State Estimation ...

Lecture 17

... of Non-Linear Dynamical Systems:

- Observability
- Extended Kalman Filter (EKF)
- Unscented Kalman Filter (UKF)
- UKF with state constraints

Non-Linear Discrete-Time Dynamical Systems
Lecture 17

We will consider the following generic non-linear, (invariant) discrete-time dynamical system with additive noise,

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{G}\mathbf{w}_k \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \end{aligned}$$

and initial state $\mathbf{x}_0$. We keep the noise description the same as in the linear context, $E[\mathbf{w}_k] = 0_n$, $E[\mathbf{v}_k] = 0_m$, $E[\mathbf{w}_k \mathbf{w}_k^T] = \mathbf{Q} \in \mathbb{R}^{n \times n}$, and $E[\mathbf{v}_k \mathbf{v}_k^T] = \mathbf{R} \in \mathbb{R}^{m \times m}$.

Observability
Lecture 17

- Generally input dependent system can be observable under a set of inputs and not observable under other inputs)
- A much less straightforward concept for non-linear systems, but the question remains very valid. The estimation problem might be ill-posed for particular combinations of dynamics and measurements.

Extended Kalman Filter

Lecture 17

- Extended Kalman Filter Predict Step:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{f}(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1})$$

$$\mathbf{P}_{k|k-1} = \mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T$$

Where

$$\mathbf{A}_k = \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \Big|_{\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}}$$

$$\mathbf{H}_k = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\hat{\mathbf{x}}_{k|k-1}}$$

- Kalman Filter Update Step:

$$\hat{\mathbf{y}}_k = \mathbf{h}(\hat{\mathbf{x}}_{k|k-1}); \; \mathbf{P}_y = \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T; \; \mathbf{P}_{xy} = \mathbf{P}_{k|k-1} \mathbf{H}^T$$

$$\mathbf{K}_k = \mathbf{P}_{xy}(\mathbf{P}_y + \mathbf{R})^{-1}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_{k|k-1}$$

EKF Example 1

Lecture 17

- A 1D rocket model, with state vector $\mathbf{x} = [s \; v \; m]^T \in \mathbb{R}^3$, where s is traveled distance, v velocity, m mass and input $u \in \mathbb{R}$

Continuous-time model:

$$\dot{\mathbf{x}} = \begin{bmatrix} v \\ (u - 0.2v^2)m \\ -0.01^2 u \end{bmatrix}$$

Discrete-time approximation:

$$\mathbf{f}(\mathbf{x}, u) = \begin{bmatrix} s + v\Delta t \\ v + (u - 0.2v^2)m\Delta t \\ m - 0.01^2 u\Delta t \end{bmatrix}$$

The measurement equation is $h(\mathbf{x}) = s$.

EKF Example 2

Lecture 17

- An autonomous pendulum model $\mathbf{x} = [\theta \quad \omega \quad L \quad \alpha]^T \in \mathbb{R}^4$, where θ is the angle, ω is the angular velocity, L is the length of the pendulum wire, and α is a coefficient of friction.

Continuous-time model:

$$\dot{\mathbf{x}} = \begin{bmatrix} \omega \\ -\frac{g}{L}\sin\theta - \alpha\omega \\ 0 \\ 0 \end{bmatrix}$$

Discrete-time approximation:

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \theta + \omega\Delta t \\ \omega - (\frac{g}{L}\sin\theta + \alpha\omega)\Delta t \\ L \\ \alpha \end{bmatrix}$$

The measurement equation is $h(\mathbf{x}) = \theta$.

EKF vs UKF

Lecture 17

- For the EKF, at each time step, the Jacobian of the nonlinear function is evaluated with current predicted states. These matrices can be then be used in the “standard” Kalman filter equations. The process essentially linearizes the non-linear function around the current estimate.
- Notice also, that the nonlinear functions **f** and **h** could be used in their original (non-linearized) form in the dynamics prediction step and for the measurement prediction. We only needed a linearized approximation for covariance estimation/calculation.
- The UKF comes with another solution to the problem of calculating these covariance terms in the non-linear context.

The Unscented Transform

Lecture 17

- Consider a random variable $\boldsymbol{a} \in \mathbb{R}^n$ with mean $E[\boldsymbol{a}] = \bar{\boldsymbol{a}}$ and (auto-)covariance $E[(\boldsymbol{a} - \bar{\boldsymbol{a}})(\boldsymbol{a} - \bar{\boldsymbol{a}})^T] = \boldsymbol{C}$
- Next consider a non-linear function $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^m$ and consider the characteristics of the random variable $\boldsymbol{b} = \mathbf{g}(\boldsymbol{a})$
- The UT is a method for approximating the mean and auto-covariance of the random variable $\boldsymbol{b}$, when we know the function $\mathbf{g}$, but only the mean and auto-covariance of $\boldsymbol{a}$ (we would need to know the entire pdf of $\boldsymbol{a}$ to attempt an exact computation).

The Unscented Transform
Lecture 17

- It is a type of heuristic.
- The main idea is to draw a finite set of **well-selected and deterministically chosen** samples points from the random variable ***a***, apply the non-linear function **g** to these samples and use a type of sample mean and sample covariance as approximates for true statistics of the random variable $\boldsymbol{b} = \mathbf{g}(\boldsymbol{a})$.
- A main idea with the UT, is to carefully select the sample points and keep their number reduced (not doing Monte Carlo here).
- The sample points are called **sigma-points**. The name **Sigma-Points Kalman Filter** is also used.

The Scaled Unscented Transform

Lecture 17

- Scalar parameters α, β, κ, e.g. $\alpha = 10^{-3}, \kappa = 0, \beta = 2$. Derived scalar quantity $\lambda = \alpha^2(n+\kappa) - n$;
- Let $(n+\lambda)\mathbf{C}_a = \mathbf{L}\mathbf{L}^T$, the Cholesky decomposition of the scaled covariance matrix $\mathbf{C}_a$ where $\mathbf{L}$ is lower triangular;
- The sigma points are defined as: $\mathbf{s}_0 = \bar{\boldsymbol{a}}$, $\mathbf{s}_{i+1} = \bar{\boldsymbol{a}} + \mathbf{L}[:, i]$, $\mathbf{s}_{i+1+n} = \bar{\boldsymbol{a}} - \mathbf{L}[:, i]$, $i = 0, \dots, n-1$;
- Two sets (set m and set c) of weights are defined as well $W_0^m = \lambda/(n+\lambda)$, $W_0^c = \lambda/(n+\lambda) + (1 - \alpha^2 + \beta)$, and $W_j^m = W_j^c = 1/\left(2(n+\lambda)\right), j = 1 \dots 2n$;
- Then $\bar{\boldsymbol{b}} = E[\mathbf{g}(\boldsymbol{a})] \approx \tilde{\boldsymbol{b}} = \sum_{j=0}^{2n} W_j^m \mathbf{g}(\mathbf{s}_j)$, and $\mathbf{C}_b = E[(\mathbf{g}(a) - \bar{\boldsymbol{b}})(\mathbf{g}(a) - \bar{\boldsymbol{b}})^T] \approx \tilde{\mathbf{C}}_b = \sum_{j=0}^{2n} W_j^c (\mathbf{g}(\mathbf{s}_j) - \tilde{\boldsymbol{b}})(\mathbf{g}(\mathbf{s}_j) - \tilde{\boldsymbol{b}})^T$.

Exercise

Lecture 17

- Generate the sigma-points according to the unscaled unscented transform for the random variable $\boldsymbol{a} \in \mathbb{R}^4$ with the following statistics:

$$\bar{\boldsymbol{a}} = \begin{bmatrix} 0 & 1 & -2 & \text{[illegible]} \end{bmatrix}^T$$

$$\mathbf{C}_a = \begin{bmatrix} \text{[illegible]} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 9 \end{bmatrix}$$

Unscented Kalman Filter

Lecture 17

- Unscented Kalman Filter Predict Step:

$$\mathbf{x}_j = \mathbf{f}\left(\mathbf{s}_j\left(\hat{\mathbf{x}}_{k-1}, \mathbf{P}_{k-1}\right)\right);\ j = 0, \ldots, 2n$$

$$\hat{\mathbf{x}}_{k|k-1} = \sum_{j=0}^{2n} W_j^m \mathbf{x}_j;\ \mathbf{P}_{k|k-1} = \sum_{j=0}^{2n} W_j^c \left(\mathbf{x}_j - \hat{\mathbf{x}}_{k|k-1}\right)\left(\mathbf{x}_j - \hat{\mathbf{x}}_{k|k-1}\right)^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T$$

- Kalman Filter Update Step:

$$\mathbf{x}_j = \mathbf{s}_j\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}\right),\ \mathbf{y}_j = \mathbf{h}(\mathbf{x}_j),\ j = 0, \ldots, 2n$$

$$\hat{\mathbf{y}}_k = \sum_{j=0}^{2n} W_j^m \mathbf{y}_j,\ \mathbf{C}_y = \sum_{j=0}^{2n} W_j^c \left(\mathbf{y}_j - \hat{\mathbf{y}}_k\right)\left(\mathbf{y}_j - \hat{\mathbf{y}}_k\right)^T$$

Unscented Kalman Filter

Lecture 17

- Kalman Filter Update Step (cont):

$$\mathbf{C}_{xy} = \sum_{j=0}^{2n} W_j^c (\mathbf{x}_j - \hat{\mathbf{x}}_{k|k-1})(\mathbf{y}_j - \hat{\mathbf{y}}_k)^T$$

$$\mathbf{K}_k = \mathbf{C}_{xy}(\mathbf{C}_y + \mathbf{R})^{-1}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k)$$

$$\mathbf{P}_k = \mathbf{P}_{k|k-1} - \mathbf{K}_k(\mathbf{C}_y + \mathbf{R})\mathbf{K}_k^T$$

- with some given initial conditions $\hat{\mathbf{x}}_0$, $\mathbf{P}_0$.

There is an alternative form where the sigma-points are extracted from an augmentated state including the noise.

Joseph Form and square Root Filter

Lecture 17

- Joseph form implements directly for EKF
- Joseph form for UKF makes use of the following approximation $\mathbf{H} = \mathbf{C}_{xy}^{T}\mathbf{P}_{k|k-1}^{-}$,
- Square Root implementations [ToDo]

UKF Exercise

Lecture 17

Implement the UKF filter and test it for the 1D rocket system and the pendulum model.

EKF and UKF
Lecture 17

Occasional problems with divergence. Some potential fixes:

- [Perea et all. 2007] "Artificial" increases in covariance e.g. B-EKF1: $\mathbf{C}_y = \alpha \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}$, $\alpha \in \mathbb{R}^+$, e.g. [illegible]; see also B-EKF2, B-EKF3, B-EKF4 in this paper.
- UKFz [Perea et all. 2007]: Replacing $\hat{\mathbf{y}}_k$ from $\hat{\mathbf{y}}_k = \sum_{j=0}^{2n} W_j^m \mathbf{y}_j$ to $\mathbf{h}(\hat{\mathbf{x}}^{k|k-1})$. Affects the calculation of $\mathbf{C}_y$ and $\mathbf{C}_{sy}$ and $\hat{\mathbf{x}}_k$;
- IUKF [Perea et all. 2008]: Use $\hat{\mathbf{y}}_k$ from $\hat{\mathbf{y}}_k = \sum_{j=0}^{2n} W_j^m \mathbf{y}_j$ to $\mathbf{h}(\hat{\mathbf{x}}^{k|k-1})$ but only in $\hat{\mathbf{x}}_k$;

Equality Constrained EKF/UKF

Lecture 17

- Same procedure as in the linear Kalman filter, project the state onto the constraint subspace, same closed analytical solution.

Interval-Constrained UKF

Lecture 17

- [Teixeira et all 2010] UKF with state interval constraints (a very useful case). See especially the combination where ICUT is used in the predict step and at the end of the update step a PDF truncation procedure is performed (the TIUKF in the paper);
- ICUT is an interval constraint sigma-point generation procedure;
- The PDF truncation procedure modifies both the $\hat{\mathbf{x}}_k$ and $\mathbf{P}_k$ using an iterative procedure to enforce each of the interval constraints one by one.

Well done!