

Propagation-Based Approximation

Sargur Srihari
srihari@cedar.buffalo.edu

Topics

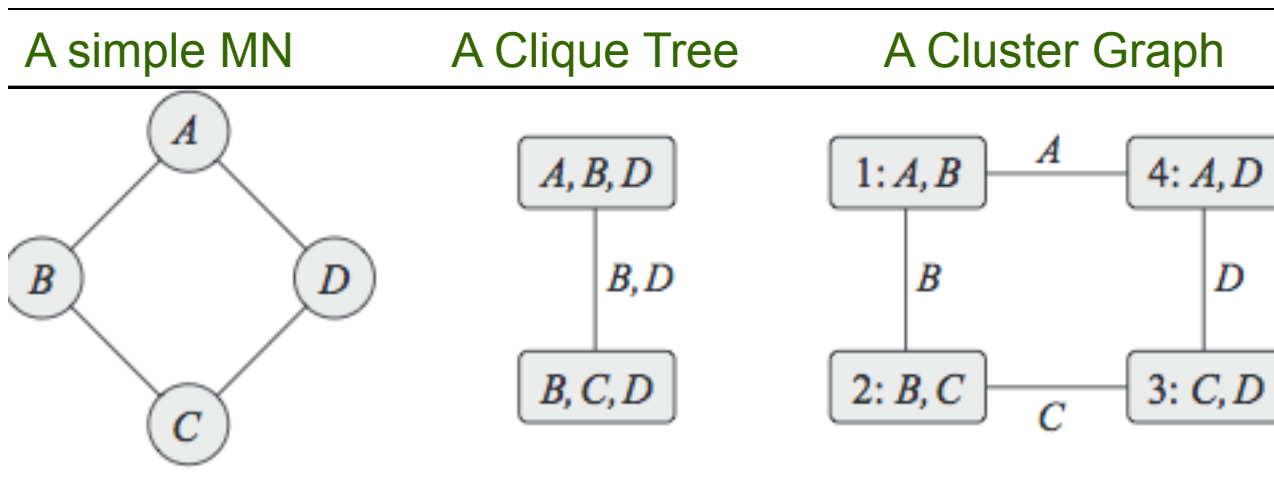
- Propagation-Based Approximation
 1. A simple example of Loopy Belief Propagation
 2. Cluster-graph belief propagation
 3. Properties of cluster graph belief propagation
 4. Analyzing convergence
 5. Constructing cluster graphs
 1. Pairwise Markov networks
 2. Bethe cluster graphs
 3. Beyond marginal probabilities
 6. Variational analysis
 7. Other entropy approximations
 8. Discussion

Propagation-Based Approximation

- These are methods that use exactly the same message propagation as in exact inference
 - However the propagation schemes use a general-purpose cluster graph instead of clique trees

Cluster graph definition:

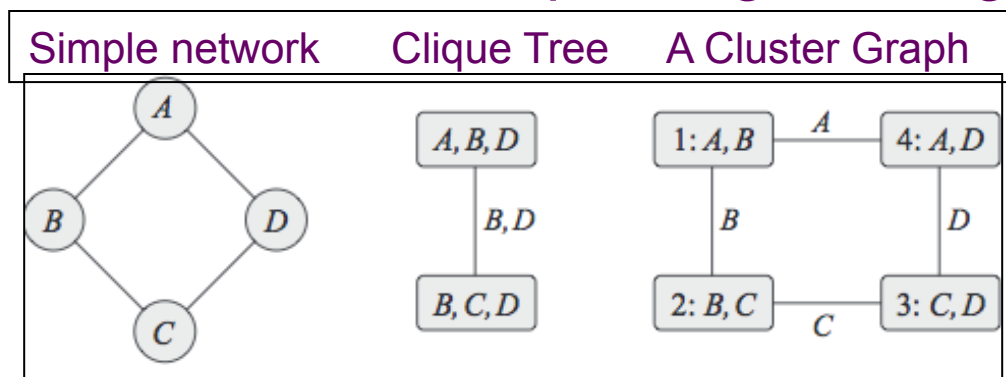
A data structure that provides a graphical flowchart of the factor manipulation process. Each *node* is a subset of variables. Each *edge* connects nodes. With nonempty intersection of scopes (called sepset)



Since constraints defining A clique tree are crucial for exact inference Message propagation Algorithms that use cluster graphs will generally not provide correct answers

A simple example

- Propagation-based approx. with a simple MN
 - To perform exact inference first reduce it to a tree
 - Inference involves passing messages over sepset $\{B,D\}$

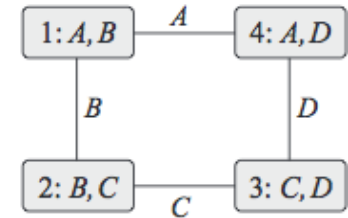


- Cluster graph has loops— so not a tree; a *loopy* graph
 - Nevertheless apply belief-update algorithm Ctree-BU-calibrate
- Clusters are smaller than in Clique Tree
 - Therefore message passing steps are less expensive
 - But what are the results of this procedure?

Loopy Belief Propagation

- Suppose we propagate messages

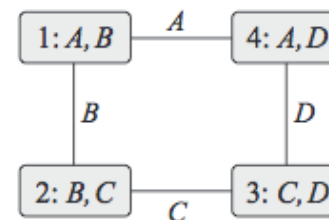
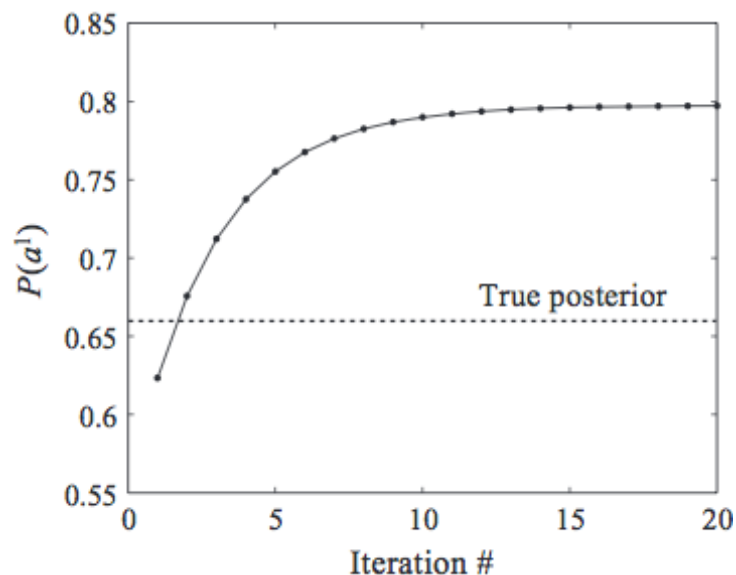
- in following order $\mu_{1,2}, \mu_{2,3}, \mu_{3,4}, \mu_{4,1}$



- In the first message the cluster $\{A,B\}$ passes information to cluster $\{B,C\}$ through a marginal distribution on B
 - In final message $\mu_{4,1}$ information reaches original cluster
- Suppose all potentials prefer consensus assignment,
 - i.e., $\beta_1(a^0, b^0)$ and $\beta_1(a^1, b^1)$ are much larger than $\beta(a^1, b^0)$ and $\beta_1(a^0, b^1)$ and similarly for other beliefs.
 - Thus if the message $\mu_{1,2}$ strengthens the belief that $B=b^1$ then the message $\mu_{2,3}$ will increase the belief in $C=c^1$, etc
 - Once we go around the loop, message $\mu_{4,1}$ will strengthen the support in $A=a^1$
 - This will be incorporated into the cluster as independent evidence
 - If we continue to apply the same sequence of propagations again, we will keep increasing the beliefs in the assignment $A=a^1$

Example run of Loopy Belief Propagation

- All potentials prefer consensus assignments over nonconsensus ones
- In each iteration, we perform message passing for all the edges in the cluster graph



Coding Theory and Loopy BP

- Sending messages over a noisy channel and recovering
- We wish to send a k -bit message u_1, \dots, u_k
- Encode the message using n bits x_1, \dots, x_n
- Resulting in corrupted outputs y_1, \dots, y_n
- Task is to recover an estimate $\tilde{u}_1, \dots, \tilde{u}_k$ from y_1, \dots, y_n
- Message decoding can be formulated as a probabilistic inference task

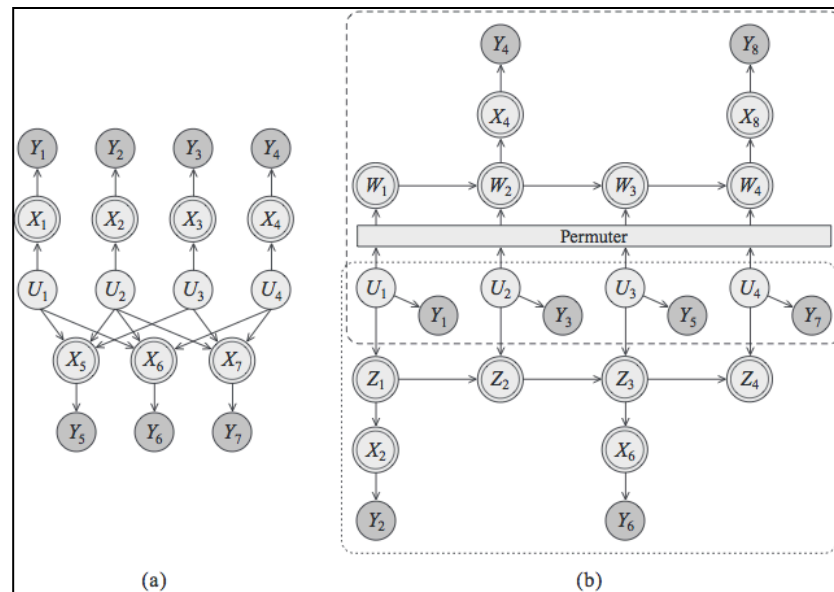
Noise Models and Error Rate

- Outputs can be discrete or continuous
 - Different channels introduce different noise
 - Addition of Gaussian noise
 - Flip bits independently with some probability p
 - Noise is added in a correlated way
- Bit error rate
 - Probability that bit is decoded incorrectly
- Rate of a code
 - k/n : ratio of no. of msg bits to no. of transmit bits
 - Repetition code: transmit each bit 3 times, decode by majority vote, has bit error rate $p^3 + 3p^2$
 - Shannon: for a given rate, max noise level tolerated while achieving a certain bit error rate

Two Examples of Codes

- A $k=4, n=7$ parity check code where every four message bits are sent along with three bits that encode parity checks
- A $k=4, n=8$ turbocode

BN formulation
With Belief propagation



Cluster Graph Belief Propagation

- Sum-Product BP in a Cluster Graph

- Procedure *Cgraph-SP-Calibrate* (

```

     $\Phi$ , // Set of factors
     $\mathcal{U}$  // Generalized cluster graph  $\Phi$ 
)
1 Initialize-CGraph
2 while graph is not calibrated
3   Select  $(i-j) \in \mathcal{E}_{\mathcal{U}}$ 
4    $\delta_{i \rightarrow j}(S_{i,j}) \leftarrow \text{SP-Message}(i, j)$ 
5   for each clique  $i$ 
6      $\beta_i \leftarrow \psi_i \cdot \prod_{k \in \text{Nb}_i} \delta_{k \rightarrow i}$ 
7   return  $\{\beta_i\}$ 

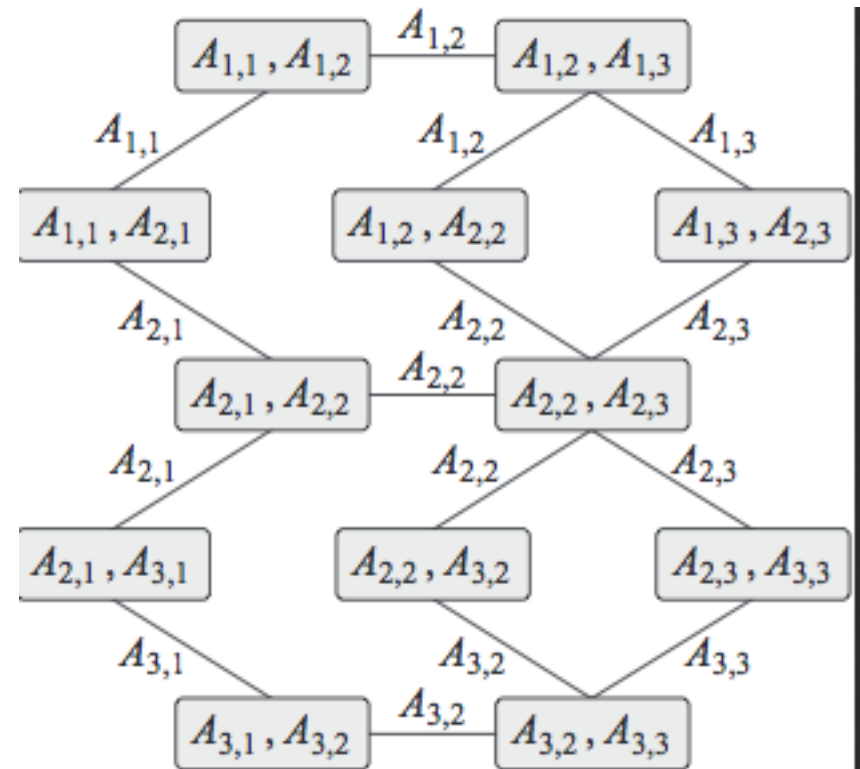
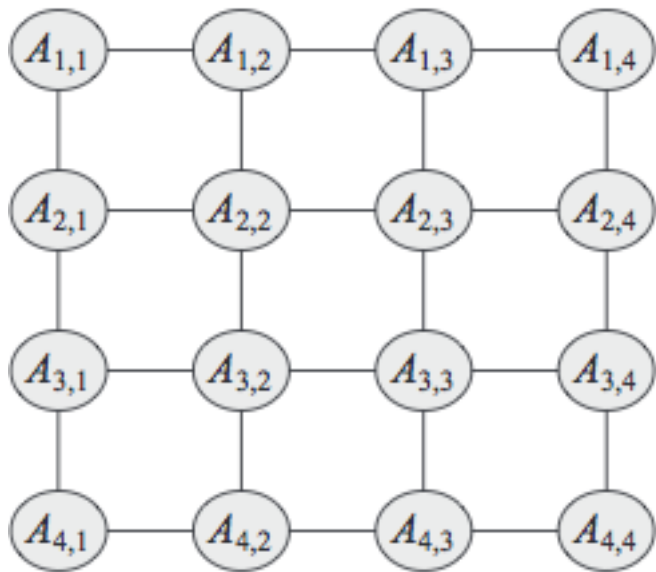
Procedure Initialize-CGraph (
   $\mathcal{U}$ 
)
1 for each cluster  $C_i$ 
2    $\beta_i \leftarrow \prod_{\phi : \alpha(\phi)=i} \phi$ 
3 for each edge  $(i-j) \in \mathcal{E}_{\mathcal{U}}$ 
4    $\delta_{i \rightarrow j} \leftarrow 1$ 
5    $\delta_{j \rightarrow i} \leftarrow 1$ 
6

Procedure SP-Message (
   $i$ , // sending clique
   $j$  // receiving clique
)
1  $\psi(C_i) \leftarrow \psi_i \cdot \prod_{k \in (\text{Nb}_i - \{j\})} \delta_{k \rightarrow i}$ 
2  $\tau(S_{i,j}) \leftarrow \sum_{C_i - S_{i,j}} \psi(C_i)$ 
3 return  $\tau(S_{i,j})$ 

```

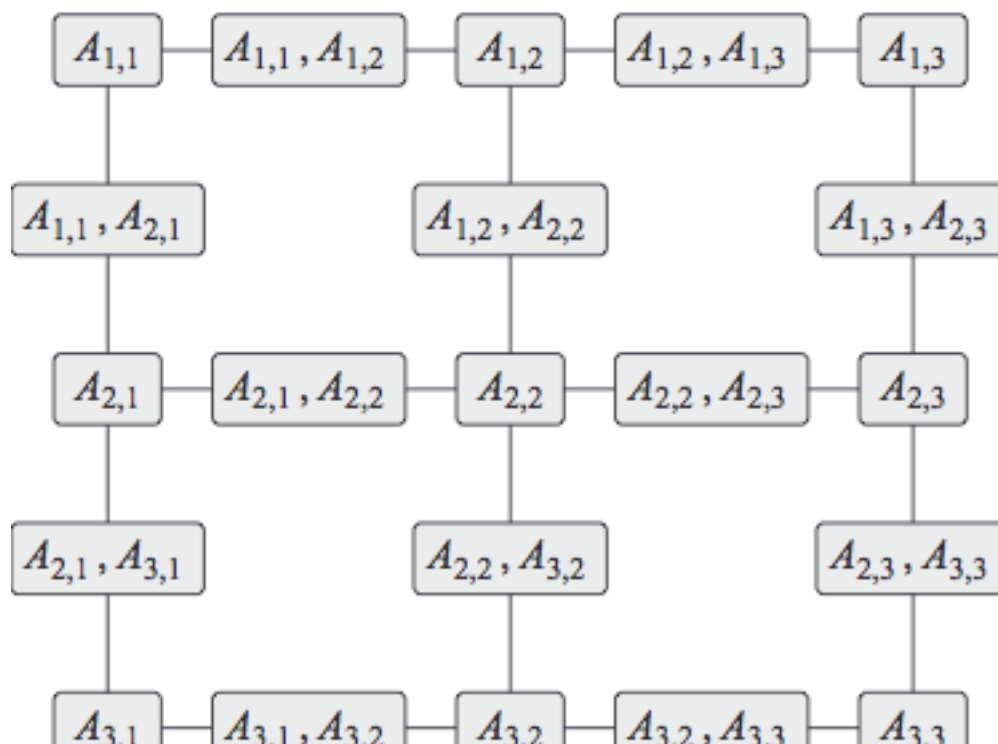
Cluster Graph Belief Propagation

- 4x4 two-dimensional grid network
- Generalized cluster graph for 3 x 3 network



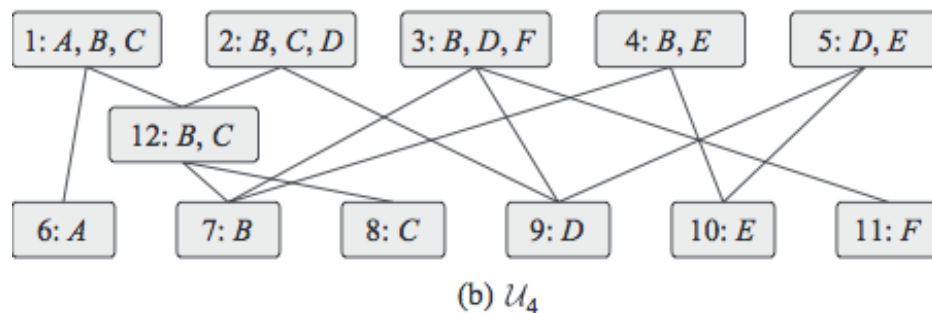
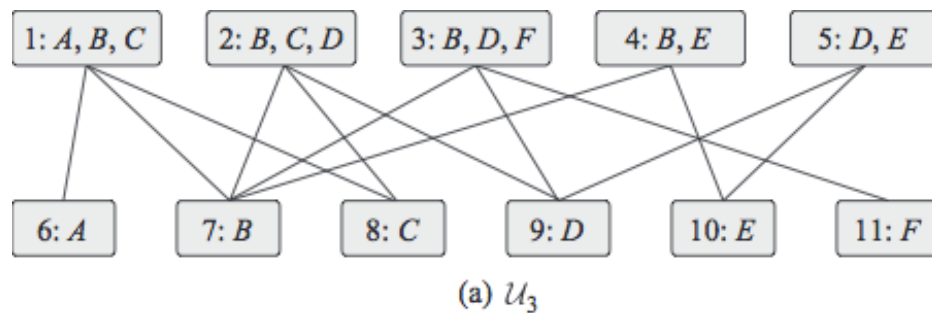
Cluster Graph for Pairwise Markov Network

- Potentials defined over nodes and edges
- For a 3x3 grid



Bethe Cluster Graph

- Generalizes pairwise clustering
- Bipartite graph; first layer of large clusters and second layer of univariate clusters



Use of Cluster Graphs

- Cluster graph belief propagation are a general purpose approximation inference method
- Can be used with trees of high width
- Many applications
 - Message decoding in communications
 - Predicting protein structure
 - Image segmentation
- Some Caveats: Need not converge, multiple optima