Topic #15

16.31 Feedback Control Systems

State-Space Systems

- Open-loop Estimators
- Closed-loop Estimators

- **Observer Theory (no noise) – Luenberger**
  IEEE TAC Vol 16, No. 6, pp. 596–602, December 1971.
- **Estimation Theory (with noise) – Kalman**

Estimators/Observers

- **Problem:** So far we have assumed that we have full access to the state $\mathbf{x}(t)$ when we designed our controllers.
  - Most often all of this information is not available.

- Usually can only feedback information that is developed from the sensors measurements.
  - Could try "output feedback"
  $$u = Kx \quad \Rightarrow \quad u = \hat{K}y$$
  - Same as the proportional feedback we looked at at the beginning of the root locus work.
  - This type of control is very difficult to design in general.

- **Alternative approach:** Develop a replica of the dynamic system that provides an "estimate" of the system states based on the measured output of the system.

- **New plan:**
  1. Develop estimate of $\mathbf{x}(t)$ that will be called $\hat{\mathbf{x}}(t)$.
  2. Then switch from $\mathbf{u}(t) = -K\mathbf{x}(t)$ to $\mathbf{u}(t) = -K\hat{\mathbf{x}}(t)$.

- Two key questions:
  - How do we find $\hat{\mathbf{x}}(t)$?
  - Will this new plan work?

Estimation Schemes

- Assume that the system model is of the form:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t)\ , \quad \mathbf{x}(0) \text{ unknown} \\ \mathbf{y}(t) &= C\mathbf{x}(t)\end{aligned}$$

where

1. A, B, and C are known.
2. $\mathbf{u}(t)$ is known
3. Measurable outputs are $\mathbf{y}(t)$ from $C \neq I$

- **Goal:** Develop a dynamic system whose state

$$\hat{\mathbf{x}}(t) = \mathbf{x}(t)$$

for all time $t \geq 0$. Two primary approaches:

  – Open-loop.

  – Closed-loop.

Open-loop Estimator

- Given that we know the plant matrices and the inputs, we can just perform a simulation that runs in parallel with the system

$$\dot{\hat{\mathbf{x}}}(t) = A\hat{\mathbf{x}}(t) + B\mathbf{u}(t)$$

- Then $\hat{\mathbf{x}}(t) \equiv \mathbf{x}(t)\ \forall\ t$ provided that $\hat{\mathbf{x}}(0) = \mathbf{x}(0)$

- **Major Problem:** We do not know $\mathbf{x}(0)$

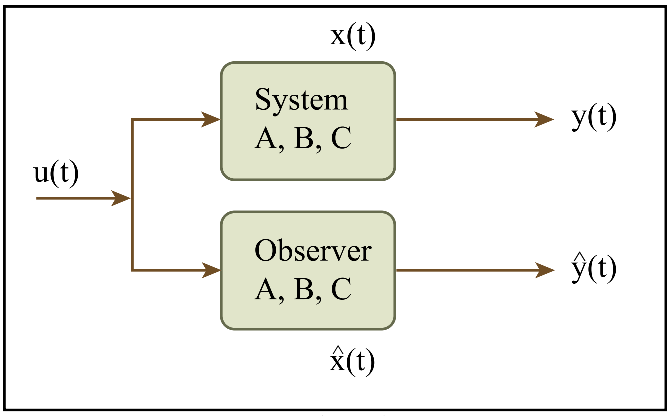

Figure by MIT OpenCourseWare.

- Analysis of this case:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t)\\ \dot{\hat{\mathbf{x}}}(t) &= A\hat{\mathbf{x}}(t) + B\mathbf{u}(t)\end{aligned}$$

- Define the **estimation error** $\tilde{\mathbf{x}}(t) = \mathbf{x}(t) - \hat{\mathbf{x}}(t)$. Now want $\tilde{\mathbf{x}}(t) = 0\ \forall\ t$. (But is this realistic?)

- Subtract to get:

$$\frac{d}{dt}(\mathbf{x}(t) - \hat{\mathbf{x}}(t)) = A(\mathbf{x}(t) - \hat{\mathbf{x}}(t)) \quad \Rightarrow \quad \dot{\tilde{\mathbf{x}}}(t) = A\tilde{\mathbf{x}}(t)$$

  which has the solution

$$\tilde{\mathbf{x}}(t) = e^{At}\tilde{\mathbf{x}}(0)$$

  – Gives the estimation error in terms of the initial error.

- Does this guarantee that $\tilde{\mathbf{x}}(t) = 0 \ \forall \ t$?
  Or even that $\tilde{\mathbf{x}}(t) \rightarrow 0$ as $t \rightarrow \infty$? (which is a more realistic goal).

  – Response is fine if $\tilde{\mathbf{x}}(0) = 0$. But what if $\tilde{\mathbf{x}}(0) \neq 0$?

- If A stable, then $\tilde{\mathbf{x}}(t) \rightarrow 0$ as $t \rightarrow \infty$, but the dynamics of the estimation error are completely determined by the open-loop dynamics of the system (eigenvalues of A).

  – Could be very slow.

  – No obvious way to modify the estimation error dynamics.

- Open-loop estimation does not seem to be a very good idea.

Closed-loop Estimator

- An obvious way to fix this problem is to use the additional information available:
  - How well does the estimated output match the measured output?

    Compare: $\mathbf{y}(t) = C\mathbf{x}(t)$ with $\hat{\mathbf{y}}(t) = C\hat{\mathbf{x}}(t)$
  - Then form $\tilde{\mathbf{y}}(t) = \mathbf{y}(t) - \hat{\mathbf{y}}(t) \equiv C\tilde{\mathbf{x}}(t)$

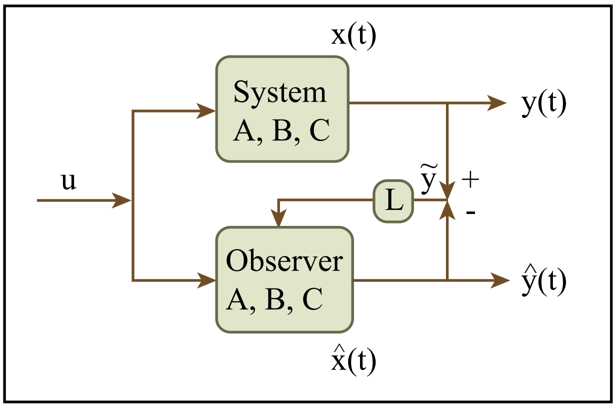

Figure by MIT OpenCourseWare.

- **Approach:** Feedback $\tilde{\mathbf{y}}(t)$ to improve our estimate of the state. Basic form of the estimator is:

$$\begin{aligned}\dot{\hat{\mathbf{x}}}(t) &= A\hat{\mathbf{x}}(t) + B\mathbf{u}(t) + \boxed{L\tilde{\mathbf{y}}(t)} \\ \hat{\mathbf{y}}(t) &= C\hat{\mathbf{x}}(t)\end{aligned}$$

  where L is the *user selectable gain matrix.*

- **Analysis:**

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

- So the closed-loop estimation error dynamics are now

$$\dot{\tilde{\mathbf{x}}}(t) = (A - LC)\tilde{\mathbf{x}}(t)$$

with solution

$$\tilde{\mathbf{x}}(t) = e^{(A-LC)t}\,\tilde{\mathbf{x}}(0)$$

- **Bottom line:** Can select the gain L to attempt to improve the convergence of the estimation error (and/or speed it up).
  - But now must worry about observability of the system model.

- Closed-loop estimator:

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

  - Which is a dynamic system with poles given by $\lambda_i(A - LC)$ and which takes the measured plant outputs as an input and generates an estimate of $\mathbf{x}(t)$.

Regulator/Estimator Comparison

- **Regulator Problem:**
  - Concerned with controllability of (A, B)

    For a controllable system we can place the eigenvalues of $A - BK$ arbitrarily.
  - Choose $K \in \mathcal{R}^{1\times n}$ (SISO) such that the closed-loop poles

$$\det(sI - A + BK) = \Phi_c(s)$$

    are in the desired locations.

- **Estimator Problem:**
  - For estimation, were concerned with observability of pair (A, C).

    For a observable system we can place the eigenvalues of $A - LC$ arbitrarily.
  - Choose $L \in \mathcal{R}^{n\times 1}$ (SISO) such that the closed-loop poles

$$\det(sI - A + LC) = \Phi_o(s)$$

    are in the desired locations.

- These problems are obviously very similar – in fact they are called **dual problems**.

Estimation Gain Selection

- The procedure for selecting L is very similar to that used for the regulator design process.
- Write the system model in **observer canonical** form

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -a_1 & 1 & 0 \\ -a_2 & 0 & 1 \\ -a_3 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

- Now very simple to form

$$\begin{aligned} A - LC &= \begin{bmatrix} -a_1 & 1 & 0 \\ -a_2 & 0 & 1 \\ -a_3 & 0 & 0 \end{bmatrix} - \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} -a_1 - l_1 & 1 & 0 \\ -a_2 - l_2 & 0 & 1 \\ -a_3 - l_3 & 0 & 0 \end{bmatrix} \end{aligned}$$

  – The closed-loop poles of the estimator are at the roots of

  $$\det(sI - A + LC) = s^3 + (a_1 + l_1)s^2 + (a_2 + l_2)s + (a_3 + l_3) = 0$$

  – Use Pole Placement algorithm with this characteristic equation.

- Note that the estimator equivalent of Ackermann's formula is that

$$L = \Phi_e(s)\mathcal{M}_o^{-1} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

- So we have the freedom to place the closed-loop poles as desired.
  – Task greatly simplified by the selection of the state-space model used for the design/analysis.

- Another approach:
  - Note that the poles of $(A-LC)$ and $(A-LC)^T$ are identical.
  - Also we have that $(A-LC)^T = A^T - C^T L^T$
  - So designing L^T for this transposed system looks like a standard regulator problem $(A-BK)$ where

$$\begin{array}{rcl} A & \Rightarrow & A^T \\ B & \Rightarrow & C^T \\ K & \Rightarrow & L^T \end{array}$$

So we can use

$$K_e = \texttt{acker}(A^T, C^T, P) \ , \quad L \equiv K_e^T$$

Estimators Example

- Simple system

$$
\begin{aligned}
A &= \begin{bmatrix} -1 & 1.5 \\ 1 & -2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \mathbf{x}(0) = \begin{bmatrix} -0.5 \\ -1 \end{bmatrix} \\
C &= \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad D = 0
\end{aligned}
$$

  – Assume that the initial conditions are not well known.

  – System stable, but $\lambda_{\max}(A) = -0.18$

  – Test observability:

$$
\texttt{rank} \begin{bmatrix} C \\ CA \end{bmatrix} = \texttt{rank} \begin{bmatrix} 1 & 0 \\ -1 & 1.5 \end{bmatrix}
$$

- Use open and closed-loop estimators

  – Since the initial conditions are not well known, use

$$
\hat{\mathbf{x}}(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}
$$

- Open-loop estimator:

$$
\begin{aligned}
\dot{\hat{\mathbf{x}}}(t) &= A\hat{\mathbf{x}}(t) + B\mathbf{u}(t) \\
\hat{\mathbf{y}}(t) &= C\hat{\mathbf{x}}(t)
\end{aligned}
$$

- Typically simulate both systems together for simplicity

- Open-loop case:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t)\\ \mathbf{y}(t) &= C\mathbf{x}(t)\\ \dot{\hat{\mathbf{x}}}(t) &= A\hat{\mathbf{x}}(t) + B\mathbf{u}(t)\\ \hat{\mathbf{y}}(t) &= C\hat{\mathbf{x}}(t)\end{aligned}$$

$$\Rightarrow \begin{bmatrix}\dot{\mathbf{x}}(t)\\ \dot{\hat{\mathbf{x}}}(t)\end{bmatrix} = \begin{bmatrix}A & 0\\ 0 & A\end{bmatrix}\begin{bmatrix}\mathbf{x}(t)\\ \hat{\mathbf{x}}(t)\end{bmatrix} + \begin{bmatrix}B\\ B\end{bmatrix}\mathbf{u}(t)\ , \quad \begin{bmatrix}\mathbf{x}(0)\\ \hat{\mathbf{x}}(0)\end{bmatrix} = \begin{bmatrix}-0.5\\ -1\\ 0\\ 0\end{bmatrix}$$

$$\begin{bmatrix}\mathbf{y}(t)\\ \hat{\mathbf{y}}(t)\end{bmatrix} = \begin{bmatrix}C & 0\\ 0 & C\end{bmatrix}\begin{bmatrix}\mathbf{x}(t)\\ \hat{\mathbf{x}}(t)\end{bmatrix}$$

- Closed-loop case:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t)\\ \dot{\hat{\mathbf{x}}}(t) &= (A - LC)\hat{\mathbf{x}}(t) + B\mathbf{u}(t) + LC\mathbf{x}(t)\end{aligned}$$

$$\Rightarrow \begin{bmatrix}\dot{\mathbf{x}}(t)\\ \dot{\hat{\mathbf{x}}}(t)\end{bmatrix} = \begin{bmatrix}A & 0\\ LC & A - LC\end{bmatrix}\begin{bmatrix}\mathbf{x}(t)\\ \hat{\mathbf{x}}(t)\end{bmatrix} + \begin{bmatrix}B\\ B\end{bmatrix}\mathbf{u}(t)$$

- Example uses a strong $\mathbf{u}(t)$ to shake things up

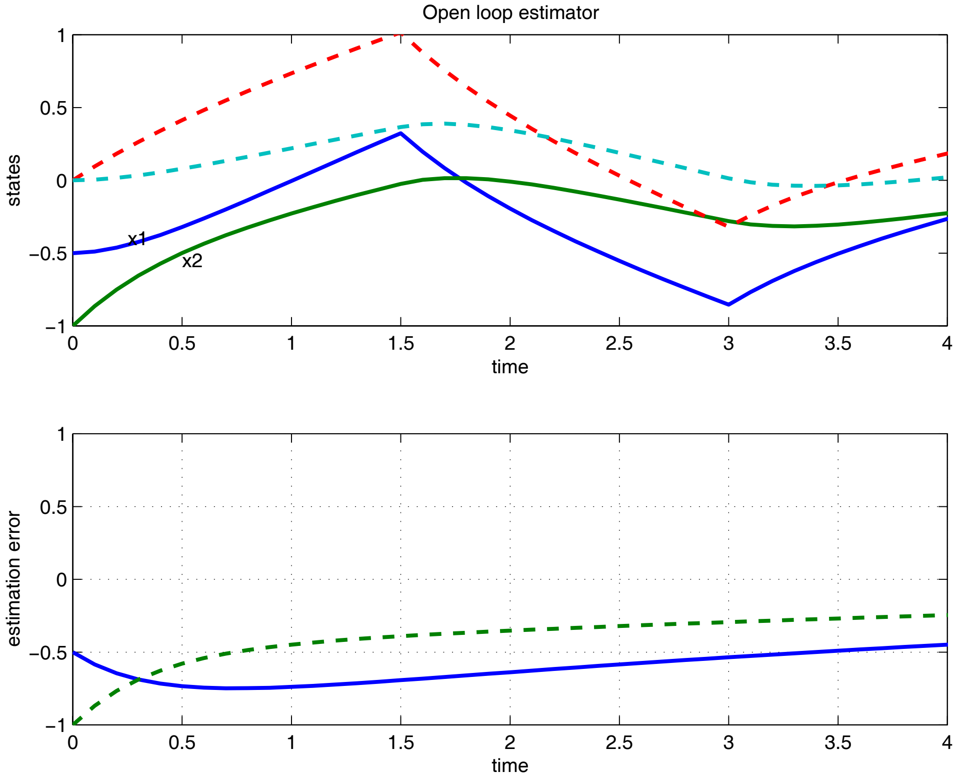

Figure 1: Open-loop estimator. Estimation error converges to zero, but very slowly.

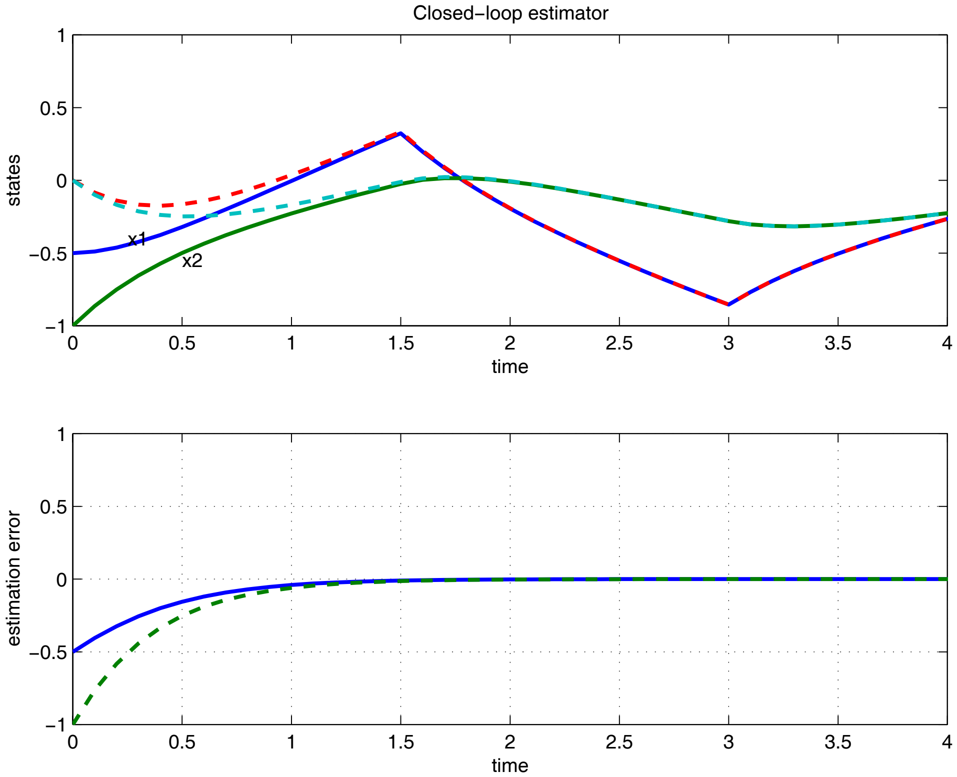

Figure 2: Closed-loop estimator. Convergence looks much better.

Where to put the Estimator Poles?

- Location heuristics for poles still apply – use Bessel, ITAE, ...
  - Main difference: probably want to make the estimator faster than you intend to make the regulator – should enhance the control, which is based on $\hat{x}(t)$.
  - ROT: Factor of 2–3 in the time constant $\zeta\omega_n$ associated with the regulator poles.

- **Note:** When designing a regulator, were concerned with "bandwidth" of the control getting too high ⇒ often results in control commands that *saturate* the actuators and/or change rapidly.

- Different concerns for the estimator:
  - Loop closed inside computer, so saturation not a problem.
  - However, the measurements y are often "noisy", and we need to be careful how we use them to develop our state estimates.

⇒ **High bandwidth estimators** tend to accentuate the effect of sensing noise in the estimate.

- State estimates tend to "track" the measurements, which are fluctuating randomly due to the noise.

⇒ **Low bandwidth estimators** have lower gains and tend to rely more heavily on the plant model

- Essentially an open-loop estimator – tends to ignore the measurements and just uses the plant model.

Final Thoughts

- Note that the feedback gain L in the estimator only stabilizes the estimation error.
  - If the system is unstable, then the state estimates will also go to ∞, with zero error from the actual states.

- Estimation is an important concept of its own.
  - Not always just "part of the control system"
  - Critical issue for guidance and navigation system

- Can develop an optimal estimate as well
  - More complete discussion requires that we study stochastic processes and optimization theory.
  - More later

- **Estimation is all about which do you trust more: your measurements or your model.**

Estimator Codes (examp1.m)

```
% Examples of estimator performance
%
% Jonathan How
% Oct 2007
%
% plant dynamics
%
a=[-1 1.5;1 -2];
b=[1 0]';
c=[1 0];
d=0;

% estimator gain calc
%
l=place(a',c',[-3 -4]);l=l'

% plant initial cond
xo=[-.5;-1];
% extimator initial cond
xe=[0 0]';

t=[0:.1:10];

% inputs
%
u=0;u=[ones(15,1);-ones(15,1);ones(15,1)/2;-ones(15,1)/2;zeros(41,1)];

%
% open-loop extimator
%
A_ol=[a zeros(size(a));zeros(size(a)) a];
B_ol=[b;b];
C_ol=[c zeros(size(c));zeros(size(c)) c];
D_ol=zeros(2,1);

%
% closed-loop extimator
%
A_cl=[a zeros(size(a));l*c a-l*c];
B_cl=[b;b];
C_cl=[c zeros(size(c));zeros(size(c)) c];
D_cl=zeros(2,1);

[y_cl,x_cl]=lsim(A_cl,B_cl,C_cl,D_cl,u,t,[xo;xe]);
[y_ol,x_ol]=lsim(A_ol,B_ol,C_ol,D_ol,u,t,[xo;xe]);

figure(1);clf;subplot(211)
set(gca,'FontSize',14)
plot(t,x_cl(:,[1 2]),t,x_cl(:,[3 4]),'--','LineWidth',2);axis([0 4 -1 1]);
title('Closed-loop estimator');ylabel('states');xlabel('time')
text(.25,-.4,'x_1','FontSize',14);text(.5,-.55,'x_2','FontSize',14);subplot(212)
plot(t,x_cl(:,[1 2])-x_cl(:,[3 4]),'LineWidth',2)
setlines;axis([0 4 -1 1]);grid on
ylabel('estimation error');xlabel('time')

figure(2);clf;subplot(211)
set(gca,'FontSize',14)
plot(t,x_ol(:,[1 2]),t,x_ol(:,[3 4]),'--','LineWidth',2);axis([0 4 -1 1])
title('Open loop estimator');ylabel('states');xlabel('time')
text(.25,-.4,'x_1','FontSize',14);text(.5,-.55,'x_2','FontSize',14);subplot(212)
plot(t,x_ol(:,[1 2])-x_ol(:,[3 4]),'LineWidth',2)
setlines;axis([0 4 -1 1]);grid on
ylabel('estimation error');xlabel('time')

print -dpng -r300 -f1 est11.png
print -dpng -r300 -f2 est12.png
```