Fall 2011 Graduate Course on
Multisensory Predictive Learning

Particle Filters

November 1, 2011

Byoung-Tak Zhang

School of Computer Science and Engineering &
Cognitive Science and Brain Science Programs
Seoul National University

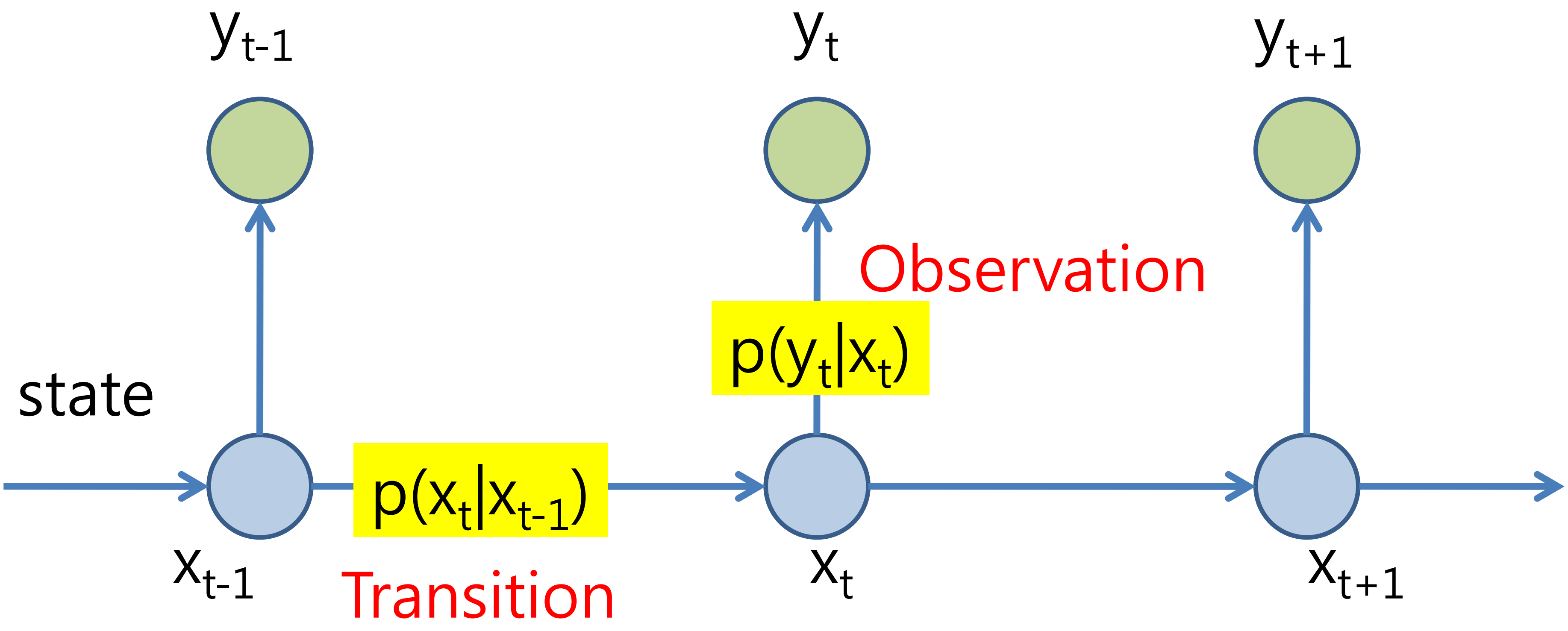
http://bi.snu.ac.kr/~btzhang/

Overview

- Filtering Problem
- Sequential Bayesian Filtering
- Particle Filter
- Monte Carlo (MC) Approximation
- MC with Importance Sampling (IS)
- Sequential Importance Sampling (SIS)
- Sampling Importance Resampling (SIR)

Filtering / Tracking

- We want to track the unknown state x of a system as it evolves over time based on the (noisy) observations y that arrive sequentially.



Dynamical System

x_t is state vector at time t, y_t is observations at time t

State equation $p(x_t \mid x_{t-1})$

Observation equation $p(y_t \mid x_t)$

Note: The forms of $p(x_t \mid x_{t-1})$ and $p(y_t \mid x_t)$ depend on the state transition function $f_X(\cdot)$ and observation function $f_Y(\cdot)$.

State equation: $x_t = f_X(x_{t-1}, u_t)$

f_X state transition function

u_t process noise with known distribution

Observation equation: $y_t = f_Y(x_t, v_t)$

f_Y observation function

u_t observation noise with known distribution

Filtering Problem

- The objective is to estimate unknown state x_t, based on a sequence of observations y_t, t=0,1, …

$$\text{Find posterior distribution } p(x_{0:t} \mid y_{1:t})$$

- By knowing posterior distribution (of the states) a number of estimates can be computed, e.g. the expected value of some function f(.) that depends on the state values:

$$E[f(x_{0:t})] = \int f(x_{0:t})\, p(x_{0:t} \mid y_{1:t})\, dx_{0:t}$$

Formally…

Let:

State vector $x_{0:t} = (x_0, \ldots, x_t)$

Observation vector $y_{1:t} = (y_1, \ldots, y_t)$

Find:

PDF $p(x_{0:t} \mid y_{1:t})$ posterior distribution

or $p(x_t \mid y_{1:t})$ filtering distribution

Given:

$p(x_0)$ prior distribution (on state)

$p(x_t \mid x_{t-1})$ transition probability (e.g., motor model)

$p(y_t \mid x_t)$ observation probability (e.g., sensor model)

$p(x_0)$ is given.

$t = 0$, observe y_0.

Update $$p(x_0 \mid y_0) = \frac{p(y_0 \mid x_0)}{p(y_0)} p(x_0)$$ (Bayes theorem)

Predict $$p(x_1 \mid y_0) = \int p(x_1 \mid x_0) p(x_0 \mid y_0) dx_0$$ (Markovian)

$t = 1$, observe y_1 from x_1

Update $$p(x_1 \mid y_1) = \frac{p(y_1 \mid x_1)}{p(y_1)} p(x_1)$$

Predict $$p(x_2 \mid y_1) = \int p(x_2 \mid x_1) p(x_1 \mid y_1) dx_1$$

$t = 2$, observe y_2 from x_2

Update $$p(x_2 \mid y_{1:2}) = \frac{p(y_2 \mid x_2)}{p(y_2 \mid y_1)} p(x_2 \mid y_1)$$

Predict $$p(x_3 \mid y_{1:2}) = \int p(x_3 \mid x_2) p(x_2 \mid y_{1:2}) dx_2$$

$t = 3$, observe y_3 from x_3

Update $$p(x_3 \mid y_{1:3}) = \frac{p(y_3 \mid x_2)}{p(y_3 \mid y_{1:3})} p(x_2 \mid y_{1:2})$$

Predict $$p(x_4 \mid y_{1:3}) = \int p(x_4 \mid x_3) p(x_3 \mid y_{1:3}) dx_3$$

Sequential Bayesian Filtering

Given $p(x_{t-1} \mid y_{1:t-1})$ ... prior (filtering) distribution (i.e., before observing y_t)

1. Prediction

$$p(x_t \mid y_{1:t-1}) = \int_{x_{t-1}} p(x_t \mid x_{t-1}) p(x_{t-1} \mid y_{1:t-1}) dx_{t-1} \qquad \text{(Eqn. 1)}$$

since $p(x_t \mid y_{1:t-1}) = \int_{x_{t-1}} p(x_t, x_{t-1} \mid y_{1:t-1}) dx_{t-1}$

$$= \int_{x_{t-1}} p(x_t \mid x_{t-1}, y_{1:t-1}) p(x_{t-1} \mid y_{1:t-1}) dx_{t-1}$$

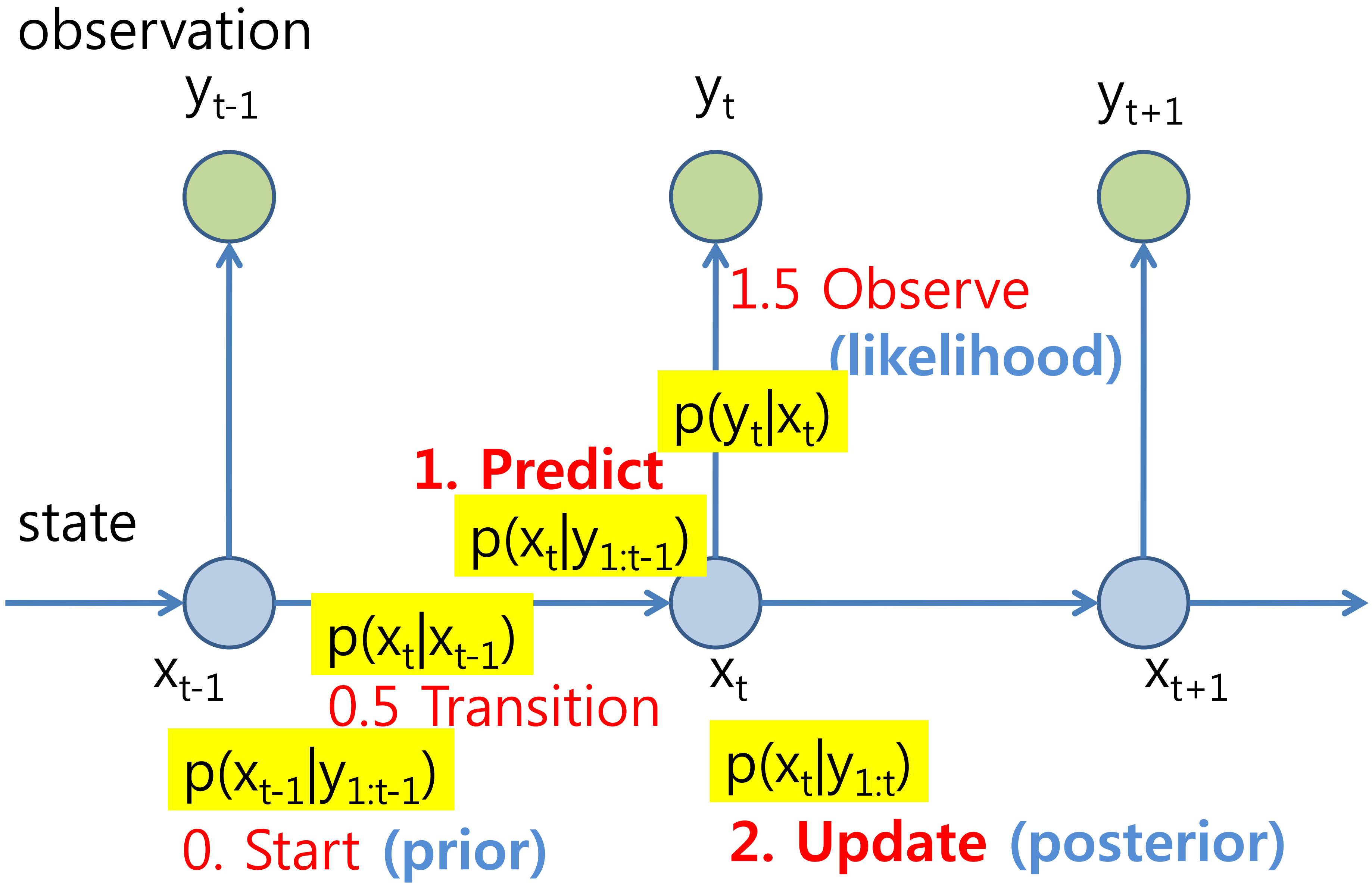
note: $p(a) = \int_b p(a,b) db$ and $p(a,b \mid c) = p(a \mid b,c) p(b \mid c)$

2. Update ... posterior distribution (after observing y_t)

$$p(x_t \mid y_{1:t}) = \frac{p(y_t \mid x_t) p(x_t \mid y_{1:t-1})}{p(y_t \mid y_{1:t-1})} \qquad \text{(Eqn. 2)}$$

where $p(y_t \mid y_{1:t-1}) = \int_{x_t} p(y_t \mid x_t) p(x_t \mid y_{1:t-1}) dx_t$

Graphically…



A Special Case: Kalman Filter

$$p(x_{t-1} \mid y_{1:t-1}) = N(x_{t-1} \mid m_{t-1|t-1}, P_{t-1|t-1})$$

$$p(x_t \mid y_{1:t-1}) = N(x_t \mid m_{t|t-1}, P_{t|t-1})$$

$$p(x_t \mid y_{1:t}) = N(x_t \mid m_{t|t}, P_{t|t})$$

$$m_{t|t-1} = F_t m_{t-1|t-1}$$

$$P_{t|t-1} = Q_{t-1} + F_t P_{t-1|t-1} F_t^T$$

...

$$x_t = F_t x_{t-1} + v_{t-1}, \qquad v_{t-1} \sim N(0, Q_{t-1})$$ ... linear and Gaussian

$$y_t = H_t x_t + n_t, \qquad n_t \sim N(0, R_t)$$

F_t : transition matrix (known)

H_t : observation matrix (known)

Particle Filters

- Particle filter is a technique for implementing recursive Bayesian filter by Monte Carlo sampling
- The idea is to represent the posterior density by a set of random samples (particles) with associated weighs.
  - Compute estimates based on these samples and weights.
- Many different names….
  - Sequential Monte Carlo (SMC)
  - Condensation method
  - Survival of the fittest (evolutionary comptutation?)

Advantages of Particle Filters

- Ability to represent arbitrary densities
  - Can deal with non-linearities
  - Non-Gaussian noise
- Particle filters focus adaptively on probable regions of state space
  - In contrast, HMM filters discretize the state space to N fixed states.
- Can be implemented in O(Ns)
  - Ns: sample size
  - Easy to implement
  - Easy to parallelize

Sample-Based PDF Representation

- Monte Carlo characterization of pdf
- Represent posterior density by a set of random i.i.d. samples (particles) from the pdf $p(x_{0:t}|y_{1:t})$
- For large number N of particles equivalent to functional description of pdf
- For $N \to \infty$, Monte Carlo method approaches optimal Bayesian estimate.

Monte Carlo (MC) Approximation

$$E_p[f(x)] = \int p(x) f(x) dx$$

$$\approx \frac{1}{N} \sum_{i=1}^{N} f(x^{(i)}), \qquad x^{(i)} \sim p(x) = N(0, \sigma^2)$$

- Monte Carlo approach

1. Simulate N random variables from p(x), e.g. Normal distribution

$$x^{(i)} \sim p(x) = N(0, \sigma^2)$$

2. Compute the average

$$E_p[f(x)] = \frac{1}{N} \sum_{i=1}^{N} f(x^{(i)}),$$

MC with Importance Sampling

$$E_p[f(x)] = \int_x p(x) f(x) dx$$

$$= \int_x \frac{p(x)}{q(x)} q(x) f(x) dx$$

$$\approx \sum_{i=1}^{N} w_i f(x^{(i)})$$

$x^{(i)} \sim q(x)$ — $q(x)$: proposal distribution

$w_i = \dfrac{p(x^{(i)})}{q(x^{(i)})}$ — w_i: importance weight

Note: $q(x)$ is easier to sample from than $p(x)$.

Importance Sampling (IS)

$$E[f(x_{0:t})] = \int f(x_{0:t}) p(x_{0:t} \mid y_{1:t}) dx_{0:t}$$

$$\approx \sum_{i=1}^{N} w_i f(x_{0:t}^{(i)})$$

$x_{0:t}^{(i)} \sim q(x_{0:t} \mid y_{1:t})$ $q(x)$: proposal distribution

$w_i = \dfrac{p(x_{0:t}^{(i)} \mid y_{1:t})}{q(x_{0:t}^{(i)} \mid y_{1:t})}$ w_i: importance weight

Importance Sampling: Procedure

1. Draw N samples $x_{0:t}^{(i)}$ from proposal distribution q(.).

$$x_{0:t}^{(i)} \sim q(x_{0:t} \mid y_{1:t})$$

2. Compute importance weight

$$w(x_{0:t}^{(i)}) = \frac{p(x_{0:t}^{(i)} \mid y_{1:t})}{q(x_{0:t}^{(i)} \mid y_{1:t})}$$

3. Estimate an arbitrary function f(.):

$$E[f(x_{0:t} \mid y_{1:t})] \approx \sum_{i=1}^{N} f(x_{0:t}^{(i)}) \tilde{w}_t^{(i)}, \qquad \tilde{w}_t^{(i)} = \frac{w(x_{0:t}^{(i)})}{\sum_{j=1}^{N} w(x_{0:t}^{(j)})}$$

Sequential Importance Sampling (SIS): Recursive Estimation

Augmenting the samples

$$q(x_{0:t} \mid y_{1:t}) = q(x_{0:t-1} \mid y_{1:t-1}) q(x_t \mid x_{0:t-1}, y_{1:t})$$
$$= q(x_{0:t-1} \mid y_{1:t-1}) q(x_t \mid x_{t-1}, y_t)$$

$$x_t^{(i)} \sim q(x_t \mid x_{t-1}, y_t)$$

(cf. non-sequential IS: $x_t^{(i)} \sim q(x_{0:t} \mid y_{1:t})$)

Weight update

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(y_t \mid x_t^{(i)}) p(x_t^{(i)} \mid x_{t-1}^{(i)})}{q(x_t^{(i)} \mid x_{t-1}^{(i)}, y_t)}$$

Sequential Importance Sampling: Idea

- Update filtering density using Bayesian filtering
- Compute integrals using importance sampling

- The filtering density $p(x_t \mid y_{1:t})$ is represented using particles and their weights

$$\{x_t^{(i)}, w_t^{(i)}\}_{i=1}^{N}$$

- Compute weights using:

$$w_t^{(i)} = \frac{p(x_t^{(i)}, y_{1:t})}{q(x_t^{(i)}, y_{1:t})}$$

Sequential Importance Sampling: Procedure

1. Particle generation $x_t^{(i)} \sim q(x_t \mid x_{t-1}^{(i)}, y_t) = p(x_t \mid x_{t-1}^{(i)})$

2. Weight computation $w_t^{(i)} = w_{t-1}^{(i)} p(y_t \mid x_t^{(i)})$

Weight normalization $\tilde{w}_t^{(i)} = \dfrac{w_t^{(i)}}{\sum_{j=1}^{N} w_t^{(j)}}$

3. Estimation computation $E[f(x_t \mid y_{1:t})] = \sum_{i=1}^{N} f(x_t^{(i)}) \tilde{w}_t^{(i)}$

Note: Step 1 above assumes the proposal density to be the prior. This does not use the information from observations. Alternatively, the proposal density could be

$$x_t^{(i)} \sim q(x_t \mid x_{t-1}^{(i)}, y_t) = p(x_t \mid x_{t-1}^{(i)}, y_t)$$

that minizes the variance of w_t (Doucet el al., 1999).

Resampling

- SIS suffers from degeneracy problems, i.e. a small number of particles have big weights and the rest have extremely small values.
- Remedy: SIR introduces a selection (resampling) step to eliminate samples with low importance ratios (weights) and multiply samples with high importance ratios.
- Resampling maps the weighted random measure on to the equally weighted random measure by sampling uniformly with replacement from $\{x_{0:t}^{(i)}\}_{i=1}^{N}$ with probabilities $\{w_t^{(i)}\}_{i=1}^{N}$:

$$\{\tilde{x}_{0:t}^{(i)}, N^{-1}\}_{i=1}^{N} \sim \{x_{0:t}^{(i)}, w_t^{(i)}(x_{0:t}^{(i)})\}_{i=1}^{N}$$

Sampling Importance Resampling (SIR) = Sequential Monte Carlo = Particle Filter

1. Initialize $t \leftarrow 0$

   - For $i = 1, \ldots, N$: sample $x_t^{(i)} \sim p(x_0)$, $\quad t \leftarrow 1$.

2. Importance sampling

   - For $i = 1, \ldots, N$: sample $x_t^{(i)} \sim q(x_t \mid x_{t-1}^{(i)}, y_t) = p(x_t \mid x_{t-1}^{(i)})$

     Let $x_{0:t}^{(i)} \sim (x_{0:t-1}^{(i)}, x_t^{(i)})$

   - For $i = 1, \ldots, N$: compute weights $\quad w_t^{(i)} = p(y_t \mid x_t^{(i)})$

   - Normalize the weights: $\tilde{w}_t^{(i)} = w_t^{(i)} \Big/ \sum_{j=1}^{N} w_t^{(j)}$

3. Resampling

   - Resample with replacement N particles $x_{0:t}^{(i)}$ according to the importance weights $w_t^{(i)}$, resulting in $\{\tilde{x}_{0:t}^{(i)}, N^{-1}\}_{i=1}^{N}$.

   - New particle population $\{x_{0:t}^{(i)}\}_{i=1}^{N} \leftarrow \{\tilde{x}_{0:t}^{(i)}\}_{i=1}^{N}$.

   - Set $t \leftarrow t+1$ and go to step 2.

Visualization of Particle Filter

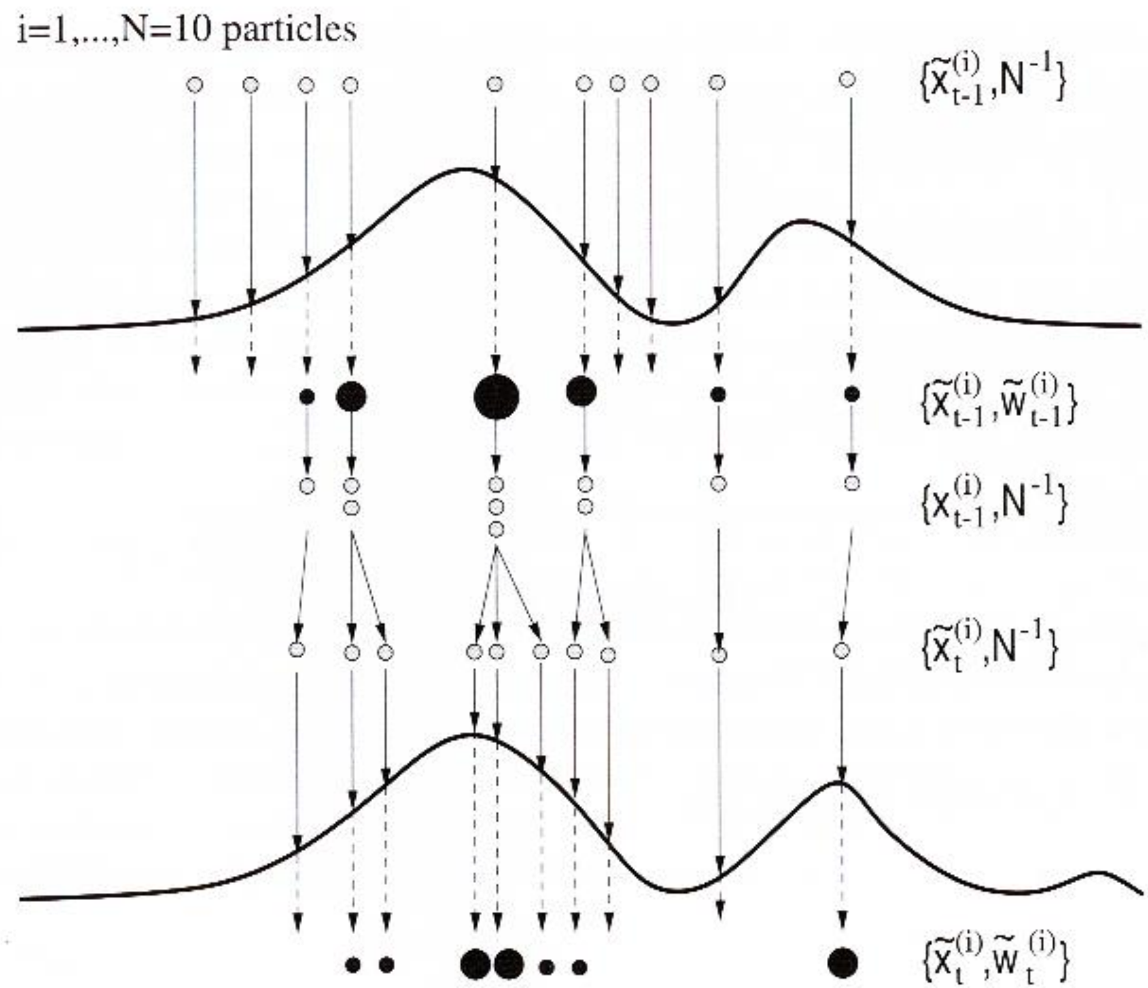
unweighted measure

compute importance weights ⇒ $p(x_{t-1}|z_{1:t-1})$

resampling

move particles

predict $p(x_t|z_{1:t-1})$



References (Slides)

- Bolic, M., Theory and Implementation of Particle Filters, University of Ottawa, slide file, 2004.
- Copsey, K., Tutorial on Particle Filters, DERA Malvern, slide file, 2001.
- Dellaert, F., Tutorial on Monte Carlo Methods: Part 2 - Particle Filters, Georgia Inst. of Tech., slide file 2007.
- Muehlich, M., Particle Filters: An Overview, University of Frankfurt, slide file, 2003.
- Pfeiffer, M., A Brief Introduction to Particle Filters, TU Graz, slide file, 2004.