Chapter 6

Localization with Centralized State Estimation

In the centralized case, anchors with known positions are present to localize tags. As shown in Chapter 4, state estimation techniques should be pursued to account for random errors, under-determined equations caused by missing measurements and for describing the estimated error by means of a covariance matrix. Chapter 2 describes various techniques to deterministically calculate the position of a tag. This chapter uses these techniques in order to design Bayesian filters for the centralized case. These filters are then tested with real and simulated data.

6.1 State Transition Model

W.l.o.g. two dimensions are used for describing the movement of a tracked object. The state vector $\underline{x}$ contains the position and velocity of the object $\begin{bmatrix} x & y & \dot{x} & \dot{y} \end{bmatrix}^{\mathrm{T}}$, which in this case is the tag. In case the acceleration system input is not available for the state transition model, the state vector can additionally contain the estimated acceleration of the tracked object $\begin{bmatrix} x & y & \dot{x} & \dot{y} & \ddot{x} & \ddot{y} \end{bmatrix}^{\mathrm{T}}$.

The state transition model maps the state from the current time-step k to the next time-step $k+1$. For describing motion with a constant step-size T, the state transition model is linear and time-invariant. For the state vector $\underline{x}$ and the system input $\underline{u}$

$$\underline{x} = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix}, \ \underline{u} = \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix}, \tag{6.1}$$

where the u_x and u_y depict the acceleration caused to the system due to the system input, the state transition model can be described as

$$\underline{x}_{k+1} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \underline{x}_k + \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix} \underline{u}_k . \tag{6.2}$$

However, it is important to note that the system input is not always available. If acceleration measurements are also not available, as in the case of the chosen hardware, non-linear motion can no longer be described using the existing model. For instance, if the system input in (6.2) is no longer present, the state transition model assumes a constant velocity. Similarly, extending the state space to include the acceleration components, would result in a constant acceleration model, which is described by Equation (6.3) for two dimensions. These two models assume a constant velocity or acceleration.

Therefore, a solution for a viable state transition model without system input, accelerometer or odometry readings is needed. A possible method is using the nearly constant velocity (NCV) or the nearly constant acceleration (NCA) model [48, 7].

For better dynamics, NCA is chosen over NCV. In this case, the system input is expressed using a white noise acceleration term σ^w. The NCA model describes the state transition of the estimated state vector $\hat{\underline{x}} = \begin{bmatrix} x & y & \dot{x} & \dot{y} & \ddot{x} & \ddot{y} \end{bmatrix}^{\mathrm{T}}$ with

$$\hat{\underline{x}}_{k+1} = \underbrace{\begin{bmatrix} 1 & 0 & T & 0 & \frac{T^2}{2} & 0 \\ 0 & 1 & 0 & T & 0 & \frac{T^2}{2} \\ 0 & 0 & 1 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{A}} \hat{\underline{x}}_k + \underline{\boldsymbol{w}}_k \tag{6.3}$$

and the noise variance matrix $\mathbf{Q}^w = (\sigma^w)^2$, which is the variance of of the white noise acceleration. This noise term is added to the covariance matrix of the system at each time step as previously shown in (4.12).

$$\mathbf{C}_{k+1} = \mathbf{A}\mathbf{C}_k\mathbf{A}^{\mathrm{T}} + \underbrace{\mathbf{B}\mathbf{Q}^w\mathbf{B}^{\mathrm{T}}}_{\mathbf{C}_k^w} \tag{6.4}$$

with

$$\mathbf{B} = \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.5}$$

This is logical because if the movement of an object is deemed to be *nearly constant*, an uncertainty is present when using the constant acceleration assumption. This uncertainty is added with every state transition used for prediction. The noise term σ^w is set based on how quickly the tracked object can accelerate. Therefore, this value would be different for tracking a person when compared to for example, tracking an airplane.

The matrix $\mathbf{Q}^w$ can also contain different noise terms depending on the direction [48]. For example, in the 2-D case, it is possible to construct this matrix from two individual terms that describe the acceleration noise terms in x and y directions.

$$\mathbf{Q}^w = \begin{bmatrix} (\sigma_x^w)^2 & 0 \\ 0 & \left(\sigma_y^w\right)^2 \end{bmatrix} \tag{6.6}$$

Because the NCA model describes the uncertainty of non-linear movement appropriately, Bayesian state estimation approaches can filter the state using measurements. If this noise term were not incorporated, the state transition model would only be true for objects that only undergo constant acceleration. If an object moving with a non-constant acceleration were to be tracked using a constant acceleration model (6.3), the result would diverge from the real value. An illustration of this is provided by Figure 6.1. In this example, three Kalman filters using NCA, constant velocity and constant acceleration models receive positions of an object and track them accordingly. Only the Kalman filter using a NCA model does not diverge from the trajectory. The non-linear trajectory ensures that the assumption of a constant velocity or a constant acceleration no longer hold. This section only explores NCA as it is eventually implemented for the centralized and cooperative scenarios. A survey paper written by Li et al. provides a general overview of other possible methods [48].

6.2 Considered Bayesian Approaches

A variety of Bayesian approaches are considered. Chapter 3 introduces UWB ranging. As this ranging technique is based on time of flight, the obtained distance is described by the equations from Section 2.4.1.2. A linear Kalman filter is used in combination with solver. The solver solves the non-linear distance equation for the linear Kalman filter. The resulting position is treated as sensor data for the linear Kalman filter. To do away with this two step process of first positioning and then filtering with a Kalman filter, the EKF and the UKF are also considered as they can incorporate a non-linear measurement model in order to map the distance measurements to the state vector.

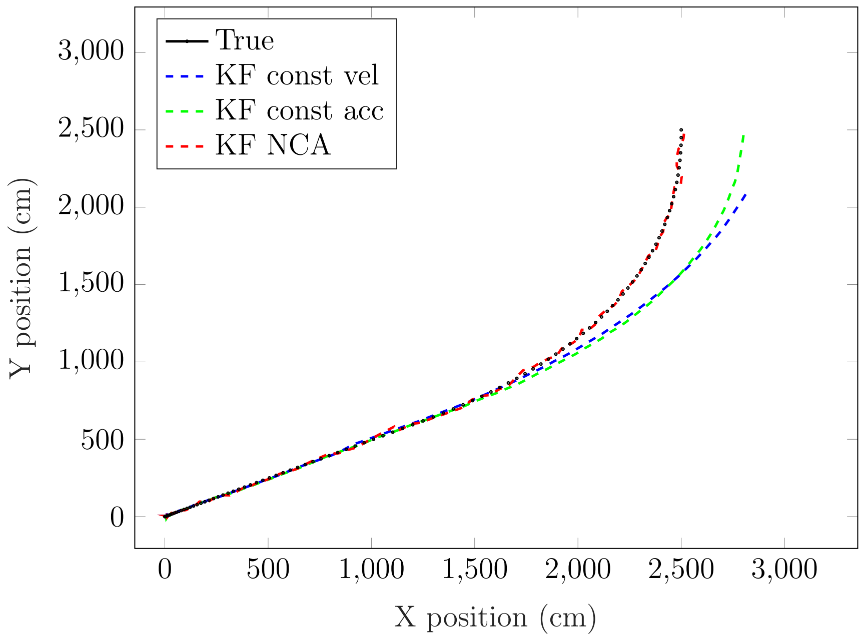

Figure 6.1: Comparison of using NCA, constant velocity and constant acceleration models for implementing a Kalman filter for tracking.

6.2.1 Linear Kalman Filter

The linear Kalman filter, which is more generally explained in Chapter 4, can be applied for this case using the equations

$$
\begin{aligned}
\hat{\underline{x}}^{\mathrm{p}}_{k+1} &= \mathbf{A}\hat{\underline{x}}^{\mathrm{e}}_{k} \\
\mathbf{C}^{\mathrm{p}}_{k+1} &= \mathbf{A}\mathbf{C}^{\mathrm{e}}_{k}\mathbf{A}^{\mathrm{T}} + \underbrace{\mathbf{B}\mathbf{Q}^{w}\mathbf{B}^{\mathrm{T}}}_{\mathbf{C}^{w}} \\
\mathbf{K}_{k} &= \mathbf{C}^{\mathrm{p}}_{k}\mathbf{H}^{\mathrm{T}}\left(\mathbf{C}^{z}_{k} + \mathbf{H}\mathbf{C}^{\mathrm{p}}_{k}\mathbf{H}^{\mathrm{T}}\right)^{-1} \\
\hat{\underline{x}}^{\mathrm{e}}_{k} &= \hat{\underline{x}}^{\mathrm{p}}_{k} + \mathbf{K}_{k}\left(\hat{\underline{z}}_{k} - \mathbf{H}\hat{\underline{x}}^{\mathrm{p}}_{k}\right) \\
\mathbf{C}^{\mathrm{e}}_{k} &= \mathbf{C}^{\mathrm{p}}_{k} - \mathbf{K}_{k}\mathbf{H}\mathbf{C}^{\mathrm{p}}_{k}.
\end{aligned}
\tag{6.7}
$$

In this case, the state transition and measurement models $\mathbf{A}$ and $\mathbf{H}$ are linear and time-invariant. The measurement model, which maps the state space to the measurement space

$$\underline{z}_k = \mathbf{H}\underline{x}_k, \tag{6.8}$$

is given by

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{6.9}$$

for the 2-D case. The realisation $\hat{\underline{z}}_k$, or, in other words, the obtained measurement, is provided by one of the trilateration methods shown in Section 2.4. For simplification, the trilateration method described as solving for the distance from the origin in this section is from now on referred to as the *origin* LSQ solver. Similarly, the method described as linearising the distance equation by subtraction is referred to as the *subtraction* LSQ solver.

6.2.2 Measurement Density

By designing the linear Kalman filter as described in the previous section, a paradoxical situation is reached. If the measurement noise of the trilaterated positions is assumed to be Gaussian, it follows that the distance measurements have a non-Gaussian density because as Chapter 4 states, passing a Gaussian density through a non-linear function results in a non-Gaussian density. Likewise, if the ranging measurements are assumed to be Gaussian, the positions after trilateration are no longer Gaussian.

In general, the raw measurement data is assumed to be Gaussian. Following this logic, similar literature assumes that the ranging measurements are Gaussian [49, 72, 29]. It is, however, clear from Section 3.2 that this is not the case if certain calibrations cannot be performed. As filtering for non-Gaussian noise distributions is outside the scope of this thesis, other considered Bayesian state estimators, such as the EKF and the UKF assume that the ranging measurements are Gaussian.

Though this decision leads to a Gaussian measurement noise assumption no longer holding true for the linear Kalman filter, its results are still compared to the UKF and EKF.

6.2.3 Extended Kalman Filter

The EKF can be used to linearise the distance equation

$$h_i(\underline{x}_k) = \sqrt{(P_i^x - x_k)^2 + (P_i^y - y_k)^2}, i = 1 \ldots N. \tag{6.10}$$

$$h_i(\underline{x}_k) = \sqrt{(P_i^x - x_k)^2 + (P_i^y - y_k)^2}, i = 1 \ldots N. \tag{6.11}$$

The N anchor positions P_i are stationary in the centralized scenario. As described in Chapter 4, the measurement matrix is approximated using a first-order Taylor approximation

$$\mathbf{H}_k = \frac{\partial h_k(\cdot)}{\partial \underline{x}_k}\bigg|_{\underline{x}_k = \hat{\underline{x}}^{\mathrm{p}}_{k+1}} = \begin{bmatrix} \frac{x_k - P^x_1}{\sqrt{(x_k-P^x_1)^2+(y_k-P^y_1)^2}} & \frac{y_k - P^y_1}{\sqrt{(x_k-P^x_1)^2+(y_k-P^y_1)^2}} & 0 & 0 & 0 & 0 \\ \frac{x_k - P^x_1}{\sqrt{(x_k-P^x_2)^2+(y_k-P^y_2)^2}} & \frac{y_k - P^y_2}{\sqrt{(x_k-P^x_2)^2+(y_k-P^y_2)^2}} & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{x_k - P^x_N}{\sqrt{(x_k-P^x_N)^2+(y_k-P^y_N)^2}} & \frac{y_k - P^y_N}{\sqrt{(x_k-P^x_N)^2+(y_k-P^y_N)^2}} & 0 & 0 & 0 & 0 \end{bmatrix} \tag{6.12}$$

for the 2-D case. The position components of $\hat{\underline{x}}^{\mathrm{p}+1}_k$, namely $\hat{x}^{\mathrm{p}}_{k+1}$ and $\hat{y}^{\mathrm{p}}_{k+1}$, are used to determine this matrix as the rest of the partial derivatives are equal to 0.

Otherwise, the implementation does not differ much from the linear Kalman filter. A similar NCA model is used. Due to the linear state transition model $\mathbf{A}$, only the estimate update equation

$$\hat{\underline{x}}^{\mathrm{e}}_k = \hat{\underline{x}}^{\mathrm{p}}_k + \mathbf{K}_k\left(\hat{\underline{z}}_k - \underline{h}\left(\hat{\underline{x}}^{\mathrm{p}}_k\right)\right) \tag{6.13}$$

is different due to the non-linear measurement model.

6.2.4 Unscented Kalman Filter

The UKF for the centralized scenario is almost adequately described in Section 4.5.2. Analogously to the EKF and the linear Kalman filter, the state transition model is linear and time-invariant, while the measurement model is time-invariant. Only the covariance prediction equation

$$\mathbf{C}^{\mathrm{p}}_{k+1} = \sum_{i=0}^{2N} w_i^{(c)}\left(\underline{\xi}^{\mathrm{p}}_i - \hat{\underline{x}}^{\mathrm{p}}_{k+1}\right)\left(\underline{\xi}^{\mathrm{p}}_i - \hat{\underline{x}}^{\mathrm{p}}_{k+1}\right)^{\mathrm{T}} + \underbrace{\mathbf{B}\mathbf{Q}^w\mathbf{B}^{\mathrm{T}}}_{\mathbf{C}^w} \tag{6.14}$$

has to be tweaked to suit the NCA model.

6.2.5 Adaptive Robust Extended Kalman Filter

An interesting idea that was pursued during the course of this thesis was the adaptive robust extended Kalman filter proposed by Li et al. specifically for UWB [49]. The filter is based on the assumption that the ranging bias error can be described as an exponential function of the range d

$$f(d) = 0.4\left(1.10 - e^{-0.17d}\right) \tag{6.15}$$