TTIC 31170: Robot Learning and Estimation

Spring 2021

Matthew Walter
TTI-Chicago

Lecture 4: Recursive Bayesian Estimation (Cont.)

Problem Set 1

- **Due**: April 16 11:59pm
- **Electronic Submission via Gradescope**
- **Collaboration Policy**:
  - Everyone must write up and implement their own solutions
  - Must acknowledge who you collaborated with (if anyone)
- **Late Submission Policy**:
  - Everyone has a budget of 3 days that can be used to avoid penalties
  - Otherwise, 10% penalty for each late day (max three days)

Topics

- **HMMs (Continued)**
- State Representations
- Environment Interaction
- Probabilistic Generative Laws
- Belief Distributions
- Bayes Filters

Filtering

- Determine distribution over current state given *current* history of observations

$$P(X_t|Z_0, Z_1, \ldots, Z_t) = P(X_t|Z^t)$$

- We can compute this distribution from forward distribution

$$\begin{aligned} P(X_t = i|Z_0, Z_1, \ldots, Z_t) &= \frac{P(X_t = i, Z_0, Z_1, \ldots, Z_t)}{P(Z_0, Z_1, \ldots, Z_t)} \\ &= \frac{\alpha_t(i)}{\sum_j \alpha_t(j)} \end{aligned}$$

Smoothing

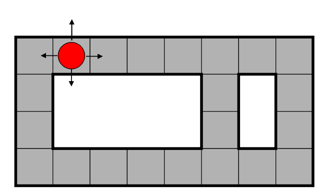
- Determine distribution over current state given *entire* history of observations

$$P(X_t|Z_0, Z_1, \ldots, Z_T) = P(X_t|Z^T)$$

- We can compute this distribution from forward distribution

$$\begin{aligned} P(X_t = i|Z_0, Z_1, \ldots, Z_T) &= \frac{P(X_t = i, Z_0, Z_1, \ldots, Z_T)}{P(Z_0, Z_1, \ldots, Z_T)} \\ &= \frac{\alpha_t(i)\beta_t(i)}{\sum_j \alpha_T(j)\beta_T(j)} \end{aligned}$$

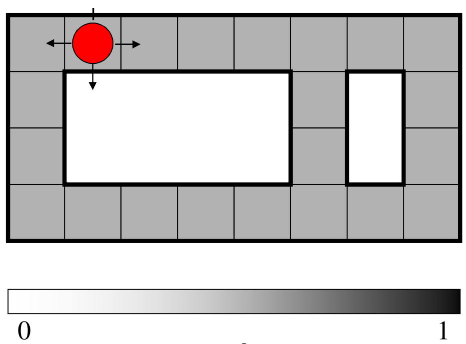
Filtering Example: Robot Localization



- Robot moves up, down, left, or right
- With small probability, robot won't move
- Sensors indicate whether there is a wall in front, back, left, and right
- With small probability, one sensor fails

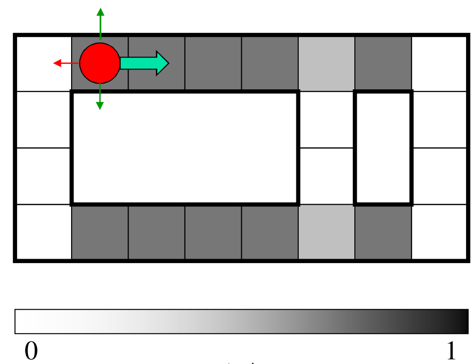
Example from MIchael Pfeiffer

Filtering Example: Robot Localization



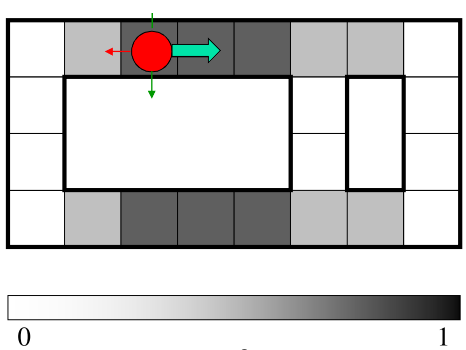
t=0

Filtering Example: Robot Localization



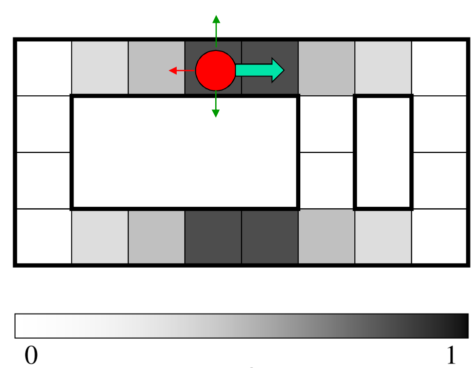
t=1

Filtering Example: Robot Localization



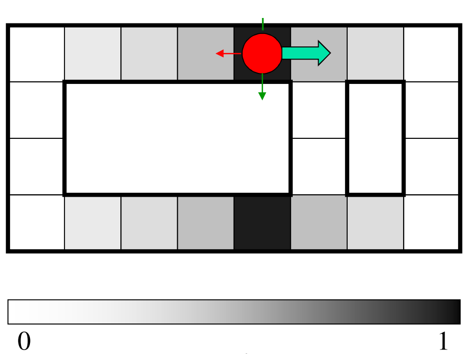
t=2

Filtering Example: Robot Localization



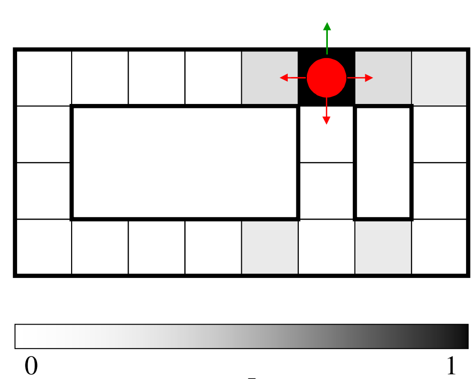
t=3

Filtering Example: Robot Localization



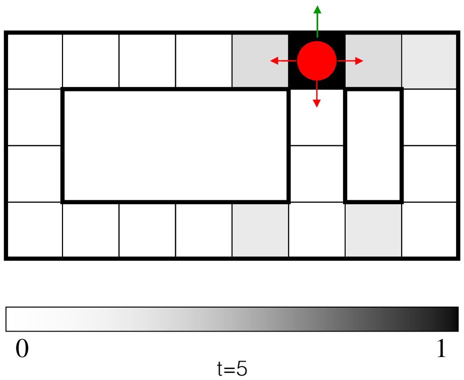
t=4

Filtering Example: Robot Localization



t=5

Filtering Example: Robot Localization



Example from MIchael Pfeiffer

Learning: HMM definition

- Formally define an HMM $\lambda = \{T, M, \pi\}$:
  - $\mathcal{X}$: finite set of states
  - $\mathcal{Z}$: finite set of observations
  - $T : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}_+$ (transition probabilities)
  - $M : \mathcal{X} \times \mathcal{Z} \rightarrow \mathbb{R}_+$ (observation/emission probabilities)
  - $\pi : \mathcal{X} \rightarrow \mathbb{R}_+$ (prior probability over initial state)

Learning

Suppose that we have a system $\{\mathcal{X}, \mathcal{Z}\}$ that can be modeled as an HMM $\lambda = \{T, M, \pi\}$. Given a set of observations Z^T, we would like to find the optimal parameter settings

$$\lambda^* = \arg\max_\lambda P(Z^T; \lambda)$$

Learning

Suppose that we have a system $\{\mathcal{X}, \mathcal{Z}\}$ that can be modeled as an HMM $\lambda = \{T, M, \pi\}$. Given a set of observations Z^T, we would like to find the optimal parameter settings

$$\begin{aligned}\lambda^* &= \arg\max_\lambda P(Z^T; \lambda) \\ &= \arg\max_\lambda \sum_{X^T} P(Z^T | X^T; \lambda) P(X^T; \lambda)\end{aligned}$$

Learning

Suppose that we have a system $\{\mathcal{X}, \mathcal{Z}\}$ that can be modeled as an HMM $\lambda = \{T, M, \pi\}$. Given a set of observations Z^T, we would like to find the optimal parameter settings

$$\begin{aligned}\lambda^* &= \arg\max_\lambda P(Z^T; \lambda) \\ &= \arg\max_\lambda \sum_{X^T} P(Z^T | X^T; \lambda) P(X^T; \lambda)\end{aligned}$$

- Challenges
  - Joint distribution doesn't factorize over t
  - Number of terms in summation is exponential in T

Learning

Suppose that we have a system $\{\mathcal{X}, \mathcal{Z}\}$ that can be modeled as an HMM $\lambda = \{T, M, \pi\}$. Given a set of observations Z^T, we would like to find the optimal parameter settings

$$\begin{aligned}\lambda^* &= \arg\max_\lambda P(Z^T; \lambda) \\ &= \arg\max_\lambda \sum_{X^T} P(Z^T | X^T; \lambda) P(X^T; \lambda)\end{aligned}$$

- Challenges
  - Joint distribution doesn't factorize over t
  - Number of terms in summation is exponential in T

- Instead, maximize likelihood via Expectation-Maximization algorithm

Learning

Given an HMM λ (transition & emission distributions) and an observation history Z^T, find a new HMM λ' that explains the observations at as well or better, i.e.,

$$P(Z^T; \lambda') \geq P(Z^T; \lambda)$$

- Ideally, we'd like to find the model that maximizes $P(Z^T; \lambda')$, however this is intractable in general
- Instead, we will be satisfied with a solution that converges to a local maxima of such probability
- Note, in order for learning to be effective, we need **lots of data**, i.e., **many, long** observation histories

Expectation of (state) counts

- Let us define

$$\gamma_t(i) = P(X_t = i | Z^T; \lambda)$$

i.e., the probability that the system is in state i at time t given observation history, according to the current HMM model

Expectation of (state) counts

- Let us define

$$\gamma_t(i) = P(X_t = i | Z^T; \lambda)$$

i.e., the probability that the system is in state i at time t given observation history, according to the current HMM model

- We already know how to compute this, i.e., via smoothing

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i \in \mathcal{X}} \alpha_t(i)\beta_t(i)}$$

Expectation of (state) counts

- Let us define

$$\gamma_t(i) = P(X_t = i | Z^T; \lambda)$$

i.e., the probability that the system is in state i at time t given observation history, according to the current HMM model

- We already know how to compute this, i.e., via smoothing

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i \in \mathcal{X}} \alpha_t(i)\beta_t(i)}$$

- Similarly, what is the expected number of visits to state i?

$$\mathbb{E}[\# \text{ of transitions from } i] = \sum_{t=0}^{T-1} \gamma_t(i)$$

Expectation of (transition) counts

- Similarly, define

$$\xi_t(i,j) = P(X_t = i, X_{t+1} = j | Z^T; \lambda)$$

i.e., the probability that the system is in state i at time t and state j at $t+1$ given observation history, according to the current HMM model

Expectation of (transition) counts

- Similarly, define

$$\xi_t(i,j) = P(X_t = i, X_{t+1} = j | Z^T; \lambda)$$

i.e., the probability that the system is in state i at time t and state j at $t+1$ given observation history, according to the current HMM model

- We can calculate this from forward and backward probabilities

$$\xi_t(i,j) \propto \alpha_t(i) P(X_{t+1} = j | X_t = i; \lambda) P(Z_{t+1} | X_{t+1} = j; \lambda) \beta_{t+1}(j)$$

Expectation of (transition) counts

- Similarly, define

$$\xi_t(i,j) = P(X_t = i, X_{t+1} = j | Z^T; \lambda)$$

i.e., the probability that the system is in state i at time t and state j at $t+1$ given observation history, according to the current HMM model

- We can calculate this from forward and backward probabilities

$$\xi_t(i,j) \propto \alpha_t(i) P(X_{t+1} = j | X_t = i; \lambda) P(Z_{t+1} | X_{t+1} = j; \lambda) \beta_{t+1}(j)$$

- Similarly, what is the expected number of visits to state i?

$$\mathbb{E}[\# \text{ of transitions from } i \text{ to } j] = \sum_{t=0}^{T-1} \xi_t(i,j)$$

Baum-Welch algorithm

- Based on probability estimates and expectations computed so far using the current HMM model $\lambda = \{T, M, \pi\}$, we can construct a new model $\lambda' = \{T', M', \pi'\}$

- The new prior over each state follows as

$$\pi'_i = \gamma_0(i)$$

- The entries in the new transition matrix can be estimated as follows

$$T'_{ji} = \frac{\mathbb{E}[\# \text{ transitions from } i \text{ to } j]}{\mathbb{E}[\# \text{ transitions from } i]} = \frac{\sum_{t=0}^{T-1} \xi_t(i,j)}{\sum_{t=0}^{T-1} \gamma_t(i)}$$

Baum-Welch algorithm

- The entries of the new observation matrix can be estimated as

$$M'_{mi} = \frac{\mathbb{E}[\# \text{ times in state } i \text{ and observing } m]}{\mathbb{E}[\# \text{ times in state } i]} = \frac{\sum_{t=0}^{T} \gamma_t(i) \cdot 1(Z_t = m)}{\sum_{t=0}^{T} \gamma_t(i)}$$

- Baum et al., 1970 showed that the new model λ' exhibits
  - $P(Z^T; \lambda') \geq P(Z^T; \lambda)$ as desired
  - $P(Z^T; \lambda') = P(Z^T; \lambda)$ only if λ is a critical point of the likelihood function $P(Z^T; \lambda)$

State

- The ***state*** is the parametrized representation expressing the properties of the world sufficient to impact the future
  - Communications: Binary sequence
  - Speech recognition: Words being uttered
  - Robot: Pose and environment

$$x^t = \{x_0, x_1, x_2, \ldots, x_t\}$$

switch to lowercase
for continuous variables

- The state is *complete* if it sufficiently describes the past

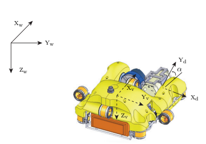
$X_0 = \text{“}wreck\text{”}$ $X_1 = \text{“}a\text{”}$ $X_2 = \text{“}nice\text{”}$ $X_3 = \text{“}beach\text{”}$

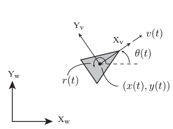
$X^T = \{X_0, X_1, X_2, \ldots, X_T\}$

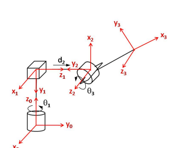
0010011001000101000001000011101101010100

State: Robot pose

- Most often continuous
- Ground robot: 2 Cartesian coordinates + heading
- Aerial/underwater robot: 3 Cartesian coordinates + roll, pitch, yaw
- Manipulator: Angles (revolute) and distances (prismatic) for each 1 DOF joint, specifying its configuration
- Optionally: Body-frame linear and angular velocities

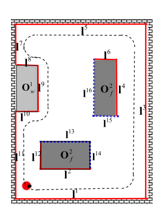




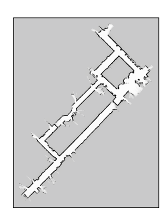


State: Environment

- Pose and properties of surrounding objects
  - Examples: Wall, tree, people, cars, etc.
  - Properties: Visual appearance, color, door open vs. closed
  - Static (landmarks) and dynamic (e.g., people)
  - May be continuous or discrete



Collection of walls parametrized by line segments



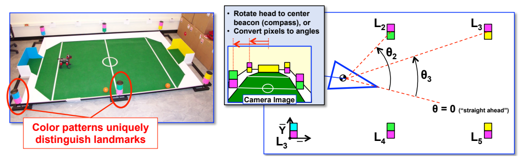
Binary occupancy or set of discrete cells

Environment interaction

- Two primary means of interaction between robot and environment
- **Control actions**: Change the state of the world
  - Robot motion
- **Sensor measurements**: Robot uses sensors (camera, LIDAR, tactile sensors) to extract information about (observe) the world

Environment interaction

- Two primary means of interaction between robot and environment
- **Control actions**: Change the state of the world
  - Robot motion
- **Sensor measurements**: Robot uses sensors (camera, LIDAR, tactile sensors) to extract information about (observe) the world



Environment interaction

- **Control data**: Convey information regarding change in state
  - Robot's velocity
  - Odometry

$$u^t = \{u_0, u_1, u_2, \ldots, u_t\} \qquad \text{where } \; u_t : (t-1; t]$$

- **Environment measurement data:** Convey information regarding the environment
  - Camera images
  - Range data

$$z^t = \{z_0, z_1, z_2, \ldots, z_t\}$$

Probabilistic generative laws

- Random variables
  - State: x_t
  - Measurements: z_t
  - Control: u_t
- Each is a stochastic process that evolves over time

$$p(x_t | x^{t-1}, z^{t-1}, u^t)$$

Probabilistic generative laws

- Random variables
  - State: x_t
  - Measurements: z_t
  - Control: u_t
- Each is a stochastic process that evolves over time

$$p(x_t | x^{t-1}, z^{t-1}, u^t)$$

- If x_{t-1} is complete, it is a sufficient statistic for z^{t-1} and u^{t-1}

$$p(x_t | x^{t-1}, z^{t-1}, u^t) = p(x_t | x_{t-1}, u_t)$$

Key to tractable inference

Probabilistic generative laws

- Random variables
  - State: x_t
  - Measurements: z_t
  - Control: u_t
- If x_{t-1} is complete, it is a sufficient statistic for z^{t-1} and u^{t-1}

$$p(x_t | x^{t-1}, z^{t-1}, u^t) = p(x_t | x_{t-1}, u_t)$$

- If x_t is complete

$$p(z_t | x^t, z^{t-1}, u^t) = p(z_t | x_t)$$

Probabilistic generative laws

- **State transition probability**: Describes how state evolves w/ time

$$p(x_t|x_{t-1}, u_t)$$

- **Measurement probability**: Probabilistic model by which measurements are generated

$$p(z_t|x_t)$$

- Describe the dynamics of the stochastic system

Belief distributions

- State can't be known exactly due to uncertainty
- **Definition**: The ***belief*** is the posterior probability distribution over the state, conditioned on history of measurement & control data

$$bel(x_t) = p(x_t|z^t, u^t)$$

Belief distributions

- State can't be known exactly due to uncertainty
- **Definition**: The ***belief*** is the posterior probability distribution over the state, conditioned on history of measurement & control data

$$bel(x_t) = p(x_t|z^t, u^t)$$

- The ***prediction*** is the posterior, without the previous measurement

$$\overline{bel}(x_t) = p(x_t|z^{t-1}, u^t)$$

Belief distributions

- State can't be known exactly due to uncertainty
- **Definition**: The ***belief*** is the posterior probability distribution over the state, conditioned on history of measurement & control data

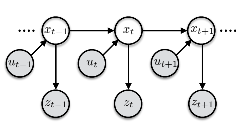
$$bel(x_t) = p(x_t|z^t, u^t)$$

- The ***prediction*** is the posterior, without the previous measurement

$$\overline{bel}(x_t) = p(x_t|z^{t-1}, u^t)$$

- The ***measurement update (correction)*** computes $bel(x_t)$ from $\overline{bel}(x_t)$

Inference over dynamic Bayesian networks



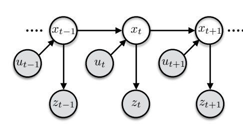
- **Filtering**: Estimate state posterior (belief) given data to-date
- **Prediction**: Estimate posterior over future state given data to-date

$$p(x_{t+k}|z^t, u^t) \text{ for } k > 0$$

- **Smoothing**: Estimate posterior over past state given data to-date

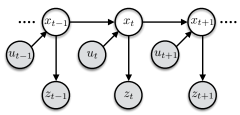
$$p(x_k|z^t, u^t) \text{ for } k < t$$

Bayes Filters



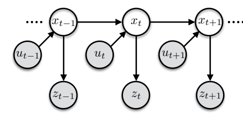
- **Given**:
  - Stream of measurement and control data
    $z^t = \{z_0, \dots, z_{t-1}, z_t\}$ $\quad u^t = \{u_0, \dots, u_{t-1}, u_t\}$
  - Measurement model: $p(z_t|x_t)$
  - Transition model: $p(x_t|x_{t-1}, u_t)$
  - Prior: $p(x_0)$
- **Desired**:
  - The belief (state posterior): $bel(x_t) = p(x_t|z^t, u^t)$

Bayes Filters



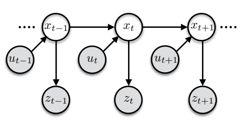
$$bel(x_t) = p(x_t | u_0, z_0, \ldots, u_t, z_t)$$

Bayes Filters



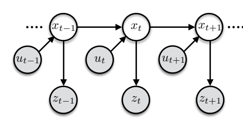
$$\begin{aligned} bel(x_t) &= p(x_t | u_0, z_0, \ldots, u_t, z_t) \\ \text{Bayes rule} \quad &= \eta\, p(z_t | x_t, u_0, z_0, \ldots, u_t)\, p(x_t | u_0, z_0, \ldots, u_t) \end{aligned}$$

Bayes Filters



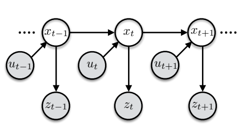
$$\begin{aligned} bel(x_t) &= p(x_t | u_0, z_0, \ldots, u_t, z_t) \\ \text{Bayes rule} \quad &= \eta\, p(z_t | x_t, u_0, z_0, \ldots, u_t)\, p(x_t | u_0, z_0, \ldots, u_t) \\ \text{C.I.} \quad &= \eta\, p(z_t | x_t)\, p(x_t | u_0, z_0, \ldots, u_t) \end{aligned}$$

Bayes Filters



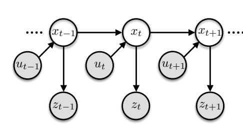
$$\begin{aligned} bel(x_t) &= p(x_t | u_0, z_0, \ldots, u_t, z_t) \\ \text{Bayes rule} \quad &= \eta\, p(z_t | x_t, u_0, z_0, \ldots, u_t)\, p(x_t | u_0, z_0, \ldots, u_t) \\ \text{C.I.} \quad &= \eta\, p(z_t | x_t)\, p(x_t | u_0, z_0, \ldots, u_t) \\ \text{Total prob.} \quad &= \eta\, p(z_t | x_t) \int p(x_t | x_{t-1}, u_0, z_0, \ldots, u_t)\, p(x_{t-1} | u_0, z_0, \ldots, u_t)\, dx_{t-1} \end{aligned}$$

Bayes Filters



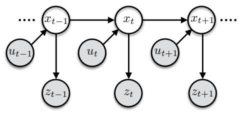
$$\begin{aligned} bel(x_t) &= p(x_t | u_0, z_0, \ldots, u_t, z_t) \\ \text{Bayes rule} \quad &= \eta\, p(z_t | x_t, u_0, z_0, \ldots, u_t)\, p(x_t | u_0, z_0, \ldots, u_t) \\ \text{C.I.} \quad &= \eta\, p(z_t | x_t)\, p(x_t | u_0, z_0, \ldots, u_t) \\ \text{Total prob.} \quad &= \eta\, p(z_t | x_t) \int p(x_t | x_{t-1}, u_0, z_0, \ldots, u_t)\, p(x_{t-1} | u_0, z_0, \ldots, u_t)\, dx_{t-1} \\ \text{Markov} \quad &= \eta\, p(z_t | x_t) \int p(x_t | x_{t-1}, u_t)\, p(x_{t-1} | u_0, z_0, \ldots, u_t)\, dx_{t-1} \end{aligned}$$

Bayes Filters



$$\begin{aligned} bel(x_t) &= p(x_t | u_0, z_0, \ldots, u_t, z_t) \\ \text{Bayes rule} \quad &= \eta\, p(z_t | x_t, u_0, z_0, \ldots, u_t)\, p(x_t | u_0, z_0, \ldots, u_t) \\ \text{C.I.} \quad &= \eta\, p(z_t | x_t)\, p(x_t | u_0, z_0, \ldots, u_t) \\ \text{Total prob.} \quad &= \eta\, p(z_t | x_t) \int p(x_t | x_{t-1}, u_0, z_0, \ldots, u_t)\, p(x_{t-1} | u_0, z_0, \ldots, u_t)\, dx_{t-1} \\ \text{Markov} \quad &= \eta\, p(z_t | x_t) \int p(x_t | x_{t-1}, u_t)\, p(x_{t-1} | u_0, z_0, \ldots, u_t)\, dx_{t-1} \\ &= \eta\, p(z_t | x_t) \int p(x_t | x_{t-1}, u_t)\, bel(x_{t-1})\, dx_{t-1} \end{aligned}$$

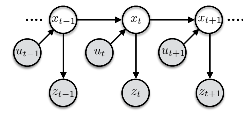
Bayes Filters



$$bel(x_t) = \eta\, p(z_t|x_t) \int p(x_t|x_{t-1}, u_t)\, bel(x_{t-1})\, dx_{t-1}$$

Algorithm Bayes_filter($bel(x_{t-1}), u_t, z_t$)
for all x_t do
$\overline{bel}(x_t) = \int p(x_t|x_{t-1}, u_t)\, bel(x_{t-1})\, dx_{t-1}$
$bel(x_t) = \eta\, p(z_t|x_t)\, \overline{bel}(x_t)$
endfor
return $bel(x_t)$

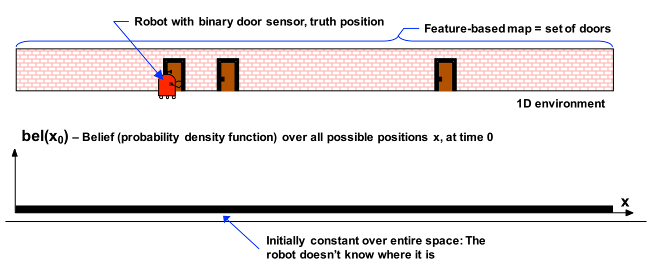
Bayes Filters



$$bel(x_t) = \eta\, p(z_t|x_t) \int p(x_t|x_{t-1}, u_t)\, bel(x_{t-1})\, dx_{t-1}$$

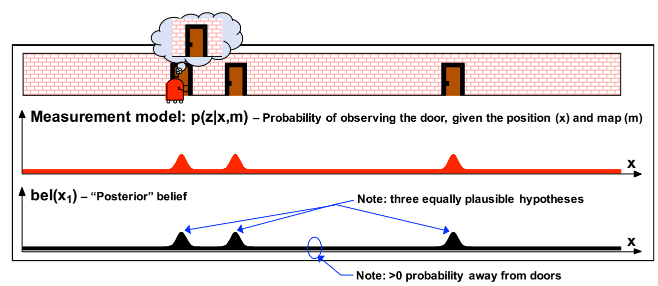
Algorithm Bayes_filter($bel(x_{t-1}), u_t, z_t$)
for all x_t do
Prediction $\overline{bel}(x_t) = \int p(x_t|x_{t-1}, u_t)\, bel(x_{t-1})\, dx_{t-1}$
Measurement update $bel(x_t) = \eta\, p(z_t|x_t)\, \overline{bel}(x_t)$
endfor
return $bel(x_t)$

Robot localization (a teaser)



Robot localization (a teaser)

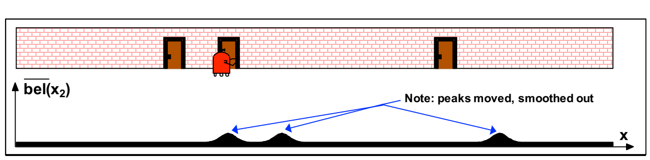
1) Robot activates it's door sensor, sees it is near a door



Robot localization (a teaser)

2) Robot moves down the hallway

3) Prediction: Incorporate uncertainty in robot motion

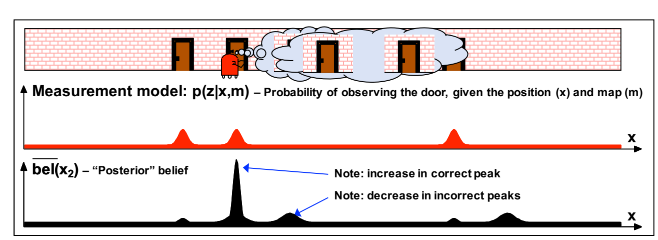


Prediction step (robot motion) reduces robot's confidence in its position belief

Robot localization (a teaser)

4) Robot activates its door sensor and sees a door

5) Measurement update: Update belief based upon measurement

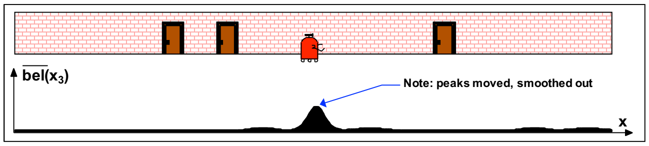


Measurement step improves robot's confidence in its position belief

Robot localization (a teaser)

6) Robot moves down the hallway

7) Prediction: Incorporate uncertainty in robot motion



Bayes Filters

$$bel(x_t) = \eta\, p(z_t|x_t) \int p(x_t|x_{t-1}, u_t)\, bel(x_{t-1})\, dx_{t-1}$$

- Kalman filters
- Particle filters
- Hidden Markov models
- Partially Observable Markov Decision Processes (POMDPs)