1 State Estimation and Observers

In practice no perfect observation of the system state is available, either because it is costly, technically unfeasible or because the measurements quality is low. In this case state feedback control laws,

$$u(t) = u(x(t)), \quad t \geq 0$$

as derived in previous lectures is often impractical. There is a need for a systematic approach for the evaluation or estimation of the system state using the information available.

One natural approach is to compute an estimate \( \hat{x} \) of the state \( x \) and apply the feedback:

$$u(t) = u(\hat{x}(t)), \quad t \geq 0$$

The idea that a stabilizing controller can consist of a state estimator plus (estimated) state feedback is called the separation principle. For linear systems this is a valid approach. Indeed, given a linear time invariant system

\[
\begin{align*}
\dot{x} &= Ax + Bu \\
y &= Cx
\end{align*}
\]

Consider the observer

\[
\begin{align*}
\dot{\hat{x}} &= A\hat{x} + Bu + L(y - C\hat{x}) \\
&= (A - LC)\hat{x} + Bu + Ly \\
&= (A - LC)x + BKx + Ly
\end{align*}
\]

Denote

$$e = x - \hat{x}$$

We see that

\[
\begin{align*}
\dot{e} &= (Ax + Bu) - (A - LC)\hat{x} - Bu - LCx \\
&= (A - LC)e \\
&= (A - LC)e
\end{align*}
\]

Let

$$u = -K\hat{x} = -K(x - e)$$
Then
\[
\begin{bmatrix}
\dot{x} \\
\dot{e}
\end{bmatrix} =
\begin{bmatrix}
A - BK & BK \\
0 & A - LC
\end{bmatrix}
\begin{bmatrix}
x \\
e
\end{bmatrix}
\]

Thus if \( A - BK \) and \( A - LC \) are stable matrices, the resulting closed loop system is also stable.

Unfortunately, for nonlinear systems this approach does not work in general. The problem is that generally speaking it is not possible to estimate the error dynamics – i.e. the dynamics of the difference between the actual state and the estimated state.

There are several approaches to state estimation that may be applied:

1. Extended Kalman Filter (EKF): Extension of linear Kalman Filter
2. Recursive Prediction Error (RPE): Based on the sensitivity equation
3. Unscented Kalman Filter (UKF): Mix of Monte-Carlo with Kalman Filter
4. Moving Horizon Estimation (MHE)
5. High Gain Observers

In the sequel we present techniques 1, 4 and 5 for estimating the state of a nonlinear system.

### 1.1 Least Squares Estimation of Constant Vectors

We consider the process model
\[
y = Cx + w, \quad x \in \mathbb{R}^n, \quad y, w \in \mathbb{R}^p, n \geq p.
\]

with \( w \) denotes white noise.

The goal is to compute the best estimate \( \hat{x} \) of \( x \) using process measurements \( y \). Given \( \hat{x} \) we can give an estimate \( \hat{y} \) of the output by
\[
\hat{y} = C\hat{x}
\]

Then we define the residual
\[
\epsilon_y = \hat{y} - y
\]
We want to obtain the optimal estimate in the sense that the quadratic function

\[ J[\hat{x}] = 0.5 \epsilon_y^T \epsilon_y \]

It is easily derived that this optimal estimate is given by

\[ \hat{x} = (C^T C)^{-1} C^T y \]

Where the matrix inversion is made in the sense of the so called “pseudo inverse”. Averaging effect takes place when the dimension of the output is larger than the dimension of the state.

**Example 1** Consider a process modelled by the simple linear relationship

\[ y = Cx + w; \quad C = \begin{pmatrix} -5 & 1 \\ 0 & 1 \end{pmatrix} \]

The magnitude \( y \) is measured directly and is affected by white noise \( w \). The actual value of \( x \) and needs to be reconstructed from the data series

\[ Y = [y_1; y_2; \ldots; y_N] = [Cx + w_1; Cx_2 + w_2; \ldots; Cx + w_N] \]

The optimal estimate \( \hat{x} \) is then given by

\[ \hat{x} = KY, K = (H^T H)^{-1} H^T; \quad H = [C; C; \ldots; C]; \]

Figure 1 shows how the estimate \( \hat{x} \) approaches the true value \( x = [5; 5] \) as \( N \), the number of measurements, grows.

Our second example shows how to use the results above in the context of dynamical system parameter estimation.

**Example 2** Consider the equations of a simple heat exchanger. Steam at temperature \( T_{si} \) enters the system and passes its energy to a gas that enters the system at temperature \( T_{gi} \). The fluids are separated by a metal inter-face, which has temperature \( T_m \). At the output, the steam and the gas have temperatures

\[ \dot{T}_g = \alpha_g (T_m - T_g) + fg(T_{gi} - T_g) \quad (1) \]
\[ \dot{T}_m = \alpha_g (T_g - T_m) + \alpha_s (T_s - T_m) \quad (2) \]
\[ \dot{T}_s = \alpha_s (T_m - T_s) + fs(T_{si} - T_s) \quad (3) \]
Figure 1: Static Estimation. Note how accuracy improves as $N$ grows.
Our task is to estimate the values of $\alpha_g$ and $\alpha_s$ from the measurements $T_{gi}, T_g, T_{si}, T_s$ and $T_m$.

We proceed by realizing that for any given point in time we can create a linear relationship

$$y = Cx + w$$

where

$$y = \begin{pmatrix} \dot{T}_g - fg(T_{gi} - T_g) \\ \dot{T}_m \\ \dot{T}_s - fs(T_{si} - T_s) \end{pmatrix}$$

and

$$C = \begin{pmatrix} (T_m - T_g) & 0 \\ (T_g - T_m) & (T_s - T_m) \\ 0 & (T_m - T_s) \end{pmatrix}$$

$$x = \begin{pmatrix} \alpha_g \\ \alpha_s \end{pmatrix}$$

Note that the derivatives $\dot{T}_g, \dot{T}_m, \dot{T}_s$ in $y$ need to be reconstructed from the temperature measurements. Due to measurement noise, one must expect that these "reconstructed" derivatives are noisy measurements of the true derivatives.

Figure 2 shows how the estimates $\alpha_{ge}, \alpha_{se}$ approach the true values $[3; 1]$ as $N$, the number of measurements, grows.

Remark 1 The static method is designed to introduce averaging effects over the whole data set $y$. Thus, in practice one needs methods to manage which data is presented to the algorithm so that results remain meaningful. For instance, if the process has large steady state phases interrupted by periods of transient behavior, then one must make sure that the steady state phases are not presented to the algorithm in the same data set because otherwise the result will be the average of two different steady state modes, which is not what is desired.

For example, the results in Figure 2 were obtained by resetting the time arrays after the change of parameters, i.e. at times $t = 50, 100$. Without this manipulation the observer does not deliver meaningful results.
Figure 2: Parametric Static Estimation. Note how accuracy improves as $N$ grows.
1.2 Weighted Least Squares Estimator

The measurement error statistical properties can change from one point to another thus it makes sense to give different weight to different points while seeking the optimum. More importantly, as discussed in the previous section, when designing an observer there is always need for methods to fine tune the significance of different values in the method trade off’s between model and noise. For instance, the designer may want to make sure that old values are considered less important than recent ones.

If the weights are distributed according to the equation

\[ J[\hat{x}] = 0.5 \epsilon_y^T \epsilon_y, \quad \epsilon_y = N^{-1}(\hat{y} - y). \]

Making the derivative of this function equal to zero, we derive the following formula for optimal estimate

\[ \hat{x} = (C^T S^{-1} C)^{-1} C^T S^{-1} y; \quad S = N^{-T} N^{-1}. \]

This formula is used exactly as in the previous section. Care is of course needed in the choice of the weights \( N \).

1.3 Propagation of the State Estimate and its Uncertainty

The state of a dynamic system changes with time and with its statistical properties. State estimators must take this fact into account. If the initial conditions, the inputs and the dynamics were perfectly known then it would be enough to integrate forward the system equations to have a good estimate of the system state. Unfortunately, this is never the case and uncertainty will always play a role. In this section we must learn to propagate the statistical properties of the state, so that we can use them when calculating the optimal estimator.

Let us assume we have a linear dynamical system

\[ \dot{x} = Ax + Bu + Lw \]

Let us replace deterministic values by the “means”. Then

\[ E[\dot{x}] = AE[x()] + BE[u()] + LE[w()]. \]
Via simple manipulations this implies
\[
\dot{m}(t) = Am(t) + Bu(t)
\]
\[
\dot{P}(t) = AP + PA^T + LQL^T.
\]

Where
1. \( m(t) = E[x(t)] \)
2. \( P(t) = E[(x(t) - m(t))^T(x(t) - m(t))] \)
3. \( Q \) is covariance of \( w \).

Note how the uncertainty (covariance) of the mean of the state will always grow due to the driving force \( LQL^T \).

### 1.4 Kalman Filter

A recursive optimal filter propagates the conditional probability density function from one sampling time to the next, incorporating measurements and statistical of the measurements in the estimate calculation.

The Kalman Filter consists of the following steps

1. State Estimate Propagation
2. State Covariance Propagation
3. Filter Gain Calculation
4. State Estimate Update using the newest measurements
5. State Covariance Update using the newest measurements

Steps 1 and 2 were considered before. Step 3 (calculation of Filter gain) is made using ideas of the first part of this lecture namely weighted least squares. The result is

\[
K = P_k C^T R^{-1}.
\]

where \( R \) is the covariance of the measurement noise. The formula shows a tradeoff between the measurement noise statistics \( R \) and the quality of our estimation as given by \( P \).

The Filter gain \( K \) is used in the Steps 4 and 5 as follows.

1. \( \dot{x}_k = x_k + K(y - C\dot{y}) \)
2. \( P_k = [P_k^{-1} + C_k R_k^{-1} C_k^T]^{-1} \).
1.5 Extended Kalman Filter

The system is now nonlinear and modifications to the schema above are needed. They are

- Steps 1 is made with the true nonlinear equations
- Steps 2 to 5 are made with linearization of the linear system at the current estimate $\hat{x}$.

A summary of the formulae is given in Figure 1. The same formalism can be used to estimate parameters. Indeed, one can augment the natural system states with states representing the parameters. The dynamic equation for the parameter vector is chosen to be zero, i.e. $\dot{p} = 0 + w_p$, where $w_p$ has covariance $Q_p$. Substituting these expressions in the standard EKF formulate produces the algorithm represented in Figure 4.

**Example 3** We consider the same heat exchanger, but now we are prepared to confront the realistic case where all measurements are noisy and thus cal-
Calculating the derivatives accurately is not realistic. Also, we should drop the assumption that we know when the heat exchanger changes the operational modes. Under these conditions, the static estimation of parameters cannot be used because both we would not have the elements to construct the matrix $C$ used in Example 2.

In this more realistic case we resort to use the Extended Kalman Filter, whereas we will estimate the states $T_g, T_m, T_s$ and the parameters $\alpha_g$ and $\alpha_s$. The equations remain the same, i.e. Equations 1. The important non trivial step is to create dynamic equations for the parameters. In this case it is enough

\[
\dot{p}_g = 0 \\
\dot{p}_s = 0
\]

and

\[
Q_{pp} = \begin{pmatrix} c_1 & 0 \\ 0 & c_2 \end{pmatrix}, \\
Q_{xp} = 0^{2 \times 3}.
\]
Note that now there is no need for manipulating the arrays: the algorithm is recursive and reacts correctly to operational mode changes.

1.6 Moving Horizon Estimation

In analogy to Model Predictive Control, one can think of the observation problem as one of optimization under a finite horizon. Variables for optimization would be in this case the system states, and parameters if parameter estimation is also required. This approach has the advantage that Model Predictive Control formalism can be reused to deal with this task.

In concrete terms, the "attack plan" is to collect

1. the history of actuators moves and measurements  $u_M(\tau), y_M(\tau), \tau \in \mathbb{R}$
nonlinear systems and control — spring 2015

\[ t-M : t \]

2. the plant model, including the constraints

with which the observer calculates the trajectory of states \( x(\tau), \tau \in [t-M : t] \) that better “explains” the given history and the model.

In a somewhat simplified notation the problem being solved is one of minimization of the function

\[
J[x(t - M), \ldots, x(0)] = \sum_{t=-M}^{t-1} \{[x(t+1) - f(x(t), u_M(t))]_{Q_x}^2 + [y_M(t) - h(x(t), u_M(t))]_{Q_y}^2 \}
\]

Tuning parameters are \( q_x, q_y > 0 \), which denote the degree of trust on the existent measurements (so called measurement noise level) and in the model (process noise level). For large estimation horizon \( M \), this method converges towards a Kalman Filter.

**Remark 2** Moving Horizon Estimation allows keeping the problem constraints explicitly, which might be important in application for the sake of eliminating ambiguity.

**Remark 3** (Findeisen, 2003): Stability results of MPC and MHE can be obtained assuming that

1. MHE can be made to converge very fast: given error bound is satisfied after a given number of iterations

2. MPC has a continuous value function: NMPC robust to small perturbations.

Neither of these facts are obvious to prove for a given system, but they show in what direction ones needs to work to guarantee stability.

Industrial implementations of this method hide complexity from the user. All is required it is to attach to the model those variables containing the current values of the measurement and actuators. Industrial grade systems automatically:

- collect the data needed, i.e. the values of actuators, measurements and model parameters,
create the associated mathematical programming problem,
calculate the estimation of the states as required,
pass these values to the controller for calculation of the optimal moves.

This capability can also be used in stand-alone fashion for construction of model based soft sensors or even simply for noise filtering purposes. One example of this usage is the implementation of material balancing applications, where the values of certain magnitudes in the process (eg, metal content and/or reagent usage) are reconstructed by looking at magnitudes measured at other points of the circuit and a (simple) mathematical model of the installation.

1.7 High Gain Observers

Another approach to estimate the state is to consider a High Gain Observer. Consider the system

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \phi(x, u) \\
y &= x_1
\end{align*}
\]

Suppose that the state feedback \( u = \gamma(x) \) is a locally Lipschitz state feedback control that stabilizes the origin \( x=0 \) of the closed loop system:

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \phi(x, \gamma(x)) \\
y &= x_1
\end{align*}
\]

To implement this feedback control using only \( y \), use the following estimation scheme:

\[
\begin{align*}
\dot{\hat{x}}_1 &= \hat{x}_2 + h_1(y - \hat{x}_1) \\
\dot{\hat{x}}_2 &= \phi_0(\hat{x}, \gamma(\hat{x})) + h_1(y - \hat{x}_1)
\end{align*}
\]

Where \( \phi_0(x, u) \) is a nominal model for \( \phi(x, u) \). The estimation error \( \tilde{x} = x - \hat{x} \) then satisfies

\[
\begin{align*}
\dot{\tilde{x}}_1 &= -h_1 \tilde{x}_1 + \tilde{x}_2 \\
\dot{\tilde{x}}_2 &= -h_2 \tilde{x}_1 + \delta(x, \hat{x})
\end{align*}
\]

where

\[
\delta(x, \hat{x}) = \phi(x, \gamma(\hat{x})) - \phi_0(\hat{x}, \gamma(\hat{x})).
\]
We want \((h_1, h_2)\) to be such that \(\tilde{x}(t) \to 0\), and if \(\delta \equiv 0\), this is easily achieved by making the matrix

\[
A_0 = \begin{bmatrix}
-h_1 & 1 \\
-h_2 & 0
\end{bmatrix}
\]

to be stable. For \(\delta \neq 0\), we design \((h_1, h_2)\) in such a way that the transfer function

\[
G_0 : \delta \mapsto \tilde{x}
\]

is as small as possible, i.e. that the gain of the system is as small as possible. In this case:

\[
G_0 = \frac{1}{s^2 + h_1 s + h_2} \begin{bmatrix}
1 \\
s + h_1
\end{bmatrix},
\]

which can be made arbitrarily small if \(h_2 >> h_1 >> 1\). In particular if

\[
h_1 = \frac{\alpha_1}{\varepsilon}, h_2 = \frac{\alpha_2}{\varepsilon^2}, \varepsilon << 1,
\]

Then

\[
\lim_{\varepsilon \to 0} \|G_0(s)\| = 0
\]

Unfortunately there is a problem, the \((2,1)\) term of \(\exp(A_0 t)\) grows as \(h_1, h_2\) grow. This gives a peaking phenomenon. This means that the estimates of \(\tilde{x}\) will become arbitrarily large for some period of time. This will make \(u = \gamma(\hat{x})\) large, and may drive the system state out of the region of attraction – the region where the error dynamics are stable.

This situation may be overcome by saturating the control action. Since the peaking phenomena is of short duration, we assume that the state estimate will converge to the system state faster than the system diverges, and then the estimated state feedback will be sufficient to stabilize the system.

For illustration purposes, let us consider the system:

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= x_2^3 + u \\
y &= x_1
\end{align*}
\]

A stabilizing state feedback is given by:

\[
u = \gamma(x) = -x_2^3 - x_1 - x_2
\]
The output feedback controller is then given by:

\[
\begin{align*}
\dot{x}_1 &= \dot{x}_2 + \left(\frac{2}{\varepsilon}\right) (y - \hat{x}_1) \\
\dot{x}_2 &= -\dot{x}_1 - \dot{x}_2 + \left(\frac{1}{\varepsilon^2}\right) (y - \hat{x}_1) \\
u &= \text{sat} \left(-\dot{x}_2 - \dot{x}_1 - \dot{x}_2\right)
\end{align*}
\]

The following plots compare the behaviour of the system with state feedback, with output feedback and with saturated output feedback for value of \(\varepsilon\) equal to 0.2 and 0.1. In Figure 6 and Figure 7 we see how for smaller values of \(\varepsilon\), the transient is faster but larger, leading to earlier instabilities of the unsaturated system, but faster convergence of the estimated state to the actual state for the saturated system.

In particular, note how for the case \(\varepsilon = 0.1\) depicted Figure 7, the brief transient is sufficient to drive the controller without saturation into an unstable region (the simulation stops around 0.9s), while in the case with saturation, the controller performs almost as well as the state feedback controller.

Further, as \(\varepsilon \to 0\) the region of attraction under output feedback approaches the region of attraction under state feedback. This holds whenever \(\gamma(x)\) is a globally bounded stabilizing function. Indeed, as \(\varepsilon \to 0\) the time in saturation becomes smaller. Eventually this time is so small that the system state does not move during this time. In general we have the following result:

**Theorem 1** Consider the SISO system

\[
\begin{align*}
\dot{x} &= A_C x + B_C \phi(x, u) \\
y &= C_C x
\end{align*}
\]

Assume that we have a stabilizing state feedback controller:

\[
\begin{align*}
\dot{v} &= \Gamma(v, x) \\
u &= \gamma(v, x)
\end{align*}
\]

, with \(g\) and \(G\) globally bounded.

Consider then the output feedback controller:

\[
\begin{align*}
\dot{v} &= \Gamma(v, \hat{x}) \\
\dot{x} &= A_C \hat{x} + B \phi_0(\hat{x}, u) + H (y - C_C \hat{x}) \\
u &= \gamma(v, \hat{x})
\end{align*}
\]

Where
Figure 6: High Gain Observer Plots for $\epsilon = 0.2$
Figure 7: High Gain Observer Plots for $\epsilon = 0.1$
\[ H = \begin{bmatrix} \alpha_1/\varepsilon \\ \alpha_2/\varepsilon^2 \\ \vdots \\ \alpha_\rho/\varepsilon^\rho \end{bmatrix} \in \mathbb{R}^\rho, \text{ (r the relative degree)} \]

and \( \varepsilon > 0 \), and \( \alpha_1, \alpha_2, \ldots, \alpha_\rho \) are such that the polynomial

\[ s^\rho + \alpha_1 s^{\rho-1} + \ldots + \alpha_\rho = 0 \]

has all its roots in the complex left half plane. Then \( \exists \varepsilon_0 > 0 \) such that for all \( \varepsilon < \varepsilon_0 \), the closed loop system is exponentially stable.

**Remark 4** In this formulation we have neglected the unobserved zero dynamics. If they were to exist, then we need to assume that the original system is minimum phase. Then the result will hold.