SLAM For Autonomous Ground Vehicles

Attached 1

Li Hong Rong

Overall

- Research Motivation
- Literature Reviews
- Research Approaches
- Progress
- Application

Research motivation

1. REDUCED ACCIDENTS

   Self-driving cars are projected to reduce traffic deaths by 90 percent, saving 30,000 lives a year

2. REDUCED TRAFFIC CONGESTION

   "Our experiments show that with as few as 5 percent of vehicles being automated and carefully controlled, we can eliminate stop-and-go waves caused by human driving behavior," said Daniel B, a lead researcher in the traffic congestion study.

3. REDUCED CO2 EMISSIONS

   The reduction in congestion will most likely result in a reduction of CO2 emissions as well.

4. TRANSPORTATION ACCESSIBILITY

   The US House Energy and Commerce Committee website adds: "With self-driving cars, tasks like commuting to work, going to the doctor, and visiting family across town could become easier for seniors and those with disabilities."

5. REDUCED TRAVEL TIME AND TRANSPORTATION COSTS

   AVs may cut travel time by up to 40 percent, recover up to 80 billion hours lost to commuting and congestion, and reduce fuel consumption by up to 40 percent

Literature Reviews

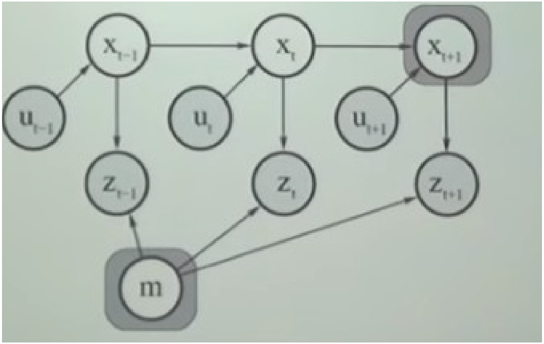
- https://www.youtube.com/playlist?
  listP̄LgnQpQtFTOGQrZ4O5QzbIHgI3b1JHimN_
- An implementation of SLAM with extended Kalman filter
  Abu Bakar Sayuti H M Saman ; Ahmed Hesham Lotfy
- 3 Dimensional application of SLAM for ground navigation
  Nak Yong Ko ; Tae Gyun Kim ; Wonkeun Youn ; Taesik Kim

Research Approaches

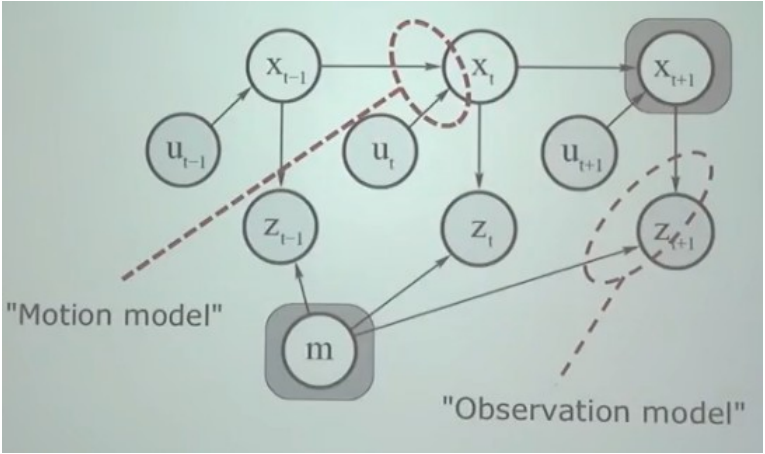
- State Esitimation
- Localization
- Mapping
- SLAM
- Navigation
- Motion Planning
- Control

Graphical Model for Online SLAM

$$p(x_t, m | z_{1:t}, u_{1:t})$$

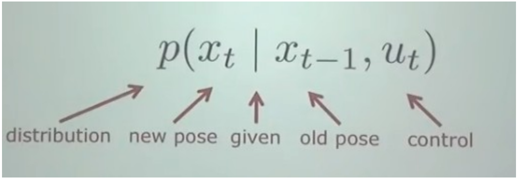


Motion and Observation model



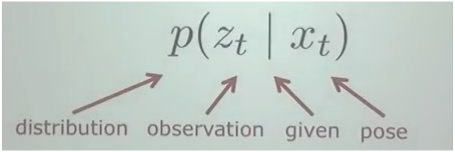
Motion Model

- The motion model describes the relative motion of the robot



Observation Model

- The observation or sensor model relates measurements with the robot's pose



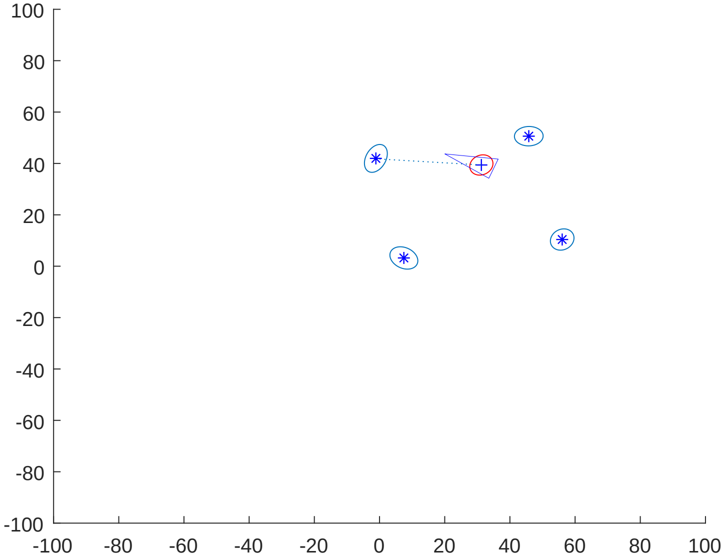
Three Main SLAM Paradigms

Kalman filter	Particle filter	Graph-based filter

EKF SLAM: Filter Cycle

1. State prediction
2. Measurement prediction
3. Measurement
4. Data association
5. Update

EKF SLAM Simulation



Goal and Application

I can build a small autonomous car in the specific space, and help me with cleaning.

Robot Mapping

EKF SLAM

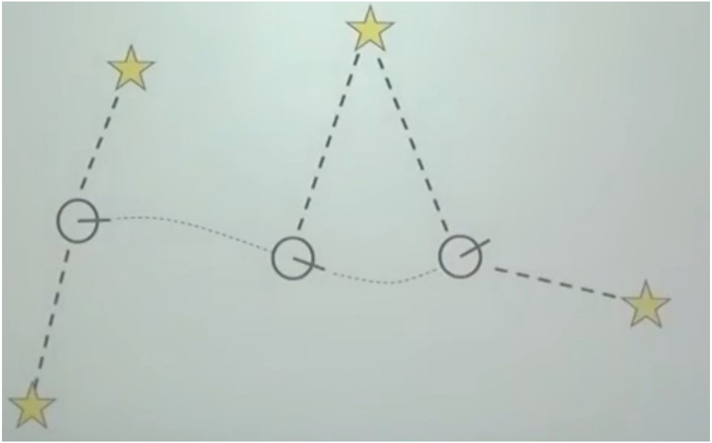
Li Hong Rong

What is SLAM

- Computing the robot's poses and the map of the environment at the same time
- **Localization**: estimating the robot's location
- **Mapping**: building a map
- **SLAM**: building a map and localizing the robot simultaneously

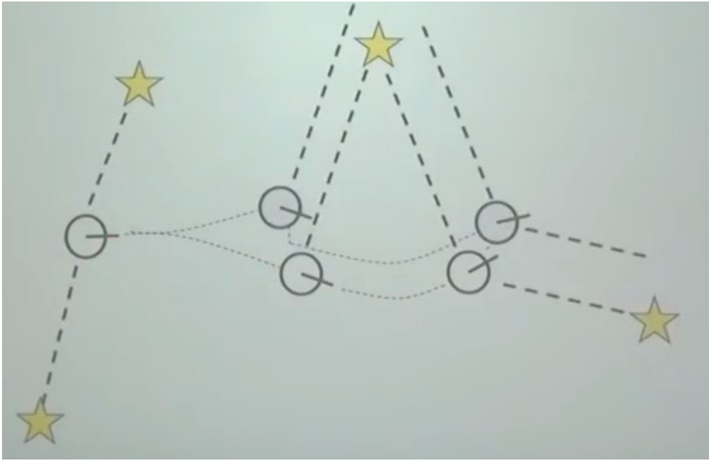
Localization Example

- Estimate the robot's poses given landmarks



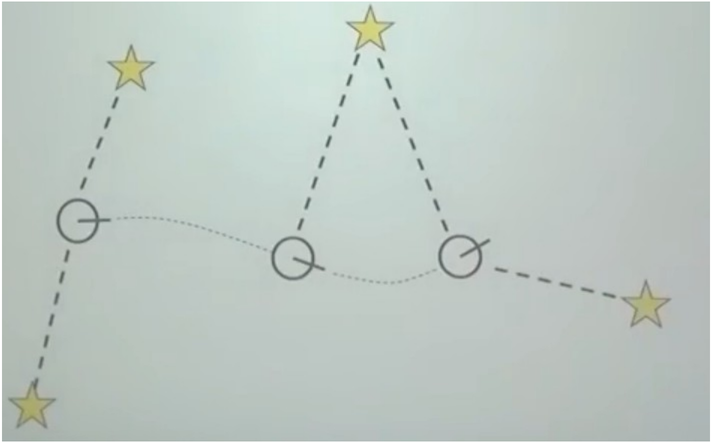
Localization Example

- Estimate the robot's poses given landmarks



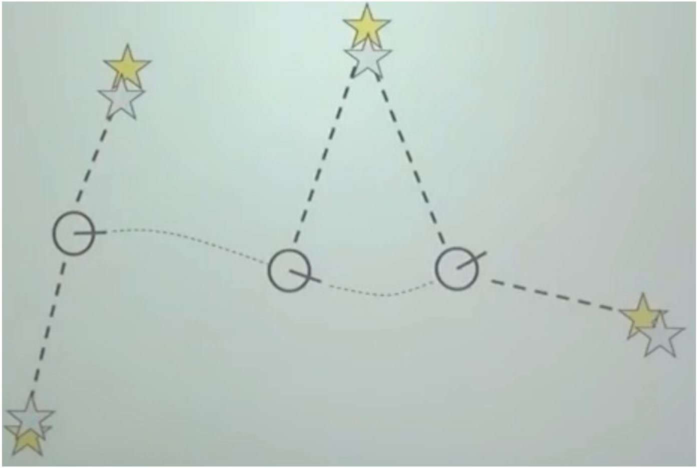
Mapping Example

- Estimate the landmarks given robot's poses



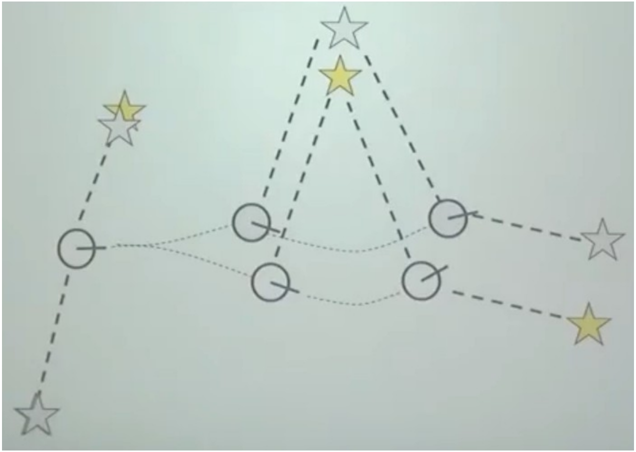
Mapping Example

- Estimate the landmarks given robot's poses



SLAM Example

- Estimate the robot's poses and the landmarks at the same time



- Building a map and locating the robot in the map at the same time
- Chicken-or-egg problem
  $\rightarrow$ a map is needed for localization and
  $\rightarrow$ a pose estimate is needed for mapping

Definition of the SLAM Problem

Given

- The robot's controls

$$u_{1:T} = \{u_1, u_2, ..., u_T\}$$

- Observations

$$z_{1:T} = \{z_1, z_2, z_3, ..., z_T\}$$

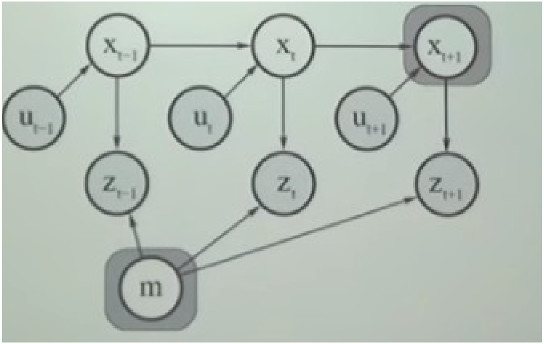
Wanted

- Map of the environment
  m
- Path of the robot

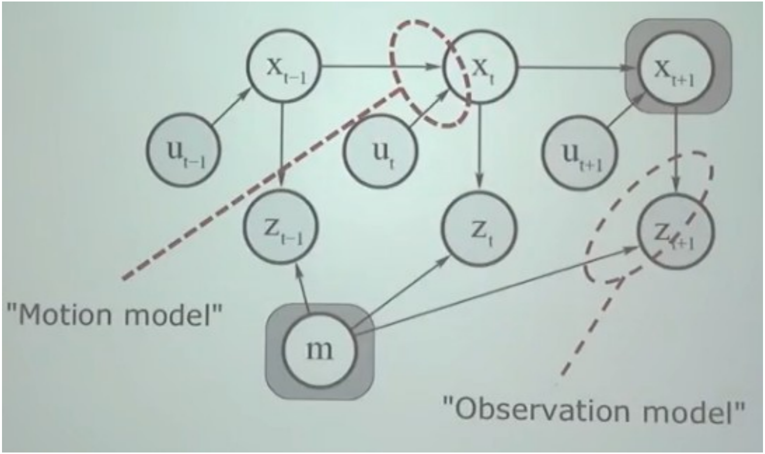
$$x_{0:T} = \{x_0, x_1, ..., x_T\}$$

Graphical Model for Online SLAM

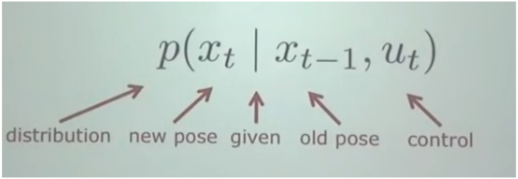
$$p(x_t, m | z_{1:t}, u_{1:t})$$



Motion and Observation model

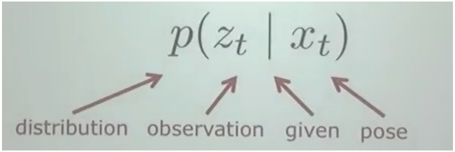


- The motion model describes the relative motion of the robot



Observation Model

- The observation or sensor model relates measurements with the robot's pose



Linear Model

- The Kalman filter assumes a linear transition and observation model
- Zero mean Gaussian noise

$$
\begin{aligned}
x_t &= A_t x_{t-1} + B_t u_t + \epsilon_t && \text{motion model} \\
z_t &= C_t x_t + \delta_t && \text{observation model}
\end{aligned}
$$

Components of a Kalman Filter

A_t Matrix ($n \times n$) that describes how the state evolves from $t-1$ to t without controls or noise

B_t Matrix ($n \times l$) that describes how the control u_t changes the state from $t-1$ to t

C_t Matrix ($k \times n$) that describes how to map the state x_t to an observation z_t

ϵ_t Random variables representing the process and measurement
δ_t noise that are assumed to be independent and normally distributed with covariance R_t and Q_t respectively

Kalman Filter Algorithm

Kalman_filter($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

$\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$

$\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t$

$K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + Q_t)^{-1}$

$\mu_t = \bar{\mu}_t + K_t(z_t - C_t \bar{\mu}_t)$

$\Sigma_t = (I - K_t C_t)\bar{\Sigma}_t$

return μ_t, Σ_t

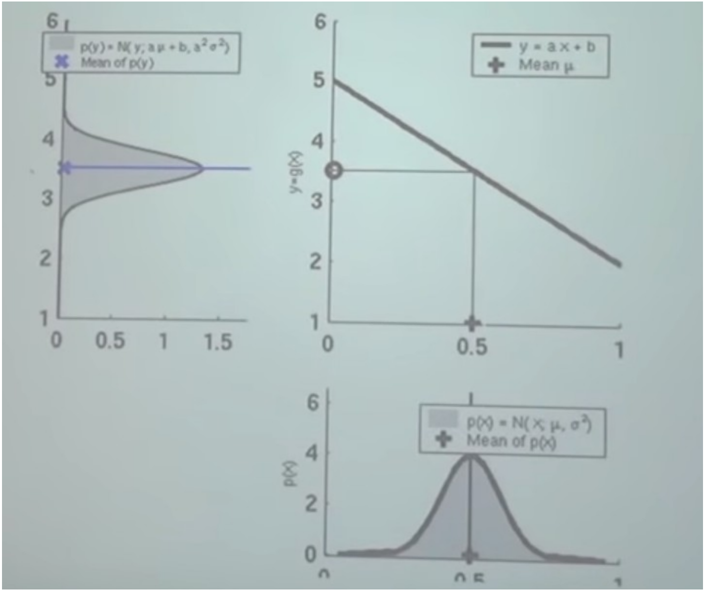
Basically computes a weighted mean between the prediction and the observation

Non-linear Dynamics Systems

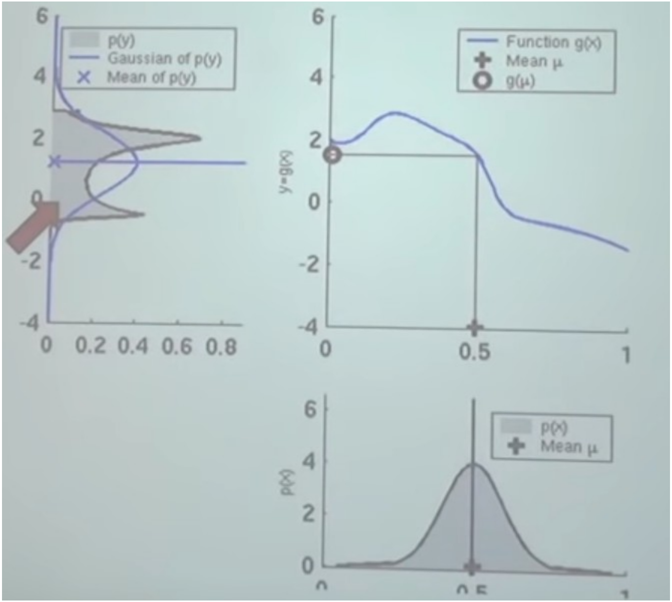
- Most realistic problems (in robotics) involve nonlinear function

$$x_t = g(u_t, x_{t-1}) + \epsilon_t$$
$$z_t = h(x_t) + \delta_t$$

Linearity Assumption Revisited



Non-Linear Function



EKF Linearization: First Order Taylor Expansion

- Prediction:

$$g(u_t, x_{t-1}) \approx g(u_t, \mu_{t-1}) + \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}}(x_{t-1} - \mu_{t-1})$$

- Correction:

$$h(x_t) \approx h(\bar{\mu}_t) + \frac{\partial h(\bar{\mu}_t)}{\partial x_t}(x_t - \bar{\mu}_t)$$

where $\frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}} := G_t$

$\frac{\partial h(\bar{\mu}_t)}{\partial x_t} := H_t$

Extended Kalman Filter Algorithm

Kalman_filter($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

$\bar{\mu}_t = g(u_t, \mu_{t-1})$

$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t$

$K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1}$

$\mu_t = \bar{\mu}_t + K_t(z_t - h(\bar{\mu}_t))$

$\Sigma_t = (I - K_t H_t)\bar{\Sigma}_t$

return μ_t, Σ_t

EKF SLAM

- Application of the EKF to SLAM
- Estimate robot's pose and locations of landmarks in the environment
- Assumption: known correspondences
- State space (for the 2D plane) is

$$x_t = (\underbrace{x, y, \theta}_{robot' spose}, \underbrace{m_{1,x}, m_{1,y}}_{landmark\ 1}, \cdots, \underbrace{m_{n,x}, m_{n,y}}_{landmark\ n})^T$$

EKF: State Representation

- Map with n landmarks: $(3+2n)$-dimensional Gaussian
- Belief is represented by

$$\underbrace{\begin{pmatrix} x \\ y \\ \theta \\ m_{1,x} \\ m_{1,y} \\ \vdots \\ m_{n,x} \\ m_{n,y} \end{pmatrix}}_{\mu} \underbrace{\begin{pmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{x\theta} & \sigma_{xm_{1,x}} & \sigma_{xm_{1,y}} & \cdots & \sigma_{xm_{n,x}} & \sigma_{xm_{n,y}} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{y\theta} & \sigma_{ym_{1,x}} & \sigma_{ym_{1,y}} & \cdots & \sigma_{ym_{n,x}} & \sigma_{ym_{n,y}} \\ \sigma_{\theta x} & \sigma_{\theta y} & \sigma_{\theta\theta} & \sigma_{\theta m_{1,x}} & \sigma_{\theta m_{1,y}} & \cdots & \sigma_{\theta m_{n,x}} & \sigma_{\theta m_{n,y}} \\ \sigma_{m_{1,x}x} & \sigma_{m_{1,x}y} & \sigma_{m_{1,x}\theta} & \sigma_{m_{1,x}m_{1,x}} & \sigma_{m_{1,x}m_{1,y}} & \cdots & \sigma_{m_{1,x}m_{n,x}} & \sigma_{m_{1,x}m_{n,y}} \\ \sigma_{m_{1,y}x} & \sigma_{m_{1,y}y} & \sigma_{m_{1,y}\theta} & \sigma_{m_{1,y}m_{1,y}} & \sigma_{m_{1,y}m_{1,y}} & \cdots & \sigma_{m_{1,y}m_{n,x}} & \sigma_{m_{1,y}m_{n,y}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \sigma_{m_{n,x}x} & \sigma_{m_{n,x}y} & \sigma_{m_{n,x}\theta} & \sigma_{m_{n,x}m_{1,x}} & \sigma_{m_{n,x}m_{1,y}} & \cdots & \sigma_{m_{n,x}m_{n,x}} & \sigma_{m_{n,x}m_{n,y}} \\ \sigma_{m_{n,y}x} & \sigma_{m_{n,y}y} & \sigma_{m_{n,y}\theta} & \sigma_{m_{n,y}m_{1,y}} & \sigma_{m_{n,y}m_{1,y}} & \cdots & \sigma_{m_{n,y}m_{n,x}} & \sigma_{m_{n,y}m_{n,y}} \end{pmatrix}}_{\Sigma}$$

- Even more compactly

$$\underbrace{\begin{pmatrix} x \\ m \end{pmatrix}}_{\mu} \underbrace{\begin{pmatrix} \Sigma_{xx} & \Sigma_{xm} \\ \Sigma_{mx} & \Sigma_{mm} \end{pmatrix}}_{\Sigma}$$

EKF SLAM: Filter Cycle

1. State prediction
2. Measurement prediction
3. Measurement
4. Data association
5. Update

Initialization

- Robots starts in its own reference frame (all landmarks unknown)
- $2N + 3$ dimensions

$$\mu_0 = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \end{pmatrix}^T$$

$$\Sigma_0 = \begin{pmatrix} 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \infty & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \infty \end{pmatrix}$$

Extended Kalman Filter Algorithm

Kalman_filter($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

$\bar{\mu}_t = g(u_t, \mu_{t-1})$

$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t$

$K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1}$

$\mu_t = \bar{\mu}_t + K_t(z_t - h(\bar{\mu}_t))$

$\Sigma_t = (I - K_t H_t)\bar{\Sigma}_t$

return μ_t, Σ_t

Prediction Step (Motion)

- Goal: Update state space based on the robot's motion
- Robot motion in the plane

$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \underbrace{\begin{pmatrix} x \\ y \\ \theta \end{pmatrix} + \begin{pmatrix} -\frac{v_t}{\omega_t} sin\theta + \frac{v_t}{\omega_t} sin(\theta + \omega_t \Delta_t) \\ \frac{v_t}{\omega_t} cos\theta - \frac{v_t}{\omega_t} cos(\theta + \omega_t \Delta_t) \\ \omega_t \Delta_t \end{pmatrix}}_{g_{x,y,\theta}(u_t, (x,y,\theta)^T)}$$

- How to map that to the $2N + 3$ dim space

Update the State Space

- From the motion in the plane

$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} + \begin{pmatrix} -\frac{v_t}{\omega_t} sin\theta + \frac{v_t}{\omega_t} sin(\theta + \omega_t \Delta_t) \\ \frac{v_t}{\omega_t} cos\theta - \frac{v_t}{\omega_t} cos(\theta + \omega_t \Delta_t) \\ \omega_t \Delta_t \end{pmatrix}$$

- to the $2N + 3$ dimensional space

$$\begin{pmatrix} x' \\ y' \\ \theta' \\ \vdots \end{pmatrix} = \underbrace{\begin{pmatrix} x \\ y \\ \theta \\ \vdots \end{pmatrix} + \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \underbrace{\cdots}_{2Ncols} & 0 \end{pmatrix}^T}_{F_x^T} \begin{pmatrix} -\frac{v_t}{\omega_t} sin\theta + \frac{v_t}{\omega_t} sin(\theta + \omega_t \Delta_t) \\ \frac{v_t}{\omega_t} cos\theta - \frac{v_t}{\omega_t} cos(\theta + \omega_t \Delta_t) \\ \omega_t \Delta_t \end{pmatrix}}_{g(u_t, x_t)}$$

Update Covariance

- The function g only affects the robot's motion and not the landmarks

$$G_t = \begin{pmatrix} \overset{\text{Jacobian of the motion } (3 \times 3)}{G_t^x} & 0 \\ 0 & \overset{\text{Identity } (2N \times 2N)}{I} \end{pmatrix}$$

Jacobian of the Motion

$$
\begin{aligned}
G_t^x &= \frac{\partial}{\partial (x,y,\theta)^T}\left[\begin{pmatrix} x \\ y \\ \theta \end{pmatrix} + \begin{pmatrix} -\frac{v_t}{\omega_t} sin\theta + \frac{v_t}{\omega_t} sin(\theta + \omega_t \Delta_t) \\ \frac{v_t}{\omega_t} cos\theta - \frac{v_t}{\omega_t} cos(\theta + \omega_t \Delta_t) \\ \omega_t \Delta_t \end{pmatrix}\right] \\
&= I + \frac{\partial}{\partial (x,y,\theta)^T} \begin{pmatrix} -\frac{v_t}{\omega_t} sin\theta + \frac{v_t}{\omega_t} sin(\theta + \omega_t \Delta_t) \\ \frac{v_t}{\omega_t} cos\theta - \frac{v_t}{\omega_t} cos(\theta + \omega_t \Delta_t) \\ \omega_t \Delta_t \end{pmatrix} \\
&= I + \begin{pmatrix} 0 & 0 & -\frac{v_t}{\omega_t} sin\theta + \frac{v_t}{\omega_t} sin(\theta + \omega_t \Delta_t) \\ 0 & 0 & \frac{v_t}{\omega_t} cos\theta - \frac{v_t}{\omega_t} cos(\theta + \omega_t \Delta_t) \\ 0 & 0 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 1 & 0 & -\frac{v_t}{\omega_t} sin\theta + \frac{v_t}{\omega_t} sin(\theta + \omega_t \Delta_t) \\ 0 & 1 & \frac{v_t}{\omega_t} cos\theta - \frac{v_t}{\omega_t} cos(\theta + \omega_t \Delta_t) \\ 0 & 0 & 1 \end{pmatrix}
\end{aligned}
$$

This Leads to the Update

$$
\begin{aligned}
\bar{\Sigma}_t &= G_t \Sigma_{t-1} G_t^T + R_t \\
&= \begin{pmatrix} G_t^x & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} \Sigma_{xx} & \Sigma_{xm} \\ \Sigma_{mx} & \Sigma_{mm} \end{pmatrix} \begin{pmatrix} (G_t^x)^T & 0 \\ 0 & I \end{pmatrix} + R_t \\
&= \begin{pmatrix} (G_t^x)\Sigma_{xx}(G_t^x)^T & G_t^x \Sigma_{xm} \\ (G_t^x \Sigma_{xm})^T & \Sigma_{mm} \end{pmatrix} + R_t
\end{aligned}
$$

Prediction Step

EKF_SLAM_Prediction$(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t, c_t, R_t)$

EKF_SLAM_Prediction$(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t, c_t, R_t)$**:**

2: $$F_x = \begin{pmatrix} 1 & 0 & 0 & 0 \cdots 0 \\ 0 & 1 & 0 & 0 \cdots 0 \\ 0 & 0 & 1 & 0 \cdots 0 \end{pmatrix}$$

3: $$\bar{\mu}_t = \mu_{t-1} + F_x^T \begin{pmatrix} -\frac{v_t}{\omega_t} \sin \mu_{t-1,\theta} + \frac{v_t}{\omega_t} \sin(\mu_{t-1,\theta} + \omega_t \Delta t) \\ \frac{v_t}{\omega_t} \cos \mu_{t-1,\theta} - \frac{v_t}{\omega_t} \cos(\mu_{t-1,\theta} + \omega_t \Delta t) \\ \omega_t \Delta t \end{pmatrix}$$

4: $$G_t = I + F_x^T \begin{pmatrix} 0 & 0 & -\frac{v_t}{\omega_t} \cos \mu_{t-1,\theta} + \frac{v_t}{\omega_t} \cos(\mu_{t-1,\theta} + \omega_t \Delta t) \\ 0 & 0 & -\frac{v_t}{\omega_t} \sin \mu_{t-1,\theta} + \frac{v_t}{\omega_t} \sin(\mu_{t-1,\theta} + \omega_t \Delta t) \\ 0 & 0 & 0 \end{pmatrix} F_x$$

5: $$\bar{\Sigma}_t = G_t \, \Sigma_{t-1} \, G_t^T + \underbrace{F_x^T \, R_t^x \, F_x}_{R_t}$$

Range-Bearing Observation

- Range-Bearing observation $z_t^i = (r_t^i, \phi_t^i)^T$
- If the landmark has not been observed

$$\underbrace{\begin{pmatrix} \bar{\mu}_{j,x} \\ \bar{\mu}_{j,y} \end{pmatrix}}_{\text{observed location of landmark j}} = \underbrace{\begin{pmatrix} \bar{\mu}_{t,x} \\ \bar{\mu}_{t,y} \end{pmatrix}}_{\text{estimated robot's location}} + \underbrace{\begin{pmatrix} r_t^i cos(\phi_t^i + \bar{\mu}_{t,\theta}) \\ r_t^i sin(\phi_t^i + \bar{\mu}_{t,\theta}) \end{pmatrix}}_{\text{relative measurement}}$$

Expected Observation

- Compute expected observation according to the current estimate

$$
\begin{aligned}
\delta &= \begin{pmatrix} \delta_x \\ \delta_y \end{pmatrix} = \begin{pmatrix} \bar{\mu}_{j,x} - \bar{\mu}_{t,x} \\ \bar{\mu}_{j,y} - \bar{\mu}_{t,y} \end{pmatrix} \\
q &= \delta^T \delta \\
\hat{z}_t^i &= \begin{pmatrix} \sqrt{q} \\ atan2(\delta_y, \delta_x) - \bar{\mu}_{t,\theta} \end{pmatrix} \\
&= h(\bar{\mu}_t)
\end{aligned}
$$

Jacobian for the Observation

- Compute the Jacobian

$$
\begin{aligned}
{}_{\text{low}}H_t^i &= \frac{\partial h(\bar{\mu}_t)}{\partial \bar{\mu}_t} \\
&= \begin{pmatrix} \frac{\partial \sqrt{q}}{\partial x} & \frac{\partial \sqrt{q}}{\partial y} & \cdots \\ \frac{atan2(\cdots)}{\partial x} & \frac{atan2(\cdots)}{\partial y} & \cdots \end{pmatrix}
\end{aligned}
$$

low-dim space $(x, y, \theta, m_{j,x}, m_{j,y})$

- We obtain the first component(by applying the chain rule)

$$
\begin{aligned}
\frac{\partial \sqrt{q}}{\partial x} &= \frac{1}{2}\frac{1}{\sqrt{q}} 2\delta_x(-1) \\
&= \frac{1}{q}(-\sqrt{q}\delta_x)
\end{aligned}
$$

Jacobian for the Observation

- Compute the Jacobian

$$
\begin{aligned}
{}_{\text{low}}H_t^i &= \frac{\partial h(\bar{\mu}_t)}{\partial \bar{\mu}_t} \\
&= \frac{1}{q}\begin{pmatrix} -\sqrt{q}\delta_x & -\sqrt{q}\delta_y & 0 & \sqrt{q}\delta_x & \sqrt{q}\delta_y \\ \delta_y & -\delta_x & -q & -\delta_y & \delta_x \end{pmatrix}
\end{aligned}
$$

- map it to the high dimensional space

$$
H_t^i = {}_{\text{low}}H_t^i F_{x,j}
$$

$$
F_{x,j} = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 0 & \cdots & 0 \end{pmatrix}
$$

Next Steps as Specified

Kalman_filter($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

$\bar{\mu}_t = g(u_t, \mu_{t-1})$

$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t$

$K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1}$

$\mu_t = \bar{\mu}_t + K_t(z_t - h(\bar{\mu}_t))$

$\Sigma_t = (I - K_t H_t)\bar{\Sigma}_t$

return μ_t, Σ_t

EKF_SLAM_Correction

$$Q_t = \begin{pmatrix} {\sigma_r}^2 & 0 \\ 0 & {\sigma_\phi}^2 \end{pmatrix}$$

for all observed features $z_t^i = (r_t^i, \phi_t^i)^T$ do

$\quad j = c_t^i$

$\quad$ if landmark j never seen before

$$\begin{pmatrix} \bar{\mu}_{j,x} \\ \bar{\mu}_{j,y} \end{pmatrix} = \begin{pmatrix} \bar{\mu}_{t,x} \\ \bar{\mu}_{t,y} \end{pmatrix} + \begin{pmatrix} r_t^i \ \cos(\phi_t^i + \bar{\mu}_{t,\theta}) \\ r_t^i \ \sin(\phi_t^i + \bar{\mu}_{t,\theta}) \end{pmatrix}$$

$\quad$ endif

$$\delta = \begin{pmatrix} \delta_x \\ \delta_y \end{pmatrix} = \begin{pmatrix} \bar{\mu}_{j,x} - \bar{\mu}_{t,x} \\ \bar{\mu}_{j,y} - \bar{\mu}_{t,y} \end{pmatrix}$$

$$q = \delta^T \delta$$

$$\hat{z}_t^i = \begin{pmatrix} \sqrt{q} \\ \operatorname{atan2}(\delta_y, \delta_x) - \bar{\mu}_{t,\theta} \end{pmatrix}$$

EKF SLAM – Correction (2/2)

$$F_{x,j} = \begin{pmatrix} 1 & 0 & 0 & 0\cdots0 & 0 & 0 & 0\cdots0 \\ 0 & 1 & 0 & 0\cdots0 & 0 & 0 & 0\cdots0 \\ 0 & 0 & 1 & 0\cdots0 & 0 & 0 & 0\cdots0 \\ 0 & 0 & 0 & 0\cdots0 & 1 & 0 & 0\cdots0 \\ 0 & 0 & 0 & \underbrace{0\cdots0}_{2j-2} & 0 & 1 & \underbrace{0\cdots0}_{2N-2j} \end{pmatrix}$$

$$H_t^i = \frac{1}{q} \begin{pmatrix} -\sqrt{q}\delta_x & -\sqrt{q}\delta_y & 0 & +\sqrt{q}\delta_x & \sqrt{q}\delta_y \\ \delta_y & -\delta_x & -q & -\delta_y & +\delta_x \end{pmatrix} F_{x,j}$$

$$K_t^i = \bar{\Sigma}_t\, H_t^{iT} (H_t^i\, \bar{\Sigma}_t\, H_t^{iT} + Q_t)^{-1}$$

$$\bar{\mu}_t = \bar{\mu}_t + K_t^i (z_t^i - \hat{z}_t^i)$$

$$\bar{\Sigma}_t = (I - K_t^i\, H_t^i)\, \bar{\Sigma}_t$$

endfor

$\mu_t = \bar{\mu}_t$

$\Sigma_t = \bar{\Sigma}_t$

return μ_t, Σ_t