概率图模型理论及应用

Theory and Applications of Probabilistic Graphical Models

(Lesson 8)

欧智坚

清华大学电子工程系

Addr: 罗姆楼 6-104

Tel: 62796193

Email: ozj@tsinghua.edu.cn

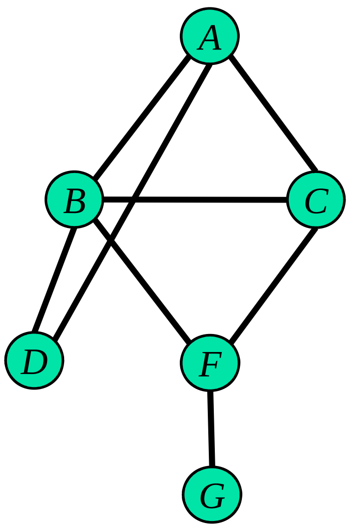
课程章节

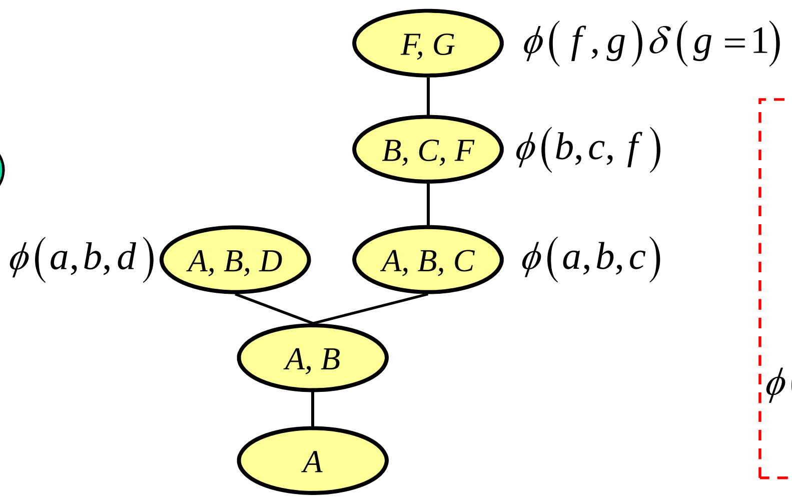
- 第一章 引言（**1**）
- 第二章 图模型的表示理论（**3**）
  - **DGM-UGM**
  - **Semantics**
  - **HMM-CRF**
- 第三章 图模型的推理理论（**6**）
  - **精确推理：variable-elimination，cluster-tree，triangulate**
  - 连续变量：Kalman
  - 采样近似：sampling
  - 变分近似：variational
- 第四章 图模型的学习理论（**3**）
  - **参数学习：maxlikelihoodEstimate，**BayesEstimate
  - 结构学习：StructureLearning
- 第五章 一个综合例子（1）

编译(寻找最优的树分解) 与 计算(树消除算法) 分离

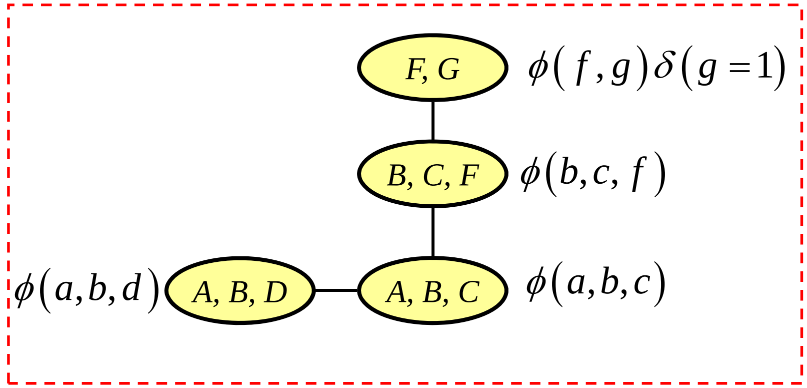
连乘积 $\xrightarrow{\text{编译}}$ 树分解　　一个连乘积的不同树分解

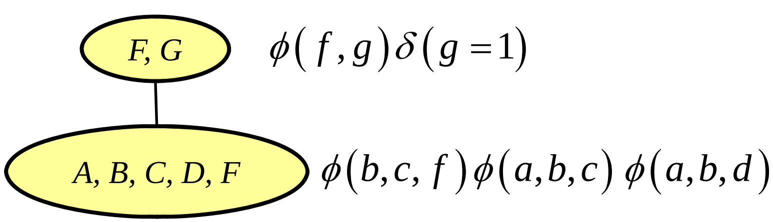
连乘积G: $\phi(a,b,c)\phi(b,c,f)\phi(a,b,d)\phi(f,g)\delta(g=1)$

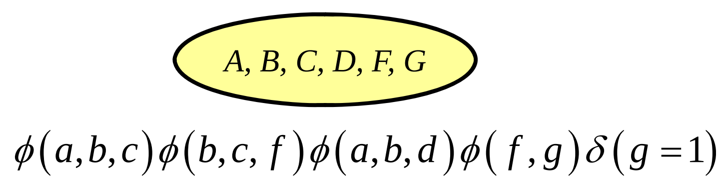




junction-tree (JT)







lesson8_triangulate

JT在什么意义下是最优？

Cluster-tree discussion

JT有什么性质？

Chordal graph, junction-tree

如何构造JT？

From primal graph to junction-tree

Cluster-tree elimination: complexity

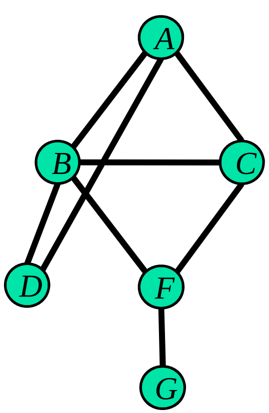
树消除算法的复杂度 = 所有边两个方向的消息计算的复杂度

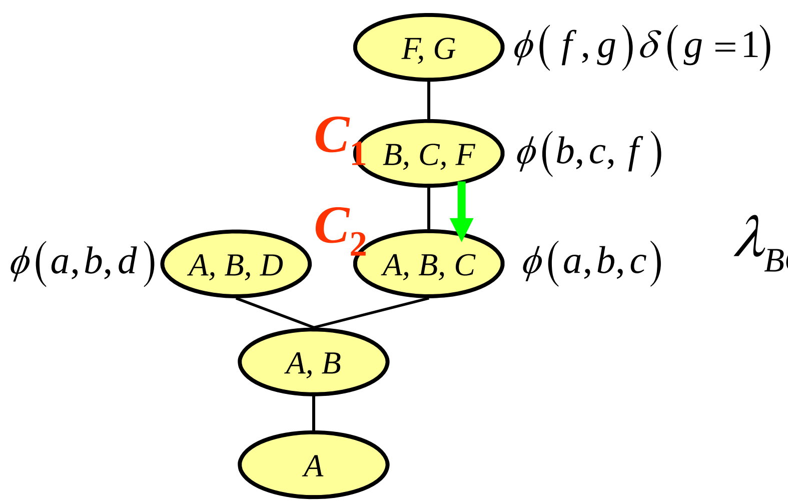
$$\lambda_{C_1 \to C_2} = \Downarrow_{sep(C_1,C_2)} \left\{ C_1\text{里的函数} \cdot \prod_{Z \in C_1\text{ 的邻居}\setminus C_2} \lambda_{Z \to C_1} \right\}$$

一次消息计算的时间复杂度 加法次数： $r^{|\chi(C_1)|}$ cluster的规模

乘法次数： $\left(|\psi(C_1)| + |ne(C_1)| - 1 - 1\right) r^{|\chi(C_1)|}$

好的cluster-tree: <u>cluster的规模</u>尽可能小，<u>cluster数目</u>尽可能少





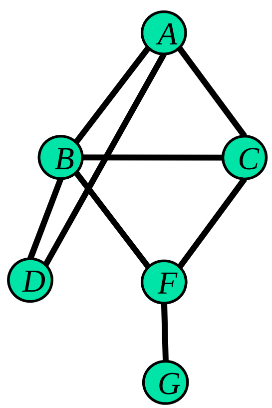
$$\lambda_{BCF \to ABC} =$$

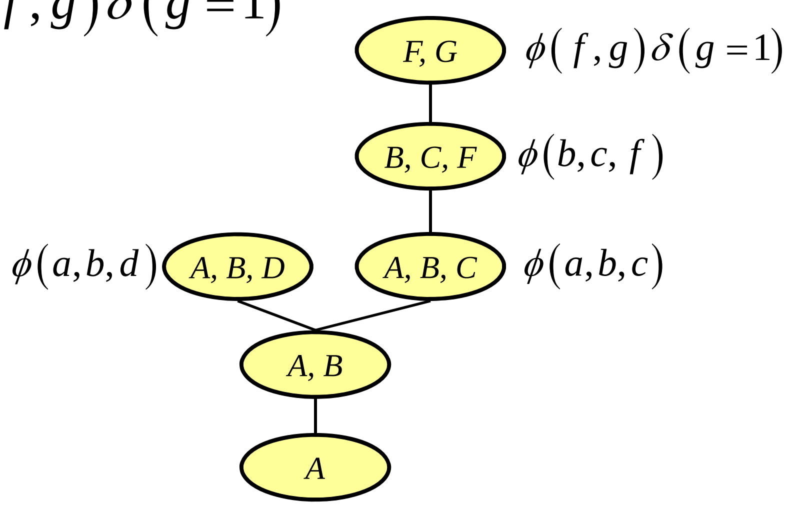
一个树分解中的cluster的性质

- 一个连乘积的素图表示如下定义:
  - 以变量为结点;
  - 在每一个局部函数的自变量间添加边。

- **性质: 素图的任一个最大簇一定包含于某个cluster**
  - If 存在以该最大簇为自变量集的局部函数
  - Otherwise 不存在以该最大簇为自变量集的局部函数

连乘积: $\phi(a,b,c)\phi(b,c,f)\phi(a,b,d)\phi(f,g)\delta(g=1)$



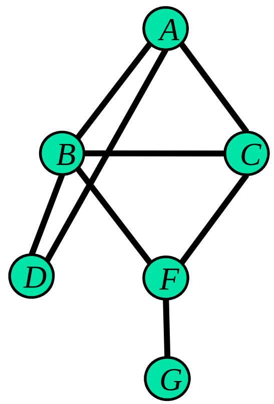


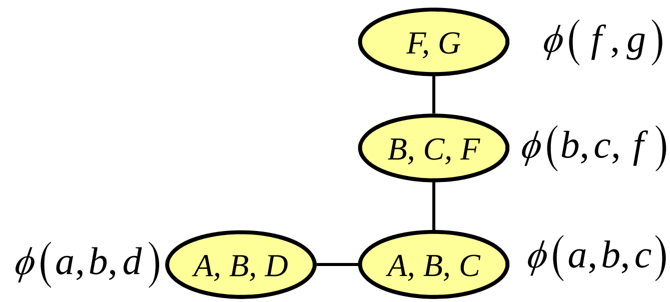
Cluster-tree discussion: existence

好的cluster-tree: cluster的规模尽可能小，cluster数目尽可能少

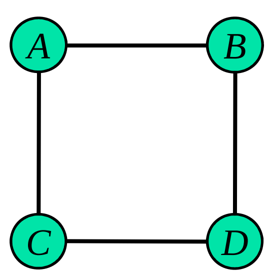
- 以素图的最大簇作为cluster，建一棵cluster-tree ?

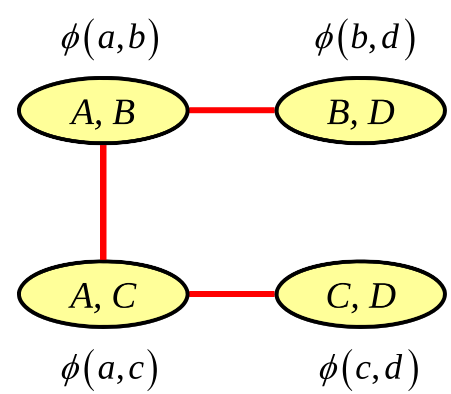
连乘积：$\phi(a,b,c)\phi(b,c,f)\phi(a,b,d)\phi(f,g)\delta(g=1)$





连乘积：$\phi(a,b)\phi(a,c)\phi(c,d)\phi(b,d)$





lesson8_triangulate

Cluster-tree discussion

Chordal graph, junction-tree

From primal graph to junction-tree

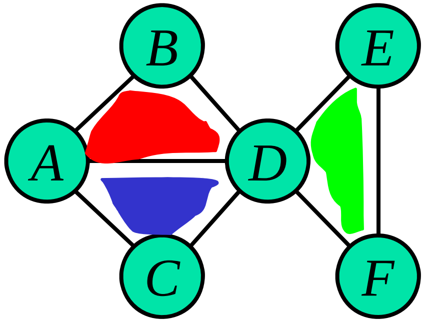
JT在什么意义下是最优？

JT有什么性质？

如何构造JT？

Basic concepts – junction-tree

- 设 C 为素图 G 的最大簇集合，如果存在以 C 作为cluster集合的一棵cluster-tree，则称图 G 存在一棵连接树。
- 这棵cluster-tree 称为是素图G的一棵连接树。



定理：素图 G 存在一棵连接树

$\Leftrightarrow$ 素图 G 是弦图

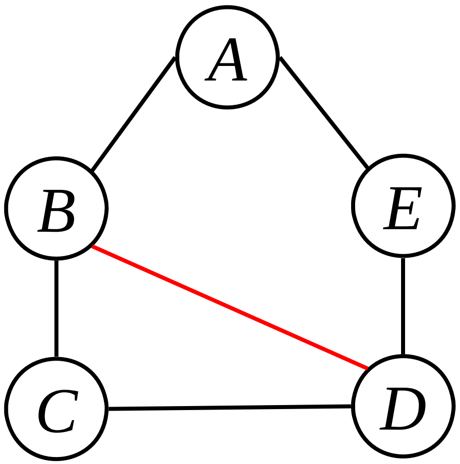
$\Leftrightarrow$ 素图 G 容许一个完美编号

$\Leftrightarrow$ 素图 G 的最大簇容许RIP排序

各有用途

Basic concepts – chord

- 环的一条弦：环上不接续的两个结点 (α_i, α_j) 间的一条边.



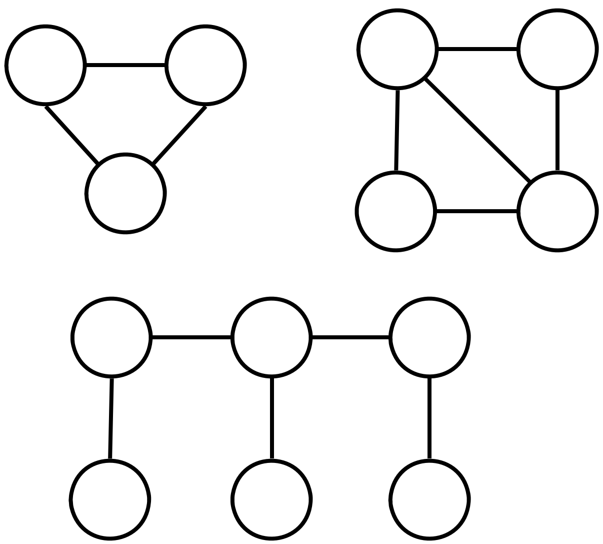
$B\sim D$ is a chord of the 5-cycle (A, B, C, D, E)

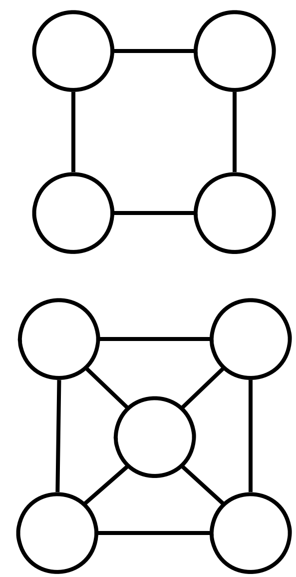
Basic concepts – chordal graph

- 一个素图 G 为弦图，如果图上每一个长度 $\geq$ 4的环都有一条弦

定理：素图 G 存在一棵连接树 $\Leftrightarrow$ 素图 G 是弦图

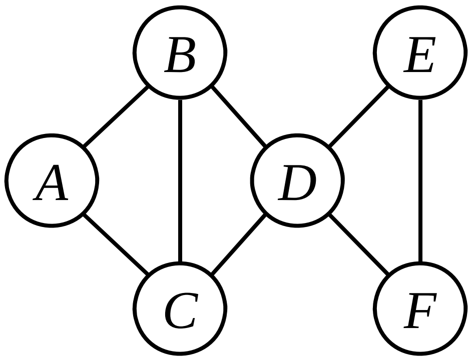
便于直观判断是否存在连接树



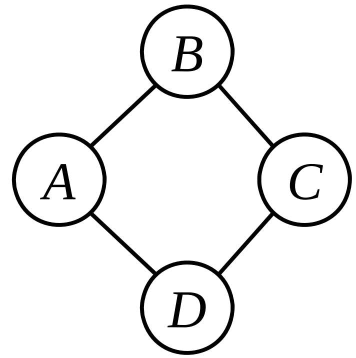


Basic concepts – perfect numbering

- 称素图 G 的一个结点编号是完美的，如果每个结点的低编号邻居 $\mathrm{ne}(v_j)\cap(v_1, v_2,\ldots, v_{j-1})$ 构成图上一个完备点集。



ABCDEF is a perfect numbering, but *ABCEFD* is not.



No perfect numbering is possible.

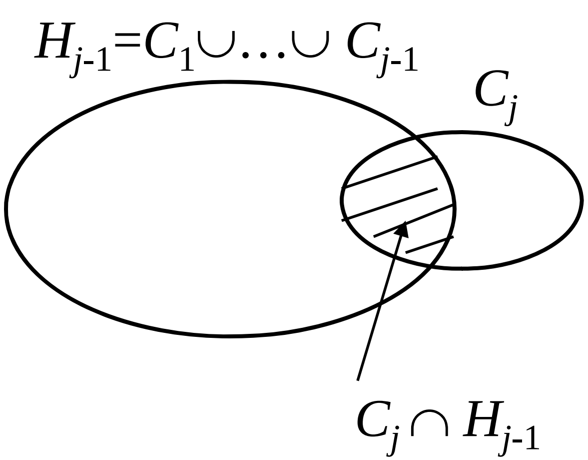
定理：素图 G 存在一棵连接树 $\Leftrightarrow$ 素图 G 容许一个完美编号

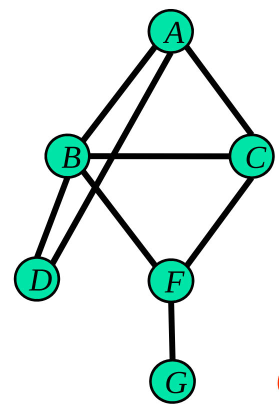
便于计算机编程判断是否存在连接树

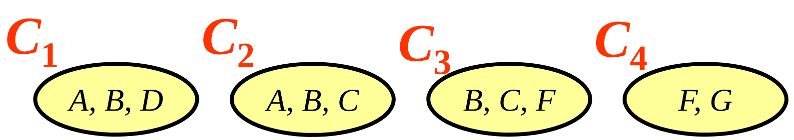
Basic concepts – RIP排序

- 一个连乘积G: $\prod_i \phi_i(\cdot)$ 的树分解：一个三元组 $<T, \chi, \psi>$
  - …
  - 树 T 满足RIP性质：树 T 上任两个cluster的交 $\chi(C_1)\cap\chi(C_2)$ 均包含于树 T 上C_1与C_2之间路径上的每一个cluster的变量集

称一个变量集的序列 $(C_1, C_2,\ldots, C_N)$ 满足RIP排序，
如果对$\forall j$，$1< j \le N$，存在一个$1\le i< j$ 使得 $C_j\cap(C_1\cup\ldots\cup C_{j-1}) \subseteq C_i$







定理：素图 G 存在一棵连接树 $\Leftrightarrow$ 素图 G 的最大簇容许RIP排序

便于构造连接树

从 满足RIP的最大簇序列 构造 连接树 的算法 (RIP2JT)

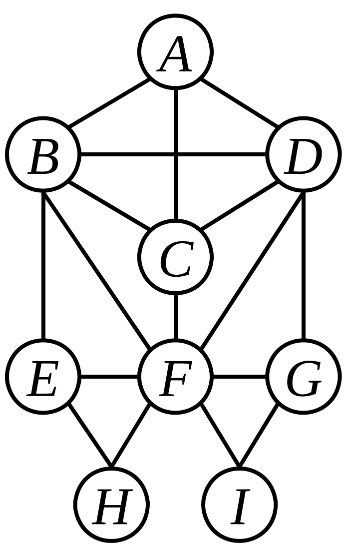
- 输入：素图 G 的一个满足RIP的最大簇序列$C_1, C_2,\ldots, C_N$

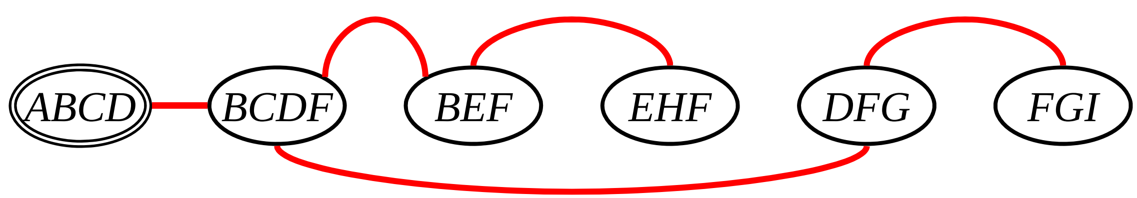
  Step 1: 一个最大簇作为一个cluster

  Step 2: for $j = 2, \ldots, N$

  考查C_j ，存在某个 $i\in\{1, \ldots, j\text{-}1\}$ 使得$C_j \cap H_{j\text{-}1} \subseteq C_i$ ，则在C_i 和 C_j 间添加一条边

- 输出：素图 G 的一棵连接树





总结

- 建立最优树分解的第一种情形
  - 可以直接用素图的最大簇作为cluster，建一棵cluster-tree

不同侧面反映了满足上述要求的素图的性质

不同侧面的表述有不同的用途

定理：素图 G 存在一棵连接树

$\Leftrightarrow$ 素图 G 是弦图

$\Leftrightarrow$ 素图 G 容许一个完美编号

$\Leftrightarrow$ 素图 G 的最大簇容许RIP排序

lesson8_triangulate

JT在什么意义下是最优？

Cluster-tree discussion

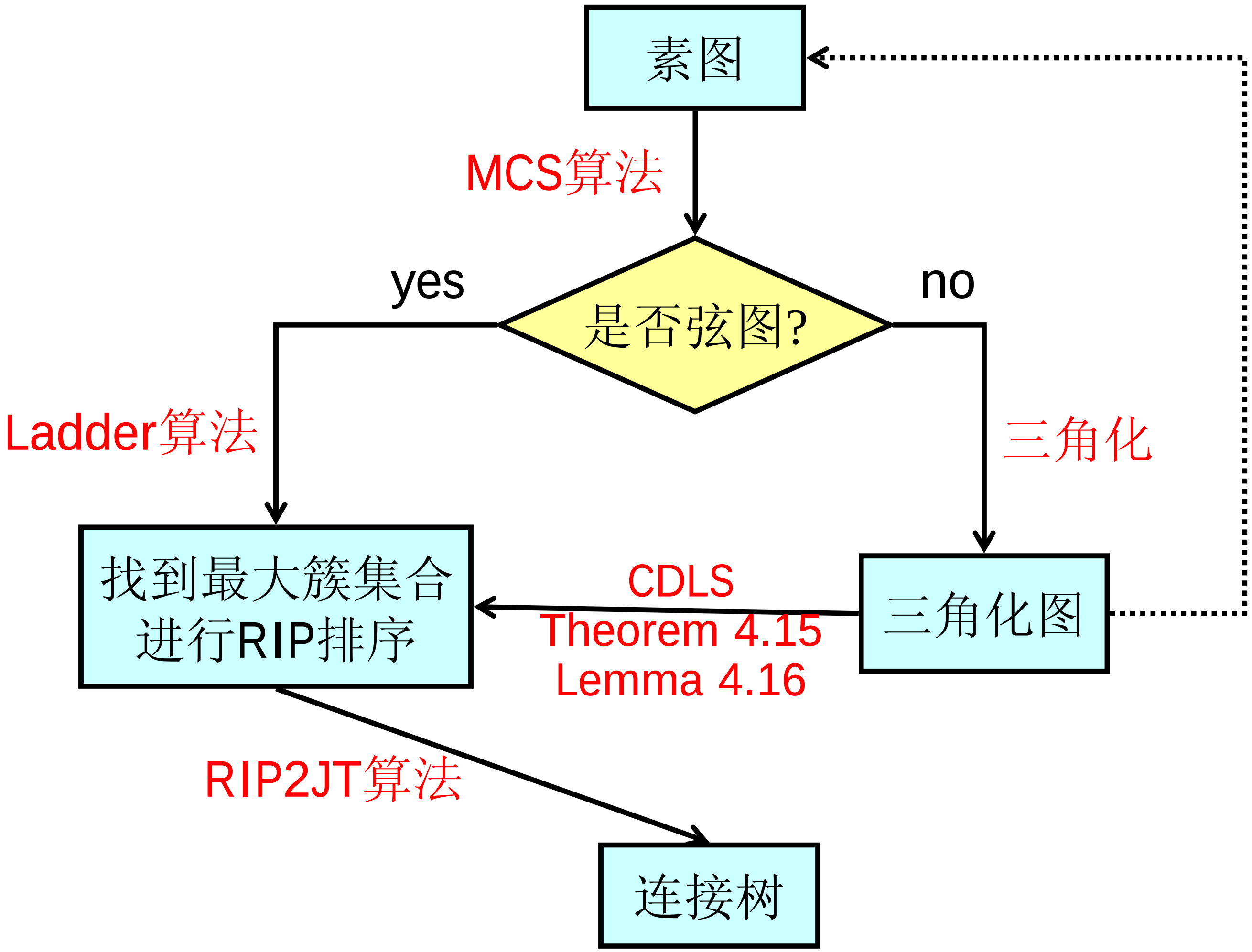
JT有什么性质？

Chordal graph, junction-tree

如何构造JT？

From primal graph to junction-tree

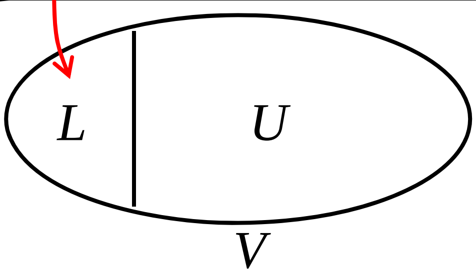
From primal graph to junction-tree



Maximum Cardinality Search (MCS)

- MCS：判断一个素图是否是弦图，复杂度：$O(|V|+|E|)$.
  - OUTPUT：1) whether G is chordal or not
    2) a perfect numbering for G, if it is chordal
- 思想：对图中结点尝试进行完美编号

L is the set of previously labeled vertices

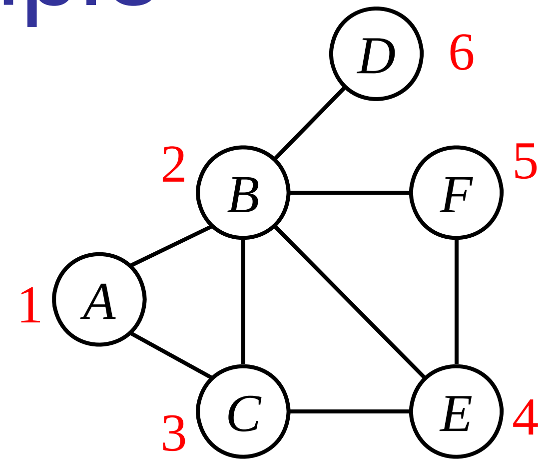


- Set counter $i = 1$
- Set $L = ø$, $U = V \setminus L$
- For $v \in U$, set $c(v)=0$

$c(v)$ is the number of previously labeled neighbors of v

- While $U \neq ø$
  - Select the vertex v maximizing $c(v)$ over $v \in U$ and label it i
  - If $\prod_{v_i} = ne(v_i) \cap L$ is not complete in G
    set OUTPUT = 'G is not chordal'
  - Otherwise, set $L = L \cup \{v_i\}$
  - Set $c(w)=c(w)+1$ for each vertex $w \in \mathrm{ne}(v_i) \cap U$
  - Increment i by 1.
- Set OUTPUT = 'G is chordal'

MCS : example



Node numbering (perfect) produced is: (A, B, C, E, F, D)

The first node in the ordering is chosen arbitrarily, since $c(v)$=0 for all nodes during the first iteration.

v =	A	B	C	D	E	F	L	U
$c(v)$	0	0	0	0	0	0	ø	$A : F$
		1	1	0	0	0	A	$B : F$
			2	1	1	1	A, B	$C : F$
				1	2	1	$A : C$	$D : F$
				1		2	$A : C, E$	D, F
				1			$A{:}C, E, F$	D

MCS: conclusion

- MCS算法从任一个结点开始，对图中结点尝试进行完美编号
  - 如果这个编号过程顺利完成，则我们得到图的一个完美编号（同时也确认这个图是一个弦图）；
  - 否则，编号失败，得不到完美编号（我们也知道这个图不是弦图）。

- **如果一个图是弦图，则以每个结点为起始编号，MCS算法能找到一个完美编号。**

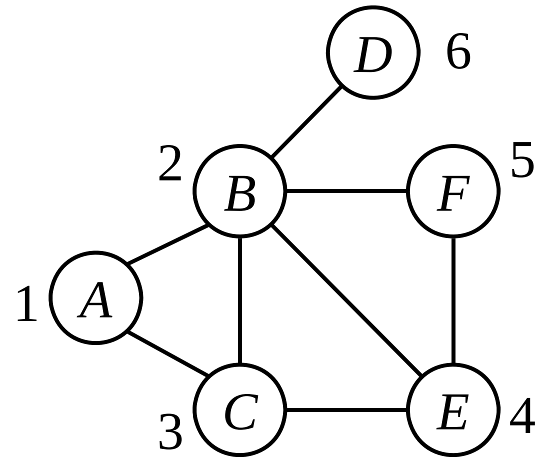
Finding max-cliques of a chordal graph

- Algorithm: Ladder-based maximal-clique construction & RIP ordering
  - 思想：如果MCS算法顺利找到了一个完美编号，则可以很方便得到这个弦图的所有最大簇，同时对其进行RIP排序。
  - Start from $(v_1, v_2, \ldots, v_{|V|})$ obtained by MCS
  - Denote $\pi_i \triangleq \left|\Pi_{v_i}\right|$
  - Call node v_i a “ladder node” if $i = |V|$ or if ($i < |V|$ and $\pi_i \geq \pi_{i+1}$)
  - Let the jth ladder node be λ_j, and define

$$C_j \triangleq \{\lambda_j\} \cup \Pi_{\lambda_j}$$

- Theorem
  - $(C_1, C_2, \ldots)$ are the maximal-cliques of G and satisfies the RIP.

Ladder-based maximal-clique construction: example



Labeling	A	B	C	E	F	D
Π_{v_i}	ø	$\{A\}$	$\{A, B\}$	$\{B, C\}$	$\{B, E\}$	$\{B\}$
π_i	0	1	2	$\geq$ 2	$\geq$ 2	$\geq$ 1
Ladder?	N	N	Y	Y	Y	Y

The maximal-cliques produced is: $\{C, A, B\}$, $\{E, B, C\}$, $\{F, B, E\}$, $\{D, B\}$.

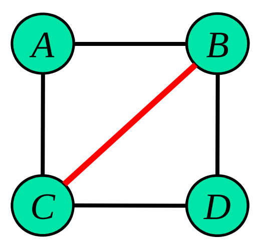
❖ 一个树分解中的cluster的性质
- 性质：素图的任一个最大簇一定包含于某个cluster

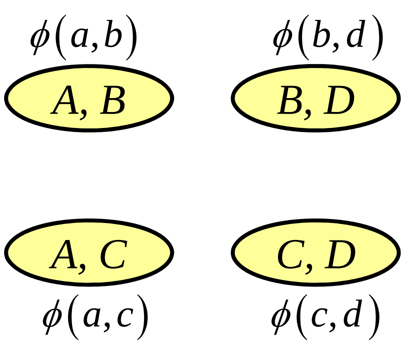
❖ 建立最优树分解的第二种情形

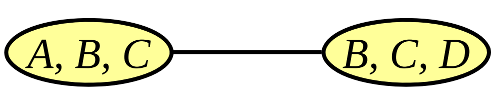
- 无法直接用素图的最大簇作为cluster，建一棵cluster-tree
- 良好愿望：一个最大簇正好对应一个cluster
- 退一步：使用更大的cluster
- 具体做法：图上添加边

——代数上，将局部函数适当组合, 使得”新连乘积”的素图表示变成弦图

连乘积：$\underbrace{\phi(a,b)\phi(a,c)}_{\phi^{new}(a,b,c)}\underbrace{\phi(c,d)\phi(b,d)}_{\phi^{new}(b,c,d)}$







是原连乘积的一个cluster-tree！

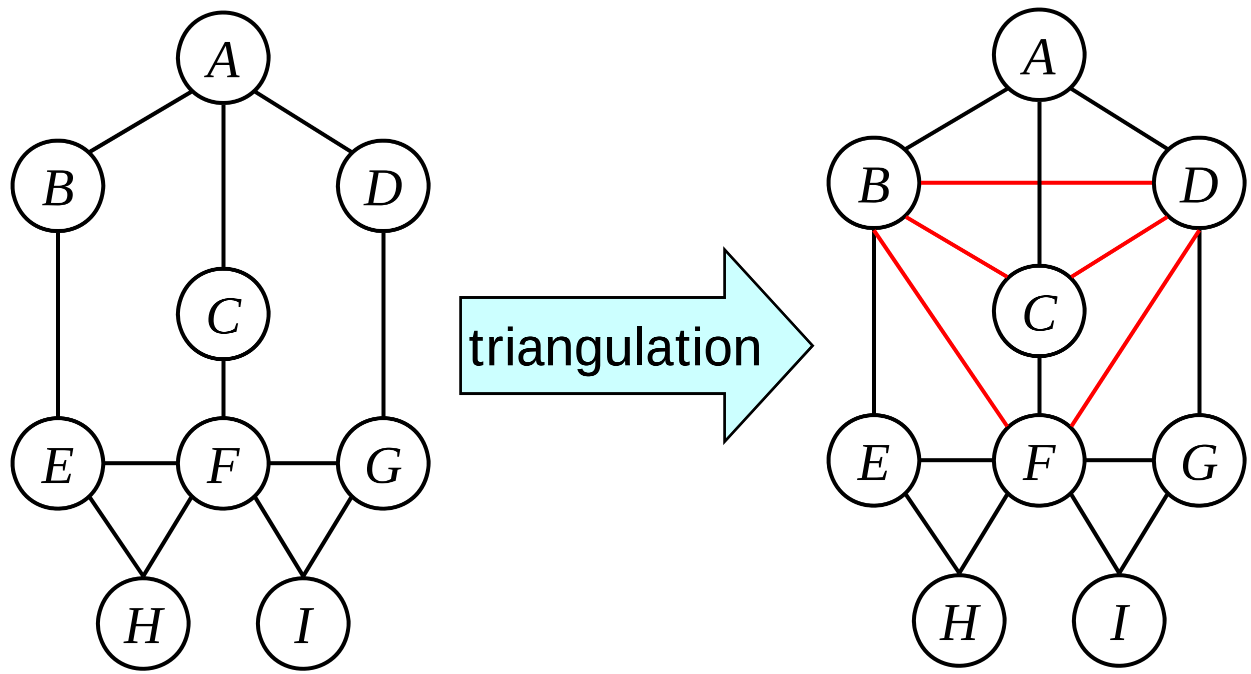
Basic concepts - Triangulation

Triangulation 三角化	如果一个图 $G = (V, E)$ 不是弦图，我们总可以添加一些边 F 使得新的图 $G' = (V, E')$（其中$E'=E\cup F$ ）是一个弦图。 这些边 F 称为 添加边（fill-in edges）. 填边后得到的弦图 G' 称为原图 G 的三角化

使得三角化后得到的弦图的最大簇的规模尽可能小

最大的最大簇规模 衡量了 三角化的质量

Original graph



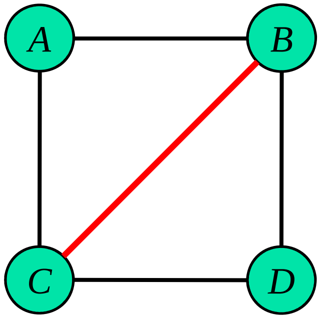
❖ 消除式三角化算法 Triangulation via elimination

- 给定素图 G 的变量排序 $d=(v_1, v_2,\ldots, v_N)$

  For $i = N$ to 1

  添加边使得 结点集 C_i = {结点v_i及编号比它小的邻居} 构成一个完备点集（称为消除集，elimination set）

- 结果得到一个弦图 G_d $\Leftarrow$ **素图 G 在结点排序 d 下的诱导图 G_d**
- 给定的变量排序 $(v_1, v_2,\ldots, v_{|V|})$ 是三角化图 G_d 的一个完美编号



消除式三角化的质量 取决于 选择的变量排序！

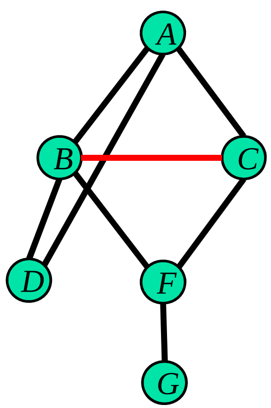
*CDLS Theorem 4.15*表明：G_d 的最大的最大簇规模 = 图G在排序d下的诱导宽度

Induced-graph, induced-width

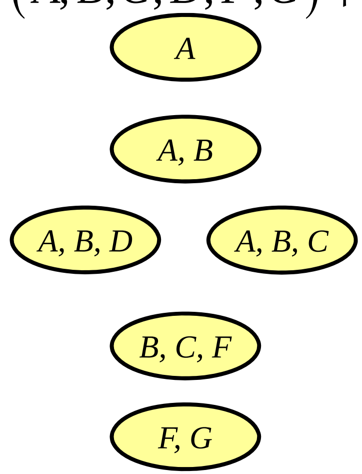
- 一个素图 G 在结点排序 $d=(v_1,\ldots,v_N)$ 下的诱导图 G_d 如下定义:
  - 从后往前依次消除各结点;
  - 消除结点 v_i 时，将此时图中 v_i 的邻居两两间添加边。
  - 称此时图中 v_i 的邻居数目为：$\max$ 结点 v_i 在排序d下的诱导宽度
  - G_d = G 加上 上述过程多添加的边。

$= \max$ 结点 v_i 的消除集规模−1

CDLS Theorem 4.15: G_d 的最大簇 $\in$ 消除集的集合 $\{C_1, C_2, \ldots, C_N\}$



排序$d=(A,B,C,D,F,G)$下消除集



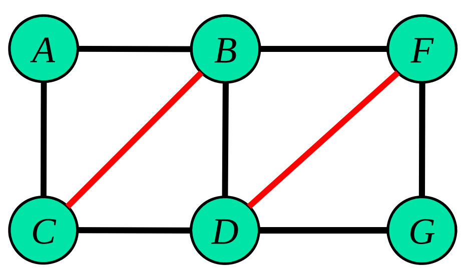
一个变量排序下的消除式三角化质量 = 该排序下的诱导宽度

- 寻找具有最小诱导宽度的变量排序 是 NP-hard.
  - 贪婪启发式搜索

One-step look-ahead triangulation

- 一步勘探三角化：有选择地消除下一个结点，动态排序
  - 一开始所有结点都未编号，当前待编号 $i = |V|$
  - 当还有未编号的结点，进行如下操作
    - 当前待编号为 i
    - 依据**某种准则 $f(v)$** 选取一个最优的未编号结点
    - 添加边使得 消除集 C_i = {结点v_i及未编号的邻居} 构成一个完备点集
    - 消除结点v_i ，下一步待编号为 i -1

$f(v_i)$ = 为使得 消除集C_i是完备的而需添加的边的数目

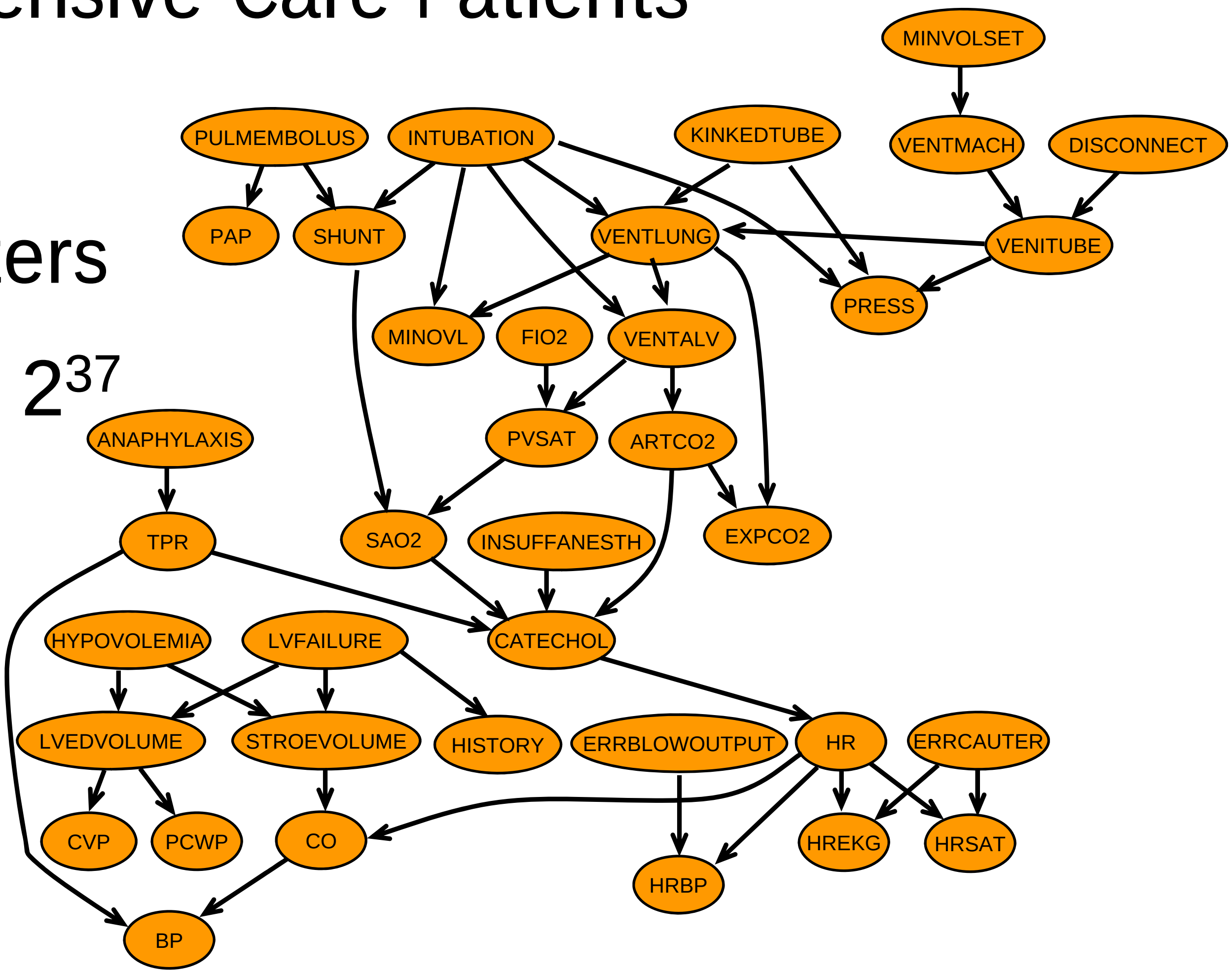


A real Bayes net

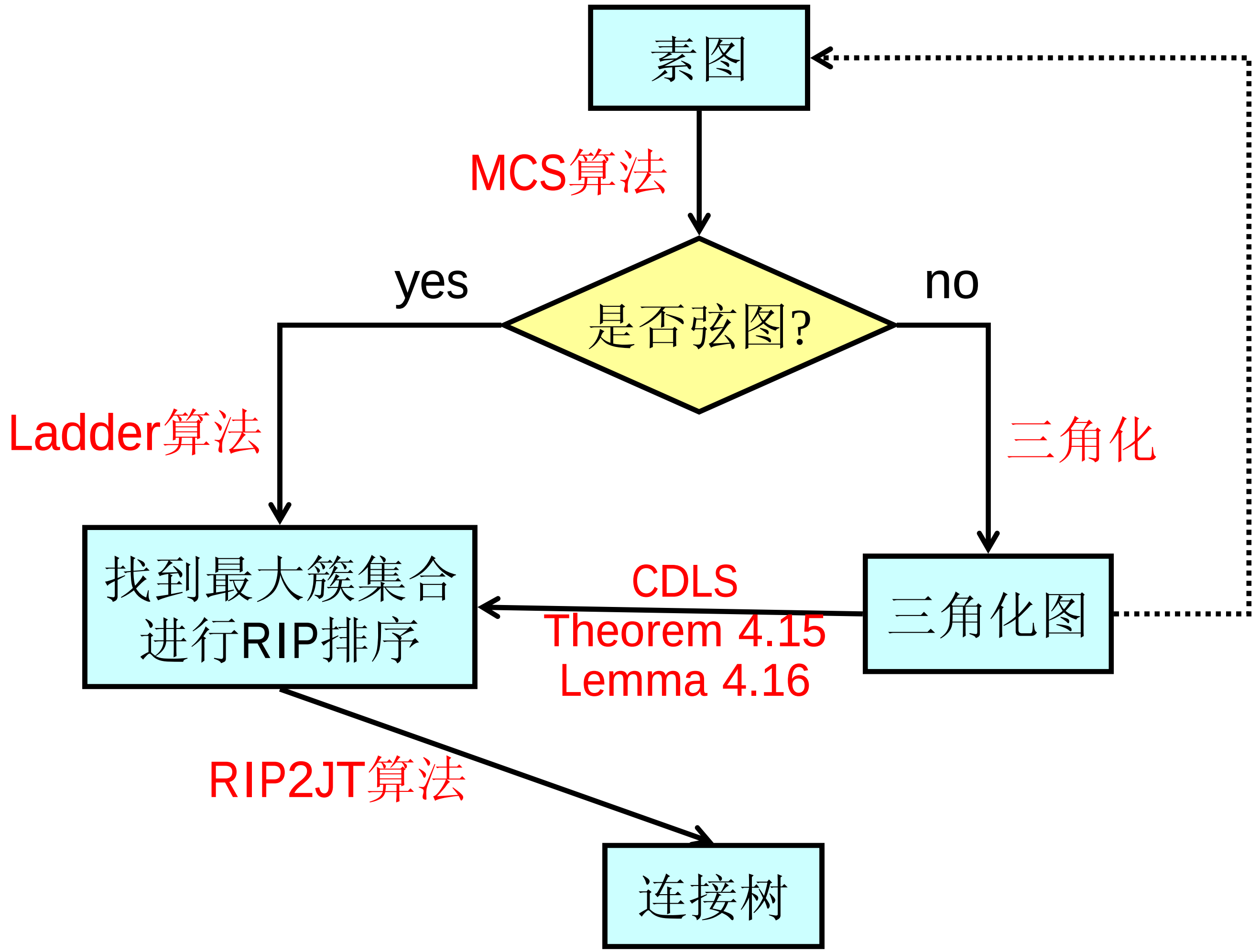
Monitoring Intensive-Care Patients

- 37 variables
- 509 parameters

  ... instead of 2^{37}



From primal graph to junction-tree



Recent progress

- Compiling Graphical Models
  - Adnan Darwiche, UCLA
  - UAI'06 Tutorial
  - Arithmetic Circuits
  - Various domains - such as relational models, genetic linkage analysis, coding, and probablistic planning
  - http://www.ics.uci.edu/~csp/uai2006/tutorials
  - "Inference in Bayesian Networks: A Historical Perspective", Darwiche 2010.

编译(寻找最优的计算结构) 与 计算 分离