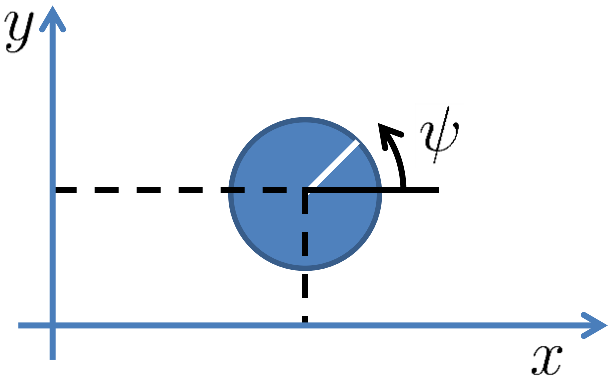
Autonomous Navigation for Flying Robots

Lecture 6.3: EKF Example

Jürgen Sturm
Technische Universität München

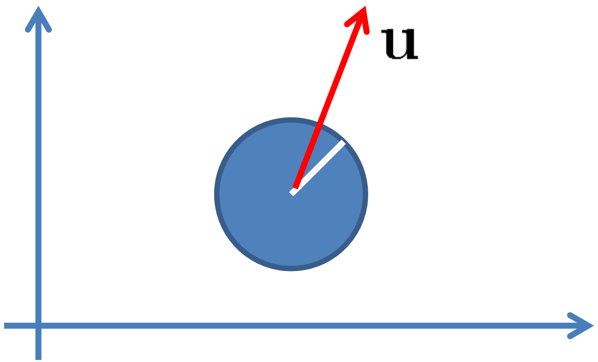
Example in 2D

- State $\mathbf{x} = \begin{pmatrix} x & y & \psi \end{pmatrix}^\top$



Example in 2D

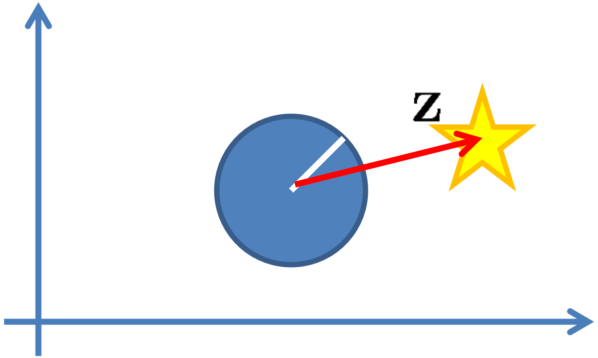
- State $\mathbf{x} = \begin{pmatrix} x & y & \psi \end{pmatrix}^\top$
- Odometry $\mathbf{u} = \begin{pmatrix} \dot{x} & \dot{y} & \dot{\psi} \end{pmatrix}^\top$



Example in 2D

- State $\mathbf{x} = \begin{pmatrix} x_x & y_x & \psi_x \end{pmatrix}^\top$ in global coordinate frame
- Odometry $\mathbf{u} = \begin{pmatrix} \dot{x}_u & \dot{y}_u & \dot{\psi}_u \end{pmatrix}^\top$
- Observations of visual marker $\mathbf{z} = \begin{pmatrix} x_z & y_z & \psi_z \end{pmatrix}^\top$

} in local frame!



See lecture 2.3 for more details on coordinate transforms

Motion Model

- Motion function

$$g(\mathbf{x}, \mathbf{u}) = \begin{pmatrix} x + (\cos(\psi)\dot{x} - \sin(\psi)\dot{y})\Delta t \\ y + (\sin(\psi)\dot{x} + \cos(\psi)\dot{y})\Delta t \\ \psi + \dot{\psi}\Delta t \end{pmatrix}$$

- Derivative of motion function

$$\mathbf{G} = \frac{\partial g(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} = \begin{pmatrix} 1 & 0 & (-\sin(\psi)\dot{x} - \cos(\psi)\dot{y})\Delta t \\ 0 & 1 & (\cos(\psi)\dot{x} + \sin(\dot{\psi})\dot{y})\Delta t \\ 0 & 0 & 1 \end{pmatrix}$$

Sensor Model

- Let's construct the sensor model
- The marker is located at $\mathbf{m} = \begin{pmatrix} x_m & y_m & \psi_m \end{pmatrix}^\top$ (given in global/world coordinates)
- We need to compute $\mathbf{z} = h(\mathbf{x})$ where $\mathbf{z}$ is the pose of the marker **relative** to the robot!

Sensor Model

- Transformation matrix corresponding to (global) robot pose

$$\mathbf{X} = \begin{pmatrix} \cos\psi_x & -\sin\psi_x & x_x \\ \sin\psi_x & \cos\psi_x & y_x \\ 0 & 0 & 1 \end{pmatrix}$$

- Relation between global and local coordinates

$$\tilde{\mathbf{t}}_{\text{global}} = \mathbf{X}\tilde{\mathbf{t}}_{\text{local}}$$

$$\tilde{\mathbf{t}}_{\text{local}} = \mathbf{X}^{-1}\tilde{\mathbf{t}}_{\text{global}}$$

Sensor Model

- Finally, we get

$$h(\mathbf{x}) = \begin{pmatrix} (x_g - x_x)\cos\psi_x + (y_g - y_x)\sin\psi_x \\ -(x_g - x_x)\sin\psi_x + (y_g - y_x)\cos\psi_x \\ \psi_g - \psi_x \end{pmatrix}$$

Sensor Model

- Now derive the observation function with respect to all components of its argument

$$H = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} = \begin{pmatrix} \frac{\partial h(\mathbf{x})}{\partial x_x} & \frac{\partial h(\mathbf{x})}{\partial y_x} & \frac{\partial h(\mathbf{x})}{\partial \psi_x} \end{pmatrix}$$

$$= \begin{pmatrix} -\cos\psi_x & -\sin\psi_x & -(x_g - x_x)\sin\psi_x + (y_g - y_x)\cos\psi_x \\ \sin\psi_x & -\cos\psi_x & -(x_g - x_x)\cos\psi_x - (y_g - y_x)\sin\psi_x \\ 0 & 0 & -1 \end{pmatrix}$$

- That's it!

Extended Kalman Filter (EKF)

For each time step, do

1. Apply motion model (prediction step)

$$\bar{\boldsymbol{\mu}}_t = g(\boldsymbol{\mu}_{t-1}, \mathbf{u}_t)$$

$$\bar{\boldsymbol{\Sigma}}_t = \mathbf{G}_t \boldsymbol{\Sigma} \mathbf{G}_t^\top + \mathbf{Q} \quad \text{with } \mathbf{G}_t = \left.\frac{\partial g(\mathbf{x}, \mathbf{u}_t)}{\partial \mathbf{x}}\right|_{\mathbf{x}=\boldsymbol{\mu}_{t-1}}$$

2. Apply sensor model (correction step)

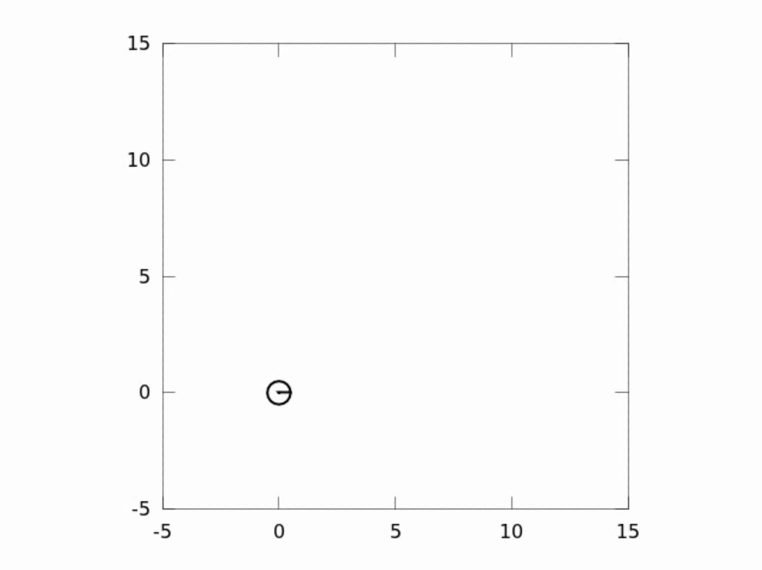
$$\boldsymbol{\mu}_t = \bar{\boldsymbol{\mu}}_t + \mathbf{K}_t(\mathbf{z}_t - h(\bar{\boldsymbol{\mu}}_t))$$

$$\boldsymbol{\Sigma}_t = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t)\bar{\boldsymbol{\Sigma}}_t$$

with $\mathbf{K}_t = \bar{\boldsymbol{\Sigma}}_t \mathbf{H}_t^\top (\mathbf{H}_t \bar{\boldsymbol{\Sigma}}_t \mathbf{H}_t^\top + \mathbf{R})^{-1}$ and $\mathbf{H}_t = \left.\frac{\partial h(\mathbf{x}, \mathbf{u}_t)}{\partial \mathbf{x}}\right|_{\mathbf{x}=\bar{\boldsymbol{\mu}}_t}$

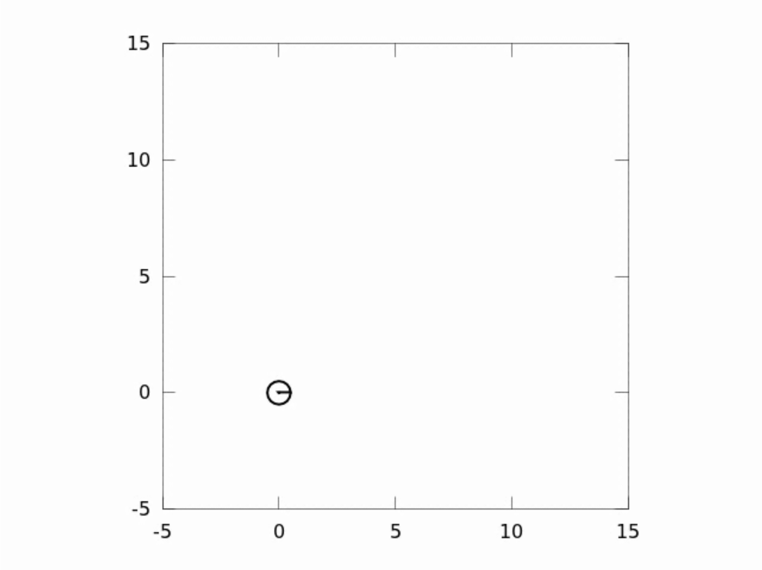
2D EKF Example

- Dead reckoning (no observations)
- Large process noise Q in x+y



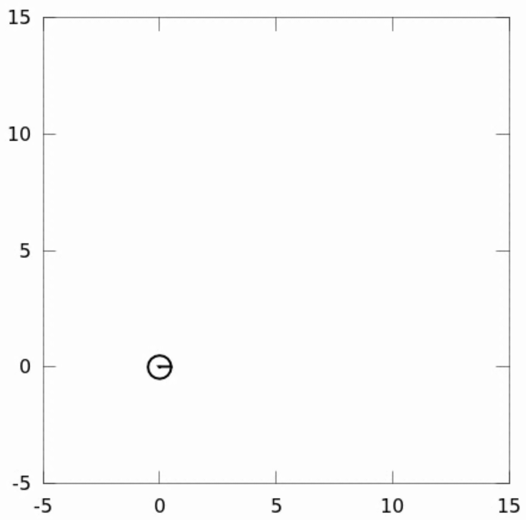
2D EKF Example

- Dead reckoning (no observations)
- Large process noise Q in x+y+yaw



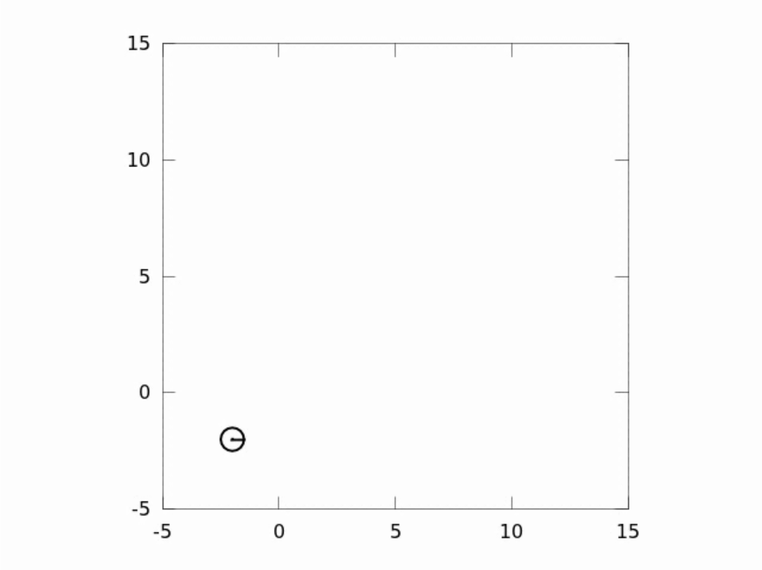
2D EKF Example

- Now with observations (limited visibility)
- Assume robot knows correct starting pose



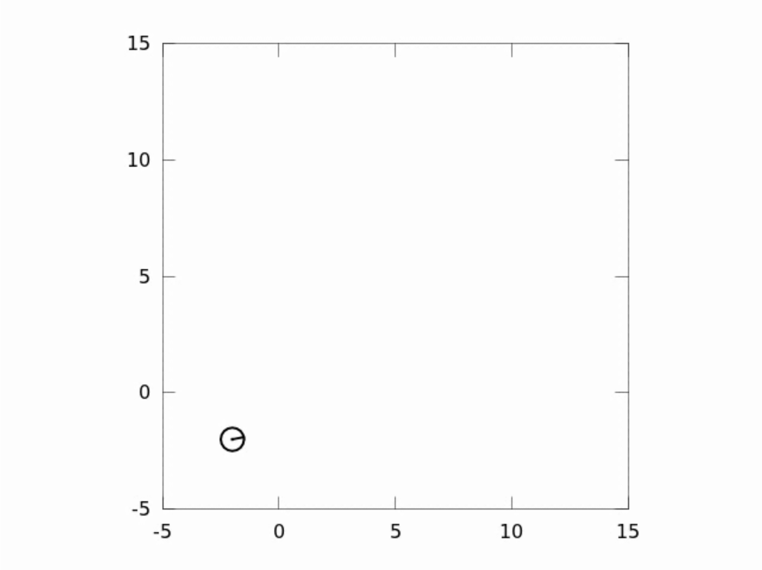
2D EKF Example

What if the initial pose (x+y) is wrong?



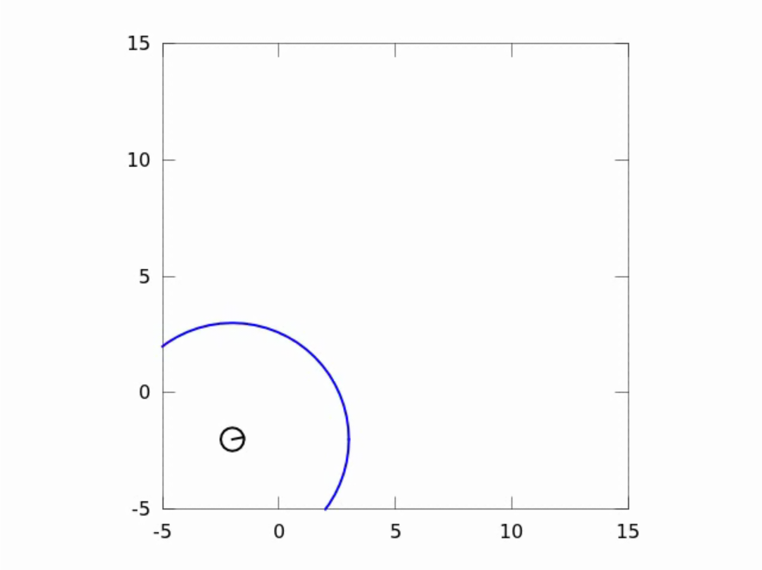
2D EKF Example

What if the initial pose (x+y+yaw) is wrong?

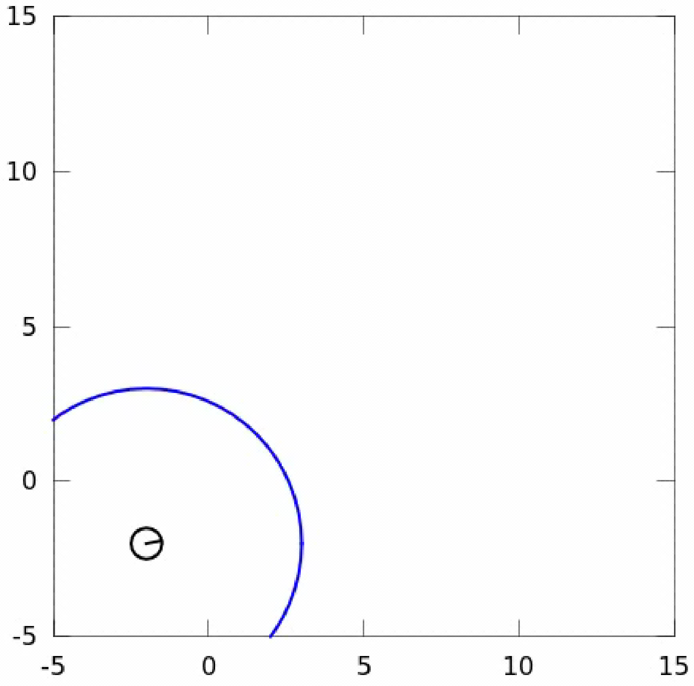


2D EKF Example

If we are aware of a bad initial guess, we set the initial sigma to a large value (large uncertainty)



2D EKF Example



Lessons Learned

- 2D example of an EKF
- Derivation of motion model
- Derivation of sensor model
- Several example runs