Low Density Parity Check – Sum-product algorithm

Grupo de Interconexión de

Digital Signal Processing for Communication and Audio-visual Systems

Contents

Contents

Contents

Contents

Log-likelihood ratio

Definition

- Definition
  - $$\mathrm{LLR}^{(1)}(x) = \ln\left(\frac{P(x=1)}{P(x=0)}\right) = -\mathrm{LLR}^{(0)}(x) \qquad (1)$$
    $$\mathrm{LLR}^{(0)}(x) = \ln\left(\frac{P(x=0)}{P(x=1)}\right) \qquad (2)$$
- Obtaining probabilities from LLR
  - $$P(x=0) = \frac{P(x=0)}{P(x=0)+P(x=1)} = \frac{\frac{P(x=0)}{P(x=1)}}{\frac{P(x=0)}{P(x=1)}+1}$$
    $$P(x=0) = \frac{e^{\mathrm{LLR}^{(0)}(x)}}{1+e^{\mathrm{LLR}^{(0)}(x)}} = \frac{e^{-\mathrm{LLR}^{(1)}(x)}}{1+e^{-\mathrm{LLR}^{(1)}(x)}}$$
    $$P(x=0) = \frac{1}{1+e^{-\mathrm{LLR}^{(0)}(x)}} = \frac{1}{1+e^{\mathrm{LLR}^{(1)}(x)}} \qquad (3)$$

Log-likelihood ratio

Obtaining probabilities

- Obtaining probabilities from *LLR*
  - $$P(x=1) = \frac{e^{\mathrm{LLR}^{(1)}(x)}}{1+e^{\mathrm{LLR}^{(1)}(x)}} = \frac{e^{-\mathrm{LLR}^{(0)}(x)}}{1+e^{-\mathrm{LLR}^{(0)}(x)}}$$
    $$P(x=1) = \frac{1}{1+e^{-\mathrm{LLR}^{(1)}(x)}} = \frac{1}{1+e^{\mathrm{LLR}^{(0)}(x)}}$$
- Property
  - $$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{1-e^{-2\cdot x}}{1+e^{-2\cdot x}} = \frac{e^{2\cdot x}-1}{e^{2\cdot x}+1} \quad (4)$$
  - $$x = \operatorname{artanh}(\tanh(x)) = \frac{1}{2}\cdot\ln\left(\frac{1+\tanh(x)}{1-\tanh(x)}\right)$$
    $$x = \operatorname{artanh}(y) = \frac{1}{2}\cdot\ln\left(\frac{1+y}{1-y}\right) \quad (5)$$

Log-likelihood ratio

Property

- Property
  - $$1 - 2 \cdot P(x=1) = 1 - 2 \cdot \frac{e^{-\mathrm{LLR}^{(0)}(x)}}{1 + e^{-\mathrm{LLR}^{(0)}(x)}}$$
    $$1 - 2 \cdot P(x=1) = \frac{1 - e^{-\mathrm{LLR}^{(0)}(x)}}{1 + e^{-\mathrm{LLR}^{(0)}(x)}}$$
    $$1 - 2 \cdot P(x=1) = \tanh\left(\frac{\mathrm{LLR}^{(0)}(x)}{2}\right) \tag{6}$$
    $$1 - 2 \cdot P(x=0) = \tanh\left(\frac{\mathrm{LLR}^{(1)}(x)}{2}\right) \tag{7}$$

Contents

Sum-product algorithm

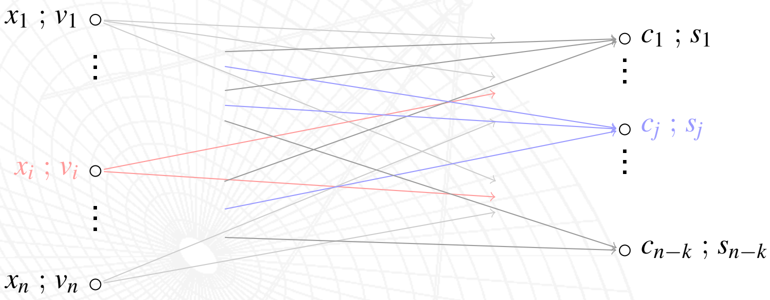
Sets of nodes

- Sets of nodes:
  - $A(c_j)$ is the set of variable nodes connected to *check* node c_j.
  - $A(v_i)$ is the set of check nodes connected to *variable* node v_i.

Sum-product algorithm

Sets of nodes

- Tanner graph, $A(v_i)$ and $A(c_j)$



Contents

Sum-product algorithm

Impact of Gaussian channel

- Probabilities corresponding to bit x_i given the received coordinate y_i assuming a *BPSK* modulation:
  - $$\begin{aligned} P(x_i = 0, v_i \to c_j, \text{channel}) &= \frac{1}{\sqrt{2\cdot\pi\cdot\sigma^2}} \cdot e^{\frac{-(y_i-(-1))^2}{2\cdot\sigma^2}} \cdot dy \\ P(x_i = 1, v_i \to c_j, \text{channel}) &= \frac{1}{\sqrt{2\cdot\pi\cdot\sigma^2}} \cdot e^{\frac{-(y_i-1)^2}{2\cdot\sigma^2}} \cdot dy \end{aligned}$$
  - $$\frac{P(x_i = 0, v_i \to c_j, \text{channel})}{P(x_i = 1, v_i \to c_j, \text{channel})} = e^{\frac{-2\cdot y_i}{\sigma^2}}$$
  - LLR value corresponding to bit x_i (influence of the channel) used in messages to check nodes $s \in A(x_i)$

$$\begin{aligned} \text{LLR}(x_i, v_i \to c_j, \text{channel}) &= \text{LLR}^{(0)}(x_i, v_i \to c_j, \text{channel}) \\ \text{LLR}(x_i, v_i \to c_j, \text{channel}) &= -\frac{2\cdot y_i}{\sigma^2} \end{aligned} \tag{8}$$

Sum-product algorithm

Estimation of probabilities at check nodes

- c_j connected to 2 variable nodes: v_{k_1} and v_{k_2}.
  $A(c_j) = \{v_{k_1}; v_{k_2}\}$
  The equation that must be satisfied is:

  $$s_j = x_{k_1} + x_{k_2} = 0$$

  - Suggestion to variable node v_{k_1}

    $$P(x_{k_1} = 0) = P(x_{k_2} = 0) = 1 - P(x_{k_2} = 1)$$
    $$P(x_{k_1} = 0) = \frac{1}{2} + \frac{1}{2} \cdot (1 - 2 \cdot P(x_{k_2} = 1))$$

Sum-product algorithm

Estimation of probabilities at check nodes

- c_j connected to 3 variable nodes: v_{k_1}, v_{k_2} and v_{k_3}.
  $A(c_j) = \{v_{k_1}; v_{k_2}; v_{k_3}\}$

$$s_j = x_{k_1} + x_{k_2} + x_{k_3} = 0$$

  - Suggestion to variable node v_{k_1}

$$\begin{aligned}
P(x_{k_1} = 0) &= P(x_{k_2} + x_{k_3} = 0) = P\left(\sum_{l=2}^{3} x_{k_l} = 0\right) \\
P(x_{k_1} = 0) &= P(x_{k_2} = 0) \cdot P(x_{k_3} = 0) + P(x_{k_2} = 1) \cdot P(x_{k_3} = 1) \\
P(x_{k_1} = 0) &= (1 - P(x_{k_2} = 1)) \cdot (1 - P(x_{k_3} = 1)) + \\
&\quad P(x_{k_2} = 1) \cdot P(x_{k_3} = 1) \\
P(x_{k_1} = 0) &= 1 - P(x_{k_2} = 1) - P(x_{k_3} = 1) + 2 \cdot P(x_{k_2} = 1) \cdot P(x_{k_3} = 1) \\
P(x_{k_1} = 0) &= \frac{1}{2} + \frac{1}{2} \cdot (1 - 2 \cdot P(x_{k_2} = 1) - 2 \cdot P(x_{k_3} = 1) + \\
&\quad 4 \cdot P(x_{k_2} = 1) \cdot P(x_{k_3} = 1))
\end{aligned}$$

Sum-product algorithm

Estimation of probabilities at check nodes

- c_j connected to 3 variable nodes: v_{k_1}, v_{k_2} and v_{k_3}.
  $A(c_j) = \{v_{k_1}; v_{k_2}; v_{k_3}\}$

$$s_j = x_{k_1} + x_{k_2} + x_{k_3} = 0$$

  - Suggestion to variable node v_{k_1}

$$P(x_{k_1} = 0) = \frac{1}{2} + \frac{1}{2} \cdot (1 - 2 \cdot P(x_{k_2} = 1)) \cdot (1 - 2 \cdot P(x_{k_3} = 1))$$

$$P(x_{k_1} = 0) = P\left(\sum_{l=2}^{3} x_{k_l} = 0\right) = \frac{1}{2} + \frac{1}{2} \cdot \prod_{l=2}^{3} (1 - 2 \cdot P(x_{k_l} = 1))$$

Sum-product algorithm

Estimation of probabilities at check nodes

- c_j connected to n variable nodes: v_{k_1}, v_{k_2},...and v_{k_n}.
  $A(c_j) = \{v_{k_1}; v_{k_2}; \ldots; v_{k_n}\}$
  $$s_j = \sum_{x \in A(s_j)} x = 0$$
  - Assumption
    $$P\left(\sum_{l=2}^{n-1} x_{k_l} = 0\right) = \frac{1}{2} + \frac{1}{2} \cdot \prod_{l=2}^{n-1} \left(1 - 2 \cdot P(x_{k_l} = 1)\right)$$
  - Suggestion to variable node v_{k_1}
    $$P(x_{k_1} = 0) = P\left(\sum_{l=2}^{n} x_{k_l} = 0\right)$$
    $$P(x_{k_1} = 0) = P(x_{k_n} = 0) \cdot P\left(\sum_{l=2}^{n-1} x_{k_l} = 0\right) + P(x_{k_n} = 1) \cdot \left(1 - P\left(\sum_{l=2}^{n-1} x_{k_l} = 0\right)\right)$$

Sum-product algorithm

Estimation of probabilities at check nodes

- c_j connected to n variable nodes: v_{k_1}, v_{k_2},...and v_{k_n}.
  $A(c_j) = \{v_{k_1}; v_{k_2}; \ldots; v_{k_n}\}$

$$s_j = \sum_{x \in A(s_j)} x = 0$$

  - Suggestion to variable node v_{k_1}

$$\begin{aligned} P(x_{k_1} = 0) &= (1 - P(x_{k_n} = 1)) \cdot P\left(\sum_{l=2}^{n-1} x_{k_l} = 0\right) + P(x_{k_n} = 1) \cdot \\ &\quad \left(1 - P\left(\sum_{l=2}^{n-1} x_{k_l} = 0\right)\right) \\ P(x_{k_1} = 0) &= P\left(\sum_{l=2}^{n-1} x_{k_l} = 0\right) \cdot (1 - 2 \cdot P(x_{k_n} = 1)) + P(x_{k_n} = 1) \end{aligned}$$

Sum-product algorithm

Estimation of probabilities at check nodes

- c_j connected to n variable nodes: v_{k_1}, v_{k_2},...and v_{k_n}.

$A(c_j) = \{v_{k_1}; v_{k_2}; \ldots; v_{k_n}\}$

$$s_j = \sum_{x \in A(s_j)} x = 0$$

- Suggestion to variable node v_{k_1}

$$P(x_{k_1} = 0) = \left(\frac{1}{2} + \frac{1}{2} \cdot \prod_{l=2}^{n-1} (1 - 2 \cdot P(x_{k_l} = 1))\right) \cdot (1 - 2 \cdot P(x_{k_n} = 1)) + P(x_{k_n} = 1)$$

$$P(x_{k_1} = 0) = \frac{1}{2} - P(x_{k_n} = 1) + \frac{1}{2} \cdot \prod_{l=2}^{n} (1 - 2 \cdot P(x_{k_l} = 1)) + P(x_{k_n} = 1)$$

$$P(x_{k_1} = 0) = \frac{1}{2} + \frac{1}{2} \cdot \prod_{l=2}^{n} (1 - 2 \cdot P(x_{k_l} = 1))$$

Contents

Sum-product algorithm

Messages from check nodes to variable nodes

- From the previous slides, c_j is connected to n variable nodes: v_{k_1}, v_{k_2},... and v_{k_n}.

$$A(c_j) = \{v_{k_1}; v_{k_2}; \ldots; v_{k_n}\}$$

$$s_j = \sum_{v_i \in A(c_j)} x_i = 0$$

  - Suggestion to variable node v_i

$$P(x_i = 0, c_j \to v_i) = \frac{1}{2} + \frac{1}{2} \cdot \prod_{v_l \in A(c_j) \setminus v_i} \left(1 - 2 \cdot P(x_l = 1, v_l \to c_j)\right) \quad (9)$$

and from (6)

$$P(x_i = 0, c_j \to v_i) = \frac{1}{2} + \frac{1}{2} \cdot \prod_{v_l \in A(c_j) \setminus v_i} \tanh\left(\frac{\mathrm{LLR}^{(0)}(x_l, v_l \to c_j)}{2}\right) \quad (10)$$

Sum-product algorithm

Messages from check nodes to variable nodes

- From the previous slides, c_j is connected to n variable nodes: v_{k_1}, v_{k_2}, ... and v_{k_n}.
  $A(c_j) = \{v_{k_1}; v_{k_2}; \ldots; v_{k_n}\}$
  - Message to node v_i (LLR)

$$\begin{aligned}
\mathrm{LLR}(x_i, c_j \to v_i) &= \mathrm{LLR}^{(0)}(x_i, c_j \to v_i) = \ln\left(\frac{P(x_i = 0, c_j \to v_i)}{P(x_i = 1, c_j \to v_i)}\right) \\
\mathrm{LLR}(x_i, c_j \to v_i) &= \ln\left(\frac{\frac{1}{2} + \frac{1}{2} \cdot \prod_{v_l \in A(c_j)\setminus v_i} \tanh\left(\frac{\mathrm{LLR}(x_l, v_l \to c_j)}{2}\right)}{1 - \frac{1}{2} - \frac{1}{2} \cdot \prod_{v_l \in A(c_j)\setminus v_i} \tanh\left(\frac{\mathrm{LLR}(x_l, v_l \to c_j)}{2}\right)}\right) \\
\mathrm{LLR}(x_i, c_j \to v_i) &= \ln\left(\frac{1 + \prod_{v_l \in A(c_j)\setminus v_i} \tanh\left(\frac{\mathrm{LLR}(x_l, v_l \to c_j)}{2}\right)}{1 - \prod_{v_l \in A(c_j)\setminus v_i} \tanh\left(\frac{\mathrm{LLR}(x_l, v_l \to c_j)}{2}\right)}\right)
\end{aligned}$$

Sum-product algorithm

Messages from check nodes to variable nodes

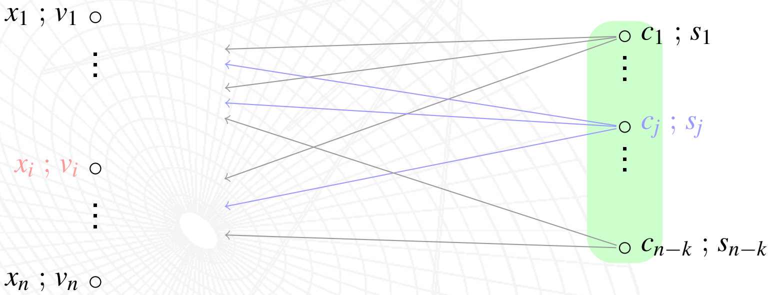
- From the previous slides, c_j is connected to n variable nodes: v_{k_1}, v_{k_2},... and v_{k_n}.
  $A(c_j) = \{v_{k_1}; v_{k_2}; \ldots; v_{k_n}\}$
  - Message to node v_i (LLR) –from (5)–

$$\text{LLR}(x_i, c_j \to v_i) \quad = \quad 2 \cdot \text{artanh}\left(\prod_{v_l \in A(c_j) \setminus v_i} \tanh\left(\frac{\text{LLR}(x_l, v_l \to c_j)}{2} \right) \right) \tag{11}$$

Sum-product algorithm

Messages from check nodes to variable nodes

$$\mathrm{LLR}(x_i)_{c_j\to v_i}^{(\text{step } m)} = 2\cdot \operatorname{artanh}\left(\prod_{v_l\in A(c_j)\setminus v_i} \tanh\left(\frac{\mathrm{LLR}(x_l)_{v_l\to c_j}^{(\text{step } m-1)}}{2}\right)\right)$$



Contents

Sum-product algorithm

Messages from variable nodes to check nodes

- From the previous slides, v_i connected m variable nodes: $c_{k_1}, c_{k_2}, \ldots$ and c_{k_m}.
  $A(v_i) = \{c_{k_1}; c_{k_2}; \ldots; c_{k_m}\}$
  - Message to check node c_j (LLR)

$$\text{LLR}(x_i, v_i \to c_j) = \text{LLR}^{(0)}(x_i, v_i \to c_j) = \ln\left(\frac{P(x_i = 0, v_i \to c_j)}{P(x_i = 1, v_i \to c_j)}\right)$$

    - LLR value sent to the check nodes includes the influence of the channel and the remaining check nodes

$$\text{LLR}^{(0)}(x_i, v_i \to c_j) = \text{LLR}^{(0)}(x_i, v_i \to c_j, \text{channel}) + \sum_{c_k \in A(v_i) \setminus c_j} \text{LLR}^{(0)}(x_i, c_k \to v_i)$$

$$\text{LLR}(x_i, v_i \to c_j) = \text{LLR}(x_i, v_i \to c_j, \text{channel}) + \sum_{c_k \in A(v_i) \setminus c_j} \text{LLR}(x_i, c_k \to v_i) \tag{12}$$

Sum-product algorithm

Messages from variable nodes to check nodes

- From the previous slides, v_i connected m variable nodes: $c_{k_1}, c_{k_2}, \ldots$ and c_{k_m}.
  $A(v_i) = \{c_{k_1}; c_{k_2}; \ldots; c_{k_m}\}$
  - Message to check node c_j (LLR)
    From (8) and (11)

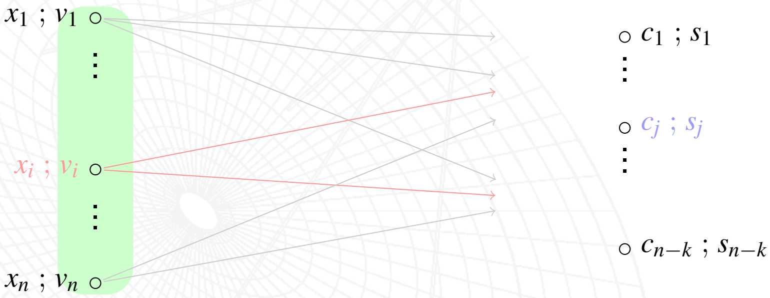
$$\text{LLR}(x_i, v_i \to c_j) = \text{LLR}(x_i, v_i \to c_j, \text{channel}) + \sum_{c_k \in A(v_i) \setminus c_j} \text{LLR}(x_i, c_k \to v_i)$$

$$\text{LLR}(x_i, v_i \to c_j) = -\frac{2 \cdot y_i}{\sigma^2} + \sum_{c_k \in A(v_i) \setminus c_j} 2 \cdot \text{artanh}\left(\prod_{v_l \in A(c_k) \setminus v_i} \tanh\left(\frac{\text{LLR}(x_l, v_l \to c_k)}{2} \right) \right) \tag{13}$$

Sum-product algorithm

Messages from variable nodes to check nodes

$$\text{LLR}(x_i)^{(\text{step } m)}_{v_i \to c_j} = -\frac{2 \cdot y_i}{\sigma^2} + \sum_{c_k \in A(v_i) \setminus c_j} \text{LLR}(x_i)^{(\text{step } m)}_{c_k \to v_i}$$



Contents

Sum-product algorithm

Stopping criterion

- Meaning of LLR
  - $\text{LLR}(x_i) > 0$, x_i more likely to be 0
  - $\text{LLR}(x_i) < 0$, x_i more likely to be 1
- Meaning of $\text{LLR}(x_{k_1}) \cdot \text{LLR}(x_{k_2})$
  - $\text{LLR}(x_{k_1}) \cdot \text{LLR}(x_{k_2}) > 0$, $x_{k_1} + x_{k_2}$ more likely to be 0
  - $\text{LLR}(x_{k_1}) \cdot \text{LLR}(x_{k_2}) < 0$, $x_{k_1} + x_{k_2}$ more likely to be 1

Sum-product algorithm

Stopping criterion

- From the previous slides, c_j is connected to n variable nodes: v_{k_1}, v_{k_2},... and v_{k_n}.
  $A(c_j) = \{v_{k_1}; v_{k_2}; \ldots; v_{k_n}\}$
  From equation (13)

$$s_j = \sum_{v_i \in A(c_j)} x_i = 0$$

  - $$\begin{aligned}\mathrm{LLR}(x_i, v_i \to c_j) &= \mathrm{LLR}(x_i, v_i \to c_j, \text{channel}) + \sum_{c_k \in A(v_i)\setminus c_j} \mathrm{LLR}(x_i, c_k \to v_i) \\ \mathrm{LLR}(x_i, v_i \to c_j) &= -\frac{2 \cdot y_i}{\sigma^2} + \sum_{c_k \in A(v_i)\setminus c_j} 2 \cdot \mathrm{artanh}\left(\prod_{v_l \in A(c_k)\setminus v_i} \tanh\left(\frac{\mathrm{LLR}(x_l, v_l \to c_k)}{2}\right)\right)\end{aligned}$$
  - For all c_j

$$\prod_{v_i \in A(c_j)} \mathrm{LLR}(x_i, v_i \to c_j) > 0 \qquad (14)$$

Sum-product algorithm

Final LLR

- LLR

$$\begin{aligned}
\mathrm{LLR}(x_i) &= \mathrm{LLR}(x_i, \text{channel}) + \sum_{c_k \in A(v_i)} \mathrm{LLR}(x_i, c_k \rightarrow v_i) \\
\mathrm{LLR}(x_i) &= -\frac{2 \cdot y_i}{\sigma^2} + \sum_{c_k \in A(v_i)} 2 \cdot \operatorname{artanh}\left(\prod_{v_l \in A(c_k) \backslash v_i} \tanh\left(\frac{\mathrm{LLR}(x_l, v_l \rightarrow c_k)}{2} \right) \right)
\end{aligned} \tag{15}$$

Contents

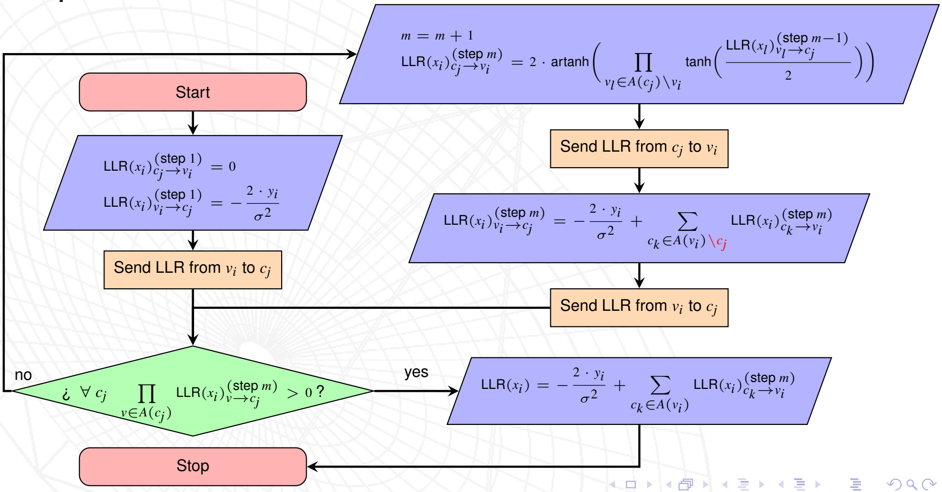
Sum-product algorithm

Algorithm

1. Step $m = 1$. Evaluate at all v_i and send to c_j.
   - $\text{LLR}(x_i)^{(\text{step } 1)}_{v_i \to c_j}$ –see equation (13)– assuming:
     - $\text{LLR}(x_i)^{(\text{step } 1)}_{c_j \to v_i} = 0$
2. The stopping criterion is applied at all s_j, equation(14)
   - 2.1 If it holds for all c_j, $\overrightarrow{x}$ is a valid codeword according to $\text{LLR}(x_i)^{(\text{step } m)}_{v_i \to c_j}$. Evaluate:
     - $\text{LLR}(x_i)$ –see equation (15)– and stop.
   - 2.2 If it doesn't hold for all c_j:
     - Step $m = m + 1$
     - Evaluate at all c_j and send to v_i
       - $\text{LLR}(x_i)^{(\text{step } m)}_{c_j \to v_i}$ –see equation (11)–
     - Evaluate at all v_i and send to c_j
       - $\text{LLR}(x_i)^{(\text{step } m)}_{v_i \to c_j}$ –see equation (13)–
     - Go to phase 2.

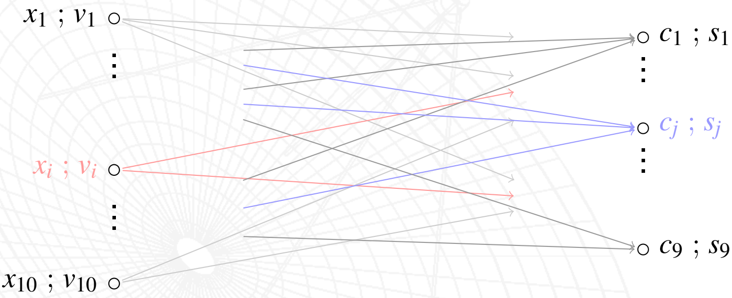
Sum-product algorithm

Algorithm



Sum-product algorithm
Algorithm

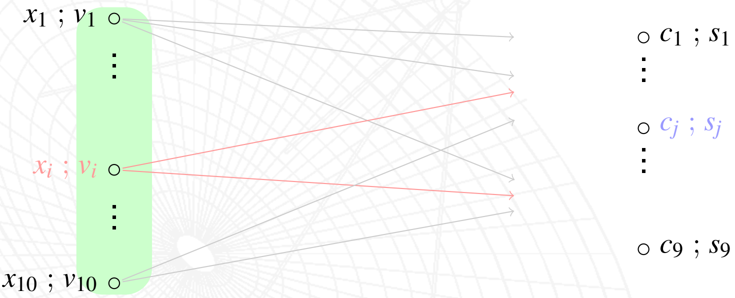
- Tanner graph



Sum-product algorithm

Algorithm

1. Step $m = 1$. Send $\mathrm{LLR}(x_i)_{v_i \to c_j}^{(\text{step } 1)}$ from all v_i



Sum-product algorithm

Algorithm

2. Evaluate $\prod_{v \in A(c_j)} \text{LLR}(x_i)^{(\text{step } m)}_{v \to c_j}$ for all c_j

   2.1 Strictly positive for all c_j. Evaluate:
   $\text{LLR}(x_i) = -\frac{2 \cdot y_i}{\sigma^2} + \sum_{c_k \in A(v_i)} \text{LLR}(x_i)^{(\text{step } m)}_{c_k \to v_i}$. Stop.

x_1 ; v_1 ○

⋮

x_i ; v_i ○

⋮

x_{10} ; v_{10} ○

○ c_1 ; s_1

⋮

○ c_j ; s_j

⋮

○ c_9 ; s_9

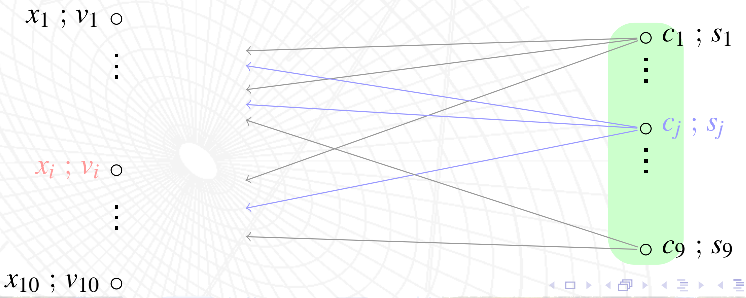
Sum-product algorithm

Algorithm

2.2 Step $m = m + 1$

Send from all c_j , $\mathrm{LLR}(x_i)^{(\text{step } m)}_{c_j \to v_i} =$

$$2 \cdot \operatorname{artanh}\left(\prod_{v_l \in A(c_j) \setminus v_i} \tanh\left(\frac{\mathrm{LLR}(x_l)^{(\text{step } m-1)}_{v_l \to c_j}}{2} \right) \right)$$

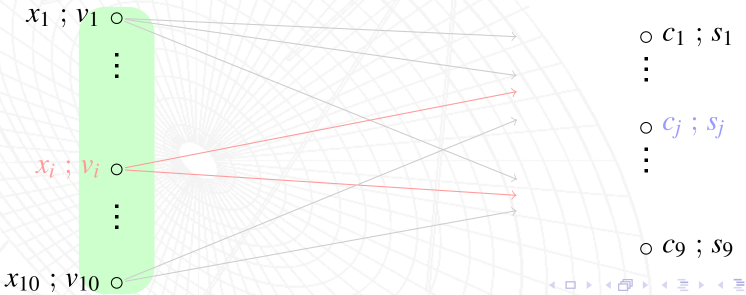


Sum-product algorithm

Algorithm

2.2 Step m

Send from all v_i , $$\text{LLR}(x_i)^{(\text{step } m)}_{v_i \to c_j} = -\frac{2 \cdot y_i}{\sigma^2} + \sum_{c_k \in A(v_i) \backslash c_j} \text{LLR}(x_i)^{(\text{step } m)}_{c_k \to v_i}$$



Sum-product algorithm

Algorithm

- Go to phase 2