Autonomous Robotics

Bayes Filter

Josiah Hanna
University of Wisconsin — Madison

Reading Responses

- Overall, great responses. Grades are published.
- If you submitted, then you likely received most of the 10 possible points.
  - Sometimes, points were lost when responses lacked detail demonstrating the text had been read.
- Reading on Kalman filter is now available on the course website.

Programming Assignments

- Due Tuesday (2/11) at 9:30am
- Any questions?
- Any comments?

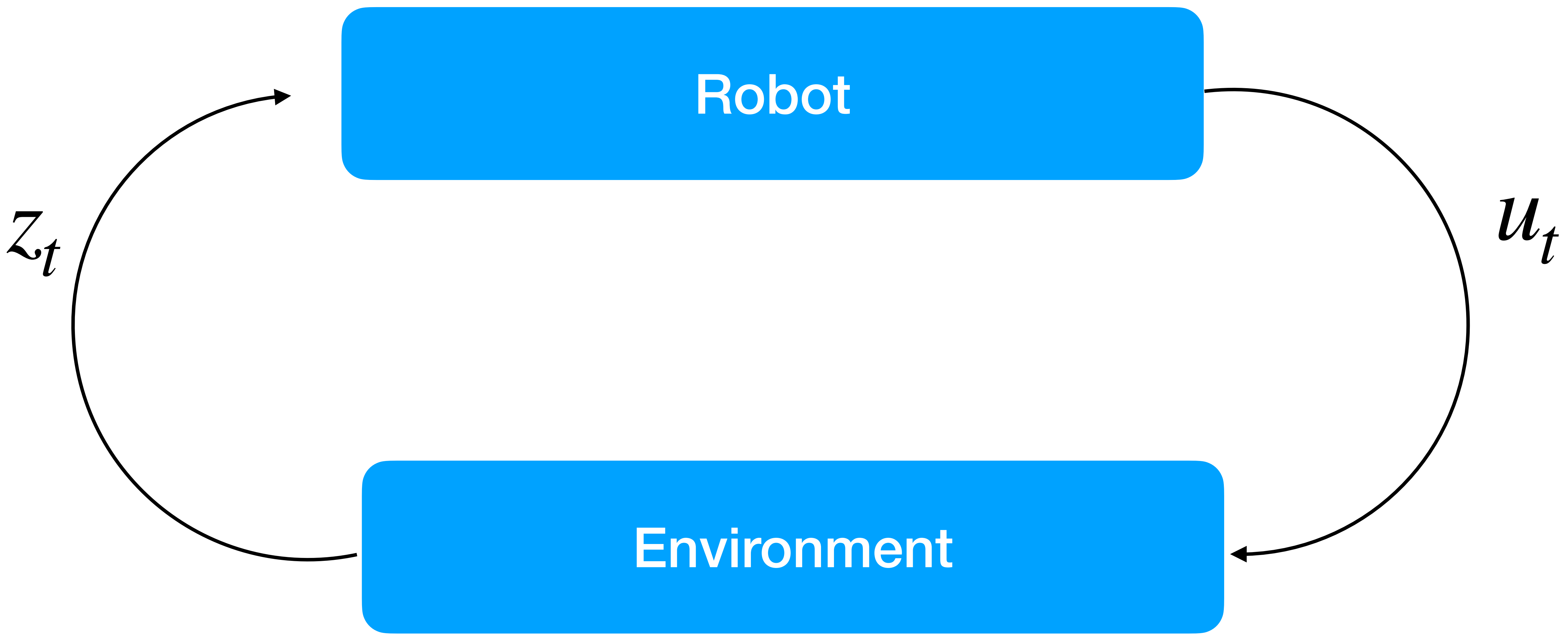
Learning Outcomes

After today's lecture, you will:

- Be able to implement the Bayes filter algorithm for recursive state estimation.
- Explain the strengths and weaknesses of the Bayes filter for real robot systems.
- Explain the difference between filtering and smoothing.

Probabilistic Interaction Model

$$\texttt{bel}(x_t) = p(x_t \mid z_{1:t}, u_{1:t})$$



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

$$z_t \sim g(x_t)$$

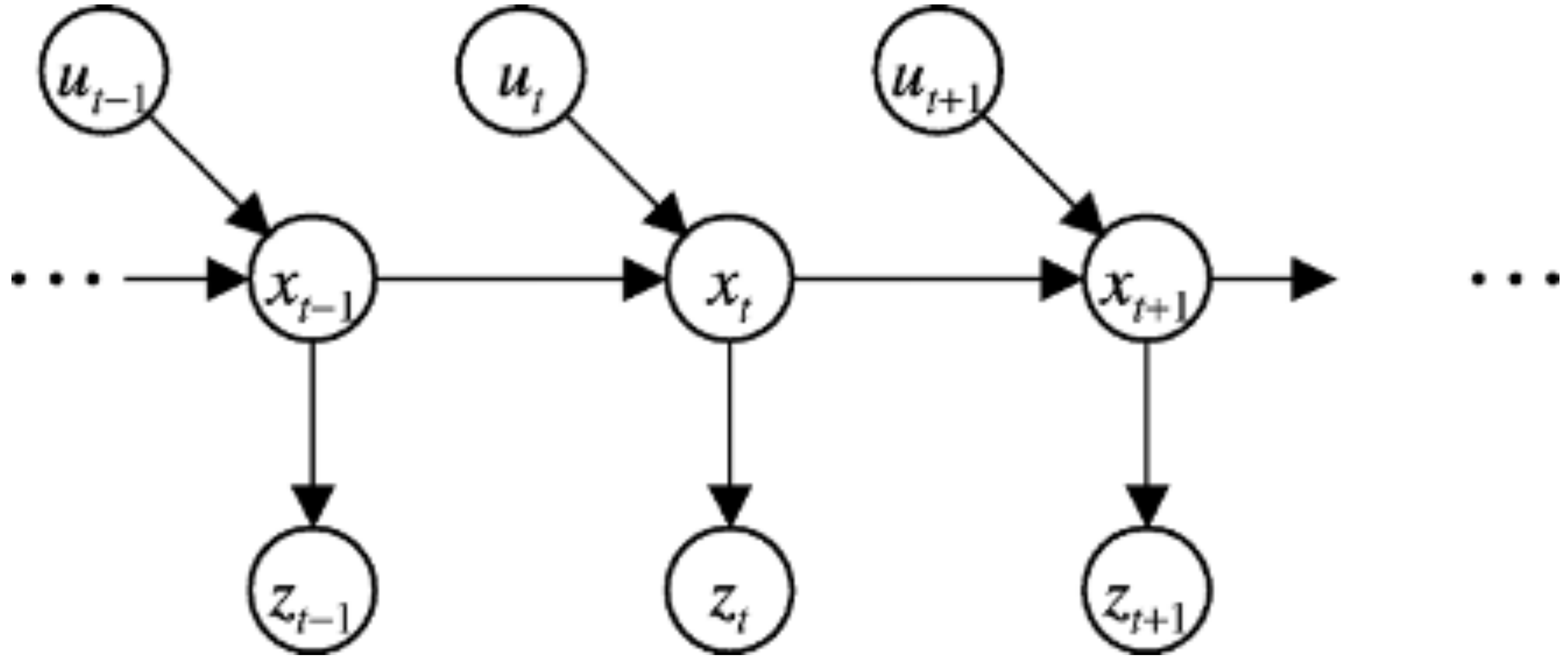
$g(z \mid x)$ is the probability of z given x.

$p(x_t \mid x_{t-1}, u_t)$ is the probability of x_t given the environment is in state x_{t-1} and control u_t is taken.

States

- State: all variables in a robot's environment that impact the future.
  - You have to define in your implementation.
  - State variables can be continuous. Example: robot pose $(x, y, \theta) \in \mathbf{R}^3$.
  - State variables can be discrete. Example: $\texttt{door} \in \{\texttt{open}, \texttt{closed}\}$
- Assumptions:
  - We know the state space, full set of possible states.
  - Markov assumption: $p(x_t \mid x_{t-1}, u_t) = p(x_t \mid x_{0:t-1}, u_{1:t})$

Dependency Graph



- Directed graphical model: nodes are variables and edges represent direct dependencies.
- Graph enables easy checking of conditional independence.
  - Two nodes, A and B are conditionally independent given C if the node for C blocks all directed paths from A to B.*

*** Represents a simplification of conditional independence check.**

Fantastic models and where to find them

- State estimation requires models of the state transition and observation functions, $p(x_t \mid x_{t-1}, u_t)$ and $g(z_t \mid x_t)$.
- In practice, you have to model these using data and/or knowledge of physics.
- Poor modeling $\approx$ poor state estimation.

Fantastic models and where to find them

- One approach: use machine learning and data $D = (x_{0:T}, u_{1:T}, z_{1:T})$.
  - Choose a set of candidate models, $\mathscr{P}$. Example: neural networks.
  - Select most likely $p \in \mathscr{P}$, given D.

$$p \leftarrow \arg\max_{p'} \sum_{t=1}^{T} \log p'(x_t \,|\, x_{t-1}, u_t)$$

  - Similarly, for the observation model:

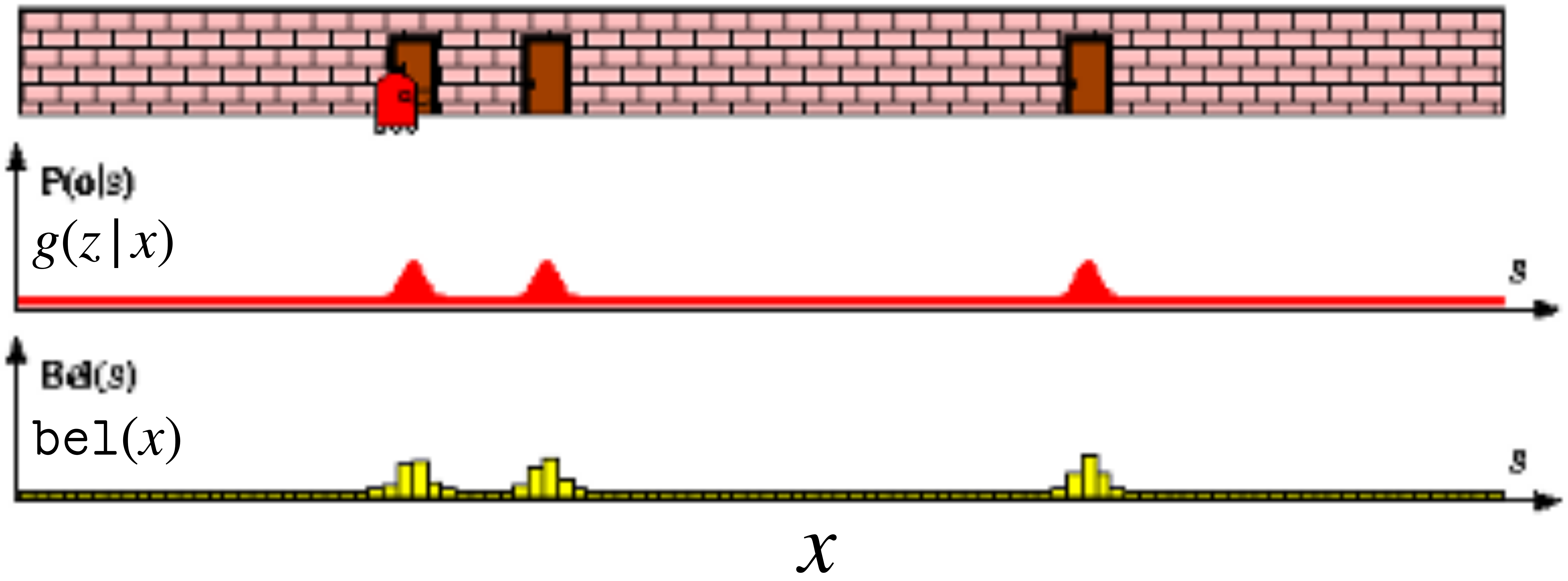
$$g \leftarrow \arg\max_{g'} \sum_{t=1}^{T} \log g'(z_t \,|\, x_t)$$

  - What could be a problem here?

State Estimation

- Define robot's belief as $\texttt{bel}(x_t)$.
- At time t, the robot has observed $z_{1:t}$ and knows it has taken actions $u_{1:t}$.
- **Goal:** Compute the posterior such that $\texttt{bel}(x_t) = p(x_t \mid z_{1:t}, u_{1:t})$.
  - Computation should not grow with t.
  - Why necessary?
- Assumptions: know models p and g, have initial belief $\texttt{bel}(x_0)$.

Illustration



Naive Approach

of terms grows exponentially!!

Bayes rule on the full sequence:

$$p(x_t \mid z_{1:t}, u_{1:t}) = \sum_{x_{1:t-1}} p(x_{1:t} \mid z_{1:t}, u_{1:t})$$

Expand the inside of the summation:

$$p(x_{1:t} \mid z_{1:t}, u_{1:t}) = \eta \cdot p(x_{1:t}, z_{1:t} \mid u_{1:t}) = \eta \cdot p(z_{1:t} \mid x_{1:t}) p(x_{1:t} \mid u_{1:t})$$

$$p(z_{1:t} \mid x_{1:t}) = \prod_{i=1}^{t} g(z_i \mid x_i) \text{ and } p(x_{1:t} \mid u_{1:t}) = \prod_{i=1}^{t} p(x_i \mid x_{i-1}, u_i).$$

Bayes Filter

Discrete States | **Continuous States**

- **Predict:**

Discrete:

$$\overline{\texttt{bel}}(x_t) \leftarrow \sum_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\texttt{bel}(x_{t-1})$$

Continuous:

$$\overline{\texttt{bel}}(x_t) \leftarrow \int_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\texttt{bel}(x_{t-1})dx_{t-1}$$

- **Correct:**

$$\texttt{bel}(x_t) \leftarrow \eta^{-1} g(z_t \mid x_t)\overline{\texttt{bel}}(x_t)$$

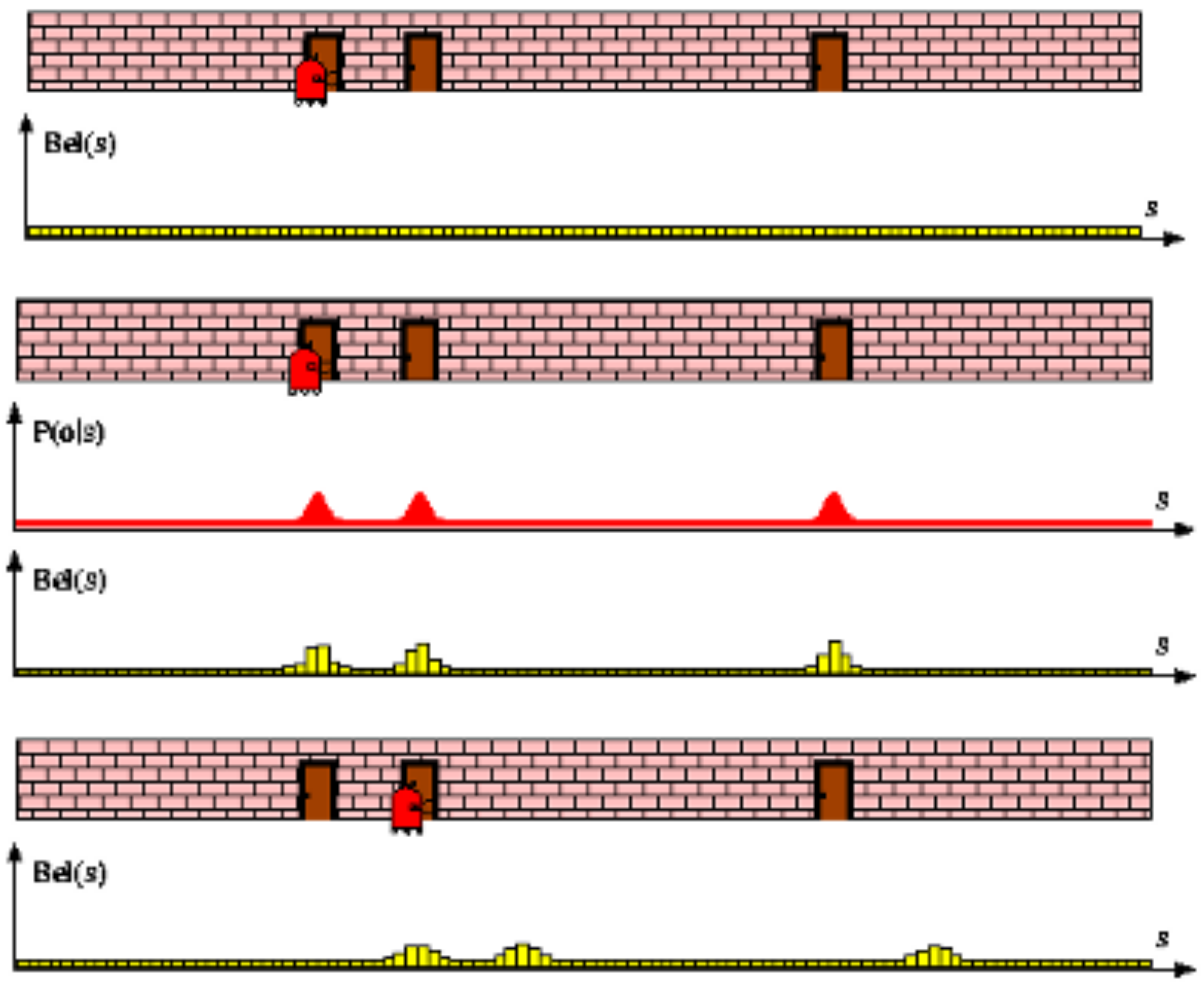
Discrete:

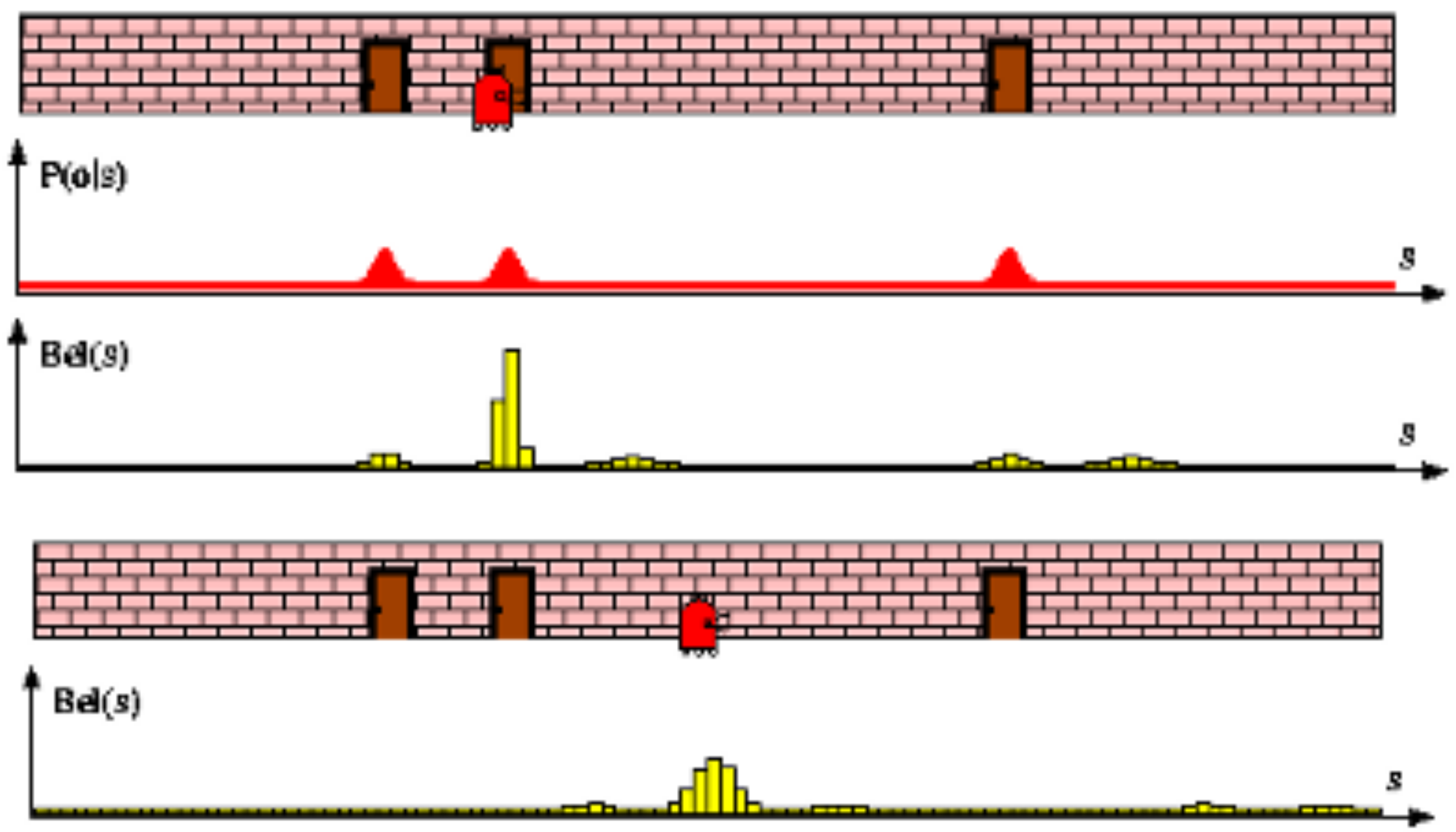
$$\eta = \sum_{x_t} g(z_t \mid x_t)\overline{\texttt{bel}}(x_t)$$

Continuous:

$$\eta = \int x_t g(z_t \mid x_t)\overline{\texttt{bel}}(x_t)dx_t$$

Bayes Filter Example





Prediction increases uncertainty; Correction step decreases uncertainty.

Credit: Probabilistic Robotics

Bayes Filter Derivation

- $\texttt{bel}(x_t) = p(x_t \mid z_{1:t}, u_{1:t})$
- $= \eta p(z_t \mid x_t, z_{1:t-1}, u_{1:t}) p(x_t \mid z_{1:t-1}, u_{1:t})$ **Bayes Rule**
- $= \eta p(z_t \mid x_t) p(x_t \mid z_{1:t-1}, u_{1:t})$ **Markov Assumption**
- $= \eta p(z_t \mid x_t) \int p(x_t \mid z_{1:t-1}, u_{1:t}, x_{t-1}) p(x_{t-1} \mid z_{1:t-1}, u_{1:t}) dx_{t-1}$ **Law of total probability**
- $= \eta p(z_t \mid x_t) \int p(x_t \mid x_{t-1}, u_t) p(x_{t-1} \mid z_{1:t-1}, u_{1:t}) dx_{t-1}$ **Markov Assumption**
- $= \eta p(z_t \mid x_t) \int p(x_t \mid x_{t-1}, u_t) p(x_{t-1} \mid z_{1:t-1}, u_{1:t-1}) dx_{t-1}$ **Markov Assumption**

Limitations

- Intractable in general case. Why?
  - Summation/integration over state space.
  - Special cases can be tractable, e.g., Kalman Filter assumes linear Gaussian models.
  - Approximation possible, e.g., Extended KF, Particle Filter

Practical Strategies

- Use special cases (more on this next week).
- Simplify state representation but watch out for Markov violations!
  - Discretize continuous states
  - Dimensionality reduction

Smoothing

- **What is it?**
  - Use new observations to refine past beliefs.
  - Expect $p(x_t \,|\, z_{1:T}, u_{1:T})$ for $T > t$ to be more reliable than $p(x_t \,|\, z_{1:t}, u_{1:t})$.
- **Why?**
  - Future observations provide information about the past.
  - Can use for offline map estimation or machine learning.

Bayes Smoother

- Forward pass: Bayes filter over the data. Get $\texttt{bel}(x_t)$ for $t \in \{1,\ldots,T\}$.
- Backward (smoothing) pass:
  - $\texttt{bel}'(x_T) \leftarrow \texttt{bel}(x_T)$

    The belief at the final time-step is just the belief from the filter.

  - Work backwards in time from T to 1 using the update:

$$\texttt{bel}'(x_t) \leftarrow \texttt{bel}(x_t) \cdot \sum_{x_{t+1}} \frac{p(x_{t+1} \mid x_t, u_{t+1}) \cdot \texttt{bel}'(x_{t+1})}{\texttt{bel}(x_{t+1})}$$

Belief from filtering

Summary

- Introduced problem of state estimation.
- Introduced Bayes filter as a method for state estimation.
- Discussed limitations of Bayes filter.
- Introduced Bayes smoother.

Action Items

- Complete the first programming assignment on control.
- Read on Kalman filter for next week; send a reading response by 12 pm on Monday.