Introduction to Graphical Models and Distributed Inference

Sang Hyun Lee
(sanghyunlee@korea.ac.kr)

Contents

1. Graphical Models

- 1.1

  Factor Graphs

- 1.2

  Graphs of Codes

- 1.3

  Belief Propagation Algorithms

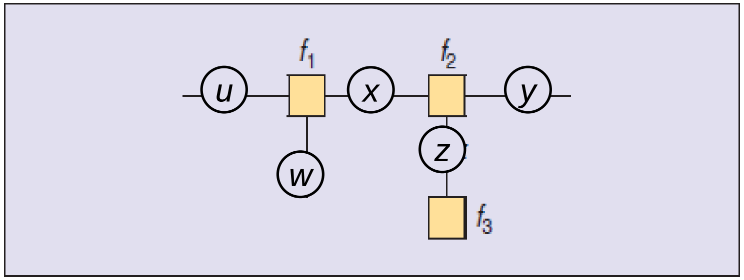
1.1 Factor Graphs (FG)

✓ Representation of factorization of a function of several variables.

$$f(u,w,x,y,z) = f_1(u,w,x) f_2(x,y,z) f_3(z)$$

f: global function

f_1, f_2, f_3: local functions



▲ 1. An FFG.

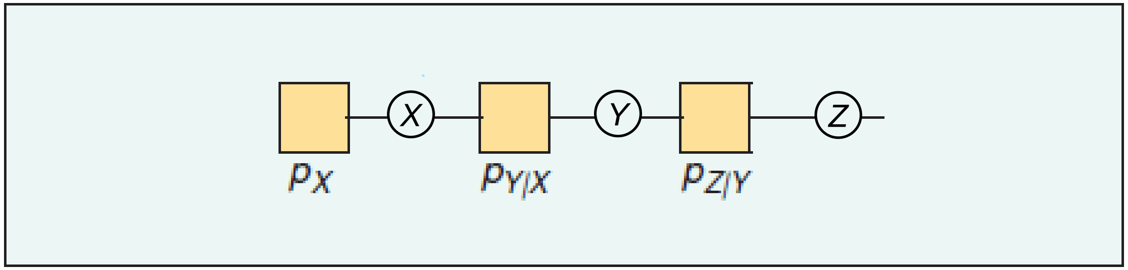
✓ Consists of
1. Factor nodes: squares □
2. Variable nodes: circles ○
3. Edges: connection of two nodes

✓ Main application: Probabilistic models

✓ FFG: a variation of FG, for simple graphs
1. Factor nodes: boxes representing factor
2. Edges: circles with two neighbors
3. Half edges: circles with one neighbors

1.1 Factor Graphs

- **Markov chain**: Chain of joint probabilities. Non-neighbor nodes are independent to each other. (all function nodes dependent)



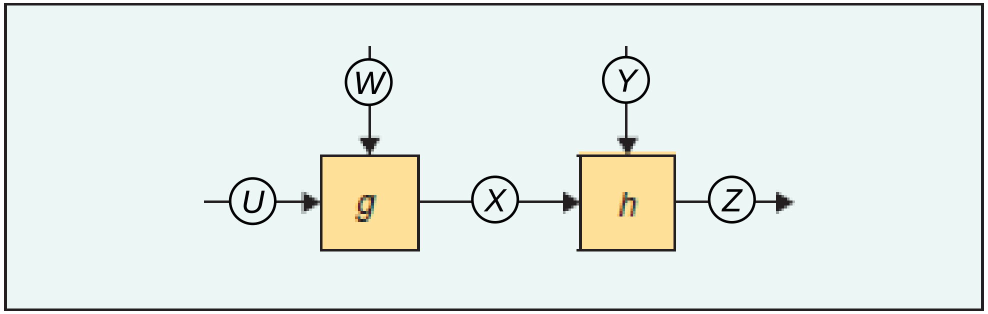
▲ *2. An FFG of a Markov chain.*

$$p_{XYZ}(x,y,z) = p_X(x)p_{Y|X}(y|x)p_{Z|Y}(z|y,x)$$
$$= p_X(x)p_{Y|X}(y|x)p_{Z|Y}(z|y)$$

1.1 Factor Graphs

- ✓ **Block Diagram Interpretation:**

$$X = g(U, W)$$
$$Z = h(X, Y)$$



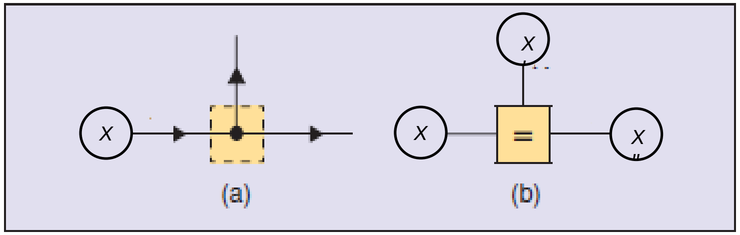
3. A block diagram.

- ✓ The function block $X = g(U, W)$ represents the factor $\delta(x - g(u, w))$
- ✓ The function block $Z = h(X, Y)$ represents the factor $\delta(z - h(x, y))$

∴ The whole graph: $f(u, w, x, y, z) = \delta(x - g(u, w)) \bullet \delta(z - h(x, y))$

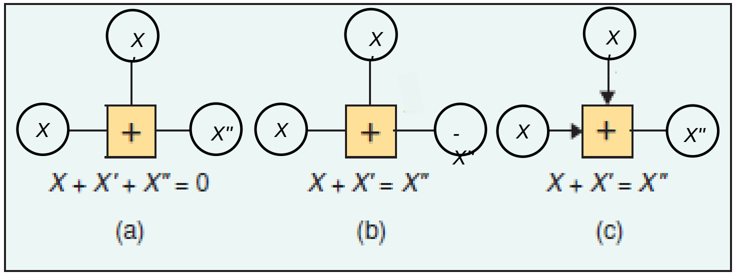
1.1 Factor Graphs

- ✓ **Branching points**:
  Becomes factor nodes, as Fig(4).
- ✓ New variables factor arises:
  $X = X' = X''$
  $f_{=}(x, x', x'') \triangleq \delta(x - x')\delta(x - x'')$



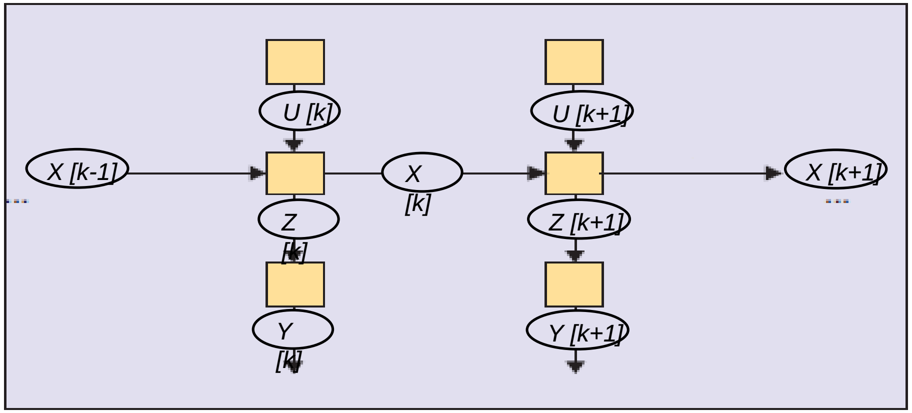
▲ 4. (a) Branching point becomes (b) an equality constraint node.

- ✓ Other symbols are also used.
  $f_{+}(x, x', x'') \triangleq \delta(x + x' + x'')$
- ✓ $X + X' = X''$ can be represented by Fig(5b) and Fig(5c)

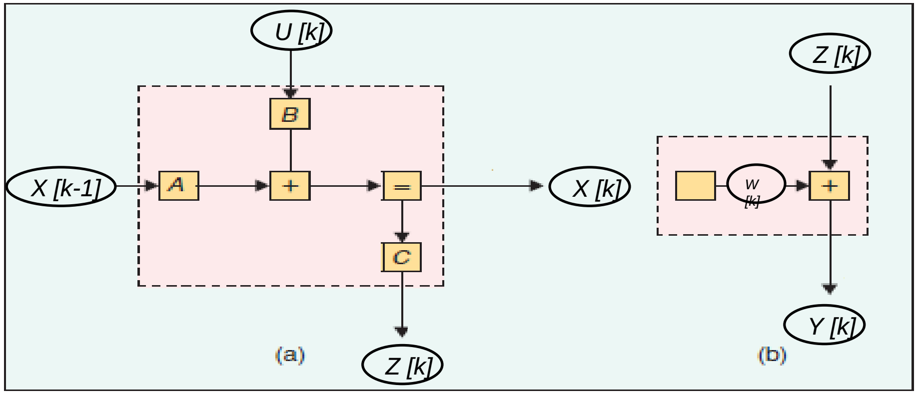


▲ 5. Zero-sum constraint node.

1.1 Factor Graphs



6. Classical state-space model.



7. Details of classical linear state-space model.

- ✓ Fig(6) and Fig(7) :

$$X[k] = AX[k-1] + BU[k]$$
$$Y[k] = CX[k] + W[k]$$

  - $k \in \mathbb{Z}$
  - $U[k], W[k], X[k], Y[k]$: real vectors
  - A, B, C: matrices of appropriate dimensions

- ✓ $U[k], W[k]$ are white Gaussian processes $\Rightarrow$ The corresponding nodes represent Gaussian probability distributions

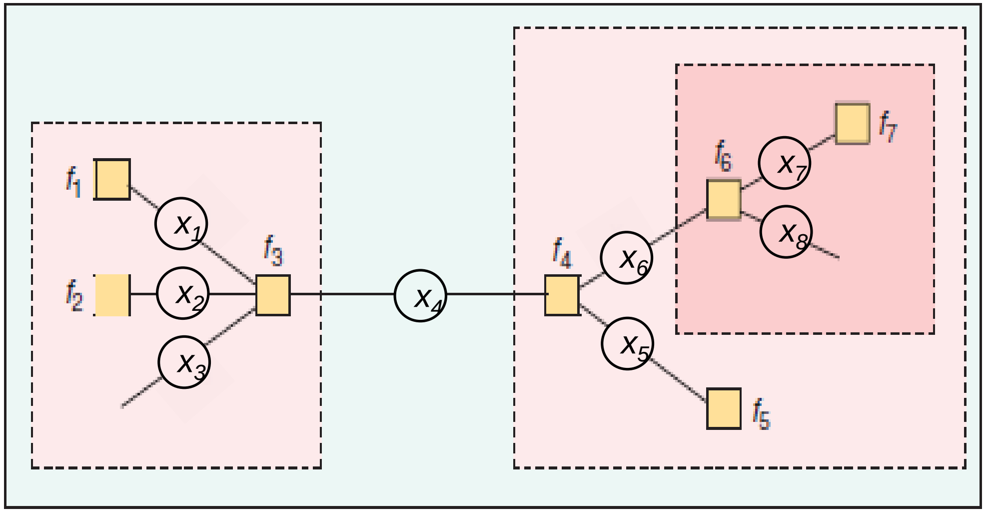
1.1 Factor Graphs

✓ **External variables**: only one edge attached

✓ **Internal variables**: two edges attached

✓ A big system f is an interconnection of subsystems
⇒ the variables connecting the subsystems are

- Internal to f
- External to the subsystems



1.2 Graphs of Codes

✓ **Error Correcting Block Code**

- C = n-length block code over A

  $\Rightarrow C \in A^n$
- $A = F$ and C is a subspace of F^n

  $\Rightarrow$ the code is linear
- $\forall$ linear code, $C = \{x \in F^n : xH^T = 0\} = \{uG : u \in F^k\}$

  : Encodes $u \in F^k$ of information symbols into the codeword $x = uG$

Ex. 3-length simple repetition code

$$C = \begin{Bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \end{Bmatrix}$$

$$H = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

$$G = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$$

1.2 Graphs of Codes

✓ Error Correction Example: **Hamming Code**

- A binary (7, 4, 3) Hamming Code:

  code length $n = 7$

  dimension $k = 4$

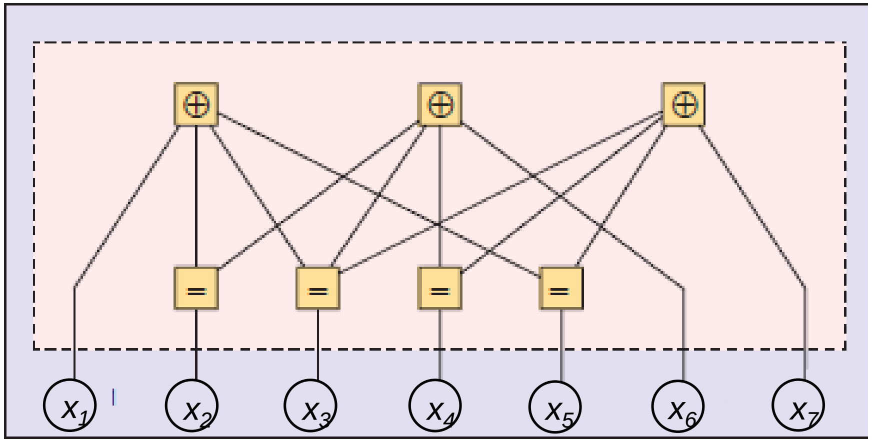
  minimum Hamming distance $= 3$

- Parity-check matrix

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{pmatrix}$$

- Membership indicator function:

$$I_C(x_1, \ldots, x_n)$$
$$= \delta(x_1 \oplus x_2 \oplus x_3 \oplus x_5) \bullet \delta(x_2 \oplus x_3 \oplus x_4 \oplus x_6) \bullet \delta(x_3 \oplus x_4 \oplus x_5 \oplus x_7)$$

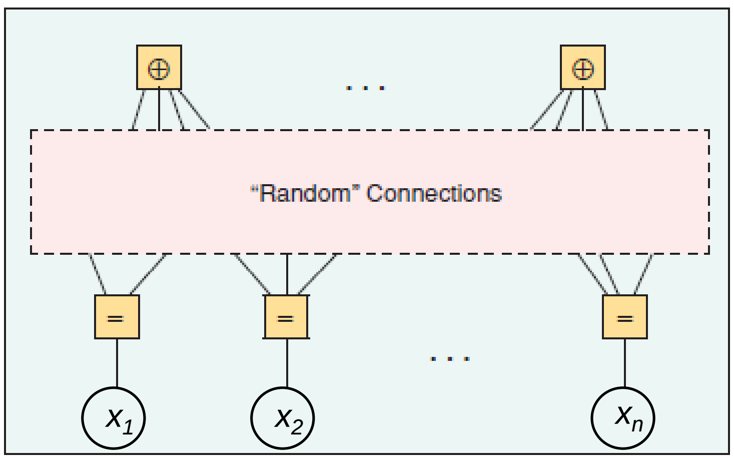


▲ 8. An FFG for the (7, 4, 3) binary Hamming code.

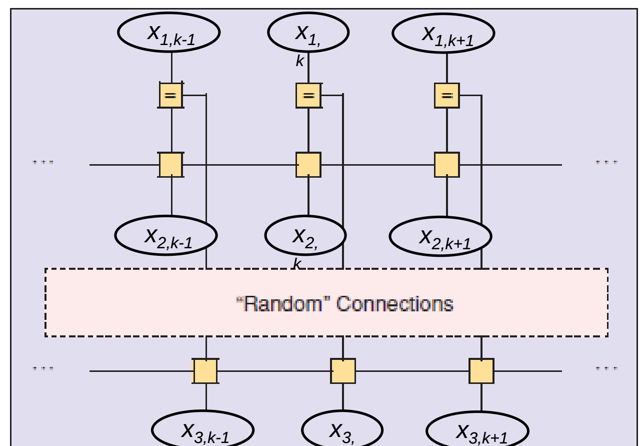
1.2 Graphs of Codes

✓ Error Correction Examples: **LDPC** and **Turbo Codes**

- **LDPC**
  - For blocks with large lengths
  - Sparse parity-check matrix
  - Main decoding algorithm: sum-product
- **Turbo**
  - Consists of two trellises sharing common symbols
  - Main decoding algorithm: sum-product



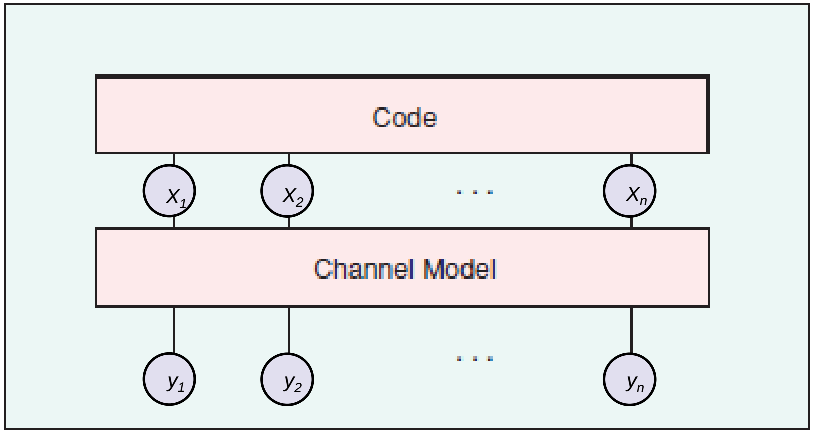
▲ 11. An FFG of a low-density parity-check code.



▲ 12. An FFG of a parallel concatenated code (turbo code).

✓ **Channel Model**

- A family $p(y|x)$ over
  - $y = (y_1, \ldots, y_n)$ as channel output
  - $x = (x_1, \ldots, x_n)$ as channel input
- FG results in $p(y|x)I_C(x)$



▲ 13. Joint code/channel FFG.

- ∀ fixed y,

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} \propto p(y|x)I_C(x)$$

- Joint channel FG represents a posteriori joint probability of $X_1, \ldots, X_n$

1.2 Graphs of Codes

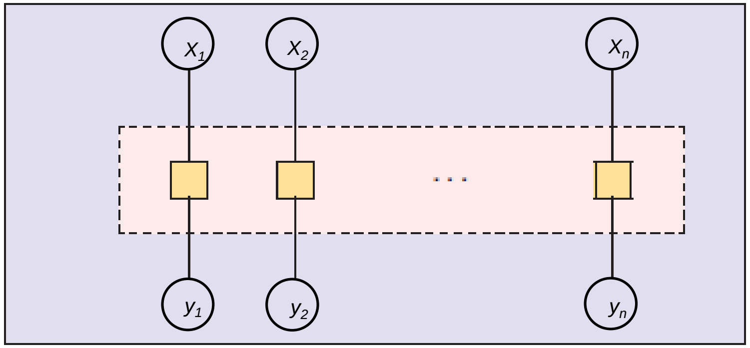
✓ Channel Model Examples: **Memoryless and State-Space Channels**

- Fig(14): **Memoryless** channel

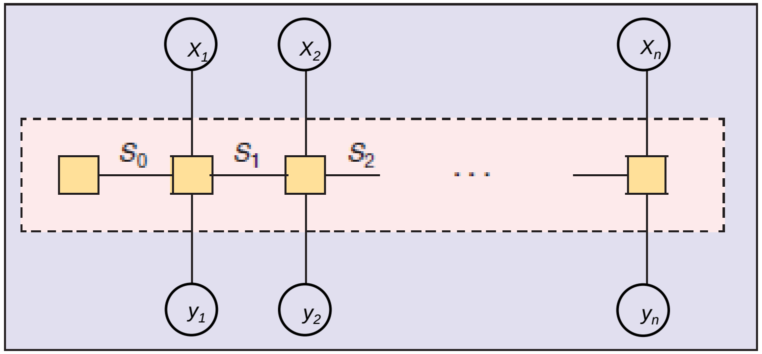
- $p(y|x) = \prod_{k=1}^{n} p(y_k|x_k)$

- Fig(15): **state-space** channel

$$p(y, s|x) = P(s_0) \prod_{k=1}^{n} p(y_k, s_k|x_k, s_{(k-1)})$$



▲ 14. Memoryless channel.



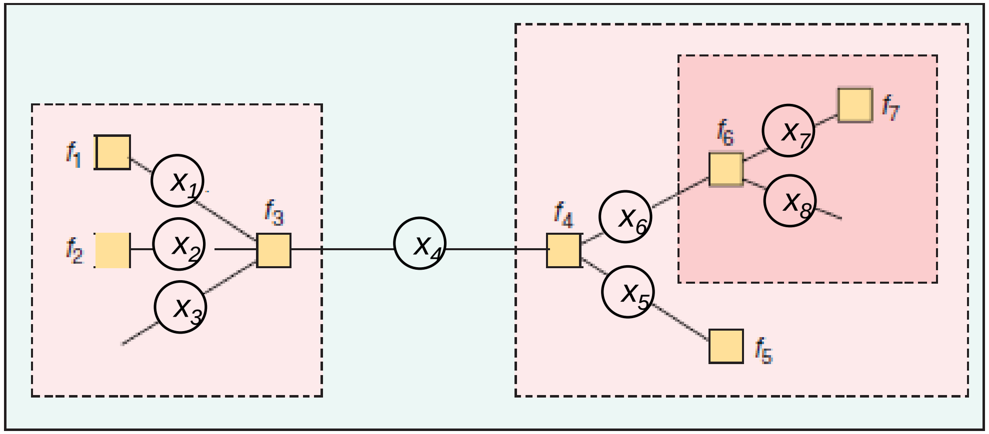
▲ 15. State-space channel model.

1.3 Belief Propagation Algorithms

✓ **Summary Operator**: Elimination of variables ("closing boxes")

- Ex. A discrete probability mass function $f(x_1, \ldots, x_8)$
  $\rightarrow$ marginal probability $p(x_4) = \sum_{x_1,x_2,x_3,x_5,x_6,x_7,x_8} f(x_1, \ldots, x_8)$
- Ex. A nonnegative function $f(x_1, \ldots, x_8)$
  $\rightarrow \rho(x_4) \triangleq \max_{x_1,x_2,x_3,x_5,x_6,x_7,x_8} f(x_1, \ldots, x_8)$

$f(x_1, \ldots x_8) =$
$\{f_1(x_1) f_2(x_2) f_3(x_1, x_2, x_3, x_4)\} \bullet$
$\{f_4(x_4, x_5, x_6) f_5(x_5)\} \bullet$
$\{f_6(x_6, x_7, x_8) f_7(x_7)\}$

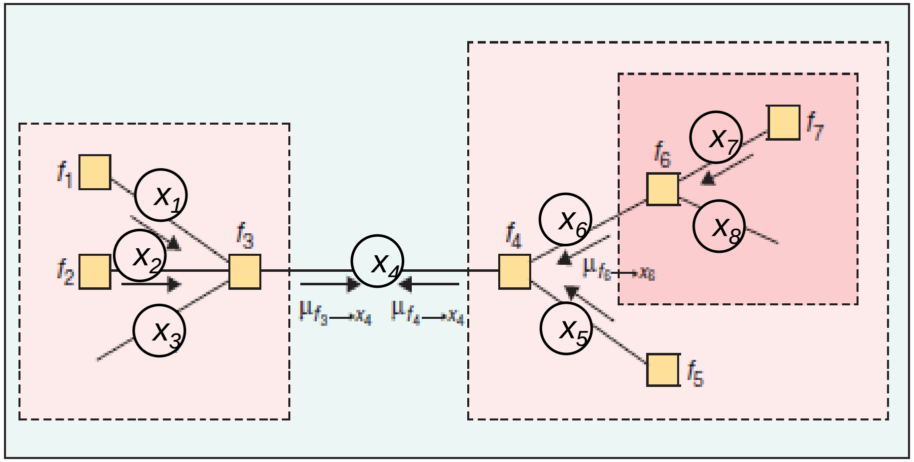


▲ 16. Elimination of variables: "closing the box" around subsystems.

1.3 Belief Propagation Algorithms

- Arithmetic manipulations to $p(x_4)$
$= \Sigma_{x_1}\Sigma_{x_2}\Sigma_{x_3}\Sigma_{x_5}\Sigma_{x_6}\Sigma_{x_7}\Sigma_{x_8} f(x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8)$
$= \{\Sigma_{x_1}\Sigma_{x_2}\Sigma_{x_3} f_3(x_1,x_2,x_3,x_4) f_1(x_1) f_2(x_2)\} \bullet$
$\{\Sigma_{x_5}\Sigma_{x_6} f_4(x_4,x_5,x_6) f_5(x_5)(\Sigma_{x_7}\Sigma_{x_8} f_6(x_6,x_7,x_8) f_7(x_7))\}$

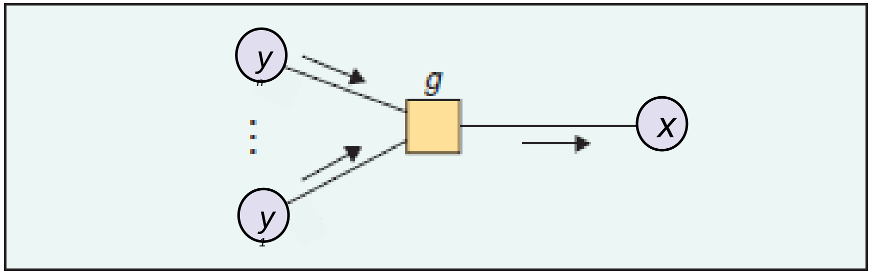
- Local Elimination Property: Successive local summaries lead to global summary
- Summary μ: "message" sent between the boxes



▲ 17. "Summarized" factors as "messages" in the FFG.

1.3 Belief Propagation Algorithms

- ✓ Message out of a terminal node = the corresponding function



▲ 18. Messages along a generic edge.

- ✓ Sum-Product Rule: for estimation

$$\mu_{g\to x}(x) \triangleq \Sigma_{y_1} \ldots \Sigma_{y_n} g(x, y_1, \ldots, y_n) \bullet \mu_{y_1\to g}(y_1) \ldots \mu_{y_n\to g}(y_n)$$

- ✓ Max-Sum Rule: for optimization

$$\mu_{g\to x}(x) \triangleq \max_{y_1} \ldots \max_{y_n} \log g(x, y_1, \ldots, y_n) + \Sigma_i \log \mu_{y_i\to g}(y_i)$$

2. Affinity Propagation

- 2.1

  Clustering by Belief Propagation: Affinity Propagation

- 2.2

  A Binary Model for Affinity Propagation

- 2.3

  Message Updates for Binary AP Model

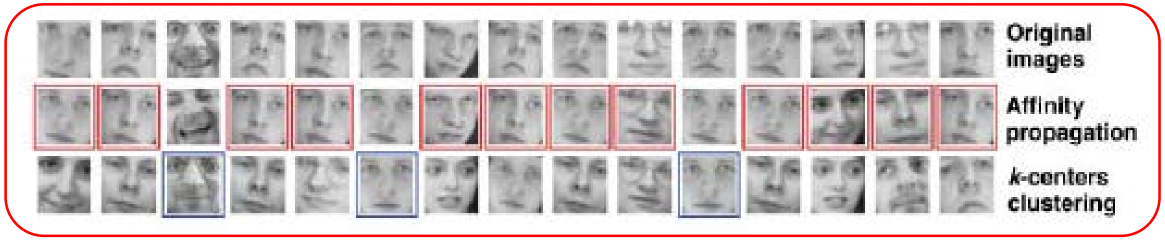
- 2.4

  Simple Applications of Binary AP Model

2.1 Clustering by Belief Propagation

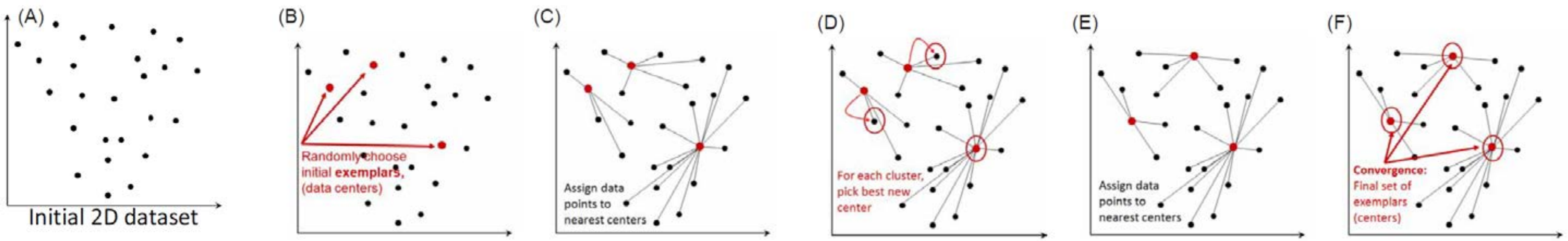
✓ Affinity Propagation(AP) Clustering: discrete variable application of belief propagation

✓ Where to use?
- detect genes in microarray data
- choose efficient facility locations
- cluster images of faces



2.1 Clustering by Belief Propagation

- Similarity: closeness of two data points
- Cluster head / exemplar: a point that represents its cluster
- Each data point belongs to its cluster head
  ⇔ each data point 'points' the exemplar of its cluster
- An exemplar point points itself as its exemplar
- Max-sum rule: maximize the sum of similarities of data points within clusters



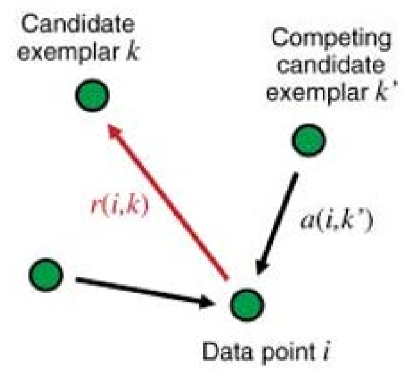
2.1 Clustering by Belief Propagation

✓ AP Input :
Real-valued similarities between data points.

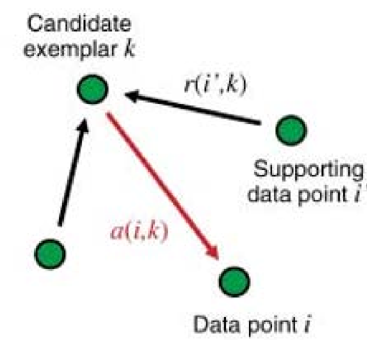
✓ Responsibility $\boldsymbol{r(i,k)}$
- **from** data point $\boldsymbol{i}$
  to candidate exemplar point $\boldsymbol{k}$
- reflects how well-suited point k is to serve as the exemplar

✓ Availability $\boldsymbol{a(i,k)}$
- **from** candidate exemplar point $\boldsymbol{k}$
  to point $\boldsymbol{i}$
- reflects how appropriate it would be for point i to choose point k as its exemplar

Sending
Responsibilities



Sending
Availabilities

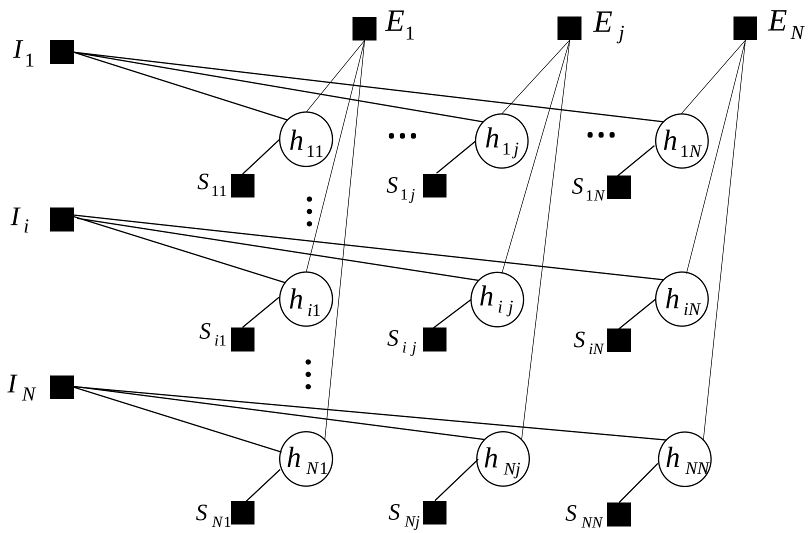


$$r(i,k) \leftarrow s(i,k) - \max_{k' \text{ s.t. } k' \neq k} \{a(i,k') + s(i,k')\}$$

$$a(i,k) \leftarrow \min\left\{0, r(k,k) + \sum_{i' \text{ s.t. } i' \notin \{i,k\}} \max\{0, r(i',k)\}\right\}$$

2.2 A Binary Model for Affinity Propagation

- ✓ Binary variables
- ✓ Each data point assigned to a single exemplar
- ✓ Pairwise Similarities $s_{ij}, \{i, j\} \subset \{1 ... N\}$
- ✓ N binary variables $\{h_{ij}\}_{j=1}^{N}$ associate with data point i
- ✓ i is pointing j as its exemplar $\Leftrightarrow h_{ij} = 1$
- ✓ $\sum_{j=1}^{N} h_{ij} = 1$



2.2 A Binary Model for Affinity Propagation

✓ Max-sum algorithm
Calculates the maximal value of the joint distribution and the corresponding variables.

✓ function → variables ■→○

“ I want you to be this value.”

$$\mu_{c\to i}(x_i) = \max_{X_c \setminus x_i} \left[\phi_c(x_c) + \sum_{j\in N(c)\setminus i} \mu_{j\to c}(x_j)\right]$$

✓ variables → function ○→■

“ I want to be this value.”

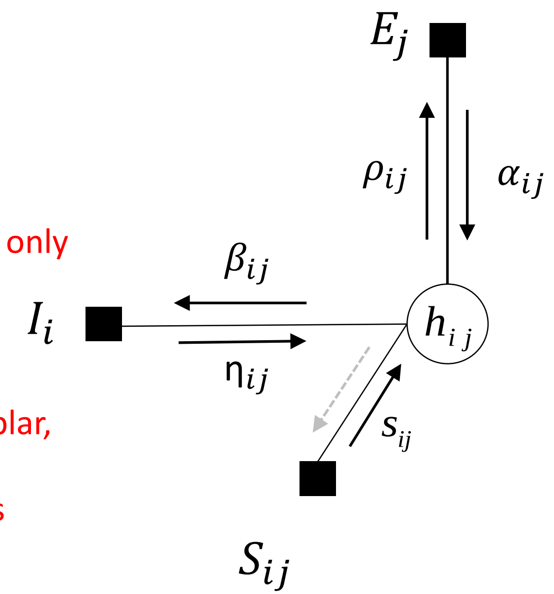
$$\mu_{i\to c}(x_i) = \sum_{b\in N(i)\setminus c} \mu_{b\to i}(x_i)$$

b = neighborhood nodes

2.2 A Binary Model for Affinity Propagation

✓ Using Max-sum formulation

✓ Five message types between variable and function nodes



✓ $I_i\ (h_{i:}) = \begin{cases} 0 & \text{if } \sum_j h_{ij} = 1, \\ -\infty & \text{otherwise.} \end{cases}$

Each data point chooses only one exemplar

✓ $E_j(h_{:j}) = \begin{cases} 0 & \text{if } h_{jj} \geq \max_i h_{ij}, \\ -\infty & \text{otherwise.} \end{cases}$

If i choose j as its exemplar, then j is its exemplar.
If no point choose j as its exemplar, j can be an exemplar of itself

✓ $S_{ij}(h_{ij}) = s_{ij} h_{ij}$

(Ex) Similarity $s_{ij} = \frac{1}{distance^2}$ = constant

Objective: maximize $F(\{h_{ij}\}) = \sum_{i,j} S_{ij}(h_{ij}) + \sum_i I_i(h_{i:}) + \sum_j E_j(h_{:j})$

2.3 Message Updates for Binary AP Model

✓ For $h_{ij} = 1$,

$$\beta_{ij}(1) = \mu_{h_{ij}\to I_i}(1) = \sum_{b\in N(h_{ij})\setminus I_i} \mu_{b\to h_{ij}}(1)$$

$$= S_{ij}(1) + \alpha_{ij}(1)$$

✓ For $h_{ij} = 0$,

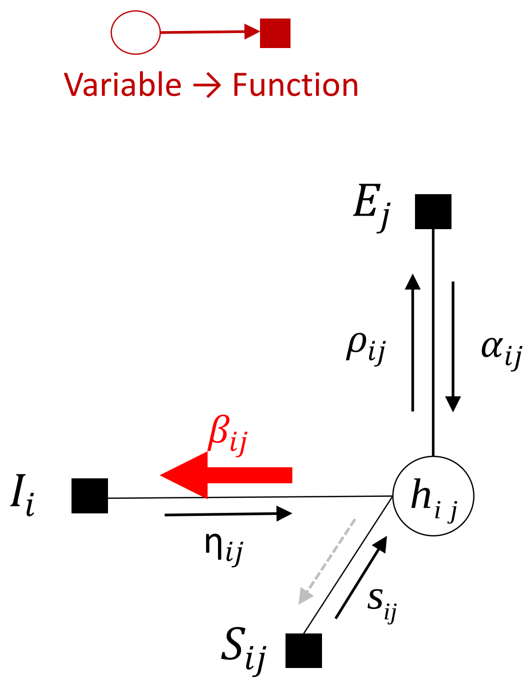
$$\beta_{ij}(0) = \underbrace{S_{ij}(0)}_{h_{ij}=0} + \alpha_{ij}(0)$$

✓ Taking the difference

$$\beta_{ij} = \beta_{ij}(1) - \beta_{ij}(0) \longrightarrow \text{denoted}$$

$$= [S_{ij}(1) - S_{ij}(0)] + [\alpha_{ij}(1) - \alpha_{ij}(0)]$$

$$= s_{ij} + \alpha_{ij}$$



2.3 Message Updates for Binary AP Model

✓ For $h_{ij} = 1$,

$$\rho_{ij}(1) = \mu_{h_{ij} \to E_j}(1) = \sum_{b \in N(h_{ij}) \setminus E_j} \mu_{b \to h_{ij}}(1)$$

$$= S_{ij}(1) + \eta_{ij}(1)$$

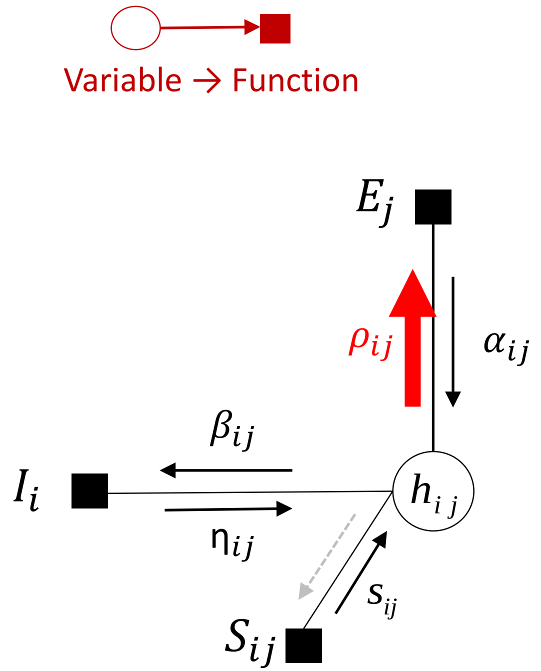
✓ For $h_{ij} = 0$,

$$\rho_{ij}(0) = S_{ij}(0) + \eta_{ij}(0)$$

$\to h_{ij} = 0$

✓ Taking the difference

$$\rho_{ij} = \rho_{ij}(1) - \rho_{ij}(0) \longrightarrow \text{denoted}$$
$$= [S_{ij}(1) - S_{ij}(0)] + [\eta_{ij}(1) - \eta_{ij}(0)]$$
$$= s_{ij} + \eta_{ij}$$



2.3 Message Updates for Binary AP Model

✓ For $h_{ij} = 1$

$$\eta_{ij}(1) = \mu_{I_i} \rightarrow h_{ij}(1)$$

$$= \max_{h_{ik}, k \neq j} [I_i(h_{i1}, \ldots, h_{ij}) = 1, \ldots, h_{iN}) + \sum_{h_{it} \in N(I_i) \setminus h_{ij}} \mu_{h_{it} \rightarrow I_i}(h_{it})]$$

$$= \max_{h_{ik}, k \neq j} [I_i(h_{i1}, \ldots, h_{ij} = 1, \ldots, h_{iN}) + \textstyle\sum_{t \neq j} \beta_{it}(h_{it})]$$

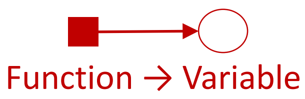
$$= \sum_{t \neq j} \beta_{it}(0)$$

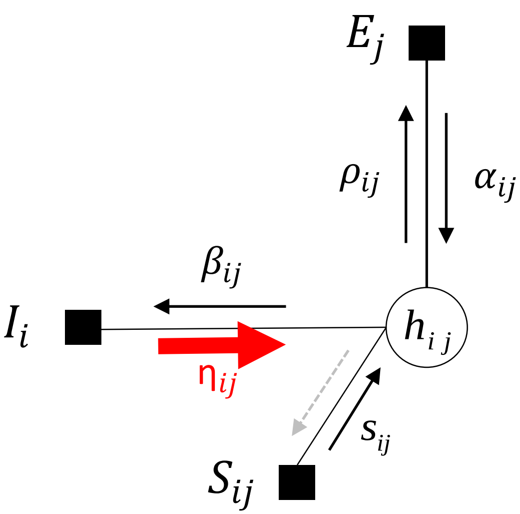
✓ For $h_{ij} = 0$

$$\eta_{ij}(0) = \max_{h_{ik}, k \neq j} [I_i(h_{i1}, \ldots, h_{ij} = 0, \ldots, h_{iN}) + \sum_{t \neq j} \beta_{it}(h_{it})]$$

$$= \max_{k \neq j} [\beta_{ik}(1) + \sum_{t \notin \{k, j\}} \beta_{it}(0)]$$ All except (j, k) are zero.

Choose one (exemplar node) of N-1





2.3 Message Updates for Binary AP Model

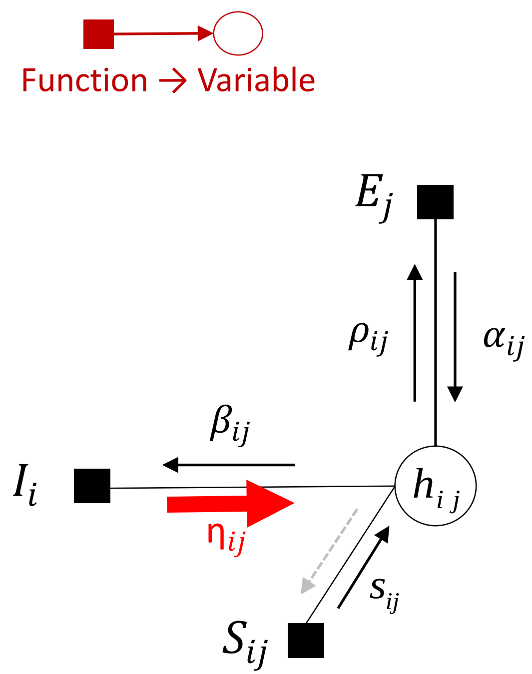
✓ Taking the difference $\eta_{ij}(1)-\eta_{ij}(0)$

$$
\begin{aligned}
\eta_{ij} &= \eta_{ij}(1) - \eta_{ij}(0) \\
&= -\max_{k\neq j}\left[\beta_{ik}(1) + \textstyle\sum_{t\notin k,j} \beta_{it}(0) - \sum_{t\neq j} \beta_{it}(0)\right] \\
&= -\max_{k\neq j}\left[\beta_{ik}(1) - \beta_{ik}(0)\right] = -\max_{k\neq j}\beta_{ik}.
\end{aligned}
$$

A – max B

= – (– A + max B)

= – (max (– A + B))



2.3 Message Updates for Binary AP Model

Whether k indicates j or not,
j can become an exemplar.

h_{kj} can be 0 , 1 both

- ✓ For $h_{ij} = 1,\ i = j$

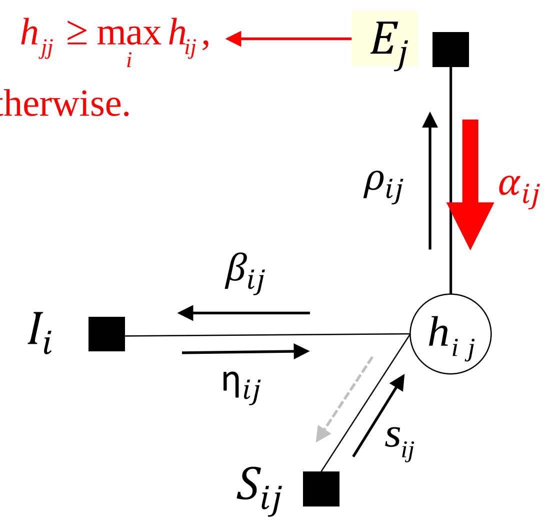
$$\alpha_{jj}(1) = \sum_{k \neq j} \max_{h_{kj}} \rho_{kj}(h_{kj}).$$

- ✓ For $h_{ij} = 0,\ i = j$

$$\alpha_{jj}(0) = \sum_{k \neq j} \rho_{kj}(0).$$

$$E_j(h_{:j}) = \begin{cases} 0 & \text{if } h_{jj} \geq \max_i h_{ij}, \\ -\infty & \text{otherwise.} \end{cases}$$

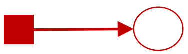
Function → Variable



- ✓ Taking the difference $\alpha_{jj}(1) - \alpha_{jj}(0)$

$$\boxed{\begin{aligned} \alpha_{jj} &= \alpha_{jj}(1) - \alpha_{jj}(0) \\ &= \sum_{k \neq j} \max(\rho_{kj}, 0) \end{aligned}}$$

2.3 Message Updates for Binary AP Model



✓ For $h_{ij} = 1, i \neq j$ → i has chosen j as its exemplar

$$\alpha_{ij}(1) = \max_{h_{kj}, k \neq i} [E_j(h_{1j}, \ldots, h_{ij} = 1, \ldots, h_{Nj}) + \sum_{k \neq i} \rho_{kj}(h_{kj})]$$

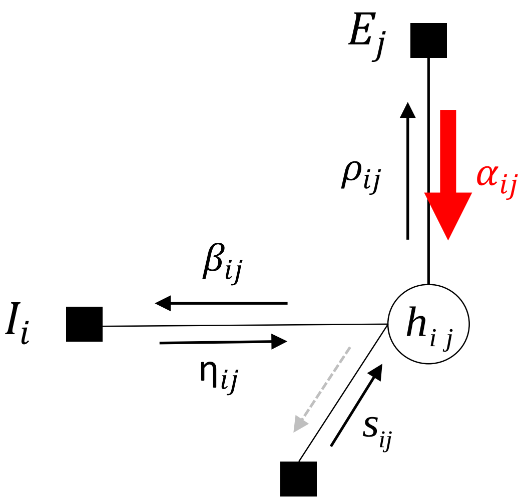
$$= \rho_{jj}(1) + \sum_{k \notin i,j} \max_{h_{kj}} \rho_{kj}(h_{kj}).$$

$\rho_{jj}(1)$ → j has chosen itself as an exemplar.

✓ For $h_{ij} = 0,\ i \neq j$

$$\alpha_{ij}(0) = \max[\underbrace{\rho_{jj}(1) + \sum_{h \notin i,j} \max_{h_{kj}} \rho_{kj}(h_{kj})}_{h_{jj} = 1},\ \underbrace{\sum_{k \neq i} \rho_{kj}(0)}_{h_{jj} = 0}].$$

No other point may choose j as an exemplar

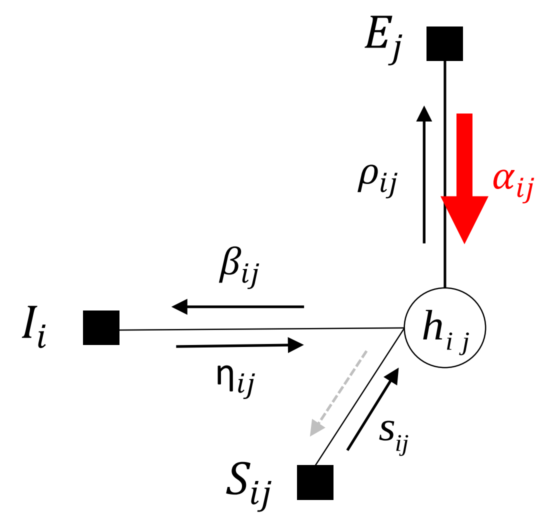


2.3 Message Updates for Binary AP Model

Function → Variable

✓ Taking the difference

$$
\begin{aligned}
\alpha_{ij} &= \alpha_{ij}(1) - \alpha_{ij}(0) \\
&= \max[0, \sum_{k \notin i} \rho_{kj}(0) - \rho_{jj}(1) - \sum_{l \notin i,j} \max(\rho_{lj}(1), \rho_{lj}(0))] \\
&= \max[\rho_{jj}(0) - \rho_{jj}(1) - \sum_{l \notin i,j} \rho_{lj}(0) - \max(\rho_{lj}(1) - \rho_{lj}(0))] \\
&= -\rho_{jj} + \sum_{l \notin i,j} [-\max(\rho_{lj}(1) - \rho_{lj}(0), 0)] \\
&= -\rho_{jj} + \sum_{l \notin i,j} [-\max(\rho_{lj}, 0)] \\
&= \min[0, \rho_{jj} + \sum_{l \notin i,j} [\max(0, \rho_{lj})]]
\end{aligned}
$$



2.3 Message Updates for Binary AP Model

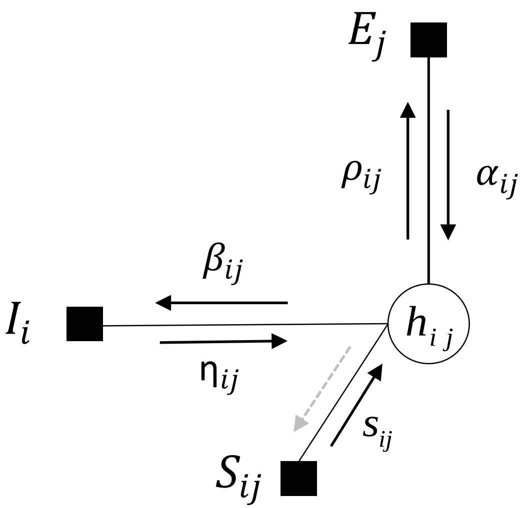
✓ To summarize, message update equations are :

$$\beta_{ij} = s_{ij} + \alpha_{ij}$$

$$\eta_{ij} = -\max_{k \neq j} \beta_{ik}$$

$$\rho_{ij} = s_{ij} + \eta_{ij}$$

$$\alpha_{ij} = \begin{cases} \sum_{k \neq j} \max(0, \ \rho_{kj}) & i = j \\ \min[0, \ \rho_{jj} + \sum_{k \notin i,j} \max(0, \rho_{kj})] & i \neq j \end{cases}$$



Availability messages $a(i,j)$

✓ Finally, express ρ in terms of α

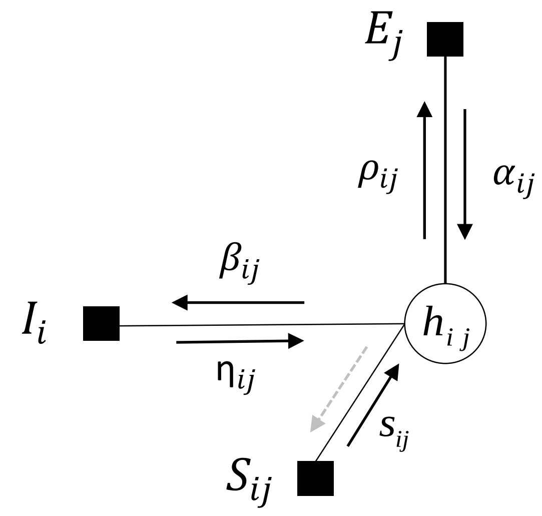
$$\rho_{ij} = s_{ij} + \eta_{ij} = s_{ij} - \max_{k\neq j}\beta_{ik} = s_{ij} - \max_{k\neq j}(s_{ik} + \alpha_{ik})$$

Responsibility messages $r(i,j)$

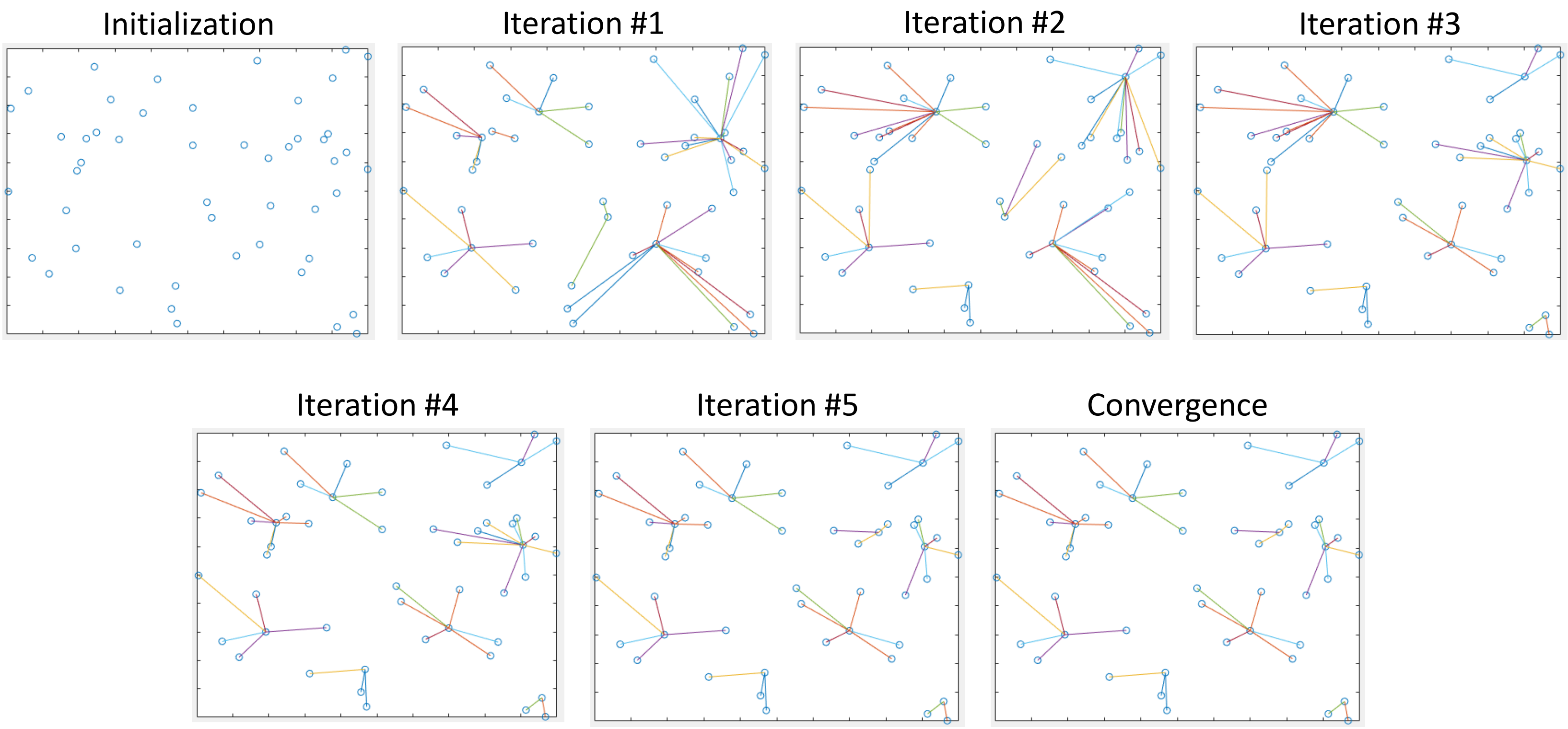
✓ Original Affinity Propagation message updates,

$$\rho_{ij} = s_{ij} - \max_{k\neq j}(s_{ik} + \alpha_{ik})$$

$$\alpha_{ij} = \begin{cases} \sum_{k\neq j} \max(0, \rho_{kj}) & i = j \\ \min[0, \rho_{jj} + \sum_{k\notin i,j} \max(0, \rho_{kj})] & i \neq j \end{cases}$$



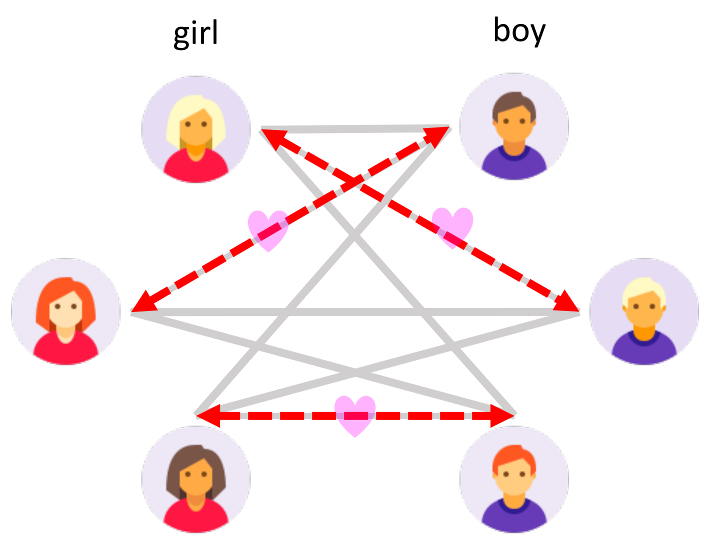
2.3 Message Updates for Binary AP Model



2.4 Simple Applications of Binary AP Model

✓ Group blind date

1. Max-Sum : maximizes the value added by all people satisfaction.
2. Max-Min : maximizes the value of the lowest satisfaction.



✓ Max-sum formulation

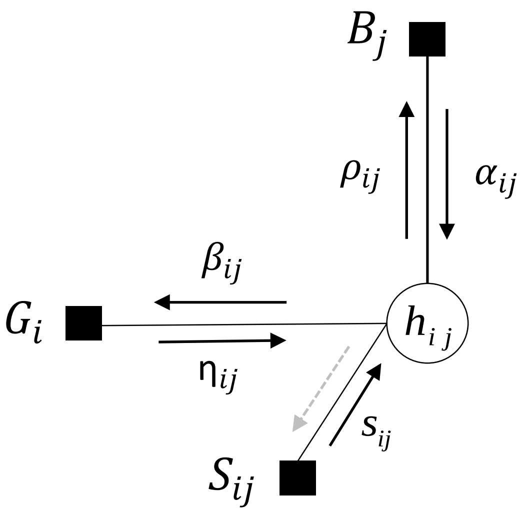
✓ $G_i\left(h_{i:}\right) = \begin{cases} 0 & \text{if } \sum_j h_{ij} = 1, \\ -\infty & \text{otherwise.} \end{cases}$

Each data point (girls) chooses only one boy

✓ $B_j\left(h_{:j}\right) = \begin{cases} 0 & \text{if } \sum_j h_{ij} = 1, \\ -\infty & \text{otherwise.} \end{cases}$

Each data point (boys) chooses only one girl

✓ $S_{ij}(h_{ij}) = s_{ij} h_{ij}$

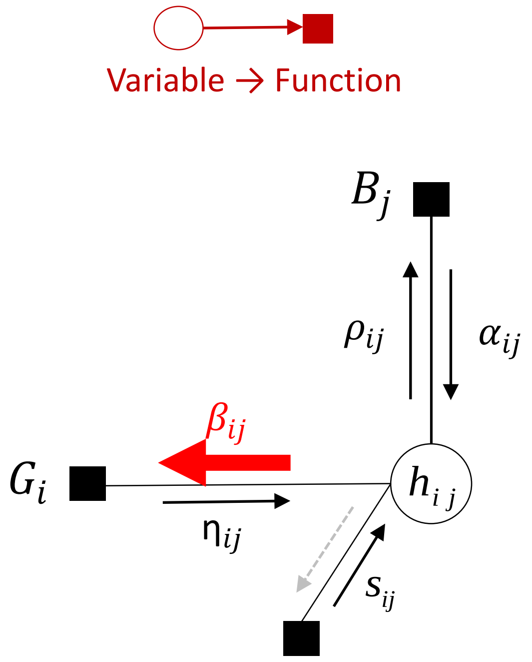


✓ For $h_{ij} = 1$,
$\beta_{ij}(1) = S_{ij}(1) + \alpha_{ij}(1)$

✓ For $h_{ij} = 0$,
$\beta_{ij}(0) = S_{ij}(0) + \alpha_{ij}(0)$

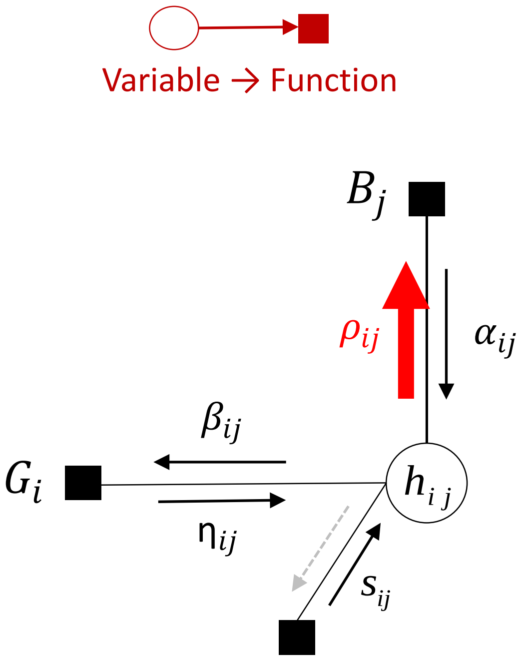
✓ Taking the difference

$\beta_{ij} = \beta_{ij}(1) - \beta_{ij}(0)$ → denoted
$= [S_{ij}(1) - S_{ij}(0)] + [\alpha_{ij}(1) - \alpha_{ij}(0)]$
$= s_{ij} + \alpha_{ij}$



2.4 Simple Applications of Binary AP Model

- For $h_{ij} = 1$,
  $\rho_{ij}(1) = S_{ij}(1) + \eta_{ij}(1)$
- For $h_{ij} = 0$,
  $\rho_{ij}(0) = S_{ij}(0) + \eta_{ij}(0)$
- Taking the difference
  $\rho_{ij} = \rho_{ij}(1) - \rho_{ij}(0)$ $\longrightarrow$ denoted
  $= [S_{ij}(1) - S_{ij}(0)] + [\eta_{ij}(1) - \eta_{ij}(0)]$
  $\boxed{= s_{ij} + \eta_{ij}}$



2.4 Simple Applications of Binary AP Model

✓ For $h_{ij} = 1$

$$\eta_{ij}(1) = \max_{h_{ik},k\neq j}[G_i(h_{i1},\dots,h_{ij}=1,\dots,h_{iN}) + \textstyle\sum_{t\neq j}\beta_{it}(h_{it})] \quad = \sum_{t\neq j}\beta_{it}(0)$$

✓ For $h_{ij} = 0$

$$\eta_{ij}(0) = \max_{h_{ik},k\neq j}[G_i(h_{i1},\dots,h_{ij}=0,\dots,h_{iN}) + \textstyle\sum_{t\neq j}\beta_{it}(h_{it})]$$

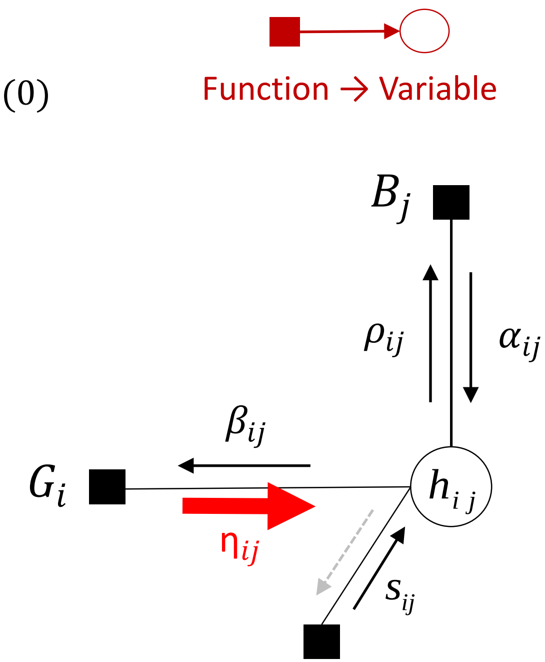
$$= \max_{k\neq j}[\beta_{ik}(1) + \sum_{t\notin\{k,j\}}\beta_{it}(0)]$$

All except $\{j,k\}$ are zero

Choose a boy of N-1 boys

✓ Taking the difference

$$\eta_{ij} = \eta_{ij}(1) - \eta_{ij}(0) = \boxed{-\max_{k\neq j}\beta_{ik}.}$$



2.4 Simple Applications of Binary AP Model

✓ For $h_{ij} = 1$

$$\alpha_{ij}(1) = \max_{h_{kj}, k\neq i}[B_j(h_{1j},\ldots,h_{ij}=1,\ldots,h_{Nj}) + \sum_{t\neq i} \rho_{tj}(h_{it})] \quad = \sum_{t\neq i} \rho_{tj}(0)$$

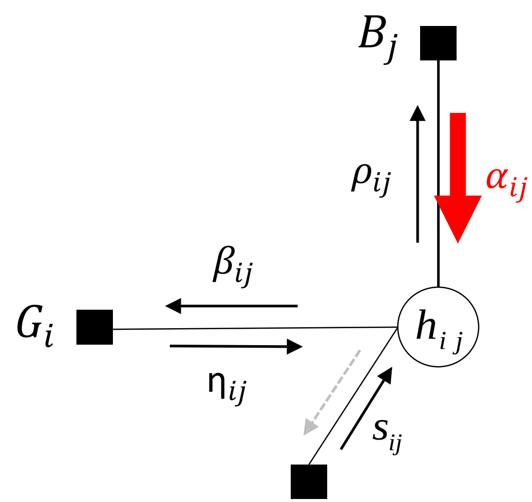
✓ For $h_{ij} = 0$

$$\alpha_{ij}(1) = \max_{h_{kj}, k\neq i}\left[B_j\left(h_{1j},\ldots,h_{ij}=1,\ldots,h_{Nj}\right) + \sum_{t\neq i} \rho_{tj}(h_{it})\right]$$

$$= \max_{k\neq i}[\rho_{kj}(1) + \sum_{t\notin\{k,i\}} \rho_{tj}(0)]$$

All except $\{j, k\}$ are zero

Choose a girl of N-1 girls



✓ Taking the difference

$$\alpha_{ij} = \alpha_{ij}(1) - \alpha_{ij}(0) \quad = \boxed{-\max_{k\neq j}\rho_{ik}.}$$

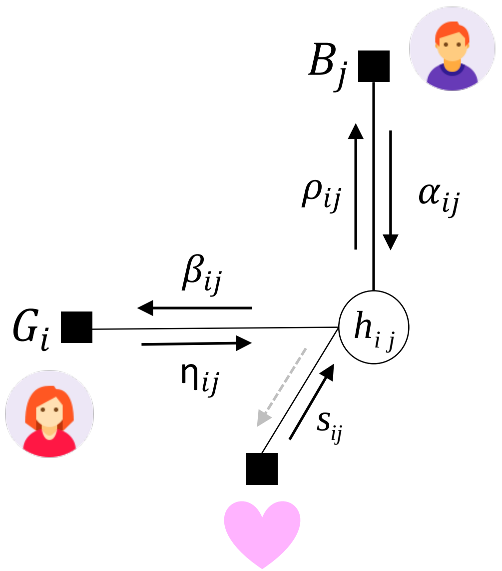
2.4 Simple Applications of Binary AP Model

✓ To summarize, the message update equations are :

$$\beta_{ij} = s_{ij} + \alpha_{ij}\,, \qquad \eta_{ij} = -\max_{k \neq j} \beta_{ik},$$
$$\rho_{ij} = s_{ij} + \eta_{ij}\,, \qquad \alpha_{ij} = -\max_{k \neq j} \rho_{ik}$$

✓ Finally Max-Sum message updates

$$\rho_{ij} = s_{ij} - \max_{k \neq j}(s_{ik} + \alpha_{ik}), \qquad \alpha_{ij} = -\max_{k \neq j} \rho_{ik}$$



2.4 Simple Applications of Binary AP Model

✓ Max-min formulation

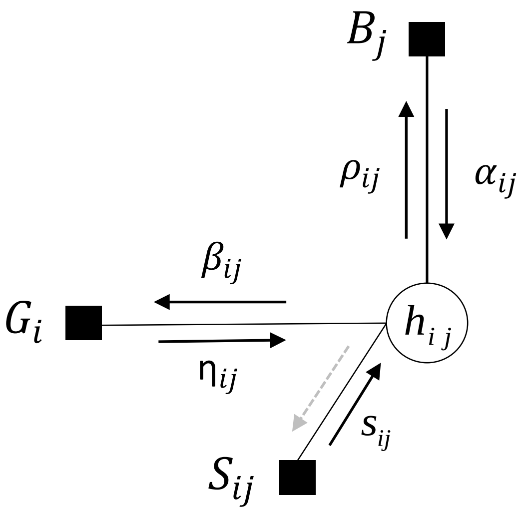
✓ $G_i\,(h_{i\,:}) = \begin{cases} 0 & \text{if } \sum_j h_{ij} = 1, \\ -\infty & \text{otherwise.} \end{cases}$

Each data point (girls) chooses only one boy

✓ $B_j\,(h_{:j}) = \begin{cases} 0 & \text{if } \sum_j h_{ij} = 1, \\ -\infty & \text{otherwise.} \end{cases}$

Each data point (boys) chooses only one girl

✓ $S_{ij}(h_{ij}) = s_{ij}h_{ij}$



2.4 Simple Applications of Binary AP Model

- ✓ Most of the process is the same as Max-Sum
- ✓ For $h_{ij} = 1$

$$\beta_{ij}(1) = \min[\alpha_{ij}(1),\ \ s_{ij}(1)]$$

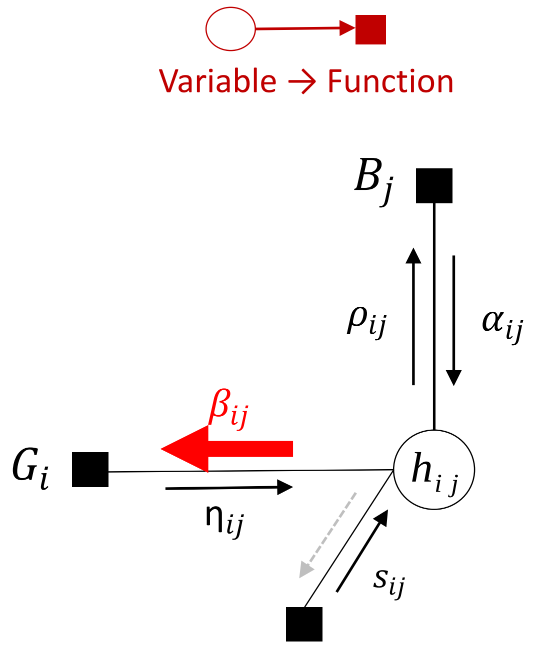
- ✓ For $h_{ij} = 0$

$$\beta_{ij}(0) = \min[\alpha_{ij}(0)] = \alpha_{ij}(0)$$

- ✓ Taking the difference,

$$\boxed{\beta_{ij} = \min[\alpha_{ij},\ \ s_{ij}(1) - \alpha_{ij}(0)]}$$

Output message difference can be expressed as function of Input message differences!!!!



2.4 Simple Applications of Binary AP Model

✓ Most of the process is the same as Max-Sum

✓ For $h_{ij} = 1$

$$\rho_{ij}(1) = \min[\eta_{ij}(1),\ \ s_{ij}(1)]$$

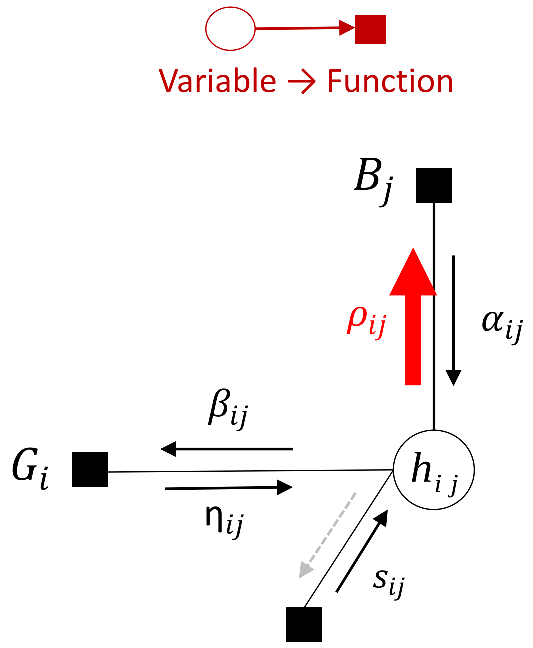
✓ For $h_{ij} = 0$

$$\rho_{ij}(0) = \min[\eta_{ij}(0)] \ = \ \eta_{ij}(0)$$

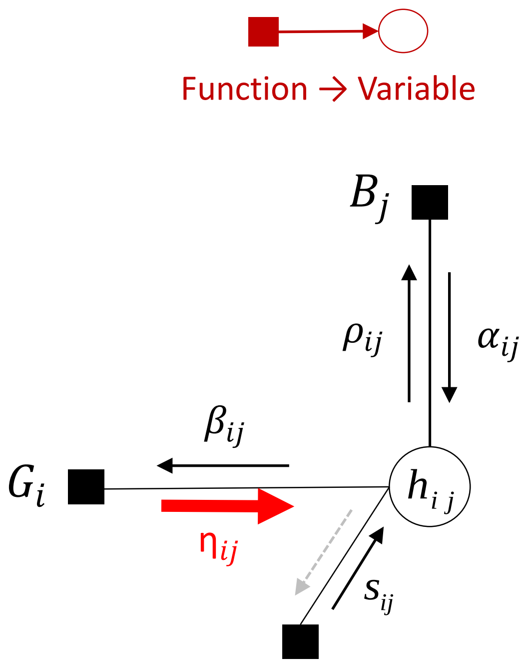
✓ Taking the difference,

$$\boxed{\rho_{ij} = \min[\eta_{ij},\ \ s_{ij}(1) - \eta_{ij}(0)]}$$

Output message difference can be expressed as function of Input message differences!!!!



2.4 Simple Applications of Binary AP Model



- ✓ For $h_{ij} = 1$

$$\eta_{ij}(1) = \max[\min_{t\neq j} \beta_{it}(0)] = \min_{t\neq j} \beta_{it}(0)$$

- ✓ For $h_{ij} = 0$

$$\eta_{ij}(0) = \max_{k\neq j}[\min[\, \beta_{ik}(1)\, , \min_{t\neq k,j} \beta_{it}(0)]\,]$$

- ✓ It is difficult to make the difference

2.4 Simple Applications of Binary AP Model

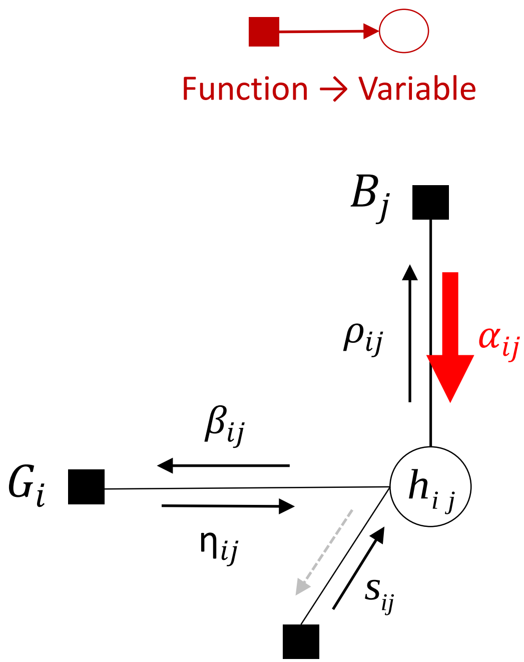
- ✓ For $h_{ij} = 1$

$$\alpha_{ij}(1) = \max[\min_{t \neq i} \rho_{tj}(0)] = \min_{t \neq i} \rho_{tj}(0)$$

- ✓ For $h_{ij} = 0$

$$\alpha_{ij}(0) = \max_{k \neq i}[\min[\rho_{kj}(1), \min_{t \neq k,i} \rho_{tj}(0)]]$$

- ✓ It is difficult to make the difference



2.4 Simple Applications of Binary AP Model

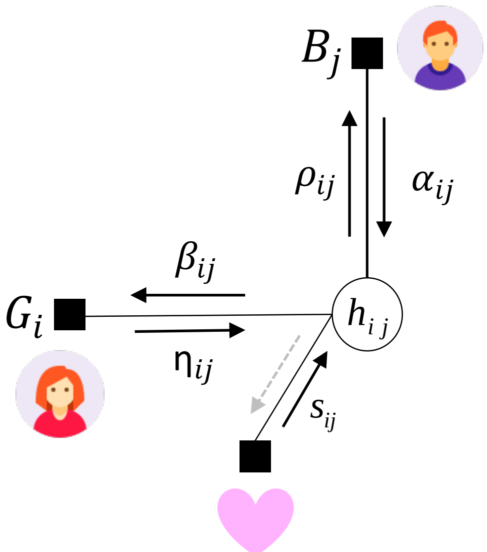
✓ To summarize, Max-Min message update equations are :

$$\beta_{ij}(h_{ij}) = \begin{cases} \min[\alpha_{ij}(1),\ \ s_{ij}(1)], & h_{ij} = 1 \\ \alpha_{ij}(0), & h_{ij} = 0 \end{cases}$$

$$\eta_{ij}(h_{ij}) = \begin{cases} \min\limits_{t \neq j} \beta_{it}(0), & h_{ij} = 1 \\ \max\limits_{t \neq j}[\min[\beta_{it}(1), \min\limits_{k \neq t,j} \beta_{ik}(0)]], & h_{ij} = 0 \end{cases}$$

$$\rho_{ij}(h_{ij}) = \begin{cases} \min[\eta_{ij}(1),\ \ s_{ij}(1)], & h_{ij} = 1 \\ \eta_{ij}(0), & h_{ij} = 0 \end{cases}$$

$$\alpha_{ij}(h_{ij}) = \begin{cases} \min\limits_{t \neq i} \rho_{tj}(0), & h_{ij} = 1 \\ \max\limits_{k \neq i}[\min[\rho_{kj}(1), \min\limits_{t \neq k,i} \rho_{tj}(0)]], & h_{ij} = 0 \end{cases}$$



2.4 Simple Applications of Binary AP Model

✓ Comparing the messages of Max-Sum and Max-Min :

Max-Sum

$$\beta_{ij}(h_{ij}) = \begin{cases} \alpha_{ij}(1) + s_{ij}(1), & h_{ij} = 1 \\ \&\alpha_{ij}(0), & h_{ij} = 0 \end{cases}$$

$$\eta_{ij}(h_{ij}) = \begin{cases} \sum_{t \neq j} \beta_{it}(0), & h_{ij} = 1 \\ \max_{k \neq j}[\beta_{ik}(1) + \sum_{t \notin \{k,j\}} \beta_{it}(0)], & h_{ij} = 0 \end{cases}$$

$$\rho_{ij}(h_{ij}) = \begin{cases} \eta_{ij}(1) + s_{ij}(1), & h_{ij} = 1 \\ \&\eta_{ij}(0), & h_{ij} = 0 \end{cases}$$

$$\alpha_{ij}(h_{ij}) = \begin{cases} \sum_{t \neq j} \rho_{tj}(0), & h_{ij} = 1 \\ \& \\ \max_{k \neq i}[\rho_{kj}(1) + \sum_{t \notin \{k,j\}} \rho_{tj}(0)], & h_{ij} = 0 \end{cases}$$

'+' changes to 'min'

'max' changes to 'max'

Max-Min

$$\beta_{ij}(h_{ij}) = \begin{cases} \min[\alpha_{ij}(1), \ s_{ij}(1)], & h_{ij} = 1 \\ \alpha_{ij}(0), & h_{ij} = 0 \end{cases}$$

$$\eta_{ij}(h_{ij}) = \begin{cases} \min_{t \neq j} \beta_{it}(0), & h_{ij} = 1 \\ \max_{t \neq j}[\min[\beta_{it}(1), \min_{k \neq t,j} \beta_{ik}(0)]], & h_{ij} = 0 \end{cases}$$

$$\rho_{ij}(h_{ij}) = \begin{cases} \min[\eta_{ij}(1), \ s_{ij}(1)], & h_{ij} = 1 \\ \eta_{ij}(0), & h_{ij} = 0 \end{cases}$$

$$\alpha_{ij}(h_{ij}) = \begin{cases} \min_{t \neq i} \rho_{tj}(0), & h_{ij} = 1 \\ \max_{k \neq i}[\min[\rho_{kj}(1), \min_{t \neq k,i} \rho_{tj}(0)]], & h_{ij} = 0 \end{cases}$$

2.4 Simple Applications of Binary AP Model

✓ $N_{girl} = 4,\ N_{boy} = 4$

random

✓ $w_{ij} = \begin{bmatrix} 8 & 2 & 8 & 1 \\ 7 & 8 & 7 & 2 \\ 1 & 2 & 9 & 9 \\ 4 & 9 & 8 & 4 \end{bmatrix}$

✓ Max-sum Discriminant

✓ $D_{ij} = \alpha_{ij} + \rho_{ij} > 0$

connect

$D_{ij} = \alpha_{ij} + \rho_{ij} < 0$

not connect

✓ Max-min Discriminant

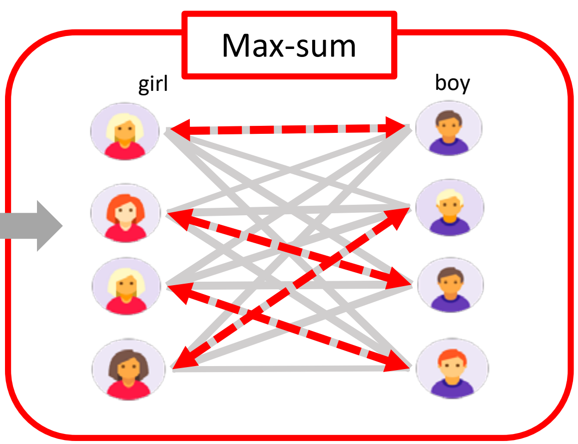
✓ $D_{ij} = \min\{\alpha_{ij}(1), \rho_{ij}(1)\} - \min\{\alpha_{ij}(0), \rho_{ij}(0)\} > 0$

connect

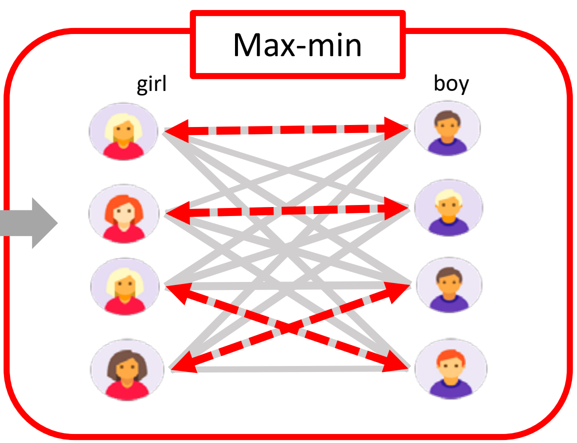
$D_{ij} = \min\{\alpha_{ij}(1), \rho_{ij}(1)\} - \min\{\alpha_{ij}(0), \rho_{ij}(0)\} < 0$

not connect

```
>> [[8. 2. 8. 1.]
    [7. 8. 7. 2.]
    [1. 2. 9. 9.]
    [4. 9. 8. 4.]]
```




```
>> [[ 1. -6. -1. -7.]
    [-1.  1. -1. -6.]
    [-7. -6. -4.  4.]
    [-4. -1.  1. -4.]]
```



3. Summary

- 1. Graphical Models

Simplified Visual Representation of Complex Systems with Local Interactions

- 2. Affinity Propagation

Simple Distributed Algorithm for General Class of Assignment Problems