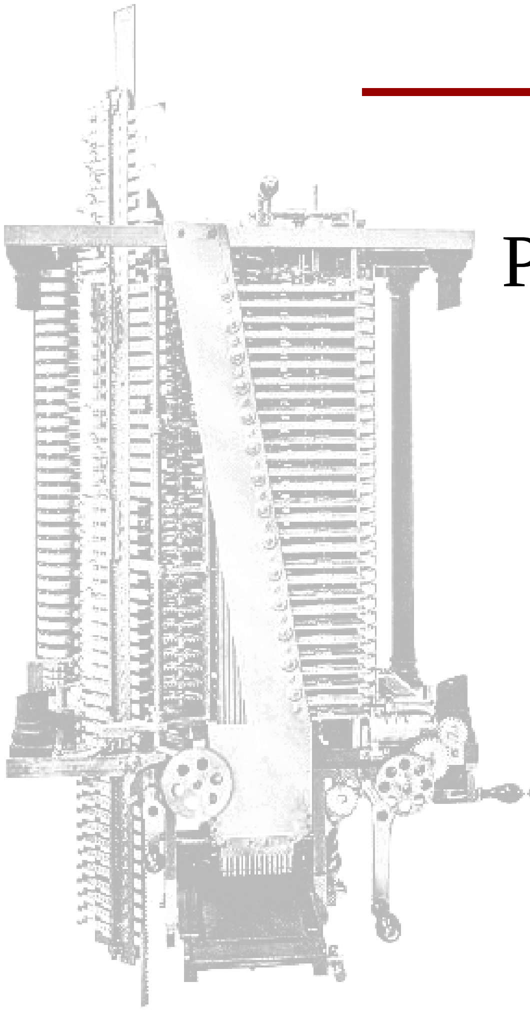
Topics in Brain Computer Interfaces
CS295-7



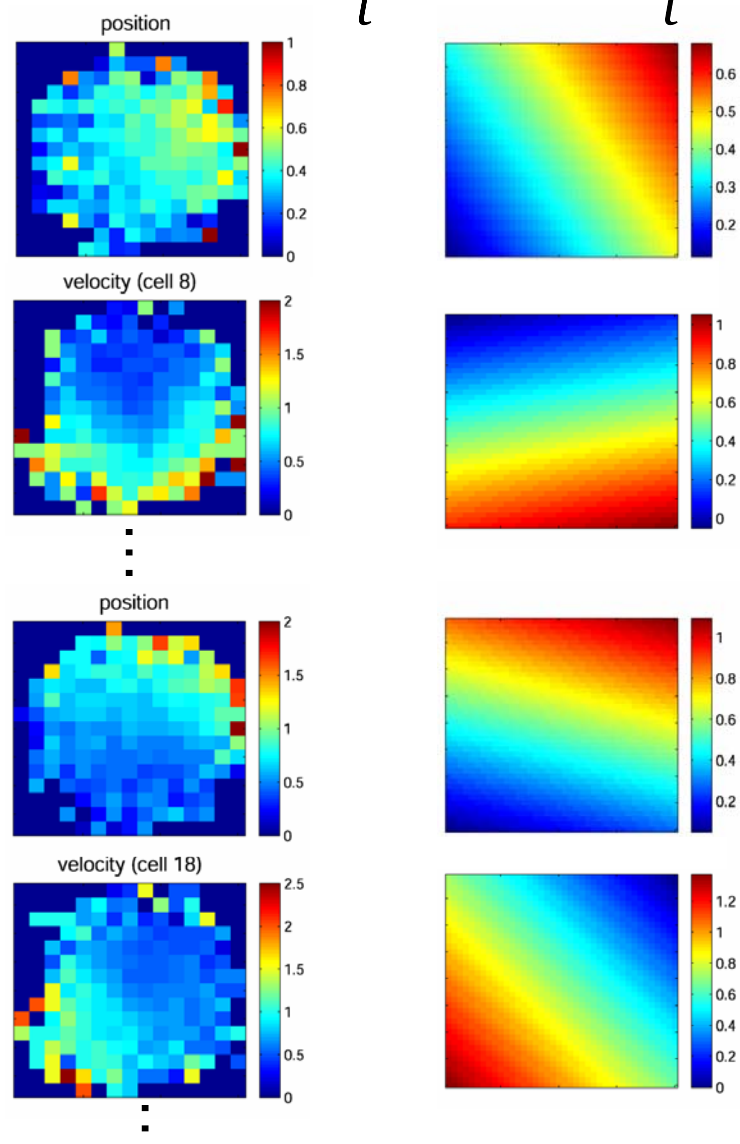
Professor: **MICHAEL BLACK**

TA: **FRANK WOOD**

Spring 2005

Kalman Filtering

$$\vec{z}_t = H\ \vec{x}_t + noise$$



Approximation: Linear Gaussian (generative) model

observation model

$$\vec{z}_{t-j} \sim \mathcal{N}(H\ \vec{x}_t,\ Q_t)$$

Full covariance Q matrix models correlations between cells.

H models how firing rates relate to full kinematic model (position, velocity, and acceleration).

Bayesian Inference

Posterior

Likelihood (evidence)

Prior (*a priori* – before the evidence)

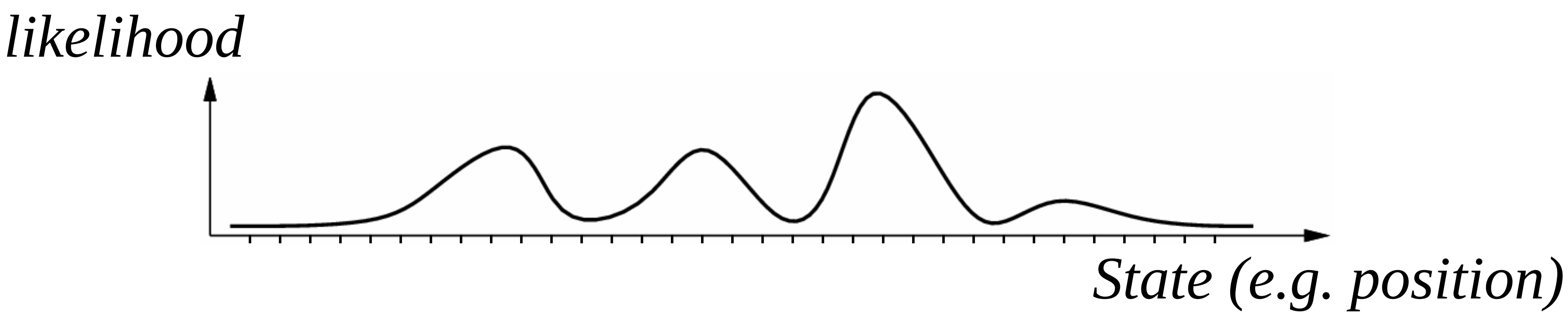
$$p(\text{kinematics} \mid \text{firing}) = \frac{p(\text{firing} \mid \text{kinematics})\, p(\text{kinematics})}{p(\text{firing})}$$

a posteriori probability (after the evidence)

normalization constant (independent of mouth)

We ***infer*** hand kinematics from uncertain evidence and our prior knowledge of how hands move.

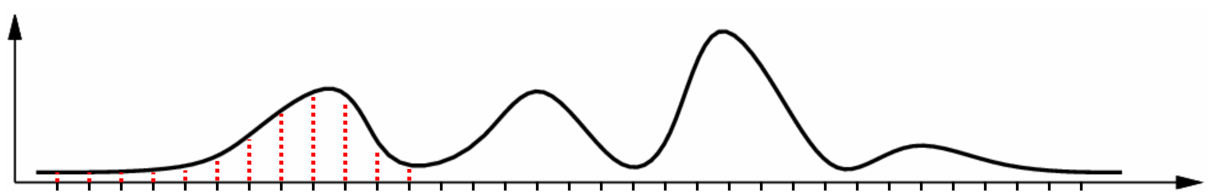
Multi-Modal Likelihood



How can we represent this?

Non-Parametric Approximation

- We could sample at regular intervals



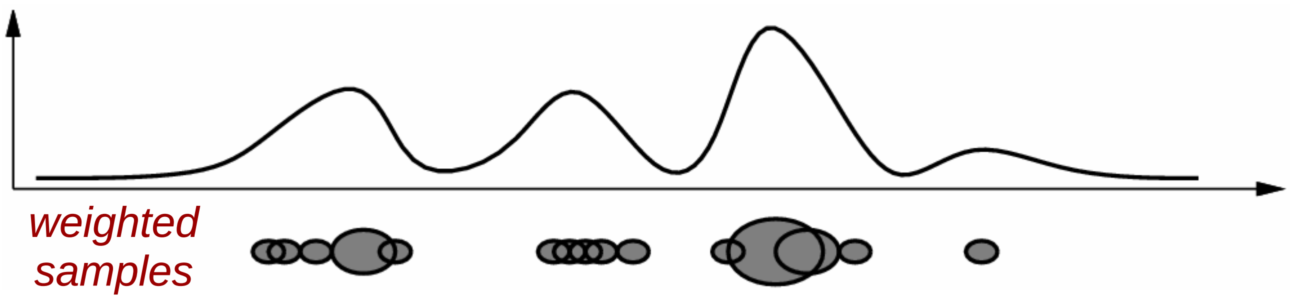
Problems?

most samples have low probability – wasted computation

How finely to discretize

High dimensional space – discretization impractical

Factored Sampling



Weighted samples $S = \{(\mathbf{x}^{(i)}, w^{(i)});\ i = 1 \ldots N\}$

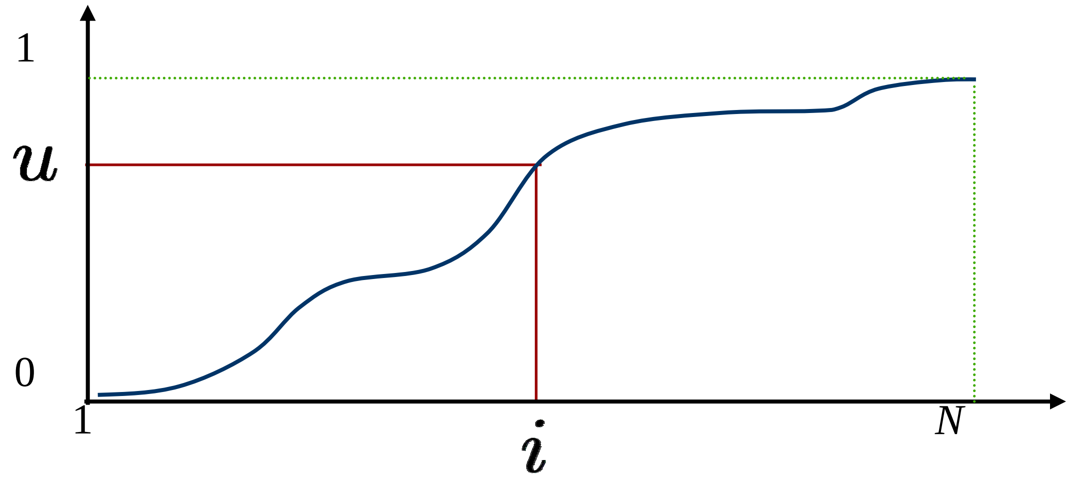
Normalized likelihood: $$w_t^{(n)} = \frac{p(\mathbf{z}_t | \mathbf{x}_t^{(n)})}{\sum_{i=1}^{N} p(\mathbf{z}_t | \mathbf{x}_t^{(i)})}$$

Monte-Carlo Sampling

Given a weighted sample set $S = \{(\mathbf{x}^{(i)}, w^{(i)});\ i = 1 \ldots N\}$

Cumulative distribution of weights

sample
$u \sim U(0,1)$



Bayesian Tracking

Posterior over model parameters given an image sequence.

$$p(\mathbf{x}_t \mid \mathbf{Z}_t) =$$

Temporal model (prior)

$$\kappa\, p(\mathbf{z}_t \mid \mathbf{x}_t) \int (p(\mathbf{x}_t \mid \mathbf{x}_{t-1})\, p(\mathbf{x}_{t-1} \mid \mathbf{Z}_{t-1}))\, d\mathbf{x}_{t-1}$$

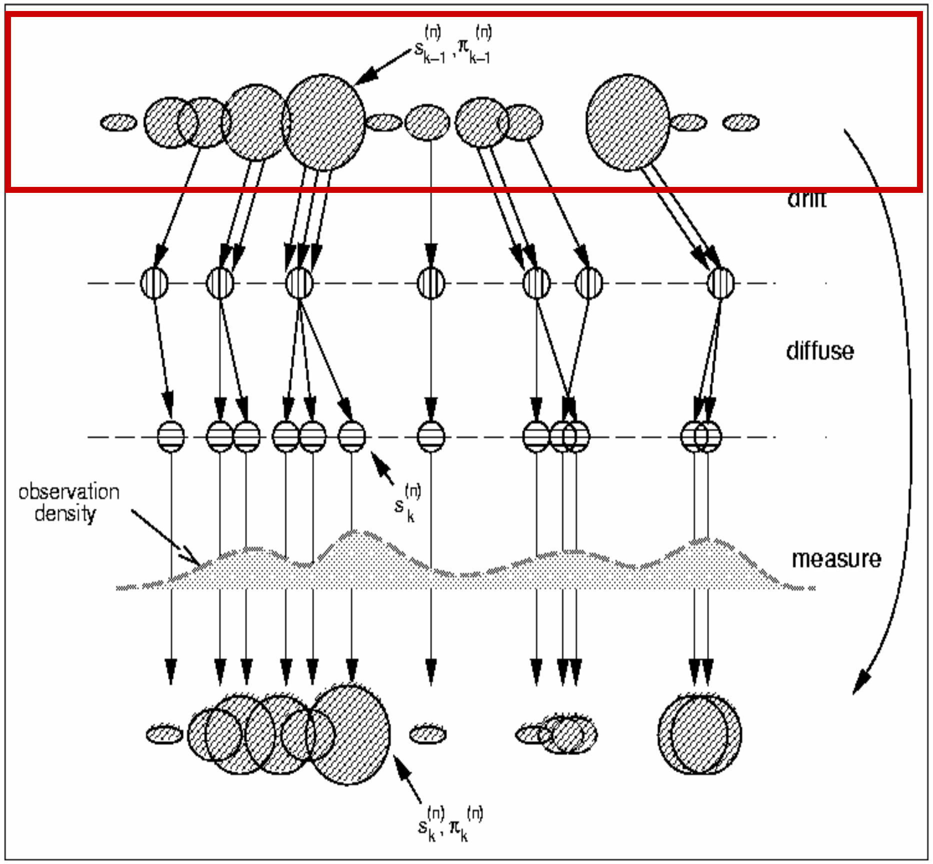
Posterior from previous time instant

Likelihood of observing the firing rates given the hand kinematics.

Monte Carlo integration

Particle Filter

Posterior $p(\mathbf{x}_{t-1} \mid \mathbf{Z}_{t-1})$

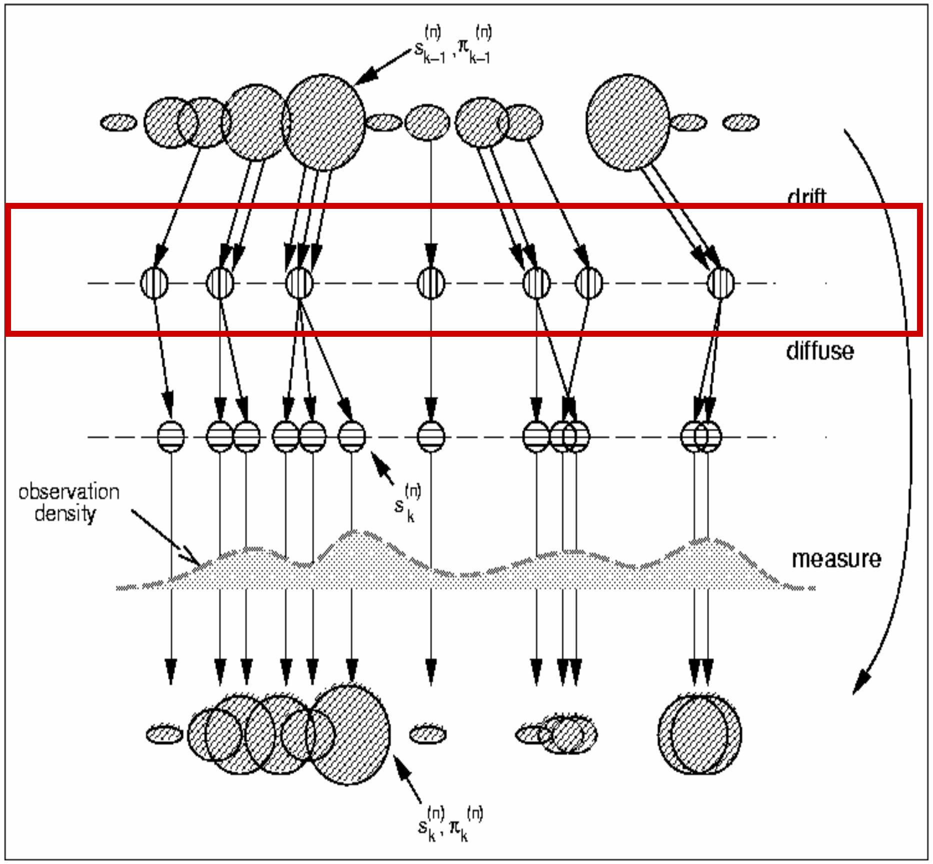


Isard & Blake '96

Particle Filter

Posterior $p(\mathbf{x}_{t-1} \mid \mathbf{Z}_{t-1})$

sample ↓



Isard & Blake '96

Particle Filter

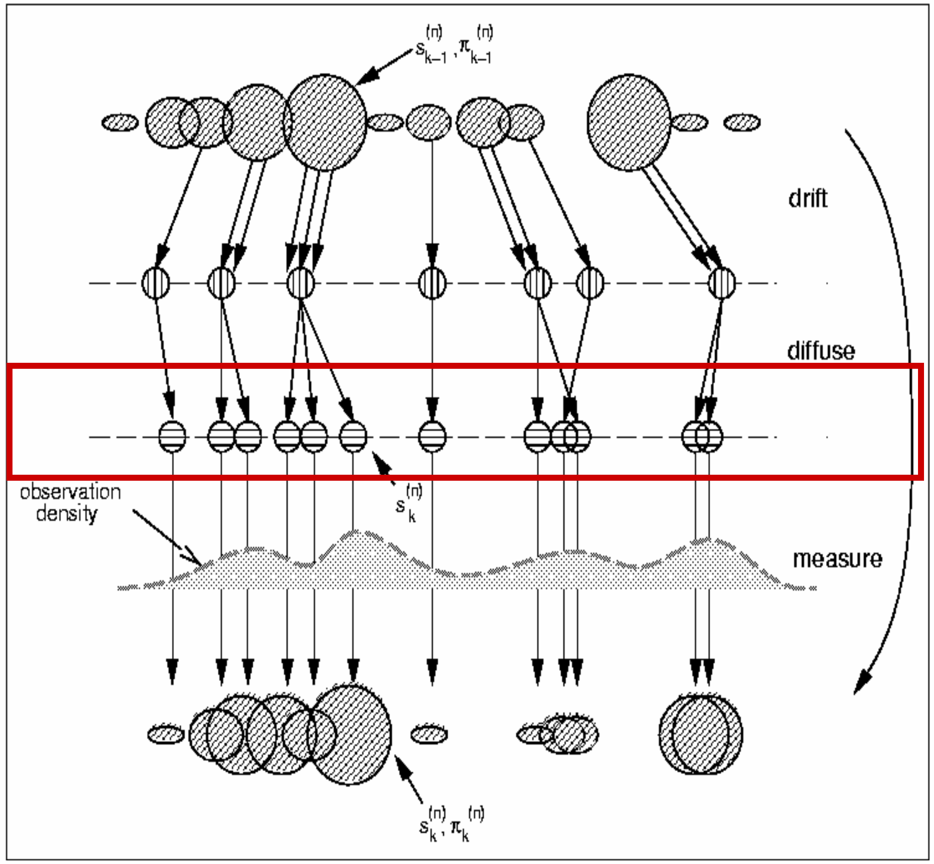
Posterior $p(\mathbf{x}_{t-1} \mid \mathbf{Z}_{t-1})$

sample

Temporal dynamics

$p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$

sample



Isard & Blake '96

Particle Filter

Posterior $p(\mathbf{x}_{t-1} \mid \mathbf{Z}_{t-1})$

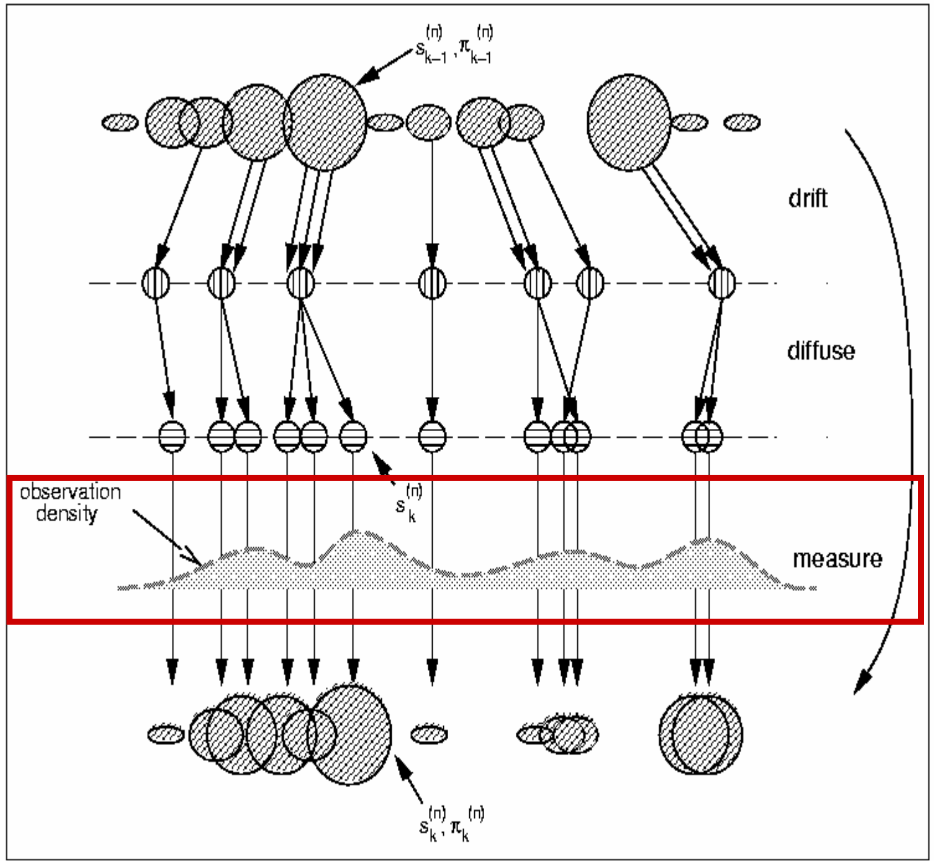
sample

Temporal dynamics

$p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$

sample

Likelihood $p(\mathbf{z}_t \mid \mathbf{x}_t)$



Isard & Blake '96

Particle Filter

Posterior $p(\mathbf{x}_{t-1} \mid \mathbf{Z}_{t-1})$

sample

Temporal dynamics

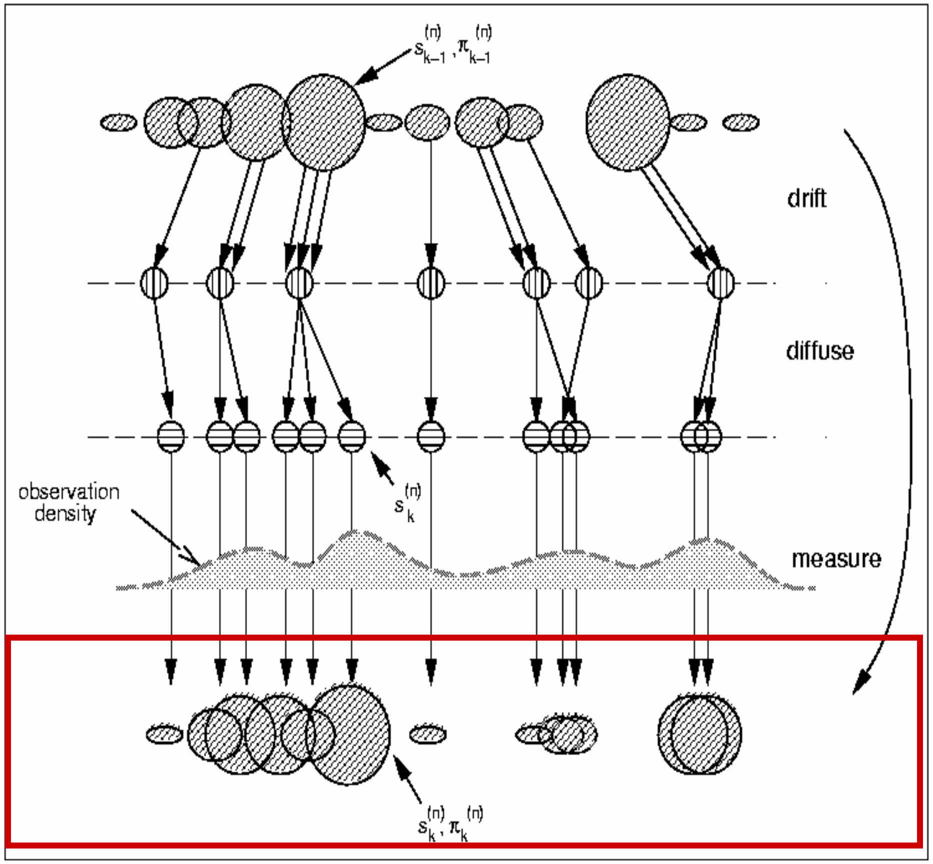
$p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$

sample

Likelihood $p(\mathbf{z}_t \mid \mathbf{x}_t)$

normalize

Posterior $p(\mathbf{x}_t \mid \mathbf{Z}_t)$

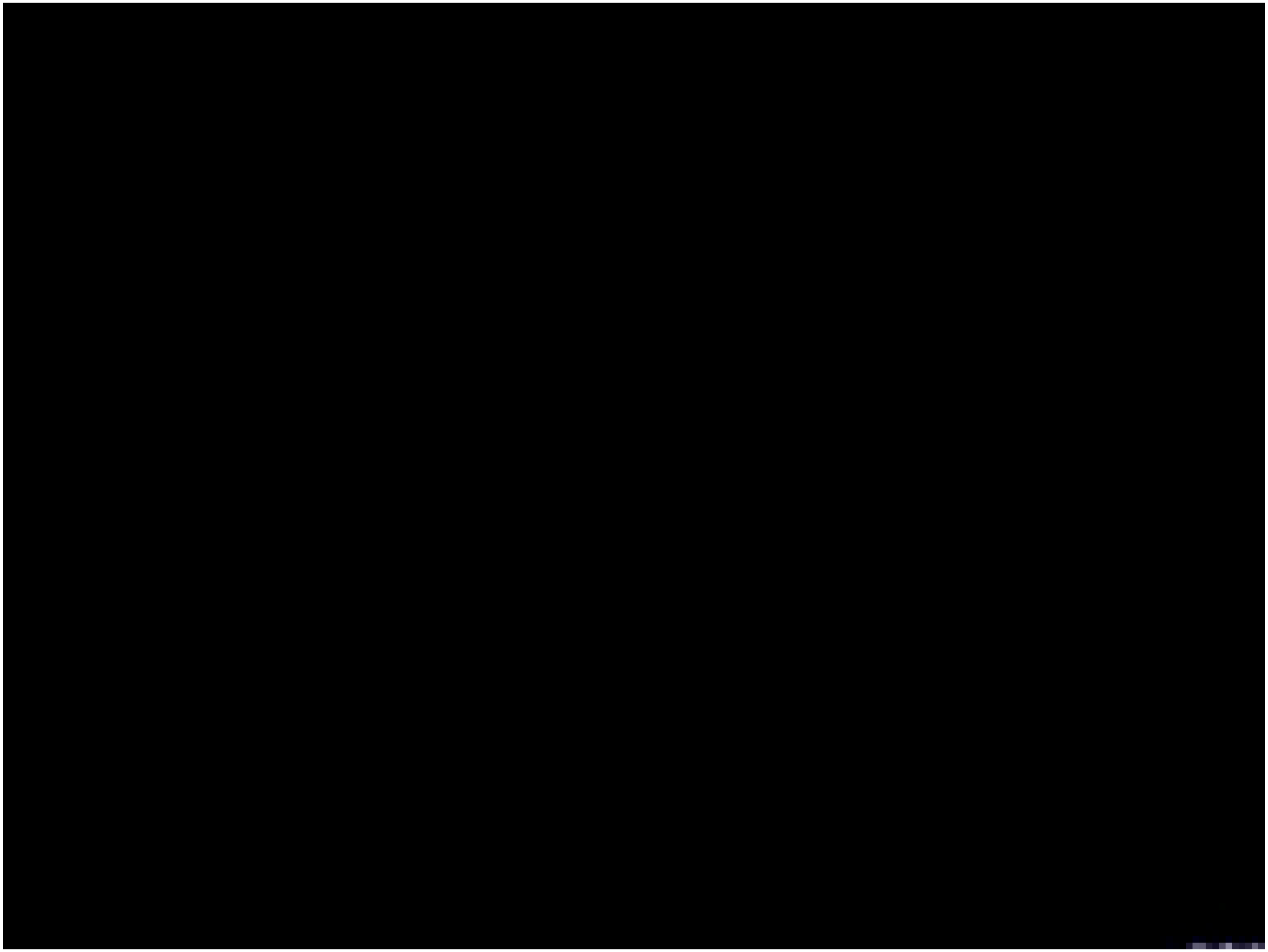


Isard & Blake '96

Pseudocode

condense1step

- % generate cumulative distribution for posterior at t-1
- ….
- % generate a vector of uniform random numbers.
- % if a the number is greater than refreshRate then
  - % generate a vector of uniform random numbers
  - % use these to search the cumulative probability
  - % find the indices of the corresponding particles
  - % for each of these particles, predict the new state (eg. Add Gaussian noise!)
  - % for each of these new states compute the log likelihood
- % else generate a particle at random and compute its log likelihood.
- % find the maximum log likelihood and subtract it from all the other log likelihoods
- % construct the posterior at time t by exponentiating all the log likelihoods and normalizing so they sum to 1.



Michael Isard

Linear Gaussian Likelihood

Generative model for the observation:

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{q}_k$$

$$\mathbf{H}_k \in \mathbf{R}^{c \times d}, \quad \mathbf{Q}_k \in \mathbf{R}^{c \times c}, \quad \mathbf{q}_k \sim N(0, \mathbf{Q}_k), \quad k = 1, 2, \cdots, M.$$

Explicit Form

The likelihood model is equivalent to that

$$\mathbf{z}_k \sim N(\mathbf{H}_k\, \mathbf{x}_k, \mathbf{Q}_k)$$

The conditional probability has explicit form:

$$p(\mathbf{z}_k | \mathbf{x}_k) = \frac{1}{((2\pi)^c \det(\mathbf{Q}_k))^{1/2}} \exp(-\frac{1}{2}(\mathbf{z}_k - \mathbf{H}_k\mathbf{x}_k)^T \mathbf{Q}_k^{-1}(\mathbf{z}_k - \mathbf{H}_k\mathbf{x}_k))$$

Linear Gaussian Temporal Prior

Temporal prior of the state:

$$\mathbf{x}_k = \mathbf{A}_k \, \mathbf{x}_{k-1} + \mathbf{w}_k$$

$$\mathbf{A}_k \in \mathbf{R}^{d \times d}, \quad \mathbf{w}_k \sim N(0, \mathbf{W}_k), \quad \mathbf{W}_k \in \mathbf{R}^{d \times d}, \quad k = 2, 3, \cdots M.$$

Explicit Form

The prior model is equivalent to that

$$\mathbf{x}_{k+1} \sim N(\mathbf{A}_k \, \mathbf{x}_k, \mathbf{W}_k)$$

The conditional probability has explicit form:

$$p(\mathbf{x}_{k+1} | \mathbf{x}_k) = \frac{1}{((2\pi)^d \det(\mathbf{W}_k))^{1/2}} \exp(-\frac{1}{2}(\mathbf{x}_{k+1} - \mathbf{A}_k \mathbf{x}_k)^T \mathbf{W}_k^{-1} (\mathbf{x}_{k+1} - \mathbf{A}_k \mathbf{x}_k))$$

Kalman Filter Model

Definition:

System Equation:

$$\mathbf{x}_k = \mathbf{A}_k \mathbf{x}_{k-1} + \mathbf{w}_k, \qquad \mathbf{w}_k \in N(0, \mathbf{W}_k) \quad k = 2, 3, \cdots$$

Measurement Equation:

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{q}_k, \qquad \mathbf{q}_k \in N(0, \mathbf{Q}_k) \quad k = 1, 2, \cdots$$

Assumption:

All random variables have Gaussian distributions and they are linearly related

Linear Gaussian Model

$$p(\mathbf{x}_t \mid \mathbf{Z}_t) =$$

$$\kappa\, p(\mathbf{z}_t \big| \mathbf{x}_t) \int (p(\mathbf{x}_t \,|\, \mathbf{x}_{t-1})\, p(\mathbf{x}_{t-1} \mid \mathbf{Z}_{t-1}))\, d\mathbf{x}_{t-1}$$

Some basic facts about Gaussians:

Marginal of a Gaussian is Gaussian

Gaussian times a Gaussian is Gaussian

$$p_1(\mathbf{x}) = N(\mu_1, \Sigma_1), \quad p_2(\mathbf{x}) = N(\mu_2, \Sigma_2)$$

$$p_1(\mathbf{x})\, p_2(\mathbf{x}) = zN(\mu_3, \Sigma_3)$$

$$\mu_3 = \Sigma_3(\Sigma_1^{-1}\mu_1 + \Sigma_2^{-1}\mu_2)$$

$$\Sigma_3 = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}$$

Linear Gaussian Model

$$p(\mathbf{x}_t \mid \mathbf{Z}_t) =$$

$$\kappa\, p(\mathbf{z}_t \big| \mathbf{x}_t) \int (p(\mathbf{x}_t \mid \mathbf{x}_{t-1})\, p(\mathbf{x}_{t-1} \mid \mathbf{Z}_{t-1}))\, d\mathbf{x}_{t-1}$$

A linear transformation of a Gaussian distributed random variable is also Gaussian:

$$\mathbf{x} \sim N(\overline{\mathbf{x}}, \mathbf{Q})$$

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b}$$

$$\mathbf{y} \sim N(\mathbf{A}\overline{\mathbf{x}} + \mathbf{b}, \mathbf{A}\mathbf{Q}\mathbf{A}^T)$$

Linear Gaussian Model

$$p(\mathbf{x}_t \mid \mathbf{Z}_t) =$$

$$\kappa\, p(\mathbf{z}_t \big| \mathbf{x}_t) \int \big(\underbrace{p(\mathbf{x}_t \mid \mathbf{x}_{t-1})}_{N(\mathbf{A}\mathbf{x}_{t-1}, \mathbf{W})} \underbrace{p(\mathbf{x}_{t-1} \mid \mathbf{Z}_{t-1})}_{N(\bar{\mathbf{x}}_{t-1}, \mathbf{P}_{t-1})} \big)\, d\mathbf{x}_{t-1}$$

$$\mathbf{x}_t \sim N(\mathbf{A}\bar{\mathbf{x}}_{t-1}, \mathbf{A}\mathbf{P}_{t-1}\mathbf{A}^T + \mathbf{W})$$

$$\mathbf{x}_t \sim N(\hat{\mathbf{x}}_t^-, \mathbf{P}_t^-)$$

Linear Gaussian Model

$$p(\mathbf{x}_t \mid \mathbf{Z}_t) =$$

$$\underbrace{\kappa\, p(\mathbf{z}_t \mid \mathbf{x}_t)}_{N(\mathbf{H}\mathbf{x}_t,\mathbf{Q})} \underbrace{\int (p(\mathbf{x}_t \mid \mathbf{x}_{t-1})\, p(\mathbf{x}_{t-1} \mid \mathbf{Z}_{t-1}))\, d\mathbf{x}_{t-1}}_{\mathbf{x}_t \sim N(\hat{\mathbf{x}}_t^-,\mathbf{P}_t^-)}$$

$$const \times \exp\left(-\frac{1}{2}(\mathbf{z}_t - \mathbf{H}\mathbf{x}_t)^T Q^{-1} (\mathbf{z}_t - \mathbf{H}\mathbf{x}_t) - \frac{1}{2}(\mathbf{x}_t - \hat{\mathbf{x}}_t^-)^T \mathbf{P}_t^{-\,-1} (\mathbf{x}_t - \hat{\mathbf{x}}_t^-) \right)$$

Linear Gaussian Model

$$p(\mathbf{x}_t \mid \mathbf{z}_t)$$

$$= const \times \exp\left(-\frac{1}{2}(\mathbf{z}_t - \mathbf{H}\mathbf{x}_t)^T Q^{-1}(\mathbf{z}_t - \mathbf{H}\mathbf{x}_t) - \frac{1}{2}(\mathbf{x}_t - \hat{\mathbf{x}}_t^-)^T \mathbf{P}_t^{-\,-1}(\mathbf{x}_t - \hat{\mathbf{x}}_t^-) \right)$$

$$= N(\hat{\mathbf{x}}_t \mid \mathbf{P}_t)$$

$$\hat{\mathbf{x}}_t = \left(\mathbf{P}_t^{-\,-1} + \mathbf{H}^T \mathbf{Q}^{-1}\mathbf{H}\right)^{-1}\left(\mathbf{P}_t^{-\,-1}\hat{\mathbf{x}}_t^- + \mathbf{H}^T\mathbf{Q}^{-1}\mathbf{z}_t\right)$$

$$\hat{\mathbf{x}}_t = \mathbf{P}_t\left(\mathbf{P}_t^{-\,-1}\hat{\mathbf{x}}_t^- + \mathbf{H}^T\mathbf{Q}^{-1}\mathbf{z}_t\right)$$

$$\mathbf{P}_t = \left(\mathbf{P}_t^{-\,-1} + \mathbf{H}^T \mathbf{Q}^{-1}\mathbf{H}\right)^{-1}$$

Linear Gaussian Model

$$\hat{\mathbf{x}}_t = \left(\mathbf{P}_t^{-\,-1} + \mathbf{H}^T\mathbf{Q}^{-1}\mathbf{H}\right)^{-1}\left(\mathbf{P}_t^{-\,-1}\hat{\mathbf{x}}_t^- + \mathbf{H}^T\mathbf{Q}^{-1}\mathbf{z}_t\right)$$

$$\mathbf{P}_t = \left(\mathbf{P}_t^{-\,-1} + \mathbf{H}^T\mathbf{Q}^{-1}\mathbf{H}\right)^{-1}$$

$\downarrow$ ***Some algebra.***

$$\hat{\mathbf{x}}_t = \hat{\mathbf{x}}_t^- + \mathbf{K}_t(\mathbf{z}_t - \mathbf{H}\hat{\mathbf{x}}_t^-)$$

$$\mathbf{K}_t = \mathbf{P}_t\mathbf{H}^T\mathbf{Q}^{-1}$$

$$\mathbf{K}_t = \left(\mathbf{P}_t^{-\,-1} + \mathbf{H}^T\mathbf{Q}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{Q}^{-1}$$

Simplifying

Matrix inversion lemma

$$\mathbf{K}_t = \underbrace{\left(\mathbf{P}_t^{-\,-1} + \mathbf{H}^T\mathbf{Q}^{-1}\mathbf{H}\right)^{-1}}\mathbf{H}^T\mathbf{Q}^{-1}$$

$$\left(\mathbf{P}_t^{-\,-1} + \mathbf{H}^T\mathbf{Q}^{-1}\mathbf{H}\right)^{-1}$$

$$= \mathbf{P}_t^- - \mathbf{P}_t^-\mathbf{H}^T(\mathbf{Q}^{-1} + \mathbf{H}\mathbf{P}_t^-\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{P}_t^-$$

$$= \mathbf{P}_t$$

KALMAN FILTER ALGORITHM

Time Update

Prior estimate

$$\hat{x}_t^- = A\hat{x}_{t-1}$$

Error covariance

$$P_t^- = AP_{t-1}A^T + W$$

Measurement Update

Posterior estimate

$$\hat{\mathbf{x}}_t = \hat{\mathbf{x}}_t^- + \mathbf{K}_t(\mathbf{z}_t - \mathbf{H}\hat{\mathbf{x}}_t^-)$$

Error covariance

$$\mathbf{P}_t = \mathbf{P}_t^- - \mathbf{P}_t^-\mathbf{H}^T(\mathbf{Q}^{-1} + \mathbf{H}\mathbf{P}_t^-\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{P}_t^-$$

Kalman gain

$$\mathbf{K}_t = \mathbf{P}_t\mathbf{H}^T\mathbf{Q}^{-1}$$

Initial estimate of $\hat{x}_{t-1}$ and P_{t-1}

Welch and Bishop 2002

Learning Kalman Model

- In practice, the parameters in the model need to be estimated from training data. (In training data, we know both hidden states and measurements.)

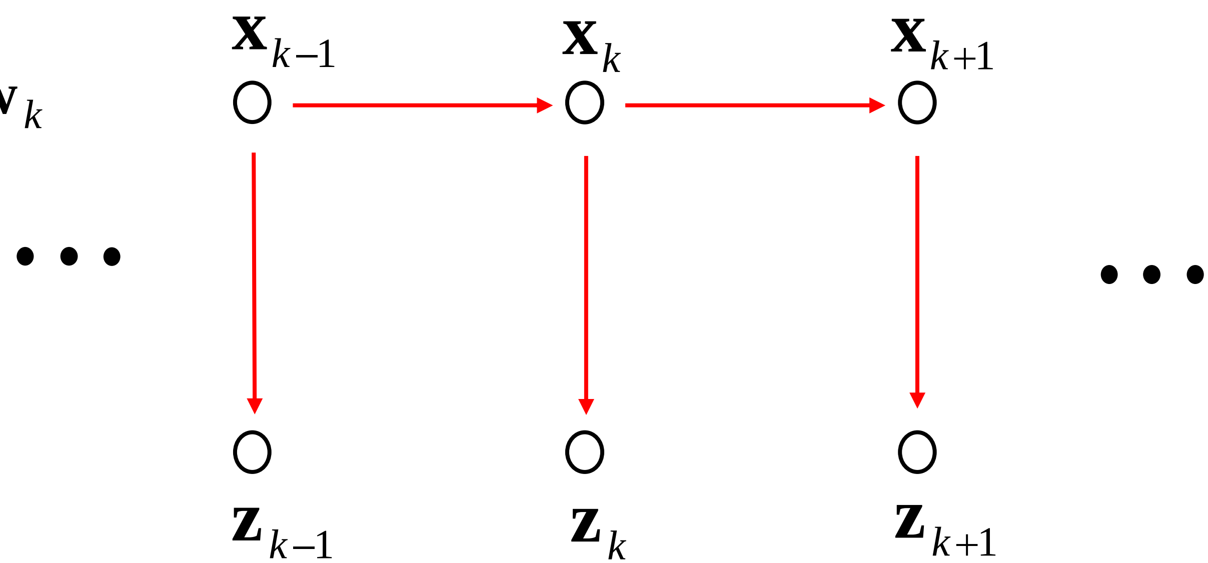
- Common simplification: $\mathbf{A}_k, \mathbf{H}_k, \mathbf{W}_k, \mathbf{Q}_k$ are constant over time (independent of k).

- The **A, H, W, Q** can be estimated by maximizing the joint probability $p(\mathbf{X}_M, \mathbf{Z}_M)$.

Bayesian Graphical Model

$$\mathbf{x}_k = \mathbf{A}_k \mathbf{x}_{k-1} + \mathbf{w}_k$$

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{q}_k$$



$$p(\mathbf{X}_M, \mathbf{Z}_M) = p(\mathbf{X}_M)\, p(\mathbf{Z}_M \mid \mathbf{X}_M)$$

$$= [\, p(\mathbf{x}_1) \prod_{k=2}^{M} p(\mathbf{x}_k | \mathbf{x}_{k-1})][\prod_{k=1}^{M} p(\mathbf{z}_k | \mathbf{x}_k)]$$

Splitting the Joint Distribution

$$\arg\max_{A,W,H,Q}\; p(\mathbf{X}_M, \mathbf{Z}_M)$$

$$= \arg\max_{A,W}\; p(\mathbf{X}_M) \arg\max_{H,Q}\; p(\mathbf{Z}_M | \mathbf{X}_M)$$

$$= \arg\min_{A,W}\; f(\mathbf{A}, \mathbf{W}) \arg\min_{H,Q}\; g(\mathbf{H}, \mathbf{Q})$$

where

$$f(\mathbf{A}, \mathbf{W}) = -\alpha \log p(\mathbf{X}_M) = \sum_{k=2}^{M} [\log(\det \mathbf{W}) + (\mathbf{x}_k - \mathbf{A}\mathbf{x}_{k-1})^T \mathbf{W}^{-1} (\mathbf{x}_k - \mathbf{A}\mathbf{x}_{k-1})],$$

$$g(\mathbf{H}, \mathbf{Q}) = -\beta \log p(\mathbf{Z}_M | \mathbf{X}_M) = \sum_{k=1}^{M} [\log(\det \mathbf{Q}) + (\mathbf{z}_k - \mathbf{H}\mathbf{x}_k)^T \mathbf{Q}^{-1} (\mathbf{z}_k - \mathbf{H}\mathbf{x}_k)].$$

How to optimize functions with matrix variables ???

Closed-form Solutions:

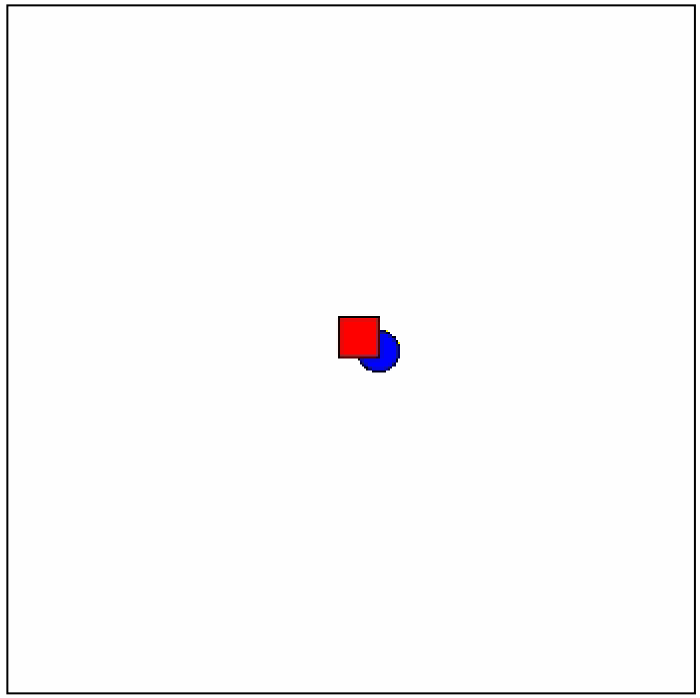
$$\mathbf{A} = \left(\sum_{k=2}^{M} \mathbf{x}_k \mathbf{x}_{k-1}^T\right)\left(\sum_{k=2}^{M} \mathbf{x}_{k-1} \mathbf{x}_{k-1}^T\right)^{-1},$$

$$\mathbf{W} = \frac{1}{M-1}\left(\sum_{k=2}^{M} \mathbf{x}_k \mathbf{x}_k^T - \mathbf{A}\sum_{k=2}^{M} \mathbf{x}_{k-1} \mathbf{x}_k^T\right),$$

$$\mathbf{H} = \left(\sum_{k=1}^{M} \mathbf{z}_k \mathbf{x}_k^T\right)\left(\sum_{k=1}^{M} \mathbf{x}_k \mathbf{x}_k^T\right)^{-1},$$

$$\mathbf{Q} = \frac{1}{M}\left(\sum_{k=1}^{M} \mathbf{z}_k \mathbf{z}_k^T - \mathbf{H}\sum_{k=1}^{M} \mathbf{x}_k \mathbf{z}_k^T\right).$$

Off-line Reconstruction



69 cells with
1.5 minutes of
training data

Actual hand position

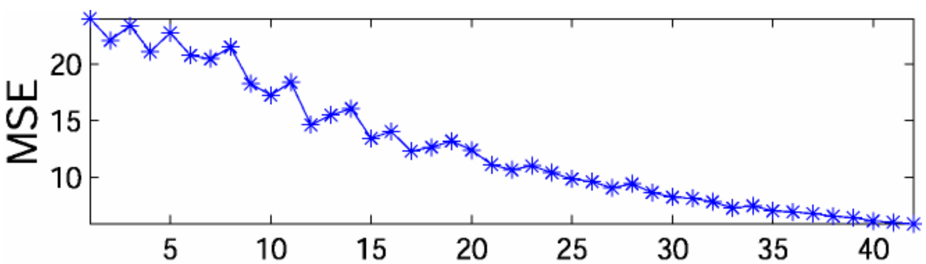
Estimated/decoded position (reconstruction)

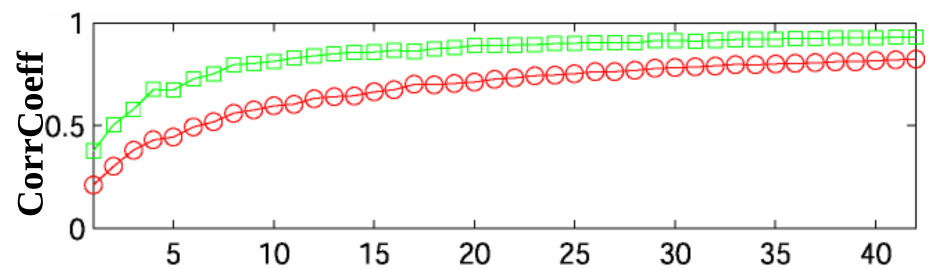
Accuracy

Continuous 2D hand motion (off-line reconstruction):

Method	RMSE (cm)
Population vector	8.66
Linear regression method	2.55
Kalman filter	2.18

As number of cells increases:





Optimal "Lag"

Measurement Equation

$$\vec{z}_k = H\ \vec{x}_k + \vec{q}_k$$

Firing precedes motion:

* Uniform: lag j time steps (1 time step = $70ms$)

$$\vec{z}_{k-j} = H\ \vec{x}_k + \vec{q}_k \qquad j = 0,1,2,3,4$$

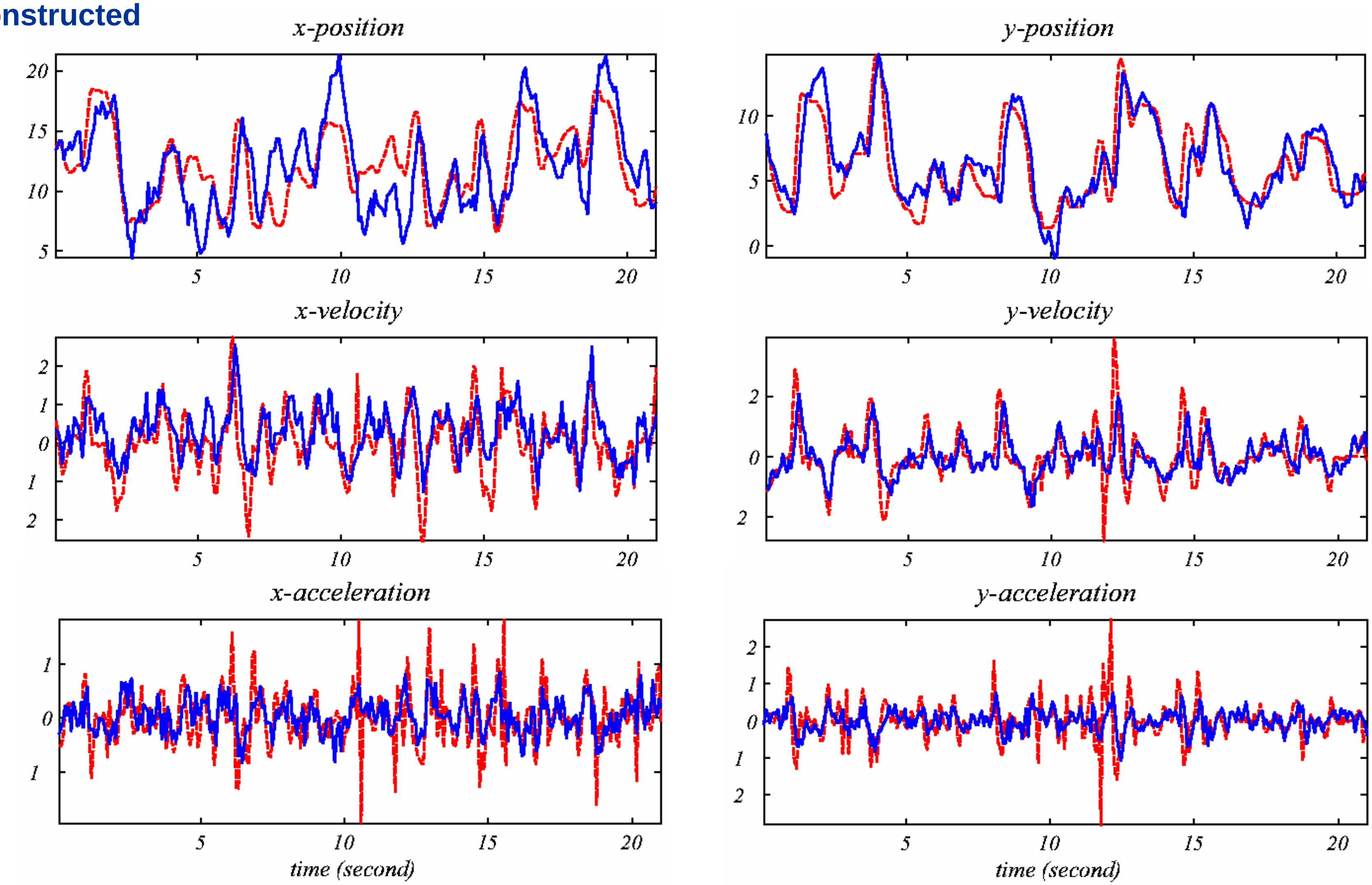
* Non-uniform: lag $(j_1, j_2, \cdots, j_{42})$ time steps

Reconstruction and Lag

Methods	CC (x, y)	MSE (cm^2)
Kalman (0ms lag)	(0.77, 0.91)	6.96
Kalman (70ms lag)	(0.79, 0.93)	6.67
Kalman (140ms lag)	**(0.81, 0.93)**	**6.09**
Kalman (210ms lag)	(0.81, 0.89)	6.98
Kalman (280ms lag)	(0.76, 0.82)	8.91
Kalman (non-uniform)	(0.82, 0.93)	5.24

Reconstruction (Test Data)

reconstructed
true



Covariance

Diagonal covariance assumes noise in firing rates is statistically independent.

$$p(\mathbf{z}_t \mid \mathbf{x}_t) = \prod_{i=1}^{C} p(z_{i,t} \mid \mathbf{x}_t) = \prod_{i=1}^{C} \frac{1}{\sqrt{2\pi\sigma_i}} \exp(-\frac{1}{2\sigma_i^2}(z_{i,t} - \mathbf{H}_i \mathbf{x}_t)^2)$$

Kalman decoding

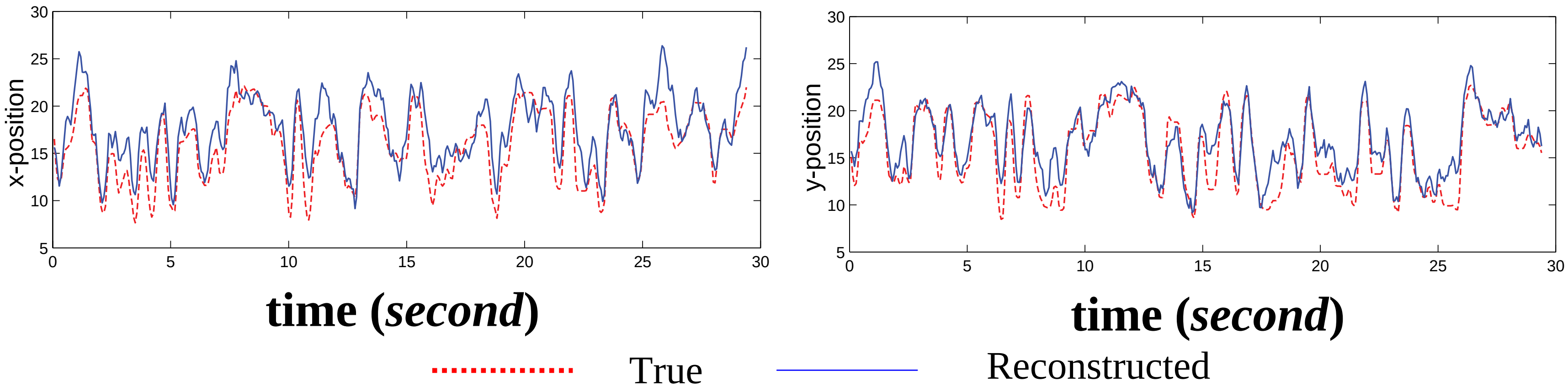
* Full covariance: MSE = 5.99 cm^2
* Diagonal covariance: MSE = 6.35 cm^2

Off-line Decoding

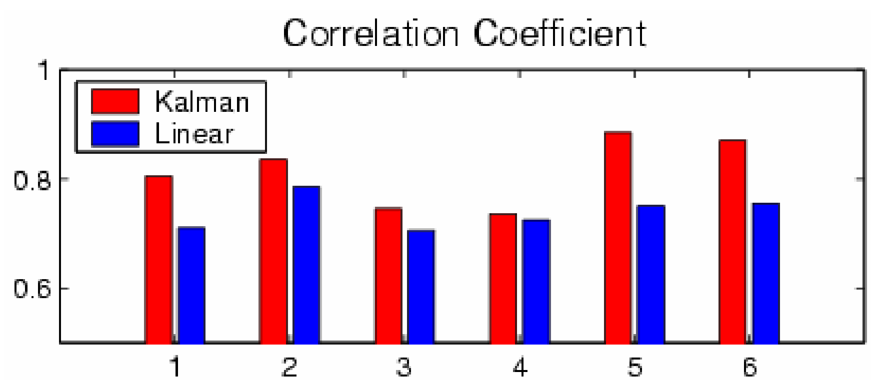
# of cells	Kalman filter		Linear regression	
	CC	*RMSE* (*cm*)	*CC*	*RMSE* (*cm*)
23	(0.79, 0.82)	3.61	(0.70, 0.72)	4.35
30	(0.88, 0.79)	3.26	(0.85, 0.72)	3.49
36	(0.75, 0.74)	4.36	(0.77, 0.64)	4.38
26	(0.71, 0.76)	4.48	(0.71, 0.74)	4.39
69	(0.88, 0.89)	3.11	(0.72, 0.78)	5.30
69	(0.86, 0.88)	3.26	(0.71, 0.80)	3.99

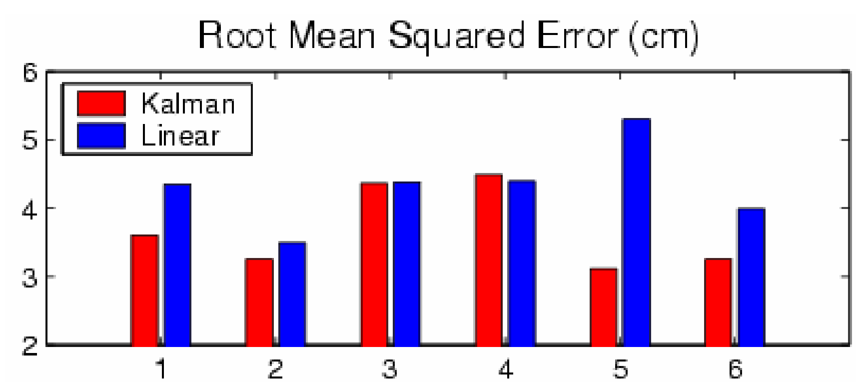
Kalman filter	***CC* = 0.81±0.06, *RMSE* = 3.7±0.6 (cm)**
Linear regression	***CC* = 0.74±0.05, *RMSE* = 4.3±0.6 (cm)**

Decoding



Decoding results over all six experiments:

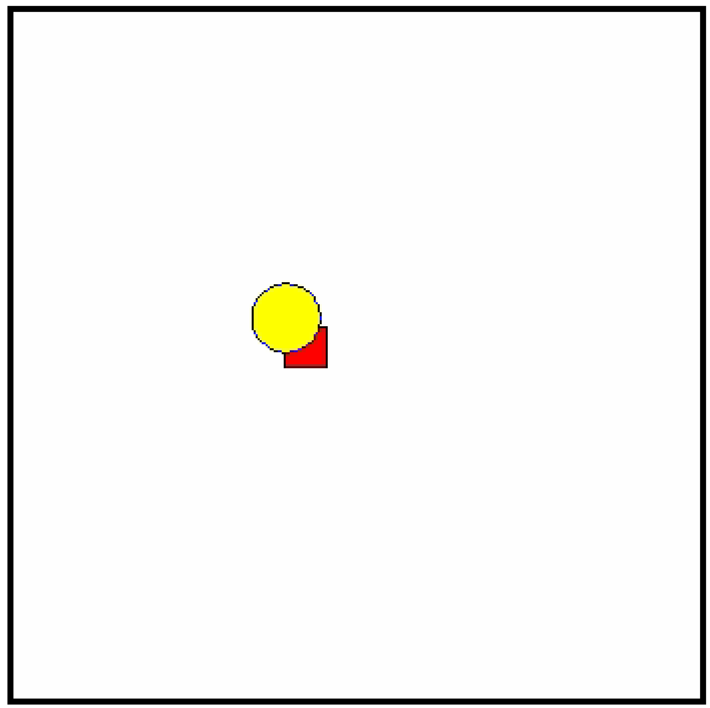




On-line Neural Control

Neural control of a computer cursor in real time.

Brain substitutes for hand.



Kalman filter decoder.
Only 18 cells.

Directly exploits the generative encoding model.

Target

Visual feedback

On-line Task Performance

# of cells	Kalman filter			Linear regression		
	time	targets	rate	time	targets	rate
17	60*sec*	38	**38/*min***			
30	105*sec*	55	**31/*min***	58*sec*	24	**25/*min***
36	57*sec*	28	**29/*min***	42*sec*	15	**21/*min***
69	45*sec*	28	**37/*min***	60*sec*	22	**22/*min***

Average results: 50% improvement

Kalman filter 33.75 targets/min

Linear regression 22.67 targets/min

Non-Gaussian Likelihood

Inhomogeneous Poisson:

$$p(z \mid \mathbf{x}) = \frac{1}{z!}(\mathbf{Hx})^z e^{-\mathbf{Hx}}$$

Hx is the predicted mean firing rate.
z is the observed rate for a *single* cell.

Problems? Hx may be negative.
No clear way to model correlated noise.