Lecture 5: Reconstruction of some non-tree networks

Elchanan Mossel

Reconstructing Networks

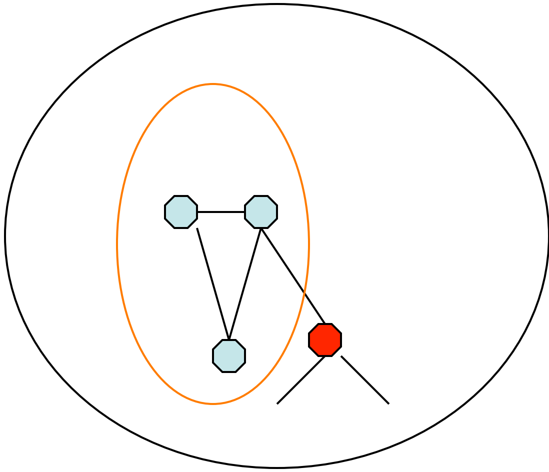
- Summary of what we did so far:
- Reconstruction of tree models from samples.
- More general problem:
- Reconstruction of network structure from samples …
- Particular interest to us: Pedigrees.
- But: Technical, do not understand so well, uses a lot of the tree technology. Instead:
- Talk a bit more about the general problem.

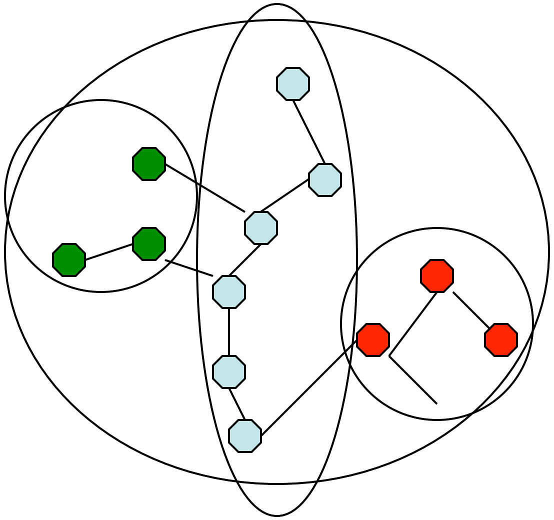
Reconstructing Networks

- Motivation: abundance of stochastic networks in biology, social networks, neuro-science etc. etc.
- Network defines a distribution as follows:
- $G=(V,E)$ = Graph on $[n] = \{1,2,\ldots,n\}$
- Distribution defined on A^V, where A is some finite set.
- Too each clique C in G, associate a function

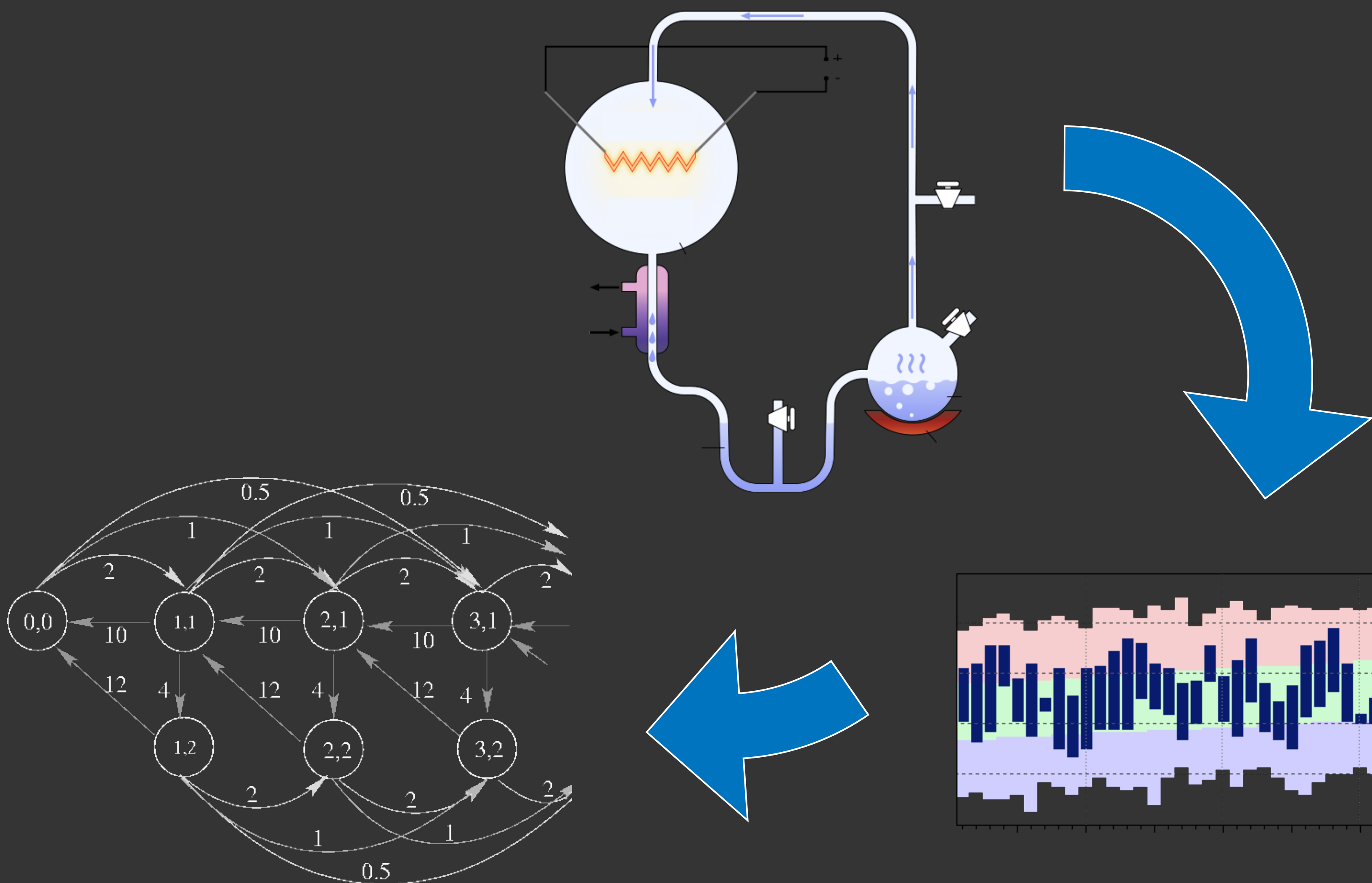
  $\psi_C : A^C \to R_+$ and:

  $P[\sigma] = \prod_C \psi_C(\sigma_C)$
- Called Markov Random Field, Factorized Distribution etc.
- Directed models also common.
- Markov Property: If S separates A from B then σ_A and σ_B are conditionally independent given σ_S



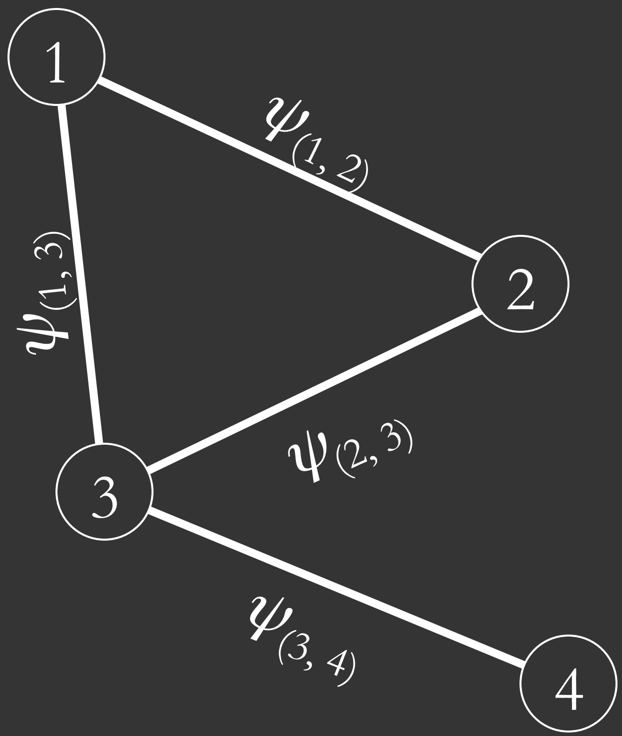


Graphical Model reconstruction



Markov random fields / Graphical Models

- **A common model for stochastic networks**

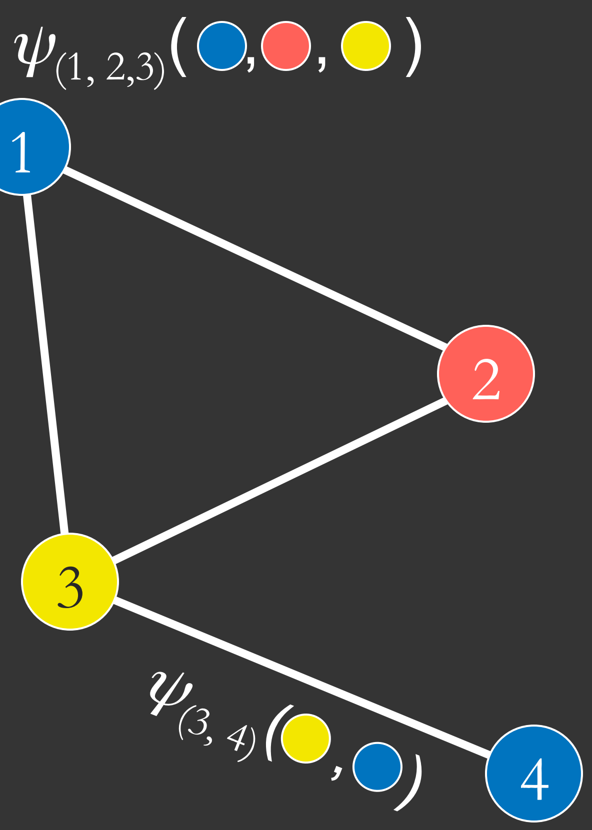


bounded degree graph $\mathrm{G} = (\mathrm{V}, \mathrm{E})$

weight functions $\psi_C: \Sigma^{|C|} \rightarrow \mathbf{R}^{\geq 0}$
for every clique C

Markov Random Fields / Graphical Models

- **A common model for stochastic networks**



bounded degree graph $G = (V, E)$

weight functions $\psi_C \colon \Sigma^{|C|} \rightarrow \mathbf{R}^{\geq 0}$
for every edge clique C

nodes v are assigned
values a_v in alphabet Σ

distribution over states
$\sigma \in \Sigma^V$ given by

$$\Pr[\sigma] \sim \prod_C \psi_C(a_w, u \in C)$$

where the product goes over all
Cliques C

Reconstruction task for Markov random fields

- Suppose we can obtain independent samples from the Markov random field
- Given observed data at the nodes, is it possible to reconstruct the model (network)?
- Important: Do we see data at all of the nodes or only a subset?

Reconstruction problem no hidden nodes

Problem: Given k independent samples of $\sigma = (\sigma_1,\ldots\sigma_n)$ at all the nodes, find the graph G

(Given activity at all the nodes, can network be reconstructed?)

- Restrict attention to graphs with max degree d: $\mathcal{G}_{n,d}$
- A structure estimator is a map $\hat{G} : \mathcal{A}^{nk} \to \mathcal{G}_{n,d}$

Questions:

1. How many samples k are required (asymptotically) in order to reconstruct MRFs with number of nodes n, max degree d with probability approaching 1, I.e. $\mathbf{P}(\hat{G}(\sigma_1, \ldots, \sigma_k) = G) = 1 - o(1)$
2. Want an *efficient* algorithm for reconstruction.

Related work

- Tree Markov Fields can be reconstructed efficiently (even with hidden nodes).
- [Erdös,Steel,Szekely,Warnow,99], [Mossel 04; Daskalakis,Mossel,Roch,06].

- PAC Setup: [Abbeel,Koller,Ng, ‘06] produce a factorized distribution that is εn close in Kullback-Leibler divergence to the true distribution.
- No guarantee to reconstruct the correct graph
- Running time and sampling complexity is $n^{O(d)}$

- More restricted problem studied by [Wainwright,Ravikumar,Lafferty, ‘06]
- Restricted to Ising model, sample complexity $\Theta(d^5 \log n)$, difficult to verify convergence condition – technique based on L_1 regularization. Moreover works for graphs not for graphical models! (clique potentials not allowed).

- Subsequent to our results, [Santhanam,Wainwright, ‘08] determine information theoretic sampling complexity and [Wainwright,Ravikumar,Lafferty, ‘08] get $\Theta(d \log n)$ sampling (restricted to Ising models; still no checkable guarantee for convergence).

Related work

Method	Abeel et al	Wainwright et al	Bresler et al.
Generative model	MRF General	Collection of Edges Ising	MRF General
Reconstruct	Dist of small KL Distance	Graph	Graph
Additional conditions	No	Yes (very hard to check)	No
Running time	n^d	n^5	n^d
Sampling Complexity	poly(n)	$d^5 \log n$ Later: $d \log n$	$d \log n$

Reconstructing General Networks - New Results

Observation: (Bresler-M-Sly-08; Lower bound on sample complexity):

- In order to recover G of max-deg d need at least $c\ d \log n$ samples, for some constant c.
- Pf follows by "counting # of networks"; information theory lower bounds.
- More formally: Given any prior distribution which is uniform over degree d graphs (no restrictions on the potentials), in order to recover correct graph with probability $\geq 2^{-n}$ need at least $c\ d \log n$ samples.

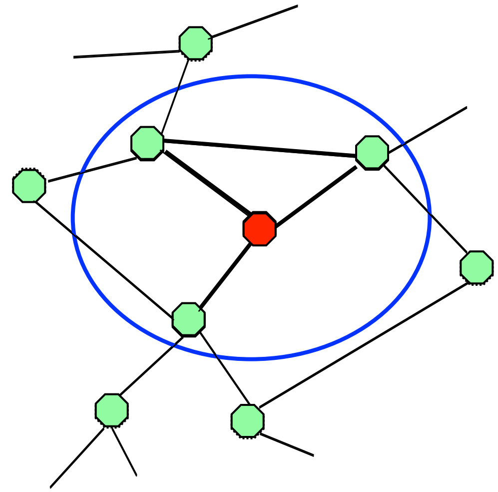
Theorem (Bresler-M-Sly-08; Asymptotically optimal algorithm):

- If distribution is "non-degenerate" $c\ d \log n$ samples suffice to reconstruct

  the model with probability $\geq 1 - 1/n^{100}$, for some (other) constant c.
- Running time is $n^{O(d)}$
- (sampling complexity tight up to a constant factor; running time – unknown)

Intuition Behind Algorithms

- Observation: Knowing graph is same as knowing neighborhoods
- But neighborhood is determined by Markov property
- Same intuition behind work of Abeel et. al.



"Algorithm":

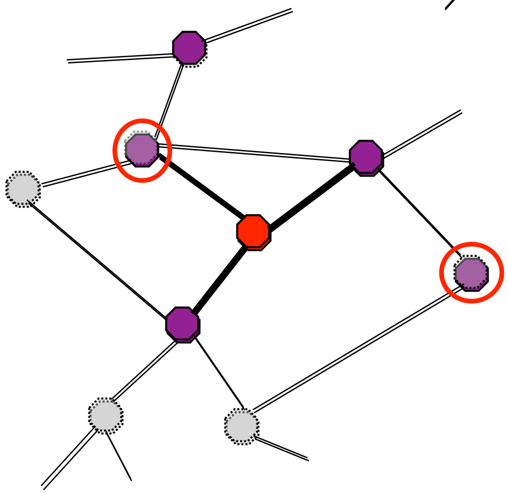
Step 1.

Compute empirical probabilities for small sets of vertices. These are concentrated.

Step 2. For each node, simply test

Markov property of each

candidate neighborhood

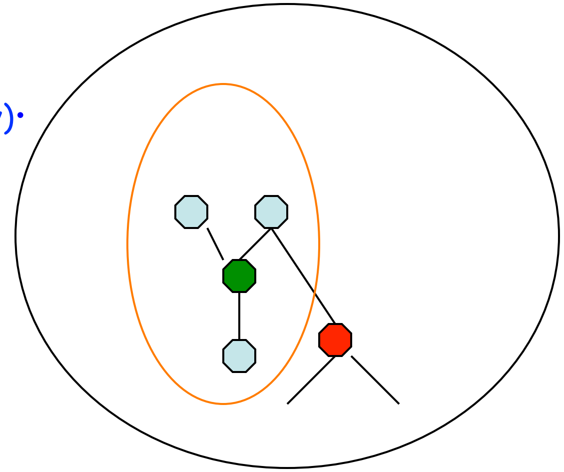


Main Challenge: Show non-degeneracy $\Rightarrow$ algorithm works

Reconstructing Networks - A Trivial Algorithm

- Upper bound (Bresler-M-Sly):
- If distribution is "non-degenerate" $c\, d \log n$ samples suffice.

- Algorithm 1:
- For each $v \in V$:
- Enumerate on $N(v)$
- For each $w \in V \setminus (N(v))$ check if σ_v ind. of σ_w given $\sigma_{N(v)}$.



- Algorithm 2:
- For each $v \in V$:
- Enumerate on $U = N(v)$
- Check that for all $u \in U$ and all W of size at most d:
- $\forall$ conditioning on σ_W,
- $\exists$ a conditioning on σ_{U-u} s.t.
- changing σ_u changes the conditional distribution at v.

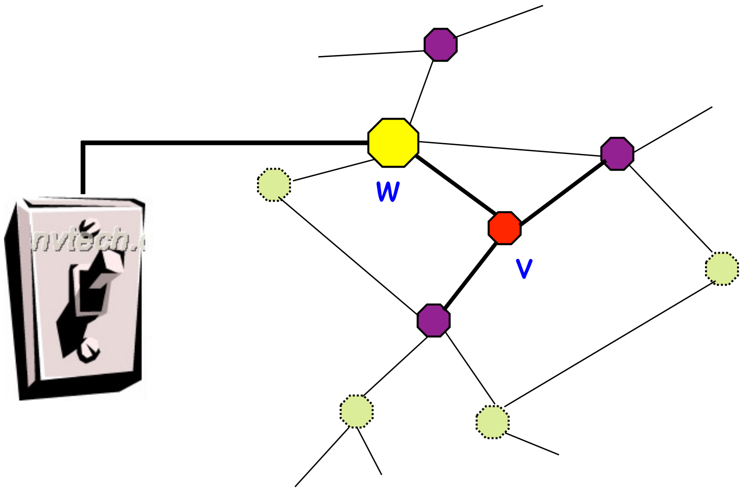
Algorithm 1

Condition N1:
For each vertex v:

For each incorrect neighborhood U, $N(v) \not\subseteq U$:

A neighbor $w \in N(v)$
has an effect on v (while conditioning on U).

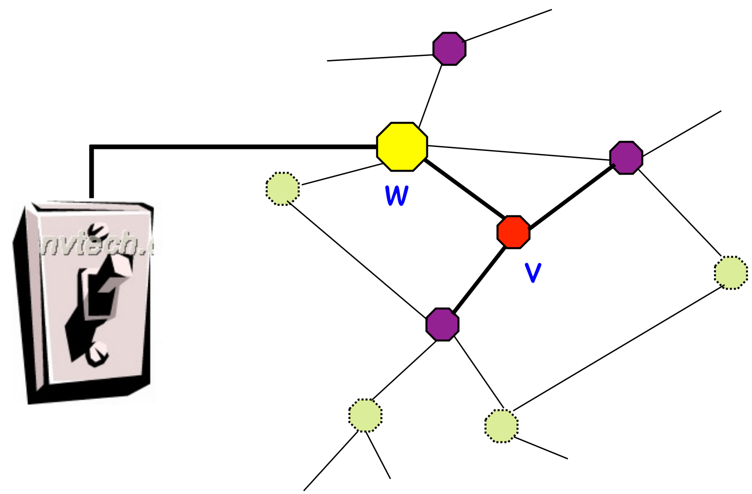
- In other words, there is a witness for the fact that $N(v) \not\subseteq U$



Algorithm:
Check each possible neighborhood U, exists witness? If not then $N(v) \subseteq U$.

Run-time:
(n nodes) x ($O(n^d)$ neighborhoods) x (n nodes) x (O(log n) samples) = $O(n^{d+2} \log n)$

Algorithm 1



<u>Condition N1 formally:</u>

There exist $\varepsilon, \delta > 0$ such that
for all $v \in V$, if $U \subset V \setminus \{v\}$
With $|U| \leq d$ and $N(v) \not\subset U$
there exist values
$x_v, x_w, x_w', x_{u_1}, \ldots, x_{u_l}$ such that
for some $w \in V \setminus (U \cup \{v\})$:

$$|P(X(v)=x_v | X(U)=x_U, X(w)=x_w) - P(X(v)=x_v | X(U)=x_U, X(w)=x_w')| > \varepsilon$$

and

$$P(X(U)=x_U, X(w)=x_w) > \delta,$$

$$|P(X(U)=x_U, X(w)=x_w') > \delta.$$

Runtime: $O(n^{d+2} \log n \, \varepsilon^{-2} \delta^{-4})$

Sampling Complexity: $O(d \log n \, \varepsilon^{-2} \delta^{-4})$

Algorithm 2

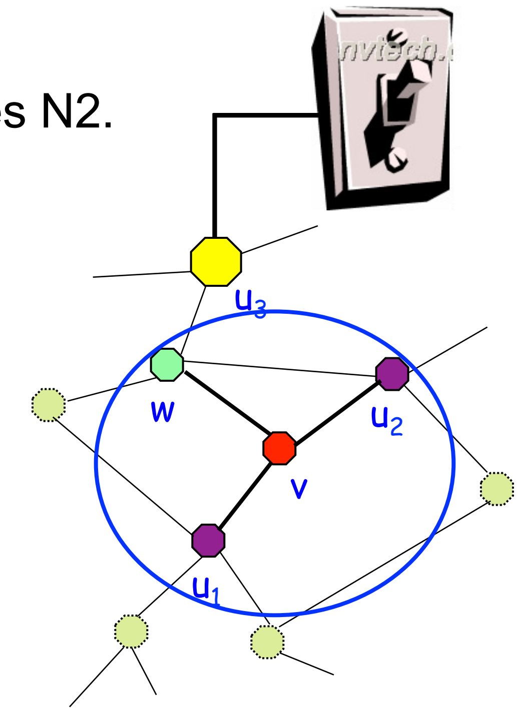
Condition N2:
For each vertex v:

Each neighbor $w \in N(v)$ has an effect on v *for some conditioning on remaining vertices in $N(v)$* .

- Weaker condition than N1: any nondegenerate MRF satisfies N2.

Witness: If U is not a subset of $N(v)$, then exists $u_i \in U$ with no effect on v while conditioning on remaining vertices in $N(v)$

- Algorithm 2:
- For each $v \in V$:
- Enumerate on $U = N(V)$
- Check that for all $u \in U$ and all W of size at most d:
- $\forall$ conditioning on σ_W,
- $\exists$ a conditioning on σ_{U-u} s.t.
- changing σ_u changes the conditional distribution at v



Algorithm 2

Condition N2:
For each vertex v:

Each neighbor $w \in N(v)$ has an effect on v *for some conditioning on remaining vertices in* $N(v)$ *.*

- Weaker condition than N1: any nondegenerate MRF satisfies N2.

Witness: If U is not a subset of $N(v)$, then exists $u_i \in U$ with no effect on v while conditioning on remaining vertices in $N(v)$

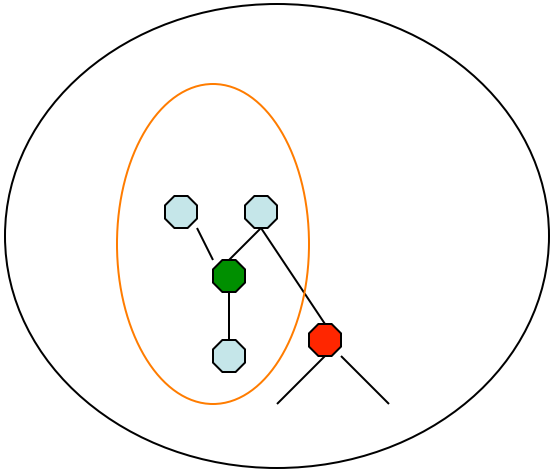
Run-time: Check (n nodes) x ($O(n^d)$ neighborhoods) x ($O(n^d)$ neighborhoods) x ($O(\log n)$ samples) = $O(n^{2d+1} \log n)$

More Exact Run-time: $O(n^{2d+2} \log n\ \varepsilon^{-2}\ \delta^{-4})$

More Exact Sampling Complexity: $O(d \log n\ \varepsilon^{-2}\ \delta^{-4})$

Reconstructing Networks - A Trivial Algorithm

- <u>Non-Degeneracy:</u>
- For algorithm 2:
- For soft-core model on graphs suffices to have for all $\psi = \psi_{u,v}$
- $\max_{a,b,c,d} |\psi(c,a)-\psi(d,a)+\psi(c,b)-\psi(d,b)| > \varepsilon$

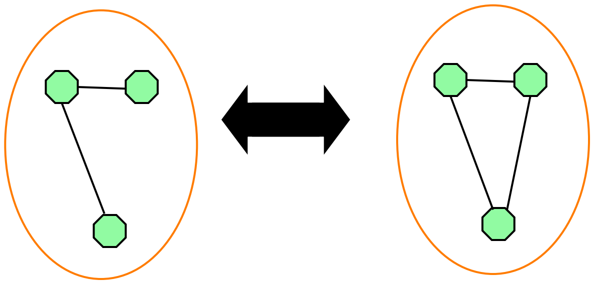


Extensions: Decay of Correlations

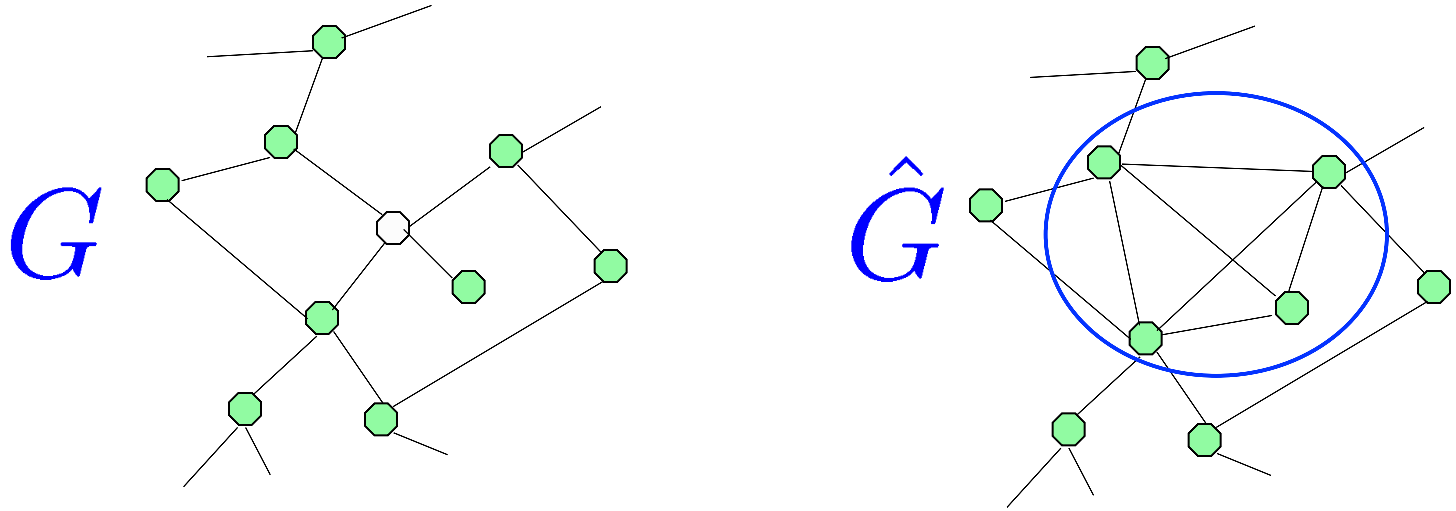
- If graph has exponential decay of correlations $Corr(\sigma_u,\sigma_v) \leq \exp(-c\ d(u,v))$
- And for each $(u,v) \in E$, $Corr(\sigma_u,\sigma_v) > \kappa$
- Then to find $N(v)$ may restrict search to nodes nearby to v.
- Running time: $O(n^2 \log n + n\ f(d))$.

Extensions: Noise & Hidden Variables

- Noise: Algorithm is robust to small amounts of noise
- Larger amount of noise often leads to non-identifiability

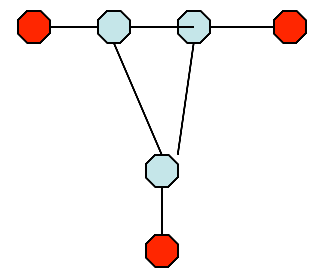


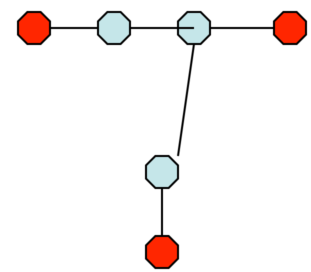
- Missing nodes: Suppose G is triangle free, then a variant of the algorithm can find hidden nodes if they are distance 2 apart.
- Idea: Run the algorithm as if the node is not hidden.



Higher Noise & Non Identifiable Example

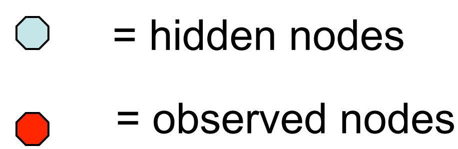
- <u>Bresler-M-Sly:</u> Example of non-identifiably
- Consider
- G_1 = path of length 2,
- G_2 = triangle + Noise.
- Assume Ising model with random interactions and random noise.
- Then with constant probability, cannot distinguish between the models.





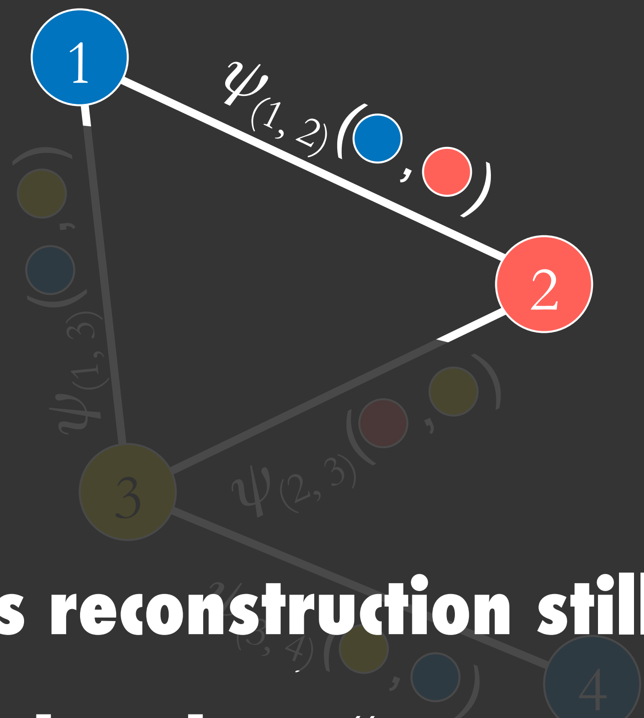
- Ising: $P[\sigma] = \prod_{u,v \in E} \exp(\beta\, \sigma(u)\, \sigma(v))$

- Intuitive reason: dimension of distribution on distributions is 3 in both cases.
- This follows from symmetry – enough to know probs of (000),(001),(010),(100)



Reconstruction of MRF with Hidden nodes

- **In many applications only some of the nodes can be observed**



visible nodes $\mathrm{W} \subseteq \mathrm{V}$

Markov random field over visible nodes is

$$\sigma_W = (\sigma_w : w \in \mathrm{W})$$

- **Is reconstruction still possible?**
- **What does "reconstruction" even mean?**

Reconstruction versus distinguishing

- Easy to construct many models that lead to the same distribution (statistical unidentifiable)
- Assuming “this is not a problem” are there computational obstacles for reconstruction?
- In particular: how hard is it to distinguish statistically different models?

Distinguishing problems

- **Let M_1, M_2 be two models with hidden nodes**

PROBLEM 1

- **Can you tell if M_1 and M_2 are statistically close or far apart (on the visible nodes)?**

PROBLEM 2

- **Assuming M_1 and M_2 are statistically far apart and given access to samples from one of them, can you tell where the samples come from?**

Hardness result with hidden nodes

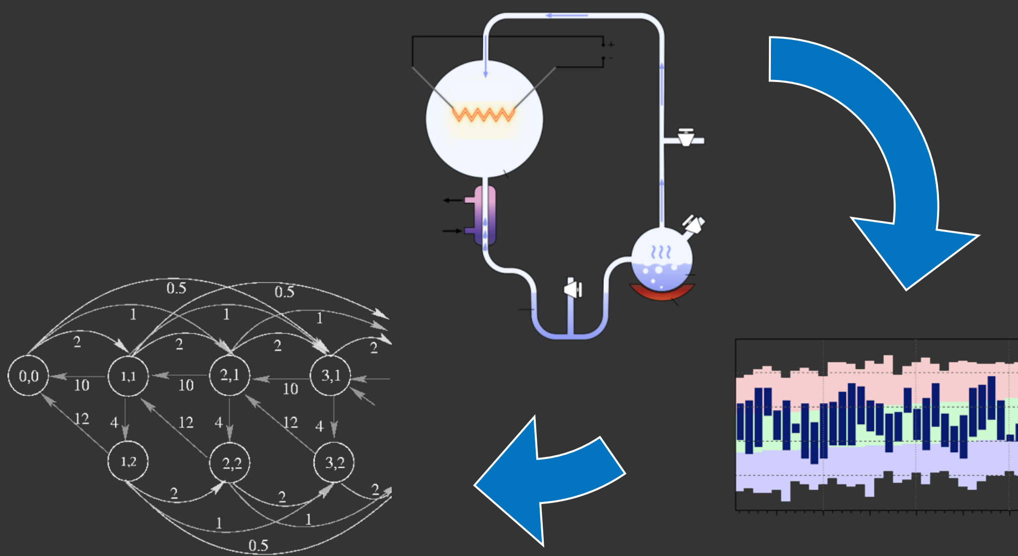
- In Bogdanov-M-Vadhan-08:

Problems 1 and 2 are intractable (in the worst case) unless $\mathrm{NP} = \mathrm{RP}$

- Conversely, if $\mathrm{NP} = \mathrm{RP}$ then distinguishing (and other forms of reconstruction) are achievable
- RP = Random Polynomial Time - with one sided error. No instance always result in no. Yes results in Yes with probability at least ½.

A possible objection

- **The "hard" models M_1, M_2 describe distributions that are not efficiently samplable**



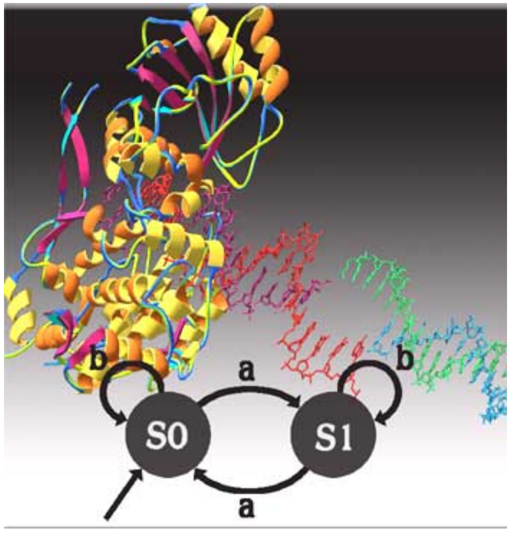
- **But if nature is efficient, we never need to worry about such distributions!**

Two Models of a Biologist

- The Computationally Limited Biologist: Cannot solve hard computational problems, in particular cannot sample from a general G-distributions.

- The Computationally Unlimited Biologist:
  Can sample from any distribution.

- Related to the following problem:
  Can nature solve computationally hard problems?



From Shapiro at Weizmann

Distinguishing problem for samplable distributions

PROBLEM 3

- If M_1 and M_2 are statistically far apart and given access to samples from one of them, can you tell where the samples came from, **assuming M_1 and M_2 are efficiently samplable**?
- Theorem

Problem 3 is intractable unless computational zero knowledge is trivial

  - We don't know if this is tight
  - Zero Knowledge: Given two circuits with total variation large

Reduction to circuits

- **Markov random fields can simulate the uniform distribution UC over satisfying assignments of a boolean circuit C**

$$\mathbf{pr}_{UC}(x) = \begin{cases} 1/\#\mathrm{SAT}(C), \text{ if } C(x) = \mathrm{TRUE} \\ 0, \text{ if } C(x) = \mathrm{FALSE} \end{cases}$$

Reduction to circuits : Proof Idea

- WLOG all gates have fun in at most 2.
- Replace each gate g by a gadget where:
- To each assignment a consistent with g add vertices $v(g,a,1),\ldots,v(g,a,r)$.
- Define MRF taking 0,1 values s.t. $\sigma(v(g,a,i)) = 1$ if a is the "asgmnt to the gate" and 0 otherwise.
- Clones allow to force consistency between different gates, at most one value.
- To force at least one value, play with weights.

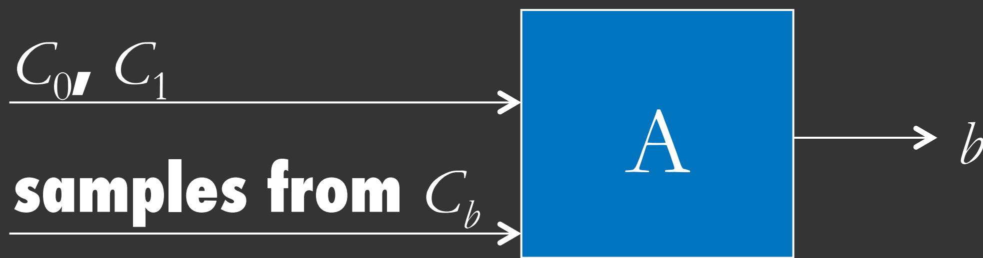
Reduction to circuits

- **Markov random fields can simulate the uniform distribution UC over satisfying assignments of a boolean circuit C**

$$\mathbf{pr}_{UC}(x) = \begin{cases} 1/\#\mathrm{SAT}(C), \text{ if } C(x) = \mathrm{TRUE} \\ 0, \text{ if } C(x) = \mathrm{FALSE} \end{cases}$$

Hardness of distinguishing circuits

- **Assume you have an algorithm A such that**



  - **If the samples come from another distribution, A can behave arbitrarily**

- **We use A to find a satisfying assignment for any circuit C: $\{0, 1\}^n \rightarrow \{0, 1\}$**

Hardness of distinguishing circuits SZK / NP-RP

$$C_0(x_1, x_2, \ldots, x_n) = C(x_1, x_2, \ldots, x_n)$$
$$C_1(x_1, x_2, \ldots, x_n) = C(\overline{x_1}, x_2, \ldots, x_n)$$

visible inputs: x_1 **hidden inputs:** $x_2, \ldots, x_n$

CLAIM

$C_0(x_1)$, $C_1(x_1)$ $\rightarrow$

samples: $0, 0, \ldots$ $\rightarrow$

A

$\rightarrow$ **value of** x_1 **in some satisfying assignment of** C

– **Proof reminiscent of argument that** $\mathrm{NP} \cap \mathrm{coNP}$ **has** NP**-hard promise problems [Even-Selman-Yacobi]**

Reconstructing Networks - the Future

- To do:
- Still a big gap between theory an practice.
- Initial simulations: Phylogenetic algorithms are fastest and most accurate on simulated data.
- Need to extend to run on “bad” data and try on real data.
- In reconstructing networks many open problems both in theory & in practice.

Collaborators

Guy Bresler
Berkeley

Andrej Bogdanov:
Hong-Kong

Allan Sly
MSR Redmond

Salil Vadhan:
Harvard


Thanks !!