The SLAM Problem

- SLAM stands for simultaneous localization and mapping
- The task of building a map while estimating the pose of the robot relative to this map

- Why is SLAM hard?
  - Chicken-or-egg problem:
    - a map is needed to localize the robot and a pose estimate is needed to build a map

From Burgard et al slides
Why is SLAM a hard problem?

- In the real world, the mapping between observations and landmarks is unknown
- Picking wrong data associations can have catastrophic consequences
- Pose error correlates data associations

From Burgard et al slides
From Burgard et al. slides
Data Association Problem

- A data association is an assignment of observations to landmarks
- In general there are more than \( \binom{n}{m} \) (n observations, m landmarks) possible associations
- Also called “assignment problem”

From Burgard et al slides
State

\[ \mathbf{x} = \begin{bmatrix} \mathcal{R} \\ \mathcal{M} \end{bmatrix} = \begin{bmatrix} \mathcal{R} \\ \mathcal{L}_1 \\ \ldots \\ \mathcal{L}_n \end{bmatrix} \]

Position and orientation of the robot

All landmark positions in original coordinate frame

Landmark 1 position in OCF
A movement model

Formally: car is non-holonomic
A movement model

\[
\begin{bmatrix}
  x \\
  y \\
  \theta
\end{bmatrix}
\rightarrow
\begin{bmatrix}
  x + v \Delta t \cos \theta \\
  y + v \Delta t \sin \theta \\
  \theta
\end{bmatrix}
\]

\[
\begin{bmatrix}
  x \\
  y \\
  \theta
\end{bmatrix}
\rightarrow
\begin{bmatrix}
  x \\
  y \\
  \theta + \Delta \theta
\end{bmatrix}
\]
A movement model

This isn’t linear!

\[
\begin{bmatrix}
x \\
y \\
\theta
\end{bmatrix} \rightarrow \begin{bmatrix}
x + R(\sin(\theta + \Delta \theta) - \sin \theta) \\
y - R(\cos(\theta + \Delta \theta) - \cos \theta) \\
\theta + \Delta \theta
\end{bmatrix}
\]
A movement model

\[
\begin{bmatrix}
x \\
y \\
\theta
\end{bmatrix} \rightarrow \begin{bmatrix}
x + v \Delta t \cos \theta \\
y + v \Delta t \sin \theta \\
\theta
\end{bmatrix}
\]

These two are limits of previous model ($\Delta \theta \rightarrow 0$; $R \rightarrow 0$)

\[
\begin{bmatrix}
x \\
y \\
\theta
\end{bmatrix} \rightarrow \begin{bmatrix}
x \\
y \\
\theta + \Delta \theta
\end{bmatrix}
\]
One kind of measurement model

• Landmark is at:
  • in global coordinate system
• We record distance and heading:
  • measurement

\[
\begin{bmatrix}
  d \\
  \phi
\end{bmatrix} = \begin{bmatrix}
  \sqrt{(x - u)^2 + (y - v)^2} \\
  \text{atan2}(y - u, x - v) - \theta
\end{bmatrix}
\]

THIS ISN’T LINEAR!
Another kind of measurement model

- Landmark is at:
  - in global coordinate system
- We record position in vehicle’s frame:

\[
\begin{bmatrix}
    x_v \\
    y_v
\end{bmatrix} = R_{-\theta} \begin{bmatrix}
    u - x \\
    v - y
\end{bmatrix}
\]

THIS ISN’T LINEAR!
Linearization and noise

• we have noise

\[ n \sim \mathcal{N}(0, \Sigma) \]

• this means \( f(x + n) \) is a random variable

• Write

\[
J_{f,x} = \begin{bmatrix}
\frac{\partial f_1}{\partial x_1} & \ldots & \frac{\partial f_1}{\partial x_j} & \ldots \\
\ldots & \frac{\partial f_i}{\partial x_j} & \ldots & \ldots \\
\ldots & \frac{\partial f_i}{\partial x_j} & \ldots & \frac{\partial f_r}{\partial x_s} \\
\ldots & \ldots & \ldots & \ldots
\end{bmatrix}
\]

• Then

\[
f(x + n) \approx f(x) + J_{f,x}n
\]

• So (approximately)

\[
f(x + n) \sim \mathcal{N}(f(x), J_{f,x} \Sigma J_{f,x}^T)
\]
Linearization and noise

- we have noise

\[ n \sim \mathcal{N}(0, \Sigma) \]

- So (approximately)

\[ f(x, n) \sim \mathcal{N}(f(x, 0), J_{f,n} \Sigma J_{f,n}^T) \]
The Kalman filter

Dynamic Model:

\[ x_i \sim N(D_i x_{i-1}, \Sigma_{d_i}) \]
\[ y_i \sim N(M_i x_i, \Sigma_{m_i}) \]

Start Assumptions: \( \overline{x}_0^- \) and \( \Sigma_0^- \) are known

Update Equations: Prediction

\[ \overline{x}_i^- = D_i \overline{x}_{i-1}^+ \]
\[ \Sigma_i^- = \Sigma_{d_i} + D_i \sigma_{i-1}^+ D_i \]

Update Equations: Correction

\[ \kappa_i = \Sigma_i^- M_i^T \left[ M_i \Sigma_i^- M_i^T + \Sigma_{m_i} \right]^{-1} \]
\[ \overline{x}_i^+ = \overline{x}_i^- + \kappa_i \left[ y_i - M_i \overline{x}_i^- \right] \]
\[ \Sigma_i^+ = \left[ I - \kappa_i M_i \right] \Sigma_i^- \]

Algorithm 11.3: The Kalman Filter.
The extended Kalman filter

- What happens if state update, measurement aren’t linear?
  - particle filter
  - linearize and approximate (EKF)

\[ x_i = f(x_{i-1}, n) \]

Noise - normal, mean 0, Cov known

\[ y_i = g(x_i, n) \]
The extended Kalman filter

- Linearize:

\[ x_i = f(x_{i-1}, n) \]

\[
\mathcal{F}_x = \begin{bmatrix}
\frac{\partial f_1}{\partial x_1} & \cdots & \cdots \\
\cdots & \frac{\partial f_i}{\partial x_j} & \cdots \\
\cdots & \cdots & \cdots 
\end{bmatrix}
\]

\[
\mathcal{F}_n = \begin{bmatrix}
\frac{\partial f_1}{\partial n_1} & \cdots & \cdots \\
\cdots & \frac{\partial f_i}{\partial n_j} & \cdots \\
\cdots & \cdots & \cdots 
\end{bmatrix}
\]

Posterior covariance of \( x_{\{i-1\}} \)

\[ x_i \sim \mathcal{N} (f(x_{i-1}, 0), \mathcal{F}_x \sum_{i-1}^+ \mathcal{F}_x^T + \mathcal{F}_n \sum_{n,i} \mathcal{F}_n^T) \]

Noise covariance
Dynamic Model:

\[ y_i \sim N(\mathcal{M}_i x_i, \Sigma_{m_i}) \]

Start Assumptions: \( \bar{x}_0^- \) and \( \Sigma_0^- \) are known

Update Equations: Prediction

\[
x_i \sim \mathcal{N} \left( f(x_{i-1}, 0), \mathcal{F}_x \Sigma_{i-1}^+ \mathcal{F}_x^T + \mathcal{F}_n \Sigma_{n,i} \mathcal{F}_n^T \right)
\]

Update Equations: Correction

\[
\mathcal{K}_i = \Sigma_i^- \mathcal{M}_i^T \left[ \mathcal{M}_i \Sigma_i^- \mathcal{M}_i^T + \Sigma_{m_i} \right]^{-1}
\]

\[
\bar{x}_i^+ = \bar{x}_i^- + \mathcal{K}_i [y_i - \mathcal{M}_i \bar{x}_i^-]
\]

\[
\Sigma_i^+ = [I_d - \mathcal{K}_i \mathcal{M}_i] \Sigma_i^-
\]

Algorithm 11.3: The Kalman Filter.
The extended Kalman filter

• Linearize:

\[ y_i = g(x_i, n) \]

\[ G_x = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \cdots \\ \vdots & \ddots & \vdots \\ \frac{\partial f_i}{\partial x_j} & \cdots & \vdots \end{bmatrix} \]

\[ G_n = \begin{bmatrix} \frac{\partial f_1}{\partial n_1} & \cdots & \cdots \\ \vdots & \ddots & \vdots \\ \frac{\partial f_i}{\partial n_j} & \cdots & \vdots \end{bmatrix} \]

\[ y_i \sim \mathcal{N}(f(x_i, 0), G_x \Sigma_i G_x^T + G_n \Sigma_{m,i} G_n^T) \]
Dynamic Model:

\[ y_i \sim N(\mathcal{M}_i x_i, \Sigma_{m_i}) \]

Start Assumptions: \( \bar{x}_0 \) and \( \Sigma_0^- \) are known
Update Equations: Prediction

\[ x_i \sim \mathcal{N}(f(x_{i-1}, 0), \mathcal{F}_x \Sigma_{i-1}^+ \mathcal{F}_x^T + \mathcal{F}_n \Sigma_{n,i} \mathcal{F}_n^T) \]

Update Equations: Correction

\[ K_i = \Sigma_i^- \mathcal{M}_i^T [\mathcal{M}_i \Sigma_i^- \mathcal{M}_i^T + \Sigma_{m_i}]^{-1} \]

\[ \bar{x}_i^+ = \bar{x}_i^- + K_i [y_i - \mathcal{M}_i \bar{x}_i^-] \]

\[ \Sigma_i^+ = [I - K_i \mathcal{M}_i] \Sigma_i^- \]

This is the inverse of the covariance of \( y_i \)

Algorithm 11.3: The Kalman Filter.
Dynamic Model:

\[ y_i \sim N(\mathcal{M}_i x_i, \Sigma_m) \]

Start Assumptions: \( \bar{x}_0^- \) and \( \Sigma_0^- \) are known

Update Equations: Prediction

\[ \bar{x}_i \sim \mathcal{N}(f(x_{i-1}, 0), \mathcal{F}_x \Sigma_{i-1}^+ \mathcal{F}_x^T + \mathcal{F}_n \Sigma_{n,i} \mathcal{F}_n^T) \]

Update Equations: Correction

\[ \mathcal{K}_i = \Sigma_i^- M_i^T \left[ M_i \Sigma_i^- M_i^T + \Sigma_m \right]^{-1} \]
\[ \bar{x}_i^+ = \bar{x}_i^- + \mathcal{K}_i \left[ y_i - M_i \bar{x}_i^- \right] \]
\[ \Sigma_i^+ = [Id - \mathcal{K}_i M_i] \Sigma_i^- \]

Algorithm 11.3: The Kalman Filter.
Dynamic Model:

\[ y_i \sim N(\mathcal{M}_i x_i, \Sigma_{m_i}) \]

Start Assumptions: \( \bar{x}_0^- \) and \( \Sigma_0^- \) are known

Update Equations: Prediction \( \bar{x}_i^- \)

\[ x_i \sim \mathcal{N}(f(x_{i-1}, 0), \mathcal{F}_x \Sigma_i^+ \mathcal{F}_x^T + \mathcal{F}_n \Sigma_n \Sigma_n, i \mathcal{F}_n^T) \]

Update Equations: Correction

\[ \mathcal{K}_i = \Sigma_i^- \mathcal{M}_i^T [\mathcal{M}_i \Sigma_i^- \mathcal{M}_i^T + \Sigma_{m_i}]^{-1} \]

\[ \bar{x}_i^+ = \bar{x}_i^- + \mathcal{K}_i [y_i - \mathcal{M}_i \bar{x}_i^-] \]

\[ \Sigma_i^+ = [\text{Id} - \mathcal{K}_i \mathcal{M}_i] \Sigma_i^- \]

Algorithm 11.3: The Kalman Filter.
Dynamic Model:

\[ x_i = f(x_{i-1}, n) \]
\[ y_i = g(x_i, n) \]

Start Assumptions: \( \bar{x}_0^- \) and \( \Sigma_0^- \) are known

Update Equations: Prediction

\[ \bar{x}_i^- \xrightarrow{\Sigma_i^-} x_i \sim \mathcal{N}(f(x_{i-1}, 0), \mathcal{F}_x \sum_{i-1}^+ \mathcal{F}_x^T + \mathcal{F}_n \Sigma_n, i \mathcal{F}_n^T) \]

Update Equations: Correction

\[ \mathcal{K}_i = \Sigma_i^- \mathcal{M}_i^T \left[ \mathcal{G}_x \Sigma_i^- \mathcal{G}_x^T + \mathcal{G}_n \Sigma_{m,i} \mathcal{G}_n^T \right]^{-1} \]
\[ \bar{x}_i^+ = \bar{x}_i^- + \mathcal{K}_i \left[ y_i - g(x_i^-, 0) \right] \]
\[ \Sigma_i^+ = [Id - \mathcal{K}_i \mathcal{G}_x] \Sigma_i^- \]

Algorithm 11.3: The Kalman Filter.
In principle, now easy

- BUT
  - $F_x$ is much simpler than it might look
    - the landmarks do not move!
  - $F_n$ ditto
    - there is no noise in the landmark updates

$$F_x = \begin{bmatrix} 3 & N \\ \frac{\partial f_R}{\partial R} & 0 \\ 0 & \mathcal{I} \end{bmatrix}$$

$N =$ Number of landmarks

$$F_n = \begin{bmatrix} \frac{\partial f_R}{\partial n} & 0 \\ 0 & 0 \end{bmatrix}$$
More simplifications

• **BUT**
  • $G_x$ is much simpler than it might look
    • each set of measurements affected by only one landmark!

$$G_x = \begin{bmatrix}
\frac{3}{\partial g_R}{\partial L_1} & \frac{3}{\partial L_1} & 0 & 0 & 0 & 0 \\
\frac{\partial g_R}{\partial R} & 0 & \frac{\partial g_L_2}{\partial L_2} & 0 & 0 & 0 \\
\cdots & \cdots & \cdots & \frac{\partial g_L_i}{\partial L_i} & \cdots \\
\frac{\partial g_R}{\partial R} & \cdots & \cdots & \cdots & \cdots \\
\end{bmatrix}$$

$N$ $N=$Number of landmarks
More simplifications

• BUT
  • $G_n$ is usually much simpler than it might look
    • noise is usually additive normal noise

\[ G_n^T \Sigma_{n,i} G_n \rightarrow \Sigma_{n,i} \]
Landmarks

• Which measurement comes from which landmark?
  • data association -
    • for the moment, assume
      • we use a bipartite graph matcher
      • or draw independent samples from posterior on landmark
        • given measurement
    • ideally, we’d average over all matchings - put that off
Landmarks

• No measurement from a landmark?
  • structure of EKF means you can process landmarks one by one
  • don’t update that landmark

• New landmark?
  • full observation (eg range+bearing, lidar)
  • partial observation (eg bearing, vision)
Full observation

• Must make estimates of
  • landmark mean state
    • invert the observation of the landmark
  • landmark covariance
    • with itself
    • with others
    • use jacobians of inverted observation
Range and bearing

Observation $\begin{bmatrix} d \\ \phi \end{bmatrix} = \begin{bmatrix} \sqrt{(x - u)^2 + (y - v)^2} \\ \text{atan2}(y - u, x - v) - \theta \end{bmatrix}$

Vehicle state

Landmark position

$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} x + d \sin(\phi + \theta) \\ y + d \cos(\phi + \theta) \end{bmatrix} = h(\bar{x}_i, y_i) = h(\bar{x}_i, l)$

Here use the current estimate of vehicle state

These are measurements of new landmark ONLY
Range and bearing

- but the measurement may be affected by noise
  - additive noise, normal, zero mean, covar \( \Sigma \)
- So I should have written

\[
\begin{bmatrix}
  u \\
v
\end{bmatrix} = \begin{bmatrix}
x + (d + \xi) \sin(\phi + \zeta + \theta) \\
y + (d + \xi) \cos(\phi + \zeta + \theta)
\end{bmatrix} = h(\bar{x}_i, y_i, \xi, \zeta) = h(\bar{x}_i, 1, n)
\]

- And I need to do some surgery
  - on the state vector
  - and on the covariance matrix
Range and bearing - state vector surgery

- Because the noise has zero mean

\[ \mathbf{x}_i^+ \rightarrow \begin{bmatrix} \mathbf{x}_i^+ \\ u \\ v \end{bmatrix} \]
Range and bearing - covariance surgery

\[
\begin{bmatrix}
  u \\
  v 
\end{bmatrix} = \begin{bmatrix}
  x + (d + \xi) \sin(\phi + \zeta + \theta) \\
  y + (d + \xi) \cos(\phi + \zeta + \theta)
\end{bmatrix} = h(\bar{x}^{-}_i, y_i, \xi, \zeta) = h(\bar{x}^{-}_i, l, n)
\]

Covariance of vehicle state with itself

- So

\[
\mathcal{H}_x \Sigma^+_{i, xx} \mathcal{H}_x^T + \mathcal{H}_n \Sigma \mathcal{H}_n^T
\]

Covariance of landmark with itself

Jacobian of landmark position wrt vehicle state

- and

\[
\mathcal{H}_x \Sigma^+_{i, \mathcal{R} M}
\]

Covariance of landmark with everything else

i’th posterior covariance of location with all other landmarks
Range and bearing - covariance surgery

\[
\Sigma^+_i \rightarrow \begin{bmatrix}
\Sigma^+_i \\
(\mathcal{H}_x \Sigma^+_i, \mathcal{R}_M) \\
\mathcal{H}_x \Sigma^+_i, xx \mathcal{H}_x^T + \mathcal{H}_n \Sigma \mathcal{H}_n^T
\end{bmatrix}
\]
Bearing only (sketch)

- Cannot determine landmark in 2D from measurement
  - it’s on a line!
  - you must come up with a prior
    - after that, it’s easy
      - find mean posterior location, covariance
      - plug in
  - Big Issue
    - True prior should have infinite covariance
      - can’t work with that
      - so linearization may fail