Chapter 7

Elements of signal estimation

In the previous chapters we discussed the methods for designing estimators for static parameter estimation. In this chapter we consider the case of dynamic or time varying parameters (signal estimation).

7.1 Introduction

In many time-varying systems, the physical quantities of interest $\boldsymbol{x}$ can be modeled as obeying a dynamic equation

$$\boldsymbol{x}_{n+1} = \boldsymbol{f}_n(\boldsymbol{x}_n, \boldsymbol{u}_n) \tag{7.1}$$

where

- $\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots$, is a sequence of vectors in $\mathbb{R}^N$, called the *state* of the system, representing the unknown quantities of interest;
- $\boldsymbol{u}_0, \boldsymbol{u}_1, \ldots$, is a sequence of vectors in $\mathbb{R}^M$, called the *state input* of the system, representing the influencing quantities acting on $\boldsymbol{x}_n$;
- $\boldsymbol{f}_0, \boldsymbol{f}_1, \ldots$, is a sequence of functions mapping $\mathbb{R}^N \times \mathbb{R}^M$ to $\mathbb{R}^M$, called the *state equation* of the system, representing the dynamic model relating $\boldsymbol{x}_n$ and $\boldsymbol{u}_n$;

A dynamic system is such that, for any fixed k and l, $\boldsymbol{x}_k$ is completely determined from the state at time l and the inputs from times l up to $k-1$. So, complete determination of $\boldsymbol{x}_n, n = 1, 2, \ldots$ requires not only the inputs $\boldsymbol{u}_n, n = 0, 1, 2, \ldots$ but also the initial condition $\boldsymbol{x}_0$.

The equation (7.1) is called the *state equation.* Associated to this equation is the *observation equation*

$$\boldsymbol{z}_n = \boldsymbol{h}_n(\boldsymbol{x}_n, \boldsymbol{v}_n) \tag{7.2}$$

where

- $\boldsymbol{z}_0, \boldsymbol{z}_1, \ldots$, is a sequence of vectors in $\mathbb{R}^P$ representing the observable quantities;
- $\boldsymbol{v}_0, \boldsymbol{v}_1, \ldots$, is a sequence of vectors in $\mathbb{R}^P$ representing the errors on the observations;
- $\boldsymbol{h}_0, \boldsymbol{h}_1, \ldots$, is a sequence of functions mapping $\mathbb{R}^N \times \mathbb{R}^P$ to $\mathbb{R}^P$ representing the *observation model.*

The main problem then is to estimate the state vector $\boldsymbol{x}_k$ from the observations $\boldsymbol{z}_0, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_l$.

Example 1: One-dimensional motion

Consider a moving target subjected to an acceleration A_t for $t > 0$. Its position X_t and its velocity V_t at time t satisfy

$$\begin{cases} X_t = \frac{\mathrm{d}P_t}{\mathrm{d}t} \\ A_t = \frac{\mathrm{d}V_t}{\mathrm{d}t} \end{cases} \tag{7.3}$$

Assume that we can measure the position V_t at time instants $t_n = nT$ and we wish to write a model of type (7.1) describing its motion. Assuming T is small, a Taylor series approximation allows us to write

$$\begin{cases} X_{n+1} \simeq X_n + TV_n \\ V_{n+1} \simeq V_n + TA_n \end{cases} \tag{7.4}$$

From these equations we see that two quantities X_n and V_n are necessary to describe the motion. So, defining

$$\begin{cases} \boldsymbol{x} = \begin{pmatrix} X \\ V \end{pmatrix} \\ U = A \\ Z = X + V \end{cases} \longrightarrow \begin{cases} \boldsymbol{x}_n = \begin{pmatrix} X_n \\ V_n \end{pmatrix} \\ U_n = A_n \\ Z_n = X_n + V_n \end{cases} \tag{7.5}$$

we can write

$$\begin{cases} \boldsymbol{x}_{n+1} & = \boldsymbol{F}\boldsymbol{x}_n + \boldsymbol{G}U_n \\ Z_n & = \boldsymbol{H}\boldsymbol{x}_n + V_n \end{cases} \text{ with } \boldsymbol{F} = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix}, \boldsymbol{G} = \begin{pmatrix} 0 \\ T \end{pmatrix}, \boldsymbol{H} = (1 \quad 0) \tag{7.6}$$

$$\begin{cases} \boldsymbol{f}_n(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{F}\boldsymbol{x} + \boldsymbol{G}\boldsymbol{u} \\ \boldsymbol{h}_n(\boldsymbol{x}, \boldsymbol{v}) = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{v} \end{cases} \tag{7.7}$$

In this example, we assumed that we can measure directly the position of the moving target. In general, however, we may observe a quantity $z(n)$ related to the unknown quantity $x(n)$ by a linear transformation:

$$\underset{\text{non observable}}{x(n)} \longrightarrow \boxed{\text{Linear System}} \longrightarrow \underset{\text{observable}}{z(n)}$$

and we want to estimate $x(n)$ from the observed values of $\{z(n), n = 1, \ldots, k\}$. The estimate $\widehat{x}(n)$ is then a function of the data $\{z(n), n = 1, \ldots, k\}$ and we note

$$\widehat{x}(n \,|\, z(1), z(2), \ldots, z(k)) \stackrel{\text{def}}{=} \widehat{x}(n \,|\, k)$$

Three cases may occur:

- we may want to estimate $x(n+k)$ from the past observations. The estimate $\widehat{x}(n+k|n)$ is called the k-th *order prediction* of $z(n)$ and the estimation procedure is called *prediction*.

- we may want to estimate $x(n)$ from present and past observations. The estimate $\widehat{x}(n|n)$ is the *filtered value* of $z(n)$ and the estimation procedure is called *filtering*.
- we may want to estimate $x(n)$ from past, present and future observations. The estimate $\widehat{x}(n|n+l)$ is the *smoothed value* of $z(n)$ and the estimation procedure is called *smoothing*.

7.2 Kalman filtering : General linear case

In this section we consider the linear systems with finite dimensions described by the following equations:

$$\begin{cases} \boldsymbol{x}_{k+1} & = \boldsymbol{F}_k\, \boldsymbol{x}_k + \boldsymbol{G}_k\, \boldsymbol{u}_k & \text{state equation,} \\ \boldsymbol{z}_k & = \boldsymbol{H}_k\, \boldsymbol{x}_k + \boldsymbol{v}_k & \text{observation equation} \end{cases}$$

where

- $k = 0, 1, 2, \ldots$ represents the discrete time ;
- $\boldsymbol{x}_k$ is a N-dimensional vector called *state vector* of the system ;
- $\boldsymbol{z}_k$ is a P-dimensional vector containing the observations (output of the system) ;
- $\boldsymbol{v}_k$ is a P-dimensional vector containing the observations errors (output noise of the system) ;
- $\boldsymbol{u}_k$ is a M-dimensional vector representing the state representation error (state space noise process) ;
- $\boldsymbol{F}_k$, $\boldsymbol{G}_k$ and $\boldsymbol{H}_k$ with respective dimensions of (N, N), (N, M) and (P, N) are the state transition, the state input and the observation matrices and are assumed to be known.
- The noise sequences $\{\boldsymbol{u}_k\}$ and $\{\boldsymbol{v}_k\}$ are assumed to be centered, white and jointly Gaussian.
- The initial state $\boldsymbol{x}_0$ is also assumed to be Gaussian and independent of $\{\boldsymbol{u}_k\}$ and $\{\boldsymbol{v}_k\}$:

$$\mathrm{E}\left[\begin{pmatrix} \boldsymbol{v}_k \\ \boldsymbol{x}_0 \\ \boldsymbol{u}_k \end{pmatrix} (\, \boldsymbol{v}_l^t, \boldsymbol{x}_0^t, \boldsymbol{u}_l^t \,)\right] = \begin{pmatrix} \boldsymbol{R}_k & 0 & 0 \\ 0 & \boldsymbol{P}_0 & 0 \\ 0 & 0 & \boldsymbol{Q}_k \end{pmatrix} \delta_{kl}$$

  where $\boldsymbol{R}_k$ is the covariance matrix of the observation noise vector $\boldsymbol{v}_k$, $\boldsymbol{Q}_k$ is the covariance matrix of the state noise vector $\boldsymbol{Q}_k$ and $\boldsymbol{P}_0$ is the covariance matrix of the initial state $\boldsymbol{x}_0$.

Remember that the aim is to find a best estimate $\widehat{\boldsymbol{x}}_{k|l}$ of $\boldsymbol{x}_k$ from the observations $\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_l$. Depending on the relative position of k with respect to i we have:

- If $k > l$ prediction

- If $k = l$ filtering
- If $k < l$ smoothing.

Of course, for fixed k and l, this problem is not different from the vector parameter estimation of the last chapter. However, we are usually interested in producing estimates eith in real time or at least on-line for increasing k.

Three different approaches can be used to obtain the Kalman filtering equations.

- Linear Mean Square (LMS) estimation :

$$\widehat{\boldsymbol{x}}_{k|l} \stackrel{\text{def}}{=} \text{LMS}(\boldsymbol{x}_k \mid \boldsymbol{z}_1, \ldots, \boldsymbol{z}_l)$$

  which minimizes

$$\text{E}\left[[\boldsymbol{x}_k - \widehat{\boldsymbol{x}}_{k|l}]^t \, \boldsymbol{W}_k \, [\boldsymbol{x}_k - \widehat{\boldsymbol{x}}_{k|l}]\right]$$

- Maximum A posteriori (MAP) estimate :

$$\widehat{\boldsymbol{x}}_{k|l} = \arg\max_{\boldsymbol{x}} \left\{p(\boldsymbol{x}_k \mid \boldsymbol{z}_1, \ldots, \boldsymbol{z}_l)\right\}$$

- Bayesian MSE estimate :

$$\widehat{\boldsymbol{x}}_{k|l} = \text{E}\left[\boldsymbol{x}_k \mid \boldsymbol{z}_1, \ldots, \boldsymbol{z}_k\right]$$

We know that, for linear relations and Gaussian assumption all these estimates are equivalent. We consider here the last approach.

The main procedure is to apply the Bayes rule recursively to find the expression of the posterior law $p(\boldsymbol{x}_k \mid \boldsymbol{z}_1, \ldots, \boldsymbol{z}_l)$. Note that, we can obtain easily this expression thanks to the following facts :

1. All variables are assumed Gaussian ;
2. $\{\boldsymbol{u}_k\}$ and $\{\boldsymbol{v}_k\}$ are assumed white, Gaussian and mutually independent ;
3. The state and the observation models are linear ;
4. All the conditional laws such as $p(\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1})$ and $p(\boldsymbol{z}_{k+1}|\boldsymbol{z}_{1:k})$ are Gaussian. So, the posterior law

$$p(\boldsymbol{x}_{k+1}|\boldsymbol{z}_{1:k+1}) = p(\boldsymbol{x}_{k+1}|\boldsymbol{z}_{1:k}) \, \frac{p(\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1})}{p(\boldsymbol{z}_{k+1}|\boldsymbol{z}_{1:k})}$$

   is also Gaussian.

To obtain the equations in the general case we note

- $\widehat{\boldsymbol{x}}_{k|k}$ the estimate of the state vector at time k from the observations up to time k ;
- $\widehat{\boldsymbol{x}}_{k+1|k}$ the estimate of the state vector at time $k+1$ from the observations up to the instant k ;

- $\boldsymbol{e}_{k+1} = \boldsymbol{z}_{k+1} - H_{k+1}\,\widehat{\boldsymbol{x}}_{k+1|k}$ the *innovation process* of the observations at the instant $k+1$
- The covariance matrix of the innovation by
$$\boldsymbol{R}^e_{k+1} = \mathrm{E}\left[\boldsymbol{e}_{k+1|k}\,\boldsymbol{e}^t_{k+1|k}\right]$$
which is diagonal;
- The covariance matrix of the prediction error by
$$\boldsymbol{P}_{k+1|k} = \mathrm{E}\left[\left[\boldsymbol{x}_{k+1} - \widehat{\boldsymbol{x}}_{k+1|k}\right]\left[\boldsymbol{x}_{k+1} - \widehat{\boldsymbol{x}}_{k+1|k}\right]^t\right],$$
- The posterior covariance matrix of the estimation error by
$$\boldsymbol{P}_{k+1|k+1} = \mathrm{E}\left[\left[\boldsymbol{x}_{k+1} - \widehat{\boldsymbol{x}}_{k+1|k+1}\right]\left[\boldsymbol{x}_{k+1} - \widehat{\boldsymbol{x}}_{k+1|k+1}\right]^t\right]$$
which is also called the covariance matrix of the filtering error.

With these definitions, we obtain easily:

$$\begin{array}{lcl}
\mathrm{E}\left[\boldsymbol{x}_k|\boldsymbol{z}_{1:k}\right] & \stackrel{\text{def}}{=} & \widehat{\boldsymbol{x}}_{k|k} \\
\mathrm{Cov}\left\{\boldsymbol{x}_k|\boldsymbol{z}_{1:k}\right\} & \stackrel{\text{def}}{=} & \boldsymbol{P}_{k|k} \\
\mathrm{E}\left[\boldsymbol{x}_{k+1}|\boldsymbol{z}_{1:k}\right] & \stackrel{\text{def}}{=} & \widehat{\boldsymbol{x}}_{k+1|k} \\
\mathrm{Cov}\left\{\boldsymbol{x}_{k+1}|\boldsymbol{z}_{1:k}\right\} & \stackrel{\text{def}}{=} & \boldsymbol{P}_{k+1|k}
\end{array}$$

$$\begin{array}{lcl}
\mathrm{E}\left[\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1}\right] & = & \boldsymbol{H}_{k+1}\,\widehat{\boldsymbol{x}}_{k+1} \\
\mathrm{Cov}\left\{\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1}\right\} & = & \boldsymbol{R}_{k+1} \\
\\
\mathrm{E}\left[\boldsymbol{x}_{k+1}|\boldsymbol{z}_{1:k}\right] & = & \boldsymbol{F}_k\,\widehat{\boldsymbol{x}}_{k|k} \\
\mathrm{Cov}\left\{\boldsymbol{x}_{k+1}|\boldsymbol{z}_{1:k}\right\} & = & \boldsymbol{F}_k\,\boldsymbol{P}_k\boldsymbol{F}^t_k + \boldsymbol{G}_k\,\boldsymbol{Q}_k\,\boldsymbol{G}^t_k \\
\\
\mathrm{E}\left[\boldsymbol{z}_{k+1}|\boldsymbol{z}_{1:k}\right] & = & \boldsymbol{H}_{k+1}\,\boldsymbol{F}_k\,\widehat{\boldsymbol{x}}_{k|k} \\
\mathrm{Cov}\left\{\boldsymbol{z}_{k+1}|\boldsymbol{z}_{1:k}\right\} & = & \boldsymbol{H}_{k+1}\,\boldsymbol{P}_{k+1|k}\boldsymbol{H}^t_{k+1} + \boldsymbol{R}_{k+1}
\end{array}$$

Replacing the expressions of $p(\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1})$ and $p(\boldsymbol{z}_{k+1}|\boldsymbol{z}_{1:k})$ we obtain :

$$p(\boldsymbol{x}_{k+1}|\boldsymbol{z}_{1:k+1}) = A\exp\left[-\frac{1}{2}[\boldsymbol{x}_{k+1} - \widehat{\boldsymbol{x}}_{k+1|k}]^t\boldsymbol{P}^{-1}_{k+1|k+1}[\boldsymbol{x}_{k+1} - \widehat{\boldsymbol{x}}_{k+1|k}]\right]$$

with

$$A = \frac{1}{(2\pi)^{n/2}}\left\{\left|\boldsymbol{H}_{k+1}\,\boldsymbol{P}_{k+1|k}\,\boldsymbol{H}^t_{k+1} + \boldsymbol{R}_k\right|^{1/2}\left|\boldsymbol{R}_k\right|^{-1/2}\left|\boldsymbol{P}_{k+1|k}\right|\right\}^{-1/2}$$

$$\begin{array}{lcl}
\widehat{\boldsymbol{x}}_{k+1|k} & = & \boldsymbol{F}_k\,\widehat{\boldsymbol{x}}_{k|k} + \boldsymbol{P}_{k+1|k}\,\boldsymbol{H}^t_{k+1}\left[\boldsymbol{R}_{k+1} + \boldsymbol{H}_{k+1}\,\boldsymbol{P}_{k+1|k}\,\boldsymbol{H}^t_{k+1}\right]^{-1}\left[\boldsymbol{z}_{k+1} - \boldsymbol{H}_{k+1}\,\boldsymbol{F}_k\,\widehat{\boldsymbol{x}}_{k|k}\right] \\
\boldsymbol{P}_{k+1|k} & = & \boldsymbol{F}_k\,\boldsymbol{P}_{k|k}\,\boldsymbol{F}^t_k + \boldsymbol{G}_k\,\boldsymbol{Q}_k\,\boldsymbol{G}^t_k \\
\boldsymbol{P}^{-1}_{k+1|k+1} & = & \boldsymbol{P}^{-1}_{k+1|k} + \boldsymbol{H}^t_{k+1}\boldsymbol{R}^{-1}_{k+1}\,\boldsymbol{H}_{k+1}
\end{array}$$

These equations can be rewritten in many different ways. Here are two of them:

- **Prediction-Correction form :**

  - **Prediction (Time update) :**

$$\begin{aligned}\widehat{\boldsymbol{x}}_{k+1|k} &= \boldsymbol{F}_k\,\widehat{\boldsymbol{x}}_{k|k}\\ \boldsymbol{P}_{k+1|k} &= \boldsymbol{F}_k\,\boldsymbol{P}_{k|k}\,\boldsymbol{F}_k^t + \boldsymbol{G}_k\,\boldsymbol{Q}_k\,\boldsymbol{G}_k^t\end{aligned}$$

  - **Correction (measurement update) :**

$$\begin{aligned}\widehat{\boldsymbol{x}}_{k+1|k+1} &= \widehat{\boldsymbol{x}}_{k+1|k} + \boldsymbol{K}_{k+1}^g[\boldsymbol{z}_{k+1} - \boldsymbol{H}_{k+1}\,\widehat{\boldsymbol{x}}_{k+1|k}]\\ \boldsymbol{K}_{k+1}^g &= \boldsymbol{P}_{k+1|k}\,\boldsymbol{H}_{k+1}^t(\boldsymbol{R}_{k+1}^e)^{-1}\\ \boldsymbol{R}_{k+1}^e &= \boldsymbol{R}_{k+1} + \boldsymbol{H}_{k+1}\,\boldsymbol{P}_{k+1|k}\,\boldsymbol{H}_{k+1}^t\\ \boldsymbol{P}_{k+1|k+1} &= [\boldsymbol{I} - \boldsymbol{K}_{k+1}^f\boldsymbol{H}_{k+1}]\boldsymbol{P}_{k+1|k}\end{aligned}$$

- **Compact form for prediction :**

$$\begin{aligned}\widehat{\boldsymbol{x}}_{k+1|k} &= \boldsymbol{F}_k\,\widehat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k(\boldsymbol{R}_k^e)^{-1}[\boldsymbol{z}_k - \boldsymbol{H}_k\,\widehat{\boldsymbol{x}}_{k|k-1}]\\ \boldsymbol{R}_k^e &= \boldsymbol{R}_k + \boldsymbol{H}_k\,\boldsymbol{P}_{k|k-1}\,\boldsymbol{H}_k^t\\ \boldsymbol{K}_k &= \boldsymbol{F}_k\,\boldsymbol{P}_{k|k-1}\,\boldsymbol{H}_k^t\\ \boldsymbol{P}_{k+1|k} &= \boldsymbol{F}_k\,\boldsymbol{P}_{k|k-1}\,\boldsymbol{F}_k^t + \boldsymbol{G}_k\,\boldsymbol{Q}_k\,\boldsymbol{G}_k^t - \boldsymbol{K}_k(\boldsymbol{R}_k^e)^{-1}\,\boldsymbol{K}_k^t\end{aligned}$$

- **Compact form for filtering :**

$$\begin{aligned}\widehat{\boldsymbol{x}}_{k|k} &= \widehat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k^g[\boldsymbol{z}_k - \boldsymbol{H}_k\,\widehat{\boldsymbol{x}}_{k|k-1}]\\ \boldsymbol{K}_k^g &= \boldsymbol{P}_{k|k-1}\,\boldsymbol{H}_k^t[\boldsymbol{R}_k + \boldsymbol{H}_k\,\boldsymbol{P}_{k|k-1}\,\boldsymbol{H}_k^t]^{-1}\\ \boldsymbol{P}_{k|k} &= \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k^g\boldsymbol{H}_k\boldsymbol{P}_{k|k-1}\\ \boldsymbol{P}_{k+1|k} &= \boldsymbol{F}_k\boldsymbol{P}_{k|k}\,\boldsymbol{F}_k^t + \boldsymbol{G}_k\,\boldsymbol{Q}_k\,\boldsymbol{G}_k^t\end{aligned}$$

- **Very compact form for prediction :**

$$\begin{aligned}\widehat{\boldsymbol{x}}_{k+1|k} &= \boldsymbol{F}_k\widehat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k^g[\boldsymbol{z}_k - \boldsymbol{H}_k\,\widehat{\boldsymbol{x}}_{k|k-1}]\\ \boldsymbol{K}_k^g &= \boldsymbol{P}_{k|k-1}\,\boldsymbol{H}_k^t[\boldsymbol{R}_k + \boldsymbol{H}_k\,\boldsymbol{P}_{k|k-1}\,\boldsymbol{H}_k^t]^{-1}\\ \boldsymbol{P}_{k+1|k} &= \boldsymbol{F}_k\boldsymbol{P}_{k|k-1}\,\boldsymbol{F}_k^t - \boldsymbol{F}_k\boldsymbol{K}_k^g\boldsymbol{H}_k\boldsymbol{P}_{k|k-1}\boldsymbol{F}_k^t + \boldsymbol{G}_k\,\boldsymbol{Q}_k\,\boldsymbol{G}_k^t\end{aligned}$$

  where $\boldsymbol{K}_k$ is called the *Kalman filter gain* and $\boldsymbol{K}_k^g = \boldsymbol{K}_k(\boldsymbol{R}_k^e)^{-1}$ the generalized *Kalman filter gain.*

In all cases the initialization is :

$$\widehat{\boldsymbol{x}}_{0|-1} = 0 \quad \boldsymbol{P}_{0|-1} = \boldsymbol{P}_0$$

7.3 Examples

7.3.1 1D case:

$$\begin{cases} x_{n+1} &= f\,x_n + u_n \\ z_n &= h\,x_n + v_n \end{cases} \tag{7.8}$$

where u_n and v_n are assumed independent, zero-mean, white and Gaussian with known variance q and r respectively. x_0 is also assumed Gaussian with known mean m_0 and known variance p_0

$$\begin{cases} u_n &\sim \mathcal{N}(0,q) \\ v_n &\sim \mathcal{N}(0,r) \\ x_0 &\sim \mathcal{N}(m_0,p_0) \end{cases} \tag{7.9}$$

The equations in this case reduce to

$$\begin{cases} \widehat{x}_{n+1|n} &= f\,\widehat{x}_{n|n} \\ \widehat{x}_{n|n} &= \widehat{x}_{n|n-1} + k_n\left(z_n - h\,\widehat{x}_{n|n-1}\right) \\ k_n &= \frac{p_{n|n-1}h}{h^2 p_{n|n-1}+r} = \frac{1}{h}\,\frac{p_{n|n-1}}{p_{n|n-1}+r/h^2} \end{cases} \tag{7.10}$$

The role of the Kalman gain in the measurement update is easily seen from these expressions. $p_{n|n-1}$ is the MSE incurred in the estimation of x_n from $z_{0:n-1}$, and the ratio r/h^2 is a measure of noisiness of the observations. It is interesting to compare these equations with the Bayesian estimation of the signal amplitude as described in Example ().

For this particular time-invariant model, we have

$$\begin{cases} p_{n+1|n} &= f^2\,p_{n|n} + q \\ p_{n|n} &= \frac{1}{h}\,\frac{p_{n|n-1}}{\frac{r}{h^2}\,p_{n|n-1+1}} \end{cases} \tag{7.11}$$

We can eliminate the coupling between these equations and obtain

$$p_{n+1|n} = \frac{f^2\,p_{n|n-1}}{\frac{h^2}{r}\,p_{n|n-1}+1} + q \tag{7.12}$$

We see here that as n increases, $p_{n+1|n}$ and so the gain k_n approaches a constant. Note that if $p_{n+1|n}$ does approach a constant, say p_∞, then p_∞ must satisfy

$$p_\infty = \frac{f^2\,p_\infty}{\frac{h^2}{r}\,p_\infty + 1} + q \tag{7.13}$$

This equation is quadratic and has a unique positive solution

$$p_\infty = \frac{1}{2}\left\{\left[\frac{r}{h^2}(1-f^2) - q\right]^2 + \frac{4rq}{h^2}\right\}^{1/2} - \frac{r}{2h^2}(1-f^2) + q \tag{7.14}$$

$$\begin{aligned} \left|p_{n+1|n} - p_\infty\right| &= f^2\left|\frac{p_{n|n-1}}{\frac{h^2}{r}\,p_{n|n-1}+1} - \frac{p_\infty}{\frac{h^2}{r}\,p_\infty+1}\right| \\ &\leq f^2\left|p_{n|n-1} - p_\infty\right| \end{aligned}$$

$$
\begin{aligned}
\left|p_{n+1|n} - p_\infty\right| \le f^{2(n+1)} \left|p_0 - p_\infty\right| \\
\le \quad f^2 \left|p_{n|n-1} - p_\infty\right|
\end{aligned}
$$

This means that if $|f| < 1$ then $p_{n+1|n}$ converges to p_∞. So $|f| < 1$ is a sufficient condition for Kalman-Bucy filter to approach a steady state.

7.3.2 Track-While-Scan (TWS) Radar

Let consider the example of one dimentional moving target and assume that the target is subject to random acceleration A_n. Then we have

$$
\begin{aligned}
\begin{pmatrix} X_{n+1} \\ V_{n+1} \end{pmatrix} &= \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} \begin{pmatrix} X_n \\ V_n \end{pmatrix} + \begin{pmatrix} 0 \\ T \end{pmatrix} A_n \\
Z_n &= (\,1 \quad 0\,) \begin{pmatrix} X_n \\ V_n \end{pmatrix} + e_n
\end{aligned}
$$

For a more general case in 3D we have a state vector with 6 components (3 positions and 3 velocities). But, if we assume that the measurement noise in 3 dimensions are independent of one another and independent to the components of the acceleration, the problem can be treated as 3 independent one-dimensional moving target.

The Kalman equations for this simple model become

$$
\begin{aligned}
\begin{pmatrix} \widehat{X}_{n+1|n} \\ \widehat{V}_{n+1|n} \end{pmatrix} &= \begin{pmatrix} \widehat{X}_{n|n} + T\widehat{V}_{n|n} \\ \widehat{V}_{n|n} \end{pmatrix} \\
\begin{pmatrix} \widehat{X}_{n|n} \\ \widehat{V}_{n|n} \end{pmatrix} &= \begin{pmatrix} \widehat{X}_{n|n-1} \\ \widehat{V}_{n|n-1} \end{pmatrix} + \begin{pmatrix} K_{n,1} \\ K_{n,2} \end{pmatrix} (\, Z_n - \widehat{X}_{n|n-1} \,) \\
\begin{pmatrix} K_{n,1} \\ K_{n,2} \end{pmatrix} &= \begin{pmatrix} \frac{P(1,1)}{P(1,1)+r} \\ \frac{P(2,1)}{P(1,1)+r} \end{pmatrix}
\end{aligned}
$$

where $P(k,l)$ is the $(k-l)$th component of the matrix $P_{n|n-1}$.

To reduce the computation, the time varying elements of the Kalman gain vector can be replaced with some constants (the steady states values) to obtain

$$
\begin{pmatrix} \widehat{X}_{n|n} \\ \widehat{V}_{n|n} \end{pmatrix} = \begin{pmatrix} \widehat{X}_{n|n-1} \\ \widehat{V}_{n|n-1} \end{pmatrix} + \begin{pmatrix} \alpha \\ \beta/T \end{pmatrix} (\, Z_n - \widehat{X}_{n|n-1} \,) \tag{7.15}
$$

with constatnt values for α and β.

7.3.3 Track-While-Scan (TWS) Radar with dependent acceleration sequences

The simple model of the previous example is not very realistic, because there was assumed that the target is subjected to random acceleration. For a heavy target, we can do a little better by assuming that the acceleration A_n is modeled as

$$
A_{n+1} = \rho A_n + W_n, \quad n = 0, 1, \ldots
$$

a first order autoregressive (AR) model. The value of ρ can be choosed in accordance of the target. ρ near to 0 means a very low inertia target and ρ near to 1 means a high inertia target.

To account for this equation, we can extend the state vector and we obtain

$$\begin{aligned}\begin{pmatrix} X_{n+1} \\ V_{n+1} \\ A_{n+1} \end{pmatrix} &= \begin{pmatrix} 1 & T & 0 \\ 0 & 1 & T \\ 0 & 0 & \rho \end{pmatrix} \begin{pmatrix} X_n \\ V_n \\ A_n \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} W_n \\ Z_n &= \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} X_n \\ V_n \\ A_n \end{pmatrix} + e_n\end{aligned}$$

We again can apply the Kalman equations to this model and obtain:

$$\begin{aligned}\begin{pmatrix} \widehat{X}_{n+1|n} \\ \widehat{V}_{n+1|n} \\ \widehat{A}_{n+1|n} \end{pmatrix} &= \begin{pmatrix} \widehat{X}_{n|n} + T\widehat{V}_{n|n} \\ \widehat{V}_{n|n} + T\widehat{A}_{n|n} \\ \rho\widehat{A}_{n|n} \end{pmatrix} \\ \begin{pmatrix} \widehat{X}_{n|n} \\ \widehat{V}_{n|n} \\ \widehat{A}_{n|n} \end{pmatrix} &= \begin{pmatrix} \widehat{X}_{n|n-1} \\ \widehat{V}_{n|n-1} \\ \widehat{A}_{n|n-1} \end{pmatrix} + \begin{pmatrix} K_{n,1} \\ K_{n,2} \\ K_{n,3} \end{pmatrix} (\, Z_n - \widehat{X}_{n|n-1} \,) \\ \begin{pmatrix} K_{n,1} \\ K_{n,2} \\ K_{n,3} \end{pmatrix} &= \begin{pmatrix} \frac{P(1,1)}{P(1,1)+r} \\ \frac{P(2,1)}{P(1,1)+r} \; \frac{P(3,1)}{P(1,1)+r} \end{pmatrix}\end{aligned}$$

where $P(k,l)$ is the $(k-l)$th component of the matrix $P_{n|n-1}$.

Again to reduce the computation, the time varying elements of the Kalman gain vector can be replaced with some constants related to its steady state value and obtain

$$\begin{pmatrix} \widehat{X}_{n|n} \\ \widehat{V}_{n|n} \\ \widehat{A}_{n|n} \end{pmatrix} \begin{pmatrix} \widehat{X}_{n|n-1} \\ \widehat{V}_{n|n-1} \\ \widehat{A}_{n|n-1} \end{pmatrix} + \begin{pmatrix} \alpha \\ \beta/T \\ \gamma/T^2 \end{pmatrix} (\, Z_n - \widehat{X}_{n|n-1} \,) \tag{7.16}$$

with constant values for α, β and γ.

7.4 Fast Kalman filter equations

The general equations of the Kalman filtering do not assume any stationarity of the system and all the matrices of the system $\boldsymbol{F}$, $\boldsymbol{G}$ and $\boldsymbol{H}$, and also $\boldsymbol{R}$ and $\boldsymbol{Q}$ may depend on the index k.

Through the simple example in previous section, we saw that, when these quantities are independent of k, *i.e.*

$$\begin{aligned}
\widehat{\boldsymbol{x}}_{k+1|k} &= \boldsymbol{F}\widehat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k(\boldsymbol{R}_k^e)^{-1}[\boldsymbol{z}_k - \boldsymbol{H}\widehat{\boldsymbol{x}}_{k|k-1}] \\
\boldsymbol{R}_k^e &= \boldsymbol{R} + \boldsymbol{H}\boldsymbol{P}_{k|k-1}\boldsymbol{H}^t \\
\boldsymbol{K}_k &= \boldsymbol{F}\boldsymbol{P}_{k|k-1}\boldsymbol{H}^t \\
\boldsymbol{P}_{k+1|k} &= \boldsymbol{F}\boldsymbol{P}_{k|k-1}\boldsymbol{F}^t + \boldsymbol{G}\boldsymbol{Q}\boldsymbol{G}^t - \boldsymbol{K}_k(\boldsymbol{R}_k^e)^{-1}\boldsymbol{K}_k^t
\end{aligned}$$

the system can reach a stationnary point and we can use the steady state values of the gain or an approximation to it to reduce the calculation cost. But doing so does not give an optimal estimate and may not give a very satisfactory solution. Here, we present shortly a slightly better way to obtain fast algorithms without loosing too much its optimality.

Let assume a constatnt system and note by $\delta\boldsymbol{P}_k$, $\delta\boldsymbol{K}_k^g$ and $\delta\boldsymbol{R}_k^e$ the increments

$$\begin{aligned}
\delta\boldsymbol{P}_k &= \boldsymbol{P}_{k|k-1} - \boldsymbol{P}_{k-1|k-2} \\
\delta\boldsymbol{K}_k^g &= \boldsymbol{K}_k^g - \boldsymbol{K}_{k-1}^g \\
\delta\boldsymbol{R}_k^e &= \boldsymbol{R}_k^e - \boldsymbol{R}_{k-1}^e
\end{aligned}$$

Then, it can be shown that the $\delta\boldsymbol{P}_{k+1}$ can be factorized by

$$\begin{aligned}
\delta\boldsymbol{P}_{k+1} &= [\boldsymbol{F} - \boldsymbol{K}_{k-1}^g\boldsymbol{H}][\delta\boldsymbol{P}_k - \delta\boldsymbol{P}_k\boldsymbol{H}^t(\boldsymbol{R}_k^e)^{-1}\boldsymbol{H}\boldsymbol{P}_k][\boldsymbol{F} - \boldsymbol{K}_{k-1}^g\boldsymbol{H}]^t \\
&= [\boldsymbol{F} - \boldsymbol{K}_k^g\boldsymbol{H}][\delta\boldsymbol{P}_k + \delta\boldsymbol{P}_k\boldsymbol{H}^t(\boldsymbol{R}_{k-1}^e)^{-1}\boldsymbol{H}\boldsymbol{P}_k][\boldsymbol{F} - \boldsymbol{K}_k^g\boldsymbol{H}]^t
\end{aligned}$$

Then, it is possible to reduce the cost of the calculations by noting that if $\delta\boldsymbol{P}_k$ can be factorized as:

$$\delta\boldsymbol{P}_1 = \boldsymbol{z}_0\boldsymbol{M}_0\boldsymbol{z}_0^t,$$

then $\delta\boldsymbol{P}_{k+1}$ can also be factorized as

$$\delta\boldsymbol{P}_{k+1} = \boldsymbol{z}_k\boldsymbol{M}_k\boldsymbol{z}_k^t$$

and we obtain then the following equations:

$$\begin{aligned}
\delta\boldsymbol{P}_{k+1} &= \boldsymbol{z}_k\boldsymbol{M}_k\boldsymbol{z}_k^t \\
\boldsymbol{z}_k &= [\boldsymbol{F} - \boldsymbol{K}_k^g\boldsymbol{H}]\boldsymbol{z}_{k-1} \\
\boldsymbol{M}_k &= \boldsymbol{M}_{k-1} + \boldsymbol{M}_{k-1}\boldsymbol{z}_{k-1}^t\boldsymbol{H}^t(\boldsymbol{R}_{k-1}^e)^{-1}\boldsymbol{H}\boldsymbol{z}_{k-1}\boldsymbol{M}_{k-1} \\
\boldsymbol{R}_{k-1}^e &= \boldsymbol{R}_k^e + \boldsymbol{H}\boldsymbol{z}_k\boldsymbol{M}_k\boldsymbol{z}_k^t\boldsymbol{H}^t \\
\boldsymbol{K}_{k+1} &= \boldsymbol{K}_{k+1}^g\boldsymbol{R}_{k+1}^e = \boldsymbol{K}_k + \boldsymbol{F}\boldsymbol{z}_k\boldsymbol{M}_k\boldsymbol{z}_k^t\boldsymbol{H}^t \\
\boldsymbol{P}_{k|k-1} &= \boldsymbol{P}_0 + \sum_{j=0}^{k-1}\boldsymbol{z}_j\boldsymbol{M}_j\boldsymbol{z}_j^t
\end{aligned}$$

which are called *Chandrasekhar* equations.

Note that if $\alpha = \text{rang}\,\{\delta \boldsymbol{P}_1\}$ where

$$\delta \boldsymbol{P}_1 = \boldsymbol{F}\boldsymbol{P}_0\boldsymbol{F}^t + \boldsymbol{G}\boldsymbol{Q}\boldsymbol{G}^t - \boldsymbol{K}_0(\boldsymbol{R}_0^e)^{-1}\boldsymbol{K}_0^t - \boldsymbol{P}_0$$

then $\boldsymbol{z}_k$ has dimensions (N, α), $\boldsymbol{M}_k$ has dimensions (α, α). So, in place of updating, at each time k the matrix $\boldsymbol{P}_k$ with dimensions (N, N) we only have to update the matrixes $\boldsymbol{z}_k$ and $\boldsymbol{M}_k$ with dimensions (N, α) and (α, α).

Note also that $\boldsymbol{M}_0$ is the signature matrix of $\delta \boldsymbol{P}_1$ and the value of α depends on the choice of the initial covariance matrix $\boldsymbol{P}_0$. It is not unusual to have $\alpha = 1$ which greately reduces the computation cost.

7.5 Kalman filter equations for signal deconvolution

Starting by the convolution equation:

$$z(k) = \sum_{i=0}^{p-1} h(i)x(k-i) + b(k)$$

rewritten in matrix form

$$\begin{pmatrix} z(1) \\ \vdots \\ z(k) \\ \vdots \\ z(M) \end{pmatrix} = \begin{pmatrix} h_{(p-1)} & \cdots & h_{(0)} & \cdots & \cdots \\ \vdots & & & & \vdots \\ 0 & h_{(p-1)} & \cdots & h_{(0)} & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & h_{(p-1)} & \cdots & h_{(0)} \end{pmatrix} \begin{pmatrix} x(-p) \\ \vdots \\ x(0) \\ \vdots \\ x(M) \end{pmatrix} + \begin{pmatrix} b(1) \\ \vdots \\ b(k) \\ \vdots \\ z(M) \end{pmatrix}$$

we can propose the following models:

- **Constant state vector model**

$$\begin{cases} \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k = \boldsymbol{x} = [x_{-p}, \ldots, x_{-1}, x_0, x_1, \ldots, x_n]^t \\ z_k &= \boldsymbol{h}_k^t \cdot \boldsymbol{x}_k + v_k \end{cases}$$

$$\boldsymbol{h}_k = (\,0 \quad 0 \quad 0 \quad h_{p-1} \quad \ldots \quad h_0 \quad 0 \quad 0\,)^t$$

where coefficient h_0 is in the k-th position. Then we have

$$\begin{cases} \boldsymbol{u}_k &= \boldsymbol{0} \\ \boldsymbol{F}_k &= \boldsymbol{G}_k = \boldsymbol{I} \quad \text{with } \boldsymbol{D} = \\ \boldsymbol{h}_{k+1}^t &= \boldsymbol{D}\boldsymbol{h}_k^t \end{cases} \begin{pmatrix} 0 & \ldots & \ldots & 0 & 0 \\ 1 & 0 & \ldots & 0 & 0 \\ 0 & 1 & 0 & \vdots & \vdots \\ \vdots & & & \vdots & \vdots \\ 0 & & \ldots & 1 & 0 \end{pmatrix}$$

If we note by

$$\mathrm{E}\,[\boldsymbol{x}] = \boldsymbol{x}_0 \qquad \mathrm{E}\left[[\boldsymbol{x}-\boldsymbol{x}_0][\boldsymbol{x}-\boldsymbol{x}_0]^t\right] = \boldsymbol{P}_0$$

$$\mathrm{E}\,[v_k] = 0 \qquad \mathrm{E}\,[v_k v_j] = r\,\delta_{kj}$$

we obtain

$$\begin{aligned} \widehat{\boldsymbol{x}}_{k|k} &= \widehat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k (r_k^e)^{-1} [y_k - \boldsymbol{h}_k^t \cdot \widehat{\boldsymbol{x}}_{k|k-1}] \\ r_k^e &= r + \boldsymbol{h}_k^t \boldsymbol{P}_{k|k-1}\,\boldsymbol{h}_k \\ \boldsymbol{K}_k &= \boldsymbol{P}_{k|k-1}\,\boldsymbol{h}_k^t \\ \boldsymbol{P}_{k+1|k} &= \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k (r_k^e)^{-1}\,\boldsymbol{K}_k^t \end{aligned}$$

Note that the observations y_k are scalar. So, r_k^e is also a scalar quantity, but $\boldsymbol{x}$ is a N-dimensional vector and so the covariance matrix $\boldsymbol{P}$ has the dimensions $(N \times N)$.

- **Non constant state space model**

  we can choose

$$\boldsymbol{h} = [h_0, \ldots, h_{p-1}]^t, \quad \boldsymbol{x}_k = [x_k, x_{k-1}, \ldots, x_{k-p+1}]^t$$

$$z(k) = \boldsymbol{h}^t \boldsymbol{x}_k + b(k), \quad \text{dimension of } \boldsymbol{x}_k = p \leq N$$

But now we need to introduce a generating state space model for $\boldsymbol{x}_k$.

One of such models is an AR model :

$$x(n+1) = \sum_{i=1}^{q} a(i)\, x(n-i+1) + u(n+1)$$

where

$$\mathrm{E}[u_n] = 0, \quad \mathrm{E}\left[|u_n|^2\right] = \beta^2, \quad \mathrm{E}[u_m u_n] = 0, \quad m \neq n$$

It is easy then to see that we can write

$$\begin{pmatrix} x(n+1) \\ x(n) \\ \vdots \\ \vdots \\ x(n-q+2) \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & \ldots & \ldots & a_q \\ 1 & 0 & \ldots & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \\ \vdots & \vdots & \vdots & \vdots & \\ \vdots & \vdots & \vdots & 1 & 0 \end{pmatrix} \begin{pmatrix} x(n) \\ x(n-1) \\ \vdots \\ \vdots \\ x(n-q+1) \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix} u(n+1)$$

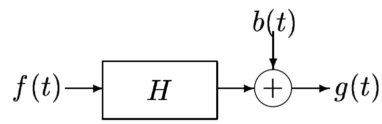
Thus we have

$$\begin{cases} \boldsymbol{x}_{k+1} & = & \boldsymbol{F}\boldsymbol{x}_k + \boldsymbol{G}\boldsymbol{u}_{k+1} \\ y_k & = & \boldsymbol{h}^t \cdot \boldsymbol{x}_k + b_k \end{cases}$$

$$\boldsymbol{F} = \begin{pmatrix} a_1 & a_2 & \ldots & \ldots & a_q \\ 1 & 0 & \ldots & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \\ \vdots & \vdots & \vdots & \vdots & \\ \vdots & \vdots & \vdots & 1 & 0 \end{pmatrix}, \quad \boldsymbol{G} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

This model has the advantage that $\boldsymbol{F}$, $\boldsymbol{G}$ and $\boldsymbol{H}$ are constant and we can use fast algorithms, but the main drawback in practical applications is the determination of q and a_k, $k = 1, \ldots, q$.

Consider the following convolution model :



$$g(t) = \int f(t')h(t-t')\,\mathrm{d}t' + b(t) = \int h(t')f(t-t')\,\mathrm{d}t' + b(t)$$

and assume the following hypotheses :

- The signals $f(t)$, $g(t)$, $h(t)$ are discretized with the same sampling period $\Delta T = 1$,
- The impulse response is finite (FIR) : $h(t) = 0$, for t such that $t < -q\Delta T$ or $\forall t > p\Delta T$.

Then we have :

$$g(m) = \sum_{k=-q}^{p} h(k)f(m-k) + b(m), \quad m = 0, \cdots, M$$

or in a matrix form

$$\begin{pmatrix} g(0) \\ g(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ g(M) \end{pmatrix} = \begin{pmatrix} h(p) & \cdots & h(0) & \cdots & h(-q) & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & & \ddots & & \ddots & & & & \vdots \\ \vdots & & h(p) & \cdots & h(0) & \cdots & h(-q) & & & \vdots \\ \vdots & & & \ddots & & & \ddots & & & \vdots \\ \vdots & & & & \ddots & & & & & 0 \\ 0 & \cdots & \cdots & & 0 & h(p) & \cdots & h(0) & \cdots & h(-q) \end{pmatrix} \begin{pmatrix} f(-p) \\ \vdots \\ f(0) \\ f(1) \\ \vdots \\ f(M) \\ f(M+1) \\ \vdots \\ f(M+q) \end{pmatrix}$$

or

$$\boldsymbol{g} = \boldsymbol{H}\boldsymbol{f} + \boldsymbol{v}$$

Note that $\boldsymbol{g}$ is a $(M+1)$-dimensional vector, $\boldsymbol{f}$ has dimension $M+p+q+1$, $\boldsymbol{h} = [h(p), \cdots, h(0), \cdots, h(-q)]$ has dimension $(p+q+1)$ and matrix $\boldsymbol{H}$ has dimensions $(M+1) \times (M+p+q+1)$.

Now, if we assume that the system is causal ($q = 0$) we obtain

$$\begin{pmatrix} g(0) \\ g(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ g(M) \end{pmatrix} = \begin{pmatrix} h(p) & \cdots & h(0) & 0 & \cdots & \cdots & 0 \\ 0 & & & & & & \vdots \\ \vdots & & h(p) & \cdots & h(0) & & \vdots \\ \vdots & & & & & & 0 \\ 0 & \cdots & \cdots & 0 & h(p) & \cdots & h(0) \end{pmatrix} \begin{pmatrix} f(-p) \\ \vdots \\ f(0) \\ f(1) \\ \vdots \\ \vdots \\ f(M) \end{pmatrix}$$

If the input signal is also assumed to be causal, we obtain :

$$\begin{pmatrix} g(0) \\ g(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ g(M) \end{pmatrix} = \begin{pmatrix} h(0) & & & & & \\ h(1) & \ddots & & & & \\ \vdots & & & & & \\ h(p) & \cdots & & h(0) & & \\ 0 & \ddots & & & \ddots & \\ \vdots & & & & & \\ 0 & \cdots & 0 & h(p) & \cdots & h(0) \end{pmatrix} \begin{pmatrix} f(0) \\ f(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ f(M) \end{pmatrix} \tag{7.17}$$

and finally if $p = M$ we have :

$$\begin{pmatrix} g(0) \\ g(1) \\ \vdots \\ g(M) \end{pmatrix} = \begin{pmatrix} h(0) & & & \\ h(1) & h(0) & & \\ \vdots & & \ddots & \\ h(M) & \cdots & h(1) & h(0) \end{pmatrix} \begin{pmatrix} f(0) \\ f(1) \\ \vdots \\ f(M) \end{pmatrix}$$

Remark that, in all cases matrix $\boldsymbol{H}$ is TOEPLITZ.

In the case where the input signal and the system are both causal, (7.17) can be rewritten as

$$\begin{pmatrix} g(0) \\ g(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ g(M) \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} h(0) & 0 & \cdots & & & 0 & & h(p) & \cdots & h(1) \\ h(1) & \ddots & & & & & & & \ddots & \vdots \\ \vdots & & & & & & & & & h(p) \\ h(p) & \cdots & & h(0) & 0 & & & & & 0 \\ 0 & \ddots & & & \ddots & & & & & \vdots \\ \vdots & & & & & & & & & \\ 0 & \cdots & 0 & h(p) & \cdots & & & h(0) & & 0 \\ \vdots & & & & & & & & & \vdots \\ & & & & \ddots & & & & \ddots & 0 \\ 0 & \cdots & & \cdots & 0 & h(p) & \cdots & & & h(0) \end{pmatrix} \begin{pmatrix} f(0) \\ f(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ f(M) \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

where $\boldsymbol{f}$ and $\boldsymbol{g}$ have been completed artificially by some zeros. This operation is called zero-filling and the main advantage to do so is that the matrix $\boldsymbol{H}$ is now a circulant matrix.

Starting by the Kalman filter equations:

$$\begin{cases} \boldsymbol{z}_k & = \boldsymbol{H}_k\, \boldsymbol{x}_k + \boldsymbol{v}_k \quad \text{observation equation} \\ \boldsymbol{x}_{k+1} & = \boldsymbol{F}_k\, \boldsymbol{x}_k + \boldsymbol{G}_k\, \boldsymbol{u}_k \quad \text{state equation} \end{cases}$$

$$\begin{aligned}
\widehat{\boldsymbol{x}}_{k+1|k} &= \boldsymbol{F}_k\, \widehat{\boldsymbol{x}}_{k|k} \\
\boldsymbol{P}_{k+1|k} &= \boldsymbol{F}_k\, \boldsymbol{P}_{k|k}\, \boldsymbol{F}_k^t + \boldsymbol{G}_k\, \boldsymbol{Q}_k\, \boldsymbol{G}_k^t \\
\widehat{\boldsymbol{x}}_{k+1|k+1} &= \widehat{\boldsymbol{x}}_{k+1|k} + \boldsymbol{K}_{k+1}^f [\boldsymbol{z}_{k+1} - \boldsymbol{H}_{k+1}\, \widehat{\boldsymbol{x}}_{k+1|k}] \\
\boldsymbol{K}_{k+1}^f &= \boldsymbol{P}_{k+1|k}\, \boldsymbol{H}_{k+1}^t (\boldsymbol{R}_{k+1}^e)^{-1} \\
\boldsymbol{R}_{k+1}^e &= \boldsymbol{R}_{k+1} + \boldsymbol{H}_{k+1}\, \boldsymbol{P}_{k+1|k}\, \boldsymbol{H}_{k+1}^t \\
\boldsymbol{P}_{k+1|k+1} &= [\boldsymbol{I} - \boldsymbol{K}_{k+1}^f \boldsymbol{H}_{k+1}] \boldsymbol{P}_{k+1|k}
\end{aligned}$$

7.5.1 AR, MA and ARMA Models

AR model

$$u(n) = \sum_{k=1}^{p} a(k)\, u(n-k) + \epsilon(n), \quad \forall n$$

$$\mathrm{E}\left[\epsilon(n)\right] = 0, \quad \mathrm{E}\left[|\epsilon(n)|^2\right] = \beta^2,$$

$$\mathrm{E}\left[\epsilon(n)\, u(m)\right] = 0, \quad m \neq n$$

$$\epsilon(n) \longrightarrow \boxed{H(z) = \frac{1}{A(z)} = \frac{1}{1 + \sum_{k=1}^{p} a(k) z^{-k}}} \longrightarrow u(n)$$

MA model

$$u(n) = \sum_{k=0}^{q} b(k)\, \epsilon(n-k), \quad \forall n$$

$$\epsilon(n) \longrightarrow \boxed{B(z) = \sum_{k=0}^{q} b(k) z^{-k}} \longrightarrow u(n)$$

ARMA model

$$u(n) = \sum_{k=1}^{p} a(k)\, u(n-k) + \sum_{l=0}^{q} b(l)\, \epsilon(n-l)$$

$$\epsilon(n) \longrightarrow \boxed{H(z) = \frac{B(z)}{A(z)} = \frac{\sum_{k=0}^{q} b(k) z^{-k}}{1 + \sum_{k=1}^{p} a(k) z^{-k}}} \longrightarrow u(n)$$

$$\epsilon(n) \longrightarrow \boxed{H(z) = B_q(z)} \longrightarrow \boxed{H(z) = \frac{1}{A_p(z)}} \longrightarrow u(n)$$

In a dynamic system, in general, we are interested in a physical quantity $\boldsymbol{x}$ through the observation of a quantity $\boldsymbol{z}$ related to $\boldsymbol{x}$ by the following system of equations

$$\left\{ \begin{array}{ll} \boldsymbol{x}_{n+1} & = \boldsymbol{f}_n(\boldsymbol{x}_n, \boldsymbol{u}_n) \\ \boldsymbol{z}_n & = \boldsymbol{h}_n(\boldsymbol{x}_n, \boldsymbol{v}_n) \end{array} \right. \tag{7.18}$$