

# Chapter 8 of Bishop's Book: “Graphical Models”

# Review of Probability

- Probability density over possible values of  $x$

$$p(x)$$

- Used to find probability of  $x$  falling in some range

$$P(x \in (a, b)) = \int_b^a p(x) dx$$

- For continuous variables, the probability of a single value is technically 0!
- For discrete values, called probability mass

# Review of Probability

- The density function must satisfy two conditions

$$p(x) \geq 0$$

$$\int_{-\infty}^{\infty} p(x) = 1$$

# Two important rules

- Sum Rule:

$$\int_y p(x, y) dy = p(x)$$

- $p(x)$  is marginal probability, obtained by marginalizing out  $y$
- Product Rule

$$p(x, y) = p(y|x)p(x)$$

# Graphical Models

- Consider the distribution

$$p(a, b, c)$$

- Can be rewritten as

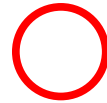
$$p(a, b, c) = p(c|a, b)p(a, b)$$

- Which can be rewritten as

$$p(a, b, c) = p(c|a, b)p(b|a)p(a)$$

# Making a Graphical Model

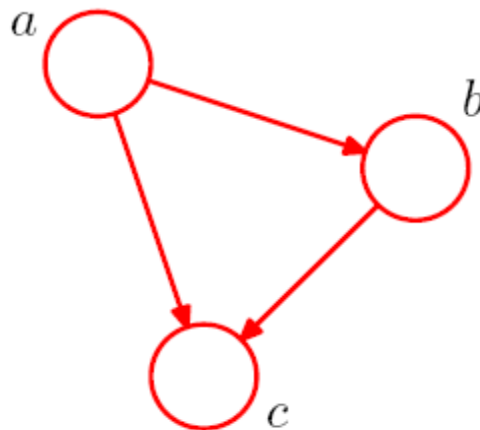
- Introduce one node per random variable



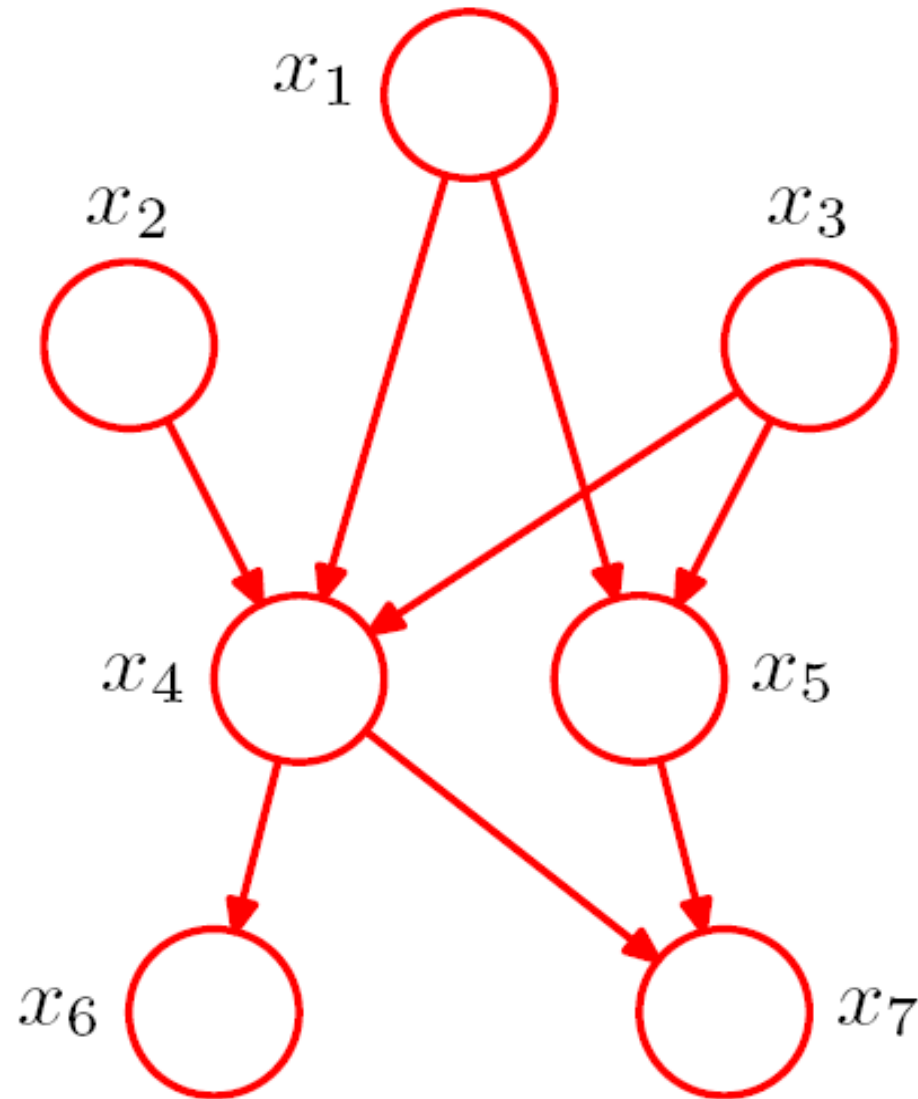
- Use the distribution

$$p(c|a, b)p(b|a)p(a)$$

- Add one edge per conditional distribution



# What's the distribution for this graph?



$$p(x_1)p(x_2)p(x_3)p(x_4|x_1, x_2, x_3)p(x_5|x_1, x_3)p(x_6|x_4)p(x_7|x_4, x_5)$$

# Big Advantage: Many Less Parameters

$$p(x_1)p(x_2)p(x_3)p(x_4|x_1, x_2, x_3)p(x_5|x_1, x_3)p(x_6|x_4)p(x_7|x_4, x_5)$$

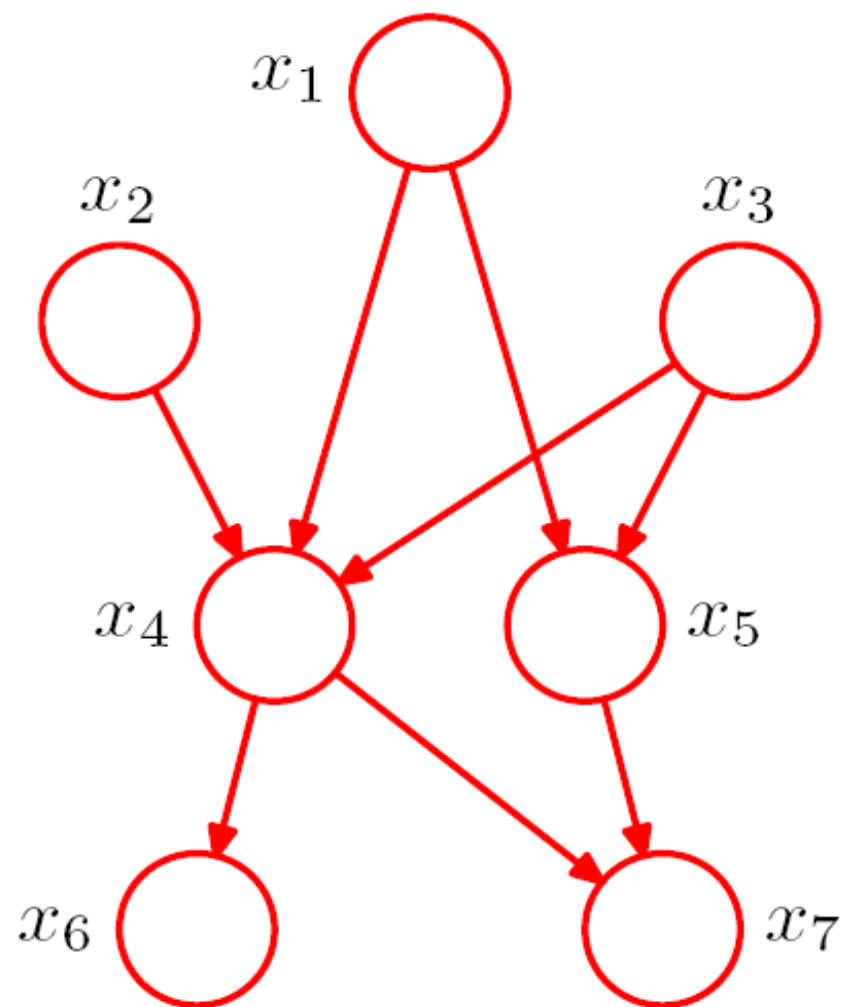
- Assume each variable can take **K** states, how many numbers do we need to specify this distribution
- What if we just wanted to express it as a giant joint distribution

# What do you do with distributions?

- Draw Samples
- Infer marginal distributions over variables

# Ancestral Sampling

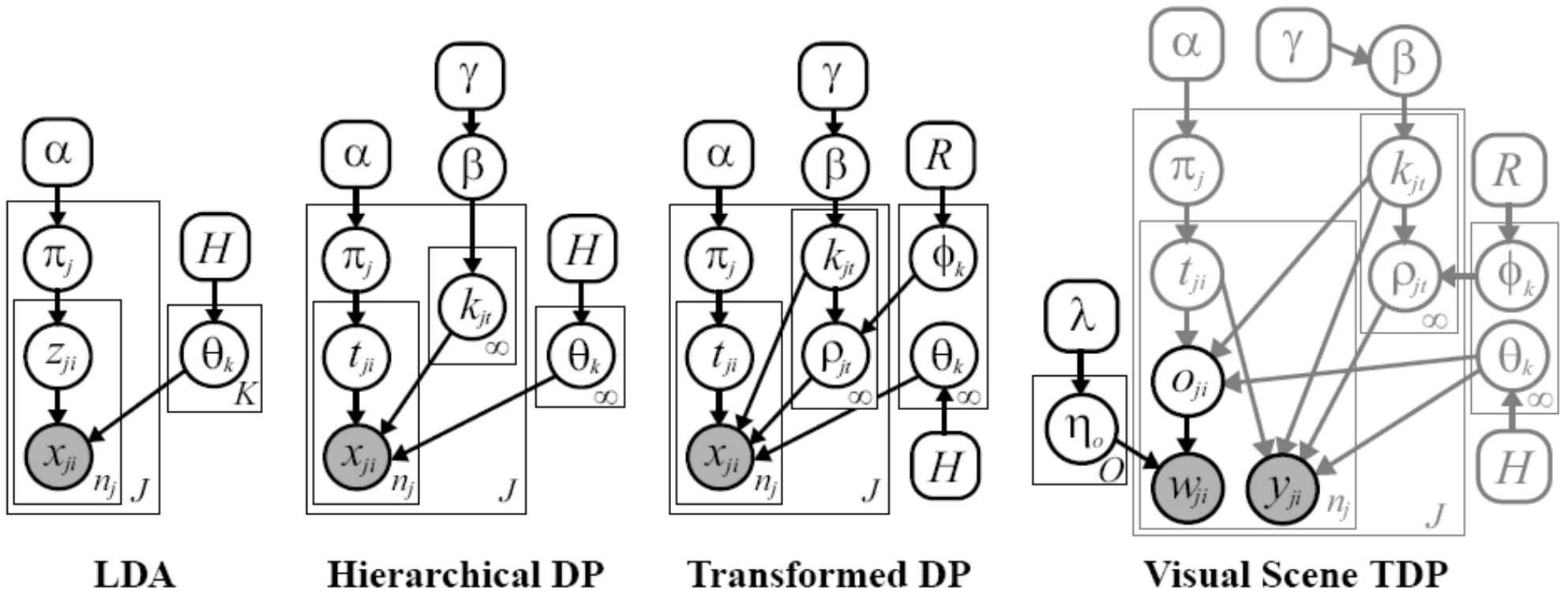
1. Start at the low numbered nodes and draw samples
2. Work your way through the graph, sampling from conditional distributions



# Generative Models

- The graph, along with the ancestral sampling method, is a model of how the data is generated
- The model expresses a *causal* process for generating the data
- These models are often called *generative* models
- They show how to generate the data

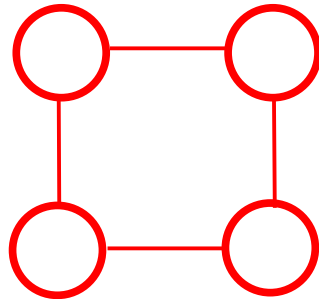
# Example of a Generative Model of Images (of Objects)



(From Sudderth, Torralba, Freeman, and Willsky - NIPS05)

# Undirected Models

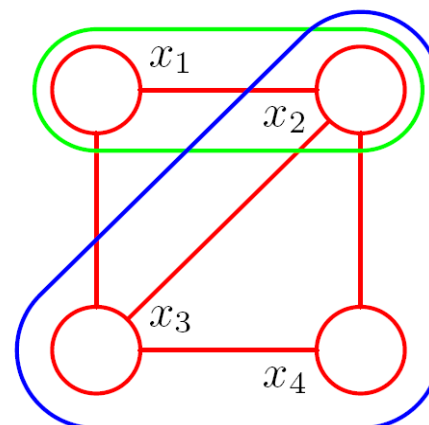
- Directed models are useful, but have some complicating limitations
  - Determining conditional independence can be hard
- Not all relationships can be expressed causally
- We can drop the arrows to make an undirected model



- Also called Markov Network or Markov Random Field

# Undirected Graphical Models

- To understand undirected models, we need to introduce the notion of a clique
  - Subset of nodes
  - Links between all nodes in subset
- *And Maximal Cliques*
  - If you add nodes to the clique, it is no longer a clique



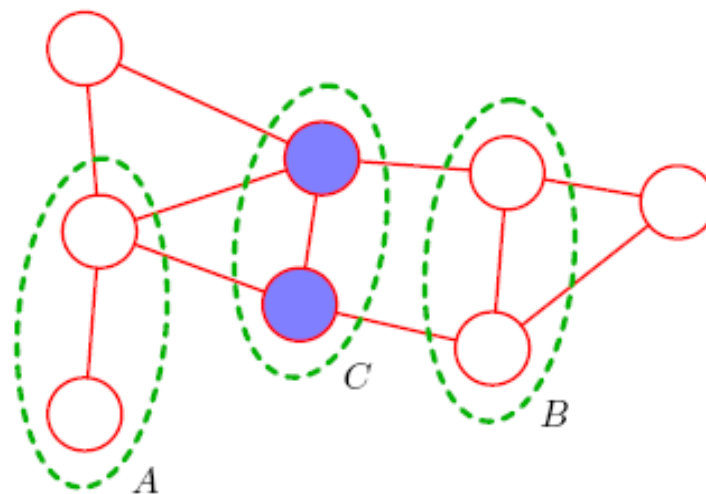
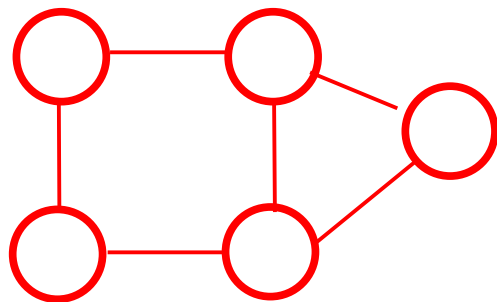
# Conditional Independence

$$p(a|b, c) = p(a|c).$$

or

$$\begin{aligned} p(a, b|c) &= p(a|b, c)p(b|c) \\ &= p(a|c)p(b|c). \end{aligned}$$

# Conditional Independence in an MRF



- Conditioned on its neighbors a node is conditionally independent from the rest of the nodes in the graph

# Representing the Distribution in an Undirected Graph

- The form of the distribution is

$$p(\mathbf{x}) = \frac{1}{Z} \prod_C \psi_C(\mathbf{x}_C)$$

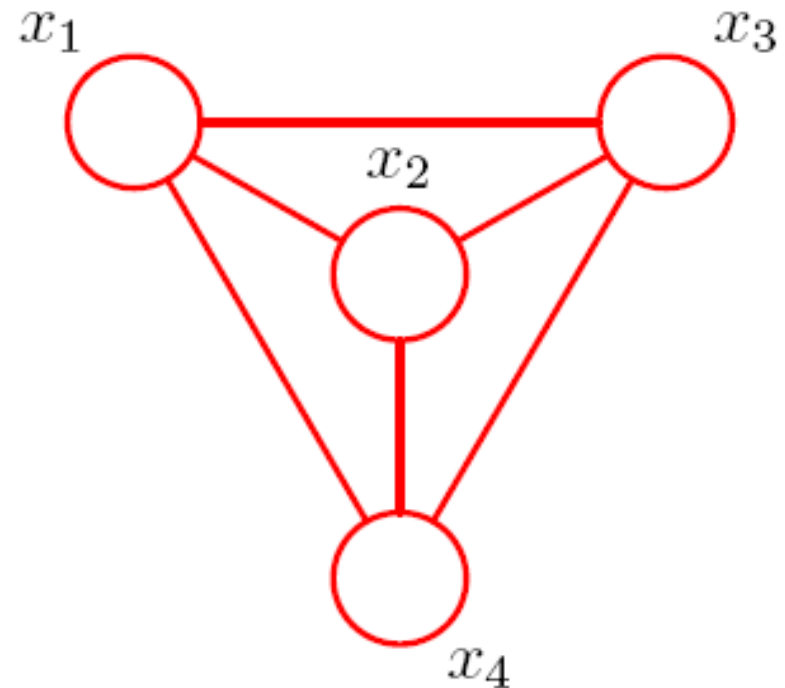
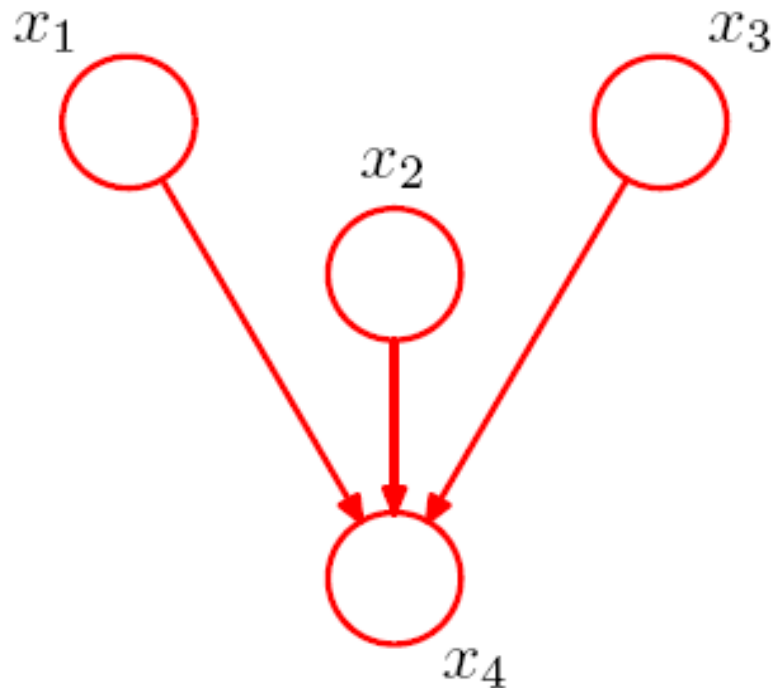
- The distribution is formed from potential functions on the maximal cliques of the graph
  - They represent compatibility between the states of different variables

$$\psi_C(\mathbf{x}_C)$$

- $Z$  is a normalization constant and is also known as the partition function

# Converting a directed model to an undirected model

- You “moralize” the graph
- Marry the parents
- Can then use inference techniques for undirected graphs

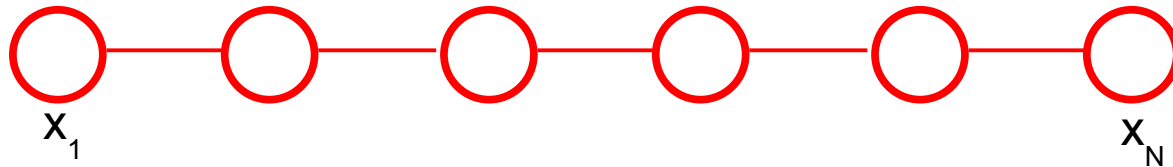


# Inference in Graphical Models

- Let's say that I've analyzed the problem
- Have designed a graphical model that relates the observations to quantities that I would like to estimate
- Get my observations
- How do estimate the hidden, or *latent*, quantities?

# Inference and Conditional Independence

- Conditional independence is important for MRF models because it makes inference much easier.
- Consider a chain



- Task: Find the marginal distribution of some variable  $x_n$

# Naïve, Slow Way

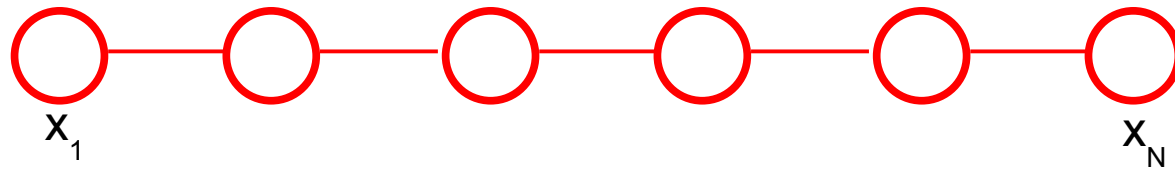
- Use the sum rule and do the sums

$$p(x_n) = \sum_{x_1} \cdots \sum_{x_{n-1}} \sum_{x_{n+1}} \cdots \sum_{x_N} p(\mathbf{x}).$$

- Implication:
  - If you have  $K$  states per node and  $N$  nodes, this will take  $K^N$  operations.

# Taking advantage of the structure

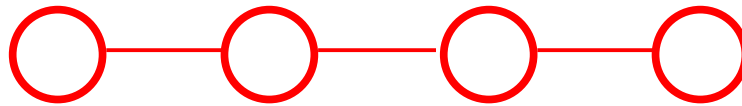
- The distribution of this chain:



- Is

$$p(\mathbf{x}) = \frac{1}{Z} \psi_{1,2}(x_1, x_2) \psi_{2,3}(x_2, x_3) \cdots \psi_{N-1,N}(x_{N-1}, x_N).$$

# 4 Node Example



$$p(x_2) = \sum_{x_1} \sum_{x_2} \sum_{x_3} \sum_{x_4} \frac{1}{Z} \psi_{1,2}(x_1, x_2) \psi_{2,3}(x_2, x_3) \psi_{3,4}(x_3, x_4)$$

- Start Re-Arranging Sums

$$p(x_2) = \sum_{x_1} \sum_{x_2} \sum_{x_3} \frac{1}{Z} \psi_{1,2}(x_1, x_2) \psi_{2,3}(x_2, x_3) \sum_{x_4} \psi_{3,4}(x_3, x_4)$$

# Re-Arranging Sums

$$p(x_2) = \sum_{x_1} \sum_{x_2} \sum_{x_3} \frac{1}{Z} \psi_{1,2}(x_1, x_2) \psi_{2,3}(x_2, x_3) \sum_{x_4} \psi_{3,4}(x_3, x_4)$$

- Make Substitution

$$m_3(x_3) = \sum_{x_4} \psi_{3,4}(x_3, x_4)$$

- Which leads to

$$p(x_2) = \sum_{x_1} \sum_{x_2} \sum_{x_3} \frac{1}{Z} \psi_{1,2}(x_1, x_2) \psi_{2,3}(x_2, x_3) m_3(x_3)$$

# Do it again

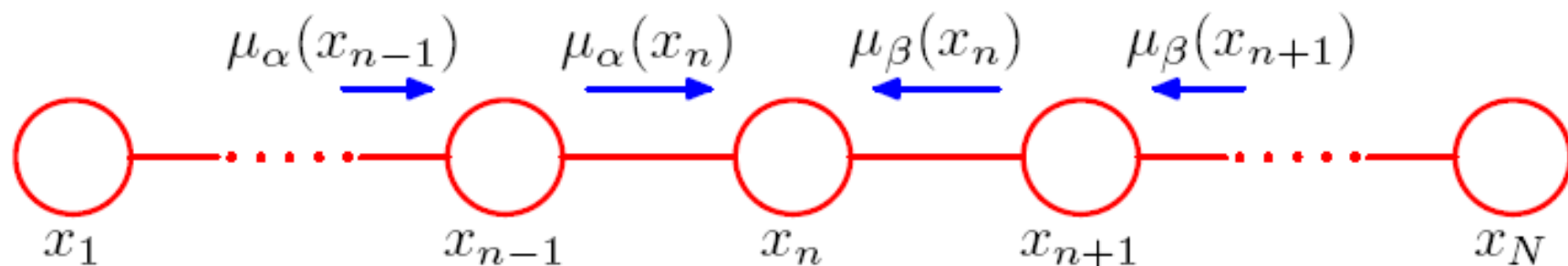
$$p(x_2) = \sum_{x_1} \sum_{x_2} \frac{1}{Z} \psi_{1,2}(x_1, x_2) \sum_{x_3} \psi_{2,3}(x_2, x_3) m_3(x_3)$$

- To Get

$$p(x_2) = \sum_{x_1} \sum_{x_2} \frac{1}{Z} \psi_{1,2}(x_1, x_2) m_2(x_2)$$

How many operations did this  
require?

# For a general chain



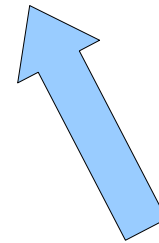
$$p(x_n) = \frac{1}{Z}$$

$$\left[ \underbrace{\sum_{x_{n-1}} \psi_{n-1,n}(x_{n-1}, x_n) \cdots \left[ \sum_{x_2} \psi_{2,3}(x_2, x_3) \left[ \sum_{x_1} \psi_{1,2}(x_1, x_2) \right] \right]}_{\mu_\alpha(x_n)} \cdots \right] \cdot \left[ \underbrace{\sum_{x_{n+1}} \psi_{n,n+1}(x_n, x_{n+1}) \cdots \left[ \sum_{x_N} \psi_{N-1,N}(x_{N-1}, x_N) \right]}_{\mu_\beta(x_n)} \cdots \right] \cdot \quad (8.52)$$

# One more type of graphical model

- Now, we'll write the distribution as

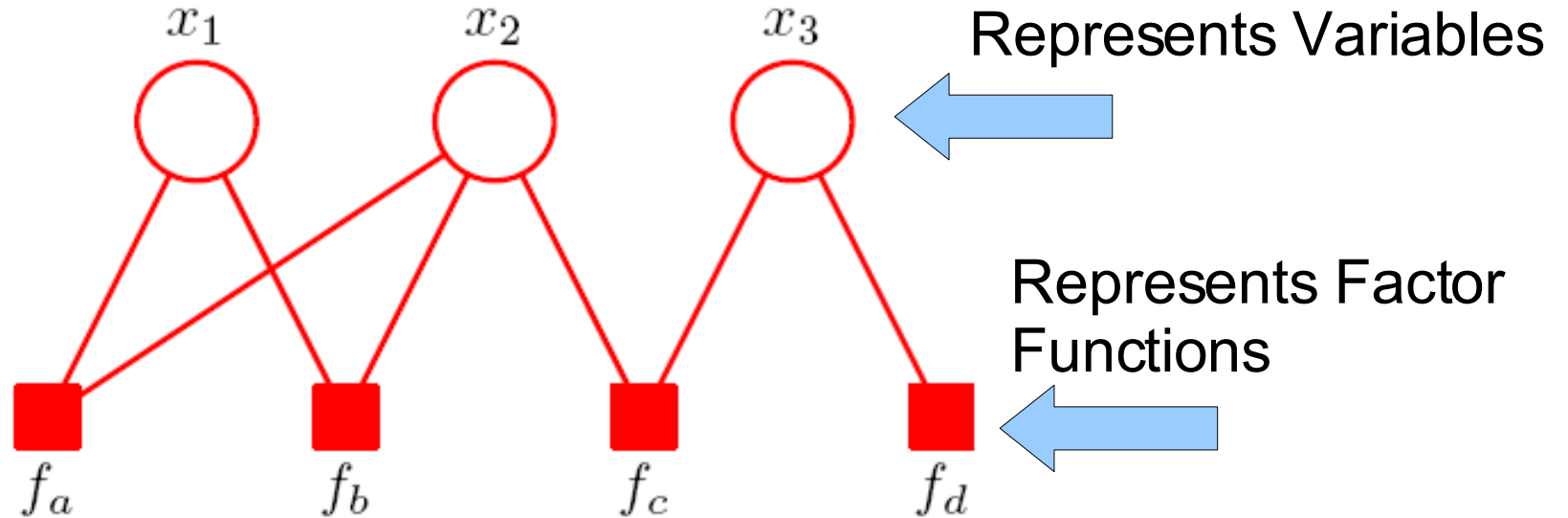
$$p(\mathbf{x}) = \prod_s f_s(\mathbf{x}_s)$$



Called a factor

# The Graph

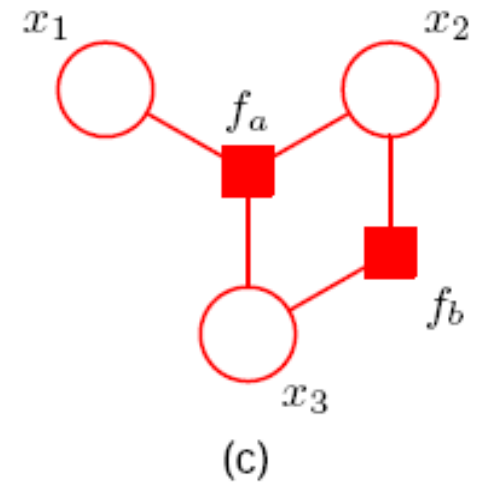
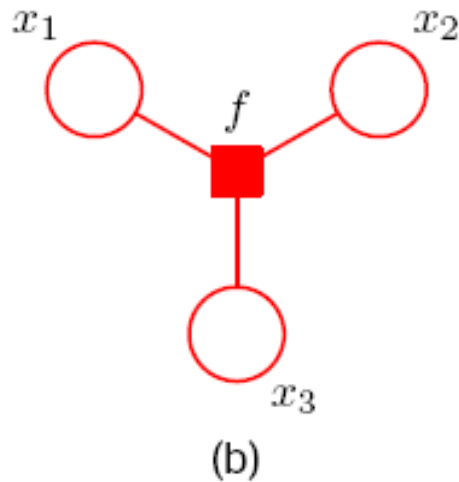
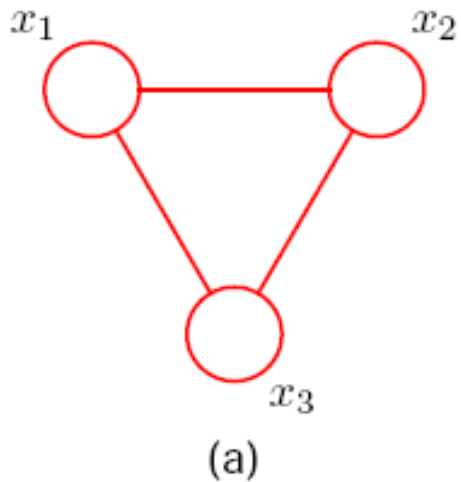
- The graphs are bipartite graphs



$$p(\mathbf{X}) = f_a(x_1, x_2) f_b(x_1, x_2) f_c(x_2, x_3) f_d(x_3).$$

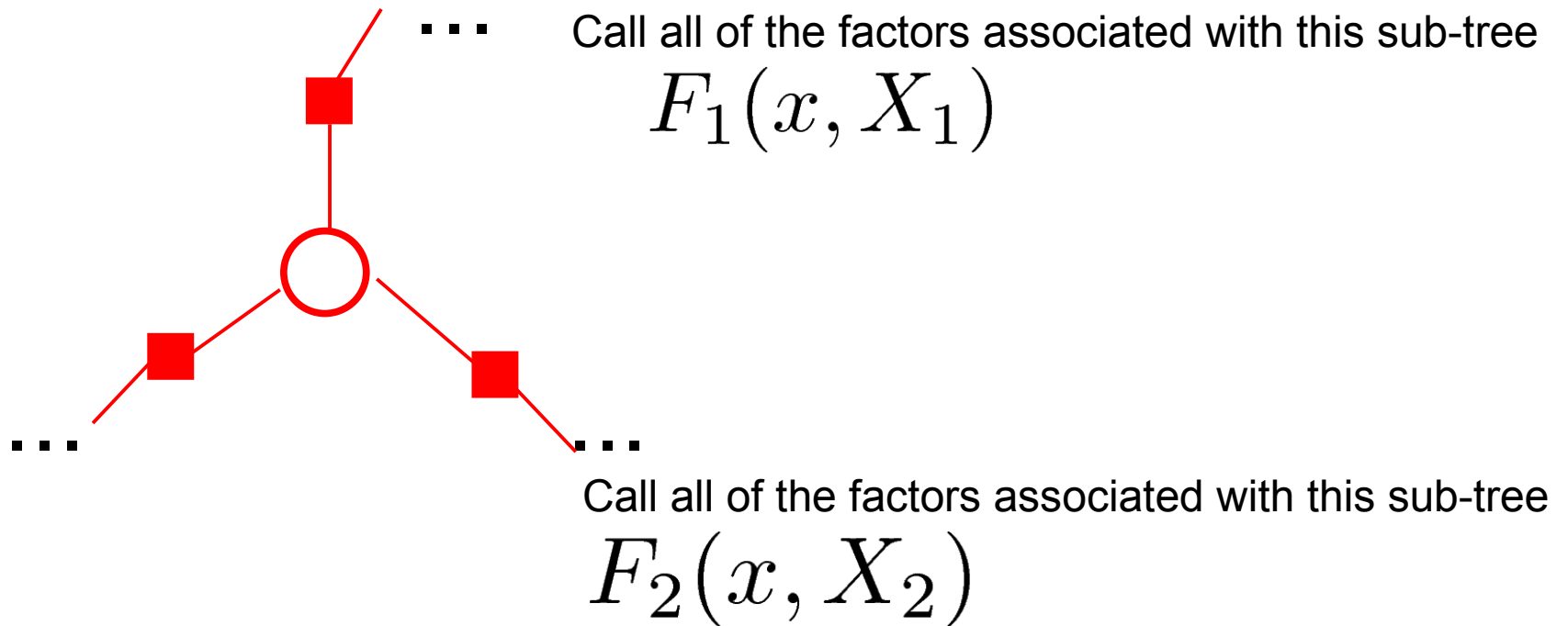
- Advantage: The graph can express distributions more exactly

# Example



- A factor graph can represent any directed or undirected graphical model

# Deriving the sum-product algorithm



Goal calculate the marginal probability of some variable  $x$

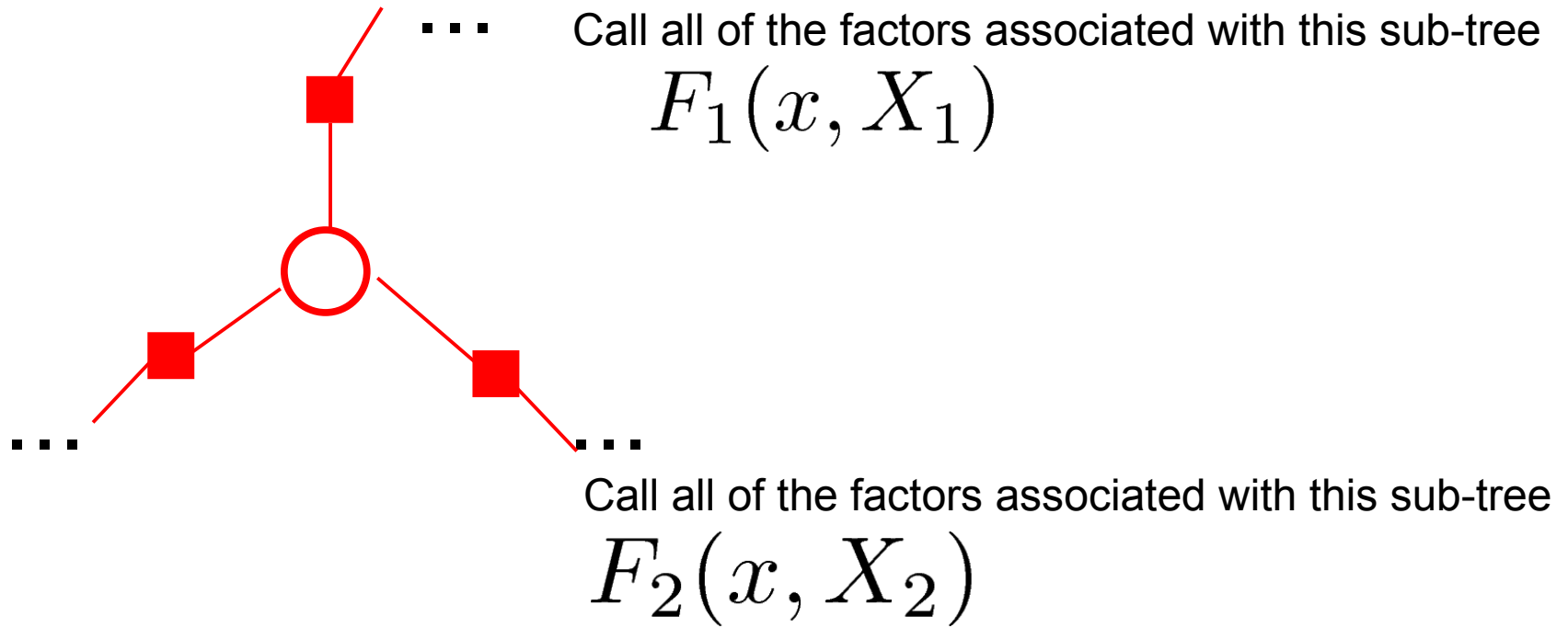
$$p(x) = \sum_{\mathbf{x} \setminus x} p(\mathbf{x})$$

Bold font represents the collection of all variables

This means all variables but  $x$



# Deriving the sum-product algorithm



$$p(\mathbf{x}) = \prod_{s \in \text{ne}(x)} F_s(x, X_s)$$

Combining the two (and flipping order)

$$\begin{aligned} p(x) &= \prod_{s \in \text{ne}(x)} \left[ \sum_{X_s} F_s(x, X_s) \right] \\ &= \prod_{s \in \text{ne}(x)} \mu_{f_s \rightarrow x}(x). \end{aligned}$$

$$\mu_{f_s \rightarrow x}(x) \equiv \sum_{X_s} F_s(x, X_s)$$

# We can also factor the $F$ variables

$$F_s(x, X_s) = f_s(x, x_1, \dots, x_M) G_1(x_1, X_{s1}) \dots G_M(x_M, X_{sM})$$

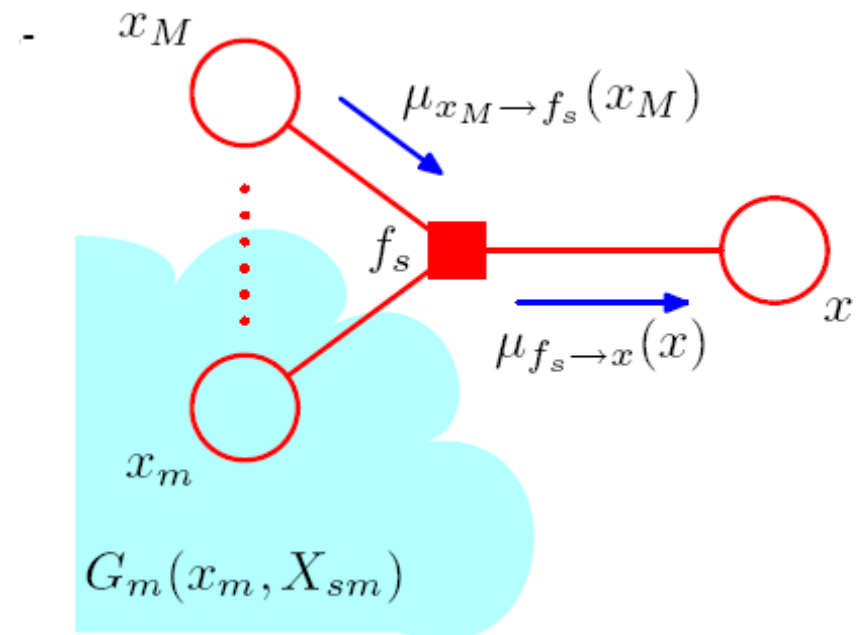
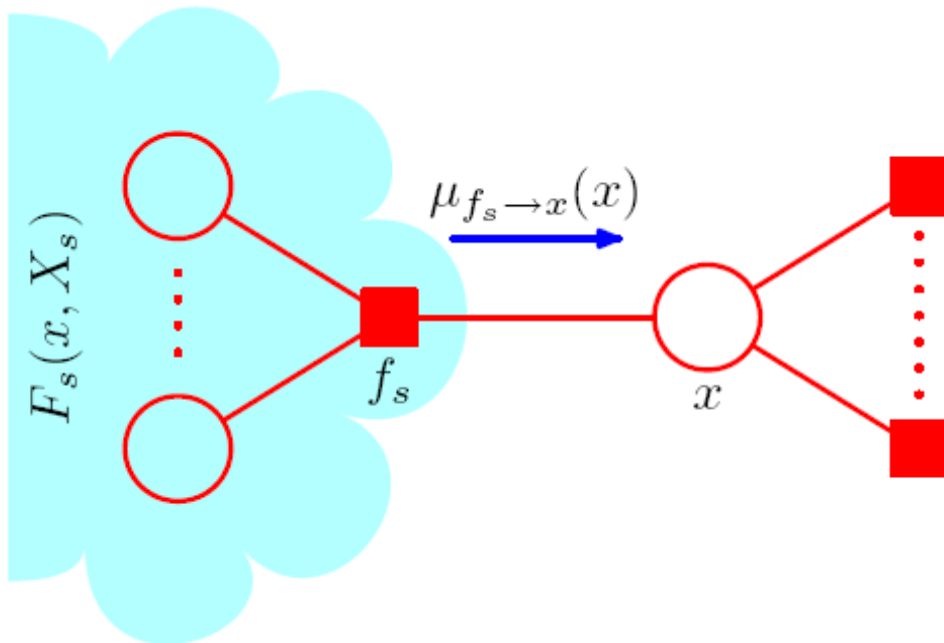
$$\mu_{x_m \rightarrow f_s}(x_m) \equiv \sum_{X_{sm}} G_m(x_m, X_{sm}).$$

$$G_m(x_m, X_{sm}) = \prod_{l \in \text{ne}(x_m) \setminus f_s} F_l(x_m, X_{ml})$$

$$\begin{aligned} \mu_{x_m \rightarrow f_s}(x_m) &= \prod_{l \in \text{ne}(x_m) \setminus f_s} \left[ \sum_{X_{ml}} F_l(x_m, X_{ml}) \right] \\ &= \prod_{l \in \text{ne}(x_m) \setminus f_s} \mu_{f_l \rightarrow x_m}(x_m) \end{aligned}$$

# Called the sum-product algorithm (or belief propagation)

- Two steps:
  - Variables pass messages to factors
  - Factors pass messages to variables



# Can use similar message-passing scheme with undirected models

- Pass messages between nodes
- The factor graph representation is nice because it is much easier to deal with non-pairwise nodes.
  - If your clique potentials involve more than two nodes, the messages are complicated

# What if you want the labeling with the highest probability?

- Use max-product
  - or max-sum in the log-domain

$$\begin{aligned}\mu_{x_n \rightarrow f_{n,n+1}}(x_n) &= \mu_{f_{n-1,n} \rightarrow x_n}(x_n) \\ \mu_{f_{n-1,n} \rightarrow x_n}(x_n) &= \max_{x_{n-1}} \left[ \ln f_{n-1,n}(x_{n-1}, x_n) + \mu_{x_{n-1} \rightarrow f_{n-1,n}}(x_n) \right]\end{aligned}$$

- Basically, replace the sum in sum-product with max

# IMPORTANT: These algorithms only work if the graph has a tree structure

- If the graph has loops, it is NP-Complete (see Boykov, Veksler, and Zabih PAMI-2000)
- Problem:
  - Many interesting models in vision have loops
- One solution:
  - Ignore the @^\$&^\$ loops and run it anyway

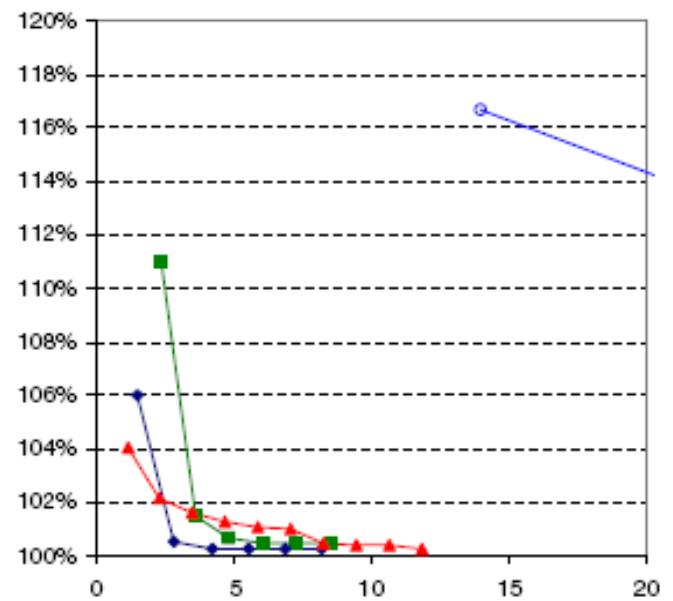
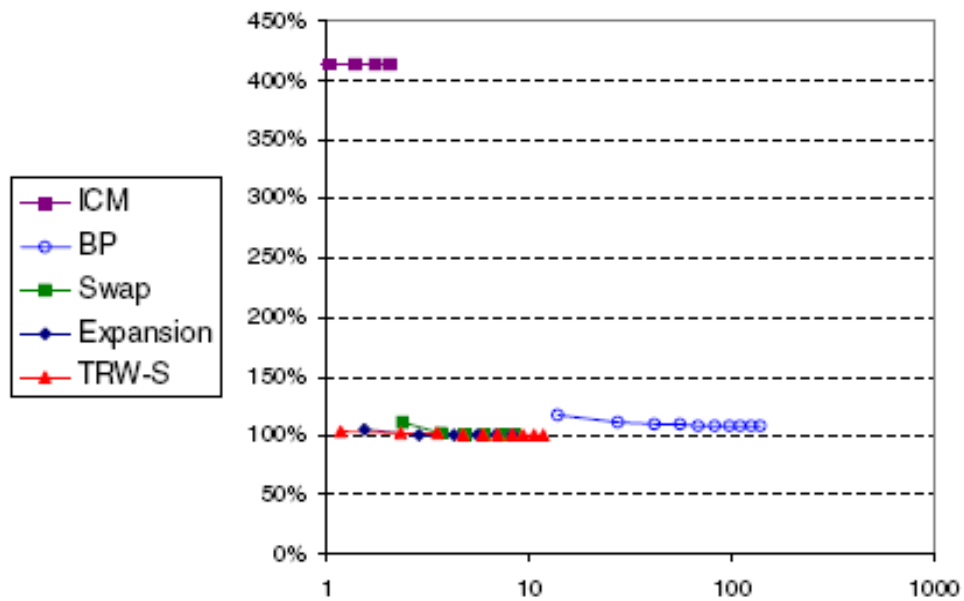
# Surprisingly, this works well

- Recent comparison of BP with other algorithms on several vision problems

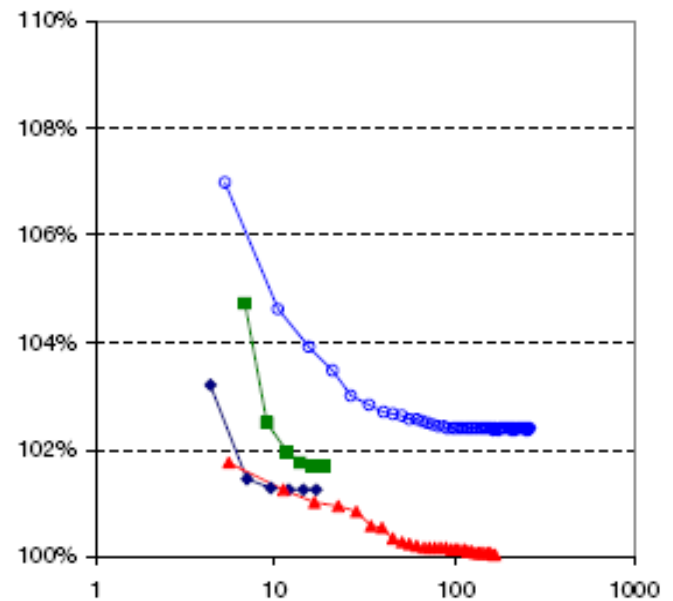
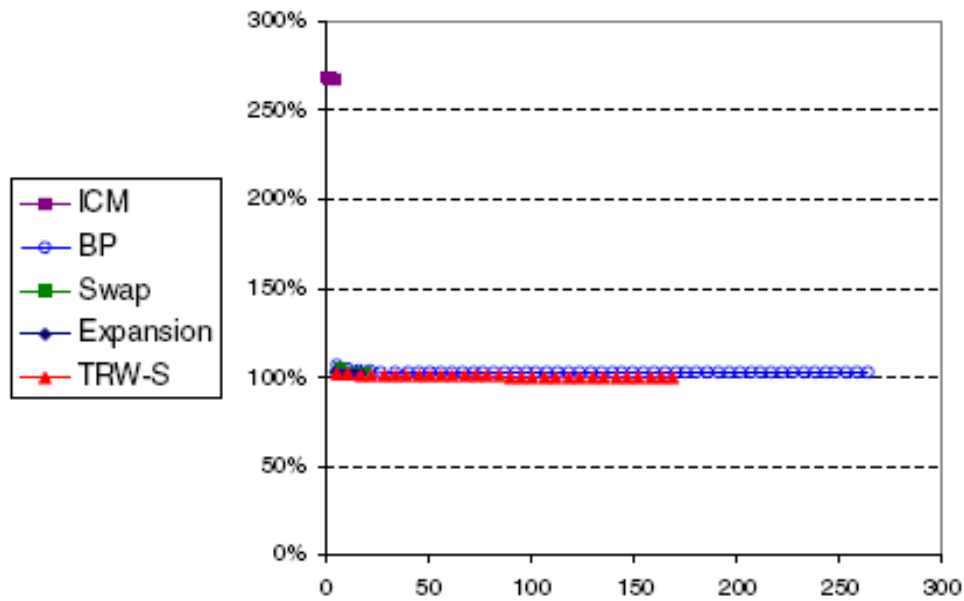
A Comparative Study of Energy Minimization Methods for Markov Random Fields  
R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother.

In Ninth European Conference on Computer Vision (ECCV 2006), volume 2, pages 19-26, Graz, Austria, May 2006.

- This paper also has references to a lot of work that relies on MRF models



“Tsukuba” energy, with the truncated  $L_1$  distance for  $V$



“Venus” energy, with the truncated  $L_2$  distance for  $V$

# To Add

- Definition of Boltzman
- Hammersley-Clifford
- Relationship of undirected to Directed