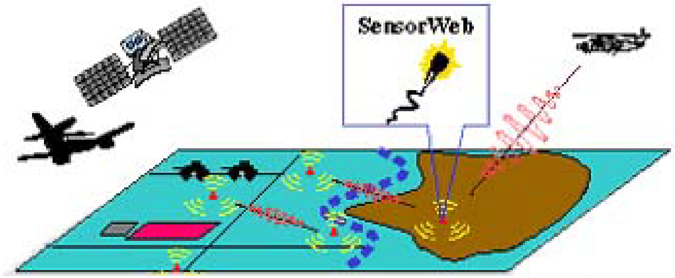
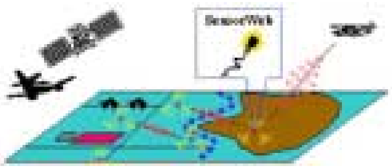


Network-Constrained Estimation

Alan S. Willsky

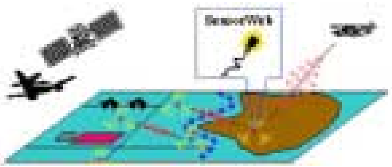
M.I.T.

June 18, 2001



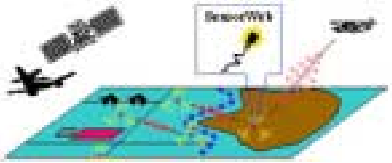
Vital Statistics

- IT-1
- RCA-5, with ties to RCA-6, 2&3
- Participants
  - Sudderth, Wainwright, Johnson, Willsky, Jaakkola
- “Outputs”
  - Several publications
  - Several invited talks
  - Initiating transition of some work already

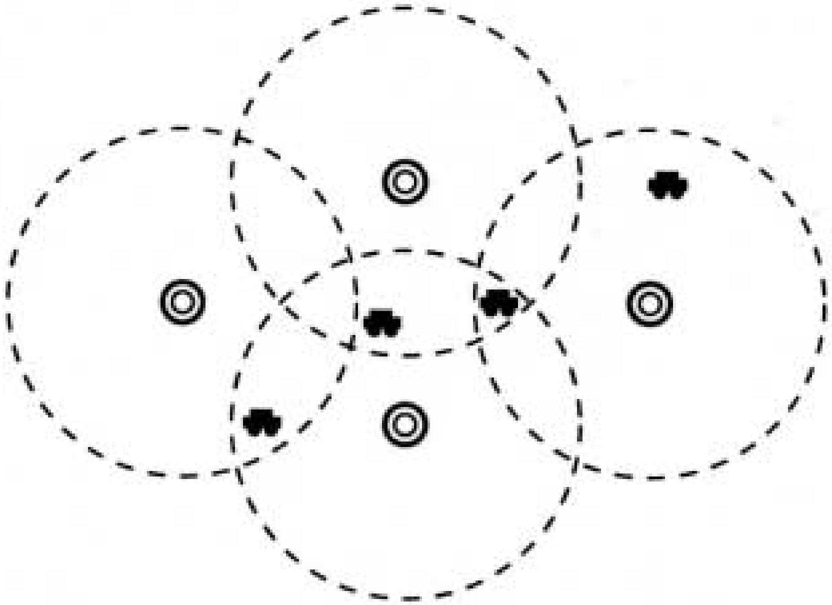


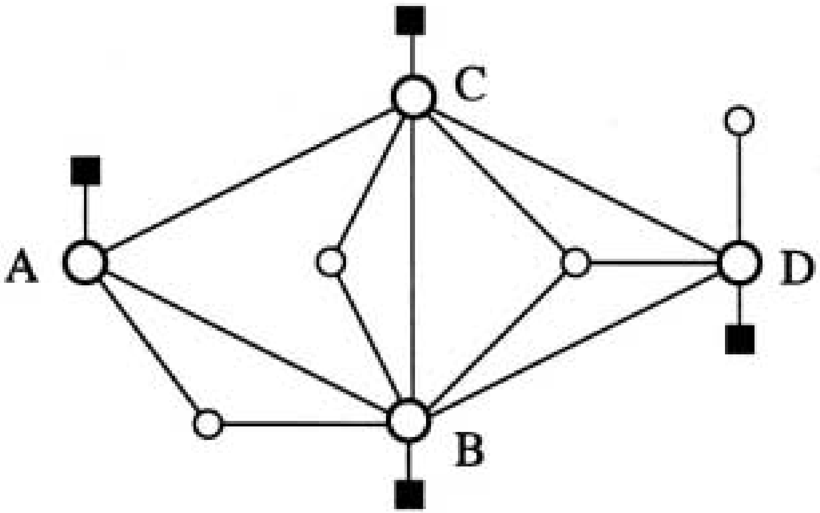
The Problem

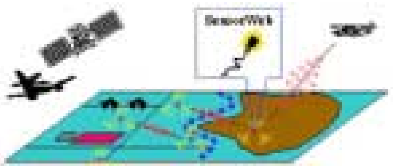
- A network of “nodes”
  - Some representing sensors, some the “hidden” variables to be estimated
  - Links between nodes represent:
    - Statistical relationships among variables (e.g., between measurements and hidden variables or between those variables themselves)
    - Communication links between sensors
- Objective: Perform optimal or provably near optimal estimation of all variables given all data, subject to network constraints



A Notional Example







Linear Estimation on Graphs

$$\begin{aligned} x &\sim \mathcal{N}(0,P) & \qquad v &\sim \mathcal{N}(0,R) \\ y &= Cx+v & \qquad y &\sim \mathcal{N}(0,CPC^T+R) \end{aligned}$$

$x = [\, x_1 \;\; x_2 \;\; \ldots \;\; x_N \,]^T \equiv$ unobserved state variables ($\dim x_i = d$)

$y = [\, y_1 \;\; y_2 \;\; \ldots \;\; y_N \,]^T \equiv$ noisy observations

Optimal MAP/BLSE estimates: $\quad p(x \mid y) \sim \mathcal{N}(\widehat{x}, \widehat{P})$

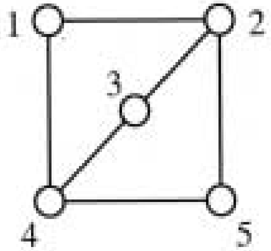
$$\begin{aligned} \hat{P}^{-1}\widehat{x} &= C^T R^{-1} y \\ \widehat{P} &= [P^{-1} + C^T R^{-1} C]^{-1} \end{aligned}$$

Goal: Compute $p(x_i \mid y) \sim \mathcal{N}(\widehat{x}_i, \widehat{P}_i)$ for each node *efficiently.*

Graph Structure and Inverse Covariances

Consider a Gaussian prior $x \sim \mathcal{N}(0, P)$. Partition P^{-1} into a grid of $N \times N$ blocks each of size d.

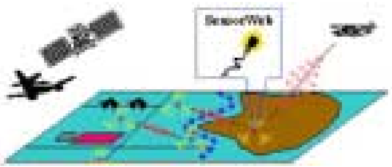
Sparse structure: By Hammersley-Clifford Thm., the $(i,j)^{th}$ block will be nonzero only if there is an edge between nodes i and j.



Graph

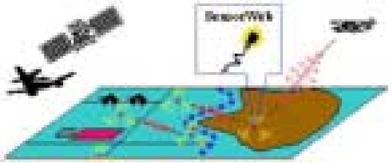
	1	2	3	4	5
1	■	■	□	■	□
2	■	■	■	□	■
3	□	■	■	■	□
4	■	□	■	■	■
5	□	■	□	■	■

Inverse Covariance



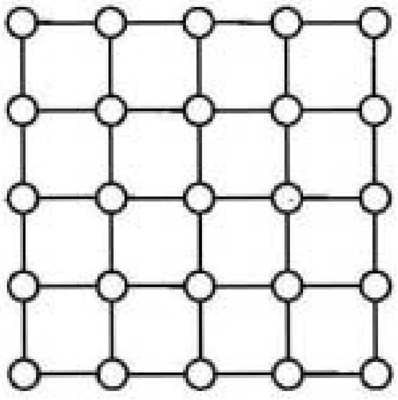
Trees Are Nice

- If the graph is acyclic (e.g., a tree), then there exist very efficient algorithms for optimal estimation
  - Belief propagation (BP)
  - Two-sweep algorithms analogous to Rauch-Tung Striebel smoothing (tree-based Gaussian elimination)
  - Key is the existence of what has been called “partially nested information structures” in decentralized control
- If the graph has cycles, optimal estimation is not so easy
  - “Fill” in Gaussian elimination
  - Iterative algorithms such as BP don’t always converge, and when they do, they give the correct estimates but not the correct covariances

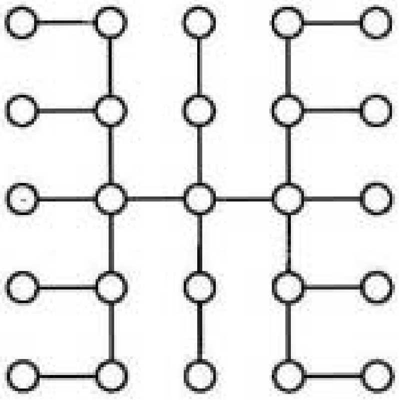


Embedded Trees

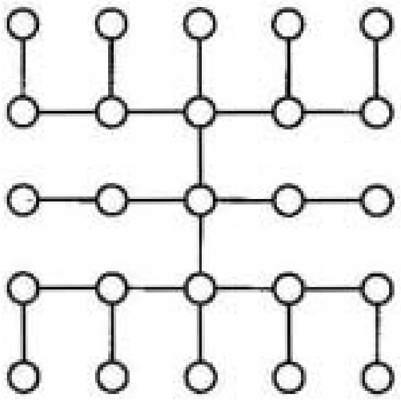
- By the Hammersley–Clifford Theorem, removing edges from a graph is equivalent to zeroing the corresponding entries in P^{-1}
- A variety of spanning trees may be obtained by using different "cutting matrices" K



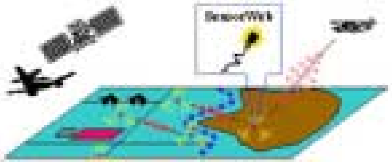
P^{-1}



$P^{-1}_{\mathtt{tree(1)}} = P^{-1} + K_1$



$P^{-1}_{\mathtt{tree(2)}} = P^{-1} + K_2$



ET: Calculation of the estimates

$$\left[P_{\mathrm{tree}}^{-1} - K + C^T R^{-1} C\right] \widehat{x} = C^T R^{-1} y$$

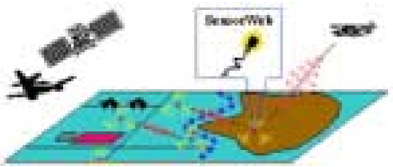
This matrix splitting naturally leads to the iterations

$$\begin{aligned} \left[P_{\mathrm{tree}(t(n))}^{-1} + C^T R^{-1} C\right] \widehat{x}^n &= K_{t(n)} \widehat{x}^{n-1} + C^T R^{-1} y \\ \widehat{x}^n &= M_{t(n)}^{-1} \left[K_{t(n)} \widehat{x}^{n-1} + C^T R^{-1} y\right] \end{aligned}$$

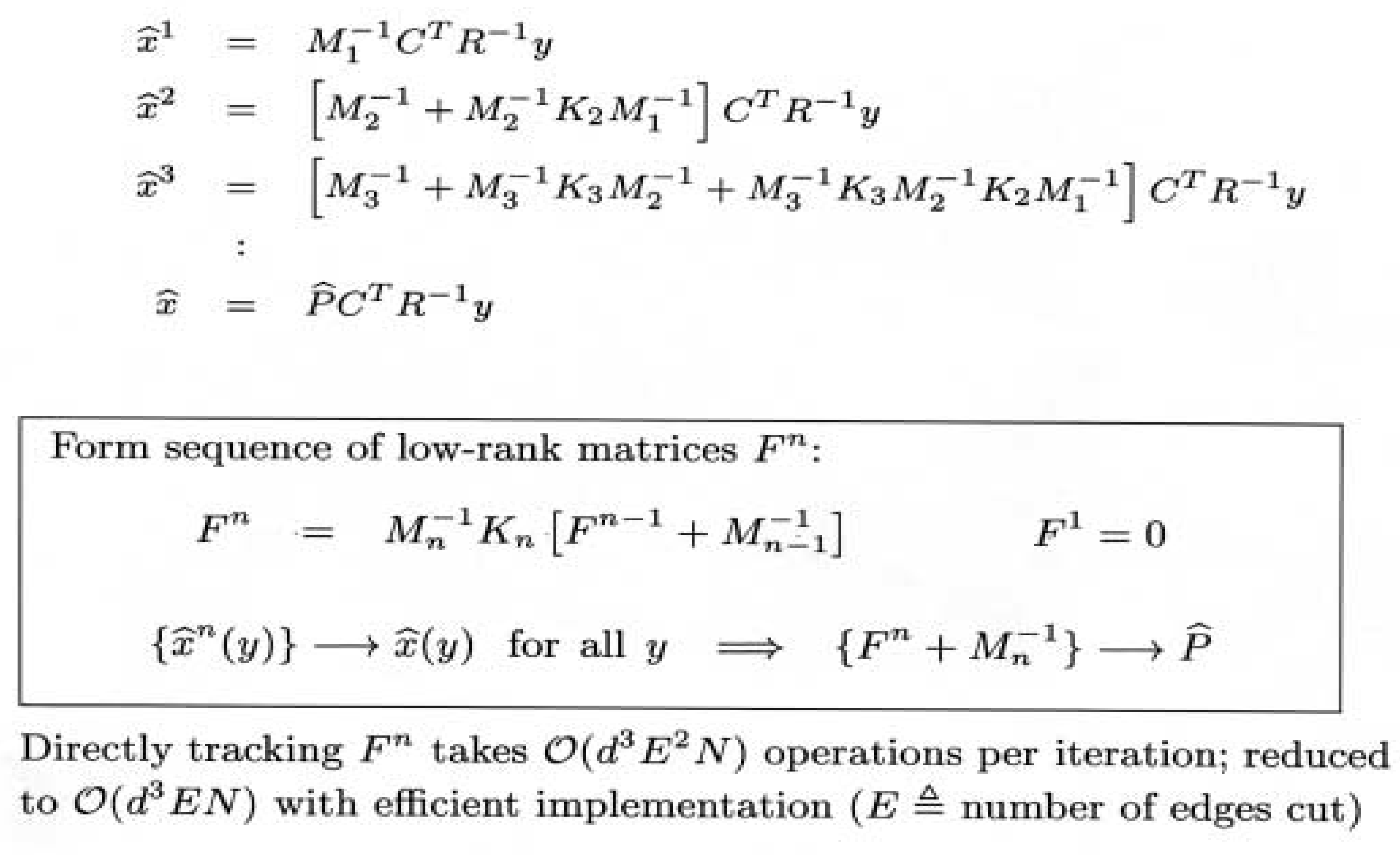
$$M_{t(n)} \triangleq \left[P_{\mathrm{tree}(t(n))}^{-1} + C^T R^{-1} C\right]$$

$t(n) \triangleq$ index of embedded tree for n^{th} iteration

Each iteration is a standard tree-structured Gaussian problem, and can be solved directly in $\mathcal{O}(Nd^3)$ operations.



ET: Calculation of the covariances



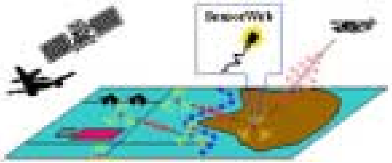
$$\begin{aligned}
\widehat{x}^1 &= M_1^{-1} C^T R^{-1} y \\
\widehat{x}^2 &= \left[M_2^{-1} + M_2^{-1} K_2 M_1^{-1} \right] C^T R^{-1} y \\
\widehat{x}^3 &= \left[M_3^{-1} + M_3^{-1} K_3 M_2^{-1} + M_3^{-1} K_3 M_2^{-1} K_2 M_1^{-1} \right] C^T R^{-1} y \\
&\vdots \\
\widehat{x} &= \widehat{P} C^T R^{-1} y
\end{aligned}$$

Form sequence of low-rank matrices F^n:

$$F^n = M_n^{-1} K_n \left[F^{n-1} + M_{n-1}^{-1} \right] \qquad F^1 = 0$$

$$\{\widehat{x}^n(y)\} \longrightarrow \widehat{x}(y) \text{ for all } y \implies \{F^n + M_n^{-1}\} \longrightarrow \widehat{P}$$

Directly tracking F^n takes $\mathcal{O}(d^3 E^2 N)$ operations per iteration; reduced to $\mathcal{O}(d^3 E N)$ with efficient implementation ($E \triangleq$ number of edges cut)



ET: Convergence

For any initial condition $\widehat{x}^0$, $\widehat{x}$ is the unique fixed point and

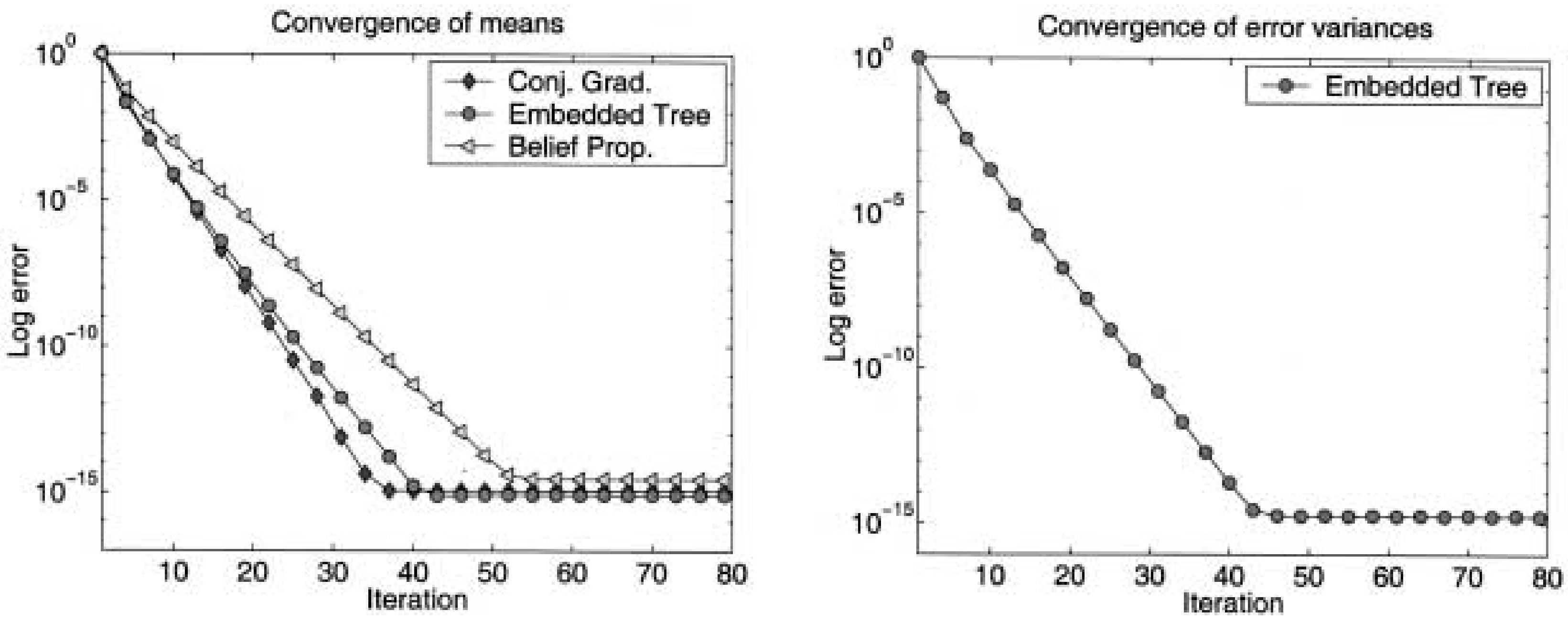
$$(\widehat{x}^n - \widehat{x}) = \left[\prod_{j=1}^{n} M_{t(j)}^{-1} K_{t(j)}\right] (\widehat{x}^0 - \widehat{x})$$

If we periodically cycle through T spanning trees, $\{(\widehat{x}^n - \widehat{x})\}$ evolves according to a linear-periodic system:

$$\mathbf{A} \equiv \prod_{j=1}^{T} M_{t(j)}^{-1} K_{t(j)}$$

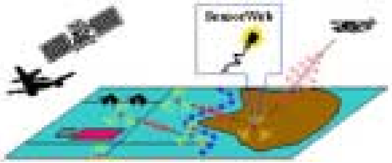
$$\rho(\mathbf{A}) < 1 \Longrightarrow \{(\widehat{x}^n - \widehat{x})\} \overset{n\rightarrow\infty}{\longrightarrow} 0 \text{ geometrically at rate } \gamma \equiv \rho(\mathbf{A})^{\frac{1}{T}}$$

Result: Inference on 20x20 Grid

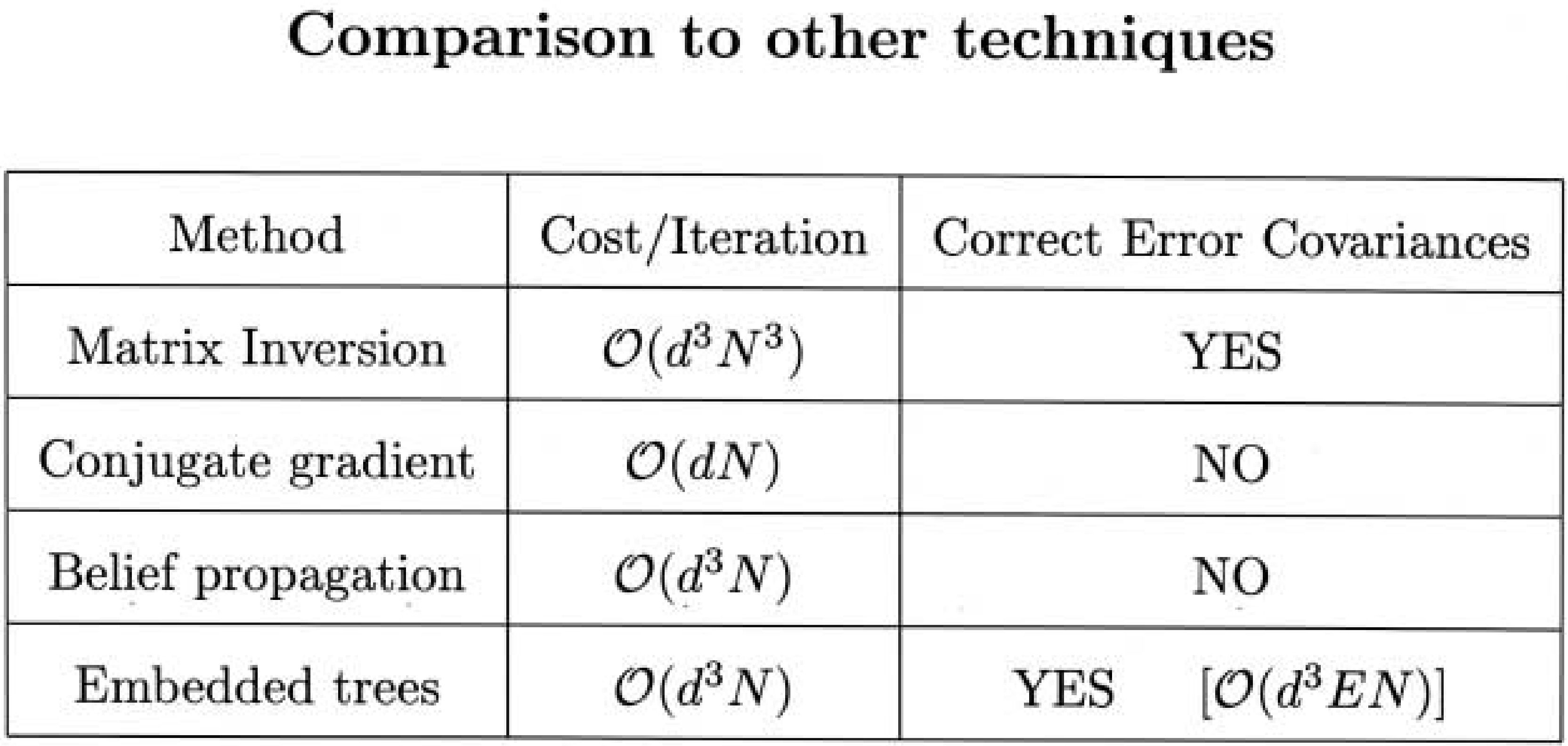


(a) Convergence of means

(b) Convergence of covariances

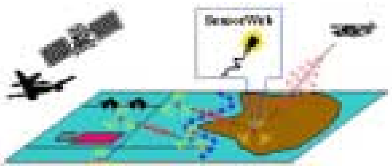


Complexity Comparisons



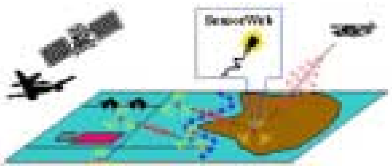
Comparison to other techniques

Method	Cost/Iteration	Correct Error Covariances
Matrix Inversion	$\mathcal{O}(d^3N^3)$	YES
Conjugate gradient	$\mathcal{O}(dN)$	NO
Belief propagation	$\mathcal{O}(d^3N)$	NO
Embedded trees	$\mathcal{O}(d^3N)$	YES $[\mathcal{O}(d^3EN)]$



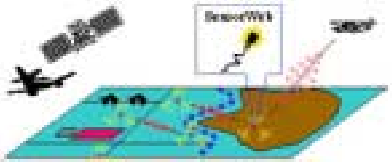
ET’s not quite ready to phone home

- Compact (and computable) sufficient or necessary & sufficient conditions for convergence
- New algorithmic structures using ET as a preconditioner for CG
- Faster results in some cases when ET *diverges*
- Asynchronous versions using only local network structure
- Optimal or at least good choices of spanning trees
- Randomized choices of spanning trees



Tree-Based Reparameterization (TRP)

- Motivated by success of ET, with focus here on discrete-valued processes
- The key idea is that distributions over trees admit very special factorizations in terms of marginal distributions at individual nodes and over maximal cliques (assumed here to be doubletons)
- The idea uses the generalization of factorizations for Markov chains



Estimation for a Markov process on a graph G

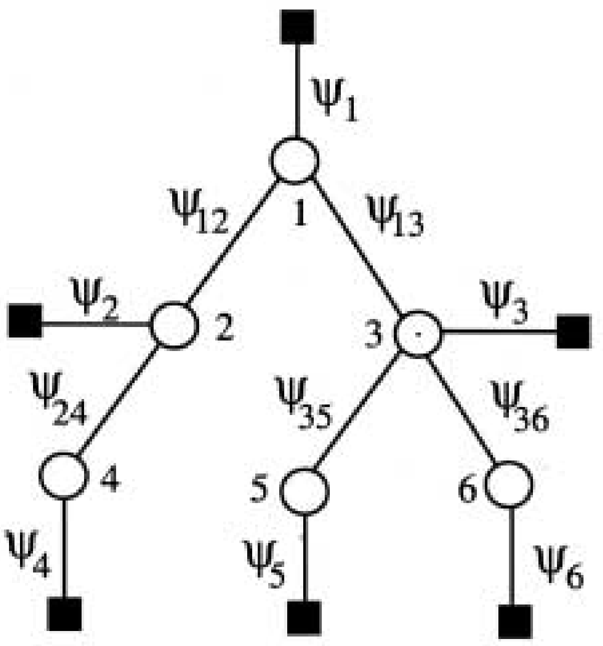
Consider stochastic process x on $\mathcal{G}$ such that $p(e) > 0 \;\; \forall\, e \,\in\, \mathcal{X}$.

$$\underbrace{x \;\; \text{is} \;\; \text{Markov} \;\; \text{w.r.t} \;\; \mathcal{G}}_{\text{Markov property}} \qquad \Longleftrightarrow \qquad \underbrace{p(x) = \frac{1}{Z} \prod_{C} \psi_C(x)}_{\text{Factorization of distribution}}$$

Here $Z = \sum_x \prod_C \psi_C(x)$ is the partition function that normalizes the distribution.

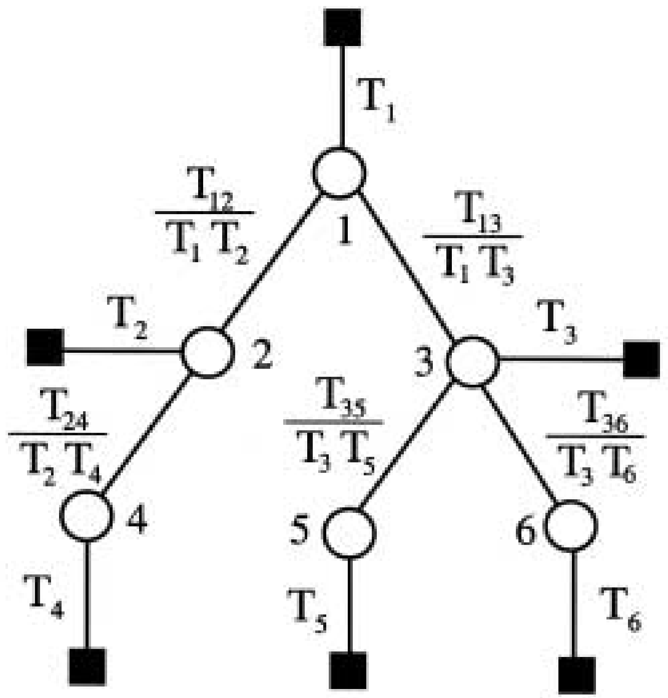
Objective : Seek exact or approximate marginals $T_s(x_s), T_{st}(x_s, x_t)$ through reparameterization of the factorized form of *p(x)*

Tree estimation as reparameterization



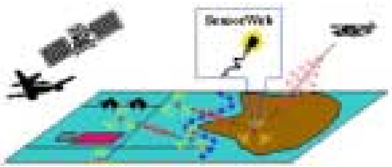
(a) Initial parameterization

$$p(x) = \frac{1}{Z}\prod_s \psi_s \prod_{(s,t)} \psi_{st}$$

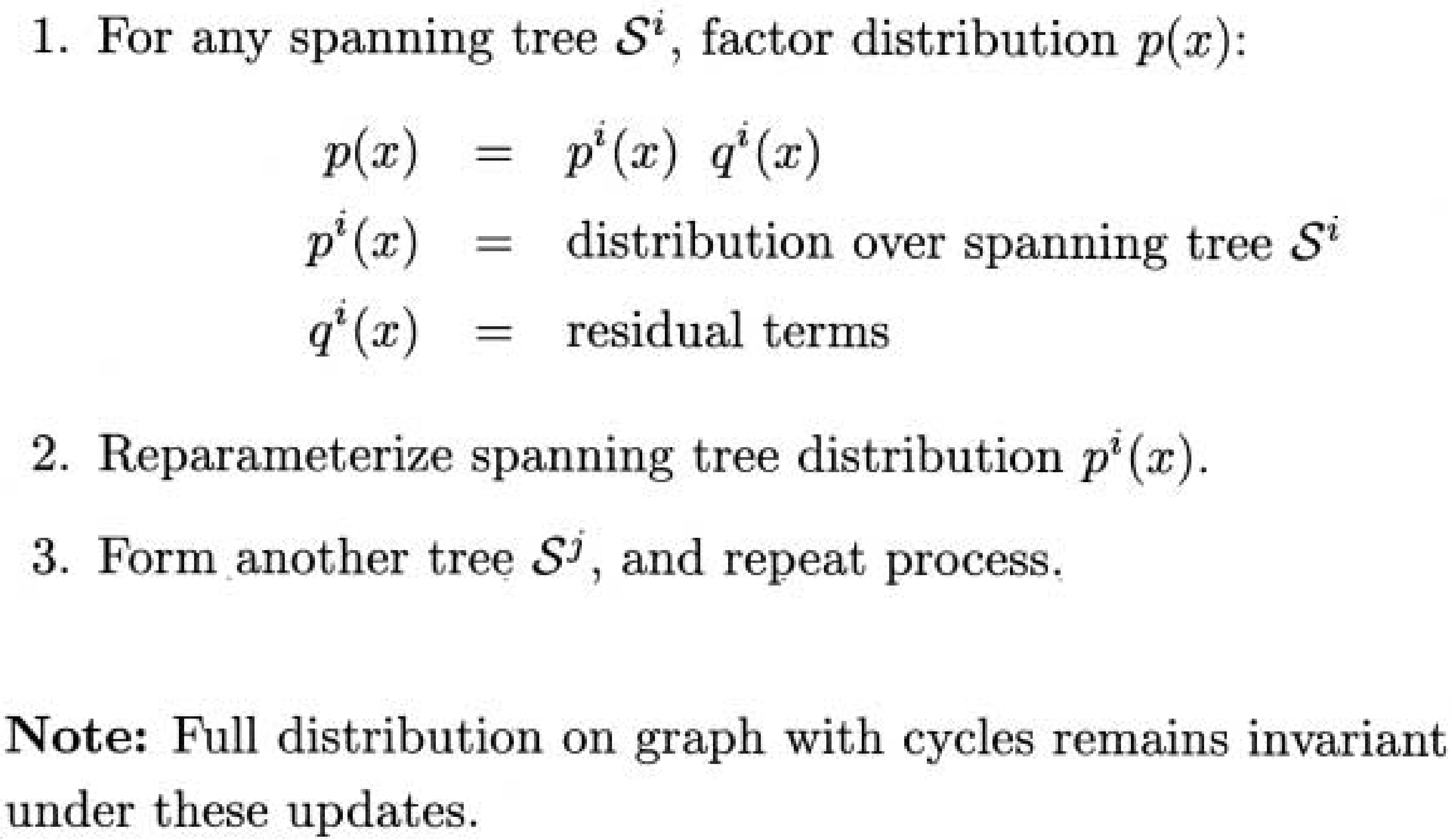


(b) Desired parameterization

$$p(x) = \prod_s T_s \prod_{(s,t)} \frac{T_{st}}{T_s\, T_t}$$



TRP: The Basic Idea



1. For any spanning tree $\mathcal{S}^i$, factor distribution $p(x)$:

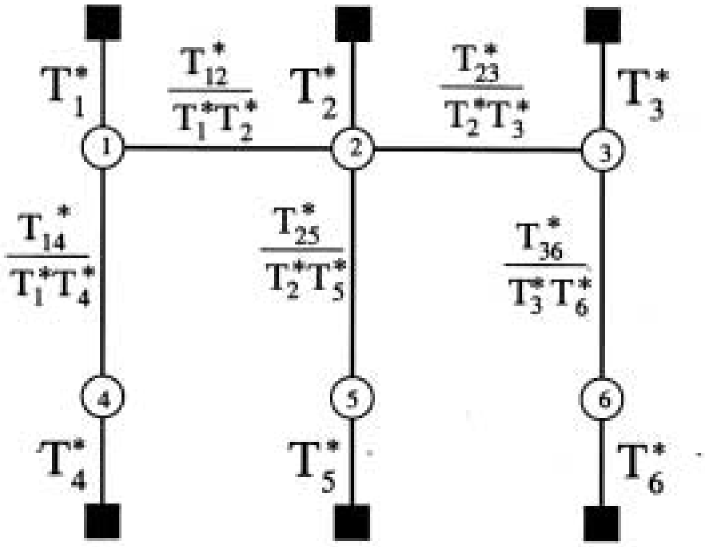
$$\begin{aligned} p(x) &= p^i(x)\ q^i(x) \\ p^i(x) &= \text{distribution over spanning tree } \mathcal{S}^i \\ q^i(x) &= \text{residual terms} \end{aligned}$$

2. Reparameterize spanning tree distribution $p^i(x)$.

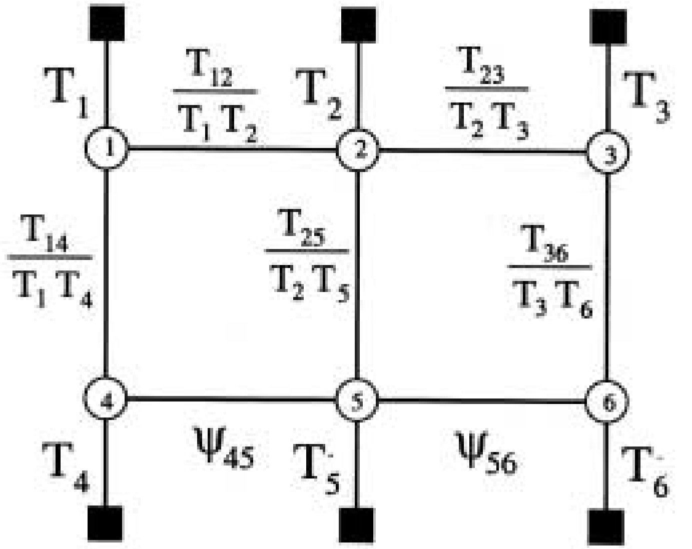
3. Form another tree $\mathcal{S}^j$, and repeat process.

Note: Full distribution on graph with cycles remains invariant under these updates.

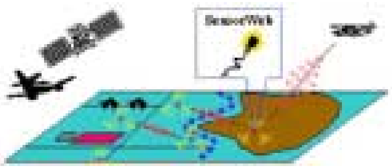
Graphical Illustration


T_1^*
$\frac{T_{12}^*}{T_1^*T_2^*}$
T_2^*
$\frac{T_{23}^*}{T_2^*T_3^*}$
T_3^*
1
2
3
$\frac{T_{14}^*}{T_1^*T_4^*}$
$\frac{T_{25}^*}{T_2^*T_5^*}$
$\frac{T_{36}^*}{T_3^*T_6^*}$
4
5
6
T_4^*
T_5^*
T_6^*

(a) Reparameterize spanning tree

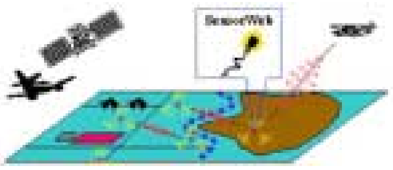


(b) Full graph after update

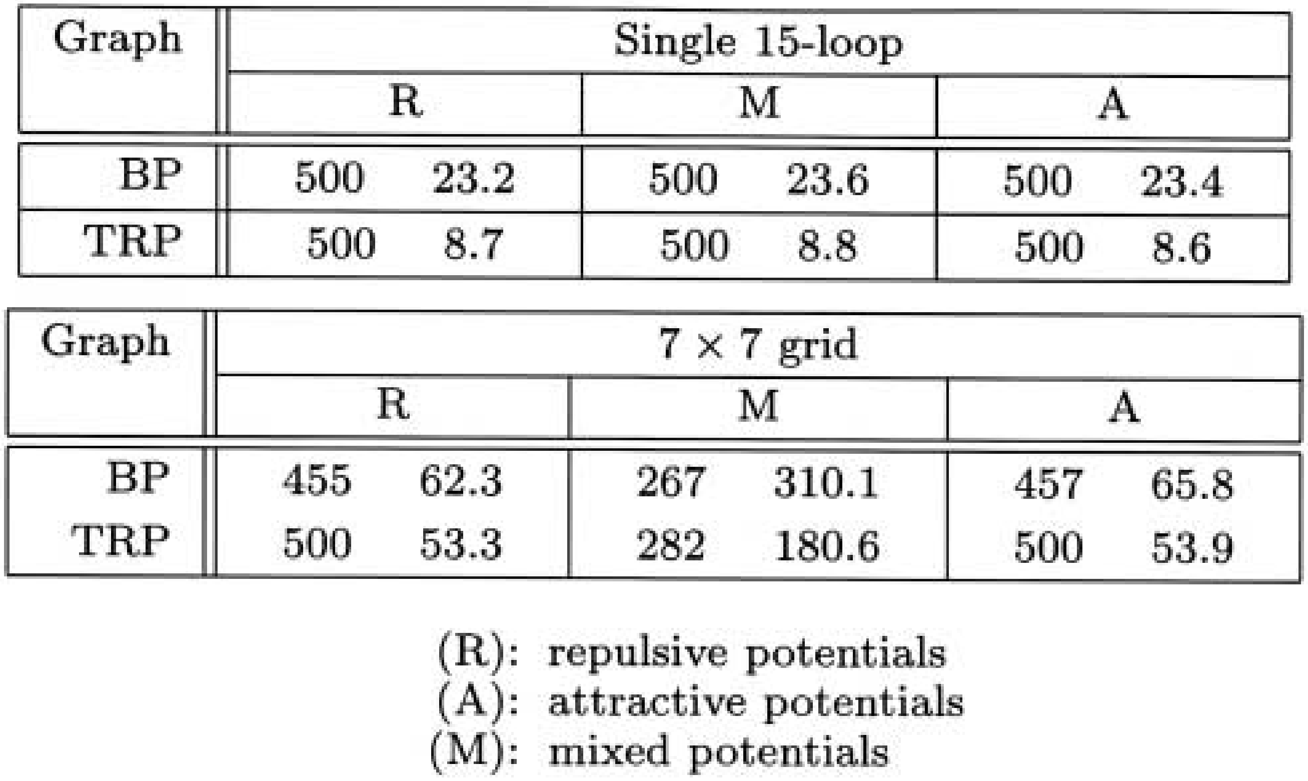


TRP and BP

- Interpretation of BP as a TRP algorithm, using two-node, non-spanning trees
  - Yields alternate algorithmic structure which cuts storage requirement in half
- Empirical results confirm intuition that more global communication structure of TRP yields gains
  - Lower total computational/communication cost
  - Converges in some cases in which BP does not and converges at least as fast or faster than BP when BP does converge



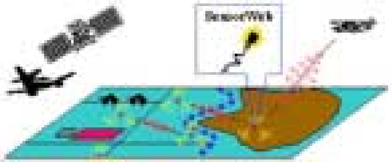
Empirical Results



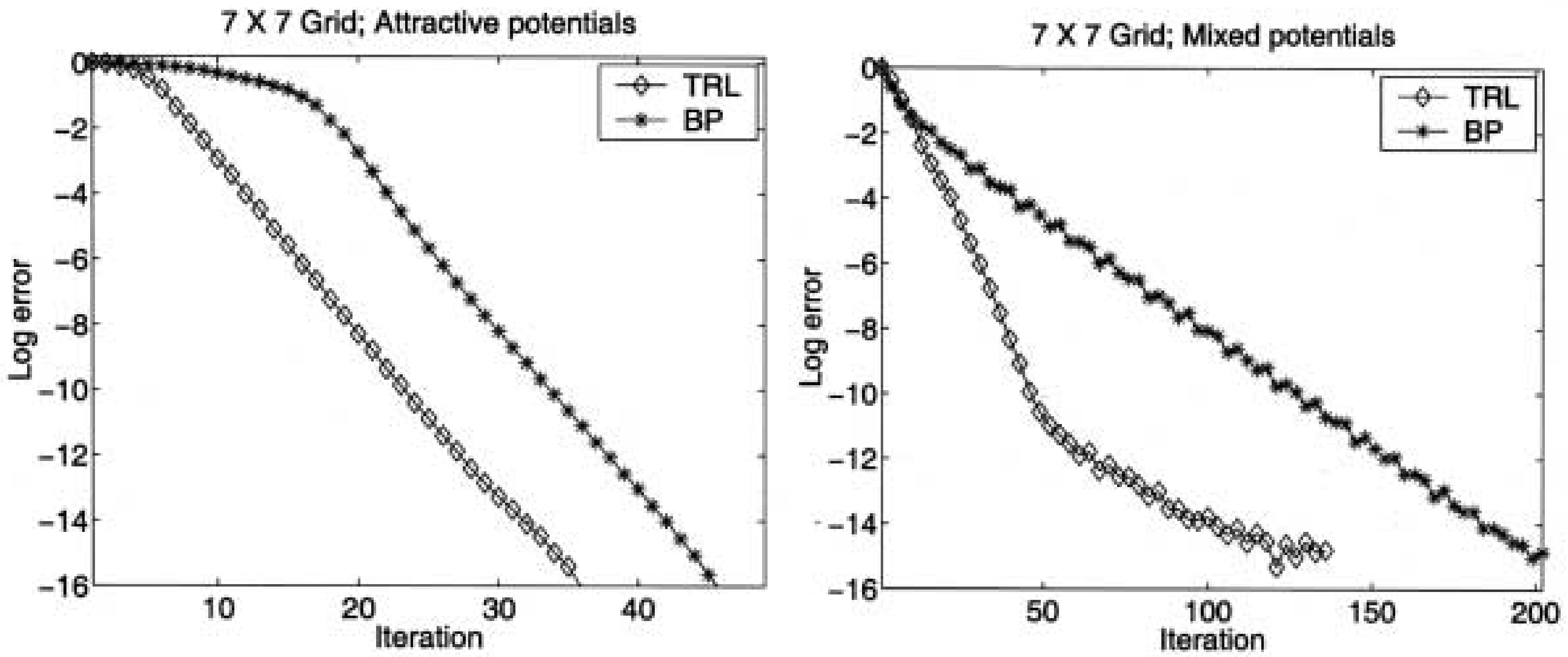
Graph	Single 15-loop					
	R		M		A	
BP	500	23.2	500	23.6	500	23.4
TRP	500	8.7	500	8.8	500	8.6

Graph	7×7 grid					
	R		M		A	
BP	455	62.3	267	310.1	457	65.8
TRP	500	53.3	282	180.6	500	53.9

(R): repulsive potentials
(A): attractive potentials
(M): mixed potentials

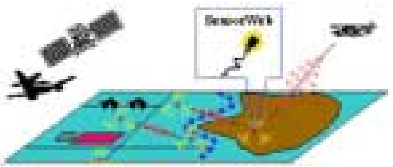


Convergence Plots



(a) Attractive potentials

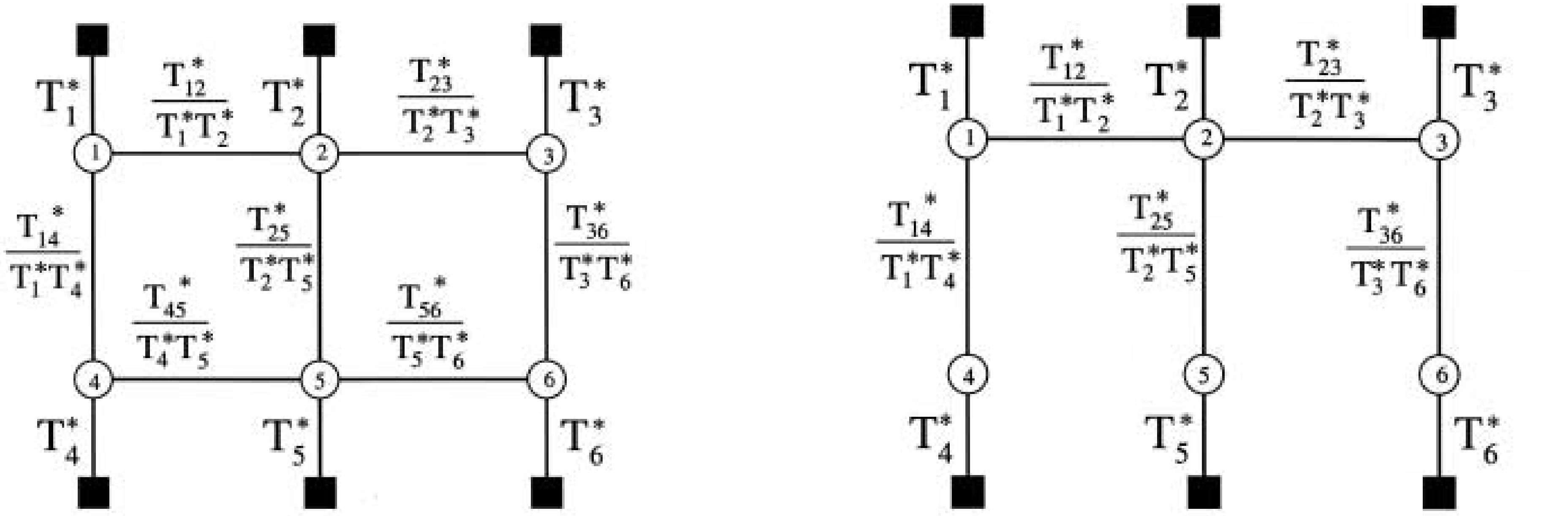
(b) Mixed potentials



Theoretical Analysis of TRP

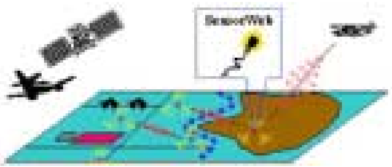
- Interpretation of TRP as successive projection operation using a “distance” related to Kullback-Liebler Divergence
    - Demonstrates ties to analysis of BP and minimization of Bethe free energy
    - Key is using an overcomplete parameterization of an exponential family of distributions
    - Leads to a characterization of fixed points

Interpretation of Fixed Points


T_1^*
$\frac{T_{12}^*}{T_1^*T_2^*}$
T_2^*
$\frac{T_{23}^*}{T_2^*T_3^*}$
T_3^*
1
2
3
$\frac{T_{14}^*}{T_1^*T_4^*}$
$\frac{T_{25}^*}{T_2^*T_5^*}$
$\frac{T_{36}^*}{T_3^*T_6^*}$
$\frac{T_{45}^*}{T_4^*T_5^*}$
$\frac{T_{56}^*}{T_5^*T_6^*}$
4
5
6
T_4^*
T_5^*
T_6^*
T_1^*
$\frac{T_{12}^*}{T_1^*T_2^*}$
T_2^*
$\frac{T_{23}^*}{T_2^*T_3^*}$
T_3^*
1
2
3
$\frac{T_{14}^*}{T_1^*T_4^*}$
$\frac{T_{25}^*}{T_2^*T_5^*}$
$\frac{T_{36}^*}{T_3^*T_6^*}$
4
5
6
T_4^*
T_5^*
T_6^*

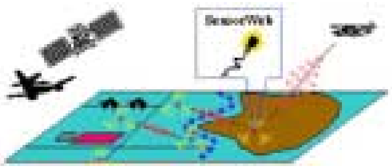
(a) Full graph labeling

(b) Consistent tree parameterization



Fixed Points and Convergence

- Fixed points exist!
- Fixed points of TRP and BP are the same
- Sufficient condition for application of TRP with two spanning trees
- Gives elementary proof that in the Linear-Gaussian case, BP (when it converges) yields the correct estimates but incorrect error variances
- Interesting question: Can the exact marginals form a fixed point?
  - Answer: There are some cases where it can, but (we believe) these form a very special (and thin) set



Error Analysis

- Conceptually useful exact representation of error
- Leads to upper and lower bounds on error in probabilities produced by TRP (or BP) when they converge

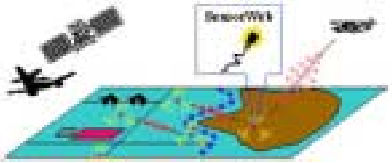
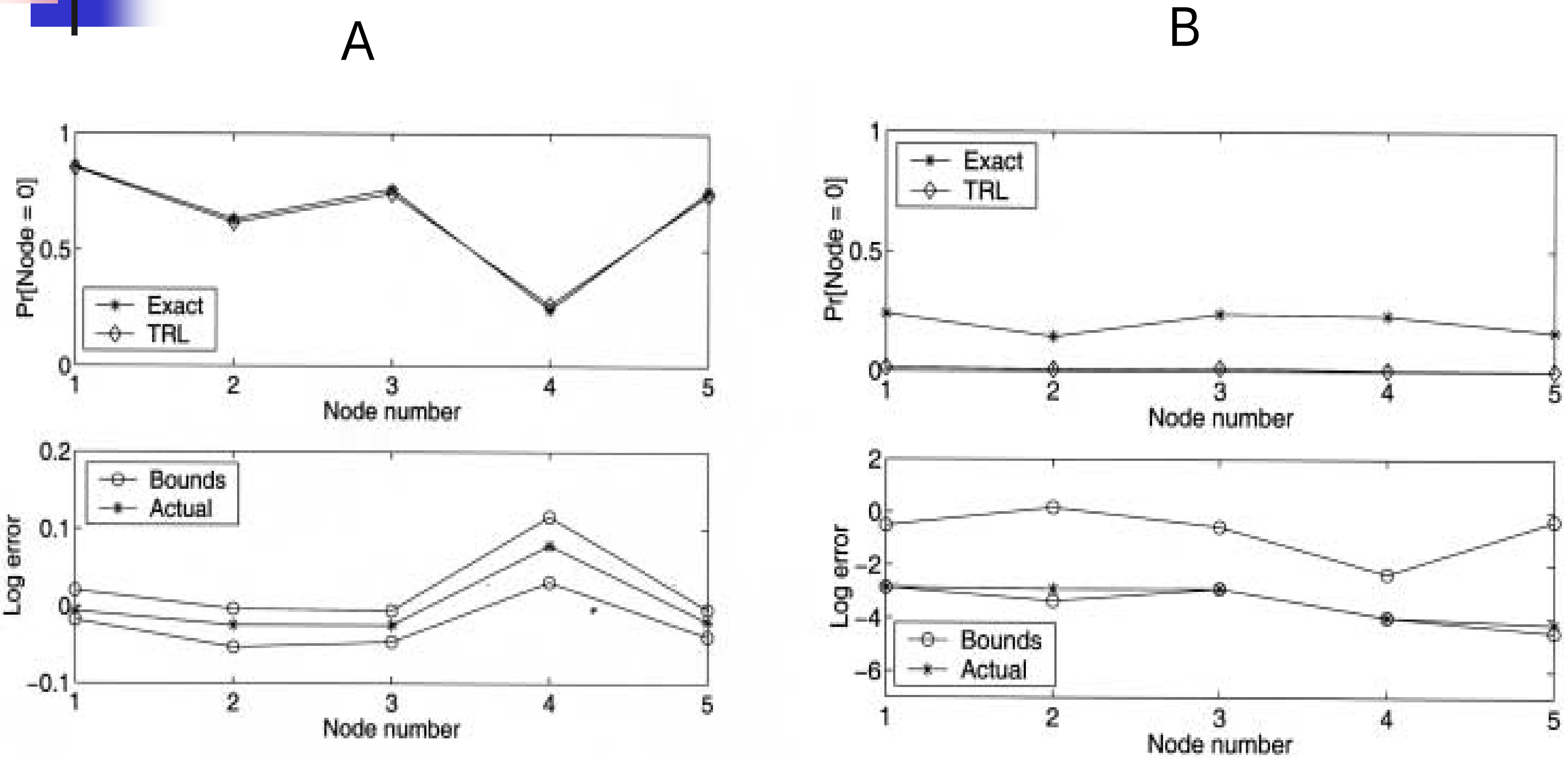
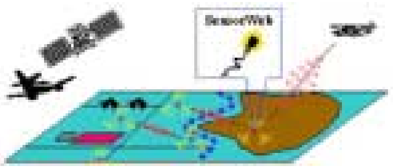


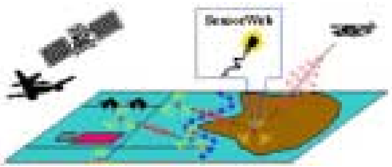
Illustration of Bounds





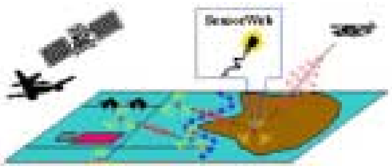
Where to from here?

- Enhanced bounds and analysis of behavior?
  - Sensitivity analysis to understand “breaking points” of the algorithm
  - Characterizating when TRP yields exact answers
- Choice of trees
  - For algorithm and for bounds
- Asynchronous, distributed implementation
  - Parallel operation à la BP
  - Without global knowledge of network structure
  - Robust to changes in network structure
- New and better algorithms!



Recursive Cavity Models (RCM’s)

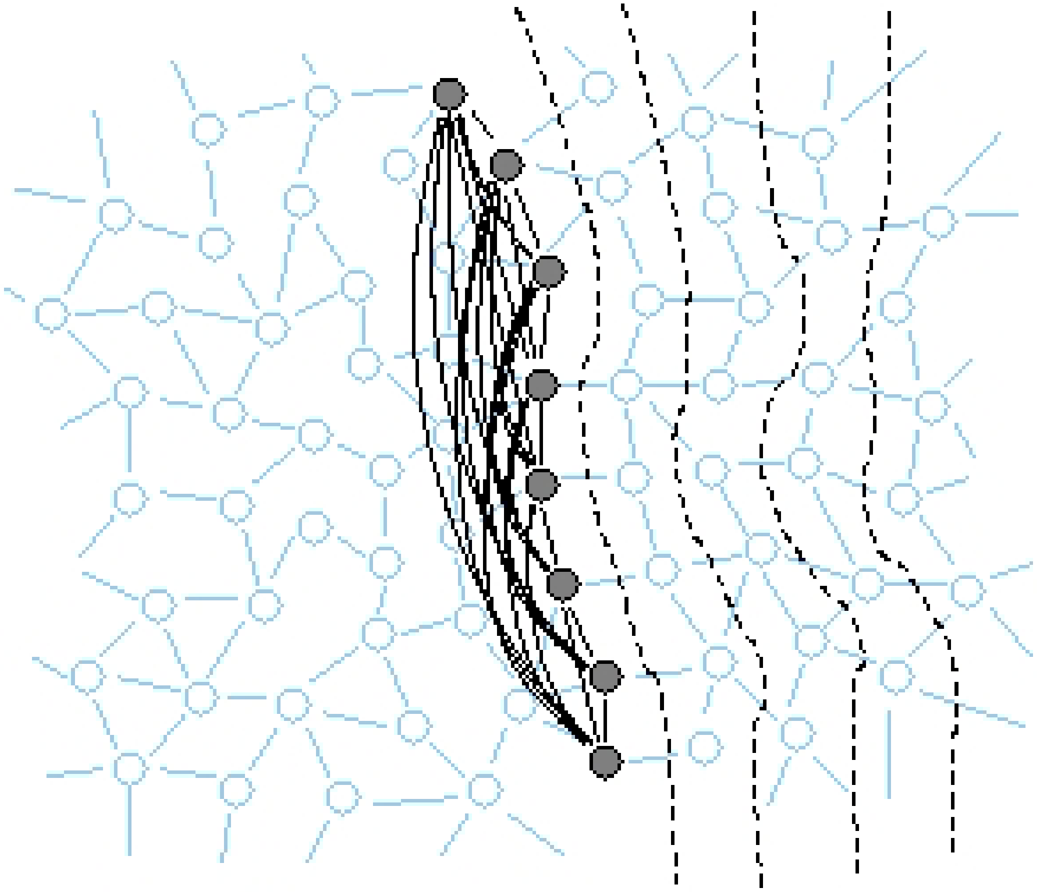
- The concept of a separator set, S
  - Partitions the nodes of a graph into disjoint sets, A and B, such that any path from one set to the other passes through S
  - Conditioned on the values on S, the values on A and B are independent
- This suggests the idea of a recursive partitioning of the graph, with the “state” of the process corresponding to the values of the process along a separating boundary
  - Closely related to the idea of “frontier models” for dynamic Bayes’ nets
  - The challenge is dealing with “fill” for boundary states



Frontier Models and RCM's

- Closely related to "marching methods" for PDE's
  - Boundary Models are propagated from frontier to frontier
  - These correspond (in the linear case) to so-called information representations (propagation of P^{-1} and $P^{-1}\hat{x}$)
  - Approximations made to keep P^{-1} sparse, based on locally available statistical quantities
  - Computation of estimates then involves separate calculations on each boundary

Notional Picture of a Frontier Model



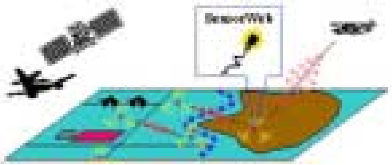
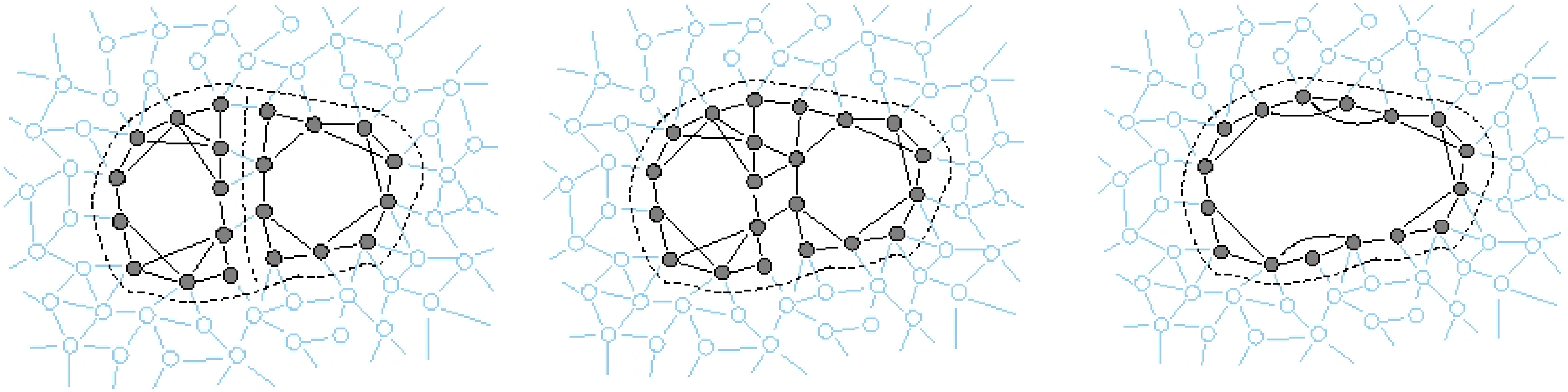
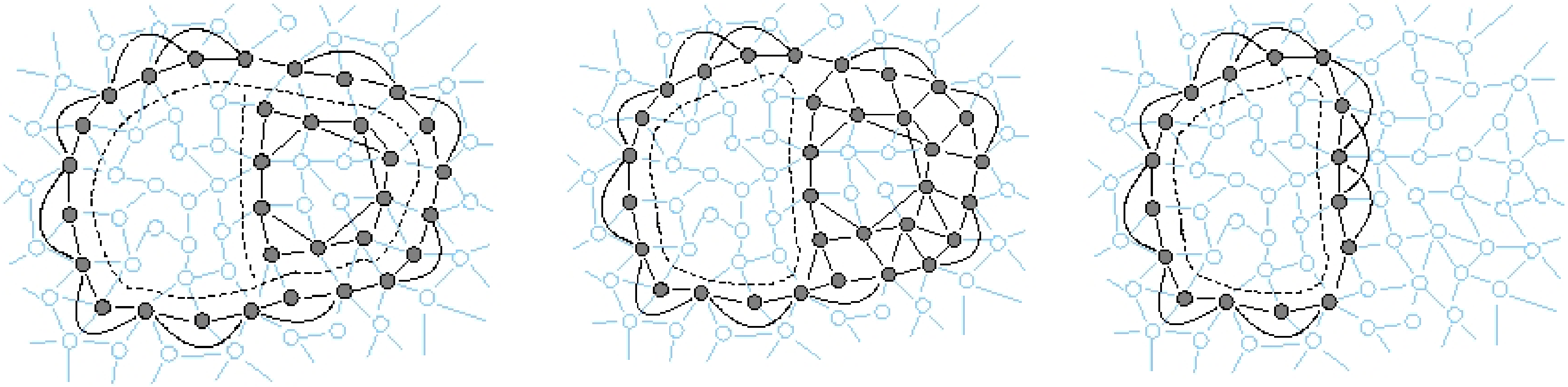
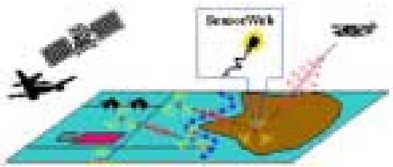


Illustration of the Upsweep of RCM



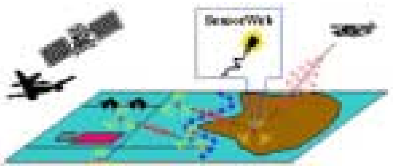
The RCM Downsweep





Computation of Estimates

- Corresponds to solving sparse/graphical equations around each boundary
  - These could also be solved, if desired, using graphical techniques (e.g., ET)
- RCM can be embedded in an iterative algorithm much as ET can, leading to very efficient iterative algorithms, in essence using RCM as a preconditioner



Where to from here?

- Global measures of approximation error and stability results
  - Ensuring that approximations made at one boundary do not cause divergence more globally
- Putting something into the cavities
  - Latent variables
    - Improving boundary models
    - Capturing more global, long-distance characteristics/correlations (à la multipole methods for PDE’s)

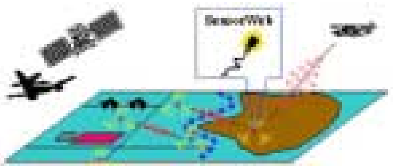


Illustration of RCM with Latent Nodes