Microsoft Research

Each year Microsoft Research hosts hundreds of influential speakers from around the world including leading scientists, renowned experts in technology, book authors, and leading academics, and makes videos of these lectures freely available.
2013 © Microsoft Corporation. All rights reserved.

Belief Propagation Algorithms: From Matching Problems to Network Discovery in Cancer Genomics

Jennifer Chayes
Microsoft Research New England
Microsoft Research New York City

Outline

1. Graphical Models and Belief Propagation
2. A Simple Example: Matching
3. A More Complex Example: Steiner Tree Problem
4. Application to Networks in Systems Biology

1. Graphical Models & Belief Propagation

- (Hyper) Graphical model: Representation of dependency structure of a collection of random variables with local constraints

$$G = (V, E)$$

1. Graphical Models & Belief Propagation

- (Hyper) Graphical model: Representation of dependency structure of a collection of random variables with local constraints

$$G = (V, E)$$

- Each node $i \in V$ has random variable σ_i with *a priori* distribution φ_i
- Each hyperedge $c \in E$ has (hard or soft) constraint ψ_c

1. Graphical Models & Belief Propagation

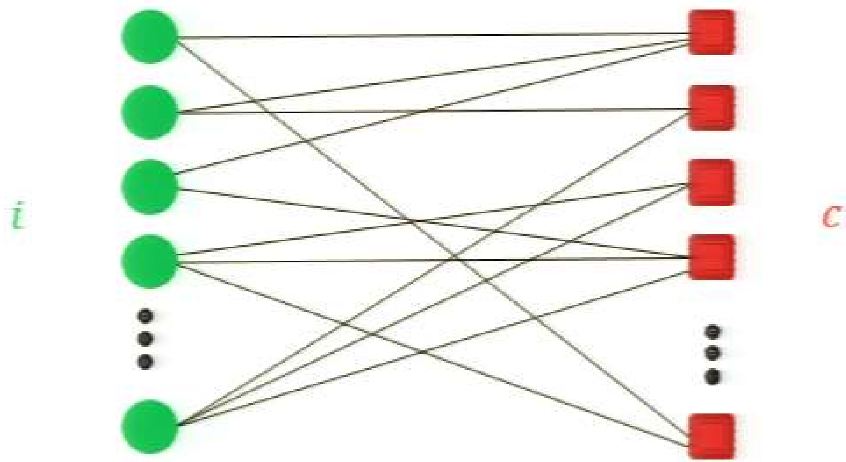
- (Hyper) Graphical model: Representation of dependency structure of a collection of random variables with local constraints

$$G = (V, E)$$

- Each node $i \in \mathrm{V}$ has random variable σ_i with *a priori* distribution φ_i
- Each hyperedge $c \in E$ has (hard or soft) constraint ψ_c
- Probability distribution of the set of variables $\sigma_V = \{\sigma_i\}_{i \in \mathrm{V}}$:

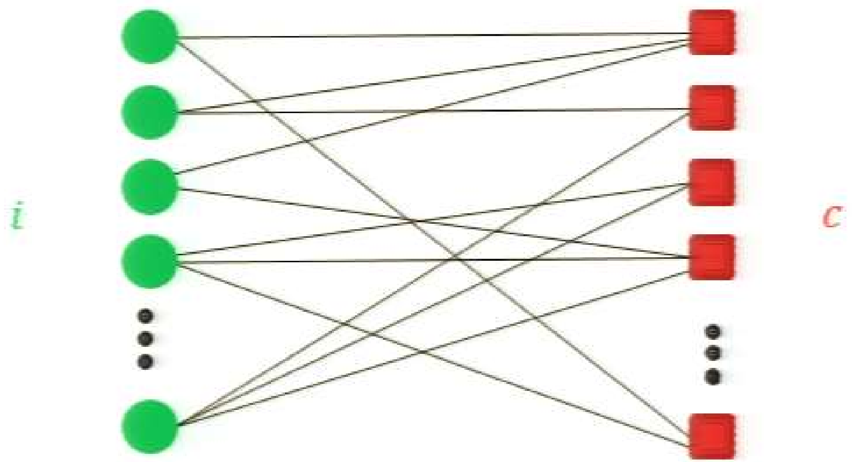
$$\mu(\sigma_V) = \frac{1}{Z} \prod_{i \in \mathrm{V}} \varphi_i(\sigma_i) \prod_{c \in \mathrm{E}} \psi_c(\sigma_c)$$

- Visualize dependency structure: Factor Graph F



ic is an edge of F
if
i is constrained by c

- Visualize dependency structure: Factor Graph F



ic is an edge of F
if
i is constrained by c

- Interested in calculating/estimating:
  - Marginals μ_i of σ_i

$$\mu_i(\sigma_i) = \sum_{\sigma_j \in \sigma_{V \setminus i}} \mu(\sigma_V)$$

  - Modes (configurations of maximal weight)

$$\sigma_{max} = \text{argmax } \mu$$

Belief Propagation

- Iterative method for approximating marginals and modes, exact if the factor graph is a tree

Belief Propagation

- Iterative method for approximating marginals and modes, exact if the factor graph is a tree
- In general, 2 sets of equations* relating:
  - "message from i to c":

    $\mu_{i \to c}$ = marginal i would have if it ignored constraint c
  - "message from c to i":

    $\mu_{c \to i}$ = marginal i would have if it were only constrained through c (and had uniform prior)

Belief Propagation

- Iterative method for approximating marginals and modes, exact if the factor graph is a tree
- In general, 2 sets of equations* relating:
  - "message from i to c":

    $\mu_{i \to c}$ = marginal i would have if it ignored constraint c
  - "message from c to i":

    $\mu_{c \to i}$ = marginal i would have if it were only constrained through c (and had uniform prior)

*Note: There are simplifications in problems in which the variables or constraints have only degree 2 in the factor graph

General Belief Propagation Equations

- Fixed-Point Equations (exact on trees):

General Belief Propagation Equations

- Fixed-Point Equations (exact on trees):

$$\mu_{i\to c}(\sigma_i) \propto \varphi_i(\sigma_i) \prod_{c' \ni i, c' \neq c} \mu_{c' \to i}(\sigma_i)$$

$$\mu_{c\to i}(\sigma_i) \propto \sum_{\sigma_k \in \sigma_{c\setminus \mathrm{i}}} \psi_c(\sigma_k) \prod_{j \in c, j \neq i} \mu_{j \to c}(\sigma_j)$$

General Belief Propagation Equations

- Fixed-Point Equations (exact on trees):

$$\mu_{i\to c}(\sigma_i) \propto \varphi_i(\sigma_i) \prod_{c' \ni i, c' \neq c} \mu_{c' \to i}(\sigma_i)$$

$$\mu_{c\to i}(\sigma_i) \propto \sum_{\sigma_k \in \sigma_{c \backslash i}} \psi_c(\sigma_k) \prod_{j \in c, j \neq i} \mu_{j \to c}(\sigma_j)$$

- Easy to implement corresponding update equations
- Often work well in practice

General Belief Propagation Equations

- Fixed-Point Equations (exact on trees):

$$\mu_{i\to c}(\sigma_i) \propto \varphi_i(\sigma_i) \prod_{c' \ni i, c' \neq c} \mu_{c' \to i}(\sigma_i)$$

$$\mu_{c\to i}(\sigma_i) \propto \sum_{\sigma_k \in \sigma_{c\setminus \mathrm{i}}} \psi_c(\sigma_k) \prod_{j \in c, j \neq i} \mu_{j \to c}(\sigma_j)$$

- Easy to implement corresponding update equations
- Often work well in practice
- Question: When does the solution converge to the right answer?

Rigorous results on BP: Convergence and correctness

Rigorous results on BP: Convergence and correctness

- Maximum weight matching
  - Bipartite graph (when solution is unique):
    - Bayati, Shah, Sharma ('08)
  - General graph, b-matching (when corresponding LP is tight):
    - Bayati, Borgs, Chayes, Zecchina ('09)
    - Sanghavi, Shah, Willsky ('09)

Rigorous results on BP: Convergence and correctness

- Maximum weight matching
  - Bipartite graph (when solution is unique):
    - Bayati, Shah, Sharma ('08)
  - General graph, b-matching (when corresponding LP is tight):
    - Bayati, Borgs, Chayes, Zecchina ('09)
    - Sanghavi, Shah, Willsky ('09)
- Nash bargaining on networks (when corresponding MWM LP is tight):
  - Bayati, Borgs, Chayes, Kanoria, Montanari ('11)

Rigorous results on BP: Convergence and correctness

- Maximum weight matching
  - Bipartite graph (when solution is unique):
    - Bayati, Shah, Sharma ('08)
  - General graph, b-matching (when corresponding LP is tight):
    - Bayati, Borgs, Chayes, Zecchina ('09)
    - Sanghavi, Shah, Willsky ('09)
- Nash bargaining on networks (when corresponding MWM LP is tight):
  - Bayati, Borgs, Chayes, Kanoria, Montanari ('11)
- Min-cost network flow:
  - Garmanik, Shah, Wei ('11)

2. A Simple Example of BP: Matching

- The model and graphical representation
- Derivation of BP for (max) weighted matching
- LP and statement of BP results

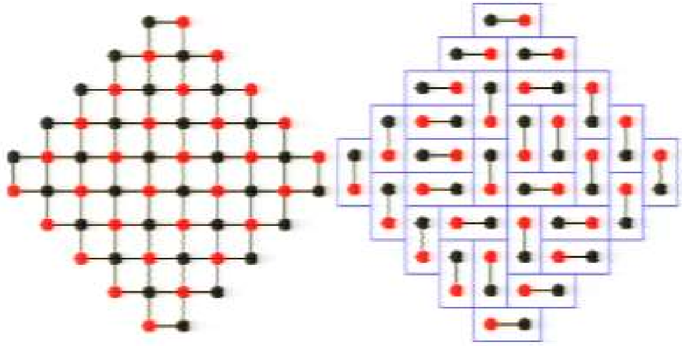
Maximum Weight Matching Problem

- Given
  - Graph $G = (V, E)$
  - Weights $\{w_{ij}\}_{ij \in E}$

Maximum Weight Matching Problem

- Given
  - Graph $G = (V, E)$
  - Weights $\{w_{ij}\}_{ij \in E}$
- Perfect matching M

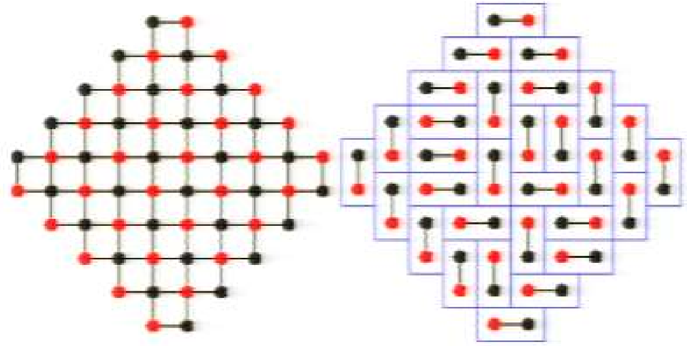
$$M \subseteq E \text{ s.t. } \forall i \in V \quad |\{e \in M \mid e \ni i\}| = 1$$



Maximum Weight Matching Problem

- Given
  - Graph $G = (V, E)$
  - Weights $\{w_{ij}\}_{ij \in E}$
- Perfect matching M

$$M \subseteq E \text{ s.t. } \forall\, i \in V \quad |\{e \in M \mid e \ni i\}| = 1$$



- Max-weight matching problem: Find

$$M_{max} \text{ s.t. } \; W(M_{max}) = \sum_{ij \in M_{max}} w_{ij} \text{ is maximal}$$

Graphical Model for Matching

Graphical Model for Matching

- Here the variables sit on the edges and the constraints on the sites of the graph $G = (V, E)$
  - Variables: $\forall\, ij \in E,\;\; x_{ij} = \begin{cases} 0 \text{ if} & \text{vacant} \\ 1 \text{ if occupied} \end{cases}$
  - Constraints: $\forall\, i \in V,\;\; \sum_{j \in N(i)} x_{ij} = 1$

M $\leftrightarrow$ edge variables $x_E = \{x_{ij}\}$ with $x_{ij} = \begin{cases} 1 \text{ if } ij \in M \\ 0 \text{ if } ij \notin M \end{cases}$

Graphical Model for Matching

- Here the variables sit on the edges and the constraints on the sites of the graph $G = (V, E)$
  - Variables: $\forall\, ij \in E,\ \ x_{ij} = \begin{cases} 0 \text{ if } & \text{vacant} \\ 1 \text{ if } & \text{occupied} \end{cases}$
  - Constraints: $\forall\, i \in V,\ \ \sum_{j \in N(i)} x_{ij} = 1$

  M $\leftrightarrow$ edge variables $x_E = \{x_{ij}\}$ with $x_{ij} = \begin{cases} 1 \text{ if } ij \in M \\ 0 \text{ if } ij \notin M \end{cases}$
- Probability distribution of x_E at "temperature" β:

$$\mu(x_E) = \frac{1}{Z} \prod_{ij \in \mathrm{E}} e^{\beta w_{ij} x_{ij}} \prod_{i \in \mathrm{V}} \mathbb{I}\Big(\sum_{j \in N(i)} x_{ij} = 1 \Big)$$

Derivation: BP Matching Equations on Trees

Derivation: BP Matching Equations on Trees

- Notational Simplification:
  - Leave out constraint in equations, and enforce constraints implicitly

$$\mu(x_E) = \frac{1}{Z}\prod_{ij\in E} e^{\beta w_{ij}x_{ij}}$$

- Messages:
  - Since variables have only degree 2 in the factor graph, we need only one set of equations, e.g. for $\mu_{\{i,j\}\to j}$ = marginal at ij if constraint at j is ignored, which we'll just call $\mu_{i\to j}$ $= \mu_{i\to j}(x_{ij})$.

Derivation: BP Matching Equations on Trees

- Notational Simplification:
  - Leave out constraint in equations, and enforce constraints implicitly

$$\mu(x_E) = \frac{1}{Z} \prod_{ij \in E} e^{\beta w_{ij} x_{ij}}$$

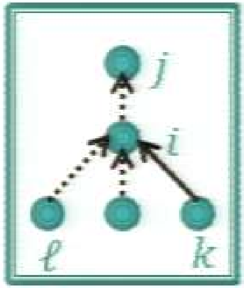
- Messages:
  - Since variables have only degree 2 in the factor graph, we need only one set of equations, e.g. for $\mu_{\{i,j\}\to j}$ = marginal at ij if constraint at j is ignored, which we'll just call $\mu_{i\to j} = \mu_{i\to j}(x_{ij})$.
  - Also, instead of taking just $\mu_{i\to j}(1)$ or $\mu_{i\to j}(0)$, as the message, try the log-ratio $m_{i\to j}$ defined by

$$e^{\beta m_{i\to j}} = \frac{\mu_{i\to j}(1)}{\mu_{i\to j}(0)}$$

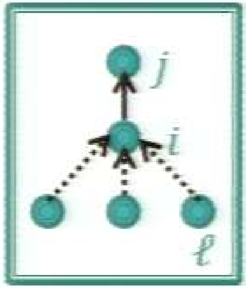
Iterative Calculations on Trees

Iterative Calculations on Trees

- $\mu_{i\to j}(0)$
$= \frac{1}{z_{ij}} \sum_{k\in N(i)\setminus j} \mu_{k\to i}(1) \prod_{\ell \in N(i)\setminus\{j,k\}} \mu_{\ell\to i}(0)$

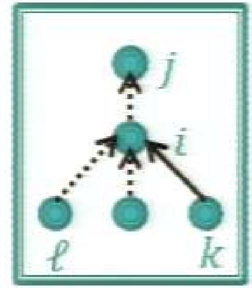


- $\mu_{i\to j}(1)$
$= \frac{e^{\beta w_{ij}}}{z_{ij}} \prod_{\ell \in N(i)\setminus j} \mu_{\ell\to i}(0)$

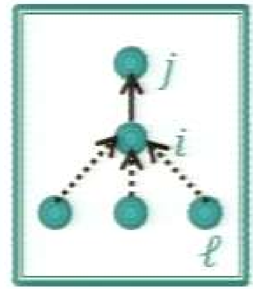


Iterative Calculations on Trees

- $\mu_{i\to j}(0)$
$= \frac{1}{Z_{ij}} \sum_{k\in N(i)\setminus j} \mu_{k\to i}(1) \prod_{\ell\in N(i)\setminus\{j,k\}} \mu_{\ell\to i}(0)$



- $\mu_{i\to j}(1)$
$= \frac{e^{\beta w_{ij}}}{Z_{ij}} \prod_{\ell\in N(i)\setminus j} \mu_{\ell\to i}(0)$



$$\Rightarrow e^{-\beta m_{i\to j}} = \frac{\mu_{i\to j}(0)}{\mu_{i\to j}(1)} = \sum_{k\in N(i)\setminus j} e^{-\beta(w_{ij} - m_{k\to i})}$$

As $\beta \to \infty$

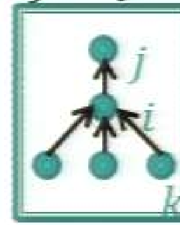
$$m_{i\to j} = w_{ij} - \max_{k\in N(i)\setminus j} m_{k\to i}$$

BP Algorithm for Matching

BP Algorithm for Matching

- Define "message" $m_{i \to j}$ on directed edge $i \to j$ by

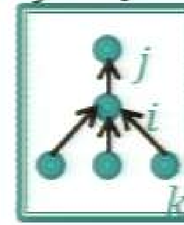
$$m_{i \to j}(0) = w_{ij}$$
$$m_{i \to j}(t+1) = w_{ij} - \max_{k \in N(i) \backslash j} m_{k \to i}(t)$$



BP Algorithm for Matching

- Define "message" $m_{i\to j}$ on directed edge $i \to j$ by

$$m_{i\to j}(0) = w_{ij}$$
$$m_{i\to j}(t+1) = w_{ij} - \max_{k\in N(i)\setminus j} m_{k\to i}(t)$$



- Similarly can show: Define M_{max} at time t, $M(t)$:

For each site i choose as the candidate edge into i the edge $i\ell$ such that

$$m_{\ell\to i}(t) = \max_{k\in N(i)} m_{k\to i}(t)$$

and add this maximum message edge to the candidate "matching" $M(t)$. (Note $M(t)$ may not be a matching.)

BP Algorithm for Matching

- Define "message" $m_{i \to j}$ on directed edge $i \to j$ by

$$m_{i \to j}(0) = w_{ij}$$
$$m_{i \to j}(t+1) = w_{ij} - \max_{k \in N(i) \setminus j} m_{k \to i}(t)$$

- Similarly can show: Define M_{max} at time t, $M(t)$:

For each site i choose as the candidate edge into i the edge $i\ell$ such that

$$m_{\ell \to i}(t) = \max_{k \in N(i)} m_{k \to i}(t)$$

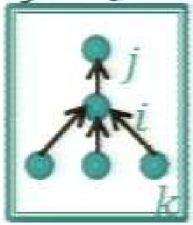
and add this maximum message edge to the candidate "matching" $M(t)$. (Note $M(t)$ may not be a matching.)

- Note: This is exact on trees.

BP Algorithm for Matching

- Define "message" $m_{i\to j}$ on directed edge $i \to j$ by

$$m_{i\to j}(0) = w_{ij}$$
$$m_{i\to j}(t+1) = w_{ij} - \max_{k\in N(i)\setminus j} m_{k\to i}(t)$$



- Similarly can show: Define M_{max} at time t, $M(t)$:

For each site i choose as the candidate edge into i the edge $i\ell$ such that

$$m_{\ell\to i}(t) = \max_{k\in N(i)} m_{k\to i}(t)$$

and add this maximum message edge to the candidate "matching" $M(t)$. (Note $M(t)$ may not be a matching.)

- Note: This is exact on trees.
- Question: Can we determine when else it converges to the correct answer, and how fast?

Rigorous Result on BP for Matching

Rigorous Result on BP for Matching

- Consider the corresponding LP relaxation and its dual:
  - LP: $\max \sum_{ij\in E} w_{ij} x_{ij}$
    subj. to $0 \le x_{ij} \le 1$
    $\sum_{j\in N(i)} x_{ij} = 1$
  - Dual: $\min \sum_{ij\in E} \lambda_{ij} - \sum_{i\in V} y_i$
    subj. to $\lambda_{ij} \ge 0$
    $\lambda_{ij} \ge w_{ij} + y_i + y_j$
- Theorem (Bayati, Borgs, Chayes, Zecchina '09): If the LP has a unique optimum which is integer, then $M(t)$ converges to the correct solution M_{max}. In particular $M(t) = M_{max}$ for

$$t \ge \frac{2|V|}{\epsilon} \max_i |y_i^*| \, ,$$

where y^* is an optimal solution of the dual LP and

$$\epsilon = \min_{ij}\{| w_{ij} + y_i^* + y_j^* | > 0\}.$$

3. The Steiner Tree Problem

3. The Steiner Tree Problem

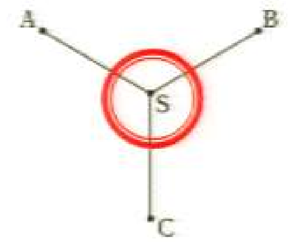
- Given
  - Graph $G = (V, E)$
  - Costs $\{c_{ij}\}_{ij \in E}$, $c_{ij} \geq 0$
  - Set of "terminals" (or "privileged nodes") $U \subseteq V$
- Problem: Find a tree $T \subseteq G$ containing all terminals, i.e. all nodes in U, which minimizes the cost:

$$C(T) = \sum_{ij \in E(T)} c_{ij}$$

3. The Steiner Tree Problem

- Given
  - Graph $G = (V, E)$
  - Costs $\{c_{ij}\}_{ij \in E}$, $c_{ij} \geq 0$
  - Set of "terminals" (or "privileged nodes") $U \subseteq V$
- Problem: Find a tree $T \subseteq G$ containing all terminals, i.e. all nodes in U, which minimizes the cost:

$$C(T) = \sum_{ij \in E(T)} c_{ij}$$

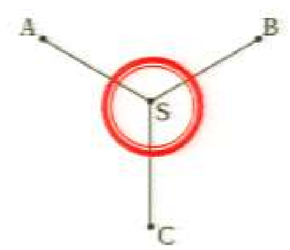


- The non-terminals which appear in the minimizing tree are called Steiner nodes
- Idea: Do belief propagation to find minimizing tree

3. The Steiner Tree Problem

- Given
  - Graph $G = (V, E)$
  - Costs $\{c_{ij}\}_{ij \in E}$, $c_{ij} \geq 0$
  - Set of "terminals" (or "privileged nodes") $U \subseteq V$
- Problem: Find a tree $T \subseteq G$ containing all terminals, i.e. all nodes in U, which minimizes the cost:

$$C(T) = \sum_{ij \in E(T)} c_{ij}$$



- The non-terminals which appear in the minimizing tree are called Steiner nodes
- Idea: Do belief propagation to find minimizing tree
- Difficulty: Don't have a local way to enforce the global constraint of a (connected) tree

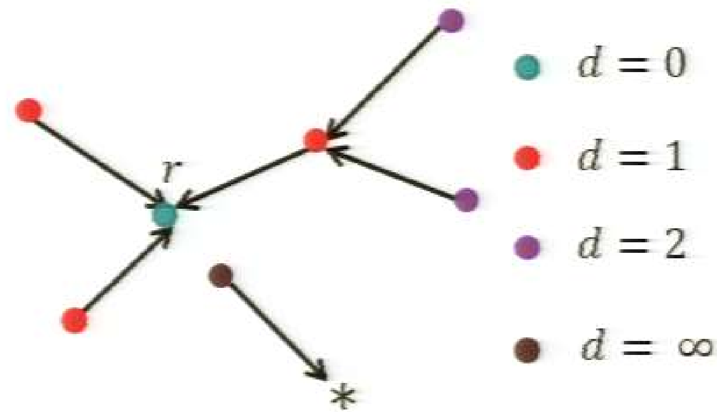
New Representation

Bayati, Borgs, Braunstein, Chayes, Ramezanpour, Zecchina ('08)

New Representation

Bayati, Borgs, Braunstein, Chayes, Ramezanpour, Zecchina ('08)

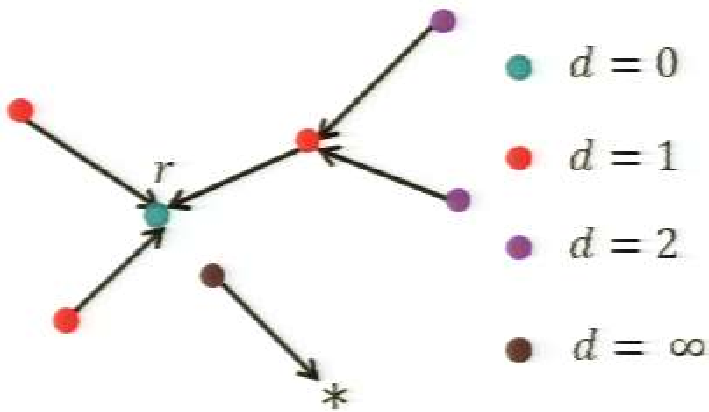
- Designate one terminal $r \in U$ as root and set $c_{rr} = 0$
- $\forall i \in V$, introduce two variables
  - Distance: $d_i \in \{0, 1, \ldots, |V| - 1\}$
  - Parent: $p_i \in N(i) \cup \{*\}$
- If T is a Steiner tree, set
  - $d_i = \mathrm{dist}_T(i, r) \quad \forall i \in V(T)$
  - $p_i = \begin{cases} * \text{ if} & i \neq V(T) \\ i \text{ if} & i = r \\ \text{parent of } i \text{ in } T & \text{otherwise} \end{cases}$



New Representation

Bayati, Borgs, Braunstein, Chayes, Ramezanpour, Zecchina ('08)

- Designate one terminal $r \in U$ as root and set $c_{rr} = 0$
- $\forall i \in V$, introduce two variables
  - Distance: $d_i \in \{0, 1, \ldots, |V| - 1\}$
  - Parent: $p_i \in N(i) \cup \{*\}$
- If T is a Steiner tree, set
  - $d_i = \mathrm{dist}_T(i, r) \quad \forall i \in V(T)$
  - $p_i = \begin{cases} * \text{ if} & i \neq V(T) \\ i \text{ if} & i = r \\ \text{parent of } i \text{ in } T & \text{otherwise} \end{cases}$



- Cost of the tree: $C(T) = \sum_{i \in V(G)} c_{ip_i} \mathbb{I}(p_i \neq *)$
- Constraints:
  - $p_i \neq * \quad \forall i \in U$
  - If $p_k = j \notin \{*, r\}$, then $p_j \neq *$ and $d_j = d_k - 1$

Graphical Model

- Define interactions enforcing these constraints (and including the weights):

$$\psi_{jk} = \left[1 - \mathbb{I}(p_k = j)\mathbb{I}\left(d_j \neq d_k - 1\right)\right]\left[1 - \mathbb{I}(p_k = j)\mathbb{I}\left(p_j = *\right)\right]$$

and

$$\varphi_i = [1 - \mathbb{I}(i \in U)\mathbb{I}(p_i = *)] \exp\left[-\beta c_{ip_i}\mathbb{I}(p_i \neq *)\right]$$

Graphical Model

- Define interactions enforcing these constraints (and including the weights):

$$\psi_{jk} = \left[1 - \mathbb{I}(p_k = j)\mathbb{I}\left(d_j \neq d_k - 1\right)\right]\left[1 - \mathbb{I}(p_k = j)\mathbb{I}\left(p_j = *\right)\right]$$

and

$$\varphi_i = \left[1 - \mathbb{I}(i \in U)\mathbb{I}(p_i = *)\right] \exp\left[-\beta c_{i p_i} \mathbb{I}(p_i \neq *)\right]$$

- Then the probability distribution is

$$\mu(\{d_i, p_i\}) = \frac{1}{z} \prod_{i \in V} \varphi_i \prod_{i,j \in V;\ ij \in E} \psi_{ij}$$

Graphical Model

- Define interactions enforcing these constraints (and including the weights):

$$\psi_{jk} = [1 - \mathbb{I}(p_k = j)\mathbb{I}(d_j \neq d_k - 1)][1 - \mathbb{I}(p_k = j)\mathbb{I}(p_j = *)]$$

and

$$\varphi_i = [1 - \mathbb{I}(i \in U)\mathbb{I}(p_i = *)]\exp[-\beta c_{ip_i}\mathbb{I}(p_i \neq *)]$$

- Then the probability distribution is

$$\mu(\{d_i, p_i\}) = \frac{1}{z}\prod_{i \in V}\varphi_i \prod_{i,j \in V;\ ij \in E}\psi_{ij}$$

- Variants:
  - Bounded diameter D tree: Take $d_i \in \{0,1,\ldots,D\}$
  - Prize-collecting Steiner tree: Replace φ_i by soft constraints, removing $\mathbb{I}(i \in U)$ and adding "prizes" to cost function

See Angel, Flaxman, Wilson ('08 –'12)

BP Results on the Steiner Tree

BP Results on the Steiner Tree

- Rigorous Results: Minimum spanning tree
  - If BP converges, then it converges to the correct solution (Bayati, Braunstein and Zecchina '08)
- Non-Rigorous Results: Minimum Steiner tree
  - Tests of our BP algorithm vs. LP algorithms for a benchmark library of several dozen Steiner tree instances (SteinLib), show that our algorithm is *much faster*. Also, it gets better optima in all but two (very small) instances (Bailley-Bechet, Borgs, Braunstein, Chayes, Dagkessamanskaia, Francois, Zecchina '11)
  - On biological data sets in the Fraenkel Lab at MIT, the LP algorithms were too slow to give any results on human data

BP Results on the Steiner Tree

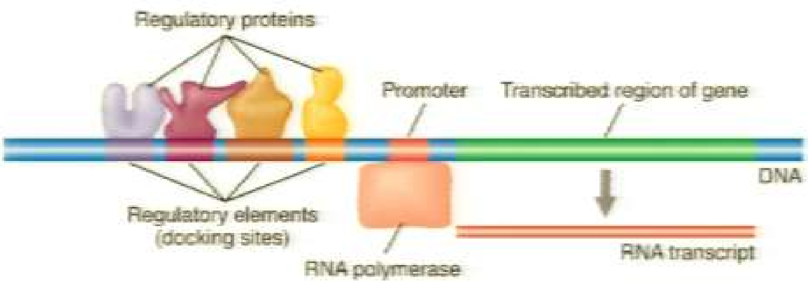
- Rigorous Results: Minimum spanning tree
  - If BP converges, then it converges to the correct solution (Bayati, Braunstein and Zecchina '08)
- Non-Rigorous Results: Minimum Steiner tree
  - Tests of our BP algorithm vs. LP algorithms for a benchmark library of several dozen Steiner tree instances (SteinLib), show that our algorithm is *much faster*. Also, it gets better optima in all but two (very small) instances (Bailley-Bechet, Borgs, Braunstein, Chayes, Dagkessamanskaia, Francois, Zecchina '11)
  - On biological data sets in the Fraenkel Lab at MIT, the LP algorithms were too slow to give any results on human data
- Open Problem: Find sufficient conditions for BP for the MWST to converge to the correct solution, or at least to a solution within ϵ of an optimizer.

4. Application to Systems Biology

- The Biological Problem
- Formulation of the Algorithmic Problem: The Prize-Collecting Steiner Tree (PCST)
- Biological Network Applications of the PCST
- A Variant Algorithmic Problem: The Prize-Collecting Steiner Forest (Parallel Networks)
- Construction of Patient-Specific Networks

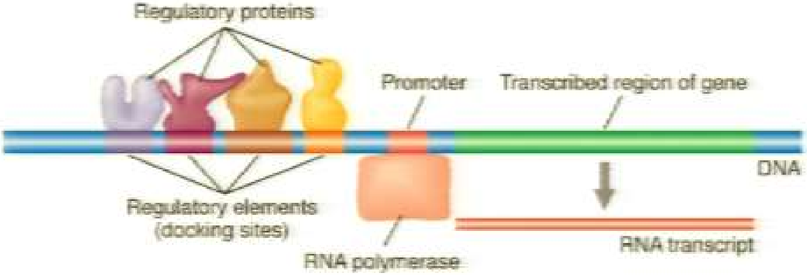
The Biological Problem

- Standard Dogma: DNA → RNA → Proteins

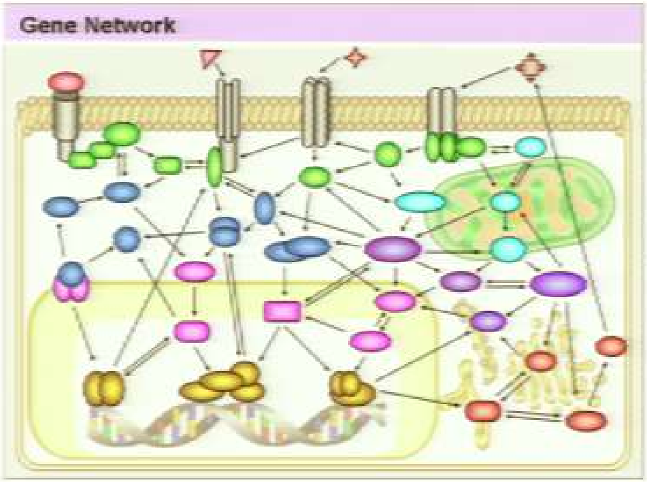


The Biological Problem

- Standard Dogma: DNA → RNA → Proteins



⇒ Gene Regulatory Network

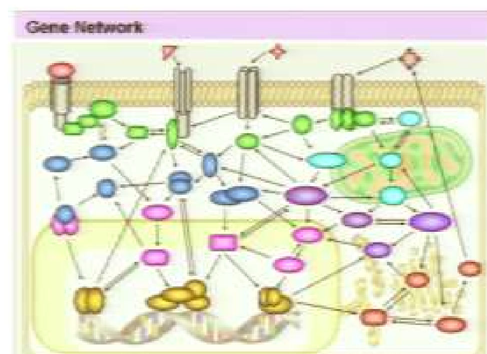


Protein Interactome

Gene Regulation and Disease

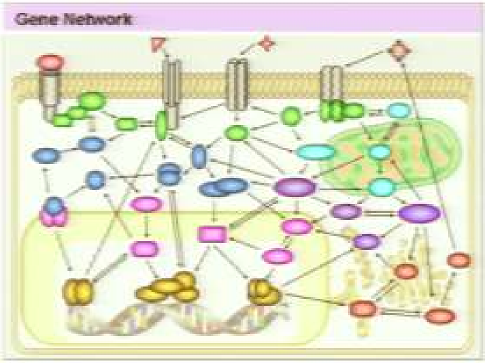
Gene Regulation and Disease

- Problems with the gene regulatory network are the sources of many diseases



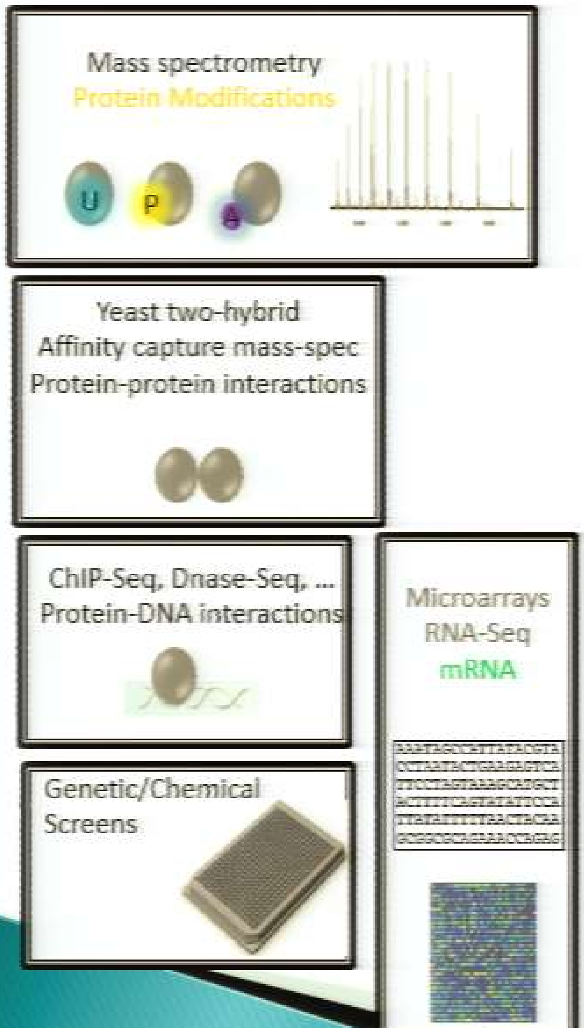
Gene Regulation and Disease

- Problems with the gene regulatory network are the sources of many diseases
- How do we infer the network structure from partial data?
- Can we identify particular nodes on the network responsible for dysregulation in certain diseases and individuals?
- Are one or more nodes in combination viable drug targets?

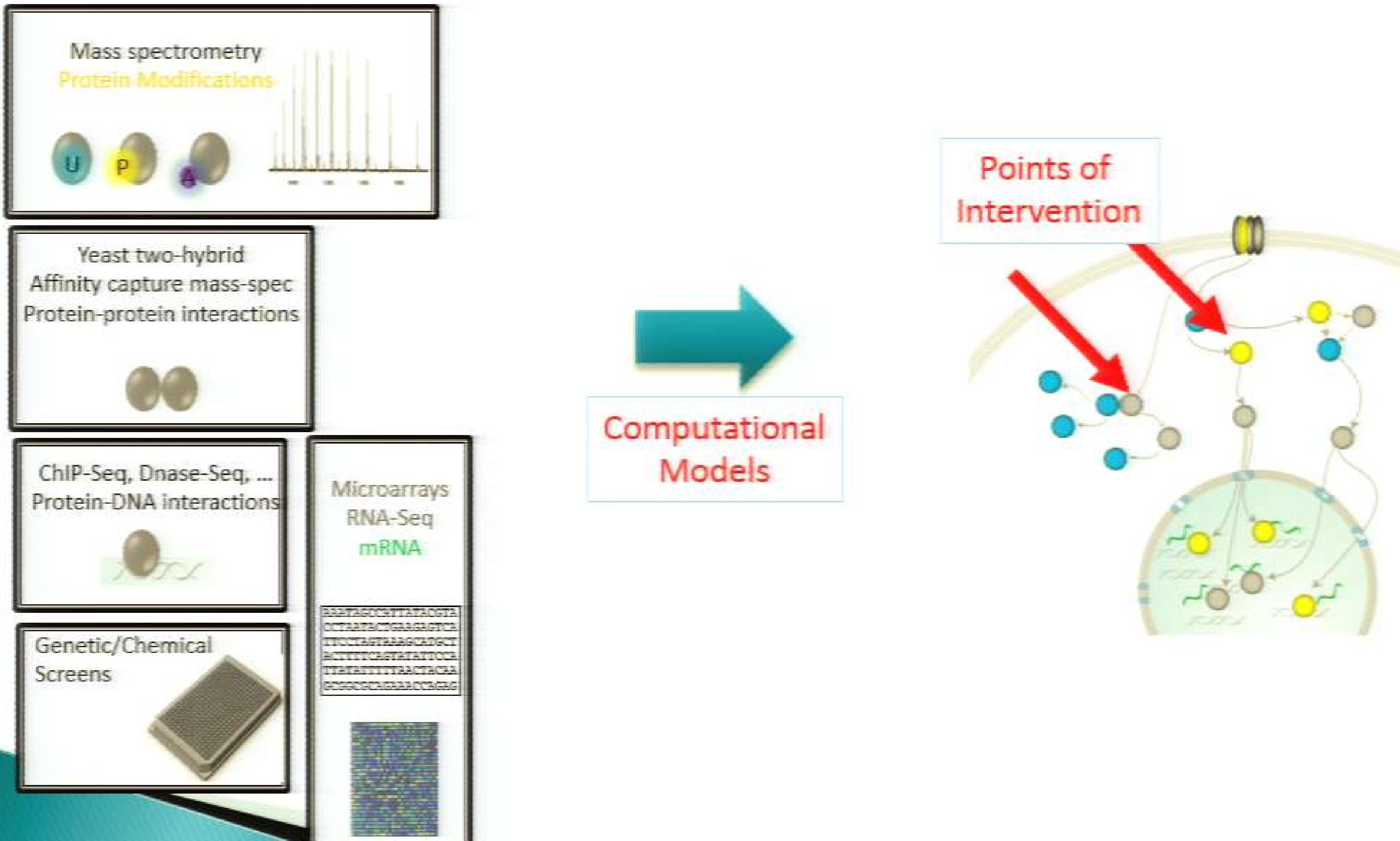


Drug Discovery Paradigm

Drug Discovery Paradigm

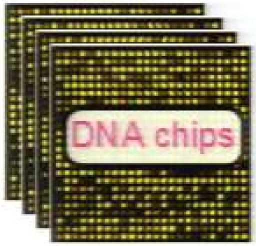


Drug Discovery Paradigm



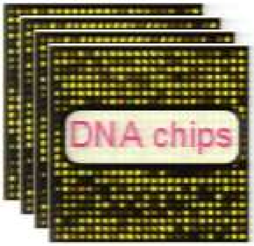
Gene Expression Data

Gene Expression Data



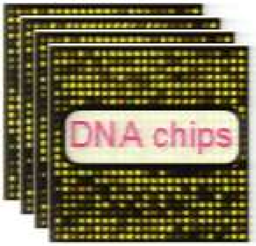
- Microarrays tell us which gene is expressed in the presence of which other gene under a particular set of conditions

Gene Expression Data



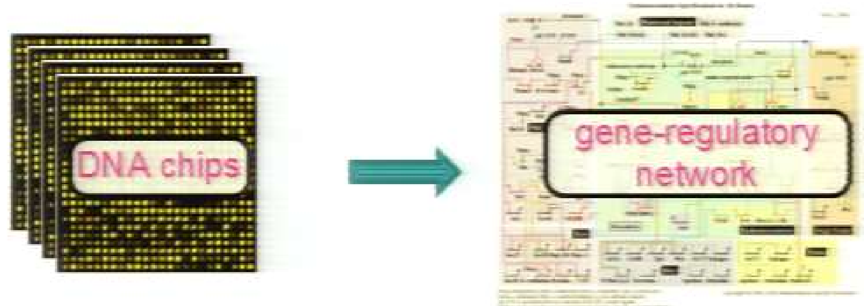
- Microarrays tell us which gene is expressed in the presence of which other gene under a particular set of conditions
- From the differential expression of a particular gene, we infer the node weight of the corresponding transcription factor protein (prize in the PCST)

Gene Expression Data



- Microarrays tell us which gene is expressed in the presence of which other gene under a particular set of conditions
- From the differential expression of a particular gene, we infer the node weight of the corresponding transcription factor protein (prize in the PCST)
- To get edge weights between two proteins, we use the probability of interaction of these two proteins inferred from (properly weighted) databases of known interactions for the given organism

Gene Expression Data



- Microarrays tell us which gene is expressed in the presence of which other gene under a particular set of conditions
- From the differential expression of a particular gene, we infer the node weight of the corresponding transcription factor protein (prize in the PCST)
- To get edge weights between two proteins, we use the probability of interaction of these two proteins inferred from (properly weighted) databases of known interactions for the given organism

Question: How do we determine the network most likely to have produced this data?

Formulation of the Problem: The Prize-Collecting Steiner Tree

Formulation of the Problem: The Prize-Collecting Steiner Tree

- Given
  - Graph $G = (V, E)$
  - Costs $\{c_{ij}\}_{ij \in E}$, $c_{ij} \geq 0$
  - Set of "prize terminals" $U \subseteq V$ with prizes $\{\pi_i\}_{i \in U}$, $\pi_i > 0$
  - Parameter $\lambda > 0$
- Problem: Find a tree $T \subseteq G$ which minimizes the cost:

$$C(T) = \sum_{ij \in E(T)} c_{ij} - \lambda \sum_{i \in V(T)} \pi_i$$

Formulation of the Problem: The Prize-Collecting Steiner Tree

- Given
  - Graph $G = (V, E)$
  - Costs $\{c_{ij}\}_{ij \in E}$, $c_{ij} \geq 0$
  - Set of "prize terminals" $U \subseteq V$ with prizes $\{\pi_i\}_{i \in U}$, $\pi_i > 0$
  - Parameter $\lambda > 0$
- Problem: Find a tree $T \subseteq G$ which minimizes the cost:

$$C(T) = \sum_{ij \in E(T)} c_{ij} - \lambda \sum_{i \in V(T)} \pi_i$$

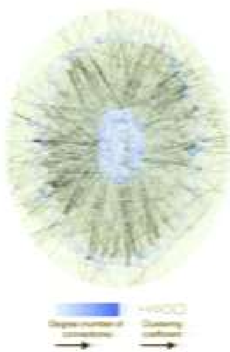
- Note: As $\lambda \rightarrow \infty$, this turns into the standard Steiner tree problem with terminals $U = \{i | \pi_i > 0\}$.

Mapping to Biological Data

Mapping to Biological Data

- Find the tree which minimizes

$$C(T) = \sum_{ij \in E(T)} c_{ij} - \lambda \sum_{i \in V(T)} \pi_i$$



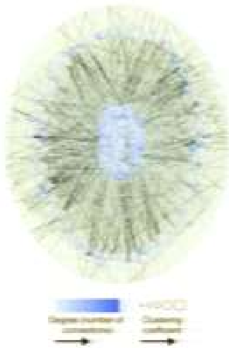
$$c_{ij} = -\log \text{prob}(ij \text{ exists})$$

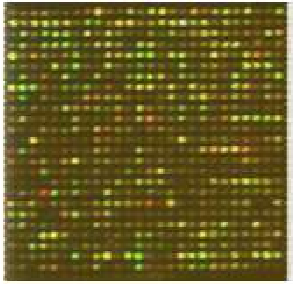
where prob(ij exists) is the probability that proteins i and j interact in the given organism (from databases)

Mapping to Biological Data

- Find the tree which minimizes

$$C(T) = \sum_{ij \in E(T)} c_{ij} - \lambda \sum_{i \in V(T)} \pi_i$$





$$c_{ij} = -\log \mathrm{prob}(ij \text{ exists})$$

where $\mathrm{prob}(ij \text{ exists})$ is the probability that proteins i and j interact in the given organism (from databases)

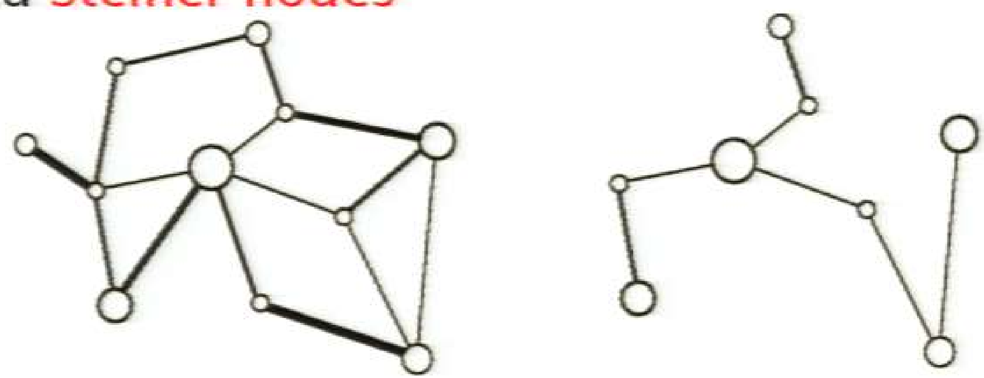
$$\pi_i = -\log p_{\mathrm{value}}(i)$$

where $p_{\mathrm{value}}(i)$ is the p-value of the differential expression of the gene corresponding to protein i, in the given experiment

Steiner Nodes

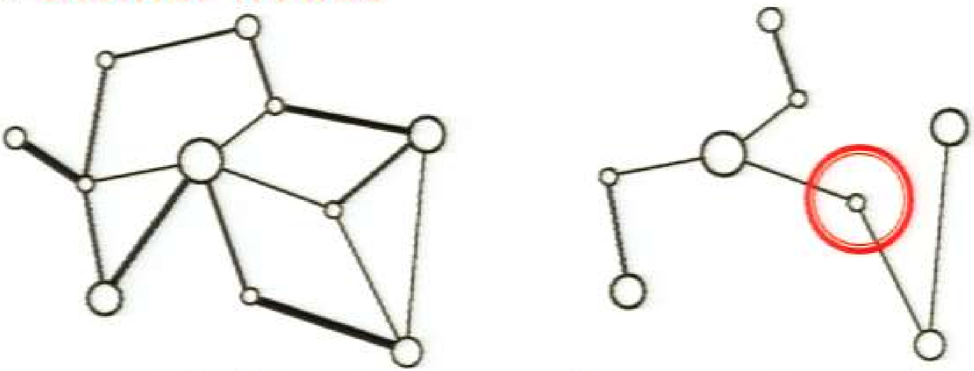
Steiner Nodes

- In the standard Steiner tree problem, nodes which are included in the minimizing solution but which are not terminals, i.e. not in the set U, are called Steiner nodes
- Similarly, in the PCST, nodes which have zero (or low) prizes but which are included in the minimizing solution are called Steiner nodes



Steiner Nodes

- In the standard Steiner tree problem, nodes which are included in the minimizing solution but which are not terminals, i.e. not in the set U, are called Steiner nodes
- Similarly, in the PCST, nodes which have zero (or low) prizes but which are included in the minimizing solution are called Steiner nodes



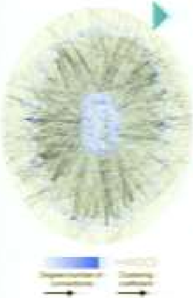
- In the context of the gene regulatory networks, Steiner nodes correspond to proteins whose genes which are not differentially expressed a lot, but which nevertheless seem likely to participate in the network ⇒ identification of proteins not previously know to participate in the pathway

Example 1: Yeast Pheromone Response Pathway

(Bailley-Bechet, Borgs, Braunstein, Chayes, Dagkessamanskaia, Francois, Zecchina: PNAS '11)

Example 1: Yeast Pheromone Response Pathway

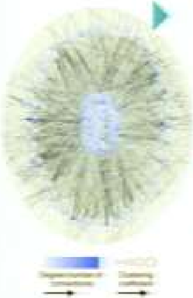
(Bailley-Bechet, Borgs, Braunstein, Chayes, Dagkessamanskaia, Francois, Zecchina: PNAS '11)



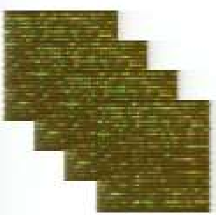
- Yeast protein signal transduction network:
  - 4689 Proteins
  - 14928 Protein-Protein interactions
  - Gives set of weights $\{c_{ij}\}$ for relevant proteins in pheromone response pathway

Example 1: Yeast Pheromone Response Pathway

(Bailley-Bechet, Borgs, Braunstein, Chayes, Dagkessamanskaia, Francois, Zecchina: PNAS '11)



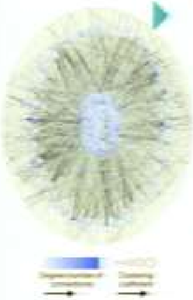
- Yeast protein signal transduction network:
  - 4689 Proteins
  - 14928 Protein-Protein interactions
  - Gives set of weights $\{c_{ij}\}$ for relevant proteins in pheromone response pathway



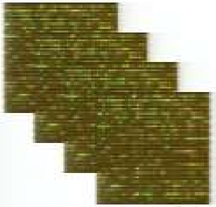
- Considered 56 large-scale gene expression data sets used to reconstruct the yeast pheromone pathway. For each data set
  - Get set of prizes $\{\pi_i\}$

Example 1: Yeast Pheromone Response Pathway

(Bailley-Bechet, Borgs, Braunstein, Chayes, Dagkessamanskaia, Francois, Zecchina: PNAS '11)



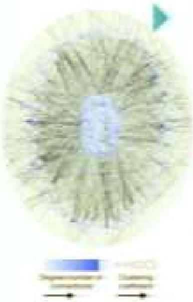
- Yeast protein signal transduction network:
  - 4689 Proteins
  - 14928 Protein-Protein interactions
  - Gives set of weights $\{c_{ij}\}$ for relevant proteins in pheromone response pathway



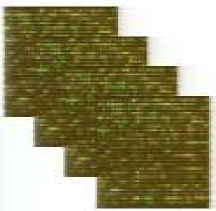
- Considered 56 large-scale gene expression data sets used to reconstruct the yeast pheromone pathway. For each data set
  - Get set of prizes $\{\pi_i\}$
- Construct 56 solutions to bounded-D PCST problem

Example 1: Yeast Pheromone Response Pathway

(Bailley-Bechet, Borgs, Braunstein, Chayes, Dagkessamanskaia, Francois, Zecchina: PNAS '11)



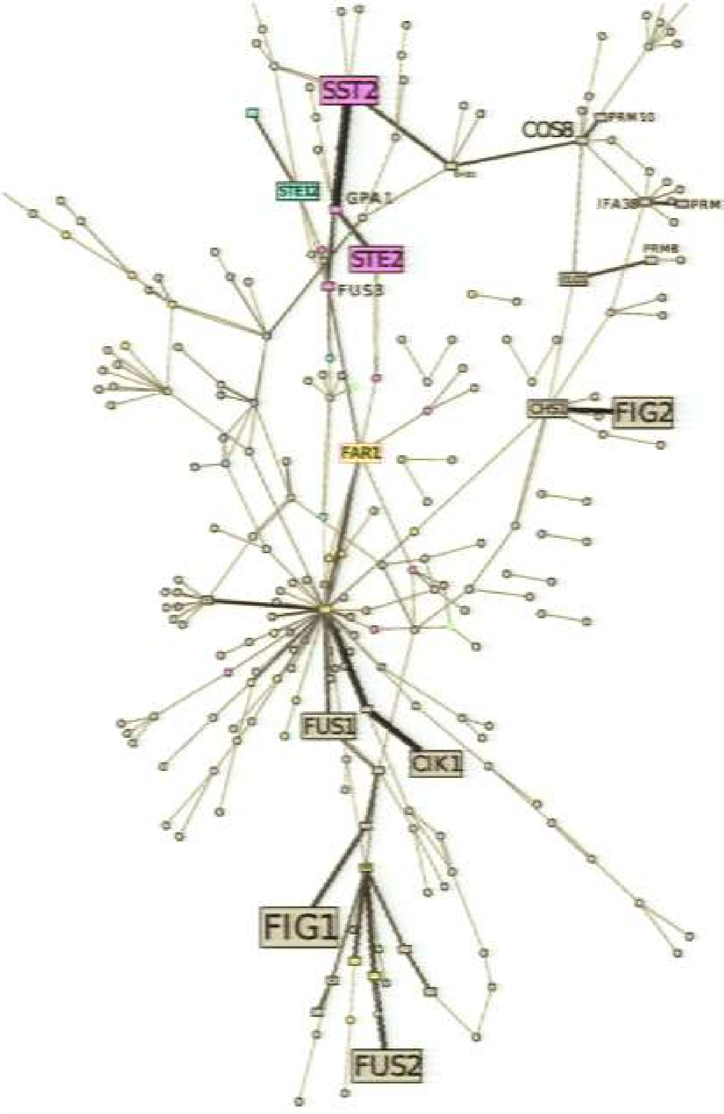
- Yeast protein signal transduction network:
  - 4689 Proteins
  - 14928 Protein-Protein interactions
  - Gives set of weights $\{c_{ij}\}$ for relevant proteins in pheromone response pathway



- Considered 56 large-scale gene expression data sets used to reconstruct the yeast pheromone pathway. For each data set
  - Get set of prizes $\{\pi_i\}$
- Construct 56 solutions to bounded-D PCST problem
- "Merge solutions" to get one network

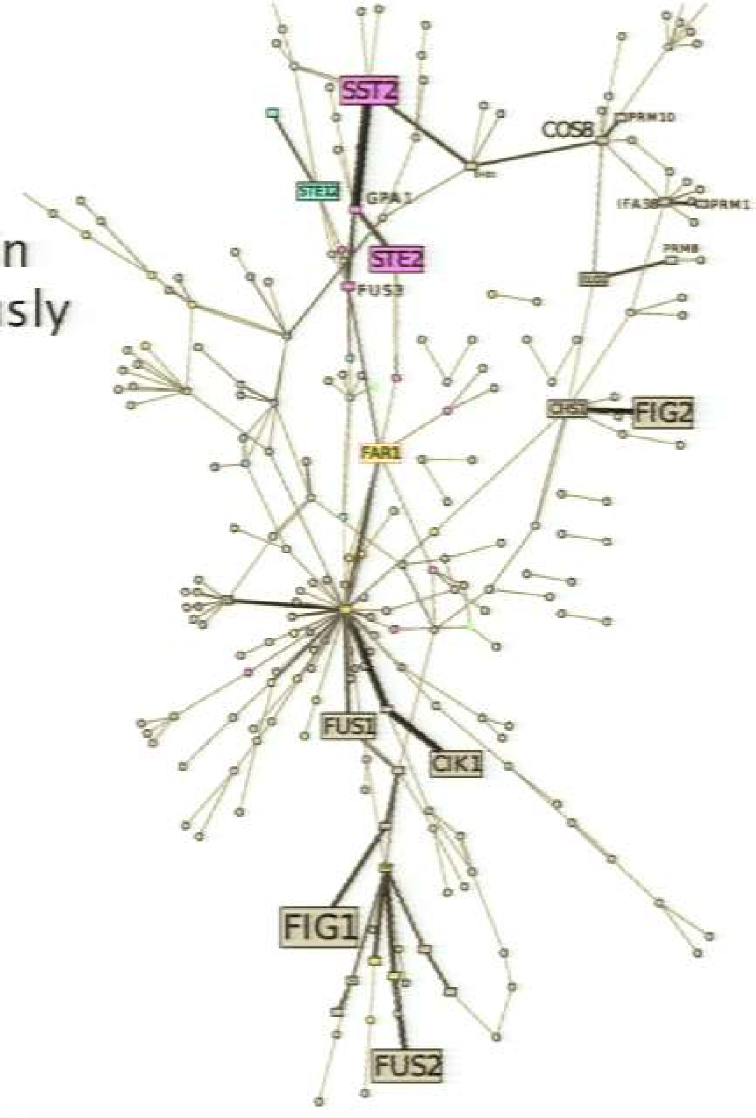
Results: Pathway identified

Results: Pathway identified



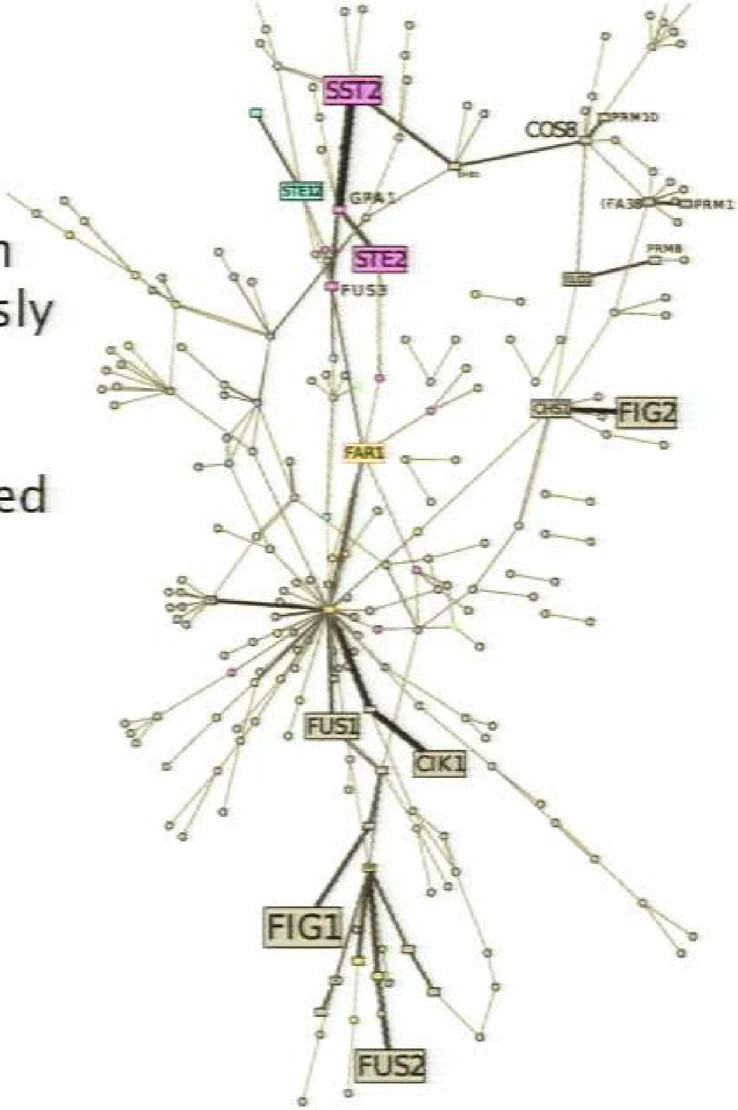
Results: Pathway identified

- Two types of proteins on network
  - Proteins differentially expressed in pheromone response and previously discovered by transcriptomic studies (terminals)



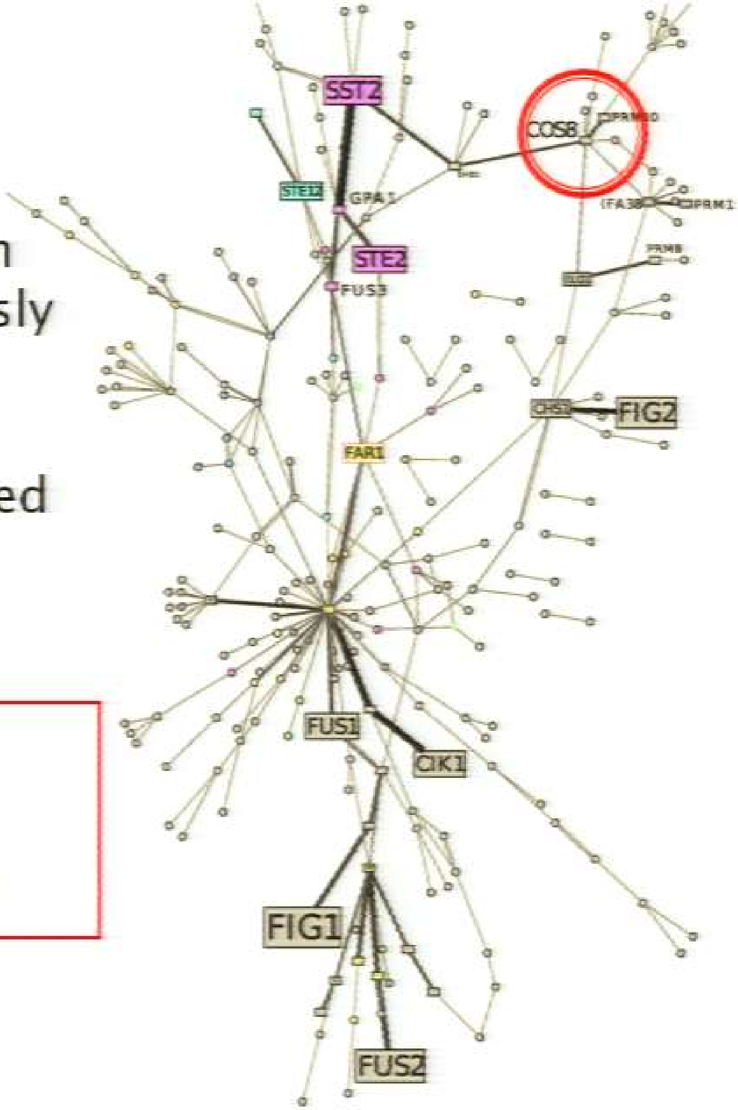
Results: Pathway identified

- Two types of proteins on network
  - Proteins differentially expressed in pheromone response and previously discovered by transcriptomic studies (terminals)
  - Proteins not differentially expressed but bridging between different subnetworks ("Steiner proteins")



Results: Pathway identified

- Two types of proteins on network
  - Proteins differentially expressed in pheromone response and previously discovered by transcriptomic studies (terminals)
  - Proteins not differentially expressed but bridging between different subnetworks ("Steiner proteins")



Question: Are the Steiner proteins important in the pheromone response pathway?

Testing a Steiner Node

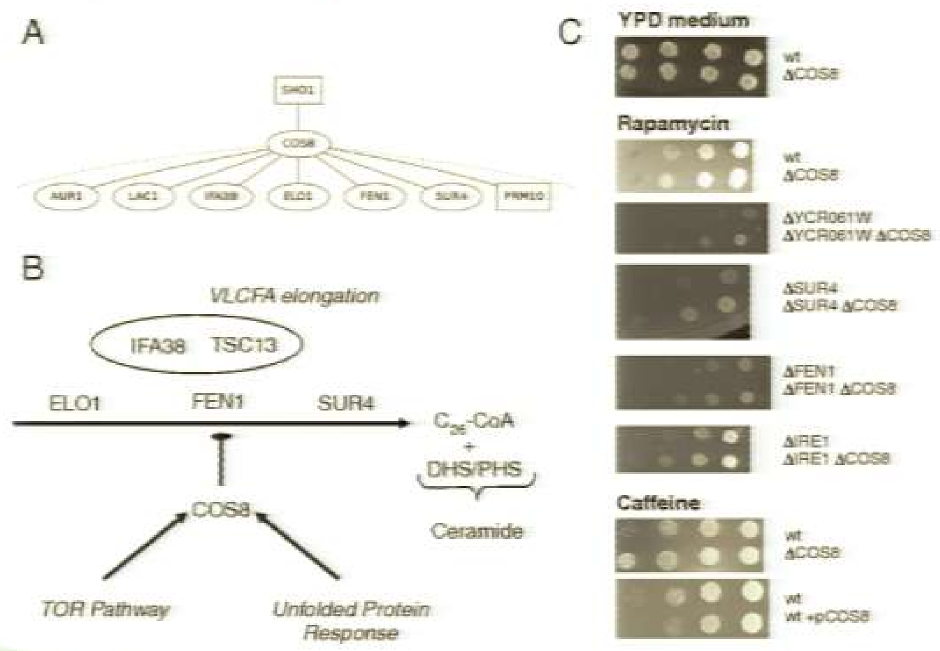
- Did an experiment to knock out the gene corresponding to COS8

Testing a Steiner Node

- Did an experiment to knock out the gene corresponding to COS8

$\Longrightarrow$

Pheromone response pathway failed.



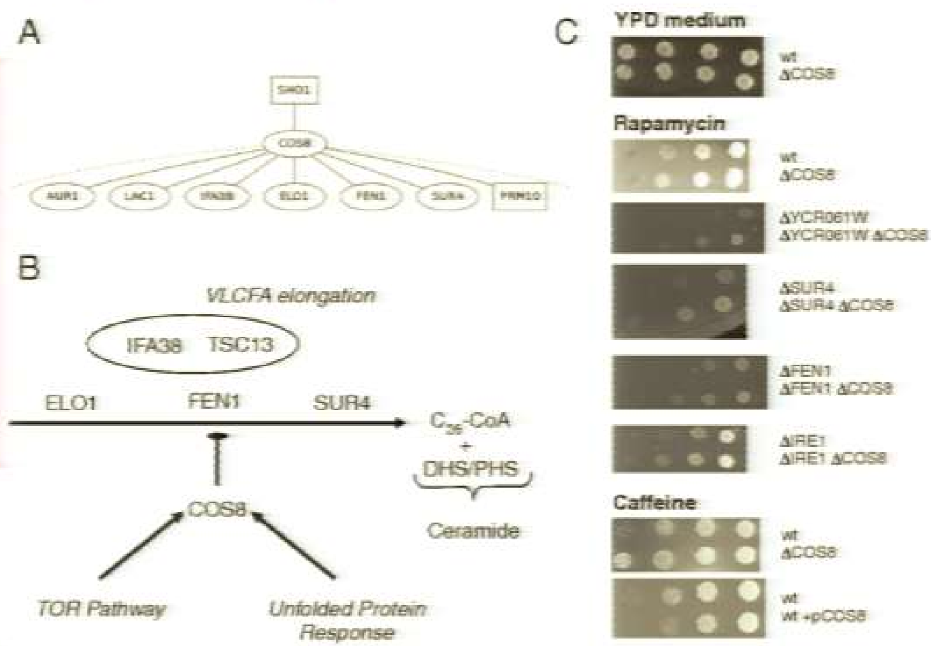
Testing a Steiner Node

- Did an experiment to knock out the gene corresponding to COS8

$\Longrightarrow$

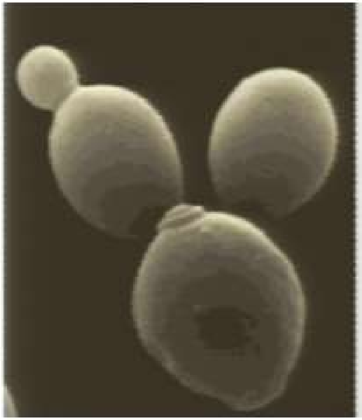
Pheromone **response pathway failed**.

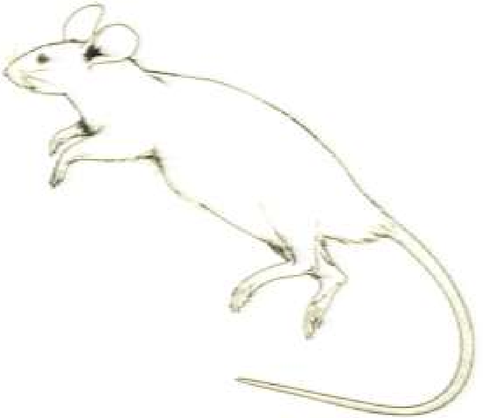
> "**Experimental proof**" of the importance of the Steiner node

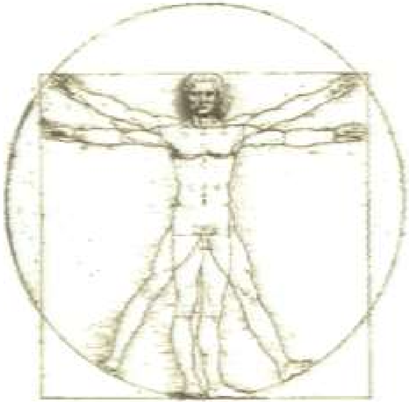


From Yeast to Mammals

- Problems (mammals relative to yeast):
  - Incomplete interactome data
  - Ten times as many transcription factors
  - Huge intergenic regions

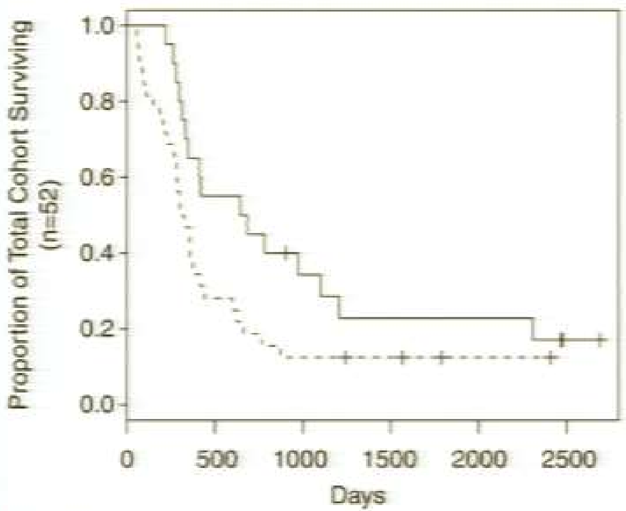




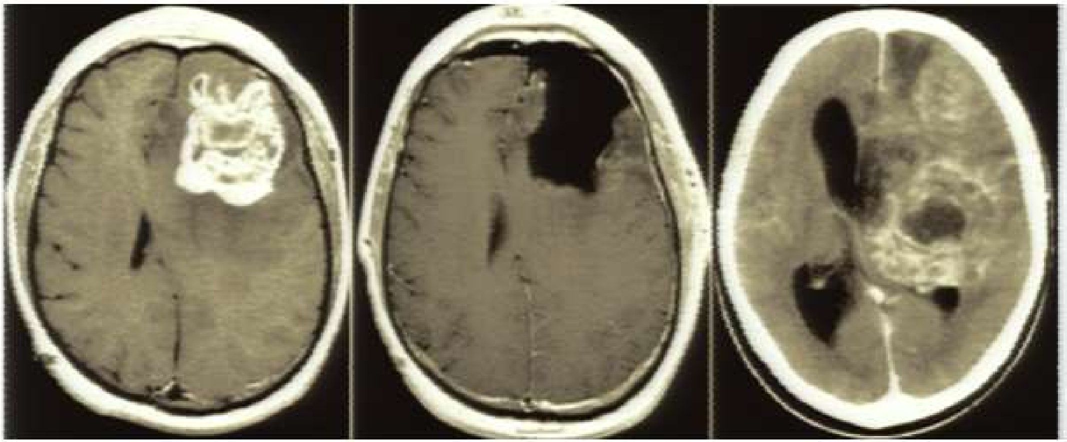


Example 2: Glioblastoma Pathways

- Glioblastoma:
  - particular form of brain cancer
  - the human cancer with the worst outcome
  - much more common in men than women



Pope W B et al. Radiology 2008;249:268-277

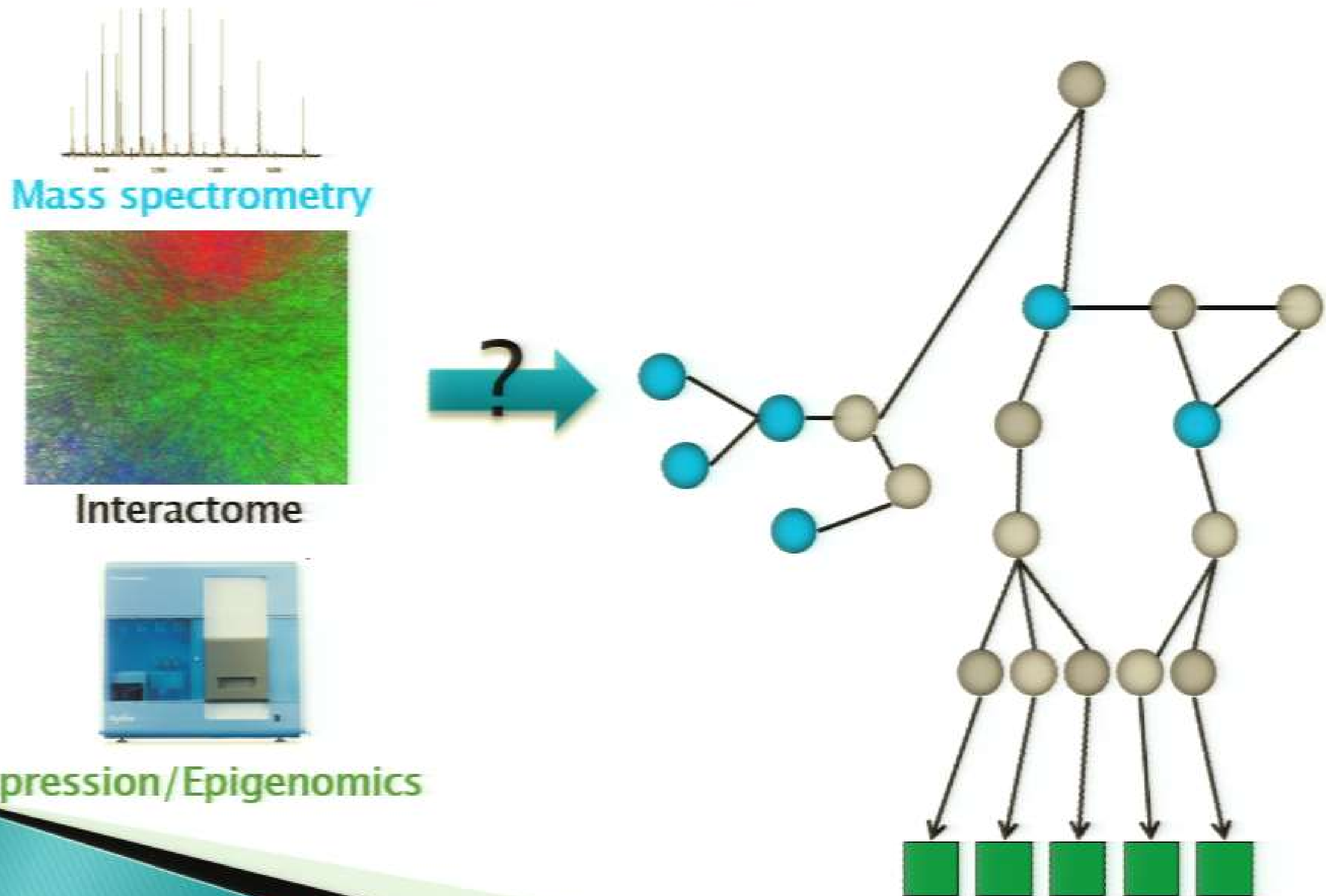


Presentation Post-op Recurrence

Weil RJ (2006) PLoS Med 3(1): e31.

Can we find GBM pathways using the PCST?

(Fraenkel Lab, MIT, work in progress using our PCST algorithm)



How to choose the root of the PCST?

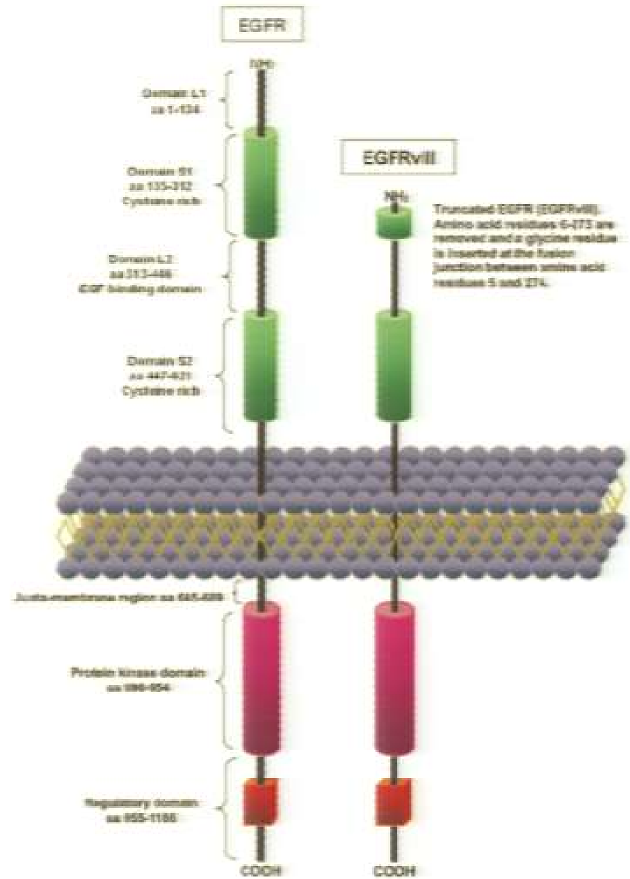
How to choose the root of the PCST?

Always good to choose receptor proteins since these often begin signaling pathways

How to choose the root of the PCST?

Always good to choose receptor proteins since these often begin signaling pathways

Try EGFR

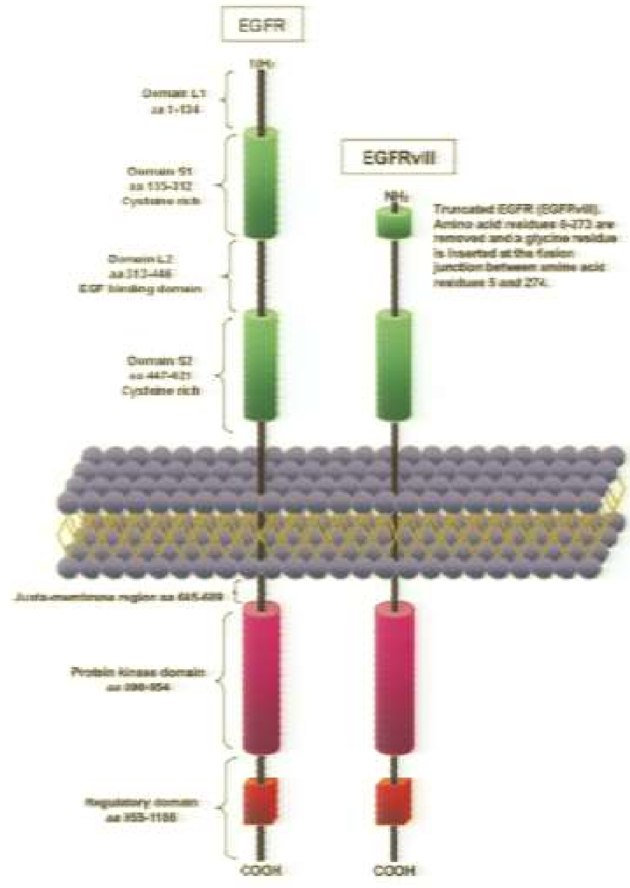


How to choose the root of the PCST?

Always good to choose receptor proteins since these often begin signaling pathways

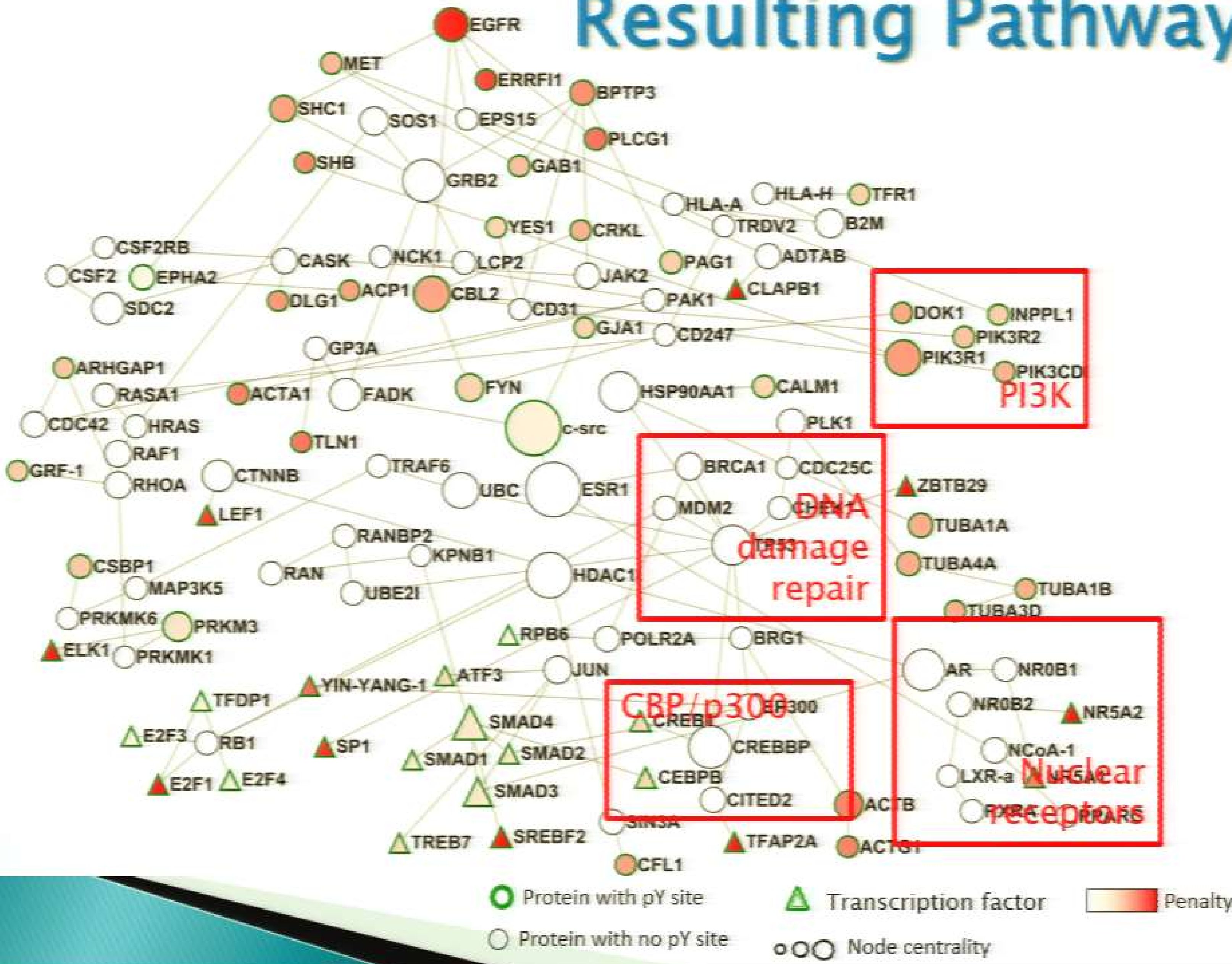
Try EGFR

- EGFR variant III mutation is most common EGFR mutation in human cancer
- Present in 60% of GBMs
- EGFRvIII expression correlates with shorter life expectancies

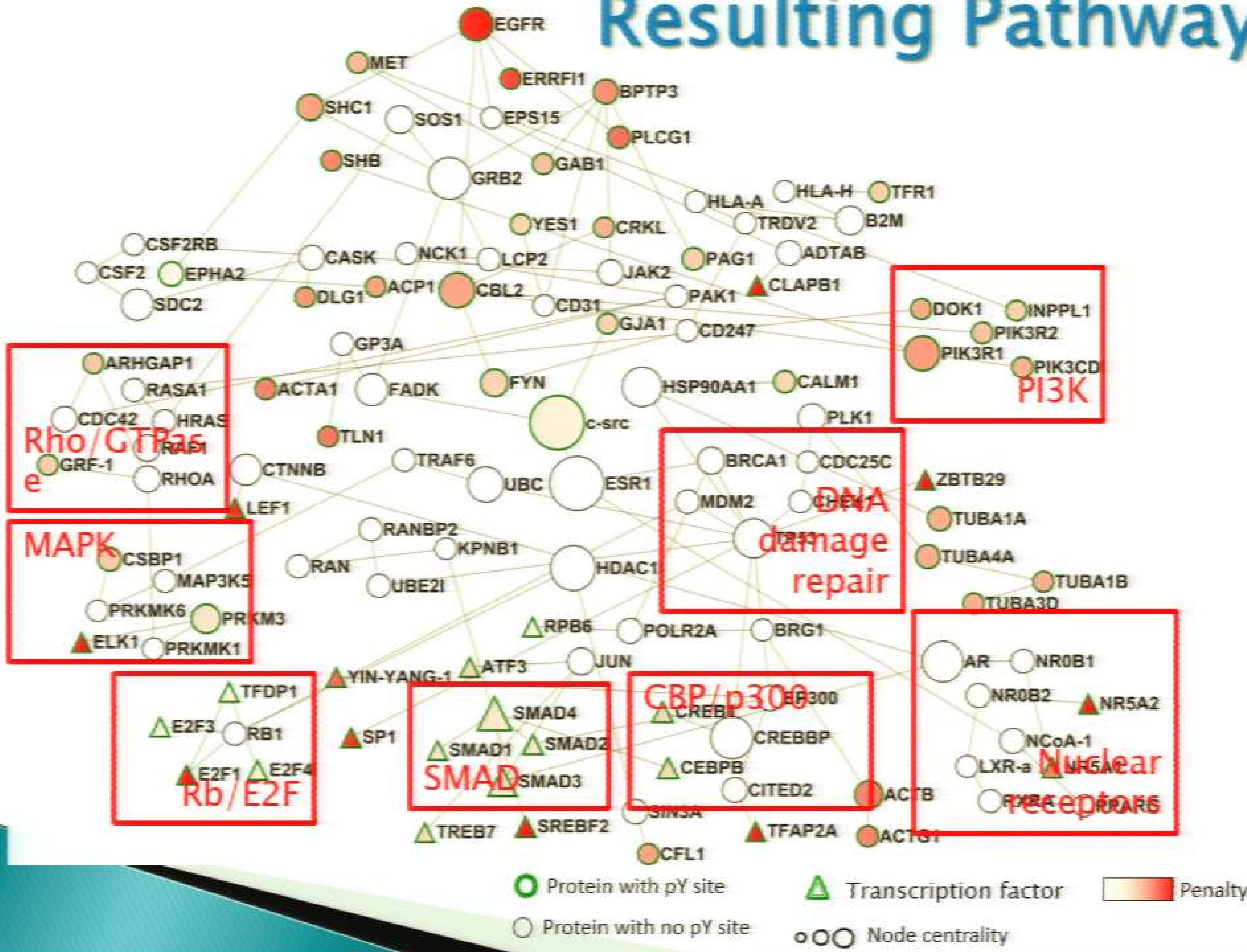


Resulting Pathway

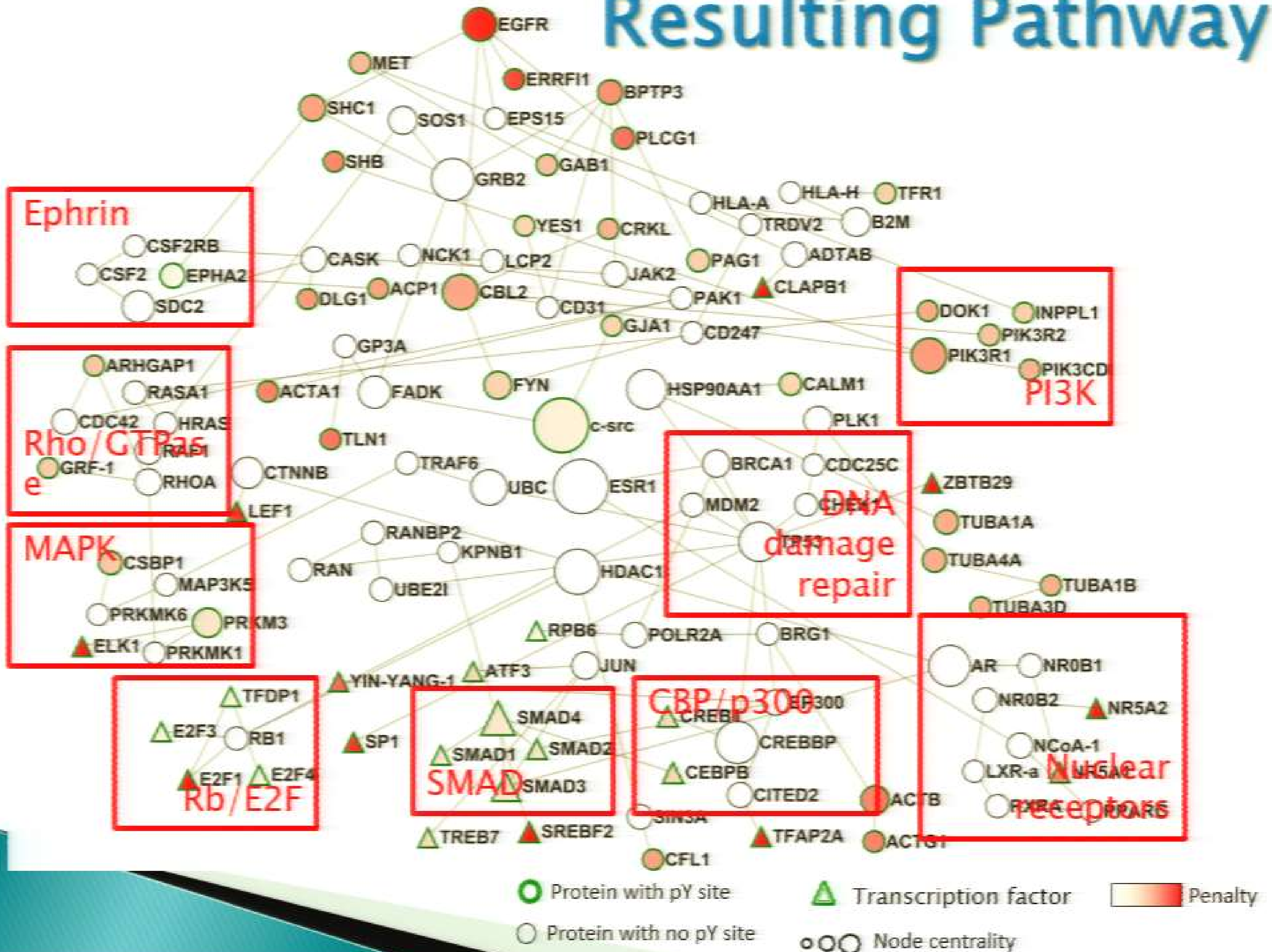
Resulting Pathway



Resulting Pathway

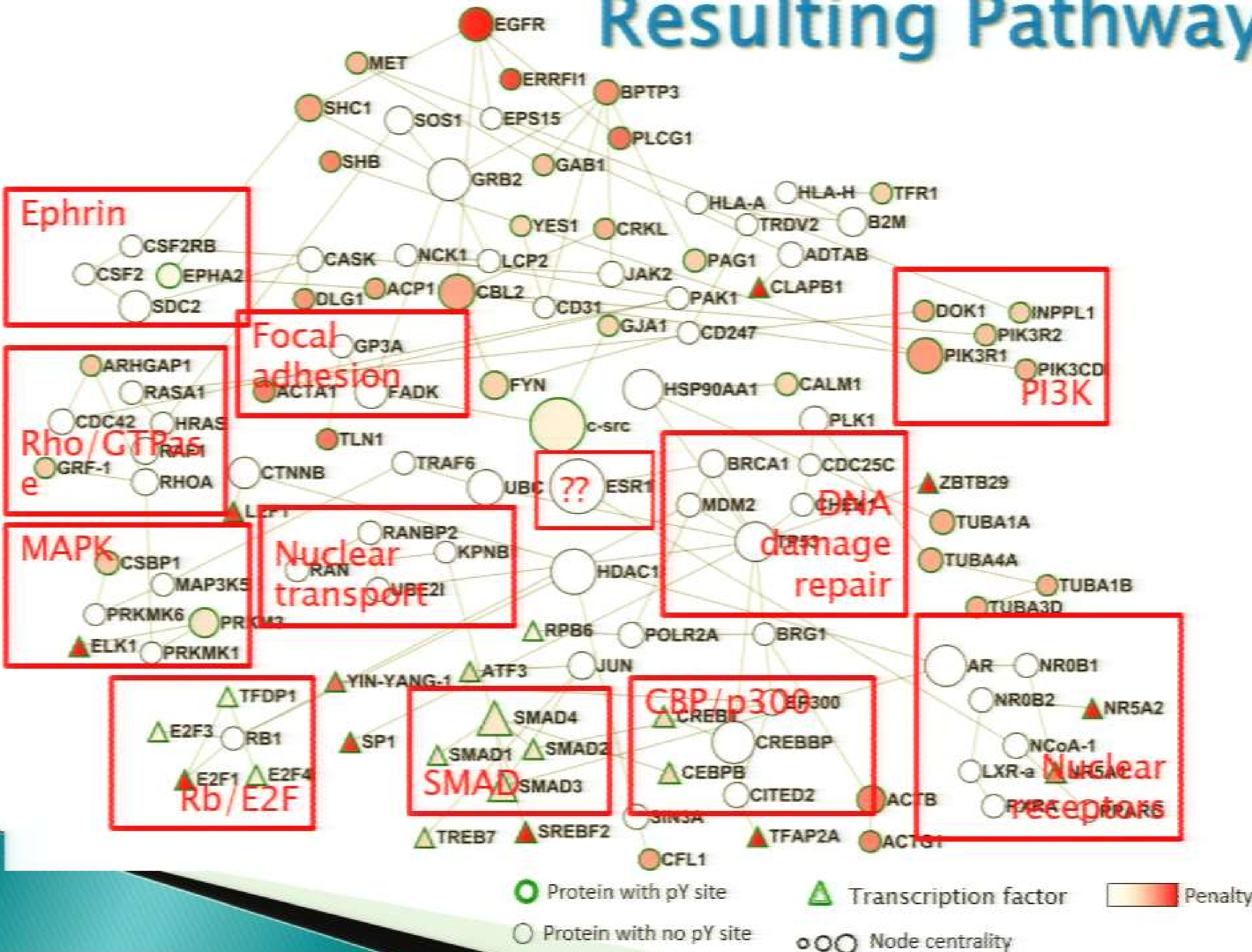


Resulting Pathway



Resulting Pathway

Resulting Pathway



Identify interesting Steiner nodes

Identify interesting Steiner nodes

- Top 5 Nodes ranked by betweeness centrality*:
  SRC, ESR1, HDAC1, CREBBP, GRB2
- SRC well-known to be active in many types of cancer, and had relatively large "prize"

*Relative percentage of shortest paths in graph through given node

Identify interesting Steiner nodes

- Top 5 Nodes ranked by betweeness centrality*:
  SRC, ESR1, HDAC1, CREBBP, GRB2
- SRC well-known to be active in many types of cancer, and had relatively large "prize"
- What about ESR1?
  - No "prize" and not previously identified for Glioblastoma
  - What is ESR1?

*Relative percentage of shortest paths in graph through given node

Identify interesting Steiner nodes

- Top 5 Nodes ranked by betweeness centrality*:
  SRC, ESR1, HDAC1, CREBBP, GRB2
- SRC well-known to be active in many types of cancer, and had relatively large "prize"
- What about ESR1?
  - No "prize" and not previously identified for Glioblastoma
  - What is ESR1?
  - This is the Estrogen Receptor
- **First pathway link between glioblastoma and gender!**

*Relative percentage of shortest paths in graph through given node

Identify interesting Steiner nodes

- Top 5 Nodes ranked by betweeness centrality*:
  SRC, ESR1, HDAC1, CREBBP, GRB2
- SRC well-known to be active in many types of cancer, and had relatively large "prize"
- What about ESR1?
  - No "prize" and not previously identified for Glioblastoma
  - What is ESR1?
  - This is the Estrogen Receptor
- **First pathway link between glioblastoma and gender!**
- Experimental test: EGFR inhibitor and Estrodiol together inhibit the growth of GBM cells in culture better than the EGFR inhibitor alone

*Relative percentage of shortest paths in graph through given node

Identify interesting Steiner nodes

- Top 5 Nodes ranked by betweeness centrality*:
  SRC, ESR1, HDAC1, CREBBP, GRB2
- SRC well-known to be active in many types of cancer, and had relatively large "prize"
- What about ESR1?
  - No "prize" and not previously identified for Glioblastoma
  - What is ESR1?
  - This is the Estrogen Receptor
- **First pathway link between glioblastoma and gender!**
- Experimental test: EGFR inhibitor and Estrodiol together inhibit the growth of GBM cells in culture better than the EGFR inhibitor alone

  $\Rightarrow$ possible drug therapy for glioblastoma

*Relative percentage of shortest paths in graph through given node

Multiple Signaling Pathways

(Tuncbag, Braunstein, Pagnani, Huang, Chayes, Borgs, Zecchina, Frankel; RECOMB '12)

Multiple Signaling Pathways

(Tuncbag, Braunstein, Pagnani, Huang, Chayes, Borgs, Zecchina, Frankel; RECOMB '12)

- How do we explain multiple disjoint signaling pathways altered in a particular condition?
- Use Prize-Collecting Steiner Forest:
- Just like prize-collecting Steiner tree, but now we also specify that there be k disjoint trees* (= forest F) as the minimizing solution of

$$C(F) = \sum_{ij \in E(F)} c_{ij} - \lambda \sum_{i \in V(F)} \pi_i$$

*Or let k vary by adding another term to C

Multiple Signaling Pathways

(Tuncbag, Braunstein, Pagnani, Huang, Chayes, Borgs, Zecchina, Frankel; RECOMB '12)

- How do we explain multiple disjoint signaling pathways altered in a particular condition?
- Use Prize-Collecting Steiner Forest:
- Just like prize-collecting Steiner tree, but now we also specify that there be k disjoint trees* (= forest F) as the minimizing solution of

$$C(F) = \sum_{ij \in E(F)} c_{ij} - \lambda \sum_{i \in V(F)} \pi_i$$

- To implement PCSF, just add an "artificial node" A, connect every node i to A with strength c_{iA} ⇒ new PCST with 1 more node and $|V|$ more edges

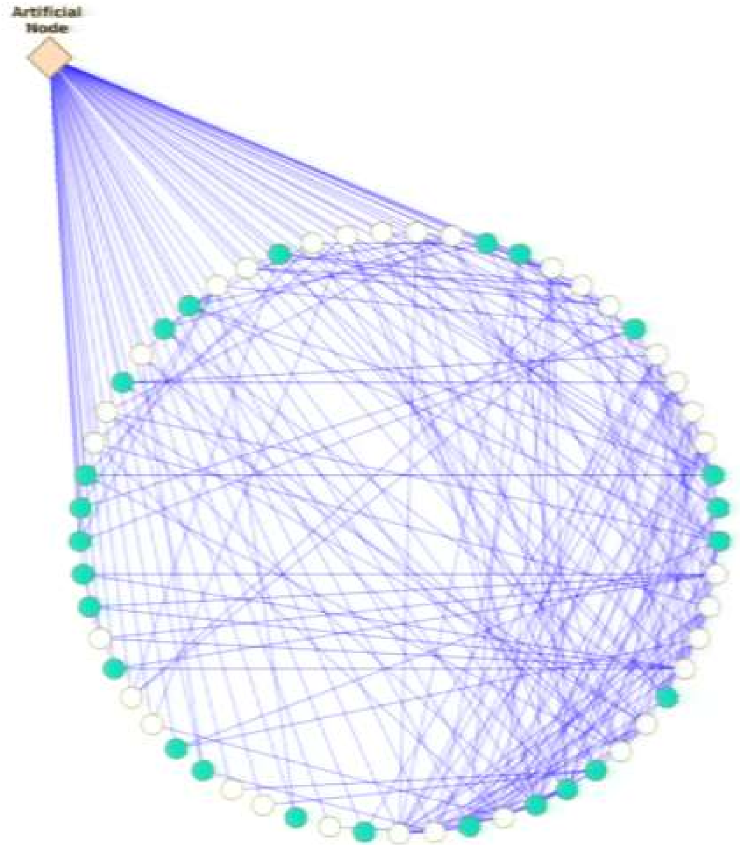
*Or let k vary by adding another term to C

Method

Prize Collecting Steiner Forest

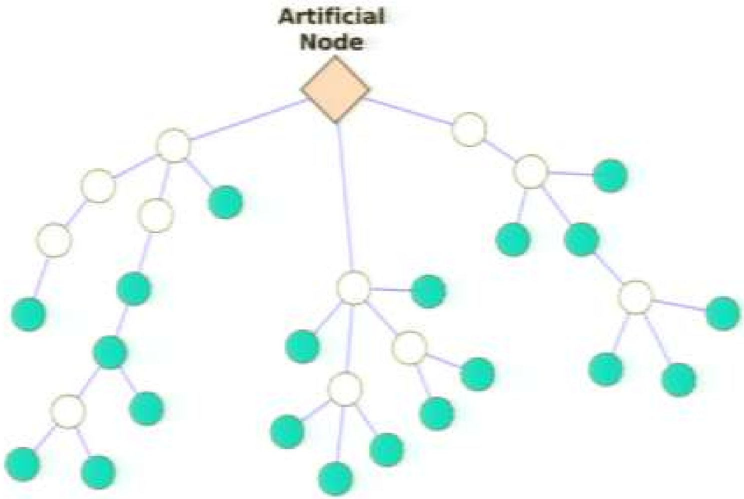
Method

Prize Collecting Steiner Forest

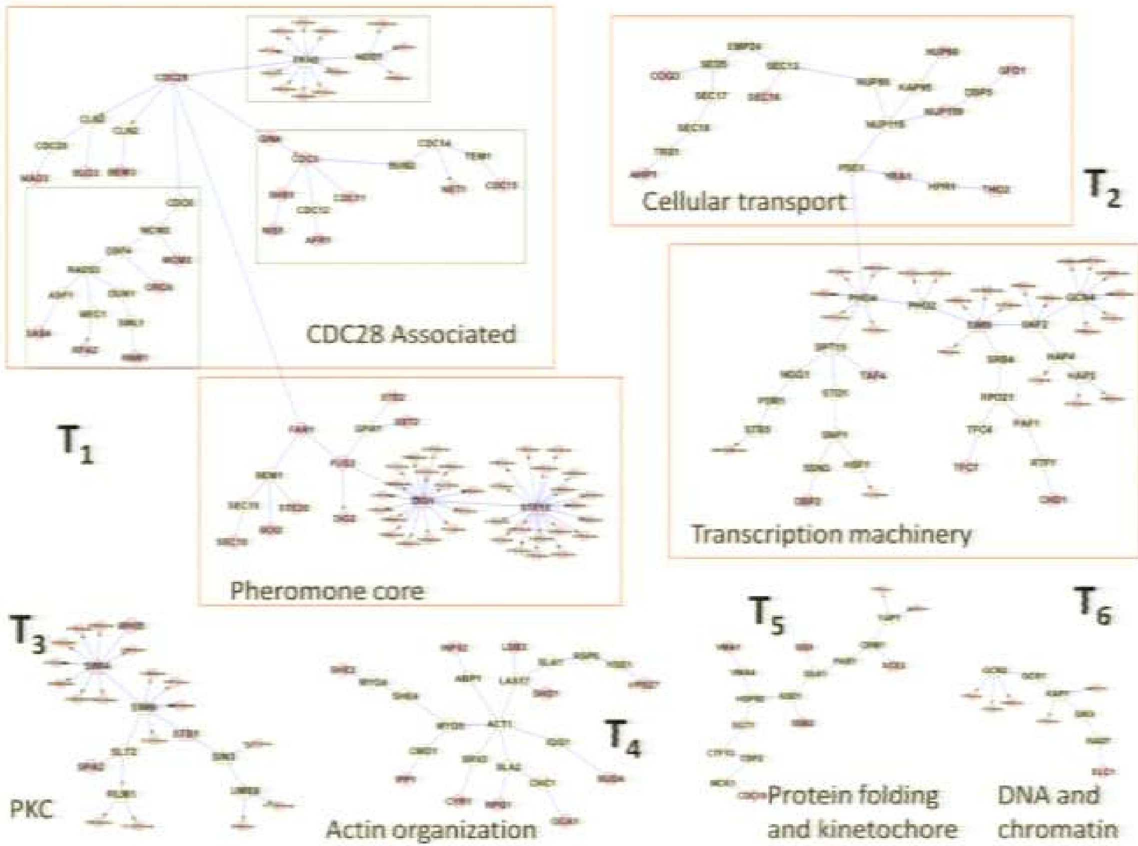


Method

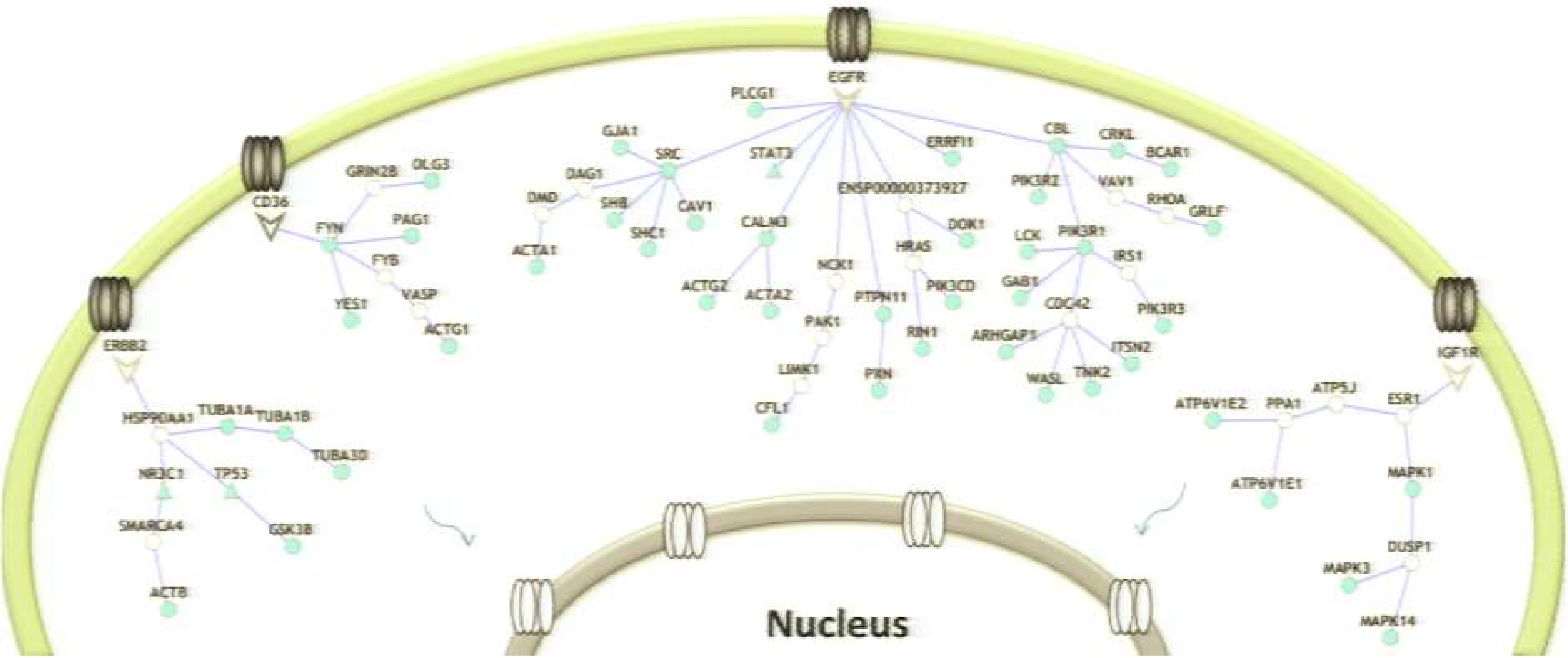
Prize Collecting Steiner Forest



Derived Forest: Yeast Pheromone Response Network



Derived Forest: Human Glioblastoma Data Set



Ex. 3: Extension to Patient-Specific Networks (Multi-PCSF) for Breast Cancer

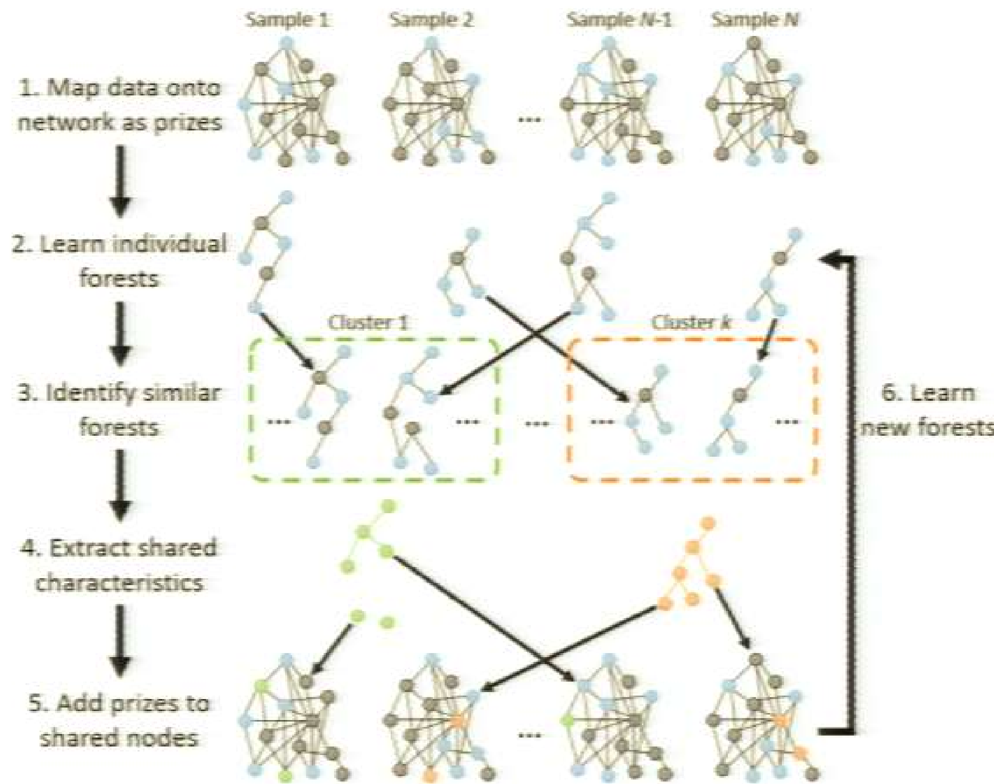
Ex. 3: Extension to Patient-Specific Networks (Multi-PCSF) for Breast Cancer

TCGA Breast Cancer Data:

Learn networks of individual breast cancer patients, extract shared features, & update algorithm for individual patients. Iterate.

Ex. 3: Extension to Patient-Specific Networks (Multi-PCSF) for Breast Cancer

TCGA Breast Cancer Data:
Learn networks of individual breast cancer patients, extract shared features, & update algorithm for individual patients. Iterate.

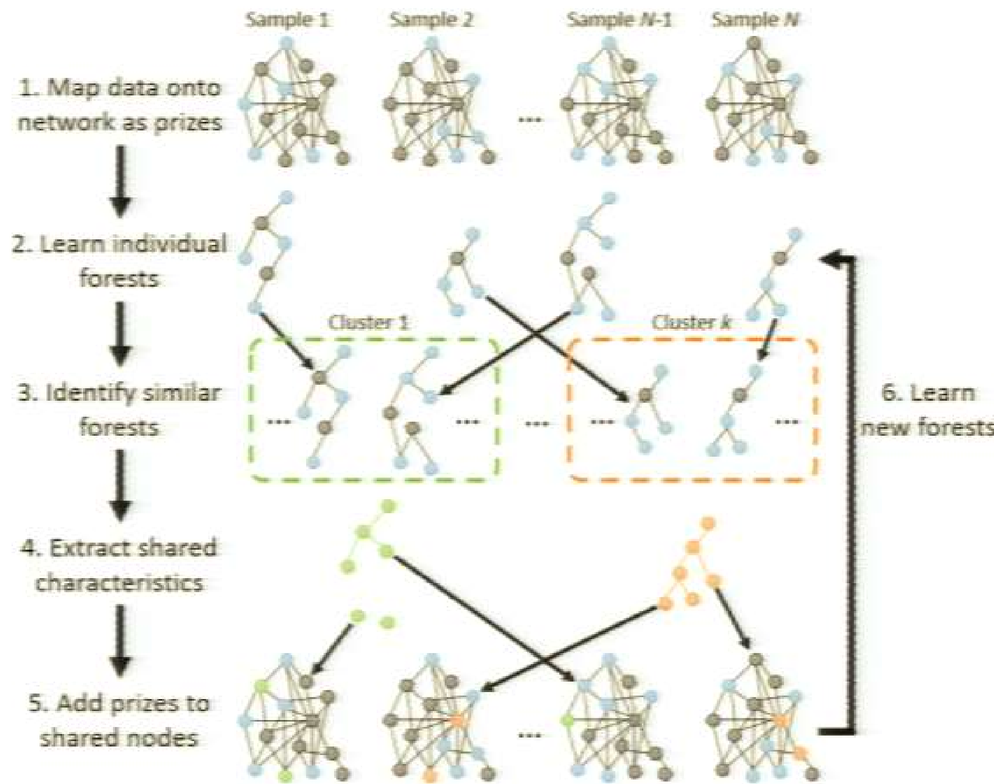


(Gitter, Braunstein, Pagnini, Baldassi, Borgs, Chayes, Zecchina, Fraenkel; PSB'14)

Ex. 3: Extension to Patient-Specific Networks (Multi-PCSF) for Breast Cancer

TCGA Breast Cancer Data:
Learn networks of individual breast cancer patients, extract shared features, & update algorithm for individual patients. Iterate.

→ Highly patient-specific networks, which have input from networks of other patients.

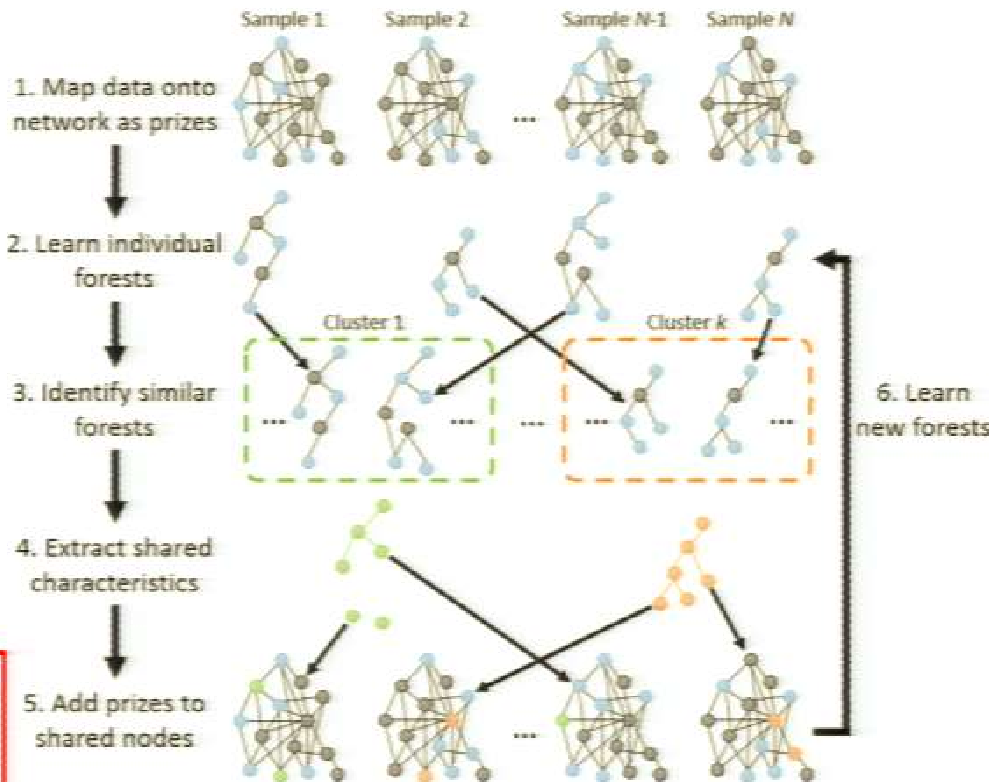


(Gitter, Braunstein, Pagnini, Baldassi, Borgs, Chayes, Zecchina, Fraenkel; PSB'14)

Ex. 3: Extension to Patient-Specific Networks (Multi-PCSF) for Breast Cancer

TCGA Breast Cancer Data:
Learn networks of individual breast cancer patients, extract shared features, & update algorithm for individual patients. Iterate.

→ Highly patient-specific networks, which have input from networks of other patients.



E.g., found subclass whose Steiner nodes implied they might be treatable with drugs for KIT-positive gastrointestinal tumors

(Gitter, Braunstein, Pagnini, Baldassi, Borgs, Chayes, Zecchina, Fraenkel; PSB'14)

Summary

- Graphical models give us succinct representations for capturing local dependencies among random variables, and (with the right representation) even some global dependencies, e.g., the prize-collecting Steiner tree

Summary

- Graphical models give us succinct representations for capturing local dependencies among random variables, and (with the right representation) even some global dependencies, e.g., the prize-collecting Steiner tree
- Belief propagation give us a way of approximiating marginals and modes of graphical models

Summary

- Graphical models give us succinct representations for capturing local dependencies among random variables, and (with the right representation) even some global dependencies, e.g., the prize-collecting Steiner tree
- Belief propagation give us a way of approximiating marginals and modes of graphical models
  - Rigorously can be proved to converge quickly to the correct solution in particular cases (e.g., b-matching when LP has only integral optima)

Summary

- Graphical models give us succinct representations for capturing local dependencies among random variables, and (with the right representation) even some global dependencies, e.g., the prize-collecting Steiner tree
- Belief propagation give us a way of approximiating marginals and modes of graphical models
  - Rigorously can be proved to converge quickly to the correct solution in particular cases (e.g., b-matching when LP has only integral optima)
  - In practice converges to near optimal solutions very rapidly on known benchmarks and new biological data sets

Summary

- Graphical models give us succinct representations for capturing local dependencies among random variables, and (with the right representation) even some global dependencies, e.g., the prize-collecting Steiner tree
- Belief propagation give us a way of approximiating marginals and modes of graphical models
  - Rigorously can be proved to converge quickly to the correct solution in particular cases (e.g., b-matching when LP has only integral optima)
  - In practice converges to near optimal solutions very rapidly on known benchmarks and new biological data sets
- There is biological evidence that BP algorithms do well in identifying regulatory pathways among proteins, and also identify "Steiner proteins", suggesting (patient-specific) drug targets for human disease

Thanks for your attention

NIPS Thanks Its Sponsors

amazon.com

Microsoft Research

Google

facebook

SKYTREE
THE MACHINE LEARNING COMPANY

TWO SIGMA

United Technologies
Research Center

D E Shaw & Co

xerox

YAHOO!
LABS

IBM
Research

TOYOTA

DRW TRADING GROUP

millionshort

criteo

PDT PARTNERS

Springer
Machine Learning Journal

Disney Research

WYSS INSTITUTE

HARVARD
INTELLIGENT PROBABILISTIC SYSTEMS

Message Passing Inference with Chemical Reaction Networks

Nils Napp
Wyss Institute for Biologically Inspired Engineering
Harvard University
Cambridge MA
nnapp@wyss.harvard.edu

Ryan P. Adams
School of Engineering and Applied Sciences
Harvard University
Cambridge MA
rpa@seas.harvard.edu

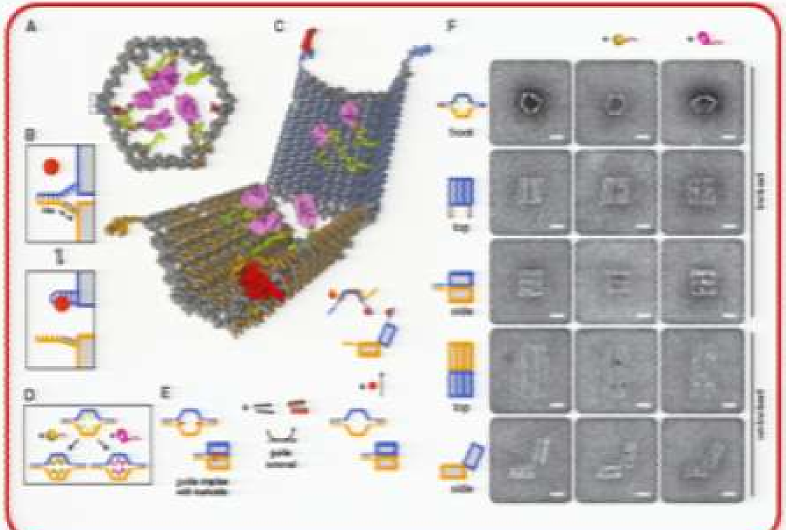
Neural Information Processing Systems
Lake Tahoe, 7 December 2013

ESET NOD32 Antivirus
An error occurred while downloading update files.

Self-Organizing Systems Research Group

HARVARD
School of Engineering and Applied Sciences

Engineering with Biological Parts



Douglas, Bachelet, Church. Science 2012

- Unidentified Parts
- "Unpredictable" Circuits Behavior
- High Complexity Circuits
- Incompatible Parts
- Variability in Behavior

Kwok, Nature 2010

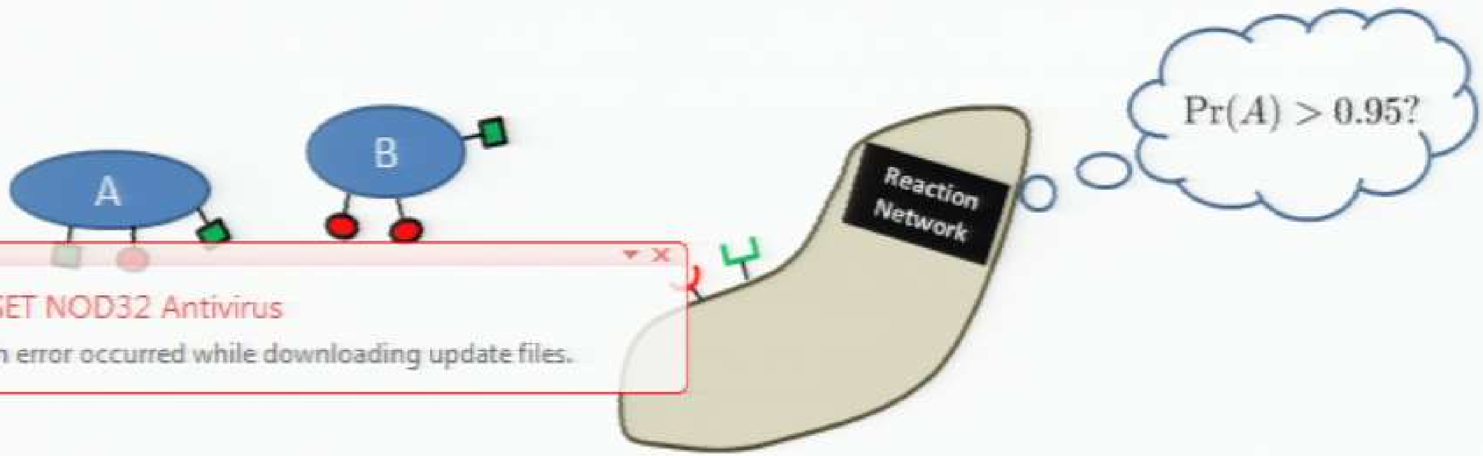
ESET NOD32 Antivirus

An error occurred while downloading update files.

Contribution

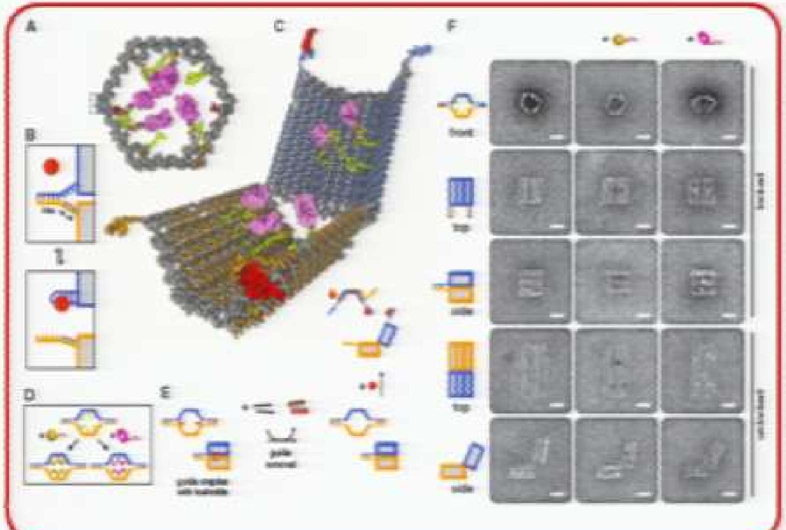
Implement inference on a molecular level

- Enable estimation of latent variables
- Take into account complex dependencies
- Extract information from noisy sensors



ESET NOD32 Antivirus
An error occurred while downloading update files.

Engineering with Biological Parts



Douglas, Bachelet, Church. Science 2012

FIVE HARD TRUTHS FOR SYNTHETIC BIOLOGY

Can engineering approaches tame the complexity of living systems? **Roberta Kwok** explores five challenges for the field and how they might be resolved.

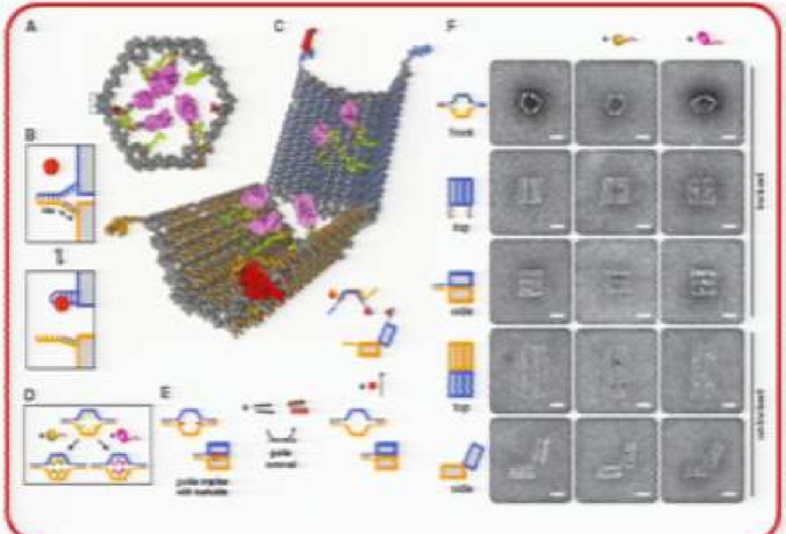
- Unidentified Parts
- "Unpredictable" Circuits Behavior
- High Complexity Circuits
- Incompatible Parts
- Variability in Behavior

Kwok, Nature 2010

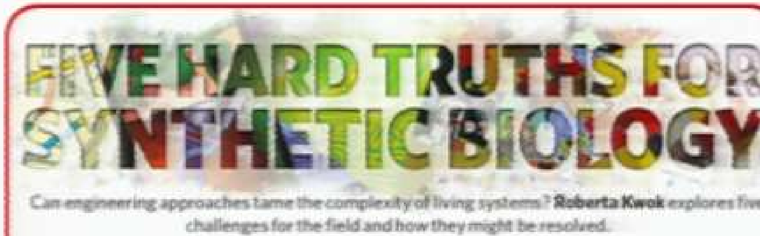
ML techniques can address these problems!

ESET NOD32 Antivirus
An error occurred while downloading update files.

Engineering with Biological Parts



Douglas, Bachelet, Church. Science 2012



- Unidentified Parts
- "Unpredictable" Circuits Behavior
- High Complexity Circuits
- Incompatible Parts
- Variability in Behavior

Kwok, Nature 2010

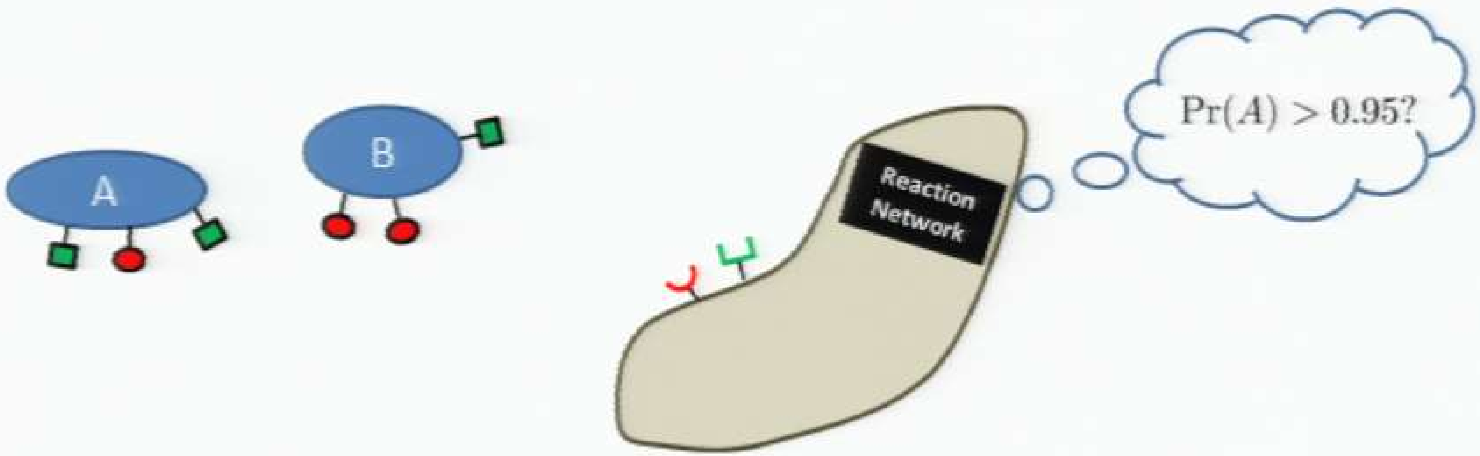
↑

ML techniques can address these problems!

Contribution

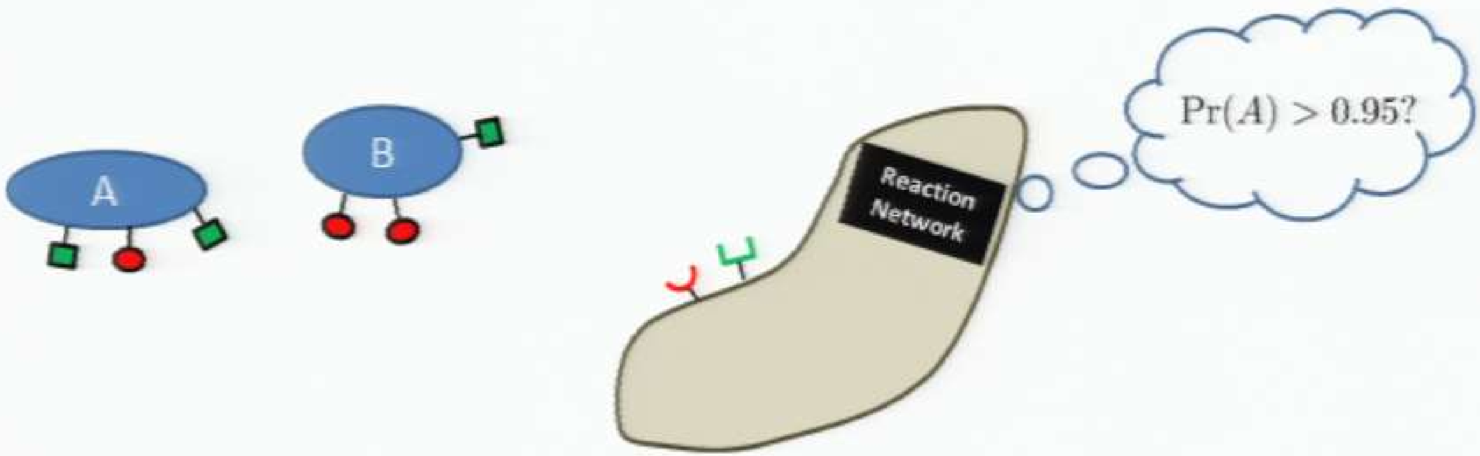
Implement inference on a molecular level

- Enable estimation of latent variables
- Take into account complex dependencies
- Extract information from noisy sensors



Contribution

Chemical reaction networks are the *assembly language* of small scale devices.



Chemical Reaction Networks

Set of species: $Z = \{\mathsf{Z}_1, \mathsf{Z}_2, ..., \mathsf{Z}_M\}$

$$r_1\mathsf{Z}_1 + ... + r_M\mathsf{Z}_M \xrightarrow{k_q} p_1\mathsf{Z}_1 + ... + p_M\mathsf{Z}_M$$

Reaction: $R_q = (\mathbf{r}^q, k_q, \mathbf{p}^q)$

Reaction Network: $\mathcal{R} = \{R_1, ..., R_Q\}$

Chemical Reaction Networks

Set of species: $Z = \{\mathsf{Z}_1, \mathsf{Z}_2, ..., \mathsf{Z}_M\}$

$$r_1\mathsf{Z}_1 + ... + r_M\mathsf{Z}_M \xrightarrow{k_q} p_1\mathsf{Z}_1 + ... + p_M\mathsf{Z}_M$$

Reaction: $R_q = (\mathbf{r}^q, k_q, \mathbf{p}^q)$

Reaction Network: $\mathcal{R} = \{R_1, ..., R_Q\}$

Example:

$$Z = \{\mathsf{A}, \mathsf{B}, \mathsf{C}\}$$

$$\mathsf{A} + 2\mathsf{B} \xrightarrow{k_1} \mathsf{A} + \mathsf{C}$$

$$\mathsf{C} \xrightarrow{k_2} \emptyset$$

$$\mathbf{r}^1 = (1, 2, 0)^T \quad \mathbf{r}^2 = (0, 0, 1)^T$$

$$\mathbf{p}^1 = (1, 0, 1)^T \quad \mathbf{p}^2 = (0, 0, 0)^T$$

Mass Action Kinetics

Concentration: $[\mathsf{Z}_m]$

The Law of Mass Action: $$\frac{d[\mathsf{Z}_m]}{dt} = \sum_{q=1}^{Q} k_q \prod_{m'=1}^{M} [\mathsf{Z}'_m]^{\mathbf{r}^q_{m'}} (\mathbf{p}^q_m - \mathbf{r}^q_m)$$

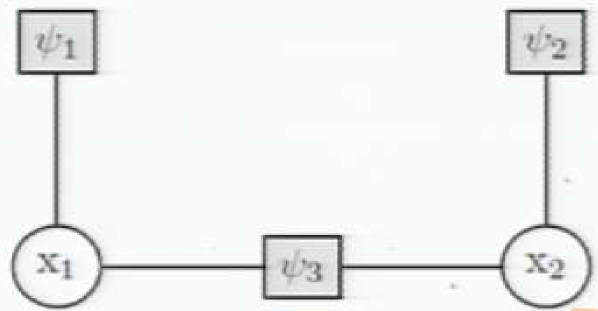
Given a Chemical Reaction Network the Law of Mass Action gives a set of non-linear ODEs that describe the evolution of concentrations.

$$\mathsf{A} + 2\mathsf{B} \xrightarrow{k_1} \mathsf{A} + \mathsf{C}$$

$$\mathsf{C} \xrightarrow{k_2} \emptyset$$

$$\frac{d[\mathsf{C}]}{dt} = k_1[\mathsf{A}][\mathsf{B}]^2 - k_2[\mathsf{C}]$$

Factor Graphs



Bipartite graph between *factor* nodes and *variable* nodes

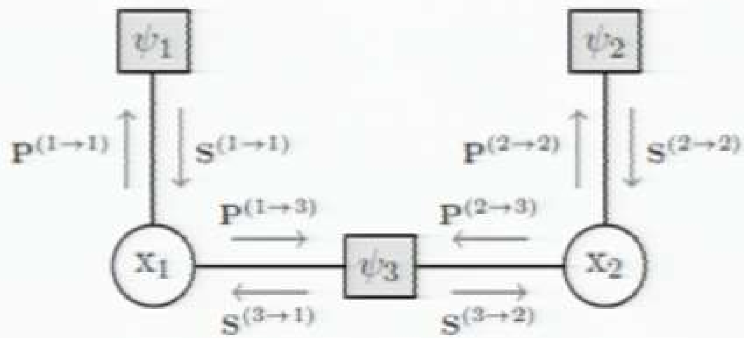
Describes how join probability of random variables represented by variable nodes factors:

$$\Pr(\mathbf{x}) = \Pr(X_1, X_2, ..., X_N) = \frac{1}{\alpha} \prod_{j=1}^{J} \psi_j(\mathbf{x^j})$$

Subset of variables connected factor j

Non-negative scalar function

Inference on Factor Graphs



- Compute marginal probabilities $\Pr(X_i)$ taking into account dependencies.
- Can be done by "message passing" two different types of messages.

Sum messages (factor to variable) : $S_k^{(j\to n)} = \sum_{\mathbf{k}_n^j = k} \psi_j(\mathbf{x}^j = \mathbf{k}^j) \prod_{n' \in \mathrm{ne}(j)\setminus n} P_{\mathbf{k}_{n'}^j}^{(n'\to j)}$

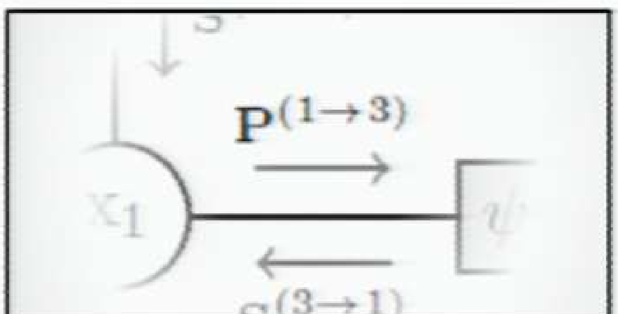
Product message (variable to factor) : $P_k^{(n\to j)} = \prod_{j' \in \mathrm{ne}(n)\setminus j} S_k^{(j'\to n)}$

Marginals at variable nodes given by: $\Pr(\mathbf{x}_n = k) = \prod_{j \in \mathrm{ne}(n)} S_k^{(j\to n)}$

Chemical Representation: Belief Species

Each message is represented by a set of *belief species*.

- If the message is k-entry vector then the set of belief species has $k+1$ species.
- The extra species represents unassigned probability.
- Other messages catalyze assignment of unassigned probability mass, but all assignments say with the set.



Message in Graph

$$\mathbf{P}^{(1\to3)} = \begin{pmatrix} \mathbf{P}_1^{(1\to3)} \\ \mathbf{P}_2^{(1\to3)} \end{pmatrix}$$

Probability Vector

$$\mathsf{P}_0^{(1\to3)}$$
$$\mathsf{P}_1^{(1\to3)}$$
$$\mathsf{P}_2^{(1\to3)}$$

Chemical Representation

Product Messages ($\mathrm{P}_k^{(n\to j)} = \prod_{j'\in \mathrm{ne}(n)\setminus j} \mathrm{S}_k^{(j'\to n)}$)

Produce messages can be implemented as

$$\mathsf{P}_0^{(n\to j)} + \sum_{j'\in \mathrm{ne}(j)\setminus n} \mathsf{S}_k^{(j'\to n)} \xrightarrow{\kappa_{\mathrm{prod}}} \mathsf{P}_k^{(n\to j)} + \sum_{j'\in \mathrm{ne}(j)\setminus n} \mathsf{S}_k^{(j'\to n)}$$

At steady state:

$$\frac{\kappa_r}{\kappa_{\mathrm{prod}}[\mathsf{P}_0^{(n\to j)}]}[\mathsf{P}_k^{(n\to j)}] = \prod_{j'\in \mathrm{ne}(j)\setminus n} [\mathsf{S}_k^{(j'\to n)}].$$

Product Messages ($\mathrm{P}_k^{(n\to j)} = \prod_{j'\in \mathrm{ne}(n)\setminus j} \mathrm{S}_k^{(j'\to n)}$)

Produce messages can be implemented as

$$\mathsf{P}_0^{(n\to j)} + \sum_{j'\in \mathrm{ne}(j)\setminus n} \mathsf{S}_k^{(j'\to n)} \xrightarrow{\kappa_{\mathrm{prod}}} \mathsf{P}_k^{(n\to j)} + \sum_{j'\in \mathrm{ne}(j)\setminus n} \mathsf{S}_k^{(j'\to n)}$$

At steady state:

$$\frac{\kappa_r}{\kappa_{\mathrm{prod}}[\mathsf{P}_0^{(n\to j)}]}[\mathsf{P}_k^{(n\to j)}] = \prod_{j'\in \mathrm{ne}(j)\setminus n} [\mathsf{S}_k^{(j'\to n)}].$$

Ratios correspond to sum messages.
When $[\mathrm{P}_0]$ is small they approximate probability directly.

Sum Messages ($S_k^{(j\to n)} = \sum_{\mathbf{k}_n^j = k} \psi_j(\mathbf{x}^j = \mathbf{k}^j) \prod_{n' \in \mathrm{ne}(j) \setminus n} P_{\mathbf{k}_{n'}^j}^{(n'\to j)}$)

Sum messages can be implemented as:

$$S_0^{(j\to n)} + \sum_{n' \in \mathrm{ne}(j) \setminus n} P_{\mathbf{k}_{n'}^j}^{(n'\to j)} \xrightarrow{\psi_j(\mathbf{x}^j = \mathbf{k}^j)} S_k^{(j\to n)} + \sum_{n' \in \mathrm{ne}(j) \setminus n} P_{\mathbf{k}_{n'}^j}^{(n'\to j)}$$

At steady state:

$$\frac{\kappa_r}{[S_0^{(j\to n)}]}[S_k^{(j\to n)}] = \sum_{\mathbf{k}_n^j = k} \psi_j(\mathbf{x}^j = \mathbf{k}^j) \prod_{n' \in \mathrm{ne}(j) \setminus n} [P_{\mathbf{k}_{n'}^j}^{(n'\to j)}]$$

Sum Messages ($S_k^{(j\to n)} = \sum_{\mathbf{k}_n^j=k} \psi_j(\mathbf{x}^j = \mathbf{k}^j) \prod_{n'\in \mathrm{ne}(j)\setminus n} \mathsf{P}_{\mathbf{k}_{n'}^j}^{(n'\to j)}$)

Sum messages can be implemented as:

$$\mathsf{S}_0^{(j\to n)} + \sum_{n'\in \mathrm{ne}(j)\setminus n} \mathsf{P}_{\mathbf{k}_{n'}^j}^{(n'\to j)} \xrightarrow{\psi_j(\mathbf{x}^j=\mathbf{k}^j)} \mathsf{S}_k^{(j\to n)} + \sum_{n'\in \mathrm{ne}(j)\setminus n} \mathsf{P}_{\mathbf{k}_{n'}^j}^{(n'\to j)}$$

At steady state:

$$\frac{\kappa_r}{[\mathsf{S}_0^{(j\to n)}]}[\mathsf{S}_k^{(j\to n)}] = \sum_{\mathbf{k}_n^j=k} \psi_j(\mathbf{x}^j = \mathbf{k}^j) \prod_{n'\in \mathrm{ne}(j)\setminus n} [\mathsf{P}_{\mathbf{k}_{n'}^j}^{(n'\to j)}]$$

Ratios correspond to sum messages.
When $[\mathsf{S}_0]$ is small they approximate probability directly.

Recycling Reactions

Recycle probability within sets of belief species.

- Messages processed continually and the system adapts to new information.
- Recycling rate determines turnover and speed.

$$\mathsf{P}_k^{(n\to j)} \xrightarrow{k_{\circlearrowleft}} \mathsf{P}_0^{(n\to j)}$$

$$\mathsf{S}_k^{(j\to n)} \xrightarrow{k_{\circlearrowleft}} \mathsf{S}_0^{(j\to n)}$$

$$\mathsf{Pr}_k^{n} \xrightarrow{k_{\circlearrowleft}} \mathsf{Pr}_0^{n}$$

Generic

$$\mathsf{P}_1^{(1\to 3)} \xrightarrow{k_{\circlearrowleft}} \mathsf{P}_0^{(1\to 3)}$$

$$\mathsf{P}_2^{(1\to 3)} \xrightarrow{k_{\circlearrowleft}} \mathsf{P}_0^{(1\to 3)}$$

Example for $\mathbf{P}^{(1\to 3)}$

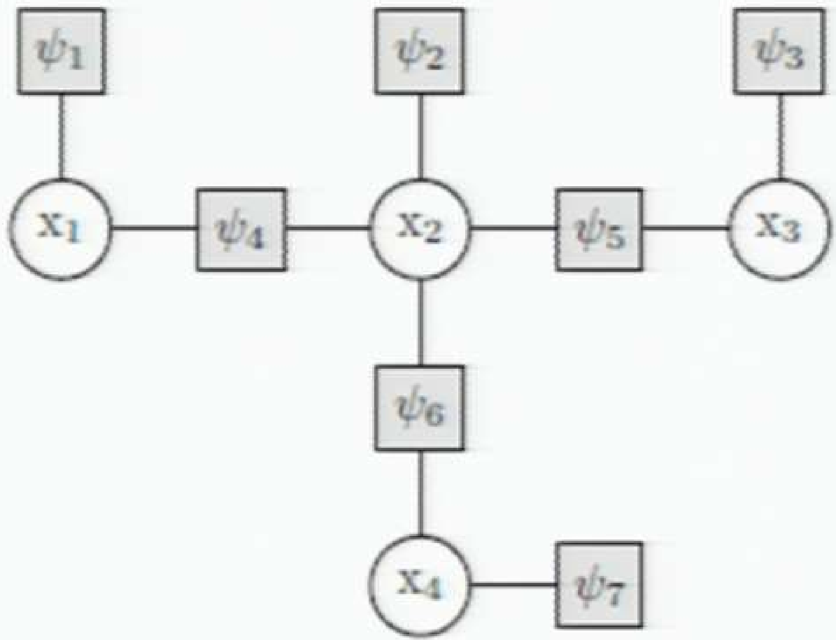
Recycling Assignment

$$\mathsf{P}_1^{(1\to 3)} \rightleftharpoons \mathsf{P}_0^{(1\to 3)} \rightleftharpoons \mathsf{P}_2^{(1\to 3)}$$

Assignment Recycling

Reaction Structure in Belief set

Example



$\psi_1(1)$	$\psi_1(2)$
1	0.1

$\psi'_1(1)$	$\psi'_1(2)$
0.1	1

$\psi_2(1)$	$\psi_2(2)$
1	0.1

$\psi_3(1)$	$\psi_3(2)$	$\psi_3(3)$
2	1	1

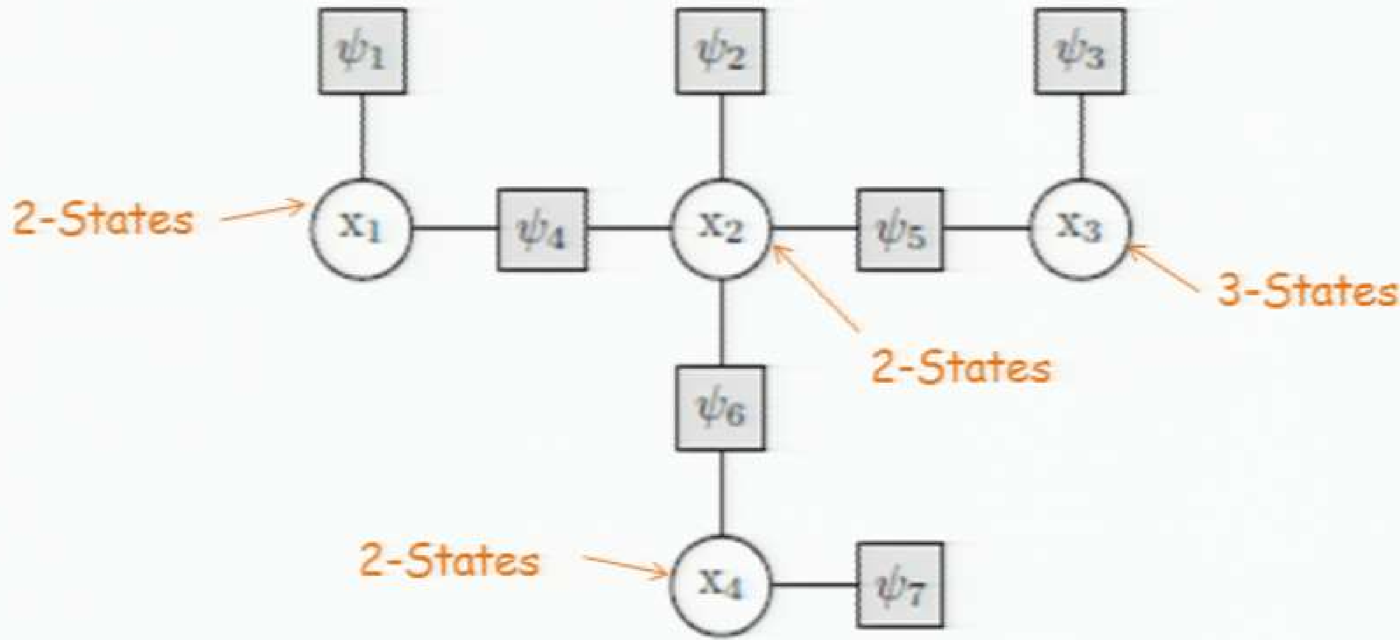
$\psi_7(1)$	$\psi_7(2)$
1	1

	$\psi_4(\cdot,1)$	$\psi_4(\cdot,2)$
$\psi_4(1,\cdot)$	1	0.1
$\psi_4(2,\cdot)$	0.1	3

	$\psi_5(\cdot,1)$	$\psi_5(\cdot,2)$	$\psi_5(\cdot,3)$
$\psi_5(1,\cdot)$	0.1	2	0.1
$\psi_5(2,\cdot)$	3	0.1	1

	$\psi_6(\cdot,1)$	$\psi_6(\cdot,2)$
$\psi_6(1,\cdot)$	0.1	0.1
$\psi_6(2,\cdot)$	1	0.1

Example



$\psi_1(1)$	$\psi_1(2)$	$\psi'_1(1)$	$\psi'_1(2)$
1	0.1	0.1	1

$\psi_2(1)$	$\psi_2(2)$
1	0.1

$\psi_3(1)$	$\psi_3(2)$	$\psi_3(3)$
2	1	1

$\psi_7(1)$	$\psi_7(2)$
1	1

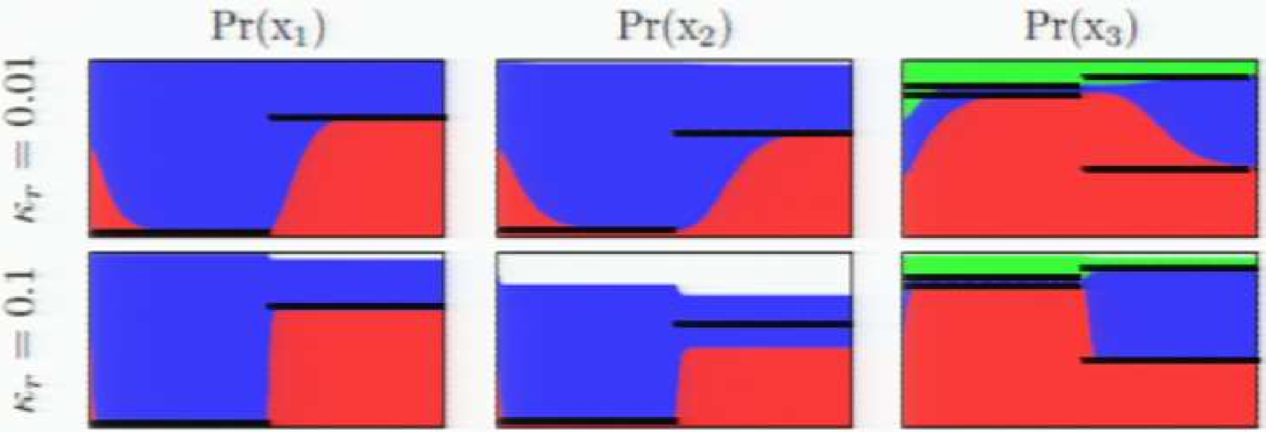
	$\psi_4(\cdot,1)$	$\psi_4(\cdot,2)$
$\psi_4(1,\cdot)$	1	0.1
$\psi_4(2,\cdot)$	0.1	3

	$\psi_5(\cdot,1)$	$\psi_5(\cdot,2)$	$\psi_5(\cdot,3)$
$\psi_5(1,\cdot)$	0.1	2	0.1
$\psi_5(2,\cdot)$	3	0.1	1

	$\psi_6(\cdot,1)$	$\psi_6(\cdot,2)$
$\psi_6(1,\cdot)$	0.1	0.1
$\psi_6(2,\cdot)$	1	0.1

Example

$\Pr(X_i = 1)$ ■ $\Pr(X_i = 2)$ ■ $\Pr(X_i = 3)$ ■ $\Pr(X_i = ?)$ □



	Pr(x1)		Pr(x2)		Pr(x3)			Pr(x4)	
exact	0.692	0.308	0.598	0.402	0.392	0.526	0.083	0.664	0.336
slow	0.690	0.306	0.583	0.393	0.394	0.520	0.083	0.665	0.333
fast	0.661	0.294	0.449	0.302	0.379	0.508	0.080	0.646	0.326

Summary

- Compile Belief Propagation on arbitrary discrete valued factor graphs into sets of chemical reactions.
- Probabilities and messages are represented sets of belief species which are conserved.
- Message species catalyze each other.
- Works because the system dynamics have the same form as the computation we would like to do.

Law of Mass Action

$$\frac{d[Z_m]}{dt} = \sum_{q=1}^{Q} k_q \prod_{m'=1}^{M} [Z_{m'}]^{\mathbf{r}^q_{m'}} (\mathbf{p}^q_m - \mathbf{r}^q_m)$$

$$S_k^{(j\to n)} = \sum_{\mathbf{k}^j_n = k} \psi_j(\mathbf{x}^j = \mathbf{k}^j) \prod_{n'\in \mathrm{ne}(j)\setminus n} P^{(n'\to j)}_{\mathbf{k}^j_{n'}}$$

Sum message in Belief Propagation

$$P_k^{(n\to j)} = \prod_{j'\in \mathrm{ne}(n)\setminus j} S_k^{(j'\to n)}$$

Product message in Belief Propagation

Where to go next

- Apply to specific bio-sensor models
- Simplify machinery for binary RVs
- Look for inference network motives
- Collaborate with sys-bio community help solve noise and uncertainty problems in current systems, e.g. parameter learning

Thanks!

Ryan P. Adams (Harvard)

Radhika Nagpal (Harvard)

David Soloveichick (USF)

HARVARD INTELLIGENT PROBABILISTIC SYSTEMS

Self-Organizing Systems Research Group

Please visit us at poster S68 this evening.

WYSS INSTITUTE

HARVARD School of Engineering and Applied Sciences

NIPS Thanks Its Sponsors

amazon.com

Microsoft Research

Google

facebook

SKYTREE
THE MACHINE LEARNING COMPANY

TWO SIGMA

United Technologies Research Center

D E Shaw & Co

xerox

YAHOO! LABS

IBM Research

TOYOTA

DRW TRADING GROUP

millionshort

criteo

PDT PARTNERS

Springer
Machine Learning Journal

Disney Research

Where to go next

- Apply to specific bio-sensor models
- Simplify machinery for binary RVs
- Look for inference network motives
- Collaborate with sys-bio community help solve noise and uncertainty problems in current systems, e.g. parameter learning

NIPS Thanks Its Sponsors

amazon.com

Microsoft Research

Google

facebook

SKYTREE
THE MACHINE LEARNING COMPANY

TWO SIGMA

United Technologies Research Center

D E Shaw & Co

IBM Research

xerox

YAHOO! LABS

TOYOTA

DRW TRADING GROUP

millionshort

criteo

PDT PARTNERS

Springer
Machine Learning Journal

Disney Research

Information-theoretic Lower Bounds for Distributed Statistical Estimation with Communication Constraints

Yuchen Zhang   John Duchi
Michael I. Jordan   Martin J. Wainwright

University of California, Berkeley

NIPS 2013

A Modern Data Center

- Holds 10,000+ servers.
- Data storage and data processing highly distributed.
- Communication cost $\gg$ computation cost.

A Fundamental Trade-off

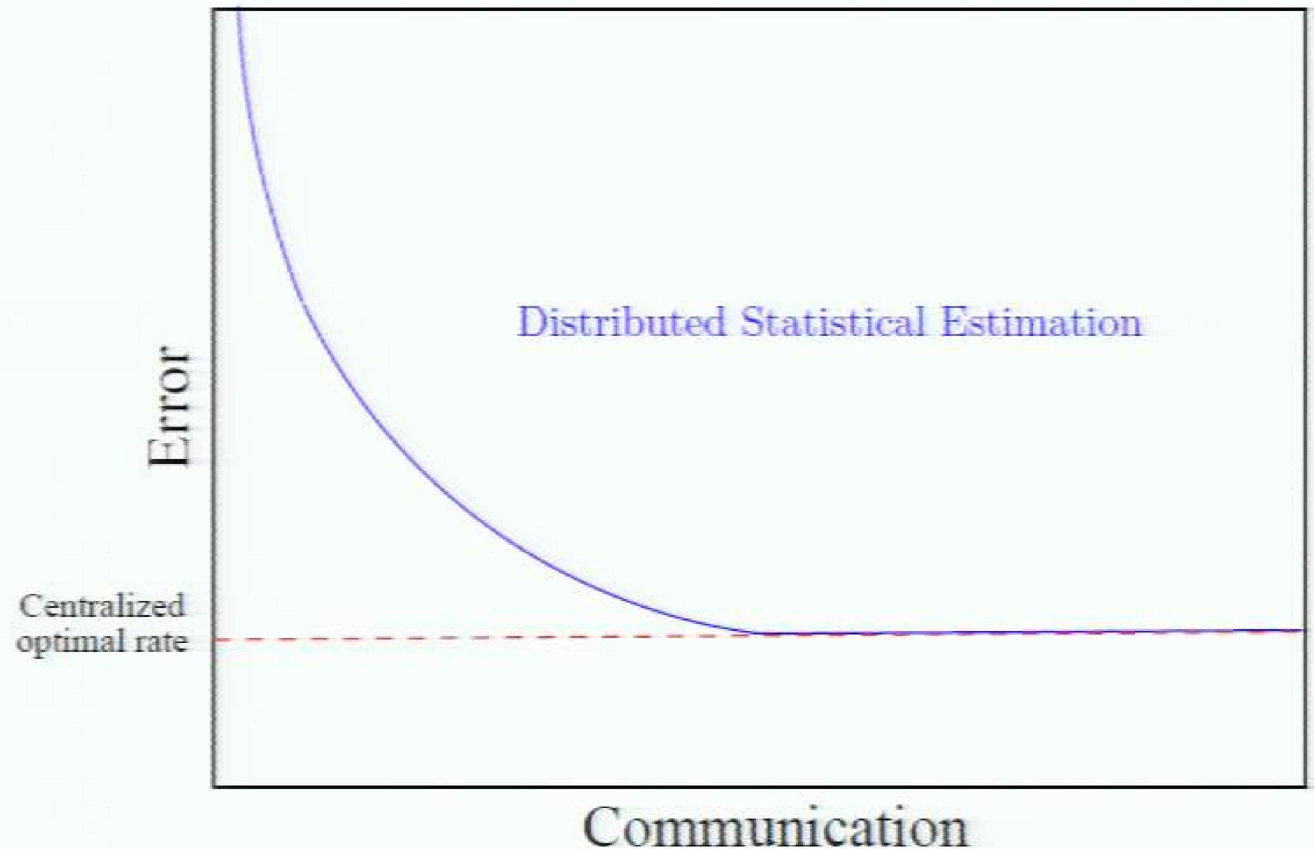
When learning from distributed data,

Target 1: maximize statistical accuracy.

Target 2: minimize communication cost.

Main Result

Communication-Accuracy trade-off:

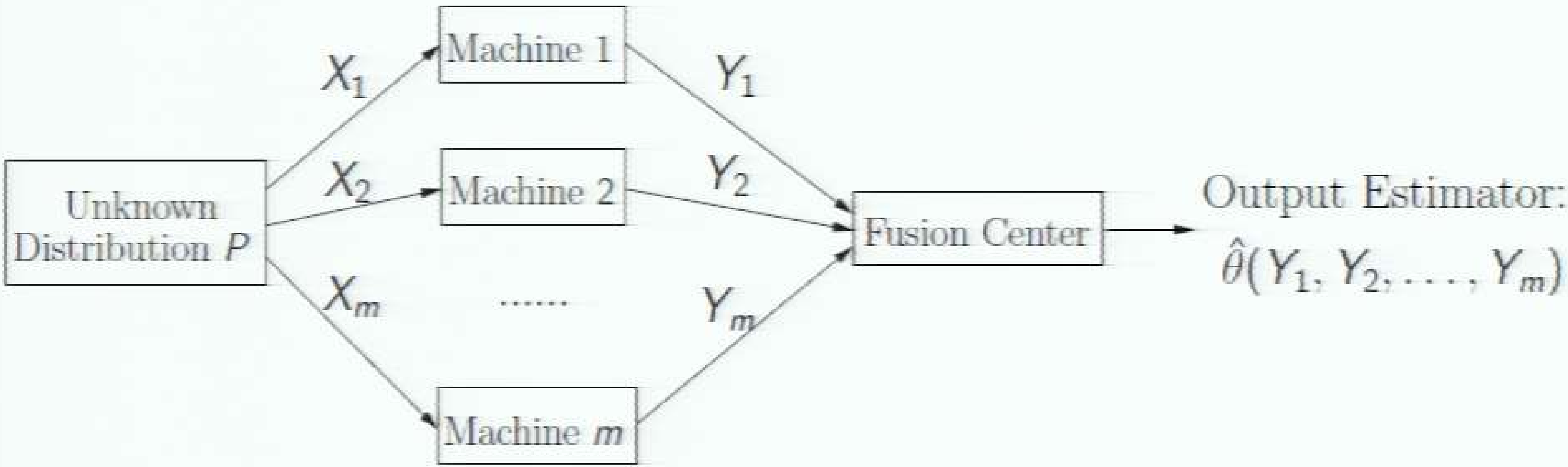


Statistical Estimation

Given: i.i.d. data drawn from unknown distribution P
Goal: estimate a parameter $\theta(P)$.

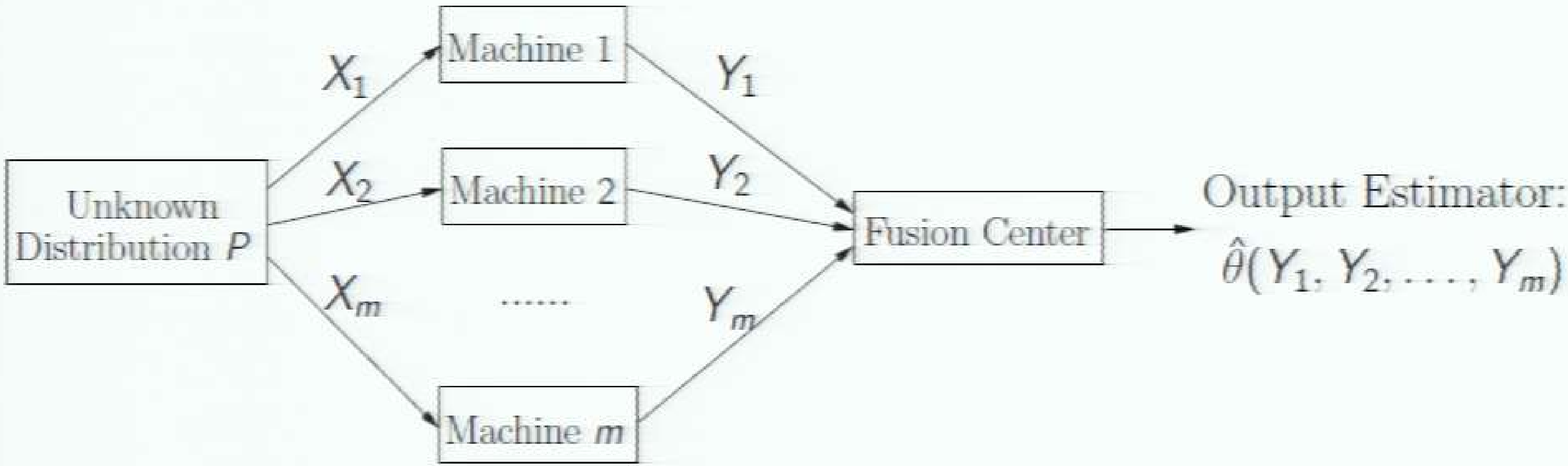
Distributed Statistical Estimation

- Data is stored on m separate machines.
- Each machine generates a message based on its local data.
- Output a message-based estimator.



Distributed Statistical Estimation

- Data is stored on m separate machines.
- Each machine generates a message based on its local data.
- Output a message-based estimator.



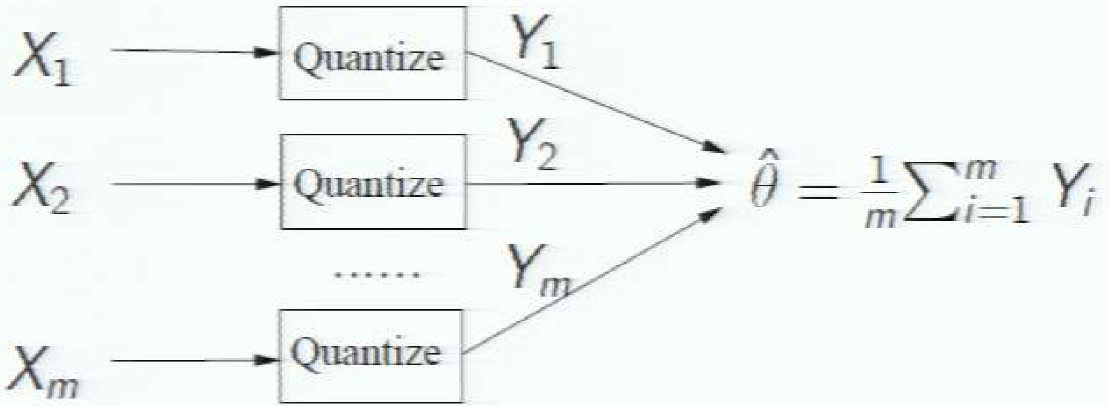
- Statistical accuracy: $\mathbb{E}[\|\hat{\theta} - \theta\|_2^2]$
- Communication cost: $\sum_{i=1}^{m} \text{Length}(Y_i)$

Example: Gaussian Location Model

m machines, each machine gets $X_i \sim \mathcal{N}(\theta, 1)$. Want to estimate θ.

Example: Gaussian Location Model

m machines, each machine gets $X_i \sim \mathcal{N}(\theta, 1)$. Want to estimate θ.



Analysis:

- Estimation error: $\mathbb{E}[(\hat{\theta} - \theta)^2] \simeq \frac{1}{m}$. (optimal rate)
- Communication cost $\simeq m$.

Question: Is there a better estimator?

Minimum Possible Communication

Answer is: NO.

Minimum Possible Communication

Answer is: NO.

Theorem

If each of m machines gets one i.i.d. sample from $N(\theta, 1)$, then any optimal estimator of θ must communicate $\widetilde{\Omega}(m)$ bits.

Gaussian Location Model ($n \geq 1$, $d \geq 1$)

Given: m machines, each machine gets n i.i.d. samples from $\mathcal{N}(\theta, \sigma^2 I_{d\times d})$.
Goal: find the Gaussian mean $\theta \in \mathbb{R}^d$.

Gaussian Location Model ($n \geq 1$, $d \geq 1$)

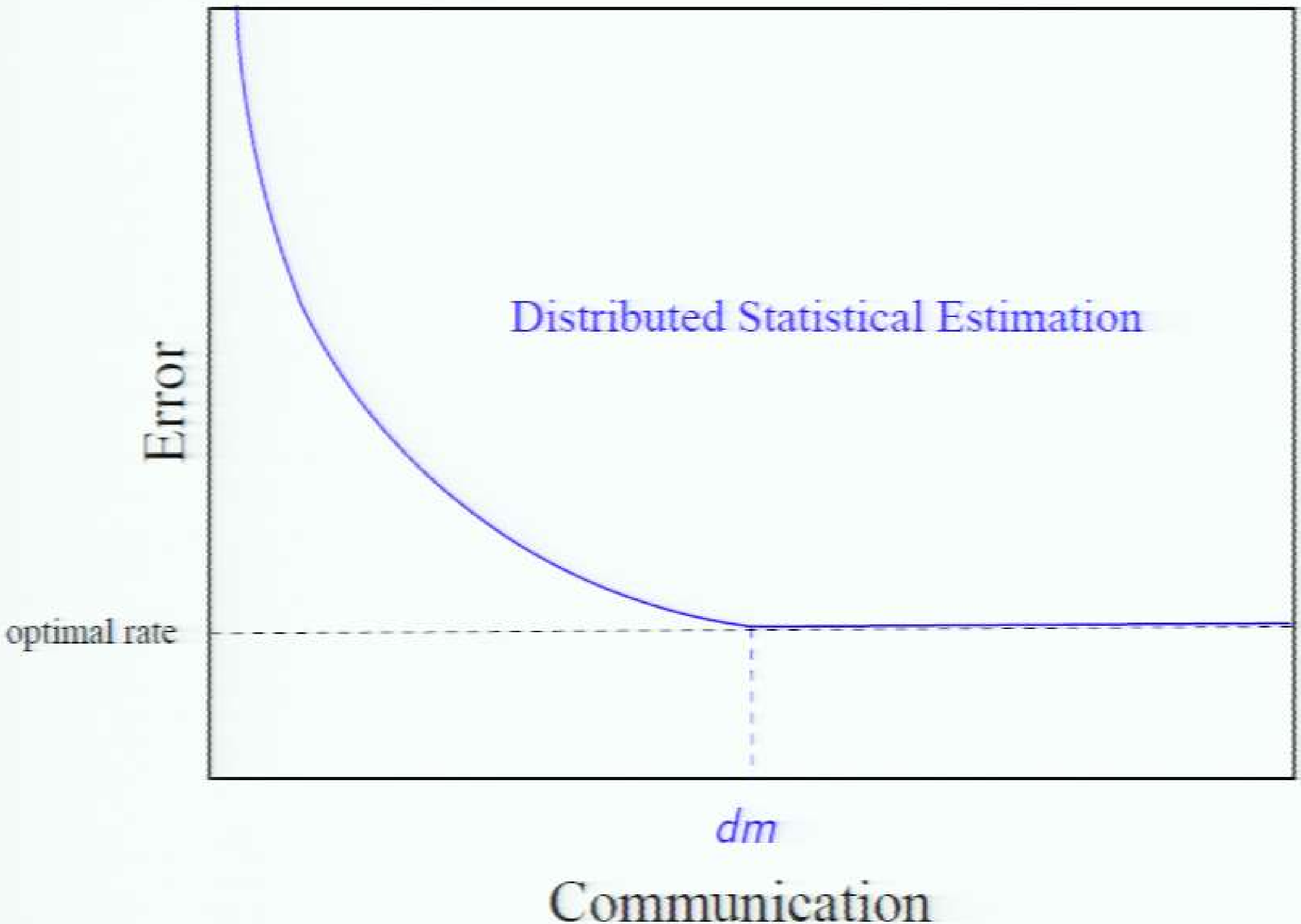
Given: m machines, each machine gets n i.i.d. samples from $\mathcal{N}(\theta, \sigma^2 I_{d\times d})$.
Goal: find the Gaussian mean $\theta \in \mathbb{R}^d$.

Theorem

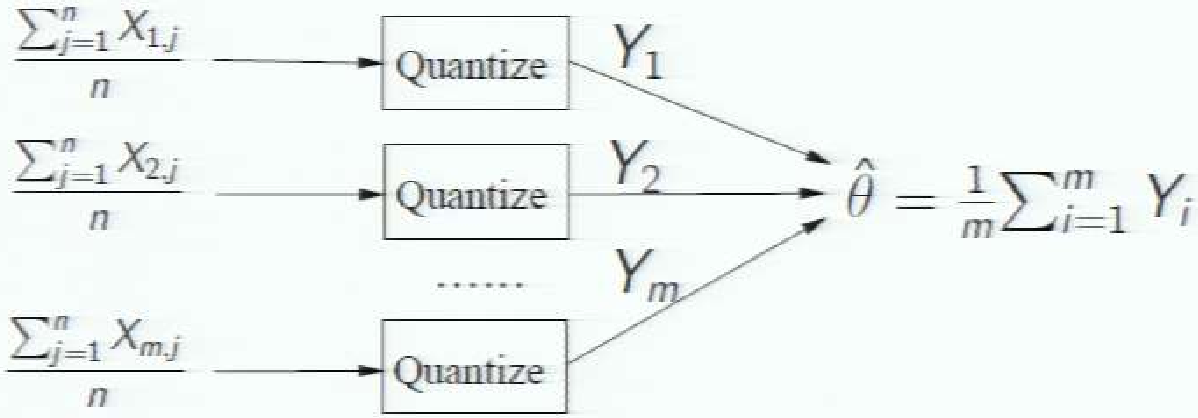
If an estimator is allowed to communicate B bits, then

$$\max_{\theta\in[-1,1]^d} \mathbb{E}[(\hat{\theta} - \theta)^2] \geq C \cdot \frac{d}{mn} \cdot \max\left\{1, \frac{dm}{B \log m}\right\}$$

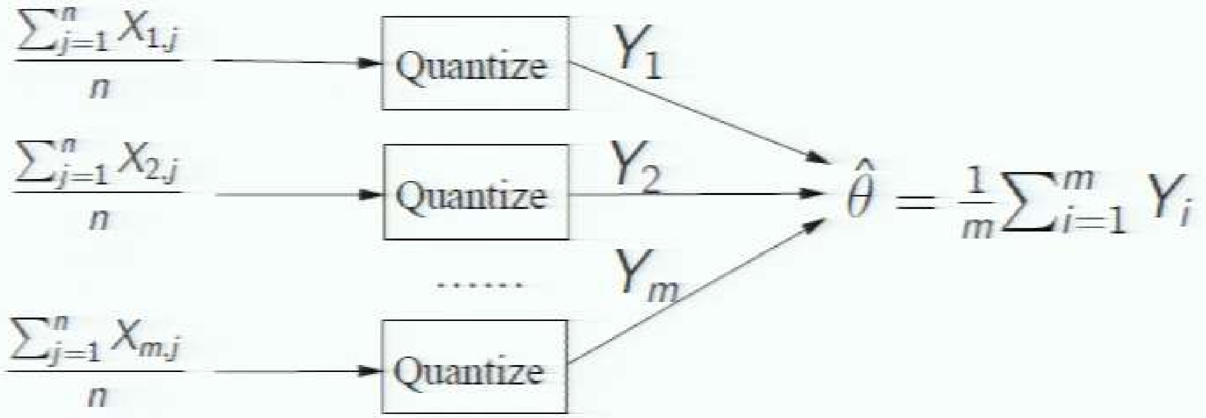
Lower Bound Curve



Achievability of Lower Bound



Achievability of Lower Bound



Analysis:

- Estimation error: $\mathbb{E}[\|\hat{\theta} - \theta\|_2^2] = \mathcal{O}\left(\frac{d}{mn}\right)$. (optimal rate)
- Communication cost: $\mathcal{O}(dm \log(mn))$.

Conclusion: $\widetilde{\Theta}(dm)$ bits of communication are necessary and sufficient.

Consequence for Regression Problems

Linear Regression

Given: m machines, each machine gets n i.i.d. inputs (x_i, z_i) satisfying

$$x_i \in \mathbb{R}^d \quad \text{and} \quad z_i = \theta^T x_i + w_i$$

where $w_i \sim \mathcal{N}(0, \sigma^2)$.
Goal: find the regression coefficient $\theta \in \mathbb{R}^d$.

Probit Regression

Given: m machines, each machine gets n i.i.d. inputs (x_i, y_i) satisfying

$$x_i \in \mathbb{R}^d \quad \text{and} \quad z_i = \begin{cases} 1 & \text{with probability } \Phi(\theta^T x_i) \\ 0 & \text{with probability } 1 - \Phi(\theta^T x_i) \end{cases}$$

where Φ is the CDF of standard normal distribution.
Goal: find the regression coefficient $\theta \in \mathbb{R}^d$.

Consequence for Regression Problems

Lower Bound

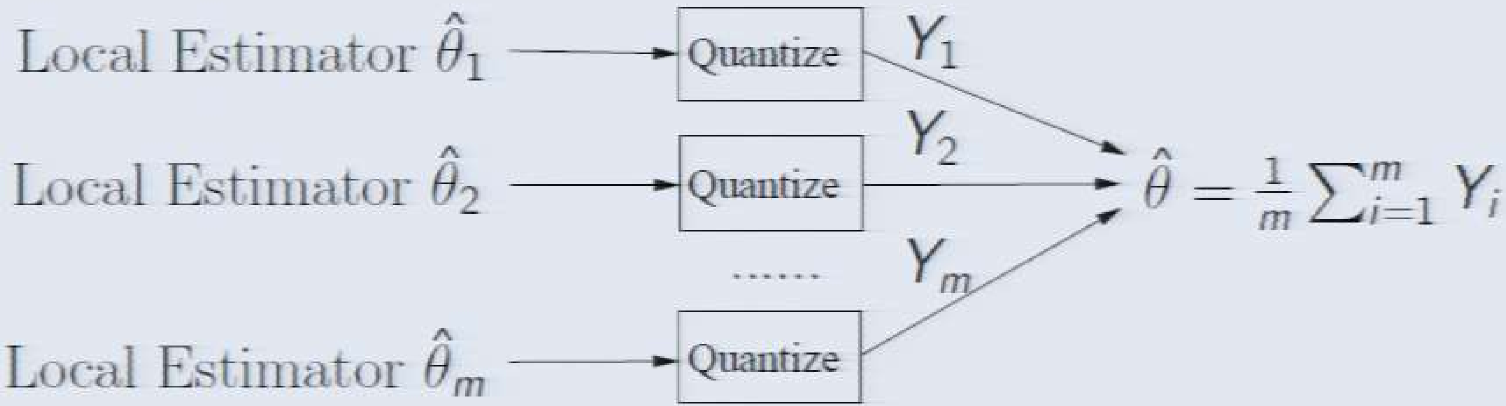
For linear regression and probit regression, any optimal estimator of θ must communicates $\Omega(dm/\log m)$ bits.

Consequence for Regression Problems

Lower Bound

For linear regression and probit regression, any optimal estimator of θ must communicates $\Omega(dm/\log m)$ bits.

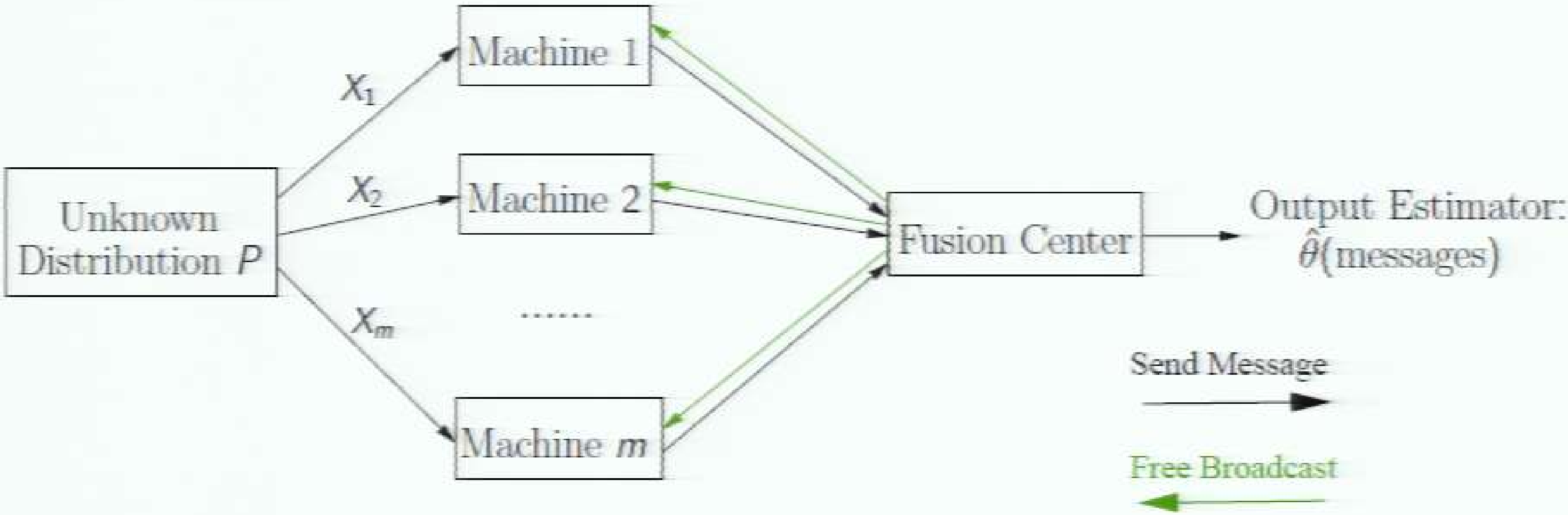
Upper Bound (Z, Duchi, Wainwright, NIPS'12)



- Estimation error: $\mathbb{E}[\|\hat{\theta} - \theta\|_2^2] = \mathcal{O}\left(\frac{d}{mn}\right)$. (optimal rate)
- Communication cost: $\mathcal{O}(dm \log(mn))$.

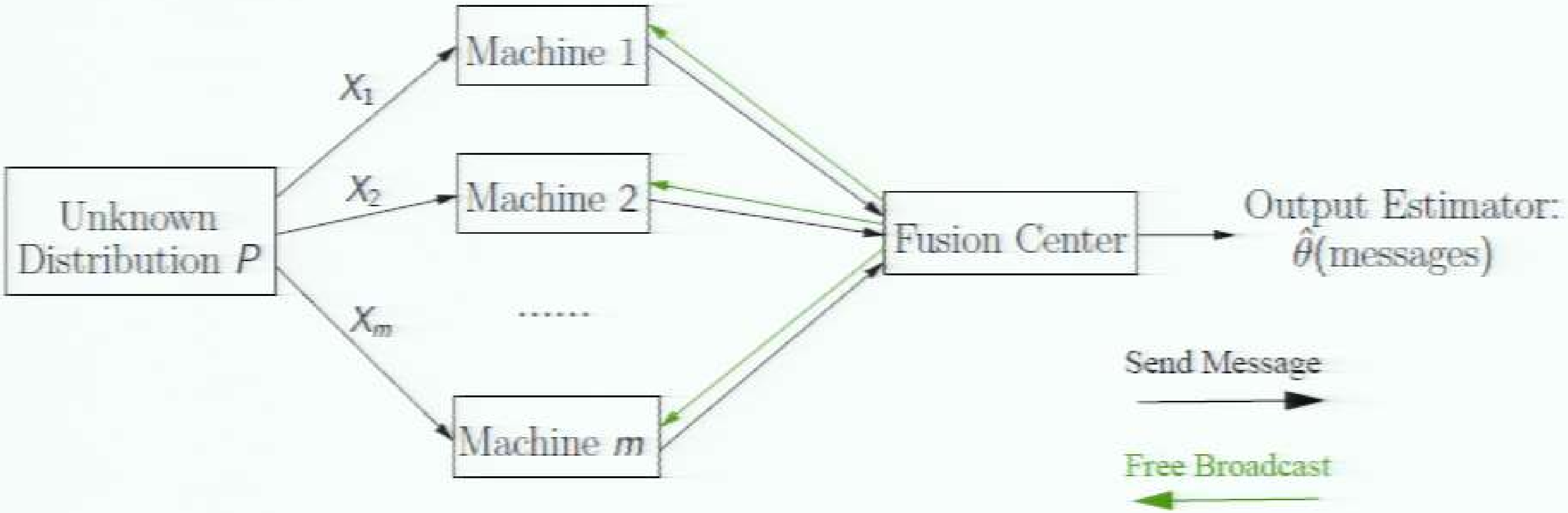
Multiple Rounds of Communication

- In each round, messages are generated by local data and old messages of previous rounds.
- Output a message-based estimator.



Multiple Rounds of Communication

- In each round, messages are generated by local data and old messages of previous rounds.
- Output a message-based estimator.



- Statistical accuracy: $\mathbb{E}[\|\hat{\theta} - \theta\|_2^2]$
- Communication cost: $\sum$ Length(message)

Multiple Rounds of Communication: Lower Bound

Theorem

For {Gaussian location model, linear regression, probit regression} of dimension $d = 1$, any optimal estimator of θ must communicates $\widetilde{\Omega}(m)$ bits.

Remark:

- Interactivity doesn't help (communication cost linear in m).
- Open: generalization to $d > 1$?

Proof Ideas

1. Fix a communication budget $B \geq \text{Length}(messages)$.

Proof Ideas

1. Fix a communication budget $B \geq \text{Length}(\textit{messages})$.
2. Data processing inequality:

$$I(\text{parameter}, \text{messages}) \leq \underbrace{I(\text{parameter}, \text{data})}_{\text{message independent}} \cdot \underbrace{I(\text{data}, \text{messages})}_{\leq B}$$

parameter → data → messages

3. Lower bound $\mathbb{E}[\|\hat{\theta} - \theta\|_2^2]$ by the bound for $I(\text{parameter}, \text{messages})$.

Conclusion

Characterize trade-off between communication and accuracy:

- Single-round communication: Gaussian location model, linear regression, probit regression.
- Interactive communication: same problem set, $d = 1$.

Conclusion

Characterize trade-off between communication and accuracy:

- Single-round communication: Gaussian location model, linear regression, probit regression.
- Interactive communication: same problem set, $d = 1$.

Future Works:

- Generalize the result to other statistical estimation problems.
- Tight lower bound for interactive communication in arbitrary dimension.

NIPS Thanks Its Sponsors

amazon.com

Microsoft Research

Google

facebook

SKYTREE
THE MACHINE LEARNING COMPANY

TWO SIGMA

United Technologies Research Center

D E Shaw & Co

xerox

YAHOO! LABS

IBM Research

TOYOTA

DRW TRADING GROUP

millionshort

criteo

PDT PARTNERS

Springer
Machine Learning Journal

Disney Research

Proof Ideas

1. Fix a communication budget $B \geq \text{Length}(\textit{messages})$.
2. Data processing inequality:

$$I(\text{parameter}, \text{messages}) \leq \underbrace{I(\text{parameter}, \text{data})}_{\text{message independent}} \cdot \underbrace{I(\text{data}, \text{messages})}_{\leq B}$$

parameter $\rightarrow$ data $\rightarrow$ messages

3. Lower bound $\mathbb{E}[\|\hat{\theta} - \theta\|_2^2]$ by the bound for $I(\text{parameter}, \text{messages})$.

For d-dimension problem, a stronger inequality:

$$I(\text{parameter}, \text{messages}) \leq \frac{I(\text{parameter}, \text{data})}{d} \cdot I(\text{data}, \text{messages})$$