1 Review and preliminaries

Exercise 1.1
Consider a plant with a model given by

$$Y(z) = G(z)U(z), \text{ where } G(z) = \frac{2}{z - 0.7} \tag{1.1}$$

is a discrete transfer function ($z-$transform of plant's impulse response $g(n)$). Design a receding horizon controller for driving this plant from initial condition $y(k) = 2$ to a constant set-point $s(k) = 3$. Assume the sampling interval $T_s = 3$ s and the prediction horizon $N = 2$. We have the following specific control requirements in this problem:

- The controller is required to drive the system output y such that it follows an 'exponential' reference trajectory $r(k)$ while approaching the set-point $s(k)$. Assume the time constant $T_{\text{ref}} = 9$ s for $r(k)$.

- The controller is not required to achieve coincidence between output prediction $\hat{y}(k+i|k)$ and $r(k+i|k)$ at each *prediction time step* 'i' inside the prediction phase. We instead impose a single *coincidence point* at the second time step (6 s into the future) i.e. we want to satisfy an equality constraint $\hat{y}(k+2|k) = r(k+2|k)$ at each *sampling time step* k.

- The controller is allowed to make only a single move ($u(k|k) = u(k+1|k)$) while optimizing the output trajectory inside the prediction phase. This implies that the control horizon $M = 1$.

We have the following data known at time step k for solving this control problem: The set-point is constant at value $s(k+i) = 3$, $\forall i$, the previous and current outputs are $y(k-1) = y(k) = 2$, the latest control input available in memory is $u(k-1) = 0.3$.

It is sufficient to find the first three optimal inputs $u(k|k)$, $u(k+1|k+1)$ and $u(k+2|k+2)$ (Note that the steady-gain of the model is $20/3$, so the input signal should converge to $3/(20/3) = 0.45$ if the control scheme is stable.) In addition, verify that the output $y(k+2)$ at sampling instant $k+2$, is not the same as the 2-step ahead prediction $\hat{y}(k+2|k)$ done at the sampling instant k, even for cases where we use a disturbance free perfect plant model.

Exercise 1.2
Consider a 'straight-line' reference trajectory with error evolution given by

$$\epsilon(k+i) = \begin{cases} \lambda(i)\epsilon(k), & \text{if } \; i\,T_s < T_{\text{ref}} \\ 0, & \text{otherwise} \end{cases} \tag{1.2}$$

where $\lambda(i) = \left(1 - \frac{iT_s}{T_{\text{ref}}}\right)$ is an error evolution profile (linear decay). Now repeat Exercise 1.1 using this profile for two cases:

a) Single coincidence point.

b) Two coincidence points.

Exercise 1.3

Are the functions $V(x)$ which appear below positive-definite?

a) Suppose $V(x) = 9x_1^2 + 25x_2^2 + 16x_3^2$. Find Q such that $V(x) = x^T Q x$.

b) Suppose $V(x) = (5x_1^2 + 2x_2^2 + x_3^2) + (100u_1^2 + 4u_2^2)$. Find Q and R such that $V(x) = x^T Q x + u^T R u$.

Exercise 1.4

The standard plant model used in the course is given by

$$x(k+1) = Ax(k) + Bu(k), \tag{1.3a}$$
$$y(k) = C_y x(k), \tag{1.3b}$$
$$z(k) = C_z x(k). \tag{1.3c}$$

Show how this model can can be transformed into a model that instead uses control moves Δu as input by introducing the new augmented state vector

$$\xi(k) = \begin{bmatrix} \Delta x(k) \\ y(k-1) \\ z(k-1) \end{bmatrix}. \tag{1.4}$$

Give the expression for new augmented state-space model along with all its matrices.

2 Receding horizon control

Exercise 2.1
Consider the first order system described by

$$y(k+1) = ay(k) + u(k).$$

You are supposed to work out the details for a simple MPC controller for this system. The prediction horizon is $N = 2$ and the control horizon is $M = 1$. The controller is based on minimization of the objective

$$V_2(y(k), u(k|k)) = (\hat{y}(k+1|k) - r(k))^2 + \alpha(\hat{y}(k+2|k) - r(k))^2$$

where α is a tuning parameter.

a) Design a receding horizon controller by solving the optimization problem. Give an expression for the control law. Use any standard assumptions that you need.

b) Determine the closed-loop poles in the two extreme cases with $\alpha = 0$ and $\alpha \to \infty$, respectively.

Exercise 2.2
Consider the first order system described by

$$y(k+1) = ay(k) + u(k-1).$$

A simple MPC controller for this system will be investigated. The controller is based on minimization of the objective

$$V_2(y(k), u(k|k)) = (\hat{y}(k+2|k) - r(k))^2 + \rho u^2(k|k)$$

Hence, the prediction horizon is $N = 2$ and the control horizon is $M = 1$, and ρ is a tuning parameter.

a) Design a receding horizon controller by solving the optimization problem. Give an expression for the control law. Use any standard assumptions that you need.

b) Determine the closed-loop poles in the two extreme cases with $\rho = 0$ and $\rho \to \infty$, respectively.

Exercise 2.3
Consider the system described by

$$y(k+1) = u(k) + bu(k-1)$$

You will investigate an MPC controller for this system that is based on minimization of the objective

$$V_2(u(k|k), u(k+1|k)) = (\hat{y}(k+1|k) - r(k))^2 + (\hat{y}(k+2|k) - r(k))^2$$

where $r(k)$ is the setpoint or reference signal, and both the prediction horizon and the control horizon are equal to 2.

a) Design a receding horizon controller by solving the optimization problem. Give an expression for the control law.

b) Determine under what conditions the closed-loop system is stable.

Exercise 2.4
Solve the following finite horizon optimal control problem using dynamic programming

$$\begin{aligned} \min \quad & \big((x(2)-1)^2 + \big(u(0)^2 + u(1)^2\big)\big) \\ \text{s.t.} \quad & x(k+1) = x(k) + u(k), \quad x(0) = 1. \end{aligned}$$

Exercise 2.5
Suppose that the cost function used in the predictive control formulation was changed to include the following penalty term

$$\sum_{i=0}^{M-1} \| u(k+i|k) - u_{\text{ref}}(k+i) \|^2_{S(i)}, \tag{2.1}$$

in addition to the terms penalizing tracking error and control moves, where M is the control horizon, u_{ref} denotes some prescribed future trajectory for the plant inputs, and $S(i)$ are time-varying penalty weight matrices.

a) Describe in detail the changes that would need to be made to the computations of predictions and the optimal solution.

b) Briefly list some reasons for and against making such a modification.

Exercise 2.6
Show that the optimal cost for the infinite horizon LQ controller is given by

$$V^*_\infty(x(0)) = x^T(0) P x(0),$$

where $x(0)$ is the initial state and P is the solution to the algebraic Riccati equation.

3 State estimation

Exercise 3.1

Detectability of Integrating Disturbance Models: The augmented system model used inside state estimator for estimating both system states and the unmeasured disturbances is given by:

$$\begin{bmatrix} x \\ d \end{bmatrix}^+ = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + w \tag{3.1}$$

$$y = \begin{bmatrix} C & C_d \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix} + v \tag{3.2}$$

a) Prove that the above augmented system is detectable if and only if the system (A, C) is detectable and rank

$$\operatorname{rank} \underbrace{\begin{bmatrix} I - A & -B_d \\ C & C_d \end{bmatrix}}_{=D_2} = n + n_d \tag{Condition 2}$$

b) Prove that the augmented system is detectable only if $n_d \leq p$ i.e. the maximal dimension n_d of the disturbance d in augmented system (3.1)-(3.2) — such that the augmented system is detectable — is equal to the number of measurements p

Exercise 3.2

Unconstrained Tracking Problem

a) For an unconstrained system, show that the following condition is sufficient for feasibility of the target problem for any z_{sp}.

$$\operatorname{rank} \begin{bmatrix} I - A & -B \\ HC & 0 \end{bmatrix} = n + n_c \tag{3.3}$$

where n is the number of states, n_c is the number of controlled variables, p is the number of measured outputs. The order of block matrix in (3.3) is $(n + n_c) \times (n + m)$.

b) Show that the condition (3.3) implies that the number of controlled variables without offset is less than or equal to the number of manipulated variables and the number of measurements, $n_c \leq m$ and $n_c \leq p$.

c) Show that (3.3) implies the rows of H are independent.

d) Does (3.3) imply that the rows of C are independent? If so, prove it, if not provide a counter-example?

e) By choosing H, how can one satisfy (3.3) if one has installed redundant sensors so several rows of C are identical?

Exercise 3.3

Kalman filter for first order system.

Consider the first order system (without control input)

$$\begin{aligned} x(k+1) &= x(k) \\ y(k) &= x(k) + v(k) \end{aligned}$$

Assume that the measurement noise v has standard deviation σ and that $x(0)$ has variance 0.5.

a) Determine the Kalman filter that estimates the constant state x from the noisy measurements.

b) Give expressions for both the estimate variance $P(k)$ and the Kalman gain $L(k)$. What happens when $k \to \infty$? Explanation?

c) Simulate the system with $x(0) = -2$ and the Kalman filter with $\hat{x}(0) = 0$, using $\sigma = 1$. Compare the Kalman filter estimate with an estimator using a constant L. Try $L = 0.01$ and $L = 0.05$.

Exercise 3.4

Kalman filter in transfer function form.

Consider the first order system (without control input) given by the difference equation

$$y(k) + ay(k-1) = e(k) + ce(k-1)$$

where the noise e has standard deviation σ. Assume that $|c| < 1$.

a) Determine a state space model for the system.

b) Determine the steady state Kalman filter for the system, and put it in transfer function form. Note that in this case, the process noise and the measurement noise will be correlated. Denoting the cross correlation S, the time-varying Kalman filter equations are modified slightly; in prediction form, they are given by

$$\begin{aligned}\hat{x}(k+1|k) &= A\hat{x}(k|k-1) + Bu(k) \\ &\quad + (AP(k)C^T + S)[CP(k)C^T + R]^{-1}(y(k) - C\hat{x}(k|k-1)) \\ P(k+1) &= AP(k)A^T + Q - (AP(k)C^T + S)[CP(k)C^T + R]^{-1}(CP(k)A^T + S)\end{aligned}$$

c) Determine an expression for the steady-state, one-step ahead prediction of the output.

Exercise 3.5

Pole of a Kalman filter.

Consider the first order system (without control input)

$$\begin{aligned}x(k+1) &= 0.5x(k) + v(k) \\ y(k) &= x(k) + e(k)\end{aligned}$$

where v and e are uncorrelated white noise with variances Q and R. Further, $x(0)$ is normally distributed with zero mean and standard deviation ρ.

a) Determine the Kalman filter for the system.

b) What is the Kalman filter in steady state? Compute the pole of the steady-state filter and compare with pole of the system.

4 Optimization and convexity

Exercise 4.1
Solve the following optimization problem

$$\begin{aligned} \text{minimize} \;\; & V(x) = \frac{1}{2}(x_1^2 + x_2^2) \;\; (\text{w.r.t. } \; x), \\ \text{Subject to} \;\; & x_1 - 1 \geq 0 \end{aligned} \tag{4.1}$$

Exercise 4.2
Softening the Constraints with quadratic penalty on slack variables:
Consider the following standard QP problem with *hard* linear inequality constraints on optimization variable θ.

$$\begin{aligned} \text{minimize} \;\; & V(\theta) = \frac{1}{2}\theta^T \Phi \theta + \phi^T \theta \\ \text{Subject to} \;\; & \Omega\theta \leq \omega \end{aligned} \tag{4.2}$$

In some practical applications, we can relax the hard constraints by introducing slack variables ϵ (this is called *constraint softening*) and rewrite the optimization problem as follows

$$\begin{aligned} \text{minimize} \;\; & V(\theta) = \frac{1}{2}\theta^T \Phi \theta + \phi^T \theta + \rho \|\epsilon\|_2^2 \\ \text{Subject to} \;\; & \Omega\theta \leq \omega + \epsilon \\ & \epsilon \geq 0 \end{aligned} \tag{4.3}$$

a) Show that the new optimization problem (4.3) is also a standard QP problem.

b) Modify the problem (4.3) to deal with the case that a subset of the constraints, $\Omega_1\theta \leq \omega_1$, is to be *hard*, while the remaining constraints, $\Omega_2\theta \leq \omega_2$, is to be *soft*. Now verify that the problem is still a QP problem after this reformulation.

Exercise 4.3
Softening the constraints with 1-norm or ∞-norm penalties on slack variables:
There are various problem formulation alternatives for constraint softening. For example, consider the following reformulation of optimization problem (4.3)

$$\begin{aligned} \text{minimize} \;\; & V(\theta) = \frac{1}{2}\theta^T \Phi \theta + \phi^T \theta + \rho \|\epsilon\|_1 = \frac{1}{2}\theta^T \Phi \theta + \phi^T \theta + \rho \sum_j \epsilon_j \\ \text{Subject to} \;\; & \Omega\theta \leq \omega + \epsilon \\ & \epsilon \geq 0 \end{aligned} \tag{4.4}$$

where, instead of quadratic, we have used 1-norm penalty on slack variables here. Note that ϵ is a vector of slack variables here.

Similarly, we also can achieve constraint softening using another problem formulation

$$\begin{aligned} \text{minimize} \;\; & V(\theta) = \frac{1}{2}\theta^T \Phi \theta + \phi^T \theta + \rho \|\epsilon\|_\infty = \frac{1}{2}\theta^T \Phi \theta + \phi^T \theta + \rho \max_j \epsilon_j \\ \text{Subject to} \;\; & \Omega\theta \leq \omega + \epsilon \\ & \epsilon \geq 0 \end{aligned} \tag{4.5}$$

where ϵ is a vector of slack variables. Note that in this problem, we have used ∞-norm penalty on slack variables.

There is yet another problem formulation to relax the constraints

$$\begin{aligned} \text{minimize} \;\; & V(\theta) = \frac{1}{2}\theta^T \Phi \theta + \phi^T \theta + \rho\epsilon \\ \text{Subject to} \;\; & \Omega\theta \leq \omega + \epsilon \\ & \epsilon \geq 0 \end{aligned} \tag{4.6}$$

where ϵ is a scalar slack variable here. Show that all three formulations are on standard QP form.

Exercise 4.4
Consider the nonlinear constrained optimization problem

$$\begin{aligned} \min_{x} \;\; & \frac{1}{2}x^T x \\ \text{s.t.} \;\; & x^T x \geq N \end{aligned}$$

with $x \in \mathbb{R}^n$. What is the solution? Is it a KKT point? Is it regular? Does it fulfill the second order sufficient conditions for an optimal solution? Justify and explain!

Exercise 4.5
A classmate of yours wants to solve the following problem:

$$\begin{aligned} \min_{x,y} \;\; & x + y \\ \text{s.t.} \;\; & x + y = ax^2 + by^2 + c \end{aligned}$$

with $a > 0$, $b > 0$. Combining the constraint and the objective, your friend concludes that an equivalent problem formulation is

$$\min_{x,y} \;\; ax^2 + by^2 + c$$

which gives the trivial solution $x = y = 0$. What is wrong?

Exercise 4.6
In the lecture notes, the following is stated: We can conclude that for any local optimum x^*, we must have $\nabla f(x^*)^T d \geq 0$ for any d such that $\nabla h(x^*)^T d = 0$, provided x^* is regular.
From this, show how the result in the lecture notes can be deduced.
Hint: Partition $\nabla f(x^*)$ into two orthogonal components.

5 Stability

Exercise 5.1
Exercise on effect of Prediction Horizons on Stability and Speed of Response

Consider the plant consisting of two delays in series:

$$x(k+1) = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} x(k) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(k) \tag{5.1}$$

The cost function is given by

$$V(k) = \sum_{i=1}^{N} \hat{x}^T(k+i|k) Q \hat{x}(k+i|k) \tag{5.2}$$

where $Q = \begin{bmatrix} 1 & 2 \\ 2 & 6 \end{bmatrix}$.

a) Suppose prediction horizon $N = 1$ and control horizon $M = 1$. Show that the predictive control law is $u^*(k) = -2x_1(k)$ and hence that the closed loop system is unstable.

b) Now suppose $N = 2$, with the cost being

$$V(k) = \hat{x}^T(k+1|k) \begin{bmatrix} 1 & 2 \\ 2 & 6 \end{bmatrix} \hat{x}(k+1|k) + \hat{x}^T(k+2|k) \begin{bmatrix} 1 & 2 \\ 2 & 6 \end{bmatrix} \hat{x}(k+2|k) \tag{5.3}$$

and keep $M = 1$, so that only $u(k)$ is to be optimized, with the assumption that $u(k+1) = u(k)$. Show that the predictive control law is $u^*(k) = -\frac{1}{6}x_1(k)$ and hence that the closed loop is stable.

c) Now let $M = 2$, so that $u(k)$ and $u(k+1)$ have to be optimized. By setting both derivatives $\frac{\partial V}{\partial u(k)}$ and $\frac{\partial V}{\partial u(k+1)}$ to zero (or $\nabla_u V = 0$) show that the predictive control law is $u^*(k) = -\frac{2}{3}x_1(k)$, and hence that the closed loop is stable.

d) From Cases 'b' and 'c' which one will give faster regulation i.e. faster convergence to origin from some non-zero initial condition?

e) Simulate the closed-loop behaviors for all these cases using MATLAB and confirm that response for Case b is faster than that for Case a.

Exercise 5.2
Show that the infinite horizon LQ controller gives a stable closed loop.
Hint: Use the Lyapunov function $V = x^T Px$, where P is the solution of the algebraic Riccati equation.

6 Beyond linear MPC

Exercise 6.1

Consider the nonlinear constrained optimization problem

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s.t.} \quad & h(x) = 0 \end{aligned}$$

Prove that, for a suitably defined matrix H, the primal Newton direction for this problem is provided by solving the following QP:

$$\begin{aligned} \min_{\Delta x} \quad & \frac{1}{2}\Delta x^T H \Delta x + \nabla f(x)^T \Delta x \\ \text{s.t.} \quad & h(x) + \nabla h(x)^T \Delta x = 0 \end{aligned}$$

and the Lagrange multipliers of the latter problem provide the update for the Lagrange multipliers of the original problem.

7 Solutions

Solution 1.1

Notation and Nomenclature Let's first define some important variables involved in receding horizon control strategy:

a) $y(t)$ is actual output of plant at any time t.

b) $s(t)$ is the set-point trajectory at any time t. The set-point trajectory is the desired output trajectory that we want the plant to follow.

c) T_s is the sampling Interval

d) k is current discrete time step

e) $r(t|k)$ is the reference trajectory during prediction phase. This starts at $y(k)$ and defines an ideal trajectory along which the plant should return to the set-point trajectory $s(t)$ after, for example, a disturbance occurs.

f) $\hat{y}(t|k)$ is the predicted output trajectory of a plant for any time '$t > k$', computed using plant model and the information (measured output and input etc) given up to time 'k'

g) $y_f(t|k)$ is the predicted free response of the plant model i.e. the response of plant model that would be obtained if the future input trajectory remained at the last input $u(k-1)$.

h) T_{ref} = time constant of exponential along which the system model response would return to $s(t)$. It defines the slope of $r(t|k)$ and hence the speed of response during prediction phase.

i) $\epsilon(k) = s(k) - y(k)$ is an initial output error.

j) $\epsilon(k+i) = e^{(-iT_s/T_{ref})}\epsilon(k) = \lambda^i \epsilon(k)$ where $\lambda = e^{-Ts/T_{ref}}$.

k) $r(k+i|k) = s(k+i) - \epsilon(k+i) = s(k+i) - \lambda^i \epsilon(k)$ defines the reference trajectory at any time step during prediction phase. Note that during calculations we have to find values of reference trajectory $r(t|k)$ at as many time steps as there are coincidence points. Generally $r(k+j+i|k+j) = s(k+j+i) - \epsilon(k+j+i) = s(k+j+i) - \lambda^i \epsilon(k+j)$ for optimization at any time step '$k+j$'.

l) $\theta(k)$ is the step response of the model to a unit step input, k steps after the unit step is applied.

m) $u(k|k)$ is the computed control input to make the predicted output $\hat{y}(t|k)$ follow the reference trajectory $r(t|k)$.

n) $\Delta u(k|k) = u(k|k) - u(k-1)$ is the first *control move* at time step 'k' i.e. change in control signal from the last input $u(k-1)$ to the next input $u(k|k)$. Similarly $\Delta u(k+1|k) = u(k+1|k) - u(k|k)$, $\Delta u(k+2|k) = u(k+2|k) - u(k+1|k)$. Generally for any time step 'k' (in real-world time axis), the control move is given by $\Delta u(k+i|k) = u(k+i|k) - u(k+i-1|k)$.

o) $\hat{y}(k+i|k) = y_f(k+i|k) + \theta(i)\Delta u(k|k)$

p) $P_i, \forall i = \{1, \cdots, n_c\}$ is the relative discrete time-step for the i^{th} coincidence (point) between $\hat{y}(t|k)$ and $r(t|k)$ at $t = k + P_i$ where $P_i \leq N$. Note there are n_c coincidence points.

q) $\boldsymbol{y_f}$ is a free response vector containing value of free response $y_f(t|k)$ at each coincidence point $P_1, P_2, \cdots, P_{n_c}$.

r) $\boldsymbol{r}$ is a reference trajectory vector containing value of reference trajectory $r(t|k)$ at each coincidence point.

s) Θ is step response vector containing value of step responses for all coincidence points.

t) N is prediction horizon.

u) M is control horizon. It sets the number of sampling intervals over which the control variables are to be optimized.It means that input is allowed to change it's value over next M steps during prediction phase, for example, so that we have to choose first M optimal values of a control signal $u(k+i|k)$: $u(k|k), u(k+1|k), \cdots, u(k+M-1|k)$ and after M control moves, control signal remains constant i.e. $u(k+M-1|k) = u(k+M|k) = \cdots = u(k+N-1|k)$ if $M < N$.

v) H_w is the window horizon. It controls when the cost function becomes active i.e. this parameters tells when to start penalizing the control variable and controlled output variable. This may be required because it is not necessary that we start penalizing the deviation of controlled output from reference trajectory immediately.

w) The objective is to equalize predicted response $\hat{y}(t|k)$ to the reference trajectory $r(t|k)$ at each conincidence point $t = \{P_1, \cdots, P_c\}$ i.e. $\hat{y}(k+P_i|k) = r(k+P_i|k)$

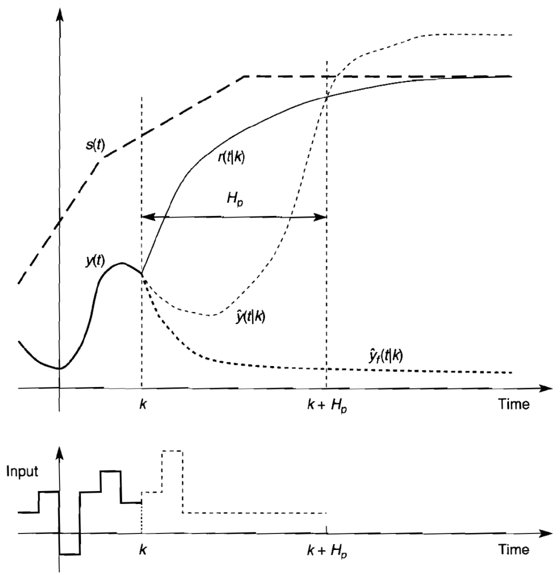


Figure 1: Predictive Control Basic Idea and definition of signals

Assumptions:

Note in the following development, we will assume that the model output $\hat{y}(k|k)$ and the actual plant output $y(k)$ are same up to time k i.e. when prediction phase starts both plant and model output are same. Also we assume that $M = 1$ which means the input remains constant over the whole prediction horizon i.e. $u(k|k) = u(k+1|k) = \cdots = u(k+N-1|k)$. It means there is only one parameter to determine. Moreover here we have only one coincidence point at $P_1 = N$ which means there is only one equation to be satisfied — $\hat{y}(k+N|k) = r(k+N|k)$— and hence we would have a unique solution. Also note that future control inputs are denoted

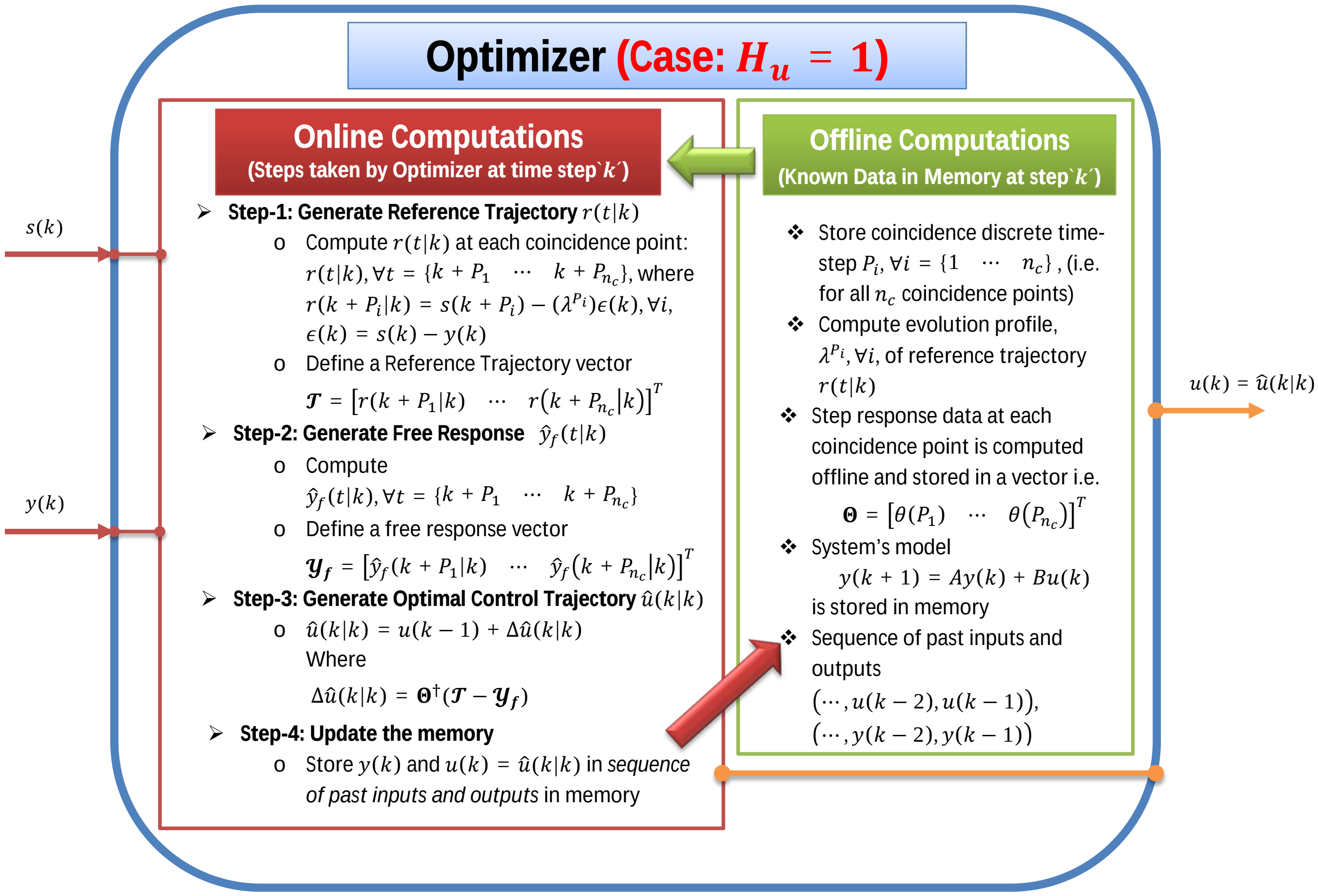


Figure 2: Block diagram of optimizer inside receding horizon controller (RHC). There are two computational phases: Offline and Online. In the offline phase all the variables shown in the green box are pre-computed once and stored inside memory and reused at each time-step. In the online phase, all variables shown in red box are computed again at each time step k using already known information stored in memory and the current plant measurement $y(k)$ and the current set-point $s(k)$. Note that, during online-phase, after doing all computations at each time step the memory inside green box is updated with new information (in order to make it available at next time-step). Note that for the case $M > 1$, all the steps will be similar to case of $M = 1$. The only difference is that now the optimal control will be a vector $\boldsymbol{\Delta u}$ instead of Δu and the step response vector Θ will be a matrix, (see Maceijowski equations (1.26) and (1.27) for details).

$u(k+i|k)$ to stress that all computations are carried out at time k and that future controls are only candidate control actions.

Given Data:
$s(t) = 3$, $T_{ref} = 9\,\text{sec}$, $T_s = 3\,\text{sec}$, $N = 2$, $M = 1$, $P_1 = 2$ (i.e. 6 sec in predicted future from current time-step) and the previous and current outputs are $y(k-1) = y(k) = 2$ and the latest control is $u(k-1) = 0.3$.

Computation of Optimal Control Input $(u(k|k))$, at step k:

Off-line Phase
There is only one coincidence point (i.e. $n_c = 1$) at $P_1 = 2$. Now we will compute step response and evolution profile at this coincidence point:

Computation of Step Response $\theta(2)$:

We need $\theta(P_1) = \theta(2)$ that is the step response 2 time steps after a unit step input is applied. We can get this from (7.3) by assuming $u(k) = u(k+1) = 1$ and zero initial conditions $y(k) = y(k-1) = 0$. Assuming to start at $k = 0$ and using $\theta(0) = y(0)$ we get following unit step responses from model at next time steps $k = 1$ and $k = 2$:

$$\theta(1) = y(1) = 0.7y(0) + 2u(0) = 0.7 \times 0 + 2 \times 1 = 2 \tag{7.1}$$
$$\theta(2) = y(2) = 0.7\theta(1) + 2u(1) = 0.7 \times 2 + 2 \times 1 = 3.4 \tag{7.2}$$

Note that $\Theta = \theta(2)$ here.

Computation of Evolution Profile $\lambda^{P_i}, \forall i$:

$$\lambda^{P_1} = e^{(-P_1 T_s / T_{ref})} \Rightarrow \lambda^2 = e^{(-2T_s/T_{ref})} = (0.7165)^2$$

Online Phase

Step 1–*Computation of Reference Trajectory at Coincidence Point P_1 (i.e. at $t = k+2$):*

Using above definitions, we have $\epsilon(k) = s(k) - y(k) = 3 - 2 = 1$, and $\lambda = 0.7165$. Hence $r(k+2|k) = s(k+2) - \lambda^2\,\epsilon(k) = 3 - 0.7165^2 \times 1 = 2.487$. Note that $\mathcal{T} = r(k+2|k)$ here

Step 2–*Computation of Free Response $y_f(t|k)$ at Coincidence Point P_1 (i.e. at $t = k+2$):*

To get the free responses $y_f(t|k)$ at $t = k+2$, we put the given transfer function model of plant into difference equation form:

$$y(k) = 0.7y(k-1) + 2u(k-1) \tag{7.3}$$

Since in computation of free response $y_f(t|k)$ it is assumed that the input is held at the last value which was applied to plant so it means $u(k+1) = u(k) = u(k-1) = 0.3$. Also note that $y_f(k|k) = y(k) = 2$. Now the evolution of free response for two time steps can be obtained by doing two recursions of (7.3) to get:

$$y_f(k+1|k) = 0.7y_f(k|k) + 2u(k) = 0.7 \times 2 + 2 \times 0.3 = 2.0 \tag{7.4}$$
$$y_f(k+2|k) = 0.7y_f(k+1|k) + 2u(k+1) = 0.7 \times 2 + 2 \times 0.3 = 2.0 \tag{7.5}$$

Note that $\boldsymbol{y_f} = y_f(k+2|k)$ here.

Step 3–*Computation of Optimal Control Input $u(k|k)$:*

We now have everything required to compute the optimal control input $u(k|k)$. Note that in this exercise Θ is scalar, thus $\Theta^{\dagger} = \Theta^{-1} = \frac{1}{\theta(2)}$. Thus:

$$\Delta u(k|k) = \frac{r(k+P_1|k) - y_f(k+P_1|k)}{\theta(P_1)} \tag{7.6}$$

$$\Delta u(k|k) = \frac{r(k+2|k) - y_f(k+2|k)}{\theta(2)} \tag{7.7}$$

$$\Delta u(k|k) = \frac{2.487 - 2.0}{3.4} = 0.1432 \tag{7.8}$$

$$u(k|k) = u(k-1) + \Delta u(k|k) = 0.4432 \tag{7.9}$$

This is the input signal applied to the plant: $u(k) = u(k|k) = 0.4432$. If our model of the plant is perfect, and there are no disturbances, then this would result in the next plant output value being $y(k+1) = 0.7 \times 2 + 2 \times 0.4432 = 2.2864$.

Computation of Optimal Control Input $(u(k+1|k+1))$, at step $k+1$:

Step 1–*Computation of Reference Trajectory at Coincidence Point P_1 (i.e. at $t = (k+1)+2$):*

We will repeat the similar three steps for computing optimal control input to be applied to plant at time step $k+1$. We have $\epsilon(k+1) = s(k+1) - y(k+1) = 3 - 2.2864 = 0.7136$, and $\lambda = 0.7165$. Hence $r(k+3|k+1) = s(k+3) - \lambda^2\,\epsilon(k+1) = 3 - 0.7165^2 \times 0.7136 = 2.6337$.

Step 2–*Computation of Free Response $y_f(t|k+1)$ at Coincidence Point P_1 (i.e. at $t = (k+1)+2$):*

To get the free responses $y_f(t|k+1)$ at $t = k+3$, we will again use the plant model (7.3). Since in computation of free response $y_f(t|k+1)$ it is assumed that the input is held at the last value which was applied to plant so it means $u(k+2) = u(k+1) = u(k) = 0.4432$. Also note that $y_f(k+1|k+1) = y(k+1) = 2.2864$. Now the evolution of free response for two time steps ahead can be obtained by doing two recursions of (7.3) to get:

$$y_f(k+2|k+1) = 0.7y_f(k+1|k+1) + 2u(k+1) = 0.7 \times 2.2864 + 2 \times 0.4432 = 2.4869 \tag{7.10}$$

$$y_f(k+3|k+1) = 0.7y_f(k+2|k+1) + 2u(k+2) = 0.7 \times 2.4869 + 2 \times 0.4432 = 2.6272 \tag{7.11}$$

Step 3–*Computation of Optimal Control Input $u(k+1|k+1)$:*

We now have everything to compute the optimal control input $u(k|k)$:

$$\Delta u(k+1|k+1) = \frac{r(k+1+P_1|k+1) - y_f(k+1+P_1|k+1)}{\theta(P_1)} \tag{7.12}$$

$$\Delta u(k+1|k+1) = \frac{r(k+3|k+1) - y_f(k+3|k+1)}{\theta(2)} \tag{7.13}$$

$$\Delta u(k+1|k+1) = \frac{2.6337 - 2.6272}{3.4} = 0.0019 \tag{7.14}$$

$$u(k+1|k+1) = u(k) + \Delta u(k+1|k+1) = 0.4451 \tag{7.15}$$

This is the input signal that will be applied to the plant at time step $k+1$: $u(k+1) = u(k+1|k+1) = 0.4451$. If our model of the plant is perfect, and there are no disturbances, then this would result in the next plant output value being $y(k+2) = 0.7y(k+1) + 2u(k+1) =$

$0.7 \times 2.2864 + 2 \times 0.4451 = 2.4907$.

Computation of Optimal Control Input $(u(k+2|k+2))$**, at step** $k+2$**:**

Similarly we will repeat the same three steps as above for computing optimal control input to be applied to plant at time step $k+2$. In short we have following results:

$$\epsilon(k+2) = s(k+2) - y(k+2) = 3 - 2.4907 = 0.5093 \tag{7.16}$$

$$r(k+4|k+2) = s(k+4) - \lambda^2 \epsilon(k+2) = 3 - 0.7165^2 \times 0.5093 = 2.7385 \tag{7.17}$$

$$y_f(k+3|k+2) = 0.7y_f(k+2|k+2) + 2u(k+2) = 0.7 \times 2.4907 + 2 \times 0.4451 = 2.6337 \tag{7.18}$$

$$y_f(k+4|k+2) = 0.7y_f(k+3|k+2) + 2u(k+3) = 0.7 \times 2.6337 + 2 \times 0.4451 = 2.7338 \tag{7.19}$$

$$\Delta u(k+2|k+2) = \frac{r(k+2+P_1|k+2) - y_f(k+2+P_1|k+2)}{\theta(P_1)} \tag{7.20}$$

$$\Delta u(k+2|k+2) = \frac{r(k+4|k+2) - y_f(k+4|k+2)}{\theta(2)} \tag{7.21}$$

$$\Delta u(k+2|k+2) = \frac{2.7385 - 2.7338}{3.4} = 0.00138 \tag{7.22}$$

$$u(k+2|k+2) = u(k+1) + \Delta u(k+2|k+2) = 0.4469 \tag{7.23}$$

This is the input signal that will be applied to the plant at time step $k+2$: $u(k+2) = u(k+2|k+2) = 0.4469$. If our model of the plant is perfect, and there are no disturbances, then this would result in the next plant output value being $y(k+3) = 0.7y(k+2) + 2u(k+2) = 0.7 \times 2.4907 + 2 \times 0.4469 = 2.6373$.
Verification: $y(k+2) = 2.4907 \neq \hat{y}(k+2|k) = y_f(k+2|k) + \theta(2)\Delta u(k|k) = 2.0 + 3.4 \times 0.1432 = 2.487$

Remark 7.1. *We know that in receding horizon strategy we want to achieve the coincidence of predicted response* $\hat{y}$ *with the reference trajectory* r *at all coincidence points. Since in this problem we had only one coincidence point and also one free variable* u *(i.e. one degree of freedom) and hence we only have to satisfy one constraint* $\hat{y} = r$ *at coincidence point* $P_1 = N$*, so we can find such* u *which satisfies this constraint exactly. We get exact solution without any approximation!*

Solution 1.2

Part a: Single Coincidence Point.

Given Data:

$s(t) = 3$, $T_{\text{ref}} = 9\,\text{sec}$, $T_s = 3\,\text{sec}$, $N = 2$, $M = 1$, $P_1 = 2$ (i.e. coincidence at 6 sec in predicted future from current time-step) and the previous and current outputs are $y(k-1) = y(k) = 2$ and the latest control is $u(k-1) = 0.3$. The system model is given. For later use, we put the given transfer function model of plant into difference equation form:

$$y(k) = 0.7y(k-1) + 2u(k-1) \tag{7.24}$$

We need $\theta(2)$, the step response 2 times steps after a unit step input is applied. We can get this from (7.24) by assuming $u(k) = u(k+1) = 1$ and zero initial conditions $y(k) = y(k-1) = 0$.

Assuming to start at $k = 0$ and using $\theta(0) = y(0)$ we get following unit step responses from model at next time steps $k = 1$ and $k = 2$:

$$\theta(1) = y(1) = 0.7y(0) + 2u(0) = 0.7 \times 0 + 2 \times 1 = 2 \tag{7.25}$$
$$\theta(2) = y(2) = 0.7\theta(1) + 2u(1) = 0.7 \times 2 + 2 \times 1 = 3.4 \tag{7.26}$$

Here we have only one coincidence point at $P_1 = 2$. Thus the evolution profile of reference trajectory is given by:

$$\lambda^{P_1} = \left(1 - \frac{P_1 T_s}{T_{\text{ref}}}\right) \Rightarrow \lambda^2 = \left(1 - \frac{2T_s}{T_{\text{ref}}}\right) = \frac{1}{3}$$

Using above definitions, we have $\epsilon(k) = s(k) - y(k) = 3 - 2 = 1$, and $r(k+i|k) = s(k+i) - \epsilon(k+i)$.

Hence $r(k+2|k) = s(k+2) - \lambda^2 \epsilon(k) = 3 - \frac{1}{3} \times 1 = 2.6667$.

To get the free responses $y_f(t|k)$ at $t = k + 2$, we use system model (7.24). Since in computation of free response $y_f(t|k)$ it is assumed that the input is held at the last value which was applied to plant so it means $u(k+1) = u(k) = u(k-1) = 0.3$. Also note that $y_f(k|k) = y(k) = 2$. Now the evolution of free response for two time steps can be obtained by applying (7.24) twice iteratively to get:

$$y_f(k+1|k) = 0.7y_f(k|k) + 2u(k) = 0.7 \times 2 + 2 \times 0.3 = 2.0 \tag{7.27}$$
$$y_f(k+2|k) = 0.7y_f(k+1|k) + 2u(k+1) = 0.7 \times 2 + 2 \times 0.3 = 2.0 \tag{7.28}$$

We now have everything required to compute the optimal control input $u(k|k)$, using:

$$\Delta u(k|k) = \frac{r(k+N|k) - y_f(k+N|k)}{\theta(N)} \tag{7.29}$$
$$\Delta u(k|k) = \frac{r(k+2|k) - y_f(k+2|k)}{\theta(2)} \tag{7.30}$$
$$\Delta u(k|k) = \frac{2.6667 - 2.0}{3.4} = 0.1961 \tag{7.31}$$
$$u(k|k) = u(k-1) + \Delta u(k|k) = 0.4961 \tag{7.32}$$

This is the input signal applied to the plant: $u(k) = u(k|k) = 0.4961$. If our model of the plant is perfect, and there are no disturbances, then this would result in the next plant output value being $y(k+1) = 0.7 \times 2 + 2 \times 0.4961 = 2.3922$.

Part b: Two Coincidence Points. Given Data:
$s(t) = 3$, $T_{\text{ref}} = 9\,\text{sec}$, $T_s = 3\,\text{sec}$, $N = 2$, $M = 1$, $P_1 = 1$, $P_2 = N = 2$ (i.e. coincidence at two points one at 3 sec and another at 6 sec in predicted future from current time-step) and the previous and current outputs are $y(k-1) = y(k) = 2$ and the latest control is $u(k-1) = 0.3$.

We need $\theta(1)$ and $\theta(2)$ which are, respectively, the step responses one-step and two-steps after a unit step input is applied. These are given by (7.26)-(7.26)

This time we have two coincidence points: $P_1 = 1, P_2 = N = 2$. Thus the evolution profile is given by

$$\lambda^{P_1} = \left(1 - \frac{P_1 T_s}{T_{\text{ref}}}\right) \Rightarrow \lambda = \left(1 - \frac{T_s}{T_{\text{ref}}}\right) = \left(1 - \frac{1 \times 3}{9}\right) = \frac{2}{3}$$

$$\lambda^{P_2} = \left(1 - \frac{P_2 T_s}{T_{\text{ref}}}\right) \Rightarrow \lambda^2 = \left(1 - \frac{2T_s}{T_{\text{ref}}}\right) = \left(1 - \frac{2 \times 3}{9}\right) = \frac{1}{3}$$

The value of $\epsilon(k) = s(k) - y(k) = 3 - 2 = 1$. Now we need to find value of reference trajectory $r(k + P_i|k) = s(k + P_i) - \epsilon(k + P_i)$ at two coincidence points:

$$r(k+1|k) = s(k+1) - \lambda\epsilon(k) = 3 - \frac{2}{3} \times 1 = 2.3333 \tag{7.33}$$

$$r(k+2|k) = s(k+2) - \lambda^2\epsilon(k) = 3 - \frac{1}{3} \times 1 = 2.6667. \tag{7.34}$$

The free responses $y_f(t|k)$ at $t = k + 1$ and $t = k + 2$ are given by (7.27)-(7.28). We now have everything to define the matrices:

$$\boldsymbol{r} = \begin{bmatrix} r(k+1|k) \\ r(k+2|k) \end{bmatrix} = \begin{bmatrix} 2.3333 \\ 2.6667 \end{bmatrix} \tag{7.35}$$

$$\Theta = \begin{bmatrix} \theta(1) \\ \theta(2) \end{bmatrix} = \begin{bmatrix} 2 \\ 3.4 \end{bmatrix} \tag{7.36}$$

$$\boldsymbol{y_f} = \begin{bmatrix} y_f(k+1|k) \\ y_f(k+2|k) \end{bmatrix} = \begin{bmatrix} 2.0 \\ 2.0 \end{bmatrix} \tag{7.37}$$

$$\tag{7.38}$$

We now have everything required to compute the optimal control input $u(k|k)$ in least square sense using:

$$\Delta u(k|k) = \Theta \backslash (\mathcal{T} - \mathcal{Y}_f) = \Theta^\dagger\,(\mathcal{T} - \mathcal{Y}_f) = (\Theta^T\,\Theta)^{-1}\,\Theta^T\,(\mathcal{T} - \mathcal{Y}_f) \tag{7.39}$$

$$\Delta u(k|k) = 0.1885 \tag{7.40}$$

$$u(k|k) = u(k-1) + \Delta u(k|k) = 0.3 + 0.1885 = 0.4885 \tag{7.41}$$

This is the input signal applied to the plant: $u(k) = u(k|k) = 0.4885$. If our model of the plant is perfect, and there are no disturbances, then this would result in the next plant output value being $y(k + 1) = 0.7 \times 2 + 2 \times 0.4885 = 2.3770$.

Remark 7.2. *Note that $\Theta^\dagger = (\Theta^T\,\Theta)^{-1}\,\Theta^T$ is called the pseudo-inverse and it exists only if the matrix Θ is $1 - 1$ i.e. it has full column rank (independent columns) and hence Θ is left invertible ($\Theta^\dagger\,\Theta = I_v$) where I_v is the $v \times v$ identity matrix with v equal to number of columns in Θ.*

Solution 1.3

a) $Q = \text{diag}(9, 25, 16)$.

b) $Q = \text{diag}(5, 2, 1)$ and $R = \text{diag}(100, 4)$.

Both functions are positive definite, here seen from the fact that the diagonal matrices have positive entries on the diagonal.

Solution 1.4

It can be readily verified that the augmented model becomes

$$\xi(k+1) = \begin{bmatrix} A & 0 & 0 \\ C_y & I & 0 \\ C_z & 0 & I \end{bmatrix} \xi(k) + \begin{bmatrix} B \\ 0 \\ 0 \end{bmatrix} \Delta u(k)$$
$$\begin{bmatrix} y(k) \\ z(k) \end{bmatrix} = \begin{bmatrix} C_y & I & 0 \\ Z_z & 0 & I \end{bmatrix} \xi(k)$$

Solution 2.1

a) The predictions are given by

$$\hat{y}(k+1|k) = ay(k) + u(k|k)$$
$$\hat{y}(k+2|k) = a\hat{y}(k+1|k) + u(k+1jk) = a^2y(k) + (1+a)u(k|k)$$

where the standard assumption $u(k+1|k) = u(k|k)$ has been used (since the control horizon is 1). To find the minimizing control, differentiate w.r.t. $u(k|k)$ and set the result equal to zero, giving (use the abbreviations $u = u(k|k)$, $y = y(k)$, $r = r(k)$):

$$2(ay + u - r) + 2\alpha(1+a)(a^2y + (1+a)u - r) = 0$$
$$a(1 + a\alpha(1+a))y + (1 + \alpha(1+a)^2)u - (1 + \alpha(1+a))r = 0$$

which results in the control law

$$u(k) = \frac{1 + \alpha(1+a)}{1 + \alpha(1+a)^2} r(k) - \frac{a(1 + a\alpha(1+a))}{1 + \alpha(1+a)^2} y(k)$$

b) The control law in the extreme cases become

$$u(k) = \begin{cases} r(k) - ay(k), & \alpha = 0 \\ \frac{1}{1+a} r(k) - \frac{a^2}{1+a} y(k), & \alpha \to \infty \end{cases}$$

which gives the closed-loop dynamics

$$y(k+1) = \begin{cases} r(k)m & \alpha = 0 \\ \frac{a}{1+a} y(k) + \frac{1}{1+a} r(k), & \alpha \to \infty \end{cases}$$

The closed-loop poles are thus $p = 0$ and $p = a/(1+a)$ in the two cases.

Solution 2.2

a) The predictions are given by

$$\hat{y}(k+1|k) = ay(k) + u(k-1)$$
$$\hat{y}(k+2|k) = a\hat{y}(k+1|k) + u(k|k) = a^2y(k) + au(k-1) + u(k|k)$$

To find the minimizing control, differentiate w.r.t. $u(k|k)$ and set the result equal to zero, giving (using the abbreviations $u = u(k|k)$, $u^- = u(k-1)$, $y = y(k)$, $r = r(k)$):

$$a^2y + au^- + u - r + \rho u = a^2y - r + au^- + (1+\rho)u = 0$$

which gives the minimizing control

$$u(k) = -\frac{a^2}{1+\rho} y(k) + \frac{1}{1+\rho} r(k) - \frac{a}{1+\rho} u(k-1)$$

i.e. the controller is dynamic.

b) The control law in z-transformed version is

$$(1+\frac{a}{1+\rho}z^{-1})U(z) = -\frac{a^2}{1+\rho}Y(z) + \frac{1}{1+\rho}R(z)$$

which, inserted into the z-transformed system model, gives the closed-loop description

$$\begin{aligned}(z^2-az)(1+\frac{a}{1+\rho}z^{-1})Y(z) &= -\frac{a^2}{1+\rho}Y(z) + \frac{1}{1+\rho}R(z),\\ z(z-a\frac{\rho}{1+\rho})Y(z) &= \frac{1}{1+\rho}R(z)\end{aligned}$$

The closed-loop poles are thus both in the origin if $\rho = 0$ and for $\rho \to \infty$, the poles are $p_1 = 0$ and $p_2 = a$, i.e. equal to the open-loop poles.

Solution 2.3

a) The predictions are given by

$$\begin{aligned}\hat{y}(k+1|k) &= u(k|k) + bu(k-1)\\ \hat{y}(k+2|k) &= u(k+1|k) + bu(k|k)\end{aligned}$$

To find the minimizing control, differentiate V_2 w.r.t. $u(k|k)$ and $u(k+1|k)$, respectively, and set the result equal to zero:

$$\begin{aligned}(u(k|k) + bu(k-1) - r(k)) \cdot 1 + (u(k+1|k) + bu(k|k) - r(k)) \cdot b = 0\\ (u(k|k) + bu(k-1) - r(k)) \cdot 0 + (u(k+1|k) + bu(k|k) - r(k)) \cdot 1 = 0\end{aligned}$$

which gives the minimizing control (we focus on $u(k|k) = u(k)$):

$$u(k) = -bu(k-1) + r(k).$$

b) The control law in z-transformed version is

$$U(z) = \frac{1}{1+bz^{-1}}R(z)$$

which implies that the plant zero in $-b$ will be canceled by the controller pole in the same location. The cancellation implies that the closed-loop input-output relation becomes $y(k+1) = r(k)$. However, the canceled pole determines the stability, so that the condition for closed-loop stability is $|b| < 1$, which means that the process must be minimum phase.

Solution 2.4

Let's first rewrite the objective function on standard form:

$$\begin{aligned}V_N(x(0), u(0:N-1)) &= V_f(x(N)) + \sum_{k=0}^{N-1} l(x(k), u(k))\\ V_2(x(0), u(0:1)) &= (x(2)-1)^2 + \sum_{k=0}^{1} u(k)^2\end{aligned}$$

Which implies that:

$$
\begin{aligned}
&\sum_{k=0}^{N-1} l(x(k),u(k)), \quad \text{Running Cost} \\
l(x(k),u(k)) &= u(k)^2, \quad \forall k = 1,2,\cdots,N-1 \;\; \text{Stage Cost} \\
V_f(x(N)) &= V_f(x(2)) = (x(2)-1)^2, \quad \text{Terminal Cost} \\
f(x(k),u(k)) &= x(k)+u(k) \;\; \text{System Dynamics}
\end{aligned}
\tag{7.42}
$$

From definition of $V_f(x(N))$, we can see that prediction horizon $N = 2$. Note that there is no *hard-constraint* on terminal state $x(N)$. The terminal state requirement is embedded into the objective function as a *soft-constraint* which means that the terminal cost $V_f(x(2)) = (x(2)-1)^2$ penalizes the deviation of state $x(2)$ from a desired terminal state $x_d(2) = 1$. There is no constraint on $u(k)$ and $x(k)$ which means we are free to choose any u and x such that the system dynamic constraint is satisfied. Now the optimization problem on standard form can be rewritten as follows:

$$
\begin{aligned}
\text{minimize} \quad & V_2(x(0),u(0:1)) \\
\text{subject to} \quad & x(k+1) = f(x(k),u(k)) \\
& x(0) = 1, \\
& x(k) \in \mathbb{R} \;\; \text{and} \;\; u(k) \in \mathbb{R}
\end{aligned}
\tag{P–I}
$$

To solve the optimization problem (P–I), we can use the following backward dynamic programming recursion:

$$
V_{N\to N}(x(N)) = V_f(x(N)), \tag{7.43}
$$

$$
V^*_{k\to N}(x(k)) = \min_{u(k)\in\mathbb{R}} \left\{ l(x(k),u(k)) + V^*_{k+1\to N}(x(k+1)) \right\}, \tag{7.44}
$$

where $V_{N\to N}(x(N))$ is *Terminal Cost-to-Pay* for deviating from desired terminal state at 'N' and $V^*_{k\to N}(x(k))$ is *Optimal Cost-to-Go* from 'k' to 'N'. The optimal feedback control at each stage 'k' is given by

$$
u^*(k,x(k)) = \text{argmin}_{u(k)\in\mathbb{R}} \left\{ l(x(k),u(k)) + V_{k+1\to N}(x(k+1)) \right\}
$$

Where $k = \{N-1, N-2, \cdots, 0\}$. In our case $N = 2$ and Cost-to-Pay for deviation at terminal point is given by:

$$
V_{2\to 2}(x(2)) = V_f(x(2)) = (x(2)-1)^2 \tag{7.45}
$$

DP Recursion–1: Optimal Decision at *Stage–1* (or time step $k = 1$)
Optimal Cost-to-Go from time step $k = 1$ to $N = 2$ is given by:

$$
V^*_{1\to 2}(x(1)) = \min_{u(1)} \left(l(x(1),u(1)) + V_{2\to 2}(x(2)) \right) \tag{7.46}
$$

Now using (7.42) and (7.45) for $k = 1$ we get:

$$
V^*_{1\to 2}(x(1)) = \min_{u(1)} \left(u(1)^2 + (x(2)-1)^2 \right) \tag{7.47}
$$

Since system dynamics gives $x(2) = x(1) + u(1)$, so we can write:

$$
\begin{aligned}
V^*_{1\to 2}(x(1)) &= \min_{u(1)} \left(u(1)^2 + (x(1)+u(1)-1)^2 \right) \\
V^*_{1\to 2}(x(1)) &= \min_{u(1)} \left(2\,u(1)^2 + 2\,(x(1)-1)\,u(1) + (x(1)-1)^2 \right)
\end{aligned}
$$

Now let's complete the square in $u(1)$ [1] with $\alpha = 2, \quad \beta = (x(1)-1)$ and $c = (x(1)-1)^2$.

$$V^*_{1\to 2}(x(1)) = \min_{u(1)} \left(2 \left(u(1) + \frac{(x(1)-1)}{2} \right)^2 + (x(1)-1)^2 - \frac{(x(1)-1)^2}{2} \right) \tag{7.48}$$

Hence the optimal control input $u(1)$ that minimizes (7.48) is given by:

$$u^*(1) = -\frac{1}{2}\,(x(1)-1) \tag{7.49}$$

and using (7.49) in (7.48), the optimal (minimum) cost-to-go from step $k = 1$ to step $N = 2$ is given by:

$$V^*_{1\to 2}(x(1)) = \frac{1}{2}(x(1)-1)^2 \tag{7.50}$$

DP Recursion–2: Optimal Decision at *Stage–0* (or time step $k = 0$)
Again using (7.44) we will do the second recursion to calculate the optimal input $u(0)$ and the optimal Cost-to-Go $V^*_{0\to 2}(x(0))$ from time step $k = 0$ to $N = 2$:

$$V^*_{0\to 2}(x(0)) = \min_{u(0)} \left(l(x(0), u(0)) + V^*_{1\to 2}(x(1)) \right)$$

Now using (7.42) and (7.50) for $k = 0$ we get:

$$V^*_{0\to 2}(x(0)) = \min_{u(0)} \left(u(0)^2 + 0.5\,(x(1)-1)^2 \right)$$

Since system dynamics gives $x(1) = x(0) + u(0)$, so we can write:

$$V^*_{0\to 2}(x(0)) = \min_{u(0)} \left(u(0)^2 + 0.5\,(x(0) + u(0) - 1)^2 \right)$$
$$V^*_{0\to 2}(x(0)) = \min_{u(0)} \left(\frac{3}{2} {u_0}^2 + (x(0)-1)\; u(0) + 0.5\;(x(0)-1)^2 \right)$$

Now let's complete the square in $u(0)$ with $\alpha = 1.5, \quad \beta = 0.5\,(x(0)-1)$ and $c = 0.5\,(x(0)-1)^2$.

$$V^*_{0\to 2}(x(0)) = \min_{u(0)} \left(\frac{3}{2} \left(u(0) + \frac{(x(0)-1)}{3} \right)^2 + 0.5\;(x(0)-1)^2 - \frac{(0.5\;(x(0)-1))^2}{1.5} \right) \tag{7.51}$$

Hence the optimal control input $u(0)$ that minimizes (7.51) is given by:

$$u^*(0) = -\frac{1}{3}\,(x(0)-1) \tag{7.52}$$

and using (7.52) in (7.51), the optimal (minimum) cost-to-go from step $k = 0$ to step $N = 2$ is given by:

$$V^*_{0\to 2}(x(0)) = \frac{1}{3}(x(0)-1)^2$$

Now for initial state $x(0) = 1$ we get following results:

$$\text{Optimal State} \;\; x^*(1) = 1$$

using Optimal Control $u^*(0) = 0$ and Optimal Cost $V^*_{0\to 2}(x(0)) = 0$. Optimal State

$$x^*(2) = 1$$

with Optimal Control $u^*(1) = 0$, and Optimal Cost $V^*_{1\to 2}(x(1)) = 0$.

[1] Generally to complete square in any variable x we rewrite: $\alpha x^2 + 2\,\beta x + c = \alpha \left(x + \frac{\beta}{\alpha} \right)^2 + c - \frac{\beta^2}{\alpha}$

Remark 7.3. *Is the solution surprising? No, not really—in fact, one could have guessed the solution, even at the very beginning, by observing that the system is already at the desired state $x(2) = 1$, so the natural optimal strategy is to do nothing* [2]*. Even though the optimal solution in this case could have been guessed intuitively, this simple exercise demonstrates that dynamic programming does the job!*

Solution 2.5
TBD

Solution 2.6
TBD

Solution 3.1

a) **Detectability of non-augmented system**
According to *Popov-Belevitch-Hautus (PBH) test* (also known as Hautus Lemma), the detectability of (A, C) is equivalent to the rank condition

$$\text{rank} \underbrace{\begin{bmatrix} \lambda I - A \\ C \end{bmatrix}}_{=D_1} = n, \quad \forall \lambda \in \mathbb{C} : |\lambda| \geq 1 \qquad \text{(Condition 1)}$$

We denote this Condition 1.
Detectability of augmented system
Also from the Hautus Lemma, the detectability of the augmented system is equivalent to the rank condition

$$\text{rank} \underbrace{\begin{bmatrix} \lambda I - A & -B_d \\ 0 & (\lambda - 1)I \\ C & C_d \end{bmatrix}}_{=D_3} = n + n_d, \quad \forall \lambda \in \mathbb{C} : |\lambda| \geq 1 \qquad \text{(Condition 3)}$$

We denote this Condition 3. Note that, now onwards, for ease of reference we call first column-block of D_3 as D_{31} and second column-block as D_{32}.

Now we are to show that Conditions 1 and 2 are equivalent to Condition 3. Note that the above conditions are for detectability which is a bit weaker notion compared to observability (compare the above conditions with that for observability!)

Proof:
Before we proceed with the proof, let $l = n + n_d$ and rewrite D_3 in terms of its column vectors

$$D_3 = \begin{bmatrix} D_{31} & D_{32} \end{bmatrix} = \begin{bmatrix} d_1 & d_2 & \cdots & d_n & d_{n+1} & \cdots & d_l \end{bmatrix} \tag{7.53}$$

where $d_i \in \mathbb{R}^{n+n_d+p}$ is a column vector [3] and let's call $D_{31} \in \{d_1, \cdots, d_n\}$ the first set of columns and $D_{32} \in \{d_{n+1}, \cdots, d_l\}$ the second set of columns.

Condition 3 implies Conditions 1 and 2. Let's assume that Condition 3 holds and hence the matrix D_3 has full column rank. Now the proof for this part will be carried out in two steps:

[2] generally it depends on system dynamics but here our system dynamics allow us to do nothing and still keep the system on desired terminal state

[3] Don't confuse d_i's with the disturbance vector d.

Condition 3 ⇒ Conditions 1:

Since Condition 3 holds and hence the matrix D_3 has full column rank which means the set of columns in D_{31} are also independent i.e. columns of $\begin{bmatrix} \lambda I - A \\ 0 \\ C \end{bmatrix}$ are independent for $|\lambda| \geq 1$, from which it is straight forward to conclude that the columns of $\begin{bmatrix} \lambda I - A \\ C \end{bmatrix}$ are also independent and hence the Condition 1 is established.

Condition 3 ⇒ Conditions 2:

Similarly, since Condition 3 holds $\forall |\lambda| \geq 1$ so substituting $\lambda = 1$ in Condition 3 and deleting the zero rows establishes Condition 2.

Condition 1 and 2 implies Condition 3. We will establish this through the method of contradiction. Assume, contrary to what is to be proven, that Condition 1 and Condition 2 hold, and Condition 3 is violated for some $|\lambda| \geq 1$. We just have to show contradiction for some λ values. The contradiction will be proved in two steps as shown below.

Step-1: Violation of Condition 3 ⇒ Violation of Condition 2:

We will first show contradiction to assumption about Condition 2. Let us suppose Condition 3 is violated for $\lambda = 1$. In this case, 2nd row-block of matrix D_3 becomes zero and it can be deleted without changing the column rank of D_3. After deleting, D_3 becomes equal to D_2 which implies D_2 does not have full column rank either. Thus Condition 2 is violated which contradicts our initial assumption about Condition 2.

Step-2: Violation of Condition 3 ⇒ Violation of Condition 1:

Now we will show contradiction to assumption about Condition 1. Let us suppose Condition 3 is violated for some $\lambda = \lambda' > 1$, it means columns of D_3 are dependent and hence the linear combination of columns of D_3

$$\alpha_1 d_1 + \alpha_2 d_2 + \cdots + \alpha_n d_n + \alpha_{n+1} d_{n+1} + \cdots + \alpha_l d_l = 0 \tag{7.54}$$

with at least one nonzero coefficient α_i. In other words we should be able to find a set $\{\alpha_1, \cdots, \alpha_l\}$ of coefficients with at least one non-zero element such that (7.54) is satisfied. Note that for ease of reference, we call $\{\alpha_1, \cdots, \alpha_n\}$ the first set of coefficients and $\{\alpha_{n+1}, \cdots, \alpha_l\}$ as second set of coefficients.

Note that submatrix $(\lambda' - 1)I$ in D_{32} is a diagonal matrix with all non-zero entries whereas the corresponding submatrix in D_{31} is zero. Now it can be easily seen that all of the coefficients $\{\alpha_{n+1}, \cdots, \alpha_l\}$ of the second set of columns $\{d_{n+1}, \cdots, d_l\}$ must be zero because $(\lambda' - 1)I$ is diagonal and thus its each row contains only one non-zero element whereas there is the corresponding zero submatrix in the first set of columns. This can also be easily seen from the following subset of equations (corresponding to second row-block of D_3) of equation (7.54) i.e.

$$\mathbf{0} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} + [(\lambda' - 1)I] \begin{bmatrix} \alpha_{n+1} \\ \vdots \\ \alpha_l \end{bmatrix} = \mathbf{0} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} + \begin{bmatrix} (\lambda' - 1) & 0 & \cdots & 0 \\ 0 & (\lambda' - 1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & (\lambda' - 1) \end{bmatrix} \begin{bmatrix} \alpha_{n+1} \\ \vdots \\ \alpha_l \end{bmatrix} = 0$$

$\Rightarrow \alpha_{n+1} = \alpha_{n+2} = \cdots = \alpha_l = 0$

Since second set of coefficients is zero, therefore to satisfy (7.54) we must have some nonzero coefficient in the set $\{\alpha_1, \cdots, \alpha_n\}$ for the first set of columns and this implies that the columns of $D_{31} = \begin{bmatrix} \lambda' I - A \\ 0 \\ C \end{bmatrix}$ are dependent. Now deleting zero row-block, D_{31} becomes equal to D_1. Since deleting zero row-block can not increase the column rank of the resulting matrix D_1 so it implies D_1 does not have full column rank either. Thus Condition 1 is also violated, which contradicts our initial assumption about Condition 1.

Here we have shown that violation of Condition 3 also implies the violation of Condition 1 and Condition 2. Therefore, if as per our assumption, Condition 1 and Condition 2 hold then Condition 3 must also hold and hence is not violated for any $|\lambda| \geq 1$. This proves the corollary.

b) **Proof:** If $n_d > p$, then Condition 2 is violated since the matrix in Condition 2 is a $(n+p) \times (n+n_d)$ matrix which means the matrix D_2 is a *fat matrix* ($n+p < n+n_d$ i.e. more columns than rows) and since the rank$(D_2) \leq \min\{n+p, n+n_d\} = n+p$ i.e. the rank is at most $n+p$ which is less than $n+n_d$. But from the previous part of the exercise, if Condition 2 is not satisfied, the augmented system is not detectable. Therefore $n_d \leq p$ is necessary for the augmented system to be detectable.

Solution 3.2

a) In the unconstrained problem, feasibility depends on the existence of a solution to

$$\begin{bmatrix} I - A & -B \\ HC & 0 \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} 0 \\ r_{sp} \end{bmatrix} \tag{7.55}$$

This solution exists if the rows of the matrix (3.3) are independent, which is the rank condition given in the exercise but this condition is not *necessary and sufficient* but only *sufficient* because the right-hand-side in (7.55) is not arbitrary, but has a vector of zeros on left-hand-side above the controlled variable set-points r_{sp}. For example, if $A = I$ and $B = 0$, a solution can exist ($H = I$, $C = I$) despite violation of rank condition (3.3).

b) For the rows to be independent, the number of rows $n + n_c$ must be less than or equal to the number of columns $(n + m)$, which gives $n_c \leq m$. Because of the zero in the second row, the rank condition implies also that rank$(HC) = n_c$. To meet this condition, since H is an $n_c \times p$ matrix, we know $n_c \leq p$. Although the exercise doesn't ask the question, we can conclude that $n_c \leq n$.

c) The n_c rows of H are independent otherwise we contradict rank$(HC) = n_c$ from the previous part.

d) The rows of C do not need to be independent to satisfy the given rank condition. But we cannot choose all of the outputs corresponding to linearly dependent rows as controlled variables and satisfy the rank condition.

e) We can have redundant sensors and choose the mean of the redundant sensors as the controlled variable, for example.

Solution 3.3

a) The Kalman filter is given by the equations

$$\begin{aligned}
\hat{x}(k+1|k) &= \hat{x}(k|k-1) + L(k)(y(k) - \hat{x}(k|k-1)) \\
L(k) &= \frac{P(k)}{\sigma^2 + P(k)} \\
P(k+1) &= \frac{\sigma^2 P(k)}{\sigma^2 + P(k)}
\end{aligned}$$

b) $P(k) \to 0$, $L(k) \to 0$, $k \to \infty$

c) Simulation!

Solution 3.4

a) A state space model is given by

$$\begin{aligned}
x(k+1) &= -ax(k) + e(k) \\
y(k) &= (c-a)x(k) + e(k)
\end{aligned}$$

b) In steady state, the P equation for the model above becomes

$$P = a^2 P + \sigma^2 - \frac{\left(\sigma^2 - aP(c-a)\right)^2}{(c-a)^2 P + \sigma^2}$$

with the solution $P = 0$. The steady state Kalman gain is then $L = S/R = 1$ and the filter becomes

$$\hat{x}(k+1|k) = -a\hat{x}(k|k-1) + 1 \cdot (y(k) - (c-a)\hat{x}(k|k-1)) = -c\hat{x}(k|k-1) + y(k)$$

which can be written in transfer function form

$$\hat{x}(k+1|k) = \frac{1}{1+cz^{-1}} y(k)$$

c) Since $e(k)$ is white noise, we have

$$\hat{y}(k+1|k) = (c-a)\hat{x}(k+1|k) = \frac{c-a}{1+cz^{-1}} y(k)$$

Solution 3.5

a) **Pole of a Kalman filter.** Revision of basic filter equations: The Correction Step of Kalman filter is given by

$$\hat{x}(k|k) = \hat{x}(k|k-1) + L(k)(y(k) - C\hat{x}(k|k-1)), \quad \hat{x}(0|0) = x0$$

where

$$L(k) = P(k|k-1)C^\top [CP(k|k-1)C^\top + R]^{-1}$$

is the Kalman Filter gain. The Prediction Step is given by

$$\begin{aligned}
\hat{x}(k+1|k) &= A\hat{x}(k|k) + Bu(k) \\
\hat{x}(k+1|k) &= A\hat{x}(k|k-1) + Bu(k) + L(k)(y(k) - C\hat{x}(k|k-1))
\end{aligned}$$

where $L(k) = AL(k)$. The closed-loop dynamics of Kalman Filter is given by

$$\hat{x}(k+1|k) = A_f^{cs}\hat{x}(k|k-1) + Bu(k) + L(k)y(k)$$

where $A_f^{cs} = A - L(k)C$ is the closed-loop system matrix of the filter and both $u(k)$ and $y(k)$ are inputs to the filter. The state estimation error covariance matrix is given by

$$P(k|k) = P(k|k-1) - P(k|k-1)C^\top[CP(k|k-1)C^\top + R]^{-1}CP(k|k-1), \quad P(0|0) = P_0$$

and the state prediction error covariance matrix is given by

$$\begin{aligned}
P(k+1|k) &= AP(k|k)A^\top + Q \\
P(k+1|k) &= AP(k|k-1)A^\top + Q - AP(k|k-1)C^\top[CP(k|k-1)C^\top + R]^{-1}CP(k|k-1)A^\top \\
P(k+1|k) &= AP(k|k-1)A^\top + Q - L(k)CP(k|k-1)A^\top
\end{aligned}$$

Now using these equations, the Kalman filter for the given system, with noise covariances $Q = \sigma_v^2$ and $R = \sigma_e^2$, is given by

$$\hat{x}(k+1|k) = 0.5\hat{x}(k|k-1) + Bu(k) + L(k)(y(k) - C\hat{x}(k|k-1)), \tag{7.56a}$$

$$L(k) = \frac{0.5P(k|k-1)}{\sigma_e^2 + P(k|k-1)}, \tag{7.56b}$$

$$P(k+1|k) = 0.25P(k|k-1) + \sigma_v^2 - \frac{0.25P(k|k-1)^2}{\sigma_e^2 + P(k|k-1)}, \tag{7.56c}$$

$$P(k+1|k) = \frac{0.25\sigma_e^2 P(k|k-1) + \sigma_v^2 P(k|k-1) + \sigma_e^2\sigma_v^2}{\sigma_e^2 + P(k|k-1)} \tag{7.56d}$$

b) **Steady-state Kalman Filter.** The steady state Kalman filter is obtained by using $P(k+1|k) = P(k|k-1) = P$ in (7.56). Thus, we get

$$\begin{aligned}
&\hat{x}(k+1|k) = 0.5\hat{x}(k|k-1) + Bu(k) + L(y(k) - C\hat{x}(k|k-1)), \\
&L = \frac{0.5P}{\sigma_e^2 + P}, \\
&P = \frac{0.25\sigma_e^2 P + \sigma_v^2 P + \sigma_v^2\sigma_e^2}{\sigma_e^2 + P}, \\
&P^2 + (0.75\sigma_e^2 - \sigma_v^2)P - \sigma_v^2\sigma_e^2 = 0.
\end{aligned}$$

The pole of Kalman filter will depend on the relative values of covariances σ_v^2 and σ_e^2. To find closed-loop pole, let us fix the measurement noise covariance $\sigma_e^2 = 1$ and then find P as a function of σ_v^2. Figure 3 shows P, L, and closed-loop pole as a function of process noise σ_v^2 with measurement noise $\sigma_e^2 = 1$.

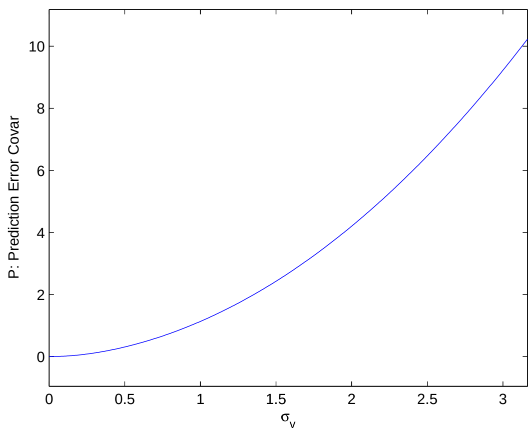
Extra Information

We can also make three cases to find analytical expression for prediction error covariance P.

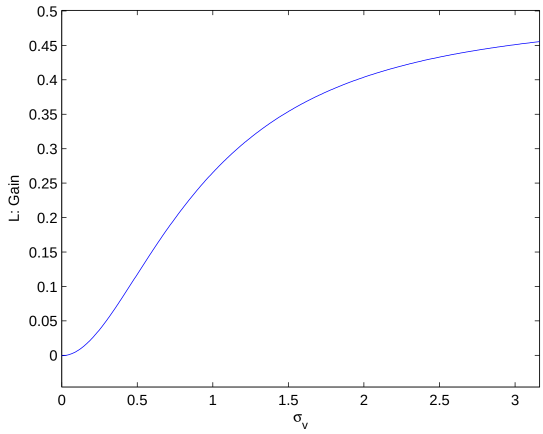
(1) Case-1: $\sigma_e^2 \gg \sigma_v^2$

Assuming $\sigma_e^2 \to \infty$, we get

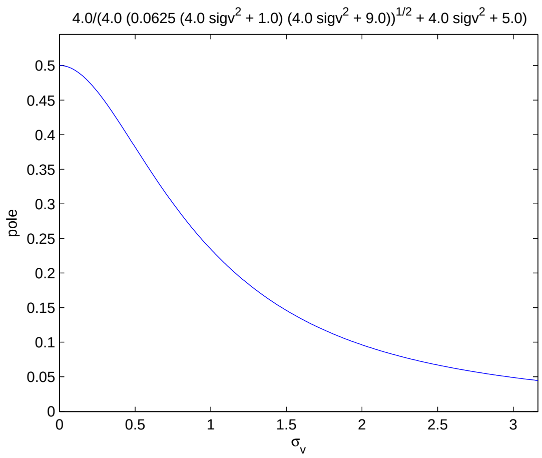
$$\begin{aligned}
&\lim_{\sigma_e^2 \to \infty}\left[\frac{P^2}{\sigma_e^2} + \left(0.75 - \frac{\sigma_v^2}{\sigma_e^2}\right)P - \sigma_v^2\right] = 0 \\
&\Rightarrow 0.75P - \sigma_v^2 = 0 \\
&\Rightarrow P = 1.33\sigma_v^2
\end{aligned}$$



(a) Prediction error covariance matrix P as a function of σ_v^2.



(b) Kalman Gain as a function of σ_v^2.



(c) Closed-loop filter pole as a function of σ_v^2.

Figure 3: Matrices P, L, and closed-loop pole as a function of process noise σ_v^2 with measurement noise $\sigma_e^2 = 1$.

(2) Case-2: $\sigma_e^2 = \sigma_v^2$

$$
\begin{aligned}
&P^2 + (0.75\sigma_v^2 - \sigma_v^2)P - \sigma_v^4 = 0 \\
&P = \frac{0.25}{2}\sigma_v^2 \pm \frac{2.01556}{2}\sigma_v^2 \\
&P > 0 \Rightarrow P = 1.1328\sigma_v^2
\end{aligned}
$$

(c) Case-3: $\sigma_e^2 \ll \sigma_v^2$
Assuming $\sigma_e^2 \approx 0$, we get

$$
\begin{aligned}
&P^2 - \sigma_v^2 P = 0 \\
&P > 0 \Rightarrow P = \sigma_v^2
\end{aligned}
$$

We can use this P for different cases to find L.

Solution 4.1
TBD

Solution 4.2

a) The optimization problem can be written as

$$\begin{aligned} &\text{minimize} \;\; V(\theta,\epsilon) = \frac{1}{2}\begin{bmatrix}\theta^T & \epsilon^T\end{bmatrix}\begin{bmatrix}\Phi & 0\\ 0 & \rho I\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} + \begin{bmatrix}\phi^T & 0\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} \\ &\text{Subject to} \;\; \begin{bmatrix}\Omega & -I\\ 0 & -I\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} \leq \begin{bmatrix}\omega\\ 0\end{bmatrix} \end{aligned} \tag{7.57}$$

which is on the standard QP form in terms of optimization variables θ and ϵ.

b) The optimization problem can be written as

$$\begin{aligned} &\text{minimize} \;\; V(\theta,\epsilon) = \frac{1}{2}\begin{bmatrix}\theta^T & \epsilon^T\end{bmatrix}\begin{bmatrix}\Phi & 0\\ 0 & \rho I\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} + \begin{bmatrix}\phi^T & 0\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} \\ &\text{Subject to} \;\; \begin{bmatrix}\Omega_1 & 0\\ \Omega_2 & -I\\ 0 & -I\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} \leq \begin{bmatrix}\omega\\ 0\end{bmatrix} \end{aligned} \tag{7.58}$$

which is on the standard QP form in terms of optimization variables θ and ϵ.

Solution 4.3

The 1-norm optimization problem (4.4) can be reformulated as

$$\begin{aligned} &\text{minimize} \;\; V(\theta,\epsilon) = \frac{1}{2}\begin{bmatrix}\theta^T & \epsilon^T\end{bmatrix}\begin{bmatrix}\Phi & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} + \begin{bmatrix}\phi^T & \rho\mathbf{1}^T\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} \\ &\text{Subject to} \;\; \begin{bmatrix}\Omega & -I\\ 0 & -I\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} \leq \begin{bmatrix}\omega\\ 0\end{bmatrix} \end{aligned} \tag{7.59}$$

where $\mathbf{1}$ is a vector of ones with the same dimension as ϵ. The problem is now on the standard QP form in terms of optimization variables θ and ϵ.

Note that both optimization problems (4.5) and (4.6) are equivalent in terms of ∞-norm penalty on slack variables. Both will results in same solution. Therefore, we can use (4.6) to represent both optimization problems on standard QP form as follows.

$$\begin{aligned} &\text{minimize} \;\; V(\theta,\epsilon) = \frac{1}{2}\begin{bmatrix}\theta^T & \epsilon\end{bmatrix}\begin{bmatrix}\Phi & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} + \begin{bmatrix}\phi^T & \rho\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} \\ &\text{Subject to} \;\; \begin{bmatrix}\Omega & -1\\ 0 & -1\end{bmatrix}\begin{bmatrix}\theta\\ \epsilon\end{bmatrix} \leq \begin{bmatrix}\omega\\ 0\end{bmatrix} \end{aligned} \tag{7.60}$$

which is also on the standard QP form in terms of optimization variables θ and ϵ.

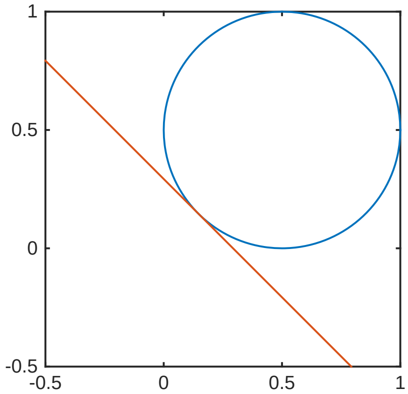
Solution 4.4

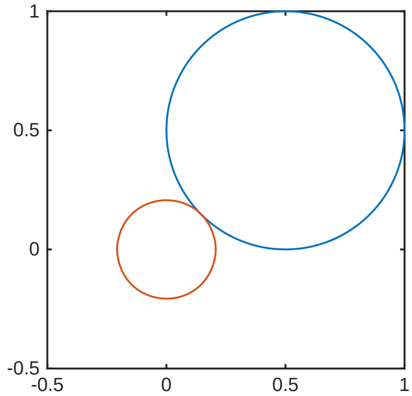
TBD

Solution 4.5

For simplicity, we illustrate the solution for the case $a = b = 1$ and $c = 0$. The constraint can then be re-written as $(x - \frac{1}{2})^2 + (y - \frac{1}{2})^2 = (\frac{1}{2})^2$, so that the feasible set is a circle with center in the point $(\frac{1}{2}, \frac{1}{2})$ and radius $\frac{1}{2}$. The level curves of the objective function $x + y$ are straight lines in the "northwest–southeast" direction, and the line through $(\frac{1}{2} + \frac{1}{2}\cos 225°, \frac{1}{2} + \frac{1}{2}\sin 225°)$ intersects the circle at the optimal point, see the left figure below.

The suggested reformulation gives an objective function with circles centered in the origin as level curves, with the optimal point in the origin. However, the constraint should be kept in the transformed problem, and the optimal point is again given by the level curve that is just touching the feasible set, see the right figure below. The two solutions are identical, as it should be.





Solution 4.6

Assume that for some d with $\nabla h(x^*)^T d = 0$ we have $\nabla f(x^*)^T d > 0$. Changing sign of d would then give $\nabla f(x^*)^T(-d) < 0$ and $\nabla h(x^*)^T(-d) = 0$. This shows that we can strengthen the condition in this case: $\nabla f(x^*)^T d = 0$ for any d such that $\nabla h(x^*)^T d = 0$.
Now, let $\nabla f(x^*) = \nabla h(x^*)\nu + v$ be a partition of $\nabla f(x^*)$ into two orthogonal components, belonging to $\mathcal{R}\nabla h(x^*)$ and $\mathcal{N}\nabla h(x^*)^T$, respectively. Note that ν is uniquely defined since x^* is regular.
Since $v \in \mathcal{N}\nabla h(x^*)^T$ we can use $d = v$ in the test, saying that $\nabla f(x^*)^T v = 0$. With the partition above, this implies $v^T v = 0$, i.e. $v = 0$, which concludes the proof in one direction. The opposite direction is immediate by multiplying the equation $\nabla f(x^*) + \nabla h(x^*)\nu^* = 0$ from the left by d^T.

Solution 5.1

Let's define

$$
\begin{aligned}
A &= \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
Q &= \begin{bmatrix} 1 & 2 \\ 2 & 6 \end{bmatrix}, \quad X = \begin{bmatrix} x(k+1)^T & x(k+2)^T \cdots x(k+N)^T \end{bmatrix}^T \\
\bar{Q} &= \begin{bmatrix} Q & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & Q \end{bmatrix} \in \mathbb{R}^{nN \times nN}
\end{aligned}
$$

and then write the following optimization problem

$$
\begin{aligned}
\text{minimize} \quad & V(x) = X^T \bar{Q} X \\
\text{subject to} \quad & x(k+1) = Ax(k) + Bu(k)
\end{aligned}
\tag{P-III}
$$

We know that

$$
X = \Psi x(k) + \Phi \mathcal{U} \tag{7.61}
$$

where:

$$\Psi = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix}, \quad \Phi = \begin{bmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \cdots & B \end{bmatrix}, \quad \text{for} \quad N = M \tag{7.62}$$

Note that for $M < N$ the form for Φ will change accordingly i.e. we have to use blocking.

Now using X the objective function can be expressed as follows:

$$\begin{aligned} V(x(k), \mathcal{U}) &= (\Psi x(k) + \Phi \mathcal{U})^T \bar{Q} (\Psi x(k) + \Phi \mathcal{U}) \\ &= x(k)^T \Psi^T \bar{Q} \Psi x(k) + x(k)^T \Psi^T \bar{Q} \Phi \mathcal{U} + \mathcal{U}^T \Phi^T \bar{Q} \Psi x(k) + \mathcal{U}^T \Phi^T \bar{Q} \Phi \mathcal{U} \\ &= x(k)^T \Psi^T \bar{Q} \Psi x(k) + 2x(k)^T g^T \mathcal{U} + \mathcal{U}^T H \mathcal{U} \end{aligned} \tag{7.63}$$

where:

$$H = \Phi^T \bar{Q} \Phi, \quad g = \Phi^T \bar{Q} \Psi \tag{7.64}$$

Since $V(x(k), \mathcal{U})$ is quadratic in $\mathcal{U}$, so using $\nabla_{\mathcal{U}} V = \left(\frac{\partial V}{\partial \mathcal{U}}\right)^T = 0$ we get

$$\mathcal{U}^* = -H^{-1} g x(k) \Rightarrow \bar{\mathcal{K}} = -H^{-1} g \in \mathbb{R}^{M \times n}, \quad \mathcal{K} = \bar{\mathcal{K}}(1,:) \tag{7.65}$$

a) **Case:** $N = 1, \quad M = 1$

$$\Psi = A = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad \Phi = B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \Rightarrow g = \begin{bmatrix} 2 & 0 \end{bmatrix}, \quad H = 1 \Rightarrow \mathcal{K} = \begin{bmatrix} -2 & 0 \end{bmatrix} \tag{7.66}$$

So the closed-loop dynamics is given by

$$x(k+1) = (A + B\mathcal{K})\, x(k) = A_{cs} x(k), \quad A_{cs} = \begin{bmatrix} -2 & 0 \\ 1 & 0 \end{bmatrix} \tag{7.67}$$

so the eigenvalues of closed-loop system: $\text{eig}(A_{cs}) = \lambda = \{-2, 0\}$, since not all $|\lambda| < 1$ which means that the closed-loop system is unstable.

b) **Case:** $N = 2, \quad M = 1$

$$\Psi = \begin{bmatrix} A \\ A^2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \Phi = \begin{bmatrix} B \\ AB + B \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \Rightarrow g = \begin{bmatrix} 2 & 0 \end{bmatrix}, \quad H = 12 \Rightarrow \mathcal{K} = \begin{bmatrix} -\frac{1}{6} & 0 \end{bmatrix} \tag{7.68}$$

So the closed-loop dynamics is given by

$$x(k+1) = (A + B\mathcal{K})\, x(k) = A_{cs} x(k), \quad A_{cs} = \begin{bmatrix} -\frac{1}{6} & 0 \\ 1 & 0 \end{bmatrix} \tag{7.69}$$

so the eigenvalues of closed-loop system: $\text{eig}(A_{cs}) = \lambda = \{-\frac{1}{6}, 0\}$, since all $|\lambda| < 1$ which means that the closed-loop system is stable.

c) **Case:** $N = 2, \quad M = 2$

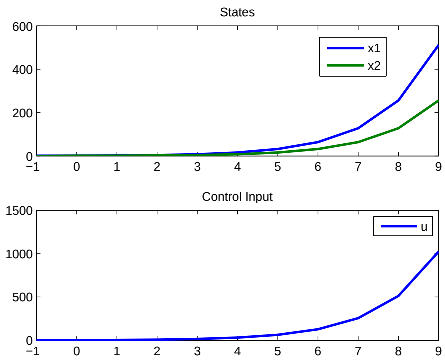
$$\Psi = \begin{bmatrix} A \\ A^2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \Phi = \begin{bmatrix} B & 0 \\ AB & B \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \Rightarrow g = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}, \quad H = \begin{bmatrix} 7 & 2 \\ 2 & 1 \end{bmatrix} \tag{7.70}$$

$$
\Rightarrow \bar{\mathcal{K}} = \begin{bmatrix} -\frac{2}{3} & 0 \\ \frac{4}{3} & 0 \end{bmatrix} \Rightarrow \mathcal{K} = \begin{bmatrix} -\frac{2}{3} & 0 \end{bmatrix} \tag{7.71}
$$

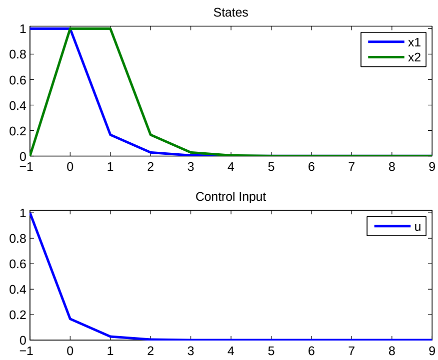
So the closed-loop dynamics is given by

$$
x(k+1) = (A + B\mathcal{K})\, x(k) = A_{cs} x(k), \quad A_{cs} = \begin{bmatrix} -\frac{2}{3} & 0 \\ 1 & 0 \end{bmatrix} \tag{7.72}
$$

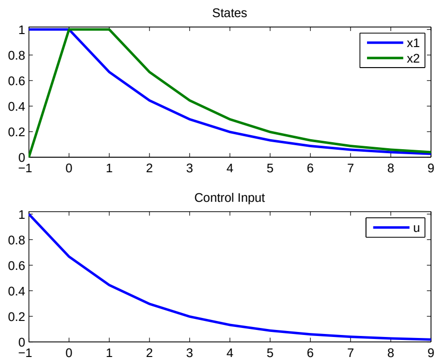
so the eigenvalues of closed-loop system: $\mathrm{eig}(A_{cs}) = \lambda = \{-\frac{2}{3}, 0\}$, since all $|\lambda| < 1$ which means that the closed-loop system is stable but response will be slower than Case b.



(a) Case a: $N = 1, \quad M = 1$



(b) Case b: $N = 2, \quad M = 1$



(c) Case c: $N = 2, \quad M = 2$

Figure 4: System Response for different values of prediction and control horizon

Solution 5.2

TBD

Solution 6.1

The Newton step for the given problem is actually given in the Lecture notes, but for clarity it is provided here using ordinary derivatives instead of using the r notation. The Newton step in the general case is given by

$$
\frac{\partial r(x)}{\partial x} \Delta x = -r(x).
$$

Applied to the KKT conditions

$$
\left(\frac{\partial L(x, \lambda)}{\partial x} \right)^{\top} = 0
$$

$$
h(x) = 0
$$

the Newton step becomes

$$\begin{bmatrix} \frac{\partial^2 L(x,\lambda)}{\partial x^2} & \frac{\partial}{\partial \lambda}\left(\frac{\partial L(x,\lambda)}{\partial x}\right)^\top \\ \frac{\partial h(x)}{\partial x} & 0 \end{bmatrix} \leq \begin{bmatrix} \Delta x \\ \Delta \lambda \end{bmatrix} = -\begin{bmatrix} \left(\frac{\partial L(x,\lambda)}{\partial x}\right)^\top \\ h(x) \end{bmatrix}.$$

Using the expressions for the Lagrangian and its derivative,

$$L = f(x) + \lambda^\top h(x), \quad \frac{\partial L}{\partial x} = \frac{\partial f}{\partial x} + \lambda^\top \frac{\partial h}{\partial x}$$

we get

$$\begin{bmatrix} \frac{\partial^2 L}{\partial x^2} & \left(\frac{\partial h}{\partial x}\right)^\top \\ \frac{\partial h}{\partial x} & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \lambda \end{bmatrix} = -\begin{bmatrix} \left(\frac{\partial f(x)}{\partial x}\right)^\top + \left(\frac{\partial h(x)}{\partial x}\right)^\top \lambda \\ h(x) \end{bmatrix},$$

or, with a slight rearrangement,

$$\begin{bmatrix} \frac{\partial^2 L}{\partial x^2} & \left(\frac{\partial h}{\partial x}\right)^\top \\ \frac{\partial h}{\partial x} & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \lambda^+ \end{bmatrix} = -\begin{bmatrix} \left(\frac{\partial f(x)}{\partial x}\right)^\top \\ h(x) \end{bmatrix}$$

where $\lambda^+ = \lambda + \Delta\lambda$. Now, recall the solution of the QP problem given in the Lecture notes:

$$\begin{aligned} &\text{minimize } f(x) = \frac{1}{2} x^\top Q x + p^\top x, \quad Q \succ 0 \\ &\text{subject to } Ax = b \end{aligned}$$

where the solution is given by the system of linear equations

$$\begin{bmatrix} Q & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} x^+ \\ \lambda^+ \end{bmatrix} = \begin{bmatrix} -p \\ b \end{bmatrix}$$

Comparing the two solutions, we have

$$\begin{aligned} Q &= \frac{\partial^2 L}{\partial x^2}, \quad p = \left(\frac{\partial f(x)}{\partial x}\right)^\top, \\ A &= \frac{\partial h}{\partial x}, \quad b = -h(x). \end{aligned}$$

Now we see that the Newton direction for Δx and the updated Lagrange multiplier λ^+ are obtained by solving the QP problem

$$\begin{aligned} &\text{minimize } \frac{1}{2} \Delta x^\top \frac{\partial^2 L(x,\lambda)}{\partial x^2} \Delta x + \left(\frac{\partial f(x)}{\partial x}\right)^\top \Delta x \\ &\text{subject to } \frac{\partial h}{\partial x} \Delta x = -h(x) \end{aligned}$$

which can be seen as a local quadratic approximation of the original optimization problem.