Hidden Markov Models (HMM)

Mon, Mar 26 : Bayesian Filtering.

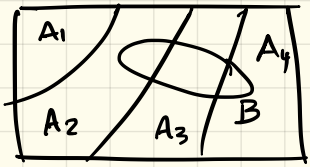
① Prob. Basics :

- Cond. Prob. :

$$P(A|B) = \frac{P(A,B)}{P(B)}$$

- Bayes' Rule:

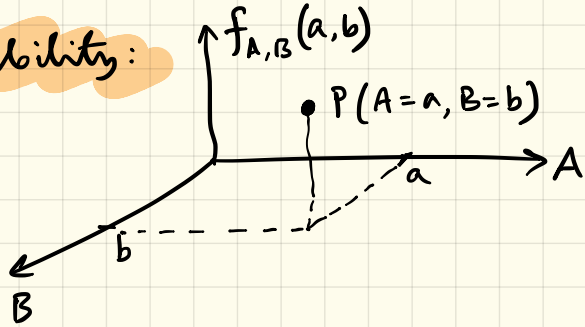
$$P(A_i|B) = \frac{P(B|A_i)\,P(A_i)}{P(B)}$$



$$\underbrace{P(A_i|B)}_{\text{A posteriori}} = \frac{\overbrace{P(B|A_i)}^{\text{likelihood}}\,P(A_i)}{\sum_{i=1}^{n} P(B|A_i)P(A_i)}$$

- Joint Probability:

$$P(A,B)$$

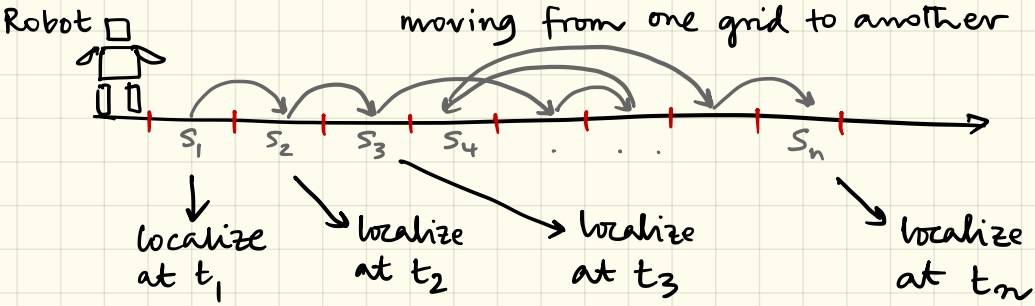


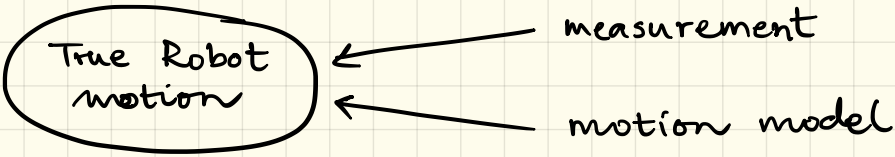
- Marginal Prob:

$$P(A) = \sum_{B} P(A,B) \qquad f_A(a) = \int_{-\infty}^{\infty} f_{A,B}(a,b)\,db$$

- Chain rule:

$$P(A,B,C) = P(A|BC)\cdot P(B|C)\cdot P(C)$$



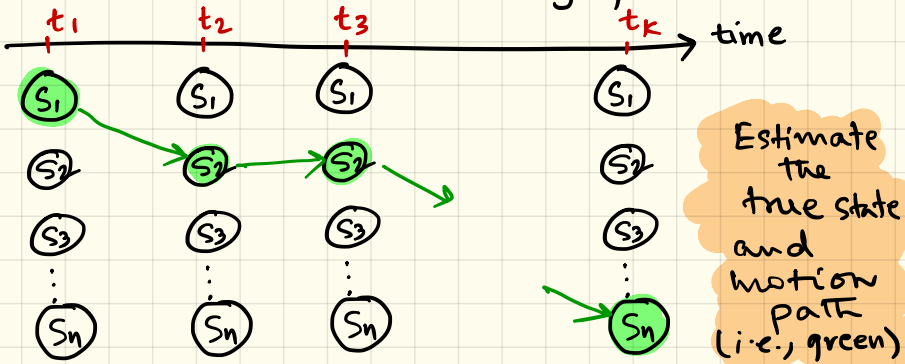


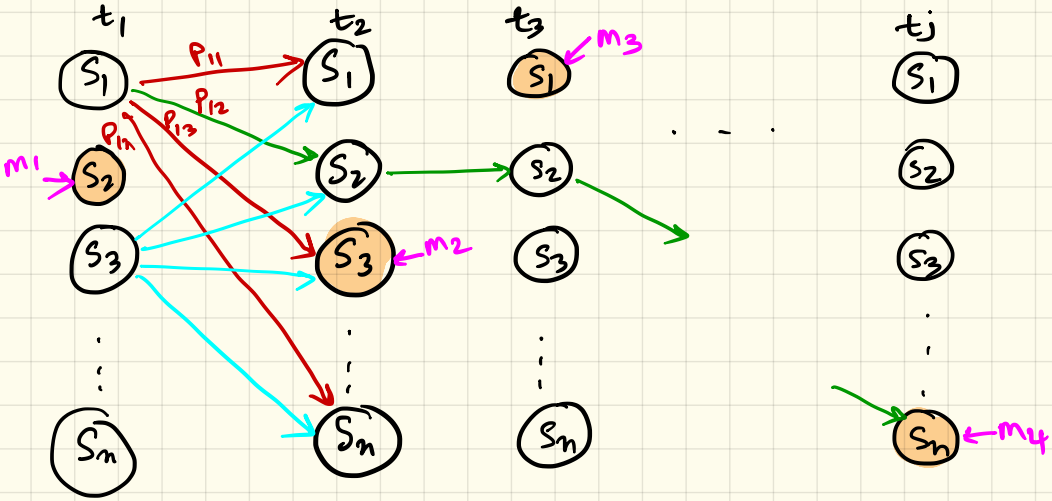
- Motion model governs how robot moves from one grid location to another.

Example:

$$P(s_i \rightarrow s_{i+1}) = 0.5$$
$$P(s_i \rightarrow s_{i+2}) = 0.2$$
$$P(s_i \rightarrow s_{i+j}, j \geq 3) = 0$$
$$P(s_i \rightarrow s_i) = 0.2$$
$$P(s_i \rightarrow s_{i-1}) = 0.1$$
$$P(s_i \rightarrow s_{i-k}, k \geq 2) = 0$$

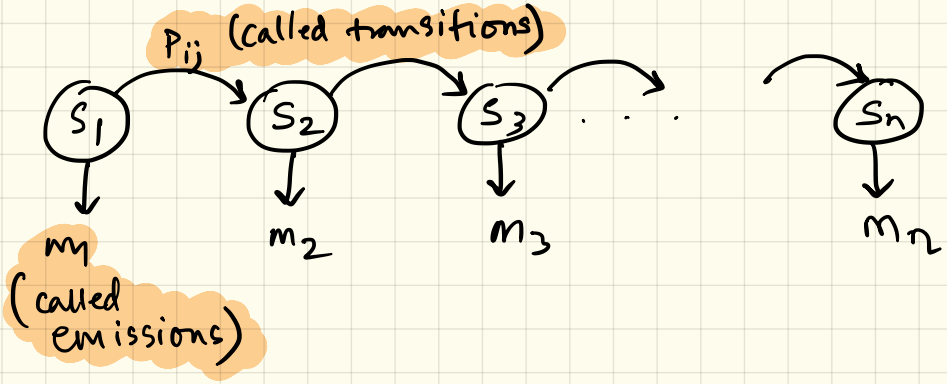
- Model the robot motion tracking problem as:



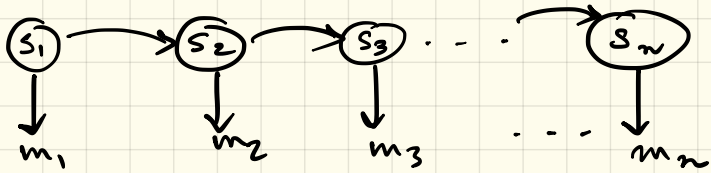


- Now goal is to measure the robot's location at every time, and combine with the motion model (i.e., state transition probabilities) to infer the true state of the robot.

- Modeled as a "State Transition Diagram"



Bayesian Filtering



What is $P(s_k \mid m_{1:n})$

Good result to keep in mind

$$\Rightarrow \quad P(s_{1:n} \mid m_{1:n}) = \frac{P(s_{1:n}, m_{1:n})}{P(m_{1:n})} \propto P(s_{1:n}, m_{1:n})$$

$$P(s_{1:n}, m_{1:n}) = P(m_n \mid m_{1:n-1}, s_{1:n})\, P(m_{n-1} \mid m_{1:n-2}\, s_{1:n}) \cdots P(m_1 \mid s_{1:n})\, P(s_n \mid s_{1:n-1}) \cdots P(s_2 \mid s_1)\, P(s_1)$$

markov

$$= P(m_n \mid s_n)\, P(m_{n-1} \mid s_{n-1}) \cdots P(m_1 \mid s_1) \cdot P(s_n \mid s_{n-1})\, P(s_{n-1} \mid s_{n-2}) \cdots P(s_2 \mid s_1)\, P(s_1)$$

$$= P(m_1 \mid s_1)\, P(s_1) \cdot \prod_{i=2}^{n} P(m_i \mid s_i)\, P(s_i \mid s_{i-1})$$

Now, we want $P(s_k | m_{1:n})$

$$P(s_k | m_{1:n}) = P(s_k, m_{1:n}) = P(s_k\, m_{1:k}\, m_{k+1:n})$$

$$= P(m_{k+1:n} | s_k, m_{1:k}) \cdot P(s_k, m_{1:k})$$

$$= P(m_{k+1:n} | s_k) \cdot P(s_k | m_{1:k}) \cdot \cancel{P(m_{1:k})}$$

$$= P(m_{k+1:n} | s_k)\, P(s_k | m_{1:k})$$

$$= \underbrace{P(s_k | m_{1:k})}_{\text{Forward (online)}} \cdot \underbrace{P(m_{k+1:n} | s_k)}_{\text{Backward (offline)}}$$

→ Let's work at the forward component

$$P(s_k | m_{1:k}) = \frac{P(s_k, m_{1:k})}{P(m_{1:k})} \propto P(s_k, m_{1:k})$$

$$= \sum_{s_{k-1}} P(s_k, s_{k-1}, m_{1:k})$$

$$= \sum_{s_{k-1}} P(m_k | s_k\, s_{k-1}\, m_{1:k-1})\, P(s_k\, s_{k-1}\, m_{1:k-1})$$

$$= \underbrace{\sum_{s_{k-1}} P(m_k | s_k) \cdot P(s_k | s_{k-1}\, m_{1:k-1})\, P(s_{k-1}\, m_{1:k-1})}_{\text{call this } \alpha_k.}$$

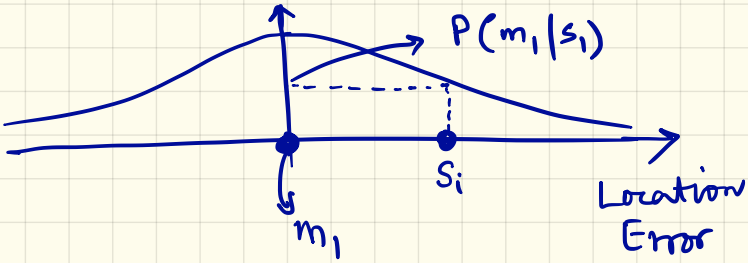
$$\therefore \quad \alpha_k = \sum_{s_{k-1}} P(m_k|s_k)\, P(s_k|s_{k-1})\, P(s_{k-1}, m_{1:k-1})$$

$$\alpha_k = \sum_{s_{k-1}} P(m_k|s_k)\, P(s_k|s_{k-1})\, \alpha_{k-1}$$

dynamic program

→ Initial condition $= \alpha_1 = P(m_1|s_1)\, P(s_1)$ should be known.

For instance, $P(s_1)$ might be equal prob. for all states, while $P(m_1|s_1)$ is a probability derived from sensor data sheet.



⊖ Backward part: $P(m_{k+1:n} \mid s_k)$

$$P(m_{k+1:n} \mid s_k) = \frac{P(m_{k+1:n}, s_k)}{P(s_k)}$$

$$= \frac{1}{P(s_k)} \sum_{s_{k+1}} P(m_{k+1:n}, s_k, s_{k+1})$$

$$= \frac{1}{P(s_k)} \sum_{s_{k+1}} P(m_{k+2:n} \mid s_{k+1}, s_k, m_{k+1}) \cdot P(m_{k+1} \mid s_{k+1}, s_k) \cdot P(s_{k+1} \mid s_k) P(s_k)$$

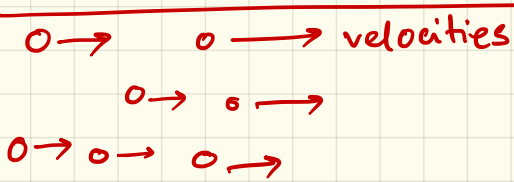
$$= \sum_{s_{k+1}} P(m_{k+2:n} \mid s_{k+1}) P(m_{k+1} \mid s_{k+1}) \cdot P(s_{k+1} \mid s_k)$$

$$\therefore \quad \beta_k = \sum_{s_{k+1}} \beta_{k+1} \cdot P(m_{k+1} \mid s_{k+1}) P(s_{k+1} \mid s_k)$$

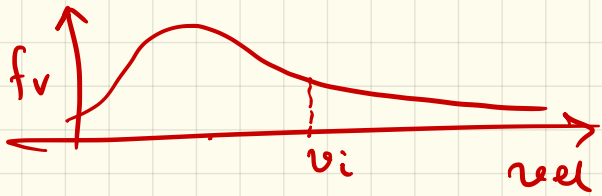
Known

Again dynamic program.

PARTICLE FILTER



Say given velocity Distribution



Step 1 Pick vel v_i for each particle i
Move the particle with this v_i.
Now, weigh the particle with $P(v_i)$ from the PDF.

Step 2 Measure the location and update each particles probabilities

$$\Psi_i = P(v_i)\, P(m_k \mid l_k) = \text{prob. of this location of the human} = \text{weight of this particle.}$$

Step 3 Resample : Redistribute N particles in each of the N locations based on Ψ_i, $\forall i \in [1, N]$

Step 4 Goto step 1

i.e., compute the transition to next step.

Kalman
- Error Gaussian
- Measurement is linear function

$$m = f(\text{location})$$

P.F works well for multi-modal, e.g., ZEE where particles disappear if gone through a wall!

HMM with ∞ particles = P.F.

P.F. continuous space.