 1 \\ \vdots \\ 1 \end{pmatrix} \begin{array}{l} \left. \begin{array}{c} \\ \\ \\ \end{array} \right\} 3N \\ \left. \begin{array}{c} \\ \\ \\ \end{array} \right\} N \end{array}$$



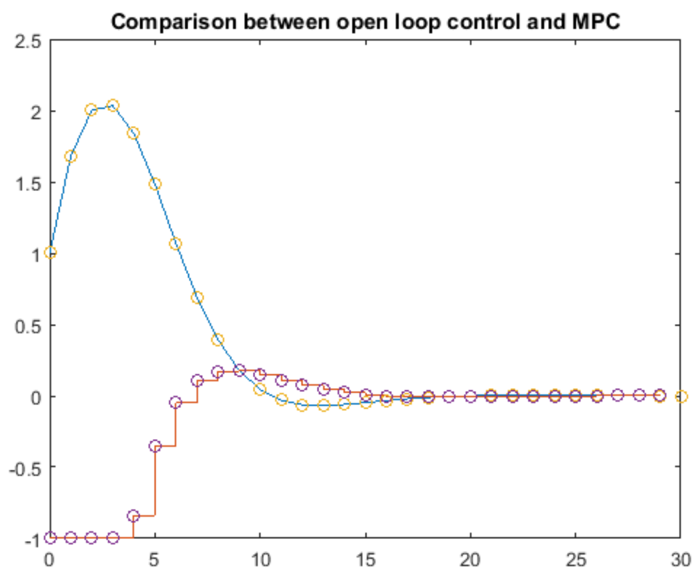
Quadprog used 5 iterations to converge.
Needs more iterations to converge since it has a more restrictive feasible set.

Problem 2

a) MPC is based on open loop optimal control, except instead of continuing to use the computed control signal u_t indefinately, we only use the first (few) "samples". After this we use feedback and recalculate the optimal openloop control.

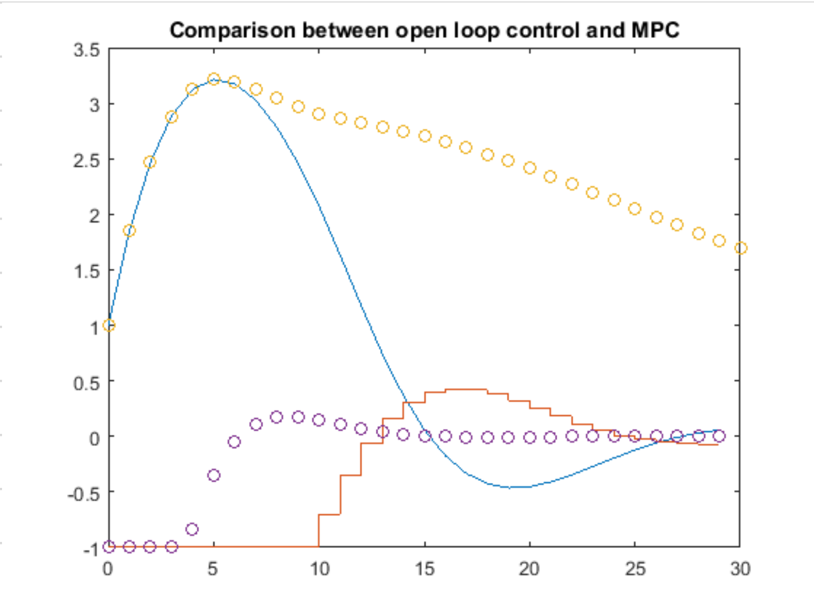


b) The scatterplot is the plot from open loop control.



The difference between open loop and closed loop in this case was very small.

c) Again the scatterplot is the result of open loop control.



In this case, the difference is striking. The open loop version has no way of accounting for modelling deviation so it starts decreasing u_t alot sooner than it should.

The closed loop controller is however able to counteract modelling deviations. The performance difference is quite apparent from the results.