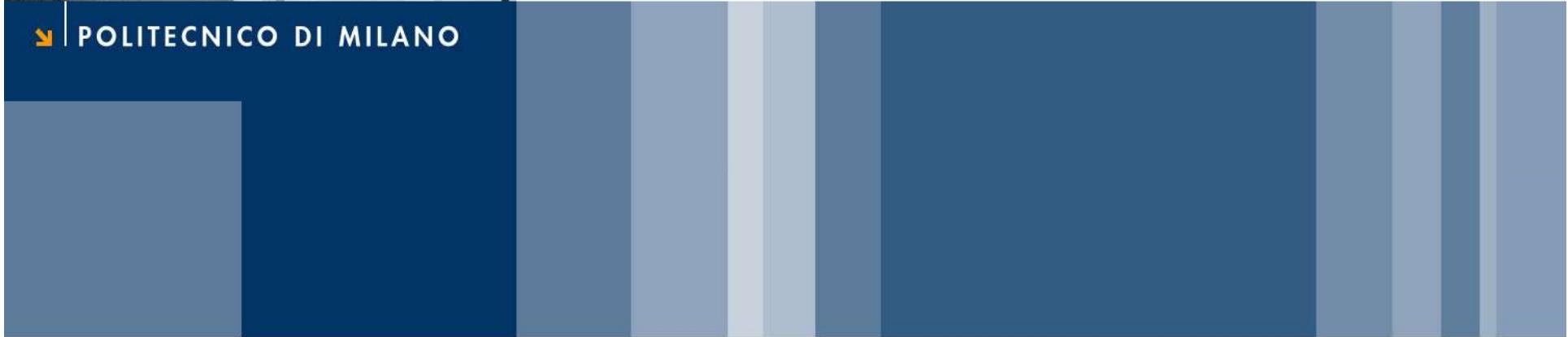




 POLITECNICO DI MILANO



Kalman prediction and filtering

Marco Lovera

Dipartimento di Scienze e Tecnologie Aerospaziali, Politecnico di Milano



Problem statement

We start from the DT-DT problem, formulated as follows.
The system under study is given by

$$\begin{aligned}x(t+1) &= Fx(t) + w(t), \quad x(1) = x_1 \\y(t) &= Hx(t) + v(t)\end{aligned}$$

where:

- v and w are DT white Gaussian noise processes with

$$w \approx G(0, W), \quad v \approx G(0, V)$$

- x_1 is a Gaussian random variable:

$$x_1 \approx G(0, P_1)$$

- v , w and x_1 are independent.



Problem statement

We want to define estimators for the state vector x on the basis of measurements of the output y :

$$y^N = [y(1), y(2), \dots, y(N)].$$

- $t > T$: *prediction* problem.
- $t = T$: *filtering* problem.
- $0 < t < T$: *smoothing* problem.

We first consider the prediction problem, starting from *one-step-ahead* prediction.



Free response:

$$x(t+1) = Fx(t), \quad x(1) = x_1$$

$$x(2) = Fx(1) = Fx_1$$

$$x(3) = Fx(2) = F^2x_1$$

...

$$x(t) = Fx(t-1) = F^{t-1}x_1.$$



Forced response:

$$x(t+1) = Fx(t) + w(t), \quad x(1) = 0$$

$$x(2) = Fx(1) + w(1) = w(1)$$

$$x(3) = Fx(2) + w(2) = Fw(1) + w(2)$$

$$x(4) = Fx(3) + w(3) = F^2w(1) + Fw(2) + w(3)$$

...

$$x(t) = \sum_{k=1}^{t-1} F^{t-k} w(k).$$



Comments:

- The free response $x(t) = Fx(t - 1) = F^{t-1}x_1$ is linear in the initial state, so if the initial condition is Gaussian the free response is also Gaussian for all t .
- The forced response $x(t) = \sum_{k=1}^{t-1} F^{t-k}w(k)$ is linear in the samples of $w(t)$, so if process noise is a Gaussian RP, the forced response is Gaussian for all t .
- Finally, since the system is linear, the response is the sum of free and forced and therefore is also Gaussian.



Using Bayes rule we can express the optimal one-step-ahead state and output predictors as

$$\begin{aligned}\hat{x}(N+1|N) &= E[x(N+1)|y^N] \\ \hat{y}(N+1|N) &= E[y(N+1)|y^N].\end{aligned}$$

We will use often the innovation

$$e(N+1) = y(N+1) - E[y(N+1)|y^N]$$

and the state prediction error

$$\nu(N+1) = x(N+1) - E[x(N+1)|y^N].$$



Consider first the output prediction:

$$\begin{aligned}\hat{y}(N+1|N) &= E[y(N+1)|y^N] = \\ &= E[Hx(N+1) + v(N+1)|y^N] = \\ &= H E[x(N+1)|y^N] + E[v(N+1)|y^N].\end{aligned}$$

The second term is zero, as:

- $y(N)$ is a function of v up to time N , of w up to time $N-1$ and of x_1 .
- $v(N+1)$ is independent of
 - previous samples of v and w
 - the initial state x_1 .

In other words, $v(N+1)$ is *unpredictable* based on past data.



Therefore we have

$$\hat{y}(N+1|N) = HE[x(N+1)|y^N] = H\hat{x}(N+1|N).$$

Note that as in the Luenberger observer the prediction of the output is expressed in terms of the prediction of the state through the output matrix H .



Consider now the state prediction:

$$\begin{aligned}\hat{x}(N+1|N) &= E[x(N+1)|y^N] = \\ &= E[x(N+1)|y^{N-1}, y(N)] = \\ &= E[x(N+1)|y^{N-1}] + E[x(N+1)|y(N)].\end{aligned}$$

The second term can be written in terms of the innovation:

$$\hat{x}(N+1|N) = E[x(N+1)|y^{N-1}] + E[x(N+1)|e(N)].$$

Next, we have to evaluate the two terms on the RHS.



The first term is given by:

$$E[x(N+1) \setminus y^{N-1}] = E[Fx(N) + w(N) \setminus y^{N-1}].$$

Equivalently:

$$E[x(N+1) \setminus y^{N-1}] = FE[x(N) \setminus y^{N-1}] + E[w(N) \setminus y^{N-1}].$$

$E[w(N) \setminus y^{N-1}]$ is zero, as:

- $y(N-1)$ is a function of v up to time $N-1$, of w up to time $N-2$ and of x_1 .
- $w(N)$ is independent of
 - previous samples of v and w
 - the initial state x_1 .

Therefore we get $E[x(N+1) \setminus y^{N-1}] = F\hat{x}(N \setminus N-1)$.



Substituting:

$$\hat{x}(N+1|N) = F\hat{x}(N|N-1) + E[x(N+1)|e(N)].$$

Using the vector Bayes rule, the second term is given by

$$E[x(N+1)|e(N)] = \Lambda_{x(N+1)e(N)} \Lambda_{e(N)e(N)}^{-1} e(N)$$

and to make it explicit we have to compute the two variance matrices:

$$\Lambda_{x(N+1)e(N)} = E[x(N+1)e^T(N)]$$

$$\Lambda_{e(N)e(N)} = E[e(N)e^T(N)].$$



For the covariance between $x(N+1)$ and $e(N)$ we have

$$\begin{aligned}\Lambda_{x(N+1)e(N)} &= E[x(N+1)e^T(N)] = \\ &= E[(Fx(N) + w(N))(Hx(N) + v(N) - H\hat{x}(N \setminus N-1))^T] = \\ &= E[(Fx(N) + w(N))(H(x(N) - \hat{x}(N \setminus N-1)) + v(N))^T].\end{aligned}$$

Computing the products:

$$\begin{aligned}\Lambda_{x(N+1)e(N)} &= E[(Fx(N) + w(N))(H(x(N) - \hat{x}(N \setminus N-1)) + v(N))^T] = \\ &= FE[x(N)(x(N) - \hat{x}(N \setminus N-1))^T]H^T + FE[x(N)v^T(N)] + \\ &\quad + E[w(N)H(x(N) - \hat{x}(N \setminus N-1)) + v(N))^T].\end{aligned}$$



Note that in

$$\begin{aligned}\Lambda_{x(N+1)e(N)} &= FE[x(N)(x(N) - \hat{x}(N \setminus N-1))^T]H^T + FE[x(N)v^T(N)] + \\ &\quad + E[w(N)H(x(N) - \hat{x}(N \setminus N-1) + v(N))^T]\end{aligned}$$

the second and the third terms are zero, so we have

$$\Lambda_{x(N+1)e(N)} = FE[x(N)(x(N) - \hat{x}(N \setminus N-1))^T]H^T.$$

To evaluate the expectation we re-write it as

$$\Lambda_{x(N+1)e(N)} = FE[(x(N) \pm \hat{x}(N \setminus N-1))(x(N) - \hat{x}(N \setminus N-1))^T]H^T$$

and compute the products.



We get

$$\begin{aligned}\Lambda_{x(N+1)e(N)} &= FE[(x(N) \pm \hat{x}(N \setminus N-1))(x(N) - \hat{x}(N \setminus N-1))^T]H^T = \\ &= FE[(x(N) - \hat{x}(N \setminus N-1))(x(N) - \hat{x}(N \setminus N-1))^T]H^T + \\ &\quad + FE[\hat{x}(N \setminus N-1)(x(N) - \hat{x}(N \setminus N-1))^T]H^T\end{aligned}$$

which can be written in terms of the prediction error:

$$\Lambda_{x(N+1)e(N)} = FE[\nu(N)\nu(N)^T]H^T + FE[\hat{x}(N \setminus N-1)\nu(N)^T]H^T.$$

The second term is zero: the prediction error at time N is the unpredictable part of $x(N)$ and therefore is independent of the prediction of $x(N)$.



Therefore, we get

$$\Lambda_{x(N+1)e(N)} = FE[\nu(N)\nu(N)^T]H^T$$

and letting $P(N) = E[\nu(N)\nu(N)^T]$

we have the final result

$$\Lambda_{x(N+1)e(N)} = FP(N)H^T.$$



For the covariance between $e(N)$ and $e(N)$, recalling that

$$e(N) = y(N) - \hat{y}(N|N-1) = H\nu(N|N-1) + v(N)$$

we have

$$\begin{aligned}\Lambda_{e(N)e(N)} &= E[e(N)e^T(N)] = \\ &= E[H\nu(N|N-1)\nu^T(N|N-1)H^T] + E[v(N)v^T(N)] + \text{cross terms} = \\ &= HP(N)H^T + V.\end{aligned}$$

The cross-terms can be shown to be zero by means of the usual arguments.



We now have:

$$\hat{x}(N+1|N) = F\hat{x}(N|N-1) + E[x(N+1)|e(N)].$$

where

$$E[x(N+1)|e(N)] = \Lambda_{x(N+1)e(N)} \Lambda_{e(N)e(N)}^{-1} e(N)$$

and

$$\begin{aligned}\Lambda_{x(N+1)e(N)} &= FP(N)H^T \\ \Lambda_{e(N)e(N)} &= HP(N)H^T + V\end{aligned}$$

therefore the complete predictor is

$$\hat{x}(N+1|N) = F\hat{x}(N|N-1) + FP(N)H^T(HP(N)H^T + V)^{-1}e(N).$$



Letting

$$K(N) = FP(N)H^T(HP(N)H^T + V)^{-1}$$

the *gain* of the predictor, we get

$$\begin{aligned}\hat{x}(N+1|N) &= F\hat{x}(N|N-1) + K(N)e(N) \\ \hat{y}(N|N-1) &= H\hat{x}(N|N-1).\end{aligned}$$

Recalling the definition of the innovation as

$$e(N) = y(N) - \hat{y}(N|N-1)$$

we recognize that the optimal predictor has the same structure as the Luenberger observer.



Note however that unlike the Luenberger observer:

- The optimal gain $K(N)$ determined using Bayes rule is NOT constant.
- The definition of the gain is not yet complete as we still need an update equation for $P(N)$.



The update equation for $P(N)$ can be derived starting from the definition of prediction error:

$$\nu(N+1) = x(N+1) - \hat{x}(N+1|N)$$

which can be also written as

$$\begin{aligned}\nu(N+1) &= x(N+1) - \hat{x}(N+1|N) = \\ &= F(x(N) - \hat{x}(N|N-1)) + w(N) - K(N)e(N)\end{aligned}$$

and recalling $e(N) = H\nu(N) + v(N)$

$$\nu(N+1) = (F - K(N)H)\nu(N) + w(N) - K(N)v(N).$$



Squaring

$$\nu(N+1) = (F - K(N)H)\nu(N) + w(N) - K(N)v(N)$$

we get

$$\begin{aligned}\nu(N+1)\nu^T(N+1) &= (F - K(N)H)\nu(N)\nu^T(N)(F - K(N)H)^T + \\ &\quad + w(N)w^T(N) - K(N)v(N)v^T(N)K^T(N) + \\ &\quad + \text{cross products}.\end{aligned}$$

Taking expectations of both sides:

$$P(N+1) = (F - K(N)H)P(N)(F - K(N)H)^T + W - K(N)VK^T(N)$$

as it can be shown that $E[\text{cross products}] = 0$.



The update equation for $P(N)$

$$P(N + 1) = (F - K(N)H)P(N)(F - K(N)H)^T + W - K(N)VK^T(N)$$

can be also written as

$$P(N + 1) = FP(N)F^T + W - FP(N)H^T[HP(N)H^T + V]^{-1}HP(N)F^T$$

where $K(N) = FP(N)H^T(HP(N)H^T + V)^{-1}$ has been used.

Or, equivalently, as

$$P(N + 1) = FP(N)F^T + W - K(N)[HP(N)H^T + V]K^T(N).$$

This equation is known as the Difference Riccati Equation (DRE).



The last form

$$P(N+1) = FP(N)F^T + W - K(N)[HP(N)H^T + V]K^T(N)$$

is interesting as it allows a simple interpretation.

- $P(N)$ is a variance matrix, so it is positive semidefinite.
- Indeed the RHS is a sum of positive sign-definite terms.
- The first two (positive: variance increase) correspond to *prediction*, i.e., pure propagation of the variance on the system's state equation.
- The last term (negative: variance reduction) corresponds to the *correction*, introduced by feedback of the innovation.



- The definition of the predictor is now complete.
- We just have to specify the initialisation for the prediction and for the variance of the prediction error.
- For the prediction, at time 1 we should condition for data at time 0, which is not available. Therefore

$$\hat{x}(1|0) = E[x(1)|y^0] = E[x(1)] = 0.$$

- For the Riccati equation:

$$P(1) = E[(x(1) - \hat{x}(1|0))^2] = E[(x(1) - E[x(1)])^2] = P_1.$$



- System:

$$x(t+1) = Fx(t) + w(t), \quad x(1) = x_1$$

$$y(t) = Hx(t) + v(t)$$

$$w \approx G(0, W), \quad v \approx G(0, V), \quad x_1 \approx G(0, P_1)$$

- State prediction:

$$\hat{x}(N+1|N) = F\hat{x}(N|N-1) + K(N)(y(N) - \hat{y}(N|N-1)), \quad \hat{x}(1|0) = x_1$$

$$\hat{y}(N|N-1) = H\hat{x}(N|N-1).$$

- Gain and prediction error variance update:

$$P(N+1) = FP(N)F^T + W - FP(N)H^T[HP(N)H^T + V]^{-1}HP(N)F^T, \quad P(1) = P_1$$

$$K(N) = FP(N)H^T(HP(N)H^T + V)^{-1}$$



We now turn to the problem of *r*-step-ahead prediction, *i.e.*, the computation of

$$\hat{x}(N + r \setminus N) = E[x(N + r) \setminus y^N].$$

We have

$$\begin{aligned}\hat{x}(N + r \setminus N) &= E[x(N + r) \setminus y^N] = \\ &= E[Fx(N + r - 1) + v(N + r - 1) \setminus y^N] = \\ &= F\hat{x}(N + r - 1 \setminus N) + \text{null terms}.\end{aligned}$$

Iterating down to $N+1$ we get

$$\begin{aligned}\hat{x}(N + r \setminus N) &= F^{r-1}\hat{x}(N + 1 \setminus N) \\ \hat{y}(N + r \setminus N) &= H\hat{x}(N + r \setminus N).\end{aligned}$$



As a particular case note that if we evaluate

$$\hat{x}(N + r \setminus N) = F\hat{x}(N + r - 1 \setminus N)$$

for $r=1$ we get $\hat{x}(N + 1 \setminus N) = F\hat{x}(N \setminus N)$.

Therefore:

- if F is invertible we can easily solve the filtering problem from the one-step-ahead prediction:

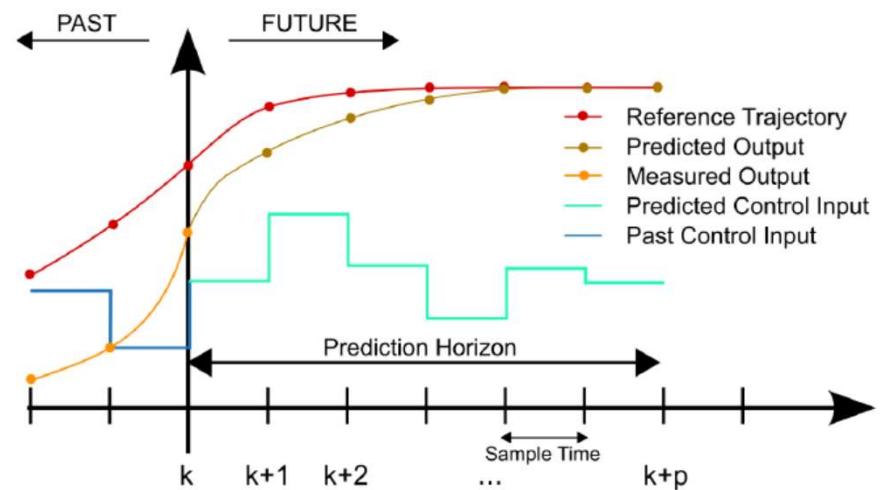
$$\hat{x}(N \setminus N) = F^{-1}\hat{x}(N + 1 \setminus N)$$

- On the contrary if the filtered estimate is available, the one-step-ahead prediction is just a one-step propagation of the state equation.



In Model Predictive Control (MPC):

- At each time instant the current output is measured and the state prediction is computed as function of future outputs.
- A performance metric is optimised with respect to future control samples.
- The first sample of the computed control sequence is applied.
- The whole procedure is repeated at the subsequent step (*receding horizon principle*).





- For conventional real-time control however we are not interested in estimating the *future* state but rather the *current* state.
- Therefore the problem we need to solve is *filtering* rather than *prediction*.
- As we will see, filtering can be solved easily by building on the optimal one-step-ahead predictor.



We want to compute $\hat{x}(N \setminus N) = E[x(N) \setminus y^N]$:

$$\begin{aligned}\hat{x}(N \setminus N) &= E[x(N) \setminus y^N] = \\&= E[x(N) \setminus y^{N-1}, y(N)] = \\&= E[x(N) \setminus y^{N-1}] + E[x(N) \setminus e(N)] \\&= \hat{x}(N \setminus N - 1) + \Lambda_{x(N)e(N)} \Lambda_{e(N)e(N)}^{-1} e(N) = \\&= F\hat{x}(N - 1 \setminus N - 1) + \Lambda_{x(N)e(N)} \Lambda_{e(N)e(N)}^{-1} e(N).\end{aligned}$$

We have seen that

$$\Lambda_{e(N)e(N)} = E[e(N)e^T(N)] = HP(N)H^T + V$$

and it can be proved that

$$\Lambda_{x(N)e(N)} = P(N)H^T.$$



Therefore, the optimal filter update is given by

$$\hat{x}(N \setminus N) = E[x(N) \setminus y^N] = F\hat{x}(N-1 \setminus N-1) + K_F(N)e(N)$$

where

$$K_F(N) = P(N)H^T(HP(N)H^T + V)^{-1}.$$

Note that

$$K(N) = FK_F(N).$$



- System:

$$x(t+1) = Fx(t) + w(t), \quad x(1) = x_1$$

$$y(t) = Hx(t) + v(t)$$

$$w \approx G(0, W), \quad v \approx G(0, V), \quad x_1 \approx G(0, P_1)$$

- State filtering:

$$\hat{x}(N \setminus N) = F\hat{x}(N-1 \setminus N-1) + K_F(N)(y(N) - \hat{y}(N \setminus N-1)), \quad \hat{x}(1 \setminus 0) = x_1$$

$$\hat{y}(N \setminus N) = H\hat{x}(N \setminus N).$$

- Gain and prediction error variance update:

$$P(N+1) = FP(N)F^T + W - FP(N)H^T[HP(N)H^T + V]^{-1}HP(N)F^T, \quad P(1) = P_1$$

$$K_F(N) = P(N)H^T(HP(N)H^T + V)^{-1}.$$



- The previous expression is somewhat hybrid, in the sense that it involves both filtered and predicted quantities.
- An expression of the filter in terms of the filter error

$$\nu_F(N) = x(N) - \hat{x}(N \setminus N)$$

and its variance

$$P_F(N) = E[\nu_F(N)\nu^T(N)]$$

can be derived, but it is very complicated and not suitable for implementation.



- In the following we will derive the so-called prediction-correction form for the optimal filter.
- This form combines predicted and filtered quantities in a systematic way.



Recall that

$$\hat{x}(N \setminus N-1) = F\hat{x}(N-1 \setminus N-1)$$

so we can obtain the prediction at $N+1$ from the filtered estimate at time N .

The new filtered estimate can be seen as a *correction* based on the measurement at time N :

$$\begin{aligned}\hat{x}(N \setminus N) &= \hat{x}(N \setminus N-1) + K_F(N)(y(N) - \hat{y}(N \setminus N-1)) \\ \hat{y}(N \setminus N) &= H\hat{x}(N \setminus N).\end{aligned}$$



For variances: we have from the Riccati equation that

$$P(N+1) = FP(N)F^T + W - K(N)[HP(N)H^T + V]K^T(N).$$

and using $K(N) = FK_F(N)$:

$$P(N+1) = FP(N)F^T + W - FK_F(N)[HP(N)H^T + V]K_F^T(N)F^T.$$

Based on

$$\hat{x}(N \setminus N) = \hat{x}(N \setminus N-1) + K_F(N)(y(N) - \hat{y}(N \setminus N-1))$$

it can be proved that

$$P_F(N) = P(N) - K_F(N)(HP(N)H^T + V)K_F^T(N)$$



Therefore if the filter error variance from the previous time instant is known, then the prediction error variance is

$$P(N) = FP_F(N-1)F^T + W$$

and the updated filter error variance is:

$$P_F(N) = P(N) - K_F(N)(HP(N)H^T + V)K_F^T(N)$$

which recalling

$$K_F(N) = P(N)H^T(HP(N)H^T + V)^{-1}$$

can be simplified to

$$P_F(N) = P(N) - K_F(N)HP(N) = (I - K_F(N)H)P(N).$$



- In the predictor/corrector form a slightly different notation is used:

$$\hat{x}(N \setminus N - 1) \rightarrow \hat{x}(N)(-)$$

$$\hat{x}(N \setminus N) \rightarrow \hat{x}(N)(+)$$

$$P(N) \rightarrow P(N)(-)$$

$$P_F(N) \rightarrow P(N)(+)$$



State estimate and error covariance extrapolation:

$$\hat{x}(N)(-) = F\hat{x}(N-1)(+)$$

$$P(N)(-) = FP(N-1)(+)F^T + W$$

Gain update:

$$K_F(N) = P(N)(-)H^T \left[HP(N)(-)H^T + V \right]^{-1}$$

State estimate update and error covariance update:

$$\hat{x}(N)(+) = \hat{x}(N)(-) + K_F(N) (y(N) - H\hat{x}(N)(-))$$

$$P(N)(+) = [I - K_F(N)H] P(N)(-)$$



- The results on Kalman prediction and filtering have been derived under some simplifying assumptions for the sake of simplicity.
- Some of the assumptions can be removed, so that the results have more general validity.



- Consider a plant model which includes a control input

$$\begin{aligned}x(t+1) &= Fx(t) + Gu(t) + w(t), \quad x(1) = x_1 \\y(t) &= Hx(t) + v(t).\end{aligned}$$

- Then the input can be included in the prediction/filtering approach developed so far just like in the Luenberger observer problem.



- System:

$$x(t+1) = Fx(t) + Gu(t) + w(t), \quad x(1) = x_1$$

$$y(t) = Hx(t) + v(t)$$

$$w \approx G(0, W), \quad v \approx G(0, V), \quad x_1 \approx G(0, P_1)$$

- State prediction:

$$\begin{aligned}\hat{x}(N+1|N) &= F\hat{x}(N|N-1) + Gu(N) + K(N)(y(N) - \hat{y}(N|N-1)), \quad \hat{x}(1|0) = x_1 \\ \hat{y}(N|N-1) &= H\hat{x}(N|N-1).\end{aligned}$$

- Gain and prediction error variance update:

$$\begin{aligned}P(N+1) &= FP(N)F^T + W - FP(N)H^T[HP(N)H^T + V]^{-1}HP(N)F^T, \quad P(1) = P_1 \\ K(N) &= FP(N)H^T(HP(N)H^T + V)^{-1}\end{aligned}$$



State estimate and error covariance extrapolation:

$$\hat{x}(N)(-) = F\hat{x}(N-1)(+) + Gu(N-1)$$

$$P(N)(-) = FP(N-1)(+)F^T + W$$

Gain update:

$$K_F(N) = P(N)(-)H^T \left[HP(N)(-)H^T + V \right]^{-1}$$

State estimate update and error covariance update:

$$\hat{x}(N)(+) = \hat{x}(N)(-) + Gu(N-1) + K_F(N) (y(N) - H\hat{x}(N)(-))$$

$$P(N)(+) = [I - K_F(N)H] P(N)(-)$$



- The assumption of LTI dynamics can be relaxed.
- The above results on prediction/filtering hold *unchanged* in the case of a time-varying linear system:

$$x(t+1) = F(t)x(t) + G(t)u(t) + w(t), \quad x(1) = x_1$$

$$y(t) = H(t)x(t) + v(t)$$

$$w \approx G(0, W(t)), \quad v \approx G(0, V(t)), \quad x_{t_1} \approx G(0, P_{t_1}).$$

- In particular, both time-varying dynamics and time-varying noise variances can be handled in the Kalman filtering framework.



- State prediction:

$$\begin{aligned}\hat{x}(N+1|N) &= F(N)\hat{x}(N|N-1) + G(N)u(N) + K(N)e(N), \quad \hat{x}(1|0) = x_1 \\ \hat{y}(N|N-1) &= H(N)\hat{x}(N|N-1) \\ e(N) &= (y(N) - \hat{y}(N|N-1)).\end{aligned}$$

- Gain and prediction error variance update:

$$\begin{aligned}P(N+1) &= F(N)P(N)F(N)^T + W(N) + \\ &\quad - F(N)P(N)H(N)^T[H(N)P(N)H(N)^T + V(N)]^{-1}H(N)P(N)F(N)^T, \quad P(1) = P_1 \\ K(N) &= F(N)P(N)H(N)^T(H(N)P(N)H(N)^T + V(N))^{-1}.\end{aligned}$$



State estimate and error covariance extrapolation:

$$\hat{x}(N)(-) = F(N-1)\hat{x}(N-1)(+) + G(N-1)u(N-1)$$

$$P(N)(-) = F(N)P(N-1)(+)F(N)^T + W(N)$$

Gain update:

$$K_F(N) = P(N)(-)H(N)^T \left[H(N)P(N)(-)H(N)^T + V(N) \right]^{-1}$$

State estimate update and error covariance update:

$$\begin{aligned} \hat{x}(N)(+) = & \hat{x}(N)(-) + G(N-1)u(N-1) + \\ & + K_F(N) (y(N) - H(N)\hat{x}(N)(-)) \end{aligned}$$

$$P(N)(+) = [I - K_F(N)H(N)] P(N)(-)$$



- In the derivation of the predictor and filter we assumed that

$$E[v(N)w(N)] = 0.$$

- Also this assumption can be relaxed and the derived solutions generalised to the case when

$$E[v(N)w(N)] = Z \neq 0.$$



- The optimal solution derived so far has as main downside that the gain $K(N)$ is time-varying even in the LTI case.
- This implies that the implementation requires the propagation of $P(N)$ besides the propagation of the estimate.
- There is evidence however that in many problems after a transient the gain converges to a constant value.



If the gain $K(N)$ converges to a constant:

$$\lim_{N \rightarrow \infty} K(N) = \bar{K}$$

then the predictor

$$\begin{aligned}\hat{x}(N+1|N) &= F\hat{x}(N|N-1) + Gu(N) + \bar{K}e(N), & \hat{x}(1|0) &= x_1 \\ \hat{y}(N|N-1) &= H\hat{x}(N|N-1) \\ e(N) &= (y(N) - \hat{y}(N|N-1))\end{aligned}$$

is called the steady-state predictor.

Note that substituting $e(N)$ we have

$$\hat{x}(N+1|N) = (F - \bar{K}H)\hat{x}(N|N-1) + Gu(N) + \bar{K}y(N), \quad \hat{x}(1|0) = x_1$$

which is a LTI system.



The following questions then arise:

- Under which conditions does the gain converge?
- Does the gain converge to a stabilising value?
- If it does, how do we compute the steady-state gain?
- What is the actual performance loss incurred by considering the *steady-state* Kalman predictor/filter?

Recall that

$$K(N) = FP(N)H^T(HP(N)H^T + V)^{-1}.$$

Therefore the convergence of the gain depends on the convergence of $P(N)$.



- Consider initially the case in which the system is asymptotically stable.
- Then, we study the variance of the state sequence.
- From $x(N + 1) = F(N)x(N) + w(N), \quad x(1) = x_1$

$$\begin{aligned} x(N + 1)x^T(N + 1) &= Fx(N)x^T(N)F^T + w(N)w^T(N) + \\ &\quad + Fx(N)w^T(N) + w(N)x^T(N)F^T, \quad x(1) = x_1 \end{aligned}$$

- And taking the expectation:

$$\Lambda(N + 1) = E[x(N + 1)x^T(N + 1)] = F\Lambda(N)F^T + W, \quad \Lambda(1) = P_1.$$



Comparing

$$\Lambda(t+1) = E[x(t+1)x^T(t+1)] = F\Lambda(t)F^T + W, \quad \Lambda(1) = P_1$$

to the Riccati equation

$$P(N+1) = FP(N)F^T + W - K(N)[HP(N)H^T + V]K^T(N), \quad P(1) = P_1$$

we conclude that $P(N) \leq \Lambda(N)$.

But if the system is stable then $\lim_{N \rightarrow \infty} \Lambda(N) = \bar{\Lambda}$

and therefore also $\lim_{N \rightarrow \infty} P(N) = \bar{P}$.



Based on this argument it can be proved that:

If the system is asymptotically stable then

- The solution of the DRE converges to

$$\lim_{N \rightarrow \infty} P(N) = \bar{P} > 0$$

and the limit is independent of the initial condition.

- The corresponding steady-state predictor is asymptotically stable.



How does one compute the steady-state gain?

If $\lim_{N \rightarrow \infty} P(N) = \bar{P}$ then by definition at steady state we have

$$P(N+1) = P(N) = \bar{P}$$

and therefore the DRE

$$P(N+1) = FP(N)F^T + W - K(N)[HP(N)H^T + V]K^T(N), \quad P(1) = P_1$$

reduces to the Discrete Algebraic Riccati Equation (DARE):

$$P = FPF^T + W - FPH^T[HPH^T + V]^{-1}HPF^T.$$



Under stability assumptions, the DARE has a unique positive definite solution from which the steady-state gain can be computed:

$$P = FPF^T + W - FPH^T[HPH^T + V]^{-1}HPF^T$$

$$\bar{K} = FPH^T(HPH^T + V)^{-1}.$$



Example: the scalar case.

In the case of a first order model, the DARE reduces to

$$P = F^2 P + W - \frac{F^2 H^2 P^2}{H^2 P + V}$$

$$H^2 P^2 + VP - (F^2 P + W)(H^2 P + V) + F^2 H^2 P^2 = 0$$

$$H^2 P^2 + (V - F^2 V - H^2 W)P - WV = 0$$

$$\bar{K} = \frac{FHP}{H^2 P + V}.$$



The state equation $x(k+1) = ax(k), \quad x(0) = x_0, \quad 0 < a < 1$

has a free response given by $x(k) = a^k x_0$

which letting $a = e^{-\frac{\Delta t}{\tau}}$ $\tau = -\frac{\Delta t}{\log(a)}$

becomes $x(k) = e^{-\frac{k\Delta t}{\tau}} x_0$

and therefore the settling time in steps is

$$\Delta t = 1 \rightarrow t_A \simeq 5\tau = -\frac{5}{\log(a)}.$$

a	round(t _A)
0.9	47
0.8	22
0.7	14
0.6	10
0.5	7
0.4	5
0.3	4
0.2	3
0.1	2
0.01	1



Example: the scalar case.

Fix for example $F = 0.5$, $H = 1$ and study the effect of W and V :

$$P^2 + (0.75V - W)P - WV = 0$$

For $W=1$:

V	P	K	\bar{F}	t_A
1	1.13	0.2656	0.2344	3.4
0.1	1.02	0.4555	0.0445	1.6
0.01	1.002	0.495	0.0049	0.94



Example: the scalar case.

Fix for example $F = 0.5$, $H = 1$ and study the effect of W and V :

$$P^2 + (0.75V - W)P - WV = 0$$

For $V=1$:

W	P	K	\bar{F}	t_A
1	1.13	0.2656	0.2344	3.4
0.1	1.18	0.0569	0.4431	6.1
0.01	0.01	0.0066	0.4934	7



- In many problems however the model is not asymptotically stable.
- For example, in the single-axis attitude estimation problem the dynamic matrix is given by

$$F = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix}$$

which has both eigenvalues equal to 1.

- Nonetheless we have seen that the filter converges to a stabilising gain.



- As in the case of the Luenberger observer the structural properties of the model play a role.
- It is intuitive that for closed-loop stability the observability of (F, H) is important.
- This however is not the only condition: we look at this using an example.



Consider again the scalar case and assume that $W=0$ (no process noise in the state equation) and $P_1=0$.

The scalar DARE

$$H^2 P^2 + (V - F^2 V - H^2 W)P - WV = 0$$

in this case reduces to

$$H^2 P^2 + V(1 - F^2)P = 0$$

which has as roots

$$P = 0, \quad P = \frac{V(F^2 - 1)}{H^2}$$



- For an unstable system the optimal solution is $P=0$.
- This is consistent with the assumptions: if the state equation is deterministic then we expect null prediction error.
- This however implies $K=0$ and therefore $(F-KH)=F$ will be unstable.
- If however we add a small process noise then the null solution of the DARE vanishes and we get a non-zero gain.



Based on these arguments it can be proved that:

If the (F, H) pair is observable and the (F, G) pair is reachable, where $G : W = GG^T$ then

- The solution of the DRE converges to

$$\lim_{N \rightarrow \infty} P(N) = \bar{P} > 0$$

and the limit is independent of the initial condition.

- The corresponding steady-state predictor is asymptotically stable.



- We now turn to the case in which the system for which we want to estimate the state has continuous-time dynamics and a discrete-time measurement equation

$$\begin{aligned}\dot{x} &= Ax + Bu + w, \quad x(0) = x_0 \\ y &= Cx + Du + v\end{aligned}$$



- To use the results on the DT solution we have to relate the CT state equation and the DT one.
- We do it using simple Euler integration:

$$\dot{x} \simeq \frac{x(N+1) - x(N)}{\Delta t} = Ax(N) + v(N)$$

$$x(N+1) = (I_n + \Delta t A)x(N) + \Delta t v(N)$$

$$x(N+1) = Fx(N) + v(N), \quad F = (I_n + \Delta t A), \quad v(N) = \Delta t v_1(N)$$