C.A.S.I.S
WORKSHOP

A Bayesian Approach
to
Nonlinear Statistical Signal Processing

James V. Candy

Chief Scientist for Engineering
Lawrence Livermore National Laboratory
Adjunct Professor UC Santa Barbara

November 16, 2006

ORGANIZATION

- Introduction
- Background into Bayesian approach
- Monte Carlo (MC) methods for Bayesian inference
- Sequential Bayesian processor (SBP)
- Model-based signal processing
- Bayesian approach to state-space processors
- Simulation-based MC approach to SBP
- Particle filtering for SBP
- Performance analysis
- Applications (nonlinear/multi-modal)
- Summary

INTRODUCTION

Bayesian signal processing

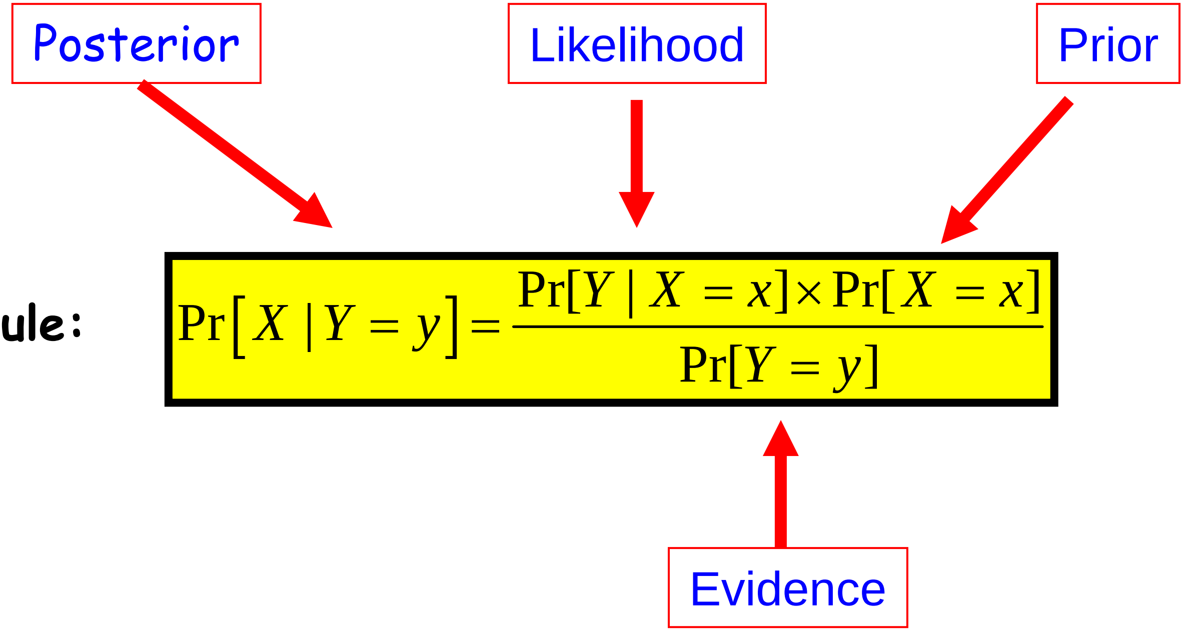
- is concerned with the estimation of the underlying probability distribution of a random signal in order to perform statistical inferences, that is,

$$\boxed{\hat{\Pr}[X \mid Y]}$$

- statistical inferences enable the extraction of the signal from noisy uncertain measurement data. For example, the maximum a-posterior (MAP)

$$\boxed{\hat{\Pr}[X \mid Y] \quad \Rightarrow \quad \hat{X} = \max_{X} \; \hat{\Pr}[X \mid Y]}$$

Bayesian techniques are based on BAYES' rule:

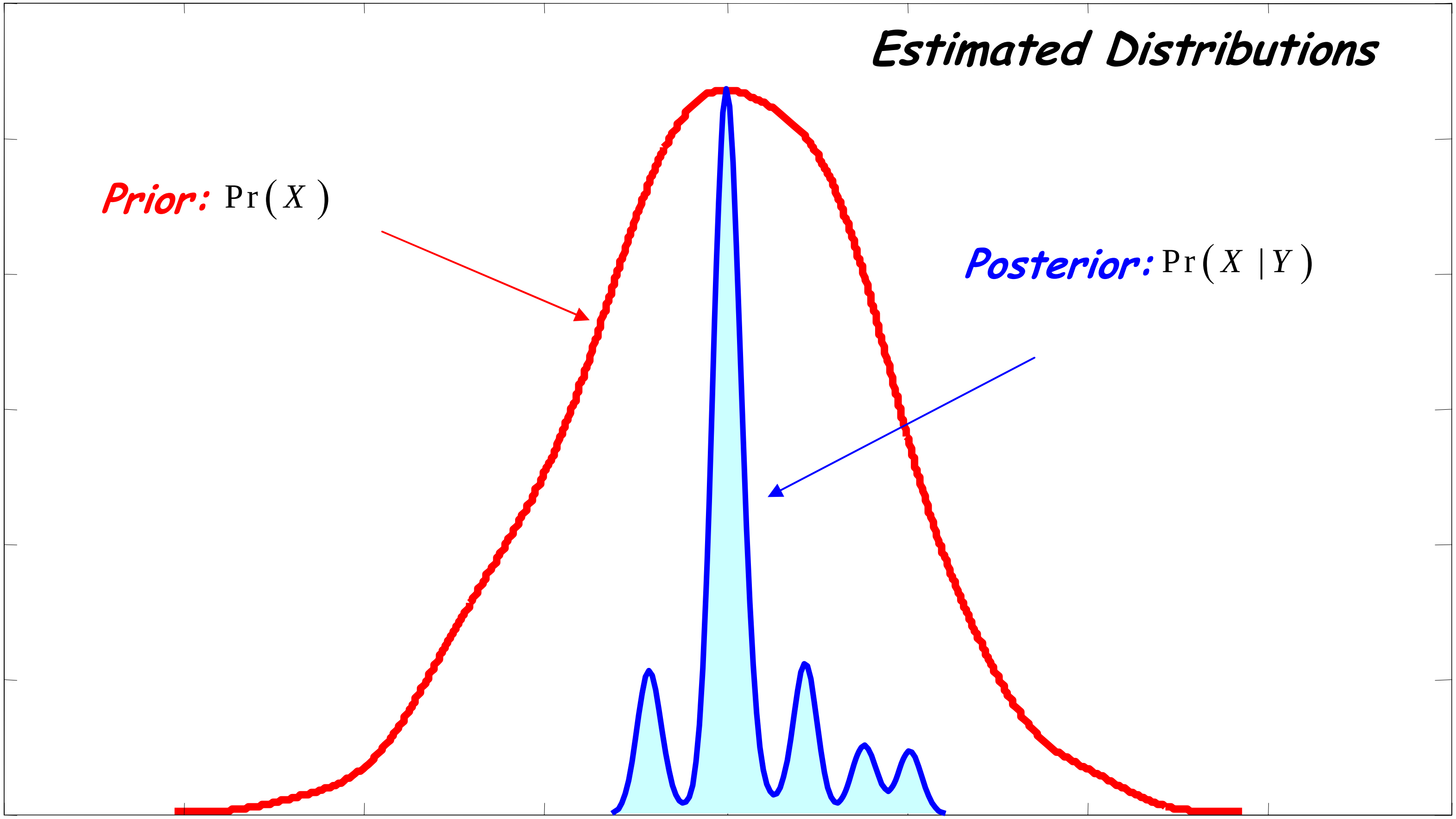


Bayes' Rule:

$$\Pr\left[X \mid Y = y\right] = \frac{\Pr[Y \mid X = x] \times \Pr[X = x]}{\Pr[Y = y]}$$

This "simple relationship" is the **principal foundation** of Bayesian signal processing both theoretically (derivations) and pragmatically (implementations)

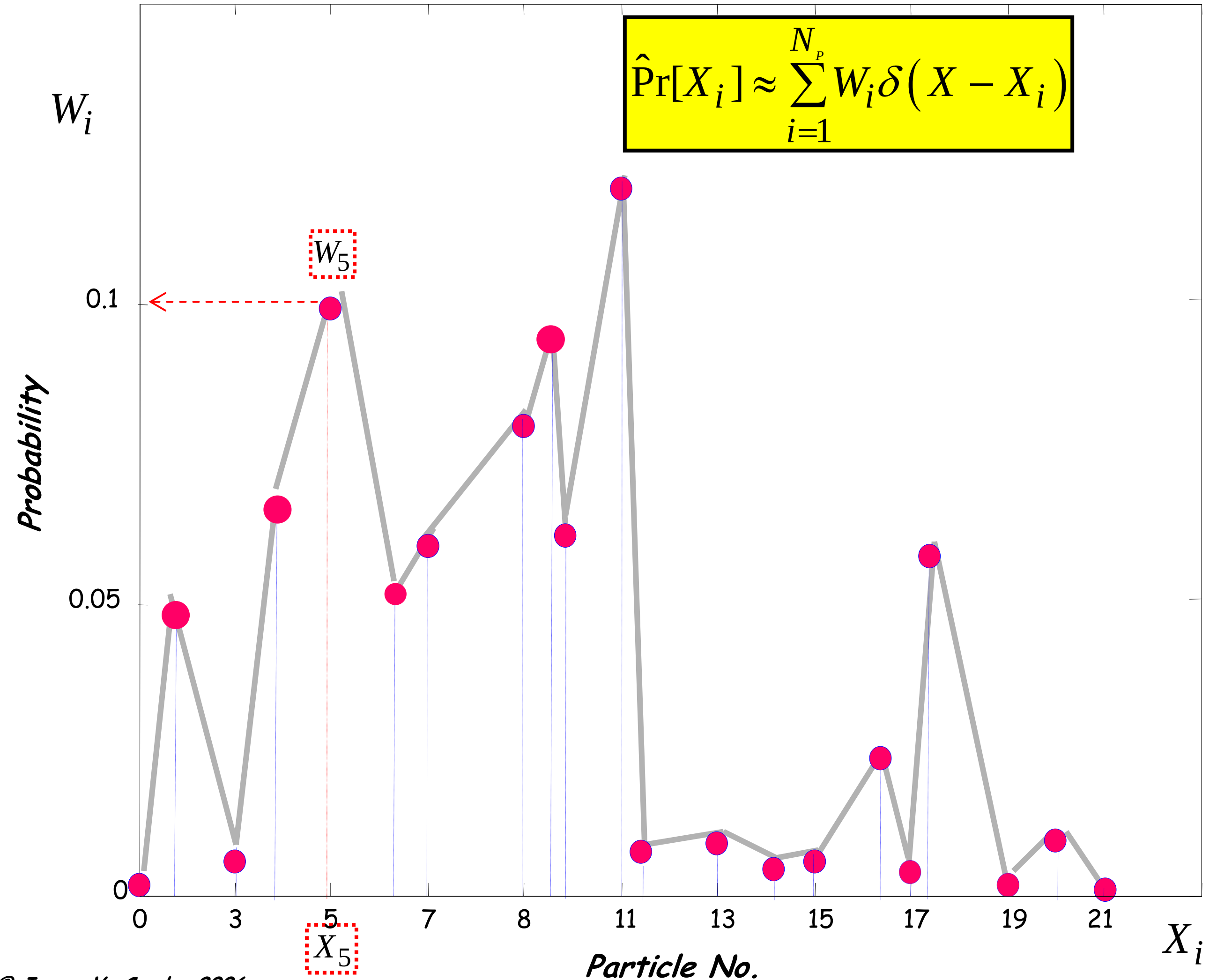
Bayesian techniques convert the prior→posterior



PARTICLE FILTERS (PF)

- PF are sequential (Monte Carlo) techniques in which the underlying posterior distribution of interest is represented by a “cloud” of random samples (particles)
- The PF is a processing algorithm that (sequentially) propagates and updates the random samples drawn from the previous stage to obtain a set of samples approximately distributed from the next stage

Particle filters can be used to construct the "empirical" posterior distributions. For example, a static parameter X given the data Y



$$\hat{\Pr}[X_i] \approx \sum_{i=1}^{N_P} W_i \delta\left(X - X_i\right)$$

Particle filters can be applied in many areas:

- **Signal processing**
  - Image processing and segmentation
  - Model selection
  - Tracking and navigation

- **Communications**
  - Channel estimation
  - Blind equalization
  - Positioning in wireless networks

- **Applications**
  - Biology & Biochemistry
  - Chemistry
  - Economics & Business
  - Genomics
  - Geosciences
  - Immunology
  - Materials Science
  - Pharmacology & Toxicology
  - Psychiatry/Psychology
  - Social Sciences

PFs: advantages and disadvantages*:

ADVANTAGES

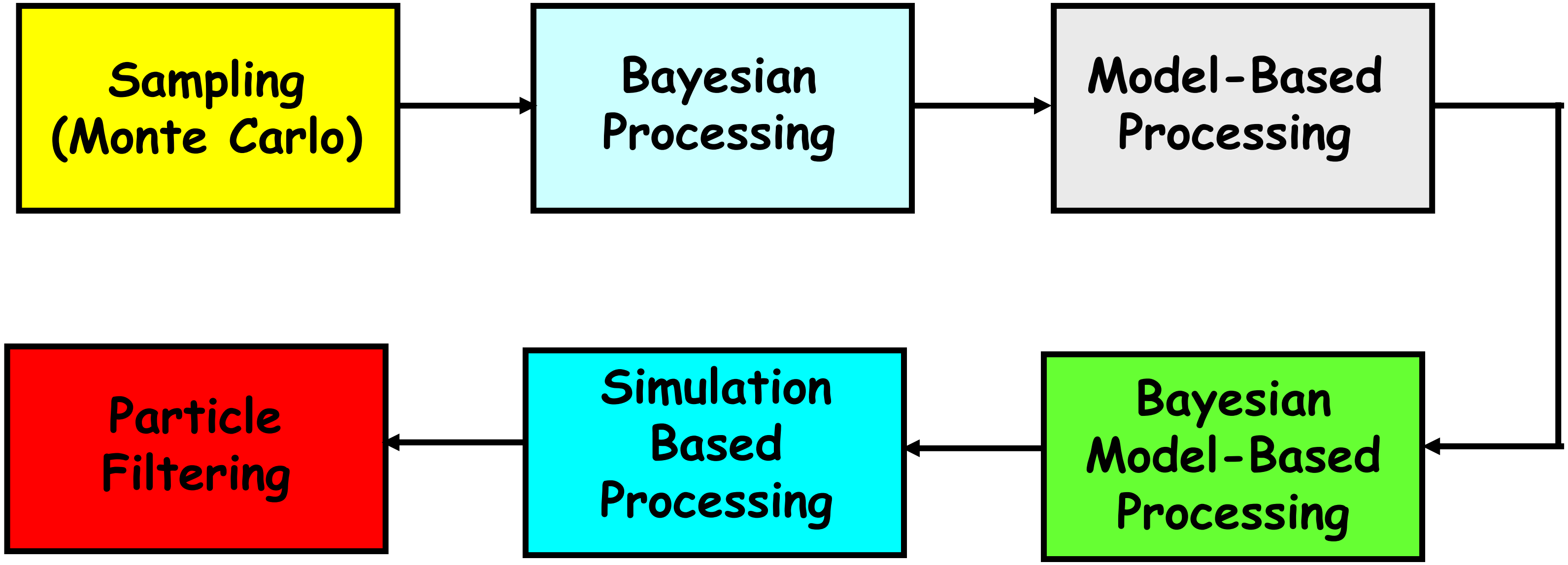
- Ability to represent arbitrary densities
- Adaptive focusing on probable regions of state-space
- Dealing with non-Gaussian noise
- The framework allows for including multiple models (tracking maneuvering targets)

DISADVANTAGES

- High computational complexity
- It is difficult to determine optimal number of particles
- Number of particles increase with increasing model dimension
- Potential problems: degeneracy and loss of diversity
- The choice of importance density is crucial

* M. Bolic, University of Ottawa, “Theory and Implementation of Particle Filters”

Our plan is to develop particle filters using the following roadmap:



Sampling (Monte Carlo)

Bayesian Processing

Model-Based Processing

Bayesian Model-Based Processing

Simulation Based Processing

Particle Filtering

BACKGROUND

Monte Carlo (MC) approach to nonparametric "posterior" probability distribution estimation

- MC techniques obtain random sample based representations of probability distributions to solve a variety of estimation problems, especially those in which a "closed-form" analytic expression is not available, that is, usual is distribution → samples, MC samples → distribution

- The MC approach enables complex, multi-modal distributions to be estimated

- Many nonlinear, non-gaussian processing problems can easily be cast into this framework

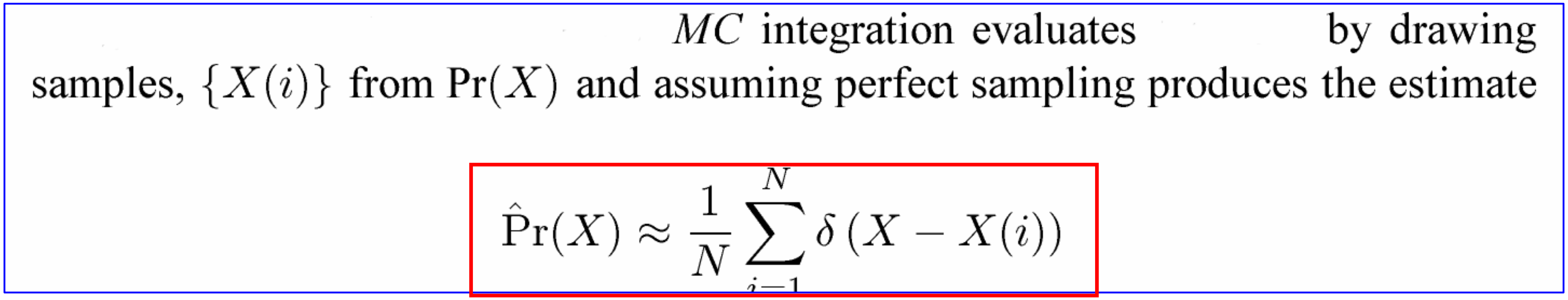
- When statistics are "nonstationary," then sequential MC methods must be employed to solve the problem

- Particle filters are a "sequential" MC method that can be applied in the nonstationary and/or real-time cases as well as the stationary case

MONTE CARLO (Sampling) METHODS

In most cases (non-gaussian), the posterior distribution is not available requiring that we use "simulation-based" or non-parametric methods for estimation and inference

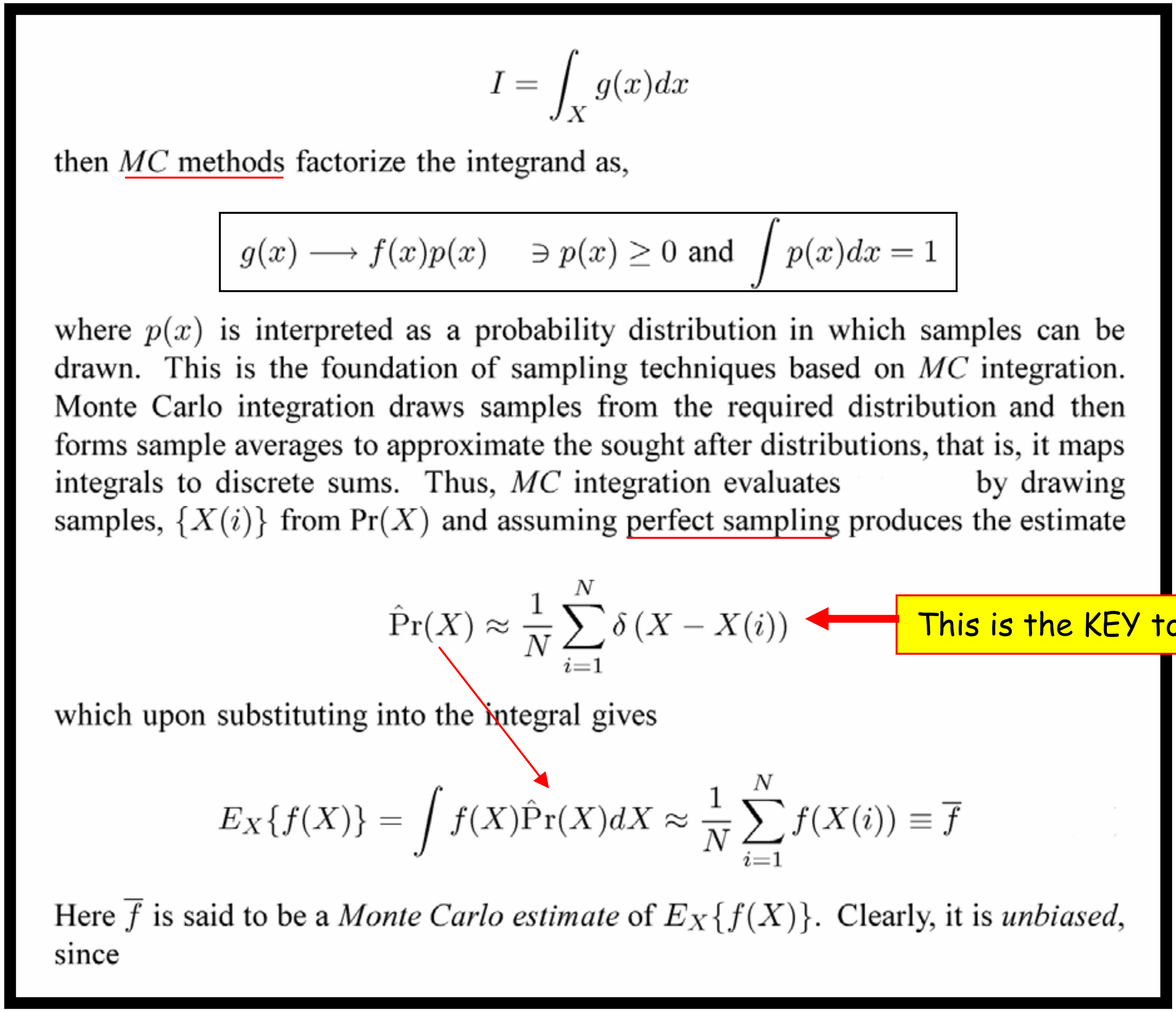
- These techniques are called SAMPLING METHODS
- They replace numerical integration techniques
- Monte Carlo (MC) sampling methods are an alternative
- The KEY idea in MC is replacing the distribution function by its SAMPLES reducing integration to summations
- As the number of samples increase the equivalence becomes established with the targeted posterior distribution enabling its moments or estimators to be calculated, that is,



MC integration evaluates by drawing samples, $\{X(i)\}$ from $\Pr(X)$ and assuming perfect sampling produces the estimate

$$\boxed{\hat{\Pr}(X) \approx \frac{1}{N}\sum_{i=1}^{N} \delta\left(X - X(i)\right)}$$

Typically integration is used to estimate expectations, but in MC the "inverse" is true



$$I = \int_X g(x)dx$$

then MC methods factorize the integrand as,

$$\boxed{g(x) \longrightarrow f(x)p(x) \quad \ni p(x) \geq 0 \text{ and } \int p(x)dx = 1}$$

where $p(x)$ is interpreted as a probability distribution in which samples can be drawn. This is the foundation of sampling techniques based on MC integration. Monte Carlo integration draws samples from the required distribution and then forms sample averages to approximate the sought after distributions, that is, it maps integrals to discrete sums. Thus, MC integration evaluates by drawing samples, $\{X(i)\}$ from $\Pr(X)$ and assuming perfect sampling produces the estimate

$$\hat{\Pr}(X) \approx \frac{1}{N}\sum_{i=1}^{N} \delta\left(X - X(i)\right)$$

which upon substituting into the integral gives

$$E_X\{f(X)\} = \int f(X)\hat{\Pr}(X)dX \approx \frac{1}{N}\sum_{i=1}^{N} f(X(i)) \equiv \overline{f}$$

Here $\overline{f}$ is said to be a *Monte Carlo estimate* of $E_X\{f(X)\}$. Clearly, it is *unbiased*, since

This is the KEY to MC techniques

Consider a gaussian example: $x \sim N\left(m_x, \sigma_x^2\right)$

the Monte Carlo approach is to generate N samples from a gaussian, that is, assuming perfect (uniform) sampling, we have that

$$\hat{\Pr}(x) \approx 1/N \sum_{i=1}^{N} \delta(x - x_i)$$

and therefore the mean and variance can be estimate from the samples directly using

$$\boxed{\hat{m}_x = \int x\hat{\Pr}(x)\ dx = \int x \left(1/N \sum_{i=1}^{N} \delta(x - x_i) \right) dx = 1/N \sum_{i=1}^{N} x_i}$$

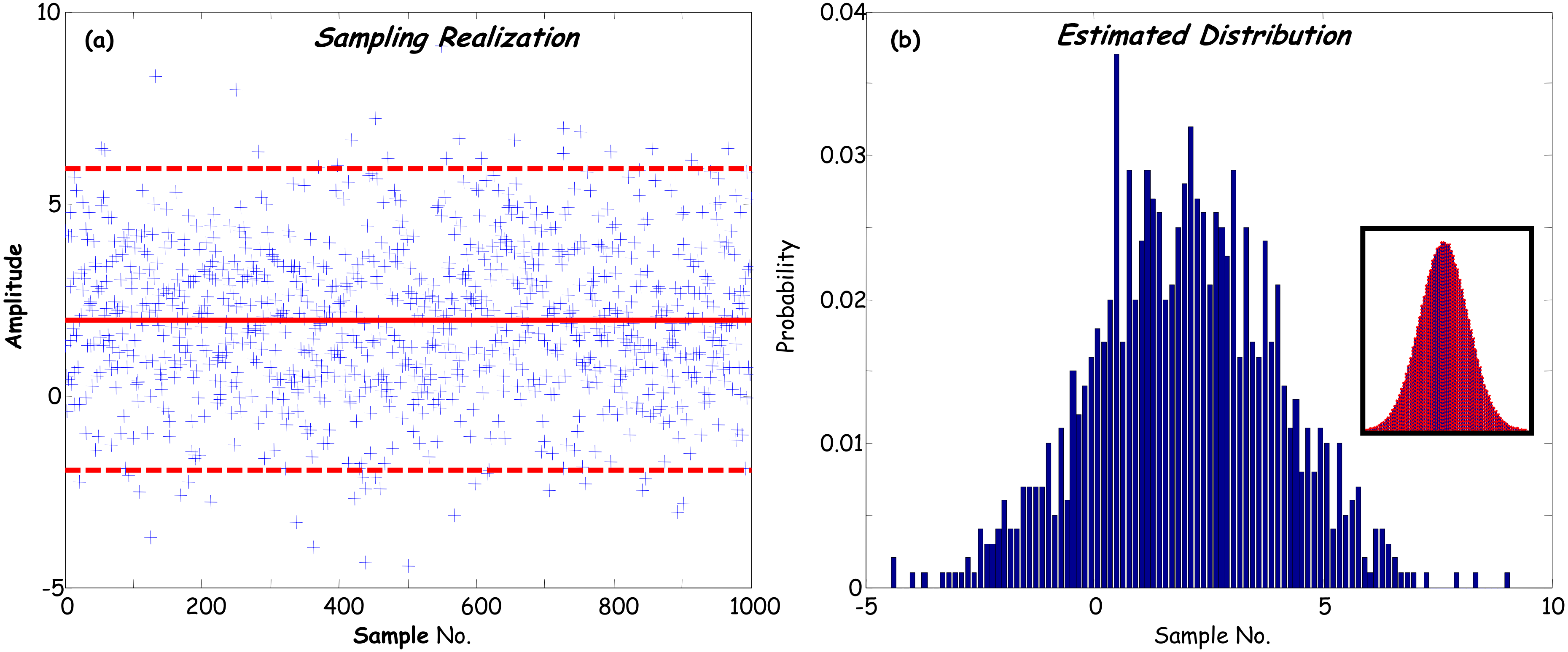
which follows directly from the sifting property of the delta function. Also, we have

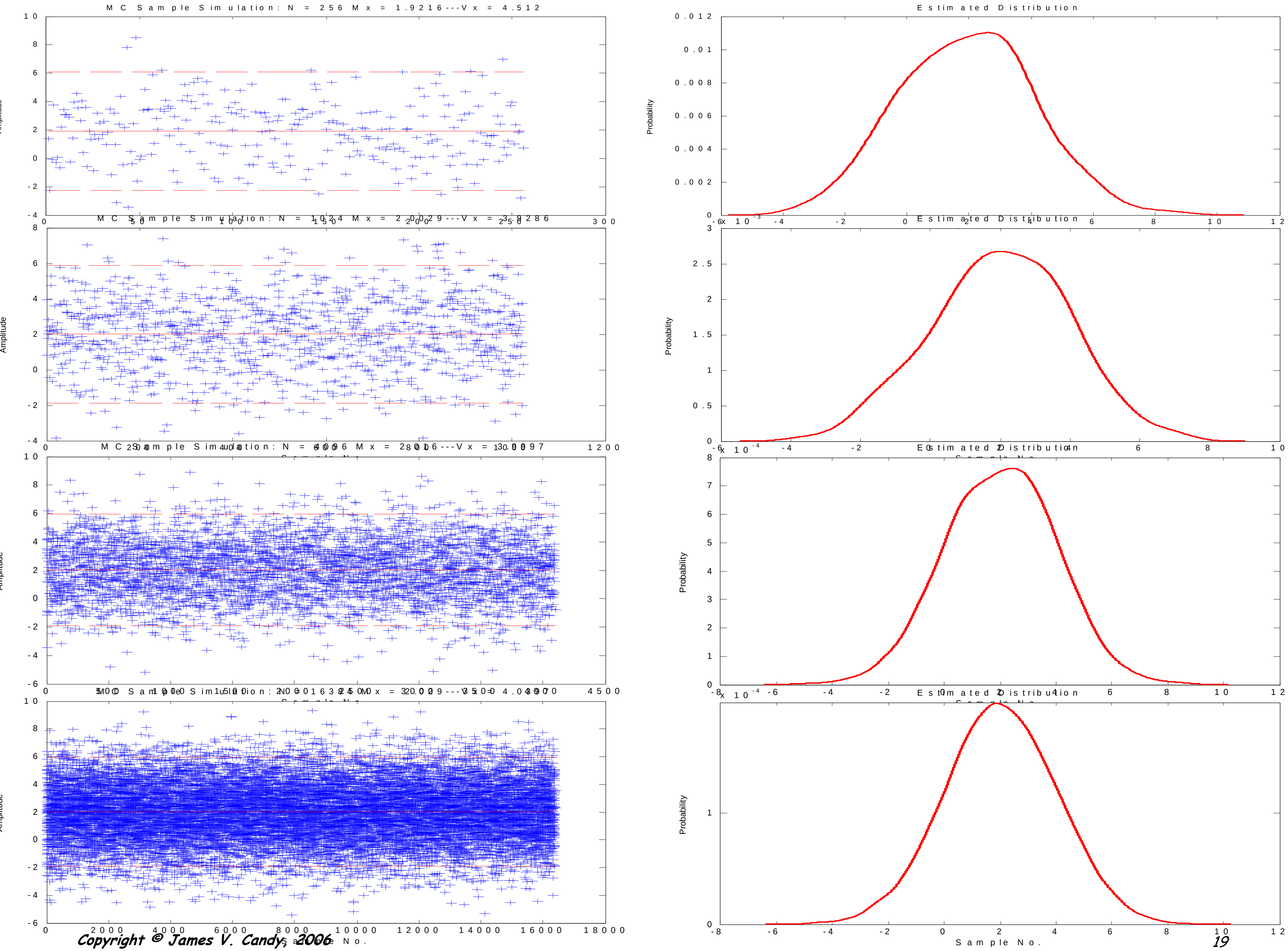
$$\boxed{\hat{\sigma}_x^2 = \int (x - m_x)^2\ \text{Pr}\hat{(}x)\ dx = \int (x - m_x)^2 \left(1/N \sum_{i=1}^{N} \delta(x - x_i) \right) dx = 1/N \sum_{i=1}^{N} (x_i - \hat{m}_x)^2}$$

Thus, summarizing the *MC* approach is:

- Generate N samples from a gaussian: $x_i \sim \mathcal{N}(m_x, \sigma_x^2)$;
- Estimate the desired statistics of the distribution from its samples as: $\hat{m}_x$ and $\hat{\sigma}_x^2$

Monte Carlo methods rely on samples generated from the sampling distribution to estimate statistics (mean, variance, etc.) as demonstrated in this Gaussian example





Copyright © James V. Candy, 2006

SEQUENTIAL BAYESIAN PROCESSOR (SBP)

Bayes' rule for dynamic variables is:

$$\Pr(X|Y) = \Pr(Y|X)\frac{\Pr(X)}{\Pr(Y)}$$

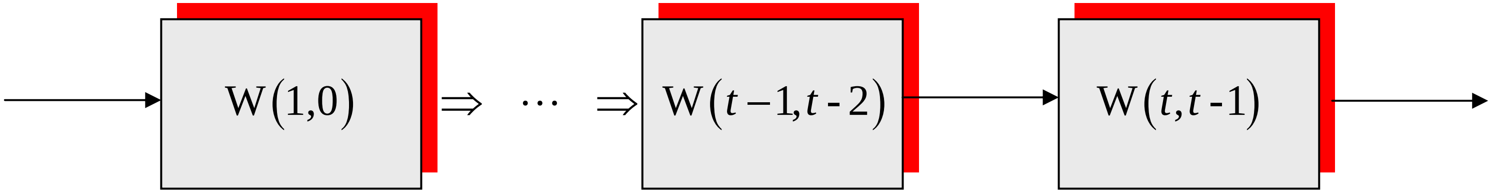
$$\Updownarrow$$

$$\Pr(X_t|Y_t) = \frac{\Pr(Y_t|X_t) \times \Pr(X_t)}{\Pr(Y_t)}$$

$$\left[X_t := \{x(0), \cdots, x(t)\}; \quad Y_t := \{y(0), \cdots, y(t)\}\right]$$

The posterior distribution can be estimated using the sequential Bayesian processor (SBP):

$$\Pr\left[\mathbf{X}_o|\mathbf{Y}_o\right] \quad\Rightarrow\quad \Pr\left[\mathbf{X}_1|\mathbf{Y}_1\right] \Rightarrow \cdots \Rightarrow \Pr\left[\mathbf{X}_{t\text{-}1}|\mathbf{Y}_{t\text{-}1}\right] \quad\Rightarrow\quad \Pr\left[\mathbf{X}_t|\mathbf{Y}_t\right]$$



$\mathrm{W}(1,0)$
$\Rightarrow \cdots \Rightarrow$
$\mathrm{W}(t-1,t\text{-}2)$
$\mathrm{W}(t,t\text{-}1)$

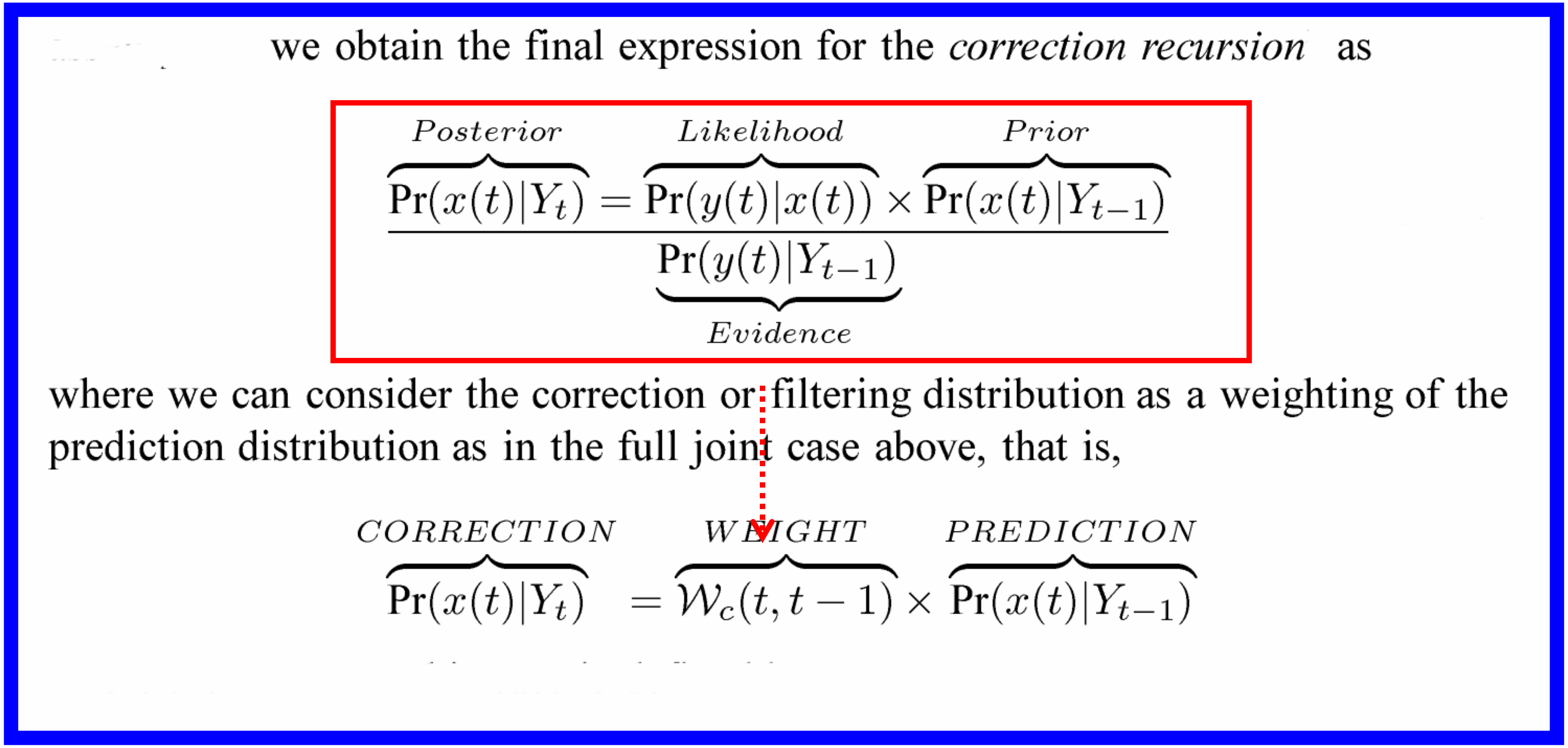
$$\Pr\left[\mathbf{X}_t|\mathbf{Y}_t\right] = \mathrm{W}(t,t\text{-}1)\times\Pr\left[\mathbf{X}_{t-1}|\mathbf{Y}_{t-1}\right]$$

where the Bayes' operator is defined at each stage by

$$\boxed{\mathrm{W}(t,t\text{-}1) := \frac{\Pr\left[\mathbf{y}(t)|\mathbf{x}(t)\right]\times\Pr\left[\mathbf{x}(t)|\mathbf{x}(t-1)\right]}{\Pr\left[\mathbf{y}(t)|\mathbf{Y}_{t-1}\right]};\quad t=1,\cdots,N}$$

Copyright © James V. Candy, 2006

The final FILTERING recursion evolves from Bayes' rule



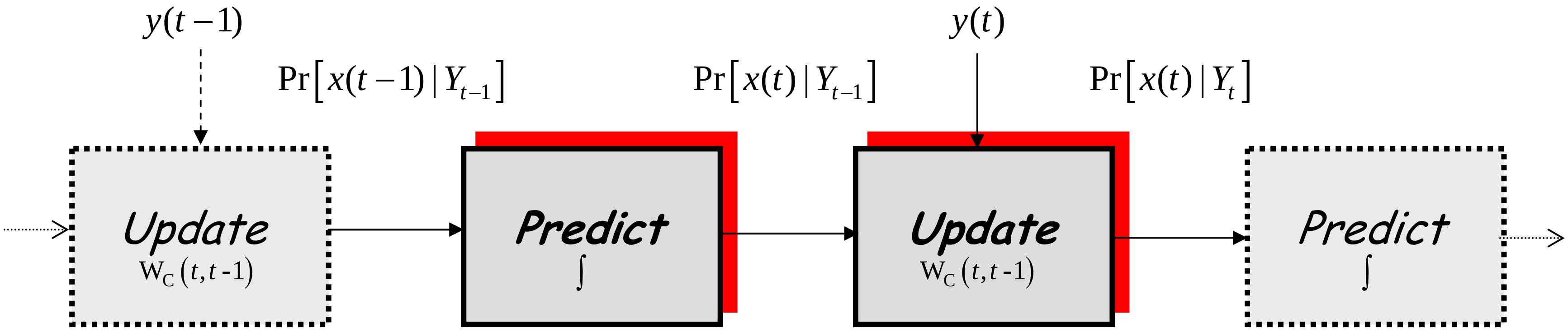
we obtain the final expression for the *correction recursion* as

$$\boxed{\overbrace{\Pr(x(t)|Y_t)}^{Posterior} = \frac{\overbrace{\Pr(y(t)|x(t))}^{Likelihood} \times \overbrace{\Pr(x(t)|Y_{t-1})}^{Prior}}{\underbrace{\Pr(y(t)|Y_{t-1})}_{Evidence}}}$$

where we can consider the correction or filtering distribution as a weighting of the prediction distribution as in the full joint case above, that is,

$$\overbrace{\Pr(x(t)|Y_t)}^{CORRECTION} = \overbrace{\mathcal{W}_c(t,t-1)}^{WEIGHT} \times \overbrace{\Pr(x(t)|Y_{t-1})}^{PREDICTION}$$

The SBP for the "filtering" posterior is shown by:



$y(t-1)$
$\Pr\left[x(t-1) \mid Y_{t-1}\right]$
$\Pr\left[x(t) \mid Y_{t-1}\right]$
$y(t)$
$\Pr\left[x(t) \mid Y_{t}\right]$
Update
$\mathrm{W_C}(t,t\text{-}1)$
Predict
$\int$
Update
$\mathrm{W_C}(t,t\text{-}1)$
Predict
$\int$

Predict: $\Pr\left[x(t) \mid Y_{t-1}\right] = \int \Pr\left[x(t) \mid x(t-1)\right] \Pr\left[x(t-1) \mid Y_{t-1}\right] dx(t-1)$

Update: $\Pr\left[x(t) \mid Y_{t}\right] = \mathrm{W_c}(t,t-1) \times \Pr\left[x(t) \mid Y_{t-1}\right]$

$$\boxed{\mathrm{W_c}(t,t-1) := \left[\frac{\Pr\left[y(t) \mid x(t)\right]}{\Pr\left[y(t) \mid Y_{t-1}\right]}\right]}$$

MODEL-BASED (Bayesian) SIGNAL PROCESSING

Model-based signal processing:

- Signal processing is the set of techniques to "extract the useful information from noisy measurements while rejecting the extraneous"

- If the SNR is high then simple non-physical techniques (FFTs, wavelets, spectral estimation, etc.) can be used to extract the desired information, BUT . . . .

- If the SNR is low and the problem uncertain (noise, random parameters, etc.), then more of the underlying physical phenomenology must somehow be incorporated into the processor

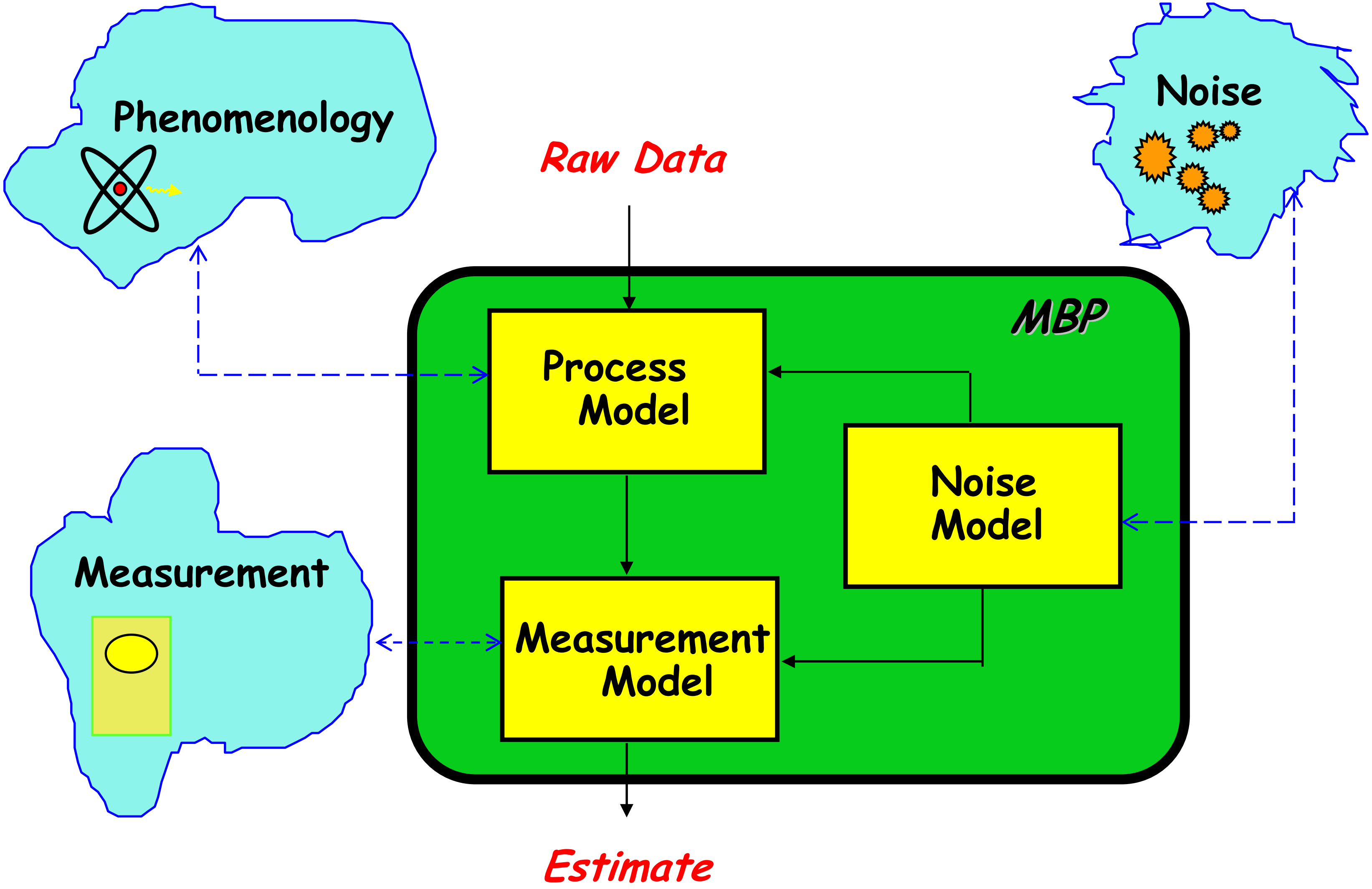
THEREFORE

- Model-based signal processing incorporates physical phenomenological, measurement, and noise/uncertainty models into the processor to extract the desired information while rejecting the extraneous --- even in highly uncertain environments

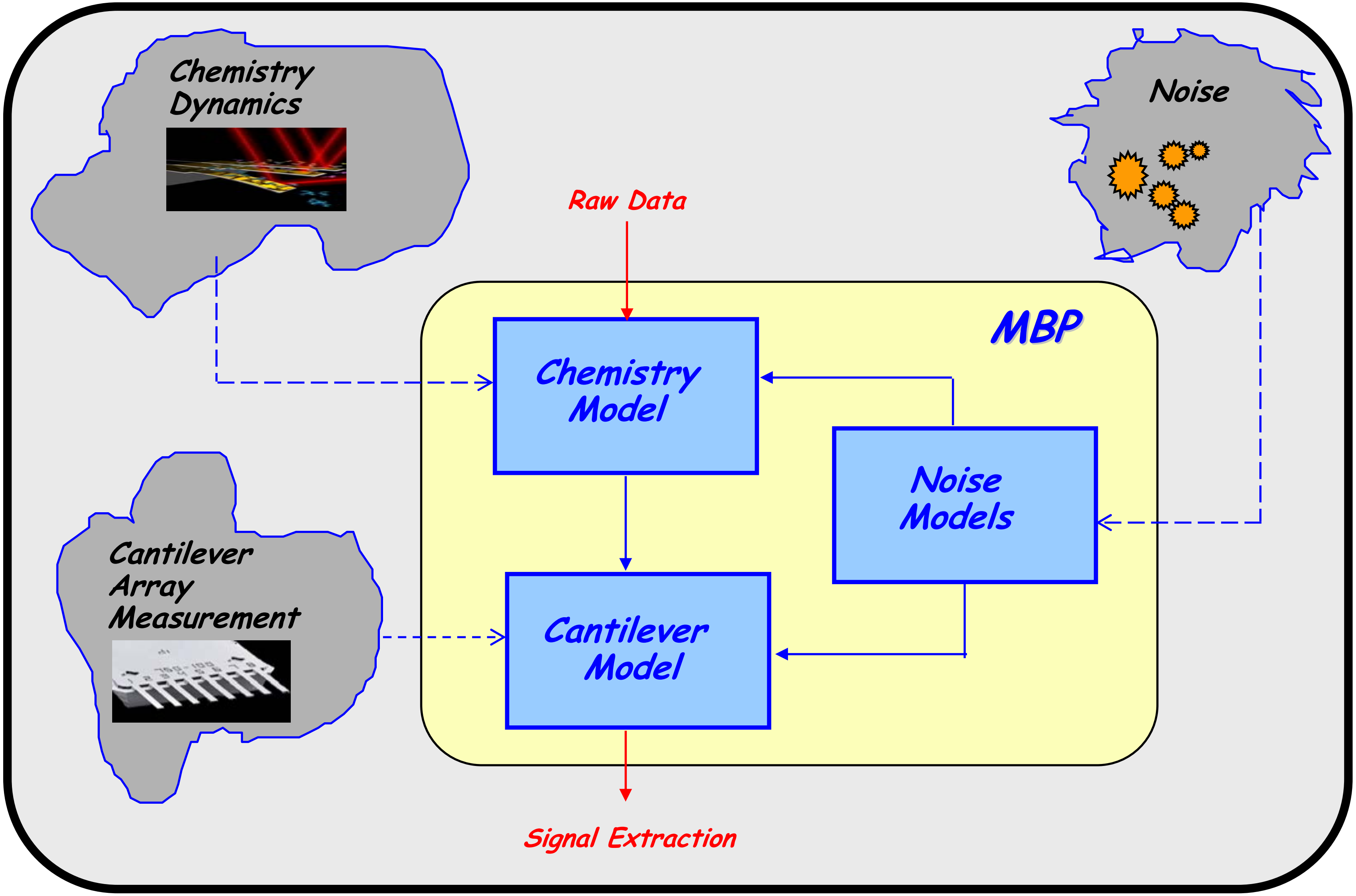
The model-based approach is simply:

"incorporating mathematical models of both physical phenomenology and the measurement processes including noise into the processor to extract the desired information"

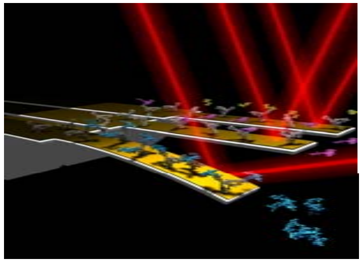
MBP are techniques that incorporate any "a priori" knowledge of the underlying phenomenology into a processing scheme



In bio-threats, MBP techniques can incorporate any "a priori" knowledge of the underlying physics into the processing scheme



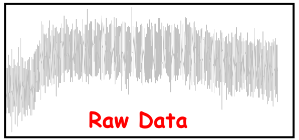
For this problem, *smart bio-sensors* incorporating a micro-cantilever array can be developed using the MBP approach



$$\Gamma(t)=\begin{cases}0, & t<t_{\mathrm{ON}}\\ \left(\dfrac{c(t)}{c(t)+k_a/k_d}\right)\{1-\exp[-(k_a c(t)+k_d)(t-t_{ON})]\}, & t_{\mathrm{ON}}\le t\le t_{\mathrm{OFF}}\\ \sqrt{\dfrac{1}{2k_d(t-t_{OFF})}}, & t>t_{\mathrm{OFF}}\end{cases}$$

Concentration

CHEMISTRY/PHYSICS

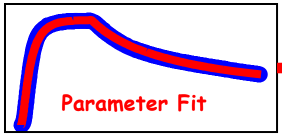


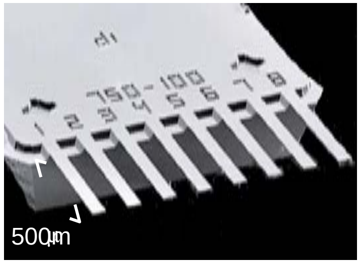
MODEL-BASED PROCESSOR

$$\Delta\hat{G}(t\,|\,t-1)=\ \Delta\hat{G}(t-1\,|\,t-1)\ \text{[Free Energy]}$$

$$\hat{\mathbf{y}}_\ell(t\,|\,t-1)\ =\beta_\ell\Gamma(t;\hat{\Theta})\Delta\hat{G}(t\,|\,t-1)+\Delta\hat{z}^T(t)\quad\text{[Deflection]}$$

$\hat{\Theta}$



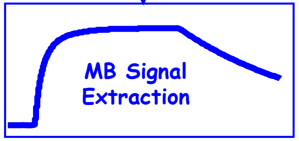


$$y_\ell(t)\ =\beta_\ell\Gamma(t;\Theta)\Delta G(t)+\Delta z^T(t)+v_\ell(t)$$

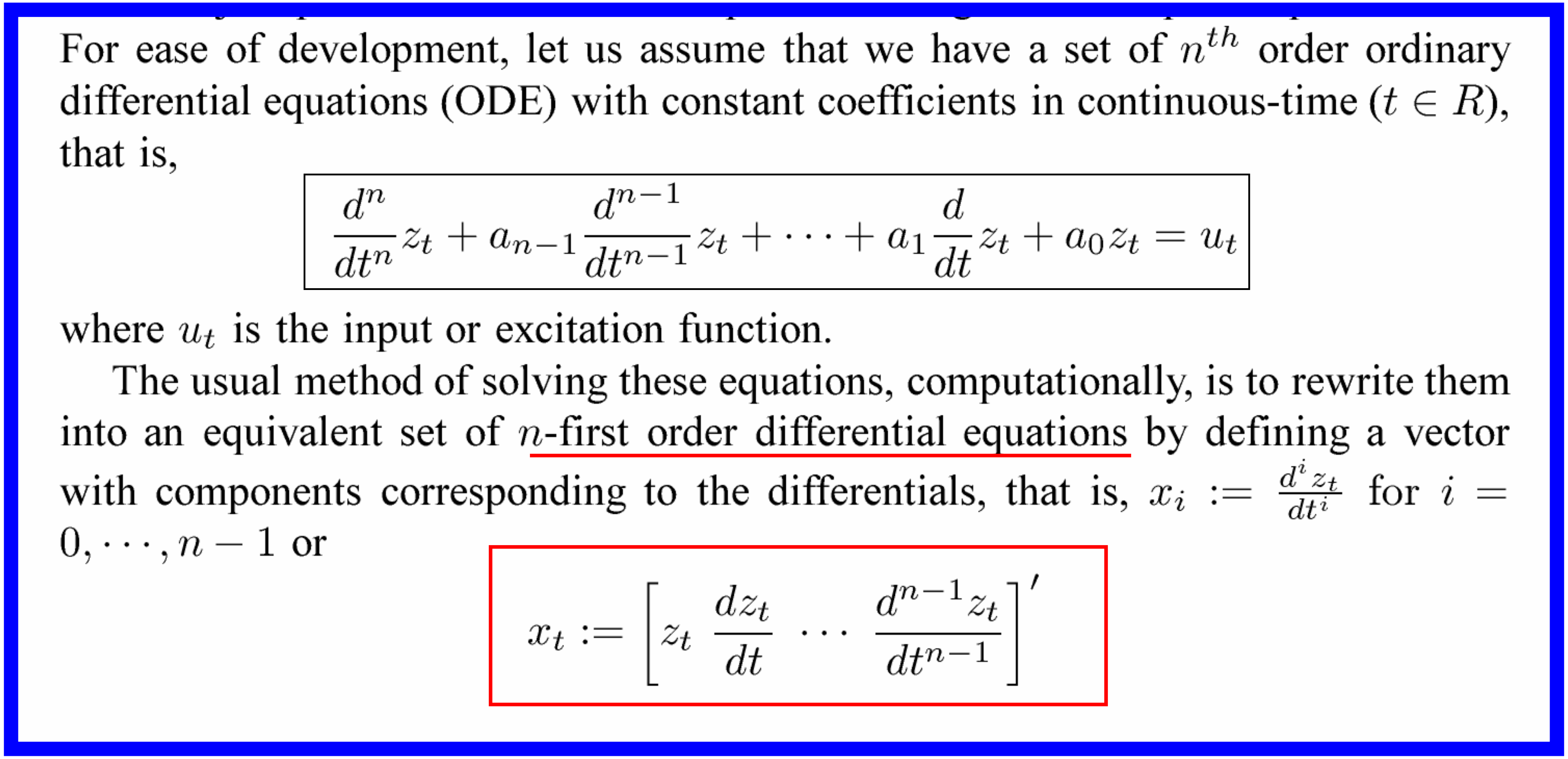
$$\text{for}\quad \ell=1,\cdots,L$$

Noise

MEASUREMENTS



State-space is a natural (mathematical) representation of a system (of equations)



For ease of development, let us assume that we have a set of n^{th} order ordinary differential equations (ODE) with constant coefficients in continuous-time ($t \in R$), that is,

$$\boxed{\frac{d^n}{dt^n} z_t + a_{n-1} \frac{d^{n-1}}{dt^{n-1}} z_t + \cdots + a_1 \frac{d}{dt} z_t + a_0 z_t = u_t}$$

where u_t is the input or excitation function.

The usual method of solving these equations, computationally, is to rewrite them into an equivalent set of n-first order differential equations by defining a vector with components corresponding to the differentials, that is, $x_i := \frac{d^i z_t}{dt^i}$ for $i = 0, \cdots, n-1$ or

$$\boxed{x_t := \left[z_t \quad \frac{dz_t}{dt} \quad \cdots \quad \frac{d^{n-1} z_t}{dt^{n-1}} \right]'}$$

State-space is a natural representation:

Taking the derivative of x_t for each component, we obtain

$$
\begin{aligned}
\dot{x}_1 &= \frac{dz_t}{dt} = x_2 \\
\dot{x}_2 &= \frac{d^2 z_t}{dt^2} = x_3 \\
&\vdots \\
\dot{x}_n &= \frac{d^n z_t}{dt^n} = -a_{n-1}\frac{d^{n-1} z_t}{dt^{n-1}} + \cdots - a_1 \frac{dz_t}{dt} - a_0 z_t + u_t
\end{aligned}
$$

Collecting all of these equations and writing them into vector-matrix form, we obtain

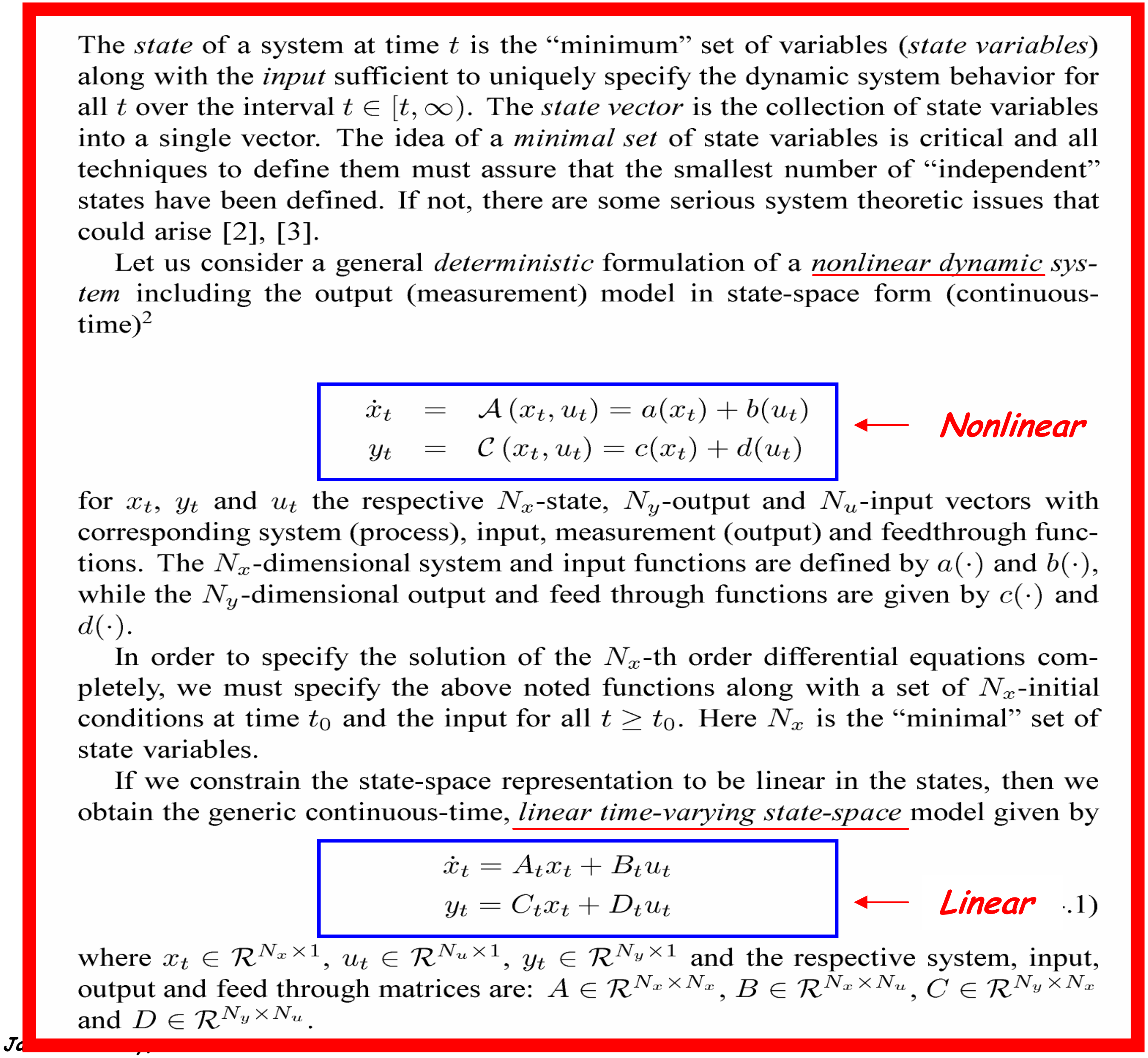
$$
\frac{d}{dt}\begin{bmatrix} x_1 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & \cdots & -a_{n-1} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u_t
$$

or in compact form

$$\boxed{\dot{x}_t = A x_t + B u_t}$$

for $x_t \in R^n$, $u_t \in R^{1\times 1}$, $A \in R^{n\times n}$ and $B \in R^{n\times 1}$. Here x_t is defined as the *state vector*, u_t the input, A is the system or process matrix and B is the input matrix. So we see from this simplistic view, the state vector is "naturally" obtained when we convert an n^{th} order ODE to a set of n-first order equations.

STATE-SPACE MODEL



The *state* of a system at time t is the "minimum" set of variables (*state variables*) along with the *input* sufficient to uniquely specify the dynamic system behavior for all t over the interval $t \in [t, \infty)$. The *state vector* is the collection of state variables into a single vector. The idea of a *minimal set* of state variables is critical and all techniques to define them must assure that the smallest number of "independent" states have been defined. If not, there are some serious system theoretic issues that could arise [2], [3].

Let us consider a general *deterministic* formulation of a *nonlinear dynamic system* including the output (measurement) model in state-space form (continuous-time)[2]

$$\boxed{\begin{aligned} \dot{x}_t &= \mathcal{A}\,(x_t, u_t) = a(x_t) + b(u_t) \\ y_t &= \mathcal{C}\,(x_t, u_t) = c(x_t) + d(u_t) \end{aligned}} \longleftarrow \textit{Nonlinear}$$

for x_t, y_t and u_t the respective N_x-state, N_y-output and N_u-input vectors with corresponding system (process), input, measurement (output) and feedthrough functions. The N_x-dimensional system and input functions are defined by $a(\cdot)$ and $b(\cdot)$, while the N_y-dimensional output and feed through functions are given by $c(\cdot)$ and $d(\cdot)$.

In order to specify the solution of the N_x-th order differential equations completely, we must specify the above noted functions along with a set of N_x-initial conditions at time t_0 and the input for all $t \geq t_0$. Here N_x is the "minimal" set of state variables.

If we constrain the state-space representation to be linear in the states, then we obtain the generic continuous-time, *linear time-varying state-space* model given by

$$\boxed{\begin{aligned} \dot{x}_t &= A_t x_t + B_t u_t \\ y_t &= C_t x_t + D_t u_t \end{aligned}} \longleftarrow \textit{Linear} \quad .1)$$

where $x_t \in \mathcal{R}^{N_x \times 1}$, $u_t \in \mathcal{R}^{N_u \times 1}$, $y_t \in \mathcal{R}^{N_y \times 1}$ and the respective system, input, output and feed through matrices are: $A \in \mathcal{R}^{N_x \times N_x}$, $B \in \mathcal{R}^{N_x \times N_u}$, $C \in \mathcal{R}^{N_y \times N_x}$ and $D \in \mathcal{R}^{N_y \times N_u}$.

BAYESIAN APPROACH TO STATE-SPACE PROCESSORS

Bayesian approach to the state-space: definitions

$$\boxed{\begin{aligned} x(t) &= \mathcal{A}\left(x(t-1), u(t-1), w(t-1)\right) \\ y(t) &= \mathcal{C}\left(x(t), u(t), v(t)\right) \end{aligned}}$$

where w and v are the respective process and measurement noise sources with u a known input. Here $\mathcal{A}(\cdot)$ is the nonlinear (or linear) dynamic state transition function and $\mathcal{C}(\cdot)$ the corresponding measurement function. Both conditional probabilistic distributions embedded within the Bayesian framework are *completely* specified by these functions and the underlying noise distributions: $\Pr(w(t-1))$ and $\Pr(v(t))$. That is, we have the equivalence

$$\boxed{\begin{aligned} \mathcal{A}\left(x(t-1), u(t-1), w(t-1)\right) &\Rightarrow \Pr\left(x(t)|x(t-1)\right) \Leftrightarrow \mathcal{A}\left(x(t)|x(t-1)\right) \\ \mathcal{C}\left(x(t), u(t), v(t)\right) &\Rightarrow \Pr\left(y(t)|x(t)\right) \qquad \Leftrightarrow \mathcal{C}\left(y(t)|x(t)\right) \end{aligned}}$$

Bayesian approach to the state-space: posteriors

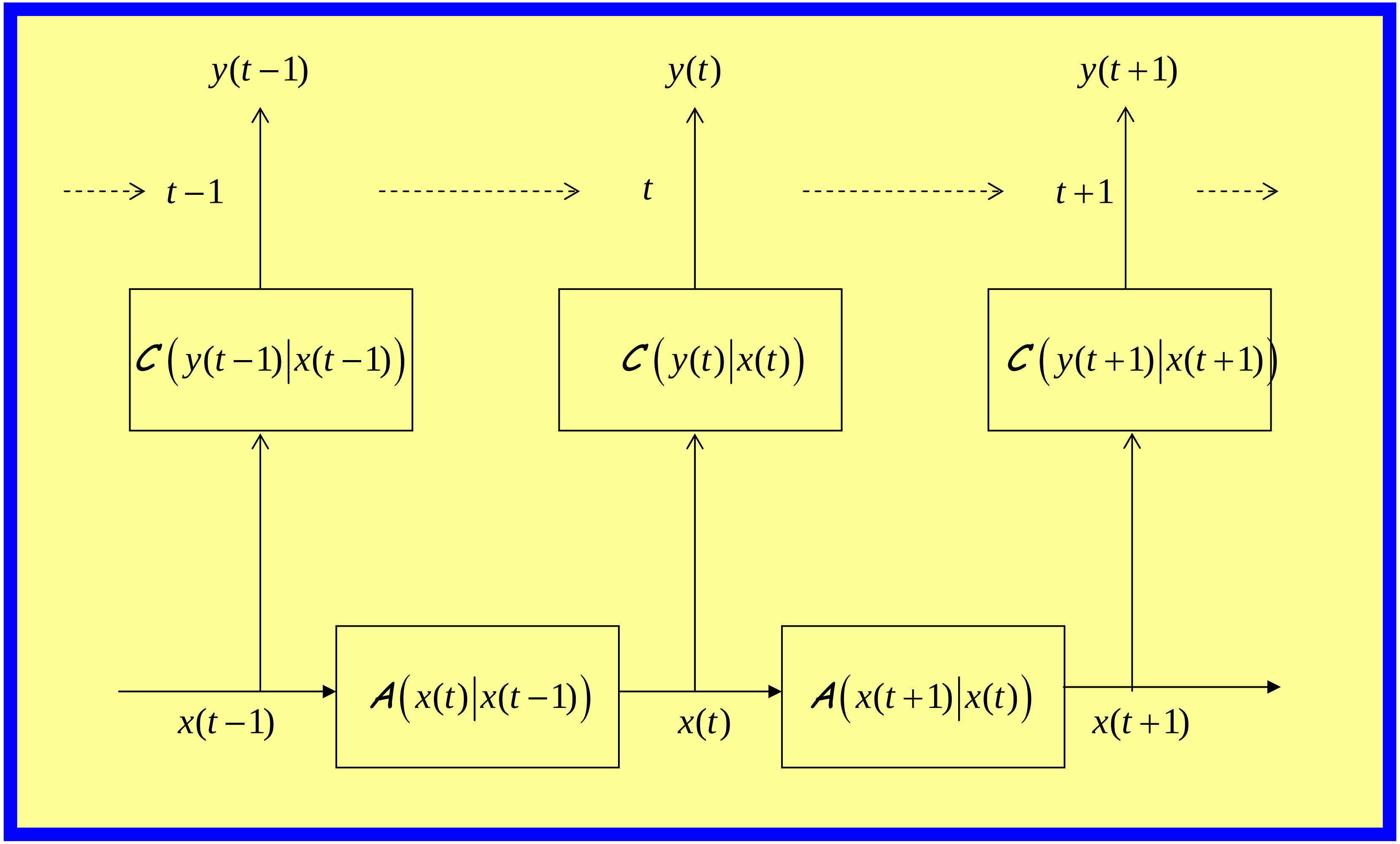
the prediction recursion characterized by the *Chapman-Kolmogorov equation* replacing transition probability with the implied model-based conditional, that is,

$$\boxed{\Pr(x(t)|Y_{t-1}) = \int \overbrace{\mathcal{A}(x(t)|x(t-1))}^{\text{Embedded Process Model}} \times \Pr(x(t-1)|Y_{t-1})dx(t-1)}$$

Next we incorporate the model-based likelihood into the posterior equation with the understanding that the process model has been incorporated into the prediction

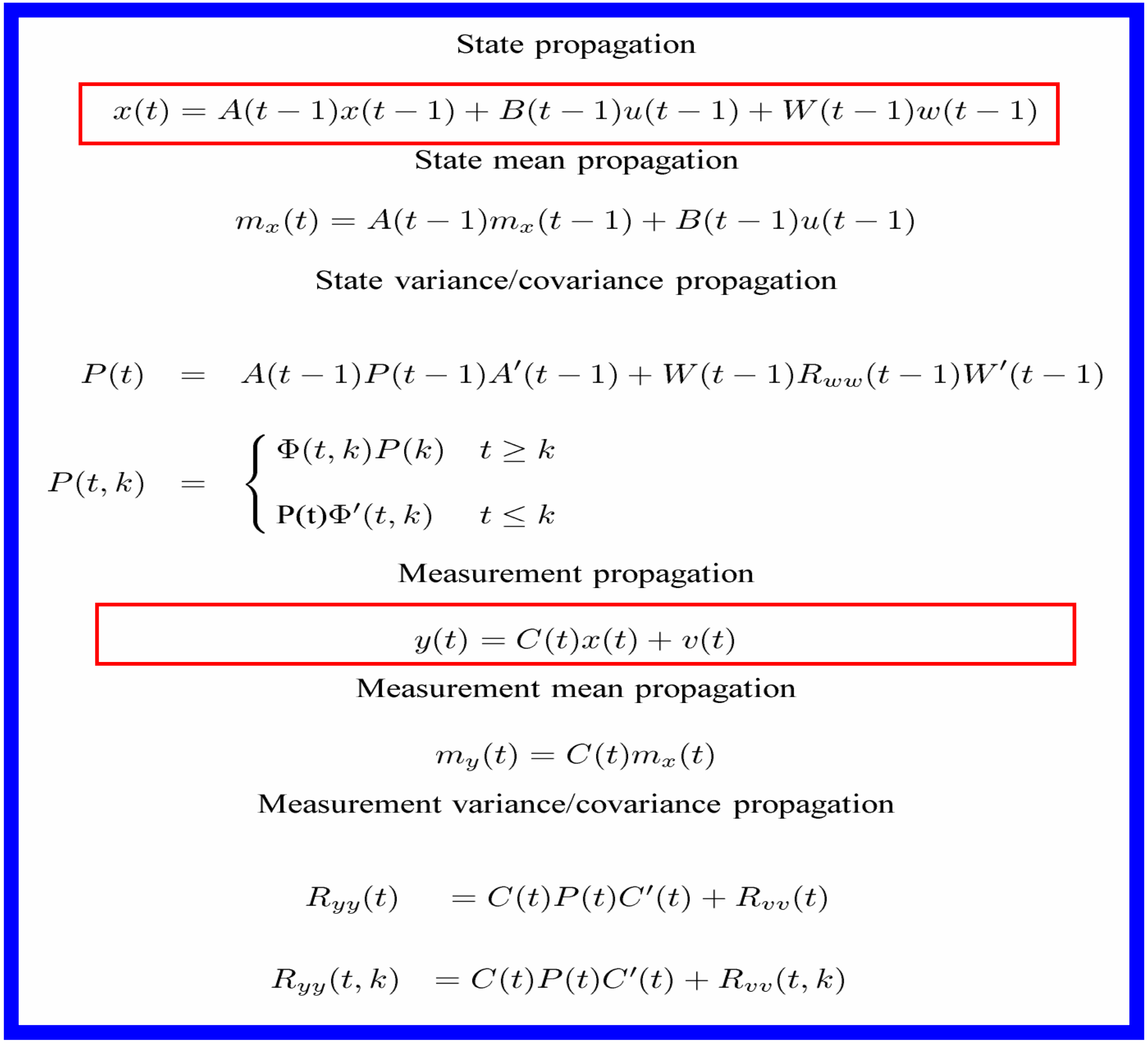
$$\boxed{\Pr(x(t)|Y_t) = \overbrace{\mathcal{C}(y(t)|x(t))}^{\text{Embedded Measurement Model}} \times \Pr(x(t)|Y_{t-1}) \Big/ \Pr(y(t)|Y_{t-1})}$$

The BSP implemented in state-space is:



$y(t-1)$
$y(t)$
$y(t+1)$
$t-1$
t
$t+1$
$\mathcal{C}\left(y(t-1)|x(t-1)\right)$
$\mathcal{C}\left(y(t)|x(t)\right)$
$\mathcal{C}\left(y(t+1)|x(t+1)\right)$
$\mathcal{A}\left(x(t)|x(t-1)\right)$
$\mathcal{A}\left(x(t+1)|x(t)\right)$
$x(t-1)$
$x(t)$
$x(t+1)$

EXAMPLE: Given a Gauss-Markov model in state-space form:



State propagation

$$\boxed{x(t) = A(t-1)x(t-1) + B(t-1)u(t-1) + W(t-1)w(t-1)}$$

State mean propagation

$$m_x(t) = A(t-1)m_x(t-1) + B(t-1)u(t-1)$$

State variance/covariance propagation

$$P(t) = A(t-1)P(t-1)A'(t-1) + W(t-1)R_{ww}(t-1)W'(t-1)$$

$$P(t,k) = \begin{cases} \Phi(t,k)P(k) & t \geq k \\ \mathrm{P(t)}\Phi'(t,k) & t \leq k \end{cases}$$

Measurement propagation

$$\boxed{y(t) = C(t)x(t) + v(t)}$$

Measurement mean propagation

$$m_y(t) = C(t)m_x(t)$$

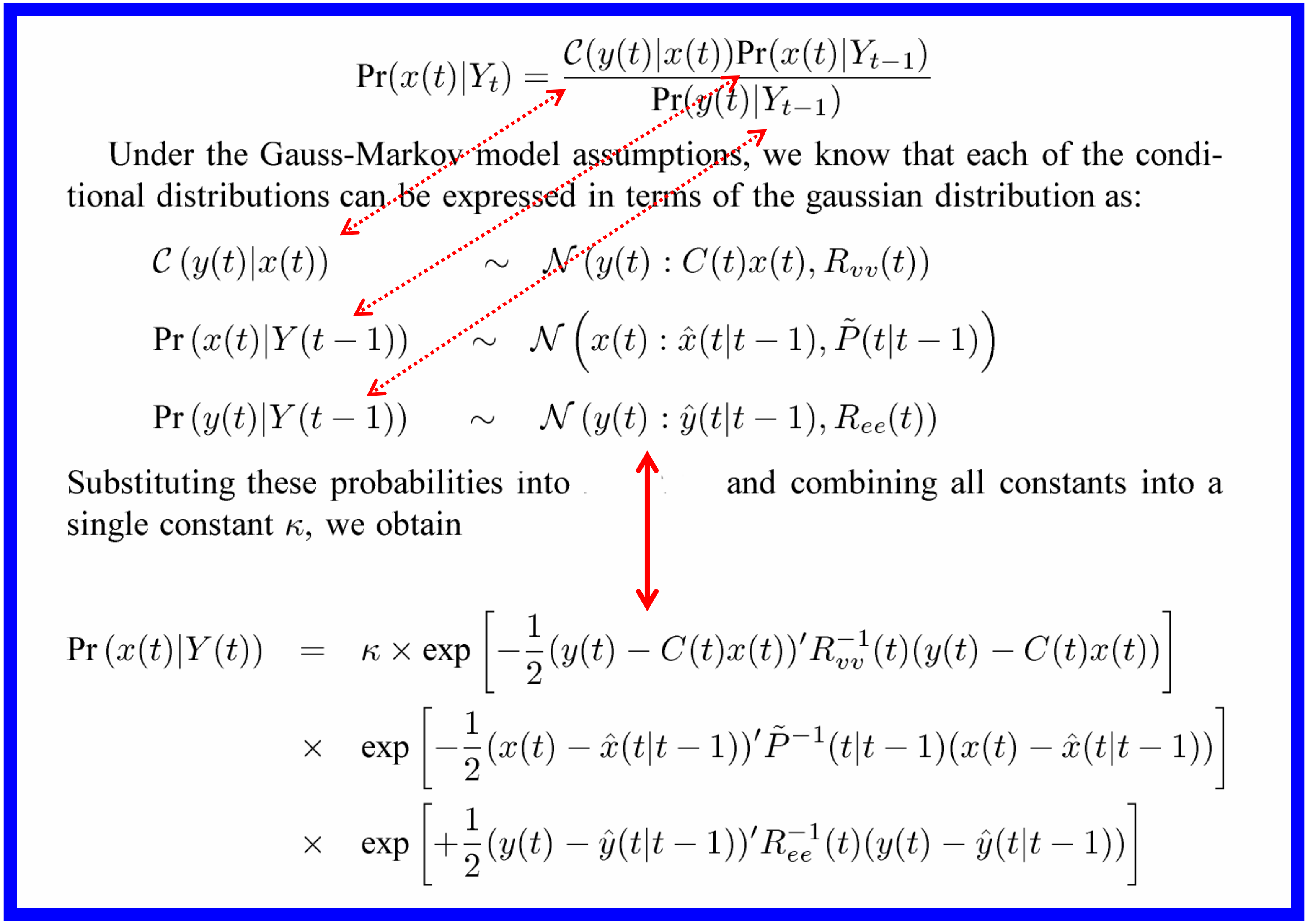
Measurement variance/covariance propagation

$$R_{yy}(t) = C(t)P(t)C'(t) + R_{vv}(t)$$

$$R_{yy}(t,k) = C(t)P(t)C'(t) + R_{vv}(t,k)$$

GM EXAMPLE: Develop a SBP where:

$$\hat{x}(t\,|\,t-1) := E\left\{x(t)\,|\,Y_{t-1}\right\};\tilde{P}(t\,|\,t-1) := \operatorname{cov}(x(t)-\hat{x}(t\,|\,t-1))$$



$$\Pr(x(t)|Y_t) = \frac{\mathcal{C}(y(t)|x(t))\Pr(x(t)|Y_{t-1})}{\Pr(y(t)|Y_{t-1})}$$

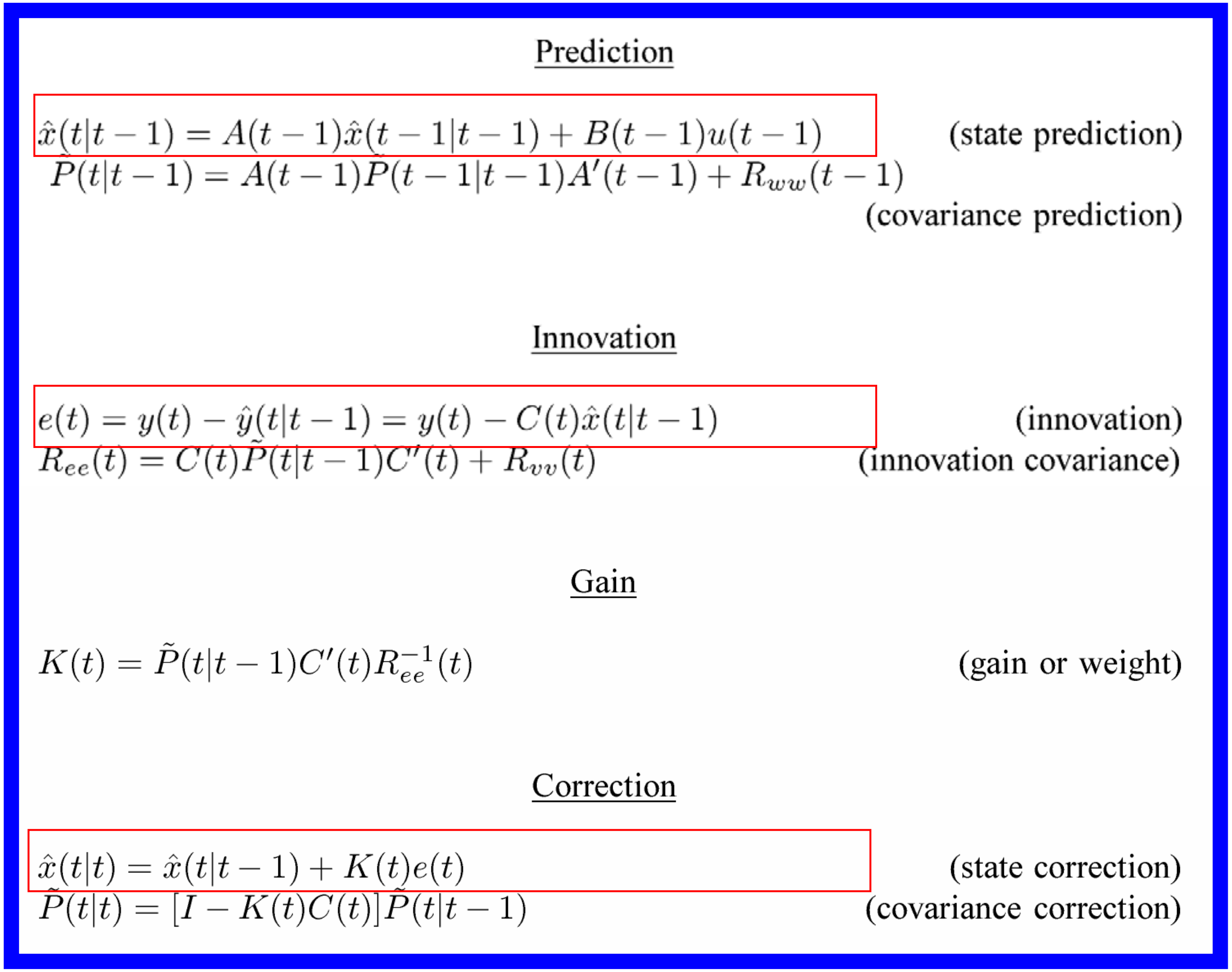
Under the Gauss-Markov model assumptions, we know that each of the conditional distributions can be expressed in terms of the gaussian distribution as:

$$\begin{aligned}
\mathcal{C}\left(y(t)|x(t)\right) &\sim \mathcal{N}\left(y(t) : C(t)x(t), R_{vv}(t)\right)\\
\Pr\left(x(t)|Y(t-1)\right) &\sim \mathcal{N}\left(x(t) : \hat{x}(t|t-1), \tilde{P}(t|t-1)\right)\\
\Pr\left(y(t)|Y(t-1)\right) &\sim \mathcal{N}\left(y(t) : \hat{y}(t|t-1), R_{ee}(t)\right)
\end{aligned}$$

Substituting these probabilities into and combining all constants into a single constant κ, we obtain

$$\begin{aligned}
\Pr\left(x(t)|Y(t)\right) &= \kappa \times \exp\left[-\frac{1}{2}(y(t)-C(t)x(t))'R_{vv}^{-1}(t)(y(t)-C(t)x(t))\right]\\
&\times \exp\left[-\frac{1}{2}(x(t)-\hat{x}(t|t-1))'\tilde{P}^{-1}(t|t-1)(x(t)-\hat{x}(t|t-1))\right]\\
&\times \exp\left[+\frac{1}{2}(y(t)-\hat{y}(t|t-1))'R_{ee}^{-1}(t)(y(t)-\hat{y}(t|t-1))\right]
\end{aligned}$$

Substituting for the individual variables gives the well-known optimal (GM) KALMAN filter algorithm as:



Prediction

$$\hat{x}(t|t-1) = A(t-1)\hat{x}(t-1|t-1) + B(t-1)u(t-1)$$ (state prediction)

$$\tilde{P}(t|t-1) = A(t-1)\tilde{P}(t-1|t-1)A'(t-1) + R_{ww}(t-1)$$ (covariance prediction)

Innovation

$$e(t) = y(t) - \hat{y}(t|t-1) = y(t) - C(t)\hat{x}(t|t-1)$$ (innovation)

$$R_{ee}(t) = C(t)\tilde{P}(t|t-1)C'(t) + R_{vv}(t)$$ (innovation covariance)

Gain

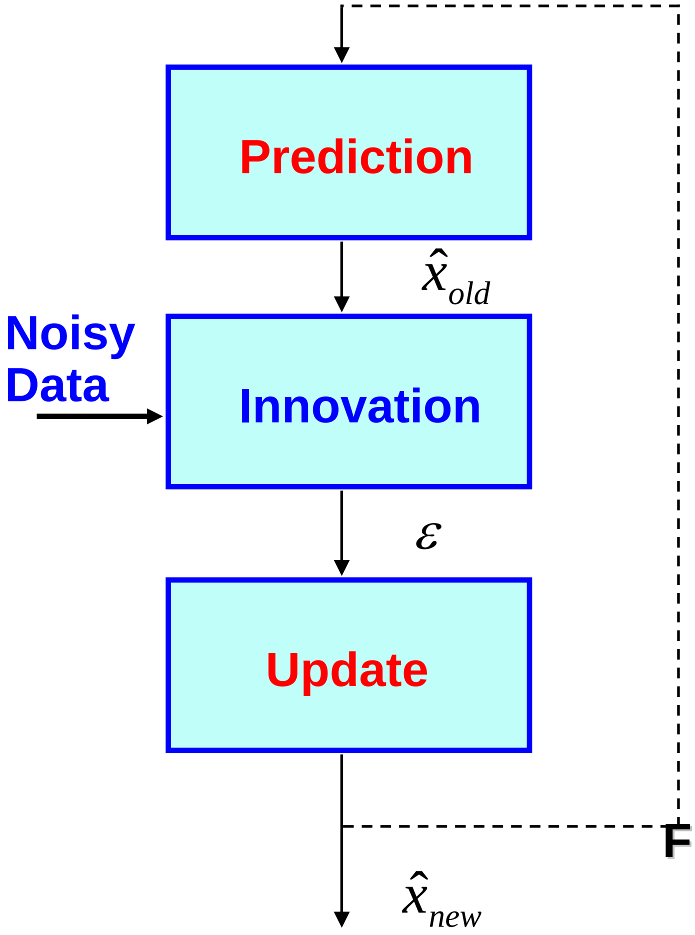
$$K(t) = \tilde{P}(t|t-1)C'(t)R_{ee}^{-1}(t)$$ (gain or weight)

Correction

$$\hat{x}(t|t) = \hat{x}(t|t-1) + K(t)e(t)$$ (state correction)

$$\tilde{P}(t|t) = [I - K(t)C(t)]\tilde{P}(t|t-1)$$ (covariance correction)

When linear state-space models are employed, the Bayesian solution is the “sequential” MODEL-BASED (Kalman filter) algorithm which as the following structure:



$$\hat{x}_{old} = f(\mathbf{model})$$

$$\varepsilon = y - \hat{y}$$

$$\hat{x}_{new} = \hat{x}_{old} + K\varepsilon$$

Filter: $\hat{x}_{new} = \hat{x}_{old} + K\varepsilon$

model **gain** **innovation**

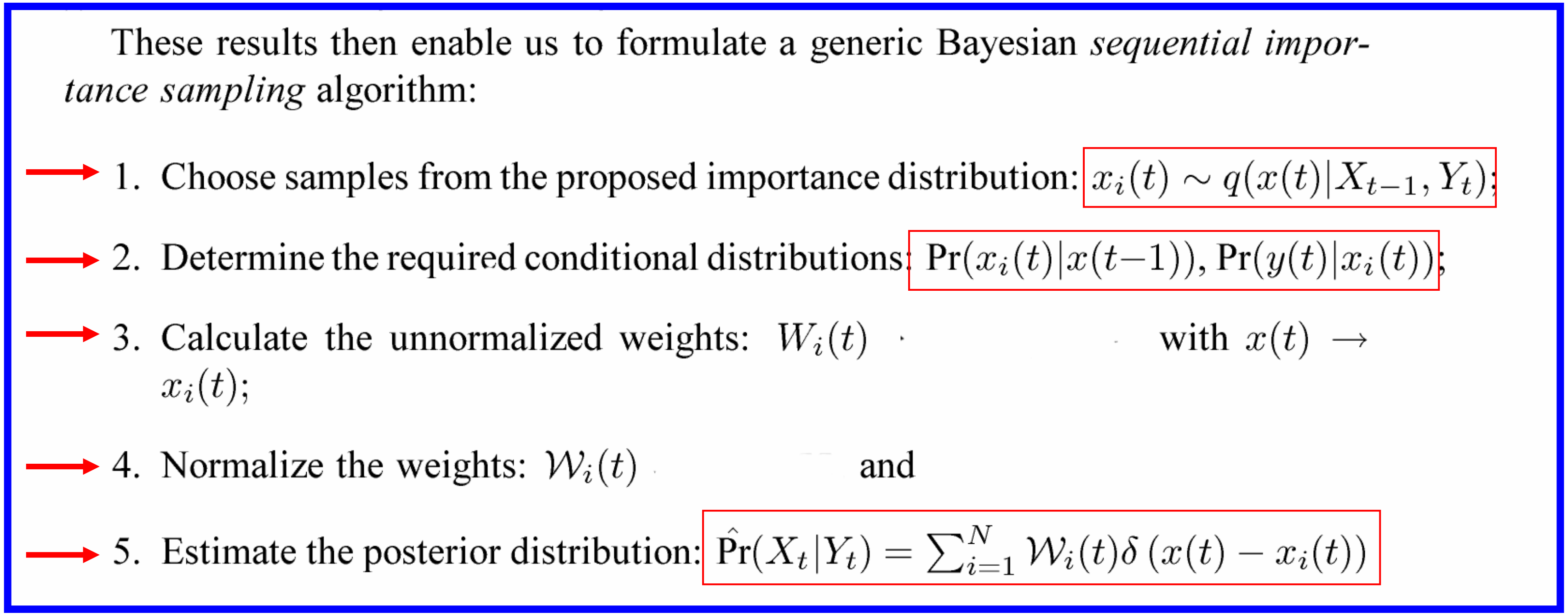
SIMULATION-BASED MC APPROACH TO SEQUENTIAL BAYESIAN PROCESSORS

Importance sampling is a generalization to the MC approach based on the *sampling distribution* that samples the targeted posterior, that is, the sampling distribution is:

$$I = \int_X g(x)dx = \int_X \left(\frac{g(x)}{q(x)}\right) \times q(x)\ dx \quad \text{for} \quad \int q(x)dx = 1$$

- a simpler distribution, *q(x),* than the "posterior" and easier to draw samples
- based on Markov chain theory and therefore will converge to the posterior
- provides non-uniform sampling of the target, *g(x),* giving "more importance" to some values than others
- it is said that the support of *q(x)* "covers" that of *g(x),* i.e., samples drawn from *q* overlap the same region of samples from *g*

This leads to the sequential importance algorithm:



These results then enable us to formulate a generic Bayesian *sequential importance sampling* algorithm:

1. Choose samples from the proposed importance distribution: $x_i(t) \sim q(x(t)|X_{t-1}, Y_t)$;
2. Determine the required conditional distributions: $\Pr(x_i(t)|x(t-1)), \Pr(y(t)|x_i(t))$;
3. Calculate the unnormalized weights: $W_i(t)$ with $x(t) \rightarrow x_i(t)$;
4. Normalize the weights: $\mathcal{W}_i(t)$ and
5. Estimate the posterior distribution: $\hat{\Pr}(X_t|Y_t) = \sum_{i=1}^{N} \mathcal{W}_i(t)\delta\left(x(t) - x_i(t)\right)$

State-space Bayesian processors based on sequential importance samplers follow easily as:

$$\boxed{W(t) = W(t-1) \times \frac{\Pr(y(t)|x(t)) \times \Pr(x(t)|x(t-1))}{q(x(t)|X_{t-1}, Y_t)}}$$

Now let us recall the general state-space characterization representing the transition and likelihood probabilities as:

$$\begin{aligned} \Pr\left(x(t)|x(t-1)\right) &\Leftrightarrow \mathcal{A}\left(x(t)|x(t-1)\right) \\ \Pr\left(y(t)|x(t)\right) &\Leftrightarrow \mathcal{C}\left(y(t)|x(t)\right) \end{aligned}$$

Assuming this is true, then the *SSPF* recursion becomes

$$\begin{aligned} x_i(t) &\sim q(x(t)|x(t-1), y(t)) \\ W_i(t) &= W_i(t-1) \times \boxed{\frac{\mathcal{C}(y(t)|x_i(t)) \times \mathcal{A}(x_i(t)|x_i(t-1))}{q(x_i(t)|x_i(t-1), y(t))}} \\ \mathcal{W}_i(t) &= \frac{W_i(t)}{\sum_{i=1}^{N_p} W_i(t)} \end{aligned}$$

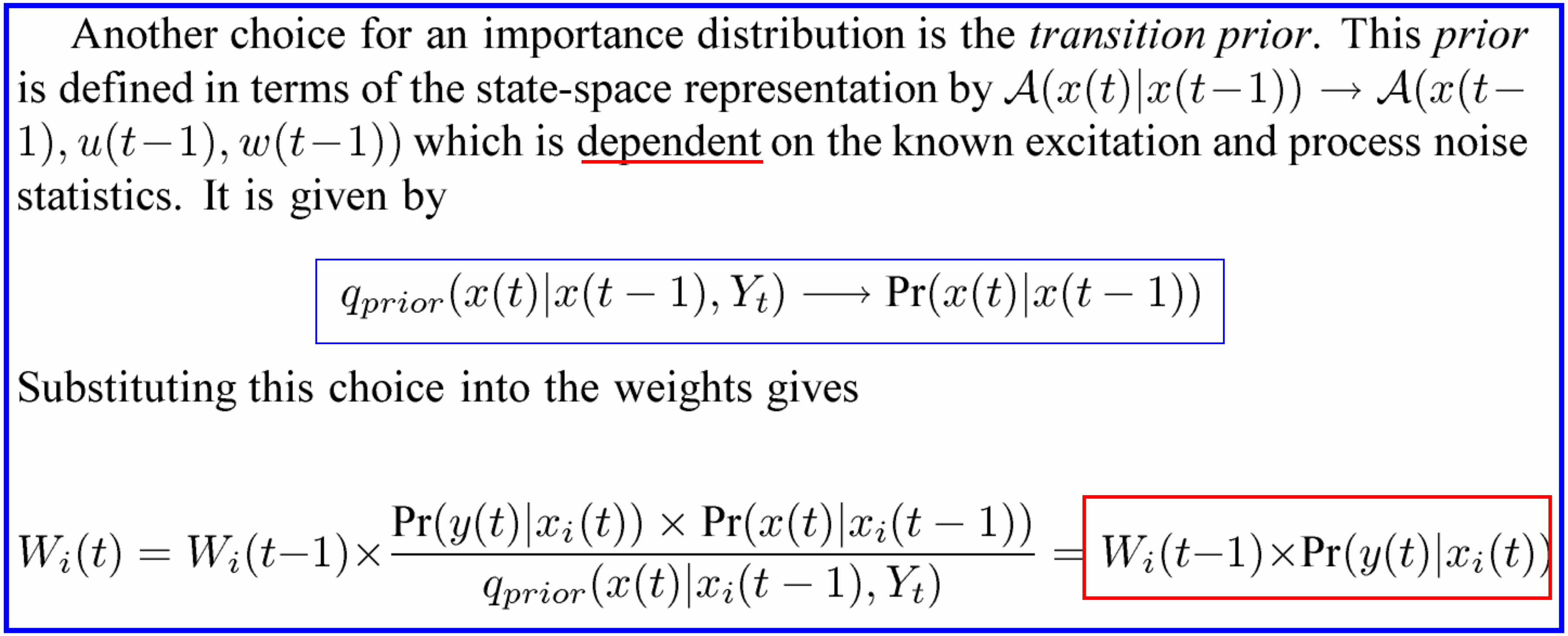
and the filtering *posterior* is estimated by

$$\boxed{\hat{\Pr}(x(t)|Y_t) \approx \sum_{i=1}^{N_p} \mathcal{W}_i(t) \times \delta(x(t) - x_i(t))}$$

Note that as N_p becomes large, in the limit, we have

$$\lim_{N_p \to \infty} \hat{\Pr}(x(t)|Y_t) \longrightarrow \Pr(x(t)|Y_t)$$

Importance distributions provide the key: “Transition Prior” (Gordon et. al. ’93)



Another choice for an importance distribution is the *transition prior*. This *prior* is defined in terms of the state-space representation by $\mathcal{A}(x(t)|x(t-1)) \rightarrow \mathcal{A}(x(t-1), u(t-1), w(t-1))$ which is dependent on the known excitation and process noise statistics. It is given by

$$\boxed{q_{prior}(x(t)|x(t-1), Y_t) \longrightarrow \Pr(x(t)|x(t-1))}$$

Substituting this choice into the weights gives

$$W_i(t) = W_i(t-1) \times \frac{\Pr(y(t)|x_i(t)) \times \Pr(x(t)|x_i(t-1))}{q_{prior}(x(t)|x_i(t-1), Y_t)} = \boxed{W_i(t-1) \times \Pr(y(t)|x_i(t))}$$

BOOTSTRAP ESTIMATOR

PARTICLE FILTERS FOR SBP

PARTICLE FILTERS (PF)

- PF are sequential MC techniques in which the underlying posterior distribution of interest is represented by a “cloud” of random samples (particles) in the state/parameter space
- The PF is an algorithm that (sequentially) propagates and updates the random samples drawn from the previous stage to obtain a set of samples approximately distributed from the next stage, that is,

$$\Pr\left[X_t \middle| \mathbf{Y}_t\right] = W_i\left(t,t-1;X_i(t)\right) \times \Pr\left[X_{t-1} \middle| \mathbf{Y}_{t-1}\right]$$

i-th Particle

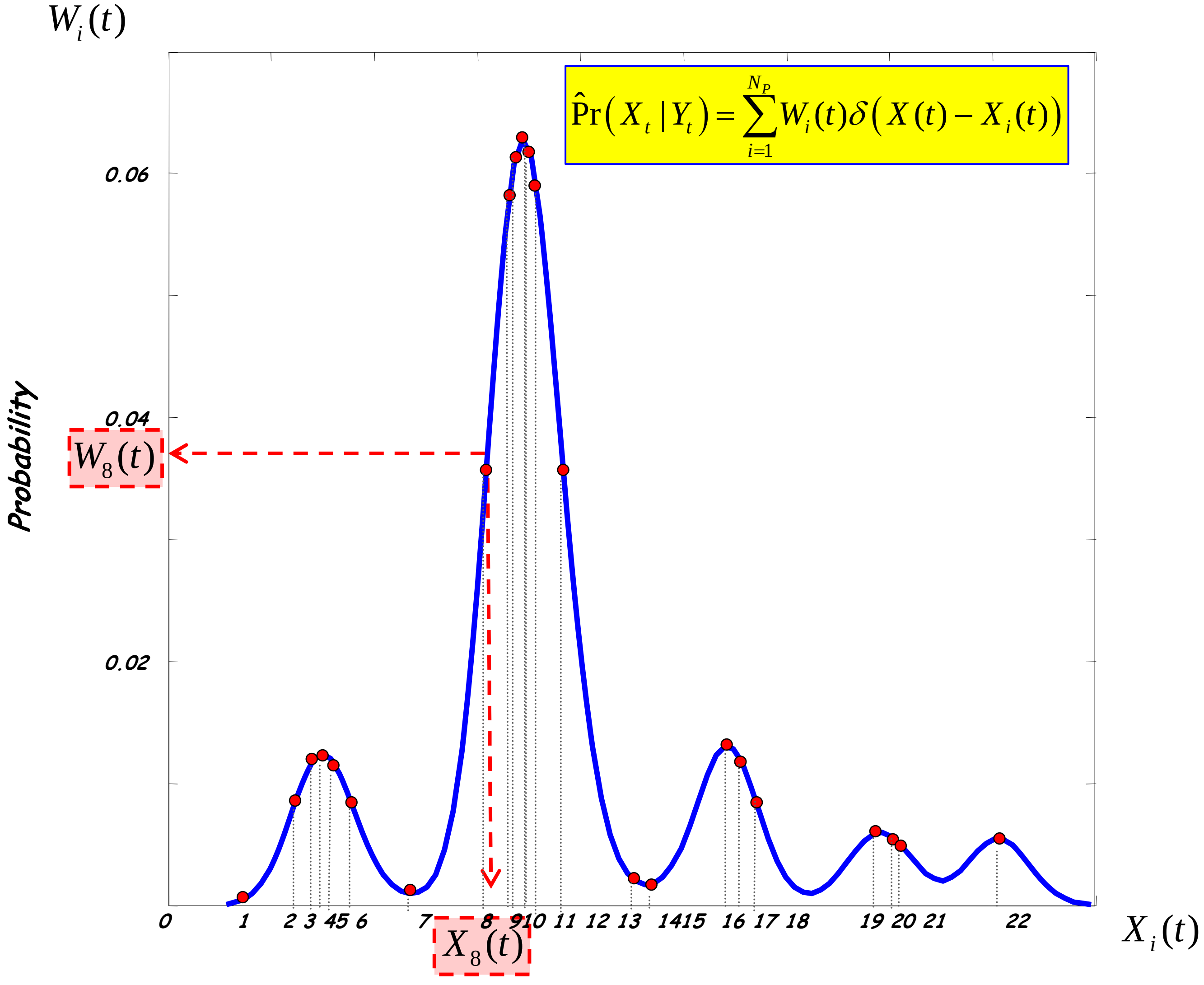
where $\mathrm{W_i}\left(t,t-1;X_i(t)\right)$ is the weight (Bayes' operator) defined earlier and

$X_i(t)$ is the i^{th} – particle at stage (time) t

Next Stage

Previous Stage

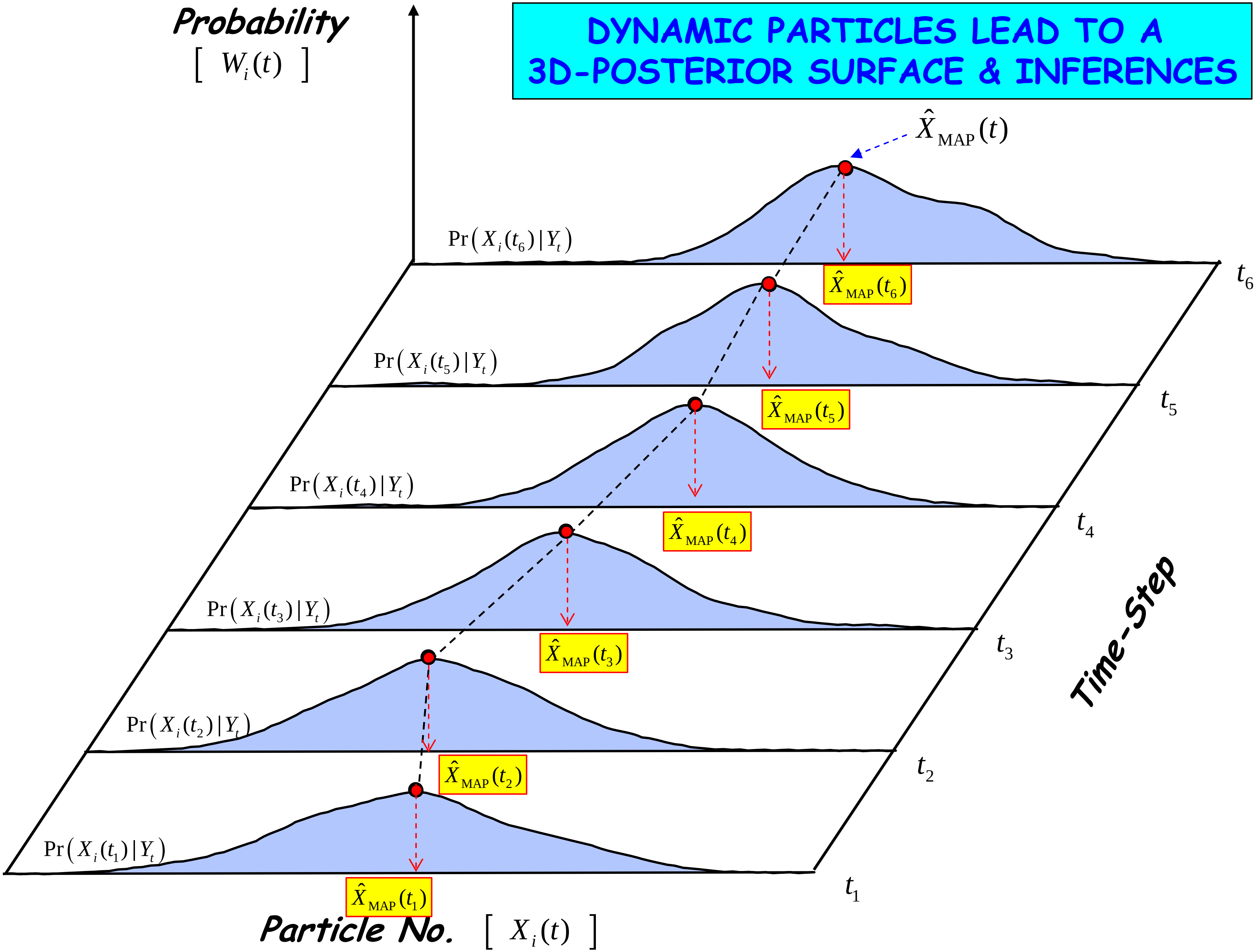
Particle filters can be used to construct posterior distributions. In the dynamic case they are used to estimate the "instantaneous" posterior



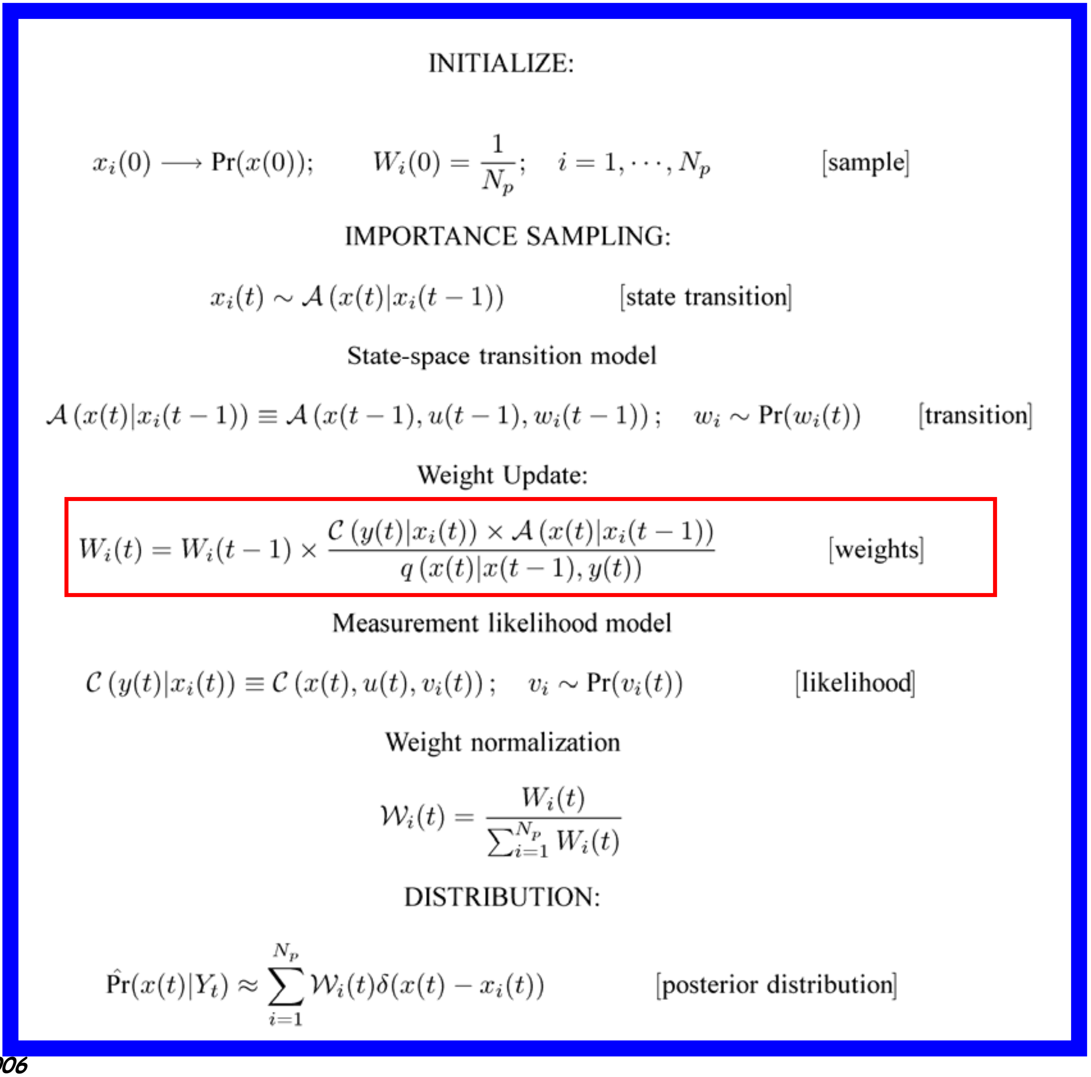
$$\hat{\Pr}\left(X_t \mid Y_t\right) = \sum_{i=1}^{N_P} W_i(t)\,\delta\left(X(t) - X_i(t)\right)$$

Particle No. (Time-Step 68)

DYNAMIC PARTICLES LEAD TO A 3D-POSTERIOR SURFACE & INFERENCES


Probability
$[\ W_i(t)\ ]$
$\hat{X}_{\text{MAP}}(t)$
$\Pr\left(X_i(t_6)\,|\,Y_t\right)$
$\Pr\left(X_i(t_5)\,|\,Y_t\right)$
$\Pr\left(X_i(t_4)\,|\,Y_t\right)$
$\Pr\left(X_i(t_3)\,|\,Y_t\right)$
$\Pr\left(X_i(t_2)\,|\,Y_t\right)$
$\Pr\left(X_i(t_1)\,|\,Y_t\right)$
$\hat{X}_{\text{MAP}}(t_6)$
$\hat{X}_{\text{MAP}}(t_5)$
$\hat{X}_{\text{MAP}}(t_4)$
$\hat{X}_{\text{MAP}}(t_3)$
$\hat{X}_{\text{MAP}}(t_2)$
$\hat{X}_{\text{MAP}}(t_1)$
t_6
t_5
t_4
t_3
t_2
t_1
Time-Step
Particle No. $[\ X_i(t)\ ]$

The "generic" State-Space particle filtering method is given by:



INITIALIZE:

$$x_i(0) \longrightarrow \Pr(x(0)); \qquad W_i(0) = \frac{1}{N_p}; \quad i = 1, \cdots, N_p \qquad \text{[sample]}$$

IMPORTANCE SAMPLING:

$$x_i(t) \sim \mathcal{A}\left(x(t)|x_i(t-1)\right) \qquad \text{[state transition]}$$

State-space transition model

$$\mathcal{A}\left(x(t)|x_i(t-1)\right) \equiv \mathcal{A}\left(x(t-1), u(t-1), w_i(t-1)\right); \quad w_i \sim \Pr(w_i(t)) \qquad \text{[transition]}$$

Weight Update:

$$\boxed{W_i(t) = W_i(t-1) \times \frac{\mathcal{C}\left(y(t)|x_i(t)\right) \times \mathcal{A}\left(x(t)|x_i(t-1)\right)}{q\left(x(t)|x(t-1), y(t)\right)} \qquad \text{[weights]}}$$

Measurement likelihood model

$$\mathcal{C}\left(y(t)|x_i(t)\right) \equiv \mathcal{C}\left(x(t), u(t), v_i(t)\right); \quad v_i \sim \Pr(v_i(t)) \qquad \text{[likelihood]}$$

Weight normalization

$$\mathcal{W}_i(t) = \frac{W_i(t)}{\sum_{i=1}^{N_p} W_i(t)}$$

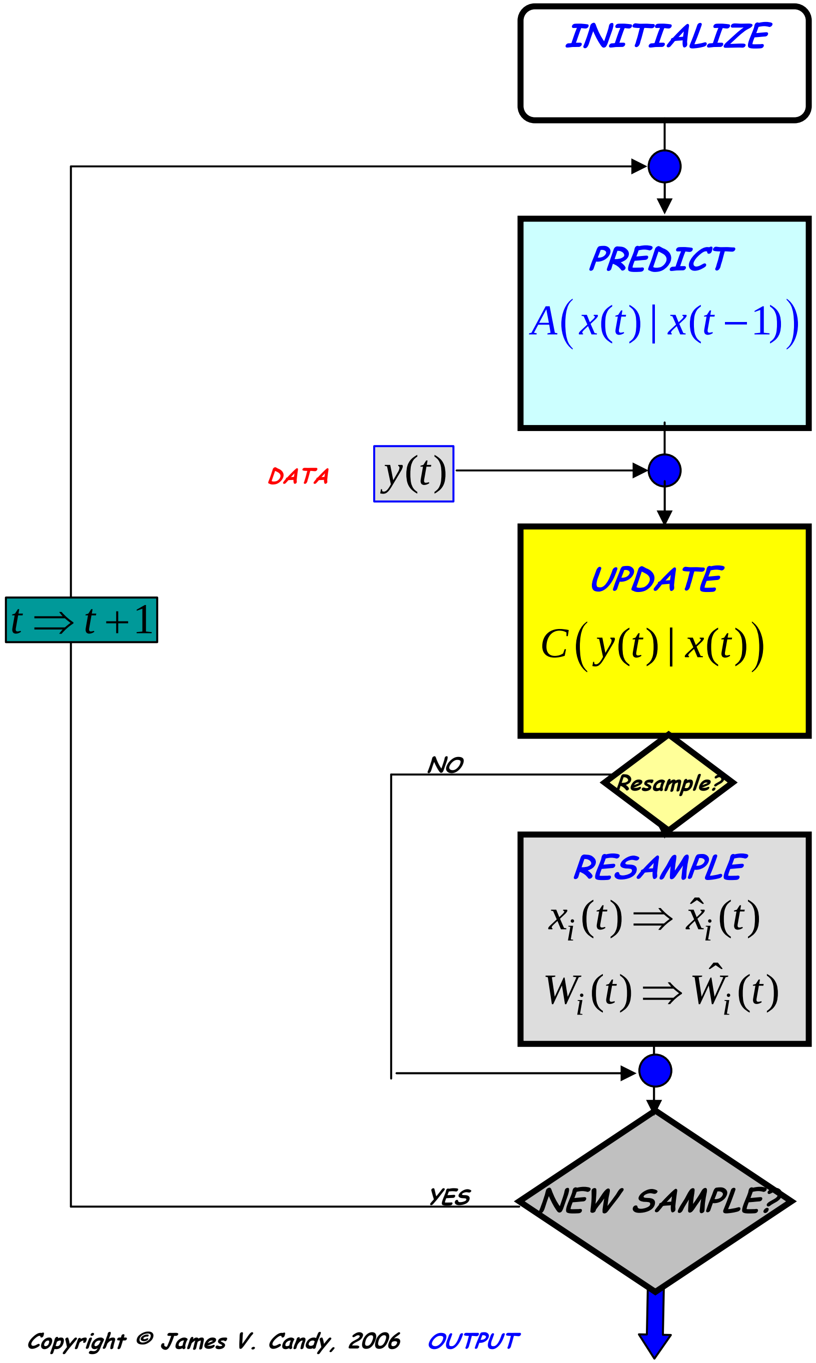
DISTRIBUTION:

$$\hat{\Pr}(x(t)|Y_t) \approx \sum_{i=1}^{N_p} \mathcal{W}_i(t)\delta(x(t) - x_i(t)) \qquad \text{[posterior distribution]}$$

PROBLEM: Particles deplete in number (degenerate) to a single particle due to the increased variance in each step; therefore,

- The particles must be "rejuvenated" or equivalently resampled
- Resampling inhibits the depletion problem, but increases the uncertainty (weight variance)
- If not implemented properly, it can also increase computational time extensively (non-parallel)
- Resampling is essentially a process that attempts to preserve particles with large weights (acceptance probabilities) while discarding those with small weights.

STATE-SPACE SIR ALGORITHM



INITIALIZE
PREDICT
$A\left(x(t)\,|\,x(t-1)\right)$
DATA
$y(t)$
UPDATE
$C\left(y(t)\,|\,x(t)\right)$
NO
Resample?
RESAMPLE
$x_i(t) \Rightarrow \hat{x}_i(t)$
$W_i(t) \Rightarrow \hat{W}_i(t)$
$t \Rightarrow t+1$
YES
NEW SAMPLE?
OUTPUT

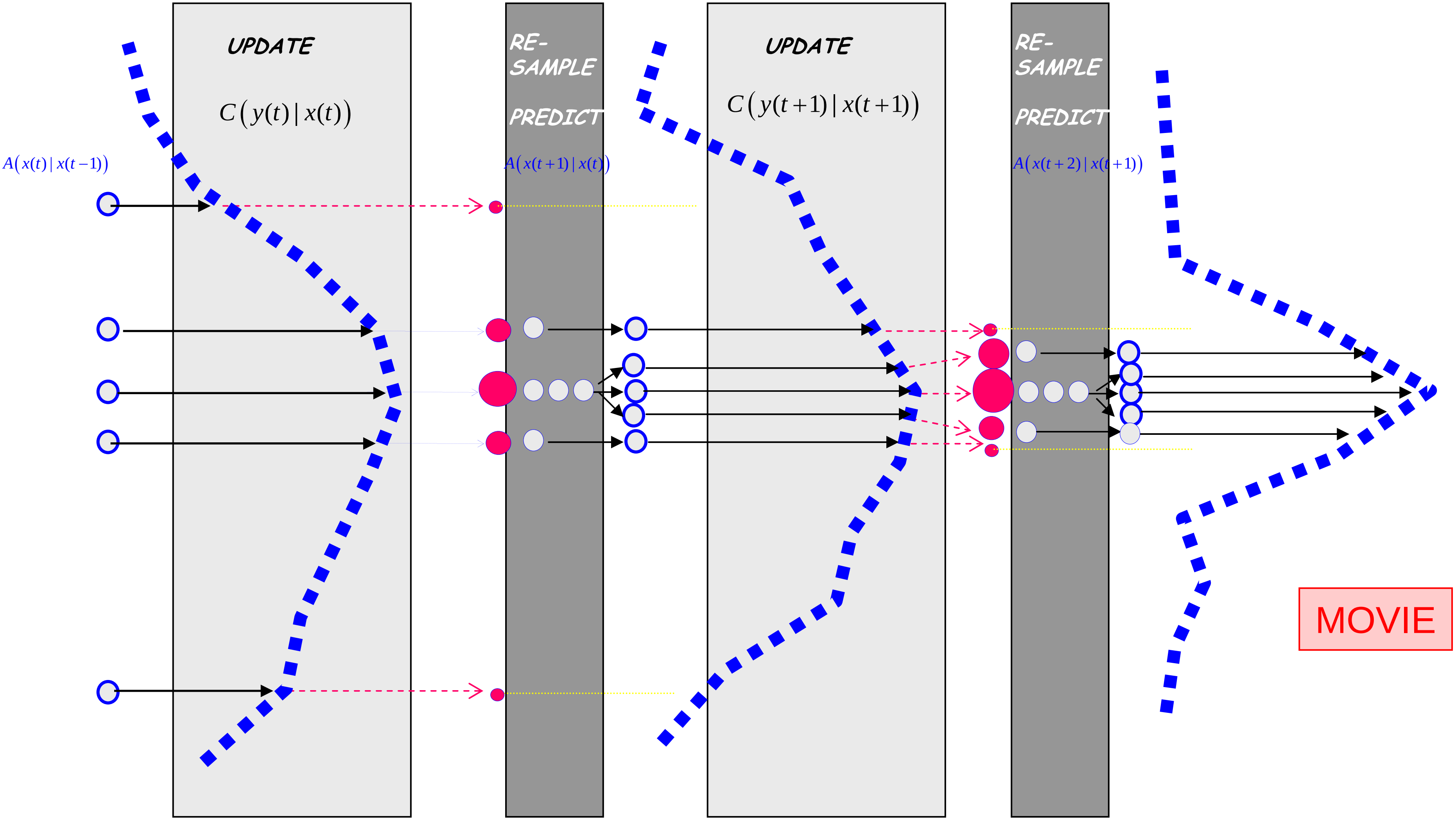
$\left\{\hat{x}_i(t-1), \hat{W}_i(t-1)\right\} \Rightarrow \Pr\left[x(t-1)\middle|Y_{t-1}\right]$

$\left\{\hat{x}_i(t), \hat{W}_i(t)\right\} \Rightarrow \Pr\left[x(t)\middle|Y_{t-1}\right]$

$\left\{x_i(t), W_i(t)\right\} \Rightarrow \Pr\left[x(t)\middle|Y_t\right]$

$\left\{x_i(t), W_i(t)\right\} \Rightarrow \Pr\left[x(t)\middle|Y_t\right]$

$\left\{\hat{x}_i(t-1),\hat{W}_i(t-1)\right\}$ → $\left\{x_i(t),W_i(t)\right\}$ $\left\{\hat{x}_i(t),\hat{W}_i(t)\right\}$ → $\left\{x_i(t+1),W_i(t+1)\right\}$ $\left\{\hat{x}_i(t+1),\hat{W}_i(t+1)\right\}$ →



$A\left(x(t)\,|\,x(t-1)\right)$
UPDATE
$C\left(y(t)\,|\,x(t)\right)$
RE-SAMPLE
PREDICT
$A\left(x(t+1)\,|\,x(t)\right)$
UPDATE
$C\left(y(t+1)\,|\,x(t+1)\right)$
RE-SAMPLE
PREDICT
$A\left(x(t+2)\,|\,x(t+1)\right)$

MOVIE

$\Pr\left[x(t-1)|Y_{t-1}\right]$ → $\Pr\left[x(t)|Y_t\right]$ → $\Pr\left[x(t+1)|Y_{t+1}\right]$ →

$t-1$ ght © James V. Candy, 2006 t $t+1$

Particle Filtering: Examples, Applications

SUMMARY

- Many problems can be cast into a Bayesian framework in order to solve a suite of problems
- No longer are we restricted to nonlinear approximations and gaussian processes
- Sequential Bayesian processors evolve as natural extensions which incorporate Markovian state-space structures
- Problem solutions for highly uncertain, noisy measurements using physics-based signal processing models and sequential Bayesian processing implemented with particle filtering techniques add a new tool for signal processors

REFERENCES

1. A. Papoulis, *Probability, Random Variables and Stochastic Processes*, McGraw-Hill: New York, 1991.

2. A. Jazwinski, *Stochastic Processes and Filtering Theory*, Academic Press, New York, 1970.

3. A. Sage and J. Melsa, *Estimation Theory with Applications to Communications and Control*, McGraw-Hill, New York, 1971.

4. B. Anderson and J. Moore, *Optimal Filtering*, Prentice-Hall: New Jersey, 1979.

5. T. Kailath, *Lectures on Kalman and Wiener Filtering Theory*, Springer-Verlag, New York, 1981.

6. H. Sorenson, Editor, *Kalman Filtering: Theory and Application*, IEEE Press, New York, 1985.

7. J. V. Candy, *Signal Processing: The Model-Based Approach*, McGraw-Hill, New York, 1986.

8. A. Harvey, *Forcasting, Structural Time Series Models and the Kalman Filter*, Cambridge Univ. Press: Cambridge, UK, 1989.

9. L. Scharf, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*, Addison-Wesley, Massachusetts, 1991.

10. M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice*, Prentice-Hall, New Jersey, 1993.

11. J. M. Mendel, *Lessons in Estimation Theory for Signal Processing, Communications, and Control*, Prentice-Hall, 1995.

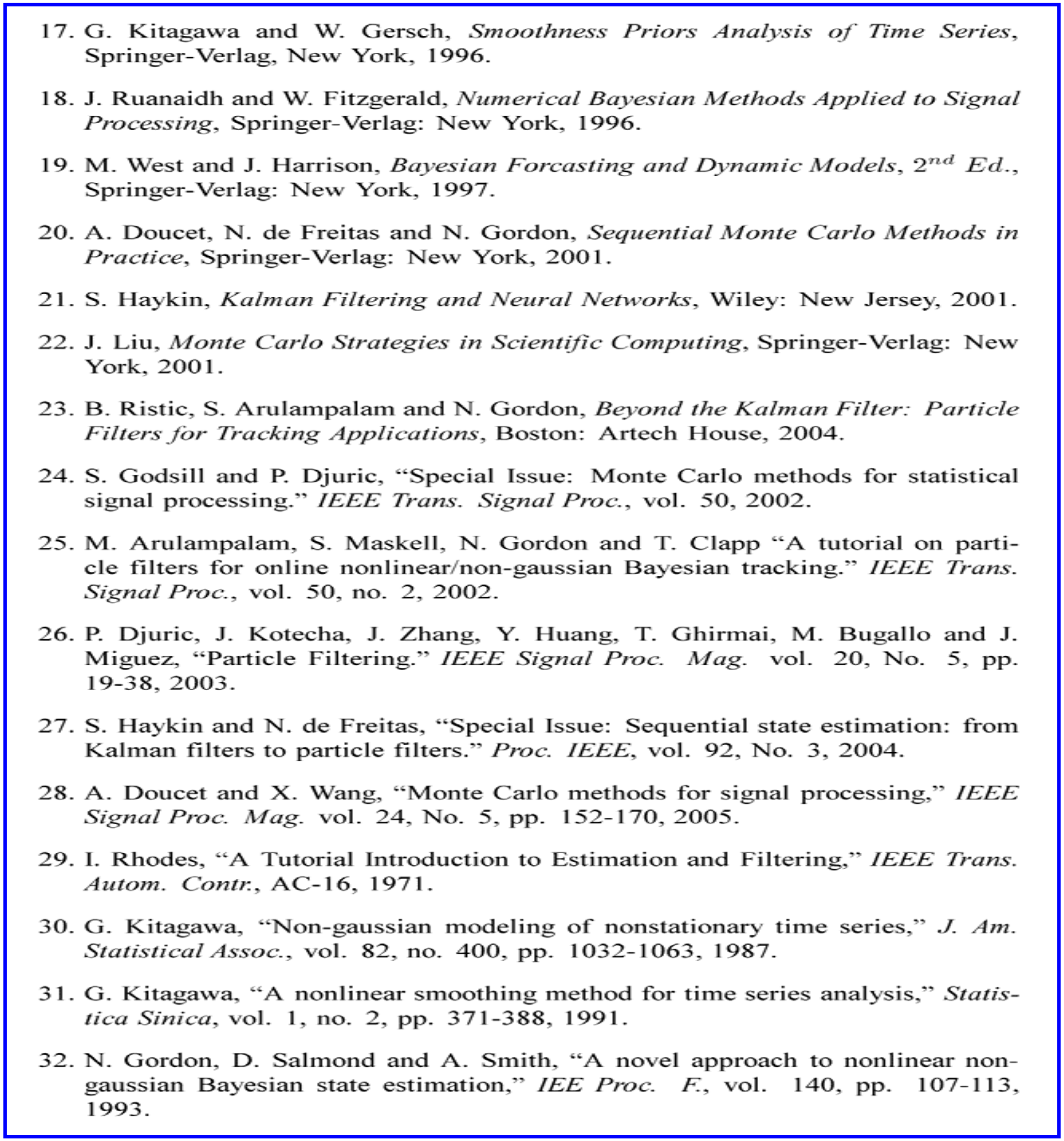
12. M. Srinath, P. Rajasekaran and R. Viswanathan. *Introduction to Statistical Signal Processing with Applications*, Prentice-Hall, New Jersey, 1996.

13. R. G. Brown and P.C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*, John Wiley, New York, 1997.

14. J. V. Candy, *Model-Based Signal Processing*, Wiley/IEEE Press, New Jersey, 2006.

15. H. Tanizaki, *Nonlinear Filters*, Springer-Verlag: New York, 1993.

16. M. Tanner, *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions*, 2nd *Ed.*, Springer-Verlag: New York,1993.



17. G. Kitagawa and W. Gersch, *Smoothness Priors Analysis of Time Series*, Springer-Verlag, New York, 1996.

18. J. Ruanaidh and W. Fitzgerald, *Numerical Bayesian Methods Applied to Signal Processing*, Springer-Verlag: New York, 1996.

19. M. West and J. Harrison, *Bayesian Forcasting and Dynamic Models*, 2^{nd} *Ed.*, Springer-Verlag: New York, 1997.

20. A. Doucet, N. de Freitas and N. Gordon, *Sequential Monte Carlo Methods in Practice*, Springer-Verlag: New York, 2001.

21. S. Haykin, *Kalman Filtering and Neural Networks*, Wiley: New Jersey, 2001.

22. J. Liu, *Monte Carlo Strategies in Scientific Computing*, Springer-Verlag: New York, 2001.

23. B. Ristic, S. Arulampalam and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Boston: Artech House, 2004.

24. S. Godsill and P. Djuric, "Special Issue: Monte Carlo methods for statistical signal processing." *IEEE Trans. Signal Proc.*, vol. 50, 2002.

25. M. Arulampalam, S. Maskell, N. Gordon and T. Clapp "A tutorial on particle filters for online nonlinear/non-gaussian Bayesian tracking." *IEEE Trans. Signal Proc.*, vol. 50, no. 2, 2002.

26. P. Djuric, J. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. Bugallo and J. Miguez, "Particle Filtering." *IEEE Signal Proc. Mag.* vol. 20, No. 5, pp. 19-38, 2003.

27. S. Haykin and N. de Freitas, "Special Issue: Sequential state estimation: from Kalman filters to particle filters." *Proc. IEEE*, vol. 92, No. 3, 2004.

28. A. Doucet and X. Wang, "Monte Carlo methods for signal processing," *IEEE Signal Proc. Mag.* vol. 24, No. 5, pp. 152-170, 2005.

29. I. Rhodes, "A Tutorial Introduction to Estimation and Filtering," *IEEE Trans. Autom. Contr.*, AC-16, 1971.

30. G. Kitagawa, "Non-gaussian modeling of nonstationary time series," *J. Am. Statistical Assoc.*, vol. 82, no. 400, pp. 1032-1063, 1987.

31. G. Kitagawa, "A nonlinear smoothing method for time series analysis," *Statistica Sinica*, vol. 1, no. 2, pp. 371-388, 1991.

32. N. Gordon, D. Salmond and A. Smith, "A novel approach to nonlinear non-gaussian Bayesian state estimation," *IEE Proc. F.*, vol. 140, pp. 107-113, 1993.

33. G. Kitagawa, "Monte Carlo filter and smoother for non-gaussian nonlinear state-space models," *J. Comput. Graphical Stat.*, vol. 5, no. 1, pp. 1-25, 1996.

34. G. Kitagawa, "Self-organizing state space model," *J. Am. Statistical Assoc.*, vol. 93, no. 443, pp. 1203-1215, 1998.

35. M. Isard and A. Blake, "Condensation–conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5-28, 1998.

36. N. Metropolis, N. Rosenbutt, A. Rosenbutt, M . Teller and A. Teller, "Equations of state calculations by fast computing machnines," *J. Chem. Phys.*, vol. 21, pp. 1087-1092, 1953.

37. W. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, pp. 97-109,1970.

38. C. Andrieu, A. Doucet, S. Singh, and V. Tadic, "Particle methods for change detection, system identification and control," *Proc. IEEE*, vol. 92, No. 3, pp. 423-438, 2004.

39. A. Doucet, S. Godsill and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statist. Comput.*, vol. 10, no. 3., pp. 197-208, 2000.

40. R. van der Merwe, A. Doucet, N. de Freitas and E. Wan, "The unscented particle filter," in *Advances in Neural Information Processing Systems 13*, MIT Press: Cambridge, MA, 2000.

41. A. Kong, J. Liu and W. Wong, "Sequential imputations and Bayesian missing data problems," *J. Am. Statistical Assoc.*, vol. 89, no. 425, pp. 278-288, 1994.

42. J. Liu and R. Chen, "Blind deconvolution via sequential imputations," *J. Am. Statistical Assoc.*, vol. 90, no. 430, pp. 567-576, 1995.

43. J. Liu, R. Chen and W. Wong, "Rejection control and sequential importance sampling," *J. Am. Statistical Assoc.*, vol. 93, no. 443, pp. 1022-1031, 1998.

44. J. Liu and R. Chen, "Sequential Monte Carlo methods for dynamic systems," *J. Am. Statistical Assoc.*, vol. 93, no. 443, pp. 1032-1044, 1998.

45. M. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *J. Am. Statistical Assoc.*, vol. 94, no. 446, pp. 590-599, 1999.

46. A. Smith and A. Gelfand, "Bayesian statistics without tears: a sampling-resampling perspective," *Am. Statistician*, vol. 44, no. 4, pp. 84-88, 1992.

47. H. Sorenson and D. Alspach, "Recursive Bayesian estimation using gaussian sums," *Automatica*, vol. 7, pp. 465-479, 1971.

48. R. van der Merwe, *Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models* **OGI School of Science & Engr.**, Oregon Health & Science Univ., PhD Dissertation, 2004.

49. T. Schoen, *Estimation of Nonlinear Dynamic Systems: Theory and Applications* **Linkopings Univ.**, Linkoping, Sweden PhD Dissertation, 2006.

50. M. Andrade Netto, L. Gimeno and J. Mendes, "On the optimal and suboptimal nonlinear filtering problem for discrete-time systems", *IEEE Trans. Auto. Control*, AC-23, No. 6, pp. 1063-1067, 1978.

51. H. Van Trees, *Detection, Estimation and Modulation Theory*, Wiley, New York, 1968.

52. R. Hogg and A. Craig, *Introduction to Mathematical Statistics*, MacMillan, New York, 1970.

53. B. Porat, *Digital Processing of Random Signals: Theory and Methods*, New Jersey:Prentice-Hall, 1994.

54. J. Candy, "The Cramer-Rao Bound: A Measure of Estimator Quality," Lawrence Livermore National Laboratory Report, UCID-17660, 1977.

55. J. Candy, *Signal Processing: The Model-Based Approach*, McGraw-Hill, New York, 1986.

56. J. Candy, "Processing of RV Signatures from Radar Tracking Data," *RTO Set Symposium on High Resolution Radar Techniques*, RTO MP-40, pp.38-1 to 38-12, 1999.

57. G. Strang, *Linear Algebra and Its Applications*, New York: Academic Press, 1976.

58. G. Golub and C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, 1989.

59. C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*, Prentice-Hall, New Jersey, 1974.

60. S. Orfanidis, *Optimum Signal Processing*, MacMillan, New York, 1985).

61. G. Stewart, *Introduction to Matrix Computations*, Academic Press, New York, 1973.

62. G. J. Bierman, *Factorization Methods for Discrete Sequential Estimation*, Academic Press, 1977.

63. G. C. Goodwin and R. L. Payne, *Dynamic System Idenfication*, Academic Press, 1976.