Fast Approximate Inference for Probabilistic Machine Learning in Complex Models

Jason Pacheco

Assistant Professor
Dept. of Computer Science

Context of This Talk

Intention Convey overview of probabilistic methods for problems where outcomes are random and uncertain

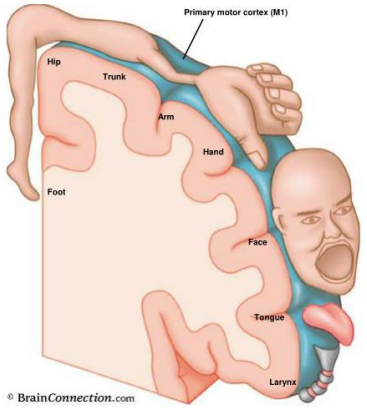
Focus of Work Develop efficient and flexible algorithms for approximate inference and decision making in above settings

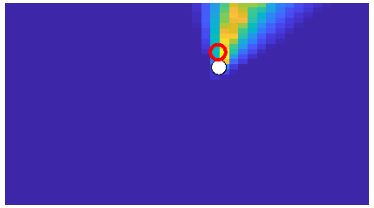
This talk will not Go into deep technical detail on any method—do pardon a little bit of necessary mathematics—I am happy to discuss details offline

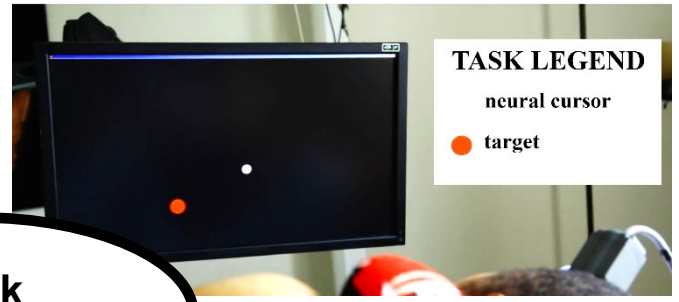
Takeaway I will discuss a methodology, not off-the-shelf methods to solve any specific problems. The best solutions are adapted to exploit individual problem structure!

Block 12: "Multiscale Semi-Markov Model"

Probabilistic Reasoning





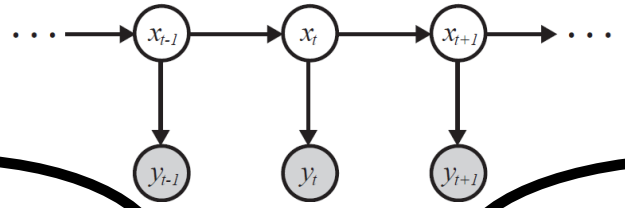


Prior Knowledge

Uncertainty Quantification

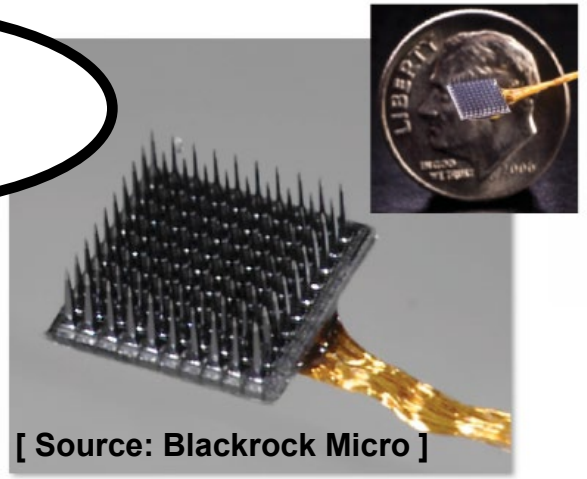
Task Execution

Representation

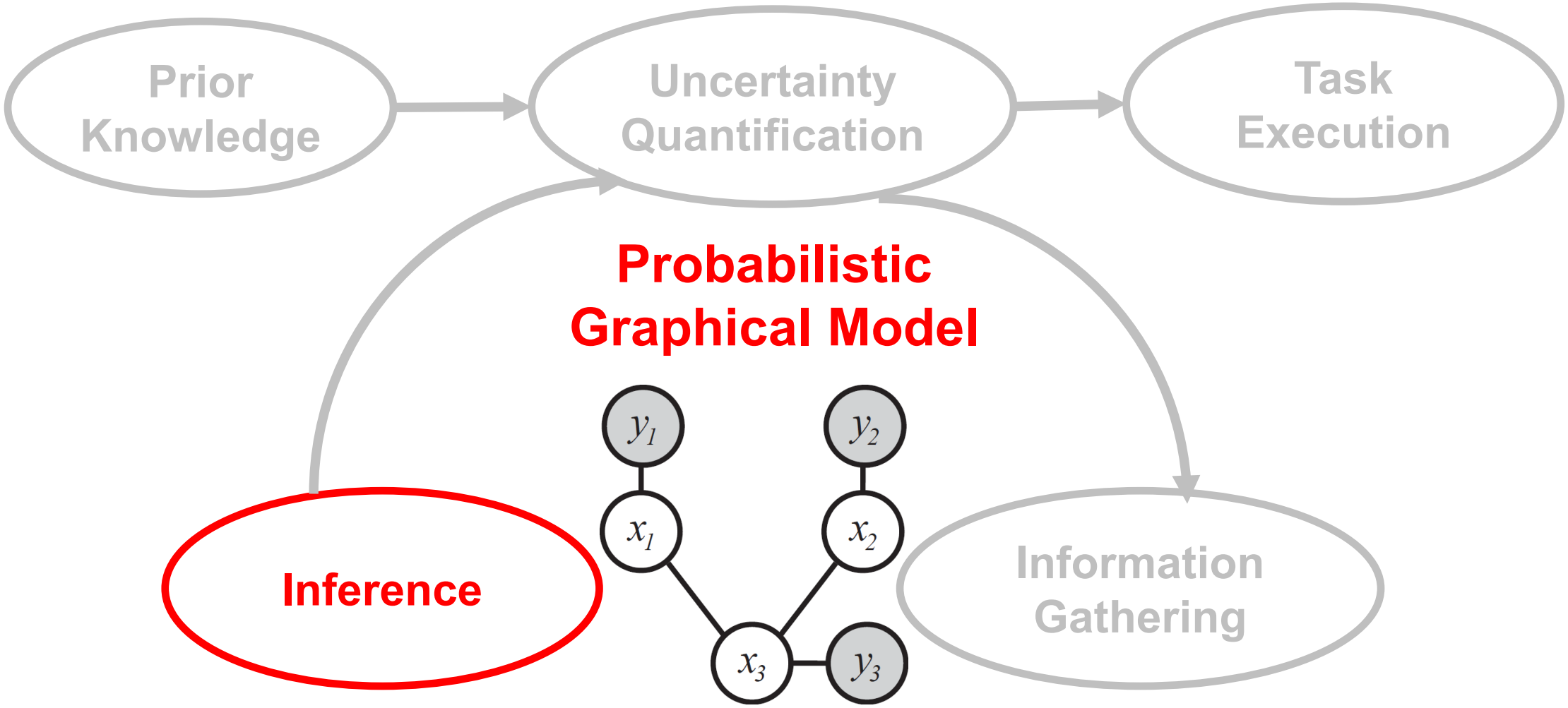


Inference

Information Gathering



Probabilistic Reasoning



Facilitates development of efficient inference algorithms.

What is a Probabilistic Graphical Model?

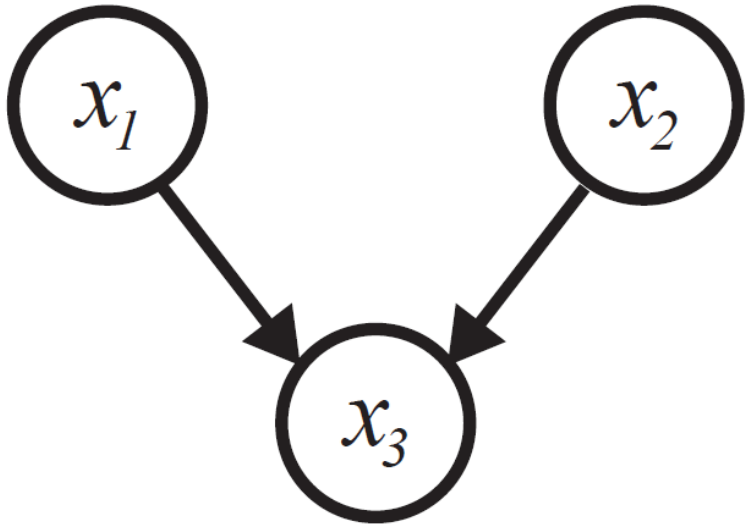
A probabilistic graphical model allows us to pictorially represent a probability distribution

Probability (or Energy) Model:

$$p(x_1, x_2, x_3) =$$

$$p(x_1)p(x_2)p(x_3 \mid x_1, x_2)$$

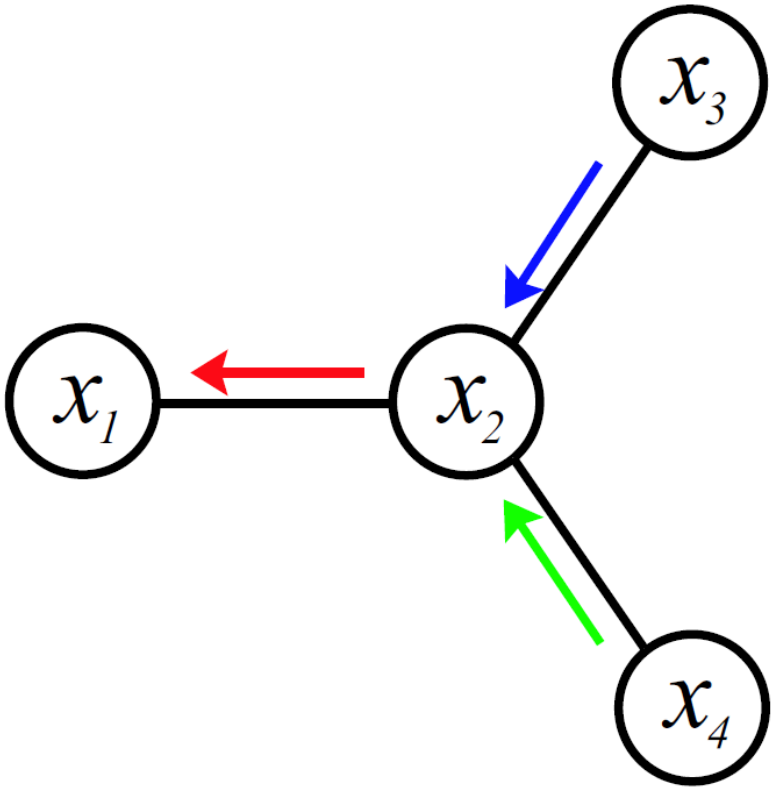
Graphical Model:



The graphical model structure *obeys* the factorization of the probability function (I will not formalize in this talk)

Why Graphical Models?

Structure simplifies both **representation** and **computation**



Representation

Complex global phenomena arise by simpler-to-specify local interactions

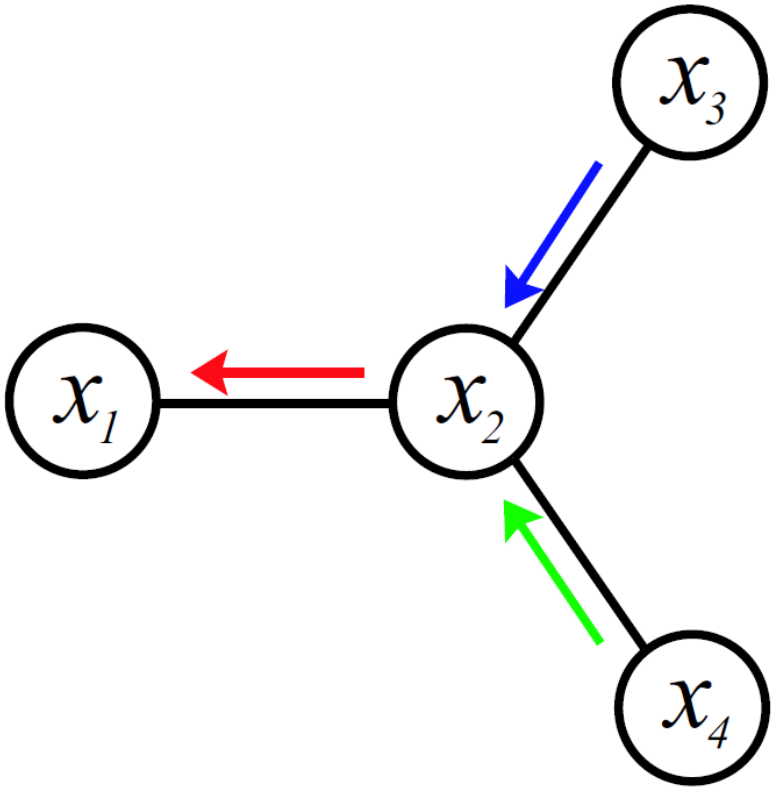
Computation

Inference / estimation depends only on subgraphs (e.g. dynamic programming, belief propagation, Gibbs sampling)

Because computer scientists like to think about graphs and structure

Why Graphical Models?

Structure simplifies both **representation** and **computation**



Representation
Complex global phenomena arise by simpler-to-specify local interactions

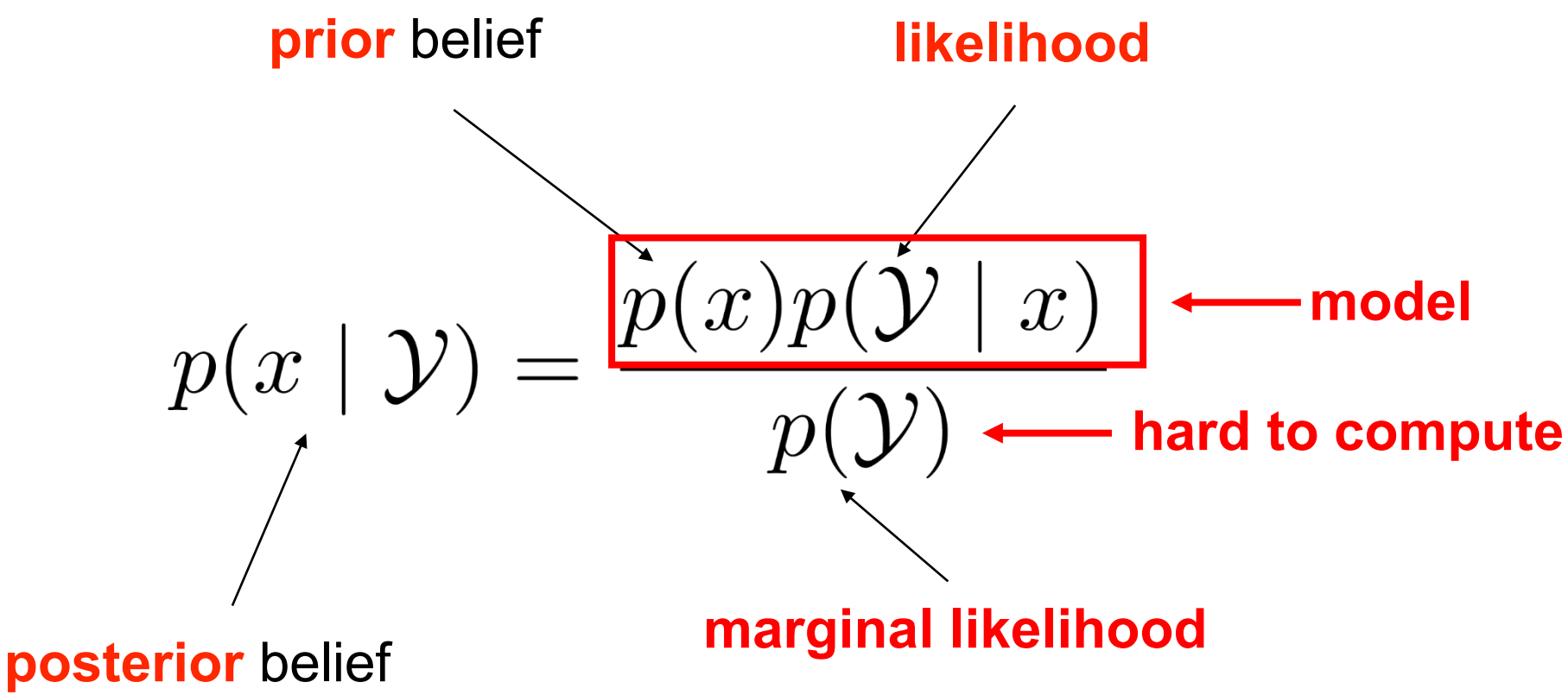
Computation
Inference / estimation depends only on subgraphs (e.g. dynamic programming, belief propagation, Gibbs sampling)

Because computer scientists like to think about graphs and structure

Bayes' Rule

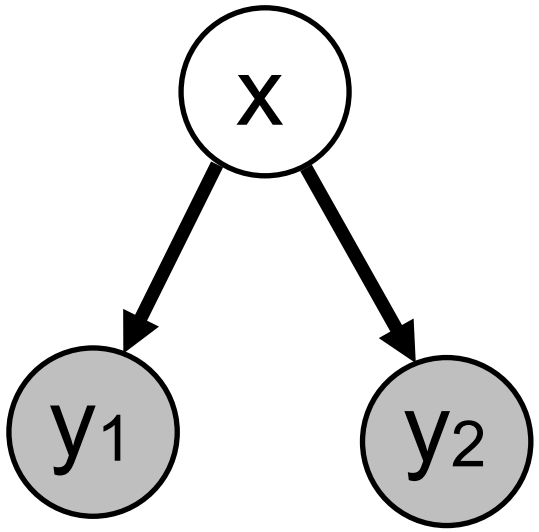
X **Quantity to be inferred**

$\mathcal{Y}$ **Collection of Observed Data**



Posterior encodes our *belief* about unknowns *given* data

Posterior Inference and Graphical Models



Denote observed data with shaded nodes,

$$Y_1 = y_1 \qquad Y_2 = y_2$$

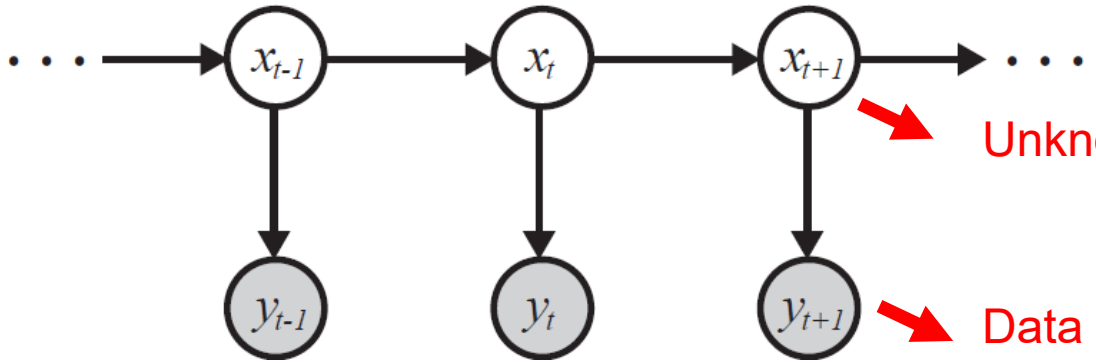
Infer *latent* variable X via Bayes' rule:

$$p(x \mid y_1, y_2) = \frac{p(x)p(y_1 \mid x)p(x_2 \mid x)}{p(y_1, y_2)}$$

- This is (obviously) a simple example
- Models and inference task can get really complicated
- But the fundamental concepts and approach are the same

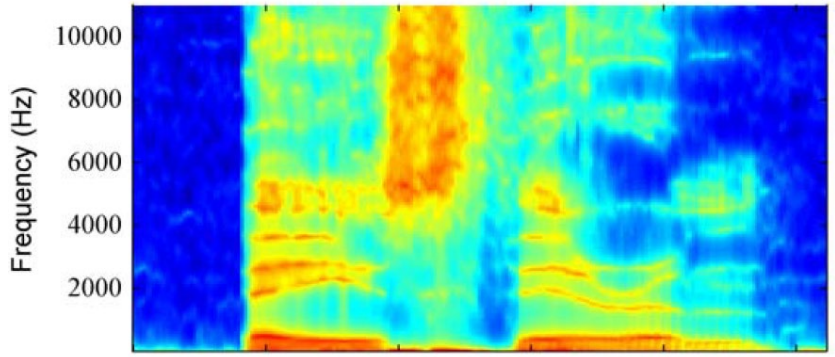
Example : Hidden Markov Models

Sequential models *of* *discrete* *quantities of interest*



Example: Speech Recognition

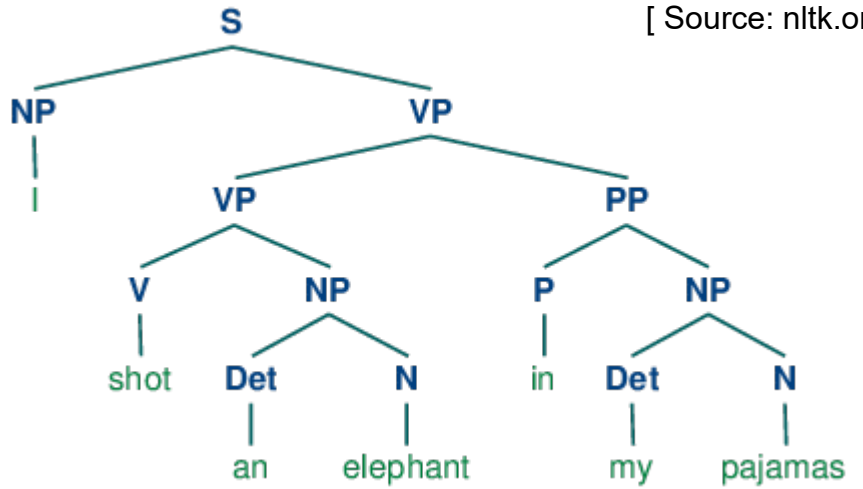
$y =$



x = b-ey-z-th-ih-er-em → Bayes' Theorem

[Source: Bishop, PRML]

[Source: nltk.org]



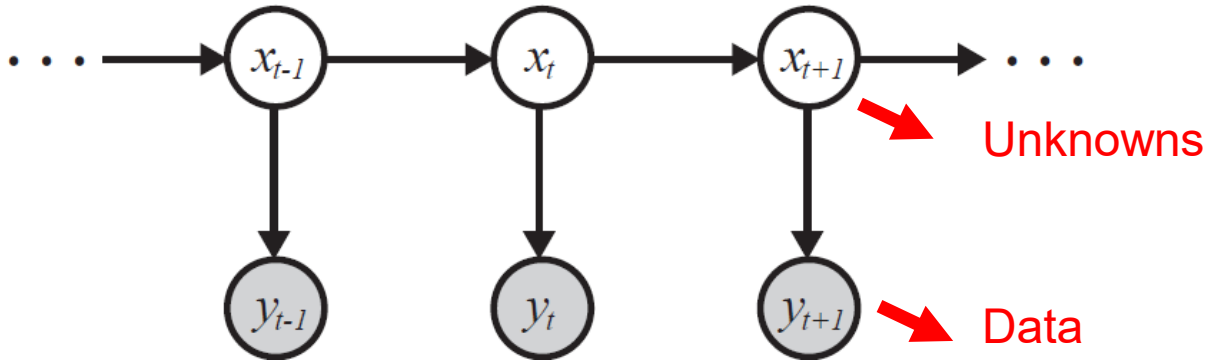
Example: Part-of-speech Tagging:

y = "I shot an elephant in my pajamas."

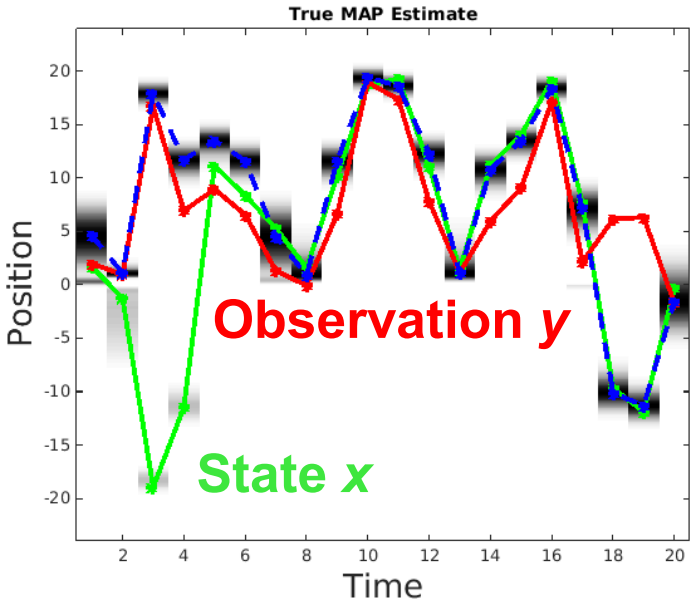
x = NP-V-Det-N-P-Det-N

Example : State-Space Models / Dynamical Systems

Latent state evolves over continuous domain with some dynamics…observations arise conditional on current state



Example: Nonlinear Time Series

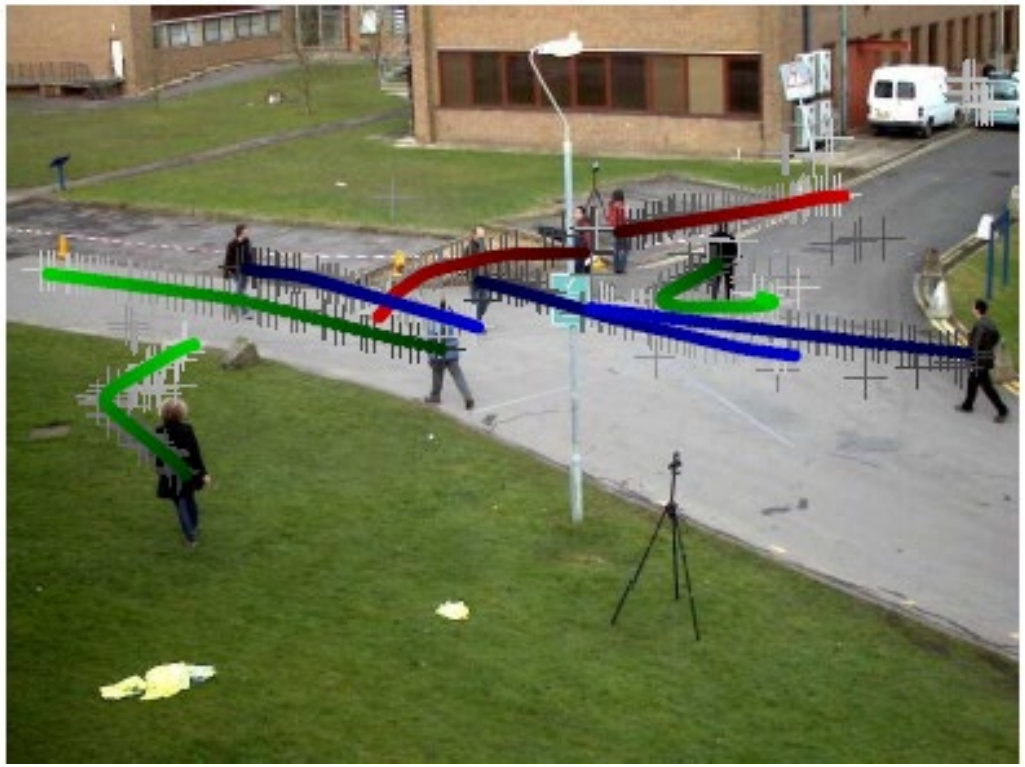


Marginal Posterior

$$p(x_t \mid y_1^T)$$

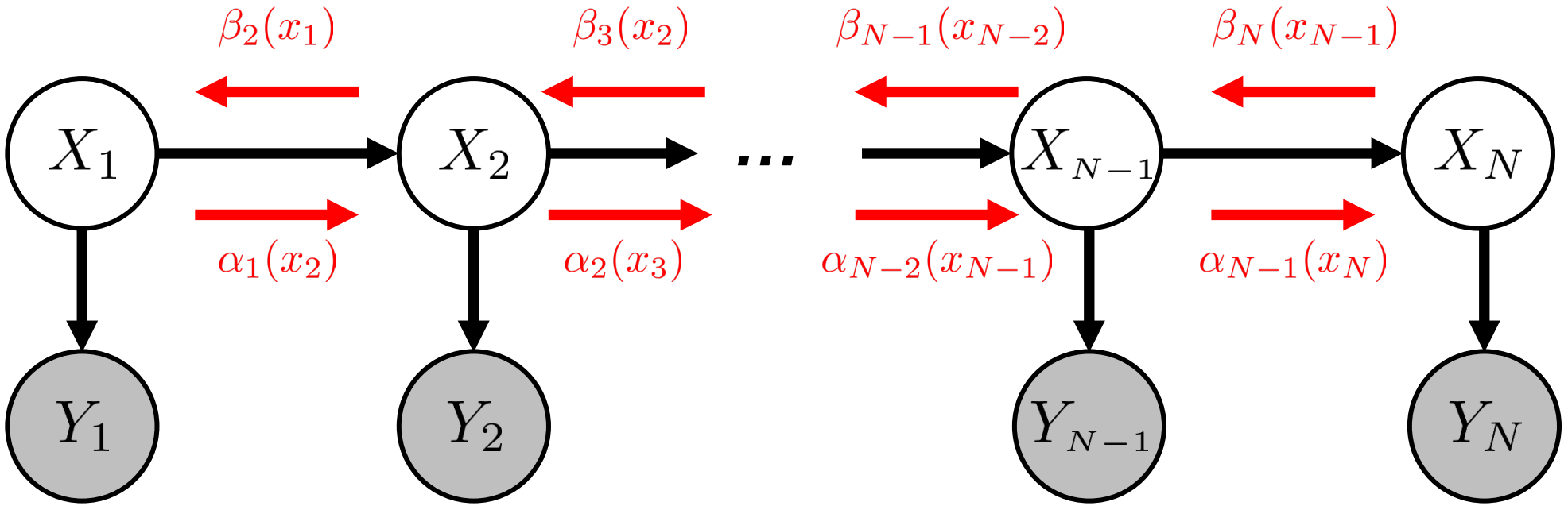
Quantifies belief of state at time t given all observations

Example: Multitarget Tracking



Forward-Backward Algorithm

Extends to HMM-style graphs with node observations…



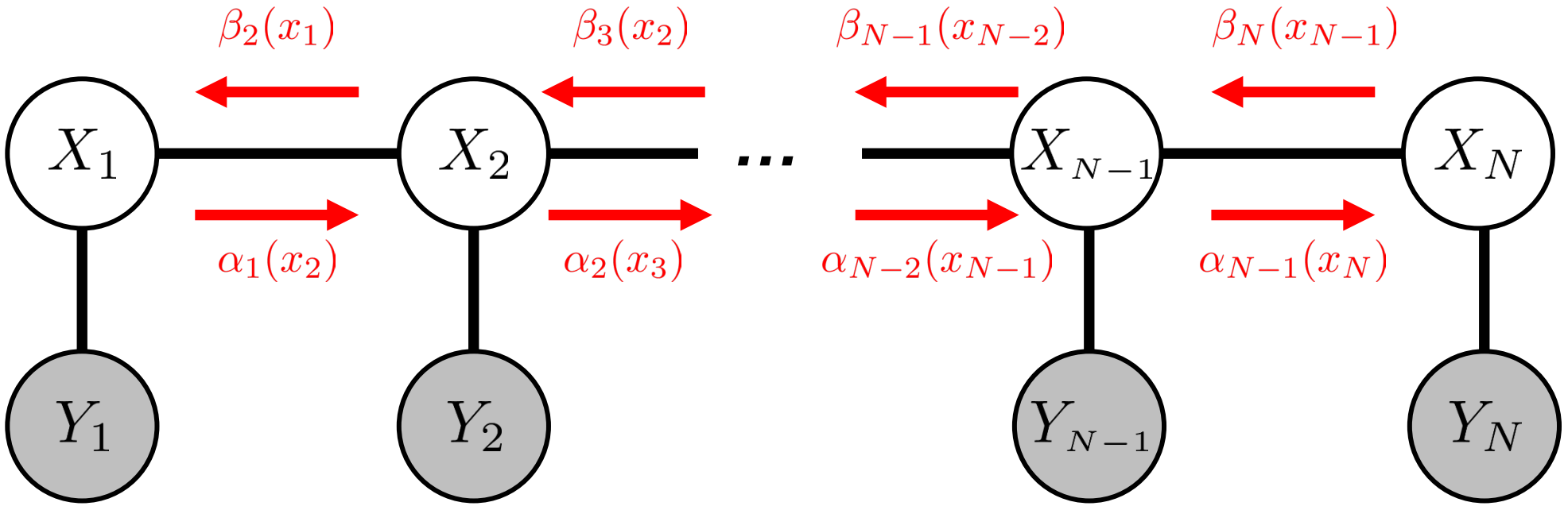
Forward message:

$$\alpha_{n-1}(x_n) = p(y_n \mid x_n) \sum_{x_{n-1}} \alpha_{n-2}(x_{n-1}) p(x_n \mid x_{n-1})$$

Backward message:

$$\beta_{n+1}(x_n) = \sum_{x_{n+1}} \beta_{n+2}(x_{n+1}) p(x_{n+1} \mid x_n) p(y_{n+1} \mid x_{n+1})$$

Forward-Backward Algorithm



Forward message gives the *filtered posterior*:

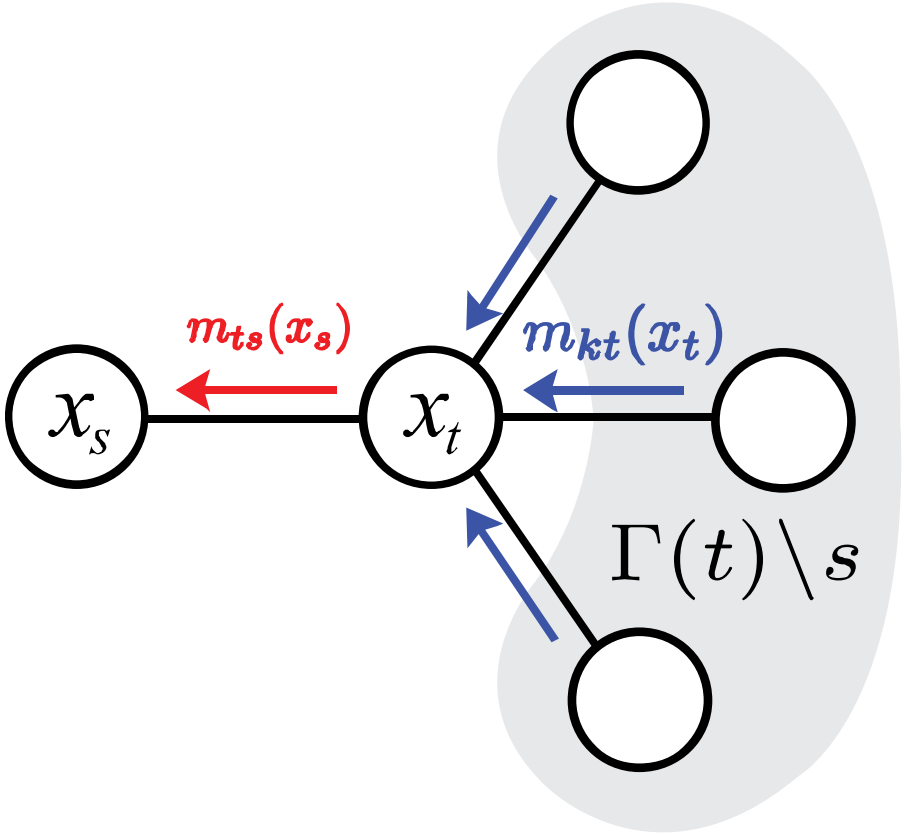
$$\alpha_{n-1}(x_n) \propto p(y_1, \ldots, y_n, x_n) \propto p(x_n \mid y_1, \ldots, y_n)$$

Smoothed posterior incorporates all observations:

$$p(x_n \mid y_1, \ldots, y_N) \propto p(x_n \mid y_1, \ldots, y_n) p(y_{n+1}, \ldots, y_N \mid x_n)$$
$$\propto \alpha_{n-1}(x_n) \beta_{n+1}(x_n)$$

Message Passing Inference

Breaks difficult global computations into simpler local updates



Many algorithms use some form of DP

- Belief propagation
- Gibbs sampling
- Particle filtering
- Viterbi decoder for HMMs
- Kalman filter (and variations of it)

This is a form of Dynamic Programming (DP)

Key Idea: Local computations only depend on the statistics of the current node and neighboring interactions

Marginal Likelihood

Recall posterior probability given by Bayes' rule:

$$p(x \mid \mathcal{Y}) = \frac{p(x)p(\mathcal{Y} \mid x)}{p(\mathcal{Y})}$$

Marginal likelihood often involves integral that lacks a closed-form, or exponential series of summations (discrete):

$$p(\mathcal{Y}) = \int p(x)p(\mathcal{Y} \mid x)\, dx \quad \text{or} \quad p(\mathcal{Y}) = \sum_{x_1,\ldots,x_N} p(x_1^N)p(\mathcal{Y} \mid x_1^N)\, dx$$

Shorthand: $x_1, \ldots, x_N$

As computer scientists, we will exploit graph structure to develop efficient algorithms to approximate each setting…

Approximate Inference

In most complex models the posterior lacks a closed-form…

Markov chain Monte Carlo (MCMC) sampling flexibly adapts to complex settings.

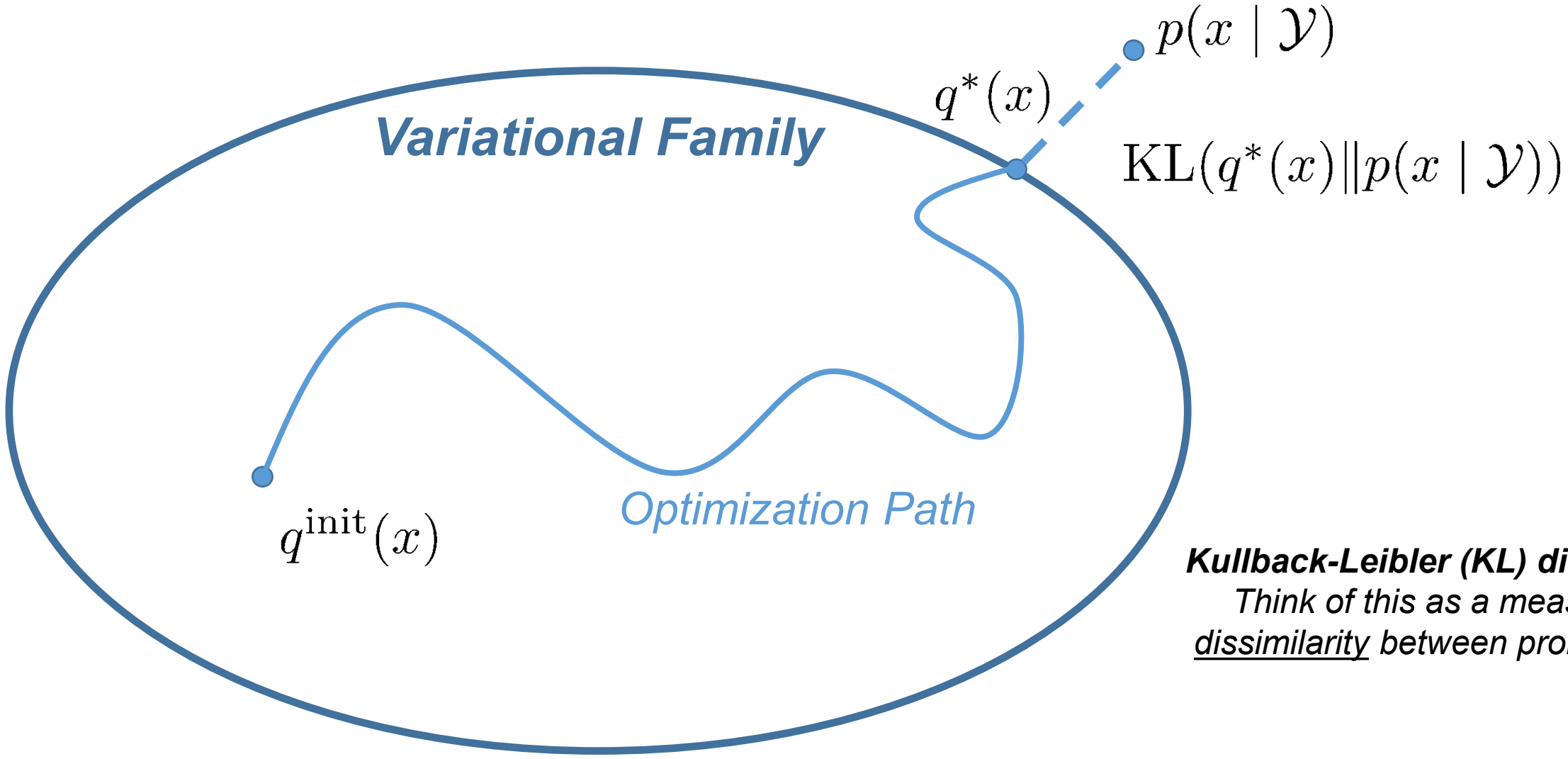
- Prohibitively slow in majority of complex (high-dim) settings
- Accuracy guarantees only hold in infinite limit
- Can be difficult to tune and diagnose convergence

Variational Inference Recasts inference as an optimization problem

- Elegantly scales to high-dimensional / large-data domains
- Diagnosing convergence is trivial
- Some downsides: minimal guarantees, biased in general, less flexible

Variational Approximation : Intuition

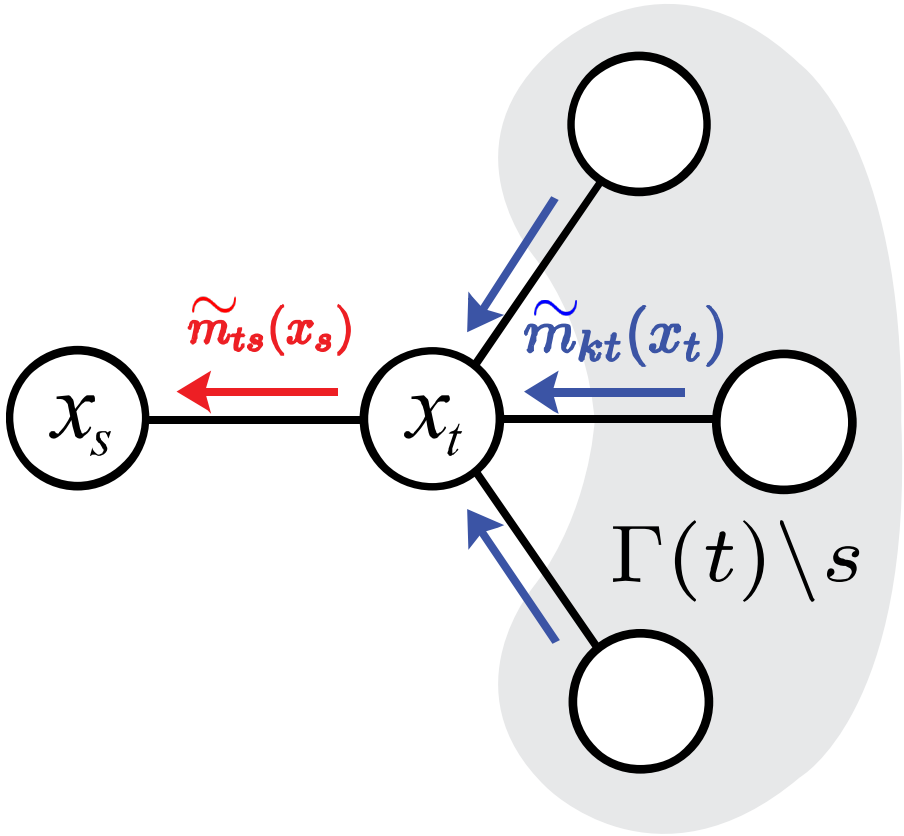
Find approximate of posterior $q(x) \approx p(x \mid \mathcal{Y})$ *in nice family*



Kullback-Leibler (KL) divergence
Think of this as a measure of dissimilarity between probabilities.

[Adapted From: David Blei]

Variational Inference as Message Passing



Key Idea: Replace *exact* message updates with approximations,

$$\widetilde{m}_{ts}(x_s) \approx m_{ts}(x_s)$$

Variational posterior is product of incoming messages from neighbors,

$$q(x_s) \propto \prod_{k \in \Gamma(t)} \widetilde{m}_{kt}(x_t)$$

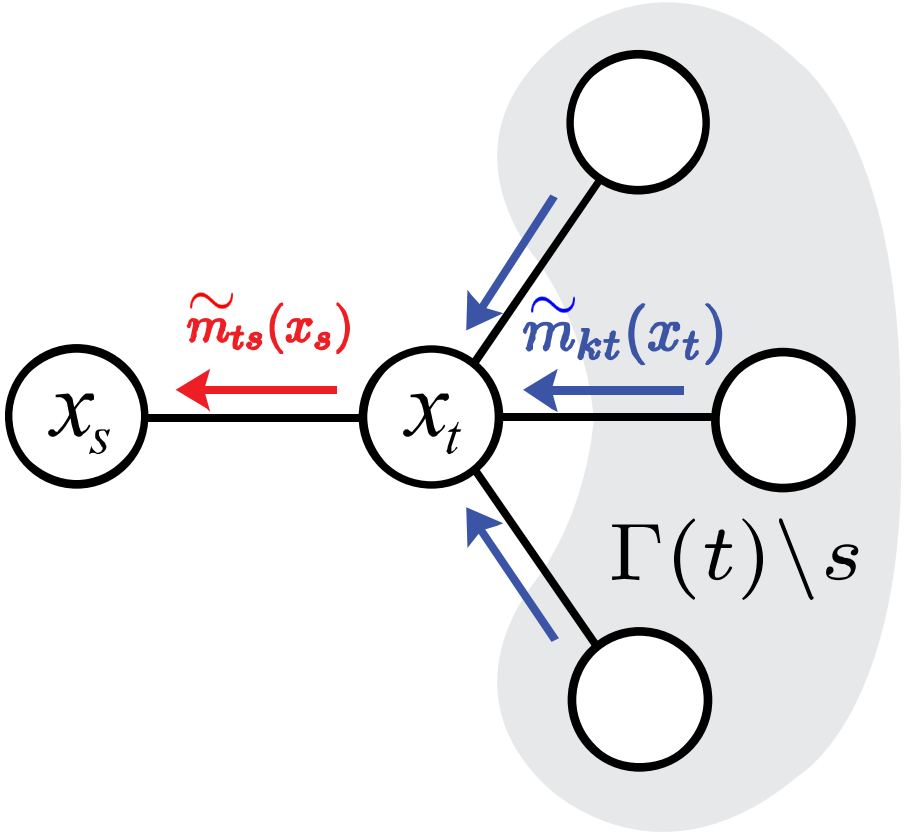
Approximates posterior in minimum KL-sense:

$$q(x_s) \approx p(x_s \mid \mathcal{Y})$$

Observe: *Updates still only rely on subgraph that involve immediate neighbors. Can be easily parallelized. Efficient!*

Approximate Variational Message Passing Inference

Different algorithms depending on chosen variational family

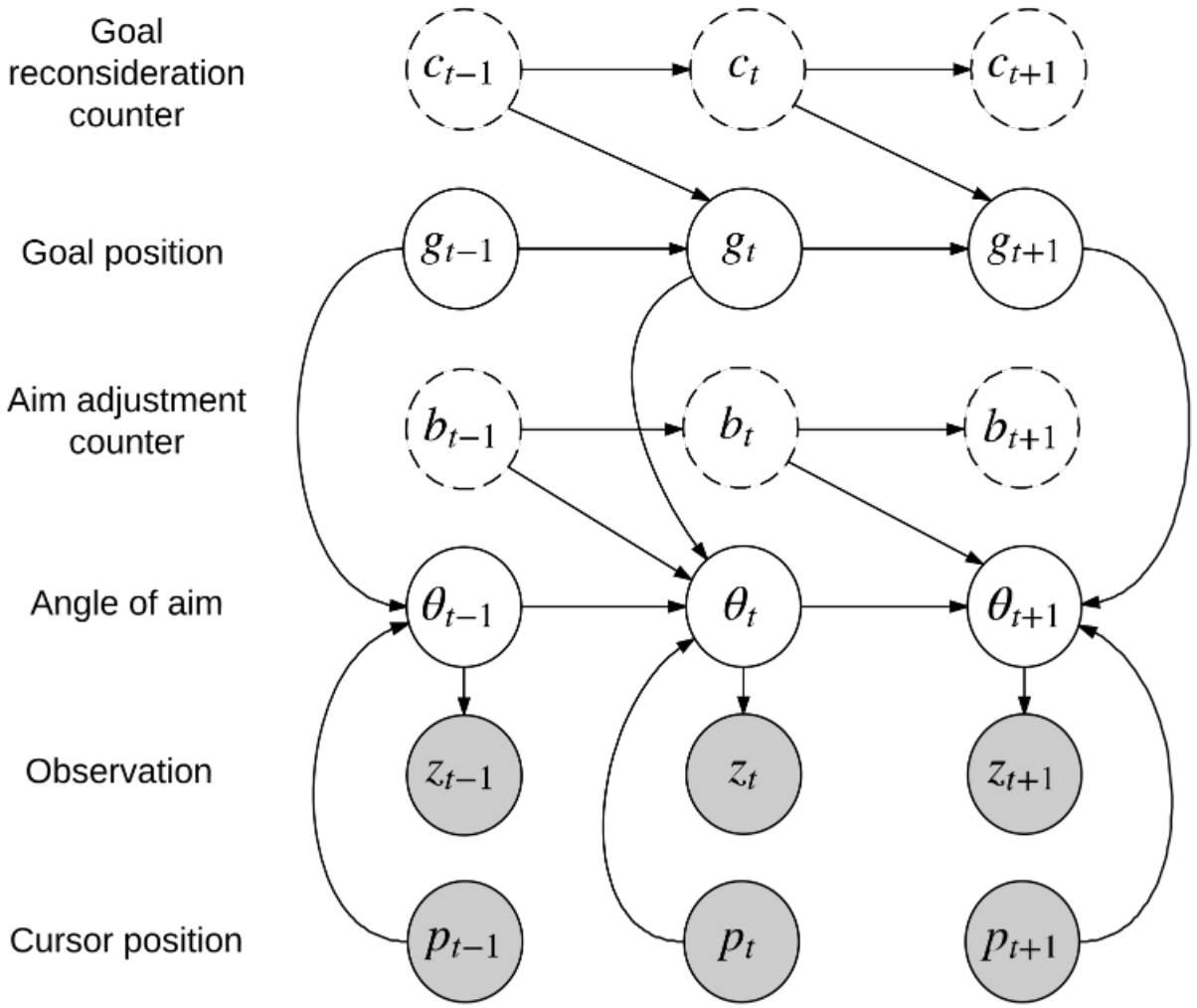


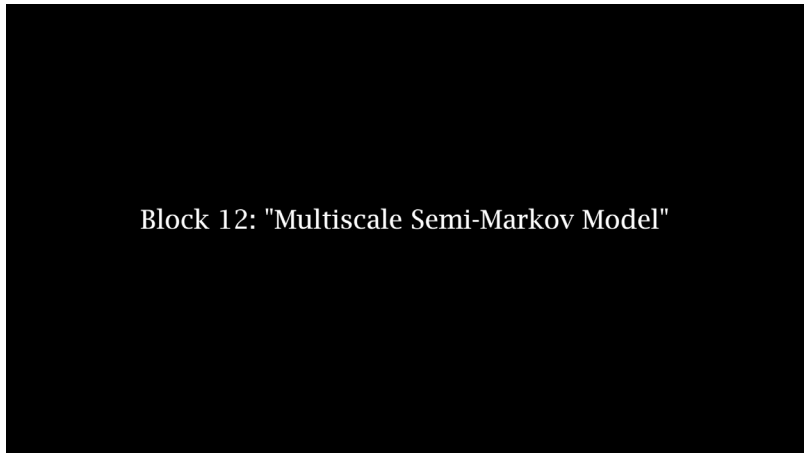
Some variational algorithms

- Loopy belief propagation (LBP)
- Expectation Propagation
- Mean Field / Structured Mean Field
- Normalizing flows variational
- Extended / Unscented Kalman Filter
- Nonparametric BP
- Particle BP

I am intentionally glossing over technical details…

Intracortical Brain-Computer Interfaces (iBCI)





Goal Given observed neural activity and cursor position infer user intended motion of cursor and goal location.

$$p(\theta_t, g_t \mid z_1^t, p_1^t)$$

Marginal Posterior

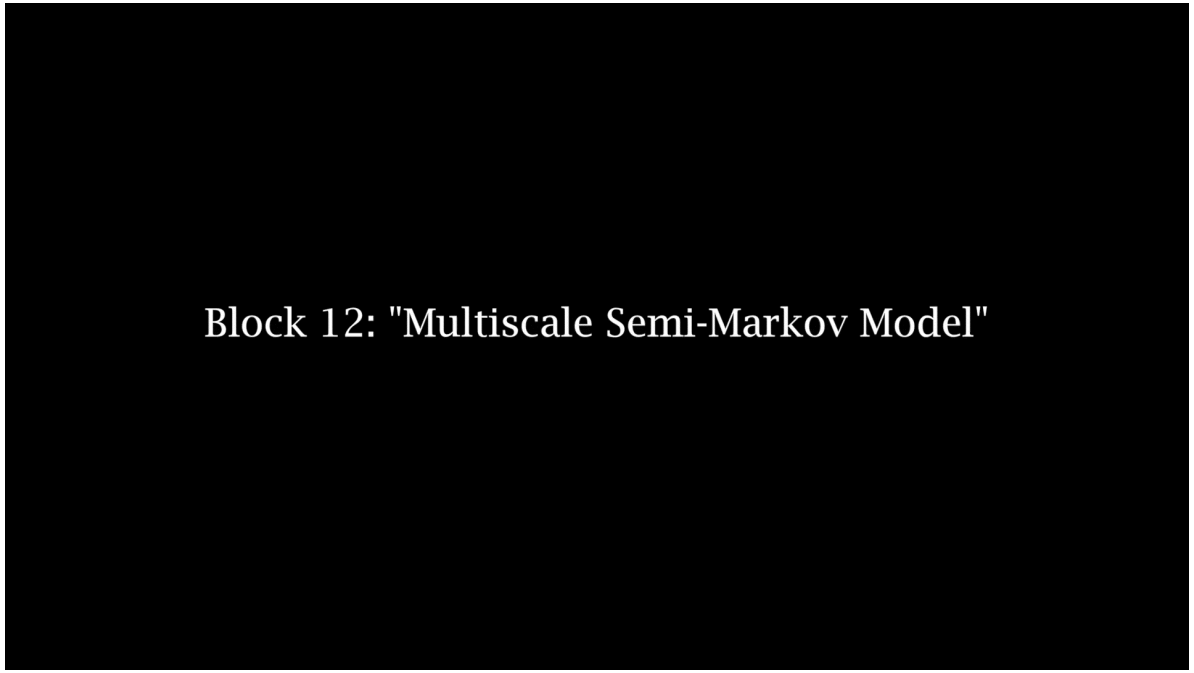
Observe Each model component sufficiently low-dimensional to discretize.

Approach Compute discrete BP messages

D. Milstein, J. Pacheco, et al. "Multiscale Semi-Markov Dynamics for Intracortical BCIs", NIPS, 2017.

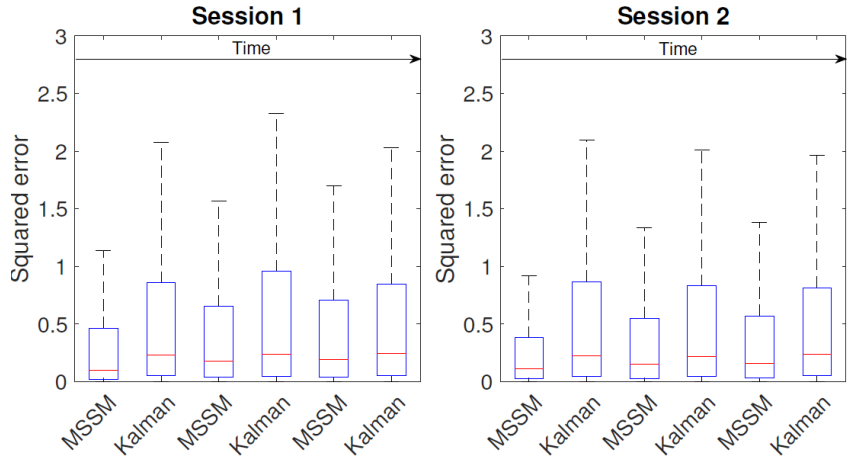
iBCI : MSSM vs. Kalman Filter

Multiscale Semi-Markov Model (MSSM)



Linear Gaussian (Kalman Filter)

- Allows modelling of nonlinear / non-Gaussian dynamics
- **Dependency structure allows for discretization of each random variable**
- Discrete approximation belief propagation meets real-time constraints

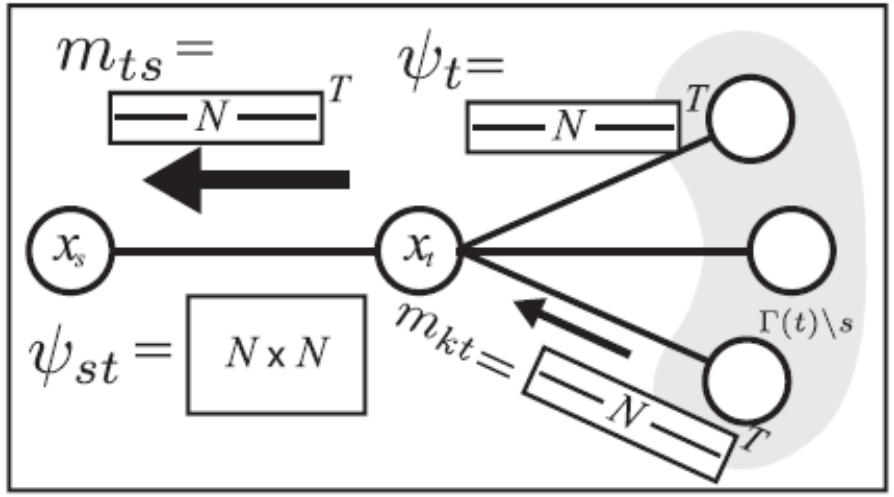


D. Milstein, J. Pacheco, et al. "Multiscale Semi-Markov Dynamics for Intracortical BCIs", NIPS, 2017.

Belief Propagation : Discrete vs. Continuous

In iBCI we were able to discretize the latent state…

Discrete

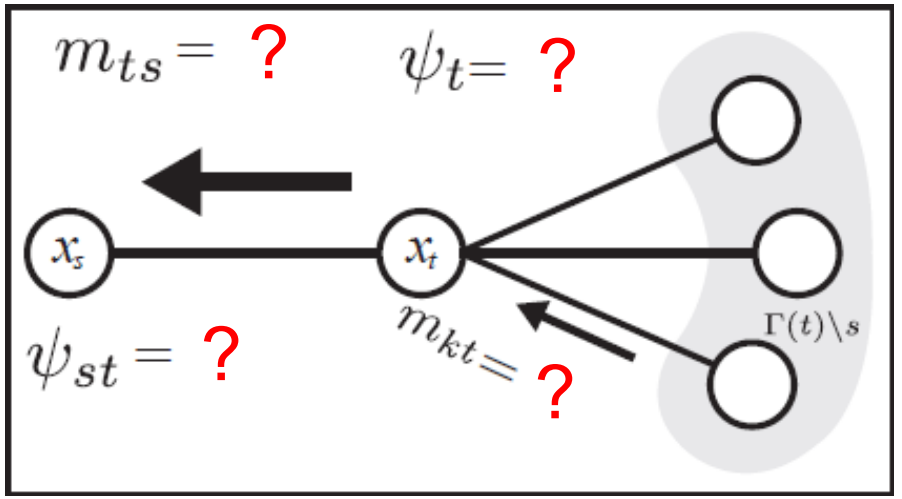


Message Update:

$$m_{ts} = \sum_{x_t} \boxed{\psi_{st}} \boxed{\psi_t \prod m_{kt}}$$

Matrix-vector multiplication

Continuous



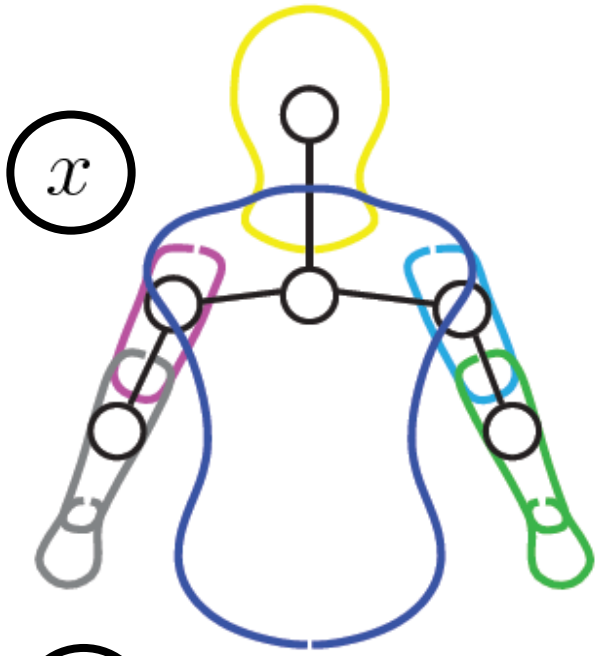
Message Update:

$$m_{ts}(x_s) = \int \psi_{st}(x_s, x_t)\psi_t(x_t) \prod_k m_{kt}(x_t)$$

No Closed-Form

…but this is often infeasible…

Articulated Pose Estimation

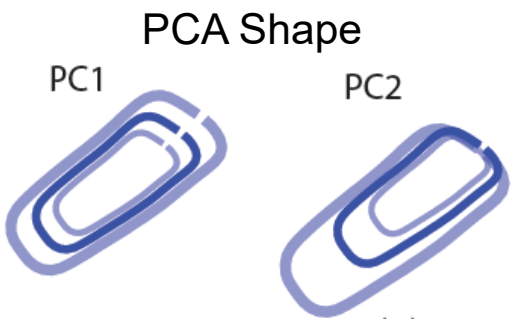


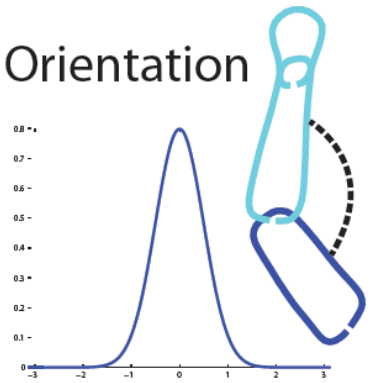
$$p(x,y) \propto \prod_{s \in \mathcal{V}} \psi_s(x_s, y) \prod_{(s,t) \in \mathcal{E}} \psi_{st}(x_s, x_t)$$

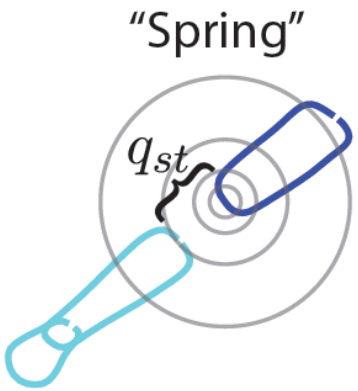
Complicated Likelihood

Non-Gaussian Prior

Latent state $x_s \in \mathcal{X}_s$ for part *shape*, *location*, *orientation* and *scale*.



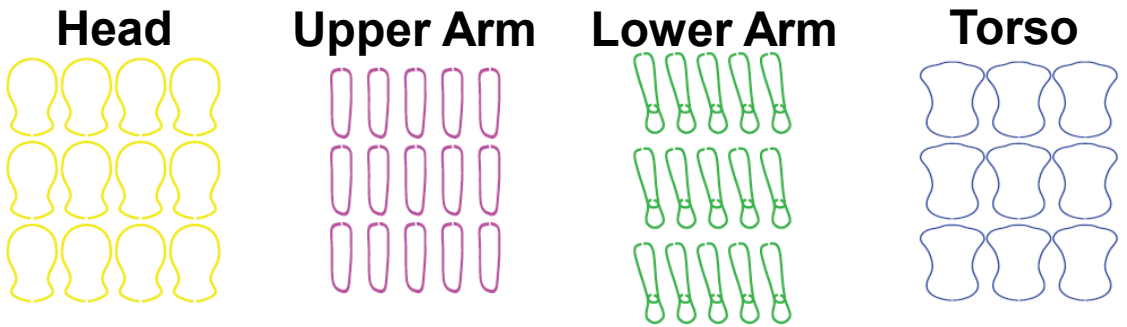




High-dimensional inference over continuous quantities…

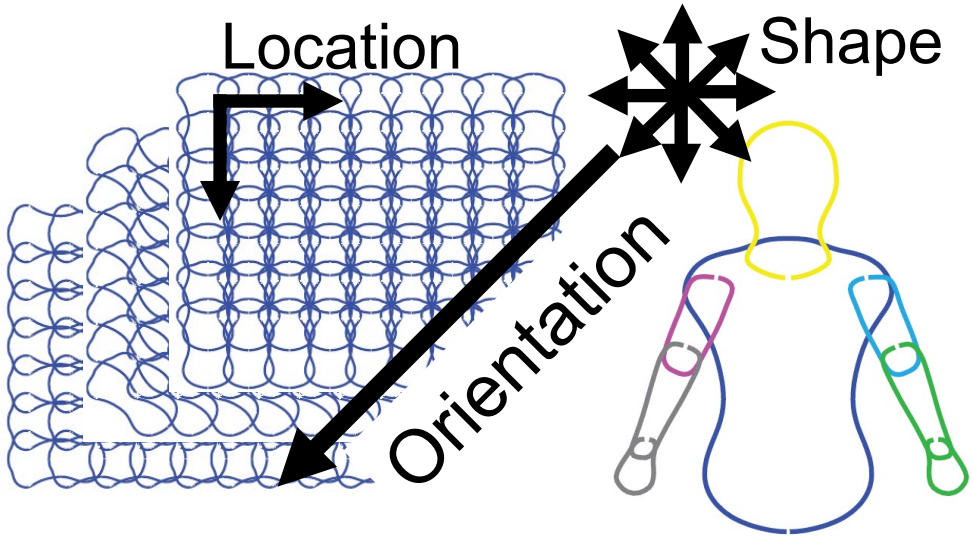
Regular Discretization

Approximate continuous max-product messages over regular grid of *points*



Example: Torso

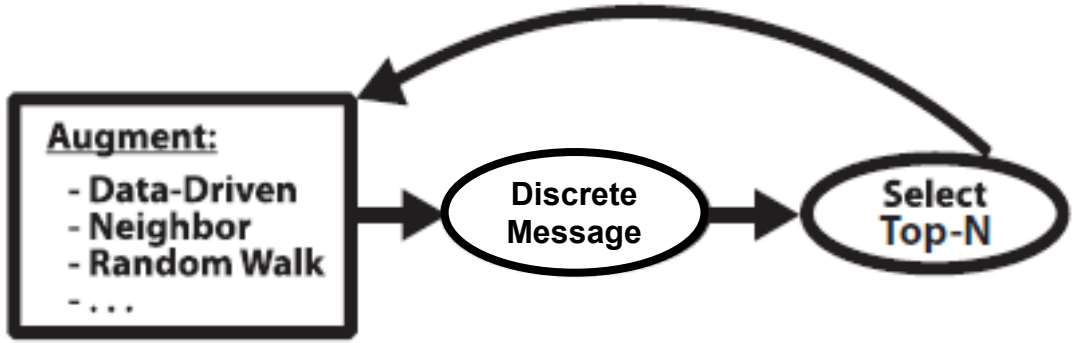
- ~10 dimensions.
- 10 grid points per dimension
- 10 Million points!

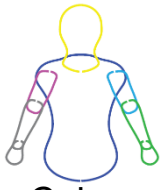


Infeasible for high dimensional models.

Top-N Particle Max Product (T-PMP)

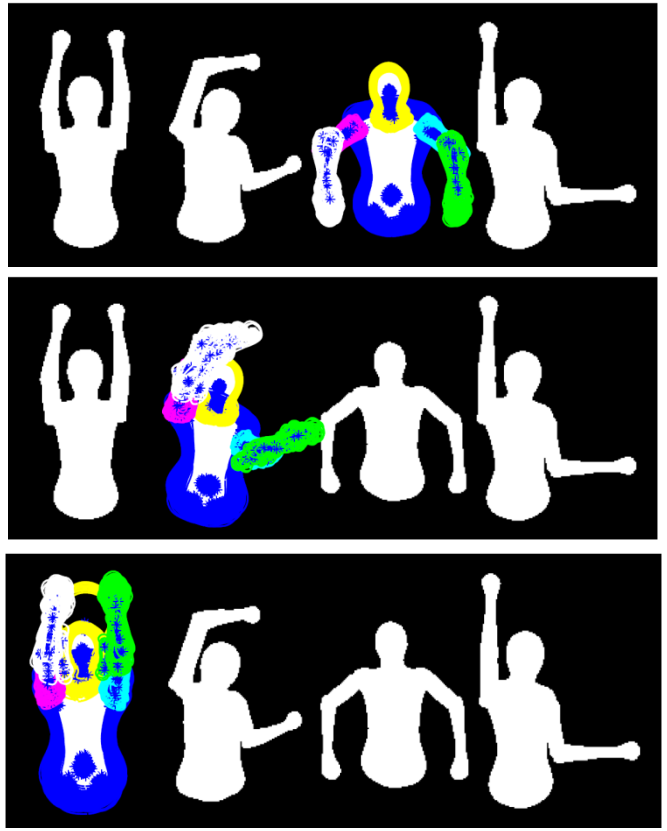
Don't use regular discretization – sample discrete configurations from proposal distributions





Colors

Example Runs



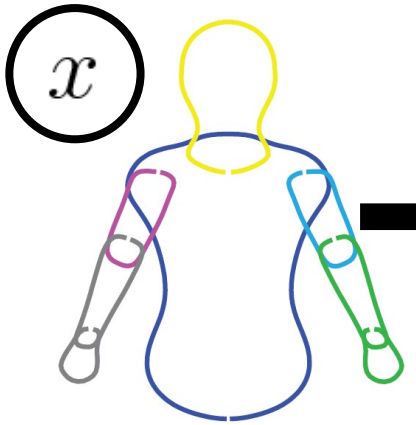
- Keep N-best particles
- Sensitive to initialization
- Selection reduces effective particles

Idea: Maintain *diversity* in particles.

J. Pacheco, et al. "Preserving Modes and Messages via Diverse Particle Selection." ICML 2014

Probabilistic Reasoning

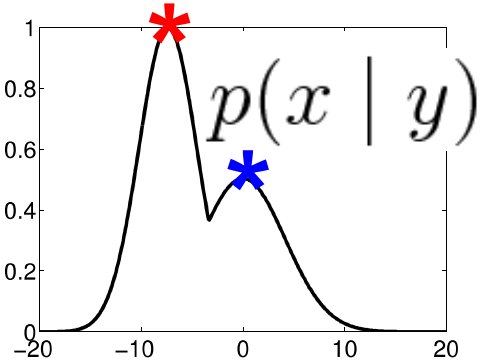
Data **Unknowns**



$p(x, y)$
Probability Model

Inference

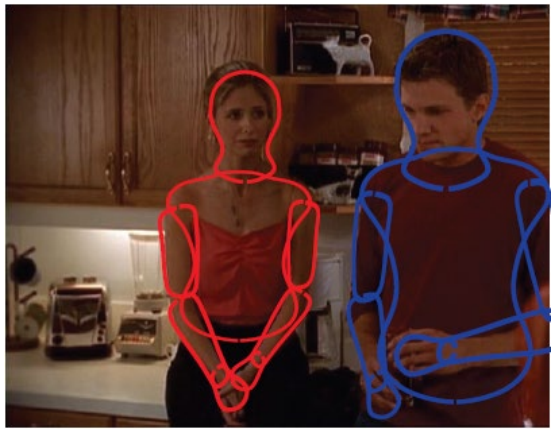
Posterior Belief



Posterior often intractable and multimodal complicating maximum a posteriori (MAP):

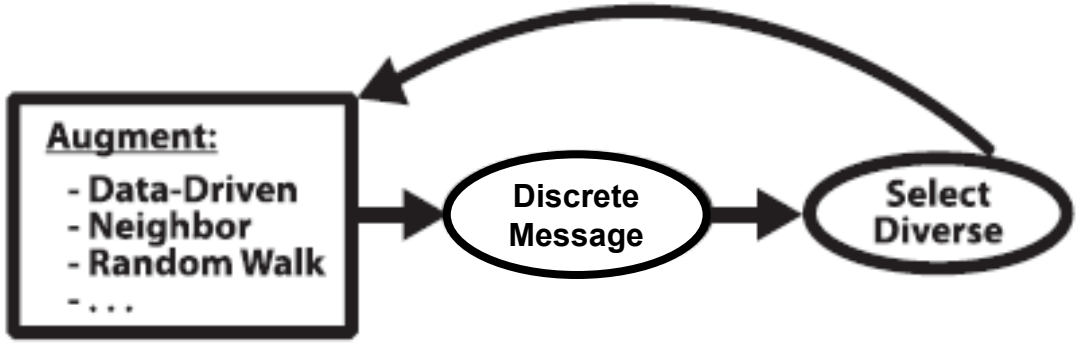
$$x^* = \underset{x}{\operatorname{argmax}}\, p(x \mid y)$$

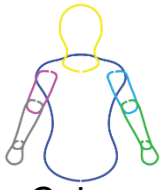
MAP Estimate



Local optima can be useful when models are inaccurate or data are noisy.

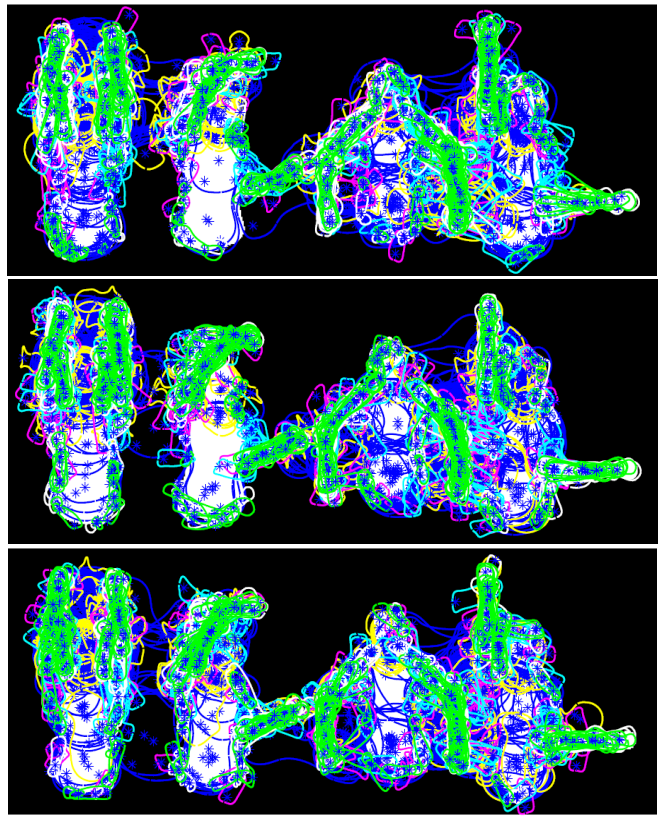
Diverse Particle Max-Product (D-PMP)





Colors

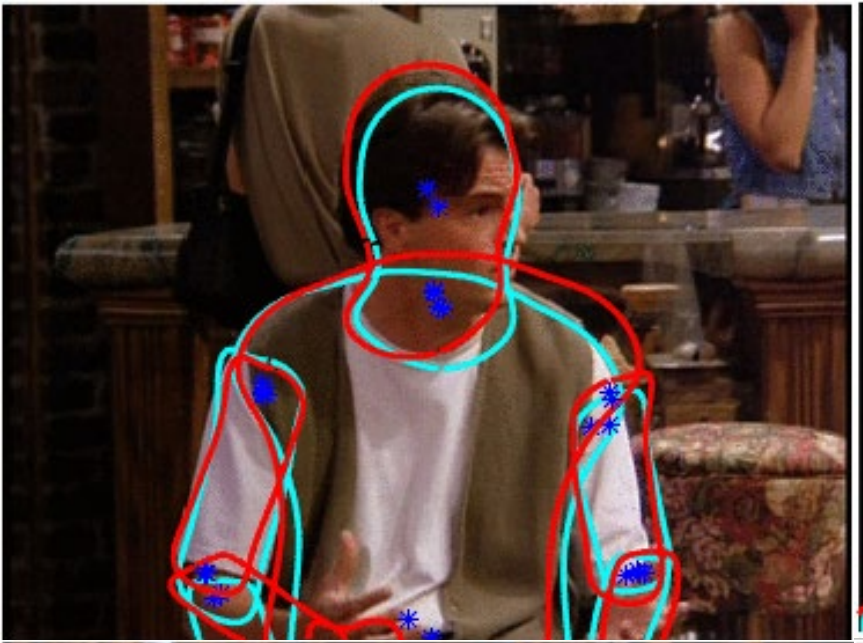
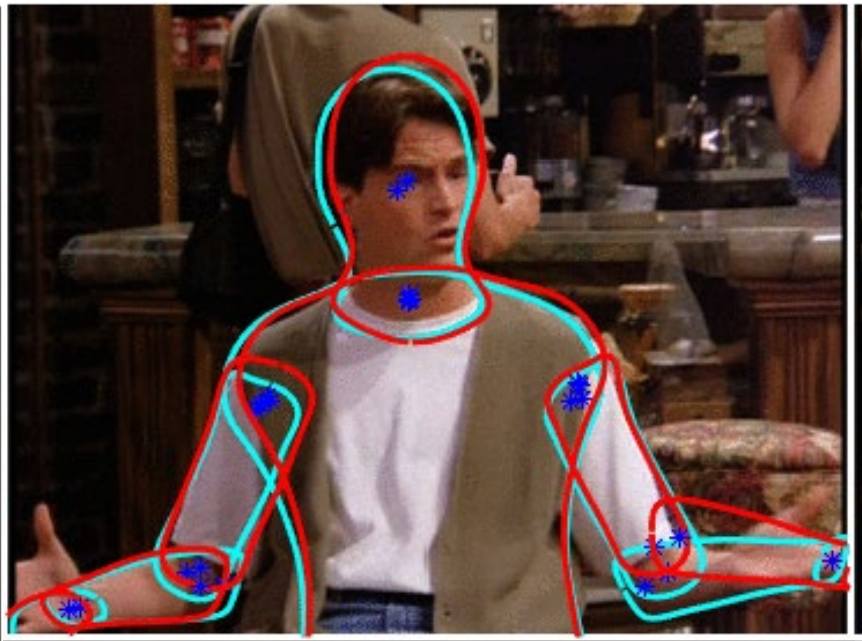
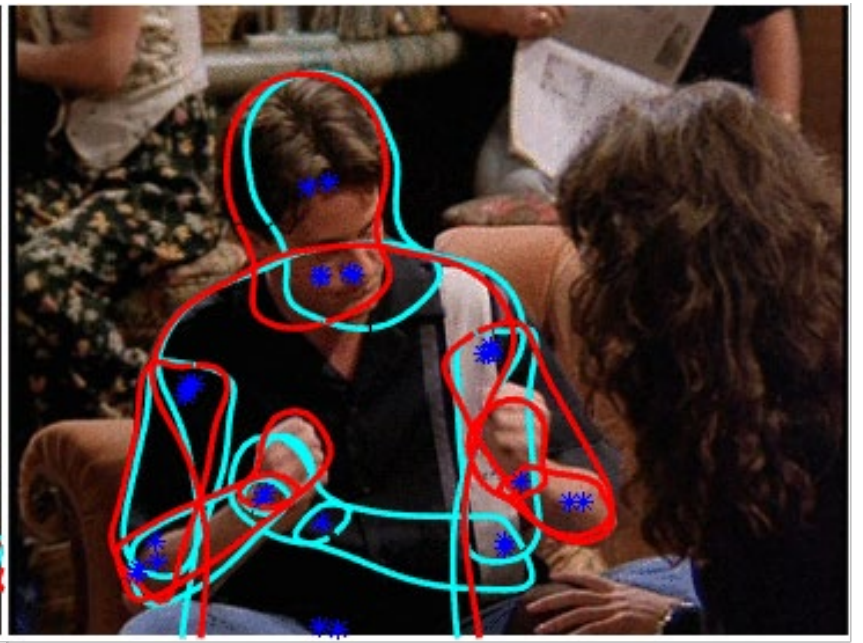
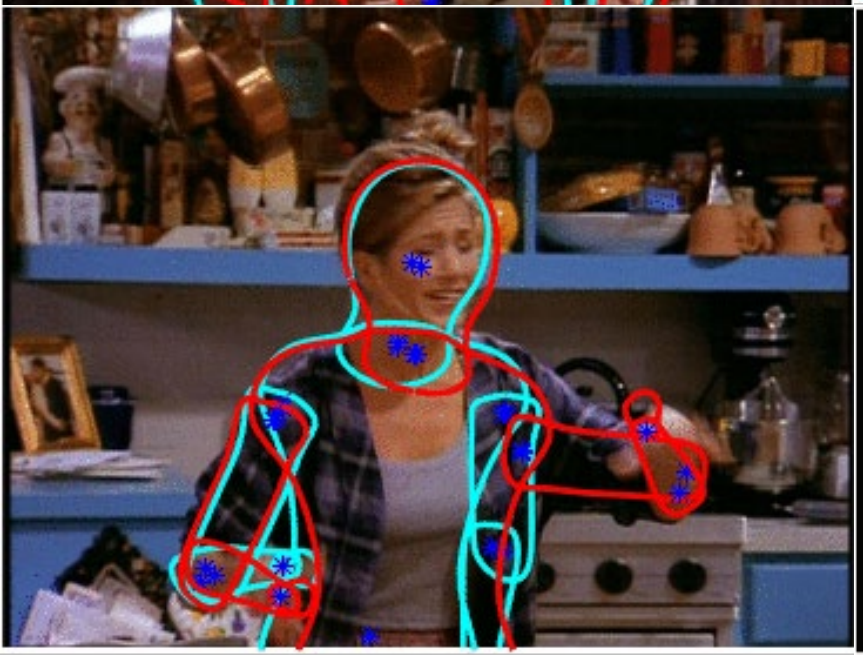
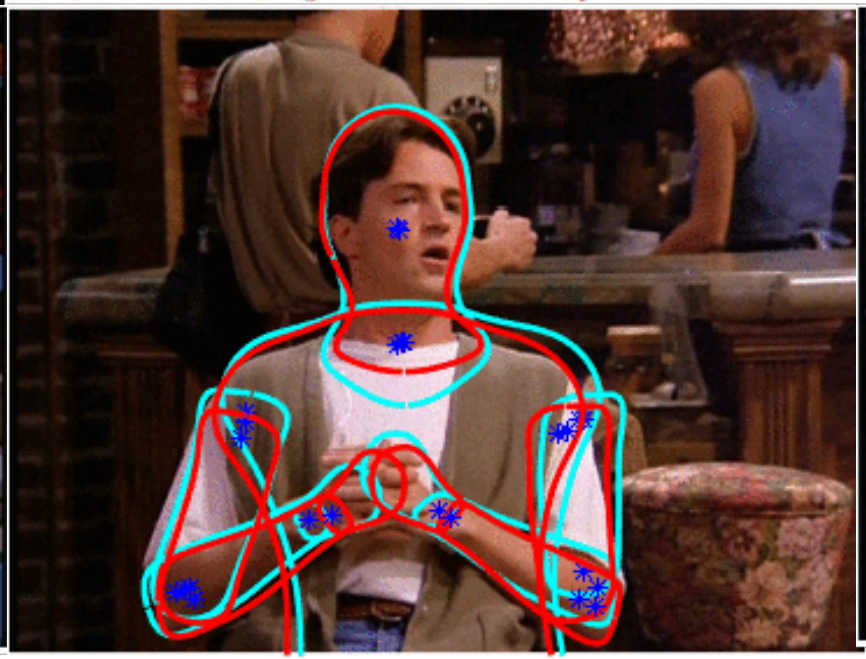
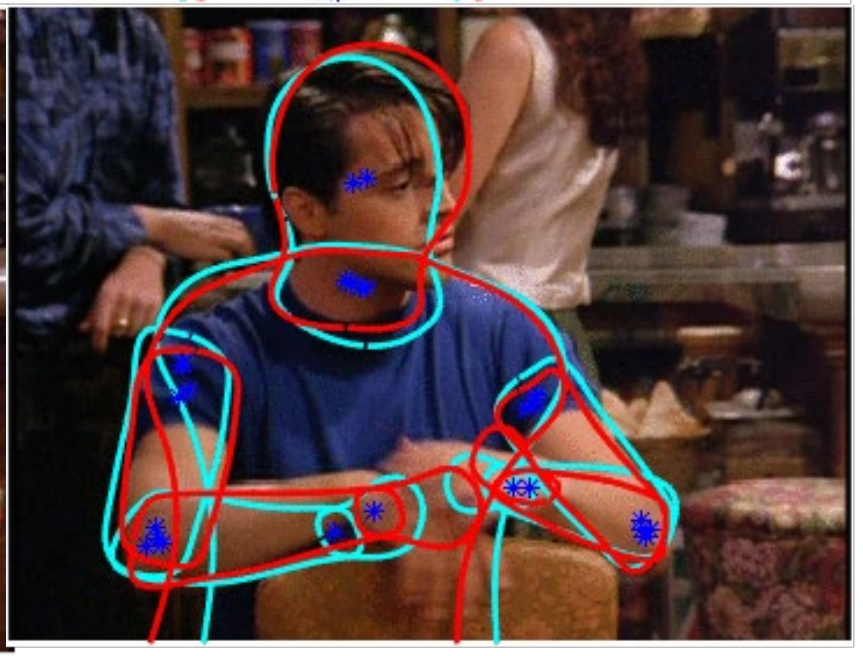
Example Runs



- No explicit diversity constraint
- Objective encourages diversity
- Efficient *Lazy* greedy algorithm
- Bounds on optimality

Avoids particle degeneracies by maintaining *ensemble* of *diverse solutions* near local modes.

J. Pacheco, et al. "Preserving Modes and Messages via Diverse Particle Selection." ICML 2014

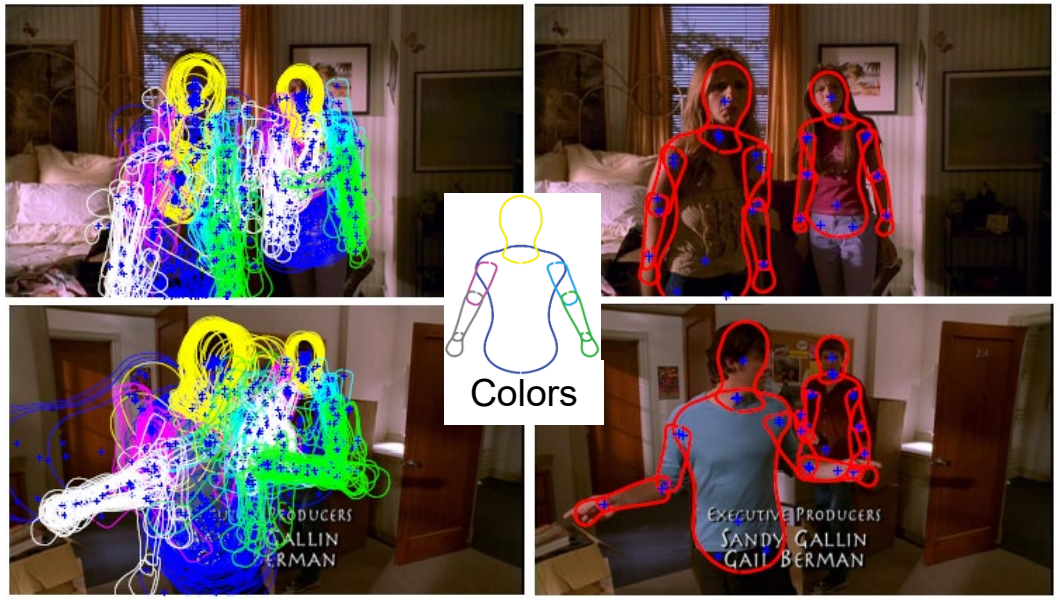







Diversity at Large-and-Small Scales

Diverse Estimates of Parts

Top 3 arm hypotheses MAP estimate, 2nd and 3rd modes for upper arm (magenta, cyan), lower arm (green, white).

Multi-Person Estimates (despite single-person model)



D-PMP Particles **Mode Estimates**

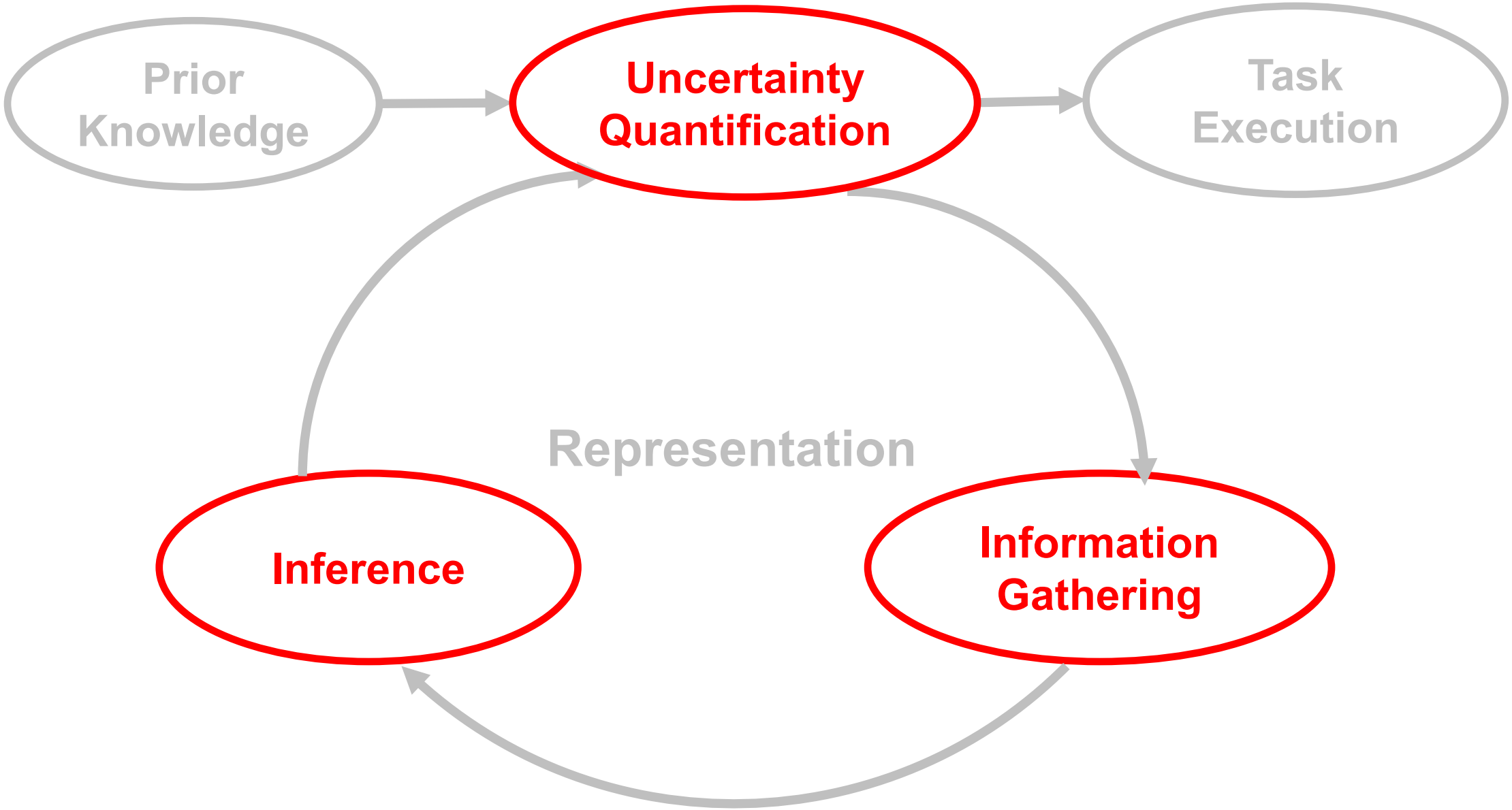
J. Pacheco, et al. "Preserving Modes and Messages via Diverse Particle Selection." ICML 2014

Current Directions of Work

Recently, my interest is in aspects of decision making:

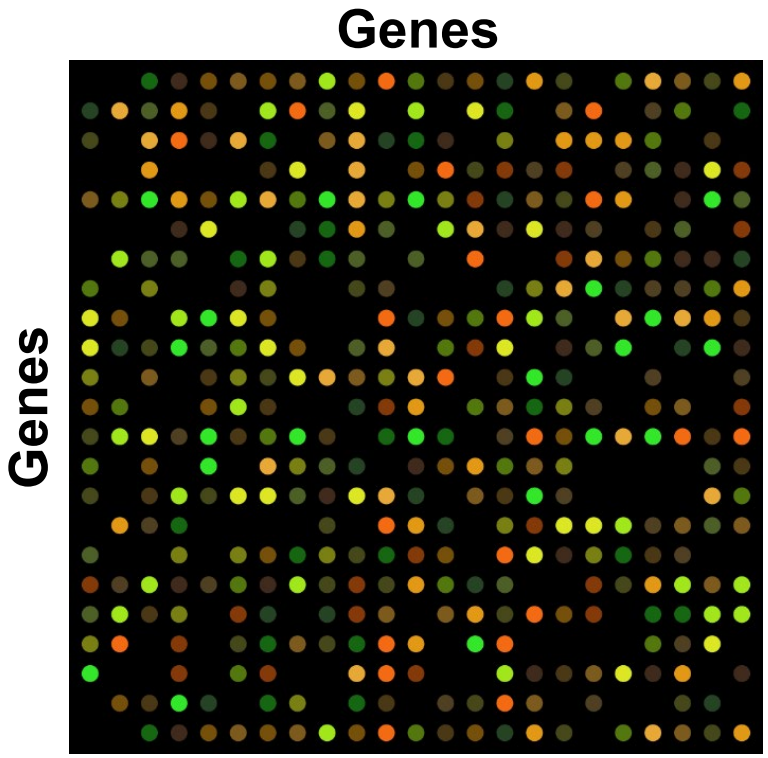
- Estimating information-theoretic measures
- Bayesian optimal experimental design (BOED)
- Control problems maximizing information gain
- Variational approaches to reinforcement learning (RL)

Probabilistic Reasoning

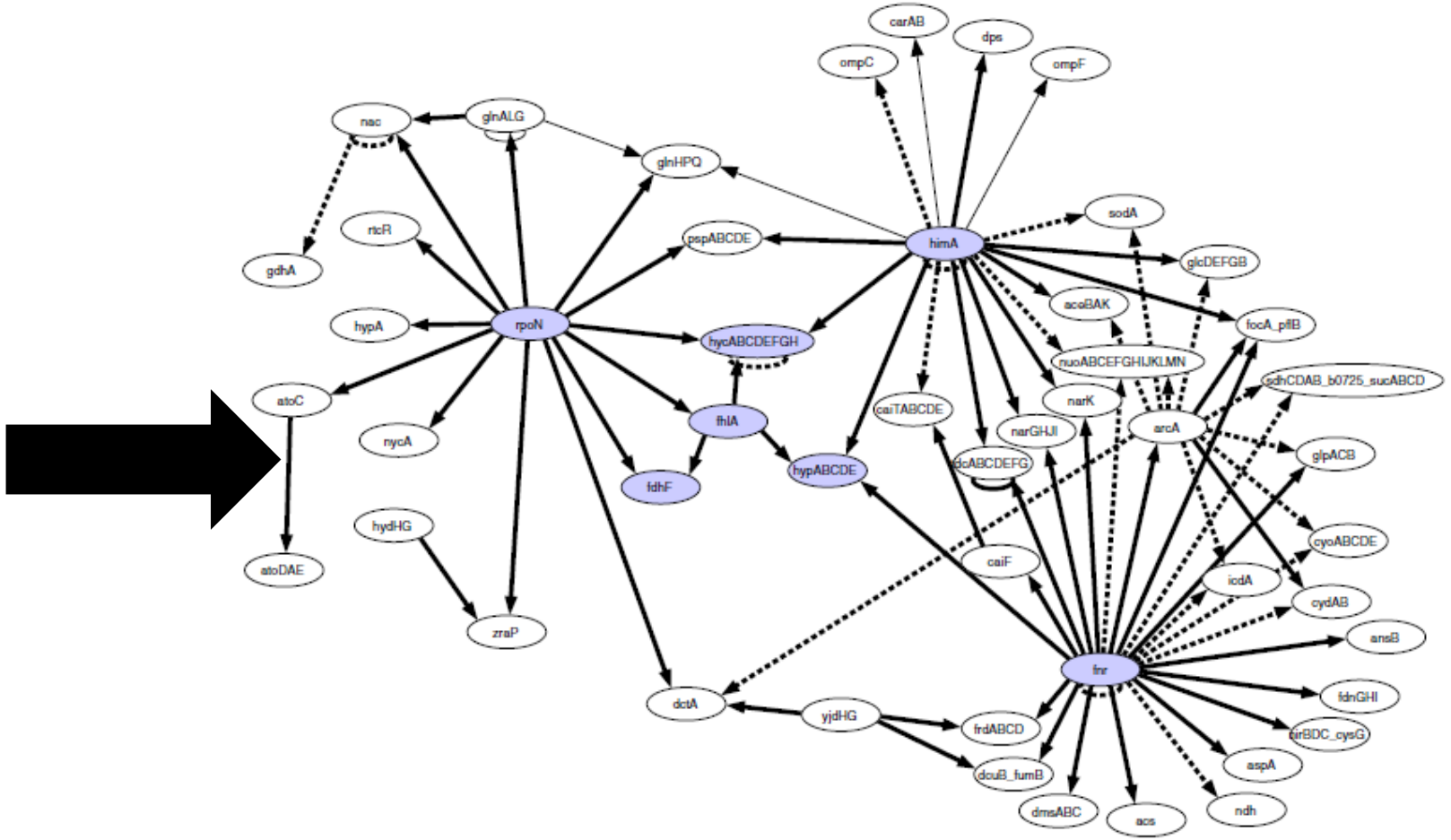


Example: Gene Regulatory Network

Gene Expression



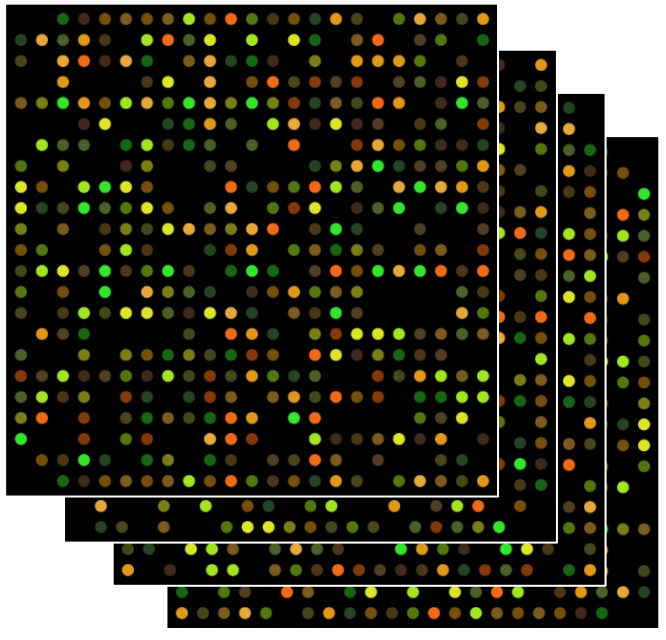
Regulatory Network



[Image: Bulcke et al., 2006]

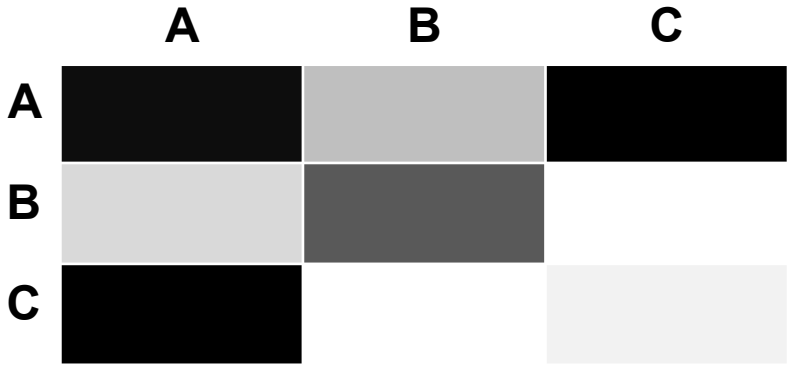
Goal: Estimate causal interaction network from expression data.

Identifying Causality

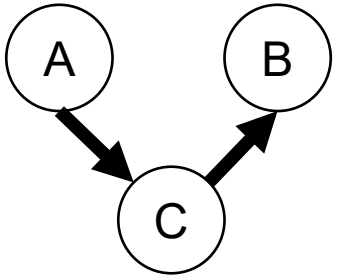


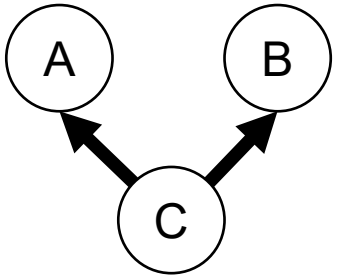
Dataset

Covariance Matrix

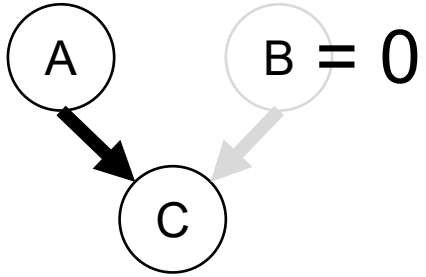


Possible Graphs



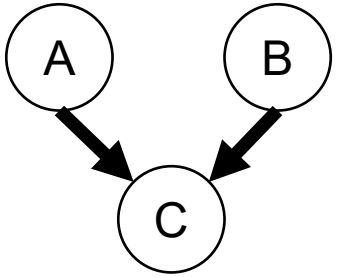


Cannot determine causality from correlations, need to perform active interventions …

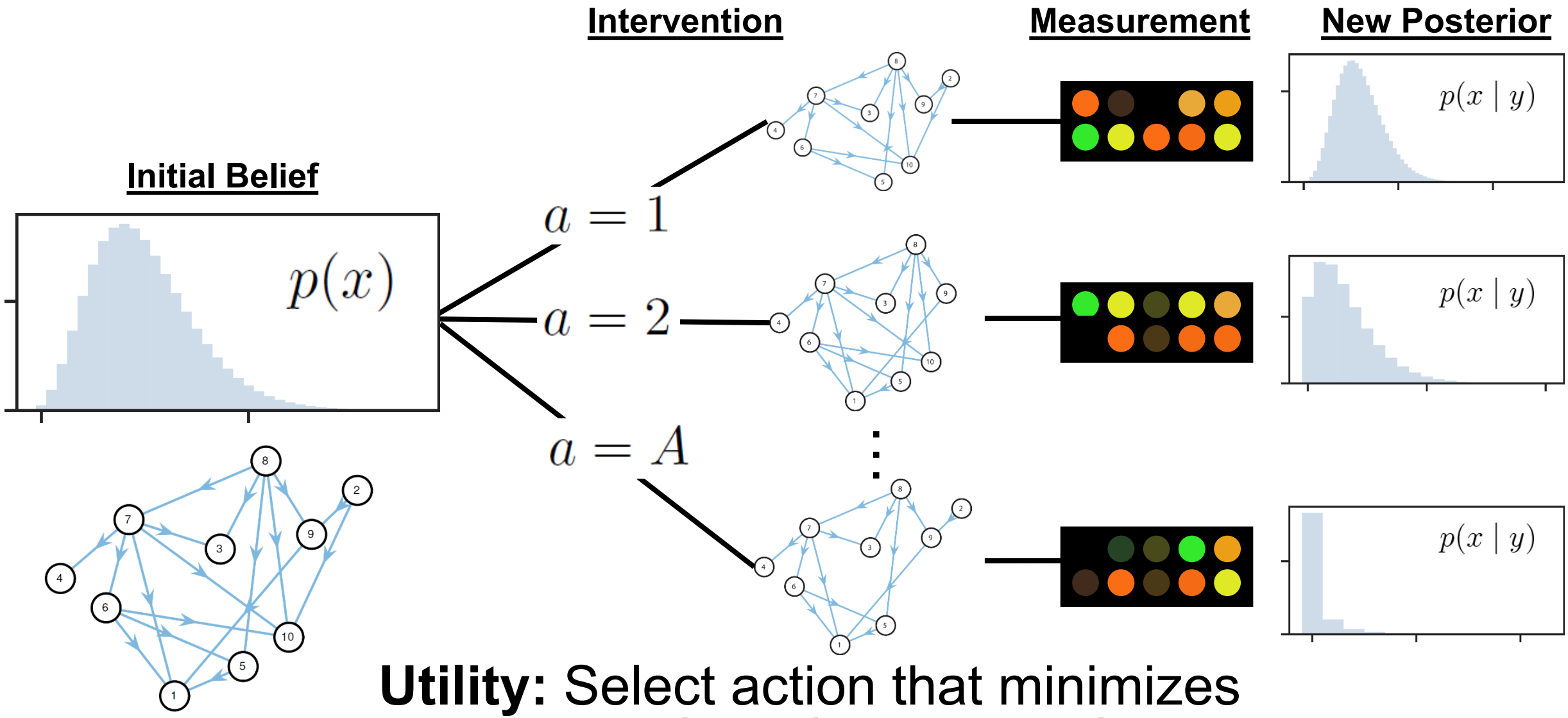


Clamp node to fixed value.

Gene Knockout



Choosing Actions

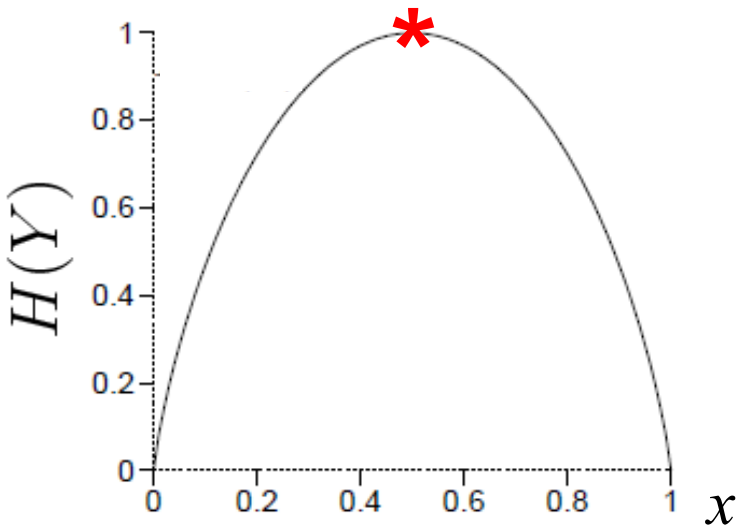


Utility: Select action that minimizes uncertainty about network

Uncertainty and Information

$$H(Y) = \mathbb{E}[-\log p(Y)]$$

Coin Flip Example: $Y \sim \text{Bernoulli}(x)$

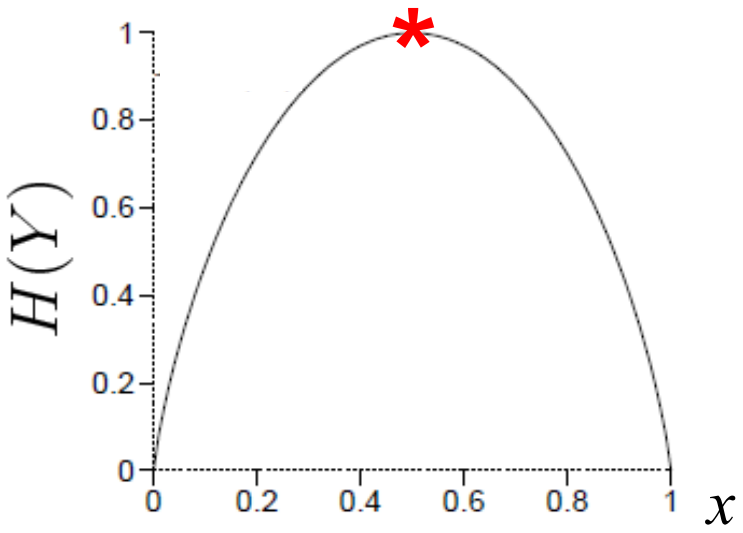


Maximum uncertainty when coin is fair.

Uncertainty and Information

$$H(Y) = \mathbb{E}[-\log p(Y)]$$

Coin Flip Example: $Y \sim \text{Bernoulli}(x)$



Maximum uncertainty when coin is fair.

Mutual Information

$$I(X;Y) = H(X) - H(X \mid Y)$$

- Measures entropy reduction after observing Y
- How much information does Y carry about X?

Computing MI as hard as doing inference.

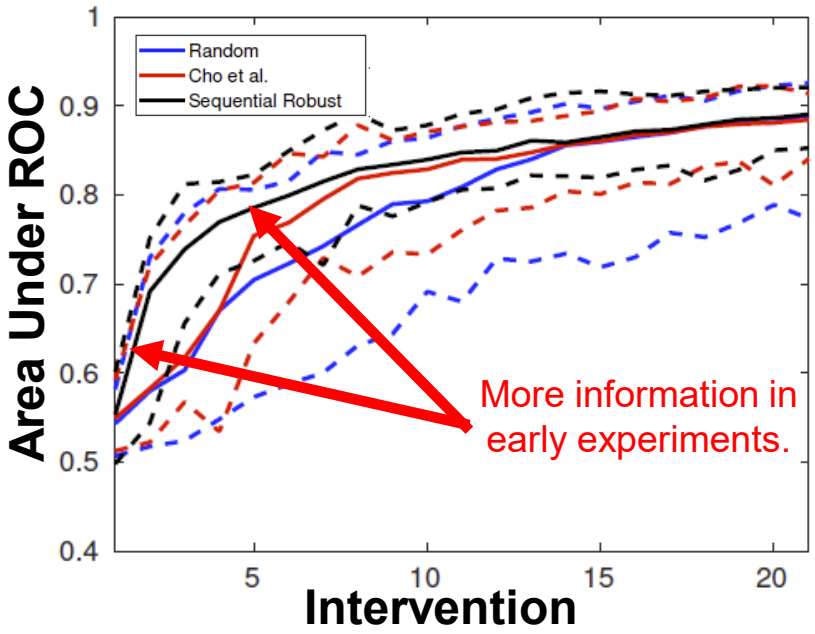
Variational MI Bound

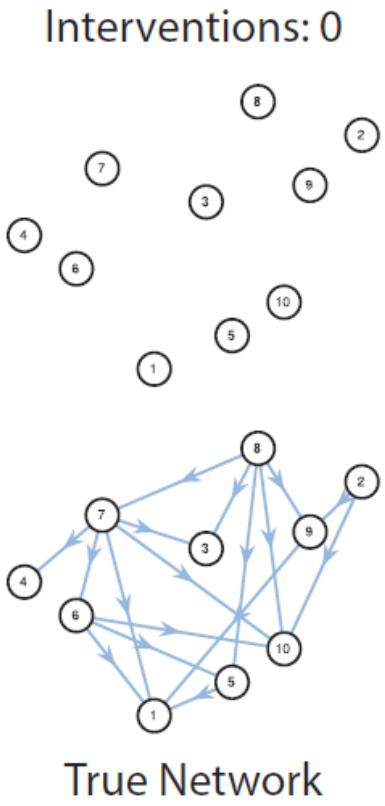
Lower bound mutual information using Gibbs' inequality,

$$I(X;Y) \geq \max_{q} H(X) + \underbrace{\mathbf{E}_p[\log q(X \mid Y)]}_{\text{(Negative) Cross Entropy } H_p(q)}$$

- $q(X|Y)$ is variational approximation in *exponential family*
- Includes many common distributions (Normal, Gamma, Bernoulli, Poisson, etc.)
- Results in efficient optimization of lower bound,

Robust Information-Theoretic Planning





Higher recall/precision w/ fewer interventions than baseline

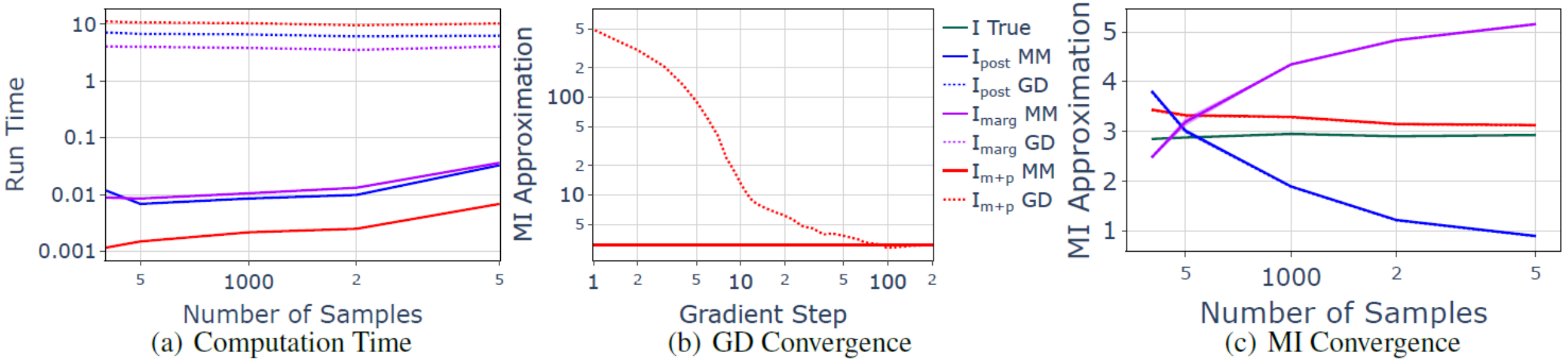
S. Zheng, J. Pacheco, J. Fisher III, "A Robust Approach to Sequential Information Theoretic Planning." ICML 2018

J. Pacheco and J. Fisher III, "Variational Information Planning for Sequential Decision Making." AISTATS 2019

Fast MI Bounds and Approximations

*With a single **fast** moment-matching operation we can solve MI upper-bounds, lower-bounds, and non-bound approximation*

(Example for high-dimensional Gaussian Mixture density)



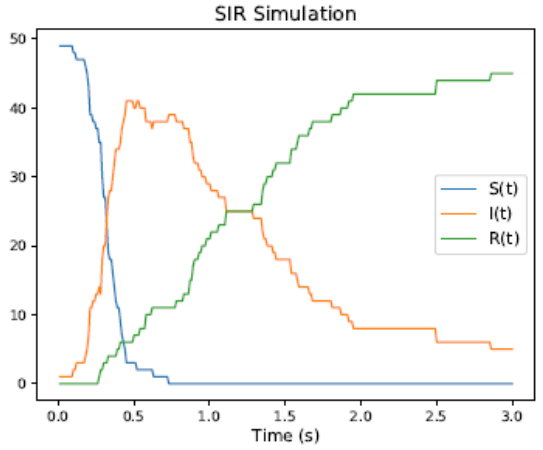
Orders of magnitude faster than standard gradient-based optimization

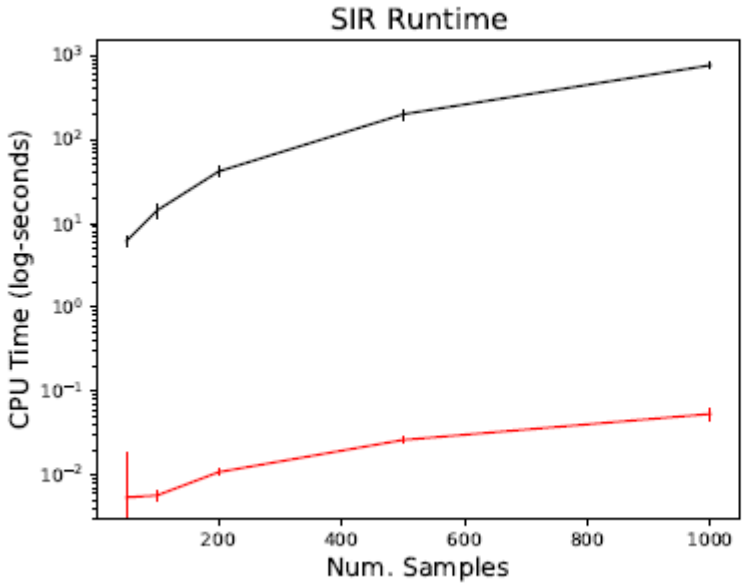
C. Dahlke, S. Zheng, J. Pacheco, AISTATS 2023 (Under Review)

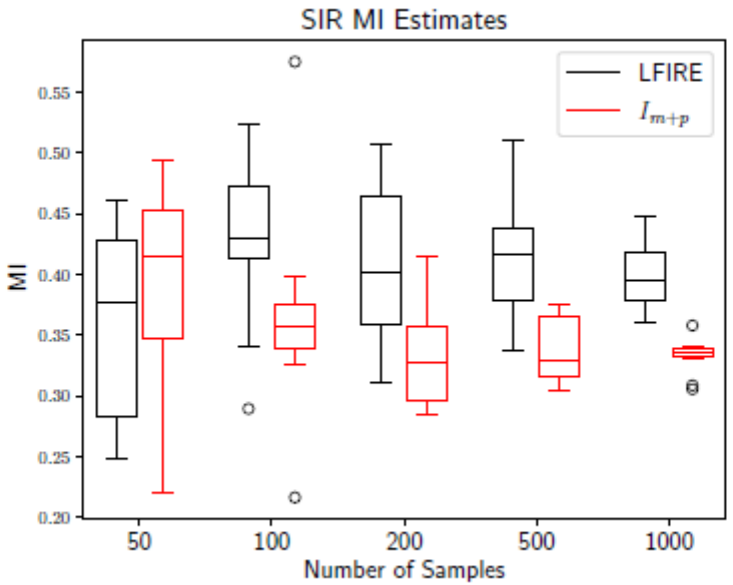
Variational MI for Implicit Likelihood Models

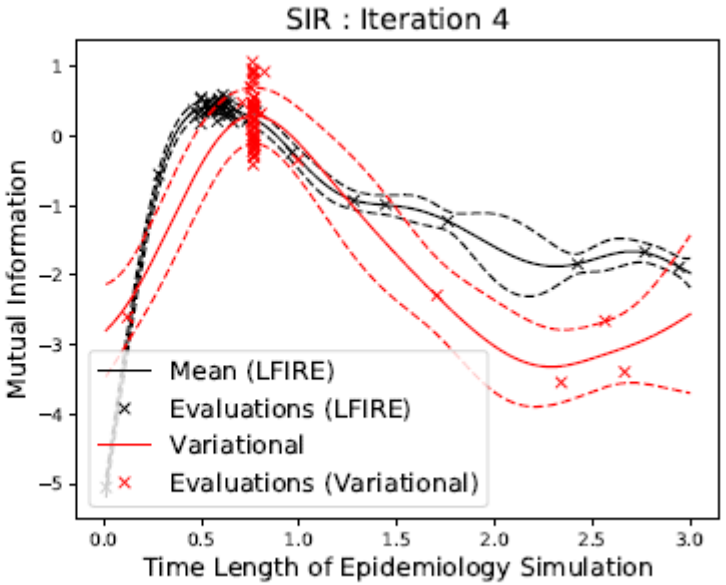
The same calculation yields MI estimation for models defined via simulation…

Example: SIR Epidemiologic Model





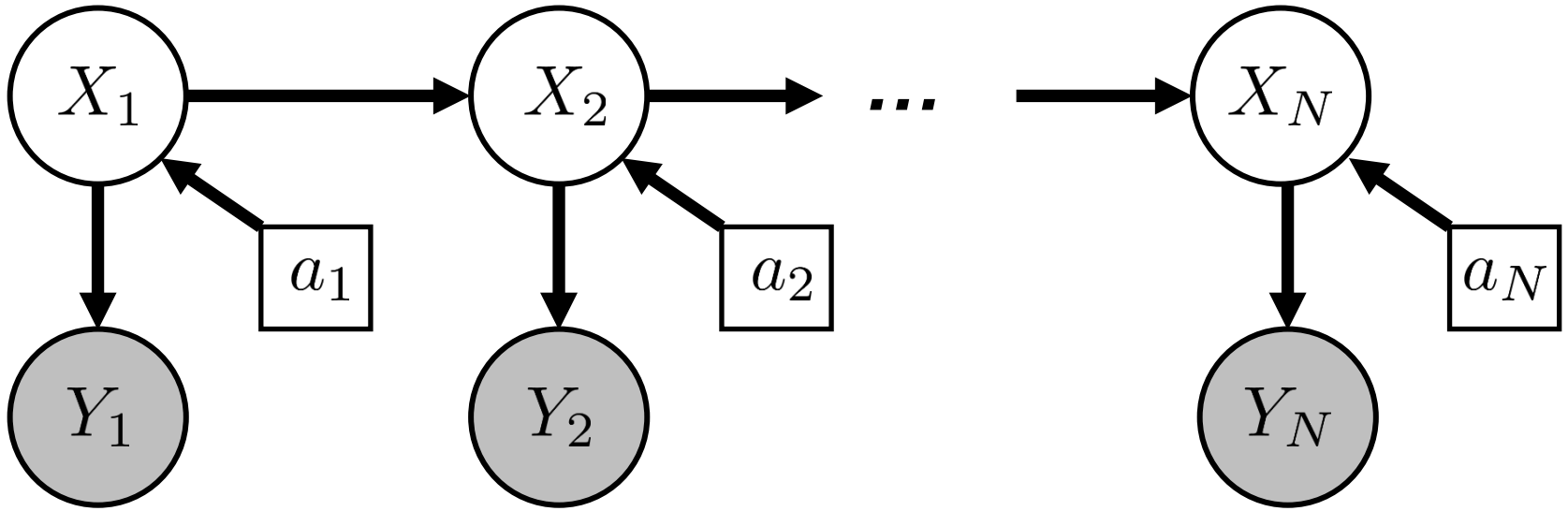




C. Dahlke, S. Zheng, J. Pacheco, AISTATS 2023 (Under Review)

Optimal Information Control

Extends to the setting where action choices alter time-varying latent state…

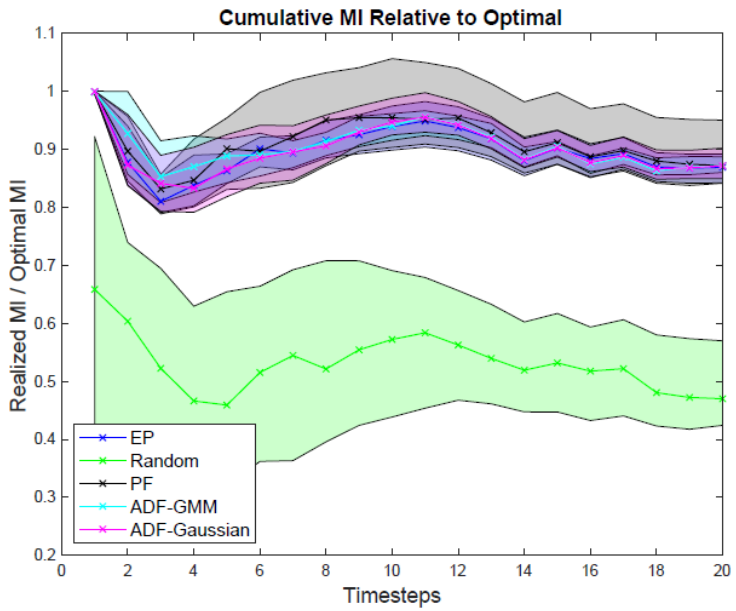


Goal Find actions / policy that maximize total information

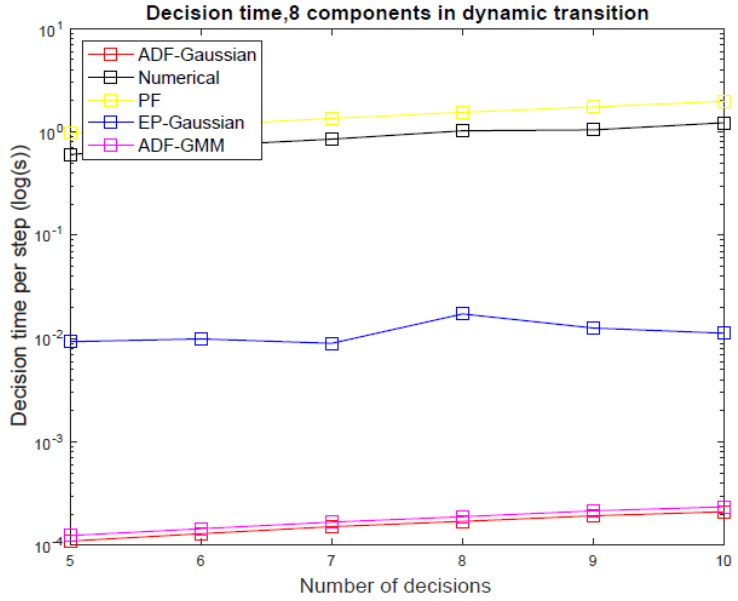
$$\max_{a_1,\ldots,a_N} I_{a_1}(X_1;Y_1) + \sum_{n=2}^{N} I_{a_n}(X_n;Y_n \mid Y_1^{n-1})$$

Optimal Information Control

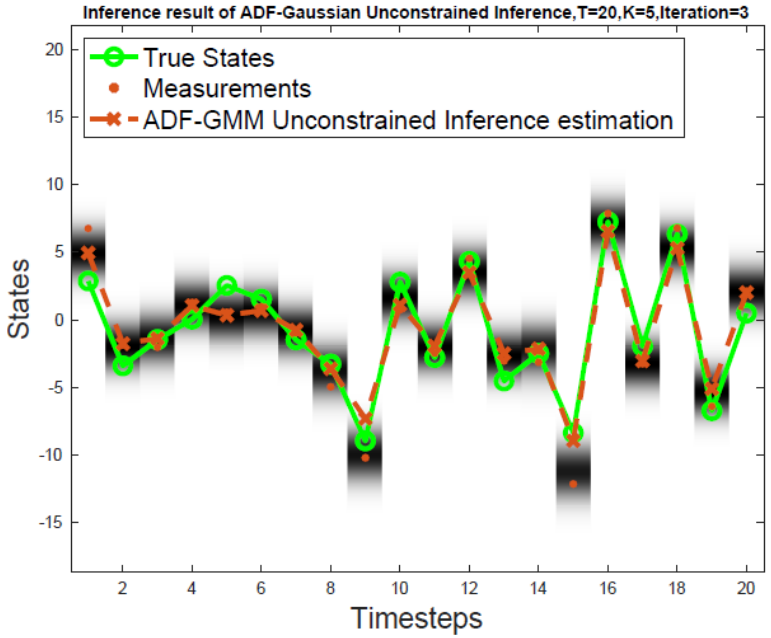
Combining efficient variational inference with fast variational MI estimates yields high-accuracy control…



(a) Information gained



(b) Runtime versus decisions

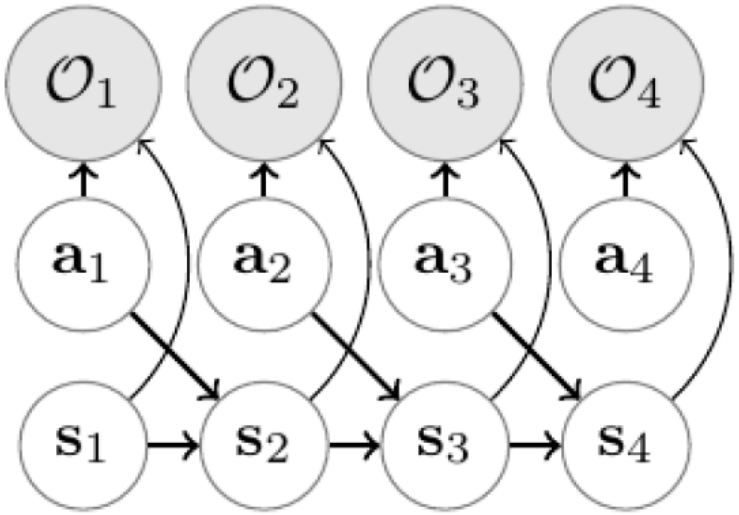


(c) Example scenario

…orders of magnitude speedup compared to Particle Filter

J. Shen and J. Pacheco, AISTATS 2023 (Under Review)

Variational Reinforcement Learning



[Image: Sergey Levine]

PGM embedding of Markov Decision Process w/

s_t : States, a_t : Actions

Binary O_t = 1 indicates optimal outcome (0 o.w.)

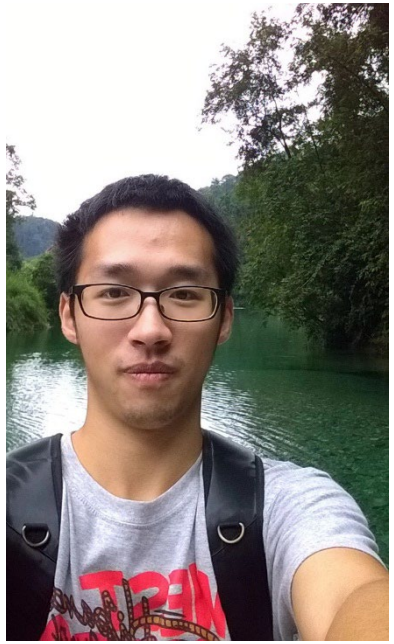
Goal: Compute the *posterior policy* that yields optimal trajectory:

$$p(a_t \mid s_t, \mathcal{O}_t = 1, \ldots, \mathcal{O}_T = 1)$$

Approach: Cannot be exactly computed. Developing variational method to approximate. Many other aspects related to risk-sensitivity.

Current work by A. Granados and J. Pacheco : supported by AFOSR YIP (PI: J. Pacheco)

Current Lab Members



Jianwei “James” Shen
CS PhD

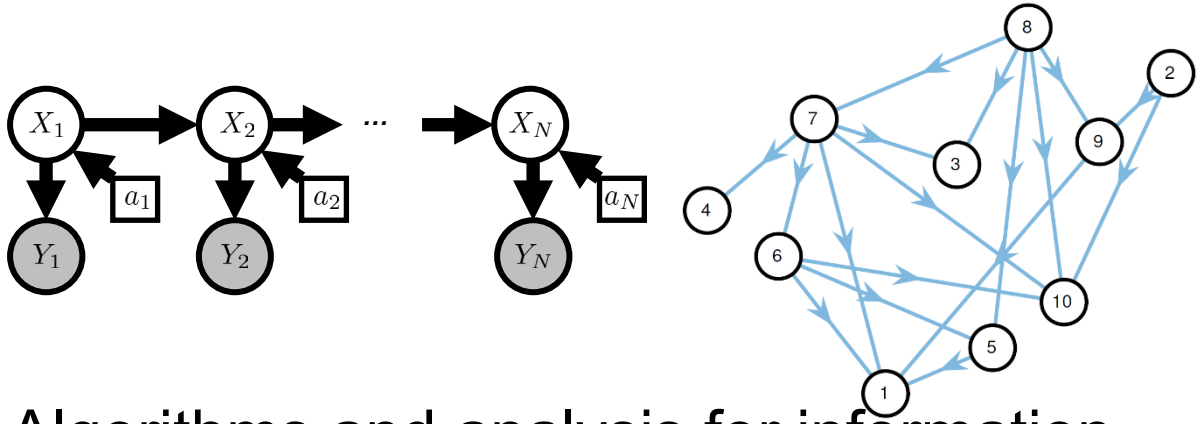
Ryan Michael Murphy
CS MS

Alonso Granados Baca
CS PhD

Caleb Dahlke
App. Math PhD

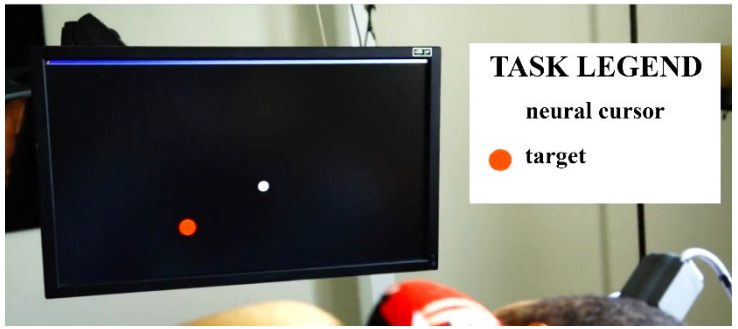
Stochastic Systems Learning Group (SSLG)

Uncertainty-Driven Decision Making

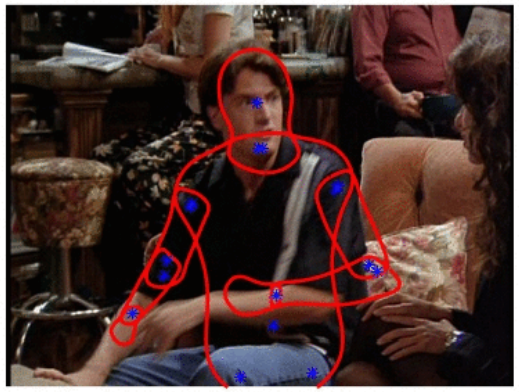


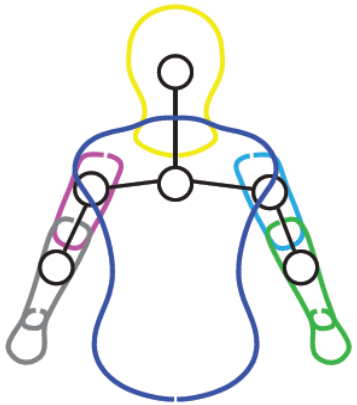
Algorithms and analysis for information-driven decision making

Spatiotemporal Reasoning



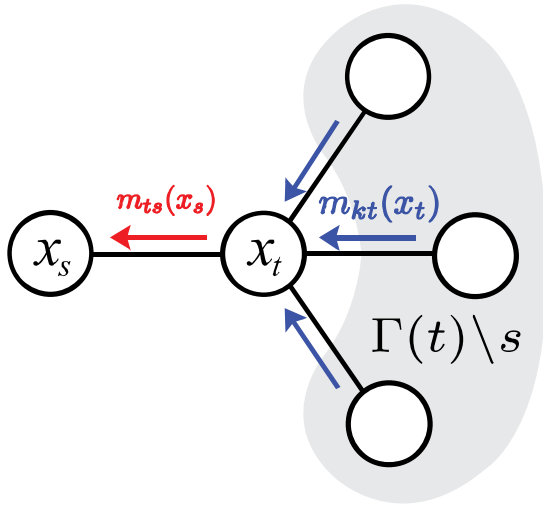
Models & inference for complex spatio-temporal reasoning tasks





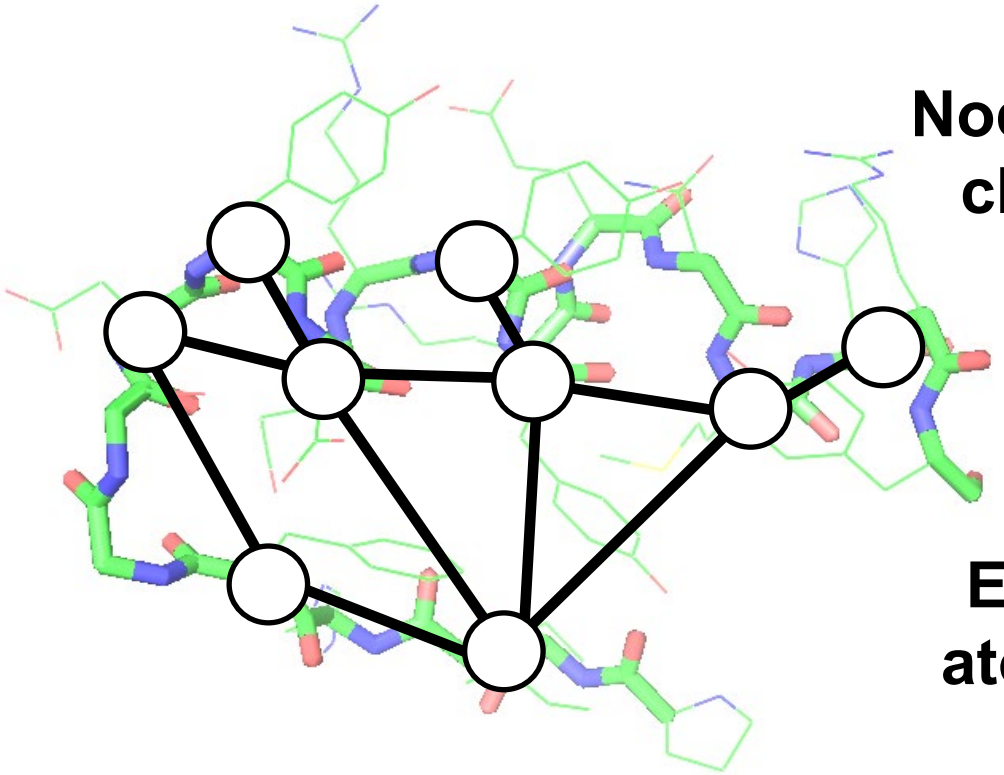
Graphical Models & Approximate Inference

General-purpose inference algorithms for high-dimensional continuous graphical models



Protein Side Chain Prediction

Graphical Model

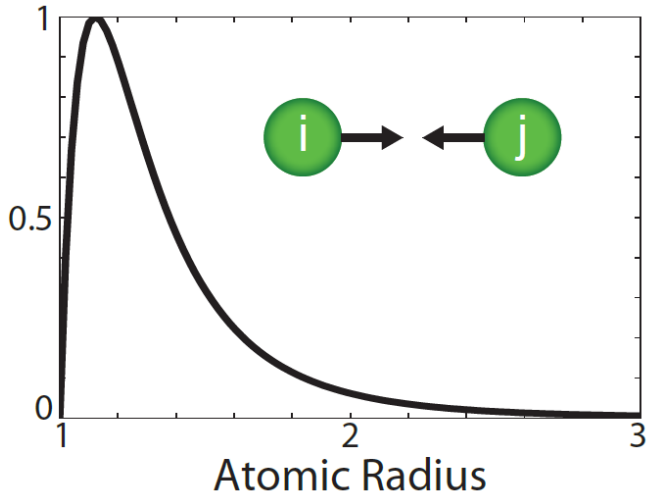


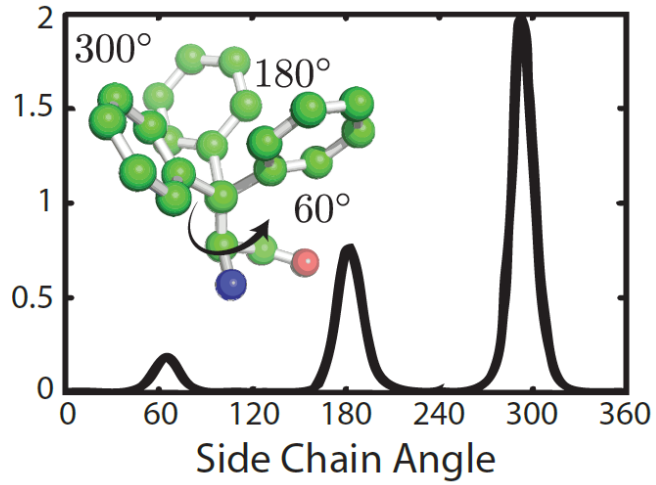
Nodes represent side chain orientations

Edges represent atomic interaction

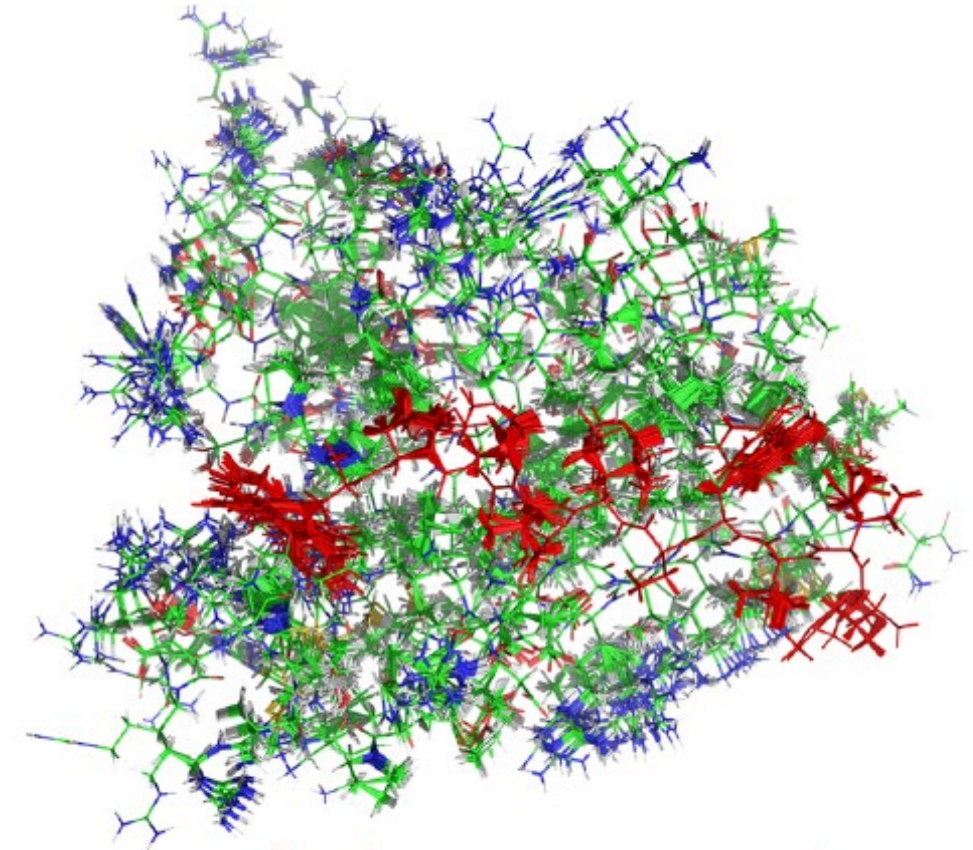
Complex phenomena specified by simpler atomic interactions

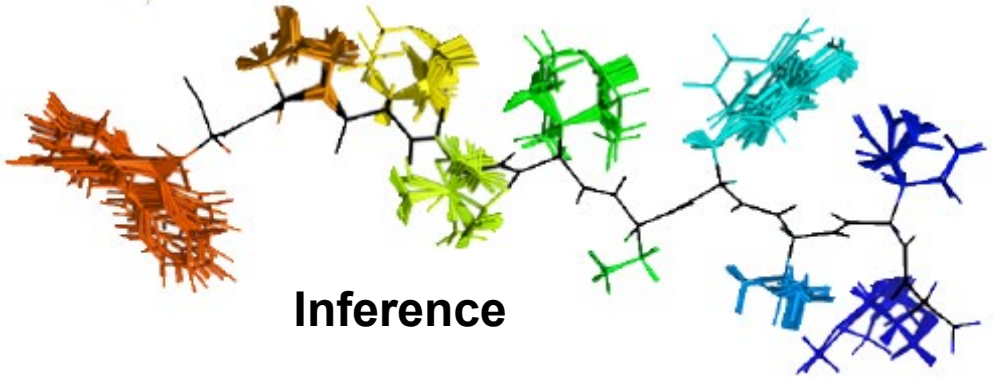
Configuration Likelihoods



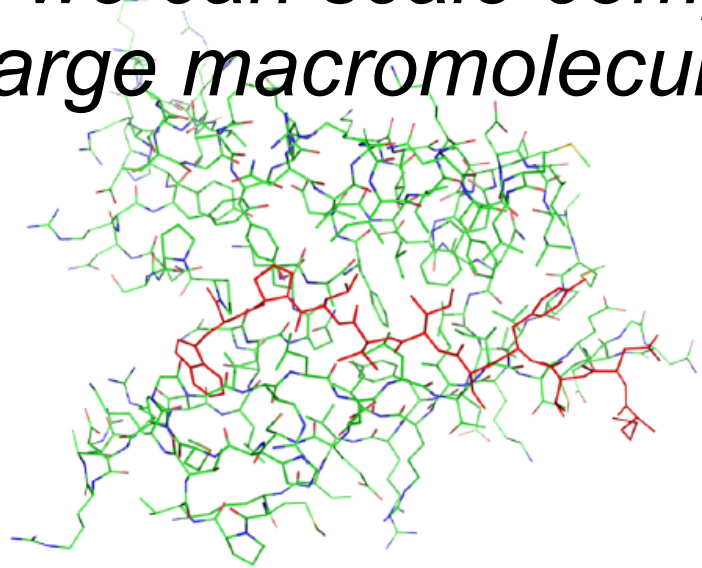


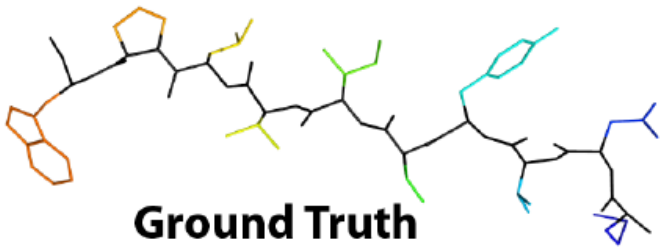
Protein Side Chain Prediction





By exploiting graphical model structure we can scale computation to large macromolecules





Pacheco and Sudderth, ICML 2015

Example: Markov Chain

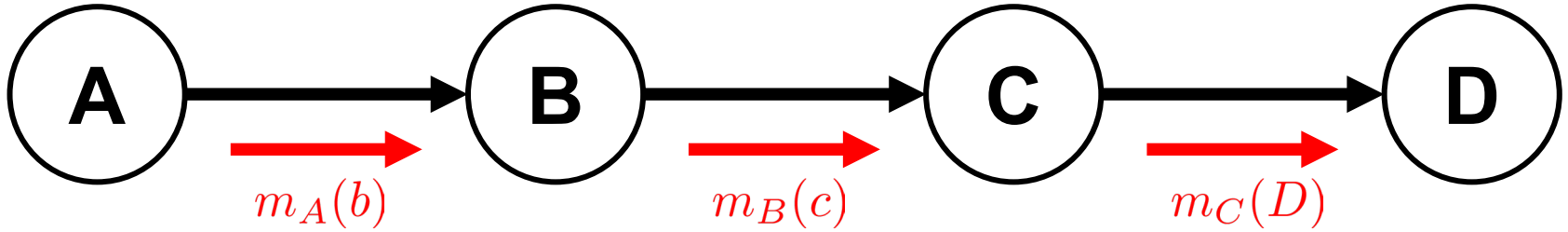
Suppose we have a chain graph…

…and want to calculate the marginal on B

$$P(D) = \sum_a \sum_b \sum_c P(a, b, c, D)$$

- For K-valued variables this is $\mathcal{O}(K^3)$
- For a Markov Chain on N variables calculating $P(X_N)$ takes $\mathcal{O}(K^{N-1})$
- We can do better by reordering operations…

Example: Markov Chain



Suppose we just care about marginal on D:

$$P(D) = \sum_a \sum_b \sum_c P(a)P(b \mid a)P(c \mid b)P(D \mid c)$$

$$= \sum_c P(D \mid c) \sum_b P(c \mid b) \underbrace{\sum_a P(a)P(b \mid a)}$$

(Distributive property)

$$= \sum_c P(D \mid c) \underbrace{\sum_b P(c \mid b) m_A(b)}$$

$$= \underbrace{\sum_c P(D \mid c) m_B(c)}$$

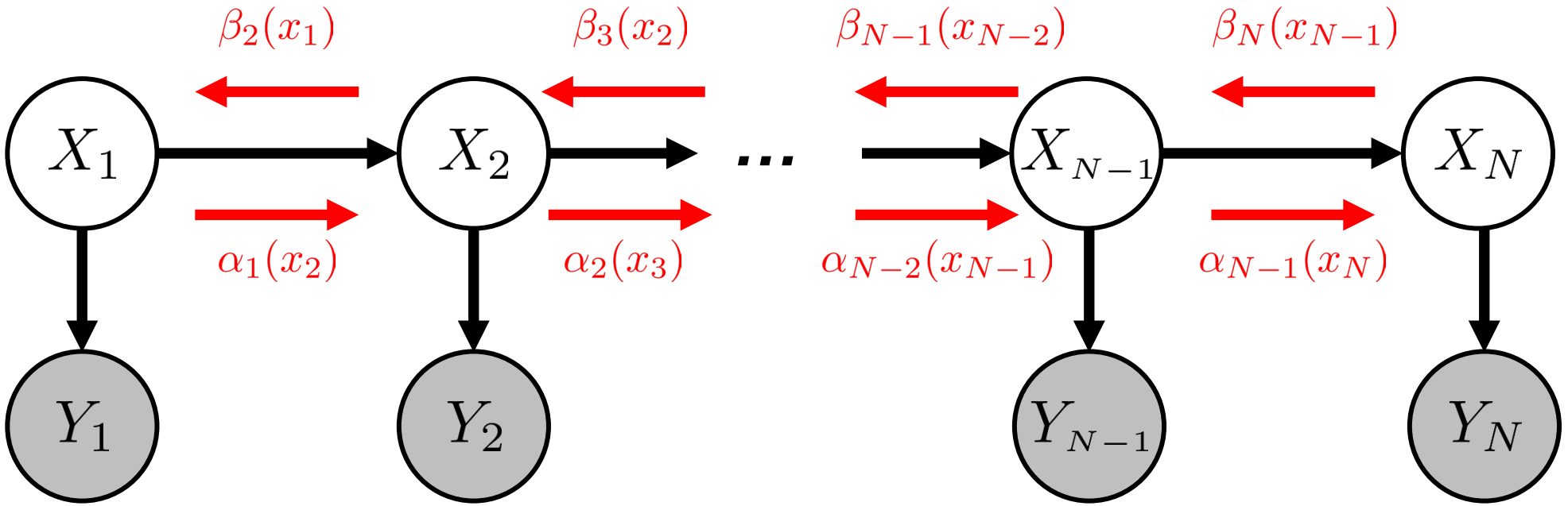
$$= m_C(D)$$

Each message takes O(K^2) time for total of O(3K^2)

On a Markov Chain of N RVs takes O((N-1)K^2)

Forward-Backward Algorithm

Extends to HMM-style graphs with node observations…



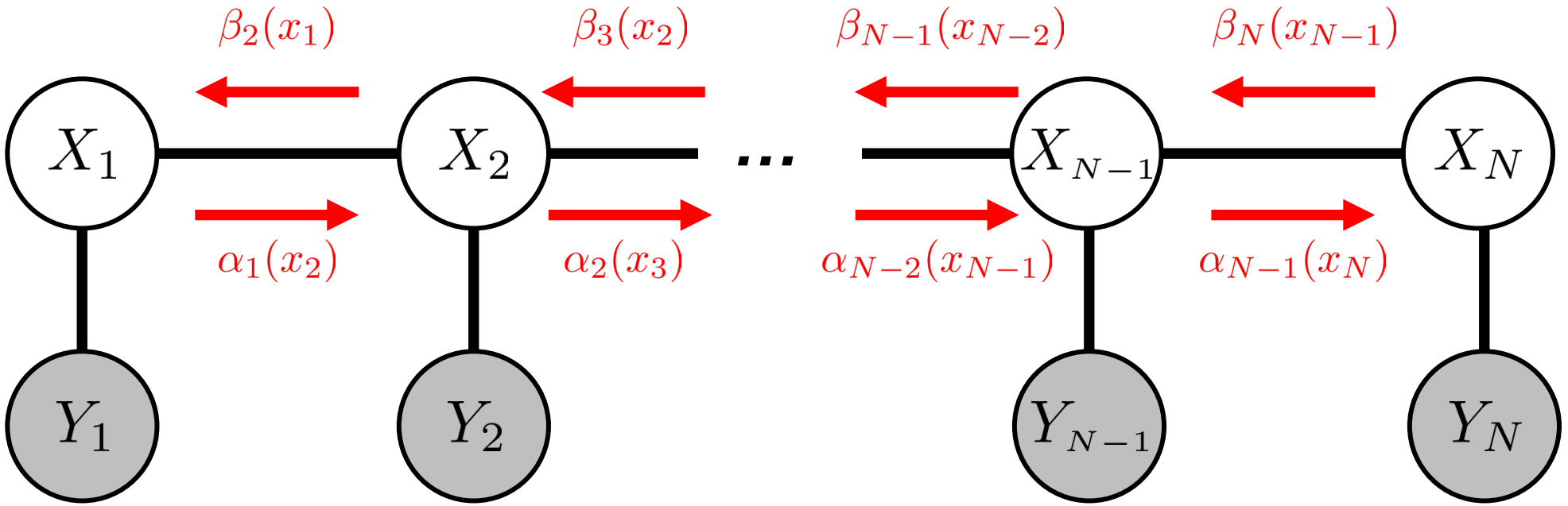
Forward message:

$$\alpha_{n-1}(x_n) = p(y_n \mid x_n) \sum_{x_{n-1}} \alpha_{n-2}(x_{n-1}) p(x_n \mid x_{n-1})$$

Backward message:

$$\beta_{n+1}(x_n) = \sum_{x_{n+1}} \beta_{n+2}(x_{n+1}) p(x_{n+1} \mid x_n) p(y_{n+1} \mid x_{n+1})$$

Forward-Backward Algorithm

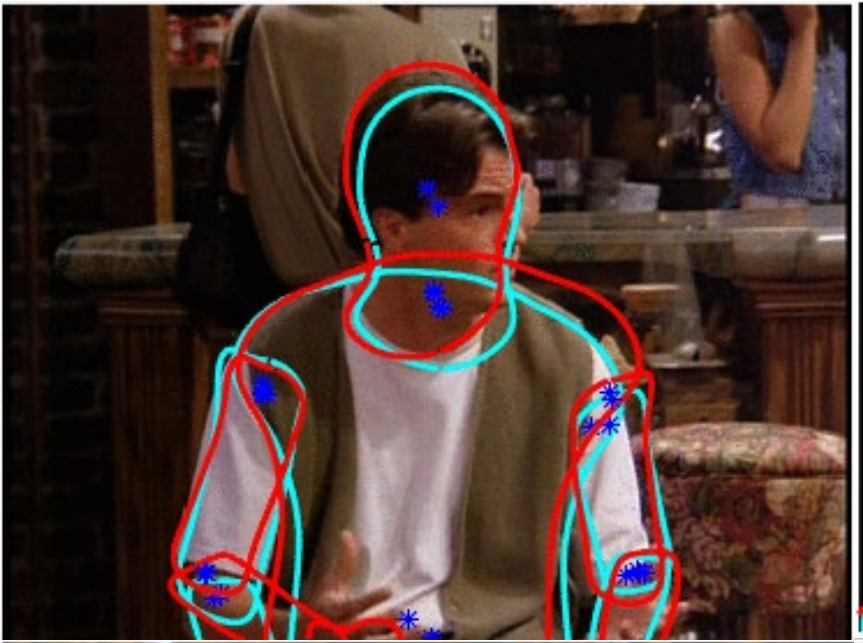
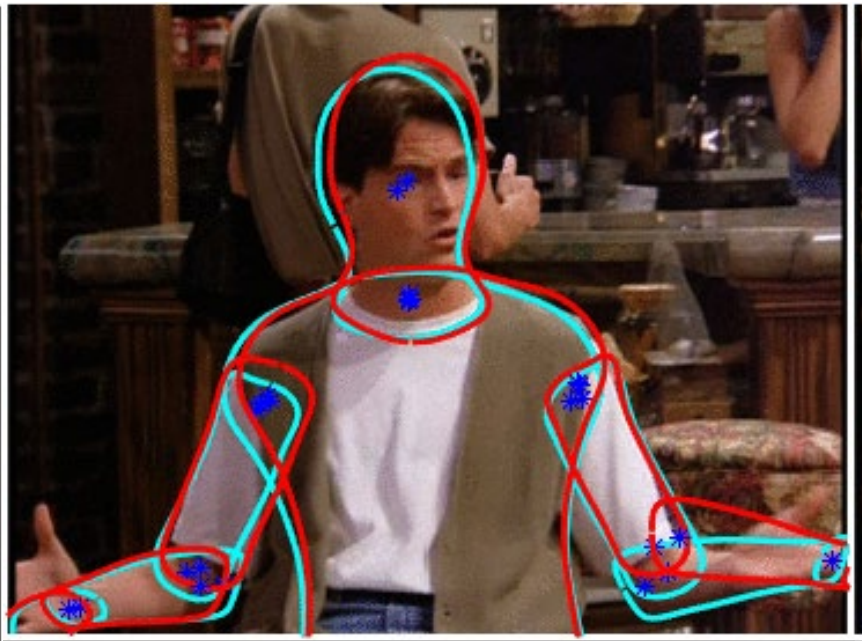
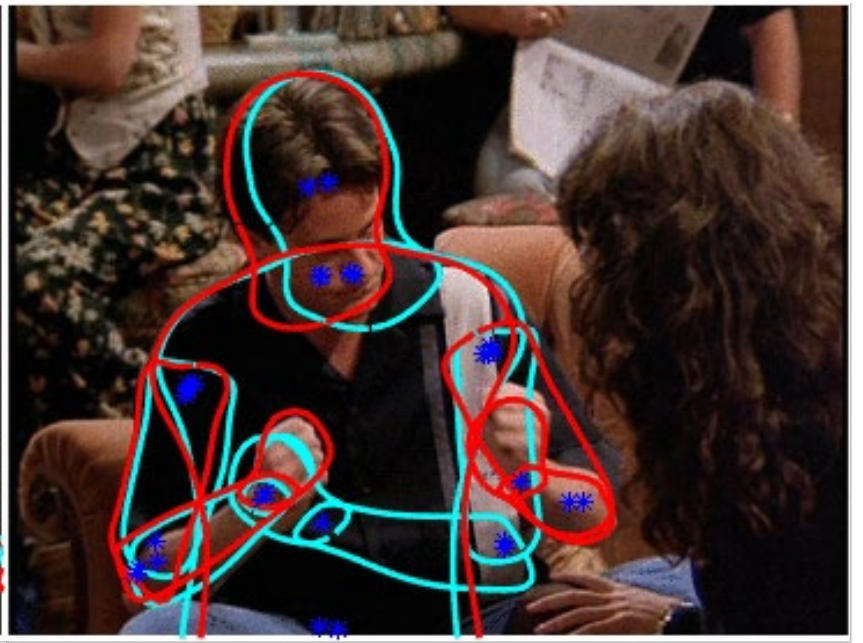
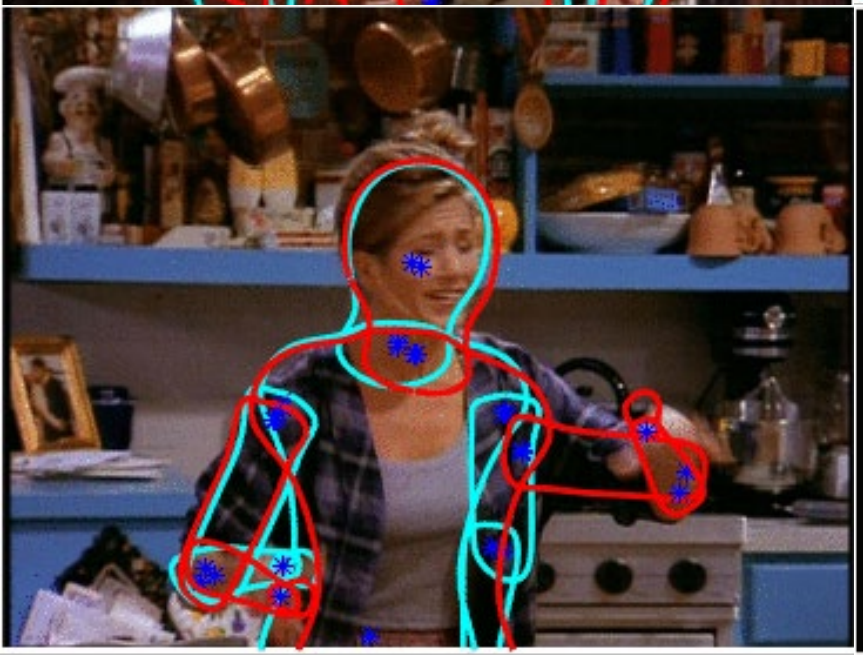
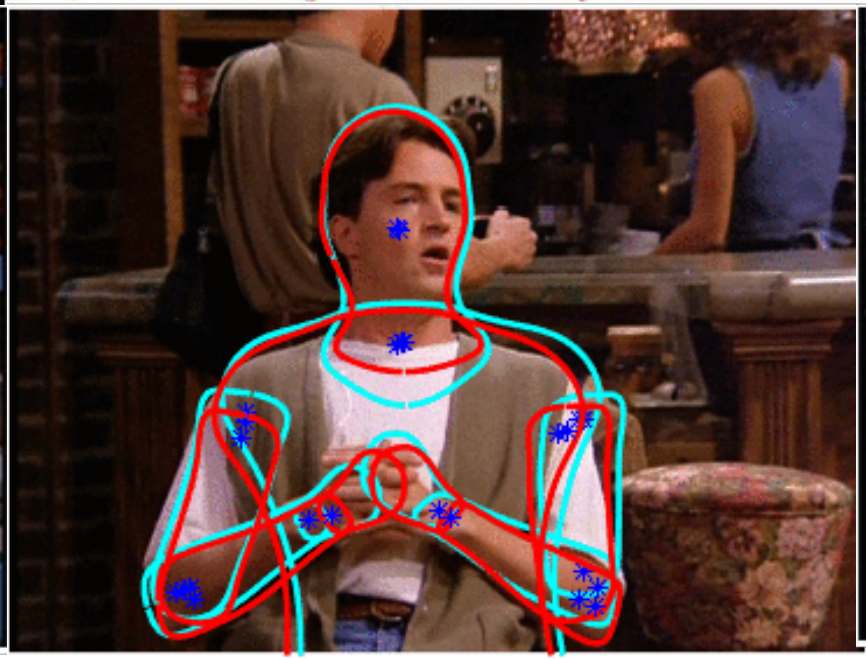
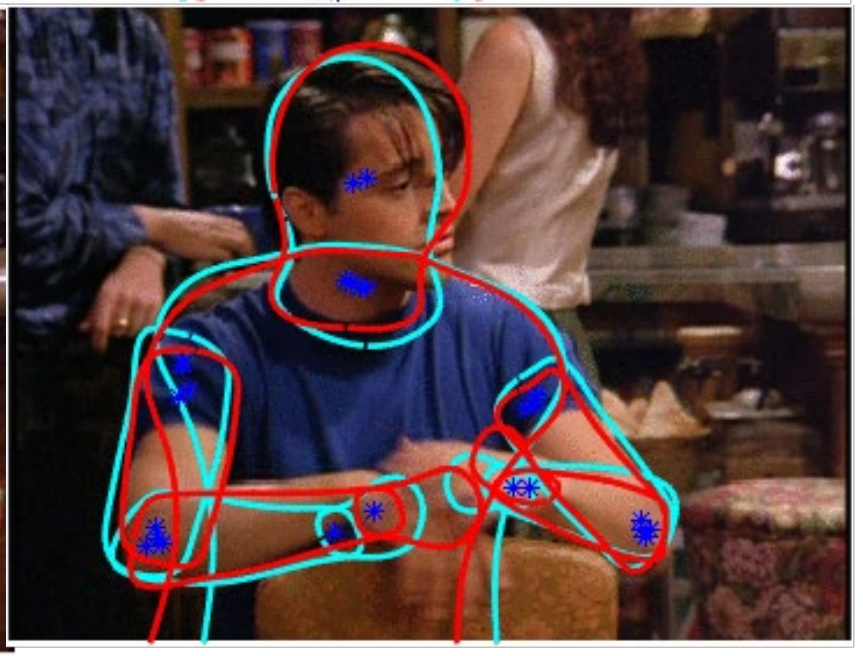


Forward message gives the *filtered posterior*:

$$\alpha_{n-1}(x_n) \propto p(y_1, \ldots, y_n, x_n) \propto p(x_n \mid y_1, \ldots, y_n)$$

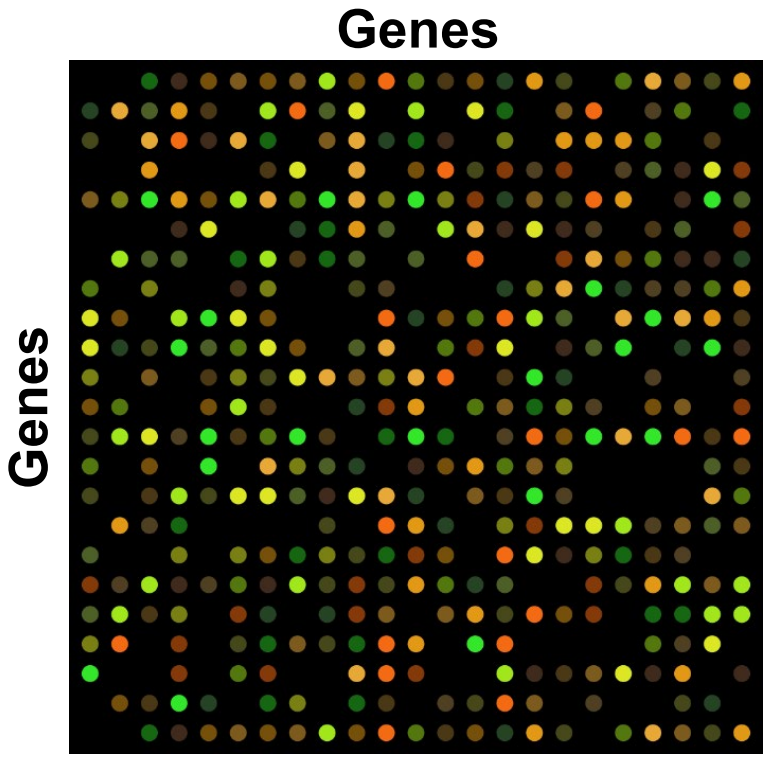
Smoothed posterior incorporates all observations:

$$p(x_n \mid y_1, \ldots, y_N) \propto p(x_n \mid y_1, \ldots, y_n) p(y_{n+1}, \ldots, y_N \mid x_n)$$
$$\propto \alpha_{n-1}(x_n) \beta_{n+1}(x_n)$$

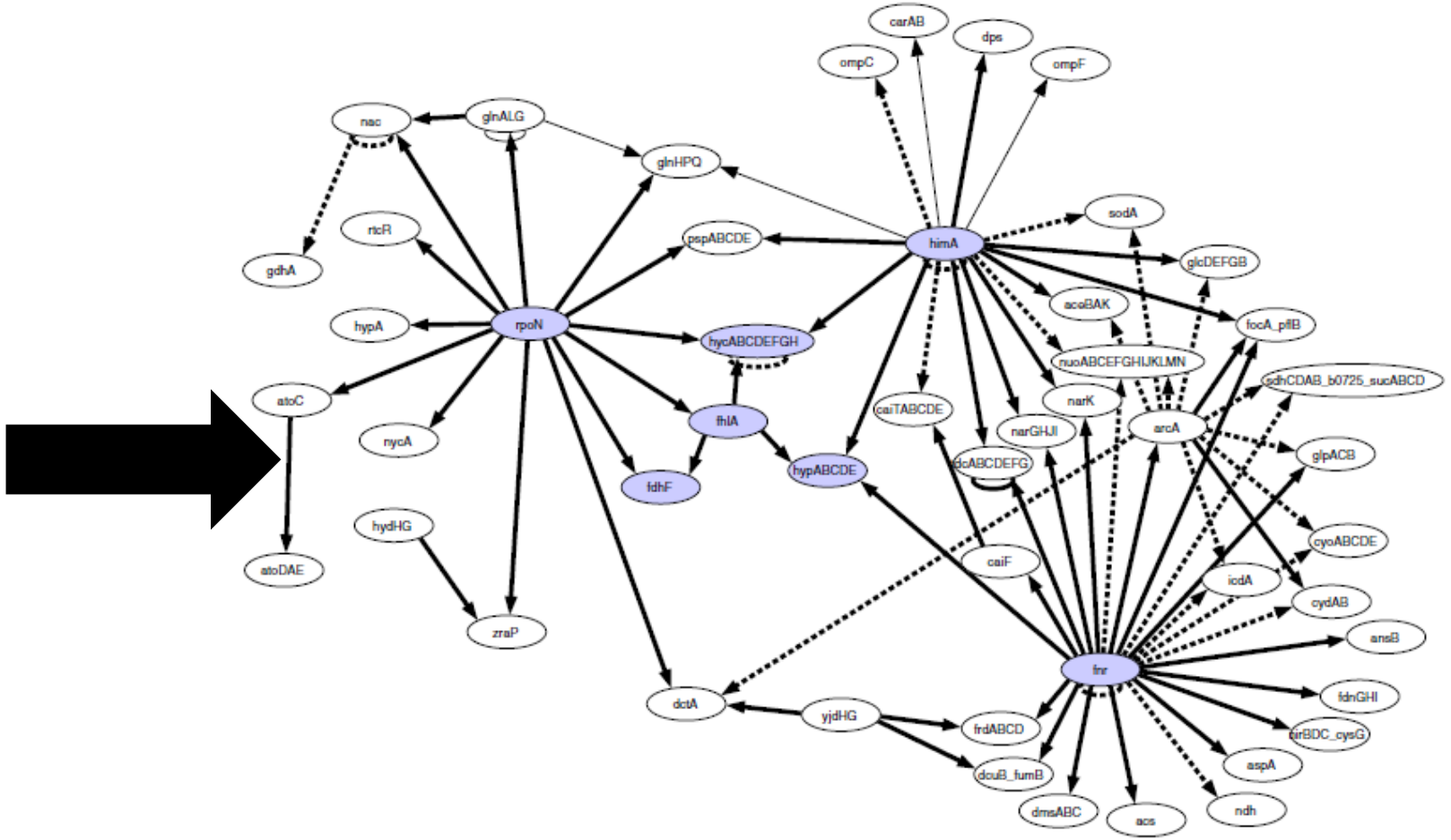







Example: Gene Regulatory Network

Gene Expression



Regulatory Network



[Image: Bulcke et al., 2006]

Goal: Estimate causal interaction network from expression data.

Graphical Models

A variety of graphical models can represent the same probability distribution