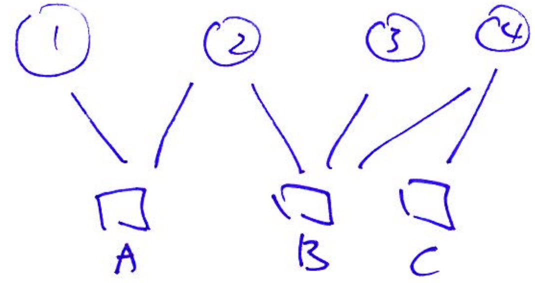
Factor Graphs

$X_1 \dots X_N$

$$p(X_1 = x_1 \dots X_N = x_n) = \frac{1}{Z} \prod_a f_a(\bar{x}_a)$$



$$p(x_1 \dots x_4) = \frac{1}{Z} f_A(x_1 x_2) f_B(x_2 x_3 x_4) f_C(x_4)$$

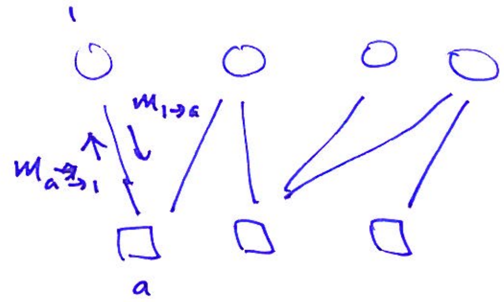
Interested in:

1. estimating marginals $P_s(X_s) = \sum_{X \setminus X_s} p(X)$, $p_i(x_i)$
2. estimating Z or $\log Z$

Belief Propagation

(Sum product, forward backward, Gallager decoding, turbo-decoding)

- Messages $m_{a\to i}(x_i)$ from factors to variables
  - what values does factor a like variable X_i to take on?
- Messages $n_{i\to a}(x_i)$ from variables to factors
  - what values X_i likes based on information from all but a.



- Beliefs

$$b_i(x_i) \propto \prod_{a\in N(i)} m_{a\to i}(x_i)$$

/ based on all pieces of information coming into X_i
→ note product is independent pieces of information

$$b_a(x_a) \propto f_a(x_a) \prod_{i\in N(a)} n_{i\to a}(x_i)$$

- Constraints:
  - nonnegative $b_i(x_i) \geq 0$. $b_a(x_a) \geq 0$
  - normalization $\sum_{x_i} b_i(x_i) = 1$ $\sum_{x_a} b_a(x_a) = 1$
  - marginalization $b_i(x_i) = \sum_{x_a \setminus x_i} b_a(x_a)$

- Message updates

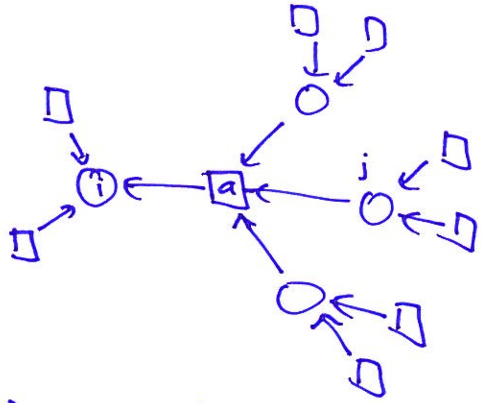
$$b_i(x_i) = \sum_{x_a\setminus i} b_a(x_a)$$

$$\prod_{a'\in N(i)} m_{a\to i}(x_i) = \sum_{x_a\setminus i} f_a(x_a) \prod_{j\in N(a)} n_{j\to a}(x_j)$$

$$= \sum_{x_a\setminus i} f_a(x_a) \prod_{j\in N(a)} \prod_{a'\in N(j)\setminus a} m_{a'\to j}(x_j)$$

$$= \prod_{a'\in N(i)\setminus a} m_{a'\to i}(x_i) \sum_{x_a\setminus i} f_a(x_a) \prod_{j\in N(a)\setminus i} m_{a'\to j}(x_j)$$

$$m_{a\to i}(x_i) = \sum_{x_a\setminus i} f_a(x_a) \prod_{j\in N(a)\setminus i} n_{j\to a}(x_j)$$



- replace $\sum$ by max
  → max product for MAP state

Free energies

- energy $E(x) = -\sum_a \log f_a(x_a)$
- entropy $H(b) = -\sum_x b(x) \log b(x)$
- Gibbs free energy

$$F(b) = \underset{\text{1/4}}{\frac{1}{T}}\sum_x b(x)E(x) - H(b)$$
$$= U(b) - H(b)$$

- system prefers low energy, high entropy

minimize $F(b)$
$\quad b$

$\Rightarrow \underset{\overset{\shortparallel}{p}}{b_{min}}(x) = \frac{1}{Z(T)} e^{-E(x)/T}$

attaining

$F(\underset{\overset{\shortparallel}{p}}{b_{min}}) = -\log Z(T) = F_H$

Helmholtz free energy

$$F(b) = F_H + KL(b \| p)$$

Two computational bottlenecks:

- ~~H(b)~~ b very large
- H(b) expensive to compute

Solutions

- assume b comes from tractable family
  - → variational approximation
- give up on b being a distribution
  - → replaced by a family of marginal beliefs $b_a(x_a)$ ...
    - → introduce constraints among them
    - → usually set of constraints ensuring global consistency is very large
    - → just use a subset of them — marginal polytope
    - → local consistency
- approximate H(b) as a function of marginal beliefs

Bethe free energy

- family of beliefs

  $b_i(x_i), \quad b_a(x_a)$

- nonnegative, normalization, marginalization constraints
- $U_{bethe} = -\sum_a \sum_{x_a} b_a(x_a) \log f_a(x_a)$

$$H_{bethe} = -\sum_a \sum_{x_a} b_a(x_a) \log b_a(x_a) + \sum_i (\underset{|N(i)|}{d_i} - 1) \sum_{x_i} b_i(x_i) \log b_i(x_i)$$

← entropy of X_i ~~e~~ over-counted by d_i times.

- ① can be not globally consistent (unrealizable)
- ② can obtain negative Bethe entropy.

+ ③ exact on trees
+ ④ maxent-normal

~~min~~

$$\mathcal{L}(b, \lambda, \pi) = -\sum_a \sum_{x_a} b_a(x_a) \log f_a(x_a) + \sum_a \sum_{x_a} \cancel{(d_i-1)}\, b_a(x_a) \log(b_a(x_a)) - \sum_i (d_i - 1) \sum_{x_i} b_i(x_i) \log b_i(x_i)$$
$$- \sum_a \gamma_a \left(\sum_{x_a} b_a(x_a) - 1\right) - \sum_i \gamma_i \left(\sum_{x_i} b_i(x_i) - 1\right) - \sum_i \sum_{a \in N(i)} \sum_{x_i} \lambda_{ai}(x_i) \left(b_i(x_i) - \sum_{x_a \setminus i} b_a(x_a)\right)$$

$$\frac{\partial \mathcal{L}(b, \lambda, \pi)}{\partial b_a(x_a)} = -\log f_a(x_a) + \log b_a(x_a) + 1 + \gamma_a - \sum_{i \in N(a)} \lambda_{ai}(x_i) \Rightarrow b_a(x_a) = f_a(x_a)\, e^{\gamma_a - 1 + \sum_i \lambda_{ai}(x_i)}$$

$$\frac{\partial \mathcal{L}(b, \lambda, \pi)}{\partial b_i(x_i)} = (1 - d_i)(\log b_i(x_i) + 1) - \gamma_i + \sum_a \lambda_{ai}(x_i) \Rightarrow b_i(x_i) = e^{-1 + \frac{1}{d_i - 1}(-\gamma_i + \sum_a \lambda_{ai}(x_i))}$$

$$\lambda_{ai}(x_i) = \log n_{i \to a}(x_i) = \log \prod_{a' \in N(i) \setminus a} m_{a' \to i}(x_i)$$

$$b_a(x_a) \propto f_a(x_a) \prod_{i \in N(a)} n_{i \to a}(x_i)$$

$$b_i(x_i) \propto \left(\prod_{a \in N(i)} \prod_{a' \in N(i) \setminus a} m_{a' \to i}(x_i)\right)^{\frac{1}{d_i - 1}} = \prod_{a \in N(i)} m_{a \to i}(x_i)$$

- BP updates from marginalization constraints again

Region Graphs

Region R is a set of variables V_R and factors A_R such that ~~for~~ if $a \in A_R$ then $N(a) \subset V_R$.

- true marginal $p_R(x_R)$
- beliefs $b_R(x_R)$
- energy $-\sum_{a \in A_R} \log f_a(x_a) = E_R(x_R)$
- free energy $F_R(b_R) = \underbrace{\sum_{x_R} b_R(x_R) E_R(x_R)}_{U_R(b_R)} - H_R(b_R)$ // $-\sum_{x_R} b_R(x_R) \log b_R(x_R)$.

Region-based free energy

$$F(\{b_R\}) = \sum_R c_R U_R(b_R) - \sum_{R} c_R H_R(b_R)$$

counting numbers.

↳ treat each region exactly

↳ approximate ~~free~~ free energy (in fact entropy)

↳ treat interactions among regions using marginalization constraints.

① Valid if $\sum_R c_R \mathbb{1}(a \in A_R) = \sum_R c_R \mathbb{1}(i \in V_R) = 1 \quad \forall a, i.$

exact energy if $b_R = p_R$

exact entropy if uniform distribution

② maxent-normal if $H(\{b_R\}) = \sum_R c_R H_R(b_R)$ achieves maximum under constraints when all b_R uniform

③ short loops should be regions (loop-based region graphs)

④ perfect-correlation if $\sum_R c_R = 1$ ← exact when all variables fully coupled.

⑤ non-singular if message passing algorithms converge to unique exact uniform fixed point.

loop graphs singular if $\sum_R c_R > 1$.

① Tree EP
② planar graphs

$\sum_R c_R = L - E + V = 1$ for faces, planar.

Region Graphs

- ~~Constraints~~ Organize regions into a directed acyclic graph

  $R_1 \longrightarrow R_2$ only if $R_2 \subset R_1$ / all factors a / all variables i.

  enforce marginalization constraint

  $$\sum_{x_{R_1} \setminus x_{R_2}} b_{R_1}(x_{R_1}) = b_{R_2}(x_{R_2})$$

  ↳ require : subgraph of all regions containing each a or i is connected

  so that region graph gives consistent beliefs about a, i.

  extended to: regions containing any subset of a's, i's are connected.

  ↳ require $c_R = 1 - \sum_{\substack{\text{ancestors} \\ R' \in A(R)}} c_{R'}$ ← degree of freedom computed properly.

- exact if region graph forms a tree (and satisfies all constraints above)

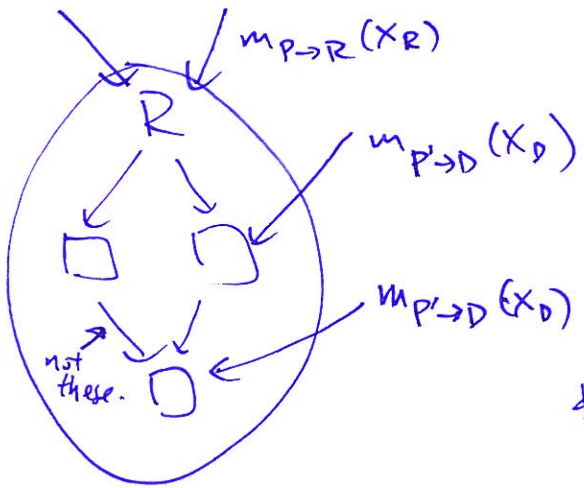
Parent-to-Child algorithm

$D(R)$ = descendents
$\mathcal{E}(R) = R \cup D(R)$
$\mathcal{P}(R)$ = parents

For every $P \to R$ in region graph

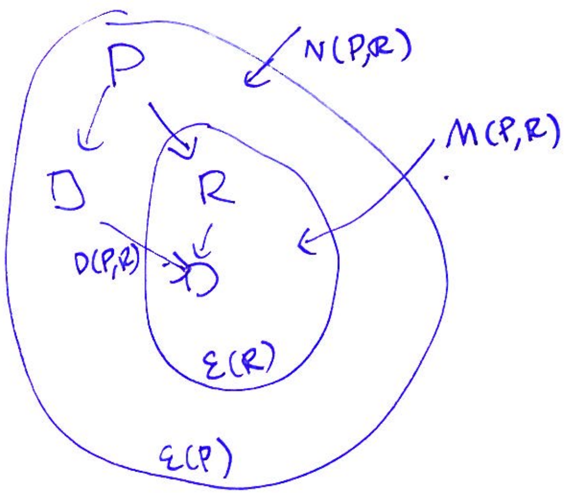
message $m_{P\to R}(x_R)$

beliefs $$b_R(x_R) \propto \prod_{a\in A_R} f_a(x_a) \underbrace{\prod_{P\in\mathcal{P}(R)} m_{P\to R}(x_R) \prod_{D\in D(R)} \prod_{P'\in\mathcal{P}(D)\setminus\mathcal{E}(R)} m_{P'\to D}(x_D)}_{\text{all messages from outside } \mathcal{E}(R)}$$



$$b_P(x_P) \propto \prod_{a\in A_P} f_a(x_a) \prod_{(I,J)\in N(P,R)\cup M(P,R)} M_{I\to J}(x_J)$$

$$b_R(x_R) \propto \prod_{a\in A_R} f_a(x_a) \prod_{\substack{(I,J)\in D(P,R)\cup M(P,R)\\ \cup\{(P,R)\}}} M_{I\to J}(x_J)$$



$$\sum_{x_P\setminus x_R} b_P(x_P) = b_R(x_R)$$

$$\sum_{x_P\setminus x_R} \prod_{a\in A_P\setminus A_R} f_a(x_a) \prod_{N(P,R)} M_{I\to J}(x_J) = M_{P\to R}(x_R) \prod_{D(P,R)} M_{I\to J}(x_J)$$

Further Details of message-passing

① initialization

- random
- uniform

② termination

- convergence criteria
  - message change
  - belief change

③ damping

$$m^{new} := \alpha\, m^{old} + (1-\alpha)\, m^{\cancel{new}\ update}$$

or $$:= (m^{old})^{\alpha} (m^{\cancel{new}\ update})^{1-\alpha}$$

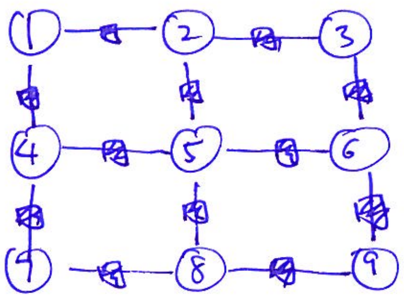
$$m^{update}_{P\to R} = \frac{\sum_{x_P \setminus x_R} \prod_{a \in A_P \setminus A_R} f_a(x_a) \prod_{N(P,R)} m^{old}_{I\to J}(x_J)}{\prod_{D(P,R)} m^{update}_{I\to J}(x_J)}$$

start from bottom of region graph, work your way up.

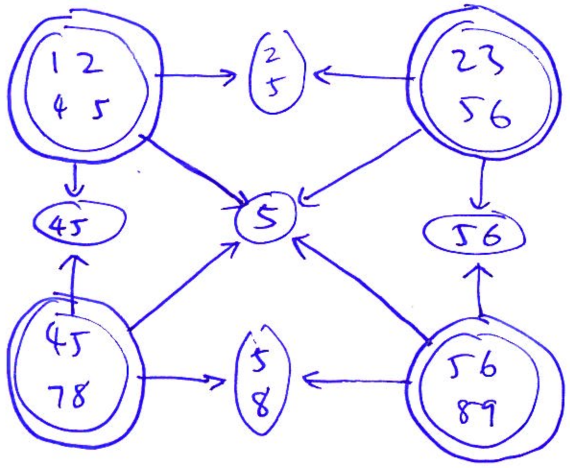
④ normalization

- normalize messages
- or work in log domain.

Construction Methods

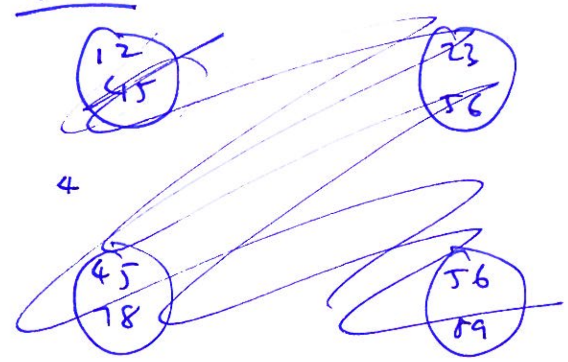


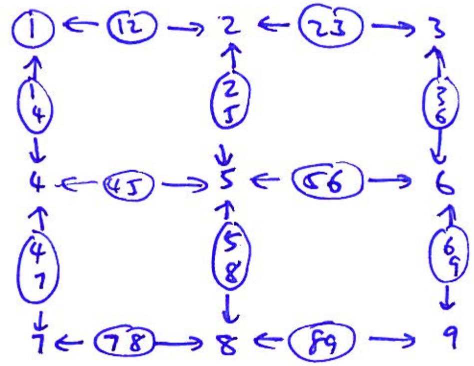
Junction graph



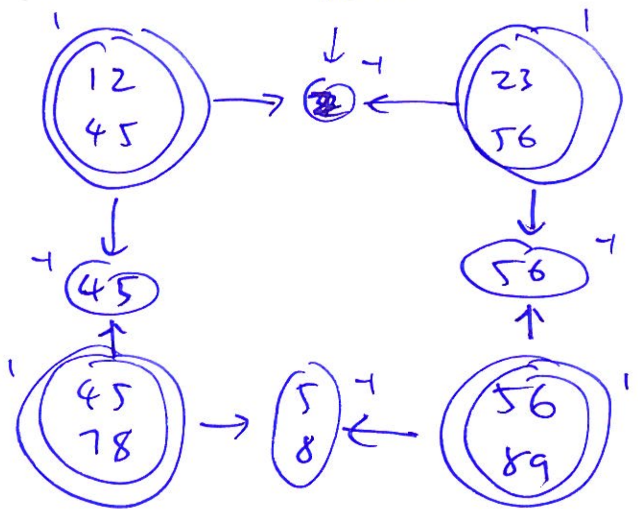
for each i remove it from regions until induced region graph forms a tree

Bethe



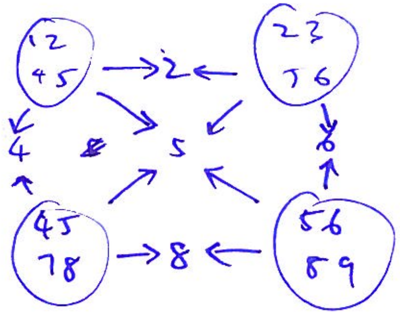


remove 5.



or: merge factors first.

e.g.



Cluster Variation method

$$
\begin{tikzcd}
{}^{1}\begin{smallmatrix}12\\45\end{smallmatrix} \arrow[r] \arrow[d] & {}^{-1}\begin{smallmatrix}2\\5\end{smallmatrix} \arrow[d] & {}^{1}\begin{smallmatrix}23\\56\end{smallmatrix} \arrow[l] \arrow[d] \\
{}^{-1}\,45 \arrow[r] & {}^{1}\,5 & {}^{-1}\,56 \arrow[l] \\
{}^{1}\begin{smallmatrix}45\\78\end{smallmatrix} \arrow[r] \arrow[u] & {}^{-1}\begin{smallmatrix}5\\8\end{smallmatrix} \arrow[u] & {}^{1}\begin{smallmatrix}56\\89\end{smallmatrix} \arrow[l] \arrow[u]
\end{tikzcd}
$$

form all intersects, intersects of intersections

note: need not be maxent-normal

Loop-based approximation,

- start with Bethe
- add strongest coupled loops first, maintaining non-singularity.
  (~~loops cycle space~~
  loops are non-singular (independent in cycle space))
- stop at maximal.

Tree EP

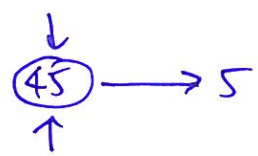
- take spanning tree
- loops are those in spanning tree + individual edges

Planar Graphs

- use faces of planar graph

produce maximally non-singular loop-based approximations

Parent-Child Updates — Cluster Variation method 3×3 grid

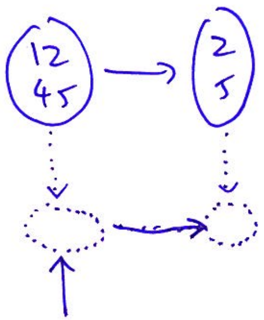


$$m_{45\to5}(x_5) = \sum_{x_4} f_{45}(x_4x_5)\, m_{1245\to45}(x_4x_5)\, m_{4578\to45}(x_4x_5) \cdot f_4(x_4)$$

$N(45,5) = \{(1245, 45), (4578, 45)\}$

$D(45,5) = \{\}$.

similarly $m_{25\to5}$, $m_{56\to5}$, $m_{58\to5}$.

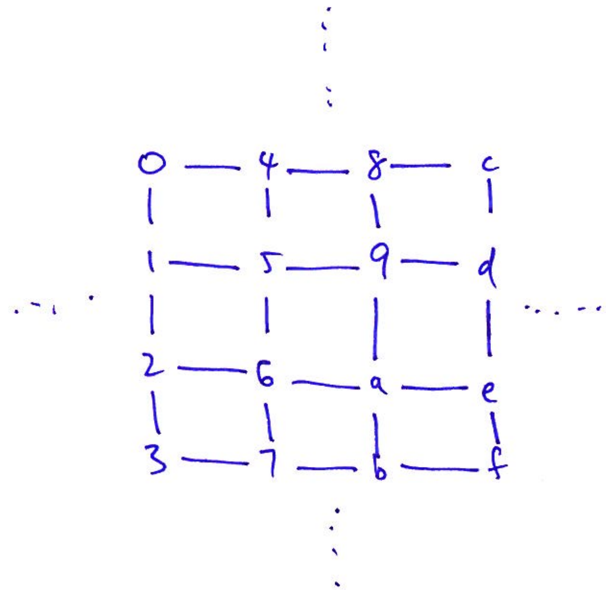


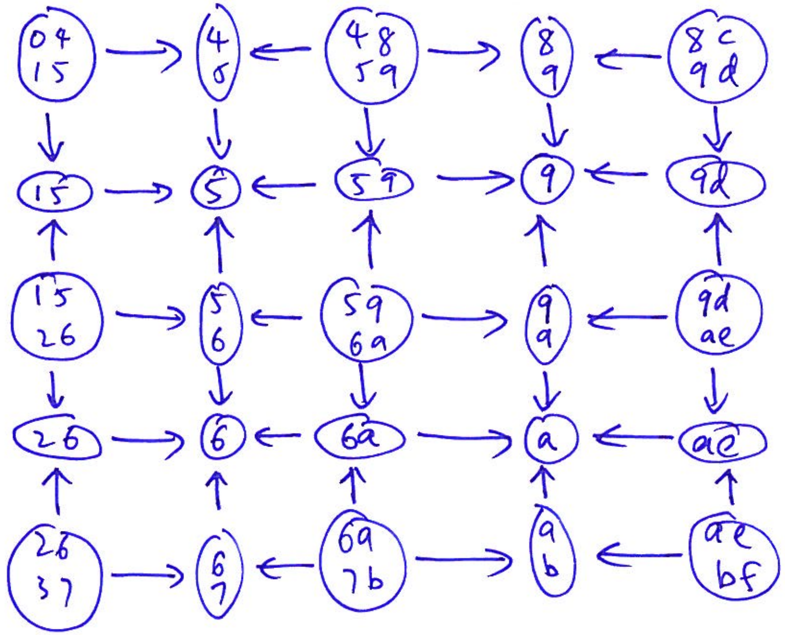
$$m_{1245\to25}(x_2x_5)\, m_{45\to5}(x_5) = \sum_{x_1x_4} f_{12} f_{45} f_1 f_4\, m_{1245\to45}(x_4x_5)$$

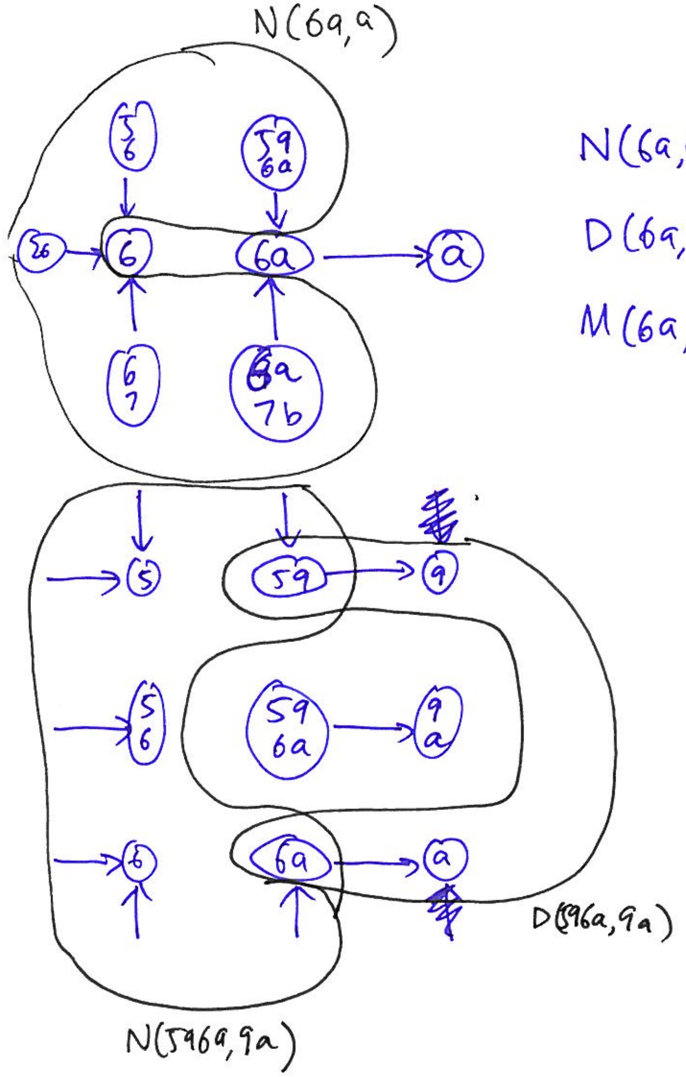
$N(1245,45) = \{(4578,45)\}$

$D(1245,45) = \{(45,5)\}$

Parent-child updates – Cluster Variation Method n×m grid







$N(6a,a) = \{(596a,6a), (6a7b,6a), (56,6), (26,6), (67,6)\}$

$D(6a,a) = \{\}$

$M(6a,a) = \{(9a,a), (ae,a), (ab,a)\}$

$$m_{596a \to 9a}(x_9 x_a) = \frac{\sum_{x_5 x_6} f_{56} f_{59} f_{6a} f_5 f_6 \; m_{4589 \to 59} \, m_{45 \to 5} \, m_{15 \to 5} \, m_{1256 \to 56} \, m_{26 \to 6} \, m_{67 \to 6} \, m_{6a7b \to 6a}}{m_{59 \to 9} \; m_{6a \to a}}$$