The Simplest SLAM

I wish to recover both state of robot AND a map of the world.

CASE: 2D translating robot,
N Beacons at \( m_i \) (in World coords)
- State of robot \( x_i \)
- robot measures location of beacons in its frame

\[
\begin{bmatrix}
  x_i \\
  m_i \\
  m_i \ldots
\end{bmatrix}
\]

Dynamics
\[
\begin{align*}
x_i+1 &= x_i + k_i \hat{m}_i \\
M_i+1 &= M_i
\end{align*}
\]

Measurement
\[
\begin{align*}
k_i \hat{a}_i &= (m_i - x_i) \\
\eta_i
\end{align*}
\]

This is all linear, so sling it in a KF
Data association
- you need to know which measurement comes from which beacon

Strategies
- \( \log P(x|\theta) \) and matching (Bipartite/Greedy)
- mark the beacons

Missing data:
- What happens if a beacon isn't found?
- Easy: Drop from measurement matrix, proceed

How accurate?
- should be very good, with more than 3-2 beacons
- 2 is a problem
- Rotating translating robot
- Like our car.
- Motion of robot depends on state

(Formal - Non-holonomic).

State of car:

\[ \text{posu in W.C} \rightarrow x \]
\[ \rightarrow y \]
\[ \rightarrow \theta \]

angle in W.C.
Pure rotation:

\[ x \rightarrow x + \Delta x = x + R(\sin(\theta + \Delta \theta) - \sin \theta) \]
\[ y \rightarrow y + \Delta y = y - R(\cos(\theta + \Delta \theta) - \cos \theta) \]
\[ \theta \rightarrow \theta + \Delta \theta \]
Notice:

- **pure rotation** = motion in a circle, $R = 0$
- **pure translation** = 

$$ R \to \infty = \frac{1}{\Delta \theta} $$
$$ \Delta \theta \to 0 $$
(take derivatives).

So our model is

$$ x \to x + R (\sin(\theta + \Delta \theta) - \sin \theta) $$
$$ y \to y - R (\cos(\theta + \Delta \theta) - \cos \theta) $$
$$ \theta \to \theta + \Delta \theta $$

where $R$ comes from velocity etc.

Notice this isn't linear

**Measurement**

$ g_i = R(\theta) [k \cdot (x_i - \bar{x})] + \eta_i $  

depends on $\theta$, nonlinear.

noise
- EKF = extended Kalman filter
  - linearize dynamics, motion around current best estimate
  - apply KF

- Particle filter:
  - this needs to be done with care for our vehicle
  - because dynamic uncertainty is quite large and meas. uncertainty is quite small.

- Options
  - propagate samples through dynamics reweight (as we did) DICEY
  - alternative: propose particles from measurements, reweight.
For particles will have slightly different in prior at start.

For some particle, find an consistent with measurements, and a

Now construct a proposal first

\[ Q(\text{state}_{i+1} \mid \text{state}_{i+1}) \]

(e.g., normal, centered on mean).

Reweight w/ dynamics.
the math:

Our Original P.F.

\[ p_r \sim P(X_{i+1} | Y_i \ldots Y_i) \]

weight:

\[ (p_r, w_r) = (p_r, P(Y_{i+1} | X_{i+1} = p_r)) \]

eval:

\[ \frac{\sum f(p_r) w_r}{\sum w_r} \approx N \int f(u) \cdot P(Y_{i+1} | X_{i+1} = u) \cdot P(X_{i+1} = u | Y_i \cdot Y_i) \, du \]

\[ \approx \int f(u) \cdot P(X_{i+1} = u | Y_i \ldots Y_{i+1}) \, du. \]
\[ p_r \sim Q(X_{i+1} \mid \text{meas}) \]

Weight:

\[ p_r \cdot \frac{\mathbb{P}(Y_{i+1} \mid X_{i+1} = p_r) \cdot \mathbb{P}(X_{i+1} = p_r \mid Y_1, \ldots, Y_i)}{Q(X_{i+1} \mid \text{meas})} \]

Check:

\[ \sum f(p_r) \cdot w_r \approx N \int f(u) \cdot \left[ \frac{\mathbb{P}(Y_{i+1} \mid X_{i+1} = u) \cdot \mathbb{P}(X_{i+1} = u \mid Y_1, \ldots, Y_i)}{Q(X_{i+1} = u \mid \text{meas})} \right] \cdot \mathbb{Q}(X_{i+1} = u \mid \text{meas}) \, du \]

OK.
How to make $\mathcal{Q}$.

- at particle, we have an est of old $\theta$.
- so we can do linear least squares

\[
R(\theta + \Delta \theta) = R(\theta)[I + \begin{bmatrix} 0 & -\Delta \theta \\ \Delta \theta & 0 \end{bmatrix}] \quad \text{or Taylor series}
\]

\[
k_i = R(\theta)[I + \begin{bmatrix} 0 & \Delta \theta \\ -\Delta \theta & 0 \end{bmatrix}] R^T(\theta) \begin{bmatrix} x_i - x_{i+1} \\ y_i - y_{i+1} \end{bmatrix}
\]

- ignore $o(\Delta^2)$, solve as linear least squares.

OR

- est rotation, tx as before!
Simple Bundle Adjustment

+ assume point csp's are known

\[ x_i \]

+ coord frame

\[ \cdot \]

+ point identity

\( (\text{so we see point } i \text{ in several frames}) \)

+ we want best est. of \( x_i \)

\[ m_i \]

+ write \( m_i \) for \( x_i \)

\[ \text{frame 1 is world} \]

+ assume

\[ \sum_{\text{kepts.}} \sum_{\text{vis.}} \left[ R(\theta_k) x_i + t_k - m_i \right]^2 = F(\theta_k, t, m_i) \]

\[ \text{Minimize this WRT } \theta_k, t, m_i \]
Iterate two phases

\[ \hat{\Theta}_k, \hat{\Theta} = \text{argmin} \ F(\Theta_k, \hat{\Theta}; \hat{M}_i) \]

Here we fix world points, adjust \( \Theta, \hat{\Theta} \).

Notice these decomposes, but isn't linear

\[ \hat{M}_i = \text{argmin} \ F(\hat{\Theta}_k, \hat{\Theta}; M_i) \]

Least squares

You can extend this to ICP if you choose

- a fixed number of points in world coords
- initialize with map constructed above
- complicate w/ IRLS,