ME 233 Spring 2012
Solution to Homework #4

1. **Finite Horizon Optimal Tracking Problem**
   The LQ tracking problem is formulated as follows:

$$\begin{aligned}
&\min_{U_0} \{J\} \\
J &= [y_d(N) - y(N)]^T \bar{Q}_f \, [y_d(N) - y(N)] \\
&\quad + \sum_{k=0}^{N-1} \left\{ [y_d(k) - y(k)]^T \bar{Q} [y_d(k) - y(k)] + u^T(k) R u(k) \right\} \\
U_k &= \{u(k), u(k+1), \cdots, u(N-1)\}
\end{aligned}$$

subject to

$$\begin{aligned}
x(k+1) &= Ax(k) + Bu(k) \\
y(k) &= Cx(k) \\
x(0) &= x_0
\end{aligned}$$

where $y_d(k)$ is specified for all k.

This is analogous to the LQ regulator problem which has been discussed in detail in class. Define:

$$\begin{aligned}
J_k^o[x(k)] &= \min_{U_k} \Big\{ [y_d(N) - y(N)]^T \bar{Q}_f \, [y_d(N) - y(N)] \\
&\quad + \sum_{i=0}^{N-1} \left\{ [y_d(i) - y(i)]^T \bar{Q} [y_d(i) - y(i)] + u^T(i) R u(i) \right\} \Big\}
\end{aligned}$$

First note that

$$\begin{aligned}
J_N^o[x(N)] &= [(y_d(N) - y(N)]^T \bar{Q}_f [y_d(N) - y(N)] \\
&= x^T(N) C^T \bar{Q}_f C x(N) - 2x^T(N) C^T \bar{Q}_f y_d(N) + y_d^T(N) \bar{Q}_f y_d(N)
\end{aligned}$$

Defining

$$P(N) = C^T \bar{Q}_f C \tag{1}$$
$$b(N) = -2C^T \bar{Q}_f y_d(N) \tag{2}$$
$$c(N) = y_d^T(N) \bar{Q}_f y_d(N) \tag{3}$$

gives

$$J_N^o[x(N)] = x^T(N) P(N) x(N) + x^T(N) b(N) + c(N)$$

which is in the form shown in the hint.

Now, we will prove using induction that $J_k^o[x(k)]$ has the form shown in the hint. Using Bellman's principle of optimality we can obtain a recursive relation between $J_{k-1}^o[x(k-1)]$, which is the optimal cost to go from $x(k-1)$ to $x(N)$, and $J_k^o[x(k)]$:

$$\begin{aligned}
J_{k-1}^o[x(k-1)] &= \min_{u(k)} \Big\{ [y_d(k-1) - y(k-1)]^T \bar{Q} \, [y_d(k-1) - y(k-1)] \\
&\quad + u^T(k-1) R u(k-1) + J_k^o(x(k)) \Big\}
\end{aligned}$$

Assuming that $J_k^o[x(k)]$ has the form shown in the hint gives

$$\begin{aligned}
J_{k-1}^o[x(k-1)] &= \min_{u(k)} \{ x^T(k-1) \left[C^T \bar{Q} C + A^T P(k) A\right] x(k-1) \\
&\quad + x^T(k-1) \left[A^T b(k) - 2C^T \bar{Q} y_d(k-1)\right] + u^T(k-1) \left[R + B^T P(k) B\right] u(k-1) \\
&\quad + u^T(k-1) B^T \left[2P(k)Ax(k-1) + b(k)\right] + y_d^T(k-1)\bar{Q} y_d(k-1) + c(k) \}
\end{aligned}$$

Taking the partial derivative of the term in the curly braces with respect to $u(k-1)$ and setting it equal to 0 gives

$$u^o(k-1) = -\left[R + B^T P(k) B\right]^{-1} B^T \left[P(k)Ax(k-1) + \frac{1}{2} b(k)\right] \tag{4}$$

And then:

$$\begin{aligned}
J_{k-1}^o[x(k-1)] &= x^T(k-1) \left\{ C^T \bar{Q} C + A^T P(k) A - A^T P(k) B \left[R + B^T P(k) B\right]^{-1} B^T P(k) A \right\} x(k-1) \\
&\quad + x^T(k-1) \left\{ A^T b(k) - 2C^T \bar{Q} y_d(k-1) - A^T P(k) B \left[R + B^T P(k) B\right]^{-1} B^T b(k) \right\} \\
&\quad + \left\{ y_d^T(k-1) \bar{Q} y_d(k-1) + c(k) - \frac{1}{4} b^T(k) B \left[R + B^T P(k) B\right]^{-1} B^T b(k) \right\}
\end{aligned}$$

Defining

$$P(k-1) = C^T \bar{Q} C + A^T P(k) A - A^T P(k) B \left[R + B^T P(k) B\right]^{-1} B^T P(k) A \tag{5}$$

$$b(k-1) = A^T b(k) - 2C^T \bar{Q} y_d(k-1) - A^T P(k) B \left[R + B^T P(k) B\right]^{-1} B^T b(k) \tag{6}$$

$$c(k-1) = y_d^T(k-1) \bar{Q} y_d(k-1) + c(k) - \frac{1}{4} b^T(k) B \left[R + B^T P(k) B\right]^{-1} B^T b(k) \tag{7}$$

gives

$$J_{k-1}^o[x(k-1)] = x^T(k-1) P(k-1) x(k-1) + x^T(k-1) b(k-1) + c(k-1)$$

which concludes our proof by induction. Thus our optimal control law is given by equations (1)–(7).

2. **Application of Dynamic Programming**
Our goal is to solve the following problem:

$$\max_{U_0} \{J\}$$

$$\begin{aligned}
J &= \prod_{i=0}^{N-1} u(i), \qquad u(i) \geq 0 \\
U_k &= \{u(k), u(k+1), \ldots, u(N-1)\} \\
x(k+1) &= x(k) + u(k) \\
x(0) &= 0 \\
x(N) &= L
\end{aligned}$$

Define

$$\begin{aligned}
J_k[x(k)] &= \prod_{i=k}^{N-1} u(i) \\
J_k^o[x(k)] &= \max_{U_k} \left\{ \prod_{i=k}^{N-1} u(i) \right\} \\
\Rightarrow J_{N-1}^o[x(N-1)] &= u^o(N-1) \\
&= L - x(N-1)
\end{aligned}$$

The central idea in dynamic programming is to express the optimal cost at time step k as a function of of the optimal cost at time step $k+1$ so that a backward recursive scheme may be used. We will do that now.

$$\begin{aligned}
J_k^o[x(k)] &= \max_{U_k} \left\{ \prod_{i=k}^{N-1} u(i) \right\} \\
&= \max_{u(k), U_{k+1}} \left\{ u(k) \prod_{i=k+1}^{N-1} u(i) \right\} \\
&= \max_{u(k)} \left\{ u(k) \max_{U_{k+1}} \left(\prod_{i=k+1}^{N-1} u(i) \right) \right\} \\
&= \max_{u(k)} \left\{ u(k) J_{k+1}^o[x(k+1)] \right\}
\end{aligned}$$

You may need to convince yourself of some of the intermediate steps in the above set of equations.

Consider the equation:

$$\begin{aligned}
J_{N-2}^o[x(N-2)] &= \max_{u(N-2)} \left(u(N-2) J_{N-1}^o[x(N-1)] \right) \\
\Rightarrow u^o(N-2) &= \arg\left(\max_{u(N-2)} \left\{ u(N-2) J_{N-1}^o[x(N-1)] \right\} \right) \\
&= \arg\left(\max_{u(N-2)} \left\{ u(N-2) \left[L - x(N-1) \right] \right\} \right) \\
&= \arg\left(\max_{u(N-2)} \left\{ u(N-2) \left[L - x(N-2) - u(N-2) \right] \right\} \right) \\
&= \frac{L - x(N-2)}{2}
\end{aligned}$$

Similarly,

$$\begin{aligned}
u^o(N-3) &= \arg\left(\max_{u(N-3)} \left\{ u(N-3) J_{N-2}^o[x(N-2)] \right\} \right) = \frac{L - x(N-3)}{3} \\
\vdots &= \qquad \vdots \\
u^o(0) &= \arg\left(\max_{u(0)} \left\{ u(0) J_1^o[x(1)] \right\} \right) = \frac{L - x(0)}{N}
\end{aligned}$$

Given $u^o(0) = L/N$, the above set of equations yield $u(i) = L/N$ for all i.

3. For this problem, it is useful to combine the plant dynamics with the noise dynamics. To do this, note that

$$\begin{aligned}
x(k+1) &= Ax(k) + Bu(k) + B_w w(k) \\
&= Ax(k) + Bu(k) + B_w C_w x_w(k).
\end{aligned}$$

Thus, if we define

$$\bar{x}(k) := \begin{bmatrix} x(k) \\ x_w(k) \end{bmatrix}, \qquad \bar{A} := \begin{bmatrix} A & B_w C_w \\ 0 & A_w \end{bmatrix}, \qquad \bar{B} := \begin{bmatrix} B \\ 0 \end{bmatrix}, \qquad \bar{B}_w := \begin{bmatrix} 0 \\ B_n \end{bmatrix}$$

we can write the augmented system dynamics as

$$\bar{x}(k+1) = \bar{A}\bar{x}(k) + \bar{B}u(k) + \bar{B}_w\eta(k)$$
$$E\{\bar{x}(0)\} = \bar{x}_o := \begin{bmatrix} x_o \\ x_{wo} \end{bmatrix}$$
$$E\left\{(\bar{x}(0) - \bar{x}_o)(\bar{x}(0) - \bar{x}_o)^T\right\} = \bar{X}_o := \begin{bmatrix} X_o & 0 \\ 0 & X_{wo} \end{bmatrix}.$$

To finish the redefinition of the problem, we need to reformulate the LQG cost. Note that if we define

$$\bar{Q} := \begin{bmatrix} Q & 0 \\ 0 & 0 \end{bmatrix}, \qquad \bar{Q}_f := \begin{bmatrix} Q_f & 0 \\ 0 & 0 \end{bmatrix}$$

we can rewrite the LQG cost as

$$J = E\left\{ \bar{x}^T(N)\bar{Q}_f\bar{x}(N) + \sum_{k=0}^{N-1} \bar{x}^T(k)\bar{Q}\bar{x}(k) + u^T(k)Ru(k) \right\}$$

(a) Since assuming that $x(k)$ and $x_w(k)$ are measurable for all k is equivalent to assuming that $\bar{x}(k)$ is measurable for all k, this problem is simply an LQG problem with exactly known state. The optimal control is thus given by

$$u(k) = -K(k+1)\bar{x}(k)$$

where

$$K(k) = \left(\bar{B}^T P(k)\bar{B} + R\right)^{-1} \bar{B}^T P(k)\bar{A}$$
$$P(k-1) = \bar{A}^T P(k)\bar{A} + \bar{Q} - \bar{A}^T P(k)\bar{B}\left(\bar{B}^T P(k)\bar{B} + R\right)^{-1} \bar{B}^T P(k)\bar{A}$$
$$P(N) = \bar{Q}_f.$$

(b) In this part, we assume that we only have access to the measurements

$$y(k) = Cx(k) + v(k).$$

If we define

$$\bar{C} := \begin{bmatrix} C & 0 \end{bmatrix}$$

the measurements can be expressed

$$y(k) = \bar{C}\bar{x}(k) + v(k).$$

Thus, this problem is simply an LQG problem. The Kalman filter for this system is given by

$$\hat{\bar{x}}^o(k+1) = \bar{A}\hat{\bar{x}}(k) + \bar{B}u(k)$$
$$\hat{\bar{x}}(k) = \hat{\bar{x}}^o(k) + F(k)\left(y(k) - \bar{C}\hat{\bar{x}}^o(k)\right)$$

where

$$F(k) = M(k)\bar{C}^T \left[\bar{C}M(k)\bar{C}^T + V\right]^{-1}$$
$$M(k+1) = \bar{A}Z(k)\bar{A}^T + \bar{B}_w\Gamma\bar{B}_w^T$$
$$Z(k) = M(k) - M(k)\bar{C}^T \left[\bar{C}M(k)\bar{C}^T + V\right]^{-1} \bar{C}M(k)$$
$$M(0) = \bar{X}_o.$$

The optimal control is thus given by

$$u(k) = -K(k+1)\hat{\bar{x}}(k)$$

where $K(k)$ is the same as in part (a).