CS649
Sensor Networks
IP Track Lecture 6: Graphical Models

I-Jeng Wang
http://hinrg.cs.jhu.edu/wsn06/

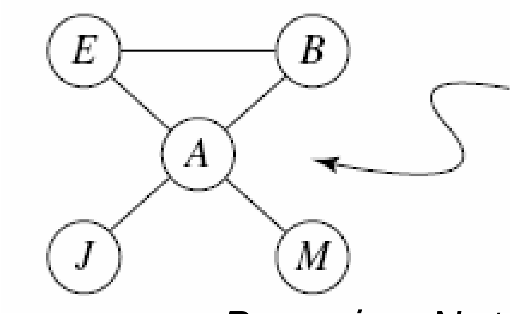
Graphical Models

- Graphical Model: A graphical representation of joint probability distribution that encodes the *conditional independency* structure of the random variables
- Types of graphical models
  - Directed graphical models: Bayesian networks
  - Undirected graphical models: factor graphs, junction trees, Markov random fields

chain rule

$$p(E,B,A,J,M) = p(E)p(B|E)p(A|E,B)p(J|E,B,A)p(M|E,B,A,J)$$
$$= p(E)p(B)p(A|E,B)p(J|A)p(M|A)$$

applying conditional independency



Bayesian Networks

$$p(X_V) = \prod_{v \in V} p(X_v \mid pa(X_v))$$

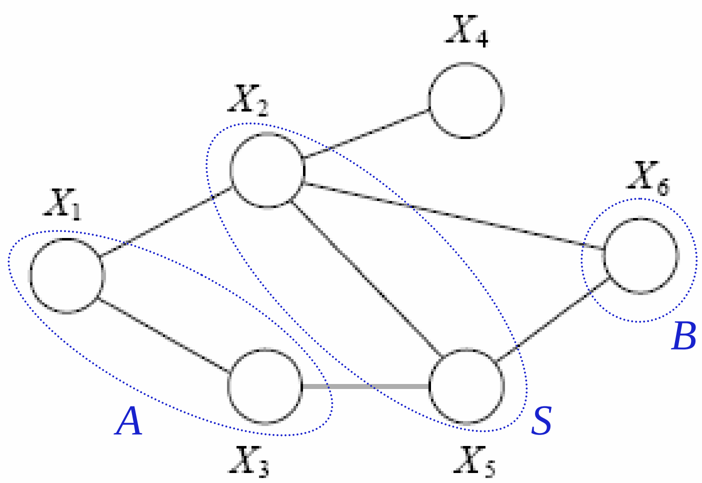
Undirected Graphical Models: Markov Random Fields

- Given an undirected graph $G=(V,E)$, each node v has an associated random variable X_v
- **Markov property**: X_A is conditional independent of X_B given X_S if S *separates* A and B, where A, B, $S \subseteq V$
- **Factorization**: The joint distribution p can be factorized into products of functions defined over all *cliques*

$$p(X_V)=\frac{1}{Z}\prod_{C\in\mathcal{C}}\psi_C(X_C)$$

compatibility function

- Hammersley-Clifford Theorem: For strictly position p, the Markov and the Factorization properties are equivalent



$$p(X_V)=\frac{1}{Z}\psi_1(X_1,X_2)\cdot\psi_2(X_1,X_3)\cdot$$
$$\psi_3(X_2,X_4)\cdot\psi_4(X_3,X_5)\cdot\psi_5(X_2,X_5,X_6)$$

Importance of Graphical Structures in Probability Models

- Provide a structure to *organize* important and difficult calculations in probability models to achieve efficient computations
- Key computation with probability models
  - Marginalization (probabilistic inference)

$$p(X_F \mid X_E), \quad V = E \bigcup F \bigcup H$$

E: Evidence (observed)
F: Inference variables
H: Hidden variables

e.g. $P(E = true \mid A = true)$

  - Modes or most probable configuration (MAP)

$$x_F^* = \arg\max_{x_F} \; p(X_F \mid X_E)$$

  - Normalization constant Z

Significance of Structure for Marginalization

$$p(x_1) = \sum_{x_2}\sum_{x_3}\sum_{x_4}\sum_{x_5}\sum_{x_6}\frac{1}{Z}\psi(x_1,x_2)\psi(x_1,x_3)\psi(x_2,x_4)\psi(x_3,x_5)\psi(x_2,x_5,x_6)$$

Direct computation lead to high complexity (r^6)!

Exploiting the distributive law and factorization structure can significantly reduce complexity (r^3)!

$$\begin{aligned}
p(x_1) &= \frac{1}{Z}\sum_{x_2}\psi(x_1,x_2)\sum_{x_3}\psi(x_1,x_3)\sum_{x_4}\psi(x_2,x_4)\sum_{x_5}\psi(x_3,x_5)\sum_{x_6}\psi(x_2,x_5,x_6) \\
&= \frac{1}{Z}\sum_{x_2}\psi(x_1,x_2)\sum_{x_3}\psi(x_1,x_3)\sum_{x_4}\psi(x_2,x_4)\sum_{x_5}\psi(x_3,x_5)m_6(x_2,x_5) \\
&= \frac{1}{Z}\sum_{x_2}\psi(x_1,x_2)\sum_{x_3}\psi(x_1,x_3)m_5(x_2,x_3)\sum_{x_4}\psi(x_2,x_4) \\
&= \frac{1}{Z}\sum_{x_2}\psi(x_1,x_2)m_4(x_2)\sum_{x_3}\psi(x_1,x_3)m_5(x_2,x_3) \\
&= \frac{1}{Z}\sum_{x_2}\psi(x_1,x_2)m_4(x_2)m_3(x_1,x_2) \\
&= \frac{1}{Z}m_2(x_1),
\end{aligned}$$

Variable Elimination
- Pick an order to eliminate
- For each variable:
- a. Push in its sum as far as possible
- b. Compute the sum

Message Passing Algorithms on Trees

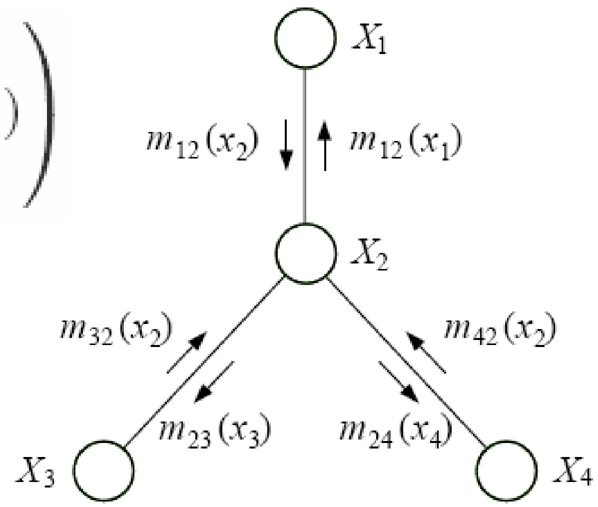
- Cliques in an undirected tree are pairs and singleton

$$p(X_V) = \prod_{i \in V} \psi(X_i) \cdot \prod_{(i,j) \in E} \psi(X_i, X_j)$$

- Sum-product algorithm (given below): Compute all marginals
- Max-product: Compute MAP configurations (replacing sums with maximizations)
- Variants with the same algebraic structure (commutative semi-ring)

$$m_{ji}(x_i) = \sum_{x_j} \left(\psi(x_j)\psi(x_i, x_j) \prod_{k \in \mathcal{N}(j) \backslash i} m_{kj}(x_j) \right)$$

$$p(x_f) \propto \psi(x_f) \prod_{e \in \mathcal{N}(f)} m_{ef}(x_f)$$



Properties of Message Passing Algorithms

- Asynchronous implementation is straightforward
- Only require “local” message passing on the graph
- Guaranteed convergence in finite number of iterations on trees (exact algorithms)
- Need not converge on graphs with cycles, however have been applied with successes
- Performance under non-stationary environments questionable

Graphical Models and Message Passing Algorithms for Sensor Networks?

- The distributed and asynchronous nature of the messaging passing seems well suited for information processing in sensor networks
- However, key questions need to be answered:
  - Mapping (constructing) a graphical model onto the sensor network
    - How do we construct a relevant probability graph given a sensor network application?
    - The "probability graph" is NOT the communication graph of the sensor network
    - Dealing with the unreliable wireless medium and low cost sensor nodes
  - Convergence issues with message passing on loopy graphs

A Robust Architecture for Distributed Inference in Sensor Networks

Mark A. Paskin

Carlos Guestrin

Jim McFadden

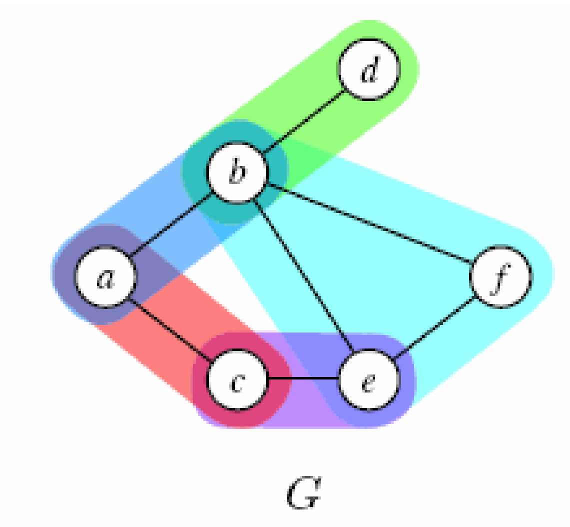
IPSN 2005

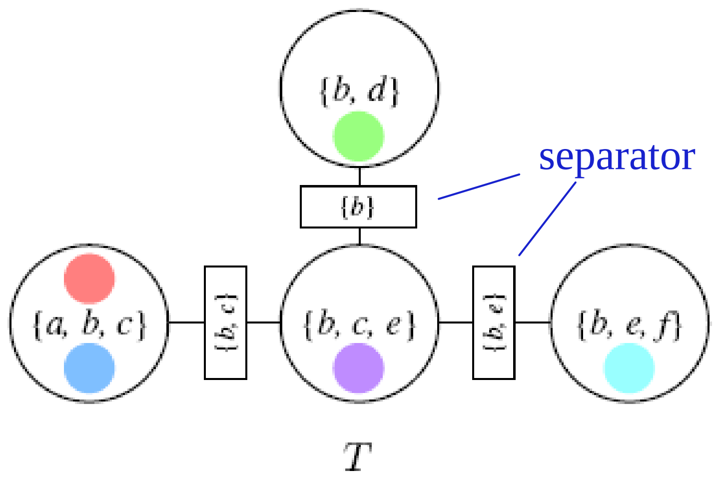
Basic Ideas

- Using junction tree structure for inference to ensure convergence of message passing algorithms
- A three-layer architecture
  - *The spanning tree layer*: Ensure the reliability of communications to support necessary information exchange in message passing algorithms; Provide a space of stable spanning tree to enable optimization of junction tree construction
  - *The junction tree layer*: Construct a tree structure for inference taking into account the underlying probability model (the graph) and the communication constraints (the spanning trees)
  - *The inference layer*: Perform appropriate message passing algorithms over the junction tree for probabilistic inference

Undirected Graphical Models and Junction Trees

- A cluster graph T is a junction tree for a graph G if it satisfies the following:
  - Single connected: exactly one path between two clusters
  - Covering: each clique in G is a subset of some cluster node in T
  - Running intersection: If two clusters B and C contain a variable X, then all clusters on the path between B and C also contains X





Junction Tree Formation

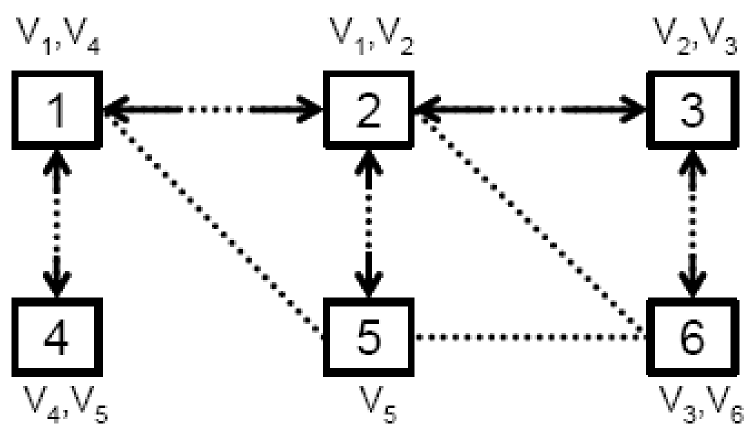
- At each sensor node, initialize a cluster node with its local variables D_i to ensure the covering property
- Update the clusters by running a message passing algorithms with reachable variable messages to ensure the running intersection property

$$\mathbf{R}_{ij} \triangleq \mathbf{D}_i \cup \bigcup_{k \in \mathrm{nbr}(i) \setminus j} \mathbf{R}_{ki}, \qquad \mathbf{C}_i \triangleq \mathbf{D}_i \cup \bigcup_{\substack{j,k \in \mathrm{nbr}(i) \\ j \neq k}} \mathbf{R}_{ji} \cap \mathbf{R}_{ki}.$$

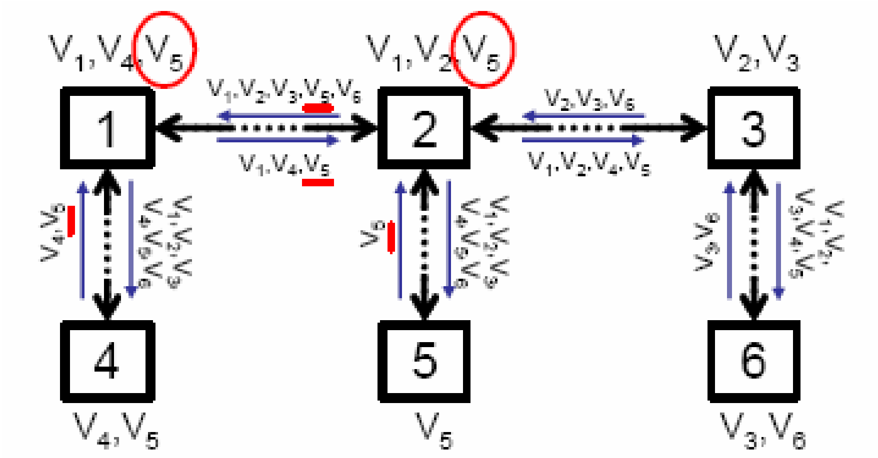
- Optimize the junction tree by minimizing the computing (sizes of clusters) and communication (sizes of the separators) costs

$$\sum_{i=1}^{N} \left[\alpha_i(\mathbf{C}_i) + \sum_{j \in \mathrm{nbr}(i)} \beta_{ij}(\mathbf{S}_{ij}) \right]$$

From a Spanning Tree to a Junction Tree



A Spanning Tree with Initial Local Variables Assignment



A Constructed Junction Tree After Message Passing

Some Open Research Issues

- How do we obtain the graphical model (the compatibility functions or factors)?
- How do we compute and communicate the inference results to where it is needed? (not necessarily co-located with the node that contains the relevant variables)
- Is it possible to derive a more robust message passing algorithm that can take advantage of the wireless local wireless broadcast capability in sensor network?