

# Lecture 3: Model Uncertainty and State Estimation

## I. Uncertainty Modeling

- **Initial Remarks: Objective Statement and Stochastic Process**

One of the main reasons for control is to **suppress the effect of disturbances** on key process outputs. A model is needed to predict the influence of the disturbance on the outputs on the basis of measured signals. For **unmeasured disturbances, stochastic models are used**.

We introduce stochastic models for disturbances and show how to integrate them into deterministic system models for estimation and control. We discuss how to construct models of the form:

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

where  $w(k)$  is a disturbance signal.

Note: here we just concatenate all the disturbance in a single  $w(k)$  instead of using different notation of  $w_1(k), w_2(k)$  so that the representation would be simpler and cleaner, the actual content of disturbance can be determined by  $F$  and  $G$ .

### Definition (Stochastic processes):

Stochastic processes are the mathematical tool used to model uncertain signals. A discrete-time stochastic process is a sequence of **random variables**:

$$\{w(0), w(1), w(2)\}$$

The realization of the process is uncertain. We can model a stochastic process **via its probability distribution**. In general, one must specify the **joint probability distribution function (pdf)** for the entire time sequence  $\mathbb{P}(w(0), w(1), \dots)$

**Problem:** Stochastic processes are modeled using data: Estimating the joint pdf usually is intractable.

**Solution:** Thus the normal distribution assumption is often made: only models of the **mean and the covariance are needed (two parameters)**

### Normal Stochastic Processes

Joint pdf is a normal distribution, completely defined by its mean and covariance function:

$$\begin{aligned}\mu_w(k) &= \mathbb{E}\{w(k)\} \\ R_w(k, \tau) &= \mathbb{E}\{w(k+\tau)w^T(k)\} - \mu_w(k+\tau)\mu_w^T(k)\end{aligned}$$

Stationary if  $\mu_w(k) = \mu_w$  and  $R_w(k, \tau) = R_w(\tau)$

Special case: Normal **white noise** stochastic process  $\varepsilon(k)$ :

$$\begin{aligned}\mu_\varepsilon &= 0 \\ R_\varepsilon(k, \tau) &= \begin{cases} R_\varepsilon & \text{if } \tau = 0 \\ 0 & \text{otherwise} \end{cases}\end{aligned}$$

Note:

1. Typically data are used to estimate  $\mu_w(k)$  and  $R_w(k, \tau)$
2. Since jointly normally distributed and uncorrelated over time, white noise stochastic process  $\varepsilon(k)$  is independent of time

- **Modeling Disturbance using State Space Descriptions**

### Stationary Case

For stationary case, we **model the stochastic process as the output  $w(k)$  of a linear system driven by normal white noise  $\varepsilon(k)$** . It will turn out that  $w(k)$  is a normal stochastic process and  $\mu_w(k)$ ,  $R_w(k, \tau)$  can be chosen through  $A_w, B_w, C_w$ . In the stationary case and using a state space description we have:

$$\begin{aligned}x_w(k+1) &= A_w x_w(k) + B_w \varepsilon(k) \\ w(k) &= C_w x_w(k) + \varepsilon(k)\end{aligned}$$

where:

$x_w$  is an **additional state** introduced to model the linear system's response to white noise.

All the eigenvalues of  $A_w$  lie **strictly inside the unit circle**.

Note: additional states **are not unique**, so as  $A_w, B_w, C_w$ , since we only specify the input and output.

The output  $w(k)$  of the stochastic process system with white noise  $\varepsilon(k)$  and stable  $A_w$  has the following properties:

$$\begin{aligned}\mathbb{E}\{x_w(k)\} &= A_w \mathbb{E}\{x_w(k-1)\} = A_w^k \mathbb{E}\{x_w(0)\} \\ \mathbb{E}\{x_w(k)x_w^T(k)\} &= A_w \mathbb{E}\{x_w(k-1)x_w^T(k-1)\} + B_w R_\varepsilon B_w^T \\ \mathbb{E}\{x_w(k+\tau)x_w^T(k)\} &= A_w^\tau \mathbb{E}\{x_w(k)x_w^T(k)\}\end{aligned}$$

From this one can deduce that:

$$\begin{aligned}\bar{w} &= \lim_{k \rightarrow \infty} \mathbb{E}\{w(k)\} = \lim_{k \rightarrow \infty} \mathbb{E}\{x_w(k)\} = \lim_{k \rightarrow \infty} A_w^k \mathbb{E}\{x_w(0)\} = 0 \\ R_w(\tau) &= \lim_{k \rightarrow \infty} \mathbb{E}\{w(k+\tau)w^T(k)\} = C_w A_w^\tau \bar{P}_w C_w^T + C_w A_w^{\tau-1} B_w R_\varepsilon\end{aligned}$$

where

$$\bar{P}_w = A_w \bar{P}_w A_w^T + B_w R_\varepsilon B_w^T$$

i.e.  $\bar{P}_w$  is a positive semi-definite solution to a Lyapunov equation. These relations can be used in order to determine  $A_w, B_w, C_w$  given certain covariance  $R_w(\tau)$


Note:

What if eigenvalues of  $A_w$  lie outside the unit circle? We can't do control since the noise system we model is not stable (the additional states would increase exponentially).

What if eigenvalues of  $A_w$  lie just on the unit circle? The states would grow, but not exponentially, because we just integrate them over time. We can still deal with it, with nonstationary case stated below.

## Nonstationary Case

If a disturbance signal has “persistent” characteristics (exhibiting shifts in the mean), it is not appropriate to model it with a stationary stochastic process. For example, controller design based on stationary stochastic processes will generally lead to offset (since in stationary case, the mean output noise would finally go to zero, meaning the disturbance would disappear in the end).

In this case one can superimpose the output of a linear system driven by  **integrated white noise**  $\varepsilon_{\text{int}}(k)$  to the stationary signal:

$$\varepsilon_{\text{int}}(k+1) = \varepsilon_{\text{int}}(k) + \varepsilon(k)$$

The state space description is then:

$$\begin{aligned}x_w(k+1) &= A_w x_w(k) + B_w \varepsilon_{\text{int}}(k) \\w(k) &= C_w x_w(k) + \varepsilon_{\text{int}}(k)\end{aligned}$$

However, we want our description to be in the standard form, i.e. in  $\varepsilon(k)$  instead of  $\varepsilon_{\text{int}}(k)$ . So the state space description can be rewritten, using differenced variables, as:

$$\begin{aligned}x_w(k+1) &= A_w x_w(k) + B_w \varepsilon_{\text{int}}(k) \\x_w(k) &= A_w x_w(k-1) + B_w \varepsilon_{\text{int}}(k-1) \\w(k) &= C_w x_w(k) + \varepsilon_{\text{int}}(k) \\w(k-1) &= C_w x_w(k-1) + \varepsilon_{\text{int}}(k-1)\end{aligned} \quad \Longrightarrow \quad \begin{aligned}\Delta x_w(k+1) &= A_w \Delta x_w(k) + B_w \varepsilon(k) \\\Delta w(k) &= C_w \Delta x_w(k) + \varepsilon(k)\end{aligned}$$

where  $\varepsilon(k)$  is **zero-mean stationary process**. And since  $\varepsilon(k)$  is a white noise signal, the above differenced expression is equivalent to:

$$\begin{aligned}\Delta x_w(k+1) &= A_w \Delta x_w(k) + B_w \varepsilon(k) \\\Delta w(k) &= C_w \Delta x_w(k) + \varepsilon(k)\end{aligned}$$

Note:

1. This is actually a very special case of time-variant noise (locally, the noise is coupled to one step before, globally, the effect is a consistent integration over time). And this is the only non-stationary thing we can deal with (worst case for us to deal, but best case in all the non-stationary problems).
2. All the above part is about **modeling the noise (disturbance) part**  $w(k)$  (with the white noise input  $\varepsilon(k)$  and state space model). However, **we still have our real physics system**, and the  $w(k)$  is only part of it, we need to combine everything together and get a standard unified form.

- **Obtaining Models from First Principles**

From first principles, after linearization, one obtains an ODE of the form:

$$\begin{aligned}\dot{x}_p &= A_p^c x_p + B_p^c u + F_p^c w \\y &= C_p^c x_p + G_p^c w\end{aligned}$$

which can be discretized, leading to:

$$\begin{aligned}x_p(k+1) &= A_p x_p(k) + B_p u(k) + F_p w(k) \\y &= C_p x_p(k) + G_p w(k)\end{aligned}$$

Note:

1. Subscript  $p$  is used here to distinguish the process model matrices from the disturbance model matrices introduced before.
2. If the physical disturbance variables cannot be identified, one can express the overall effect of the disturbances as a signal directly added to the output  $\rightarrow$  **output disturbance**,  $G_p = I$ .  $F_p = 0$
3.  $w(k)$  in  $x_p(k+1) = A_p x_p(k) + B_p u(k) + F_p w(k)$  is just what we discussed and modeled in the previous section (the output of stochastic process with input white noise  $\varepsilon(k)$ )

### Stationary Case

Now, combine and stack up what we have, we can get:

$$\begin{bmatrix} x_p(k+1) \\ x_w(k+1) \end{bmatrix} = \begin{bmatrix} A_p & F_p C_w \\ 0 & A_w \end{bmatrix} \begin{bmatrix} x_p(k) \\ x_w(k) \end{bmatrix} + \begin{bmatrix} B_p \\ 0 \end{bmatrix} u(k) + \begin{bmatrix} F_p \\ B_w \end{bmatrix} \varepsilon(k)$$

$$y(k) = [C_p \quad G_p C_w] \begin{bmatrix} x_p(k) \\ x_w(k) \end{bmatrix} + G_p \varepsilon(k)$$

re-define the following:

$$x(k) = \begin{bmatrix} x_p(k) \\ x_w(k) \end{bmatrix}, A = \begin{bmatrix} A_p & F_p C_w \\ 0 & A_w \end{bmatrix}, B = \begin{bmatrix} B_p \\ 0 \end{bmatrix}, C = [C_p \quad G_p C_w], F\varepsilon = \begin{bmatrix} F_p \\ B_w \end{bmatrix}, G = G_p$$

the above expression can be transformed in the standard state-space form of:

$$x(k+1) = Ax(k) + Bu(k) + \underbrace{F\varepsilon(k)}_{\varepsilon_1(k)}$$

$$y(k) = Cx(k) + \underbrace{Gw(k)}_{\varepsilon_2(k)}$$

### Nonstationary Case

In nonstationary case, we would come up with differenced version:

$$\begin{bmatrix} \Delta x_p(k+1) \\ \Delta x_w(k+1) \end{bmatrix} = \begin{bmatrix} A_p & F_p C_w \\ 0 & A_w \end{bmatrix} \begin{bmatrix} \Delta x_p(k) \\ \Delta x_w(k) \end{bmatrix} + \begin{bmatrix} B_p \\ 0 \end{bmatrix} u(k) + \begin{bmatrix} F_p \\ B_w \end{bmatrix} \varepsilon(k)$$

$$\Delta y(k) = [C_p \quad G_p C_w] \begin{bmatrix} \Delta x_p(k) \\ \Delta x_w(k) \end{bmatrix} + G_p \varepsilon(k)$$

Similarly, redefine:

$$x(k) = \begin{bmatrix} x_p(k) \\ x_w(k) \end{bmatrix}, A = \begin{bmatrix} A_p & F_p C_w \\ 0 & A_w \end{bmatrix}, B = \begin{bmatrix} B_p \\ 0 \end{bmatrix}, C = [C_p \quad G_p C_w], F\varepsilon = \begin{bmatrix} F_p \\ B_w \end{bmatrix}, G = G_p$$

$$\Delta x(k) = x(k) - x(k-1), \quad \Delta y(k) = y(k) - y(k-1), \quad \Delta u(k) = u(k) - u(k-1)$$

the above expression can be transformed in the standard state-space form of:

$$\Delta x(k+1) = A \Delta x(k) + B \Delta u(k) + \underbrace{F\varepsilon(k)}_{\varepsilon_1(k)}$$

$$\Delta y(k) = C \Delta x(k) + \underbrace{Gw(k)}_{\varepsilon_2(k)}$$

For estimation and control, it is further desired that **the model output be  $y$  rather than  $\Delta y$** . This

requires yet another augmentation of the state (we need to get  $y(k+1)$ ):

$$\begin{bmatrix} \Delta x(k+1) \\ y(k+1) \end{bmatrix} = \begin{bmatrix} A & 0 \\ C & I \end{bmatrix} \begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} \Delta u(k) + \begin{bmatrix} F \\ G \end{bmatrix} \varepsilon(k)$$

$$y(k) = [0 \quad I] \begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix}$$

Again, redefine the following:

$$\bar{x}(k) = \begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix}, \quad \bar{A} = \begin{bmatrix} A & 0 \\ C & I \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} B \\ 0 \end{bmatrix}, \quad \bar{F} = \begin{bmatrix} F \\ G \end{bmatrix}, \quad \bar{C} = [0 \quad I]$$

It can be brought into the standard state-space form:

$$\bar{x}(k+1) = \bar{A}\bar{x}(k) + \bar{B}\Delta u(k) + \bar{F}\varepsilon(k)$$

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

Note:

1. For nonstationary case: the system input is  $\Delta u$  instead of  $u$
2. The final combined system would have  $n_y$  (number of output dimension) integrators to express the effects of the white noise disturbances and the system input  $\Delta u$  on the output.

## II. State Estimation

### • Initial Remarks: Motivation and General Estimation Problem

#### Motivation:

In many control applications, information about the previous, current or future states is required but it is usually not possible to measure all state variables. Hence the previous/current/future states need to be estimated from past input and measurement sequences. State estimation is an integral part of MPC, which is why good knowledge of it is crucial for designing an effective model predictive control.

#### General Estimation Problem

Consider the following nonlinear time-varying discrete-time system subject to disturbances  $w_1(\cdot)$  and  $w_2(\cdot)$

$$x(k+1) = g(x(k), u(k), w_1(k), k)$$

$$y(k) = h(x(k), u(k), w_2(k), k)$$

The **goal** of state estimation at timestep  $k$  is to estimate  $x(k+i)$  given  $\{y(j), u(j)\}_{j=0,1,\dots,k}$  (i.e. we have the information of control input and output till timestep  $k$ )

Note:

1. Estimating  $x(k+i)$  with  $i > 0$  (i.e. future) is called **prediction**, while the case with  $i = 0$  (i.e. current) is referred to as **filtering**, some applications require the estimation of  $i < 0$  (i.e. past) which is called **smoothing**. We will mostly focus on filtering and prediction.
2. Disturbance modeling is very important in estimation, simply adding white noise to the equations of a deterministic model can give very poor estimation results

**Notation:** we will denote by  $\hat{x}_{i|j}$  an estimate of  $x(i)$  using **all information available up to and at timestep  $j$**

## State Estimation of Linear Systems

In this course, we would mainly focus on the estimation for linear systems

Consider the following model with white process and measurement noise sequences  $\{\varepsilon_1(i)\}_{i=0,1,\dots}$  and  $\{\varepsilon_2(i)\}_{i=0,1,\dots}$

$$\begin{aligned}x(k+1) &= Ax(k) + Bu(k) + \varepsilon_1(k) \\y(k) &= Cx(k) + \varepsilon_2(k)\end{aligned}$$

The (white) noise sequences are assumed to have zero mean, i.e.:

$$\mathbb{E}\{\varepsilon_1(k)\} = 0 \quad \text{and} \quad \mathbb{E}\{\varepsilon_2(k)\} = 0, \quad \forall k \geq 0$$

and to have covariance without time correlation:

$$\mathbb{E}\left\{\begin{bmatrix} \varepsilon_1(i) \\ \varepsilon_2(i) \end{bmatrix} \begin{bmatrix} \varepsilon_1(j) \\ \varepsilon_2(j) \end{bmatrix}^T\right\} = \begin{cases} \begin{bmatrix} R_1 & R_{1,2} \\ R_{1,2}^T & R_2 \end{bmatrix} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

Additionally, it is assumed that the noise sequences are independent of the initial state estimate  $\hat{x}_{0|0}$

Note:

1. Always remember what is mentioned above, here  $x(k)$  is a large state that stack  $x_p(k)$  (physical system process states) and  $x_w(k)$  (stochastic noise process additional state) etc.
2. As mentioned before, we can also use a single  $\varepsilon(k)$  to represent two different noise sequences with different noise input matrix. However, here we write in different  $\varepsilon_1, \varepsilon_2$ , it is because  $\varepsilon_1(k)$  is a process disturbance effecting the states, we can not filter it out and should try to understand it while  $\varepsilon_2(k)$  is just measurement noise. **We want to account for  $\varepsilon_1(k)$  to canceled  $\varepsilon_2(k)$  out.**
3. Our goal: first no error  $e = \hat{x}_{i|i} - x_i = 0$  (i.e. the filter is asymptotic unbiased) and then minimize the covariance  $R$ , see more details below.

### Linear State Estimation

#### Linear Estimator Structure

We assume here that we want to design a linear estimator with the following structure:

1. The **prediction step** propagates the last estimate  $\hat{x}_{k-1|k-1}$  using the nominal (assume  $\varepsilon_1 = 0$ ) model to generate the a **priori estimate**  $\hat{x}_{k|k-1}$ :

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} + Bu(k-1)$$

2. The **update step** corrects the a priori estimate based on the **prediction error**  $y(k) - C\hat{x}_{k|k-1}$  multiplied with the filter gain  $K_k$ :

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y(k) - C\hat{x}_{k|k-1})$$

Define the state estimation error  $x_{i|j}^e$  as the difference between the true value  $x(i)$  and the estimate

$\hat{x}_{i|j}$ , that is to say:

$$x_{i|j}^e = x(i) - \hat{x}_{i|j}$$

We focus on estimating  $x(k)$  using information up to time  $k$ , i.e. focus on calculating  $\hat{x}_{k|k}$ . We want to choose the filter gain  $K_k$  in order to minimize  $x_{k|k}^e$  in some meaningful way.

Note: It can be proved that the linear estimator structure listed here is the optimal one we can do for the system we are discussing.

### Linear Estimation Process

Recall the model and postulated filter equations:

$$x(k) = Ax(k-1) + Bu(k-1) + \varepsilon_1(k-1) \quad (\text{model})$$

$$y(k) = Cx(k) + \varepsilon_2(k)$$

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} + Bu(k-1) \quad (\text{filter})$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y(k) - C\hat{x}_{k|k-1})$$

Substituting  $y(k)$  we can get:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(C[x(k) - \hat{x}_{k|k-1}] + \varepsilon_2(k))$$

Calculating the estimation error, we get:

$$\begin{aligned} x_{k|k}^e &= x(k) - \hat{x}_{k|k} \\ &= x(k) - \hat{x}_{k|k-1} - K_k(C[x(k) - \hat{x}_{k|k-1}] + \varepsilon_2(k)) \\ &= (I - K_k C)x_{k|k-1}^e - K_k \varepsilon_2(k) \end{aligned}$$

Subtract the model and filter equation we can get:

$$x_{k|k-1}^e = Ax_{k-1|k-1}^e + \varepsilon(k-1)$$

Then the estimation error can finally be represented as:

$$x_{k|k}^e = (A - K_k CA)x_{k-1|k-1}^e + (I - K_k C)\varepsilon_1(k-1) - K_k \varepsilon_2(k)$$

Note: In the dynamics of estimation error, we should regard  $(A - K_k CA)x_{k-1|k-1}^e$  as the inherent error transition dynamics (autonomous system), and regard  $(I - K_k C)\varepsilon_1(k-1) - K_k \varepsilon_2(k)$  as the input. The next task is to choose proper  $K_k$  such that the error dynamics  $(A - K_k CA)$  is stable.

#### • State Observer

The theory of state observers is based on the behavior of the estimation error in a deterministic setting ( $\varepsilon_1 \equiv 0, \varepsilon_2 \equiv 0$ ) with a fixed  $K_k = K$ , hence we get:

$$x_{k|k}^e = (A - KCA)x_{k-1|k-1}^e$$

An estimator for system is said to be **observer-stable** if  $x_{k|k}^e \rightarrow 0$  as  $k \rightarrow \infty$  for any  $x_{0|0}^e \rightarrow 0$  when  $\varepsilon_1 \equiv 0, \varepsilon_2 \equiv 0$ .

For observer stability, it is clear that the eigenvalues of  $A - KCA$  must lie strictly inside the unit circle in the complex plane.

Note: the white noise inputs are random and has no time correlation, we generally not tend to deal with

them. Instead, tuning the inherent dynamics and making it stable should be more important and basic. Therefore, in the state observer theory, we can assume white noise inputs to be zero.

### Important Property

1. If and only if  $(CA, A)$  is observable, the eigenvalues of  $A - KCA$  can be placed arbitrarily by choosing an appropriate  $K$ . That is, we can specify the eigenvalue inside the unit circle.
2. If  $A$  has full rank then observability of  $(CA, A)$  is equivalent to observability of  $(C, A)$
3. Determining  $K$  such that the eigenvalues of  $A - KCA$  are at desired locations is called observer pole placement

Note:

1. A slightly different observer structure  $(\hat{x}_{k+1|k} = A\hat{x}_{k|k-1} + K(y(k) - C_{k|k-1}) + Bu(k))$  yields an error dynamics matrix  $A - KC$  where observability of  $(C, A)$  is necessary and sufficient to arbitrarily place the eigenvalues through  $K$ .
2. Observer pole placement is not frequently (That is also why we need Kalman filter) used since
  - a. In general, pole placement does not **determine the observer gain  $K$  uniquely** and it is unclear how to best use the remaining degrees of freedom.
  - b. Although the eigenvalues can be freely chosen, the eigenvectors cannot and the observer may recover only very slowly from certain types of errors

Though it is not that useful (or, only useful for simple cases), at least it ensures the **minimum requirement** for  $K$  in theory.

It shows that we need methods that could choose  $K$  optimally, instead of just ensuring the stability.

### • Kalman Filter

Intuitively, incorporating statistical information on how the disturbances affect the state can improve the quality of the estimation. The **Kalman filter** accounts for this information in an optimal way.

Denote the covariance of the estimation error  $x_{i|j}^e$  by:

$$P_{i|j} = \text{Cov}\{x_{i|j}^e\}$$

The optimality criterion of the Kalman filter is the **error covariance  $P_{k|k}$** , i.e. we choose  $K_k$  such that  $\alpha^T P_{k|k} \alpha$  is minimized for any  $\alpha$  of appropriate dimension

Still, for simplicity it is assumed that  $\mathbb{E}\{\varepsilon_1(i)\varepsilon_2(i)^T\} = R_{1,2} = 0$  and  $R_2 \succ 0$  (positive definite)

### Unbiasedness of estimation error $x_{k|k}^e$

Recall the error update equations:

$$x_{k|k}^e = (A - K_k CA)x_{k-1|k-1}^e + (I - K_k C)\varepsilon_1(k-1) - K_k \varepsilon_2(k)$$

Doing the dynamics rollout from the initial state,  $x_{k|k}^e$  can be written in explicit form:



$$x_{k|k}^e = \left[ \prod_{i=1}^k (A - K_i C A) \right] x_{0|0}^e + \sum_{i=1}^k \left( \left[ \prod_{j=i+1}^k (A - K_j C A) \right] (I - K_i C) \varepsilon_1(i-1) \right) - \sum_{i=1}^k \left( \left[ \prod_{j=i+1}^k (A - K_j C A) \right] K_i \varepsilon_2(i) \right)$$

Since by assumption  $\mathbb{E}\{\varepsilon_1(i)\} = 0$  and  $\mathbb{E}\{\varepsilon_2(i)\} = 0$ , taking the expectation of the estimation error we have:

$$\mathbb{E}\{x_{k|k}^e\} = \left[ \prod_{i=1}^k (A - K_i C A) \right] \mathbb{E}\{x_{0|0}^e\}$$

If  $K_k \rightarrow K_\infty$  as  $k \rightarrow \infty$  and if the eigenvalues of  $(A - K_\infty C A)$  are stable, then the filter is **asymptotically unbiased**, i.e.:

$$\lim_{k \rightarrow \infty} \mathbb{E}\{x_{k|k}^e\} = 0$$

**Facts used for derivation** (Note: we have not done since we in addition want to minimize covariance)

1.  $x_{k-1|k-1}^e$  and  $\varepsilon_1(k-1)$  are uncorrelated

This is obvious since  $\varepsilon_1(k-1)$  only influences  $x(k)$ , which itself is not part of  $x_{k-1|k-1}^e$

2.  $x_{k|k-1}^e$  and  $\varepsilon_2(k-1)$  are uncorrelated

This is obvious since  $\varepsilon_2(k)$  only influences  $y(k)$ , which itself is not part of  $x_{k|k-1}^e$

3.  $P_{i|j} = P_{i|j}^T$  and  $P_{i|j} \succeq 0$

Any covariance matrix is symmetric positive-semidefinite by definition

4.  $S_k = C P_{k|k-1} C^T + R_2 \succ 0$

This follows from  $P_{k|k-1} \succ 0$  and  $R_2 \succ 0$

**Derivation of covariances**  $P_{k|k-1}$  and  $P_{k|k}$

Start from the estimation error equations:

$$x_{k|k-1}^e = A x_{k-1|k-1}^e + \varepsilon(k-1)$$

$$x_{k|k}^e = (I - K_k C) x_{k|k-1}^e - K_k \varepsilon_2(k)$$

Using that  $x_{k-1|k-1}^e$  and  $\varepsilon_1(k-1)$  are uncorrelated we can compute  $P_{k|k-1}$  as:

$$\begin{aligned} P_{k|k-1} &= \text{Cov}\{A x_{k-1|k-1}^e + \varepsilon_1(k-1)\} \\ &= \text{Cov}\{A x_{k-1|k-1}^e\} + \text{Cov}\{\varepsilon_1(k-1)\} \\ &= A \text{Cov}\{x_{k-1|k-1}^e\} A^T + R_1 \\ &= A P_{k-1|k-1} A^T + R_1 \end{aligned}$$

Similarly for  $P_{k|k}$  using that  $x_{k|k-1}^e$  and  $\varepsilon_2(k-1)$  are uncorrelated:

$$\begin{aligned} P_{k|k} &= \text{Cov}\{(I - K_k C) x_{k|k-1}^e - K_k \varepsilon_2(k)\} \\ &= \text{Cov}\{(I - K_k C) x_{k|k-1}^e\} + \text{Cov}\{K_k \varepsilon_2(k)\} \\ &= (I - K_k C) \text{Cov}\{x_{k|k-1}^e\} (I - K_k C)^T + K_k R_2 K_k^T \\ &= (I - K_k C) P_{k|k-1} (I - K_k C)^T + K_k R_2 K_k^T \end{aligned}$$

Rewrite the  $P_{k|k}$  :

$$\begin{aligned} P_{k|k} &= (I - K_k C) P_{k|k-1} (I - K_k C)^T + K_k R_2 K_k^T \\ &= K_k \underbrace{(C P_{k|k-1} C^T + R_2)}_{S_k} K_k^T - K_k C P_{k|k-1} - (K_k C P_{k|k-1})^T + P_{k|k-1} \end{aligned}$$

The next step is called “completing the square”. i.e. rewrite the expression with respect to  $K_k$ , Since  $S_k = C P_{k|k-1} C^T + R_2 \succ 0$ , it always can be inverted, resulting in:

$$P_{k|k} = (K_k - P_{k|k-1} C^T S_k^{-1}) S_k (K_k - P_{k|k-1} C^T S_k^{-1})^T + P_{k|k-1} - P_{k|k-1} C^T S_k^{-1} C P_{k|k-1}$$

From the expression above, we can see the “quadratic” part about  $K_k$  and “constant” part about  $K_k$ .

Then, it is clear that  $\alpha^T P_{k|k} \alpha$  is minimized for any  $\alpha$  if we choose  $K_k = P_{k|k-1} C^T S_k^{-1}$  and :

$$\begin{aligned} P_{k|k} &= P_{k|k-1} - P_{k|k-1} C^T S_k^{-1} C P_{k|k-1} = (I - P_{k|k-1} C^T S_k^{-1} C) P_{k|k-1} \\ &= (I - K_k C) P_{k|k-1} \end{aligned}$$

### Kaiman Filter Algorithm

1. Initialize  $\hat{x}_{0|0}$  and  $P_{0|0}$
2. Compute the filter gain  $K_k$  and the error covariance matrix  $P_{k|k}$  online in advance using:

$$\begin{aligned} P_{k|k-1} &= A P_{k-1|k-1} A^T + R_1 \\ K_k &= P_{k|k-1} C^T (C P_{k|k-1} C^T + R_2)^{-1} = P_{k|k-1} C^T S_k^{-1} \\ P_{k|k} &= (I - K_k C) P_{k|k-1} \end{aligned}$$

i.e. we calculate successively from  $P_{k-1|k-1} \rightarrow P_{k|k-1} \rightarrow P_{k|k}$  and thus get the time sequence of  $P$  and  $K$  ahead of time.

3. At time  $k$ 
  - a. Compute the priori estimate:

$$\hat{x}_{k|k-1} = A \hat{x}_{k-1|k-1} + B u(k-1)$$

- b. Get measurement  $y(k)$

- c. Compute new estimate

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (y(k) - C \hat{x}_{k|k-1})$$

4.  $k \rightarrow k+1$

Note:

1. For precomputing  $P$  and  $R$ , we need to know  $A, C, R_1$  and  $R_2$ , then everything can be calculated in step 2 in advance and call out them online.
2. The derivation of the Kalman filter with  $R_{1,2}$  follows the same but requires a bit more algebra.
3. Generalization of the Kalman filter equations for a system where  $A, B, C$  are time-varying is straightforward. It is possible to replace  $A$  by  $A(k)$ ,  $B$  by  $B(k)$ , and  $C$  by  $C(k)$  in all equations (except the steady-state equations in the next slides) without changing the arguments used for the derivation.

## Kaiman Filter Analysis

In our derivation for Kaiman Filter, we show that it is optimal in minimizing the covariance. However, we should always forget all the optimal stuff should build on the stability:

If  $P_{k|k-1}$  converges to a steady-state solution  $P_\infty$  and the corresponding steady-state Kalman gain  $K_\infty$  satisfies the condition that  $A - K_\infty CA$  has all eigenvalues strictly inside the unit circle, then the expected estimation error goes asymptotically to zero. Together with the bounded covariance, **it is said that the Kalman filter is stable.**

Therefore, we should investigate **under which conditions the Kalman gain converges** to  $K_\infty$  and the asymptotic filter gain is stabilizing, i.e. the eigenvalues of  $A - K_\infty CA$  lie inside the unit circle. Analysis start from the successive calculation of covariance matrix:

$$P_{k|k} = (I - K_k C) P_{k|k-1} = P_{k|k-1} - P_{k|k-1} C^T S_k^{-1} C P_{k|k-1}$$

Substituting  $P_{k-1|k-1}$  in the calculation of  $P_{k|k-1}$  (inverted rollout) would result:

$$\begin{aligned} P_{k|k-1} &= A P_{k-1|k-1} A^T + R_1 \\ &= A P_{k-1|k-2} A^T - A P_{k-1|k-2} C^T S_{k-1}^{-1} C P_{k-1|k-2} A^T + R_1 \end{aligned}$$

From the last equation together with  $S_{k-1} = C P_{k-1|k-2} C^T + R_2$  it can be seen that if  $P_{k|k-1}$  converges (i.e.  $P_{k|k-1} = P_{k-1|k-2} = P_\infty$ ) it must satisfy the **Algebraic Riccati Equation:**

$$P_\infty = A P_\infty A^T - A P_\infty C^T (C P_\infty C^T + R_2)^{-1} C P_\infty A^T + R_1$$

$P_\infty$  is called a **stabilizing** solution if all eigenvalues of  $A - K_\infty CA$  are strictly inside the unit circle. We have the following properties:

If  $(C, A)$  is detectable, then  $P_{k|k-1}$  converges regardless of  $P_{0|0}$

If  $(C, A)$  is detectable,  $(A, R_1^{1/2})$  is stabilizable and  $P_{0,0} \succeq 0$  then  $P_{k|k-1}$  converges to  $P_\infty$  where  $P_\infty$  is a stabilizing solution to the ARE

Note:

1. In special case of  $C = 0$  (not observing anything), the above Algebraic Riccati equation returns to the Lyapunov equation. It means if the system is stable, then under some noise input, the variance of the process is  $P_\infty$ , which is an open-loop variance.
2. Intuition about detectability: if  $(C, A)$  is not detectable, first it is not observable (can not see the whole state), also, the instability can not be seen. Therefore, it is natural that the covariance will not converge. However, remember that  $(C, A)$  detectable only guarantee the convergence of  $P$ , but not stability! (see the next note)
3. Intuition about  $(A, R_1^{1/2})$  stabilizable:  $R_1^{1/2}$  in some sense is just  $\varepsilon_1$ . Remember that the noise is **the input** in our process, we hope that the noise can somewhat perturb/affect the unstable modes of  $A$  so that it can be seen in the output. **The more important part of this property is that it tells us how to design  $R_1$** , as previously mentioned, we need to know  $A, C, R_1$  and  $R_2$  to help us design the Kalman filter. And also, in practice, generally designing  $R_1$  usually means **designing**

**the structure of the covariance matrix (e.g. blocks corresponding certain noises), not the value (gain) of its components.**

4. Determine the observer gain: Observer Pole Placement (theorem) and Kalman filter (Practice)

Observer Pole Placement: Tune  $K$  to make the system work, really hard in real life.

Kalman filter: Manually tune  $R_1, R_2$  instead of directly measure it them, then calculate  $K$

5. Practice always somehow differs from theory, and we need to choose the best practical method we can actually use! Therefore, although manually tune  $R_1, R_2$  in Kalman filter instead of directly measure seems a bit tricky, but it is a good method. In fact, Kalman filter is nearly a perfect example since it gives us a practical way to design and the result is also guaranteed by the theorem!