Sequential processing of measurements

- The standard Kalman filter is an efficient recursive algorithm.
- However, under some conditions, we can improve its speed by rewriting some of its equations.
- One of the computationally intensive operations in the Kalman filter is the matrix inverse operation in Step 2a: $L_k = \Sigma_{\tilde{x},k}^{-} C_k^T \Sigma_{\tilde{z},k}^{-1}$.
- Inverting $\Sigma_{\tilde{z},k}^{-1}$ via Gaussian elimination (the most straightforward approach), it is an $\mathcal{O}(m^3)$ operation, where m is the dimension of the measurement vector.
- If there is a single sensor, this matrix inverse becomes a scalar division, which is an $\mathcal{O}(1)$ operation—very fast; otherwise it is slow.
- Therefore, if we can break the m measurements into m single-sensor measurements and update the Kalman filter that way, there is opportunity for significant computational savings.

Start by assuming v_k are uncorrelated

- We start by assuming that the sensor measurements are uncorrelated. This implies that:

$$\Sigma_{\tilde{v}} = \operatorname{diag}\left[\ \sigma_{\tilde{v}_1}^2,\quad \cdots,\quad \sigma_{\tilde{v}_m}^2\ \right].$$

- We will use colon ":" notation to refer to the measurement number. For example, $z_{k:1}$ is the measurement from sensor 1 at time k.
- Then, the measurement is

$$z_k = \begin{bmatrix} z_{k:1} \\ \vdots \\ z_{k:m} \end{bmatrix} = C_k x_k + v_k = \begin{bmatrix} C_{k:1}^T x_k + v_{k:1} \\ \vdots \\ C_{k:m}^T x_k + v_{k:m} \end{bmatrix},$$

where $C_{k:1}^T$ is the first row of C_k (for example), and $v_{k:1}$ is the sensor noise of the first sensor at time k, for example.

The i th gain matrix

- We will consider z_k to be sequence of scalar measurements $z_{k:1} \ldots z_{k:m}$ and update the state estimate and covariance estimates in m steps.
- We initialize the measurement update process with $\hat{x}_{k:0}^{+} = \hat{x}_k^{-}$ and $\Sigma_{\tilde{x},k:0}^{+} = \Sigma_{\tilde{x},k}^{-}$.
- Consider the measurement update for the ith measurement, $z_{k:i}$:

$$\begin{aligned} \hat{x}_{k:i}^{+} &= \mathbb{E}[x_k \mid \mathbb{Z}_{k-1}, z_{k:1} \ldots z_{k:i}] \\ &= \mathbb{E}[x_k \mid \mathbb{Z}_{k-1}, z_{k:1} \ldots z_{k:i-1}] + L_{k:i}(z_{k:i} - \mathbb{E}[z_k \mid \mathbb{Z}_{k-1}, z_{k:1} \ldots z_{k:i-1}]) \\ &= \hat{x}_{k:i-1}^{+} + L_{k:i}(z_{k:i} - C_{k:i}^T \hat{x}_{k:i-1}^{+}). \end{aligned}$$

- Generalizing from before:

$$L_{k:i} = \mathbb{E}[\tilde{x}_{k:i-1}^{+} \tilde{z}_{k:i}^T] \Sigma_{\tilde{z}_{k:i}}^{-1}.$$

State and error-covariance update

- Next, we recognize that the variance of the innovation corresponding to measurement $z_{k:i}$ is:

$$\Sigma_{\tilde{z}_{k:i}} = \sigma^2_{\tilde{z}_{k:i}} = C_{k:i}^T \Sigma^+_{\tilde{x},k:i-1} C_{k:i} + \sigma^2_{\tilde{v}_i}.$$

- The corresponding gain is $L_{k:i} = \Sigma^+_{\tilde{x},k:i-1} C_{k:i} / \sigma^2_{\tilde{z}_{k:i}}$, and the updated state is:

$$\hat{x}^+_{k:i} = \hat{x}^+_{k:i-1} + L_{k:i}\left[z_{k:i} - C_{k:i}^T \hat{x}^+_{k:i-1}\right]$$

  having covariance: $\Sigma^+_{\tilde{x},k:i} = \Sigma^+_{\tilde{x},k:i-1} - L_{k:i} C_{k:i}^T \Sigma^+_{\tilde{x},k:i-1}$.
- The covariance update can be implemented as:

$$\Sigma^+_{\tilde{x},k:i} = \Sigma^+_{\tilde{x},k:i-1} - \Sigma^+_{\tilde{x},k:i-1} C_{k:i} C_{k:i}^T \Sigma^+_{\tilde{x},k:i-1} / \big(C_{k:i}^T \Sigma^+_{\tilde{x},k:i-1} C_{k:i} + \sigma^2_{\tilde{v}_i}\big).$$

- The final measurement update gives $\hat{x}^+_k = \hat{x}^+_{k:m}$ and $\Sigma^+_{\tilde{x},k} = \Sigma^+_{\tilde{x},k:m}$.

Sequentially processing correlated measurements

- This process must be modified to accommodate situations where sensor noise is correlated among the measurements.
- Assume that we can factor the matrix $\Sigma_{\tilde{v}} = \mathcal{S}_v \mathcal{S}_v^T$, where $\mathcal{S}_v$ is a lower-triangular matrix (for symmetric positive-definite $\Sigma_{\tilde{v}}$, we can).
  - Recall that the factor $\mathcal{S}_v$ is a kind of a matrix square root, known as the "Cholesky" factor of the original matrix.
  - In Octave, `Sv = chol(SigmaV,'lower');`
  - Be careful: Octave's default answer (without specifying "`lower`") is an upper-triangular matrix, which is not what we're after.
- The Cholesky factor has strictly positive elements on its diagonal (positive eigenvalues), so is guaranteed to be invertible.

Decorrelating correlated measurement noise

- Consider a modification to the output equation of a system having correlated measurements:

$$\begin{aligned} z_k &= C x_k + v_k \\ \bar{z}_k &= \mathcal{S}_v^{-1} z_k = \mathcal{S}_v^{-1} C x_k + \mathcal{S}_v^{-1} v_k = \bar{C} x_k + \bar{v}_k. \end{aligned}$$

  - Note that I am not using the "bar" symbol to indicate the mean of a quantity here.
  - Instead, $\bar{z}_k$ is a computed output, similar in interpretation to measured output z_k.
- Consider now the covariance of the modified noise input $\bar{v}_k = \mathcal{S}_v^{-1} v_k$:

$$\begin{aligned} \Sigma_{\tilde{\bar{v}}_k} &= \mathbb{E}[\bar{v}_k \bar{v}_k^T] \\ &= \mathbb{E}[\mathcal{S}_v^{-1} v_k v_k^T \mathcal{S}_v^{-T}] = \mathcal{S}_v^{-1} \Sigma_{\tilde{v}} \mathcal{S}_v^{-T} = I. \end{aligned}$$

- Therefore, we have identified a transformation that de-correlates (and normalizes) measurement noise.

Revised measurement-update process

- Using this revised output equation, we use the prior method.
- We start the process with $\hat{x}^+_{k:0} = \hat{x}^-_k$ and $\Sigma^+_{\tilde{x},k:0} = \Sigma^-_{\tilde{x},k}$.
- The Kalman gain is $\bar{L}_{k:i} = \dfrac{\Sigma^+_{\tilde{x},k:i-1}\bar{C}_{k:i}}{\bar{C}^T_{k:i}\Sigma^+_{\tilde{x},k:i-1}\bar{C}_{k:i}+1}$ and the updated state is:

$$\begin{aligned}\hat{x}^+_{k:i} &= \hat{x}^+_{k:i-1} + \bar{L}_{k:i}\left[\bar{z}_{k:i} - \bar{C}^T_{k:i}\hat{x}^+_{k:i-1}\right]\\ &= \hat{x}^+_{k:i-1} + \bar{L}_{k:i}\left[(\mathcal{S}_v^{-1}z_k)_i - \bar{C}^T_{k:i}\hat{x}^+_{k:i-1}\right],\end{aligned}$$

having covariance:

$$\Sigma^+_{\tilde{x},k:i} = \Sigma^+_{\tilde{x},k:i-1} - \bar{L}_{k:i}\bar{C}^T_{k:i}\Sigma^+_{\tilde{x},k:i-1}.$$

- The final measurement update gives $\hat{x}^+_k = \hat{x}^+_{k:m}$ and $\Sigma^+_{\tilde{x}:k} = \Sigma^+_{\tilde{x},k:m}$.

LDL updates for correlated measurements

- An alternative to the Cholesky decomposition for factoring the covariance matrix is the LDL decomposition:

$$\Sigma_{\tilde{v}} = \mathcal{L}_v\mathcal{D}_v\mathcal{L}_v^T,$$

where $\mathcal{L}_v$ is lower-triangular and $\mathcal{D}_v$ is diagonal (with positive entries).
- In MATLAB (not available in Octave), `[L,D] = ldl(SigmaV);`
- The Cholesky decomposition is related to the LDL decomposition via: $\mathcal{S}_v = \mathcal{L}_v\mathcal{D}_v^{1/2}$.
- We can use the LDL decomposition to perform a sequential measurement update.[1]
  - A computational advantage of LDL over Cholesky is that no square-root operations are required. (We can avoid finding $\mathcal{D}_v^{1/2}$.)
  - A pedagogical advantage of Cholesky is that we use it elsewhere.

[1] For example, see: Dan Simon, *Optimal State Estimation: Kalman, H_∞, and Nonlinear Approaches*, Wiley Interscience, Hoboken, New Jersey, 2006.

Summary

- When the system we are monitoring has multiple outputs, it is possible to speed up the Kalman filter by processing measurement sequentially.
- If the sensor noises are uncorrelated, the approach directly updates the state estimate and its covariance using one sensor at a time.
- If the sensor noises are correlated, the measurements must be jointly preprocessed by multiplying by a decorrelating Cholesky factor; then, the processed measurements can be used to update the state estimate and its covariance one at a time.