Lecture 8 - Sparse Extended Information Filter - for SLAM

- matrices in computation are sparse
  ↳ eventually results in a constant complexity

Motivation

- can do involved ops in constant time complexity
- inverting correlation is expensive
- information mtx is just as dense as correlation mtx but some elements are much stronger than others
  ↳ we can set all elements $< \epsilon$ to 0 then we have a sparse information mtx → Sparse EIF is not that simple

☆ most features have only a small number of strong links
↳ this can be visualized as a graph (the information mtx)

for later
- can show an equivalence between the information mtx & a Gaussian Markov random field

Information Matrix

- can be interpreted as a graph of constraints / links between nodes
- can be interpreted as a MRF
- two non-connected nodes (missing link) indicates conditional independance given all other nodes

1 — 2 — 3

↳ if you know node 2, then 1 & 3 are conditionally independent

- $\Omega_{i,j}$ tells us the strength of a link
- most off-diagonal elements are close to 0 (but $\neq 0$)

Create Sparcity

- "Set" most links to zero / avoid fill-in
- Sparse = finite number of off-diagonals, independant of the mtx size

SEIF SLAM - Steps

- Measurement Update
  - before any measurement - Slide 8
  - Slide 10 - second measurement doesn't fill m_1, m_2 or m_2, m_1
  - ↳ Since we have robot's pose, knowing m_2 doesn't increase knowledge about m_1 (or vice versa)
  - ↳ adds info between robot's pose & observed features

- Motion Update
  - pose is updated in mtx, prev pose is marginalized out
  - all neighbors of prev. pose are connected as a result of the marginalization
  - ↳ by moving from x_t to x_{t+1}, we add additional uncertainty to the current (new) pose → not added to prev. observed landmarks which results in them being correlated w/ eachother
  - links to new pose & prev landmarks get weaker

- Sparsification
  - this is an approximation
  - ↳ Gaussian estimation becomes worse but gain compute efficiency
  - ignore certain links → become 0
  - ↳ resulting belief assumes conditional independance
  - Some information propagates to other off-diagonals

Active & Passive Landmarks

- Key element to obtain efficient algorithm
- Active Landmarks: - Subset of all landmarks including currently observed ones — there may be more but we set a constant # of active landmarks
- Passive Landmarks: - all other landmarks

- only keep direct links to active landmarks
- if all landmarks are active ⇒ this is just EIF/EKF efficiency
- landmarks have links to only nearby landmarks (landmarks that have been active at the same time)

↳ the more you limit your active set, the more you limit links among landmarks in general

Four Steps of SEIF SLAM

1) Motion Update
2) Measurement Update
3) Update State Estimate (mean) ← Should probably be first?
   ↳ needed for to apply motion update, computing expected measurement, & sparsification
4) Sparsification

⇓

SEIF_SLAM($\xi_{t-1}, \Omega_{t-1}, \mu_{t-1}, u_t, z_t$):

$\bar{\xi}, \bar{\Omega}, \bar{\mu}$ = SEIF_motion_update($\xi_{t-1}, \Omega_{t-1}, \mu_{t-1}, u_t$)

ξ_t, Ω_t = SEIF_measurement_update($\bar{\xi}_t, \bar{\Omega}_t, \bar{\mu}_t, z_t$)

μ_t = SEIF_update_State_Estimate($\xi_t, \Omega_t, \bar{\mu}_t$)

$\tilde{\xi}, \tilde{\Omega}$ = SEIF_Sparsification(ξ_t, Ω_t, μ_t)

return $\tilde{\xi}, \tilde{\Omega}, \mu_t$

ideally, P maps R^{-1} to a lower dimensionality, so it can be added to Q^{-1}. Q needs to be lower dimensionality than R for this to be useful.

Matrix Inversion Lemma

- will be useful soon
- For any invertible quadratic matrices R & Q, and any mtx P, the following holds true:

$$(R + PQP^T)^{-1} = R^{-1} - R^{-1}P(Q^{-1} + P^T R^{-1} P)^{-1} P^T R^{-1}$$

↳ for ex. if we know R^{-1} & Q^{-1} is low dimension (sparse), this can be computed very efficiently.

SEIF SLAM - Prediction Step (motion update) (1/3)

Goal: Compute $\bar{\Sigma}, \bar{\Omega}, \bar{\mu}$ from motion & $\Sigma_{t-1}, \Omega_{t-1}, \mu_{t-1}$

↳ Efficiency by exploiting sparseness of information mtx

SEIF_motion_update($\Sigma_{t-1}, \Omega_{t-1}, \mu_{t-1}, u_t$):

$$F_x = \begin{bmatrix} 1 & 0 & 0 & \overbrace{0 \cdots 0}^{2N} \\ 0 & 1 & 0 & 0 \cdots 0 \\ 0 & 0 & 1 & 0 \cdots 0 \end{bmatrix}$$ → from line 2. of EKF SLAM

$$\delta = \begin{bmatrix} -\frac{v_t}{w_t}\sin(\mu_{t-1,\theta}) + \frac{v_t}{w_t}\sin(\mu_{t-1,\theta} + w_t\Delta t) \\ \frac{v_t}{w_t}\cos(\mu_{t-1,\theta}) - \frac{v_t}{w_t}\cos(\mu_{t-1,\theta} + w_t\Delta t) \\ w_t\Delta t \end{bmatrix}$$ → from line of "

$$\Delta = \begin{bmatrix} 0 & 0 & \frac{v_t}{w_t}\cos(\mu_{t-1,\theta}) - \frac{v_t}{w_t}\cos(\mu_{t-1,\theta} + w_t\Delta t) \\ 0 & 0 & \frac{v_t}{w_t}\sin(\mu_{t-1,\theta}) - \frac{v_t}{w_t}\sin(\mu_{t-1,\theta} + w_t\Delta t) \\ 0 & 0 & 0 \end{bmatrix}$$ → from Line 4. of "

→ if following EKF SLAM, now compute inverse of Cov. matrix →

We know $F_x^T R_t^x F_x$ maps R_t^x ($\to$ 3 dims.) to a higher space.
↳ So $F_x \Phi_t F_x^T$ will map Φ_t down to 3 dims. & can be added to R_t^x

Compute Information Matrix

$$\bar{\Omega}_t = \bar{\Sigma}_t^{-1}$$
$$= [G_t \Omega_{t-1}^{-1} G_t^T + R_t]^{-1}$$
$$= [\Phi_t^{-1} + R_t]^{-1}$$

$\to$ where $\Phi_t = [G_t \Omega_{t-1}^{-1} G_t^T]^{-1}$ } inverse of Product
$= [G_t^T]^{-1} \Omega_{t-1} G_t^{-1}$

$$= [\Phi_t^{-1} + F_x^T R_t^x F_x]^{-1}$$
$$= \Phi_t - \Phi_t F_x^T (R_t^{x-1} + F_x \Phi_t F_x^T)^{-1} F_x \Phi_t$$

$\to$ apply m+x inv. lemma

$\to$ 3x3 m+x w/ R_t is 3x3 $\to$ fast inverse comp.

$\to$ Zero's except 3x3 block

$\to$ So $F_x^T(\dots)^{-1}F_x$ is possibly large but only 3x3 is populated & the rest is zero's

$\to$ Constant Complexity iff Φ_t is sparse!

$$= \Phi_t - \underbrace{\Phi_t F_x^T (R_t^{x-1} + F_x \Phi_t F_x^T)^{-1} F_x \Phi_t}_{K_t}$$

$$\bar{\Omega}_t = \Phi_t - K_t$$

Computing $\Phi_t = [G_t^T]^{-1} \Omega_{t-1} G_t^{-1}$

- Goal: Const. time if Ω_{t-1} is sparse

$$G_t^{-1} = (I + F_x^T \Delta F_x)^{-1}$$
$$= \begin{bmatrix} \Delta + I_3 & 0 \\ 0 & I_{2N} \end{bmatrix}^{-1}$$
$$= \begin{bmatrix} (\Delta + I_3)^{-1} & 0 \\ 0 & I_{2N} \end{bmatrix}$$

$\to$ inverse goes from depending on whole matrix to only on 3x3 m+x. (constant of # of landmarks)

$$= I_{3+2N} + \begin{bmatrix} (\Delta + I_3)^{-1} - I_3 & 0 \\ 0 & 0 \end{bmatrix}$$
$$= I + \underbrace{F_x^T [(I+\Delta)^{-1} - I] F_x}_{\Psi_t}$$
$$= I + \Psi_t$$

$\Sigma = \Omega^{-1}$ $\Omega = \Sigma^{-1}$

$\mu = \Omega^{-1}\xi$ $\xi = \Sigma^{-1}\mu$

$= \Sigma\xi$ $= \Omega\mu$

- So we have

$$G_t^{-1} = I + \Psi \quad \text{and} \quad [G_t^T]^{-1} = I + \Psi_t^T$$

- with

$$\Psi = F_x^T[(I+\Delta)^{-1} - I]F_x$$

meaning sparse: ↳ zero's except 3x3 block ↳ identity except 3x3 block } ∴ Φ_t can be computed in constant time

Constant Time Comp. of Φ_t

- if Ω_{t-1} is sparse, the constant time update can be seen by:

$$\Phi_t = [G_t^T]^{-1}\Omega_{t-1}G_t^{-1} \longrightarrow \underbrace{[\text{constant \# entries}][\text{sparse}][\text{constant \# entries}]}_{\text{constant computation time}}$$

$$= (I + \Psi_t^T)\Omega_{t-1}(I + \Psi_t)$$

$$= \Omega_{t-1} + \underbrace{\Psi_t^T\Omega_{t-1} + \Omega_{t-1}\Psi_t + \Psi_t^T\Omega_{t-1}\Psi_t}_{\lambda_t}$$

$\Phi = \Omega_{t-1} + \lambda$ → all elements 0 except const. # of entries

SEIF - Prediction Step (2/3)

SEIF_motion_update($\xi_{t-1}, \Omega_{t-1}, \mu_{t-1}, u_t$):

$F_x = \ldots$

$\delta = \ldots$

$\Delta = \ldots$

$\Psi = F_x^T[(I+\Delta)^{-1} - I]F_x$

$\lambda_t = \Psi_t^T\Omega_{t-1} + \Omega_{t-1}\Psi_t + \Psi_t^T\Omega_{t-1}\Psi_t$

$\Phi = \Omega_{t-1} + \lambda_t$

$K_t = \Phi_t F_x^T(R_t^{-1} + F_x\Phi_t F_x^T)^{-1}F_x\Phi_t$

$\bar{\Omega}_t = \Phi - K_t$

↳ Information mtx now computed efficiently

↳ now lets do the same w/ info. vector & mean

Compute Mean

↳ computed same way as EKF SLAM

$$\bar{\mu}_t = \mu_{t-1} + \underbrace{F_x^T \delta}_{\text{the non-lin func.}}$$

Compute Information Vector

$\bar{\xi}_t = \bar{\Omega}_t \bar{\mu}_t$ → ok, but this is linear complexity, would be stupid since we came all this way for efficiency.

$$= \bar{\Omega}_t (\mu_{t-1} + F_x^T \delta)$$
$$= \bar{\Omega}_t (\Omega_{t-1}^{-1} \xi_{t-1} + F_x^T \delta_t)$$
$$= \bar{\Omega}_t \Omega_{t-1}^{-1} \xi_{t-1} + \bar{\Omega}_t F_x^T \delta_t$$

take out this term for next step

$$= (\bar{\Omega}_t \underbrace{- \Phi_t + \Phi_t}_{=0} \underbrace{- \Omega_{t-1} + \Omega_{t-1}}_{=0}) \Omega_{t-1}^{-1} \xi_{t-1} + \bar{\Omega}_t F_x^T \delta_t$$

$$= (\underbrace{\bar{\Omega}_t - \Phi_t}_{=-K_t} + \underbrace{\Phi_t - \Omega_{t-1}}_{\lambda_t}) \underbrace{\Omega_{t-1}^{-1} \xi_{t-1}}_{\mu_{t-1}} + \underbrace{\Omega_{t-1} \Omega_{t-1}^{-1}}_{=I} \xi_{t-1} + \bar{\Omega}_t F_x^T \delta$$

$$= (\lambda_t - K_t) \mu_{t-1} + \xi_{t-1} + \bar{\Omega}_t F_x^T \delta_t$$
$$= \xi_{t-1} + (\lambda_t - K_t) \mu_{t-1} + \bar{\Omega}_t F_x^T \delta_t$$

SEIF - Prediction Step (3/3)

SEIF_Motion_Update$(\xi_{t-1}, \Omega_{t-1}, \mu_{t-1}, u_t)$:

→ if this guy is sparse & motion update only affects pose of robot (not landmarks), we can compute this whole function in constant time w/o any additional assumptions than b4

$F_x = \ldots$

$\delta = \ldots$

$\Delta = \ldots$

$\Psi_t = F_x^T [(I + \Delta)^{-1} - I] F_x$

$\lambda_t = \Psi_t^T \Omega_{t-1} + \Omega_{t-1} \Psi_t + \Psi_t^T \Omega_{t-1} \Psi_t$

$\Phi = \Omega_{t-1} + \lambda_t$

$K_t = \Phi_t F_x^T (R_t^{-1} + F_x \Phi_t F_x^T)^{-1} F_x \Phi_t$

$\bar{\Omega}_t = \Phi_t - K_t$

$\bar{\xi}_t = \xi_{t-1} + (\lambda_t - K_t) \mu_{t-1} + \bar{\Omega}_t F_x^T \delta_t$

$\bar{\mu}_t = \mu_{t-1} + F_x^T \delta_t$

return $\bar{\xi}_t, \bar{\Omega}_t, \bar{\mu}_t$

✓ Done motion update

SEIF - Measurement Update (1/2)

SEIF_measurement_update($\bar{\xi}_t, \bar{\Omega}_t, \mu_t, z_t$) $\rightarrow (c_t, R_t)$?

Same as EKF SLAM

$Q_t = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\phi^2 \end{bmatrix}$ $\rightarrow$ observation uncertainty (sensors)

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

$j = c_t^i$ $\rightarrow$ Data association

if landmark j never seen before:

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

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

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

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

$$H_t^i = \frac{1}{q} \begin{bmatrix} -\sqrt{q}\,\delta_x & -\sqrt{q}\,\delta_y & 0 & \underbrace{\begin{matrix} 0 \dots 0 \\ 0 \dots 0 \end{matrix}}_{2j-2} & \underbrace{\begin{matrix} +\sqrt{q}\,\delta_x & +\sqrt{q}\,\delta_y \\ -\delta_y & +\delta_x \end{matrix}}_{\text{landmark } j} & \underbrace{\begin{matrix} 0 \dots 0 \\ 0 \dots 0 \end{matrix}}_{2N-2j} \\ \delta_y & -\delta_x & -q & & & \end{bmatrix}$$

$$\xi_t = \bar{\xi}_t + \sum_i H_t^{i\,T} Q_t^{-1} \left[z_t^i - \hat{z}_t + H_t^i \mu_t \right]$$
$$\Omega_t = \bar{\Omega}_t + \sum_i H_t^{i\,T} Q_t^{-1} H_t^i$$

taken from EIF alg.
$\hookrightarrow$ the Kalman gain is integrated in these derived terms as well

return ξ_t, Ω_t