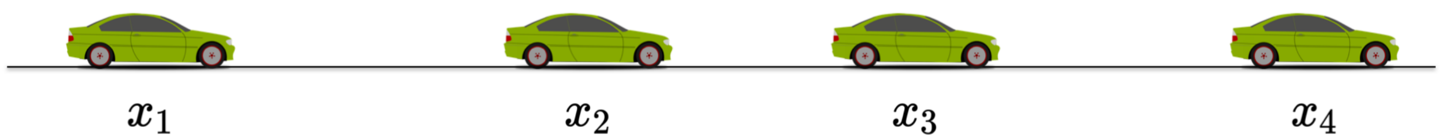
Problem 1. [100 Points] Platooning

In a single lane straight road, N vehicles are moving to the right with respective 1D position coordinates $x_1, x_2, \ldots, x_N$. See Fig. showing an example scenario for $N = 4$.



Suppose that all vehicles have identical discrete time controlled dynamics $x_i(k+1) = x_i(k) + h(u_i(k) - v)$, $i = 1, 2, \ldots, N$, for time index $k = 0, \ldots, T-1$. The parameter $h > 0$ is given constant sampling time. The control u_i can be thought of as the speed of the ith vehicle, and v is a known speed limit.

Here is the high level question of interest: what should be the optimal controls such that all consecutive vehicles maintain a separation close to some (known) desired distance d at all times? A separation smaller than d may be unsafe, and thus undesirable. A separation more than d reduces traffic throughput, and therefore also undesirable. We also want all vehicles to move at a speed close to the known speed limit v.

(a) [35 points] OCP formulation

Motivated by the aforesaid objective, consider minimizing

$$\frac{1}{2}\sum_{i=1}^{N-1}(x_{i+1}(T) - x_i(T) - d)^2 + \frac{1}{2}\sum_{k=0}^{T-1}\left\{\sum_{i=1}^{N-1}(x_{i+1}(k) - x_i(k) - d)^2 + \sum_{i=1}^{N}(u_i(k) - v)^2\right\}$$

subject to $x_i(k+1) = x_i(k) + h(u_i(k) - v)$, $i = 1, 2, \ldots, N$. Consider the final time T fixed.

Recast this problem as discrete time finite horizon LQ tracking by clearly defining the **state vector x and its dimension**, the **control vector u and its dimension**, the **output vector y and its dimension**, the **desired output trajectory y_d to track**, the **system matrices A, B, C**, and the **weight matrices M, Q, R in the cost function.**

Hint: Take a look at Lec. 10, p. 3 to see how LQ tracking problem was formulated in the continuous time case. See also part (b) to get a hint on the problem structure.

Solution for part (a):

Let $\mathbf{1}_n$ denote an $n \times 1$ column vector of ones, and let I_n denote the $n \times n$ identity matrix. We define the state vector x, the control vector u, the output vector y, and the desired output vector y_d as

$$x := \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} \in \mathbb{R}^N, \quad u := \begin{pmatrix} u_1 - v \\ u_2 - v \\ \vdots \\ u_N - v \end{pmatrix} \in \mathbb{R}^N, \quad y := \begin{pmatrix} x_2 - x_1 \\ x_3 - x_2 \\ \vdots \\ x_N - x_{N-1} \end{pmatrix} \in \mathbb{R}^{N-1}, \quad y_d := d\mathbf{1}_{N-1} \in \mathbb{R}^{N-1}.$$

Consequently, the system matrices are

$$A := I_N, \qquad B := hI_N, \qquad C := \begin{pmatrix} -1 & 1 & 0 & 0 & \dots & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 \\ 0 & 0 & -1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & -1 & 1 \end{pmatrix} \in \mathbb{R}^{(N-1)\times N}.$$

The cost weight matrices are

$$M = Q = I_{N-1} \succ 0, \qquad R = I_N \succ 0.$$

With the above definitions in place, we can transcribe the given problem into standard discrete time LQR tracking problem:

$$\underset{\{u_k\}_{k=0}^{T-1}}{\text{minimize}} \quad \frac{1}{2}\left\{(y(T) - y_d(T))' M (y(T) - y_d(T)) + \sum_{k=0}^{T-1} (y(k) - y_d(k))' Q (y(k) - y_d(k)) + (u(k))' R u(k)\right\}$$

subject to $x(k+1) = Ax(k) + Bu(k)$, $y(k) = Cx(k)$, $k = 0, 1, \dots, T-1$.

(b) [35 points] Discrete time LQ tracking solution

Extend the derivation in Lec. 10, p. 15-23 for the tracking case:

$$\underset{\{u_k\}_{k=0}^{T-1}}{\text{minimize}} \quad \frac{1}{2}\left\{(y(T) - y_d(T))' M (y(T) - y_d(T)) + \sum_{k=0}^{T-1} (y(k) - y_d(k))' Q (y(k) - y_d(k)) + (u(k))' R u(k)\right\}$$

subject to $x(k+1) = Ax(k) + Bu(k)$, $y(k) = Cx(k)$, $k = 0, 1, \dots, T-1$.

Hint: Just like the continuous time LQ tracking solution given in Lec. 10, here too you should get optimal control as a sum of a linear state feedback term, and a feedforward term. You need to derive a backward **vector** recursion for the feedforward control. Also derive how the Riccati backward recursion needs to be modified in this case, compared to the same for LQR.

Solution for part (b):

As usual, we start with the Hamiltonian

$$H(k)=\frac{1}{2}(Cx(k)-d\mathbf{1})'Q(Cx(k)-d\mathbf{1})+\frac{1}{2}(u(k))'Ru(k)+(\lambda(k+1))'(Ax(k)+Bu(k)),$$

which gives the following necessary conditions for optimality:

$$x(k+1)=Ax(k)+Bu(k),\quad \lambda(k)=\frac{\partial H(k)}{\partial x(k)}=C'QCx(k)-C'Qd\mathbf{1}+A'\lambda(k+1),$$
$$0=\frac{\partial H(k)}{\partial u(k)}=Ru(k)+B'\lambda(k+1),$$

with boundary conditions

$$\lambda(T)=\frac{\partial\phi(x(T),T)}{\partial x(T)}=C'M\left(Cx(T)-d\mathbf{1}\right),\qquad x(0)\text{ given.}$$

This gives us the 2PBVP

$$\begin{pmatrix}x(k+1)\\ \lambda(k)\end{pmatrix}=\begin{pmatrix}A & -BR^{-1}B'\\ C'QC & A'\end{pmatrix}\begin{pmatrix}x(k)\\ \lambda(k+1)\end{pmatrix}+\begin{pmatrix}0\\ -C'Q\end{pmatrix}d\mathbf{1}$$

subject to the above boundary conditions.

We consider the ansatz $\lambda(k)=P(k)x(k)-v(k)$, which upon comparing with our boundary ocndition, yields

$$P(T)=C'MC,\quad v(T)=C'Md\mathbf{1}.$$

Now we proceed as in Lec. 10, p. 18-19, mutatis mutandis, to obtain

$$\lambda(k+1)=P(k+1)x(k+1)-v(k+1)=P(k+1)(Ax(k)-BR^{-1}B'\lambda(k+1))-v(k+1)$$
$$\Rightarrow\ \lambda(k+1)=\left(I+P(k+1)BR^{-1}B'\right)^{-1}P(k+1)Ax(k)-\left(I+P(k+1)BR^{-1}B'\right)^{-1}v(k+1).$$

On the other hand, the costate equation gives $\lambda(k)=C'QCx(k)+A'\lambda(k+1)-C'Qd\mathbf{1}$, wherein the LHS equals $P(k)x(k)-v(k)$, and in the RHS, we substitute the expression for $\lambda(k+1)$ derived above, to arrive at

$$P(k)x(k)-v(k)=\left(C'QC+A'\left(I+P(k+1)BR^{-1}B'\right)^{-1}P(k+1)A\right)x(k)-A'\left(I+P(k+1)BR^{-1}B'\right)^{-1}v(k+1)$$
$$-\,C'Qd\mathbf{1}.$$

Since the above must hold for all initial conditions x_0, and thus for arbitrary $x(k)$, we must have:

$$P(k)=C'QC+A'\left(I+P(k+1)BR^{-1}B'\right)^{-1}P(k+1)A,\quad P(T)=C'MC,$$
$$v(k)=A'\left(I+P(k+1)BR^{-1}B'\right)^{-1}v(k+1)+C'Qd\mathbf{1},\quad v(T)=C'Md\mathbf{1},$$

which may be put in slightly more symmetric form as in Lec. 10, p. 20.

Once we solve the above matrix-vector backward recursions, we can write the optimal control u^{opt} as follows:

$$
\begin{aligned}
u^{\text{opt}}(k) &\overset{\text{(PMP)}}{=} -R^{-1}B'\lambda(k+1) = -R^{-1}B^{\top}P(k+1)(Ax(k)+Bu^{\text{opt}}(k)) + R^{-1}B'v(k+1)\\
\Rightarrow \quad \left(I + R^{-1}B'P(k+1)B\right)u^{\text{opt}}(k) &= -R^{-1}B'P(k+1)Ax(k) + R^{-1}B'v(k+1)\\
\Rightarrow \quad u^{\text{opt}}(k) &= -\left(I + R^{-1}B'P(k+1)B\right)^{-1}R^{-1}B'P(k+1)Ax(k) + \left(I + R^{-1}B'P(k+1)B\right)^{-1}R^{-1}B'v(k+1)\\
&= -\left(R^{-1}R + B'P(k+1)B\right)^{-1}R^{-1}B'P(k+1)Ax(k) + \left(R^{-1}R + B'P(k+1)B\right)^{-1}R^{-1}B'v(k+1)\\
&= -\left(R + B'P(k+1)B\right)^{-1}RR^{-1}B'P(k+1)Ax(k) + \left(R + B'P(k+1)B\right)^{-1}RR^{-1}B'v(k+1)\\
&= -\left(R + B'P(k+1)B\right)^{-1}B'P(k+1)Ax(k) + \left(R + B'P(k+1)B\right)^{-1}B'v(k+1)\\
&= u_{\text{feedback}} + u_{\text{feedforward}},
\end{aligned}
$$

where

$$
u_{\text{feedback}} := -K(k)x(k), \quad K(k) := K_v P(k+1)A, \quad K_v := \left(R + B'P(k+1)B\right)^{-1}B', \quad u_{\text{feedforward}} := K_v v(k+1).
$$

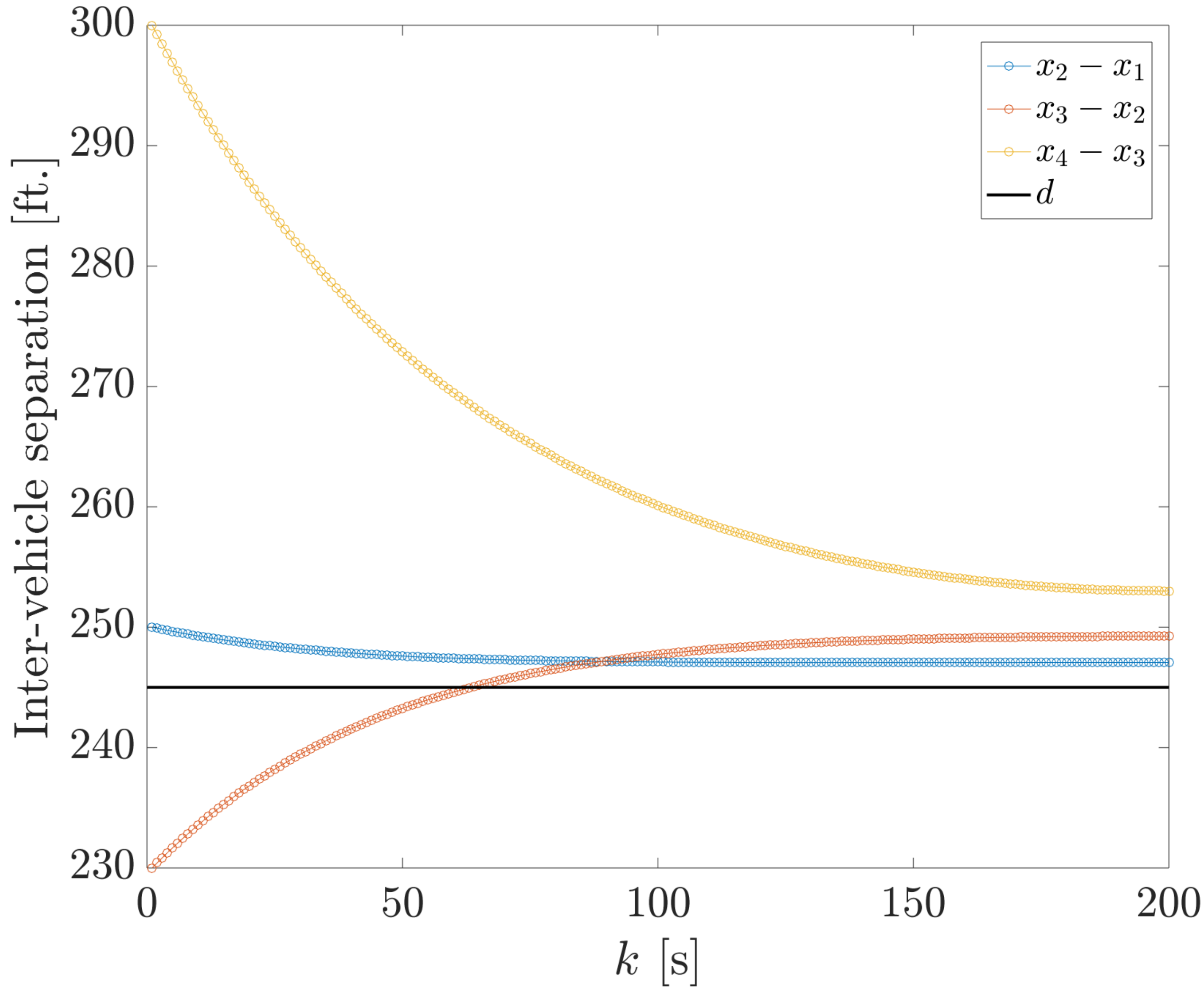
(c) [30 points] Optimal control for platooning

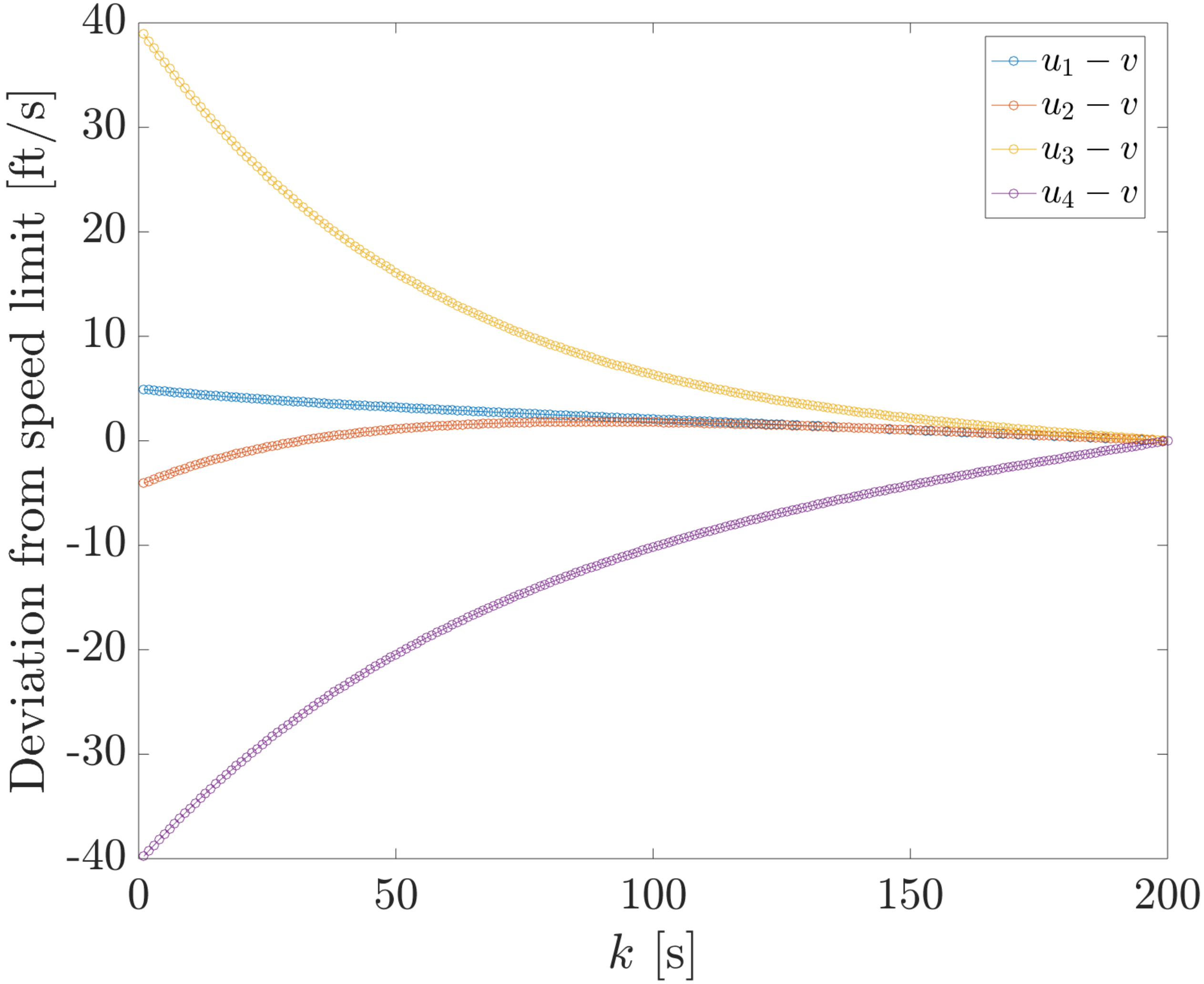
Apply your answer in part (b) to the formulation in part (a), to **compute and plot** $y^{\text{opt}}(k)$ superimposed with $y_d(k)$. Also plot $u^{\text{opt}}(k)$. To make these plots, fix $T = 200$, $h = 0.01$, $N = 4$, $d = 245$ ft, and the initial conditions $x_1(0) = 0$ ft, $x_2(0) = 250$ ft, $x_3(0) = 480$ ft, $x_4(0) = 780$ ft.

Please submit your single MATLAB code generating these plots.

Solution for part (c):

Please see the posted MATLAB code `UCSC-AM232-S21-HW4.m` posted in CANVAS Files section, folder: HW. The plots are shown below.


300
290
280
270
260
250
240
230
0
50
100
150
200
Inter-vehicle separation [ft.]
k [s]
$x_2 - x_1$
$x_3 - x_2$
$x_4 - x_3$
d



$u_1 - v$
$u_2 - v$
$u_3 - v$
$u_4 - v$
Deviation from speed limit [ft/s]
k [s]
40
30
20
10
0
-10
-20
-30
-40
0
50
100
150
200