Chapter Presentation

Outline

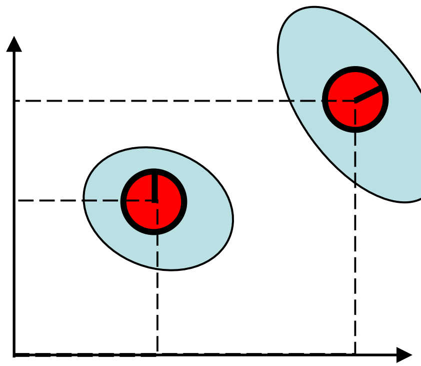
1. Introduction
2. The Bayes Filter
3. Gaussian filters
4. The Kalman filter

Labs:
SLAM Toolbox with Matlab

Assessment:
Labs + Exam (100%)

3. Gaussian Filters

- **The hypothesis:** Uncertainty in robot pose can be represented through a Gaussian random vector (grv)



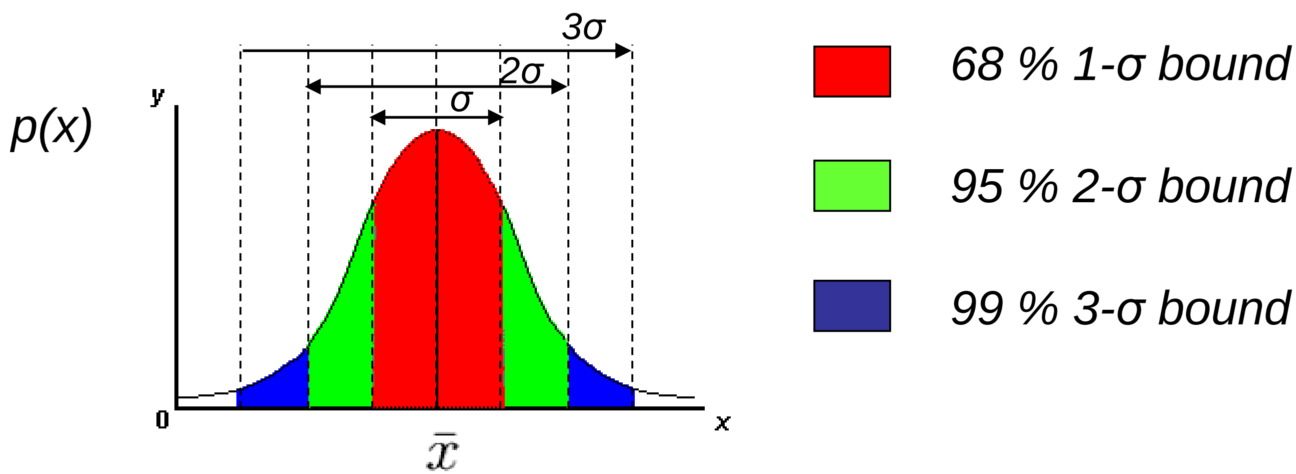
Position corrupted with Random Gaussian noise

- **Why the Gaussian hypothesis is so important?**
  - **Simplicity:** With only two parameters the pdf is completely defined.
  - **Computational efficiency:** Make some problems tractable.
  - **Central Limit Theorem (CLT):** The sum of a sequence of n independent random variables tends to a Gaussian pdf

Gaussian Random Variables

The pdf of a (scalar) ***Gaussian*** or ***normal random variable*** is

$$p(x) = \mathcal{N}(x; \bar{x}, \sigma^2) \triangleq \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\bar{x})^2}{2\sigma^2}}$$



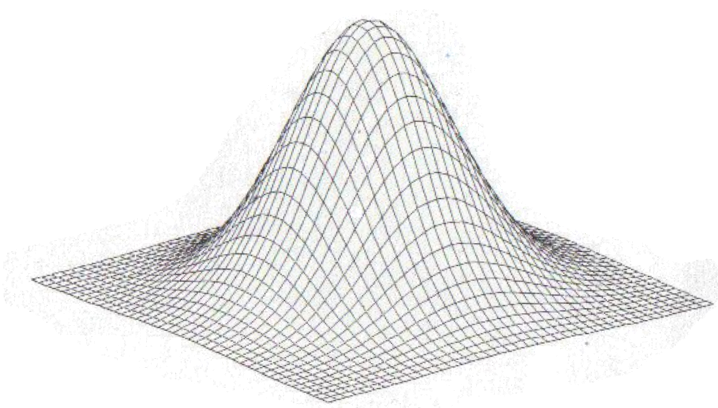
Gaussian Random Vectors (GRV)

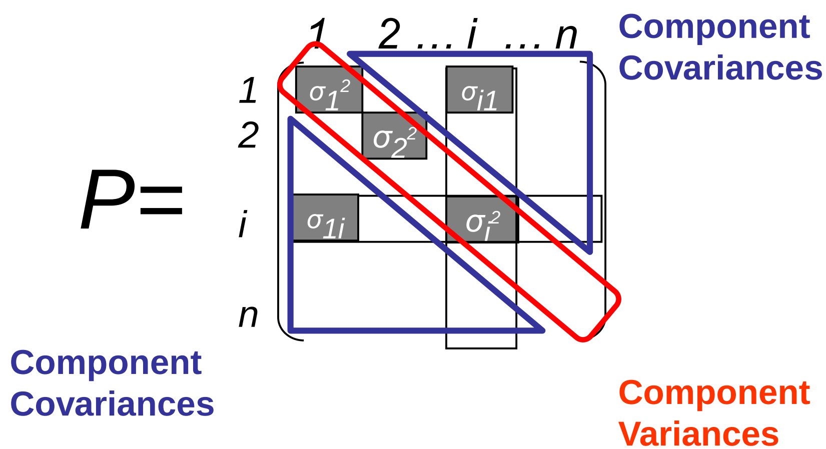
A ***vector-valued Gaussian*** random variable has the density

$$\boxed{\mathcal{N}(x;\bar{x},P) \triangleq |2\pi P|^{-1/2}\, e^{-\frac{1}{2}(x-\bar{x})'P^{-1}(x-\bar{x})}}$$

where

$$\bar{x} = E[x]$$
$$P = E[(x-\bar{x})(x-\bar{x})']$$





σ_i^2: Variance of component i

σ_{ij}: Covariance between component i & j

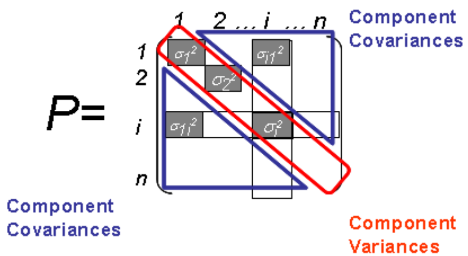
Gaussian Random Vectors (GRV)

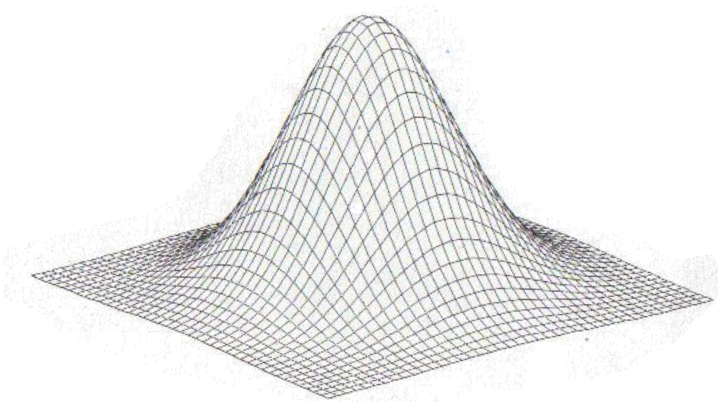
A ***vector-valued Gaussian*** random variable has the density

$$\boxed{\mathcal{N}(x;\bar{x},P) \triangleq |2\pi P|^{-1/2}\, e^{-\frac{1}{2}(x-\bar{x})' P^{-1}(x-\bar{x})}}$$

where

$$\bar{x} = E[x]$$
$$P = E[(x-\bar{x})(x-\bar{x})']$$





- P=P^T (symetric matrix) $\sigma_{ij} = \sigma_{ji}$
- P diagonal **=> uncorrelated => independent**
- P > 0 **=>** eigenvalues (P)>0
- P^{-1}>0

Normally:

independence ~~<=~~ uncorrelation
=>

But **For GRV:**

independence ⇔ uncorrelation

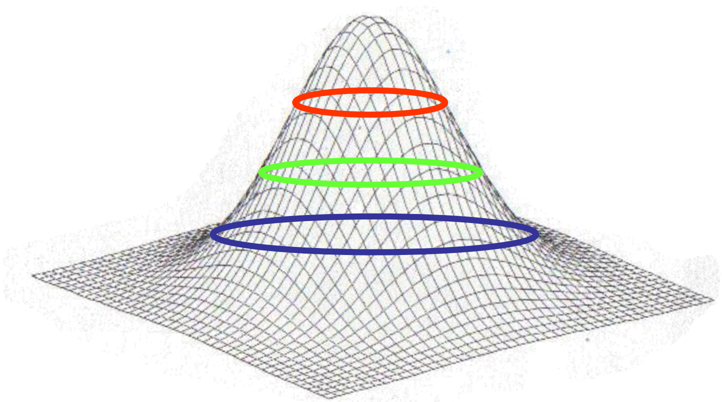
Gaussian Random Vectors (GRV)

A ***vector-valued Gaussian*** random variable has the density

$$\boxed{\mathcal{N}(x;\bar{x},P) \triangleq |2\pi P|^{-1/2}\, e^{-\frac{1}{2}(x-\bar{x})' P^{-1}(x-\bar{x})}}$$

where

$$\bar{x} = E[x]$$
$$P = E[(x-\bar{x})(x-\bar{x})']$$

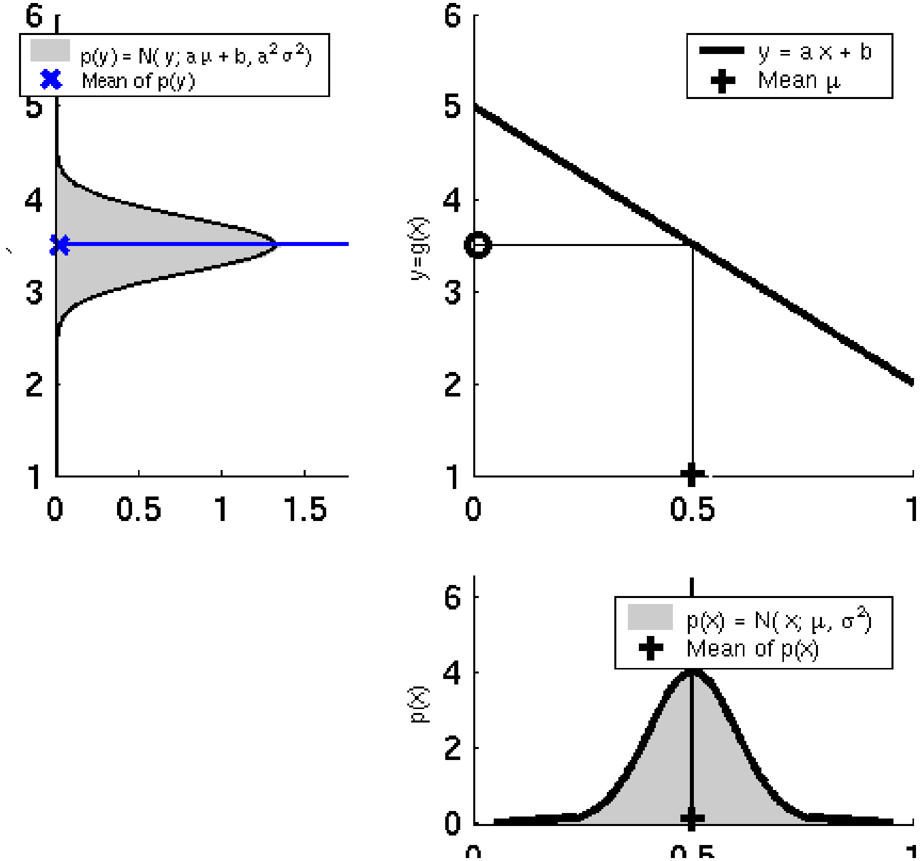


40 %   1- σ bound

95 % 2.45-σ bound

99 %   3.03-σ bound

Properties: Linear transformations of GRV



- We stay in the "Gaussian world" as long as we start with Gaussians and perform only linear transformations.

4. Kalman Filter

- **Recall Bayesian Filter.**

2.1 The Bayes Filter

Algorithm Bayes_filter($\boxed{bel(x_{t-1}), u_t, z_t}$**):**

for all x_t do

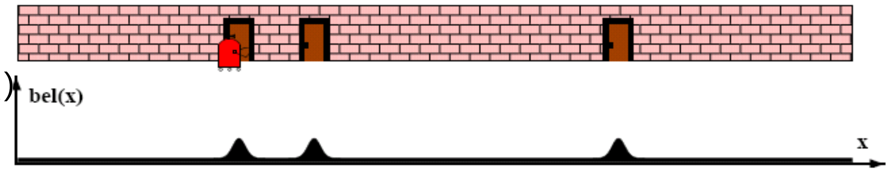
$\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1})\, bel(x_{t-1})\, dx$ — *Total Probability Theorem*

$bel(x_t) = \eta\, p(z_t \mid x_t)\, \overline{bel}(x_t)$ — *Bayes Rule*

endfor

return $bel(x_t)$

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

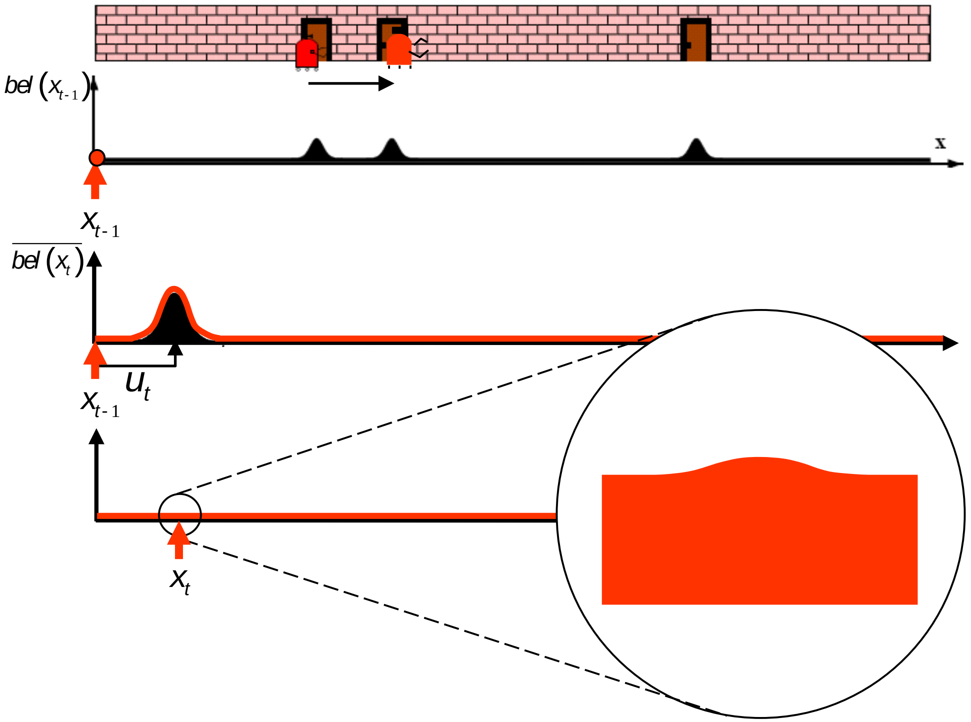


2.1 The Bayes Filter

Algorithm Bayes_filter($bel(x_{t-1}), u_t, z_t$):
  for all x_t do
    $\boxed{\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1})\, bel(x_{t-1})\, dx}$ *Total Probability Theorem*
    $bel(x_t) = \eta\, p(z_t \mid x_t)\, \overline{bel}(x_t)$ *Bayes Rule*
  endfor
  return $bel(x_t)$

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

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



2.1 The Bayes Filter

Algorithm Bayes_filter($bel(x_{t-1}), u_t, z_t$)**:**

for all x_t do

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

$bel(x_t) = \eta\, p(z_t \mid x_t)\, \overline{bel}(x_t)$

endfor

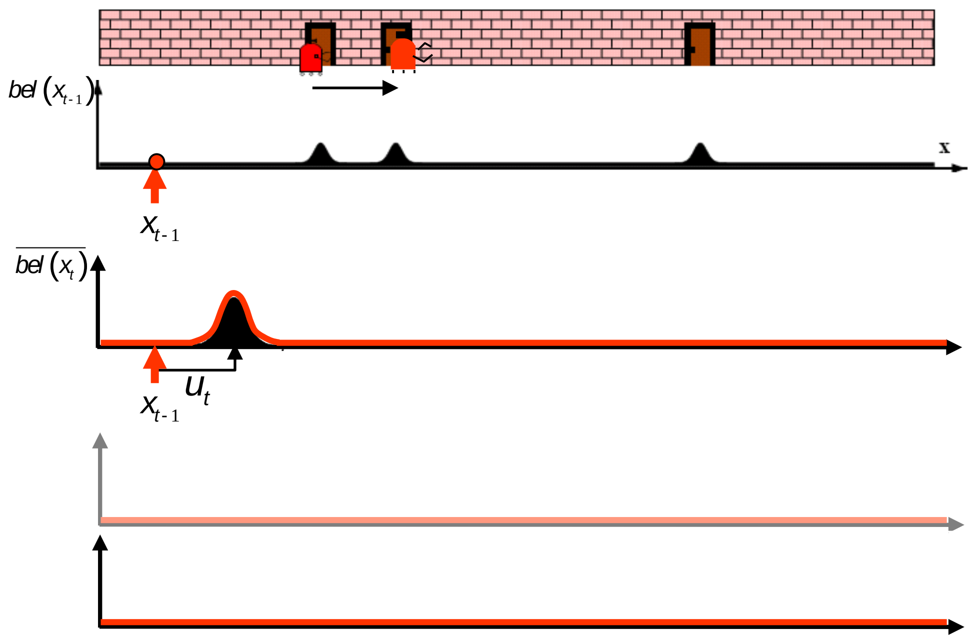
return $bel(x_t)$

Total Probability Theorem

Bayes Rule

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

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



2.1 The Bayes Filter

Algorithm Bayes_filter($bel(x_{t-1}), u_t, z_t$)**:**

  for all x_t do

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

    $bel(x_t) = \eta\; p(z_t \mid x_t)\; \overline{bel}(x_t)$

  endfor

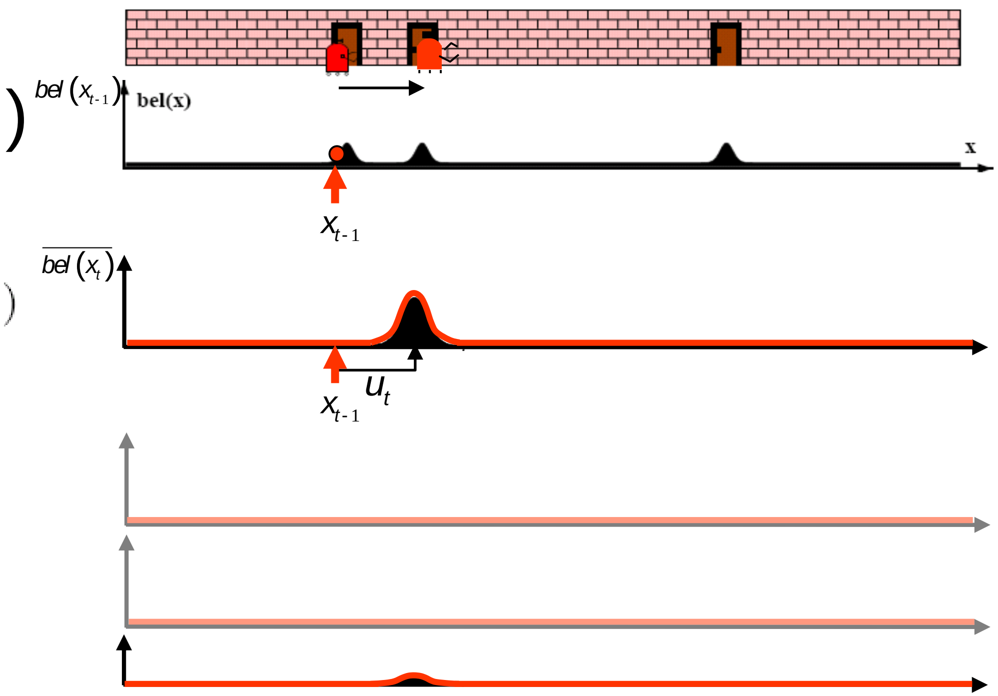
  return $bel(x_t)$

Total Probability Theorem

Bayes Rule

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

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

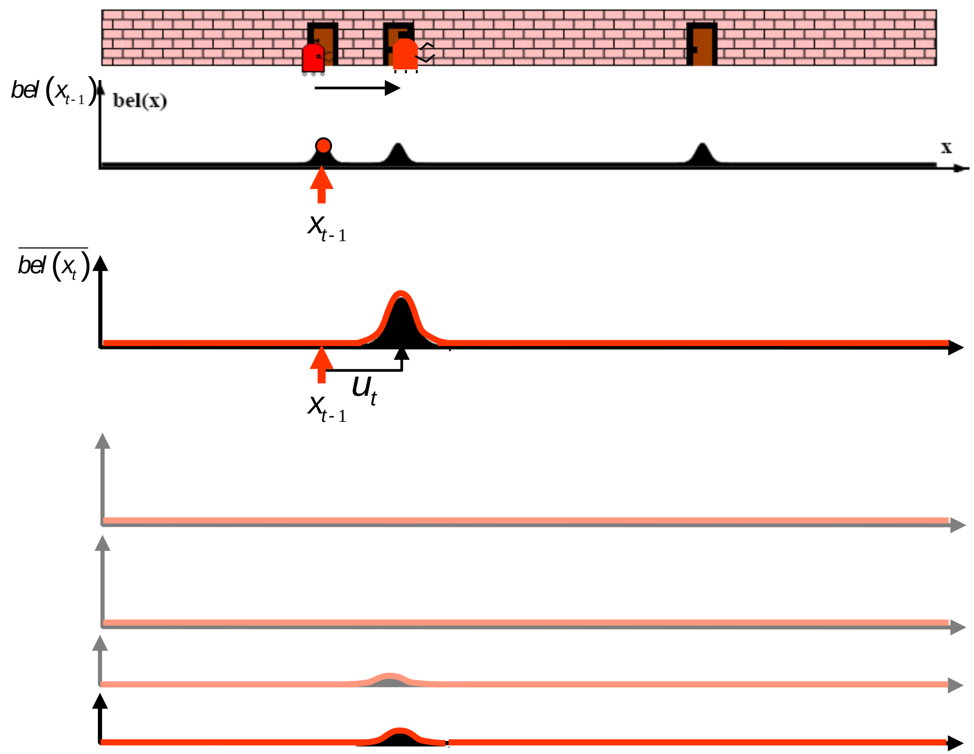


2.1 The Bayes Filter

Algorithm Bayes_filter($bel(x_{t-1}), u_t, z_t$):
  for all x_t do
    $\boxed{\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1})\, bel(x_{t-1})\, dx}$ *Total Probability Theorem*
    $bel(x_t) = \eta\, p(z_t \mid x_t)\, \overline{bel}(x_t)$ *Bayes Rule*
  endfor
  return $bel(x_t)$

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

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



2.1 The Bayes Filter

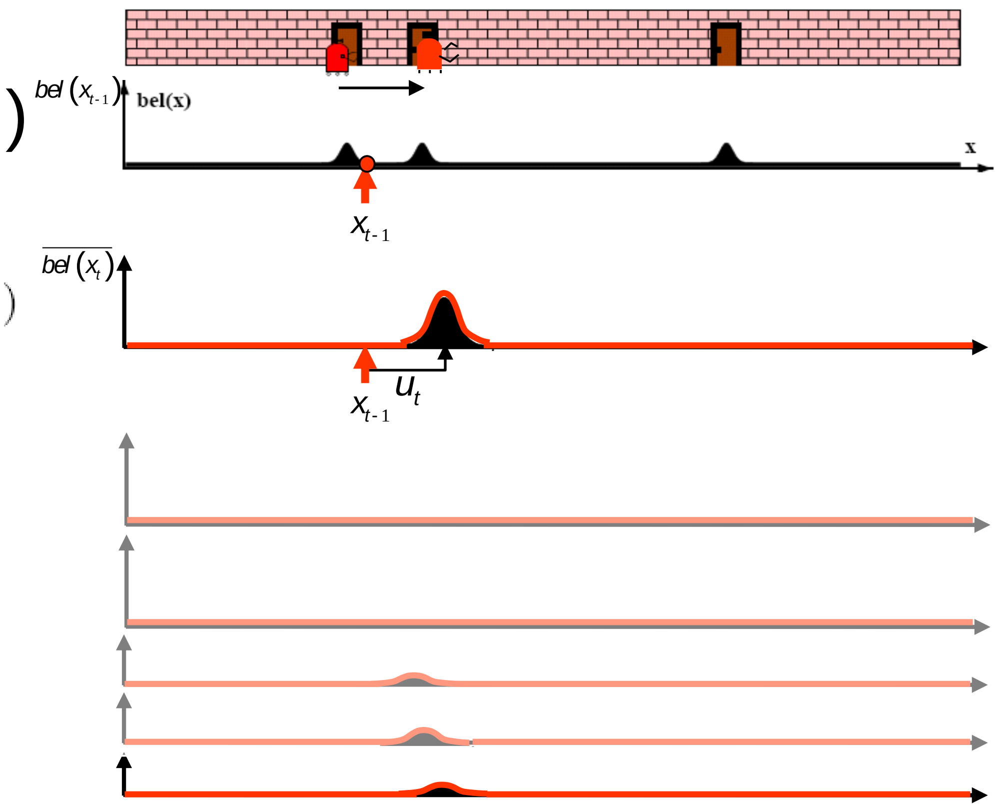
Algorithm Bayes_filter$(bel(x_{t-1}), u_t, z_t)$**:**
  for all x_t do
    $\boxed{\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1})\, bel(x_{t-1})\, dx}$
    $bel(x_t) = \eta\, p(z_t \mid x_t)\, \overline{bel}(x_t)$
  endfor
  return $bel(x_t)$

Total Probability Theorem

Bayes Rule

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

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



2.1 The Bayes Filter

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

Total Probability Theorem

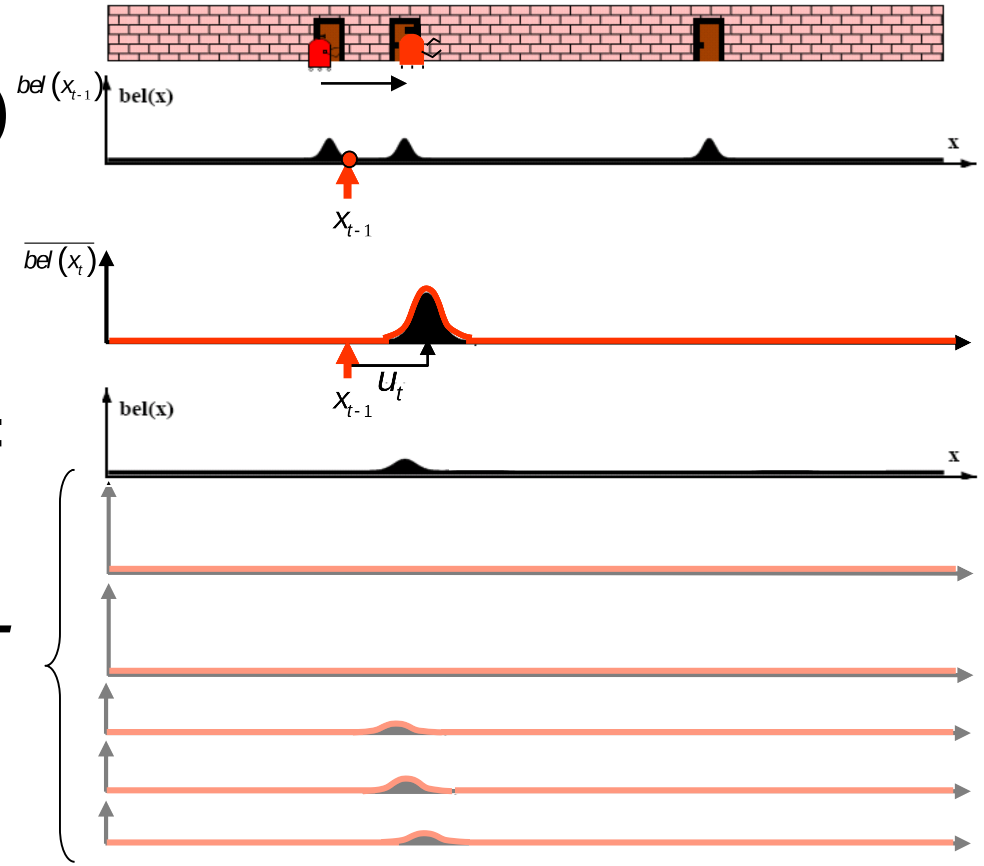
Bayes Rule

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

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

$$=$$

$$\Sigma$$



2.1 The Bayes Filter

Algorithm Bayes_filter($bel(x_{t-1}), u_t, z_t$)**:**

for all x_t do

$\boxed{\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1})\ bel(x_{t-1})\ dx}$ ***Total Probability Theorem***

$\boxed{bel(x_t) = \eta\ p(z_t \mid x_t)\ \overline{bel}(x_t)}$ ***Bayes Rule***

endfor

return $\boxed{bel(x_t)}$

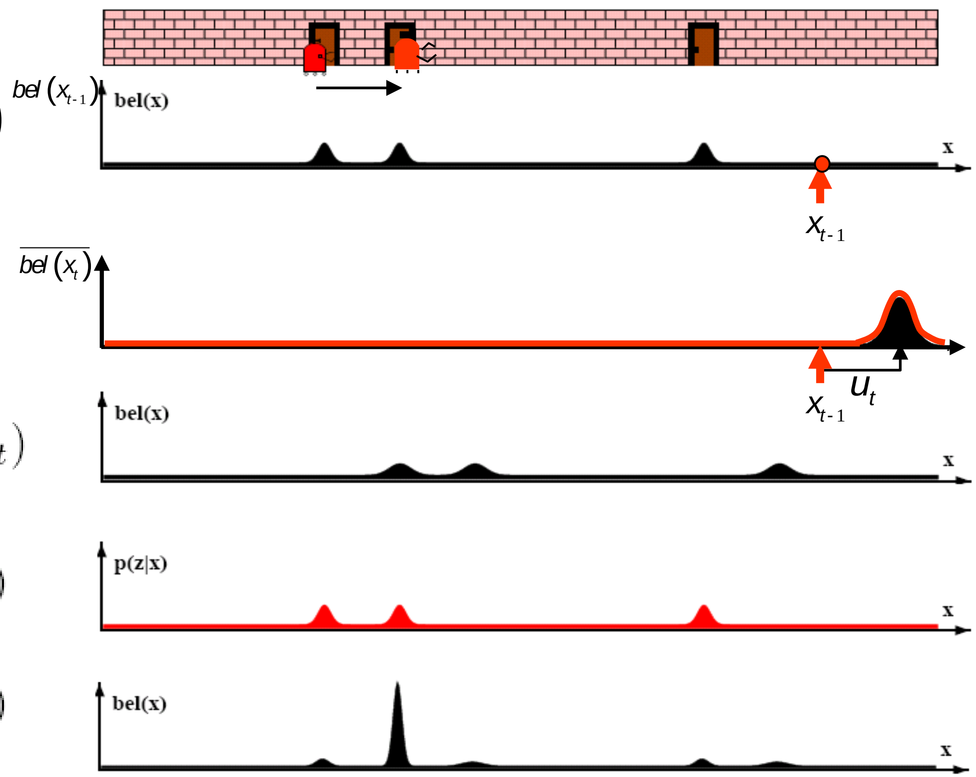
$$bel(x_{t-1}) = p(x_{t-1} \mid z_{1:t-1}, u_{1:t-1})$$

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

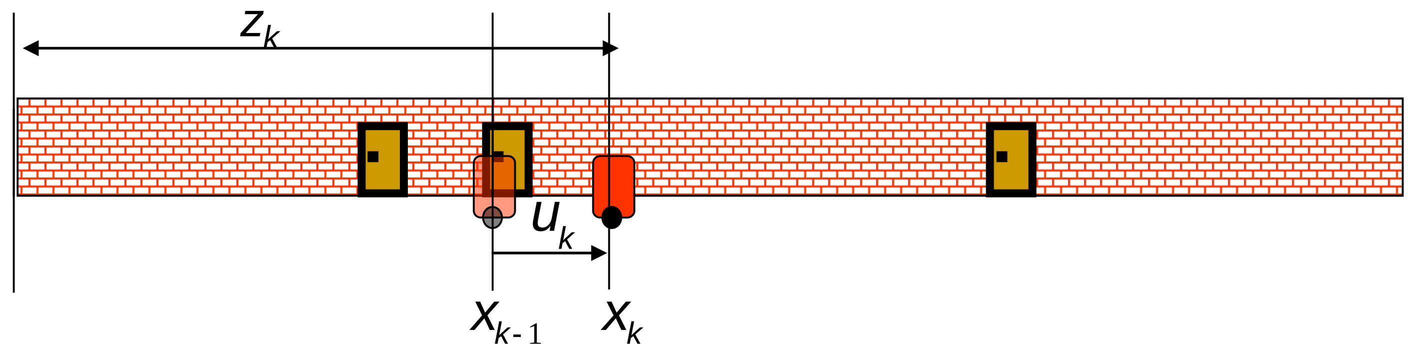
$$\overline{bel}(x_t) = p(x_t \mid z_{1:t-1}, u_{1:t})$$

$$p(z_t \mid x_t)$$

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



Example I: Monobot with odometry and position fixes



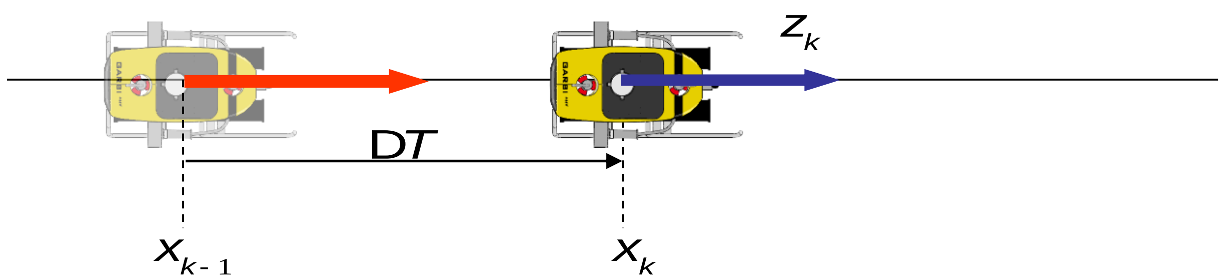
1. t_{k-1} : The Monobot is at position x_{k-1}
2. t_k: The Monobot Makes a displacement u_k
3. t_k: The Monobot Senses the distance wrt to beginning of the corridor

$$x_k = x_{k-1} + u_k + w_k \quad \Rightarrow \quad \boxed{x_k = A_k x_{k-1} + B_k u_k + w_k}$$

$$z_k = x_k + v_k \quad \Rightarrow \quad \boxed{z_k = H_k x_k + v_k}$$

$$A_k = 1$$
$$B_k = 1$$
$$H_k = 1$$
$$w_k = N(0, s_w)$$
$$v_k = N(0, s_v)$$

Example II: 1DOF AUV Ct Velocity Model & Velocity updates



1. t_{k-1} : The AUV is at position x_{k-1}
2. t_{k-1}: The AUV Moves at ct. velocity $\dot{x}_{k-1}$
3. t_k: The AUV reaches position x_k
4. t_k: The AUV measures its velocity z_k

$$x_k = x_{k-1} + \dot{x}_{k-1} DT + \frac{DT^2}{2} w_k$$

$$\dot{x}_k = \dot{x}_{k-1} + w_k DT$$

$$w_k = N(0, s_w)$$

$$z_k = \dot{x}_k + v_k$$

$$\mathbf{x}_k = \begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix}$$

$$z_k = \dot{x}_{k,measured}$$

$$\mathbf{x}_k = \mathbf{A}_k \mathbf{x}_{k-1} + \mathbf{B}_k \mathbf{u}_k + \mathbf{w}_k$$

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k$$

$$\overbrace{\begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix}}^{\mathbf{x}_k} = \overbrace{\begin{bmatrix} 1 & DT \\ 0 & 1 \end{bmatrix}}^{\mathbf{A}_k} \overbrace{\begin{bmatrix} x_{k-1} \\ \dot{x}_{k-1} \end{bmatrix}}^{\mathbf{x}_{k-1}} + \overbrace{\begin{bmatrix} \frac{DT^2}{2} \\ DT \end{bmatrix} w_k}^{\mathbf{w}_k}$$

$$z_k = \overbrace{\begin{bmatrix} 0 & 1 \end{bmatrix}}^{\mathbf{H}_k} \overbrace{\begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix}}^{\mathbf{x}_k} + \overbrace{v_k}^{\mathbf{v}_k}$$

Linear Stochastic Systems

- **General case:**

$$x_k = A_k x_{k-1} + B_k u_k + w_k$$

$$z_k = H_k x_k + v_k$$

A_k	Matrix (nxn) that describes how the state evolves from t-1 to t without controls or noise.
B_k	Matrix (nxl) that describes how the control u_t changes the state from t-1 to t.
H_k	Matrix (kxn) that describes how to map the state x_t to an observation z_t.
x_k	State Vector (nx1).
u_k	Input Vector (nx1).
w_k v_k	Random variables representing the process and measurement noise that are assumed to be independent and normally distributed with covariance Q_k and R_k respectively.

4.1 The Kalman Filter

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

Algorithm Kalman Filter $\left(\hat{x}_{k-1}, P_{k-1}, u_k, z_k\right)$

$$\hat{x}_k^- = A_k \hat{x}_{k-1} + B_k u_k$$
$$P_k^- = A_k P_{k-1} A_k^T + Q_k$$
$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k\right)^{-1}$$
$$\hat{x}_k = \hat{x}_k^- + K_k \left(z_k - H_k \hat{x}_k^-\right)$$
$$P_k = \left(I - K_k H_k\right) P_k^-$$
$$return\left(\hat{x}_k, P_k\right)$$

- **Optimal implementation of the Bayes filter for:**
  - **Linear Stochastic Systems:**
    - Possibly time varying
  - **Noise (process & measurement) should be:**
    - Gaussian zero mean
    - White
    - Possibly non stationary
    - Mutually independent

    Mathematically Tractable
  - **Computes de MMSE estimator for LG systems**

4.1 The Kalman Filter

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

Algorithm Kalman Filter $(\boxed{\hat{x}_{k-1}, P_{k-1}, u_k, z_k})$

$$\boxed{\begin{aligned} \hat{x}_k^- &= A_k \hat{x}_{k-1} + B_k u_k \\ P_k^- &= A_k P_{k-1} A_k^T + Q_k \end{aligned}}$$

$$K_k = P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k\right)^{-1}$$

$$\hat{x}_k = \hat{x}_k^- + K_k \left(z_k - H_k \hat{x}_k^-\right)$$

$$P_k = \left(I - K_k H_k\right) P_k^-$$

$$return\left(\hat{x}_k, P_k\right)$$

> Bel(x_{k-1}) is Gaussian

> A linear equation relates $x_k = f(x_{k-1}, u_k)$

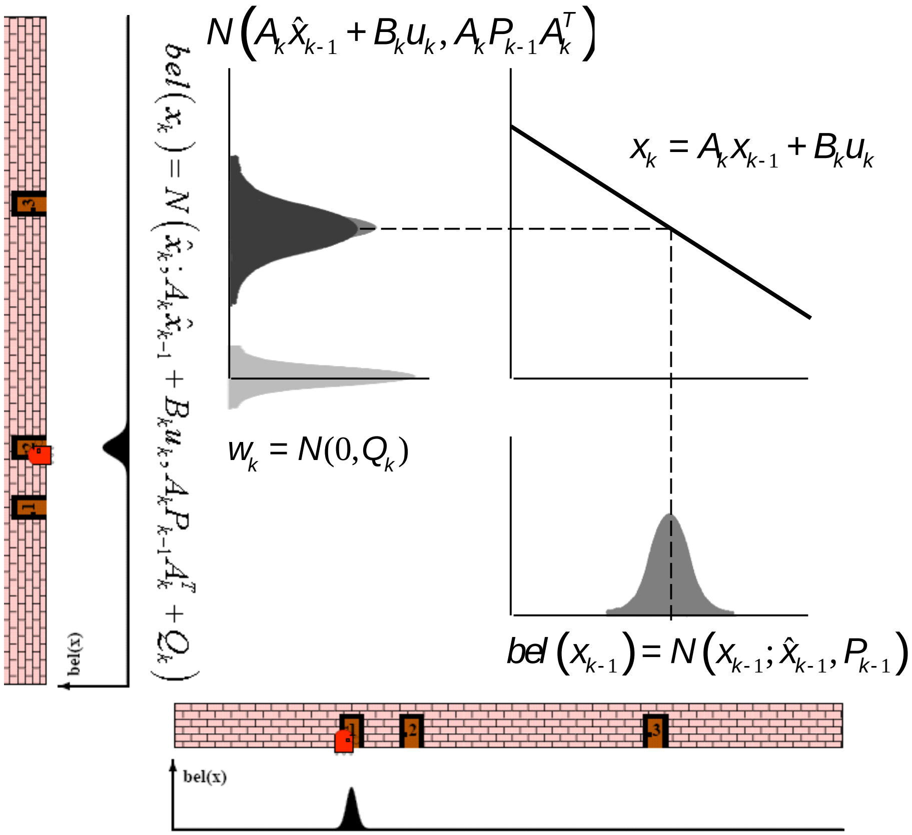
$$\mathbf{x_k = A_k x_{k-1} + B_k u_k}$$

> Gaussianity is kept through linear transformations

> The model is noisy

$$\mathbf{x_k = A_k x_{k-1} + B_k u_k + w_k}$$

> Bel(x_k) is Gaussian too



4.1 The Kalman Filter

Algorithm Bayes_filter($bel(x_{t-1}), u_t, z_t$):

for all x_t do

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

$$\boxed{bel(x_t) = \eta\, p(z_t \mid x_t)\, \overline{bel}(x_t)}$$

endfor

return $bel(x_t)$

Algorithm Kalman Filter $(\hat{x}_{k-1}, P_{k-1}, u_k, z_k)$

$$\hat{x}_k^- = A_k \hat{x}_{k-1} + B_k u_k$$

$$P_k^- = A_k P_{k-1} A_k^T + Q_k$$

$$\boxed{\begin{aligned} K_k &= P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k\right)^{-1} \\ \hat{x}_k &= \hat{x}_k^- + K_k \left(z_k - H_k \hat{x}_k^-\right) \\ P_k &= \left(I - K_k H_k\right) P_k^- \end{aligned}}$$

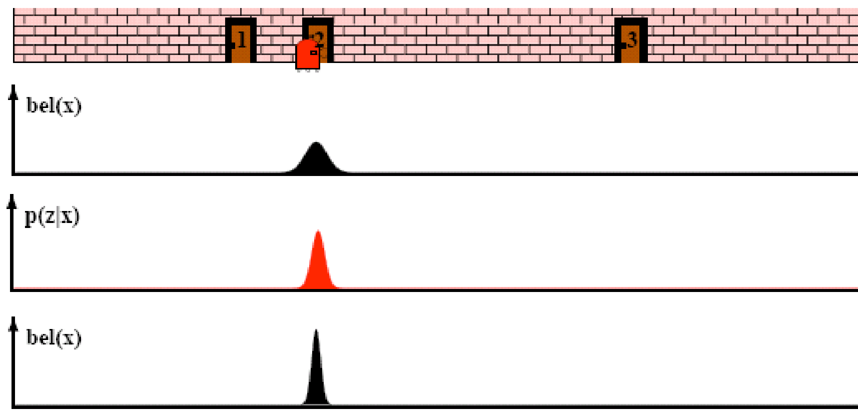
$$return\left(\hat{x}_k, P_k\right)$$

> Given a Gaussian prior & measurement the a posterior belief is Gaussian too

$$bel(x_k) = \eta\, p(z_k \mid x_k)\, \overline{bel}(x_k)$$

- $$\overline{bel}(x_k) = N\left(\underbrace{A_k \hat{x}_{k-1} + B_k u_k}_{\mu}, \underbrace{A_k P_{k-1} A_k^T + Q_k}_{P}\right)$$
- $$p(z_k \mid x_k) = N\left(\underbrace{z_k}_{\mu}, \underbrace{R_k}_{P}\right)$$

$$\boxed{\begin{aligned} bel(x_k) &= N\left(\underbrace{\hat{x}_k^- + K_k\left(z_k - H_k \hat{x}_k^-\right)}_{\mu}, \underbrace{\left(I - K_k H_k\right) P_k^-}_{P}\right) \\ K_k &= P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k\right)^{-1} \end{aligned}}$$



- See [Thrun05 ch. 3.3.4] for a formal probe.
- The KF eq, can also be derived from the fundamental eq. of linear estimation (see Bar-shallom)