Short Course

State Space Models, Generalized Dynamic Systems
and
Sequential Monte Carlo Methods,
and
their applications
in Engineering, Bioinformatics and Finance

Rong Chen
Rutgers University
Peking University

Part One: Introduction

1.1 State Space Models and Probabilistic Dynamic Systems, with Examples in Enginering, Bioinformatics and Finance

1.2 Review: Basic Monte Carlo Methods

1.3 Introduction of Sequential Monte Carlo Methods

Part Two: Sequential Monte Carlo Methods – the Framework and Implementation

2.1 A Framework

2.1.1 (Optional) Intermediate Distributions

2.1.2 Propogation: Sampling Distribution

2.1.3 Resampling/Rejuvenation

2.1.4 Inference: Rao-Blackwellization

2.2 Some Theoretically Results

2.3 Some Applications (in detail)

Part Three: Advanced Sequential Monte Carlo

3.1 Mixture Kalman Filter

3.1.1 Conditional Dynamic Linear Models

3.1.2 Mixture Kalman Filters

3.1.3 Partial Conditional Dynamic Linear Models

3.1.4 Extend Mixture Kalman Filters

3.1.5 Future Directions

3.2 Constrained SMC

3.3 Parametrer Estimation in SMC

1.1. State Space Models

state equation: $x_t = g_t(x_{t-1}, \varepsilon_t)$ **or** $x_t \sim q_t(\cdot \mid x_{t-1})$

observation equation: $y_t = h_t(x_t, e_t)$ **or** $y_t \sim f_t(\cdot \mid x_t)$

$$\begin{array}{ccccccc} \cdots & & y_t & & y_{t+1} & & \cdots \\ & & \uparrow & & \uparrow & & \\ \cdots & \longrightarrow & x_t & \rightarrow & x_{t+1} & \longrightarrow & \cdots \end{array}$$

$$\pi_t(\boldsymbol{x}_t) = p(x_1, \ldots, x_t \mid y_1, \ldots, y_t) \propto \prod_{s=1}^{t} f_s(y_s \mid x_s) q_s(x_s \mid x_{s-1})$$

Objective:

(1) Estimation: $p(x_t \mid y_1, \ldots y_t)$

(2) Prediction: $p(x_{t+1} \mid y_1, \ldots, y_t)$

(3) Smoothing: $p(x_1, \ldots, x_{t-1} \mid y_1, \ldots, y_t)$

(3.1) delayed estimation: $p(x_{t-d} \mid y_1, \ldots, y_t)$

On-Line in Real Time

Linear and Gaussian Systems:

$$\begin{aligned} x_t &= H_t x_{t-1} + W_t w_t \\ y_t &= G_t x_t + V_t v_t \end{aligned}$$

where $w_t \sim N(0, I)$ **and** $v_t \sim N(0, I)$**.**

$$p(x_t \mid y_1, \ldots y_t) \sim N(\mu_t, \Sigma_t)$$

<u>Kalman Filter</u>:

Recursive updating:

$$(\mu_t, \Sigma_t) \rightarrow (\mu_{t+1}, \Sigma_{t+1})$$

Very easy and fast!

Nonlinear and Non-Gaussian:

Easy: $p(x_1, \ldots, x_t \mid \boldsymbol{y}_t)$

Difficult:

$$E(x_t \mid \boldsymbol{y}_t) = \int \cdots \int x_t p(x_1, \ldots, x_t \mid \boldsymbol{y}_t) dx_1 \ldots dx_t$$

where $\boldsymbol{y}_t = (y_1, \ldots, y_t)$**.**

<u>Our approach</u>: Monte Carlo method

Generate samples $\boldsymbol{x}_t^{(1)}, \ldots, \boldsymbol{x}_t^{(m)}$ **from the target distribution** $p(\boldsymbol{x}_t \mid \boldsymbol{y}_t)$**, then use approximation**

$$E[h(\boldsymbol{x}_t) \mid \boldsymbol{y}_t] \approx \frac{\sum_{i=1}^m h(\boldsymbol{x}_t^{(i)})}{m}$$

Example 1: Target Tracking

A single target moving on a straight line with random (Gaussian) acceleration in a clutter environment: $x_t = (d_t, v_t)$**.**

Constant acceleration within a period $a_t = w_t/T$**.** T **is the time duration between two observations.**

State Equation (motion model):

$$\begin{aligned} d_t &= d_{t-1} + v_{t-1}T + w_t T/2, \qquad w_t \sim N(0, q^2) \\ v_t &= v_{t-1} + w_t \end{aligned}$$

Observation equation:

$$z_t = d_t + e_t, \qquad e_t \sim N(0, r^2)$$

The system is linear and Gaussian.

Example 2: Tracking a target in clutter

State Equation (motion model):

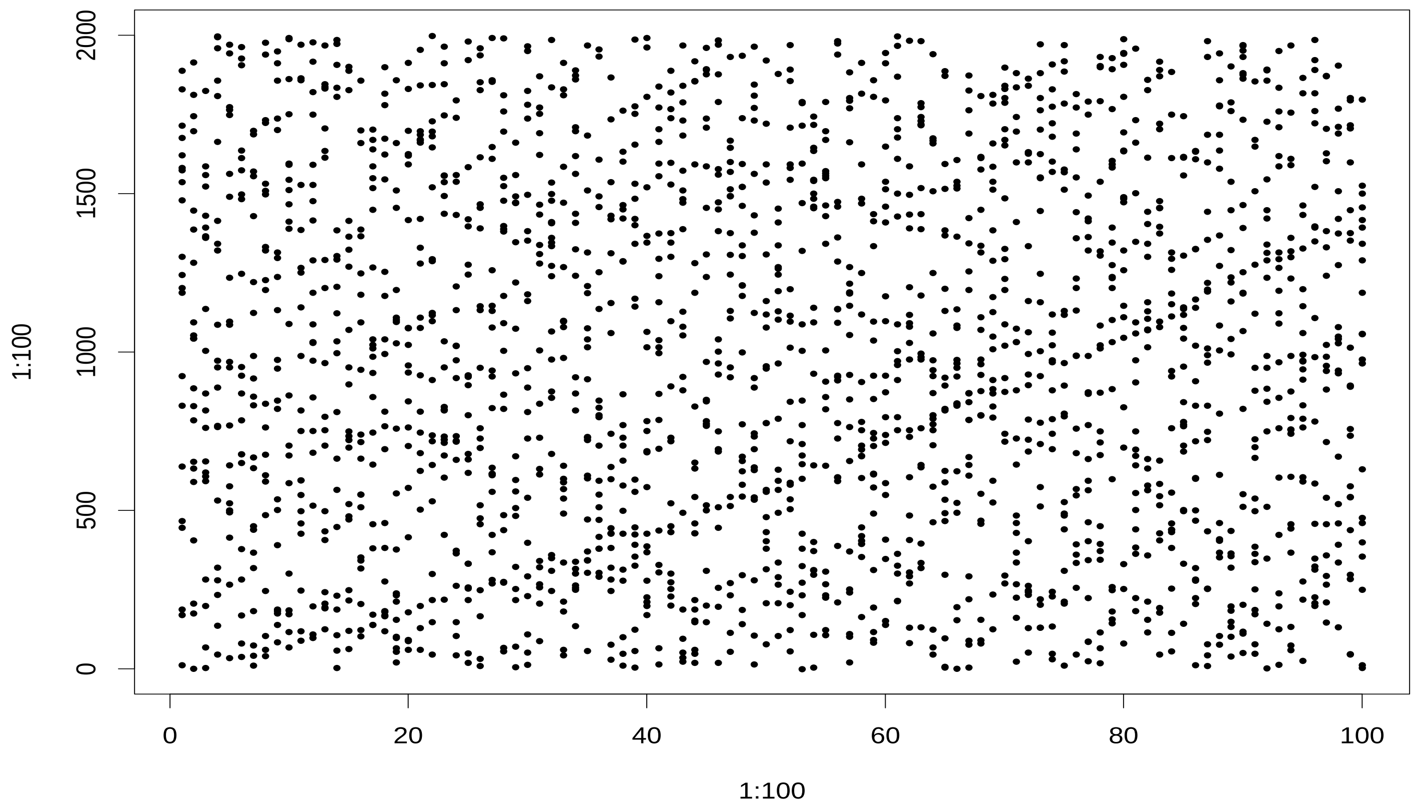
$$
\begin{aligned}
d_t &= d_{t-1} + v_{t-1}T + w_t T/2 \\
v_t &= v_{t-1} + w_t
\end{aligned}
$$

At time t, observe m_t signals, where

$$m_t \sim \textbf{Bernoulli}(p_d) + \textbf{Poisson}(\lambda\Delta)$$

The true signal $z_t = d_t + e_t$ has probability p_d to be observed, The false signals are uniformly distributed in the detection region Δ.

$$T = 1, \; p_d = 0.9, \; \lambda = 0.1, \; Var(w_t) = 0.1, \; Var(e_t) = 1$$



Example 3: Tracking a maneuvering target

A single target moving in a 2-d space with random (Gaussian) acceleration plus maneuvering

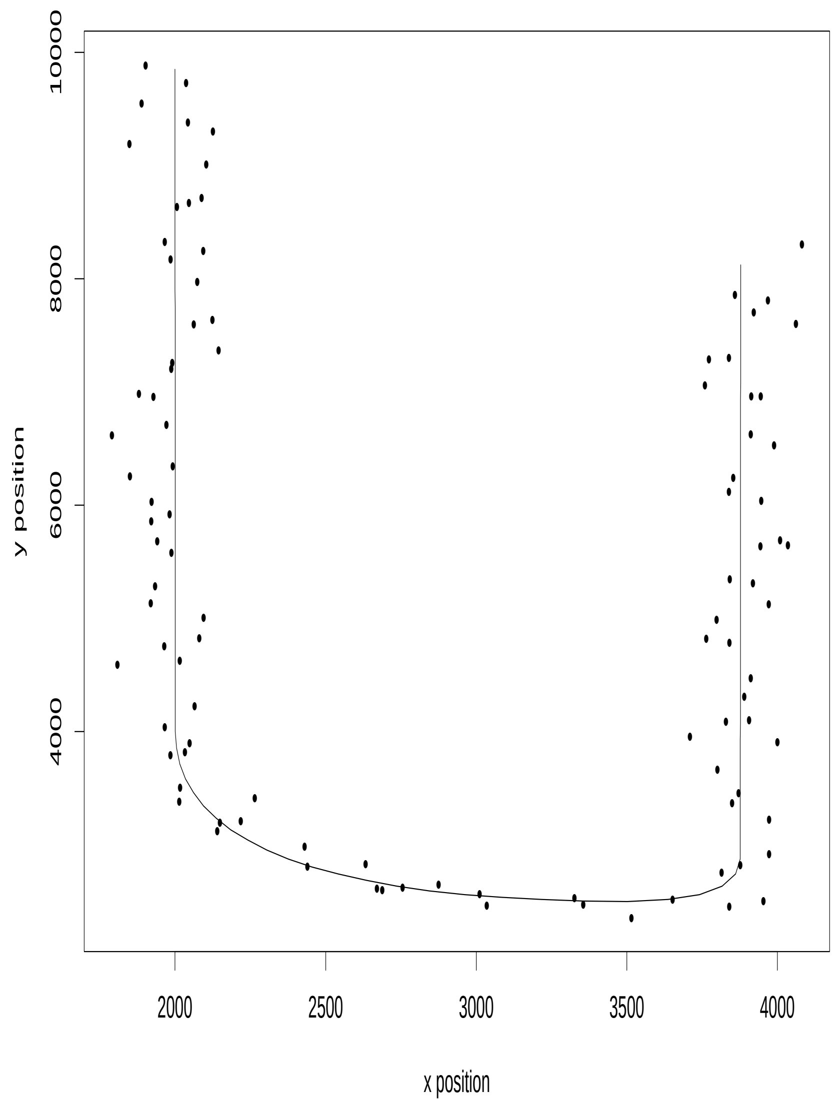
$$
\begin{aligned}
x_t &= Hx_{t-1} + Fu_t + Ww_t \\
y_t &= Gx_t + Vv_t
\end{aligned}
$$

where $w_t \sim N(0, I)$ **and** $v_t \sim N(0, I)$ **are independent.**

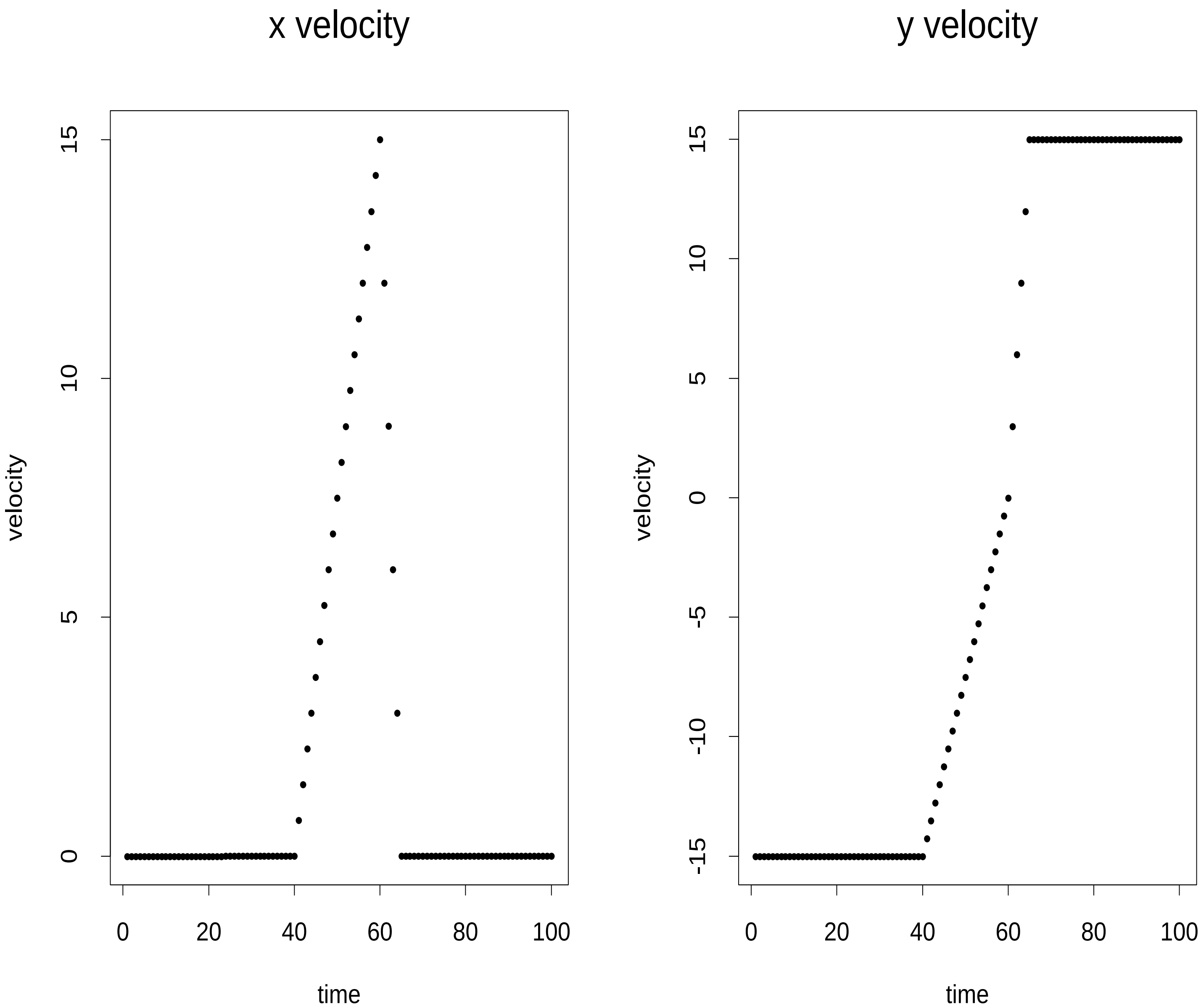
I_t **maneuvering status:**

$I_t = 0$**, no maneuvering,** $u_t = 0$
$I_t = 1$**, slow maneuvering,** $u_t \sim N(0, s_1^2 I)$
$I_t = 2$**, fast maneuvering,** $u_t \sim N(0, s_2^2 I)$

With known transition matrix $P = P(I_{t+1} \mid I_t)$**.**



y position
10000
8000
6000
4000
2000
2500
3000
3500
4000
x position


x velocity
velocity
time
0
5
10
15
0
20
40
60
80
100
y velocity
velocity
time
-15
-10
-5
0
5
10
15
0
20
40
60
80
100

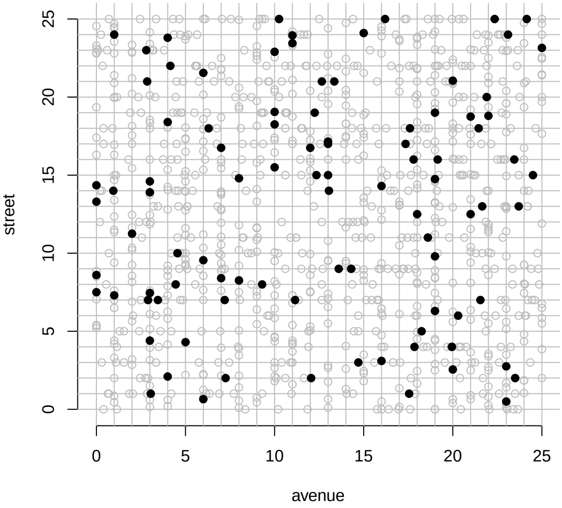
Example 4: Mobile network for nuclear material surveilliance

- Detection of nuclear material in large cities
  - radiation dispersion devices (dirty bombs)
  - enriched uranium and weapon grade plutonium (nuclear weapons)
- Mobile sensor network
  - inexpensive sensor with GPS mounted on taxi cabs and police vehicles
  - command center receives signals and does the analysis in real time

Source and sensor specification

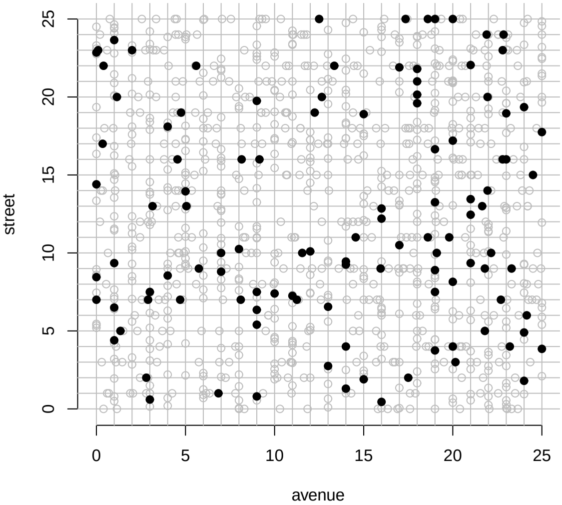
- **Signal source intensity function** $z(r) = c/r^2$
  - r **distance to source.**
  - c **related to the total energy of the source.**
- **Multi-source intensity:**
  - **Same spectra:** $z(r) = \sum c_i/r_i^2$
  - **Different spectra:** $z(r) = \max\{c_i/r_i^2\}$
- **Sensor: True signal** $S = 1$ **if** $z(r) \geq d$
  - **Source visible range:** $r^2 < c/d$
- **Sensor error:**
  - **Sensor False Positive rate** $\gamma = P(D = 1 \mid S = 0)$
  - **Sensor False Negative rate** $\delta = P(D = 0 \mid S = 1)$

Data received at 8 consecutive times ...



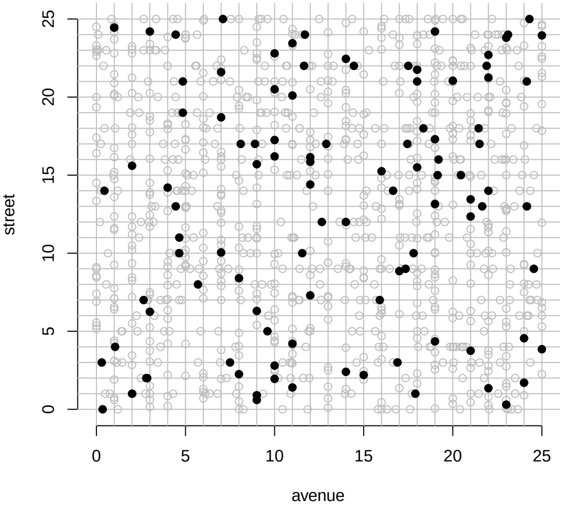
t=1

Data received at 8 consecutive times ...



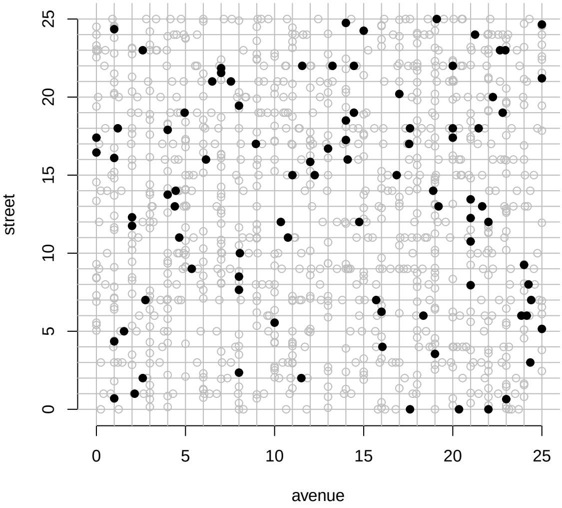
t=2

Data received at 8 consecutive times ...



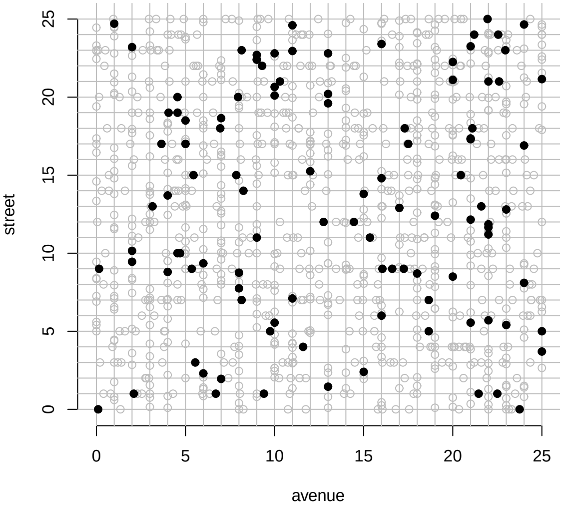
t=3

Data received at 8 consecutive times ...



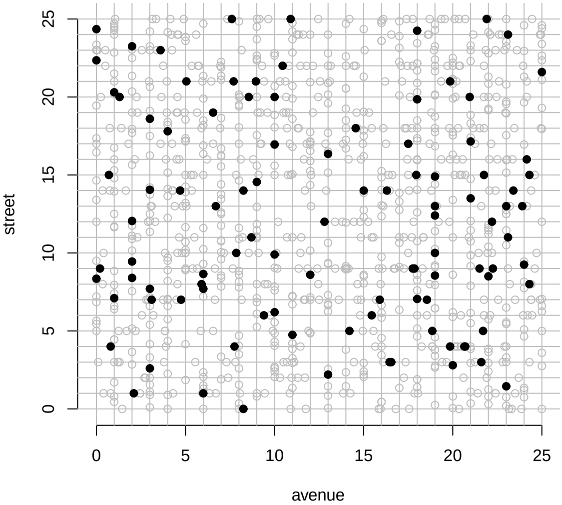
t=4

Data received at 8 consecutive times ...



t=5

Data received at 8 consecutive times ...



t=6

Nuclear surveillance with mobile sensor network using a dynamic model approach

Aude GRELAUD

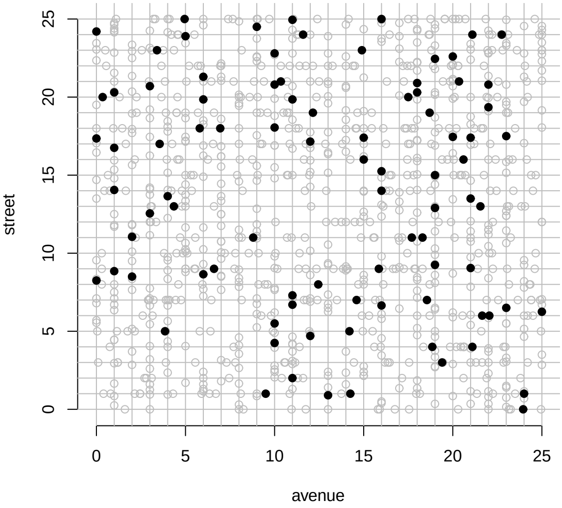
Introduction

Data and model

Estimation by particle filtering

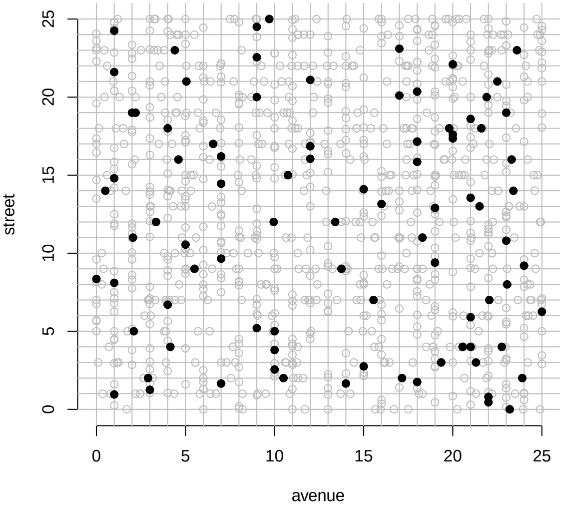
Perspectives

Data received at 8 consecutive times ...



t=7

Data received at 8 consecutive times ...



t=8

State Space Model

State Equation:

- **Number of sources k follow a Markov chain (a state variable)**
- **Source location and source power as state variables**
  - **Motion model for the source location (with map)**
  - **visible range modeled as random walk**

Observation Equation:

- **observations: $(y_i(t), D_i(t))$ (for every sensor)**
- **let $d_i(t) = 1$ if $||y_i(t) - s(t)|| < c^*$ where c^* is sensor range**
- **likelihood**

$$P(D_i = 1) = (1-\delta)^{d_i(t)}\delta^{1-d_i(t)}\gamma^{1-d_t(t)}(1-\gamma)^{d_i(t)}$$

Example 5: All Source Positioning and Navigation System

- *enable[s] low cost, robust, and seamless navigation solutions*
- *for military users on any operational platform and in any environment*,
- *with or without GPS.*

Objectives:

- **rapid integration and reconfiguration of any combination of sensors.**
- **using** *Images, Maps, Signal databases, Location lookup tables (with landmarks, ranging signal sources, etc.)*
- **with platforms including** *Dismounts, UAVs (all sizes), Submersibles, Wheeled vehicles, Tracked Vehicles, Aircraft, Small robots*
- **under environments:** *Underwater, Underground, Jungle, Forest canopy, Suburban, Urban canyon, Building interior, Open field*

Computing resources

- **Portable:**
  - small, light and limited battery power
  - single target
- **Vehicle mounted**
  - medium size, good power source.
  - a small group of targets, close to each other
- **Central command**
  - unlimited computing source
  - many many targets
  - new source generation

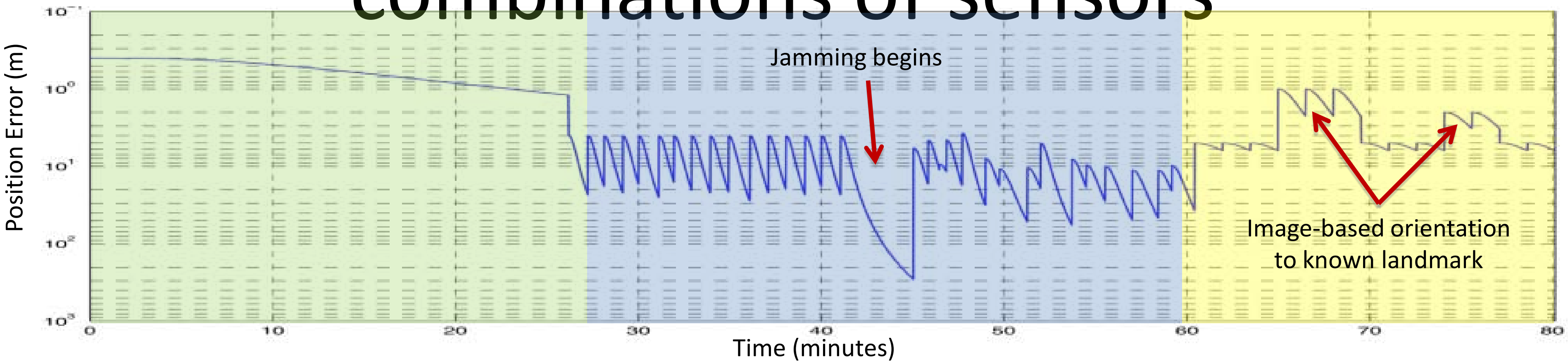
Navigational sensors:

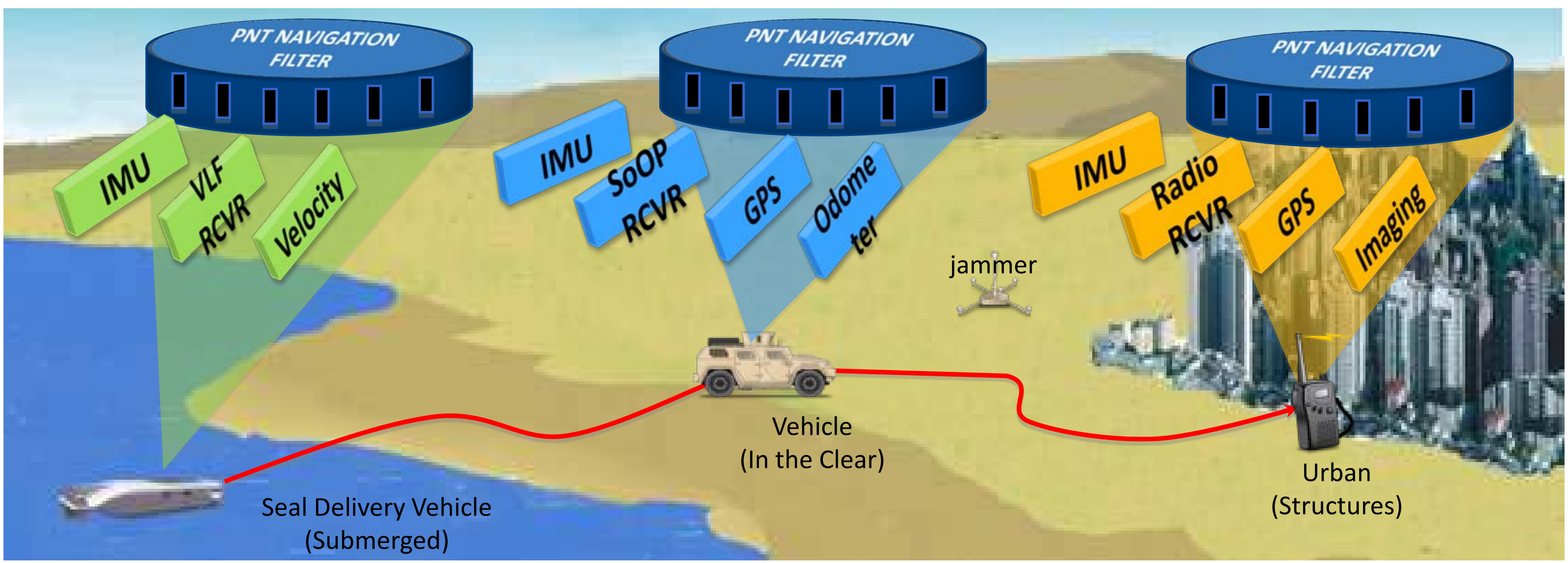
- **GPS – signals from multiple satellites**
  - **can be blocked by overcast, weak in the city**
  - **not available indoor, in tunnels etc**
  - **can be jammed**
- **Wi-Fi/RF receivers**
  - **measures distance to known locations**
  - **using signal power**
- **Inertial measurement unit (IMU)**
  - **a combination of accelerometers and gyroscopes**
  - **measures acceleration related to its own frame**
  - **measures rotational acceleration of the unit (frame)**
  - **often in combination with gravity sensors, barometer, and magnetic compass**

- **Range Finder**
  - **Measure distance to a known landmark**
  - **with the assistance of a map**
- **Millimeter-Wave Radar (based on radio-frequency technology)**
  - **distance and relative velocity**
  - **weather independent**
- **Star tracker, acoustic sensors, GyroCompass, inclinometer, ...**




Example 2: Quickly integrate different combinations of sensors





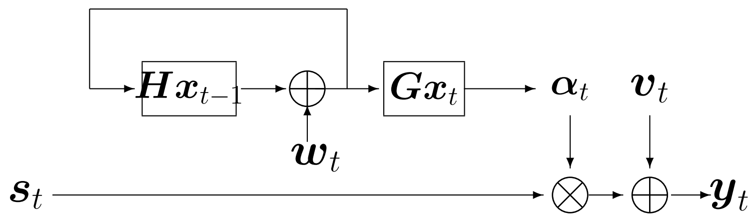
State Space Models

- **State:**
  - **3-D position, velocity and acceleration**
  - **other sensor related states (IMU facing angles)**
  - **Motion states: stationary, walking, on vehicle ...**
  - **Environmental states: open field, highway, jungle ...**
- **Observations: sensor readings**

Special features:

- **Plug-and-Play: Sensor in-and-out, hence changing system configuations**
- **Limited computational power – approximation, sensor selection and adpatation**
- **Sensor network – a group of devices moving together.**

Example 6: Digital Signal Extraction in Fading Channels

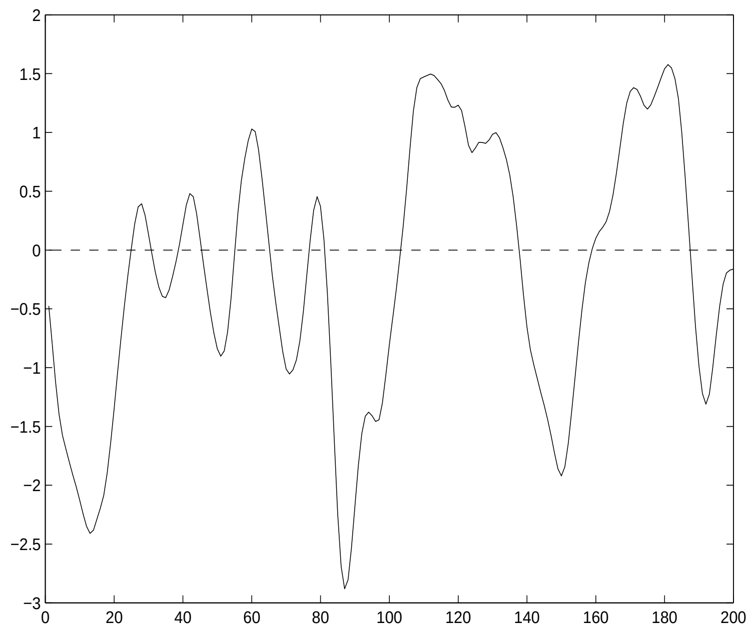


- **State Equations:** $\begin{cases} x_t = Hx_{t-1} + w_t \\ \alpha_t = Gx_t \\ s_t \sim p(\cdot \mid s_{t-1}) \end{cases}$

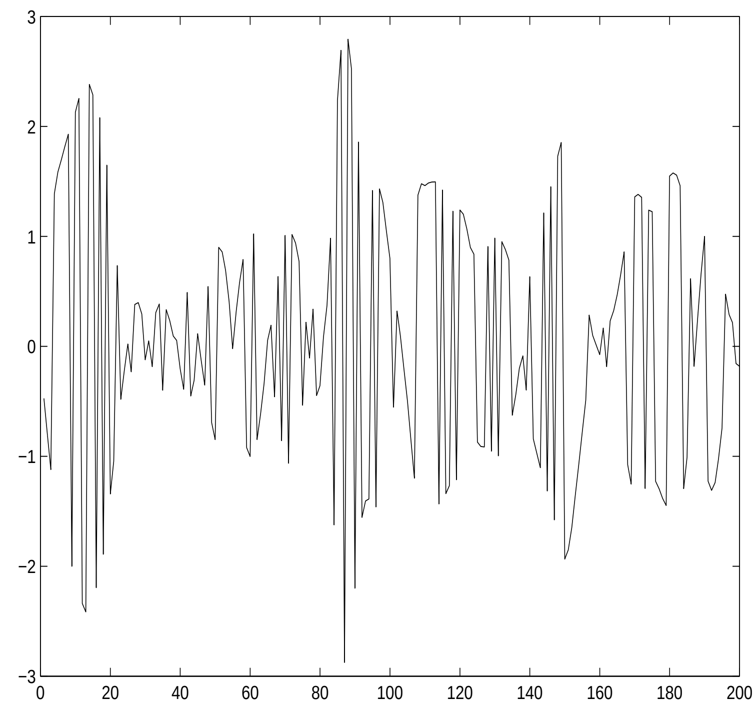
- **Observation equation:** $y_t = \alpha_t s_t + v_t$

- $\alpha_t = Gx_t$: **Butterworth filter of order** $r = 3$ **i.e. ARMA(3,3)**
  Cutoff frequency 0.1

- **Noise:** **(1)** $v_t \sim N(0, \sigma^2)$ **(2)** $v_t \sim (1-\alpha)N(0, \sigma_1^2) + \alpha N(0, \sigma_2^2)$



2
1.5
1
0.5
0
−0.5
−1
−1.5
−2
−2.5
−3
0
20
40
60
80
100
120
140
160
180
200



3
2
1
0
−1
−2
−3
0
20
40
60
80
100
120
140
160
180
200

Phase Ambiguity: $p(\boldsymbol{\alpha}_t, \boldsymbol{s}_t \mid \boldsymbol{y}_t) = p(-\boldsymbol{\alpha}_t, -\boldsymbol{s}_t \mid \boldsymbol{y}_t)$

Differential coding:

Information sequence: $s_1, \ldots, s_t$**.**

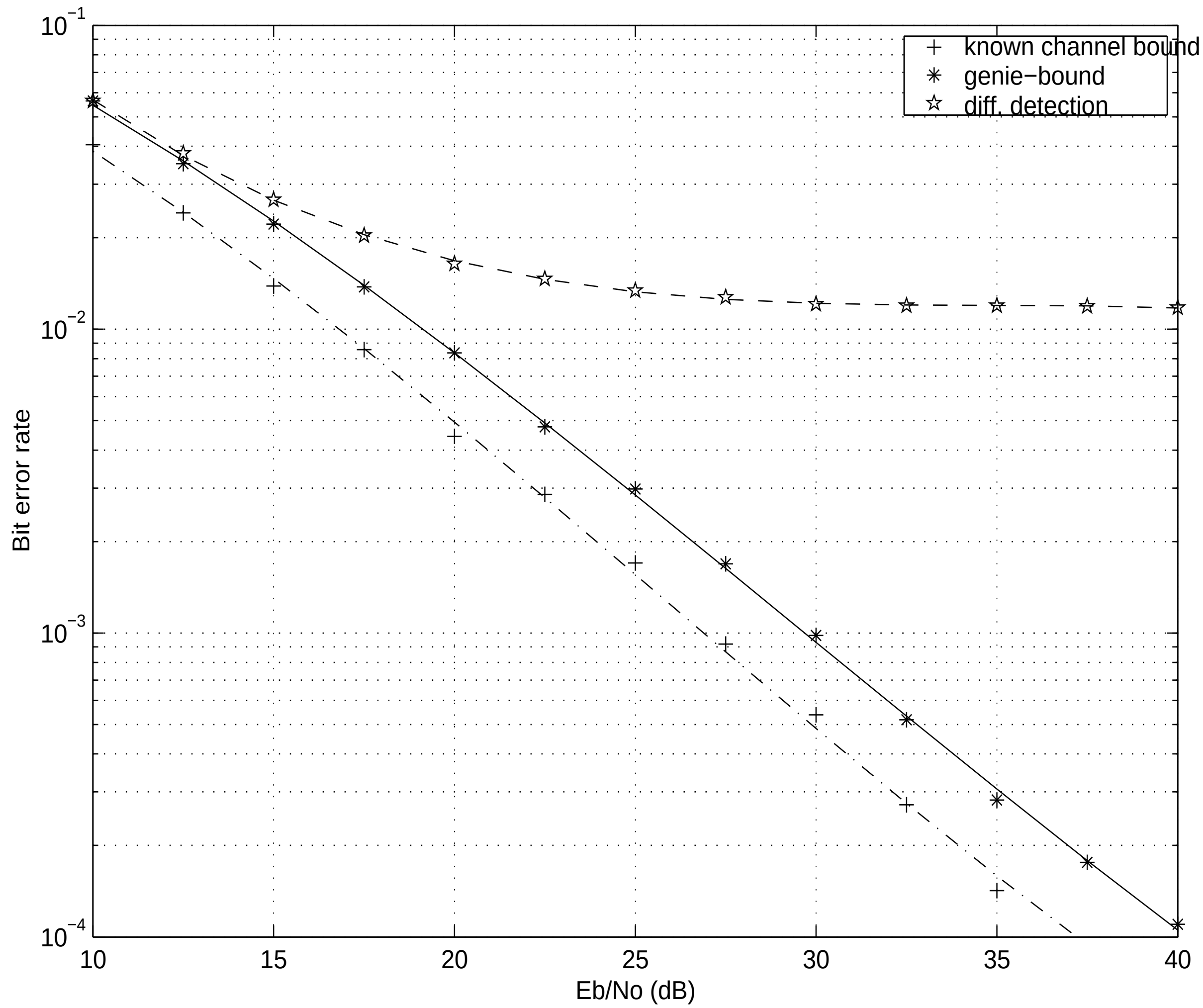
Transmitted sequence: $s_1^*, \ldots, s_t^*$**, s.t** $s_{t-1}^* s_t^* = s_t$**,** $s_1^* = s_1$**.**

Differential detector:

$$\hat{s}_t = sign(y_t y_{t-1}) = sign(\alpha_t \alpha_{t-1} s_t + \alpha_t s_t^* e_{t-1} + \alpha_{t-1} s_{t-1}^* e_t + e_{t-1} e_t)$$

Assumption: α_t **changing slowly.**

Error floor: the frequency that α_t **changes the sign.**



known channel bound
genie−bound
diff. detection
Bit error rate
Eb/No (dB)
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10
15
20
25
30
35
40

Example 7: Blind Equalization

$$\begin{array}{ccccccc} & & & & e_t & & \\ & & & & \downarrow & & \\ s_t(\textbf{digital}) & \longrightarrow & \boxed{\sum_{i=1}^{q} \theta_i s_{t-i}} & \longrightarrow & \bigoplus & \longrightarrow & y_t \end{array}$$

Objective: On-line estimation of s_t using the observed output y_t without knowing the system coefficients θ_i.

$\boldsymbol{\theta}_t = (\theta_{t1}, \ldots, \theta_{tq})$ **and** $x_t = (s_t, s_{t-1}, \ldots, s_{t-q})'$

State Equation:

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1}$$

$$x_t = \begin{pmatrix} 0 & 1 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & 1 \\ 0 & 0 & \cdots & 0 \end{pmatrix} x_{t-1} + \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} s_t$$

Observation equation:

$$y_t = \boldsymbol{\theta}_t x_t + \varepsilon_t$$

Example 8: Stochastic Volatility Models:

Y_t **stock returns (zero mean).** α_t **volatility**

State Equation: $\alpha_t = c + \phi\alpha_{t-1} + \eta_t$

Observation Equation: $Y_t \sim N(0, \exp(\alpha_t))$

where $\eta_t \sim N(0, \sigma^2)$**, and** $c > 0$**.**

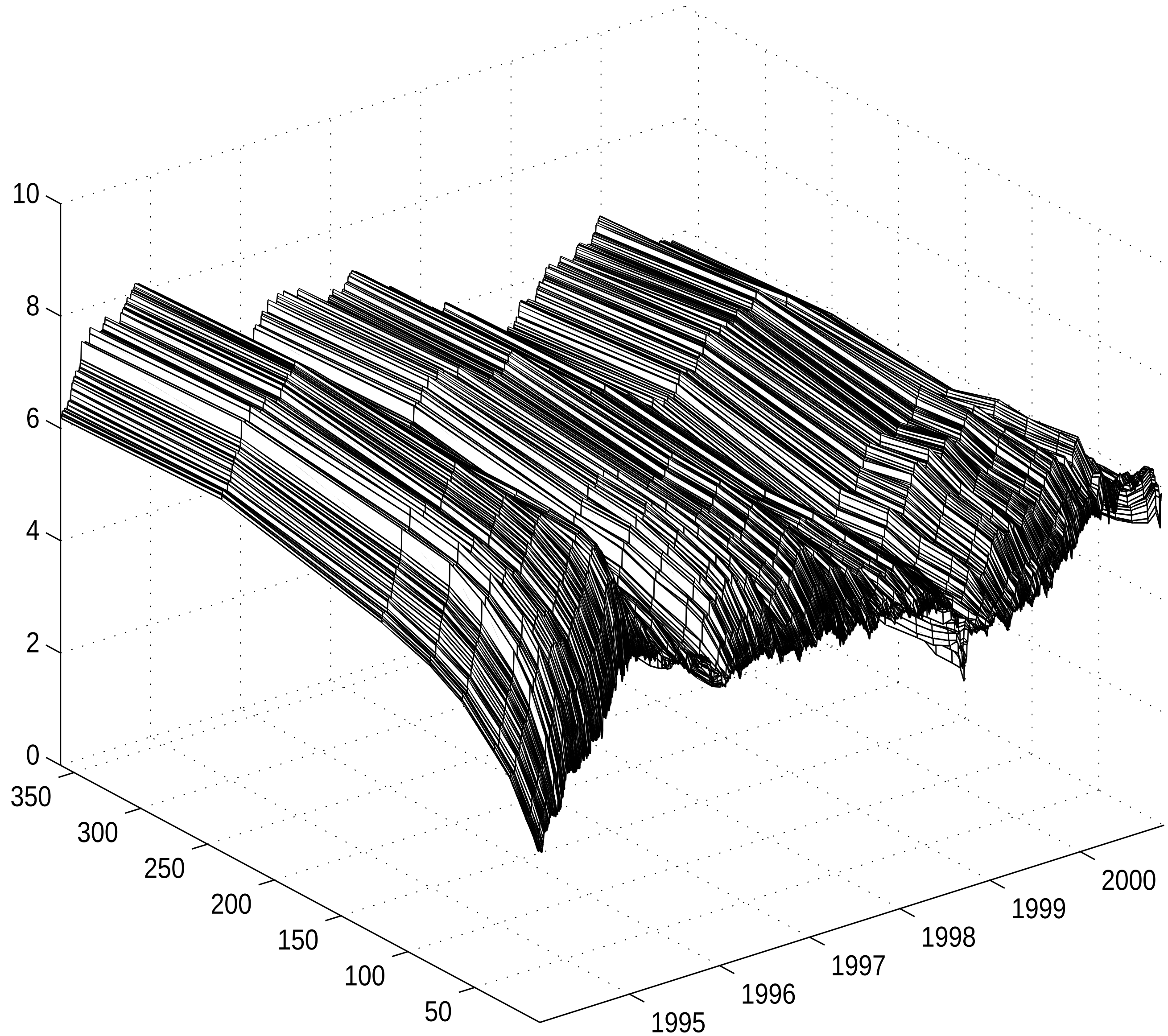
Or

State Equation: $\alpha_t = c + \phi\alpha_{t-1} + \eta_t$

Observation Equation: $log(Y_t^2) = \alpha_t + v_t$

where $v_t = log(e_t^2)$**,** $e_t \sim N(0, 1)$**.**

Example 9: Yield curve (Interest rate) over time



Curve time series $X_t(s)$ driven by dynamic processes

- **Any fixed t,** $\quad X_t(s) = f_t(s;\theta_t) + \varepsilon_t(s), \quad$ **and** $\quad s \in \Omega$
- **The function $f_t(\cdot)$ is known, except (parameter) θ_t.**
- $\varepsilon_t(s)$**: a white noise process defined on Ω with $E(\varepsilon_t(s)) = 0$.**
- θ_t**: a random (driving) process over t.**

- **The dependency between $X_t(s)$ is completely characterized by the parameter process θ_t and the noise process ε_t.**
- **We call $\{\theta_t\}$ the driving process.**
- **In most applications, $X_t(\cdot)$ is only observed at a finite number of locations $\{X_t(s_{ti}), i = 1, \ldots, m_t\}$.**

Finite dimensional driving processes

θ_t follows a parametric ARMA process:

$$
\begin{aligned}
X_t(s_{ti}) &= f(s_{ti}, \theta_t) + \varepsilon_t(s_{ti}), \quad i = 1, \ldots, m_t, \\
\theta_t &= g(\theta_{t-1}, \ldots, \theta_{t-p}, e_t, \ldots, e_{t-q}, \boldsymbol{\gamma}),
\end{aligned}
$$

$$
\begin{array}{ccccccc}
\cdots & & \varepsilon_t(s) & & \varepsilon_{t+1}(s) & & \cdots \\
 & & \downarrow & & \downarrow & & \\
\cdots & & X_t(s) & & X_{t+1}(s) & & \cdots \\
 & & \uparrow & & \uparrow & & \\
\cdots & \longrightarrow & \theta_t & \rightarrow & \theta_{t+1} & \longrightarrow & \cdots
\end{array}
$$

- **A generalized state space model**
- $g(\cdot)$ **is a known function with unknown parameters** $\boldsymbol{\gamma}$ **and** e_t **is a sequence of scalar or vector white noises.**

Probabilistic Dynamic Systems

Definition: A probabilistic dynamic system is abstracted as a sequence of evolving probability distributions $\pi_t(\boldsymbol{x}_t)$.

$\boldsymbol{x}_t$: state variable:

(i) increasing dimension: $\boldsymbol{x}_{t+1} = (\boldsymbol{x}_t, x_{t+1})$

(ii) discharging: $\boldsymbol{x}_t = (\boldsymbol{x}_{t+1}, d_t)$

(iii) no change: $\boldsymbol{x}_{t+1} = \boldsymbol{x}_t$

State space model is a special case of probabilistic dynamic system.

$$\pi_t(\boldsymbol{x}_t) = p(x_1, \ldots, x_t \mid y_1, \ldots, y_t)$$

Example 10: Sequential Bayesian Inference:

$$\boldsymbol{x}_t = \boldsymbol{\theta}$$

and

$$\pi_t(\boldsymbol{x}_t) = p(\boldsymbol{\theta} \mid y_1, \ldots, y_t)$$

Example 11: Bayesian Missing Data

- $z_1, \ldots z_n$ **iid from** $p(z, \theta)$**.**
- $z_i = (y_i, x_i)$**:** y_i **observed** x_i **missing**

Let $\boldsymbol{x}_t = (x_0, x_1, \ldots, x_t)$**.** $x_0 = \theta$**.** $\boldsymbol{y}_t = (y_1, \ldots, y_t)$**.**

Then the dynamic system is $\pi_t(\boldsymbol{x}_t) = p(\boldsymbol{x}_t \mid \boldsymbol{y}_t)$

Note: for fixed θ**, the likelihood can be evaluated**

The Growth Principle

Decompose a complex problem into a sequence of simpler problems,

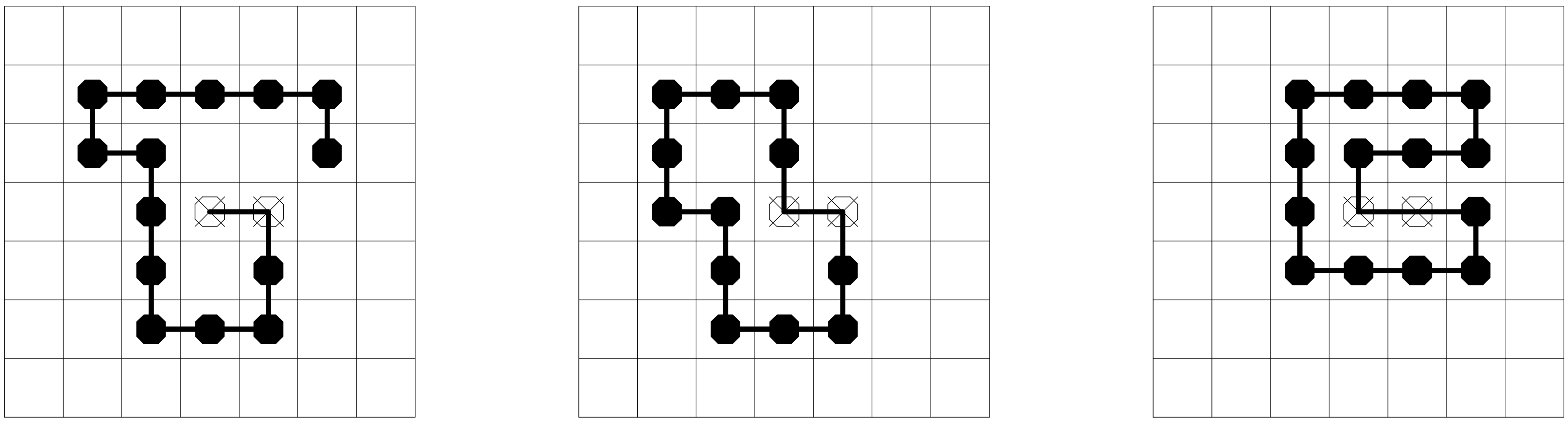
— forming a dynamic system from a fixed dimensional problem

- **Target distribution** $\pi(\boldsymbol{x})$ **where** $\boldsymbol{x} = (x_1, \ldots, x_N)$
- **Let** $\boldsymbol{x}_t = (x_1, \ldots, x_t) = (\boldsymbol{x}_{t-1}, x_t)$
- **Define a sequence of intermediate distributions** $\pi_t(\boldsymbol{x}_t)$**.**
- **Moving from** $\pi_{t-1}(\boldsymbol{x}_{t-1})$ **to** $\pi_t(\boldsymbol{x}_t)$ **is simple.**
- **Moving from** $\pi_{t-1}(\boldsymbol{x}_{t-1})$ **to** $\pi_t(\boldsymbol{x}_t)$ **is smooth.**

$$\int \pi_t(\boldsymbol{x}_{t-1}, x_t) dx_t \approx \pi_{t-1}(\boldsymbol{x}_{t-1})$$

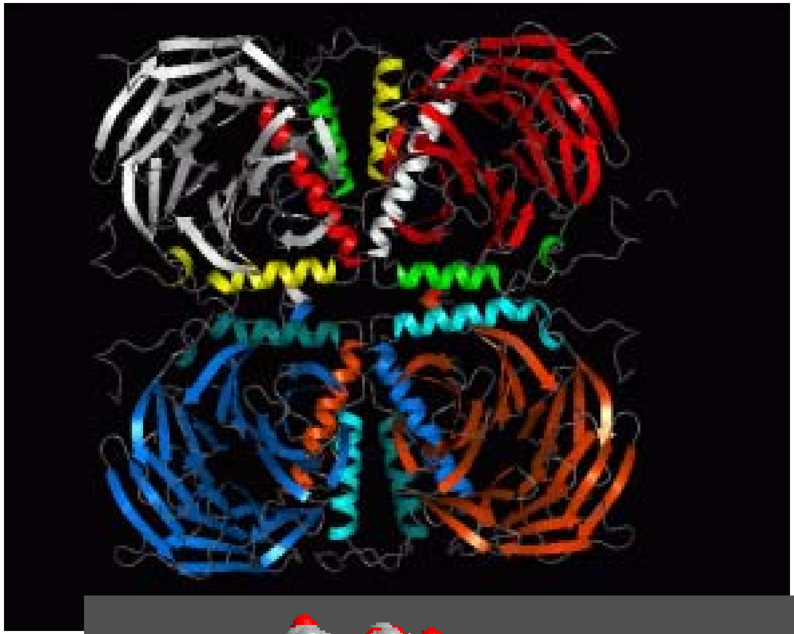
- $\pi_N(\boldsymbol{x}_N) = \pi(\boldsymbol{x})$

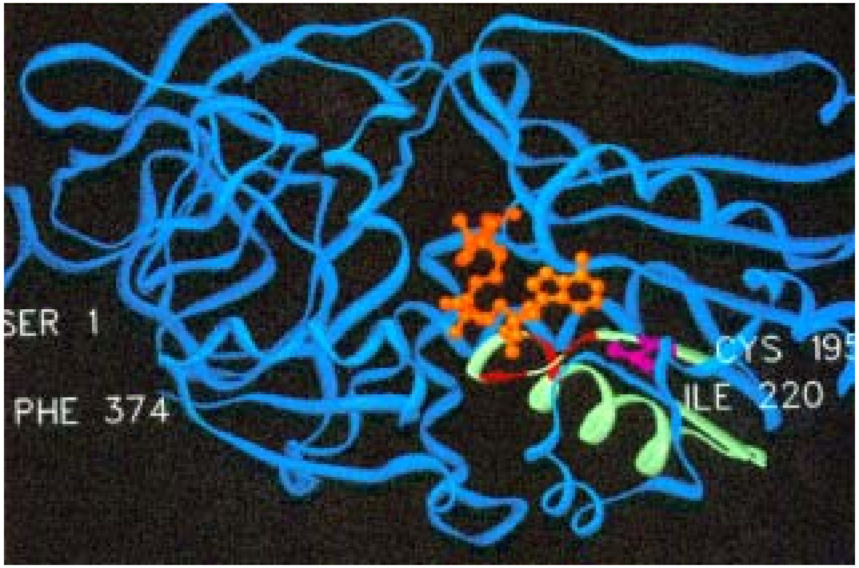
Example 12: Self-avoiding walks (SAW) and Self-avoiding loops (SAL)

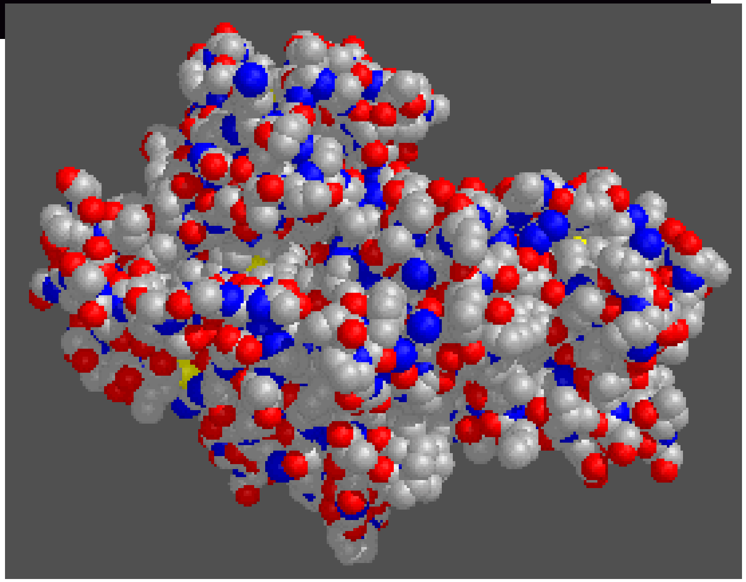


- **How many are there?**
- **What is the average size of enclosed void of SAL(n)?**
- **What is the average number of contacts?**

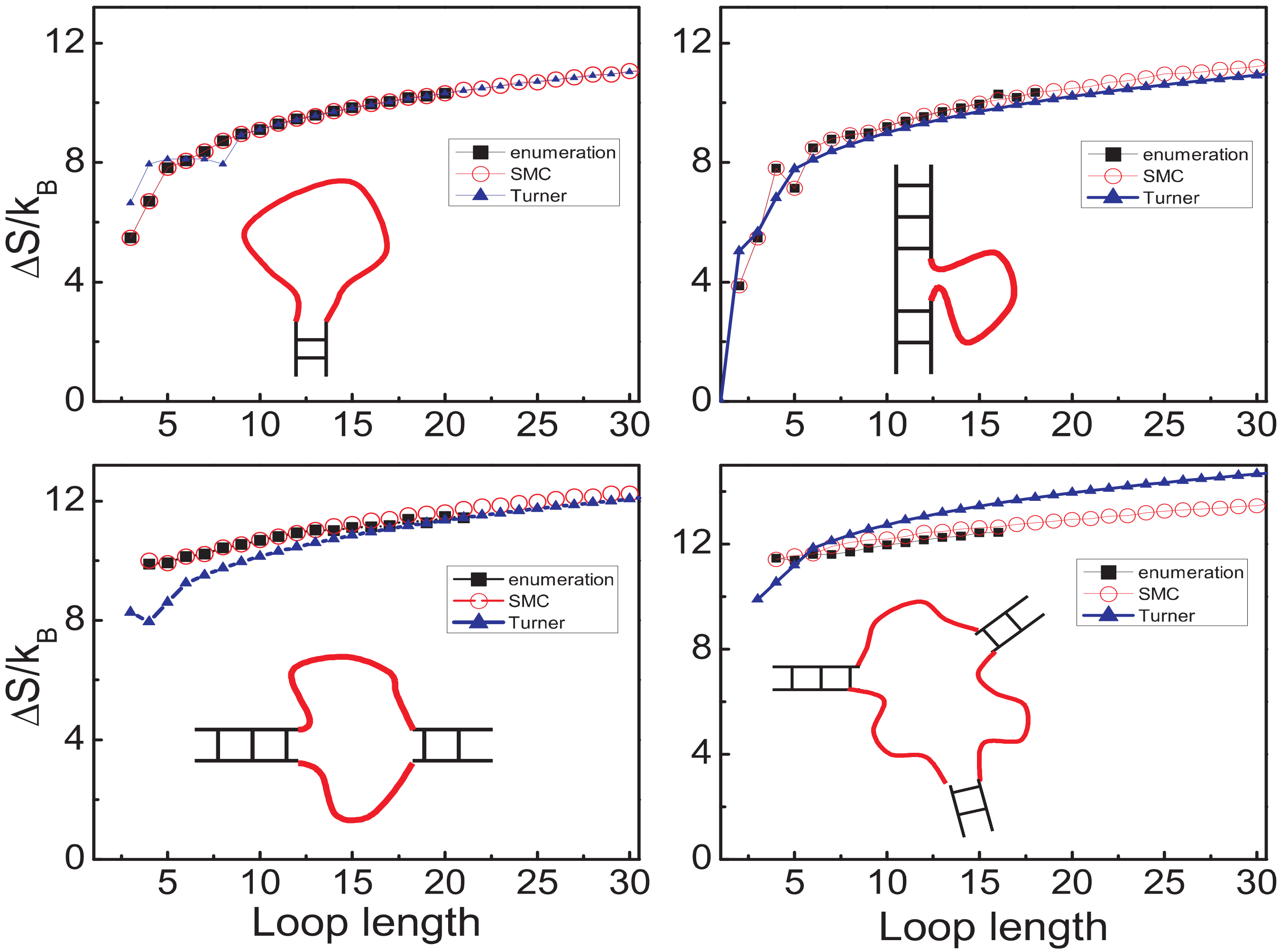
Protein Structures







RNA local structure



Enumeration:

n	SAWs	SALs	void size
14	110,188	4,116	1.40
16	802,075	23,504	2.06
18	5,808,335	137,412	2.82
20	41,889,578	818,210	3.68
22	301,100,754	4,945,292	4.63
24	2,158,326,727	30,255,240	5.68

SALs:

- **Starting at** $(0,0)$ **and** $(1,0)$ **and ending at** $(0,0)$
- **Target distribution: uniform of all SALs of length** n**.**
- **Partial chain** $(x_1,\ldots,x_t)$ **is a SAW.**
- **Strong constraint at the end – shrinking support**

Hence

- **Intermediate distributions**

  $\pi_t(\boldsymbol{x}_t)$**: uniform of all SAWs of length** t **such that**

  $$d(\boldsymbol{x}_t) < n - t \quad \textbf{(support)}$$

  where $d(\boldsymbol{x}_t) = |x_{t,1}| + |x_{t,2}|$

Example 13: Diffusion Bridges

(Multivariate) Diffusion process v_t satisfying SDE:

$$dv_t = b(v_t; \theta)dt + \sigma(v_t; \theta)dB_t$$

where B_t: Brownian Motion. θ: unknown parameters

Observations: $v_i = v_{t_i}$, $i = 0, \ldots, n$, observed at discrete time points $t_0, \ldots, t_n$.

Aims:

- **Estimate the log-likelihood function (conditioned on v_0)**

$$L(\theta) = logp(v_1, \ldots, v_n \mid v_0, \theta) = \sum_{i=1}^{n} logp(v_i \mid v_{i-1}, \theta)$$

- **Obtain the MLE of θ**
- **Estimate $E(h(v[t_0, t_n]))$ (e.g. quantile of the path $v([t_0, t_n])$)**

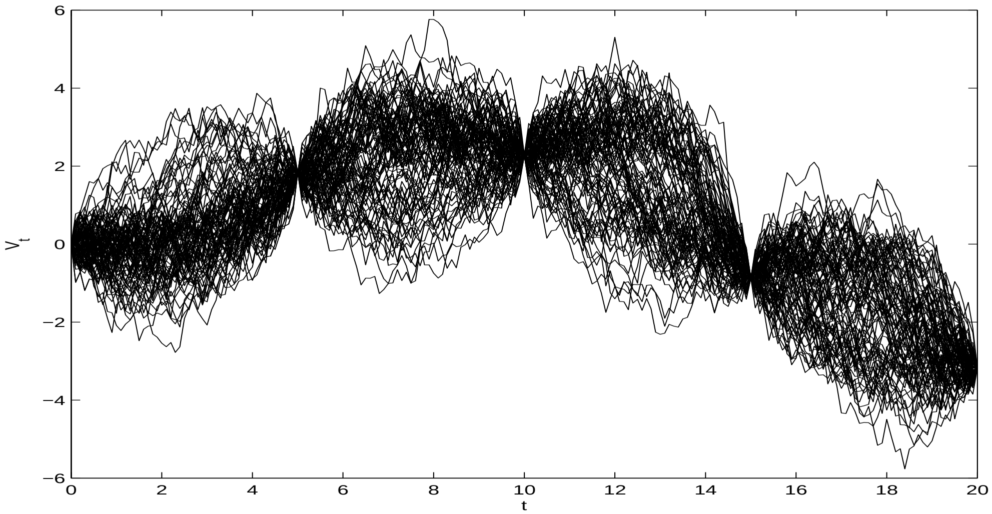
Consider the transition density $p(v_1 \mid v_0, \theta)$.

- **Except for a few cases, no analytical form for $p(v_1 \mid v_0, \theta)$**
- **If $v[t_0, t_1]$ is known or observed completely, then $p(v[t_0, t_1] \mid \theta)$ can be found since**

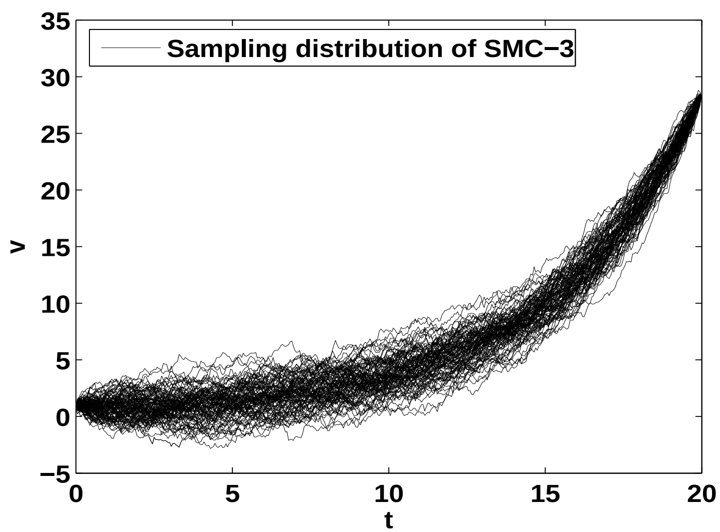
$$v(t_1) - v(t_0) = \int_{t_0}^{t_1} b(v(t), \theta)dt + \int_{t_0}^{t_1} \sigma(v(t), \theta)dB_t.$$

  The integration can be obtained analytically or numerically.

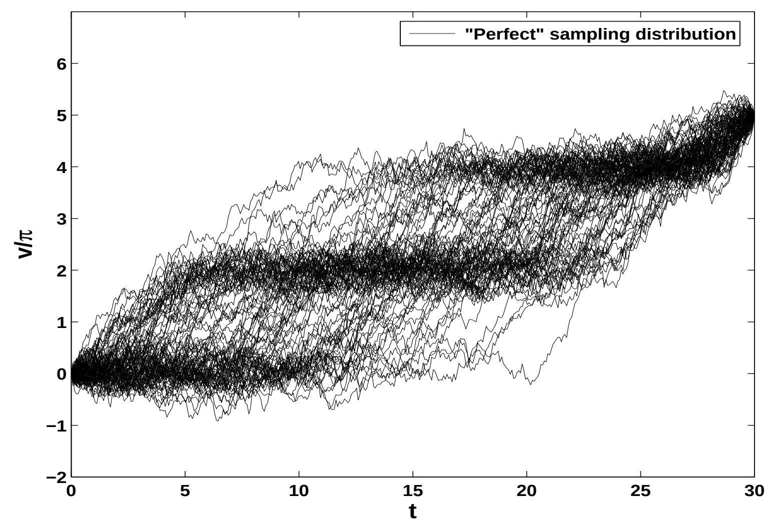
- **But $v(t_0, t_1)$ is missing.**
- **Monte Carlo: simulate the diffusion bridge $v(t_0, t_1)$ following $p(v(t_0, t_1) \mid v_0, v_1, \theta)$**



v_t
t



Sampling distribution of SMC−3
v
t



"Perfect" sampling distribution
v/π
t

- **Consider one segment $[V_0, V_1]$ (due to Markovian property).**
- **Euler-Maruyama approximation (Gaussian system)**

$$V_t \approx V_{t-\Delta t} + b(V_{t-\Delta t}; \theta)\Delta t + \sigma(V_{t-\Delta t}; \theta)(W_t - W_{t-\Delta t})$$

- **Divide $[0, 1]$ into small intervals $0 < s_1 < \ldots < s_m < 1$.**
- $V_{s_1}, \ldots, V_{s_m}$ **missing.**
- **Target distribution $\pi(V_0, V_{s_1}, \ldots, V_{s_m}, V_1 \mid \theta)$ can be easily evaluated.**
- **The intermediate distributions: $\pi_t(V_0, V_{s_1}, \ldots, V_{s_t}, V_1 \mid \theta)$ defined by the same linearized system, with special treatment of the last *large* step from V_{s_t} to V_1.**

Fixed dimension with augmentation

Augmentation (expanding dimensions): [Moral et al (2006)]

- **Target distribution $\pi(x)$, for $x \in \Omega$**
- **Let $\boldsymbol{x}_t = (x_1, \ldots, x_t) \in \Omega^t$ where $x_i \in \Omega$.**
- **Construct a sequence of intermediate distributions $\pi_t(\boldsymbol{x}_t)$.**
- **The marginal distribution of $\pi_n(x_n) = \pi(x_n)$.**

Note:

- **This is very similar to MCMC.**
- **Samples are *moved* within the same space with the trial distribution $g_t(x_t \mid \boldsymbol{x}_{t-1})$.**
- **Eventually the final marginal dist. is the target dist.**
- **Finite steps of movements, but with weights**

For example:

- **Design a sequence of** *marginal* **intermediate distributions** $\pi_t(x_t)$**, with** $\pi_n(x_n) = \pi(x)$**.**

  For example:

  - $\pi_t(x) \propto \pi(x)^{\phi_t}\mu_1(x)^{1-\phi_t}$ **with** $0 \leq \phi_1 < \ldots < \phi_n = 1$**.**
  - $\pi_t(x) \propto \pi(x \mid y_1, \ldots, y_t)$**: sequential new observations**
  - $\pi_t(x) \propto \pi(x)^{\phi_t}$ **with** $\phi_t \to \infty$ **(simulated annealing)**

- **Let**

$$\boldsymbol{\pi}_t(x_1, \ldots, x_t) = \pi_t(x_t) \prod_{k=1}^{t-1} L_k(x_{k+1}, x_k)$$

  where $L_k(x, y)$ **is a Markov kernel with** π_k **as the invariant distribution.**

Other Applications:

- **Target tracking (Gordon et al 1993, Avitzour 1995, Bølviken et al 1997, McGinnity and Irwin, 2001, Irwin et al 2002. Salmond and Gordon, 2001, Arulampalam et al 2002, Orton and Fitzgerald, 2002, Hueet al, 2002, Gustafsson et al 2002)**
- **Target recognition (Srivastava et al 2001).**
- **Blind equalization (Liu and Chen 1995)**
- **Speech recognition (Rabiner 1989)**
- **Computer vision (Isard and Blake 1996, 1998, 2001, Torma and Szepesvari, 2001)**
- **Mobile robot localization (Dellaert et al 1999, Fox et al 1999, 2001)**
- **Freeway traffic vision (for vehicle control) (Huang et al 1994)**

- **DNA sequence analysis (Churchill 1989)**
- **Stochastic volatility model (Pitt and Shephard 1997, Barndorff-Nielsen and Shephard, 2002)**
- **Expert systems (Spiegelhalter et al 1990, Kong et al 1994, Berzuini et al. 1997)**
- **Switching (auto)regression models (Kaufmann, 2002)**
- **Dynamic Bayesian networks (Koller and Lerner, 2001, Murphy and Russell, 2001)**
- **On-line control of industrial production (Marrs, 2001)**
- **Combinatorial optimizations (Wong and Liang 1997)**
- **Wireless communications (Chen et al 2000, Wang et al 2000)**

- **Signal processing (Djuric, 2001, Wang et al 2002)**
- **Audio signal enhancing (Fong et al, 2002)**
- **Data network analysis (Coates and Nowak, 2002)**
- **Chain polymer (Liang et al 2002, Liu et al 2002, Zhang et al 2003, Zhang et al 2004)**
- **Counting 0-1 tables (Liu 2001)**
- **Neural networks (Andriew et al 1999, de Freitas et al. 2000)**