

# Sequential Monte Carlo

---



Reverend Thomas Bayes 1702-1761

***Nicholas Zabaras***

***Materials Process Design and Control Laboratory  
Sibley School of Mechanical and Aerospace Engineering***

***188 Rhodes Hall***

***Cornell University***

***Email: [zabaras@cornell.edu](mailto:zabaras@cornell.edu)***

***URL: <http://mpdc.mae.cornell.edu/>***

# Sequential Importance Sampling

- When sampling from distributions on spaces of different dimensions.
- Suppose we need to sample  $\mathbf{x} = (x_1, x_2, \dots, x_d)$  from  $\pi(\mathbf{x})$ .
- The target can be rewritten as

$$\pi(\mathbf{x}) = \pi(x_1)\pi(x_2 | x_1)\dots\pi(x_d | x_1, \dots, x_{d-1})$$

- Similarly, we can construct an importance sampling density sequentially:

$$q(\mathbf{x}) = q_1(x_1)q_2(x_2 | x_1)\dots q_d(x_d | x_1, \dots, x_{d-1})$$

- Accordingly, the importance weight is

$$w(\mathbf{x}) = \frac{\pi(\mathbf{x})}{q(\mathbf{x})} = \frac{\pi(x_1)\pi(x_2 | x_1)\dots\pi(x_d | x_1, \dots, x_{d-1})}{q_1(x_1)q_2(x_2 | x_1)\dots q_d(x_d | x_1, \dots, x_{d-1})}$$

# Sequential Importance Sampling

- Knowing that  $\mathbf{x}_{1:t} = (x_1, \dots, x_t)$  and  $\mathbf{x}_{1:d} = \mathbf{x}$ , this suggests a recursive way of computing and monitoring the importance weights

$$w_t(\mathbf{x}_{1:t}) = w_{t-1}(\mathbf{x}_{1:t-1}) \frac{\pi(x_t | \mathbf{x}_{1:t-1})}{q_t(x_t | \mathbf{x}_{1:t-1})}, \quad w_1(x_1) = \frac{\pi(x_1)}{q_1(x_1)}$$

- Potential advantages of this recursion:
  - Be selective: we can stop generating further components of  $\mathbf{x}$  if the *partial weight*  $w_t$  ( $t < d$ ) is too small.
  - Be adaptive: we can take advantage of  $\pi(x_t | \mathbf{x}_{1:t-1})$  in designing  $q_t(x_t | \mathbf{x}_{1:t-1})$ , rather than selecting one a priori.
- Although the idea seems promising, it is impractical, because determining the conditional  $\pi(x_t | \mathbf{x}_{1:t-1})$  requires the marginal  $\pi(\mathbf{x}_{1:t-1}, x_t)$ :

$$\pi(\mathbf{x}_{1:t}) = \int \pi(x_1, \dots, x_d) dx_{t+1} \dots dx_d$$

which can be even more difficult than the original problem!

# Sequential Importance Sampling

- In order to carry out the sequential sampling idea, we suppose that a sequence of auxiliary distributions,  $\rho_1(x_1), \rho_2(\mathbf{x}_{1:2}), \dots, \rho_d(\mathbf{x}_{1:d})$ , can be found to approximate the marginal distribution  $\pi(\mathbf{x}_{1:t})$ , for  $t=1, \dots, d$ .
  - $\rho_d(\mathbf{x}_{1:d}) = \pi(\mathbf{x}_{1:d}) = \pi(\mathbf{x})$
  - the rest are reasonable approximations to the marginals
  - $\rho_t(\mathbf{x}_{1:t})$  are only required to be known up to a normalization constant.

$\rho_t(\mathbf{x}_{1:t}) = \gamma_t(\mathbf{x}_{1:t}) / Z_t$  and  $\gamma_t(\mathbf{x}_{1:t}) = 1$ . Thus approximate  $\pi(\mathbf{x}_{1:t})$  by  $\gamma_t(\mathbf{x}_{1:t})$

- The sequential importance sampling can then be defined as the following recursive procedure.
  - Draw  $X_t = x_t$  from  $q_t(x_t | \mathbf{x}_{1:t-1})$ , and let  $\mathbf{x}_{1:t} = (\mathbf{x}_{1:t-1}, x_t)$ .
  - Compute weight as:

$$w_t(\mathbf{x}_{1:t}) = w_{t-1}(\mathbf{x}_{1:t-1}) u_t$$

where the incremental weight

$$u_t = \frac{\gamma_t(\mathbf{x}_{1:t})}{\gamma_{t-1}(\mathbf{x}_{1:t-1}) q_t(x_t | \mathbf{x}_{1:t-1})}$$

# Sequential Importance Sampling

- This SIS approximate the difficult target density by breaking it into manageable pieces.
- Is this process correct?

- Note  $w_1(x_1) = \frac{\pi(x_1)}{q_1(x_1)} = \frac{\gamma_1(x_1)}{q_1(x_1)}$

$$\begin{aligned}w_t(\mathbf{x}_{1:t}) &= w_1(x_1) \prod_{j=2}^t \frac{\gamma_j(\mathbf{x}_{1:j})}{\gamma_{j-1}(\mathbf{x}_{1:j-1}) q_j(x_j | \mathbf{x}_{1:j-1})} \\ &= \frac{\gamma_1(x_1)}{q_1(x_1)} \frac{\gamma_2(\mathbf{x}_{1:2})}{\gamma_1(x_1) q_2(x_2 | x_1)} \cdots \frac{\gamma_t(\mathbf{x}_{1:t})}{\gamma_{t-1}(\mathbf{x}_{1:t-1}) q_t(x_t | \mathbf{x}_{1:t-1})} \\ &= \frac{\gamma_t(\mathbf{x}_{1:t})}{q_1(x_1) q_2(x_2 | x_1) \cdots q_t(x_t | \mathbf{x}_{1:t-1})} \\ &= \frac{\gamma_t(\mathbf{x}_{1:t})}{q_t(\mathbf{x}_{1:t})}\end{aligned}$$

# Sequential Importance Sampling

- From the previous slides, we constructed

$$w_t(\mathbf{x}_{1:t}) = w_{t-1}(\mathbf{x}_{1:t-1}) \frac{\gamma_t(x_t | \mathbf{x}_{1:t-1})}{q_t(x_t | \mathbf{x}_{1:t-1})} = w_{t-1}(\mathbf{x}_{1:t-1}) \frac{\gamma_t(\mathbf{x}_{1:t})}{\gamma_{t-1}(\mathbf{x}_{1:t-1}) q_t(x_t | \mathbf{x}_{1:t-1})},$$

$$w_1(x_1) = \frac{\gamma_1(x_1)}{q_1(x_1)}$$

- A good sequence of auxiliary distributions  $\{\rho_t\}$  can help us build good trial densities  $\{q_t\}$ . For example, we can choose

$$q_t(x_t | \mathbf{x}_{1:t-1}) = \rho_t(x_t | \mathbf{x}_{1:t-1}) = \frac{\gamma_t(\mathbf{x}_{1:t})}{\gamma_t(\mathbf{x}_{1:t-1})}$$

in which case:

$$w_t(\mathbf{x}_{1:t}) = w_{t-1}(\mathbf{x}_{1:t-1}) \frac{\gamma_t(\mathbf{x}_{1:t-1})}{\gamma_{t-1}(\mathbf{x}_{1:t-1})}$$

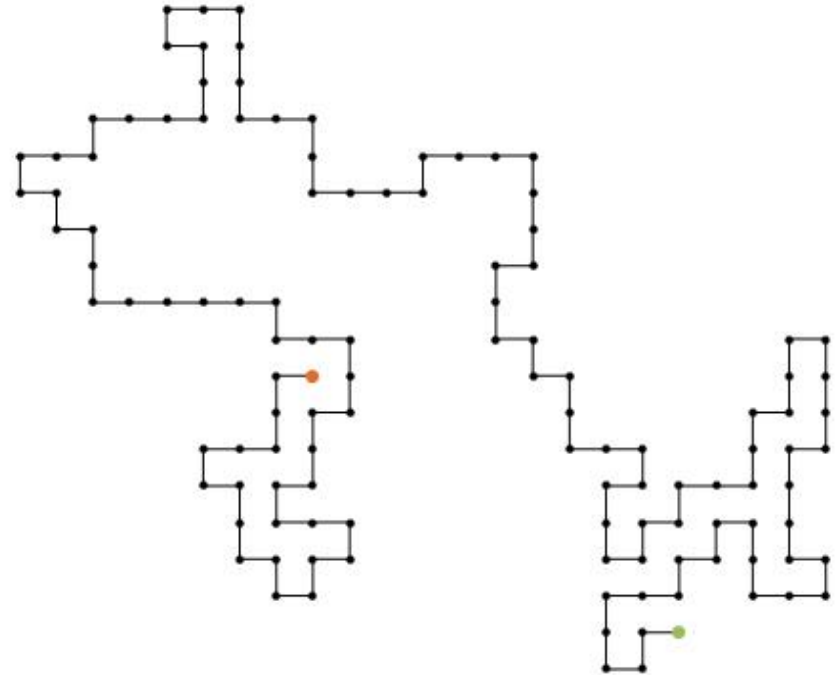
it does not depend on the current state  $x_t$ .

# Example: Growing A Polymer

- Growing a polymer on Lattice (Hammersley & Morton 1954)
  - A polymer chain is approximated by a self-avoiding path on a square lattice.
  - Its target distribution is assumed to be uniform  $\pi(\mathbf{x}) = 1 / Z_T$ .  $Z_T$  is the number of paths (the total number of different self-avoiding random walk (SAW) with  $T$  atoms) of length  $T$  and which we do not know.
  - Of interest is to estimate certain descriptive statistics, i.e.

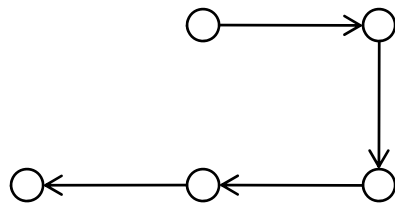
$$E \left[ \|x_T - x_1\|^2 \right] = \int \|x_T - x_1\|^2 dx$$

The mean squared extension of the chain.



# Example: Growing A Polymer

- We need a sequence of intermediate distributions.
- A natural selection will be to select the uniform distributions  $1/Z_t$  on the space of polymers of length  $t$  for  $t = 1, \dots, T$ .
- Note that since  $Z_t$  are unknown. These distributions are only known up to a constant i.e.  $\rho_t(\mathbf{x}_{1:t}) = 1/Z_t$ , and  $\gamma_t(\mathbf{x}_{1:t}) = 1$ .
- We start all paths from  $x_1 = (0, 0)$  and always move to  $x_2 = (1, 0)$  (due to symmetry).
- At step  $t$  it is located at node  $x_t = (i, j)$  and it can move to any of its neighbors  $(i \pm 1, j)$ ,  $(i, j \pm 1)$  as long as it has not been there before.
- Let  $n_t(\mathbf{x}_{1:t})$  the “available” nodes after step  $t$  (obviously  $n_t \leq 3$ ).



$$1/3 \times 1/3 \times 1/3 \times 1/2$$



$$1/3 \times 1/3 \times 1/3 \times 1/3$$

# Example: Growing A Polymer

- Under  $\rho_t(\mathbf{x}_{1:t})=1/Z_t$ , the marginal distribution of the partial sample  $\mathbf{x}_{1:t-1}=\{x_1, \dots, x_{t-1}\}$  is

$$\rho_t(\mathbf{x}_{1:t-1}) = \sum_{\text{all possible } x_t} \rho_t(\mathbf{x}_{1:t-1}, x_t) = \frac{n_{t-1}}{Z_t}$$

- Thus, the conditional distribution  $\rho_t(x_t|\mathbf{x}_{1:t-1})$  is

$$\rho_t(x_t | \mathbf{x}_{1:t-1}) = \frac{\rho_t(\mathbf{x}_{1:t})}{\rho_t(\mathbf{x}_{1:t-1})} = \frac{1}{n_{t-1}}$$

- We can use this as the importance sampling density at step  $t$ :

$$q_t(x_t | \mathbf{x}_{1:t-1}) = \rho_t(x_t | \mathbf{x}_{1:t-1}) = \frac{1}{n_{t-1}}$$

which implies selecting one the available (have never been visited) neighbors with equal probability.

- Note: this “growth” method tends to bias in favor of more “compact” configurations. The correct for the bias is by introducing the weights.

# Example: Growing A Polymer

- This suggests that the importance sampling weights

$$w_t(\mathbf{x}_{1:t}) = w_{t-1}(\mathbf{x}_{1:t-1}) \frac{\gamma_t(x_{1:t-1})}{\gamma_{t-1}(x_{1:t-1})} = w_{t-1}(\mathbf{x}_{1:t-1}) n_{t-1}$$

if  $n_{t-1} > 0$ , otherwise  $w_t = 0$ .

- Hence ultimately each sample  $\mathbf{x}_{1:T}$  will have a weight:

$$w_T(\mathbf{x}_{1:T}) = n_1 n_2 \cdots n_T$$

- The statistics can be studied by repeating the sampling procedure multiple times.
- Note: The multiple sampling process can be done in parallel. More precisely, at stage 1, we generate  $m$  i.i.d samples  $\{x_1^{(1)}, \dots, x_1^{(m)}\}$  from  $q_1$ ; at stage 2, we generate  $x_2^{(j)}$  from  $q_2(\cdot | x_1^{(j)})$  for  $j = 1, \dots, m$  and so on.

Sequential Monte Carlo (SMC) is taking over Markov Chain Monte Carlo (MCMC) in some areas due to its parallel nature and adaptive nature for transient problems where the transient kernel changes over time.

## **Applications: Inverse Problems**

# Inverse Problem

---

- Solution of an inverse problem entails determining unknown *causes* based on observation of their *effects*.

- known: results
- unknown: causes
- For example, elasticity problem

Forward problem:

known: boundary conditions (tractions, displacements)

unknown: internal strains, displacements

Inverse problem:

known: internal strains, displacements

unknown: boundary conditions

- Generally, an inverse problem is to find  $\mathbf{x}$  such that (at least approximately)

$$\mathbf{y} = \mathbf{G}(\mathbf{x})$$

where  $\mathbf{G}$  is a operator describing the relationship between observed data  $\mathbf{y}$  and model parameters  $\mathbf{x}$ .

# Filtering

➤ Filtering is the problem of sequentially estimating the states  $\{X\}$  (parameters or hidden variables) of a system as a set of observations  $\{Y\}$  become available on-line.

➤ General problem statement

- Let  $\{X_k\}_{k=0}^{\infty}, \{Y_k\}_{k=1}^{\infty}$  be two stochastic process.
- state vector  $X_k \in \mathbb{R}^{n_k}$ , represents quantities we are interested in.
- observations  $Y_k \in \mathbb{R}^{m_k}$ , represents the measurement.

➤ Postulates

- The process  $\{X_k\}_{k=0}^{\infty}$  is a Markov process

$$\pi(x_{k+1} | x_0, x_1, \dots, x_k) = \pi(x_{k+1} | x_k)$$

- The process  $\{Y_k\}_{k=1}^{\infty}$  is a Markov process w.r.t. the history of  $\{X_k\}_{k=0}^{\infty}$

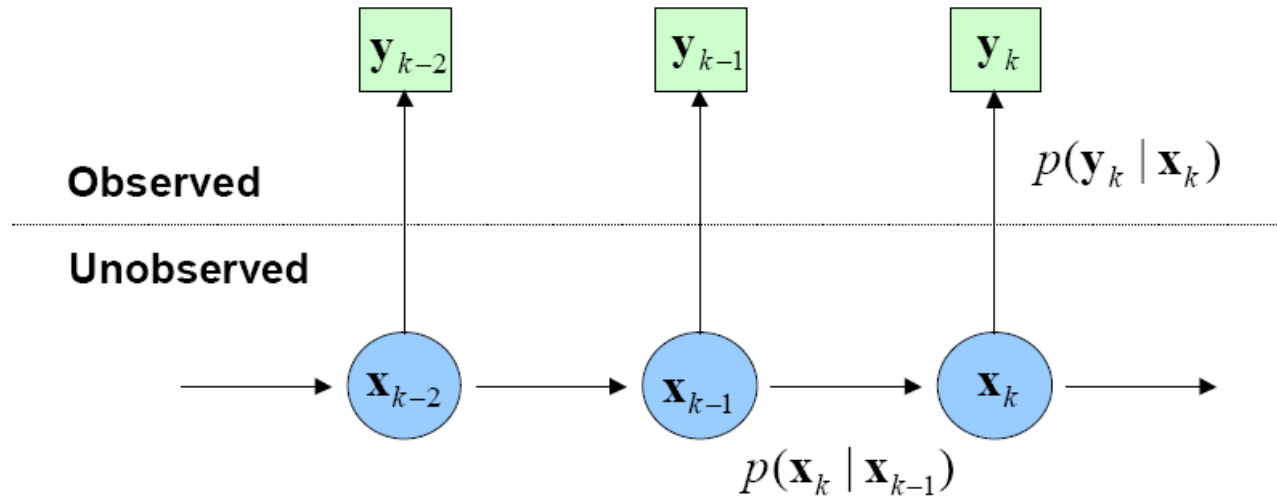
$$\pi(y_k | x_0, x_1, \dots, x_k) = \pi(y_k | x_k)$$

- The process  $\{X_k\}_{k=0}^{\infty}$  depends on the past observations only through its own history

$$\pi(x_{k+1} | x_k, y_1, \dots, y_k) = \pi(x_{k+1} | x_k)$$

# Filtering

- These postulates can be illustrated as



- The pairs satisfy these three properties are called *evolution-observation model*.
- In filtering problem, we are interested in estimating the conditional probability  $\pi(x_k | \mathbf{y}_{1:k})$ , the filtering density.

# Filtering

- Given the state space model, how do we recursively estimate the filtering density?

$$\begin{aligned}\pi(x_{k+1} | \mathbf{y}_{1:k+1}) &= \frac{\pi(\mathbf{y}_{1:k+1} | x_{k+1})\pi(x_{k+1})}{\pi(\mathbf{y}_{1:k+1})} \\ &= \frac{\pi(y_{k+1}, \mathbf{y}_{1:k} | x_{k+1})\pi(x_{k+1})}{\pi(\mathbf{y}_{1:k+1})} \\ &= \frac{\pi(y_{k+1} | \mathbf{y}_{1:k}, x_{k+1})\pi(\mathbf{y}_{1:k} | x_{k+1})\pi(x_{k+1})}{\pi(y_{k+1} | \mathbf{y}_{1:k})\pi(\mathbf{y}_{1:k})} \\ &= \frac{\pi(y_{k+1} | \mathbf{y}_{1:k}, x_{k+1})\pi(x_{k+1} | \mathbf{y}_{1:k})\pi(\mathbf{y}_{1:k})\pi(x_{k+1})}{\pi(y_{k+1} | \mathbf{y}_{1:k})\pi(\mathbf{y}_{1:k})\pi(x_{k+1})} \\ &= \frac{\pi(y_{k+1} | x_{k+1})\pi(x_{k+1} | \mathbf{y}_{1:k})}{\pi(y_{k+1} | \mathbf{y}_{1:k})}\end{aligned}$$

# Filtering

$$\pi(x_{k+1} | \mathbf{y}_{1:k+1}) = \frac{\text{likelihood} \quad \text{prior}}{\text{evidence}}$$
$$\pi(x_{k+1} | \mathbf{y}_{1:k+1}) = \frac{\pi(y_{k+1} | x_{k+1})\pi(x_{k+1} | \mathbf{y}_{1:k})}{\pi(y_{k+1} | \mathbf{y}_{1:k})}$$

posterior

- Prior: time evolution updating

$$\pi(x_{k+1} | \mathbf{y}_{1:k}) = \int \pi(x_{k+1} | x_k)\pi(x_k | \mathbf{y}_{1:k})dx_k$$

- Likelihood: defined in terms of observation model

- Evidence:

$$\pi(y_{k+1} | \mathbf{y}_{1:k}) = \int \pi(y_{k+1} | x_{k+1})\pi(x_k | \mathbf{y}_{1:k})dx_{k+1}$$

- To be completely specified, we need to know

- The probability density of initial state  $X_0$ .
  - The Markov transition kernels  $\pi(x_{k+1} | x_k), k = 0, 1, 2, \dots$  (needs not be time homogeneous).
  - The likelihood functions  $\pi(y_k | x_k), k = 1, 2, \dots$
- } Hidden Markov process
- Observed process

# Filtering

➤ Then we need to find formulas for the updating steps

▪ Time evolution updating:

$$\pi(x_k | \mathbf{y}_{1:k}) \xrightarrow{\pi(x_{k+1}|x_k)} \pi(x_{k+1} | \mathbf{y}_{1:k})$$

▪ Observation updating:

$$\pi(x_{k+1} | \mathbf{y}_{1:k}) \xrightarrow{\pi(y_{k+1}|x_{k+1})} \pi(x_{k+1} | \mathbf{y}_{1:k+1})$$

➤ Those equations can be calculated by marginals

▪ Time evolution updating

$$\pi(x_{k+1} | \mathbf{y}_{1:k}) = \int \pi(x_{k+1} | x_k) \pi(x_k | \mathbf{y}_{1:k}) dx_k$$

▪ Observation updating

$$\pi(x_{k+1} | \mathbf{y}_{1:k+1}) = \frac{\pi(y_{k+1} | x_{k+1}) \pi(x_{k+1} | \mathbf{y}_{1:k})}{\pi(y_{k+1} | \mathbf{y}_{1:k})}$$

where

$$\pi(y_{k+1} | \mathbf{y}_{1:k}) = \int \pi(y_{k+1} | x_{k+1}) \pi(x_{k+1} | \mathbf{y}_{1:k}) dx_{k+1}$$

Proof can be found in Kaipio's *Statistical Computational Inverse Problem*.

# Filtering

- General Problem: Assume a Markov model describing the evolution of states  $X_k$  and an observation model for  $Y_k$  depending on the current state  $X_k$ ,

$$X_{k+1} = F_{k+1}(X_k, W_{k+1}), \quad k = 0, 1, 2, \dots$$

$$Y_k = G_k(X_k, V_k), \quad k = 1, 2, \dots$$

$F_{k+1}$  and  $G_k$ : known functions;

$W_{k+1} \in \mathbb{R}^{p_{k+1}}$ : state noise;  $V_k \in \mathbb{R}^{q_k}$ : observation noise.

- The inverse problem considered here is to extract information of the state vectors  $X_k$  based on the measurements  $Y_k$ .
- Example:
  - Consider a dynamical system governed by a set of ODEs:  $\dot{x}_t = h_t(x_t)$
  - This might be inherently stochastic or there might be uncertainty w.r.t. the fidelity of the model. e.g.  $\dot{x}_t = h_t(x_t) + noise$
  - We rarely observe  $x_t$  directly, but usually a (noisy) function of  $x_t$ :  
 $y_t = r_t(x_t) + noise$
  - The observables  $y_t$  can be of lower or higher dimension than  $x_t$ .

# Kalman Filters

- A simple case (Harvey, 1989; Anderson and Moore, 1979)
- In cases where the state space model is linear and Gaussian, the classic Kalman filter is optimal. The state equations are linear with additive noise processes,

$$X_{k+1} = F_{k+1} X_k + W_{k+1}, \quad k = 0, 1, 2, \dots$$

$$Y_k = G_k X_k + V_k, \quad k = 1, 2, \dots$$

Here we assume  $F_{k+1}$  and  $G_k$  are known matrices. The noise vectors  $W_{k+1}$  and  $V_k$  are Gaussian with known means and covariances (for simplicity, mean zero vectors). These noises are independent over time and also mutually independent, that is,

$$W_k \perp W_l, \quad V_k \perp V_l, \quad \text{for } k \neq l$$

and 
$$W_k \perp V_l, \quad \text{for all } k \text{ and } l$$

We also require that the initial state  $X_0$  be Gaussian distributed, and without a loss of generality, the mean of  $X_0$  is assumed to be zero.

# Kalman Filters

- Assume that the pair  $\{X_k\}_{k=0}^{\infty}, \{Y_k\}_{k=0}^{\infty}$  of stochastic processes is an evolution-observation model.
- Time evolution updating is

$$\pi(x_{k+1} | \mathbf{y}_{1:k}) = \int \pi(x_{k+1} | x_k) \pi(x_k | \mathbf{y}_{1:k}) dx_k$$

- Observation updating is

$$\pi(x_{k+1} | \mathbf{y}_{1:k+1}) = \frac{\pi(y_{k+1} | x_{k+1}) \pi(x_{k+1} | \mathbf{y}_{1:k})}{\pi(y_{k+1} | \mathbf{y}_{1:k})}$$

- We Denote  $x_{k|l} = E(x_k | \mathbf{y}_{1:l})$ ,  $\Gamma_{k|l} = \text{cov}(x_k | \mathbf{y}_{1:l})$  and  $\pi(x_0) = \pi(x_0 | y_0)$ .  
Then based on the linear Gaussian assumption, the above time evolution and observation updating formulas take the following forms:

# Kalman Filters

- Time evolution updating: Assume we know

$$\pi(x_k | \mathbf{y}_{1:k}) \sim N(x_{k|k}, \Gamma_{k|k})$$

and 
$$\pi(x_{k+1} | \mathbf{y}_{1:k}) \sim N(x_{k+1|k}, \Gamma_{k+1|k})$$

where

$$x_{k+1|k} = F_{k+1} x_{k|k},$$

$$\Gamma_{k+1|k} = F_{k+1} \Gamma_{k|k} F_{k+1}^T + \Gamma_{w_{k+1}}$$

in which  $\Gamma_{w_{k+1}}$  is the covariance of the noise.

- Observation updating: knowing that

$$\pi(x_{k+1} | \mathbf{y}_{1:k}) \sim N(x_{k+1|k}, \Gamma_{k+1|k})$$

then, 
$$\pi(x_{k+1} | \mathbf{y}_{1:k+1}) \sim N(x_{k+1|k+1}, \Gamma_{k+1|k+1})$$

where 
$$x_{k+1|k+1} = x_{k+1|k} + K_{k+1} (y_{k+1} - G_{k+1} x_{k+1|k}),$$

$$\Gamma_{k+1|k+1} = (1 - K_{k+1} G_{k+1}) \Gamma_{k+1|k}$$

The Kalman gain matrix is given by

$$K_{k+1} = \Gamma_{k+1|k} G_{k+1}^T \left( G_{k+1} \Gamma_{k+1|k} G_{k+1}^T + \Gamma_{v_{k+1}} \right)^{-1}$$

# Kalman Filters

- The complete Kalman filtering recursion can be summarized as
  - given the state equations that are linear with additive noise processes

$$X_{k+1} = F_{k+1} X_k + W_{k+1}, \quad k = 0, 1, 2, \dots$$

$$Y_k = G_k X_k + V_k, \quad k = 1, 2, \dots$$

- Time evolution updating

$$x_{k+1|k} = F_{k+1} x_{k|k},$$

$$\Gamma_{k+1|k} = F_{k+1} \Gamma_{k|k} F_{k+1}^T + \Gamma_{w_{k+1}}$$

- Calculate Kalman gain matrix

$$K_{k+1} = \Gamma_{k+1|k} G_{k+1}^T \left( G_{k+1} \Gamma_{k+1|k} G_{k+1}^T + \Gamma_{v_{k+1}} \right)^{-1}$$

- Observation updating

$$x_{k+1|k+1} = x_{k+1|k} + K_{k+1} (y_{k+1} - G_{k+1} x_{k+1|k}),$$

$$\Gamma_{k+1|k+1} = (1 - K_{k+1} G_{k+1}) \Gamma_{k+1|k}$$

# Particle filters

- In practical, the observation and evolution models may be cumbersome or impossible to linearize. One may try to use Monte Carlo methods to simulate the distribution by random samples. Such methods are known as particle filters.

- General discrete-time nonlinear, non-Gaussian dynamics system

$$X_{k+1} = F_{k+1}(X_k, W_{k+1}), \quad k = 0, 1, 2, \dots$$

$$Y_k = G_k(X_k, V_k), \quad k = 1, 2, \dots$$

- Goal: Produce sequentially an ensemble of random samples  $\{x_k^{(1)}, x_k^{(2)}, \dots, x_k^{(N)}\}$  distributed according to the conditional probability distributions  $\pi(x_k | \mathbf{y}_{1:k})$ , which is the marginal of the posterior:

$$\pi(x_{1:k} | y_{1:k}) \propto \underbrace{\mu_0(x_0)}_{\text{Initial condition}} \underbrace{\prod_{j=1}^k \pi(x_j | x_{j-1})}_{\text{prior}} \underbrace{\prod_{j=0}^k \pi(y_j | x_j)}_{\text{likelihood}}$$

- Vectors  $x_k^{(j)}$  are called particles of the sample.

# Particle filters

- It is often impossible to sample directly from the true posterior

$$\pi(x_{k+1} | \mathbf{y}_{1:k+1})$$

- However, we can adopt Sequential Importance Sampling to sample from the known proposal distribution  $q(x_{k+1} | \mathbf{y}_{1:k+1})$  and make use of the following substitution

$$\begin{aligned} E[f(x_{k+1})] &= \int f(x_{k+1}) \frac{\pi(x_{k+1} | \mathbf{y}_{1:k+1})}{q(x_{k+1} | \mathbf{y}_{1:k+1})} q(x_{k+1} | \mathbf{y}_{1:k+1}) dx_{k+1} \\ &= \int f(x_{k+1}) \frac{\pi(\mathbf{y}_{1:k+1} | x_{k+1}) \pi(x_{k+1})}{\pi(\mathbf{y}_{1:k+1}) q(x_{k+1} | \mathbf{y}_{1:k+1})} q(x_{k+1} | \mathbf{y}_{1:k+1}) dx_{k+1} \\ &= \int f(x_{k+1}) \frac{w_{k+1}(x_{k+1})}{\pi(\mathbf{y}_{1:k+1})} q(x_{k+1} | \mathbf{y}_{1:k+1}) dx_{k+1} \end{aligned}$$

$$w_{k+1}(x_{k+1}) = \frac{\pi(\mathbf{y}_{1:k+1} | x_{k+1}) \pi(x_{k+1})}{q(x_{k+1} | \mathbf{y}_{1:k+1})}$$

# Particle filters

$$\begin{aligned} E[f(x_{k+1})] &= \frac{1}{\pi(\mathbf{y}_{1:k+1})} \int f(x_{k+1}) w_{k+1}(x_{k+1}) q(x_{k+1} | \mathbf{y}_{1:k+1}) dx_{k+1} \\ &= \frac{\int f(x_{k+1}) w_{k+1}(x_{k+1}) q(x_{k+1} | \mathbf{y}_{1:k+1}) dx_{k+1}}{\int \pi(\mathbf{y}_{1:k+1} | x_{k+1}) \pi(x_{k+1}) \frac{q(x_{k+1} | \mathbf{y}_{1:k+1})}{q(x_{k+1} | \mathbf{y}_{1:k+1})} dx_{k+1}} \\ &= \frac{\int f(x_{k+1}) w_{k+1}(x_{k+1}) q(x_{k+1} | \mathbf{y}_{1:k+1}) dx_{k+1}}{\int w_{k+1}(x_{k+1}) q(x_{k+1} | \mathbf{y}_{1:k+1}) dx_{k+1}} \end{aligned}$$

- Draw samples from  $q(x_{k+1} | \mathbf{y}_{1:k+1})$ , we can approximate expectations of interests as

$$E[f(x_{k+1})] = \frac{\frac{1}{N} \sum_{i=1}^N w_{k+1}(x_{k+1}^{(i)}) f(x_{k+1}^{(i)})}{\frac{1}{N} \sum_{i=1}^N w_{k+1}(x_{k+1}^{(i)})} = \sum_{i=1}^N \tilde{w}_{k+1}(x_{k+1}^{(i)}) f(x_{k+1}^{(i)})$$

The normalized importance weights are

$$\tilde{w}_{k+1}(x_{k+1}^{(i)}) = \frac{w_{k+1}(x_{k+1}^{(i)})}{\sum_{i=1}^N w_{k+1}(x_{k+1}^{(i)})}$$

# Particle filters

- Using the state space assumptions (1st order Markov / observational independence given state), the importance weights can be estimated recursively by (De Freitas (2000)).

$$w_{k+1} = w_k \frac{\pi(y_{k+1} | x_{k+1})\pi(x_{k+1} | x_k)}{q(x_{k+1} | \mathbf{x}_{1:k})}$$

- The choice of proposal distribution is critical in designing a successful particle filter.
- Requirements
  - Support of proposal distribution must include support of true posterior distribution.
  - Must include most recent observations.
- Most popular choice of proposal distribution does not satisfy these requirements though:

$$q(x_{k+1} | \mathbf{y}_{1:k+1}) = \pi(x_{k+1} | x_k)$$

# Particle filters

---

- One problem with this Sequential Importance Sampling is that the variance of the importance weights increase stochastically over time (Kong et al. (1994), Doucet et al. (1999)).
- Resampling the particles keeping them with high importance weights and discarding those with low weights can solve this problem.
- Here comes *Sampling-importance Resampling (SIR)* (Gordon, Salmond & Smith (1993))

Maps the  $N$  *unequally weighted particles* into a new set of  $N$  equally weighted samples.

$$\{x_k^{(i)}, \tilde{w}_k^{(i)}\} \rightarrow \{x_k^{(i)}, N^{-1}\}$$

# Particle filters

➤ A straightforward *Sampling Importance Resampling (SIR)* algorithm:

- 1. Draw a random sample  $\{x_0^n\}_{n=1}^N$  from the initial distribution  $\pi(x_0) = \pi(x_0 | y_0)$  of the random variable  $X_0$  and set  $k=0$ .
- 2. Prediction step: For given  $k \geq 0$ , let  $\{x_k^n\}_{n=1}^N$  be a sample distributed according to  $\pi(x_k | \mathbf{y}_{1:k})$ . Approximate the integral using MC

$$\pi(x_{k+1} | \mathbf{y}_{1:k}) = \int \pi(x_{k+1} | x_k) \pi(x_k | \mathbf{y}_{1:k}) dx_k \approx \frac{1}{N} \sum_{n=1}^N \pi(x_{k+1} | x_k^n)$$

- 3. Sample from predicted density: Draw one new particle  $\tilde{x}_{k+1}^n$  from  $\pi(x_{k+1} | x_k^n)$ ,  $1 \leq n \leq N$ .
- 4. Calculate relative likelihoods

$$w_{k+1}^n = \frac{1}{W} \pi(y_{k+1} | \tilde{x}_{k+1}^n), \quad W = \sum_{n=1}^N \pi(y_{k+1} | \tilde{x}_{k+1}^n)$$

- 5. Resample: Draw  $x_{k+1}^n$ ,  $1 \leq n \leq N$  from the set  $\{\tilde{x}_{k+1}^n\}$ , where the probability of drawing the particle  $\tilde{x}_{k+1}^n$  is  $w_{k+1}^n$ . Increase  $k \rightarrow k+1$  and repeat from 2.

# Particle filters

## ➤ Example:

- Consider a simple one dimensional model. Assume that for some  $k$ ,  $\pi(x_k | \mathbf{y}_{1:k})$  is a Rayleigh distribution,

$$\pi(x_k | \mathbf{y}_{1:k}) = x_k \exp\left(-\frac{1}{2} x_k^2\right), \quad x_k \geq 0.$$

- Let  $\{x_k^{(n)}\}$  be a random sample drawn from this distribution.
- Consider a simple random walk model,

$$X_{k+1} = X_k + W_{k+1}, \quad W_{k+1} \sim N(0, \sigma^2)$$

corresponding to the transition density

$$\pi(x_{k+1} | x_k) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2} (x_k - x_{k-1})^2\right)$$

- For each  $n$ , draw  $w^{(n)}$  from the density  $N(0, \sigma^2)$  and set

$$\tilde{x}_{k+1}^{(n)} = x_k^{(n)} + w^{(n)}$$

# Particle filters

- Thus we have produced the prediction cloud of particles. Due to our assumptions, the particles should be distributed according to the density

$$\begin{aligned}\pi(x_{k+1} | \mathbf{y}_{1:k}) &= \int \pi(x_{k+1} | x_k) \pi(x_k | \mathbf{y}_{1:k}) dx_k \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} \int_0^\infty x_k \exp\left(-\frac{1}{2}x_k^2 - \frac{1}{2\sigma^2}(x_k - x_{k+1})^2\right) dx_k\end{aligned}$$

- Assume that the observation model is again the simplest imaginable

$$Y_{k+1} = X_{k+1} + V_{k+1}, \quad V_{k+1} \sim N(0, \sigma^2)$$

then the likelihood density is

$$\pi(y_{k+1} | x_{k+1}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(y_{k+1} - x_{k+1})^2\right)$$

then the likelihood are simply

$$w_{k+1}^{(n)} = \frac{1}{W} \exp\left(-\frac{1}{2\sigma^2}(y_{k+1} - \tilde{x}_{k+1}^{(n)})^2\right)$$

# Particle filters

- Finally, we do the resampling: for every  $n$ ,  $1 \leq n \leq N$ , draw a random number  $\alpha \sim U([0,1])$  and set

$$x_{k+1}^{(n)} = \tilde{x}_{k+1}^{(l)}, \quad \text{when } \sum_{k=1}^{l-1} w_{k+1}^{(j)} < \alpha \leq \sum_{k=1}^l w_{k+1}^{(j)}$$

## Notes:

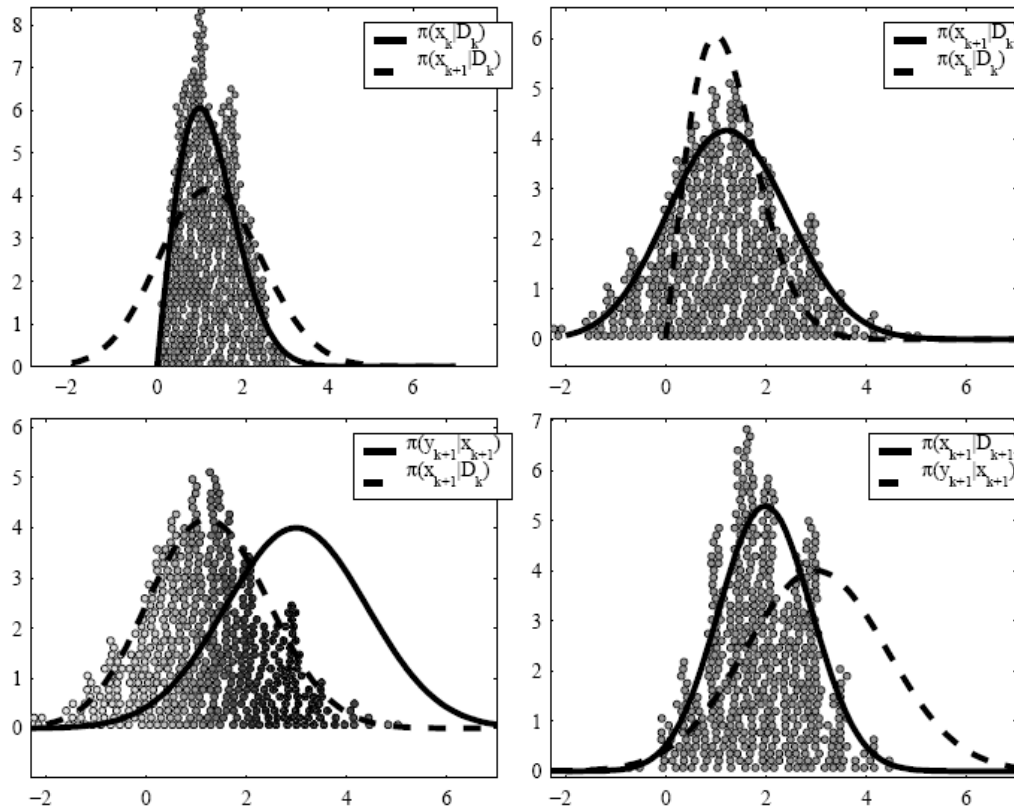
- Why resampling: the resampling reduces the variance of variable, it could give more accurate estimates.
- How large particles samples should be: A good understanding of the adequacy of the number of particles can be obtained by repeating the filtering with the same data  $M$  times.

$$\text{var}(\tilde{\theta}_g) \approx \frac{1}{M} \sum_{m=1}^M (\tilde{\theta}_{mg})^2 - (\bar{\theta}_g)^2$$

$$\int g(x_{k+1}) \pi(x_{k+1} | \mathbf{y}_{1:k}) dx_{k+1} \approx \frac{1}{N} \sum_{n=1}^N w_{k+1}^{(n)} g(\tilde{x}_{k+1}^{(n)}) = \tilde{\theta}_g$$

- The thinning (or impoverishment) of the sample: the relative likelihoods are very unevenly distributed.

# Particle filters



**Fig. 4.3.** Schematic representation of the SIR filtering. Top left: Particles drawn from  $\pi(x_k | D_k)$ . Top right: Particles propagated to approximate the predicted distribution  $\pi(x_{k+1} | D_k)$ . Bottom left: Relative likelihoods coded by a gray scale. Bottom right: Resampled particles approximating  $\pi(x_{k+1} | D_{k+1})$ .

From: Kaipio's [Statistical Computational Inverse Problem](#).