16.30/31, Fall 2010 — Recitation # 10

Brandon Luders

November 15, 2010

In this recitation, we explore the Linear Quadratic Estimator (LQE) problem.

1 Linear Quadratic Estimator

1.1 Motivation

In Topic 14, when discussing estimators, we pointed out that there is a duality between the *regulator* problem (choosing a feedback K to set the closed-loop poles of $A - BK$) and the *estimator* problem (choosing a feedback L to set the closed-loop poles of $A - LC$). We have also talked at length about designing an optimal feedback K using the Linear Quadratic Regulator (LQR) problem, whose parameters are the state matrix A, input matrix B, state weighting matrix Q, and input weighting matrix R. Thus, one might expect to be able to identify an "optimal" (in some sense) estimator using the dual quantities: A^T, C^T, and some matrices $\tilde{Q}$ and $\tilde{R}$. This is easily done in Matlab:

```
K = lqr(A,B,Q,R)
L = (lqr(A',C',Q,R))'
```

But, what exactly is this "optimal" estimator? In what sense is it optimal? And how are the $\tilde{Q}$ and $\tilde{R}$ matrices supposed to be selected? This is the Linear Quadratic Estimator (LQE) problem, and is discussed below. (Notes adapted from Prof. How's 16.323 materials.)

You already knew how to design LQE estimators from Topic 14; the purpose of this recitation is to provide context, aiding in the selection of $\tilde{Q}$ and $\tilde{R}$ matrices.

1.2 Deriving LQE

Consider the standard state space model, now augmented with multiple sources of noise:

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} + \mathbf{w}, \tag{1}$$

$$\mathbf{y} = C\mathbf{x} + \mathbf{v}. \tag{2}$$

Here $\mathbf{w}$ is a *process noise*, modeling uncertainty in the system model (e.g., wind disturbance), while $\mathbf{v}$ is a *sensing noise*, modeling uncertainty in the measurements (e.g., a poor rate gyro). We typically assume that $\mathbf{w}(t)$ and $\mathbf{v}(t)$ have zero mean, such that $E[\mathbf{w}(t)] = E[\mathbf{v}(t)] = 0$. We will model the noises as uncorrelated, Gaussian white random noises, meaning there is no correlation between the noise at one time instant and another:

$$\begin{aligned} E[\mathbf{w}(t_1)\mathbf{w}(t_2)^T] &= R_{ww}(t_1)\delta(t_1 - t_2), \\ E[\mathbf{v}(t_1)\mathbf{v}(t_2)^T] &= R_{vv}(t_1)\delta(t_1 - t_2), \\ E[\mathbf{w}(t_1)\mathbf{v}(t_2)^T] &= 0. \end{aligned}$$

We write these noises as $\mathbf{w}(t) \sim \mathcal{N}(0, R_{ww})$ and $\mathbf{v}(t) \sim \mathcal{N}(0, R_{vv})$, where the notation $\mathcal{N}(a, P_a)$ indicates a Gaussian noise with mean a and covariance matrix P_a. The covariance represents the "spread" of the distribution – a large covariance corresponds to a broad distribution, representing a larger uncertainty. By contrast, a low covariance corresponds to a distribution tightly distributed around the mean.

Recall that our estimator takes the form

$$\dot{\hat{\mathbf{x}}} = A\hat{\mathbf{x}} + B\mathbf{u} + L(\mathbf{y} - \hat{\mathbf{y}}), \tag{3}$$

$$\hat{\mathbf{y}} = C\hat{\mathbf{x}}. \tag{4}$$

We form the estimation error ($\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$) dynamics, now incorporating the new noise terms in (1)-(2):

$$\begin{aligned}\dot{\tilde{\mathbf{x}}} &= [A\mathbf{x} + B\mathbf{u} + \mathbf{w}] - [A\hat{\mathbf{x}} + B\mathbf{u} + L(\mathbf{y} - \hat{\mathbf{y}})] \\ &= A\tilde{\mathbf{x}} + \mathbf{w} - L(C\mathbf{x} + \mathbf{v} - C\hat{\mathbf{x}}) \\ &= (A - LC)\tilde{\mathbf{x}} + \mathbf{w} - L\mathbf{v}\end{aligned}$$

This equation of the estimation error explicitly shows the conflict in the estimator design process. The estimator must balance between the speed of the estimator decay rate, governed by $\mathrm{Re}[\lambda_i(A-LC)]$, and the impact of the sensing noise $\mathbf{v}$ through the gain L. In particular, fast state reconstruction requires a rapid decay rate, which typically requires a large L. However, this tends to magnify the effect of $\mathbf{v}$ on the estimation process. (The effect of the process noise is always there.) The *Kalman filter* developed below provides an optimal balance between the two conflicting problems, for a given "size" of the process and sensing noises.

Presumably, our optimal estimator should minimize the estimation error $\tilde{\mathbf{x}}(t)$ over time in some sense. Suppose our objective is to design an estimator $\hat{\mathbf{x}}(t)$ which minimizes the trace of the *mean square value* of the estimation error:

$$\begin{aligned}\min \quad & J = \mathrm{trace}(Q(t)) \\ \text{s.t.} \quad & Q(t) = E\left[\{\mathbf{x}(t) - \hat{\mathbf{x}}(t)\}\{\mathbf{x}(t) - \hat{\mathbf{x}}(t)\}^T\right].\end{aligned}$$

If we further assume the estimator is a linear function of the measurements $\mathbf{y}$, it can be shown that the solution to this problem is the estimator (3)-(4), where

$$\begin{aligned}L(t) &= Q(t)C^T R_{vv}^{-1}, \quad Q(t) \succeq 0, \\ \dot{Q}(t) &= AQ(t) + Q(t)A^T + R_{ww} - Q(t)C^T R_{vv}^{-1} C Q(t).\end{aligned}$$

Note that $\hat{\mathbf{x}}(0)$ and $Q(0)$ are assumed to be known, such that this differential equation in Q can be solved forward in time. This is called the *Kalman-Bucy filter*, or the *linear quadratic estimator* (LQE).

An increase in $Q(t)$ corresponds to increased uncertainty in the state estimate. The differential equation in $\dot{Q}$ has three contributions:

- $AQ + QA^T$ is the homogeneous part
- R_{ww} is an increase attributed to the process noise
- $QC^T R_{vv}^{-1} CQ$ is a decrease attributed to measurements

The estimator gain $L(t) = Q(t)C^T R_{vv}^{-1}$ is a feedback on the *innovation*, $\mathbf{y} - \hat{\mathbf{y}}$. If the uncertainty about the state is high, then $Q(t)$ is large, and so the innovation is weighted more heavily (by increasing L). On the other hand, if the measurements are very accurate (such that R_{vv} is low), the measurements are also heavily weighted.

1.3 Steady-State LQE

As with the LQR problem, the covariance of the LQE quickly settles down to a steady-state value Q, independent of $Q(0)$. It is computed by solving the algebraic Riccati equation

$$0 = AQ + QA^T + R_{ww} - QC^T R_{vv}^{-1} CQ, \quad (5)$$

$$L = QC^T R_{vv}^{-1}. \quad (6)$$

This is the version of LQE we have referred to in this course, and that you will use in Lab 2.

Now, what are the assumptions on this LQE problem? (compare to the LQR assumptions)

1. The matrix R_{ww} must be positive semidefinite, i.e. $R_{ww} \succeq 0$. This should make sense, since noise cannot have "negative" variance. We may also write $B_w\mathbf{w}$ in (1) instead of $\mathbf{w}$, in which case we typically assume $R_{ww} \succ 0$. This formulation allows the dimension of $\mathbf{w}$ to differ from the dimension from $\mathbf{x}$, which can be useful.
2. The matrix R_{vv} must be positive definite, i.e. $R_{vv} \succ 0$. Even if your sensors are supposedly perfect, you should always incorporate a small positive value for this covariance; otherwise the last term in (5) blows up.
3. The solution Q to the algebraic Riccati equation is always symmetric, such that $Q^T = Q$.
4. If (A, B_w, C) is stabilizable and detectable, then the correct solution of the algebraic Riccati equation is the unique solution Q for which $Q \succeq 0$.
5. If (A, B_w, C) is also controllable, then $Q \succ 0$.

With this, the full duality with LQR is now in plain view:

$$\begin{aligned}
(\text{LQR} &\Leftrightarrow \text{LQE}) \\
A &\Leftrightarrow A^T \\
B &\Leftrightarrow C^T \\
K &\Leftrightarrow L^T \\
P &\Leftrightarrow Q \\
Q &\Leftrightarrow R_{ww} \\
R &\Leftrightarrow R_{vv} \\
C_z &\Leftrightarrow B_w
\end{aligned}$$

So, ultimately, how should R_{ww} and R_{vv} be selected? Ultimately, it comes down to which you trust more, your model or your measurements. (Similar to LQR, there is no point in scaling up both – all that matters is the relative values of the two matrices.) If you have a great system model measured by poor sensors, you would probably choose R_{vv} to be larger than R_{ww}. Similarly, if the dynamics are poorly modeled but well-observed, you would probably choose R_{ww} to be larger than R_{vv}. Despite the role of R_{ww} and R_{vv} as the noise covariances, for this course it is more useful to think of them simply as design knobs, much like Q and R in LQR.

Aside: Even though LQE can be performed in Matlab via the `lqr` command, using the previously-discussed duality, there actually is an `lqe` command you can use:

```
L = lqe(A,G,C,Q,R)
```

Here `A` $= A$, `G` $= B_w$, `C` $= C$, `Q` $= R_{ww}$, and `R` $= R_{vv}$.

16.30 / 16.31 Feedback Control Systems
Fall 2010