Framework

Given

- Stream of observations z and action data u
- Sensor Model $P(z \mid x)$
- Action Model $P(x' \mid u, x)$
- Prior probability of the state $P(x)$

we want to estimate belief

Bayes Filter

Continuous environments

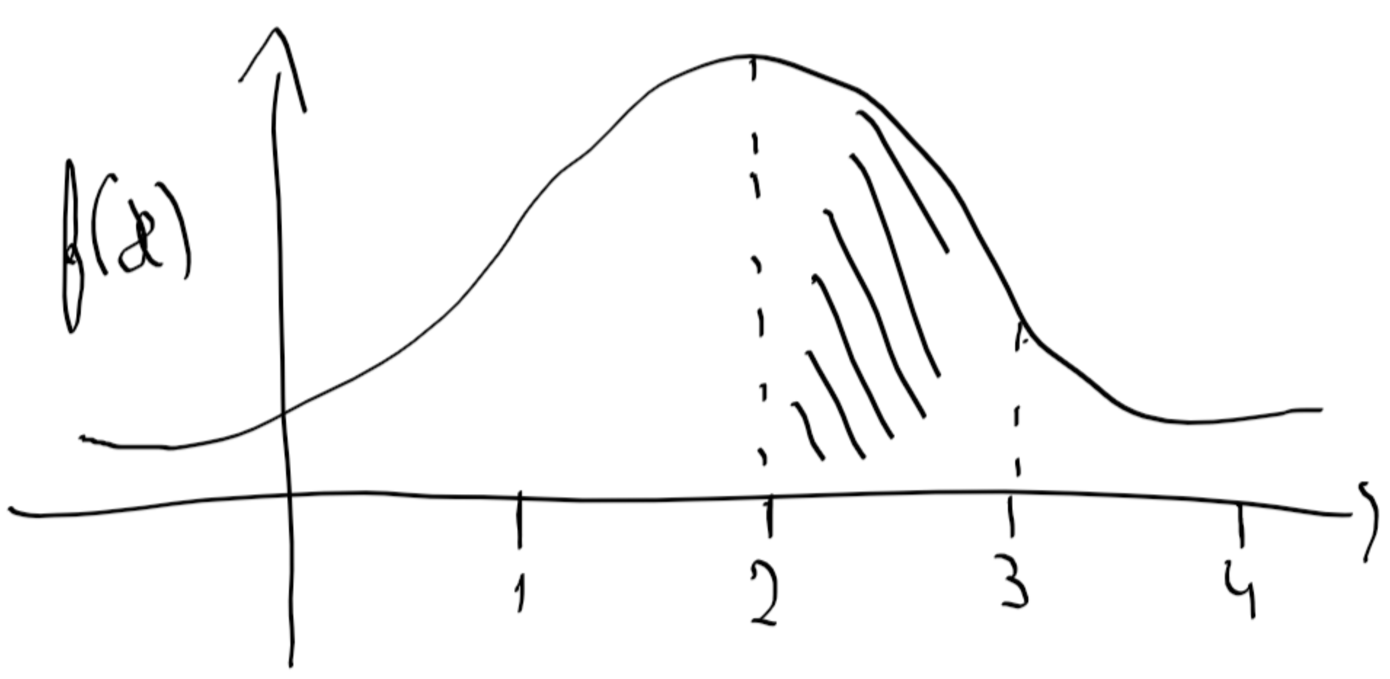
- Measurement Integration – $bel(x)_t = \eta \, p(z_t \mid x)$

  ↳ accounts for normalization

Belief integration

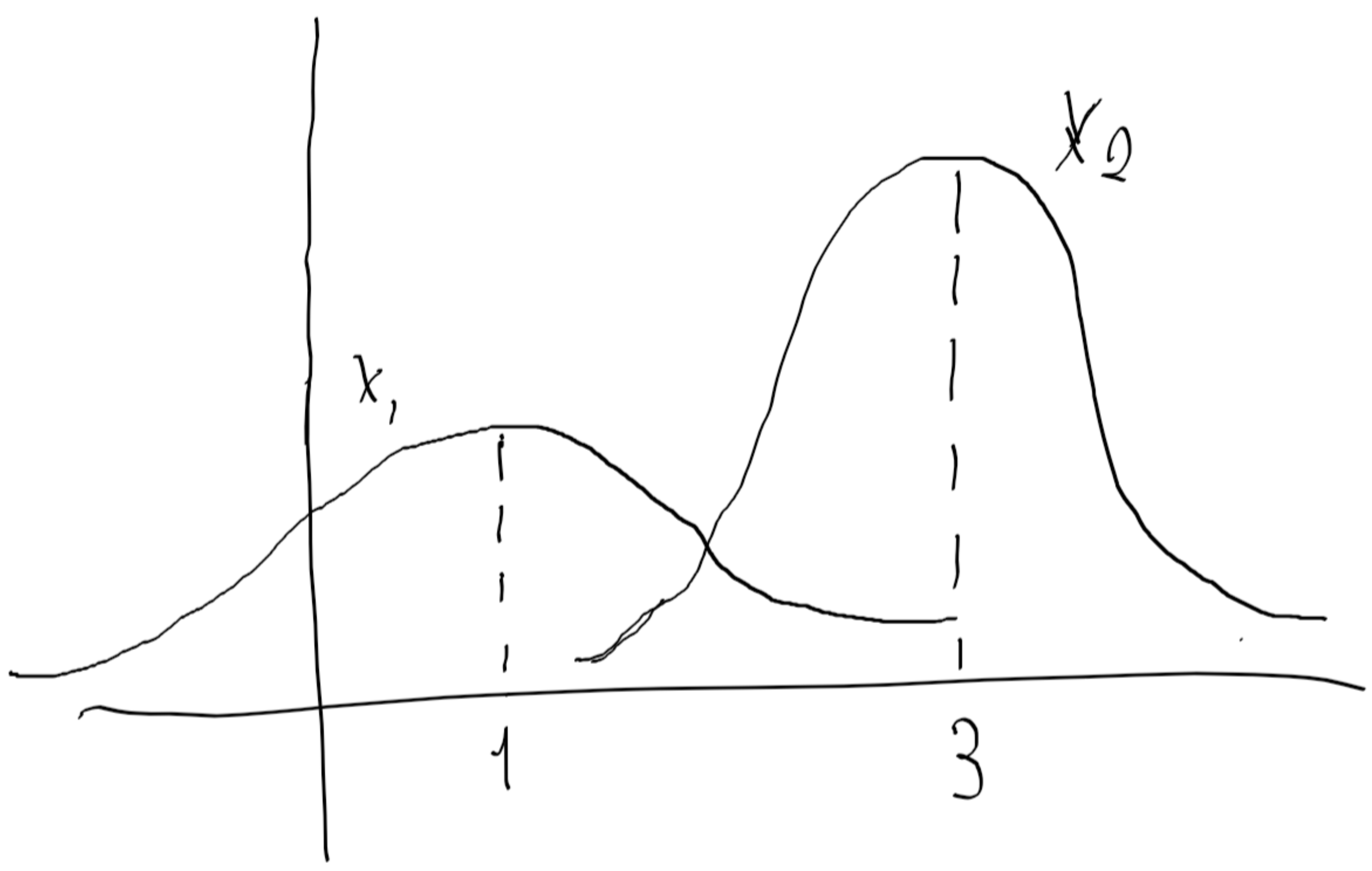
This method is not particularly easy to implement

Converting probability density function to probability



$$P([2,3]) = \int_2^3 f(x)\,dx$$

Gaussians



$\mu_1 = 1 \; ; \; \sigma_1^2 = 2$

$\mu_2 = 3 \; ; \; \sigma_2^2 = 1$

- μ product $= \dfrac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}\,\mu_1 + \dfrac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}\,\mu_2 = \dfrac{1}{5} + \dfrac{2}{5}\cdot 3 = \dfrac{6}{5}$

- $\sigma^2_{\text{product}} = \dfrac{1}{\sigma_1^{-2} + \sigma_2^{-2}} = \dfrac{1}{\frac{1}{2} + 1} = \dfrac{1}{\frac{3}{2}} = \dfrac{2}{3}$

Kalman Filter

| Assumes
- Linear model
- Every estimate is a gaussian
- Noise is gaussian (mean = 0)
- The next estimate depends only on the previous estimate as it inherently contains all the information from the past (Markov assumption)

| Product of gaussian distributions (measurement integration)

| Convolution of gaussians (motion forecast)

Discrete Kalman Filter

Estimates the state x of a discrete-time controlled process that is governed by the linear stochastic difference equation $\boxed{x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t}$ with a measurement $\boxed{z_t = C_t x_t + \delta_t}$

Nesta parte não apanhei nada do que o stor disse na aula então vou só meter os slides

Product of gaussians distributions (**measurement integration**)

$$\mu_P = \frac{\mu_1 . {\sigma_2}^2 + \mu_2 . {\sigma_1}^2}{{\sigma_1}^2 + {\sigma_2}^2} \qquad {\sigma_P}^2 = \frac{{\sigma_1}^2 . {\sigma_2}^2}{{\sigma_1}^2 + {\sigma_2}^2}$$

Convolution of gaussians distributions (**motion forecast**)

$$\mu_C = \mu_1 + \mu_2 \qquad {\sigma_C}^2 = {\sigma_1}^2 + {\sigma_2}^2$$

Kalman Filter – Motion Model

$$\hat{X}_t = F_t \hat{X}_{t-1} + B_t U_t + \omega_t$$

- $\hat{X}_t$ is the **estimated state**
- F_t is the **state transition model**
- B_t is the **control-input model**
- U_t is the **control vector**
- w_t is the **process noise** with covariance

$$Q_t : \omega_t \sim N(0, Q_t)$$

Kalman Filter – Observation Model

$$\hat{Z}_t = H_t X_t + \upsilon_t$$

- $\hat{z}_t$ is the **measurement** taken at time t
- H_t is the **observation model** of the state/event
- υ_t is the **observation noise** with covariance:

$$R_t : \upsilon_t \sim N(0, R_t)$$

Kalman Filter – Implementation

- The filter state is represented by two variables:
  - X_t is the **estimate of the state** at time t
  - P_t is the **measure of estimated accuracy** of the process
- The filter works in **two steps**:

Forecast integration

$$\overline{X}_t = F_t X_{t-1} + B_t U_t$$
$$\overline{P}_t = F_t P_{t-1} F_t^T + Q_t$$

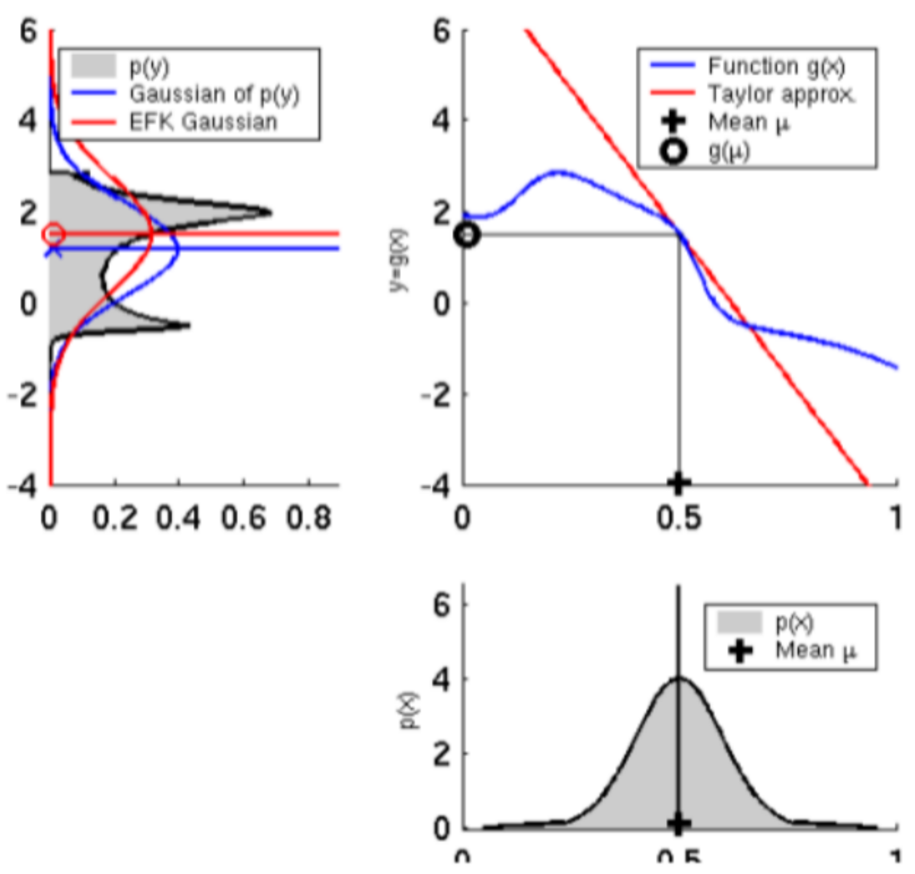
Measurement

$$K_t = \frac{\overline{P}_t H_t^T}{H_t \overline{P}_t H_t^T + R_t}$$
$$X_t = \overline{X}_t + K_t(Z_t - H_t \overline{X}_t)$$
$$P_t = (I - K_t H_t)\overline{P}_t$$

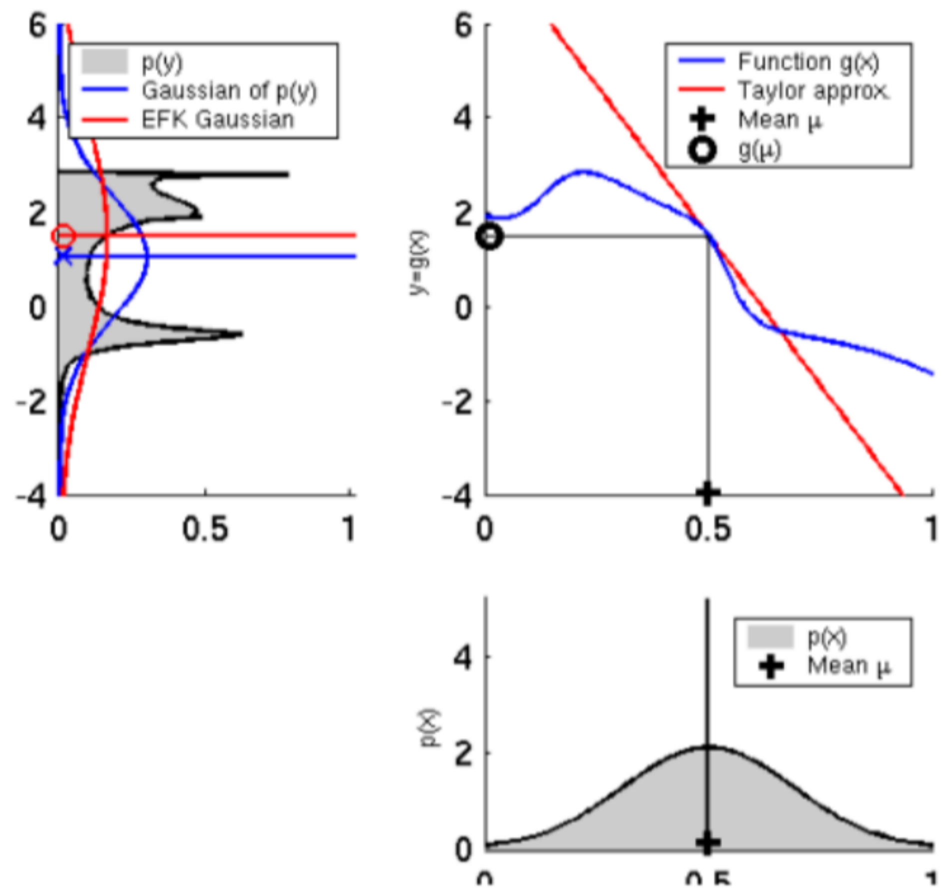
Non linear dynamic functions

- Most robotic problems involve nonlinear functions
- To these systems we apply Extended Kalman Filter (EKF)
- To apply EKF we consider that the function is linear at the point of interest
  - This results in a gaussian that is distinct from the gaussian that best describes the probability distribution of $p(y)$. This is combatted by restricting x's variance

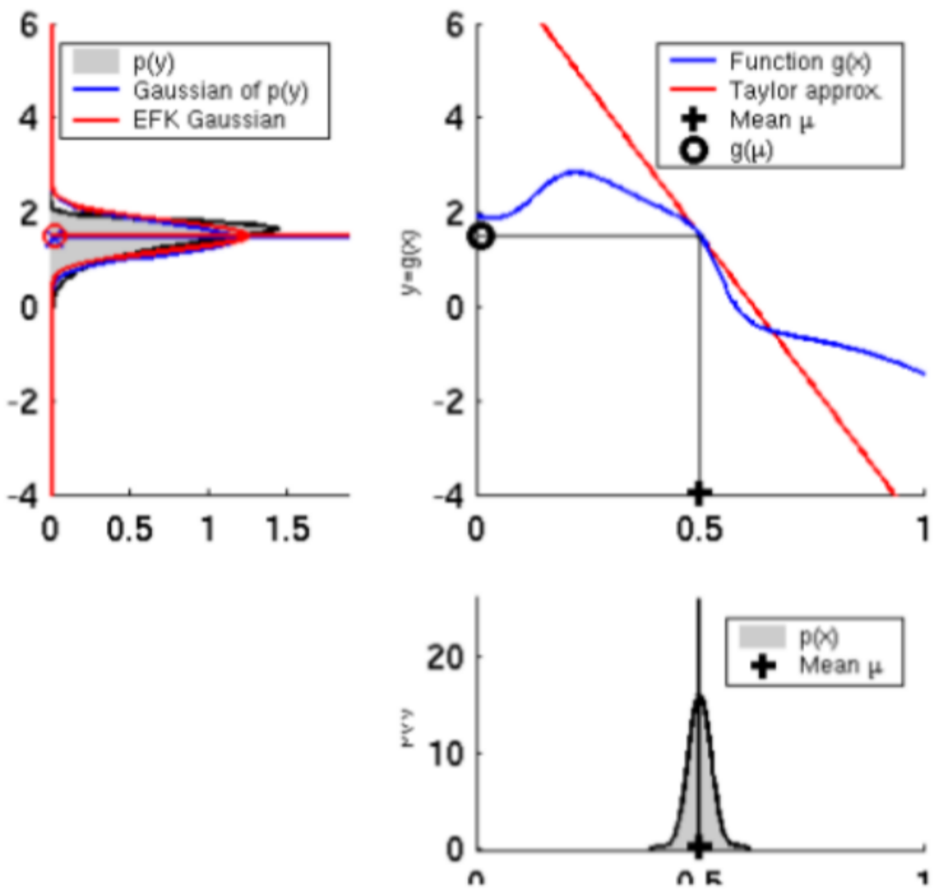
1)



2)



3)



First order Taylor Series

$$X = \begin{bmatrix} x \\ v \end{bmatrix} \; ; \; g(X) = \begin{bmatrix} x_t + \Delta t\, v_t \\ v_t \end{bmatrix} = \begin{bmatrix} g_x(X) \\ g_v(X) \end{bmatrix}$$

$$G = \begin{bmatrix} \frac{d g_x}{d x} & \frac{d g_x}{d v} \\ \frac{d g_v}{d x} & \frac{d g_v}{d v} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}$$

Prediction:

$$g(u_t, x_{t-1}) \approx g(u_t, \mu_{t-1}) + \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}} (x_{t-1} - \mu_{t-1})$$

$$g(u_t, x_{t-1}) \approx g(u_t, \mu_{t-1}) + G_t\, (x_{t-1} - \mu_{t-1})$$

Correction:

$$h(x_t) \approx h(\bar{\mu}_t) + \frac{\partial h(\bar{\mu}_t)}{\partial x_t} (x_t - \bar{\mu}_t)$$

$$h(x_t) \approx h(\bar{\mu}_t) + H_t\, (x_t - \bar{\mu}_t)$$

EKF summary

- Highly efficient
- Not optimal
- Can diverge if nonlinearities are large
- works surprisingly well even when all assumptions are violated

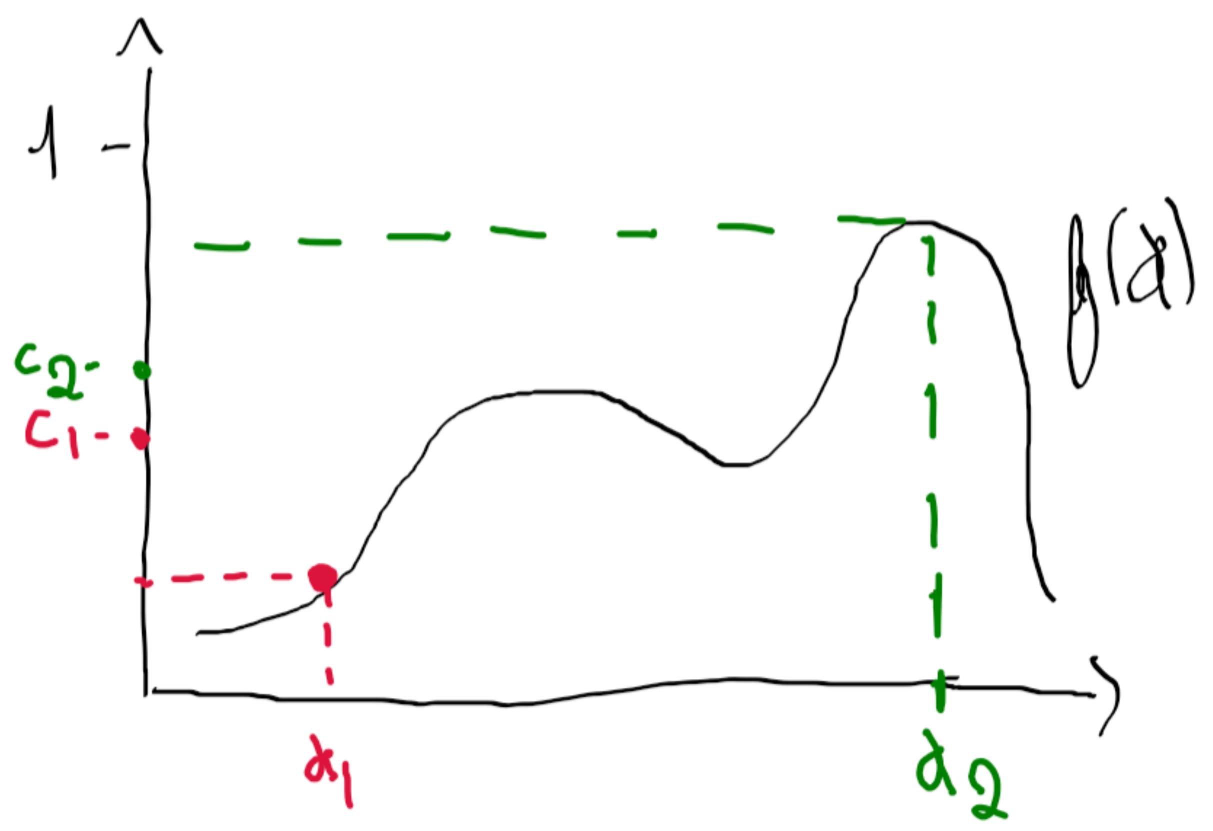
Particle Filter (Monte Carlo)

- One of the main issues with Kalman Filters is that the probabilities are gaussians
- Integrates measures over time
- The samples of the state are called particles. Each is a concrete instantiation of the state at t

$$S = \{ (s_i, \underbrace{w_i}_{\text{weight}}) \}, \ i \in \{1, 2, 3, \dots, N\}$$

$$p(x) = \sum_{i=1}^{N} w_i \cdot s_i(x)$$

Rejection sampling



- Let us assume that $f(x) < 1, \forall x$
- To apply rejection sampling
  - Sample x from uniform distribution
  - Sample c from $[0,1]$
  - Keep sample if $f(x) > c$, reject it otherwise

Sample 1 will be rejected because $f(x_1) < c_1$ and sample 2 will be accepted as $f(x_2) > c_2$

Resampling

Consider X_t the particle set at time t and $M = |X_t|$

$\{ \langle x_1, w_1 \rangle, \langle x_2, w_2 \rangle, \langle x_3, w_3 \rangle \}$, $w_1 = 0{,}1$; $w_2 = 0{,}3$; $w_3 = 0{,}6$

1^{st} ite

i was chosen to be 3 $\qquad X_t = \{ x_3 \}$

2^{nd} ite

i was chosen to be 3 $\qquad X_t = \{ x_3, x_3 \}$

3^{rd} ite

i was chosen to be 2 $\qquad X_t = \{ x_3, x_3, x_2 \}$