

SIMULTANEOUS STATE AND PARAMETER ESTIMATION USING KALMAN FILTERS

9.1: Parameters versus states

- Until now, we have assumed that the state-space model of the system whose state we are estimating is known and constant.
- However, the system model may not be entirely known: We may wish to adapt numeric values within the model to better match the model's behavior to the true system's behavior.
- Also, certain values within the system may change very slowly over the lifetime of the system—it would be good to track those changes.
- For example, consider a battery cell. Its state-of-charge can traverse its entire range within minutes. However, its internal resistance might change as little as 20% in a decade or more of regular use.
 - The quantities that tend to change quickly comprise the state of the system, and
 - The quantities that tend to change slowly comprise the time-varying parameters of the system.
- We know that Kalman filters may be used to estimate the state of a dynamic system given known parameters and noisy measurements.
- We may also use (nonlinear) Kalman filters to estimate parameters given a known state and noisy measurements.

- In this section of notes we first consider how to estimate the parameters of a system if its state is known.
- Next, we consider how to simultaneously estimate *both* the state and parameters of the system using two different approaches.

The generic approach to parameter estimation

- We denote the true parameters of a particular model by θ .
- We will use Kalman filtering techniques to estimate the parameters much like we have estimated the state. Therefore, we require a model of the dynamics of the parameters.
- By assumption, parameters change very slowly, so we model them as constant with some small perturbation:

$$\theta_k = \theta_{k-1} + r_{k-1}.$$

- The small white noise input r_k is fictitious, but models the slow drift in the parameters of the system plus the infidelity of the model structure.
- The output equation required for Kalman-filter system identification must be a measurable function of the system parameters. We use

$$d_k = h_k(x_k, u_k, \theta, e_k),$$

where $h(\cdot)$ is the output equation of the system model being identified, and e_k models the sensor noise and modeling error.

- Note that d_k is usually the same measurement as z_k , but we maintain a distinction here in case separate outputs are used. Then, $\mathbb{D}_k = \{d_0, d_1, \dots, d_k\}$. Also, note that e_k and v_k often play the same role, but are considered distinct here.

- We also slightly revise the mathematical model of system dynamics

$$x_k = f_{k-1}(x_{k-1}, u_{k-1}, \theta, w_{k-1})$$

$$z_k = h_k(x_k, u_k, \theta, v_k),$$

to explicitly include the parameters θ in the model.

- Non-time-varying numeric values required by the model may be embedded within $f(\cdot)$ and $h(\cdot)$, and are not included in θ .

9.2: EKF for parameter estimation

- Here, we show how to use EKF for parameter estimation.
- As always, we proceed by deriving the six essential steps of sequential inference.

EKF step 1a: Parameter estimate time update.

- The parameter prediction step is approximated as

$$\begin{aligned}\hat{\theta}_k^- &= \mathbb{E}[\theta_{k-1} + r_{k-1} \mid \mathbb{D}_{k-1}] \\ &= \hat{\theta}_{k-1}^+\end{aligned}$$

- This makes sense, since the parameters are assumed constant.

EKF step 1b: Error covariance time update.

- The covariance prediction step is accomplished by first computing $\tilde{\theta}_k^-$.

$$\begin{aligned}\tilde{\theta}_k^- &= \theta_k - \hat{\theta}_k^- = \theta_{k-1} + r_k - \hat{\theta}_{k-1}^+ \\ &= \tilde{\theta}_{k-1}^+ + r_k.\end{aligned}$$

- We then directly compute the desired covariance

$$\begin{aligned}\mathbb{E}[\tilde{\theta}_k^- (\tilde{\theta}_k^-)^T] &= \mathbb{E}[(\tilde{\theta}_{k-1}^+ + r_k)(\tilde{\theta}_{k-1}^+ + r_k)^T] \\ &= \Sigma_{\tilde{\theta}, k-1}^+ + \Sigma_{\tilde{r}}.\end{aligned}$$

- The time-updated covariance has additional uncertainty due to the fictitious noise “driving” the parameter values.

EKF step 1c: Output estimate.

- The system output is estimated to be

$$\begin{aligned}\hat{d}_k &= \mathbb{E}[h(x_k, u_k, \theta, e_k) \mid \mathbb{D}_{k-1}] \\ &\approx h_k(x_k, u_k, \hat{\theta}_k^-, \bar{e}_k).\end{aligned}$$

- That is, it is assumed that propagating $\hat{\theta}_k^-$ and the mean estimation error is the best approximation to estimating the output.

EKF step 2a: Estimator gain matrix.

- The output prediction error may then be approximated

$$\begin{aligned}\tilde{d}_k &= d_k - \hat{d}_k \\ &= h_k(x_k, u_k, \theta, e_k) - h_k(x_k, u_k, \hat{\theta}_k^-, \bar{e}_k),\end{aligned}$$

using again a Taylor-series expansion on the first term.

$$\begin{aligned}d_k &\approx h_k(x_k, u_k, \hat{\theta}_k^-, \bar{e}_k) \\ &\quad + \underbrace{\frac{dh_k(x_k, u_k, \theta, e_k)}{d\theta} \Big|_{\theta=\hat{\theta}_k^-}}_{\text{Defined as } \hat{C}_k^\theta} (\theta - \hat{\theta}_k^-) \\ &\quad + \underbrace{\frac{dh_k(x_k, u_k, \theta, e_k)}{de_k} \Big|_{e_k=\bar{e}_k}}_{\text{Defined as } \hat{D}_k^\theta} (e_k - \bar{e}_k).\end{aligned}$$

- From this, we can compute such necessary quantities as

$$\begin{aligned}\Sigma_{\tilde{d},k} &\approx \hat{C}_k^\theta \Sigma_{\tilde{\theta},k}^- (\hat{C}_k^\theta)^T + \hat{D}_k^\theta \Sigma_{\tilde{e}} (\hat{D}_k^\theta)^T, \\ \Sigma_{\tilde{\theta},k}^- &\approx \mathbb{E}[(\tilde{\theta}_k^-)(\hat{C}_k^\theta \tilde{\theta}_k^- + \hat{D}_k^\theta \tilde{e}_k)^T] \\ &= \Sigma_{\tilde{\theta},k}^- (\hat{C}_k^\theta)^T.\end{aligned}$$

- These terms may be combined to get the Kalman gain

$$L_k^\theta = \Sigma_{\tilde{\theta},k}^- (\hat{C}_k^\theta)^T [\hat{C}_k^\theta \Sigma_{\tilde{\theta},k}^- (\hat{C}_k^\theta)^T + \hat{D}_k^\theta \Sigma_{\tilde{e}} (\hat{D}_k^\theta)^T]^{-1}.$$

- Note, by the chain rule of total differentials,

$$dh_k(x_k, u_k, \theta, e_k) = \frac{\partial h_k(x_k, u_k, \theta, e_k)}{\partial x_k} dx_k + \frac{\partial h_k(x_k, u_k, \theta, e_k)}{\partial u_k} du_k + \frac{\partial h_k(x_k, u_k, \theta, e_k)}{\partial \theta} d\theta + \frac{\partial h_k(x_k, u_k, \theta, e_k)}{\partial e_k} de_k$$

$$\begin{aligned} \frac{dh_k(x_k, u_k, \theta, e_k)}{d\theta} &= \frac{\partial h_k(x_k, u_k, \theta, e_k)}{\partial x_k} \frac{dx_k}{d\theta} + \frac{\partial h_k(x_k, u_k, \theta, e_k)}{\partial u_k} \underbrace{\frac{du_k}{d\theta}}_0 + \frac{\partial h_k(x_k, u_k, \theta, e_k)}{\partial \theta} + \frac{\partial h_k(x_k, u_k, \theta, e_k)}{\partial e_k} \underbrace{\frac{de_k}{d\theta}}_0 \\ &= \frac{\partial h_k(x_k, u_k, \theta, e_k)}{\partial \theta} + \frac{\partial h_k(x_k, u_k, \theta, e_k)}{\partial x_k} \frac{dx_k}{d\theta}. \end{aligned}$$

■ But,

$$\frac{dx_k}{d\theta} = \frac{\partial f_{k-1}(x_{k-1}, u_{k-1}, \theta, w_{k-1})}{\partial \theta} + \frac{\partial f_{k-1}(x_{k-1}, u_{k-1}, \theta, w_{k-1})}{\partial x_{k-1}} \frac{dx_{k-1}}{d\theta}.$$

- The derivative calculations are recursive in nature, and evolve over time as the state evolves.
- The term $dx_0/d\theta$ is initialized to zero unless side information gives a better estimate of its value.
- To calculate \hat{C}_k^θ for any specific model structure, we require methods to calculate all of the above the partial derivatives for that model.

EKF step 2b: State estimate measurement update.

- The fifth step is to compute the *a posteriori* state estimate by updating the *a priori* estimate using the estimator gain and the output prediction error $d_k - \hat{d}_k$

$$\hat{\theta}_k^+ = \hat{\theta}_k^- + L_k^\theta (d_k - \hat{d}_k).$$

EKF step 2c: Error covariance measurement update.

- Finally, the updated covariance is computed as

$$\begin{aligned}
 \Sigma_{\tilde{\theta},k}^+ &= \Sigma_{\tilde{\theta},k}^- - L_k^\theta \Sigma_{\tilde{d},k} (L_k^\theta)^T \\
 &= \Sigma_{\tilde{\theta},k}^- - L_k^\theta \Sigma_{\tilde{d},k} (\Sigma_{\tilde{d},k})^{-T} (\Sigma_{\tilde{\theta},k}^-)^T \\
 &= \Sigma_{\tilde{\theta},k}^- - L_k^\theta \hat{C}_k^\theta \Sigma_{\tilde{\theta},k}^- \\
 &= (I - L_k^\theta \hat{C}_k^\theta) \Sigma_{\tilde{\theta},k}^-.
 \end{aligned}$$

- EKF for parameter estimation is summarized in a later table.

Notes:

- We initialize the parameter estimate with our best information re. the parameter value: $\hat{\theta}_0^+ = \mathbb{E}[\theta_0]$.
- We initialize the parameter estimation error covariance matrix:

$$\Sigma_{\tilde{\theta},0}^+ = \mathbb{E}[(\theta - \hat{\theta}_0^+)(\theta - \hat{\theta}_0^+)^T].$$

- We also initialize $dx_0/d\theta = 0$ unless side information is available.

Summary of the nonlinear extended Kalman filter for parameter estimation

Nonlinear state-space model:

$$\theta_{k+1} = \theta_k + r_k,$$

$$d_k = h_k(x_k, u_k, \theta_k, e_k).$$

where r_k and e_k are independent Gaussian noise processes with means zero and \bar{e} , respectively, and having covariance matrices $\Sigma_{\tilde{r}}$ and $\Sigma_{\tilde{e}}$, respectively.

Definition:

$$\hat{C}_k^\theta = \left. \frac{dh_k(x_k, u_k, \theta, e_k)}{d\theta} \right|_{\theta=\hat{\theta}_k^-} \quad \hat{D}_k^\theta = \left. \frac{dh_k(x_k, u_k, \theta, e_k)}{de_k} \right|_{e_k=\bar{e}_k}$$

Caution: Be careful to compute \hat{C}_k^θ using recursive chain rule described in notes!

Initialization: For $k = 0$, set

$$\hat{\theta}_0^+ = \mathbb{E}[\theta_0]$$

$$\Sigma_{\tilde{\theta},0}^+ = \mathbb{E}[(\theta_0 - \hat{\theta}_0^+)(\theta_0 - \hat{\theta}_0^+)^T].$$

$$\frac{dx_0}{d\theta} = 0, \text{ unless side information is available.}$$

Computation: For $k = 1, 2, \dots$ compute:

Param. estimate time update: $\hat{\theta}_k^- = \hat{\theta}_{k-1}^+.$

Error covariance time update: $\Sigma_{\tilde{\theta},k}^- = \Sigma_{\tilde{\theta},k-1}^+ + \Sigma_{\tilde{r}}.$

Output estimate $\hat{d}_k = h_k(x_k, u_k, \hat{\theta}_k^-, \bar{e}_k).$

Kalman gain matrix: $L_k^\theta = \Sigma_{\tilde{\theta},k}^- (\hat{C}_k^\theta)^T [\hat{C}_k^\theta \Sigma_{\tilde{\theta},k}^- (\hat{C}_k^\theta)^T + \hat{D}_k^\theta \Sigma_{\tilde{e}} (\hat{D}_k^\theta)^T]^{-1}.$

Param. estimate meas. update: $\hat{\theta}_k^+ = \hat{\theta}_k^- + L_k^\theta [d_k - \hat{d}_k].$

Error covariance meas. update: $\Sigma_{\tilde{\theta},k}^+ = (I - L_k^\theta \hat{C}_k^\theta) \Sigma_{\tilde{\theta},k}^-.$

9.3: SPKF for parameter estimation

- To use SPKF in a parameter estimation problem, we first define an augmented random vector θ^a that combines the randomness of the parameters and sensor noise. This augmented vector is used in the estimation process as described below.

SPKF step 1a: Parameter estimate time update.

- Due to the linearity of the parameter dynamics equation, we have $\hat{\theta}_k^{a,-} = \hat{\theta}_{k-1}^{a,+}$ (same as for EKF).

SPKF step 1b: Error covariance time update.

- Again, due to the linearity of the parameter dynamics equation, we have $\Sigma_{\tilde{\theta},k}^{a,-} = \Sigma_{\tilde{\theta},k-1}^{a,+} + \Sigma_{\tilde{r},k-1}^a$ (same as for EKF).

SPKF step 1c: Estimate system output d_k .

- To estimate the system output, we require a set of sigma points describing the output.
- This in turn requires a set of $p + 1$ sigma points describing $\theta_k^{a,-}$, which we will denote as $\mathcal{W}_k^{a,-}$.

$$\mathcal{W}_k^{a,-} = \left\{ \hat{\theta}_k^{a,-}, \hat{\theta}_k^{a,-} + \gamma \sqrt{\Sigma_{\tilde{\theta},k}^{a,-}}, \hat{\theta}_k^{a,-} - \gamma \sqrt{\Sigma_{\tilde{\theta},k}^{a,-}} \right\}.$$

- From the augmented sigma points, the $p + 1$ vectors comprising the parameters portion $\mathcal{W}_k^{\theta,-}$ and the $p + 1$ vectors comprising the modeling error portion $\mathcal{W}_k^{e,-}$ are extracted.
- The output equation is evaluated using all pairs of $\mathcal{W}_{k,i}^{\theta,-}$ and $\mathcal{W}_{k,i}^{e,-}$ (where the subscript i denotes that the i th vector is being extracted from the original set), yielding the sigma points $\mathcal{D}_{k,i}$ for time step k .

- That is, $\mathcal{D}_{k,i} = h(x_k, u_k, \mathcal{W}_{k,i}^{\theta,-}, \mathcal{W}_{k,i}^{e,-})$.
- Finally, the output estimate is computed as

$$\begin{aligned}\hat{d}_k^- &= \mathbb{E}[h_k(x_k, u_k, \theta, e_k) \mid \mathbb{D}_{k-1}] \\ &\approx \sum_{i=0}^p \alpha_i^{(m)} h_k(x_k, u_k, \mathcal{W}_{k,i}^{\theta,-}, \mathcal{W}_{k,i}^{e,-}) = \sum_{i=0}^p \alpha_i^{(m)} \mathcal{D}_{k,i}.\end{aligned}$$

SPKF step 2a: Estimator gain matrix L_k^θ .

- To compute the estimator gain matrix, we must first compute the required covariance matrices.

$$\begin{aligned}\Sigma_{\tilde{d},k} &= \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{D}_{k,i} - \hat{d}_k) (\mathcal{D}_{k,i} - \hat{d}_k)^T \\ \Sigma_{\tilde{\theta}\tilde{d},k}^- &= \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{W}_{k,i}^{\theta,-} - \hat{\theta}_k^{a,-}) (\mathcal{D}_{k,i} - \hat{d}_k)^T.\end{aligned}$$

- Then, we simply compute $L_k^\theta = \Sigma_{\tilde{\theta}\tilde{d},k}^- \Sigma_{\tilde{d},k}^{-1}$.

SPKF step 2b: State estimate measurement update.

- The fifth step is to compute the *a posteriori* state estimate by updating the *a priori* estimate using the estimator gain and the output prediction error $d_k - \hat{d}_k$

$$\hat{\theta}_k^{a,+} = \hat{\theta}_k^{a,-} + L_k^\theta (d_k - \hat{d}_k).$$

SPKF step 2c: Error covariance measurement update.

- The final step is calculated directly from the optimal formulation:

$$\Sigma_{\tilde{\theta},k}^{a,+} = \Sigma_{\tilde{\theta},k}^{a,-} - L_k^\theta \Sigma_{\tilde{d},k} (L_k^\theta)^T.$$

Summary of the nonlinear SPKF for parameter estimation

Nonlinear state-space model:

$$\theta_{k+1} = \theta_k + r_k,$$

$$d_k = h_k(x_k, u_k, \theta_k, e_k).$$

where r_k and e_k are independent Gaussian noise processes with means zero and \bar{e} , respectively, and having covariance matrices $\Sigma_{\tilde{r}}$ and $\Sigma_{\tilde{e}}$, respectively.

Definitions: Let

$$\theta_k^a = [\theta_k^T, e_k^T]^T, \quad \mathcal{W}_k^a = [(\mathcal{W}_k^\theta)^T, (\mathcal{W}_k^e)^T]^T, \quad p = 2 \times \dim(\theta_k^a).$$

Initialization: For $k = 0$, set

$$\hat{\theta}_0^+ = \mathbb{E}[\theta_0] \qquad \hat{\theta}_0^{a,+} = \mathbb{E}[\theta_0^a] = [(\hat{\theta}_0^+)^T, \bar{e}]^T.$$

$$\Sigma_{\tilde{\theta},0}^+ = \mathbb{E}[(\theta_0 - \hat{\theta}_0^+)(\theta_0 - \hat{\theta}_0^+)^T] \qquad \Sigma_{\tilde{\theta},0}^{a,+} = \mathbb{E}[(\theta_0^a - \hat{\theta}_0^{a,+})(\theta_0^a - \hat{\theta}_0^{a,+})^T]$$

$$= \text{diag}(\Sigma_{\tilde{\theta},0}^+, \Sigma_{\tilde{e}}).$$

Computation: For $k = 1, 2, \dots$ compute:

Param. estimate time update: $\hat{\theta}_k^- = \hat{\theta}_{k-1}^+.$

Error covariance time update: $\Sigma_{\tilde{\theta},k}^- = \Sigma_{\tilde{\theta},k-1}^+ + \Sigma_{\tilde{r}}.$

Output estimate: $\mathcal{W}_k^{a,-} = \{\hat{\theta}_k^{a,-}, \hat{\theta}_k^{a,-} + \gamma \sqrt{\Sigma_{\tilde{\theta},k}^{a,-}}, \hat{\theta}_k^{a,-} - \gamma \sqrt{\Sigma_{\tilde{\theta},k}^{a,-}}\}.$

$$\mathcal{D}_{k,i} = h_k(x_k, u_k, \mathcal{W}_{k,i}^{\theta,-}, \mathcal{W}_{k,i}^{e,-}).$$

$$\hat{d}_k = \sum_{i=0}^p \alpha_i^{(m)} \mathcal{D}_{k,i}.$$

Summary of the nonlinear SPKF for parameter estimation (cont.)

Computation (continued):

Estimator gain matrix:

$$\Sigma_{\tilde{d},k} = \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{D}_{k,i} - \hat{d}_k)(\mathcal{D}_{k,i} - \hat{d}_k)^T.$$

$$\Sigma_{\tilde{\theta}\tilde{d},k}^- = \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{W}_{k,i}^{\theta,-} - \hat{\theta}_k^-)(\mathcal{D}_{k,i} - \hat{d}_k)^T.$$

$$L_k^\theta = \Sigma_{\tilde{\theta}\tilde{d},k}^- \Sigma_{\tilde{d},k}^{-1}.$$

Param. estimate measurement update: $\hat{\theta}_k^+ = \hat{\theta}_k^- + L_k^\theta (d_k - \hat{d}_k).$

Error covariance measurement update: $\Sigma_{\tilde{\theta},k}^+ = \Sigma_{\tilde{\theta},k}^- - L_k^\theta \Sigma_{\tilde{d},k} (L_k^\theta)^T.$

Parameter estimation using SR-SPKF

- We can also apply SR-SPKF to estimate parameters.
- The biggest difficulty is updating the predicted state covariance in a computationally efficient way.
- We want an efficient square-root match to $\Sigma_{\tilde{\theta},k}^- = \Sigma_{\tilde{\theta},k-1}^+ + \Sigma_{\tilde{r}}$.
- We can approximate this in a square-root sense by $S_{\tilde{\theta},k}^- = S_{\tilde{\theta},k-1}^+ + D_{r,k-1}$, where we define (a diagonal matrix) $D_{r,k-1} = -\text{diag}\{S_{\tilde{\theta},k-1}^+\} + \sqrt{\text{diag}\{S_{\tilde{\theta},k-1}^+\}^2 + \text{diag}\{\Sigma_{\tilde{r}}\}}$.
- This is not an exact update, but it does force the diagonal of $\Sigma_{\tilde{\theta},k}^-$ to be correct.
- In the following table, the measurement noises are assumed to be additive. If they are not, then we must augment the parameters $\hat{\theta}_k^-$ with measurement noise in the output estimate section.

Summary of the square-root sigma-point Kalman filter for parameter estimation

Nonlinear state-space model:

$$\begin{aligned}\theta_k &= \theta_{k-1} + r_{k-1} \\ d_k &= h_k(x_k, u_k, \theta_k) + e_k,\end{aligned}$$

where r_k and e_k are independent, Gaussian noise processes with means 0 and \bar{e} and having covariance matrices $\Sigma_{\tilde{r}}$ and $\Sigma_{\tilde{e}}$, respectively.

Definition: Let

$$D_{r,k-1} = -\text{diag}\{S_{\tilde{\theta},k-1}^+\} + \sqrt{\text{diag}\{S_{\tilde{\theta},k-1}^+\}^2 + \text{diag}\{\Sigma_{\tilde{r}}\}} \quad p = 2 \times \dim(\theta_k).$$

Initialization: For $k = 0$, set

$$\hat{\theta}_0^+ = \mathbb{E}[\theta_0] \quad S_{\tilde{\theta},0}^+ = \text{chol} \left\{ \mathbb{E}[(\theta_0 - \hat{\theta}_0^+)(\theta_0 - \hat{\theta}_0^+)^T] \right\}$$

Computation: For $k = 1, 2, \dots$ compute:

Param. estimate time update: $\hat{\theta}_k^- = \hat{\theta}_{k-1}^+.$

Error covariance time update: $S_{\tilde{\theta},k}^- = S_{\tilde{\theta},k-1}^+ + D_{r,k-1}.$

Output estimate: $\mathcal{W}_k = \{\hat{\theta}_k^-, \hat{\theta}_k^- + \gamma S_{\tilde{\theta},k}^-, \hat{\theta}_k^- - \gamma S_{\tilde{\theta},k}^-\}$

$$\mathcal{D}_{k,i} = h_k(x_k, u_k, \mathcal{W}_{k,i}) + \bar{e}.$$

$$\hat{d}_k = \sum_{i=0}^p \alpha_i^{(m)} \mathcal{D}_{k,i}.$$

Estimator gain matrix: $S_{\tilde{d},k} = \text{qr} \left\{ \left[\begin{array}{c} \sqrt{\alpha_i^{(c)}} (\mathcal{D}_{k,(0:p)} - \hat{d}_k) \quad \sqrt{\Sigma_e} \end{array} \right]^T \right\}^T.$

$$\Sigma_{\tilde{\theta}\tilde{d},k}^- = \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{W}_{k,i} - \hat{\theta}_k^-) (\mathcal{D}_{k,i} - \hat{d}_k)^T.$$

$$L_k^\theta = \Sigma_{\tilde{\theta}\tilde{d},k}^- \left(S_{\tilde{d},k}^T S_{\tilde{d},k} \right)^{-1} \quad (\text{solved by backsubst.}).$$

Param. estimate meas. update: $\hat{\theta}_k^+ = \hat{\theta}_k^- + L_k^\theta (d_k - \hat{d}_k).$

Error covar. meas. update: $S_{\tilde{\theta},k}^+ = \text{downdate} \left\{ S_{\tilde{\theta},k}^-, L_k^\theta S_{\tilde{d},k} \right\}.$

9.4: Simultaneous state and parameter estimation

- We have now seen how to use Kalman filters to perform state estimation and parameter estimation independently.
- How about both at the same time?
- There are two approaches to doing so: Joint estimation and dual estimation. These are discussed in the next sections.

Generic joint estimation

- In joint estimation, the state vector and parameter vector are combined, and a Kalman filter simultaneously estimates the values of this augmented state vector.
- It has the disadvantages of large matrix operations due to the high dimensionality of the resulting augmented model and potentially poor numeric conditioning due to the vastly different time scales of the states (including parameters) in the augmented state vector.
- However, it is quite straightforward to implement. We first combine the state and parameter vectors to form an augmented state such that the dynamics may be represented by

$$\begin{bmatrix} x_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} f_{k-1}(x_{k-1}, u_{k-1}, \theta_{k-1}, w_{k-1}) \\ \theta_{k-1} + r_{k-1} \end{bmatrix}$$

$$z_k = h_k(x_k, u_k, \theta_k, v_k).$$

- Note that to simplify notation, we will refer to the vector comprising both the present state and the present parameters as \mathbf{X}_k , the vector comprising the present process noise and present parameter noise

as \mathbf{W}_k , and the equation combining the dynamics of the state and the dynamics of the parameters as \mathbf{F} . This allows us to write

$$\mathbf{X}_k = \mathbf{F}_{k-1}(\mathbf{X}_{k-1}, u_{k-1}, \mathbf{W}_{k-1})$$

$$z_k = h_k(\mathbf{X}_k, u_k, v_k).$$

- With the augmented model of the system state dynamics and parameter dynamics defined, we apply a nonlinear KF method.

Generic dual estimation

- In dual estimation, separate Kalman filters are used for state estimation and parameter estimation.
- The computational complexity is smaller and the matrix operations may be numerically better conditioned.
- However, by decoupling state from parameters, any cross-correlations between changes are lost, leading to potentially poorer accuracy.
- The mathematical model of state dynamics again explicitly includes the parameters as the vector θ_k :

$$x_k = f_{k-1}(x_{k-1}, u_{k-1}, w_{k-1}, \theta_{k-1})$$

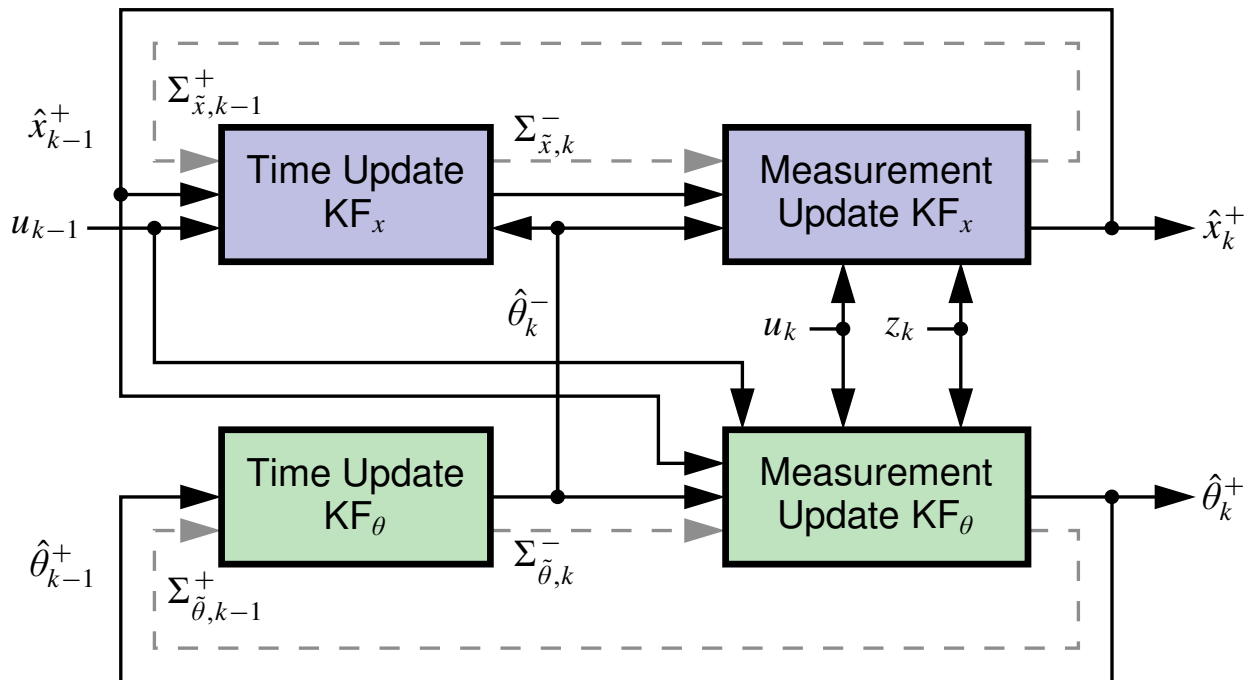
$$z_k = h_k(x_k, u_k, v_k, \theta_{k-1}).$$

- Non-time-varying numeric values required by the model may be embedded within $f(\cdot)$ and $h(\cdot)$, and are not included in θ_k . We also slightly revise the mathematical model of parameter dynamics to explicitly include the effect of the state equation.

$$\theta_k = \theta_{k-1} + r_{k-1}$$

$$d_k = h_k\left(f_{k-1}(x_{k-1}, u_{k-1}, \bar{w}_{k-1}, \theta_{k-1}), u_k, e_k, \theta_{k-1}\right).$$

- The dual filters can be viewed by drawing a block diagram. (The interactions will be made clearer later)



- We see that the process essentially comprises two Kalman filters running in parallel—one adapting the state and one adapting parameters—with some information exchange between the filters.

9.5: EKF and SPKF joint and dual estimation

Joint state and parameter estimation via EKF

- Applying EKF to the joint estimation problem is straightforward. But, don't forget the recursive calculation of $d\mathbf{F}/d\mathbf{X}$.

Dual state and parameter estimation via EKF

- Two EKFs are implemented, with their signals mixed.
- Again, we need to be careful when computing \hat{C}_k^θ , which requires a total-differential expansion to be correct

$$\hat{C}_k^\theta = \left. \frac{dh_k(\hat{x}_k^-, u_k, \theta)}{d\theta} \right|_{\theta=\hat{\theta}_k^-}$$

$$\frac{dh_k(\hat{x}_k^-, u_k, \theta)}{d\theta} = \frac{\partial h_k(\hat{x}_k^-, u_k, \theta)}{\partial \theta} + \frac{\partial h_k(\hat{x}_k^-, u_k, \theta)}{\partial \hat{x}_k^-} \frac{d\hat{x}_k^-}{d\theta}$$

$$\frac{d\hat{x}_k^-}{d\theta} = \frac{\partial f_{k-1}(\hat{x}_{k-1}^+, u_{k-1}, \theta)}{\partial \theta} + \frac{\partial f_{k-1}(\hat{x}_{k-1}^+, u_{k-1}, \theta)}{\partial \hat{x}_{k-1}^+} \frac{d\hat{x}_{k-1}^+}{d\theta}$$

$$\frac{d\hat{x}_{k-1}^+}{d\theta} = \frac{d\hat{x}_{k-1}^-}{d\theta} - L_{k-1}^x \frac{dh_{k-1}(\hat{x}_{k-1}^-, u_{k-1}, \theta)}{d\theta},$$

- This assumes that L_{k-1}^x is not a function of θ (Its is—weakly—so that it is not usually worth the extra computation to include the effects of L_{k-1}^x as a function of θ).
- The three total derivatives are computed recursively, initialized with zero values.

Summary of the nonlinear extended Kalman filter for joint estimation

State-space model:
$$\begin{bmatrix} x_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} f_{k-1}(x_{k-1}, u_{k-1}, w_{k-1}, \theta_{k-1}) \\ \theta_{k-1} + r_{k-1} \end{bmatrix} \quad \text{or} \quad \begin{aligned} \mathbf{X}_k &= \mathbf{F}_{k-1}(\mathbf{X}_{k-1}, u_{k-1}, \mathbf{V}_{k-1}) \\ z_k &= h_k(\mathbf{X}_k, u_k, v_k), \end{aligned}$$

$$z_k = h_k(x_k, u_k, v_k, \theta_k),$$

where w_k , r_k , and v_k are independent, Gaussian noise processes of covariance matrices $\Sigma_{\tilde{w}}$, $\Sigma_{\tilde{r}}$, and $\Sigma_{\tilde{v}}$, respectively. For brevity, we let $\mathbf{X}_k = [x_k^T, \theta_k^T]^T$, $\mathbf{W}_k = [w_k^T, r_k^T]^T$ and $\Sigma_{\tilde{\mathbf{W}}} = \text{diag}(\Sigma_{\tilde{w}}, \Sigma_{\tilde{r}})$.

Definitions:

$$\begin{aligned} \hat{A}_k &= \left. \frac{d\mathbf{F}_k(\mathbf{X}_k, u_k, \mathbf{W}_k)}{d\mathbf{X}_k} \right|_{\mathbf{X}_k = \hat{\mathbf{X}}_k^+} & \hat{B}_k &= \left. \frac{d\mathbf{F}_k(\mathbf{X}_k, u_k, \mathbf{W}_k)}{d\mathbf{W}_k} \right|_{\mathbf{W}_k = \bar{\mathbf{W}}_k} \\ \hat{C}_k &= \left. \frac{dh_k(\mathbf{X}_k, u_k, v_k)}{d\mathbf{X}_k} \right|_{\mathbf{X}_k = \hat{\mathbf{X}}_k^-} & \hat{D}_k &= \left. \frac{dh_k(\mathbf{X}_k, u_k, v_k)}{dv_k} \right|_{v_k = \bar{v}_k} \end{aligned}$$

Initialization: For $k = 0$, set

$$\begin{aligned} \hat{\mathbf{X}}_0^+ &= \mathbb{E}[\mathbf{X}_0] \\ \Sigma_{\tilde{\mathbf{X}},0}^+ &= \mathbb{E}[(\mathbf{X}_0 - \hat{\mathbf{X}}_0^+)(\mathbf{X}_0 - \hat{\mathbf{X}}_0^+)^T] \end{aligned}$$

Computation: For $k = 1, 2, \dots$ compute:

State estimate time update:

$$\hat{\mathbf{X}}_k^- = \mathbf{F}_{k-1}(\hat{\mathbf{X}}_{k-1}^+, u_{k-1}, \bar{\mathbf{W}}_{k-1}).$$

Error covariance time update:

$$\Sigma_{\tilde{\mathbf{X}},k}^- = \hat{A}_{k-1} \Sigma_{\tilde{\mathbf{X}},k-1}^+ \hat{A}_{k-1}^T + \hat{B}_{k-1} \Sigma_{\tilde{\mathbf{W}}} \hat{B}_{k-1}^T.$$

Output estimate:

$$\hat{z}_k = h_k(\hat{\mathbf{X}}_k^-, u_k, \bar{v}_k).$$

Estimator gain matrix:

$$L_k = \Sigma_{\tilde{\mathbf{X}},k}^- \hat{C}_k^T [\hat{C}_k \Sigma_{\tilde{\mathbf{X}},k}^- \hat{C}_k^T + \hat{D}_k \Sigma_{\tilde{v}} \hat{D}_k^T]^{-1}.$$

State estimate measurement update:

$$\hat{\mathbf{X}}_k^+ = \hat{\mathbf{X}}_k^- + L_k(z_k - \hat{z}_k).$$

Error covariance measurement update: $\Sigma_{\tilde{\mathbf{X}},k}^+ = (I - L_k \hat{C}_k) \Sigma_{\tilde{\mathbf{X}},k}^-.$

Summary of the dual extended Kalman filter for state and parameter estimation

Nonlinear state-space models:

$$\begin{aligned} x_{k+1} &= f_k(x_k, u_k, \theta_k, w_k) & \theta_{k+1} &= \theta_k + r_k, \\ z_k &= h_k(x_k, u_k, \theta_k, v_k) & \text{and} & & d_k &= h_k(x_k, u_k, \theta_k, e_k). \end{aligned}$$

where w_k , v_k , r_k and e_k are independent Gaussian noise processes with means \bar{w} , \bar{v} , \bar{r} , and \bar{e} and covariace matrices $\Sigma_{\bar{w}}$, $\Sigma_{\bar{v}}$, $\Sigma_{\bar{r}}$ and $\Sigma_{\bar{e}}$, respectively.

Definitions:

$$\begin{aligned} \hat{A}_k &= \left. \frac{df_k(x_k, u_k, \hat{\theta}_k^-, w_k)}{dx_k} \right|_{x_k=\hat{x}_k^+} & \hat{B}_k &= \left. \frac{df_k(x_k, u_k, \hat{\theta}_k^-, w_k)}{dw_k} \right|_{w_k=\bar{w}} \\ \hat{C}_k^x &= \left. \frac{dh_k(x_k, u_k, \hat{\theta}_k^-, v_k)}{dx_k} \right|_{x_k=\hat{x}_k^-} & \hat{D}_k^x &= \left. \frac{dh_k(x_k, u_k, \hat{\theta}_k^-, v_k)}{dv_k} \right|_{v_k=\bar{v}} \\ \hat{C}_k^\theta &= \left. \frac{dh_k(\hat{x}_k^-, u_k, \theta, e_k)}{d\theta} \right|_{\theta=\hat{\theta}_k^-} & \hat{D}_k^\theta &= \left. \frac{dh_k(\hat{x}_k^-, u_k, \theta, e_k)}{de_k} \right|_{e_k=\bar{e}} \end{aligned}$$

Initialization: For $k = 0$, set

$$\begin{aligned} \hat{\theta}_0^+ &= \mathbb{E}[\theta_0], & \Sigma_{\hat{\theta},0}^+ &= \mathbb{E}[(\theta_0 - \hat{\theta}_0^+)(\theta_0 - \hat{\theta}_0^+)^T], \\ \hat{x}_0^+ &= \mathbb{E}[x_0], & \Sigma_{\hat{x},0}^+ &= \mathbb{E}[(x_0 - \hat{x}_0^+)(x_0 - \hat{x}_0^+)^T]. \end{aligned}$$

Computation: For $k = 1, 2, \dots$ compute:

Time update for the weight filter:

$$\begin{aligned} \hat{\theta}_k^- &= \hat{\theta}_{k-1}^+ \\ \Sigma_{\hat{\theta},k}^- &= \Sigma_{\hat{\theta},k-1}^+ + \Sigma_{\bar{r}} \end{aligned}$$

Time update for the state filter:

$$\begin{aligned} \hat{x}_k^- &= f_{k-1}(\hat{x}_{k-1}^+, u_{k-1}, \hat{\theta}_k^-, \bar{w}) \\ \Sigma_{\hat{x},k}^- &= \hat{A}_{k-1} \Sigma_{\hat{x},k-1}^+ \hat{A}_{k-1}^T + \hat{B}_{k-1} \Sigma_{\bar{w}} \hat{B}_{k-1}^T \end{aligned}$$

Meas. update for the state filter:

$$\begin{aligned} L_k^x &= \Sigma_{\hat{x},k}^- (\hat{C}_k^x)^T [\hat{C}_k^x \Sigma_{\hat{x},k}^- (\hat{C}_k^x)^T + \hat{D}_k^x \Sigma_{\bar{v}} (\hat{D}_k^x)^T]^{-1} \\ \hat{x}_k^+ &= \hat{x}_k^- + L_k^x [z_k - h_k(\hat{x}_k^-, u_k, \hat{\theta}_k^-, \bar{v})] \\ \Sigma_{\hat{x},k}^+ &= (I - L_k^x \hat{C}_k^x) \Sigma_{\hat{x},k}^- \end{aligned}$$

Summary of the dual EKF for state and parameter estimation (cont.)

Computation (cont.): For $k = 1, 2, \dots$ compute:

$$\begin{aligned} \text{Meas. update for the weight filter: } \quad L_k^\theta &= \Sigma_{\tilde{\theta},k}^- (\hat{C}_k^\theta)^T [\hat{C}_k^\theta \Sigma_{\tilde{\theta},k}^- (\hat{C}_k^\theta)^T + \hat{D}_k^\theta \Sigma_{\tilde{e}} (\hat{D}_k^\theta)^T]^{-1} \\ \hat{\theta}_k^+ &= \hat{\theta}_k^- + L_k^\theta [z_k - h_k(\hat{x}_k^-, u_k, \hat{\theta}_k^-, \bar{e})] \\ \Sigma_{\tilde{\theta},k}^+ &= (I - L_k^\theta \hat{C}_k^\theta) \Sigma_{\tilde{\theta},k}^- \end{aligned}$$

Summary of the nonlinear sigma-point Kalman filter for joint estimation

$$\text{State-space model: } \begin{bmatrix} x_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} f_{k-1}(x_{k-1}, u_{k-1}, w_{k-1}, \theta_{k-1}) \\ \theta_{k-1} + r_{k-1} \end{bmatrix} \quad \text{or} \quad \begin{aligned} \mathbf{X}_k &= \mathbf{F}_{k-1}(\mathbf{X}_{k-1}, u_{k-1}, \mathbf{V}_{k-1}) \\ z_k &= h_k(\mathbf{X}_k, u_k, v_k), \end{aligned}$$

$$z_k = h_k(x_k, u_k, v_k, \theta_k),$$

where w_k , r_k , and v_k are independent, Gaussian noise processes with means \bar{w} , \bar{r} , and \bar{v} , and covariance matrices $\Sigma_{\tilde{w}}$, $\Sigma_{\tilde{r}}$, and $\Sigma_{\tilde{v}}$, respectively. For brevity, we let $\mathbf{X}_k = [x_k^T, \theta_k^T]^T$, $\mathbf{W}_k = [w_k^T, r_k^T]^T$ and $\Sigma_{\tilde{\mathbf{W}}} = \text{diag}(\Sigma_{\tilde{w}}, \Sigma_{\tilde{r}})$.

Definitions: Let

$$\mathbf{X}_k^a = [\mathbf{X}_k^T, \mathbf{W}_k^T, v_k^T]^T, \quad \mathcal{X}_k^a = [(\mathcal{X}_k^{\mathbf{X}})^T, (\mathcal{X}_k^{\mathbf{W}})^T, (\mathcal{X}_k^v)^T]^T, \quad p = 2 \times \dim(\mathbf{X}_k^a).$$

Initialization: For $k = 0$, set

$$\begin{aligned} \hat{\mathbf{X}}_0^+ &= \mathbb{E}[\mathbf{X}_0] & \hat{\mathbf{X}}_0^{a,+} &= \mathbb{E}[\mathbf{X}_0^a] = [(\hat{\mathbf{X}}_0^+)^T, \bar{\mathbf{W}}, \bar{v}]^T. \\ \Sigma_{\tilde{\mathbf{X}},0}^+ &= \mathbb{E}[(\mathbf{X}_0 - \hat{\mathbf{X}}_0^+)(\mathbf{X}_0 - \hat{\mathbf{X}}_0^+)^T] & \Sigma_{\tilde{\mathbf{X}},0}^{a,+} &= \mathbb{E}[(\mathbf{X}_0^a - \hat{\mathbf{X}}_0^{a,+})(\mathbf{X}_0^a - \hat{\mathbf{X}}_0^{a,+})^T] \\ & & &= \text{diag}(\Sigma_{\tilde{\mathbf{X}},0}^+, \Sigma_{\tilde{\mathbf{W}}}, \Sigma_{\tilde{v}}). \end{aligned}$$

Summary of the nonlinear SPKF for joint estimation (cont.)

Computation: For $k = 1, 2, \dots$ compute:

$$\text{State estimate time update: } \mathcal{X}_{k-1}^{a,+} = \{ \hat{\mathbf{X}}_{k-1}^{a,+}, \hat{\mathbf{X}}_{k-1}^{a,+} + \gamma \sqrt{\Sigma_{\tilde{\mathbf{X}},k-1}^{a,+}}, \hat{\mathbf{X}}_{k-1}^{a,+} - \gamma \sqrt{\Sigma_{\tilde{\mathbf{X}},k-1}^{a,+}} \}.$$

$$\mathcal{X}_{k,i}^{\mathbf{X},-} = \mathbf{F}_{k-1}(\mathcal{X}_{k-1,i}^{\mathbf{X},+}, u_{k-1}, \mathcal{X}_{k-1,i}^{\mathbf{W},+}).$$

$$\hat{\mathbf{X}}_k^- = \sum_{i=0}^p \alpha_i^{(m)} \mathcal{X}_{k,i}^{\mathbf{X},-}.$$

$$\text{Error covariance time update: } \Sigma_{\tilde{\mathbf{X}},k}^- = \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{X}_{k,i}^{\mathbf{X},-} - \hat{\mathbf{X}}_k^-) (\mathcal{X}_{k,i}^{\mathbf{X},-} - \hat{\mathbf{X}}_k^-)^T.$$

$$\text{Output estimate: } \mathcal{Z}_{k,i} = h_k(\mathcal{X}_{k,i}^{\mathbf{X},-}, u_k, \mathcal{X}_{k-1,i}^{v,+}).$$

$$\hat{z}_k = \sum_{i=0}^p \alpha_i^{(m)} \mathcal{Z}_{k,i}.$$

$$\text{Estimator gain matrix: } \Sigma_{\tilde{z},k} = \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{Z}_{k,i} - \hat{z}_k) (\mathcal{Z}_{k,i} - \hat{z}_k)^T.$$

$$\Sigma_{\tilde{\mathbf{X}}\tilde{z},k}^- = \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{X}_{k,i}^{\mathbf{X},-} - \hat{\mathbf{X}}_k^-) (\mathcal{Z}_{k,i} - \hat{z}_k)^T.$$

$$L_k = \Sigma_{\tilde{\mathbf{X}}\tilde{z},k}^- \Sigma_{\tilde{z},k}^{-1}.$$

$$\text{State estimate meas. update: } \hat{\mathbf{X}}_k^+ = \hat{\mathbf{X}}_k^- + L_k (z_k - \hat{z}_k).$$

$$\text{Error covariance meas. update: } \Sigma_{\tilde{\mathbf{X}},k}^+ = \Sigma_{\tilde{\mathbf{X}},k}^- - L_k \Sigma_{\tilde{z},k} L_k^T.$$

Joint state and parameter estimation via SPKF

- This is a standard SPKF with the state vector augmented with parameters.

Dual state and parameter estimation via SPKF

- This, just like dual estimation using EKF, uses two filters. Both employ the SPKF algorithm and intermix signals.

Summary of the sigma-point Kalman filter for dual state and parameter estimation

$$x_k = f_{k-1}(x_{k-1}, u_{k-1}, w_{k-1}, \theta_{k-1}) \quad \theta_k = \theta_{k-1} + r_{k-1},$$

Nonlinear state-space models: $z_k = h_k(x_k, u_k, v_k, \theta_k)$ and $d_k = h_k(f_{k-1}(x_{k-1}, u_k, v_k, \bar{v}_k, \theta_{k-1}))$

where w_k , v_k , r_k and e_k are independent, Gaussian noise processes with means \bar{w} , \bar{v} , \bar{r} , and \bar{e} , and covariance matrices $\Sigma_{\bar{w}}$, $\Sigma_{\bar{v}}$, $\Sigma_{\bar{r}}$ and $\Sigma_{\bar{e}}$, respectively.

Definitions: $x_k^a = [x_k^T, w_k^T, v_k^T]^T$, $\mathcal{X}_k^a = [(\mathcal{X}_k^x)^T, (\mathcal{X}_k^w)^T, (\mathcal{X}_k^v)^T]^T$, $p = 2 \times \dim(x_k^a)$.

Initialization: For $k = 0$, set

$$\begin{aligned} \hat{\theta}_0^+ &= \mathbb{E}[\theta_0], & \Sigma_{\bar{\theta},0}^+ &= \mathbb{E}[(\theta_0 - \hat{\theta}_0^+)(\theta_0 - \hat{\theta}_0^+)^T]. \\ \hat{x}_0^+ &= \mathbb{E}[x_0], & \hat{x}_0^{a,+} &= \mathbb{E}[x_0^a] = [(\hat{x}_0^+)^T, \bar{w}, \bar{v}]^T. \\ \Sigma_{\bar{x},0}^+ &= \mathbb{E}[(x_0 - \hat{x}_0^+)(x_0 - \hat{x}_0^+)^T], & \Sigma_{\bar{x},0}^{a,+} &= \mathbb{E}[(x_0^a - \hat{x}_0^{a,+})(x_0^a - \hat{x}_0^{a,+})^T] \\ & & &= \text{diag}(\Sigma_{\bar{x},0}^+, \Sigma_w, \Sigma_v). \end{aligned}$$

Computation: For $k = 1, 2, \dots$ compute:

Parameter estimate time update: $\hat{\theta}_k^- = \hat{\theta}_{k-1}^+$.

Parameter covariance time update: $\Sigma_{\bar{\theta},k}^- = \Sigma_{\bar{\theta},k-1}^+ + \Sigma_{\bar{r}}$.

State estimate time update: $\mathcal{X}_{k-1}^{a,+} = \{\hat{x}_{k-1}^{a,+}, \hat{x}_{k-1}^{a,+} + \gamma \sqrt{\Sigma_{\bar{x},k-1}^{a,+}}, \hat{x}_{k-1}^{a,+} - \gamma \sqrt{\Sigma_{\bar{x},k-1}^{a,+}}\}$.

$$\mathcal{X}_{k,i}^{x,-} = f_{k-1}(\mathcal{X}_{k-1,i}^{x,+}, u_{k-1}, \mathcal{X}_{k-1,i}^{w,+}, \hat{\theta}_k^-).$$

$$\hat{x}_k^- = \sum_{i=0}^p \alpha_i^{(m)} \mathcal{X}_{k,i}^{x,-}.$$

State covariance time update: $\Sigma_{\bar{x},k}^- = \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{X}_{k,i}^{x,-} - \hat{x}_k^-)(\mathcal{X}_{k,i}^{x,-} - \hat{x}_k^-)^T.$

(continued...)

Summary of the dual sigma-point Kalman filter (continued)

Computation (continued): For $k = 1, 2, \dots$ compute:

$$\begin{aligned}
 \text{Output estimate, param. filter:} \quad \mathcal{W}_k &= \{\hat{\theta}_k^-, \hat{\theta}_k^- + \gamma \sqrt{\Sigma_{\tilde{\theta},k}^-}, \hat{\theta}_k^- - \gamma \sqrt{\Sigma_{\tilde{\theta},k}^-}\}. \\
 \mathcal{D}_{k,i} &= h_k(f_{k-1}(\hat{x}_{k-1}^+, u_{k-1}, \bar{w}_{k-1}, \mathcal{W}_{k,i}), u_k, \bar{v}_k, \mathcal{W}_{k,i}). \\
 \hat{d}_k &= \sum_{i=0}^p \alpha_i^{(m)} \mathcal{D}_{k,i} \\
 \text{Output estimate, state filter:} \quad \mathcal{Z}_{k,i} &= h_k(\mathcal{X}_{k,i}^{x,-}, u_k, \mathcal{X}_{k-1,i}^{v,+}, \hat{\theta}_k^-). \\
 \hat{z}_k &= \sum_{i=0}^p \alpha_i^{(m)} \mathcal{Z}_{k,i}. \\
 \text{State filter gain matrix:} \quad \Sigma_{\tilde{z},k} &= \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{Z}_{k,i} - \hat{z}_k)(\mathcal{Z}_{k,i} - \hat{z}_k)^T. \\
 \Sigma_{\tilde{x}\tilde{z},k}^- &= \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{X}_{k,i}^{x,-} - \hat{x}_k^-)(\mathcal{Z}_{k,i} - \hat{z}_k)^T. \\
 L_k^x &= \Sigma_{\tilde{x}\tilde{z},k}^- \Sigma_{\tilde{z},k}^{-1}. \\
 \text{Parameter filter gain matrix:} \quad \Sigma_{\tilde{d},k} &= \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{D}_{k,i} - \hat{d}_k)(\mathcal{D}_{k,i} - \hat{d}_k)^T. \\
 \Sigma_{\tilde{\theta}\tilde{d},k}^- &= \sum_{i=0}^p \alpha_i^{(c)} (\mathcal{W}_{k,i} - \hat{\theta}_k^-)(\mathcal{D}_{k,i} - \hat{d}_k)^T. \\
 L_k^\theta &= \Sigma_{\tilde{\theta}\tilde{d},k}^- \Sigma_{\tilde{d},k}^{-1}. \\
 \text{State estimate meas. update:} \quad \hat{x}_k^+ &= \hat{x}_k^- + L_k^x (z_k - \hat{z}_k). \\
 \text{State covariance meas. update:} \quad \Sigma_{\tilde{x},k}^+ &= \Sigma_{\tilde{x},k}^- - L_k^x \Sigma_{\tilde{z},k} (L_k^x)^T. \\
 \text{Param. estimate meas. update:} \quad \hat{\theta}_k^+ &= \hat{\theta}_k^- + L_k^\theta (z_k - \hat{d}_k). \\
 \text{Parameter covar. meas. update:} \quad \Sigma_{\tilde{\theta},k}^+ &= \Sigma_{\tilde{\theta},k}^- - L_k^\theta \Sigma_{\tilde{d},k} (L_k^\theta)^T.
 \end{aligned}$$

Ensuring correct convergence

- Dual and joint filtering will adapt \hat{x} and $\hat{\theta}$ so that the model input-output relationship matches the system's input-output data as closely as possible.
- There is no built-in guarantee that the state of the model converges to anything with physical meaning.
- Usually, when employing a Kalman filter, we are concerned that the state converge to a very specific meaning.
- Special steps must be taken to ensure that this occurs.