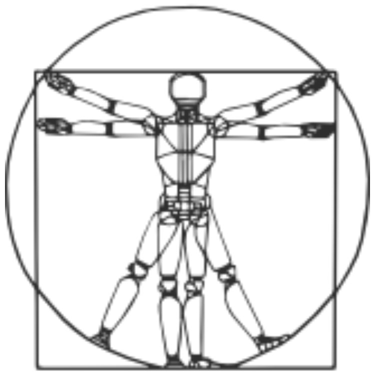
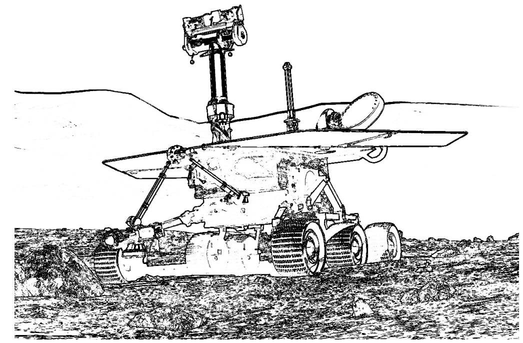




COMP417
Introduction to Robotics and Intelligent Systems

State Estimation, Localization and the Kalman Filter

McGill

MRL Mobile Robotics Lab at McGill University

Fundamental Problems In Robotics

- How to Go From A to B ? (Path Planning)
- What does the world looks like? (mapping)
  - sense from various positions
  - integrate measurements to produce map
  - assumes perfect knowledge of position
- Where am I in the world? (localization)
  - Sense
  - relate sensor readings to a world model
  - compute location relative to model
  - assumes a perfect world model
- Together, the above two are called SLAM
  (Simultaneous Localization and Mapping)

Localization

- Tracking: Known initial position
- Global Localization: Unknown initial position
- Re-Localization: Incorrect known position
  - (kidnapped robot problem)

Uncertainty

- Central to any real system!

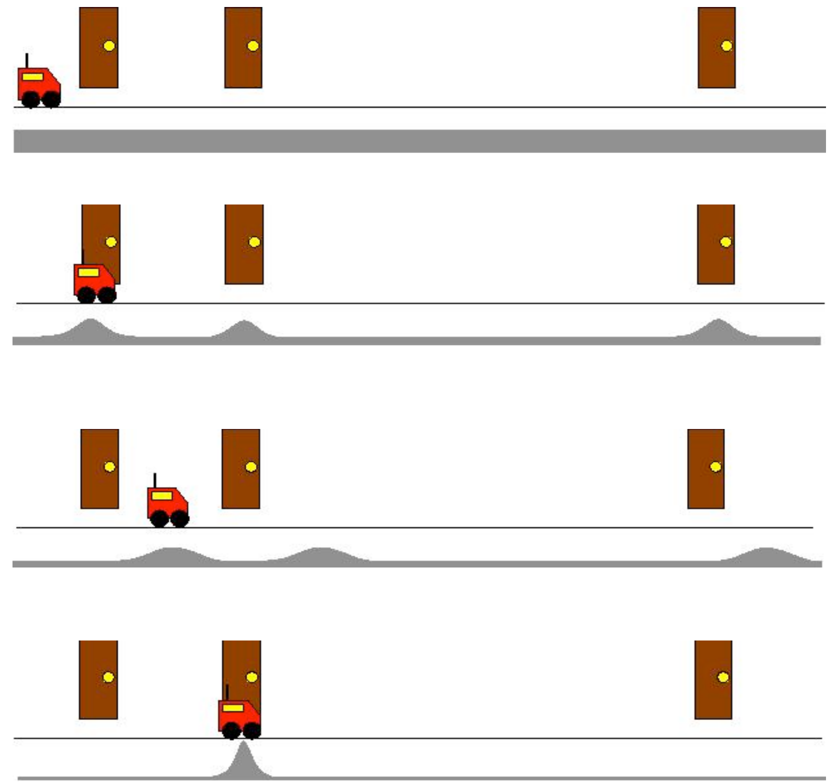
Localization

Initial state
detects nothing:

Moves and
detects landmark:

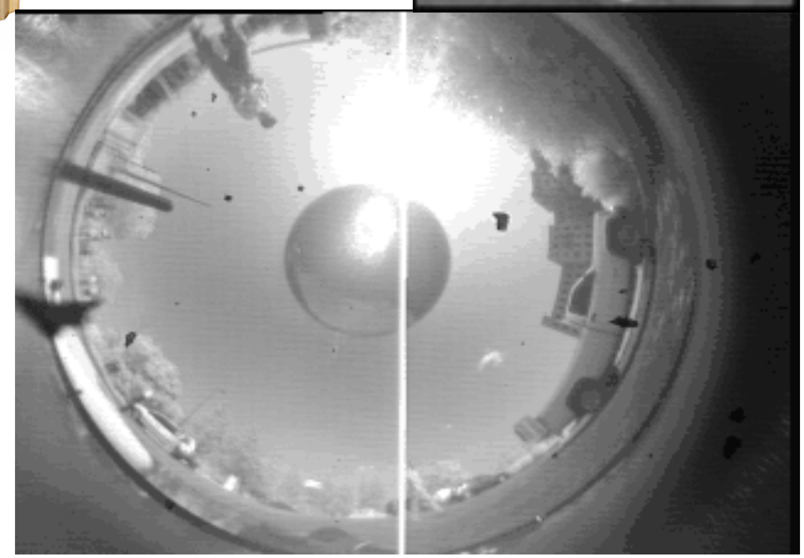
Moves and
detects nothing:

Moves and
detects landmark:



Sensors

- **Proprioceptive Sensors**
  (monitor state of vehicle-propagate)
  - IMU (accels & gyros)
  - Wheel encoders
  - Doppler radar …
    - **Noise**
- **Exteroceptive Sensors**
  (monitor environment-update)
  - Cameras (single, stereo, omni, FLIR …)
  - Laser scanner
  - MW radar
  - Sonar
  - Tactile…
    - **Uncertainty**

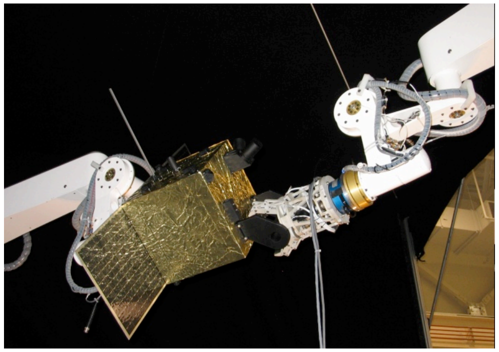


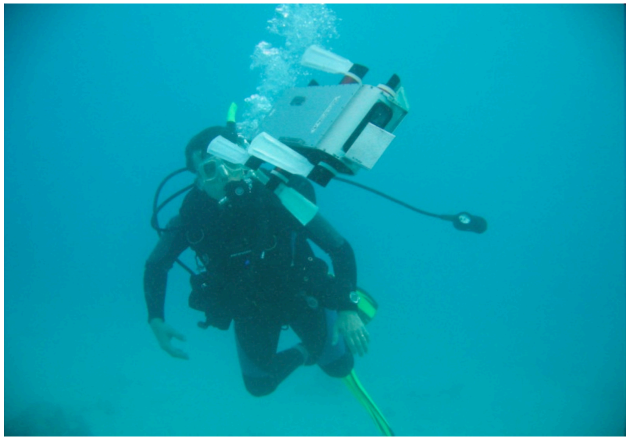
Bayesian Filter

- "**Filtering**" is a name for combining data.
- Nearly all algorithms that exist for spatial reasoning make use of this approach
  - If you're working in robotics, you'll see it over and over!
- Efficient state estimators
  - Recursively compute the robot's current state based on the previous state of the robot

State Estimation

- *What is the robot's state?*
- Depends on the robot
  - Indoor mobile robot
    - $\boldsymbol{x}$=[x, y, θ]
  - 6DOF mobile vehicle
    - $\boldsymbol{x}$=[x, y, z, φ, ψ, θ]
  - Manipulators





Bayesian Filter

- Estimate state ***x*** from data *Z*
  - *What is the probability of the robot being at x?*
- ***x*** could be robot location, map information, locations of targets, etc…
- *Z* could be sensor readings such as range, actions, odometry from encoders, etc…)
- This is a general formalism that does not depend on the particular probability representation
- Bayes filter **recursively** computes the posterior distribution:

$$Bel(x_T) = P(x_T \mid Z_T)$$

Derivation of the Bayesian Filter

Estimation of the robot's state given the data:

$$Bel(x_t) = p(x_t \mid Z_T)$$

The robot's data, Z, is expanded into two types: observations o_i and actions a_i

$$Bel(x_t) = p(x_t \mid o_t, a_{t-1}, o_{t-1}, a_{t-2}, \ldots, o_0)$$

Invoking the Bayesian theorem

$$Bel(x_t) = \frac{p(o_t \mid x_t, a_{t-1}, \ldots, o_0)\, p(x_t \mid a_{t-1}, \ldots, o_0)}{p(o_t \mid a_{t-1}, \ldots, o_0)}$$

Recommended reading

- Chapters 2 and 3.2 from Probabilistic Robotics
- Chapters 4.9 and 8.3 from Computational Principles of Mobile Robotics
- Lesson 2 in https://www.udacity.com/course/artificial-intelligence-for-robotics--cs373
- This illustrative blog post:
  http://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/
  Careful: the figure between equations (9) and (10) is wrong. The blue Gaussian should be taller and peakier than the other two Gaussians, the prior and the measurement models. This is not fixed as of March 15, 2017.

Filtering vs. Smoothing

- Smoothing/Batch Estimation

$$p(\mathbf{x}_{0:T} \mid \mathbf{z}_{0:T}, \mathbf{u}_{0:T-1})$$

Sequence of states

Sequence of sensor measurements

Sequence of commands

- Filtering Estimation

$$p(\mathbf{x}_t \mid \mathbf{z}_{0:t}, \mathbf{u}_{0:t-1})$$

What's the difference?

- Smoothing/Batch Estimation

$$p(\mathbf{x}_{0:T} \mid \mathbf{z}_{0:T}, \mathbf{u}_{0:T-1})$$

All measurements and controls are known in advance

- Filtering Estimation

$$p(\mathbf{x}_t \mid \mathbf{z}_{0:t}, \mathbf{u}_{0:t-1})$$

Measurements and controls are processed online as they come: called a **recursive filter.** Future measurements are unknown.

Why do we use filtering?

- Online **belief** updates: **filters** provide a principled way to incorporate noisy information from sensor measurements, which can change our prior belief, in an online fashion.
  - Recursive filters to it efficiently

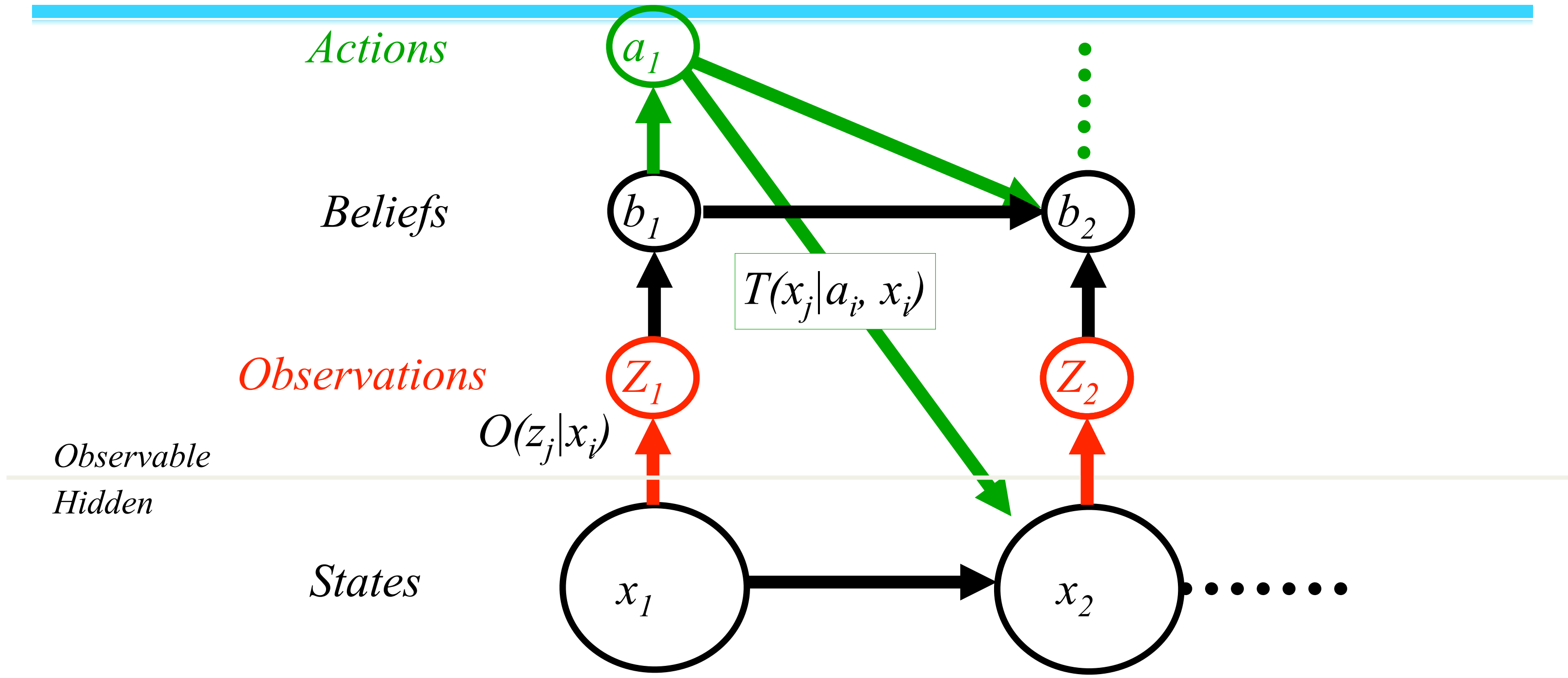
- Sensor fusion: filters enable us to combine measurements from multiple different noisy sensors into one coherent state estimate. E.g. camera + laser, camera + IMU, multiple cameras, sonar and IMU, GPS and IMU etc.

Technically speaking, this is also true for smoothing estimators.

Probabilistic (Bayes') Filter

- A generic class of filters that make use of rules of probability:
  - **Markov Assumption: known only limited amount of recent history is enough,**
  - **Static World Assumption:** the current observation depends only on the current state and the fixed map
- Now let's elaborate this...

Graphical Models, Bayes' Rule and the Markov Assumption



Bayes rule : $p(x \mid y) = \dfrac{p(y \mid x)\,p(x)}{p(y)}$

Markov : $p(x_t \mid x_{t-1}, a_t, a_0, z_0, a_1, z_1, \ldots, z_{t-1}) = p(x_t \mid x_{t-1}, a_t)$

Bayes' Filter

- A generic class of filters that make use of Bayes' rule and assume the following:
  - **Markov Assumption For Dynamics**: the state x_t is conditionally independent of past states and controls, given the previous state x_{t-1}. In other words, the dynamics model is assumed to satisfy

$$p(x_t|x_{0:t-1}, u_{0:t-1}) = p(x_t|x_{t-1}, u_{t-1})$$

  - **Static World Assumption:** the current observation is conditionally independent of past observations and controls, given the current state

$$p(z_t|x_t, u_{0:t-1}, z_{0:t-1}) = p(z_t|x_t)$$

Bayes' Filter

- A generic class of filters that make use of Bayes' rule and assume the following:
  - <u>**Markov Assumption**</u> **For Dynamics**: the state x_t is conditionally independent of past states and controls, given the previous state x_{t-1}. In other words, the dynamics model is assumed to satisfy

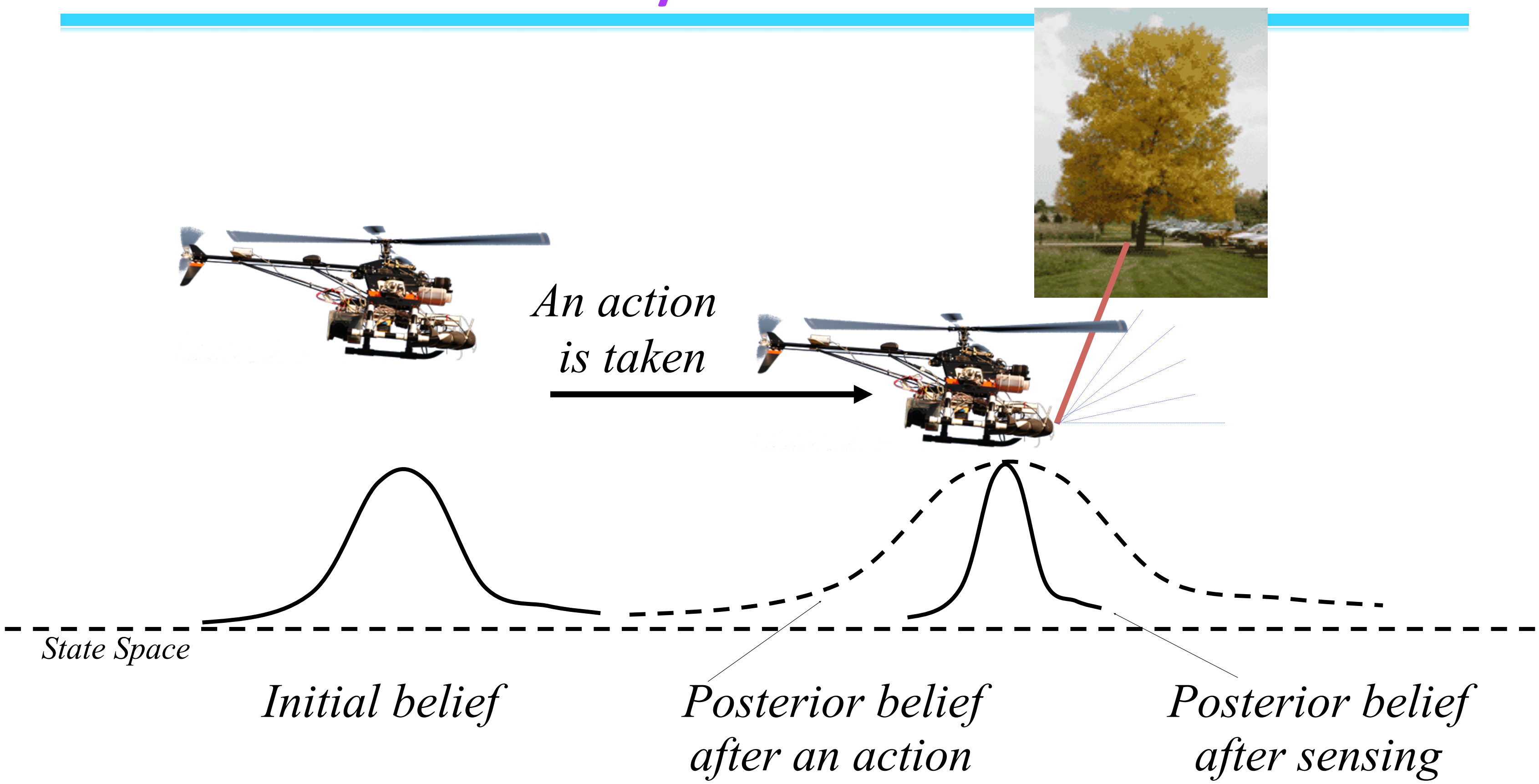
$$p(x_t|x_{0:t-1}, u_{0:t-1}) = p(x_t|x_{t-1}, u_{t-1})$$

  - **Static World Assumption:** the current observation is conditionally independent of past observations and controls, given the current state

$$p(z_t|x_t, u_{0:t-1}, z_{0:t-1}) = p(z_t|x_t)$$

Note: the Markov assumption is more general than what we have presented here.

Bayes Filter



Bayes' Filter: Derivation

$$
\begin{aligned}
bel(x_t) &= p(x_t|u_{0:t-1}, z_{0:t}) \\
&= \eta \; p(z_t|x_t, u_{0:t-1}, z_{0:t-1}) \; p(x_t|u_{0:t-1}, z_{0:t-1})
\end{aligned}
$$

Normalizing factor that makes the integral/sum of the numerator in Bayes' Rule be 1.

Conditional Bayes' Rule

$$
\underbrace{p(\overbrace{A}^{x_t}|\overbrace{B}^{u_{0:t-1},\, z_{0:t-1}}, \overbrace{C}^{z_t})}_{} = \frac{p(C|A,B)\; p(A|B)}{\underbrace{p(C|B)}_{1/\eta}}
$$

Bayes' Filter: Derivation

$$
\begin{aligned}
bel(x_t) &= p(x_t | u_{0:t-1}, z_{0:t}) \\
&= \eta \, p(z_t | x_t, \boxed{u_{0:t-1}, z_{0:t-1}}) \, p(x_t | u_{0:t-1}, z_{0:t-1}) \\
&= \eta \, p(z_t | x_t) \, p(x_t | u_{0:t-1}, z_{0:t-1})
\end{aligned}
$$

Static World Assumption

Bayes' Filter: Derivation

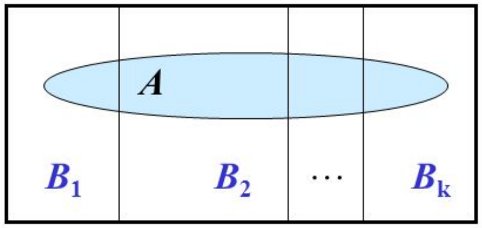
$$
\begin{aligned}
bel(x_t) &= p(x_t|u_{0:t-1}, z_{0:t}) \\
&= \eta\ p(z_t|x_t, u_{0:t-1}, z_{0:t-1})\ p(x_t|u_{0:t-1}, z_{0:t-1}) \\
&= \eta\ p(z_t|x_t)\ p(x_t|u_{0:t-1}, z_{0:t-1}) \\
&= \eta\ p(z_t|x_t)\ \int p(x_t, x_{t-1}|u_{0:t-1}, z_{0:t-1})\ dx_{t-1}
\end{aligned}
$$

All the ways I can get here from all places I might have been

Marginalization, or law of total probability

$$p(A) = \sum_{B_i} p(A, B_i)$$

where the sum enumerates all possibilities over the variable Bi. If we see Bi as a set, then the collection of Bi's must be pairwise disjoint. I.e. the collection of subsets Bi must be a partition of the sample space.



Bayes' Filter: Derivation

$$
\begin{aligned}
bel(x_t) &= p(x_t|u_{0:t-1}, z_{0:t}) \\
&= \eta\, p(z_t|x_t, u_{0:t-1}, z_{0:t-1})\, p(x_t|u_{0:t-1}, z_{0:t-1}) \\
&= \eta\, p(z_t|x_t)\, p(x_t|u_{0:t-1}, z_{0:t-1}) \\
&= \eta\, p(z_t|x_t) \int p(x_t, x_{t-1}|u_{0:t-1}, z_{0:t-1})\, dx_{t-1}
\end{aligned}
$$

Marginalization, or law of total probability

$$p(A) = \sum_{B_i} p(A, B_i)$$

Here we are actually using the law of total probability for conditional distributions, so

$$p(A|C) = \sum_{B_i} p(A, B_i|C)$$

Bayes' Filter: Derivation

$$
\begin{aligned}
bel(x_t) &= p(x_t|u_{0:t-1}, z_{0:t}) \\
&= \eta\, p(z_t|x_t, u_{0:t-1}, z_{0:t-1})\; p(x_t|u_{0:t-1}, z_{0:t-1}) \\
&= \eta\, p(z_t|x_t)\; p(x_t|u_{0:t-1}, z_{0:t-1}) \\
&= \eta\, p(z_t|x_t) \int p(x_t, x_{t-1}|u_{0:t-1}, z_{0:t-1})\; dx_{t-1} \\
&= \eta\, p(z_t|x_t) \int p(x_t|u_{0:t-1}, z_{0:t-1}, x_{t-1})\; p(x_{t-1}|z_{0:t-1}, u_{0:t-1})\; dx_{t-1}
\end{aligned}
$$

Previous state to
current state
&
previous state alone

Definition of conditional distribution

$p(A, B|C) = p(A|B, C)p(B|C)$

Bayes' Filter: Derivation

$$
\begin{aligned}
bel(x_t) &= p(x_t|u_{0:t-1}, z_{0:t}) \\
&= \eta\, p(z_t|x_t, u_{0:t-1}, z_{0:t-1})\, p(x_t|u_{0:t-1}, z_{0:t-1}) \\
&= \eta\, p(z_t|x_t)\, p(x_t|u_{0:t-1}, z_{0:t-1}) \\
&= \eta\, p(z_t|x_t) \int p(x_t, x_{t-1}|u_{0:t-1}, z_{0:t-1})\, dx_{t-1} \\
&= \eta\, p(z_t|x_t) \int p(x_t|u_{0:t-1}, z_{0:t-1}, x_{t-1})\, p(x_{t-1}|z_{0:t-1}, u_{0:t-1})\, dx_{t-1} \\
&= \eta\, p(z_t|x_t) \int p(x_t|u_{t-1}, x_{t-1})\, p(x_{t-1}|z_{0:t-1}, u_{0:t-1})\, dx_{t-1}
\end{aligned}
$$

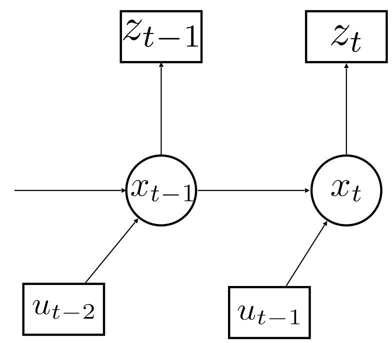
Markov assumption for dynamics

Do not need all past history

Bayes' Filter: Derivation

$$
\begin{aligned}
bel(x_t) &= p(x_t | u_{0:t-1}, z_{0:t}) \\
&= \eta\, p(z_t | x_t, u_{0:t-1}, z_{0:t-1})\; p(x_t | u_{0:t-1}, z_{0:t-1}) \\
&= \eta\, p(z_t | x_t)\; p(x_t | u_{0:t-1}, z_{0:t-1}) \\
&= \eta\, p(z_t | x_t) \int p(x_t, x_{t-1} | u_{0:t-1}, z_{0:t-1})\; dx_{t-1} \\
&= \eta\, p(z_t | x_t) \int p(x_t | u_{0:t-1}, z_{0:t-1}, x_{t-1})\; p(x_{t-1} | z_{0:t-1}, u_{0:t-1})\; dx_{t-1} \\
&= \eta\, p(z_t | x_t) \int p(x_t | u_{t-1}, x_{t-1})\; p(x_{t-1} | z_{0:t-1}, u_{0:t-1})\; dx_{t-1} \\
&= \eta\, p(z_t | x_t) \int p(x_t | u_{t-1}, x_{t-1})\; p(x_{t-1} | z_{0:t-1}, u_{0:t-2})\; dx_{t-1}
\end{aligned}
$$

This is the belief at the previous time step!
This means we can perform filtering recursively.



Control at time t-1 only affects state at time t

Bayes' Filter: Derivation

$$
\begin{aligned}
bel(x_t) &= p(x_t|u_{0:t-1}, z_{0:t}) \\
&= \eta \, p(z_t|x_t) \int p(x_t|u_{t-1}, x_{t-1}) \, bel(x_{t-1}) \, dx_{t-1}
\end{aligned}
$$

Bayes' Filter: Derivation

$$
\begin{aligned}
bel(x_t) &= p(x_t|u_{0:t-1}, z_{0:t}) \\
&= \eta\, p(z_t|x_t) \qquad \underbrace{\int p(x_t|u_{t-1}, x_{t-1})\, bel(x_{t-1})\, dx_{t-1}}_{\substack{\text{Computes the probability density of reaching state} \\ x_t \text{ from any possible previous state } x_{t-1} \\ \text{via the command } u_{t-1}}}
\end{aligned}
$$

Bayes' Filter: Derivation

$$
\begin{aligned}
bel(x_t) &= p(x_t|u_{0:t-1}, z_{0:t}) \\
&= \underbrace{\eta\, p(z_t|x_t) \int p(x_t|u_{t-1}, x_{t-1})\, bel(x_{t-1})\, dx_{t-1}}_{\substack{\text{Computes the probability density of reaching state} \\ x_t \text{ from any possible previous state } x_{t-1} \\ \text{via the command } u_{t-1} \text{ and observing } z_t}}
\end{aligned}
$$

Bayes' Filter: Derivation

$$
\begin{aligned}
bel(x_t) &= p(x_t | u_{0:t-1}, z_{0:t}) \\
&= \underbrace{\eta\, p(z_t | x_t) \underbrace{\int p(x_t | u_{t-1}, x_{t-1})\, bel(x_{t-1})\, dx_{t-1}}_{\text{Belief after prediction step}}}_{\text{Belief after update step}}
\end{aligned}
$$