# Belief Propagation

Michael Ruderer | 13. Juli 2018 | Uni Ulm

# Inhaltsverzeichnis

# SAT

- SAT formula in CNF

$$\mathcal{F} = (x_1 \vee x_2 \vee \overline{x_3}) \wedge (x_3 \vee x_4)$$

  - Boolean variables $x_1, x_2, \ldots, x_n$
  - Negations $\overline{x_1}, \ldots, \overline{x_n}$
  - Clauses: Disjunction of variables and their negations
  - $\mathcal{F}$: Conjuction of clauses
- Is there an assignmnent of the variables that satisfies $\mathcal{F}$?
- How does the assignment look like?

# Factor Graphs

Factor graphs represent a function's factorization

- Function $f(X)$ over variables $X = \{x_1, x_x, \dots, x_n\}$
- Global function $f$ factorizes to local functions

$$f(X) = \prod_{j=1}^{m} f_j(S_j)$$

- Local functions have smaller input $S_j \subset X$

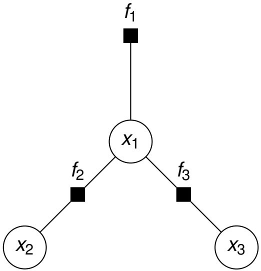
# Factor Graphs

Factor graphs represent a function’s factorization

- Two types of nodes
  - Variable nodes: represent variables
  - Factor nodes : represent local functions
- Edges connect variable and factor nodes
- Factor nodes are connected to all variable nodes of their input variables

# Example

$$\begin{aligned} f(x_1, x_2, x_3) &= x_1^3 - x_1^2 x_2 + x_1^2 x_3 - x_1 x_2 x_3 \\ &= \underbrace{(x_1)}_{f_1(x_1)} * \underbrace{(x_1 - x_2)}_{f_2(x_1,x_2)} * \underbrace{(x_1 + x_3)}_{f_3(x_1,x_2)} \end{aligned}$$

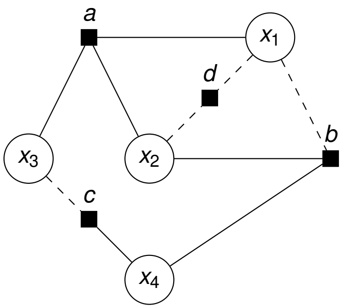


# Factor Graphs

Factor Graph of a CNF formula

$$\mathcal{F} = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee x_4) \wedge (\overline{x_3} \vee x_4) \wedge (\overline{x_1} \vee \overline{x_2})$$

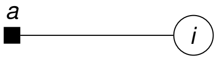
- $\mathcal{F}$ is a product of clauses
- Clauses $\cong$ local functions



# Message Passing

Message Passing Algorithms on factor graphs

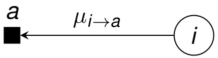
- Nodes communicate through messages
- Messages are passed over the graph's edges
- Two types of messages
  - $\mu_{i \to a}$ sent from factor $a$ to variable $i$
  - $\mu_{a \to i}$ sent from variable $i$ to factor $a$



# Message Passing

Message Passing Algorithms on factor graphs

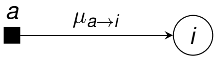
- Nodes communicate through messages
- Messages are passed over the graph’s edges
- Two types of messages
  - $\mu_{i \to a}$ sent from factor $a$ to variable $i$
  - $\mu_{a \to i}$ sent from variable $i$ to factor $a$



# Message Passing

Message Passing Algorithms on factor graphs

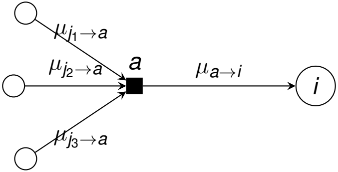
- Nodes communicate through messages
- Messages are passed over the graph's edges
- Two types of messages
  - $\mu_{i \to a}$ sent from factor $a$ to variable $i$
  - $\mu_{a \to i}$ sent from variable $i$ to factor $a$



# Message Passing

Message Passing Algorithms on factor graphs

- Nodes communicate through messages
- Messages are passed over the graph's edges
- Message $\mu_{a\to i}$ determined by incoming messages $\mu_{j\to a}$ from neighbours $j \neq i$

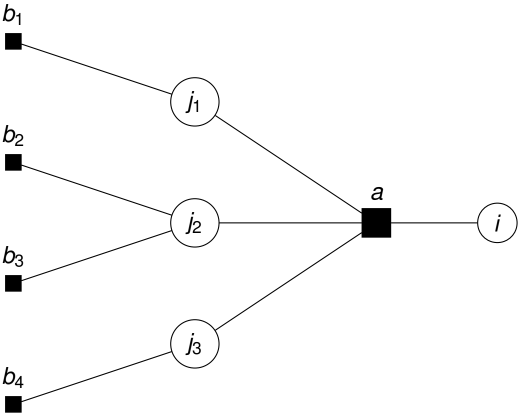


# Message Passing

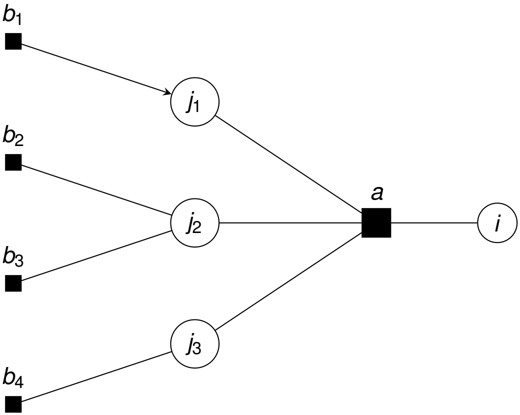
Message Passing Algorithms on trees

- Messages genereated bottom up
- Leaves start sending messages
- Messages are propagated forward in the tree
- Does **not** work on graphs with cycles

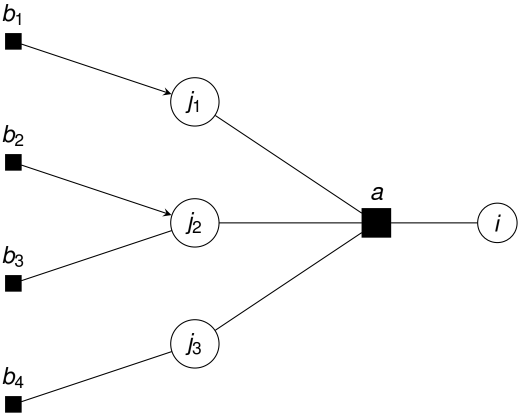
# Example



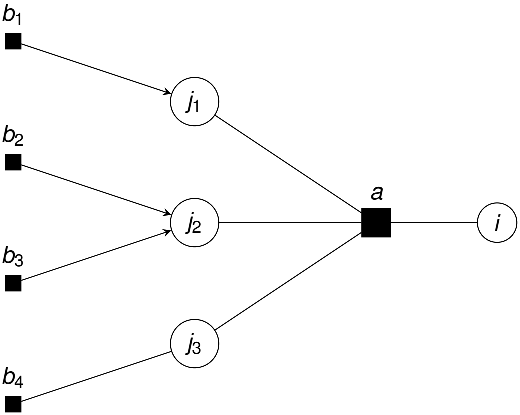
# Example



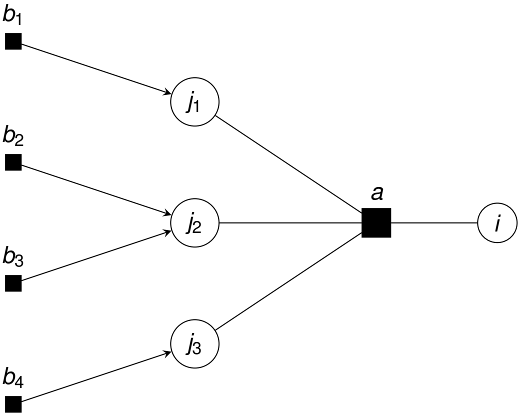
# Example



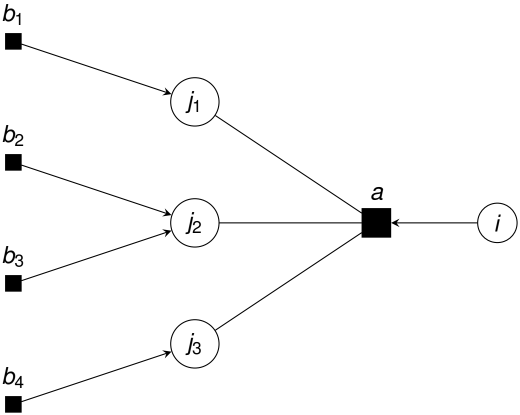
# Example



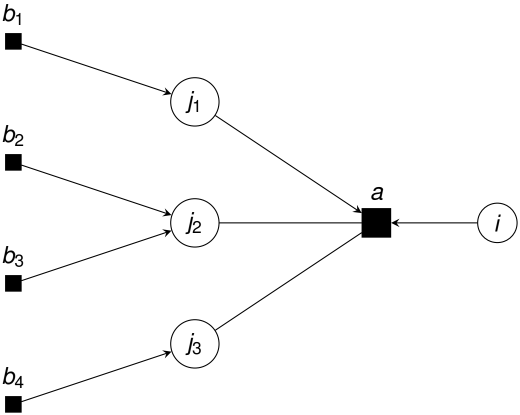
# Example



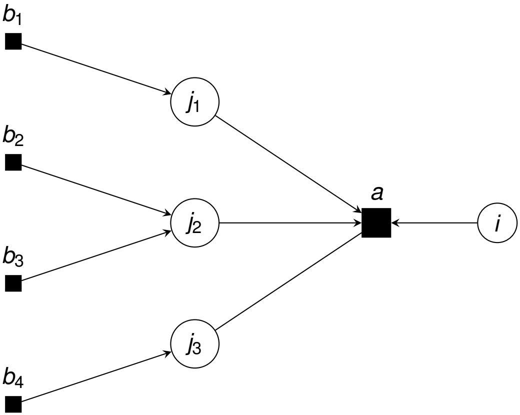
# Example



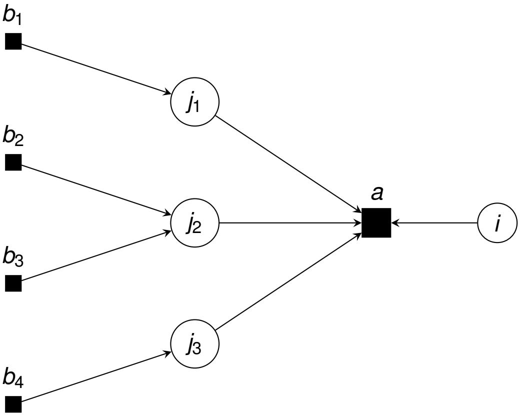
# Example



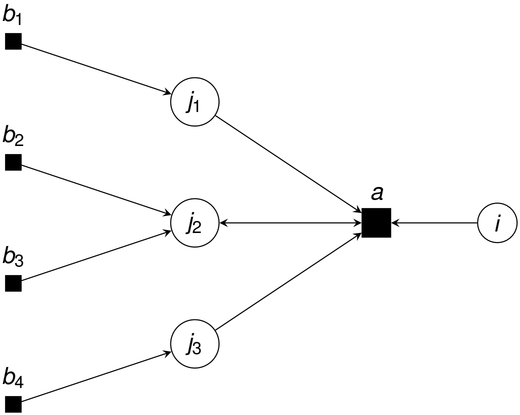
# Example



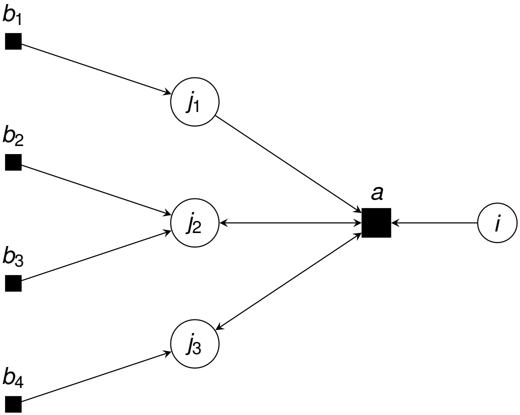
# Example



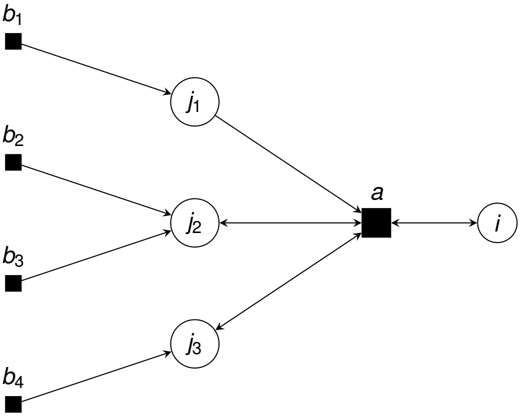
# Example



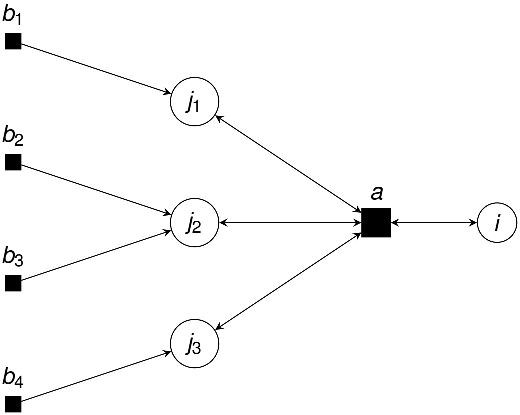
# Example



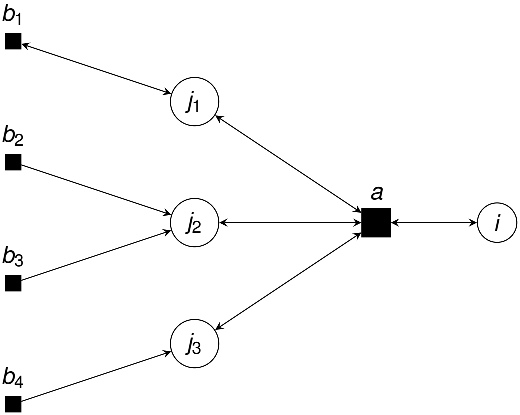
# Example



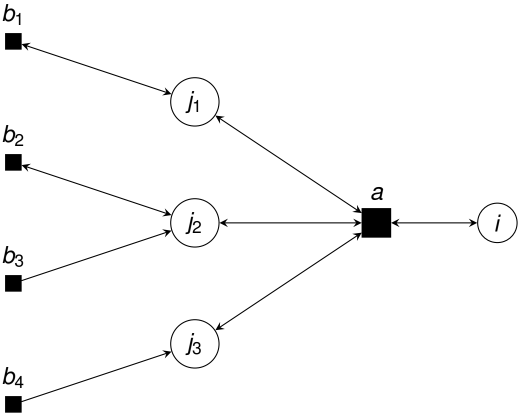
# Example



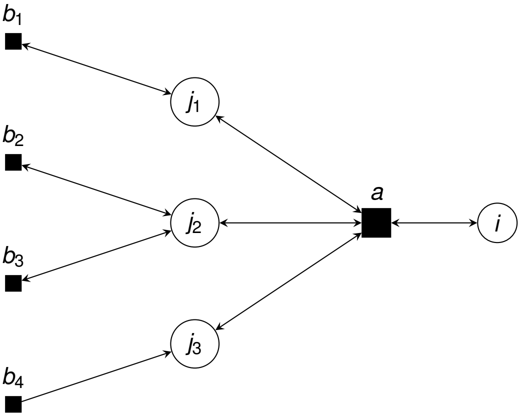
# Example



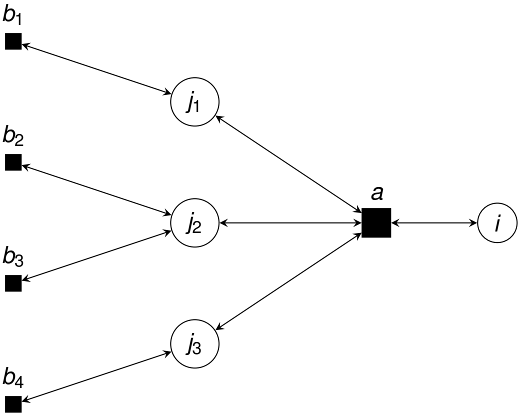
# Example



# Example



# Example



# Message Passing

- In general graphs: *Loopy* Message Passing
- Randomly initialize all messages
- Apply the Update rule until messages have converged
- Scheduling important

# Message Passing

## Generic Message Passing Algorithm

```
1. Randomly initialize all warnings μ_{i→a}, μ_{a→i}
2. For t = 0 to t_max
   2.1 Apply the update rule to all edges
       in random order
   2.2 If no message has changed goto 3
3. If t = t_max return UNGONVERGED
   Else         return the converged messages
```

## Warning Propagation Algorithm

Apply Message Passing to SAT

- Idea: Find out which variables must take a certain value

# Belief Propagation Algorithm

Generalization of Warning Propagation

- Compute the *probability* that a warning is sent