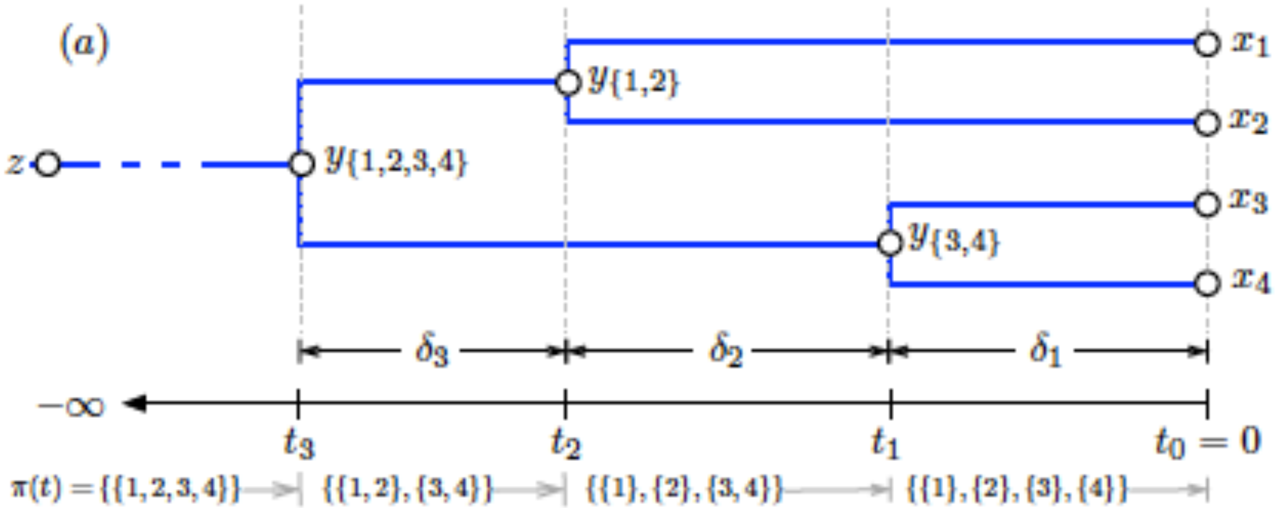
Applied Bayesian Nonparametrics

Special Topics in Machine Learning
Brown University CSCI 2950-P, Fall 2011

November 10: Coalescent Processes, Hierarchical Clustering

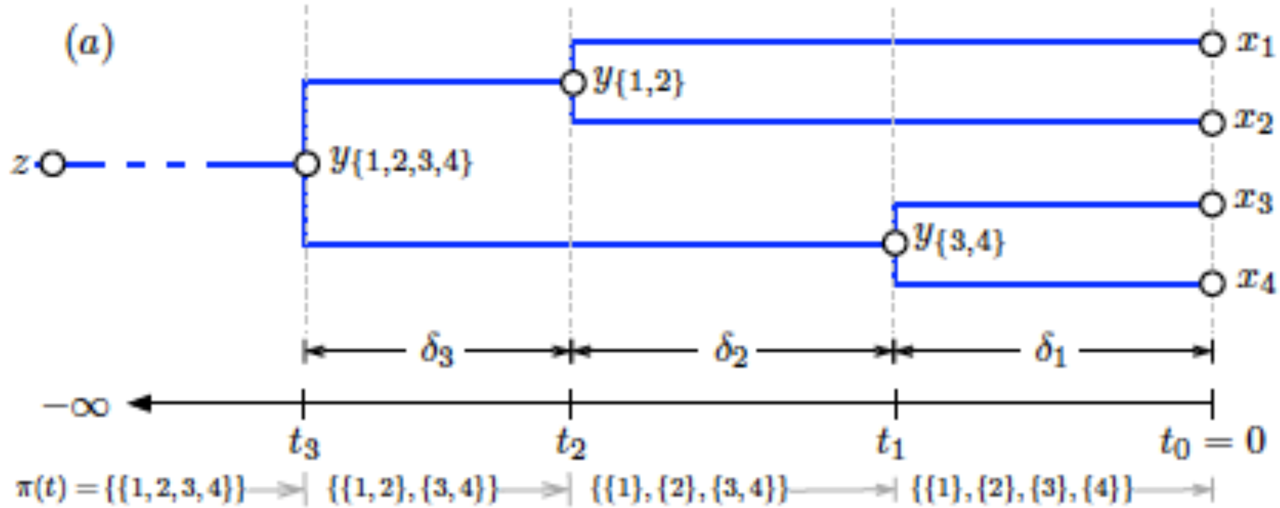
Prior: The Coalescent



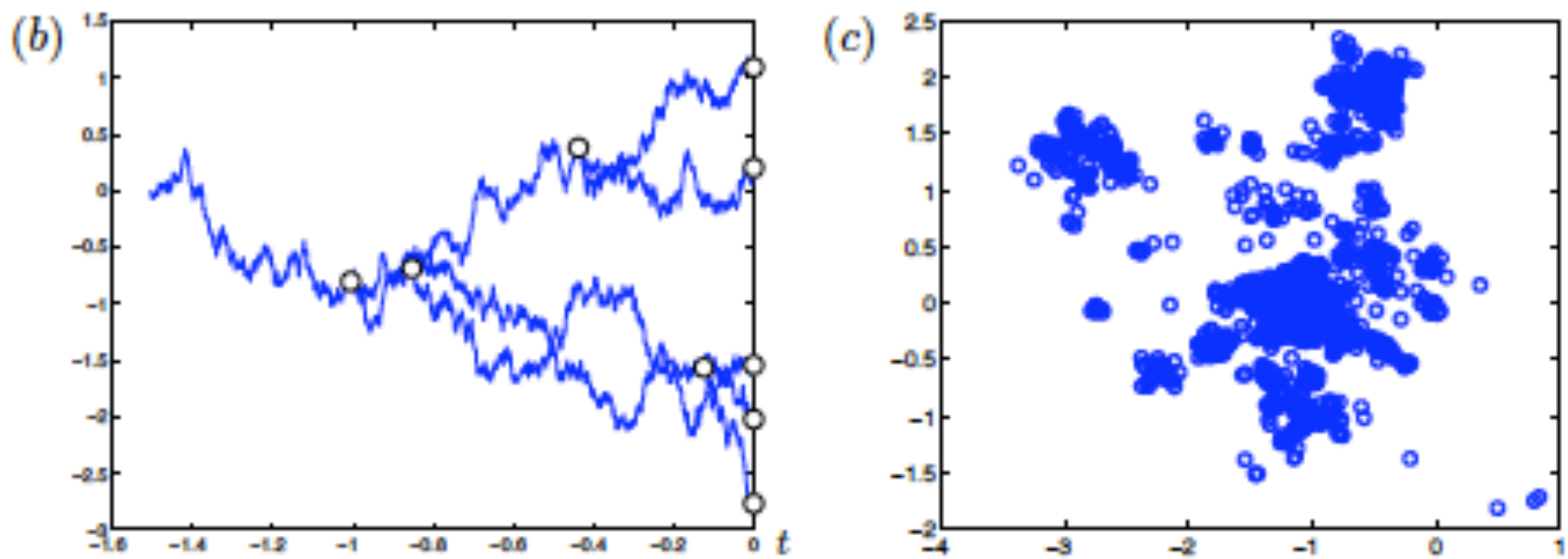
$$\pi(t) = \begin{cases} \{\{1\}, \ldots, \{n\}\} & \text{if } t = 0; \\ \pi_{t_{i-1}} - \rho_{li} - \rho_{ri} + (\rho_{li} \cup \rho_{ri}) & \text{if } t = t_i; \\ \pi_{t_i} & \text{if } t_{i+1} < t < t_i. \end{cases}$$

$$\delta_i \sim \mathrm{Exp}\left(\binom{n-i+1}{2}\right) \qquad \qquad \delta_i = t_{i-1} - t_i > 0$$

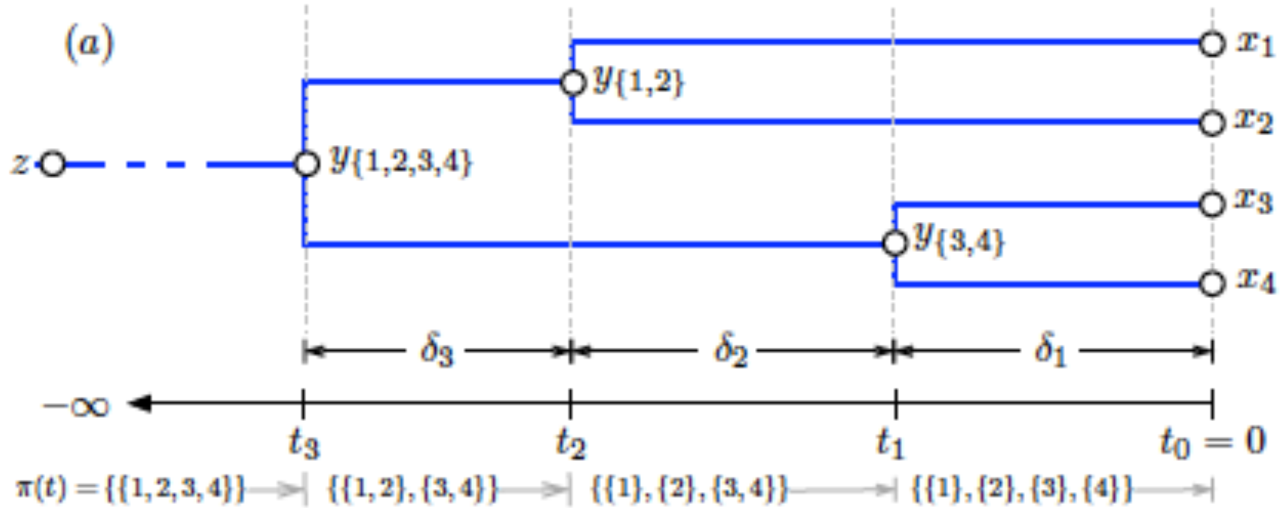
Likelihood: Markov Process on Tree



$$p(\mathbf{x}, \mathbf{y}, z|\pi) = q(z) k_{-\infty\, t_{n-1}}(z, y_{\rho_{n-1}}) \prod_{i=1}^{n-1} k_{t_i t_{li}}(y_{\rho_i}, y_{\rho_{li}}) k_{t_i t_{ri}}(y_{\rho_i}, y_{\rho_{ri}})$$



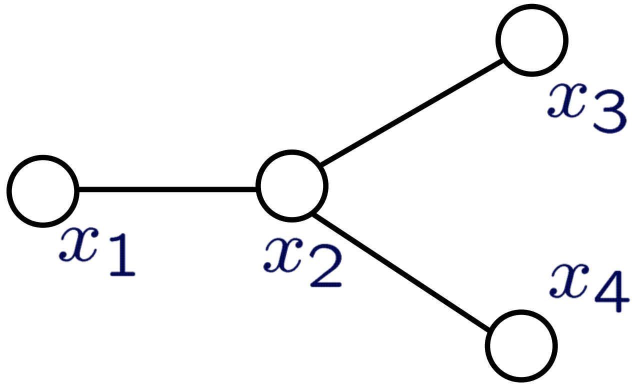
Inference Algorithms



$$p(\mathbf{x}, \mathbf{y}, z|\pi) = q(z)k_{-\infty\, t_{n-1}}(z, y_{\rho_{n-1}}) \prod_{i=1}^{n-1} k_{t_i t_{li}}(y_{\rho_i}, y_{\rho_{li}}) k_{t_i t_{ri}}(y_{\rho_i}, y_{\rho_{ri}})$$

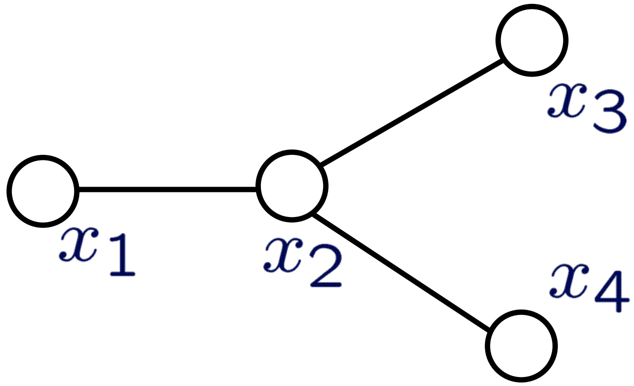
- *Collapsed* inference algorithms: Markov process on latent nodes is marginalized analytically using belief propagation (sum-product)
- *Greedy:* Bottom-up search for a single good tree
- *Sequential Monte Carlo:* Approximate true posterior on trees by a weighted set of samples (particles)

Inference via the Distributed Law



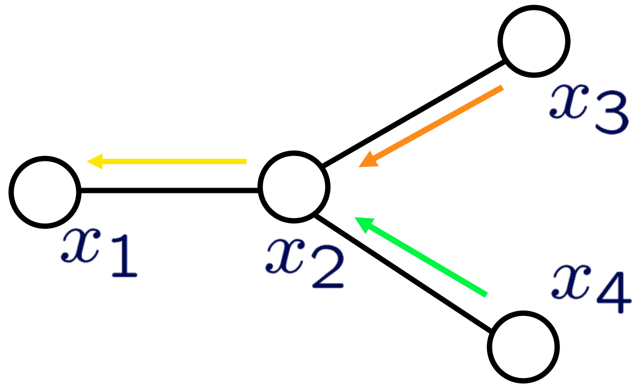
$$
\begin{aligned}
p_1(x_1) &= \sum_{x_2,x_3,x_4} \psi_1(x_1)\psi_{12}(x_1,x_2)\psi_2(x_2)\psi_{23}(x_2,x_3)\psi_3(x_3)\psi_{24}(x_2,x_4)\psi_4(x_4) \\
&= \psi_1(x_1) \sum_{x_2,x_3,x_4} \psi_{12}(x_1,x_2)\psi_2(x_2)\psi_{23}(x_2,x_3)\psi_3(x_3)\psi_{24}(x_2,x_4)\psi_4(x_4)
\end{aligned}
$$

Inference via the Distributed Law



$$
\begin{aligned}
p_1(x_1) &= \sum_{x_2,x_3,x_4} \psi_1(x_1)\psi_{12}(x_1,x_2)\psi_2(x_2)\psi_{23}(x_2,x_3)\psi_3(x_3)\psi_{24}(x_2,x_4)\psi_4(x_4) \\
&= \psi_1(x_1) \sum_{x_2,x_3,x_4} \psi_{12}(x_1,x_2)\psi_2(x_2)\psi_{23}(x_2,x_3)\psi_3(x_3)\psi_{24}(x_2,x_4)\psi_4(x_4) \\
&= \psi_1(x_1) \sum_{x_2} \psi_{12}(x_1,x_2)\psi_2(x_2) \sum_{x_3,x_4} \psi_{23}(x_2,x_3)\psi_3(x_3)\psi_{24}(x_2,x_4)\psi_4(x_4)
\end{aligned}
$$

Inference via the Distributed Law



$$p_1(x_1) = \sum_{x_2,x_3,x_4} \psi_1(x_1)\psi_{12}(x_1,x_2)\psi_2(x_2)\psi_{23}(x_2,x_3)\psi_3(x_3)\psi_{24}(x_2,x_4)\psi_4(x_4)$$

$$= \psi_1(x_1) \sum_{x_2,x_3,x_4} \psi_{12}(x_1,x_2)\psi_2(x_2)\psi_{23}(x_2,x_3)\psi_3(x_3)\psi_{24}(x_2,x_4)\psi_4(x_4)$$

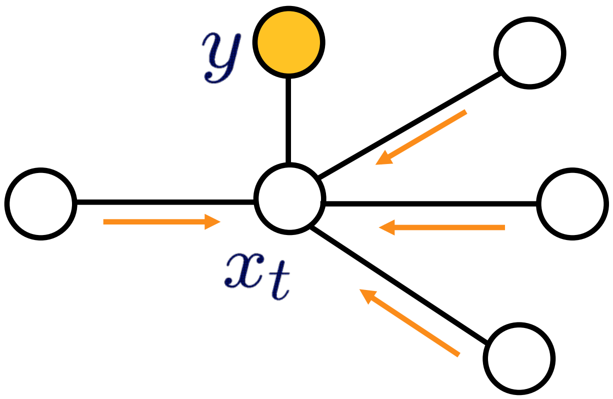
$$= \psi_1(x_1) \sum_{x_2} \psi_{12}(x_1,x_2)\psi_2(x_2) \sum_{x_3,x_4} \psi_{23}(x_2,x_3)\psi_3(x_3)\psi_{24}(x_2,x_4)\psi_4(x_4)$$

$$= \underbrace{\psi_1(x_1) \sum_{x_2} \psi_{12}(x_1,x_2)\psi_2(x_2) \underbrace{\left[\sum_{x_3} \psi_{23}(x_2,x_3)\psi_3(x_3)\right]}_{m_{32}(x_2)} \cdot \underbrace{\left[\sum_{x_4} \psi_{24}(x_2,x_4)\psi_4(x_4)\right]}_{m_{42}(x_2)}}$$

$$m_{21}(x_1) = \sum_{x_2} \psi_{12}(x_1,x_2)\psi_2(x_2)m_{32}(x_2)m_{42}(x_2)$$

Belief Propagation (Sum-Product)

BELIEFS: Posterior marginals (possibly approximate)

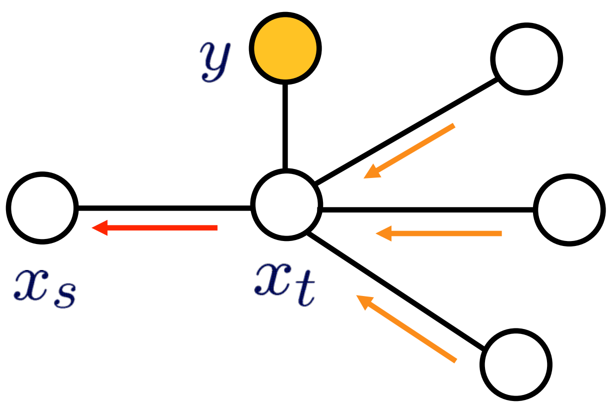


$$q_t(x_t \mid y) = \alpha \psi_t(x_t, y) \prod_{u \in \Gamma(t)} m_{ut}(x_t)$$

$\Gamma(t)$ ⟶ neighborhood of node t (adjacent nodes)

MESSAGES: Sufficient statistics (possibly approximate)

$$m_{ts}(x_s) = \alpha \sum_{x_t} \psi_{st}(x_s, x_t) \psi_t(x_t, y) \prod_{u \in \Gamma(t) \setminus s} m_{ut}(x_t)$$



I) Message Product

II) Message Propagation

Belief Propagation for Trees

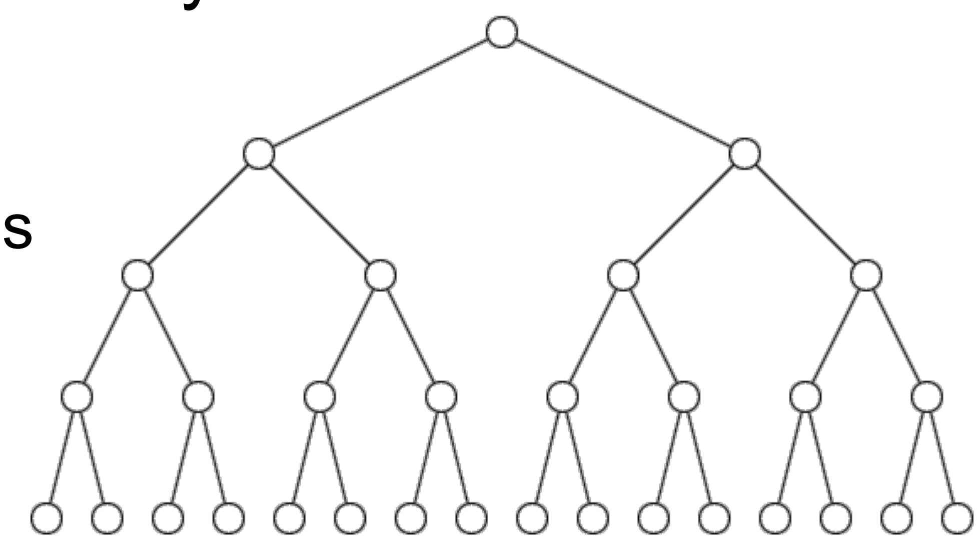
- Dynamic programming algorithm which exactly computes all marginals
- On Markov chains, BP equivalent to alpha-beta or forward-backward algorithms for HMMs
- Sequential message schedules require each message to be updated only once
- Computational cost:

$N \longrightarrow$ number of nodes

$M \longrightarrow$ discrete states for each node

Belief Prop: $\mathcal{O}(NM^2)$

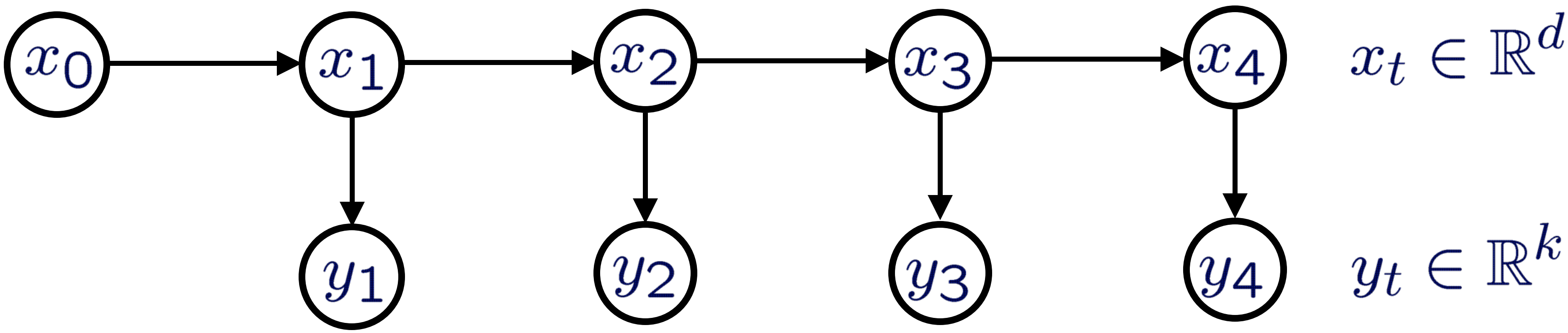
Brute Force: $\mathcal{O}(M^N)$



Greedy Coalescent Clustering

- *Belief propagation* allows likelihoods to be computed by bottom-up message passing, integrates with bottom-up greedy merging — $\mathcal{O}(n)$
- *Greedy-MaxProb:* At each iteration, find the optimal time for each candidate merge (pair of nodes), select the most likely pair+time — $\mathcal{O}(n^3)$
- *Greedy-Rate1:* Find the most likely time for each pair to merge under equivalent formulation as independent rate 1 processes, take soonest — $\mathcal{O}(n^2)$
- Algorithmic structure nearly identical to Bayesian hierarchical clustering, but model is hierarchical

Nonlinear State Space Models

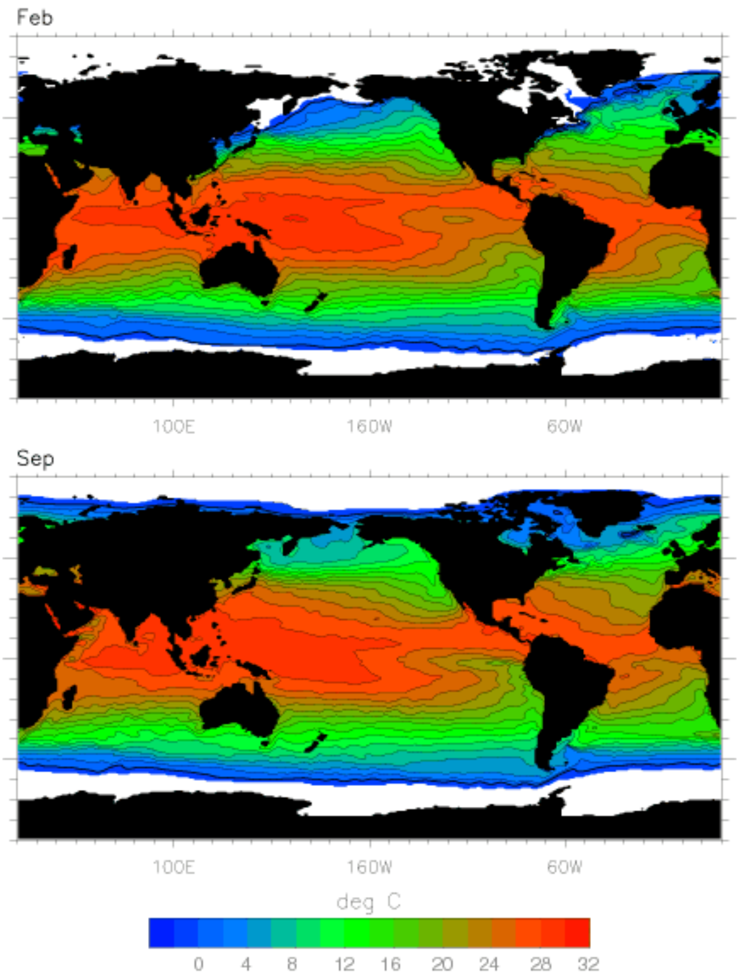


$$x_{t+1} = f(x_t, w_t) \qquad w_t \sim \mathcal{F}$$

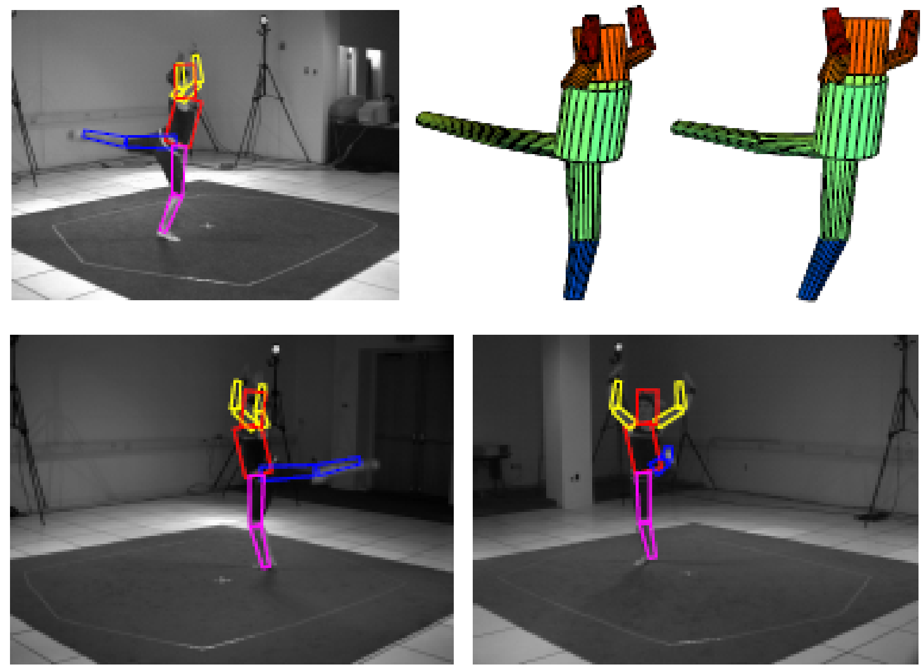
$$y_t = g(x_t, v_t) \qquad v_t \sim \mathcal{G}$$

- State dynamics and measurements given by potentially complex nonlinear functions
- Noise sampled from non-Gaussian distributions

Examples of Nonlinear Models

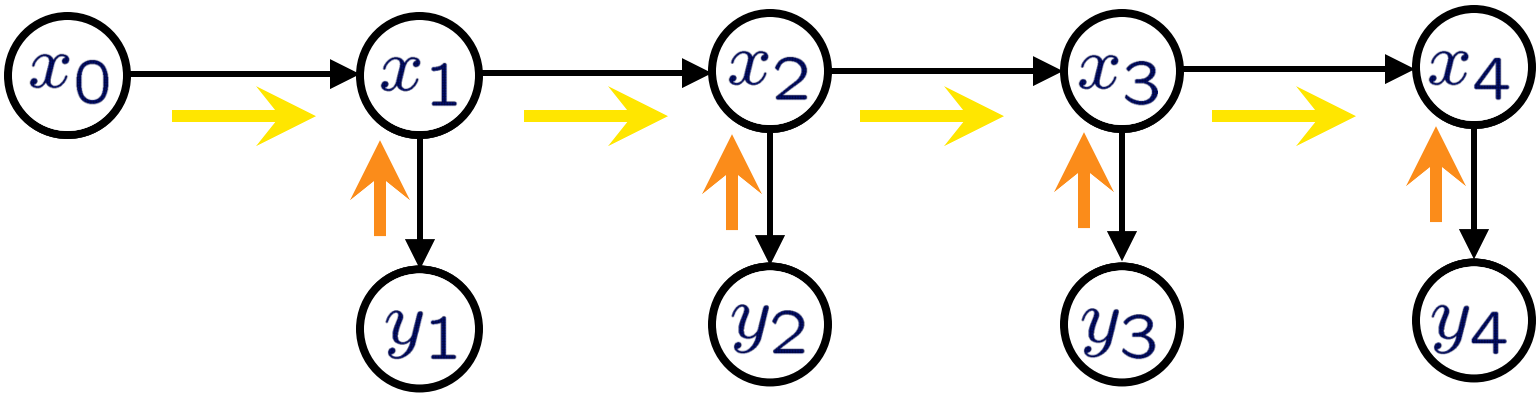


Dynamics implicitly determined by geophysical simulations



Observed image is a complex function of the 3D pose, other nearby objects & clutter, lighting conditions, camera calibration, etc.

Nonlinear Filtering



$$p(x_t \mid y_1, \ldots, y_{t-1}) = \tilde{q}_t(x_t)$$

$$p(x_t \mid y_1, \ldots, y_t) = q_t(x_t)$$

Prediction:

$$\tilde{q}_t(x_t) = \int p(x_t \mid x_{t-1}) q_{t-1}(x_{t-1}) \, dx_{t-1}$$

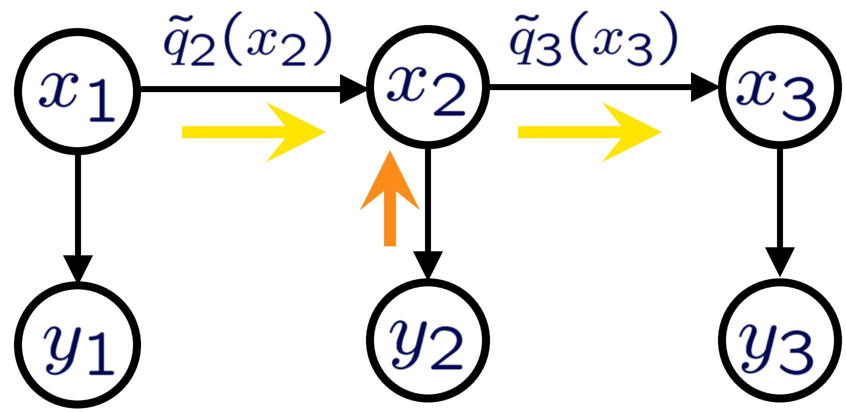
Update:

$$q_t(x_t) = \frac{1}{Z_t} \tilde{q}_t(x_t) p(y_t \mid x_t)$$

Particle Filters

Condensation, Sequential Monte Carlo, Survival of the Fittest,…

- Represent state estimates using a set of samples
- Propagate over time using importance sampling



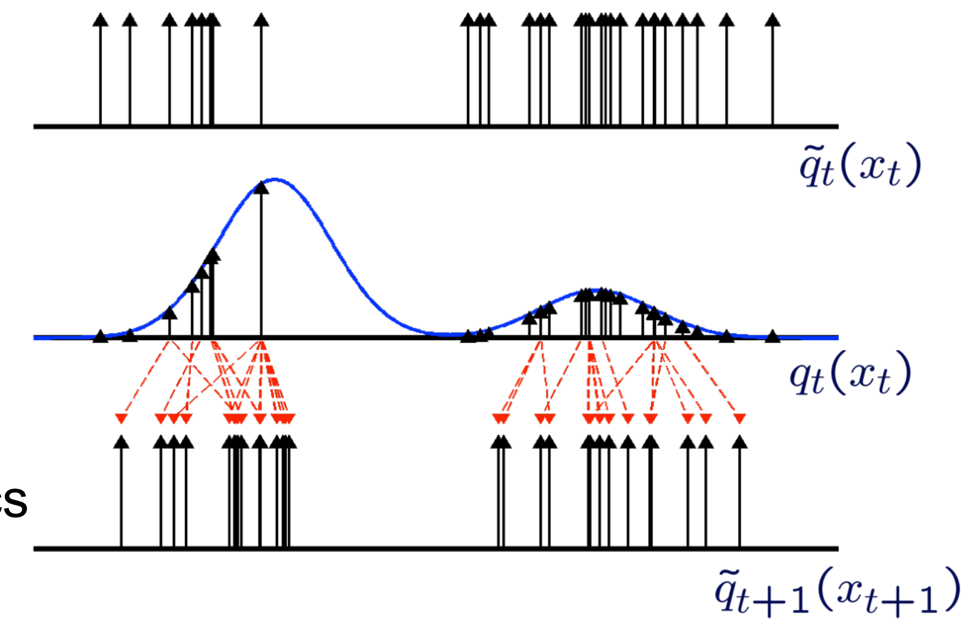
Sample-based density estimate

↓

Weight by observation likelihood

↓

Resample & propagate by dynamics



Sequential Monte Carlo (SMC)

- SMC methods can be used to sample approximately from any sequence of growing distributions $\{\pi_n\}_{n\geq 1}$

$$\pi_n(x_{1:n}) = \frac{f_n(x_{1:n})}{Z_n}$$

where

- $f_n : \mathcal{X}^n \to \mathbb{R}^+$ is known point-wise.
- $Z_n = \int f_n(x_{1:n}) dx_{1:n}$

- We introduce a proposal distribution $q_n(x_{1:n})$ to approzimate Z_n:

$$Z_n = \int \frac{f_n(x_{1:n})}{q_n(x_{1:n})} q_n(x_{1:n}) dx_{1:n} = \int W_n(x_{1:n})\, q_n(x_{1:n}) dx_{1:n}$$

de Freitas & Doucet, Tutorial at NIPS 2009

SMC Algorithm

1. Initialize at time $n = 1$

2. At time $n \geq 2$

   - Sample $\overline{X}_n^{(i)} \sim q_n\left(x_n \middle| X_{1:n-1}^{(i)}\right)$ and augment $\overline{X}_{1:n}^{(i)} = \left(X_{1:n-1}^{(i)}, \overline{X}_n^{(i)}\right)$

   - Compute the sequential weight

$$W_n^{(i)} \propto \frac{f_n\left(\overline{X}_{1:n}^{(i)}\right)}{f_{n-1}\left(\overline{X}_{1:n-1}^{(i)}\right) q_n\left(\overline{X}_n^{(i)} \middle| \overline{X}_{1:n-1}^{(i)}\right)}.$$

   Then the target approximation is:

$$\widetilde{\pi}_n\left(x_{1:n}\right) = \sum_{i=1}^{N} W_n^{(i)} \delta_{\overline{X}_{1:n}^{(i)}}\left(x_{1:n}\right)$$

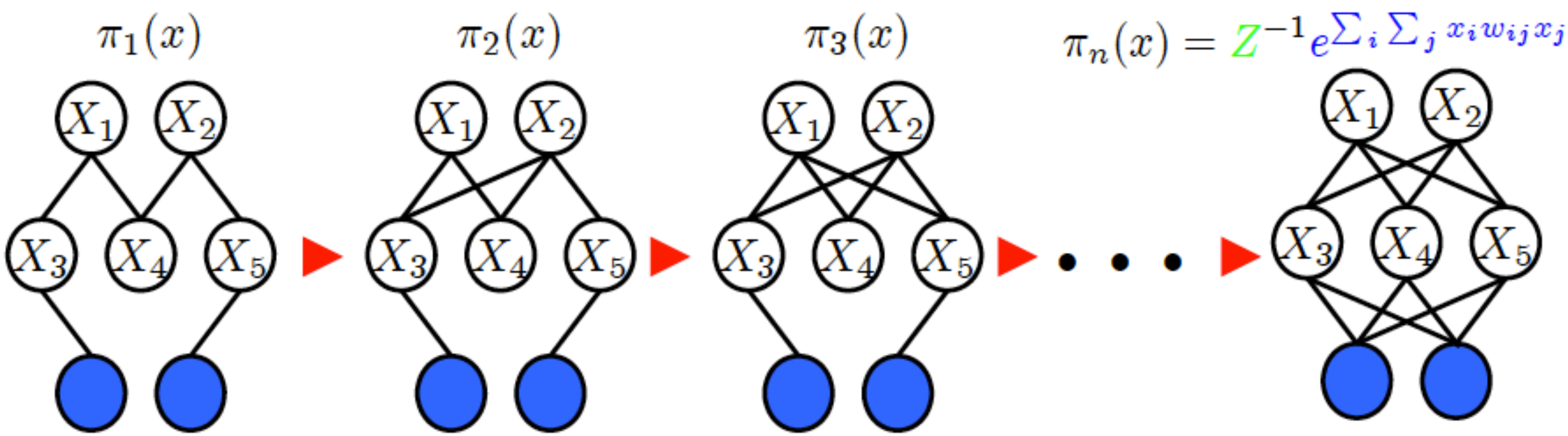
   - Resample $X_{1:n}^{(i)} \sim \widetilde{\pi}_n\left(x_{1:n}\right)$ to obtain $\widehat{\pi}_n\left(x_{1:n}\right) = \frac{1}{N} \sum_{i=1}^{N} \delta_{X_{1:n}^{(i)}}\left(x_{1:n}\right)$.

de Freitas & Doucet, Tutorial at NIPS 2009

SMC for Static Models

$$\pi_n(x) = \underbrace{Z_n^{-1}}_{\text{unknown}} \underbrace{f_n(x)}_{\text{known}}$$

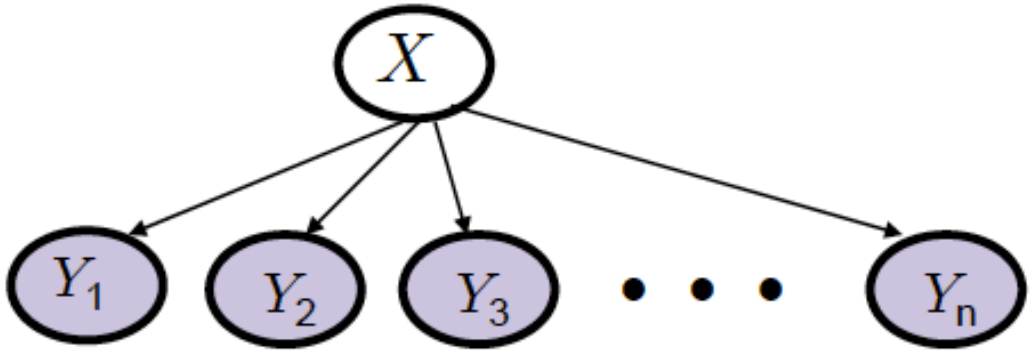
- We want to sample approximately from $\pi_n(x)$ and compute Z_n sequentially.
- This differs from the standard SMC, where $\pi_n(x_{1:n})$ is defined on $\mathcal{X}^n$.



de Freitas & Doucet, Tutorial at NIPS 2009

Static SMC Applications

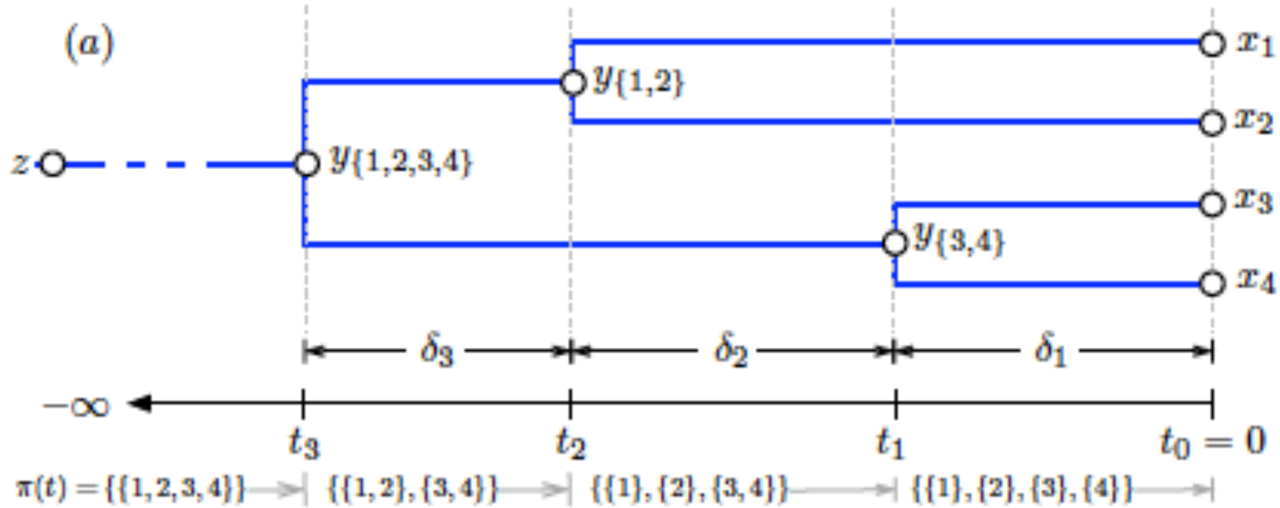
- **Sequential Bayesian Inference:** $\pi_n(x) = p(x|y_{1:n})$.



- **Global optimization:** $\pi_n(x) \propto [\pi(x)]^{\eta_n}$ with $\{\eta_n\}$ increasing sequence such that $\eta_n \to \infty$.
- **Sampling from a fixed target** $\pi_n(x) \propto [\mu_1(x)]^{\eta_n} [\pi(x)]^{1-\eta_n}$ where μ_1 is easy to sample from. Use sequence $\eta_1 = 1 > \eta_{n-1} > \eta_n > \eta_{final} = 0$. Then $\pi_1(x) \propto \mu(x)$ and $\pi_{final}(x) \propto \pi(x)$
- **Rare event simulation** $\pi(A) \ll 1$: $\pi_n(x) \propto \pi(x) 1_{E_n}(x)$ with Z_1 known. Use sequence $E_1 = \mathcal{X} \supset E_{n-1} \supset E_n \supset E_{final} = A$. Then $Z_{final} = \pi(A)$.

de Freitas & Doucet, Tutorial at NIPS 2009

SMC for Coalescents



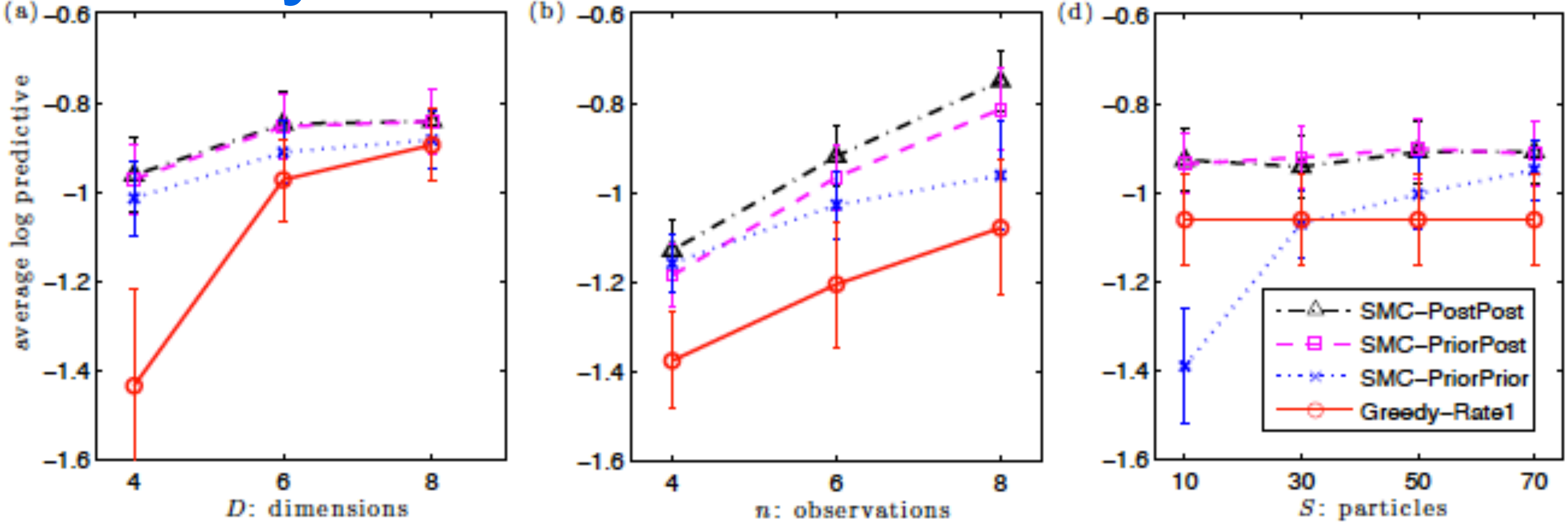
$$\theta_{i-1}^s = \{\delta_j^s, \rho_{lj}^s, \rho_{rj}^s \text{ for } j < i\}$$

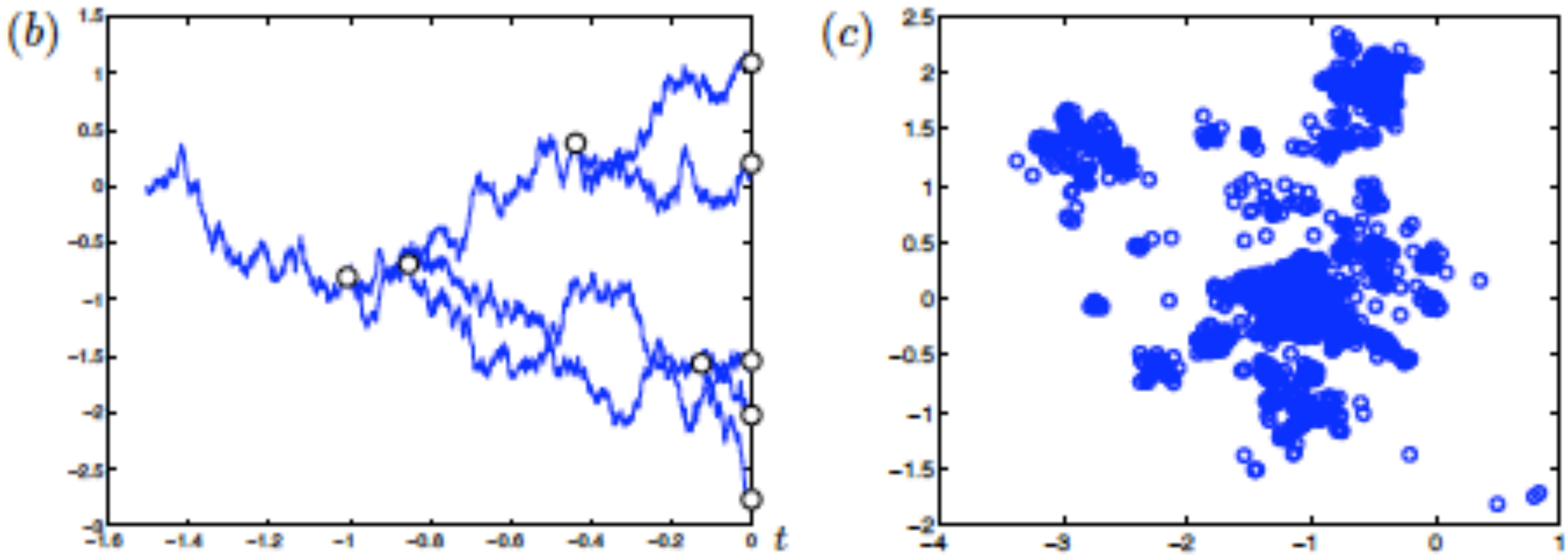
$$w_i^s = w_{i-1}^s \exp\left(-\binom{n-i+1}{2}\delta_i^s\right) Z_{\rho_i}(\mathbf{x}, \theta_i^s) / f_i(\delta_i^s, \rho_{li}^s, \rho_{ri}^s | \theta_{i-1}^s)$$

$$p(\pi, \mathbf{x}) \approx \sum_s w_{n-1}^s \delta_{\theta_{n-1}^s}(\pi)$$

- *SMC-PriorPrior:* Sample time & pair of nodes from prior
- *SMC-PriorPost:* Sample time from prior, sample pair of nodes from posterior given data and that time (discrete distribution)
- *SMC-PostPost:* Sample time & pair of nodes from posterior

Toy Brownian Diffusion Data

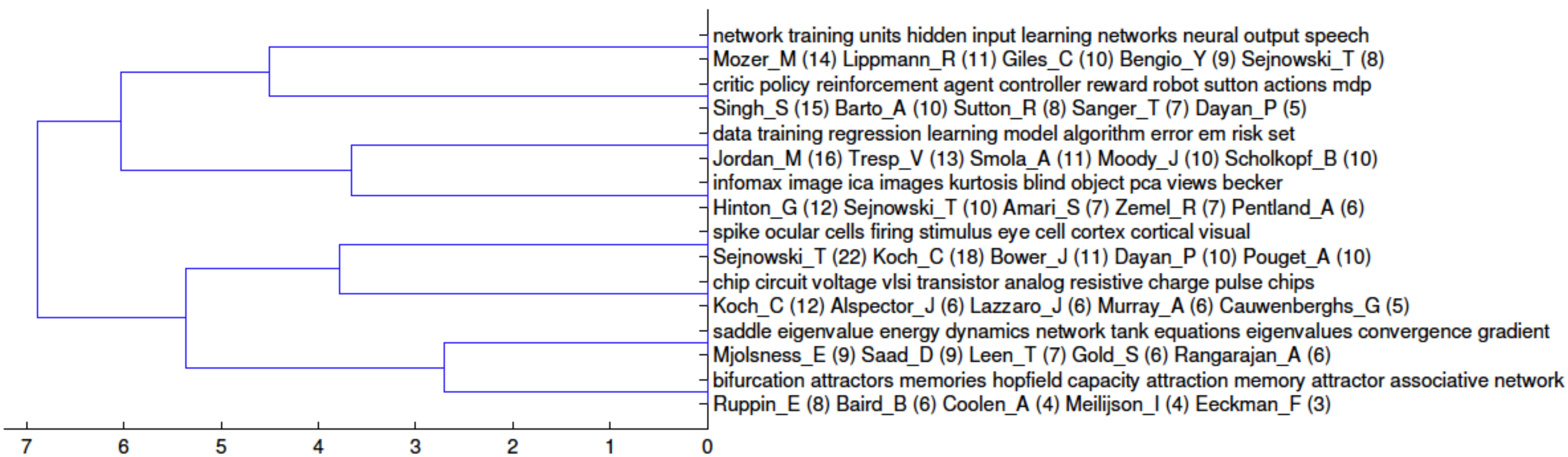




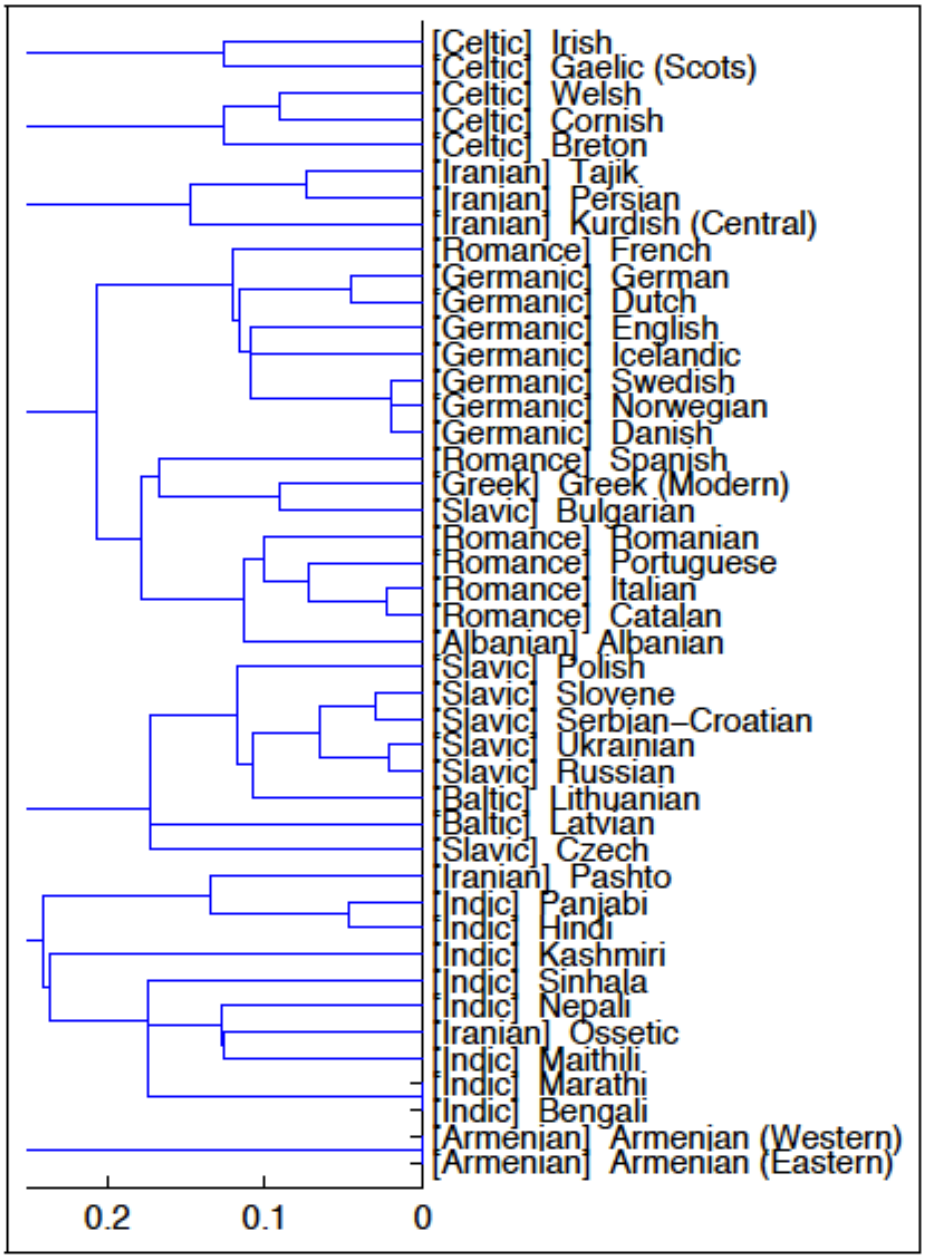
Greedy Hierarchical Clustering

	MNIST Avg-link	MNIST BHC	MNIST Coalescent	SPAMBASE Avg-link	SPAMBASE BHC	SPAMBASE Coalescent
Purity	.363±.004	.392±.006	**.412±.006**	.616±.007	**.711±.010**	.689±.008
Subtree	.581±.005	.579±.005	**.610±.005**	.607±.011	.549±.015	**.661±.012**
LOO-acc	.755±.005	.763±.005	**.773±.005**	.846±.010	.832±.010	**.861±.008**

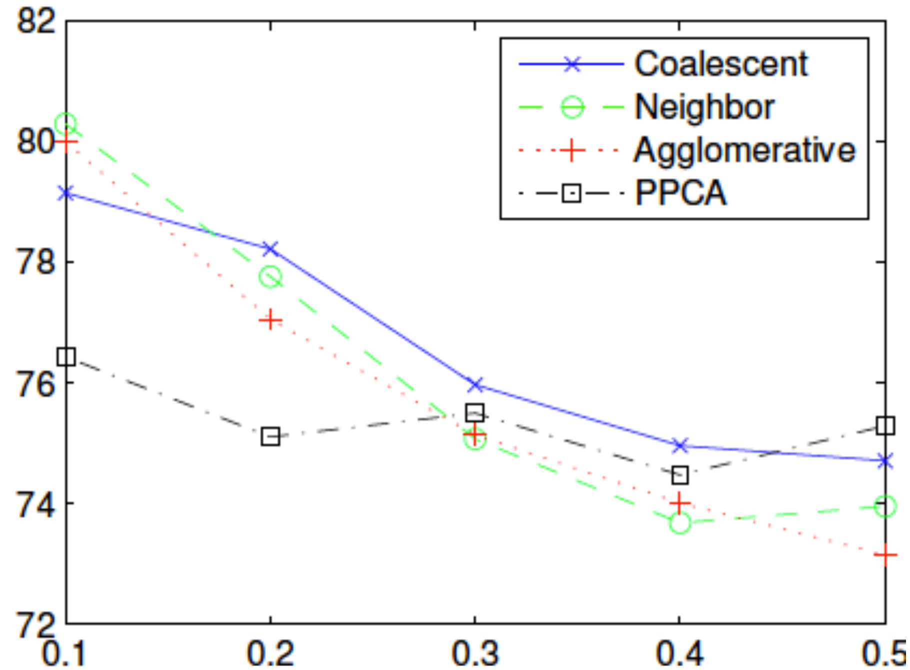
NIPS Documents with Binary Encoding of Common Words



World Atlas of Language Structures



Data Restoration Accuracy:



Indo-European Data

	Avg-link	BHC	Coalescent
Purity	0.510	0.491	**0.813**
Subtree	0.414	0.414	**0.690**
LOO-acc	0.538	0.590	**0.769**

Whole World Data

	Avg-link	BHC	Coalescent
Purity	0.162	0.160	**0.269**
Subtree	0.227	0.099	**0.177**
LOO-acc	0.080	0.248	**0.369**