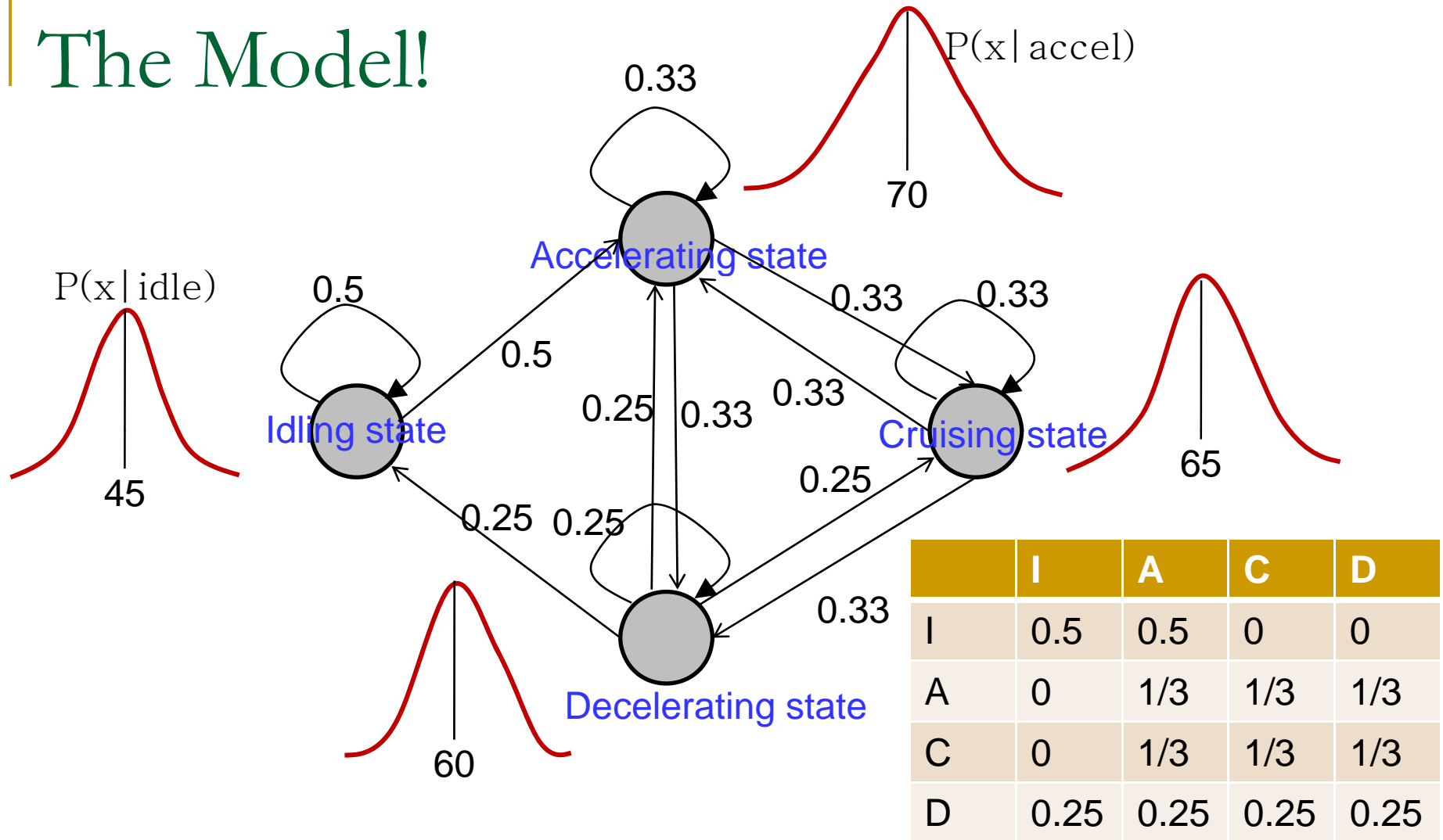# Prediction and Estimation, Part II

Class 27. 23 Nov 2010
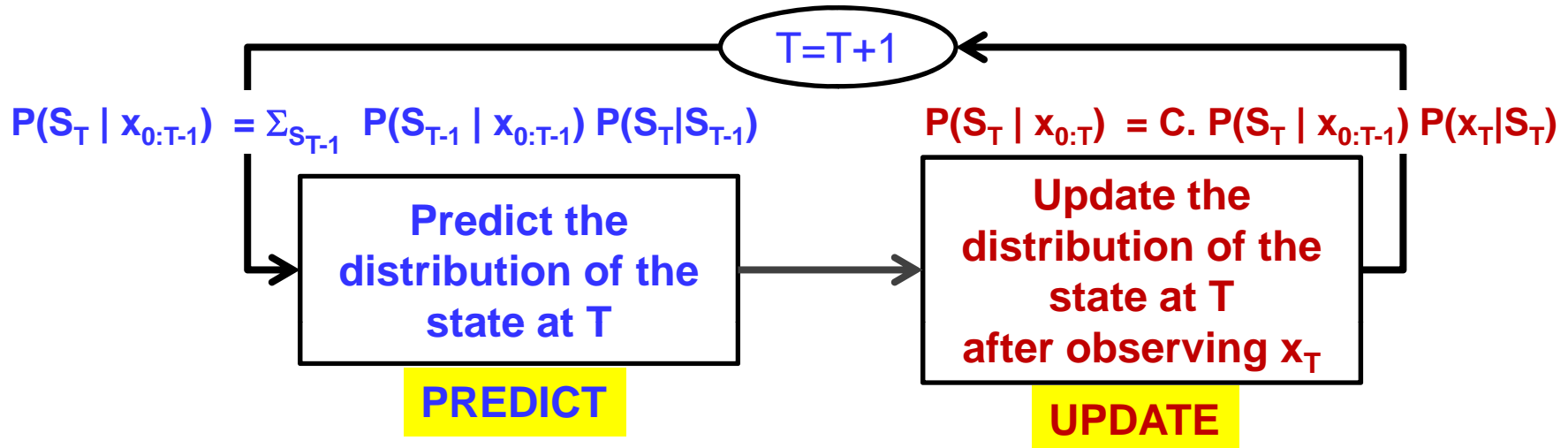
Recap: An automotive example

- Determine automatically, by only *listening* to a running automobile, if it is:
  - Idling; or
  - Travelling at constant velocity; or
  - Accelerating; or
  - Decelerating
- Assume (for illustration) that we only record energy level (SPL) in the sound
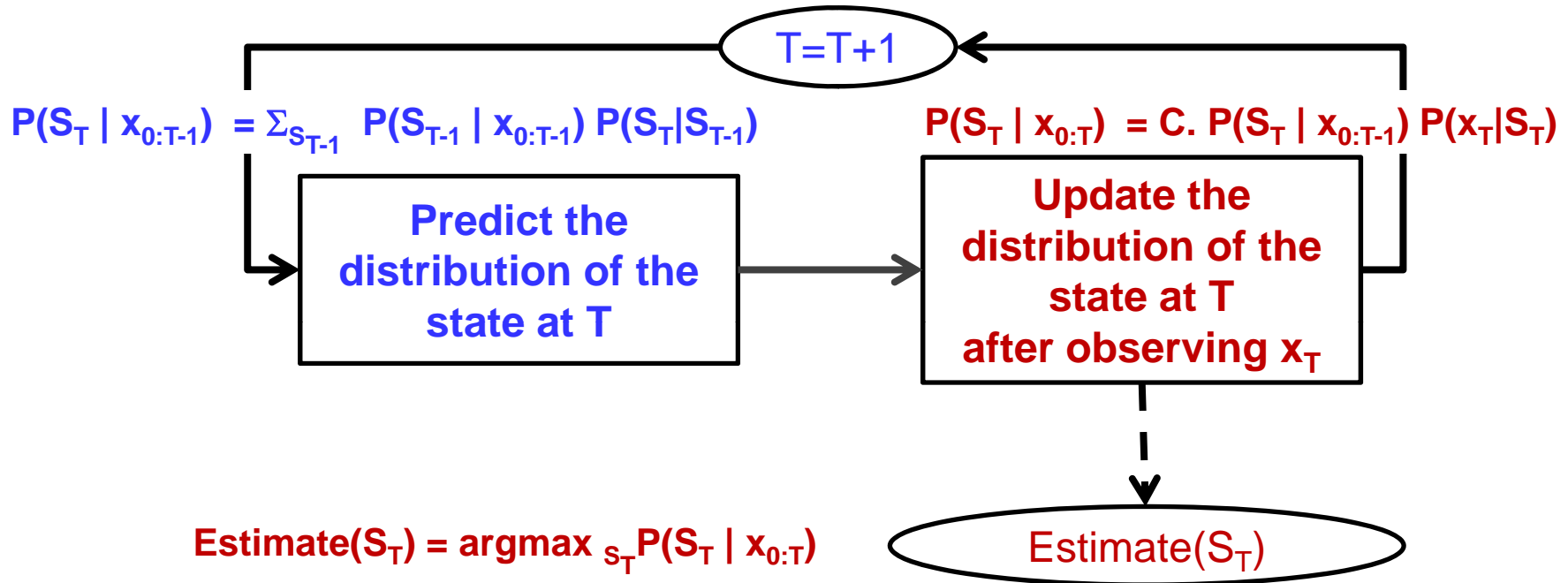 - The SPL is measured once per second

The Model!

- **The state-space model**
  - Assuming all transitions from a state are equally probable

	I	A	C	D
I	0.5	0.5	0	0
A	0	1/3	1/3	1/3
C	0	1/3	1/3	1/3
D	0.25	0.25	0.25	0.25

Overall procedure

$P(S_T \mid x_{0:T-1}) = \Sigma_{S_{T-1}} \ P(S_{T-1} \mid x_{0:T-1}) \ P(S_T \mid S_{T-1})$ $P(S_T \mid x_{0:T}) = C. \ P(S_T \mid x_{0:T-1}) \ P(x_T \mid S_T)$

$T=T+1$

Predict the distribution of the state at T

PREDICT

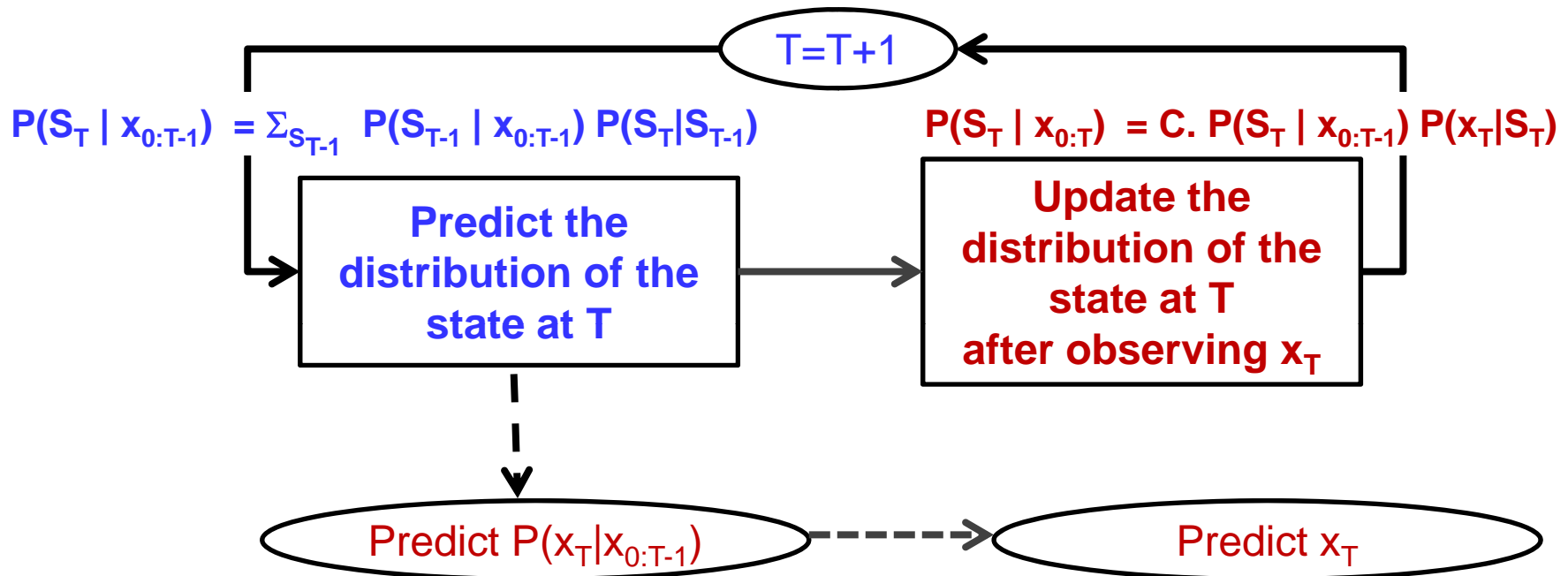**Update the distribution of the state at T after observing x_T**

UPDATE

- At T−0 the predicted state distribution is the initial state probability

- At each time T, the current estimate of the distribution over states considers *all* observations x_0 ... x_T
    - A natural outcome of the Markov nature of the model

- The prediction+update is identical to the forward computation for HMMs to within a normalizing constant

Estimating the *state*

$P(S_T \mid x_{0:T-1}) = \Sigma_{S_{T-1}} \, P(S_{T-1} \mid x_{0:T-1}) \, P(S_T|S_{T-1})$

$P(S_T \mid x_{0:T}) = C. \, P(S_T \mid x_{0:T-1}) \, P(x_T|S_T)$

T=T+1

Predict the distribution of the state at T

Update the distribution of the state at T after observing x_T

Estimate(S_T) = argmax $_{S_T}$P($S_T \mid x_{0:T}$)

Estimate(S_T)

- ## The state is estimated from the updated distribution

  - ## The updated distribution is propagated into time, not the state

Predicting the *next observation*

$P(S_T \mid x_{0:T-1}) = \Sigma_{S_{T-1}} P(S_{T-1} \mid x_{0:T-1}) P(S_T \mid S_{T-1})$

$P(S_T \mid x_{0:T}) = C. P(S_T \mid x_{0:T-1}) P(x_T \mid S_T)$

T=T+1

Predict the distribution of the state at T

Update the distribution of the state at T after observing x_T

Predict $P(x_T \mid x_{0:T-1})$

Predict x_T

- The probability distribution for the observations at the next time is a mixture:

  - $P(x_T \mid x_{0:T-1}) = \Sigma_{S_T} P(x_T \mid S_T) P(S_T \mid x_{0:T-1})$

- The actual observation can be predicted from $P(x_T \mid x_{0:T-1})$

Continuous state system

$$s_t = f(s_{t-1}, \varepsilon_t)$$

$$o_t = g(s_t, \gamma_t)$$

- The state is a continuous valued parameter that is not directly seen

  - The state is the position of navlab or the star

- The observations are dependent on the state and are the only way of knowing about the state

  - Sensor readings (for navlab) or recorded image (for the telescope)

Discrete vs. Continuous State Systems
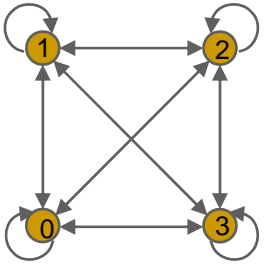
$$s_t = f(s_{t-1}, \varepsilon_t)$$

$$o_t = g(s_t, \gamma_t)$$

Prediction at time t:

$$P(s_t \mid O_{0:t-1}) = \sum_{s_{t-1}} P(s_{t-1} \mid O_{0:t-1}) P(s_t \mid s_{t-1})$$

$$P(s_t \mid O_{0:t-1}) = \int_{-\infty}^{\infty} P(s_{t-1} \mid O_{0:t-1}) P(s_t \mid s_{t-1}) ds_{t-1}$$

Update after O_t:

$$P(s_t \mid O_{0:t}) = CP(s_t \mid O_{0:t-1}) P(O_t \mid s_t)$$

$$P(s_t \mid O_{0:t}) = CP(s_t \mid O_{0:t-1}) P(O_t \mid s_t)$$

Special case: Linear Gaussian model

$$s_t = A_t s_{t-1} + \varepsilon_t$$

$$o_t = B_t s_t + \gamma_t$$

$$P(\varepsilon) = \frac{1}{\sqrt{(2\pi)^d |\Theta_\varepsilon|}} \exp\left(-0.5(\varepsilon - \mu_\varepsilon)^T \Theta_\varepsilon^{-1}(\varepsilon - \mu_\varepsilon)\right)$$

$$P(\gamma) = \frac{1}{\sqrt{(2\pi)^d |\Theta_\gamma|}} \exp\left(-0.5(\gamma - \mu_\gamma)^T \Theta_\gamma^{-1}(\gamma - \mu_\gamma)\right)$$

- A *linear* state dynamics equation
  - Probability of state driving term ε is Gaussian
  - Sometimes viewed as a driving term μ_ε and additive zero-mean noise

- A *linear* observation equation
  - Probability of observation noise γ is Gaussian

- A_t, B_t and Gaussian parameters assumed known
  - May vary with time

The Linear Gaussian model (KF)

$$P_0(s) = Gaussian(s; \bar{s}, R)$$

$$P(s_t \mid s_{t-1}) = Gaussian(s_t; \mu_\varepsilon + A_t s_{t-1}, \Theta_\varepsilon)$$

$$P(o_t \mid s_t) = Gaussian(o_t; B_t s_t, \Theta_\gamma)$$

$$P(s_t \mid o_{0:t-1}) = Gaussian\left(s; \bar{s}_t, R_t\right)$$

$$\bar{s}_t = \mu_\varepsilon + A_t \hat{s}_{t-1}$$

$$R_t = \Theta_\varepsilon + A_t \hat{R}_{t-1} A_t^T$$

$$\updownarrow$$

$$P(s_t \mid o_{0:t}) = Gaussian\left(s; \hat{s}_t, \hat{R}_t\right)$$

$$\hat{s}_t = \bar{s}_t + R_t B_t^T (B_t R_t B_t^T + \Theta_\gamma)^{-1} \left(o - B_t \bar{s}_t\right)$$

$$\hat{R}_t = \left(I - R_t B_t^T (B_t R_t B_t^T + \Theta_\gamma)^{-1} B_t\right) R_t$$

- Iterative prediction and update

The Kalman filter

- Prediction

$$\bar{s}_t = A_{t-1}\hat{s}_t + \mu_\varepsilon$$

$$R_t = \Theta_\varepsilon + A_{t-1}\hat{R}_{t-1}A_{t-1}^T$$

- Update

$$K_t = R_t B_t^T \left(B_t R_t B_t^T + \Theta_\gamma\right)^{-1}$$

$$\hat{s}_t = \bar{s}_t + K_t\left(o_t - B_t\bar{s}_t\right)$$

$$\hat{R}_t = \left(I - K_t B_t\right)R_t$$

Linear Gaussian Model
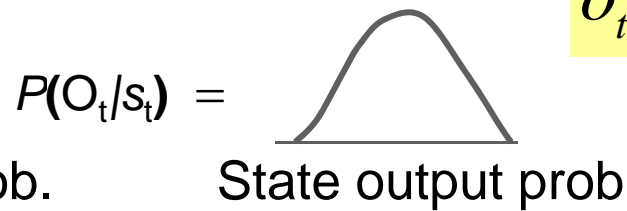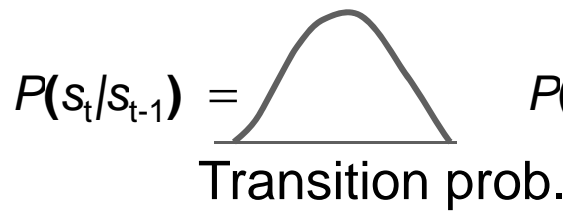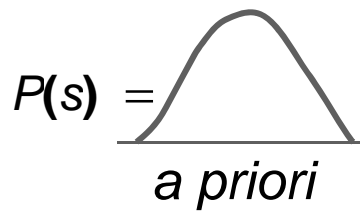
$$s_t = A_t s_{t-1} + \varepsilon_t$$

$$o_t = B_t s_t + \gamma_t$$

$P(s) =$ ⏜ $P(s_t/s_{t-1}) =$ ⏜ $P(O_t/s_t) =$ ⏜

a priori   Transition prob.   State output prob

$P(s_0) = P(s)$

$P(s_0| O_0) = C\, P(s_0)\, P(O_0| s_0)$

$$P(s_1 | O_0) = \int_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0) ds_0$$

$P(s_1| O_{0:1}) = C\, P(s_1| O_0)\, P(O_1| s_0)$

$$P(s_2 | O_{0:1}) = \int_{-\infty}^{\infty} P(s_1 | O_{0:1}) P(s_2 | s_1) ds_1$$

$P(s_2| O_{0:2}) = C\, P(s_2| O_{0:1})\, P(O_2| s_2)$

All distributions remain Gaussian

Problems

$$s_t = f(s_{t-1}, \varepsilon_t)$$

$$o_t = g(s_t, \gamma_t)$$

- f() and/or g() may not be nice linear functions
  - Conventional Kalman update rules are no longer valid

- ε and/or γ may not be Gaussian
  - Gaussian based update rules no longer valid

Problems

$$s_t = f(s_{t-1}, \varepsilon_t)$$

$$o_t = g(s_t, \gamma_t)$$

- f() and/or g() may not be nice linear functions
  - Conventional Kalman update rules are no longer valid

- ε and/or γ may not be Gaussian
  - Gaussian based update rules no longer valid

The problem with non-linear functions

$$s_t = f(s_{t-1}, \varepsilon_t)$$

$$o_t = g(s_t, \gamma_t)$$

$$P(s_t \mid o_{0:t-1}) = \int_{-\infty}^{\infty} P(s_{t-1} \mid o_{0:t-1}) P(s_t \mid s_{t-1}) ds_{t-1}$$

$$P(s_t \mid o_{0:t}) = CP(s_t \mid o_{0:t-1}) P(o_t \mid s_t)$$

- Estimation requires knowledge of P(o|s)
  - Difficult to estimate for nonlinear g()
  - Even if it can be estimated, may not be tractable with update loop

- Estimation also requires knowledge of $P(s_t | s_{t-1})$
  - Difficult for nonlinear f()
  - May not be amenable to closed form integration

The problem with nonlinearity

$$o_t = g(s_t, \gamma_t)$$

- The PDF may not have a closed form

$$P(o_t \mid s_t) = \sum_{\gamma : g(s_t, \gamma) = o_t} \frac{P_\gamma(\gamma)}{\mid J_{g(s_t, \gamma)}(o_t) \mid}$$

$$\mid J_{g(s_t, \gamma)}(o_t) \mid = \begin{vmatrix} \dfrac{\partial o_t(1)}{\partial \gamma(1)} & \cdots & \dfrac{\partial o_t(1)}{\partial \gamma(n)} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial o_t(n)}{\partial \gamma(1)} & \cdots & \dfrac{\partial o_t(n)}{\partial \gamma(n)} \end{vmatrix}$$

- Even if a closed form exists initially, it will typically become intractable very quickly

Example: a simple nonlinearity

$$o = \gamma + \log(1 + \exp(s))$$

$\gamma=0$; s
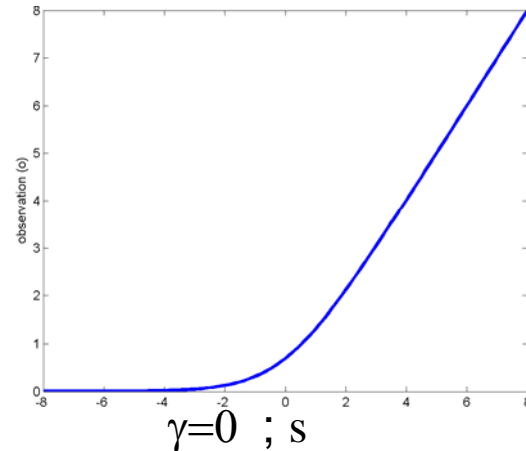
- ### P(o|s) = ?
  - Assume γ is Gaussian
  - $P(\gamma) = Gaussian(\gamma; \mu_\gamma, \Theta_\gamma)$

Example: a simple nonlinearity

$$o = \gamma + \log(1 + \exp(s))$$

$\gamma = 0$; s

- P(o|s) = ?
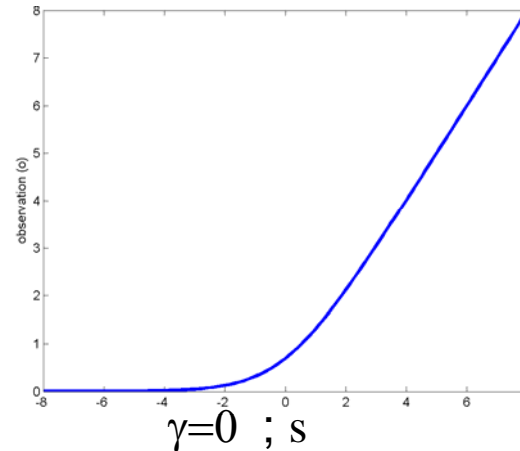
$$P(\gamma) = Gaussian(\gamma; \mu_\gamma, \Theta_\gamma)$$

$$P(o \mid s) = Gaussian(o; \mu_\gamma + \log(1 + \exp(s)), \Theta_\gamma)$$

Example: At T=0.

$$o = \gamma + \log(1 + \exp(s))$$

γ ; s=0
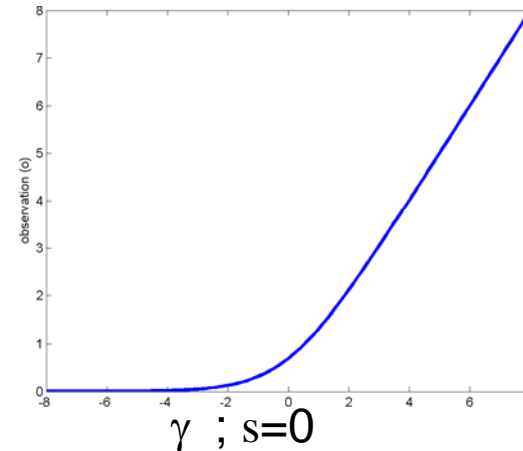
- **Assume initial probability P(s) is Gaussian**

$$P(s_0) = P_0(s) = Gaussian(s; \bar{s}, R)$$

- **Update** $\quad P(s_0 \mid o_0) = CP(o_0 \mid s_0)P(s_0)$

$$P(s_0 \mid o_0) = CGaussian(o; \mu_\gamma + \log(1 + \exp(s_0)), \Theta_\gamma)Gaussian(s_0; \bar{s}, R)$$

UPDATE: At T=0.

$$o = \gamma + \log(1 + \exp(s))$$

$\mu_\gamma = 0; \quad \bar{s} = 0$

$\Theta_\gamma = 1; \quad R = 1$

observation (o)

γ ; s=0

P(s|o=0)

s
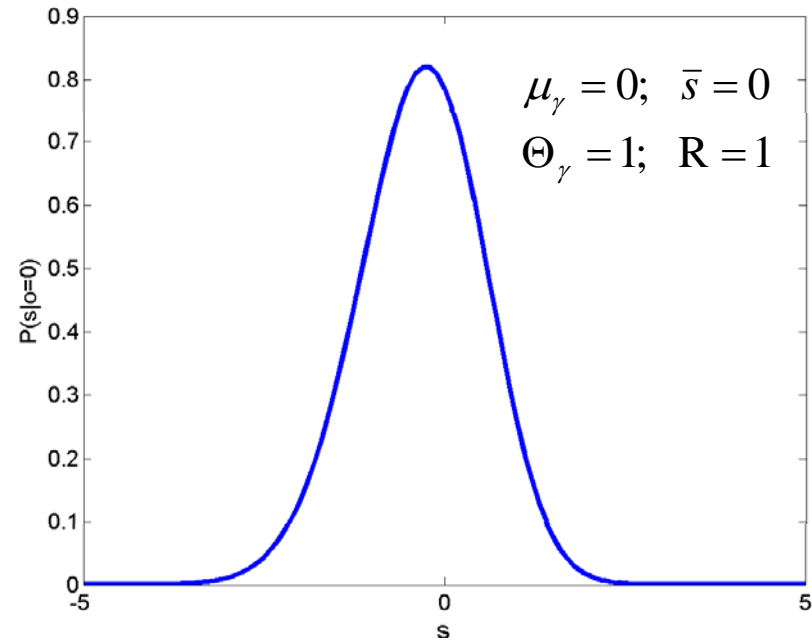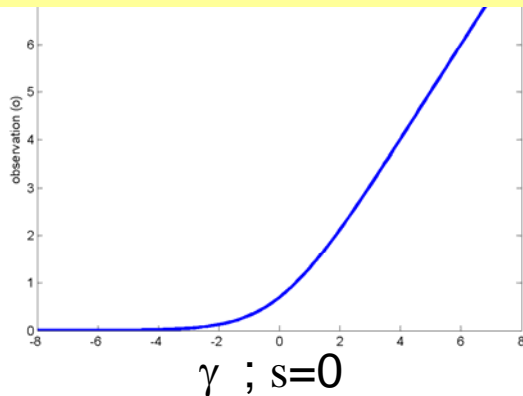
$$P(s_0 \mid o_0) = C\,Gaussian(o; \mu_\gamma + \log(1 + \exp(s_0)), \Theta_\gamma)\,Gaussian(s_0; \bar{s}, R)$$

$$P(s_0 \mid o_0) = C\exp\left(\begin{array}{c} -0.5(\mu_\gamma + \log(1 + \exp(s_0)) - o)^T \Theta_\gamma^{-1}(\mu_\gamma + \log(1 + \exp(s_0)) - o) \\ -0.5(s_0 - \bar{s})^T R^{-1}(s_0 - \bar{s}) \end{array} \right)$$

- = Not Gaussian

Prediction for $T = 1$

$$s_t = s_{t-1} + \varepsilon \qquad\qquad P(\varepsilon) = Gaussian(\varepsilon; 0, \Theta_\varepsilon)$$

- Trivial, linear state transition equation

$$P(s_t \mid s_{t-1}) = Gaussian(s_t; s_{t-1}, \Theta_\varepsilon)$$

- Prediction $\displaystyle P(s_1 \mid o_0) = \int_{-\infty}^{\infty} P(s_0 \mid o_0) P(s_1 \mid s_0) ds_0$

$$P(s_1 \mid o_0) = \int_{-\infty}^{\infty} C \exp\left(\begin{array}{c} -0.5(\mu_\gamma + \log(1 + \exp(s_0)) - o)^T \Theta_\gamma^{-1} (\mu_\gamma + \log(1 + \exp(s_0)) - o) \\ -0.5(s_0 - \bar{s})^T R^{-1}(s_0 - \bar{s}) \end{array} \right) \exp\left((s_1 - s_0)^T \Theta_\varepsilon^{-1}(s_1 - s_0) \right) ds_0$$

- = intractable

Update at T=1 and later

- ## Update at T=1

$$P(s_t \mid o_{0:t}) = CP(s_t \mid o_{0:t-1})P(o_t \mid s_t)$$

  - ❑ Intractable

- ## Prediction for T=2

$$P(s_t \mid o_{0:t-1}) = \int_{-\infty}^{\infty} P(s_{t-1} \mid o_{0:t-1})P(s_t \mid s_{t-1})ds_{t-1}$$

  - ❑ Intractable

The State prediction Equation

$$s_t = f(s_{t-1}, \varepsilon_t)$$

- Similar problems arise for the state prediction equation

- $P(s_t|s_{t-1})$ may not have a closed form

- Even if it does, it may become intractable within the prediction and update equations

  - Particularly the prediction equation, which includes an integration operation

Simplifying the problem: Linearize
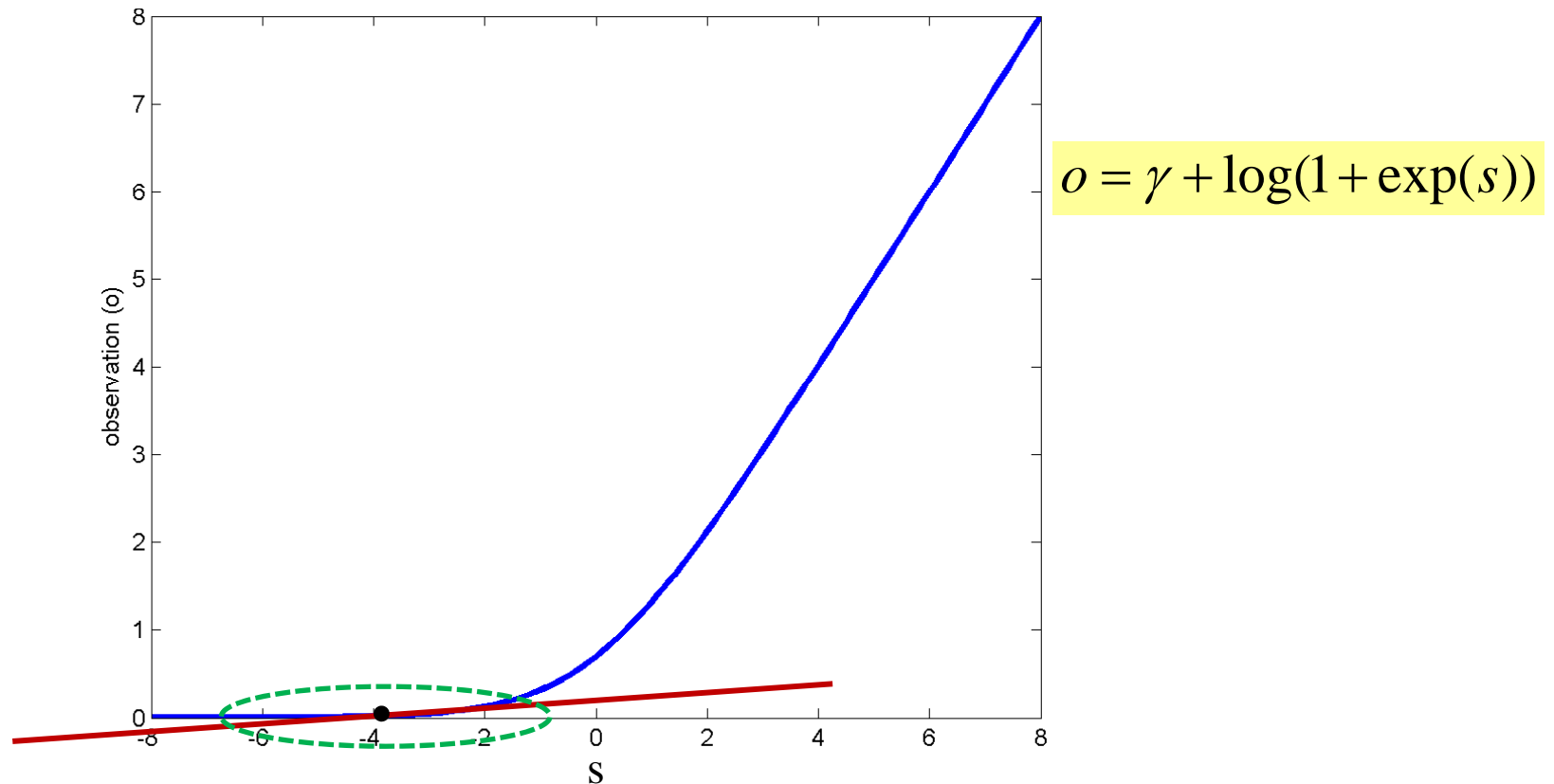
$$o = \gamma + \log(1 + \exp(s))$$

- The *tangent* at any point is a good *local* approximation if the function is sufficiently smooth

Simplifying the problem: Linearize

$$o = \gamma + \log(1 + \exp(s))$$

- The *tangent* at any point is a good *local* approximation if the function is sufficiently smooth

Simplifying the problem: Linearize

$$o = \gamma + \log(1 + \exp(s))$$

- The *tangent* at any point is a good *local* approximation if the function is sufficiently smooth

Simplifying the problem: Linearize
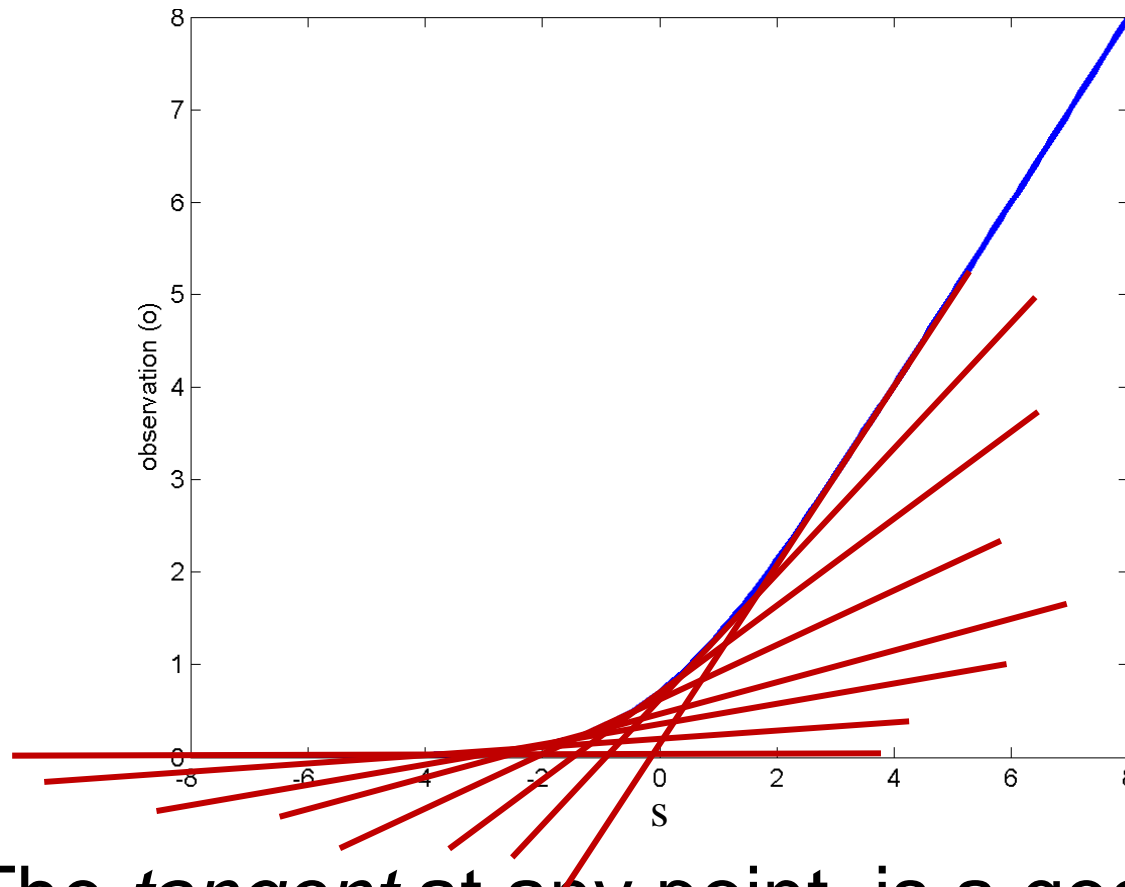
- The *tangent* at any point is a good *local* approximation if the function is sufficiently smooth

Linearizing the observation function
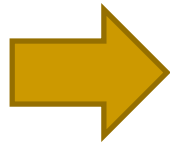
$$P(s) = Gaussian(\bar{s}, R)$$

$$o = \gamma + g(s) \quad \Longrightarrow \quad o \approx \gamma + g(\bar{s}) + J_g(\bar{s})(s - \bar{s})$$

- Simple first-order Taylor series expansion
  - J() is the Jacobian matrix
    - Simply a determinant for scalar state

- Expansion around *a priori* (or predicted) mean of the state

Most probability is in the low-error region
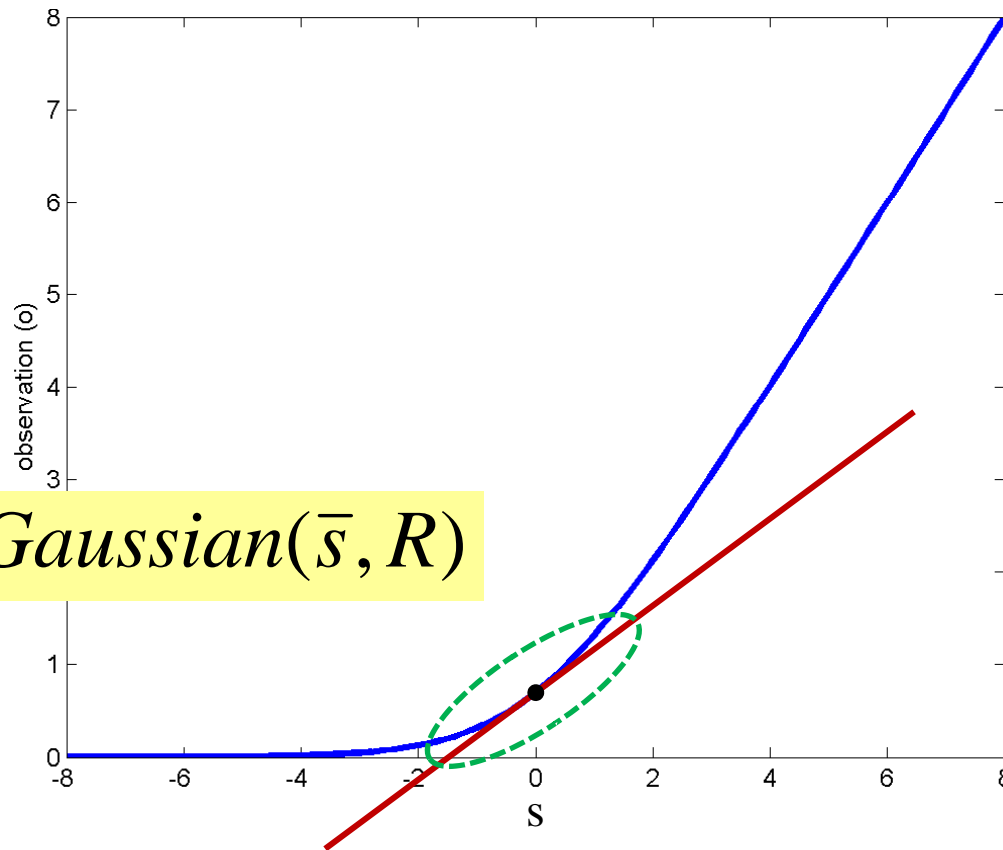
$$P(s) = Gaussian(\overline{s}, R)$$

- **P(s) is small approximation error is large**
  - Most of the probability mass of *s* is in low-error regions

Linearizing the observation function

$$P(s) = Gaussian(\overline{s}, R)$$

$$o = \gamma + g(s)$$ ⟹ $$o \approx \gamma + g(\overline{s}) + J_g(\overline{s})(s - \overline{s})$$

- **Observation PDF is Gaussian**

$$P(\gamma) = Gaussian(\gamma; 0, \Theta_\gamma)$$

$$P(o \mid s) = Gaussian(o; g(\overline{s}) + J_g(\overline{s})(s - \overline{s}), \Theta_\gamma)$$

UPDATE.

$$o \approx \gamma + g(\bar{s}) + J_g(\bar{s})(s - \bar{s})$$

$$P(o \mid s) = Gaussian(o; g(\bar{s}) + J_g(\bar{s})(s - \bar{s}), \Theta_\gamma)$$

$$P(s) = Gaussian(s; \bar{s}, R) \quad P(s \mid o) = CP(o \mid s)P(s)$$

$$P(s \mid o) = CGaussian(o; g(\bar{s}) + J_g(\bar{s})(s - \bar{s}), \Theta_\gamma)Gaussian(s; \bar{s}, R)$$

$$P(s \mid o) = Gaussian\left(s; \bar{s} + RJ_g(\bar{s})^T(J_g(\bar{s})RJ_g(\bar{s})^T + \Theta_\gamma)^{-1}(o - g(\bar{s})), (I - RJ_g(\bar{s})^T(J_g(\bar{s})RJ_g(\bar{s})^T + \Theta_\gamma)^{-1}J_g(\bar{s})R\right)$$

- **Gaussian!!**
  - **Note: This is actually only an approximation**

Prediction?

$$s_t = f(s_{t-1}) + \varepsilon \qquad P(\varepsilon) = Gaussian(\varepsilon; 0, \Theta_\varepsilon)$$

- Again, direct use of f() can be disastrous

- Solution: Linearize

$$P(s_{t-1} \mid o_{0:t-1}) = Gaussian(s_{t-1}; \hat{s}_{t-1}, \hat{R}_{t-1})$$

$$s_t = f(s_{t-1}) + \varepsilon \quad \Longrightarrow \quad s_t \approx \varepsilon + f(\hat{s}_{t-1}) + J_f(\hat{s}_{t-1})(s_{t-1} - \hat{s}_{t-1})$$

- Linearize around the mean of the updated distribution of s at t-1
  - Which should be Gaussian

Prediction

$$s_t = f(s_{t-1}) + \varepsilon \quad \Rightarrow \quad s_t \approx \varepsilon + f(\hat{s}_{t-1}) + J_f(\hat{s}_{t-1})(s_{t-1} - \hat{s}_{t-1})$$

$$P(s_{t-1} \mid o_{0:t-1}) = Gaussian(s_{t-1}; \hat{s}_{t-1}, \hat{R}_{t-1}) \qquad P(\varepsilon) = Gaussian(\varepsilon; 0, \Theta_\varepsilon)$$

- The state transition probability is now:

$$P(s_t \mid s_{t-1}) = Gaussian(s_t; f(\hat{s}_{t-1}) + J_f(\hat{s}_{t-1})(s_{t-1} - \hat{s}_{t-1}), \Theta_\varepsilon)$$

- The predicted state probability is:

$$P(s_t \mid o_{0:t-1}) = \int_{-\infty}^{\infty} P(s_{t-1} \mid o_{0:t-1}) P(s_t \mid s_{t-1}) ds_{t-1}$$

Prediction

$$P(s_{t-1} \mid o_{0:t-1}) = Gaussian(s_{t-1}; \hat{s}_{t-1}, \hat{R}_{t-1})$$

$$P(s_t \mid s_{t-1}) = Gaussian(s_t; f(\hat{s}_{t-1}) + J_f(\hat{s}_{t-1})(s_{t-1} - \hat{s}_{t-1}), \Theta_\varepsilon)$$

$$P(s_t \mid o_{0:t-1}) = \int_{-\infty}^{\infty} P(s_{t-1} \mid o_{0:t-1}) P(s_t \mid s_{t-1}) ds_{t-1}$$

$$P(s_t \mid o_{0:t-1}) = \int_{-\infty}^{\infty} Gaussian(s_{t-1}; \hat{s}_{t-1}, \hat{R}_{t-1}) Gaussian(s_t; f(\hat{s}_{t-1}) + J_f(\hat{s}_{t-1})(s_{t-1} - \hat{s}_{t-1}), \Theta_\varepsilon) ds_{t-1}$$

- The predicted state probability is:

$$P(s_t \mid o_{0:t-1}) = Gaussian\left(s_t; \hat{f}(s_{t-1}), J_f(\hat{s}_{t-1})\hat{R}_{t-1}J_f(\hat{s}_{t-1})^T + \Theta_\varepsilon\right)$$

- **Gaussian!!**

  - This is actually only an approximation

The linearized prediction/update

$$o_t = g(s_t) + \gamma$$

$$s_t = f(s_{t-1}) + \varepsilon$$

- Given: two non-linear functions for state update and observation generation

- Note: the equations are *deterministic* non-linear functions of the state variable
  - They are *linear* functions of the noise!
  - Non-linear functions of stochastic noise are slightly more complicated to handle

Linearized Prediction and Update

- Prediction for time t

$$P(s_t \mid o_{0:t-1}) = Gaussian(s_t; \bar{s}_t, R_t)$$

$$\bar{s}_t = f(\hat{s}_{t-1}) \qquad R_t = J_f(\hat{s}_{t-1})\hat{R}_{t-1}J_f(\hat{s}_{t-1})^T + \Theta_\varepsilon$$

- Update at time t

$$P(s_t \mid o_{0:t}) = Gaussian(s_t; \hat{s}_t, \hat{R}_t)$$

$$\hat{s}_t = \bar{s}_t + R_t J_g(\bar{s}_t)^T (J_g(\bar{s}_t)R_t J_g(\bar{s}_t)^T + \Theta_\gamma)^{-1}(o_t - g(\bar{s}_t))$$

$$\hat{R}_t = \left(I - R_t J_g(\bar{s}_t)^T (J_g(\bar{s}_t)R_t J_g(\bar{s}_t)^T + \Theta_\gamma)^{-1} J_g(\bar{s}_t)\right)R_t$$

Linearized Prediction and Update

$$A_t = J_f(\hat{s}_{t-1})$$

$$B_t = J_g(\bar{s}_t)$$

- **Prediction for time t**

$$P(s_t \mid o_{0:t-1}) = Gaussian(s_t; \bar{s}_t, R_t)$$

$$\bar{s}_t = f(\hat{s}_{t-1}) \qquad R_t = A_t \hat{R}_{t-1} A_t^T + \Theta_\varepsilon$$

- **Update at time t**

$$P(s_t \mid o_{0:t}) = Gaussian(s_t; \hat{s}_t, \hat{R}_t)$$

$$\hat{s}_t = \bar{s}_t + R_t B_t^T (B_t R_t B_t^T + \Theta_\gamma)^{-1} (o_t - g(\bar{s}_t))$$

$$\hat{R}_t = \left(I - R_t B_t^T (B_t R_t B_t^T + \Theta_\gamma)^{-1} B_t\right) R_t$$

The Extended Kalman filter

- Prediction

$$\bar{s}_t = f(\hat{s}_{t-1})$$

$$R_t = \Theta_\varepsilon + A_t \hat{R}_{t-1} A_t^T$$

$$A_t = J_f(\hat{s}_{t-1})$$

$$B_t = J_g(\bar{s}_t)$$

- Update

$$K_t = R_t B_t^T \left(B_t R_t B_t^T + \Theta_\gamma \right)^{-1}$$

$$\hat{s}_t = \bar{s}_t + K_t \left(o_t - g(\bar{s}_t) \right)$$

$$\hat{R}_t = \left(I - K_t B_t \right) R_t$$

The Kalman filter

- Prediction

$$\bar{s}_t = A_t \hat{s}_{t-1} + \mu_\varepsilon$$

$$R_t = \Theta_\varepsilon + A_t \hat{R}_{t-1} A_t^T$$

- Update

$$K_t = R_t B_t^T \left(B_t R_t B_t^T + \Theta_\gamma \right)^{-1}$$

$$\hat{s}_t = \bar{s}_t + K_t \left(o_t - B_t \bar{s}_t \right)$$

$$\hat{R}_t = \left(I - K_t B_t \right) R_t$$

EKFs

- EKFs are probably the most commonly used algorithm for tracking and prediction
  - Most systems are non-linear
  - Specifically, the relationship between state and observation is usually nonlinear
  - The approach can be extended to include non-linear functions of noise as well

- The term "Kalman filter" often simply refers to an *extended* Kalman filter in most contexts.

- But..

EKFs have limitations
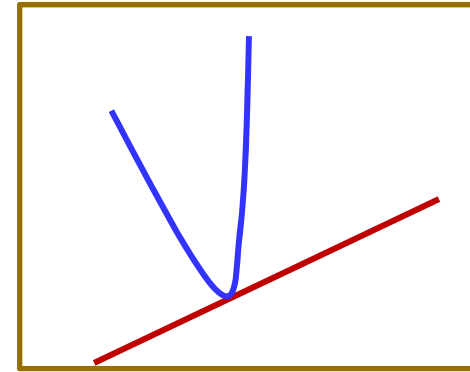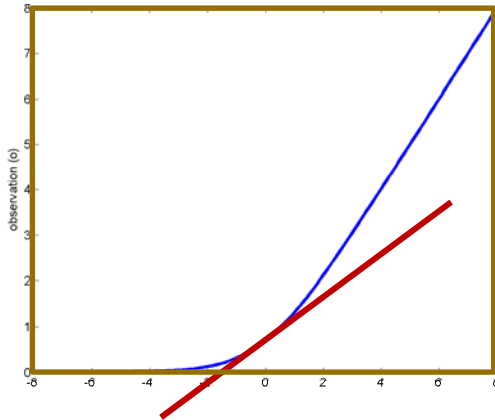
- If the non-linearity changes too quickly with s, the linear approximation is invalid
  - Unstable

- The estimate is often biased
  - The true function lies entirely on one side of the approximation

- Various extensions have been proposed:
  - Invariant extended Kalman filters (IEKF)
  - Unscented Kalman filters (UKF)

A different problem: Non-Gaussian PDFs

$$o_t = g(s_t) + \gamma \qquad\qquad s_t = f(s_{t-1}) + \varepsilon$$

- We have assumed so far that:
  - $P_0(s)$ is Gaussian or can be approximated as Gaussian
  - $P(\varepsilon)$ is Gaussian
  - $P(\gamma)$ is Gaussian

- This has a happy consequence: All distributions remain Gaussian

Linear Gaussian Model
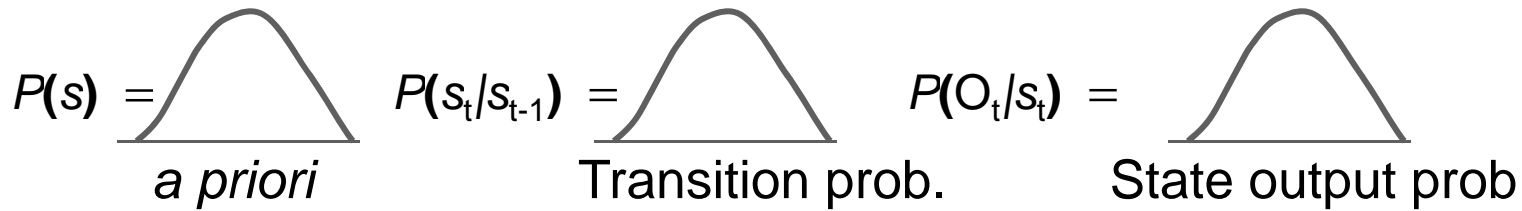
$P(s) = $ 〈curve〉 $P(s_t/s_{t-1}) = $ 〈curve〉 $P(O_t/s_t) = $ 〈curve〉

a priori   Transition prob.   State output prob

$P(s_0) = P(s)$

$P(s_0 | O_0) = C \, P(s_0) \, P(O_0 | s_0)$

$$P(s_1 | O_0) = \int_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0) ds_0$$

$P(s_1 | O_{0:1}) = C \, P(s_1 | O_0) \, P(O_1 | s_0)$

$$P(s_2 | O_{0:1}) = \int_{-\infty}^{\infty} P(s_1 | O_{0:1}) P(s_2 | s_1) ds_1$$

$P(s_2 | O_{0:2}) = C \, P(s_2 | O_{0:1}) \, P(O_2 | s_2)$

All distributions remain Gaussian

A different problem: Non-Gaussian PDFs

$$o_t = g(s_t) + \gamma \qquad s_t = f(s_{t-1}) + \varepsilon$$

- We have assumed so far that:
  - $P_0(s)$ is Gaussian or can be approximated as Gaussian
  - $P(\varepsilon)$ is Gaussian
  - $P(\gamma)$ is Gaussian

- This has a happy consequence: All distributions remain Gaussian

- But when any of these are not Gaussian, the results are not so happy

A simple case

$$o_t = Bs_t + \gamma$$

$$P(\gamma) = \sum_{i=0}^{1} w_i Gaussian(\gamma; \mu_i, \Theta_i)$$

- P(γ) is a mixture of only two Gaussians

- o is a linear function of s
  - Non-linear functions would be linearized anyway
- P(o|s) is also a Gaussian mixture!

$$P(o_t \mid s_t) = P(\gamma = o_t - Bs_t) = \sum_{i=0}^{1} w_i Gaussian(o; \mu_i + Bs_t, \Theta_i)$$

$$P(\gamma)$$

$$P(o_t \mid s_t)$$

When distributions are not Gaussian

$P(s) =$ $P(s_t/s_{t-1}) =$ $P(O_t/s_t) =$

a priori   Transition prob.   State output prob
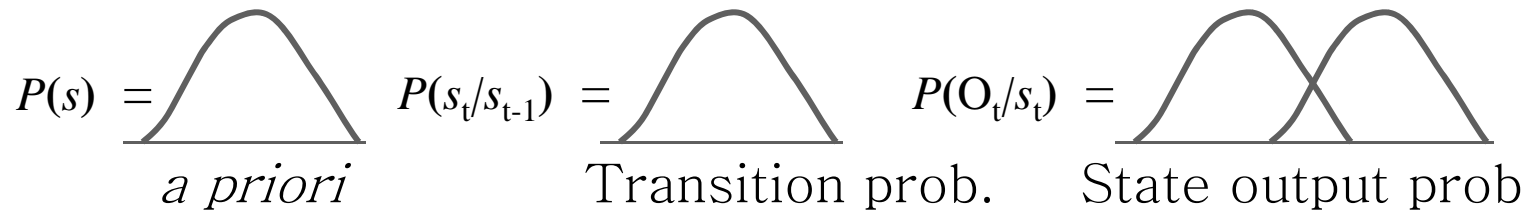
$P(s_0) = P(s)$

When distributions are not Gaussian

$P(s) =$ $P(s_t/s_{t-1}) =$ $P(O_t/s_t) =$

a priori   Transition prob.   State output prob
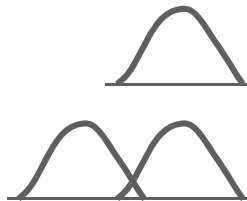
$P(s_0) = P(s)$

$P(s_0 | O_0) = C\, P(s_0)\, P(O_0 | s_0)$

When distributions are not Gaussian

$P(s) =$ $P(s_t/s_{t-1}) =$ $P(O_t/s_t) =$

a priori   Transition prob.   State output prob
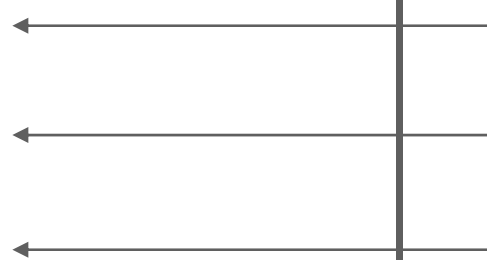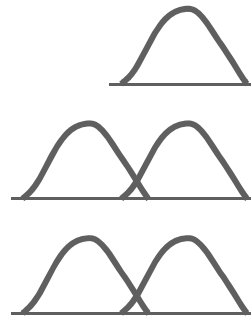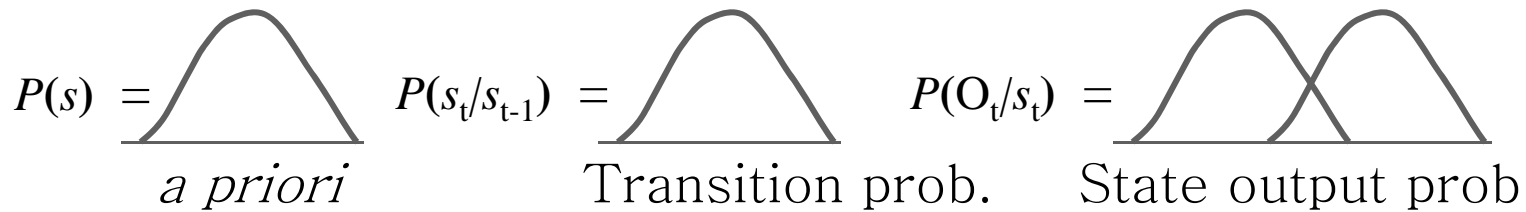
$P(s_0) = P(s)$

$P(s_0 | O_0) = C\, P(s_0)\, P(O_0 | s_0)$

$$P(s_1 | O_0) = \int_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0)\, ds_0$$

When distributions are not Gaussian

$P(s) = $ $P(s_t/s_{t-1}) = $ $P(O_t/s_t) = $

a priori   Transition prob.   State output prob
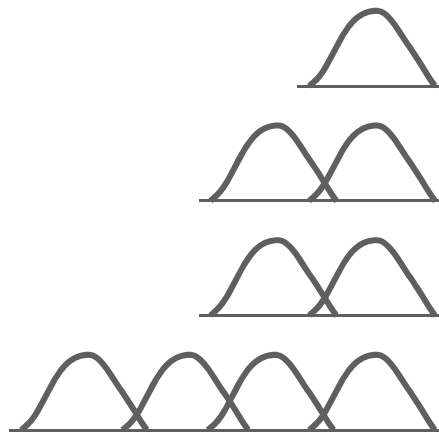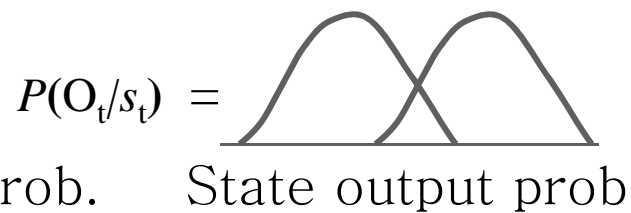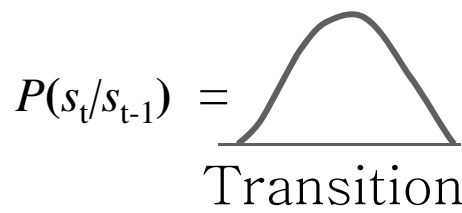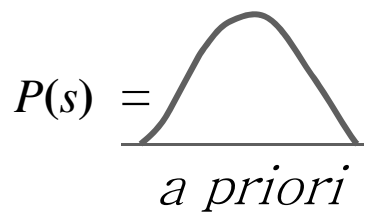
$P(s_0) = P(s)$

$P(s_0 | O_0) = C\, P(s_0)\, P(O_0 | s_0)$

$P(s_1 | O_0) = \int_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0)\, ds_0$

$P(s_1 | O_{0:1}) = C\, P(s_1 | O_0)\, P(O_1 | s_0)$

When distributions are not Gaussian

$P(s) =$ $P(s_t/s_{t-1}) =$ $P(O_t/s_t) =$

a priori Transition prob. State output prob
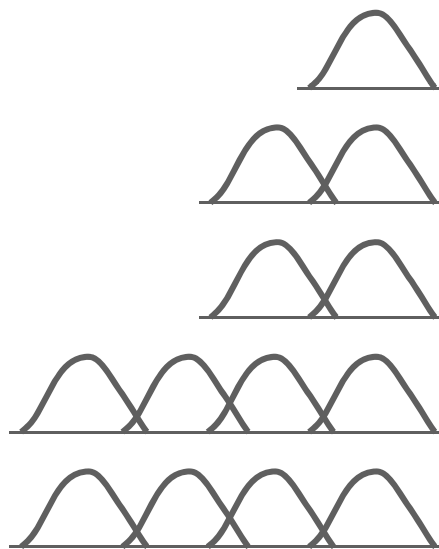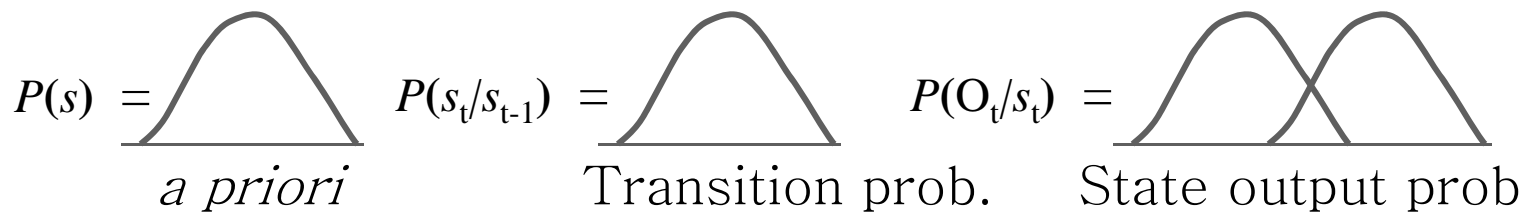
$P(s_0) = P(s)$

$P(s_0|\, O_0) = C\, P(s_0)\, P(O_0|\, s_0)$

$$P(s_1 \,|\, O_0) = \int_{-\infty}^{\infty} P(s_0 \,|\, O_0) P(s_1 \,|\, s_0)\, ds_0$$

$P(s_1|\, O_{0:1}) = C\, P(s_1|\, O_0)\, P(O_1|\, s_0)$

$$P(s_2 \,|\, O_{0:1}) = \int_{-\infty}^{\infty} P(s_1 \,|\, O_{0:1}) P(s_2 \,|\, s_1)\, ds_1$$

When distributions are not Gaussian

$P(s) =$

$P(s_t/s_{t-1}) =$

$P(O_t/s_t) =$

a priori   Transition prob.   State output prob

$P(s_0) = P(s)$

$P(s_0| O_0) = C\, P(s_0)\, P(O_0| s_0)$

$P(s_1 | O_0) = \int_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0) ds_0$

$P(s_1| O_{0:1}) = C\, P(s_1| O_0)\, P(O_1| s_0)$

$P(s_2 | O_{0:1}) = \int_{-\infty}^{\infty} P(s_1 | O_{0:1}) P(s_2 | s_1) ds_1$

$P(s_2| O_{0:2}) = C\, P(s_2| O_{0:1})\, P(O_2| s_2)$
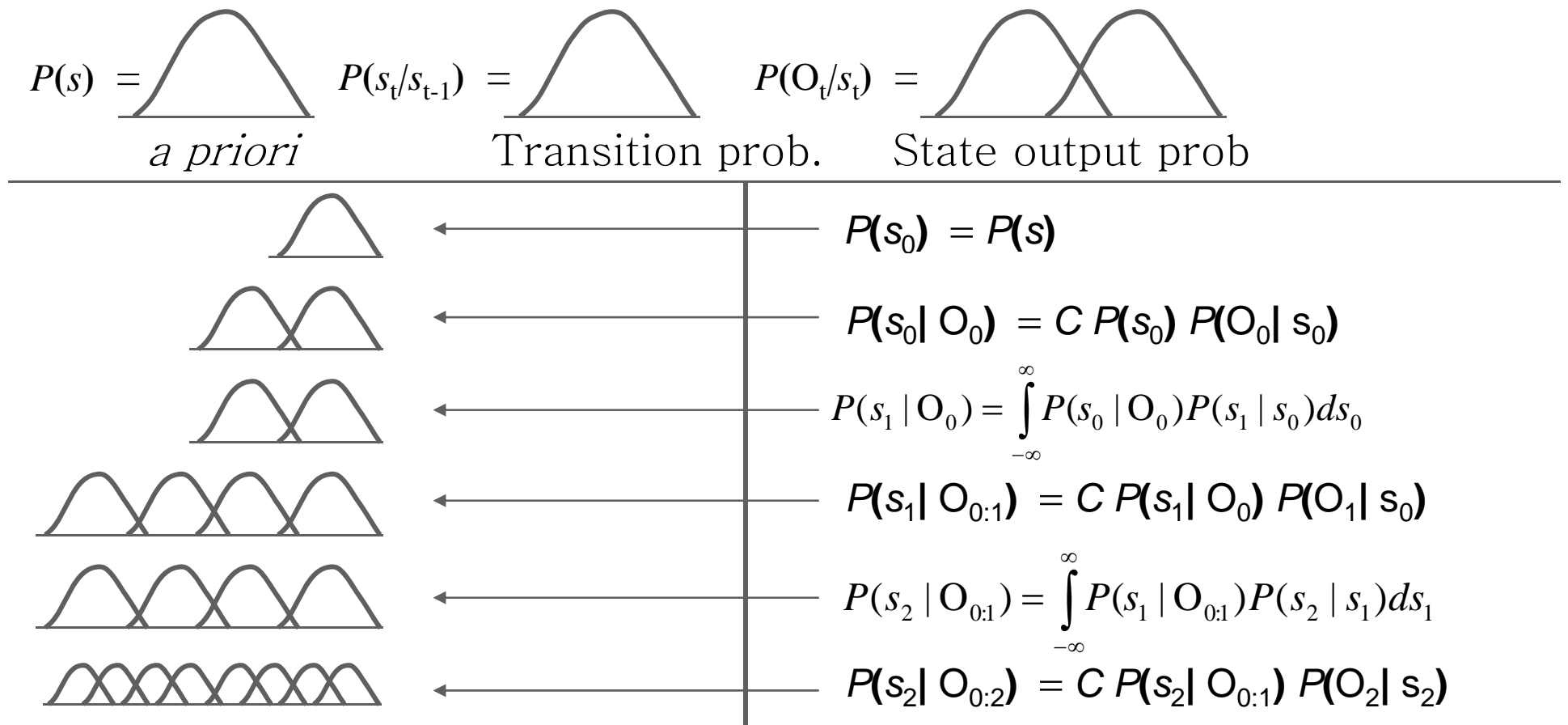
When $P(O_t/s_t)$ has more than one Gaussian, after only a few time steps…

When distributions are not Gaussian

$P(s_t \mid O_{0:t}) =$
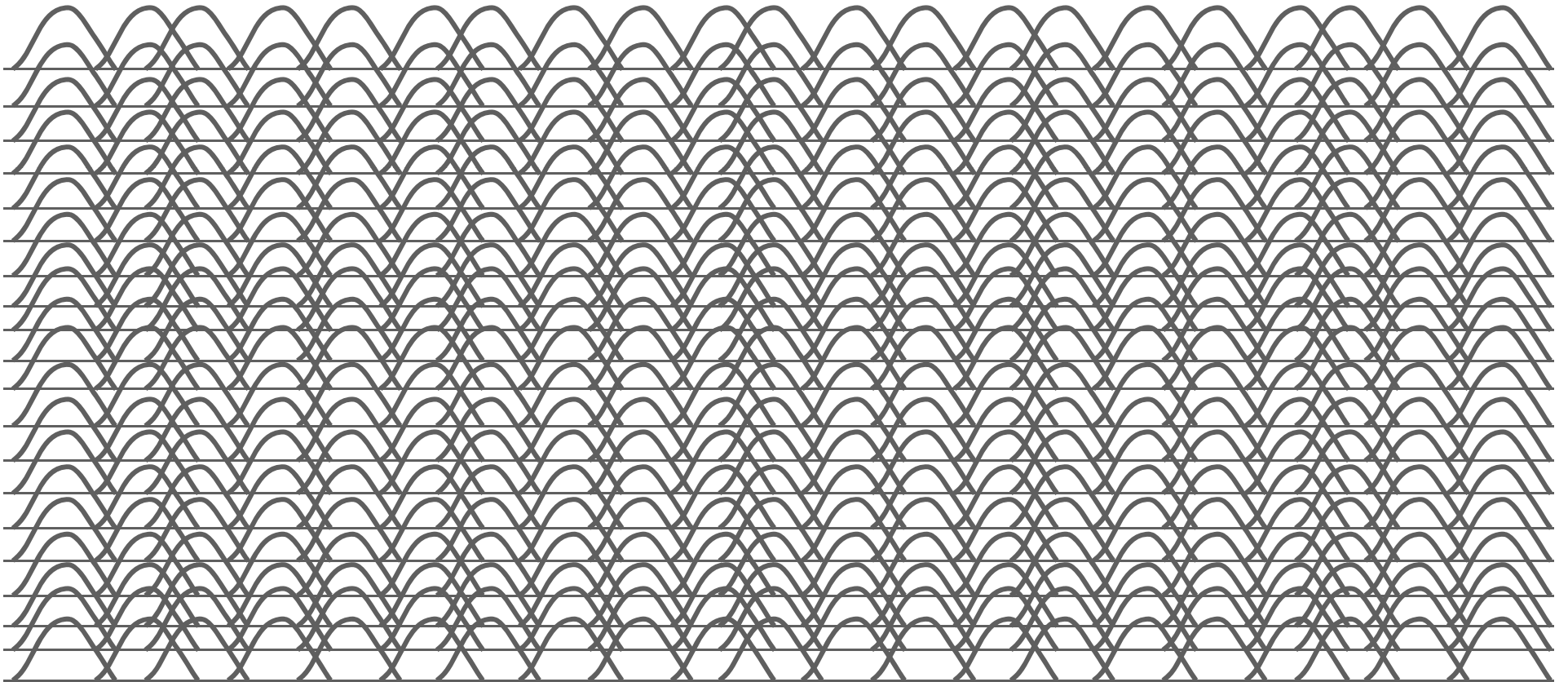
We have too many Gaussians for comfort..

Related Topic: How to sample from a Distribution?

- "Sampling from a Distribution P(x; Γ) with parameters Γ"
- Generate random numbers such that
  - The distribution of a large number of generated numbers is P(x; Γ)
  - The parameters of the distribution are Γ

- Many algorithms to generate RVs from a variety of distributions
  - Generation from a uniform distribution is well studied
  - Uniform RVs used to sample from multinomial distributions
  - Other distributions: Most commonly, transform a uniform RV to the desired distribution

Sampling from a multinomial

- Given a multinomial over N symbols, with probability of ith symbol = P(i)

- Randomly generate symbols from this distribution

- Can be done by sampling from a uniform distribution

Sampling a multinomial

1.0

P(1)   P(2)   P(3)   P(N)
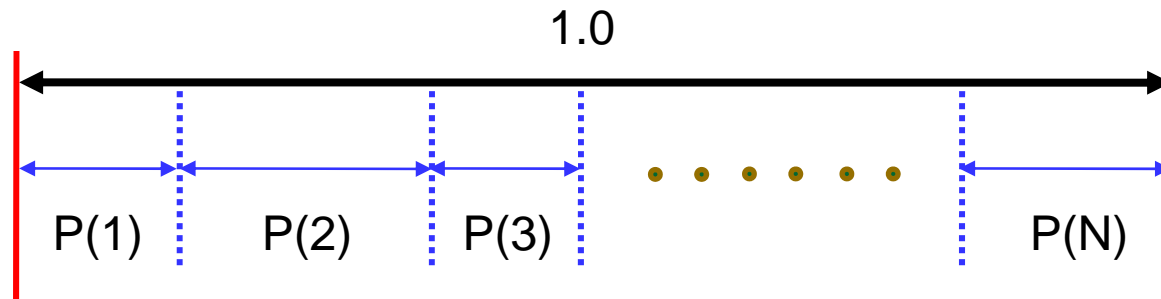
- Segment a range (0,1) according to the probabilities P(i)
  - The P(i) terms will sum to 1.0

Sampling a multinomial
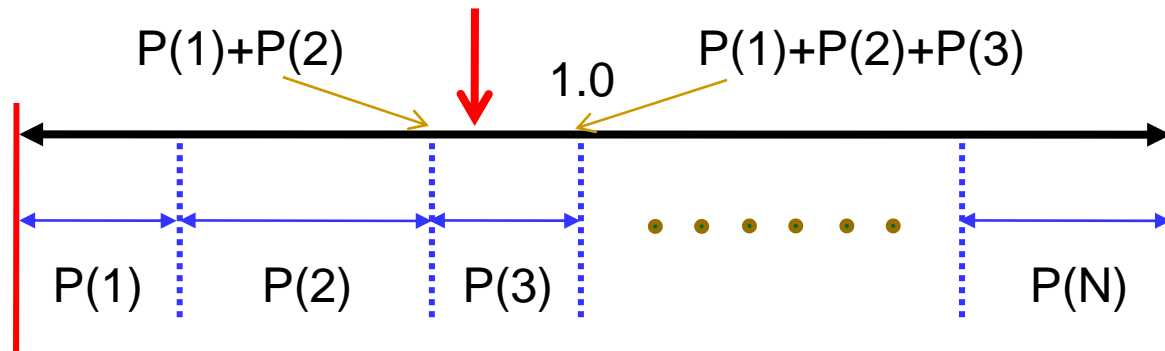
- Segment a range (0,1) according to the probabilities P(i)
  - The P(i) terms will sum to 1.0

- Randomly generate a number from a uniform distribution
  - Matlab: "rand".
  - Generates a number between 0 and 1 with uniform probability

- If the number falls in the ith segment, select the ith symbol

Related Topic: Sampling from a Gaussian

- **Many algorithms**
  - Simplest: add many samples from a uniform RV
  - The sum of 12 uniform RVs (uniform in (0,1)) is approximately Gaussian with mean 6 and variance 1
  - For scalar Gaussian, mean μ, std dev σ:

$$x = \sum_{i=1}^{12} r_i - 6$$

- **Matlab :   x = μ + randn* σ**
  - "randn" draws from a Gaussian of mean=0, variance=1

Related Topic: Sampling from a Gaussian

- **Multivariate (d-dimensional) Gaussian with mean μ and covariance Θ**

  - Compute eigen value matrix Λ and eigenvector matrix E for Θ

    - $\Theta = E \, \Lambda \, E^T$

  - Generate d 0-mean unit-variance numbers $x_1 .. x_d$

  - Arrange them in a vector:

    $$X = [x_1 \, .. \, x_d]^T$$

  - Multiply X by the square root of Λ and E, add μ

    $$Y = \mu + E \, \text{sqrt}(\Lambda) \, X$$

Sampling from a Gaussian Mixture

$$\sum_i w_i Gaussian(X; \mu_i, \Theta_i)$$

- Select a Gaussian by sampling the multinomial distribution of weights:

$$j \sim \text{multinomial}(w_1, w_2, \ldots)$$

- Sample from the selected Gaussian

$$Gaussian(X; \mu_j, \Theta_j)$$

Returning to our problem:

$P(s) =$ $P(s_t/s_{t-1}) =$ $P(O_t/s_t) =$

a priori   Transition prob.   State output prob

$P(s_0) = P(s)$

$P(s_0| O_0) = C\, P(s_0)\, P(O_0| s_0)$

$P(s_1 | O_0) = \int\limits_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0)\, ds_0$

$P(s_1| O_{0:1}) = C\, P(s_1| O_0)\, P(O_1| s_0)$

$P(s_2 | O_{0:1}) = \int\limits_{-\infty}^{\infty} P(s_1 | O_{0:1}) P(s_2 | s_1)\, ds_1$

$P(s_2| O_{0:2}) = C\, P(s_2| O_{0:1})\, P(O_2| s_2)$
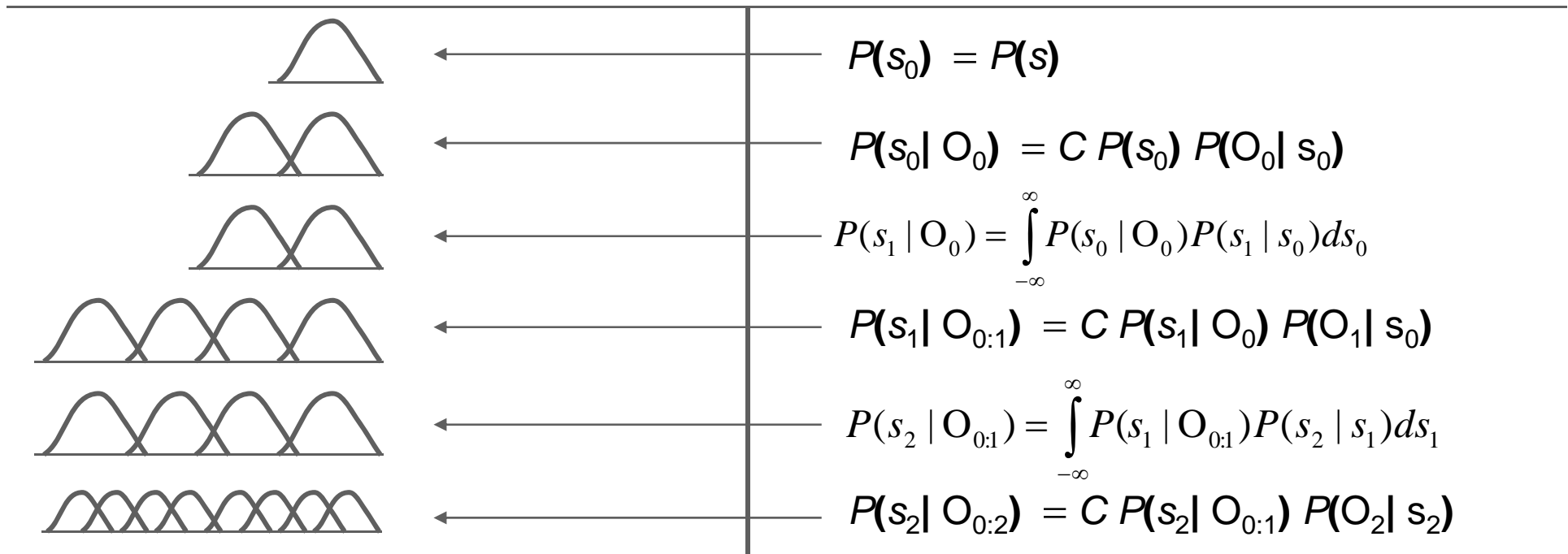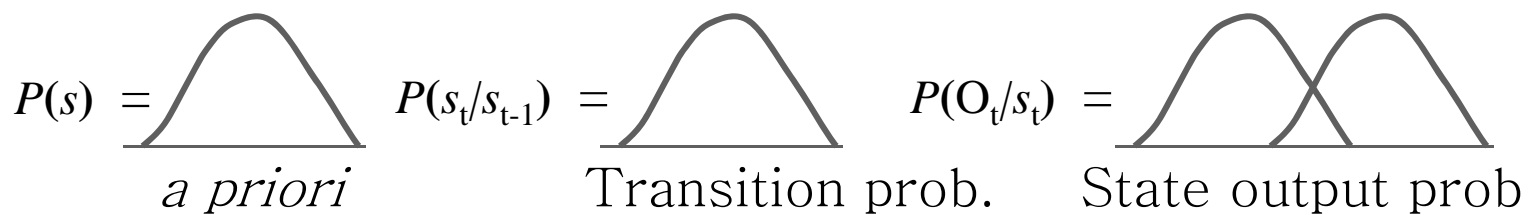
When $P(O_t/s_t)$ has more than one Gaussian, after only a few time steps…

The problem of the exploding distribution

- The complexity of the distribution increases exponentially with time

- This is a consequence of having a *continuous* state space
  - Only Gaussian PDFs propagate without increase of complexity

- *Discrete-state* systems do not have this problem
  - The number of states in an HMM stays fixed
  - However, discrete state spaces are too coarse

- Solution: Combine the two concepts
  - *Discretize* the state space dynamically

Discrete approximation to a distribution
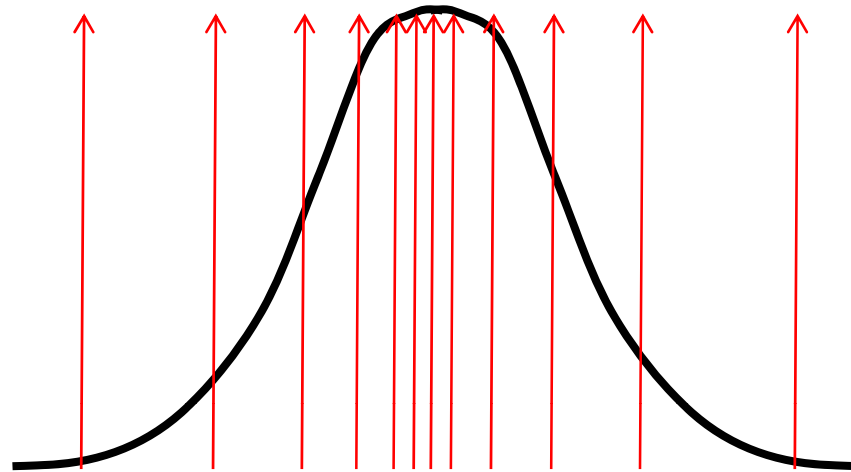
- **A large-enough collection of randomly-drawn samples from a distribution will approximately quantize the space of the random variable into equi-probable regions**
  - We have more random samples from high-probability regions and fewer samples from low-probability reigons

Discrete approximation: Random sampling
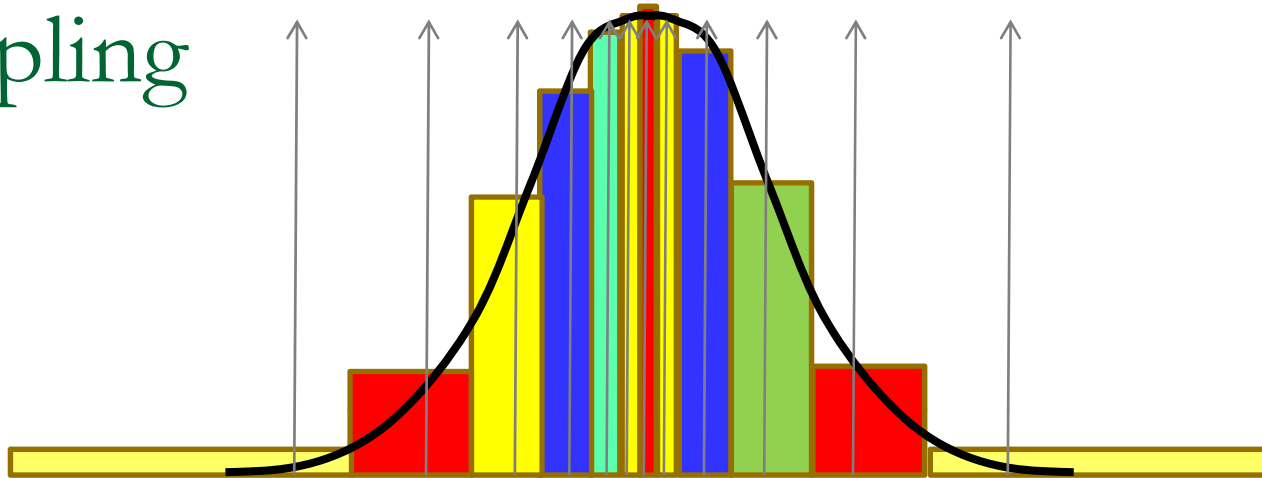
- A PDF can be approximated as a uniform probability distribution over randomly drawn samples
  - Since each sample represents approximately the same probability mass (1/M if there are M samples)

$$P(x) \approx \frac{1}{M} \sum_{i=0}^{M-1} \delta(x - x_i)$$

Note: Properties of a discrete distribution

$$P(x) \approx \frac{1}{M} \sum_{i=0}^{M-1} \delta(x - x_i)$$

$$P(x)P(y \mid x) \propto \sum_{i=0}^{M-1} P(y \mid x_i)\delta(x - x_i)$$

- The product of a discrete distribution with another distribution is simply a weighted discrete probability

$$P(x) \approx \sum_{i=0}^{M-1} w_i \delta(x - x_i)$$

$$\int_{-\infty}^{\infty} P(x)P(y \mid x)dx = \sum_{i=0}^{M-1} w_i P(y \mid x_i)$$

- The integral of the product is a mixture distribution

Discretizing the state space

- At each time, discretize the predicted state space

$$P(s_t \mid o_{0:t}) \approx \frac{1}{M} \sum_{i=0}^{M-1} \delta(s_t - s_i)$$

  - s_i are randomly drawn samples from $P(s_t \mid o_{0:t})$

- Propagate the discretized distribution

Particle Filtering

$P(s) =$ $P(s_t/s_{t-1}) =$ $P(O_t/s_t) =$

a priori   Transition prob.   State output prob

predict   ← $P(s_0) = P(s)$

Assuming that we only generate *FOUR* samples from the predicted distributions

Particle Filtering

$P(s) = $ $P(s_t/s_{t-1}) = $ $P(O_t/s_t) = $

a priori   Transition prob.   State output prob

$P(s_0) = P(s)$

Assuming that we only generate *FOUR* samples from the predicted distributions

Particle Filtering

$P(s) =$

a priori

$P(s_t/s_{t-1}) =$

Transition prob.

$P(O_t/s_t) =$

State output prob

$P(s_0) = P(s)$

$P(s_0| O_0) = C\ P(s_0)\ P(O_0| s_0)$

update

Assuming that we only generate *FOUR* samples from the predicted distributions

Particle Filtering

$P(s) =$ [curve] $P(s_t/s_{t-1}) =$ [curve] $P(O_t/s_t) =$ [curves]

a priori   Transition prob.   State output prob

$P(s_0) = P(s)$

$P(s_0| O_0) = C\ P(s_0)\ P(O_0| s_0)$

$P(s_1 | O_0) = \int_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0) ds_0$

predict

Assuming that we only generate *FOUR* samples from the predicted distributions

Particle Filtering

$P(s)$ =
a priori

$P(s_t/s_{t-1})$ =
Transition prob.

$P(O_t/s_t)$ =
State output prob

$P(s_0) = P(s)$

$P(s_0 | O_0) = C\, P(s_0)\, P(O_0 | s_0)$

$P(s_1 | O_0) = \int_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0)\, ds_0$

Assuming that we only generate *FOUR* samples from the predicted distributions

Particle Filtering

$P(s) = $ /⌒\ $P(s_t/s_{t-1}) = $ /⌒\ $P(O_t/s_t) = $ /⌒\⌒\

a priori   Transition prob.   State output prob

$P(s_0) = P(s)$

$P(s_0| O_0) = C\ P(s_0)\ P(O_0| s_0)$

$$P(s_1 \mid O_0) = \int_{-\infty}^{\infty} P(s_0 \mid O_0) P(s_1 \mid s_0)\, ds_0$$

$P(s_1| O_{0:1}) = C\ P(s_1| O_0)\ P(O_1| s_0)$

update

Assuming that we only generate *FOUR* samples from the predicted distributions

Particle Filtering

$P(s) =$ $P(s_t/s_{t-1}) =$ $P(O_t/s_t) =$

a priori   Transition prob.   State output prob

$P(s_0) = P(s)$

$P(s_0 | O_0) = C\, P(s_0)\, P(O_0 | s_0)$

$P(s_1 | O_0) = \int_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0)\, ds_0$

$P(s_1 | O_{0:1}) = C\, P(s_1 | O_0)\, P(O_1 | s_0)$

$P(s_2 | O_{0:1}) = \int_{-\infty}^{\infty} P(s_1 | O_{0:1}) P(s_2 | s_1)\, ds_1$

predict

Assuming that we only generate *FOUR* samples from the predicted distributions

Particle Filtering

$P(s)$ =

a priori

$P(s_t/s_{t-1})$ =

Transition prob.

$P(O_t/s_t)$ =

State output prob

$P(s_0) = P(s)$

$P(s_0 | O_0) = C\, P(s_0)\, P(O_0 | s_0)$

$$P(s_1 | O_0) = \int_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0)\, ds_0$$

$P(s_1 | O_{0:1}) = C\, P(s_1 | O_0)\, P(O_1 | s_0)$

$$P(s_2 | O_{0:1}) = \int_{-\infty}^{\infty} P(s_1 | O_{0:1}) P(s_2 | s_1)\, ds_1$$

Assuming that we only generate *FOUR* samples from the predicted distributions

Particle Filtering

$P(s) =$ *a priori*

$P(s_t/s_{t-1}) =$ Transition prob.

$P(O_t/s_t) =$ State output prob

$P(s_0) = P(s)$

$P(s_0| O_0) = C\, P(s_0)\, P(O_0| s_0)$

$$P(s_1 | O_0) = \int_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0)\, ds_0$$

$P(s_1| O_{0:1}) = C\, P(s_1| O_0)\, P(O_1| s_0)$

$$P(s_2 | O_{0:1}) = \int_{-\infty}^{\infty} P(s_1 | O_{0:1}) P(s_2 | s_1)\, ds_1$$

$P(s_2| O_{0:2}) = C\, P(s_2| O_{0:1})\, P(O_2| s_2)$

update

Assuming that we only generate *FOUR* samples from the predicted distributions

Particle Filtering

- **Discretize state space at the prediction step**
  - By sampling the continuous predicted distribution
    - If appropriately sampled, all generated samples may be considered to be equally probable
  - Sampling results in a **discrete** uniform distribution

- **Update step updates the distribution of the quantized state space**
  - Results in a **discrete** non-uniform distribution

- **Predicted state distribution for the next time instant will again be continuous**
  - Must be **discretized** again by sampling

- **At any step, the current state distribution will not have more components than the number of samples generated at the previous sampling step**
  - The complexity of distributions remains constant

Particle Filtering

$P(s) =$ *a priori* $P(s_t/s_{t-1}) =$ Transition prob. $P(O_t/s_t) =$ State output prob
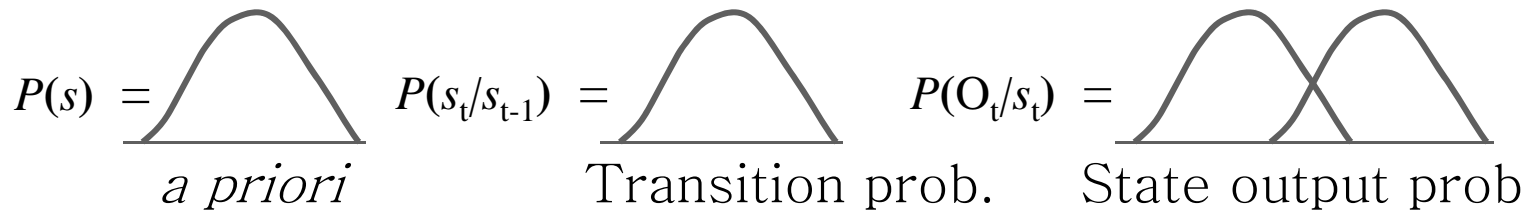
 predict

Prediction at time t:

$$P(s_t \mid O_{0:t\text{-}1}) = \int_{-\infty}^{\infty} P(s_{t-1} \mid O_{0:t\text{-}1}) P(s_t \mid s_{t-1}) ds_{t-1}$$

 update

Update at time t:

$$P(s_t \mid O_{0:t}) = C P(s_t \mid O_{0:t\text{-}1}) P(O_t \mid s_t)$$

Number of mixture components in predicted distribution governed by number of samples in discrete distribution

By deriving a small (100–1000) number of samples at each time instant, all distributions are kept manageable

Particle Filtering

$$o_t = g(s_t) + \gamma$$

$$s_t = f(s_{t-1}) + \varepsilon$$

$$P_\gamma(\gamma)$$

$$P_\varepsilon(\varepsilon)$$

- At t = 0, sample the initial state distribution

$$P(s_0 \mid o_{-1}) = P(s_0) \approx \frac{1}{M} \sum_{i=0}^{M-1} \delta(s_0 - \bar{s}_i^0) \text{ where } \bar{s}_i^0 \leftarrow P_0(s)$$

- Update the state distribution with the observation

$$P(s_t \mid o_{0:t}) = C \sum_{i=0}^{M-1} P_\gamma(o_t - g(\bar{s}_i^t)) \delta(s_t - \bar{s}_i^t)$$

$$C = \frac{1}{\sum_{i=0}^{M-1} P_\gamma(o_t - g(\bar{s}_i^t))}$$

Particle Filtering

$$o_t = g(s_t) + \gamma$$

$$s_t = f(s_{t-1}) + \varepsilon$$

$$P_\gamma(\gamma)$$

$$P_\varepsilon(\varepsilon)$$

- Predict the state distribution at the next time

$$P(s_t \mid o_{0:t-1}) = C \sum_{i=0}^{M-1} P_\gamma(o_{t-1} - g(\bar{s}_i^{t-1})) P_\varepsilon(s_t - f(\bar{s}_i^{t-1}))$$

- Sample the predicted state distribution

$$P(s_t \mid o_{0:t-1}) \approx \frac{1}{M} \sum_{i=0}^{M-1} \delta(s_t - \bar{s}_i^t) \quad \text{where} \quad \bar{s}_i^t \leftarrow P(s_t \mid o_{0:t-1})$$

Particle Filtering

$$o_t = g(s_t) + \gamma \qquad P_\gamma(\gamma) \qquad\qquad s_t = f(s_{t-1}) + \varepsilon \qquad P_\varepsilon(\varepsilon)$$

- Predict the state distribution at t

$$P(s_t \mid o_{0:t-1}) = C \sum_{i=0}^{M-1} P_\gamma(o_{t-1} - g(\bar{s}_i^{t-1})) P_\varepsilon(s_t - f(\bar{s}_i^{t-1}))$$

- Sample the predicted state distribution at t

$$P(s_t \mid o_{0:t-1}) \approx \frac{1}{M} \sum_{i=0}^{M-1} \delta(s_t - \bar{s}_i^t) \quad \text{where} \quad \bar{s}_i^t \leftarrow P(s_t \mid o_{0:t-1})$$

- Update the state distribution at t

$$P(s_t \mid o_{0:t}) = C \sum_{i=0}^{M-1} P_\gamma(o_t - g(\bar{s}_i^t)) \delta(s_t - \bar{s}_i^t) \qquad C = \frac{1}{\sum_{i=0}^{M-1} P_\gamma(o_t - g(\bar{s}_i^t))}$$

Estimating a state

- The algorithm gives us a discrete updated distribution over states:

$$P(s_t \mid o_{0:t}) = C \sum_{i=0}^{M-1} P_\gamma(o_t - g(\bar{s}_i^t)) \delta(s_t - \bar{s}_i^t)$$

- The actual state can be estimated as the mean of this distribution

$$\hat{s}_t = C \sum_{i=0}^{M-1} \bar{s}_i^t P_\gamma(o_t - g(\bar{s}_i^t))$$

- Alternately, it can be the most likely sample

$$\hat{s}_t = \bar{s}_j^t \; : \; j = \arg\max_i P_\gamma(o_t - g(\bar{s}_i^t))$$

Simulations with a Linear Model

$$s_t = s_{t-1} + \varepsilon_t \qquad o_t = s_t + x_t$$

- ε_t has a Gaussian distribution with 0 mean, known variance
- X_t has a mixture Gaussian distribution with known parameters
- Simulation:

  - Generate state sequence S_t from model

  - Generate sequence of X_t from model with one X_t term for every S_t term

  - Generate observation sequence O_t from S_t and X_t

  - Attempt to estimate S_t from O_t

Simulation: Synthesizing data

Generate state sequence according to:

ε_t is Gaussian with mean 0 and variance 10

$$s_t = s_{t-1} + \varepsilon_t$$

Simulation: Synthesizing data

Generate state sequence according to: $s_t = s_{t-1} + \varepsilon_t$

ε_t is Gaussian with mean 0 and variance 10

Generate observation sequence from state sequence according to: $o_t = s_t + x_t$

x_t is mixture Gaussian with parameters:

Means = [-4, 0, 4, 8, 12, 16, 18, 20]

Variances = [10, 10, 10, 10, 10, 10, 10, 10]

Mixture weights = [0.125, 0.125, 0.125, 0.125, 0.125, 0.125, 0.125, 0.125]
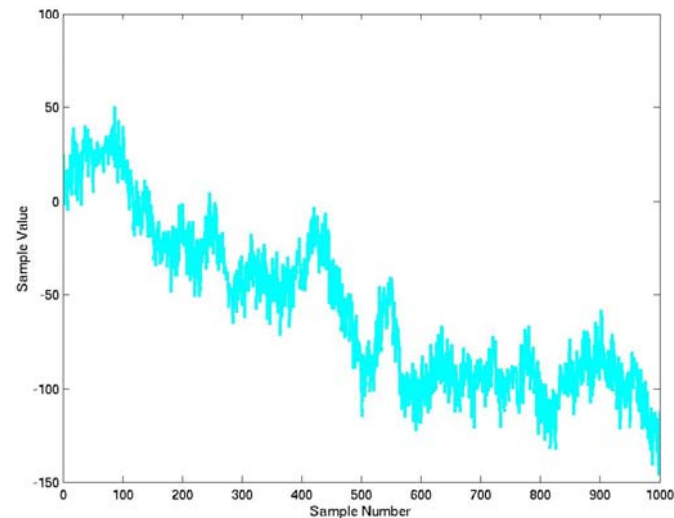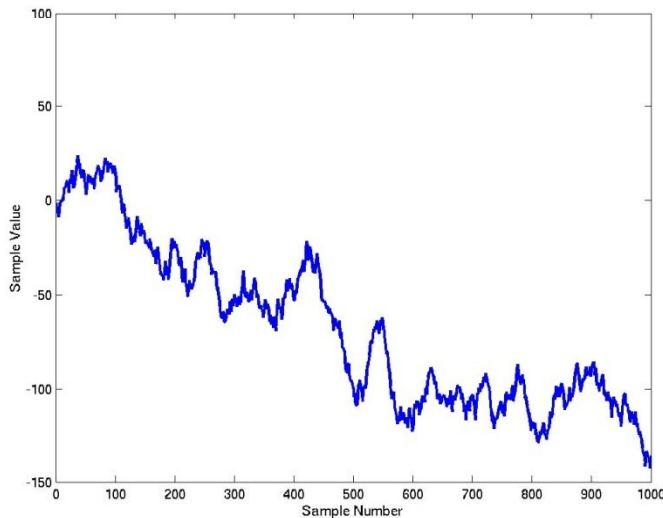
Simulation: Synthesizing data
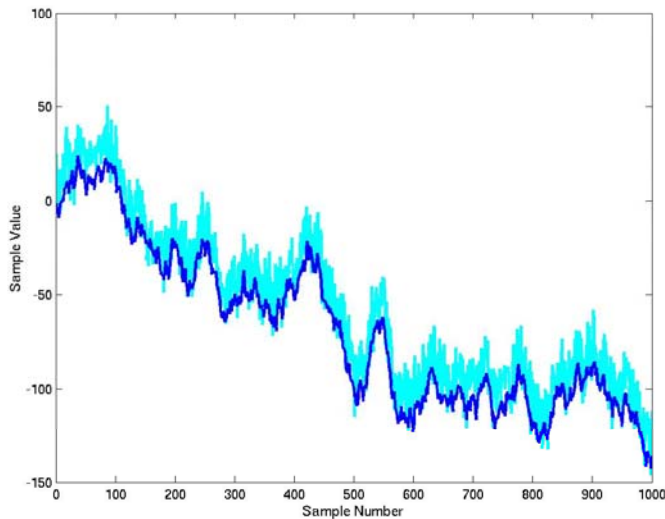
Combined figure for more compact representation

SIMULATION: TIME = 1

predict

PREDICTED STATE DISTRIBUTION
AT TIME = 1

SIMULATION: TIME = 1
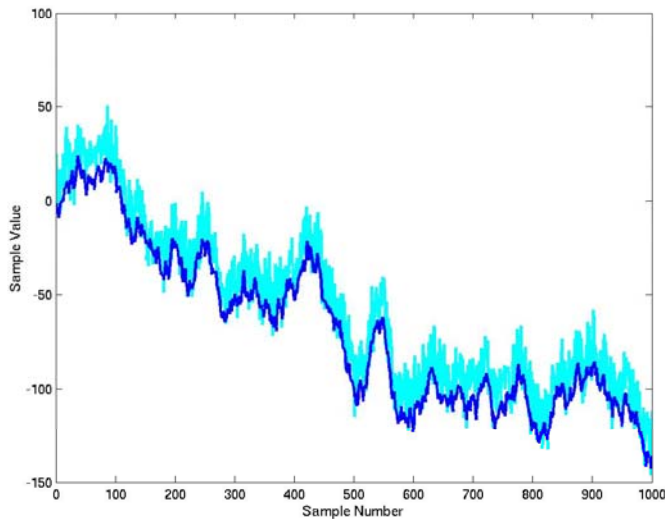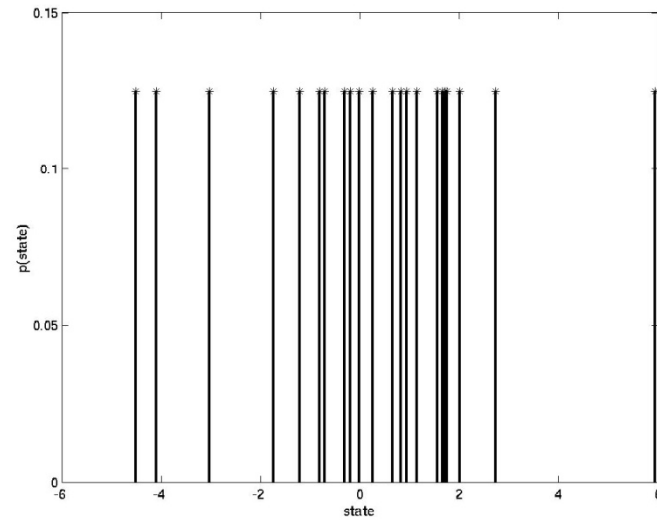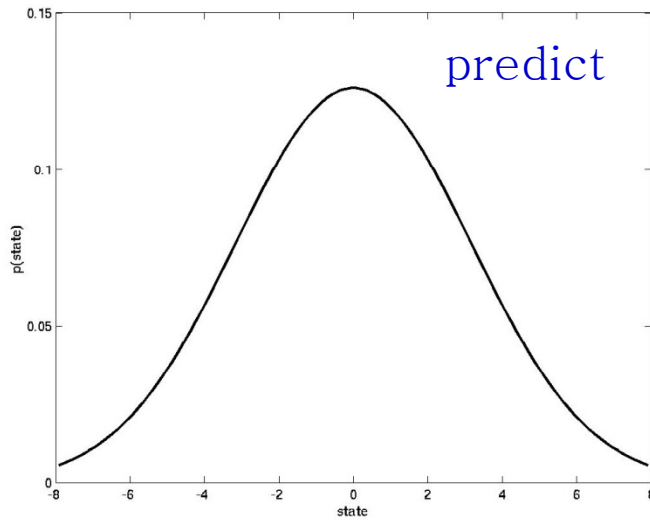
predict

SAMPLED VERSION OF
PREDICTED STATE DISTRIBUTION
AT TIME = 1

SIMULATION: TIME = 1

SAMPLED VERSION OF
PREDICTED STATE DISTRIBUTION
AT TIME = 1
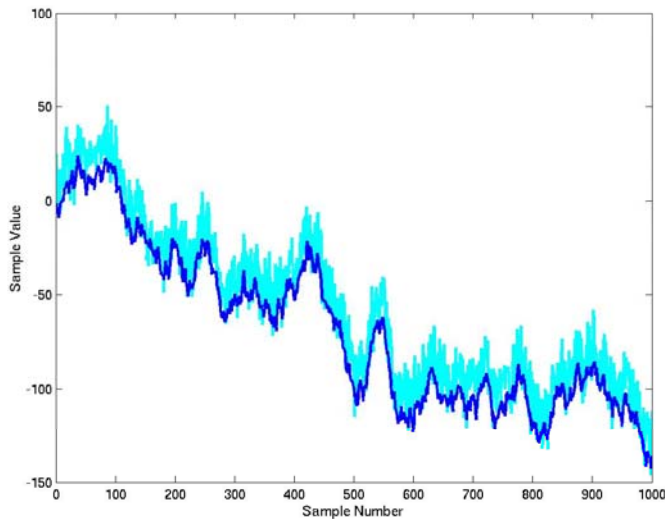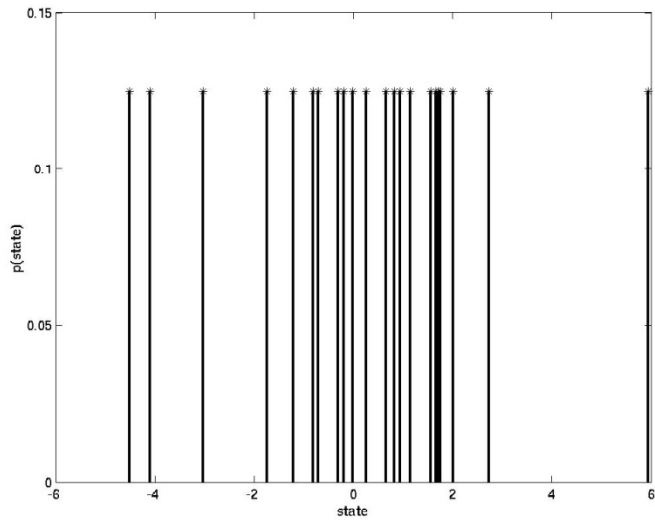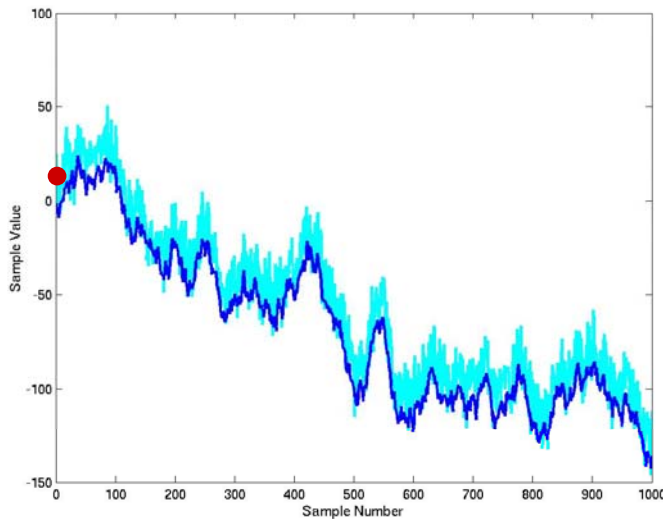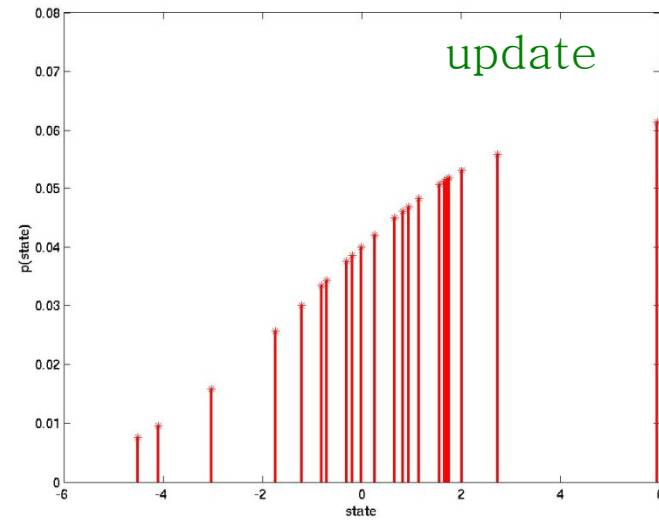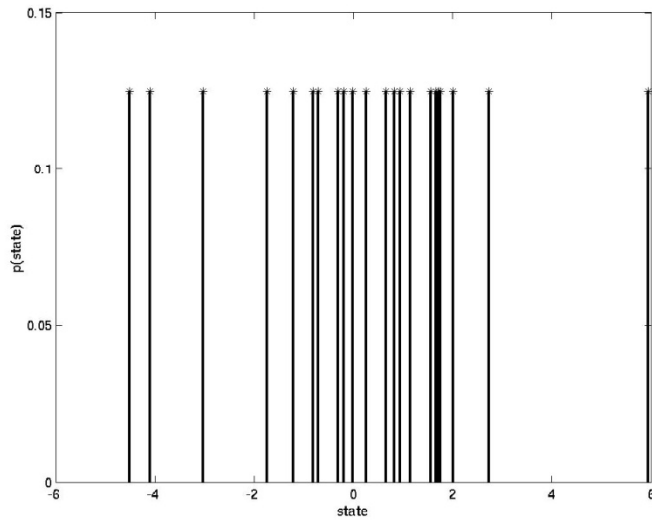
SIMULATION: TIME = 1

UPDATED VERSION OF
SAMPLED VERSION OF
PREDICTED STATE DISTRIBUTION
AT TIME = 1
AFTER SEEING FIRST OBSERVATION

SIMULATION: TIME = 1
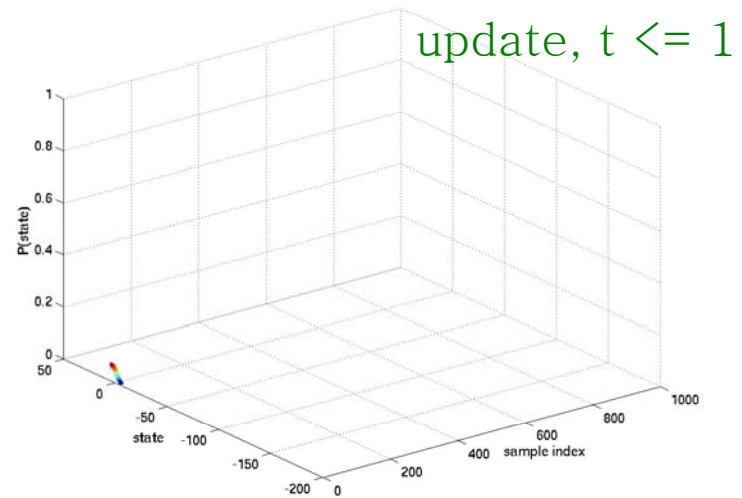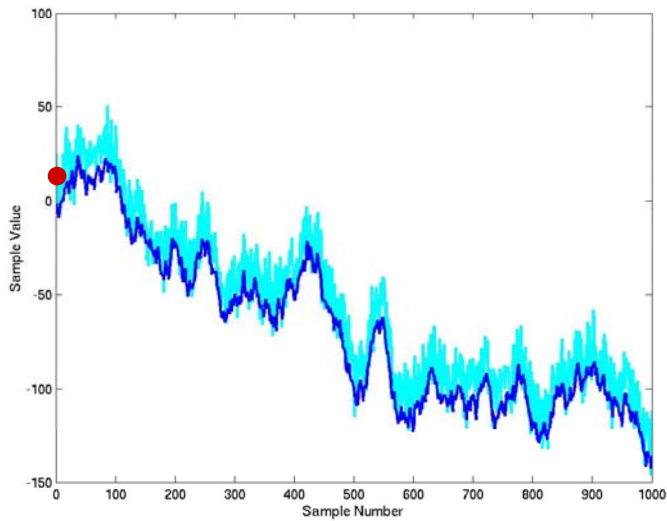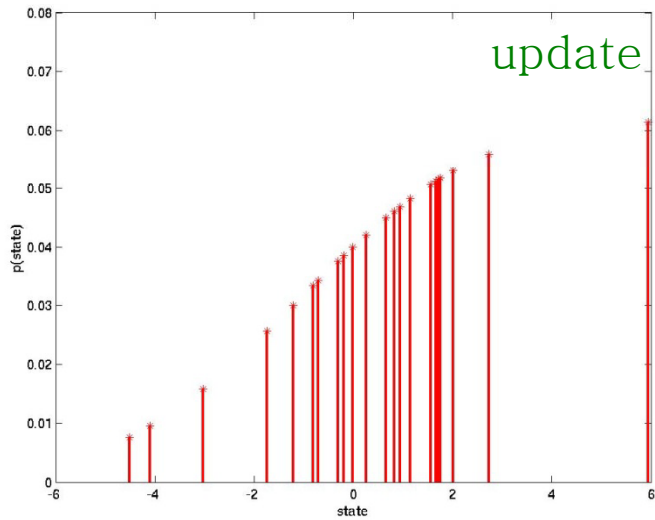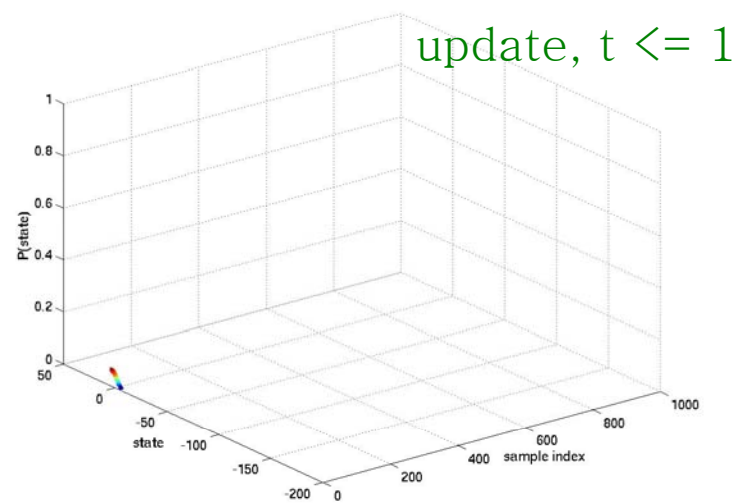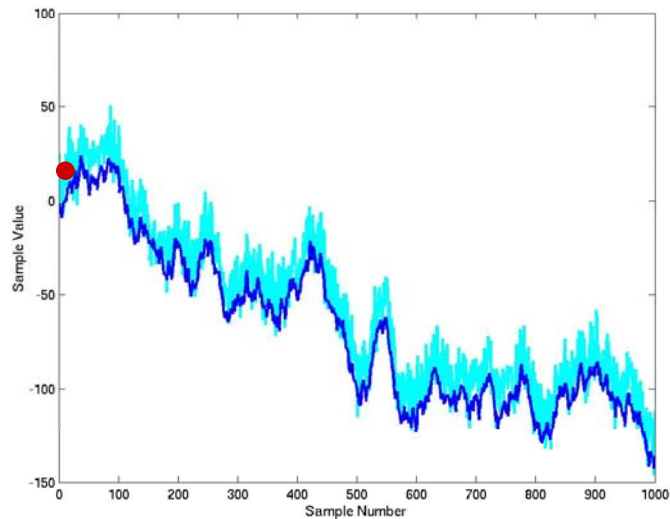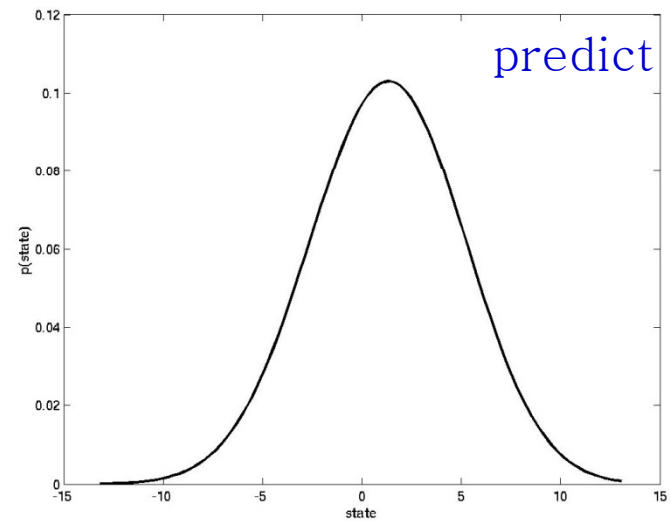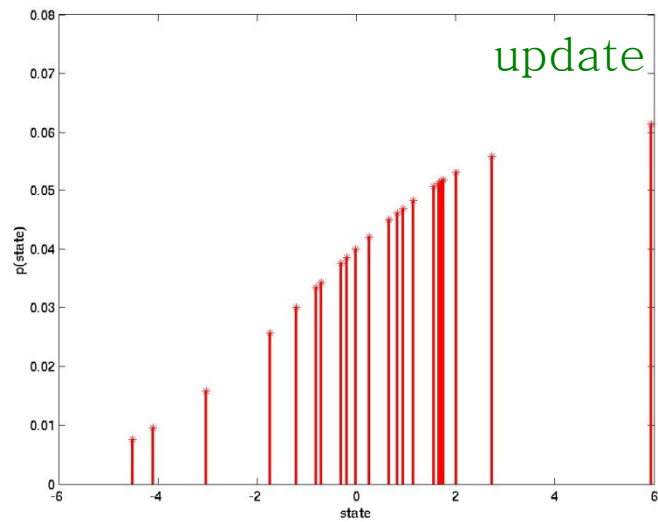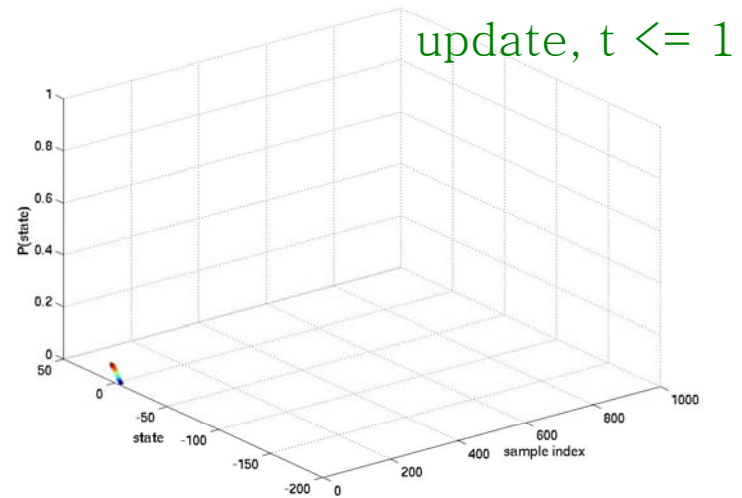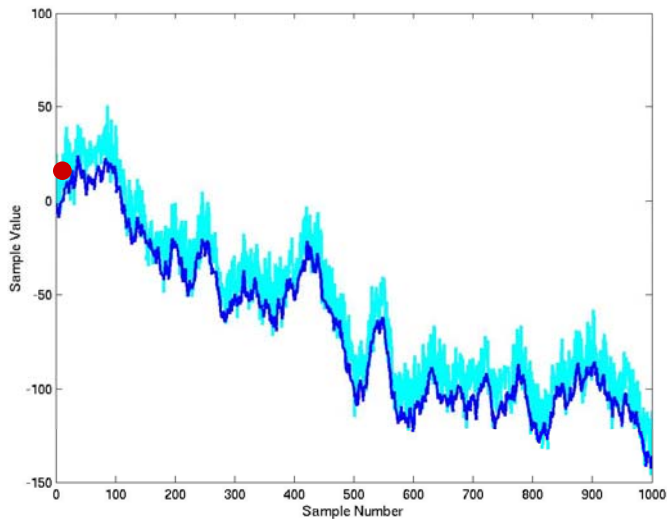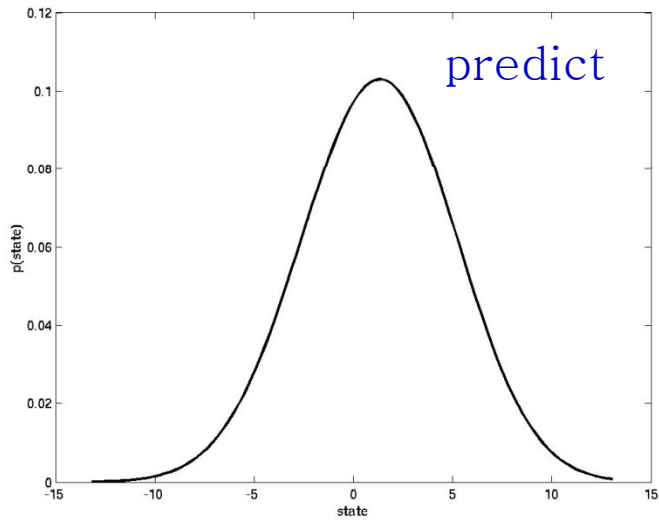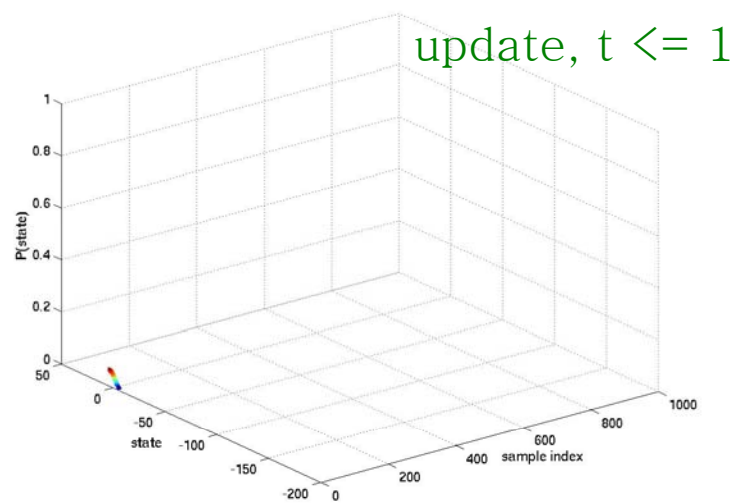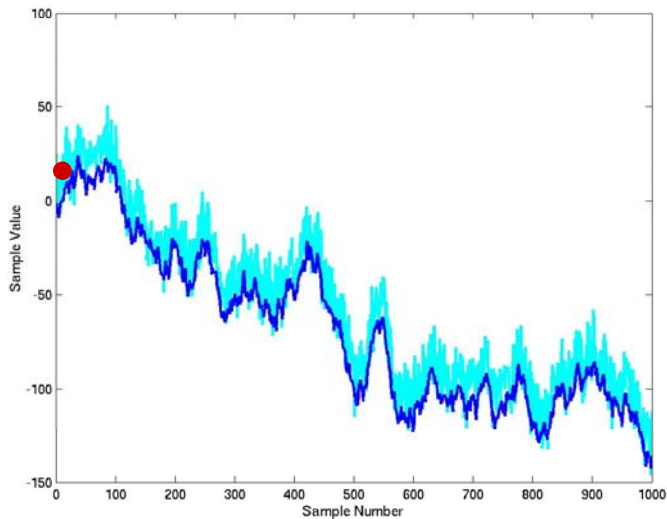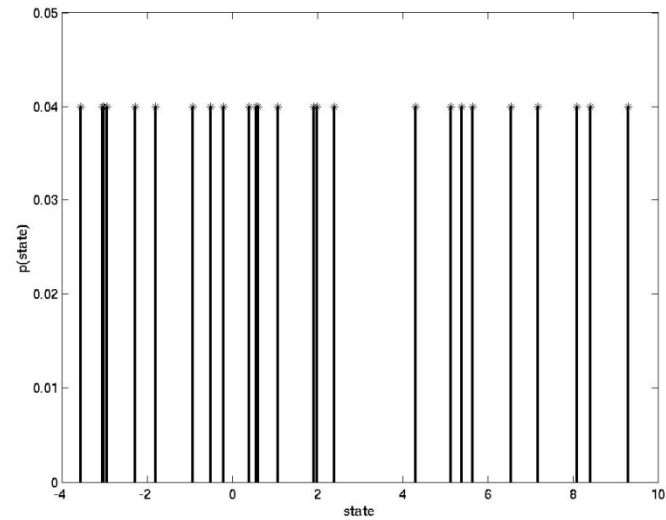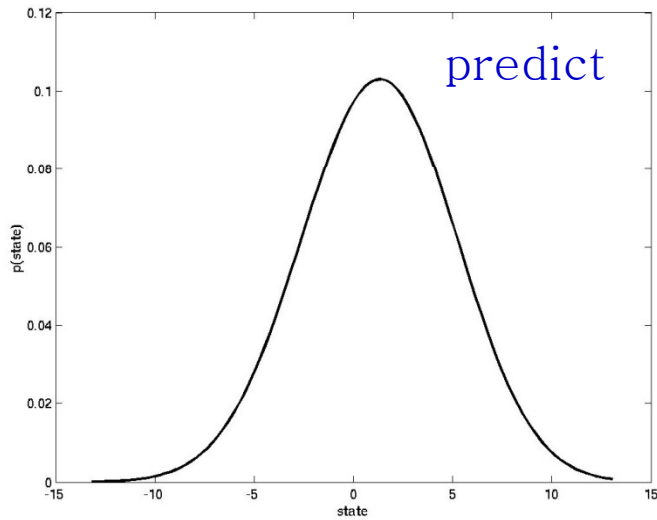
update

update, t <= 1

SIMULATION: TIME = 2

update

predict

update, t <= 1

SIMULATION: TIME = 2

predict

update, t <= 1

SIMULATION: TIME = 2

predict

update, t <= 1

SIMULATION: TIME = 2

update, t <= 1

SIMULATION: TIME = 2

update

update, t <= 1

SIMULATION: TIME = 2

update

update, t <= 2

SIMULATION: TIME = 3

update

predict

update, t <= 2

SIMULATION: TIME = 3

predict

update, t <= 2

SIMULATION: TIME = 3

predict
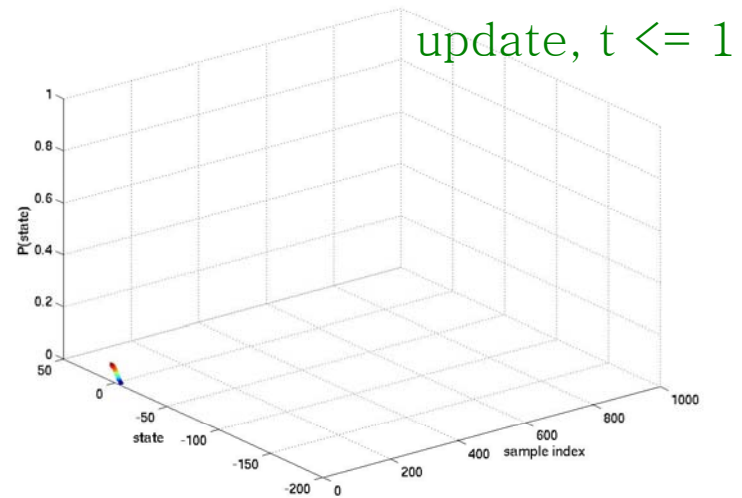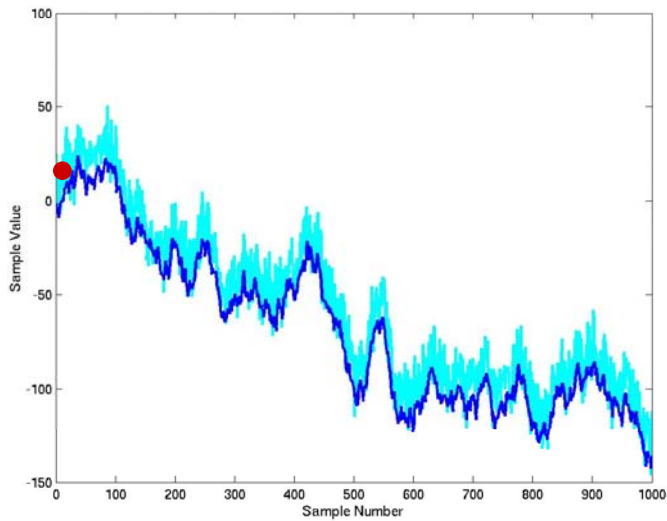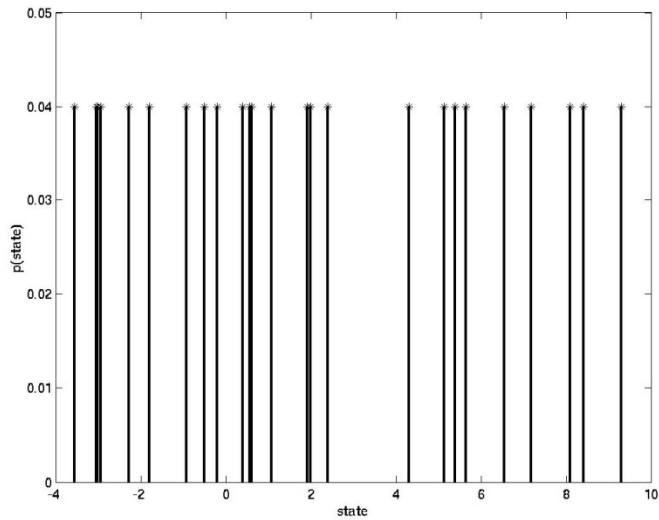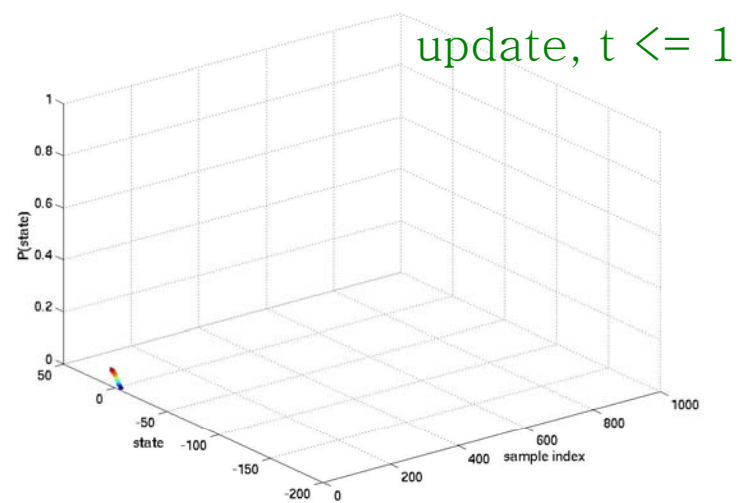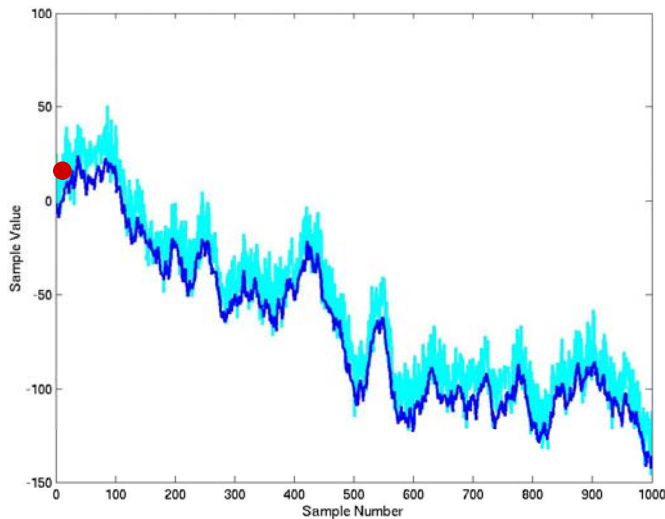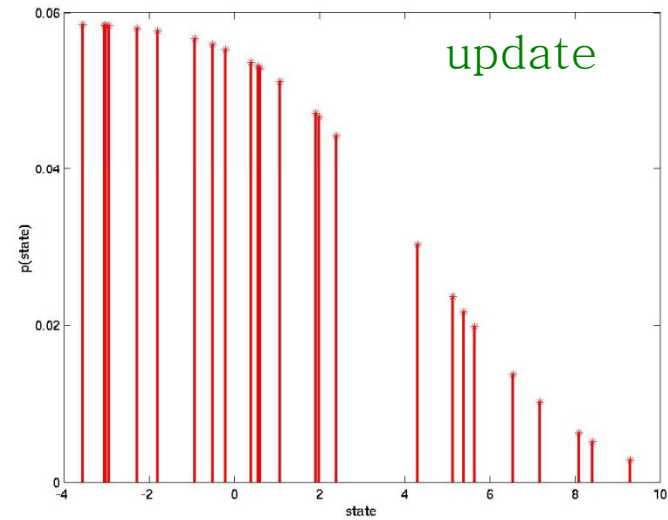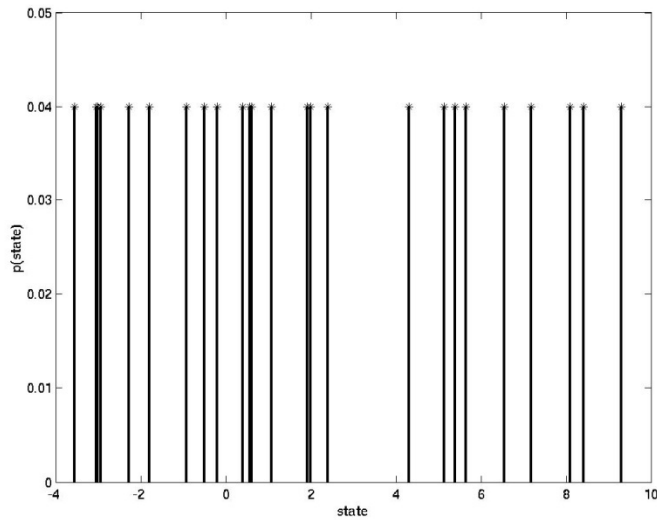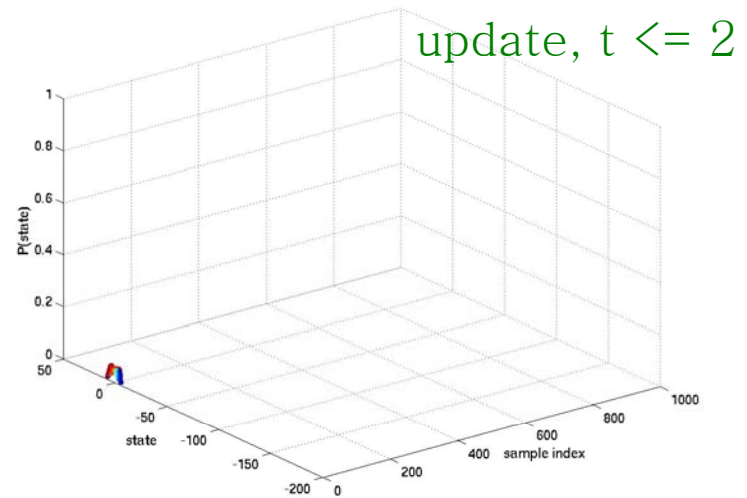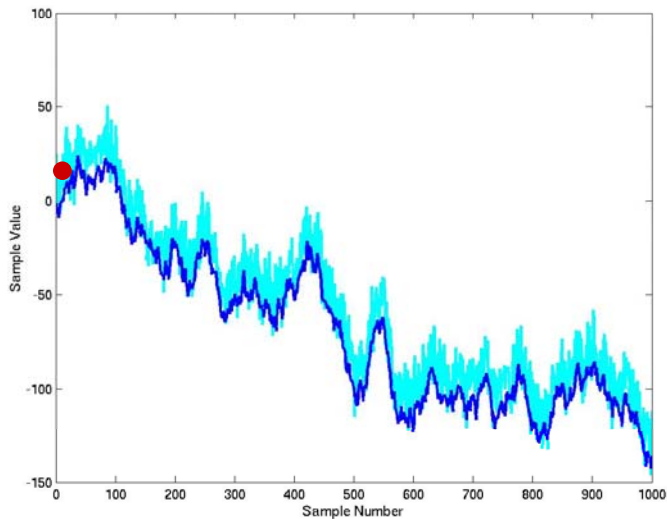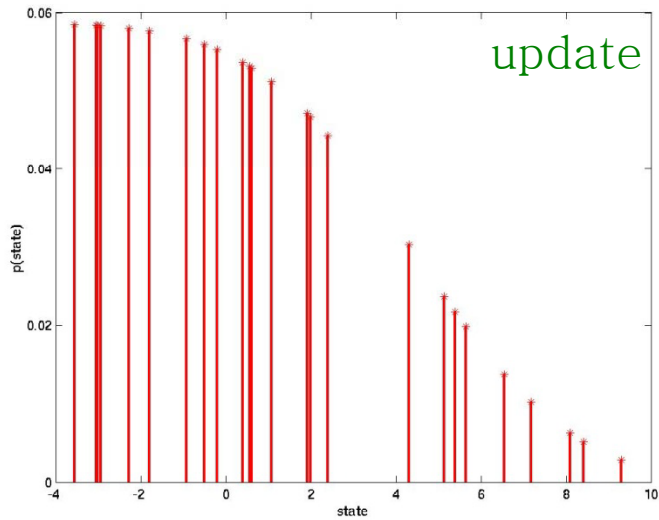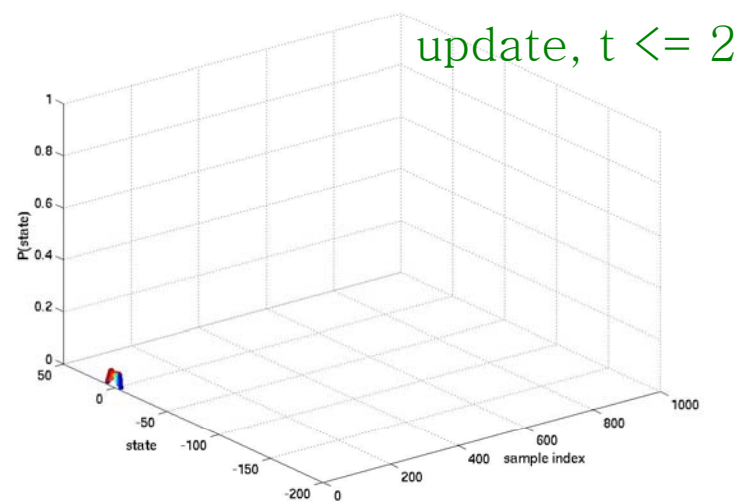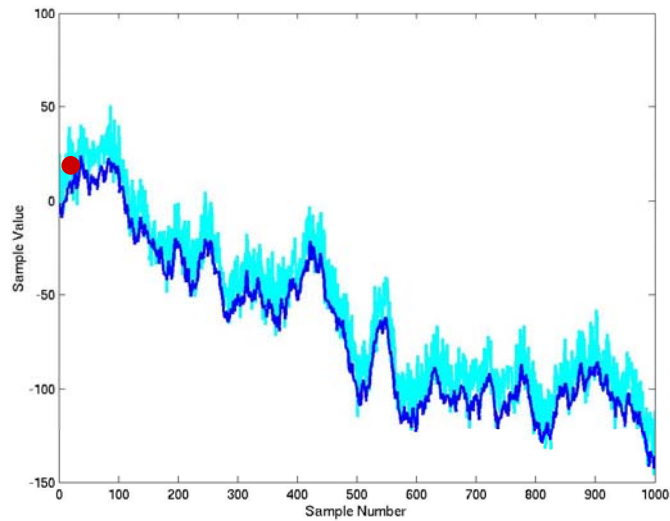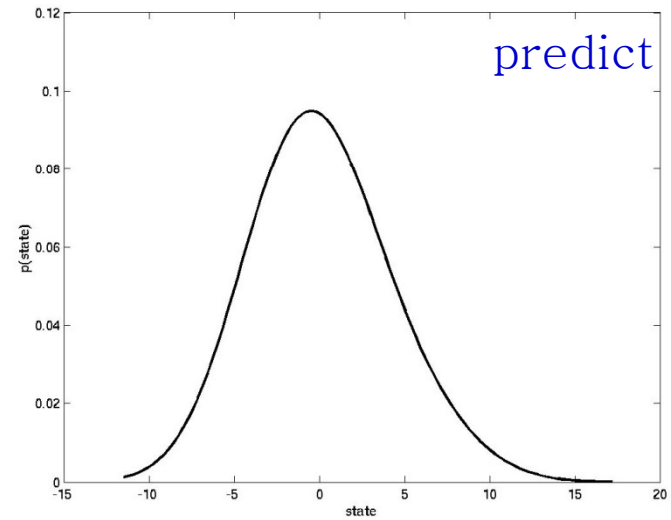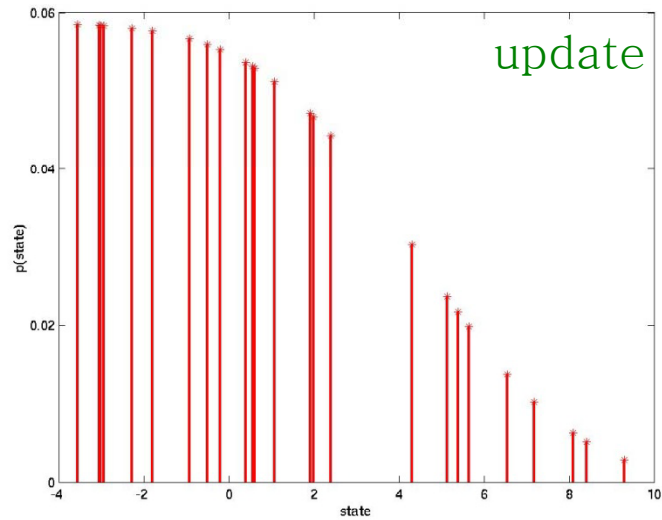
update, t <= 2

SIMULATION: TIME = 3
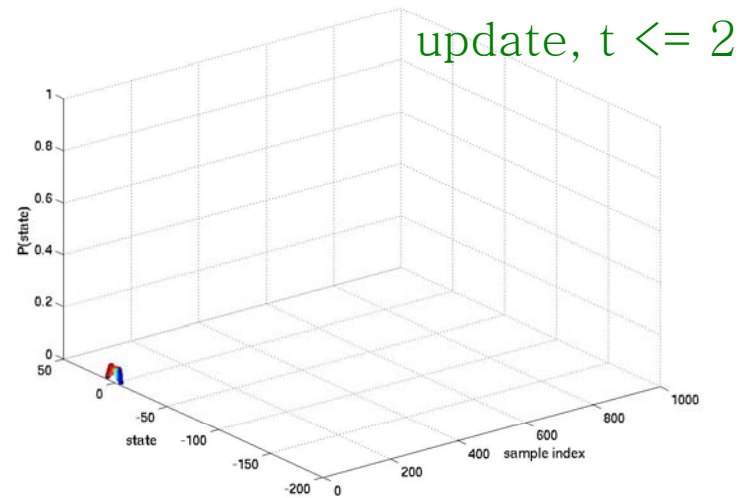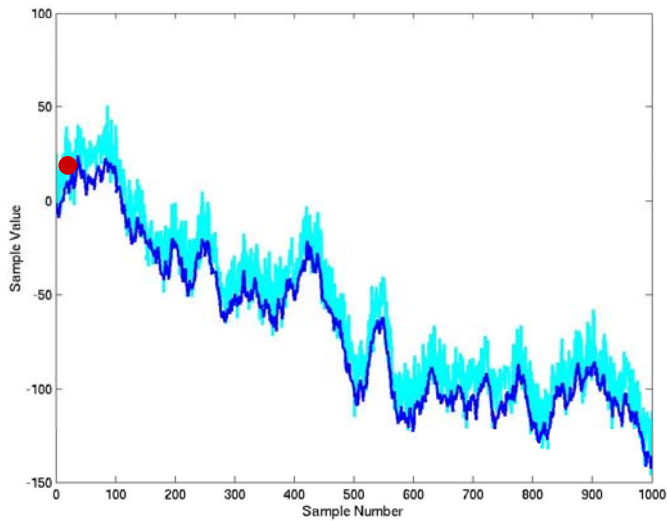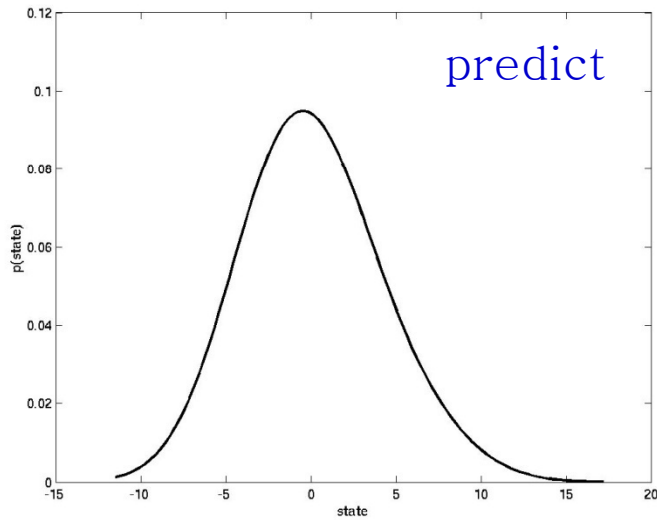
update, t <= 2
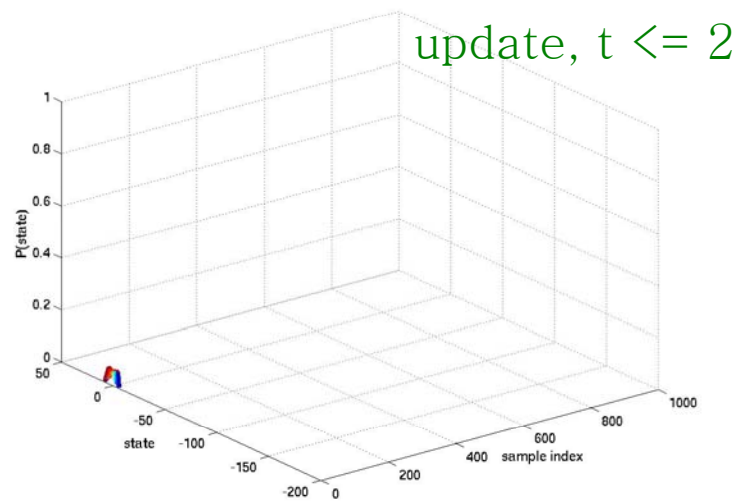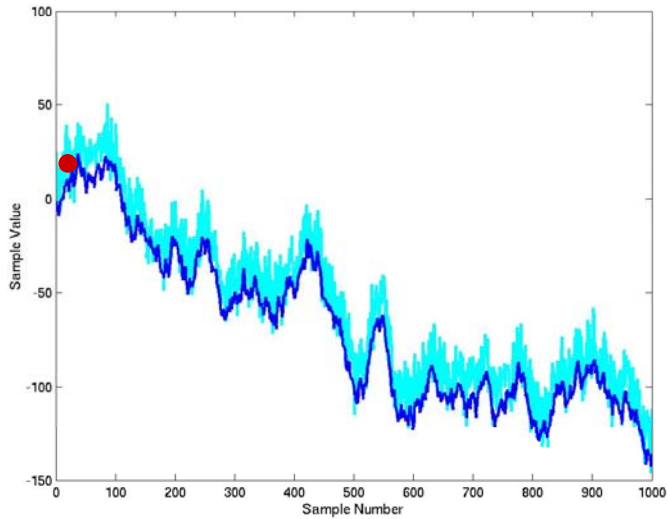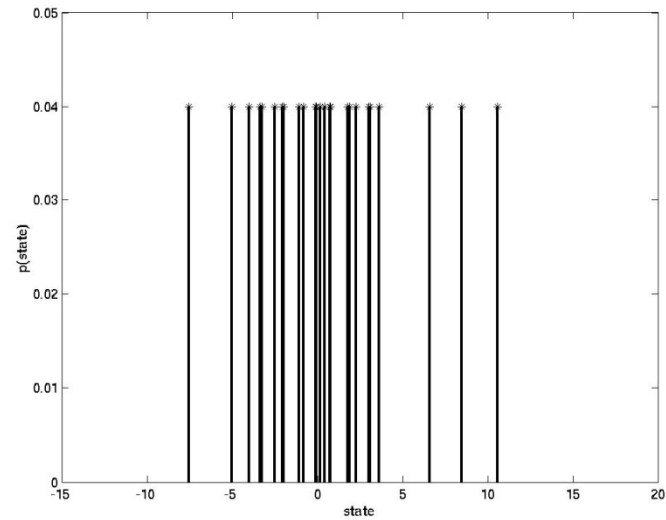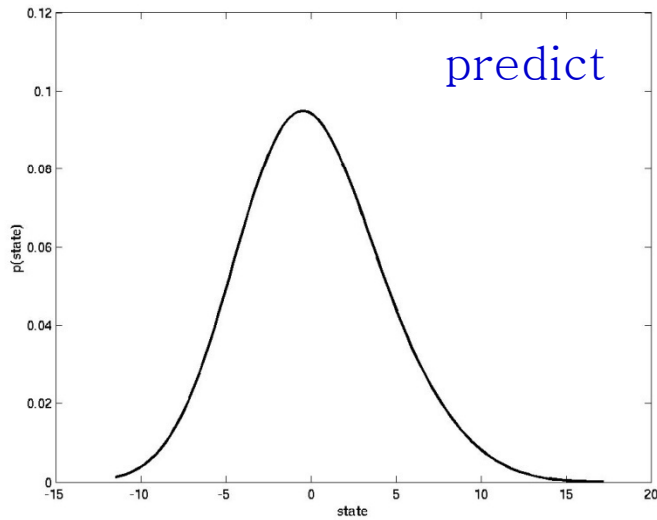
SIMULATION: TIME = 3
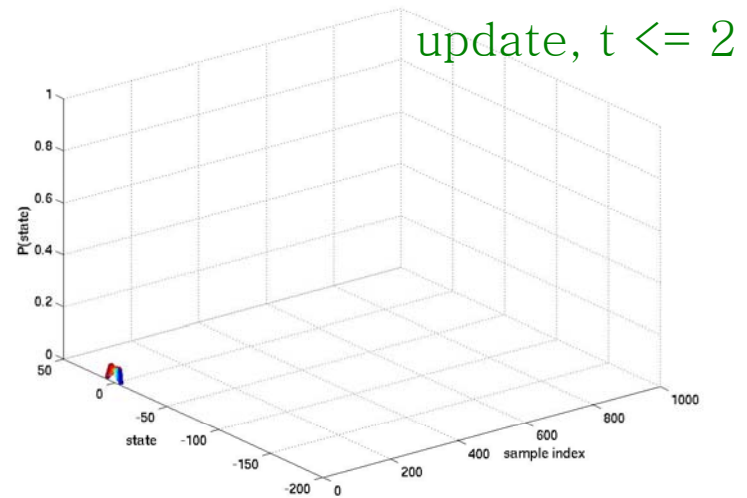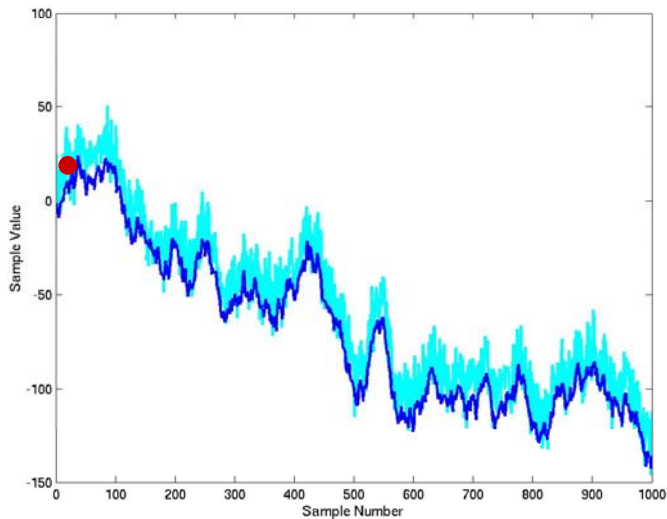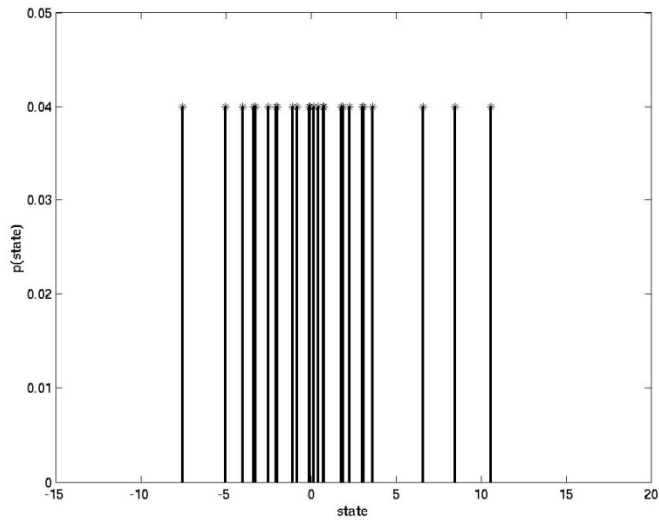
update

update, t <= 2
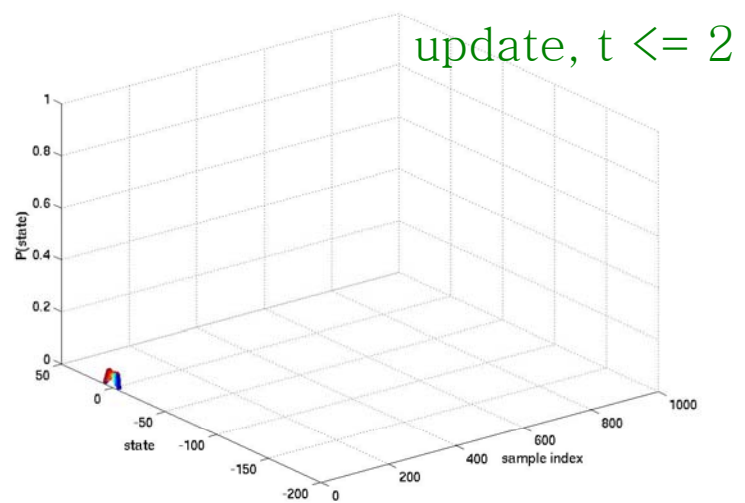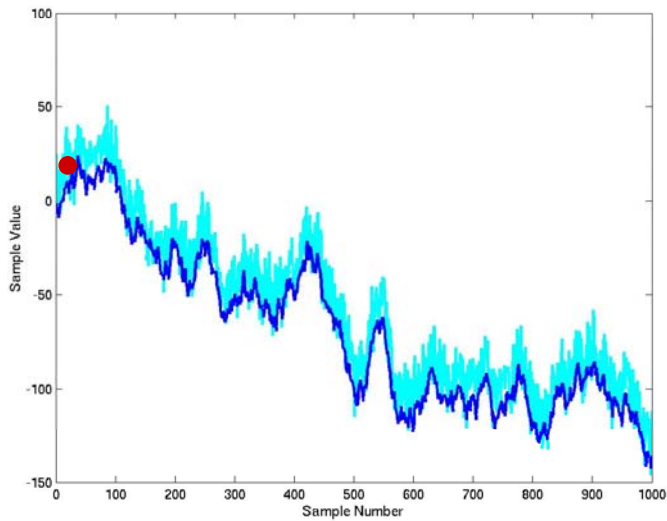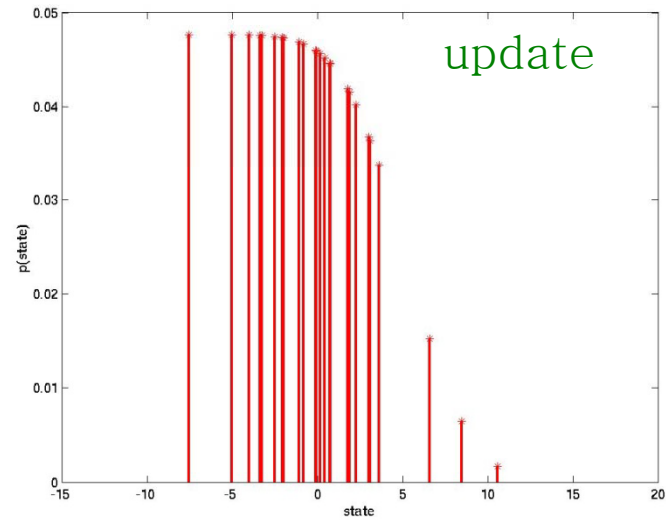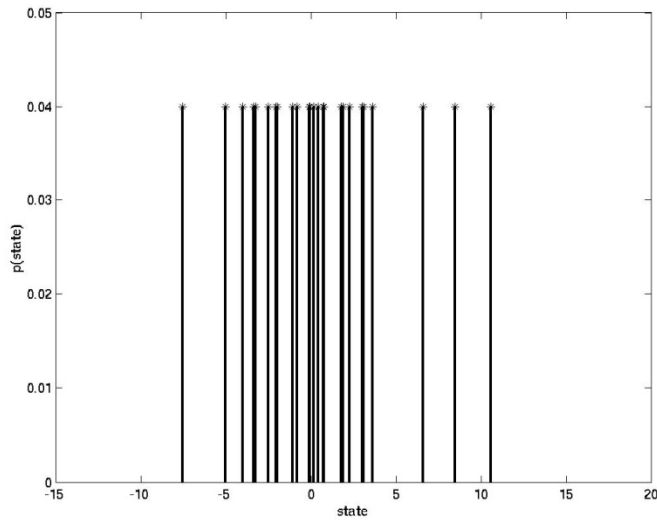
SIMULATION: TIME = 3
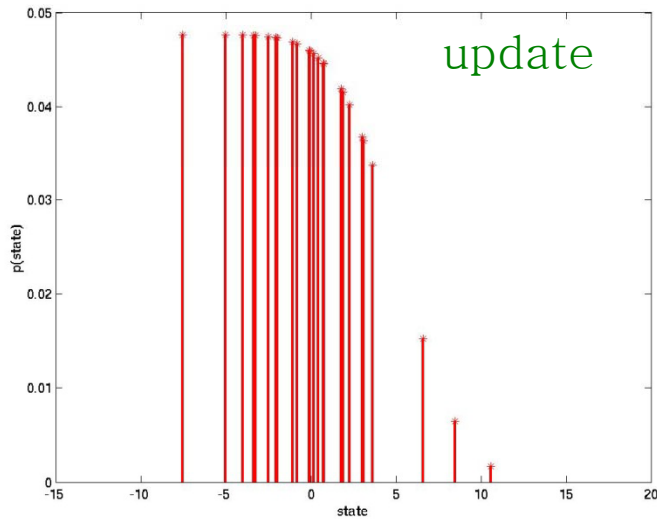
update

update, t <= 3

The figure below shows the contour of the updated state probabilities for all time instants until the current instant
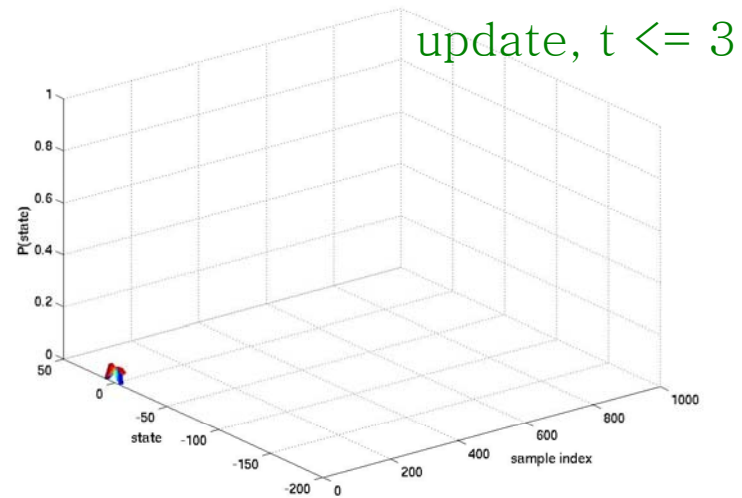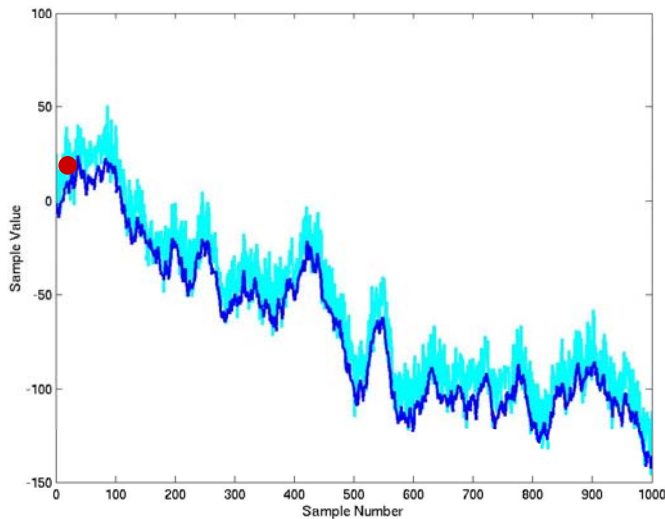
Simulation: Updated Probs Until T=3

update, t <= 3
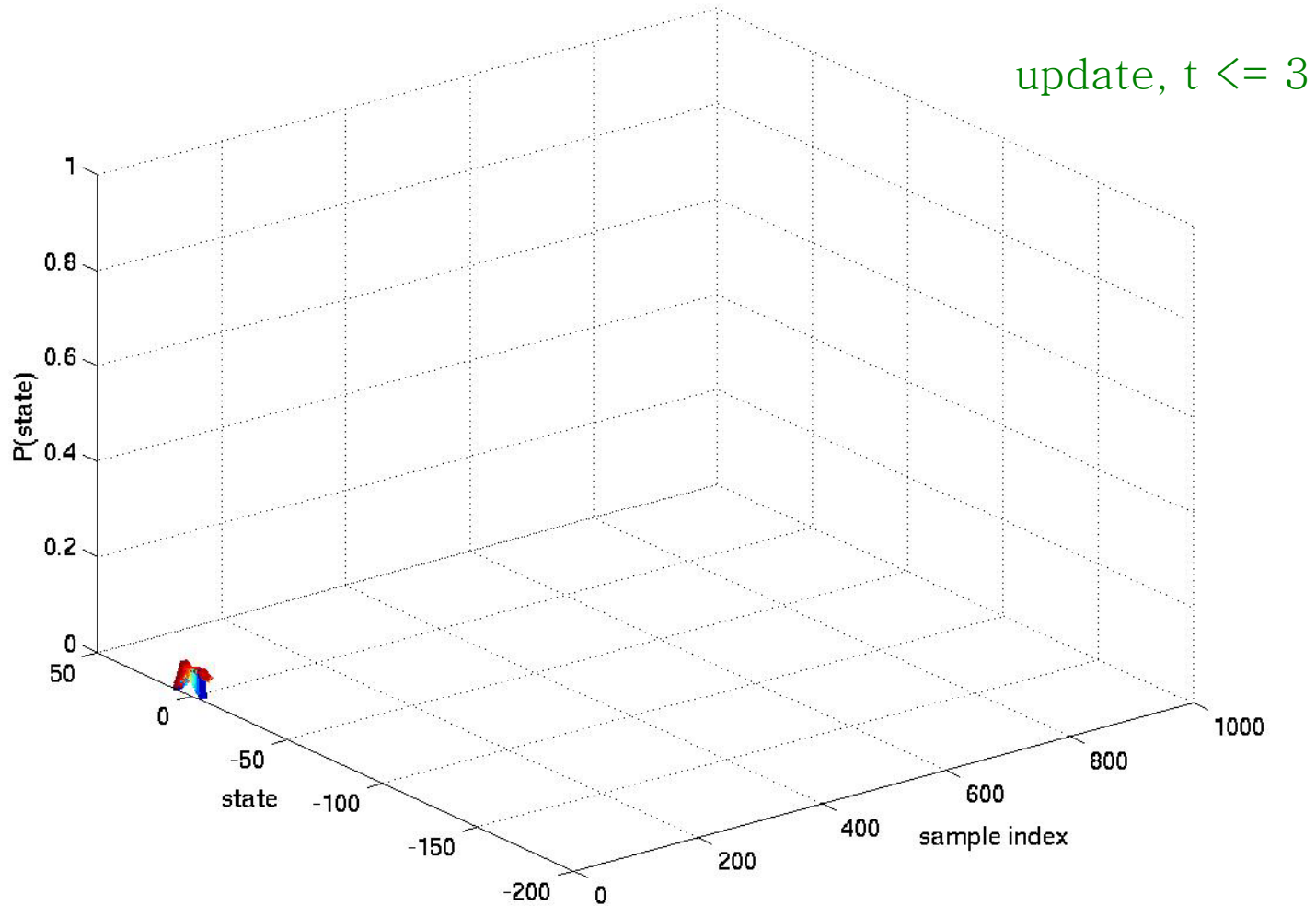
Simulation: Updated Probs Until T=100

update, t <= 100
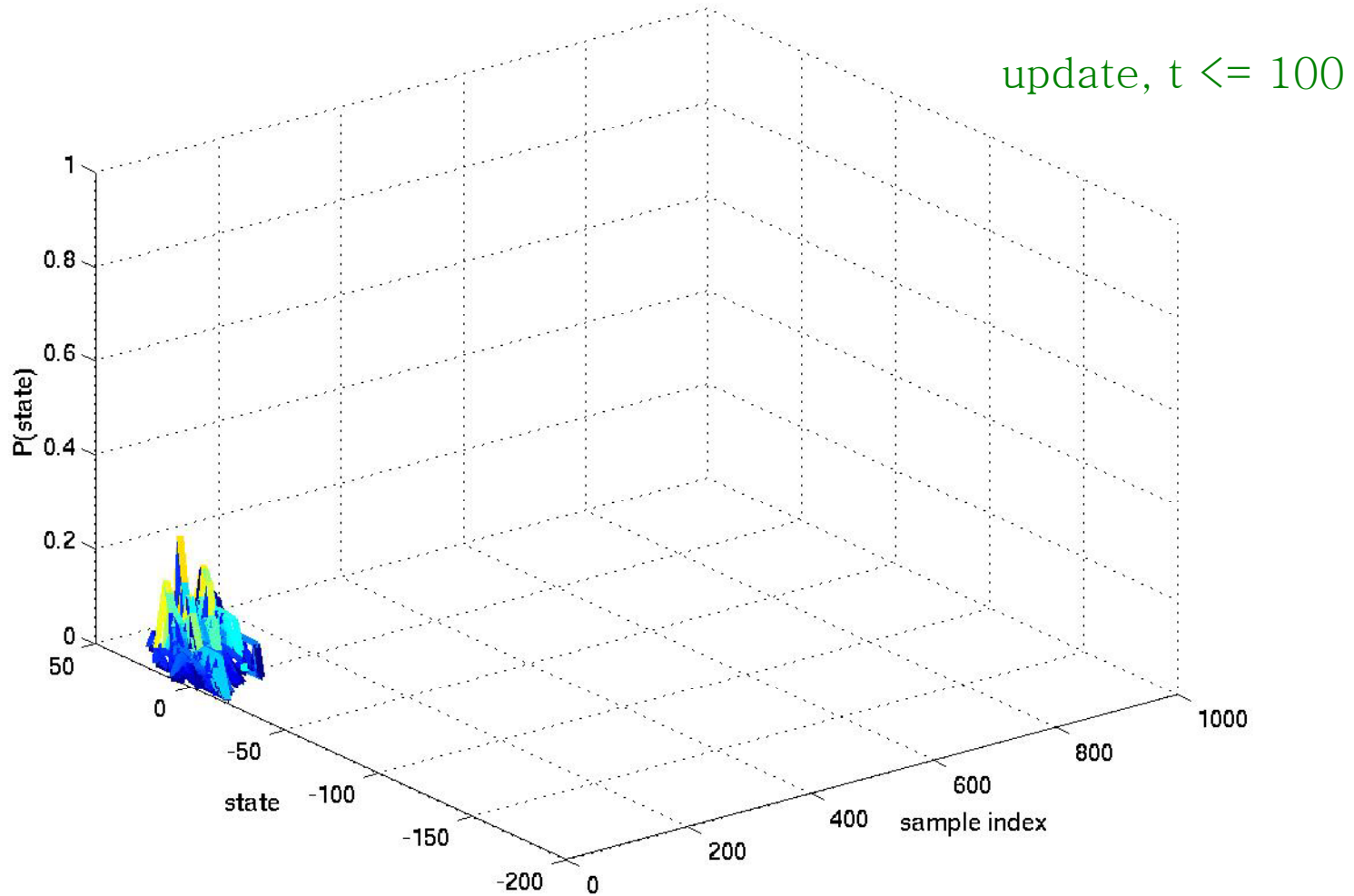
Simulation: Updated Probs Until T=200

update, t <= 200
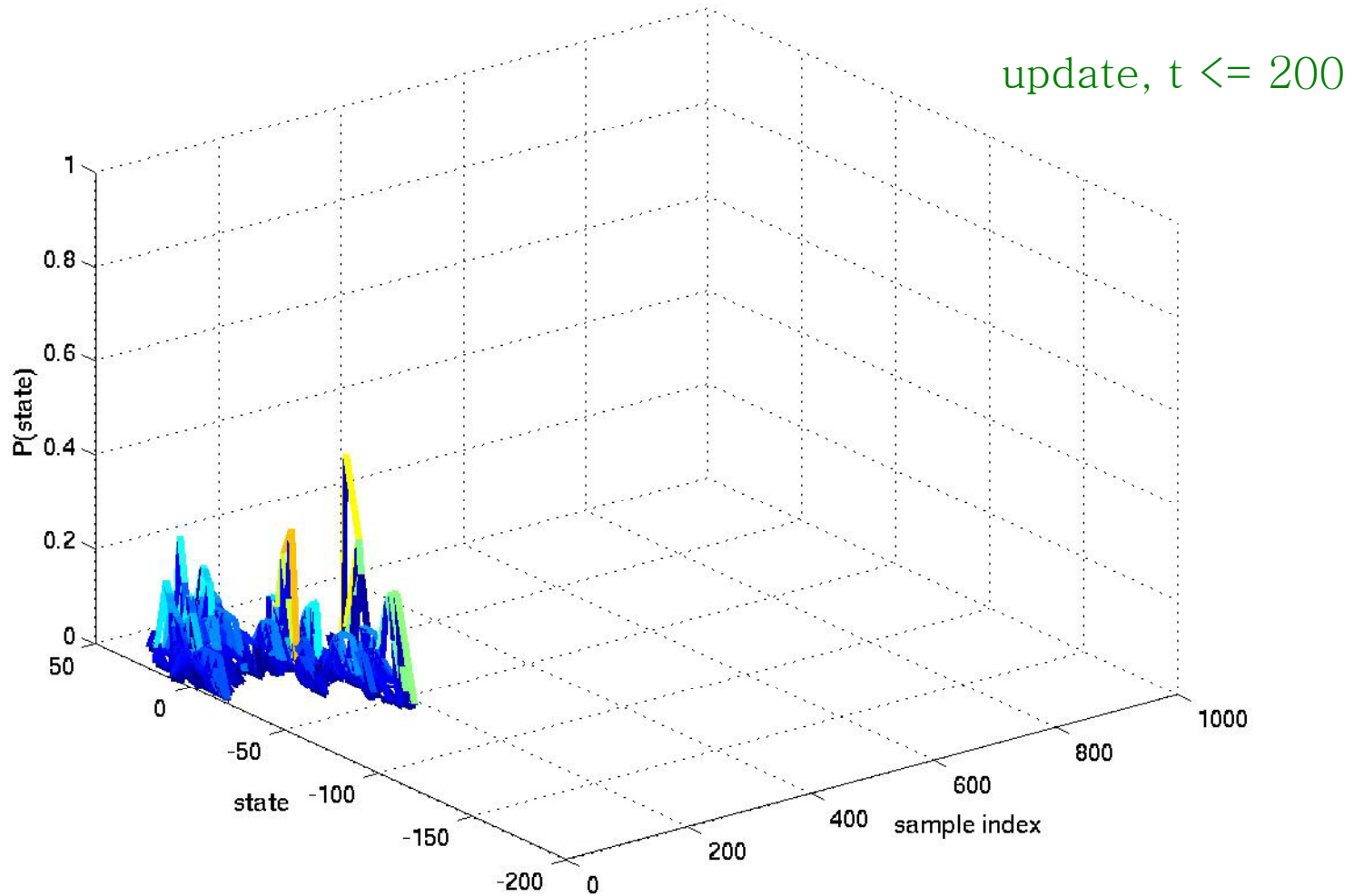
Simulation: Updated Probs Until T=300
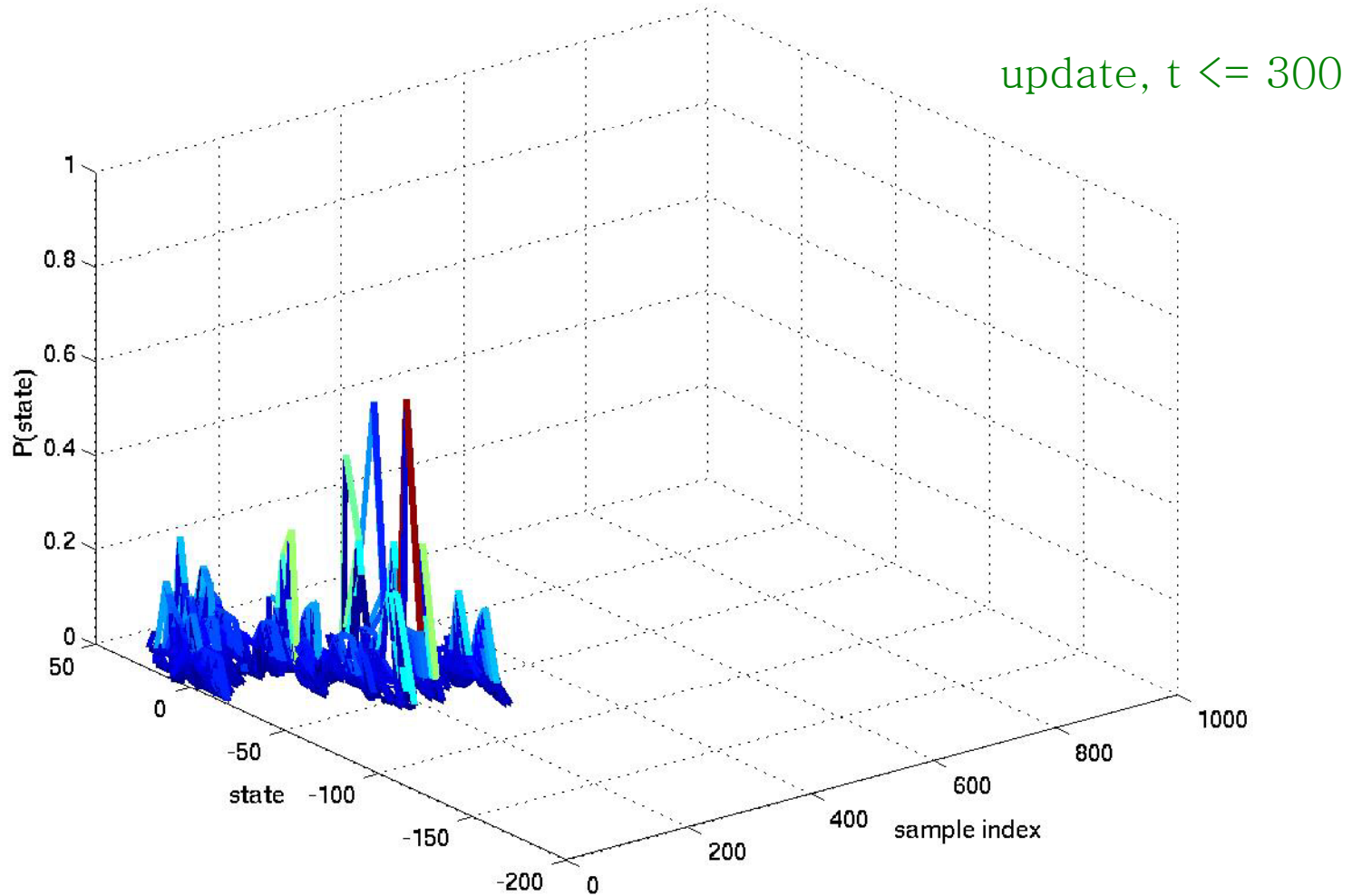
update, t <= 300

Simulation: Updated Probs Until T=500
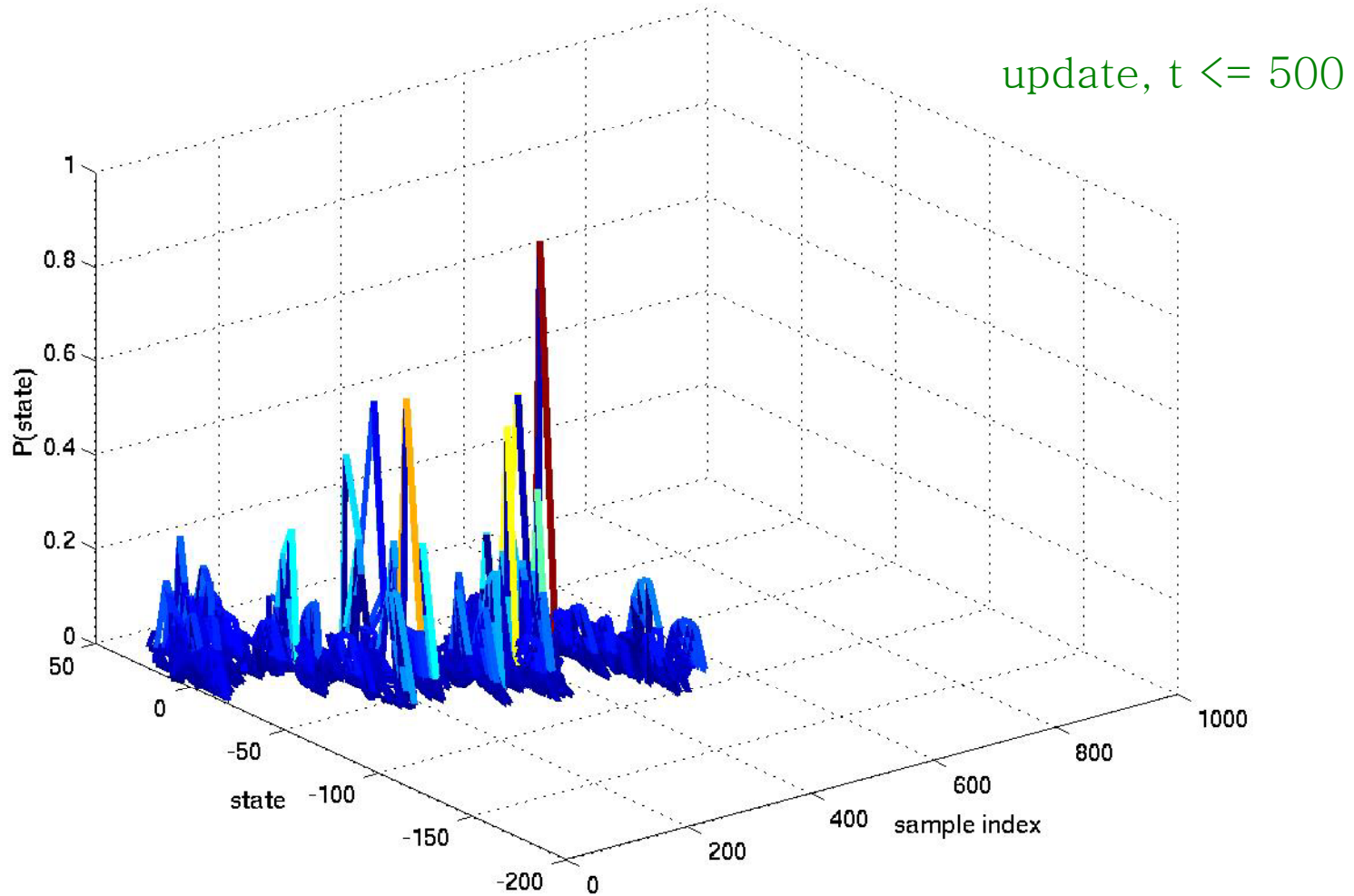
update, t <= 500
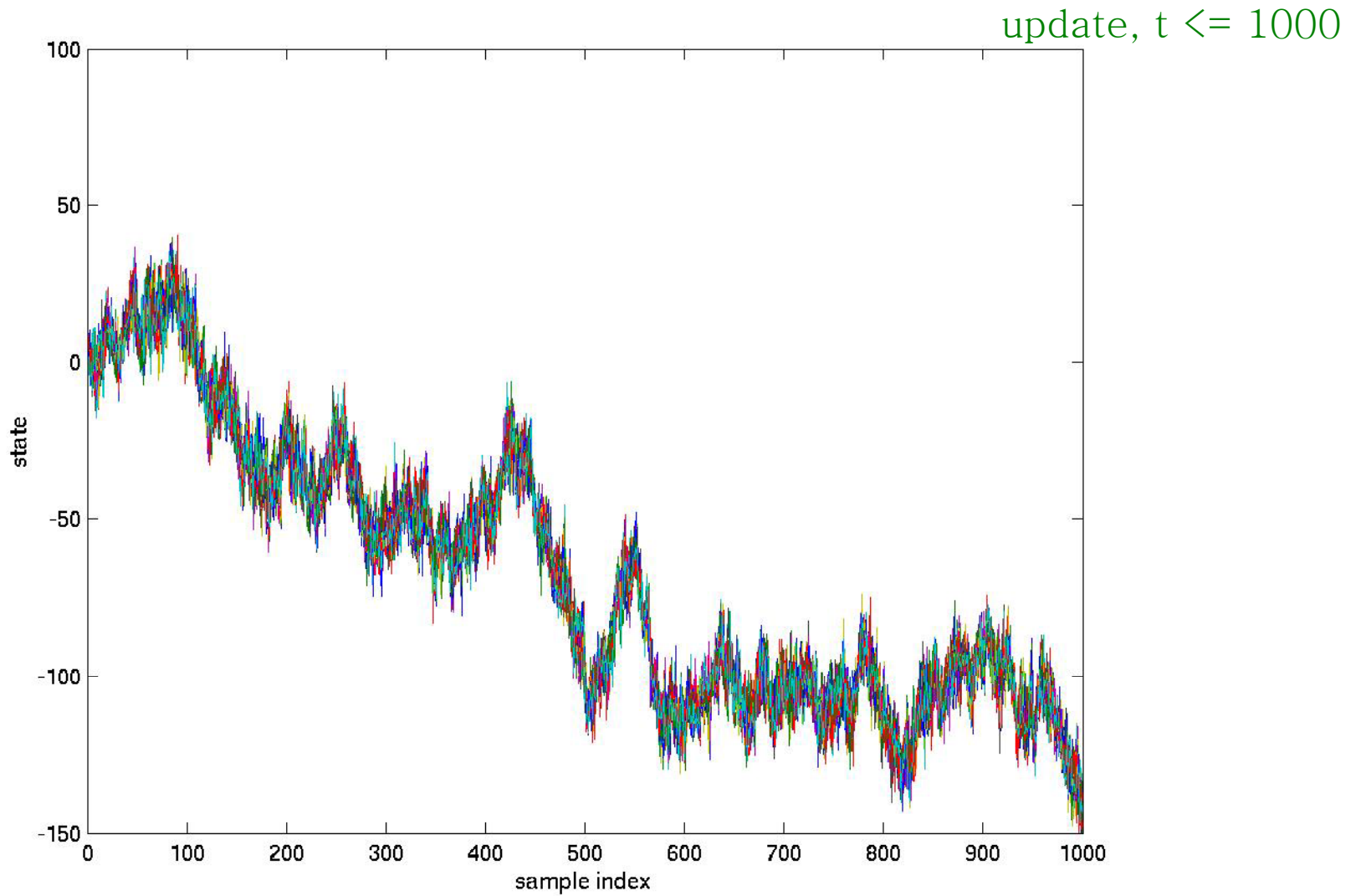
Simulation: Updated Probs Until T=1000
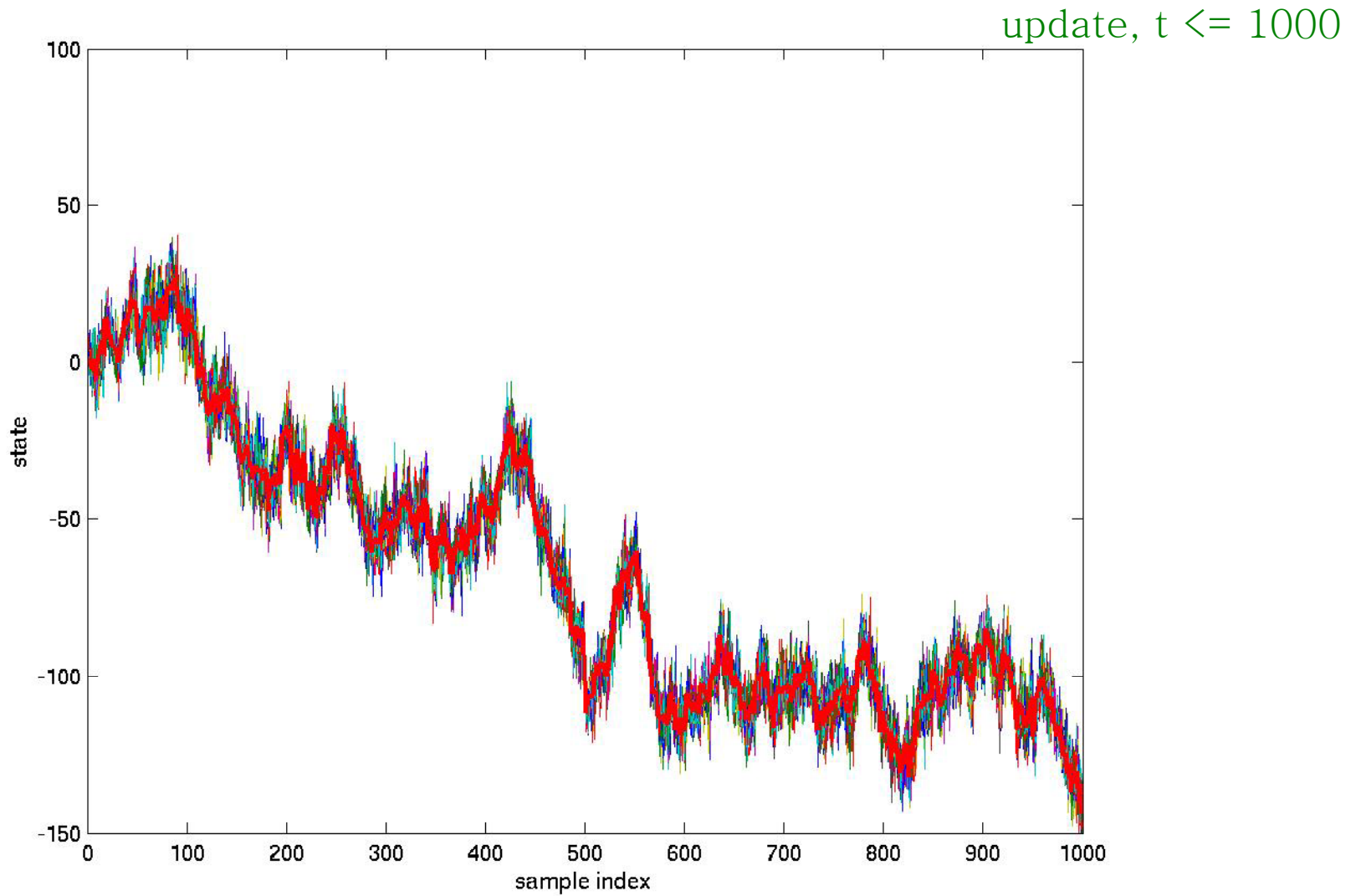
update, t <= 1000
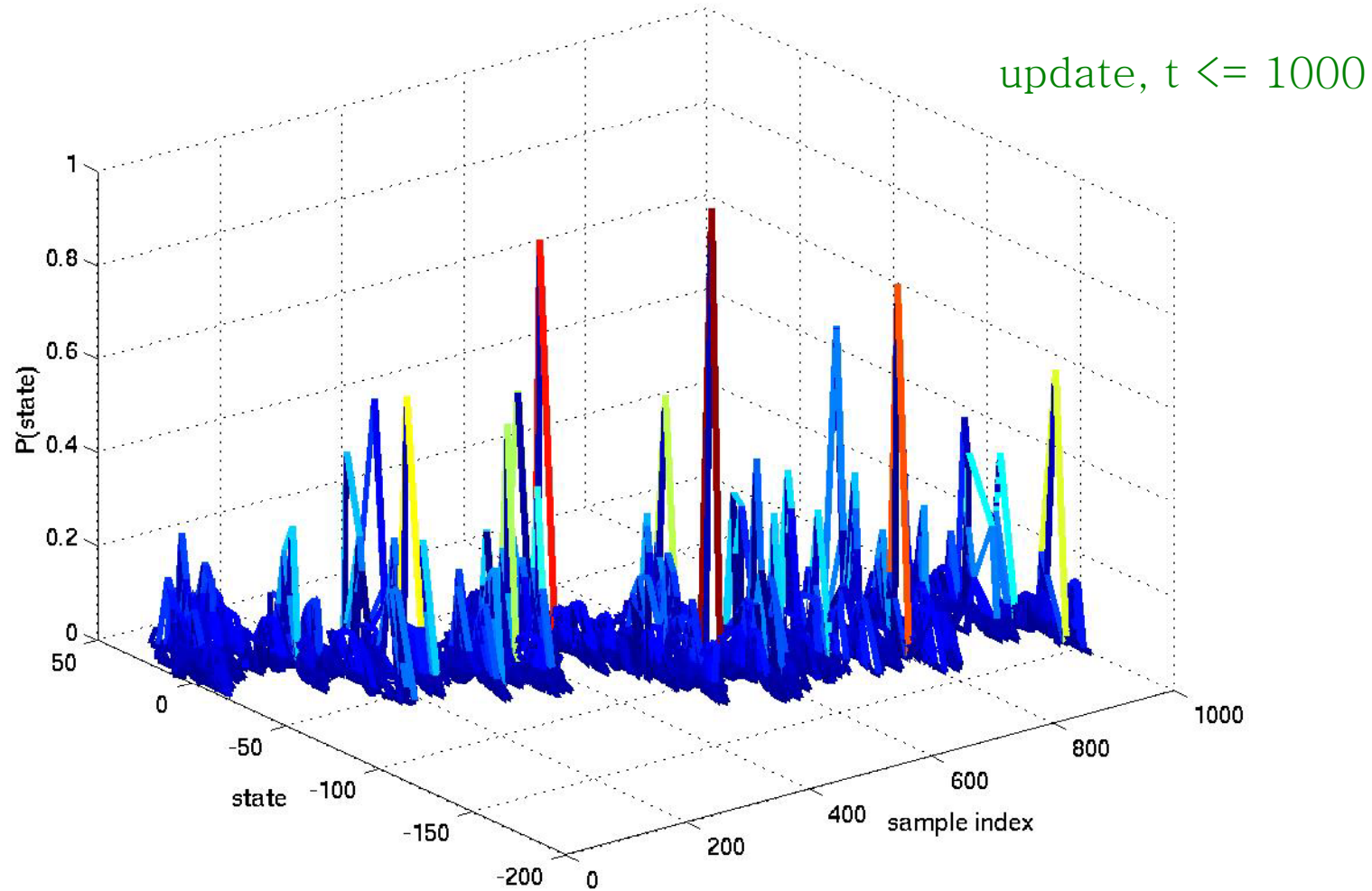
Updated Probs Until T = 1000
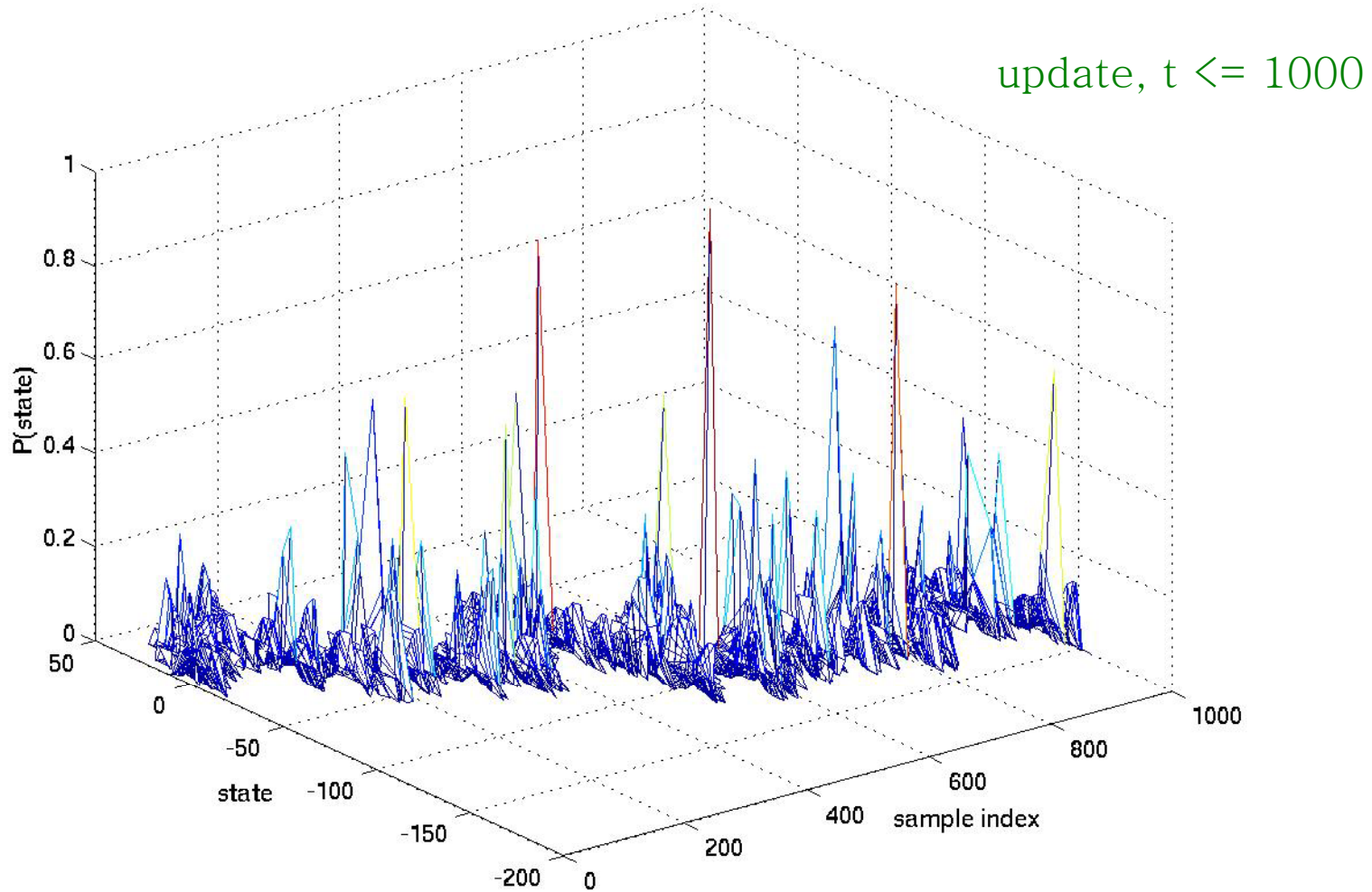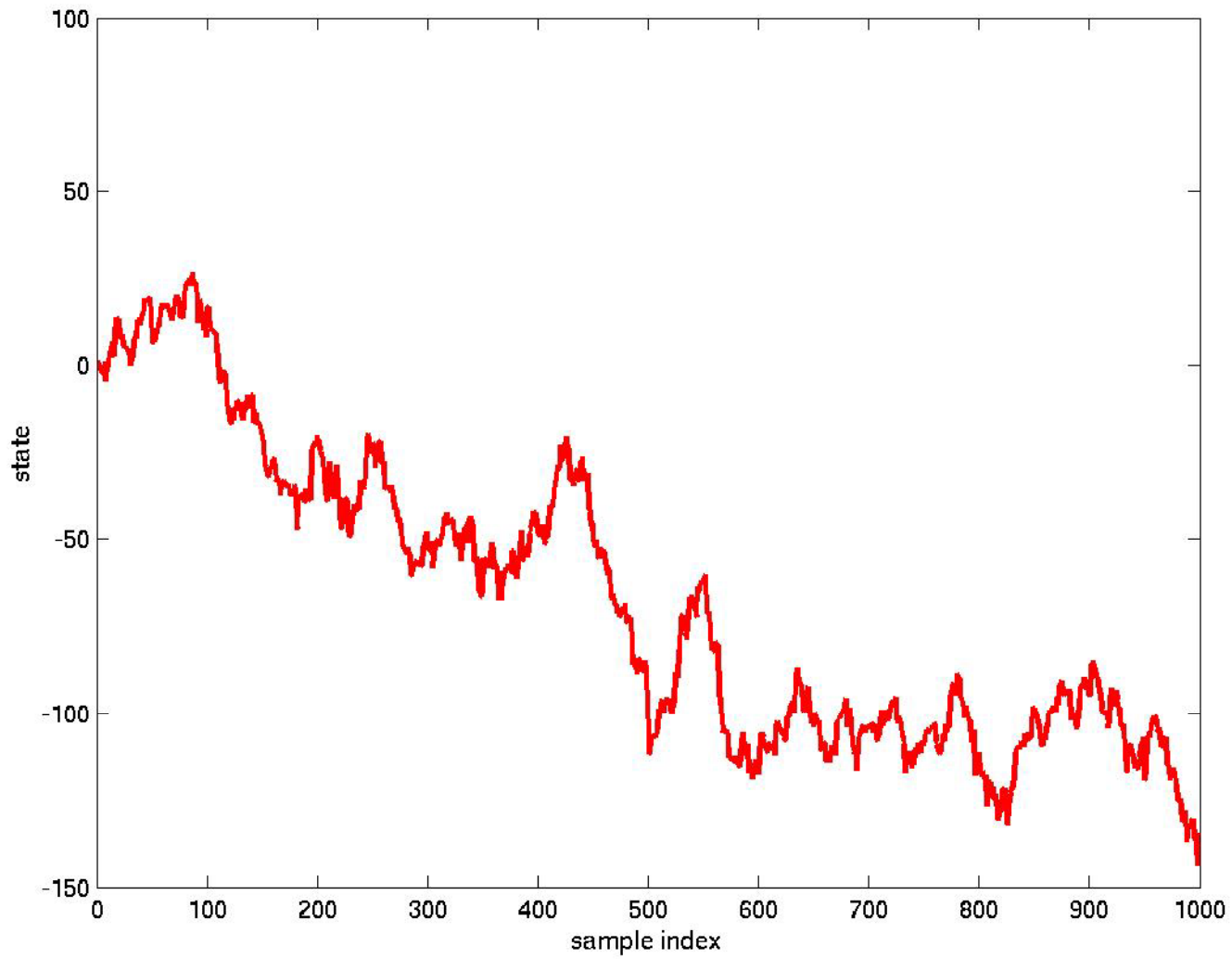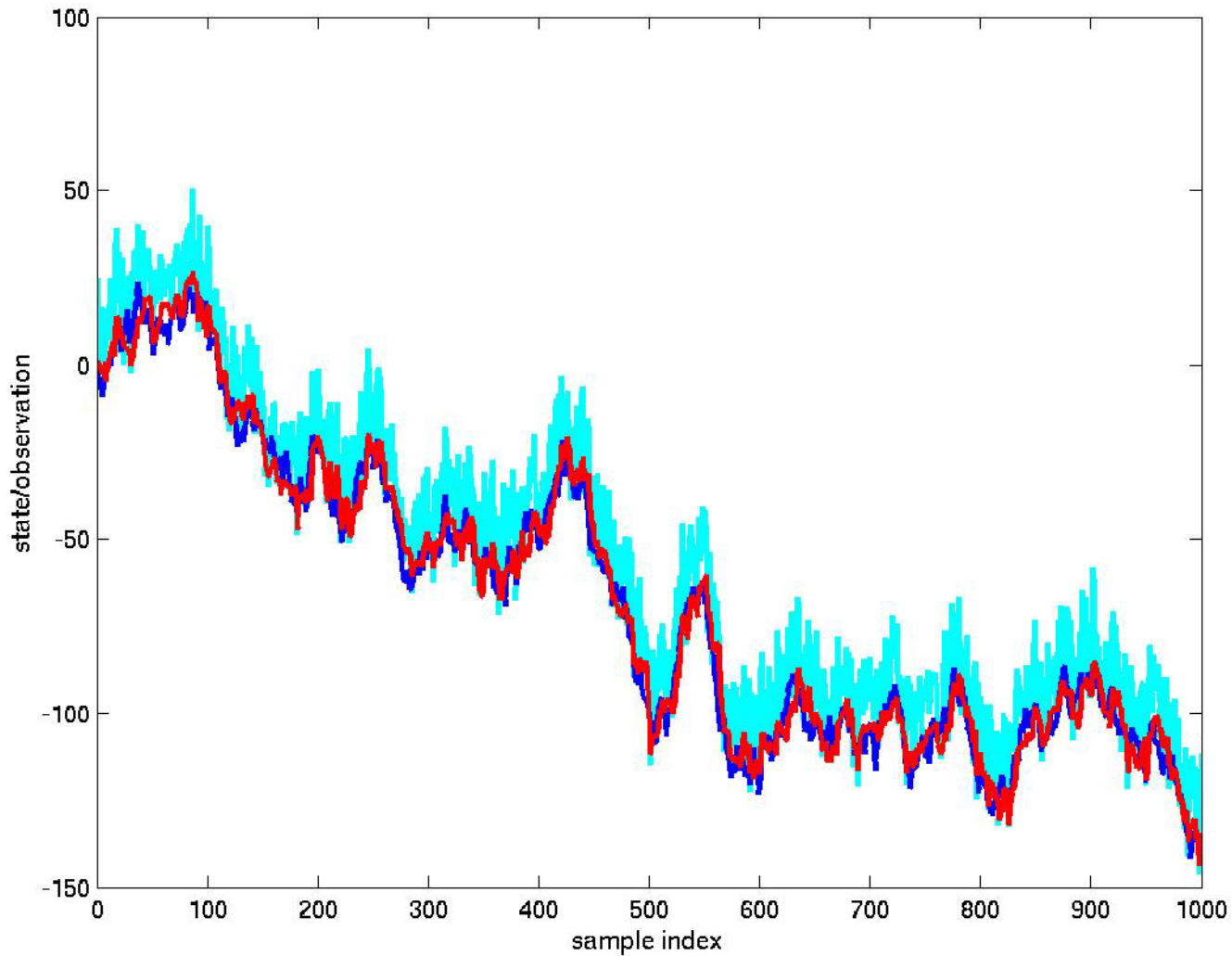
update, t <= 1000

Updated Probs Until T = 1000

Updated Probs: Top View

ESTIMATED STATE

Observation, True States, Estimate

Particle Filtering

- **Generally quite effective in scenarios where EKF/UKF may not be applicable**
  - Potential applications include tracking and edge detection in images!
  - Not very commonly used however

- **Highly dependent on sampling**
  - A large number of samples required for accurate representation
  - Samples may not represent mode of distribution
  - Some distributions are not amenable to sampling
    - Use importance sampling instead: Sample a Gaussian and assign non-uniform weights to samples

Additional Topics

- NMF / ICA.
- Class on 30 Nov?