Gaussian Filters: Extended Kalman Filter

Corina Barbalata

Mechanical and Industrial Engineering

iCORE Laboratory

cbarbalata@lsu.edu

Readings for this class

- Chapter 3.3: The Extended Kalman Filters (pages 48-54) from *Probabilistic Robotics*, Sebastian Thrun

Motivation

- The assumptions of linear state transitions and linear measurements with added Gaussian noise are rarely fulfilled in practice.
- Most realistic robotic problems involve nonlinear functions.

$$\cancel{\begin{aligned} x_k &= A_k x_{k-1} + B_k u_k + w_k \\ z_k &= H_k x_k + v_k \end{aligned}} \quad \Rightarrow \quad \begin{aligned} x_k &= f(u_k, x_{k-1}, w_k) \\ z_k &= h(x_k, v_k) \end{aligned}$$

The Extended Kalman Filter (EKF)

EKF is a Kalman Filter that makes use of a 1st order Linear Approximation (first order Taylor expansion)

Prediction:

$$f(x_{k-1}, u_k, w_k) \approx f(\hat{x}_{k-1}, u_k, 0) + \left.\frac{\partial f(x_{k-1}, u_k, w_k)}{\partial x_{k-1}}\right|_{(\hat{x}_{k-1}, u_k, 0)} (x_{k-1} - \hat{x}_{k-1}) + \left.\frac{\partial f(x_{k-1}, u_k, w_k)}{\partial w_k}\right|_{(\hat{x}_{k-1}, u_k, 0)} w_k$$

$$\boxed{f(x_{k-1}, u_k, w_k) \approx f(\hat{x}_{k-1}, u_k, 0) + A_k (x_{k-1} - \hat{x}_{k-1}) + W_k w_k}$$

Linear equations to be used

Correction:

$$h(x_k, v_k) \approx h(\hat{x}_k^-, 0) + \left.\frac{\partial h(x_k, v_k)}{\partial x_k}\right|_{(\hat{x}_k^-, 0)} (x_k - \hat{x}_k^-) + \left.\frac{\partial h(x_k, v_k)}{\partial v_k}\right|_{(\hat{x}_k^-, 0)} v_k$$

$$\boxed{h(x_k, v_k) \approx h(\hat{x}_k^-, 0) + H_k (x_k - \hat{x}_k^-) + V_k v_k}$$

The Extended Kalman Filter (EKF)

- The state is predicted using the nonlinear model

$$\hat{x}_k = f(u_k, \hat{x}_{k-1}, 0)$$

- The covariance is projected using the linear approximation

$$\hat{x}_k^- = f(x_{k-1}, u_k, w_k) \simeq f(\hat{x}_{k-1}, u_k, 0) + A_k(x_{k-1} - \hat{x}_{k-1}) + W_k w_k$$

$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_k W_k^T$$

where

$$A_k = \frac{\partial f(x_{k-1}, u_k, w_k)}{\partial x_{k-1}} \,\Big|_{(\hat{x}_{k-1}, u_k, 0)}$$

$$W_k = \frac{\partial f(x_{k-1}, u_k, w_k)}{\partial w_k} \,\Big|_{(\hat{x}_{k-1}, u_k, 0)}$$

The Extended Kalman Filter (EKF)

- The state is predicted using the nonlinear model
- The covariance is projected using the linear approximation

Kalman Filter $\left(\hat{x}_{k-1}, P_{k-1}, u_k, z_k\right)$

$$\boxed{\begin{aligned} \hat{x}_k^- &= A_k \hat{x}_{k-1} + B_k u_k \\ P_k^- &= A_k P_{k-1} A_k^T + Q_k \end{aligned}}$$

$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k\right)^{-1}$$

$$\hat{x}_k = \hat{x}_k^- + K_k \left(z_k - H_k \hat{x}_k^-\right)$$

$$P_k = \left(I - K_k H_k\right) P_k^-$$

$$return\left(\hat{x}_k, P_k\right)$$

$\Rightarrow$

Extended Kalman Filter $\left(\hat{x}_{k-1}, P_{k-1}, u_k, z_k\right)$

$$\boxed{\begin{aligned} \hat{x}_k^- &= f\left(\hat{x}_{k-1}, u_k, 0\right) \\ P_k^- &= A_k P_{k-1} A_k^T + W_k Q_k W_k^T \end{aligned}}$$

$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T\right)^{-1}$$

$$\hat{x}_k = \hat{x}_k^- + K_k \left(z_k - h\left(\hat{x}_k^-, 0\right)\right)$$

$$P_k = \left(I - K_k H_k\right) P_k^-$$

$$return\left(\hat{x}_k, P_k\right)$$

The Extended Kalman Filter (EKF)

- Since the measurement noise is nonlinear

$$z_k = h(x_k, v_k)$$

- Its covariance has to be linearized with the K gain

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1}$$

where

$$H_k = \frac{\partial h(x_k, v_k)}{\partial x_k} \Big|_{(\hat{x}_k^-, 0)}$$

$$V_k = \frac{\partial h(x_k, v_k)}{\partial v_k} \Big|_{(\hat{x}_k^-, 0)}$$

The Extended Kalman Filter (EKF)

- Since the measurement noise is nonlinear
- Its covariance has to be linearized with the K gain

Kalman Filter $(\hat{x}_{k-1}, P_{k-1}, u_k, z_k)$

$$\hat{x}_k^- = A_k \hat{x}_{k-1} + B_k u_k$$
$$P_k^- = A_k P_{k-1} A_k^T + Q_k$$
$$\boxed{K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k\right)^{-1}}$$
$$\hat{x}_k = \hat{x}_k^- + K_k \left(z_k - H_k \hat{x}_k^-\right)$$
$$P_k = \left(I - K_k H_k\right) P_k^-$$
$$return \left(\hat{x}_k, P_k\right)$$

Extended Kalman Filter $(\hat{x}_{k-1}, P_{k-1}, u_k, z_k)$

$$\hat{x}_k^- = f\left(\hat{x}_{k-1}, u_k, 0\right)$$
$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_k W_k^T$$
$$\boxed{K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T\right)^{-1}}$$
$$\hat{x}_k = \hat{x}_k^- + K_k \left(z_k - h\left(\hat{x}_k^-, 0\right)\right)$$
$$P_k = \left(I - K_k H_k\right) P_k^-$$
$$return \left(\hat{x}_k, P_k\right)$$

The Extended Kalman Filter (EKF)

- Since the measurement noise is nonlinear, the estimate based on the motion model is:
$$z_k = h(x_k, v_k)$$
- Update equations remain almost the same as for Kalman Filter:

$$\hat{x} = \hat{x}_k^- + K_k(z_k - h(\hat{x}_k^-, 0))$$

$$P_k = (I - K_k H_k) P_k^-$$

The Extended Kalman Filter (EKF)

- Since the measurement noise is nonlinear ....
- Update equations remain almost the same

Kalman Filter $(\hat{x}_{k-1}, P_{k-1}, u_k, z_k)$

$$\hat{x}_k^- = A_k \hat{x}_{k-1} + B_k u_k$$
$$P_k^- = A_k P_{k-1} A_k^T + Q_k$$
$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k \right)^{-1}$$
$$\boxed{\begin{aligned} \hat{x}_k &= \hat{x}_k^- + K_k \left(z_k - H_k \hat{x}_k^- \right) \\ P_k &= \left(I - K_k H_k \right) P_k^- \end{aligned}}$$
$$return \left(\hat{x}_k, P_k \right)$$

$\Rightarrow$

Extended Kalman Filter $(\hat{x}_{k-1}, P_{k-1}, u_k, z_k)$

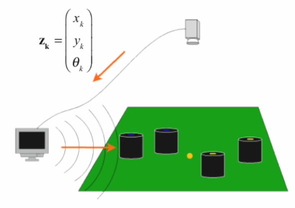
$$\hat{x}_k^- = f\left(\hat{x}_{k-1}, u_k, 0 \right)$$
$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_k W_k^T$$
$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T \right)^{-1}$$
$$\boxed{\begin{aligned} \hat{x}_k &= \hat{x}_k^- + K_k \left(z_k - h\left(\hat{x}_k^-, 0 \right) \right) \\ P_k &= \left(I - K_k H_k \right) P_k^- \end{aligned}}$$
$$return \left(\hat{x}_k, P_k \right)$$

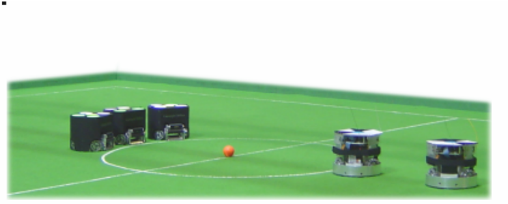
How good is the linearization?

- The accuracy of Taylor expansions depends on two factors:
  1. The degree of nonlinearity in the system
  2. The width of the posteriori.
- Extended filters tend to yield good results if the state of the system is known with relatively high accuracy, so the remaining covariance is small.
- The larger the uncertainty, the higher the error introduced by the linearization.

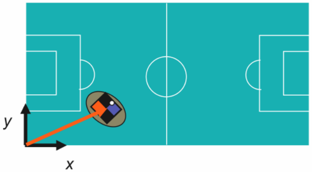
Example: 3 DOF Mobile Robot

- 3 DOF: (x, y, θ)
- Differential drive robot with odometry $(\Delta x, \Delta y, \Delta\theta)$
- Top view camera provides absolute position and heading fixes





Example: 3 DOF Mobile Robot



1. Robot position is sensed with vision

Algorithm: EKF Localization of a 3DOF Mobile Robot
With odometry & position upd

{- Pose initialization}

$\mathbf{x}_0^B = \hat{\mathbf{x}}_0^B; \mathbf{P}_0^B = \hat{\mathbf{P}}_0^B;$

for k=1 to steps **do**

$\left[\mathbf{u}_k^{R_{k-1}}, \mathbf{Q}_k\right] = get\_odometry()$

{-EKF prediction}

$\left[\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B\right] = move\_vehicle\left(\mathbf{x}_{k-1}^B, \mathbf{P}_{k-1}^B, \mathbf{u}_k^{R_{k-1}}, \mathbf{Q}_k\right)$

$\left[\mathbf{z}_k, \mathbf{R}_k\right] = get\_measurements$

{-EKF update}

$\left[\mathbf{x}_k^B, \mathbf{P}_k^B\right] = update\_position\left(\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B, \mathbf{z}_k, \mathbf{R}_k\right)$

end for

Example: 3 DOF Mobile Robot

1. Robot position is sensed with vision

2. The robot moves and its displacement is measured with odometry

Algorithm: EKF Localization of a 3DOF Mobile Robot
With odometry & position upd

{- Pose initialization}

$\mathbf{x}_0^B = \hat{\mathbf{x}}_0^B; \mathbf{P}_0^B = \hat{\mathbf{P}}_0^B;$

for k=1 to steps **do**

$\left[\mathbf{u}_k^{R_{k-1}}, \mathbf{Q}_k\right] = get\_odometry()$

{-EKF prediction}

$\left[\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B\right] = move\_vehicle\left(\mathbf{x}_{k-1}^B, \mathbf{P}_{k-1}^B, \mathbf{u}_k^{R_{k-1}}, \mathbf{Q}_k\right)$

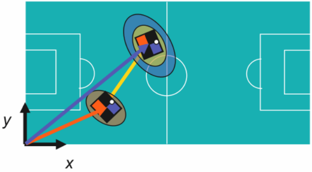
$\left[\mathbf{z}_k, \mathbf{R}_k\right] = get\_measurements$

{-EKF update}

$\left[\mathbf{x}_k^B, \mathbf{P}_k^B\right] = update\_position\left(\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B, \mathbf{z}_k, \mathbf{R}_k\right)$

end for

Example: 3 DOF Mobile Robot



1. Robot position is sensed with vision

2. The robot moves and its displacement is measured with the odometry

3. The robot position is predicted using the odometry

Algorithm: EKF Localization of a 3DOF Mobile Robot
With odometry & position upd

$\{$- Pose initialization$\}$

$\mathbf{x}_0^B = \hat{\mathbf{x}}_0^B;\ \mathbf{P}_0^B = \hat{\mathbf{P}}_0^B;$

for k=1 to steps **do**

$\left[\mathbf{u}_k^{R_{k-1}}, \mathbf{Q}_k\right] = \mathit{get\_odometry}()$

$\{$-EKF prediction$\}$

$\left[\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B\right] = \mathit{move\_vehicle}\left(\mathbf{x}_{k-1}^B, \mathbf{P}_{k-1}^B, \mathbf{u}_k^{R_{k-1}}, \mathbf{Q}_k\right)$

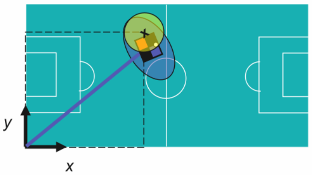
$\left[\mathbf{z}_k, \mathbf{R}_k\right] = \mathit{get\_measurements}$

$\{$-EKF update$\}$

$\left[\mathbf{x}_k^B, \mathbf{P}_k^B\right] = \mathit{update\_position}\left(\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B, \mathbf{z}_k, \mathbf{R}_k\right)$

end for

Example: 3 DOF Mobile Robot



1. Robot position is sensed with vision
2. The robot moves and its displacement is measured with the odometry
3. The robot position is predicted using the odometry
4. An absolute position fixe is read from the vision system

Algorithm: EKF Localization of a 3DOF Mobile Robot
With odometry & position upd

$\{$- Pose initialization$\}$

$\mathbf{x}_0^B = \hat{\mathbf{x}}_0^B; \mathbf{P}_0^B = \hat{\mathbf{P}}_0^B;$

for k=1 to steps **do**

$\left[\mathbf{u}_k^{R_{k-1}}, \mathbf{Q}_k\right] = get\_odometry()$

$\{$-EKF prediction$\}$

$\left[\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B\right] = move\_vehicle\left(\mathbf{x}_{k-1}^B, \mathbf{P}_{k-1}^B, \mathbf{u}_k^{R_{k-1}}, \mathbf{Q}_k\right)$

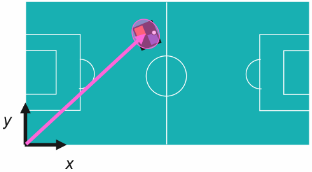
$\left[\mathbf{z}_k, \mathbf{R}_k\right] = get\_measurements$

$\{$-EKF update$\}$

$\left[\mathbf{x}_k^B, \mathbf{P}_k^B\right] = update\_position\left(\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B, \mathbf{z}_k, \mathbf{R}_k\right)$

end for

Example: 3 DOF Mobile Robot



1. Robot position is sensed with vision

2. The robot moves and its displacement is measured with the odometry

3. The robot position is predicted using the odometry

4. An absolute position fixe is read from the vision system

6. The robot position is updated.

Algorithm: EKF Localization of a 3DOF Mobile Robot
With odometry & position upd

$\{$- Pose initialization$\}$

$\mathbf{x}_0^B = \hat{\mathbf{x}}_0^B;\ \mathbf{P}_0^B = \hat{\mathbf{P}}_0^B;$

for k=1 to steps **do**

$\left[\mathbf{u}_k^{R_{k-1}}, \mathbf{Q}_k\right] = get\_odometry()$

$\{$-EKF prediction$\}$

$\left[\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B\right] = move\_vehicle\left(\mathbf{x}_{k-1}^B, \mathbf{P}_{k-1}^B, \mathbf{u}_k^{R_{k-1}}, \mathbf{Q}_k\right)$

$\left[\mathbf{z}_k, \mathbf{R}_k\right] = get\_measurements$

$\{$-EKF update$\}$

$\left[\mathbf{x}_k^B, \mathbf{P}_k^B\right] = update\_position\left(\mathbf{x}_{k|k-1}^B, \mathbf{P}_{k|k-1}^B, \mathbf{z}_k, \mathbf{R}_k\right)$

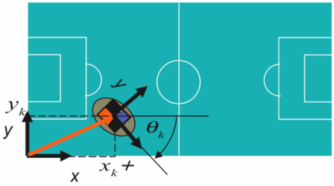
end for

Example: 3 DOF Mobile Robot

EKF Formulation:

- Define the state vector
- Define the nonlinear process model with an explicit noise representation
- Compute the matrixes

Example: 3 DOF Mobile Robot



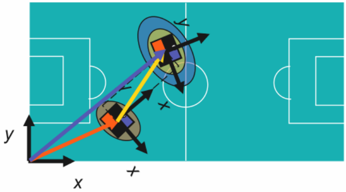
Extended Kalman Filter $(\hat{x}_{k-1}, P_{k-1}, u_k, z_k)$

$$\hat{x}_k^- = f(\hat{x}_{k-1}, u_k, 0)$$
$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_k W_k^T$$
$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T\right)^{-1}$$
$$\hat{x}_k = \hat{x}_k^- + K_k\left(z_k - h(\hat{x}_k^-, 0)\right)$$
$$P_k = (I - K_k H_k) P_k^-$$
$$return(\hat{x}_k, P_k)$$

State Definition

$$\mathbf{x}_k = \begin{pmatrix} x_k \\ y_k \\ \theta_k \end{pmatrix}$$

Example: 3 DOF Mobile Robot



Extended Kalman Filter $\left(\hat{x}_{k-1}, P_{k-1}, u_k, z_k\right)$

$$\hat{x}_k^- = f\left(\hat{x}_{k-1}, u_k, 0\right)$$

$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_k W_k^T$$

$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T\right)^{-1}$$

$$\hat{x}_k = \hat{x}_k^- + K_k\left(z_k - h\left(\hat{x}_k^-, 0\right)\right)$$

$$P_k = \left(I - K_k H_k\right) P_k^-$$

$$\textit{return}\left(\hat{x}_k, P_k\right)$$

State Definition

$$\mathbf{x}_k^{\mathrm{B}} = \begin{pmatrix} x_k \\ y_k \\ \theta_k \end{pmatrix}$$

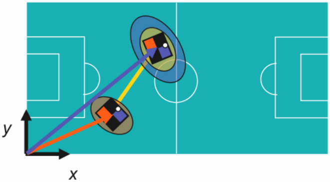
Proces Model

$$\mathbf{x}_k^{\mathrm{B}} = \mathbf{x}_{k-1}^{\mathrm{B}} \oplus \mathbf{u}_k^{R_{k-1}};$$

Next Robot Position	=	Previous Robot Position	+	Displacement Measured with the odometry

Next position is the previous one ***compounded*** *with the odometry*

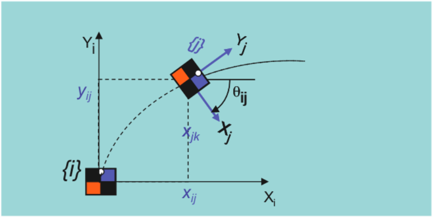
Example: 3 DOF Mobile Robot



Extended Kalman Filter $\left(\hat{x}_{k-1}, P_{k-1}, u_k, z_k\right)$

$$\hat{x}_k^- = f\left(\hat{x}_{k-1}, u_k, 0\right)$$
$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_k W_k^T$$
$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T\right)^{-1}$$
$$\hat{x}_k = \hat{x}_k^- + K_k\left(z_k - h\left(\hat{x}_k^-, 0\right)\right)$$
$$P_k = \left(I - K_k H_k\right) P_k^-$$
$$return\left(\hat{x}_k, P_k\right)$$

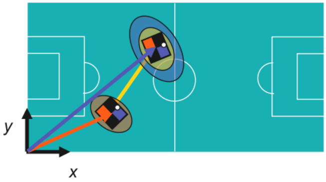
What is compounding?



$$\mathbf{x}_{ij} = \begin{pmatrix} x_{ij} \\ y_{ij} \\ \theta_{ij} \end{pmatrix}$$

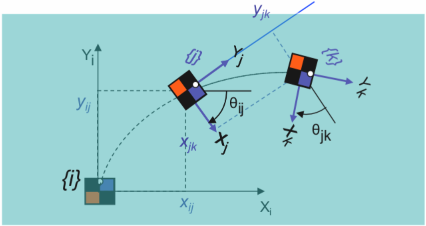
$\mathbf{x}_{ij}$: Represents an instant robot motion from {i} to {j}

Example: 3 DOF Mobile Robot



Extended Kalman Filter $(\hat{x}_{k-1}, P_{k-1}, u_k, z_k)$

$$\hat{x}_k^- = f(\hat{x}_{k-1}, u_k, 0)$$
$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_k W_k^T$$
$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T\right)^{-1}$$
$$\hat{x}_k = \hat{x}_k^- + K_k\left(z_k - h(\hat{x}_k^-, 0)\right)$$
$$P_k = (I - K_k H_k) P_k^-$$
$$return(\hat{x}_k, P_k)$$

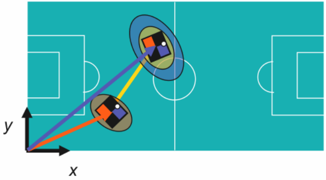


$$\mathbf{x}_{ij} = \begin{pmatrix} x_{ij} \\ y_{ij} \\ \theta_{ij} \end{pmatrix} \quad \mathbf{u}_{\mathbf{jk}} = \begin{pmatrix} x_{jk} \\ y_{jk} \\ \theta_{jk} \end{pmatrix}$$

$\mathbf{x}_{ij}$: Represents an instant robot motion from {i} to {j}

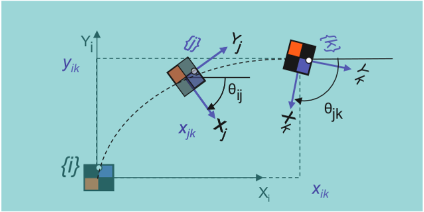
$\mathbf{u}_{\mathbf{jk}}$: Represents an instant robot motion from {j} to {k}

Example: 3 DOF Mobile Robot



Extended Kalman Filter$\left(\hat{x}_{k-1}, P_{k-1}, u_k, z_k\right)$

$$\hat{x}_k^- = f\left(\hat{x}_{k-1}, u_k, 0\right)$$
$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_k W_k^T$$
$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T\right)^{-1}$$
$$\hat{x}_k = \hat{x}_k^- + K_k\left(z_k - h\left(\hat{x}_k^-, 0\right)\right)$$
$$P_k = \left(I - K_k H_k\right) P_k^-$$
$$return\left(\hat{x}_k, P_k\right)$$

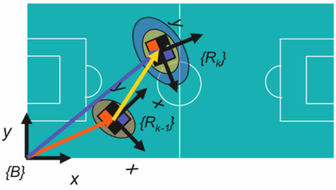


$$\mathbf{x}_{ij} = \begin{pmatrix} x_{ij} \\ y_{ij} \\ \theta_{ij} \end{pmatrix} \quad \mathbf{u}_{\mathbf{jk}} = \begin{pmatrix} x_{jk} \\ y_{jk} \\ \theta_{jk} \end{pmatrix} \quad \mathbf{x}_{ik} = \mathbf{x}_{ij} \oplus \mathbf{u}_{\mathbf{jk}} = \begin{pmatrix} c\theta_{ij}\, x_{jk} - s\theta_{ij}\, y_{jk} + x_{ij} \\ s\theta_{ij}\, x_{jk} + c\theta_{ij}\, y_{jk} + y_{ij} \\ \theta_{ij} + \theta_{jk} \end{pmatrix}$$

$\mathbf{x}_{ij}$: Represents an instant robot motion from {i} to {j}

$\mathbf{u}_{\mathbf{jk}}$: Represents an instant robot motion from {j} to {k}

$\mathbf{x}_{ik} = \mathbf{x}_{ij} \oplus \mathbf{u}_{\mathbf{jk}}$: The compounding function allows to merge 2 consecutive transformations into a unique one



Extended Kalman Filter $\left(\hat{x}_{k-1}, P_{k-1}, u_k, z_k\right)$

$$
\begin{aligned}
&\hat{x}_k^- = f\left(\hat{x}_{k-1}, u_k, 0\right)\\
&P_k^- = A_k P_{k-1} A_k^T + W_k Q_k W_k^T\\
&K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T\right)^{-1}\\
&\hat{x}_k = \hat{x}_k^- + K_k\left(z_k - h\left(\hat{x}_k^-, 0\right)\right)\\
&P_k = \left(I - K_k H_k\right) P_k^-\\
&return\left(\hat{x}_k, P_k\right)
\end{aligned}
$$

In our case …

> The robot pose wrt {B} is compounded with the last displacement (odometry) to get the next pose wrt {B}

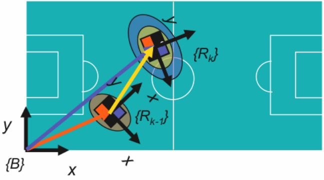
$$
\mathbf{x}_k^B = \overbrace{\begin{pmatrix} x^B_{R_{k-1}} \\ y^B_{R_{k-1}} \\ \theta^B_{R_{k-1}} \end{pmatrix}}^{\mathbf{x}^B_{k-1}} \oplus \left[\overbrace{\begin{pmatrix} x^{R_{k-1}}_{R_k} \\ y^{R_{k-1}}_{R_k} \\ \theta^{R_{k-1}}_{R_k} \end{pmatrix}}^{\mathbf{u}_k^{R_{k-1}}} + \overbrace{\begin{pmatrix} w_{x_k} \\ w_{y_k} \\ w_{\theta_k} \end{pmatrix}}^{\mathbf{w}_k} \right] \quad where \quad \begin{cases} \mathbf{w}_k \equiv N\left(0_{3x1}, Q_k\right) \\ Q_k = diag\left\{ \sigma_{w_x} \quad \sigma_{w_y} \quad \sigma_{w_\theta} \right\} \end{cases}
$$

> The compounding is a nonlinear function depending on 6 parameters, affected now by noise

$$
\mathbf{x}_k^B = \mathbf{x}_{k-1}^B \oplus \left(\mathbf{u}_k^{R_{k-1}} + \mathbf{w}_k\right) = \begin{pmatrix} c\theta^B_{R_{k-1}} \cdot \left(x^{R_{k-1}}_{R_k} + w_{x_{jk}}\right) - s\theta^B_{R_{k-1}} \cdot \left(y^{R_{k-1}}_{R_k} + w_{y_{jk}}\right) + x^B_{R_{k-1}} \\ s\theta^B_{R_{k-1}} \cdot \left(x^{R_{k-1}}_{R_k} + w_{x_{jk}}\right) + c\theta^B_{R_{k-1}} \cdot \left(y^{R_{k-1}}_{R_k} + w_{y_{jk}}\right) + y^B_{R_{k-1}} \\ \theta^B_{R_{k-1}} + \theta^{R_{k-1}}_{R_k} + w_{\theta_k} \end{pmatrix}
$$

> Now the nonlinear process model affected by the noise can be formulated…

$$
f\left(\mathbf{x}_{k-1}^B, \mathbf{u}_k^{R_{k-1}}, \mathbf{w}_k\right) = \mathbf{x}_{k-1}^B \oplus \left(\mathbf{u}_k^{R_{k-1}} + \mathbf{w}_k\right)
$$



Extended Kalman Filter $\left(\hat{x}_{k-1}, P_{k-1}, u_k, z_k\right)$

$$\hat{x}_k^- = f\left(\hat{x}_{k-1}, u_k, 0\right)$$
$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_k W_k^T$$
$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T\right)^{-1}$$
$$\hat{x}_k = \hat{x}_k^- + K_k\left(z_k - h\left(\hat{x}_k^-, 0\right)\right)$$
$$P_k = \left(I - K_k H_k\right) P_k^-$$
$$return\left(\hat{x}_k, P_k\right)$$

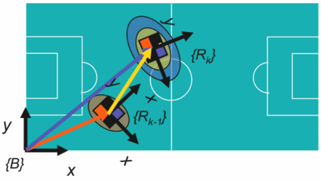
Now A & W can be computed …

$$\mathbf{A}_k = \left.\frac{\partial \mathbf{f}\left(\mathbf{x}_{k-1}^{B}, \mathbf{u}_k^{R_{k-1}}, \mathbf{w}_k\right)}{\partial \mathbf{x}_{k-1}}\right|_{\left(\hat{\mathbf{x}}_k^{B}, \mathbf{u}_k^{R_{k-1}}, 0\right)}$$

$$\mathbf{f}\left(\mathbf{x}_{k-1}^{B}, \mathbf{u}_k^{R_{k-1}}, \mathbf{w}_k\right) = \begin{pmatrix} c\theta_{R_{k-1}}^{B} \cdot \left(x_{R_k}^{R_{k-1}} + w_{x_{jk}}\right) - s\theta_{R_{k-1}}^{B} \cdot \left(y_{R_k}^{R_{k-1}} + w_{y_{jk}}\right) + x_{R_{k-1}}^{B} \\ s\theta_{R_{k-1}}^{B} \cdot \left(x_{R_k}^{R_{k-1}} + w_{x_{jk}}\right) + c\theta_{R_{k-1}}^{B} \cdot \left(y_{R_k}^{R_{k-1}} + w_{y_{jk}}\right) + y_{R_{k-1}}^{B} \\ \theta_{R_{k-1}}^{B} + \theta_{R_k}^{R_{k-1}} + w_{\theta_k} \end{pmatrix}$$

$$\mathbf{A}_k = \frac{\mathbf{f}\left(\mathbf{x}_{k-1}^{B}, \mathbf{u}_k^{R_{k-1}}, \mathbf{w}_k\right)}{\partial\left(\mathbf{x}_{R_{k-1}}^{B}\right)} = \begin{pmatrix} \frac{\partial f_x}{\partial x_{R_{k-1}}^B} & \frac{\partial f_x}{\partial y_{R_{k-1}}^B} & \frac{\partial f_x}{\partial \theta_{R_{k-1}}^B} \\ \frac{\partial f_y}{\partial x_{R_{k-1}}^B} & \frac{\partial f_y}{\partial y_{R_{k-1}}^B} & \frac{\partial f_y}{\partial \theta_{R_{k-1}}^B} \\ \frac{\partial f_\theta}{\partial x_{R_{k-1}}^B} & \frac{\partial f_\theta}{\partial y_{R_{k-1}}^B} & \frac{\partial f_\theta}{\partial \theta_{R_{k-1}}^B} \end{pmatrix} = \begin{pmatrix} 1 & 0 & -s\theta_{R_{k-1}}^{B} \cdot x_{R_k}^{R_{k-1}} - c\theta_{R_{k-1}}^{B} \cdot y_{R_k}^{R_{k-1}} \\ 0 & 1 & c\theta_{R_{k-1}}^{B} \cdot x_{R_k}^{R_{k-1}} - s\theta_{R_{k-1}}^{B} \cdot y_{R_k}^{R_{k-1}} \\ 0 & 0 & 1 \end{pmatrix}$$

Example: 3 DOF Mobile Robot



Extended Kalman Filter$(\hat{x}_{k-1}, P_{k-1}, u_k, z_k)$

$$\begin{aligned}
\hat{x}_k^- &= f(\hat{x}_{k-1}, u_k, 0) \\
P_k^- &= A_k P_{k-1} A_k^T + W_k Q_k W_k^T \\
K_k &= P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T\right)^{-1} \\
\hat{x}_k &= \hat{x}_k^- + K_k\left(z_k - h(\hat{x}_k^-, 0)\right) \\
P_k &= (I - K_k H_k) P_k^- \\
&\mathit{return}(\hat{x}_k, P_k)
\end{aligned}$$

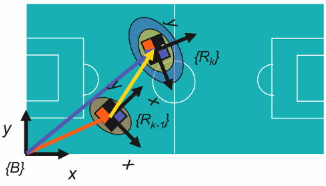
Now A & W can be computed …

$$\mathbf{W}_k = \left.\frac{\mathbf{f}\left(\mathbf{x}^B_{k-1}, \mathbf{u}^{R_{k-1}}_k, \mathbf{w}_k\right)}{\partial \mathbf{w}_k}\right|_{\left(\hat{\mathbf{x}}^B_k, \mathbf{u}^{R_{k-1}}_k, 0\right)}$$

$$\mathbf{f}\left(\mathbf{x}^B_{k-1}, \mathbf{u}^{R_{k-1}}_k, \mathbf{w}_k\right) = \begin{pmatrix} c\theta^B_{R_{k-1}} \cdot \left(x^{R_{k-1}}_{R_k} + w_{x_{jk}}\right) - s\theta^B_{R_{k-1}} \cdot \left(y^{R_{k-1}}_{R_k} + w_{y_{jk}}\right) + x^B_{R_{k-1}} \\ s\theta^B_{R_{k-1}} \cdot \left(x^{R_{k-1}}_{R_k} + w_{x_{jk}}\right) + c\theta^B_{R_{k-1}} \cdot \left(y^{R_{k-1}}_{R_k} + w_{y_{jk}}\right) + y^B_{R_{k-1}} \\ \theta^B_{R_{k-1}} + \theta^{R_{k-1}}_{R_k} + w_{\theta_k} \end{pmatrix}$$

$$\mathbf{W}_k = \frac{\mathbf{f}\left(\mathbf{x}^B_{k-1}, \mathbf{u}^{R_{k-1}}_k, \mathbf{w}_k\right)}{\partial\left(\mathbf{w}_k\right)} = \begin{pmatrix} \frac{\partial f_x}{\partial w_x} & \frac{\partial f_x}{\partial w_y} & \frac{\partial f_x}{\partial w_\theta} \\ \frac{\partial f_y}{\partial w_x} & \frac{\partial f_y}{\partial w_y} & \frac{\partial f_y}{\partial w_\theta} \\ \frac{\partial f_\theta}{\partial w_x} & \frac{\partial f_\theta}{\partial w_y} & \frac{\partial f_\theta}{\partial w_\theta} \end{pmatrix} = \begin{pmatrix} c\theta^B_{R_{k-1}} & -s\theta^B_{R_{k-1}} & 0 \\ s\theta^B_{R_{k-1}} & c\theta^B_{R_{k-1}} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Example: 3 DOF Mobile Robot

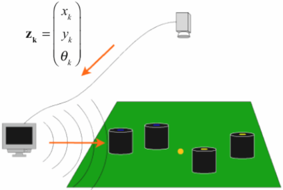


Extended Kalman Filter $\left(\hat{x}_{k-1}, P_{k-1}, u_k, z_k\right)$

$$\hat{x}_k^- = f\left(\hat{x}_{k-1}, u_k, 0\right)$$
$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_k W_k^T$$
$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + V_k R_k V_k^T\right)^{-1}$$
$$\hat{x}_k = \hat{x}_k^- + K_k\left(z_k - h\left(\hat{x}_k^-, 0\right)\right)$$
$$P_k = \left(I - K_k H_k\right) P_k^-$$
$$return\left(\hat{x}_k, P_k\right)$$

What about the measurement equation?

$$\mathbf{z_k} = \begin{pmatrix} x_k \\ y_k \\ \theta_k \end{pmatrix}$$



> In this case the measurement equation can be formulated as a linear function $z_k = H_k x_k + v_k$

$$\mathbf{z}_{R_k}^B = \overbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}^{\mathbf{H_k}} \overbrace{\begin{pmatrix} x_{R_k}^B \\ y_{R_k}^B \\ \theta_{R_k}^B \end{pmatrix}}^{\mathbf{x}_{R_k}^B} + \overbrace{\begin{pmatrix} v_{x_k} \\ v_{y_k} \\ v_{\theta_k} \end{pmatrix}}^{\mathbf{v_k}} \quad where \begin{cases} \mathbf{v_k} \equiv N\left(\mathbf{0_{3x1}}, \mathbf{R_k}\right) \\ R_k = diag\left\{\sigma_{v_x} \quad \sigma_{v_y} \quad \sigma_{v_\theta}\right\} \end{cases}$$

> Hence, $V_k = I_{3x3}$

Questions?