Introduction to Bayesian Filters
for Location Estimation

National Tsing Hua University, Taiwan
NMSL
Shu-Ting Wang


Where the hell am I?

Localization is Hard!

- To figure out where our robots are , we make use of sensors on the robots.
- But remember that sensors have inherent problems
  - Sensor noise
  - Sensor aliasing

Sensor Noise & Aliasing

- Sensor noise is mainly influenced by environment
- The solution can be
  - To take multiple readings into account
  - Employ temporal or multi-sensor fusion
- Non-uniqueness of sensors' readings is normal
- Even with multiple sensors, there is a many-to-one mapping from environmental states to robots' perceptual inputs

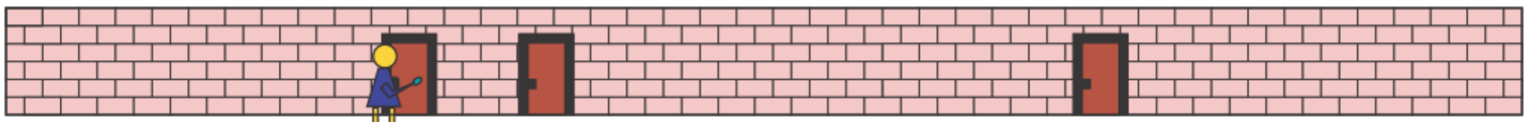
Try Bayesian Filters?

- Bayes filters represent the state at time ***t*** by random variables $\boldsymbol{x_t}$.
- At each time spot, a probability distribution over $\boldsymbol{x_t}$, called belief ***bel***($\boldsymbol{x_t}$) represents the uncertainty.
- Bayes filters aim to estimate beliefs over the state space conditioned on all information from the sensor data sequentially.

Belief?

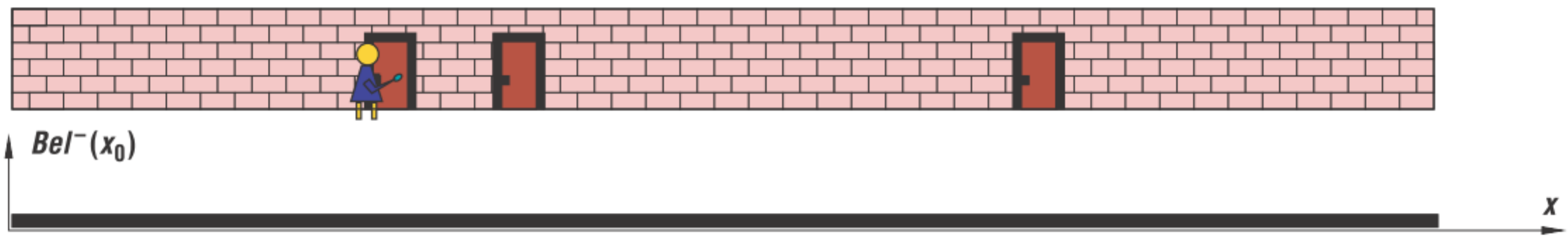
- ***bel***(x_t) is defined by the posterior density over the random variable x_t conditioned on all sensor data available at time t:
  - ***bel***$(x_t) = p(x_t \mid z_1, z_2, \ldots, z_t)$
- We can use belief to answer a question.
  - What is the probability that the person is at location x if the sensor observations are $z_1, z_2, \ldots, z_t$?

1D Example of Bayes Filters



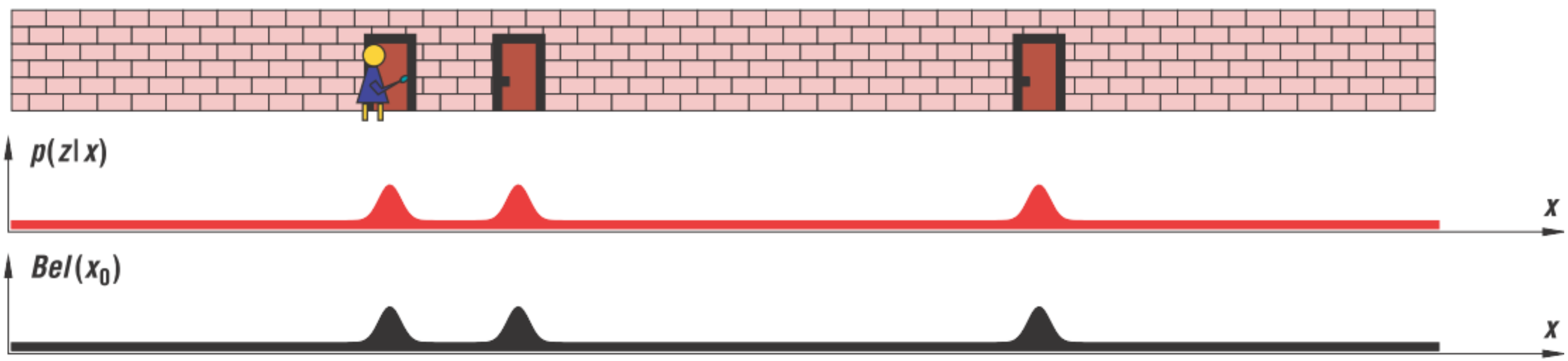
- Assume that you are in a dark world with three doors that all look the same and we can tell the difference between the doors and the wall using sensors.

1D Example of Bayes Filters



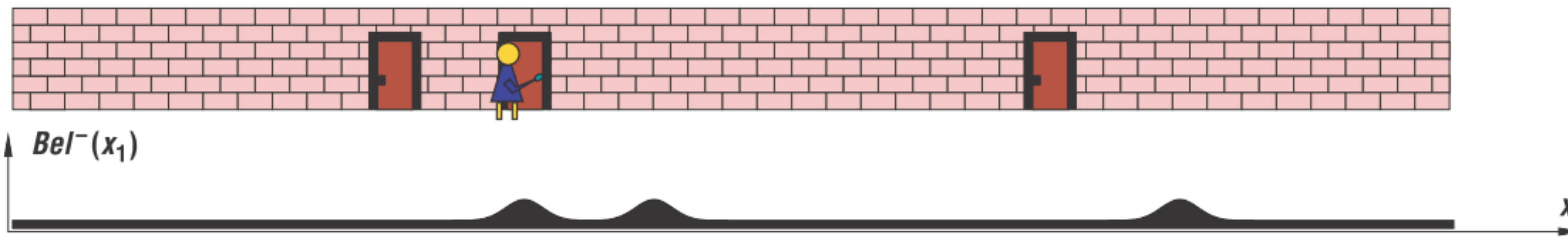
- We don't know where it is. Thus, a reasonable initial belief of its position is a uniform distribution.

1D Example of Bayes Filters



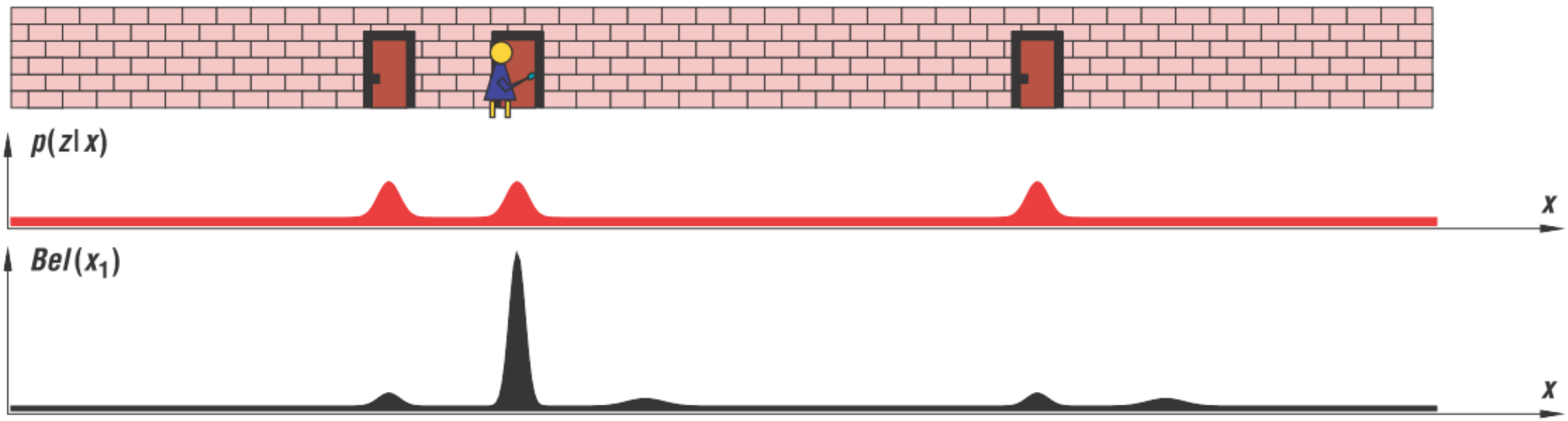
- A sensor reading is indicating a door at certain locations. This sensor reading should be integrated with prior believe to update our belief using a Bayes filter.

1D Example of Bayes Filters



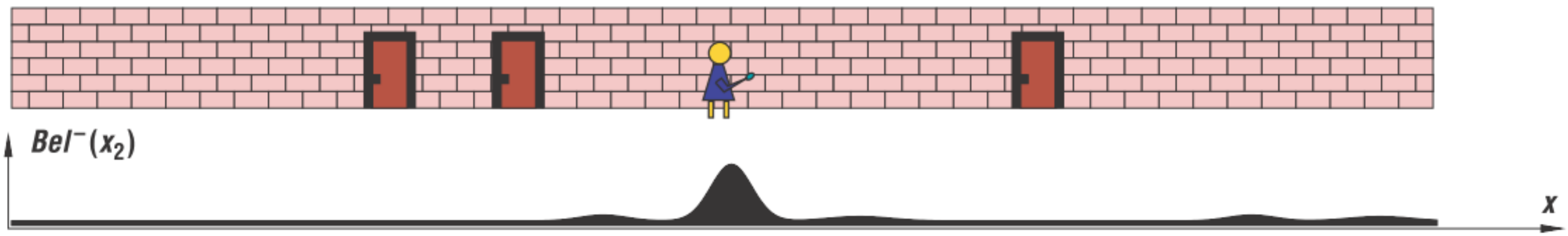
- We are moving and the Bayes filter shifts the belief in the direction of our movement

1D Example of Bayes Filters



- The sensor observes another door, and updates its belief model. The movement results in a very high belief at the second door.

1D Example of Bayes Filters



- Finally, we move on and the belief is updated after the next movement

Combine Observations & Movements(1/7)

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

Definition of the belief

Combine Observations & Movements(2/7)

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

Bayes' rule

Combine Observations & Movements(3/7)

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

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

$$= \eta\, \underline{p(z_t \mid x_t)}\, p(x_t \mid z_{1:t-1}, u_{1:t})$$

Markov assumption

Combine Observations & Movements(4/7)

$$
\begin{aligned}
bel(x_t) &= p(x_t \mid z_{1:t}, u_{1:t}) \\
&= \eta \, p(z_t \mid x_t, z_{1:t-1}, u_{1:t}) \, p(x_t \mid z_{1:t-1}, u_{1:t}) \\
&= \eta \, p(z_t \mid x_t) \, p(x_t \mid z_{1:t-1}, u_{1:t}) \\
&= \eta \, p(z_t \mid x_t) \, \underline{\int_{x_{t-1}} p(x_t \mid x_{t-1}, z_{1:t-1}, u_{1:t})} \\
&\qquad \underline{p(x_{t-1} \mid z_{1:t-1}, u_{1:t}) \, dx_{t-1}}
\end{aligned}
$$

Law of total probability

Combine Observations & Movements(5/7)

$$
\begin{aligned}
bel(x_t) &= p(x_t \mid z_{1:t}, u_{1:t}) \\
&= \eta\, p(z_t \mid x_t, z_{1:t-1}, u_{1:t})\, p(x_t \mid z_{1:t-1}, u_{1:t}) \\
&= \eta\, p(z_t \mid x_t)\, p(x_t \mid z_{1:t-1}, u_{1:t}) \\
&= \eta\, p(z_t \mid x_t) \int_{x_{t-1}} p(x_t \mid x_{t-1}, z_{1:t-1}, u_{1:t}) \\
&\qquad\qquad p(x_{t-1} \mid z_{1:t-1}, u_{1:t})\, dx_{t-1} \\
&= \eta\, p(z_t \mid x_t) \int_{x_{t-1}} \underline{p(x_t \mid x_{t-1}, u_t)}\, p(x_{t-1} \mid z_{1:t-1}, u_{1:t})\, dx_{t-1}
\end{aligned}
$$

Markov assumption

Combine Observations & Movements(6/7)

$$
\begin{aligned}
bel(x_t) &= p(x_t \mid z_{1:t}, u_{1:t}) \\
&= \eta\, p(z_t \mid x_t, z_{1:t-1}, u_{1:t})\, p(x_t \mid z_{1:t-1}, u_{1:t}) \\
&= \eta\, p(z_t \mid x_t)\, p(x_t \mid z_{1:t-1}, u_{1:t}) \\
&= \eta\, p(z_t \mid x_t) \int_{x_{t-1}} p(x_t \mid x_{t-1}, z_{1:t-1}, u_{1:t}) \\
&\qquad p(x_{t-1} \mid z_{1:t-1}, u_{1:t})\, dx_{t-1} \\
&= \eta\, p(z_t \mid x_t) \int_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\, p(x_{t-1} \mid z_{1:t-1}, u_{1:t})\, dx_{t-1} \\
&= \eta\, p(z_t \mid x_t) \int_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\, p(x_{t-1} \mid z_{1:t-1}, u_{\underline{1:t-1}})\, dx_{t-1}
\end{aligned}
$$

Markov assumption

Combine Observations & Movements(7/7)

$$
\begin{aligned}
bel(x_t) &= p(x_t \mid z_{1:t}, u_{1:t}) \\
&= \eta\, p(z_t \mid x_t, z_{1:t-1}, u_{1:t})\, p(x_t \mid z_{1:t-1}, u_{1:t}) \\
&= \eta\, p(z_t \mid x_t)\, p(x_t \mid z_{1:t-1}, u_{1:t}) \\
&= \eta\, p(z_t \mid x_t) \int_{x_{t-1}} p(x_t \mid x_{t-1}, z_{1:t-1}, u_{1:t}) \\
&\qquad\qquad p(x_{t-1} \mid z_{1:t-1}, u_{1:t})\, dx_{t-1} \\
&= \eta\, p(z_t \mid x_t) \int_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\, p(x_{t-1} \mid z_{1:t-1}, u_{1:t})\, dx_{t-1} \\
&= \eta\, p(z_t \mid x_t) \int_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\, p(x_{t-1} \mid z_{1:t-1}, u_{1:t-1})\, dx_{t-1} \\
&= \eta\, p(z_t \mid x_t) \int_{x_{t-1}} p(x_t \mid x_{t-1}, u_t)\, \underline{bel(x_{t-1})}\, dx_{t-1}
\end{aligned}
$$

Recursive term

Prediction and Correction Step

- Bayes filter can be written as a two step process
- Prediction step

$$\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1})\ bel(x_{t-1})\, dx_{t-1}$$

- Correction step

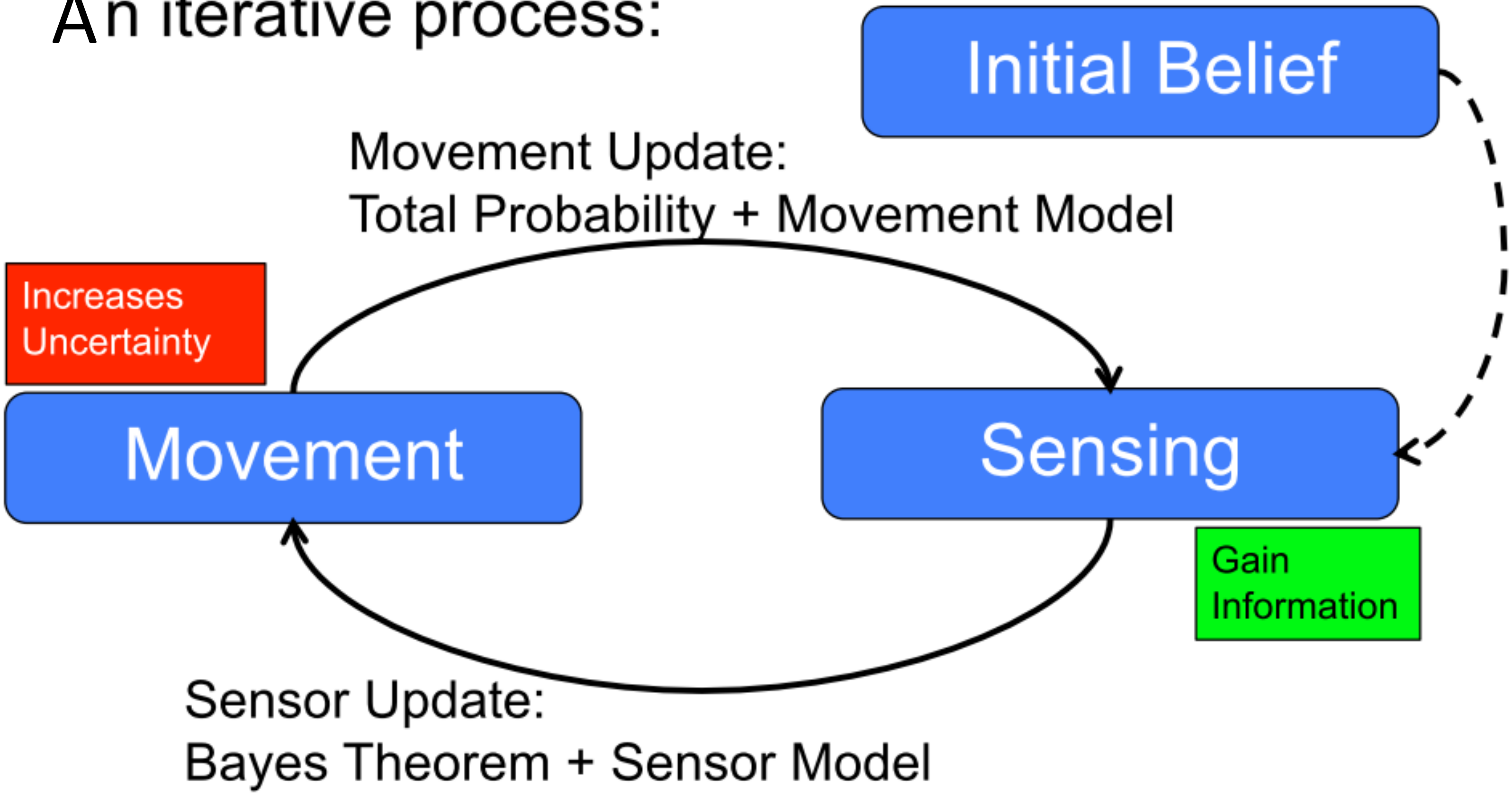
$$bel(x_t) = \eta\, p(z_t \mid x_t)\ \overline{bel}(x_{t-1})$$

Pseudo Code Discrete Bayes Filter

1. **Discrete_Bayes_Filter**(Bel(x),d):
2. η=0
3. If d is a perceptual data item z then
4. For all x do
5. $$Bel'(x) = P(z \mid x) Bel(x)$$
6. $$\eta = \eta + Bel'(x)$$
7. For all x do
8. $$Bel'(x) = \eta^{-1} Bel'(x)$$
9. Else if d is an action data item u then
10. For all x do
11. $$Bel'(x) = \sum_{x'} P(x \mid u, x')\ Bel(x')$$
12. Return Bel'(x)

Workflow of Discrete Bayes Filter

An iterative process:



Kalman Filter(1/3)

- For most cases, the state matrices drop out and we obtain the below equation, which is much easier to start with

measured value

current estimation

$$\hat{X}_k = K_k \, . \, Z_k + (1 - K_k) \, . \, \hat{X}_{k-1}$$

previous estimation

Kalman Gain

Kalman Filter(2/3)

- Time Update *(prediction)*

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k$$ $\hat{x}_k^-$ is prior estimate

$$P_k^- = AP_{k-1}A^T + Q$$ P_k^- is prior error covariance

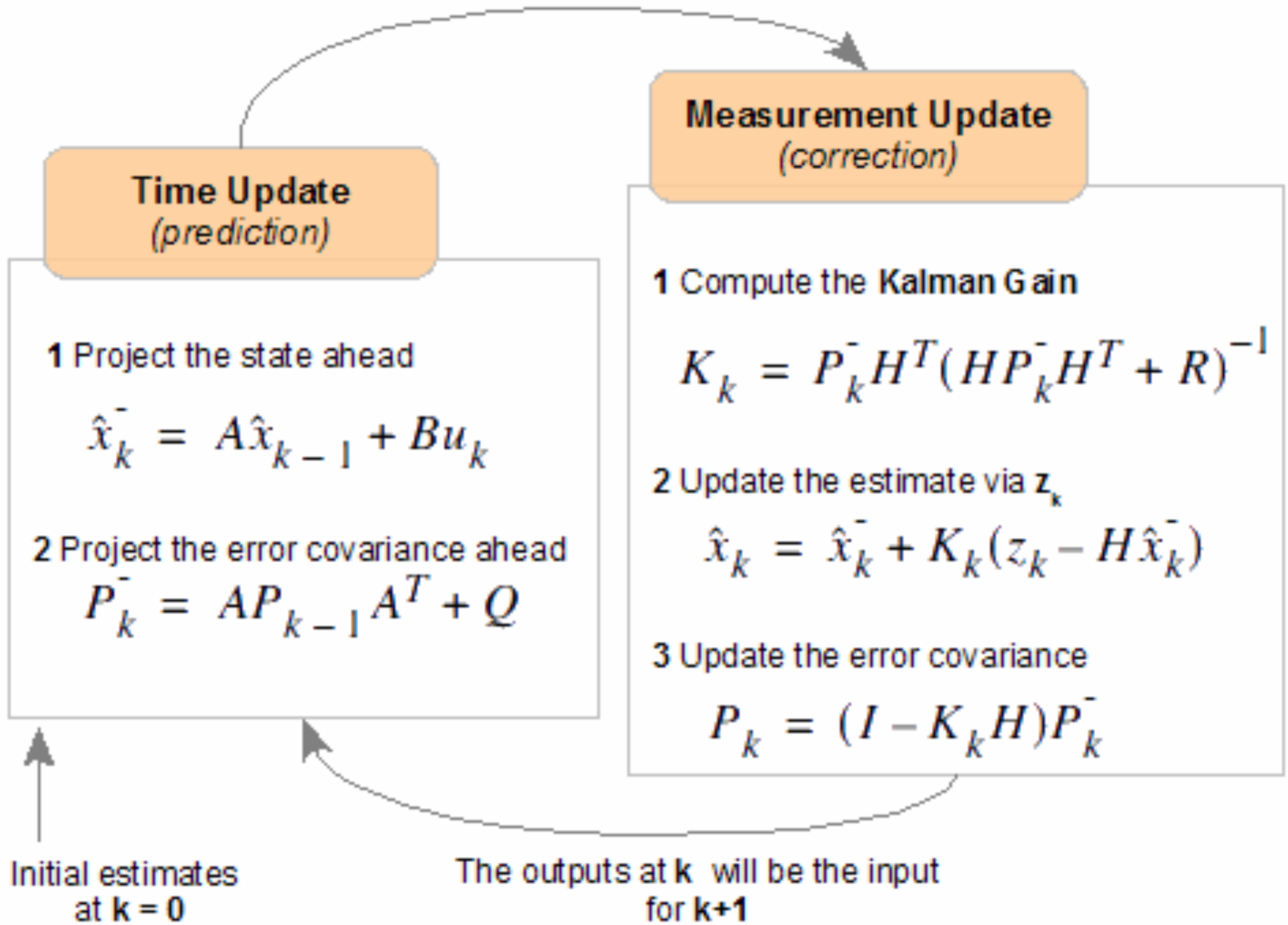
- Measurement Update *(correction)*

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-)$$ $\hat{x}_k$ is which is the estimate of **x** at time **k**

$$P_k = (I - K_k H)P_k^-$$ K_k is Kalman gain

Kalman Filter (3/3)



Pseudo Code Kalman Filter

1. **Kalman_Filter**(μ_{t-1}, Σ_{t-1}, u_t, z_t):
2. Prediction:
3. $\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$
4. $\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t$
5. Correction:
6. $K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + Q_t)^{-1}$
7. $\mu_t = \bar{\mu}_t + K_t (z_t - C_t \bar{\mu}_t)$
8. $\Sigma_t = (I - K_t C_t) \bar{\Sigma}_t$
9. Return μt , Σt

Q&A