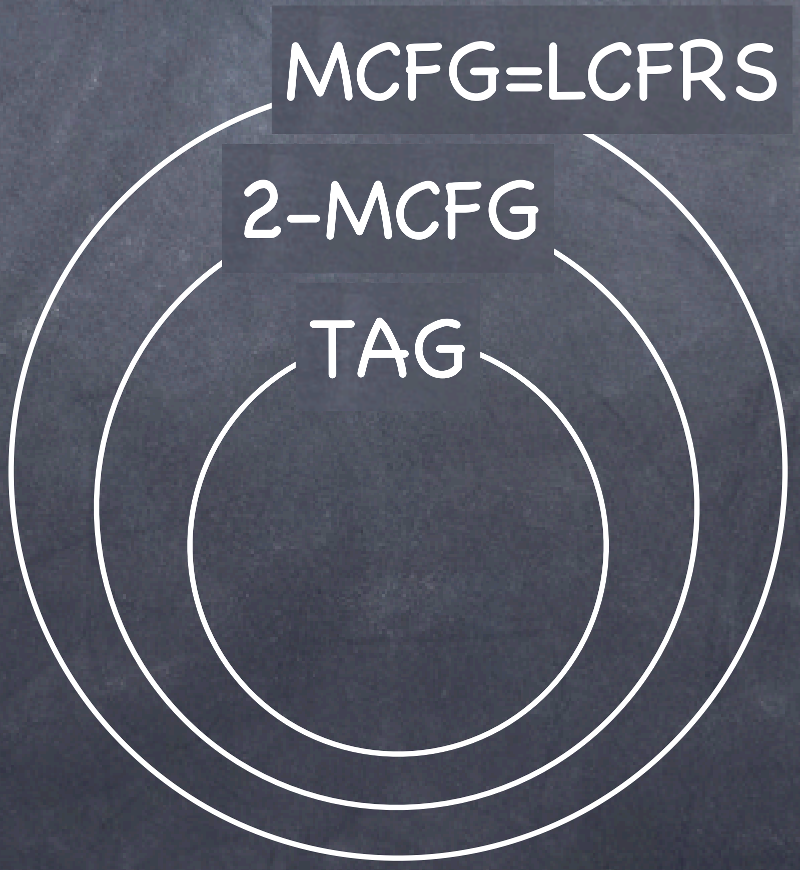
A Prefix-Correct Earley Recognizer for Multiple Context-Free Grammars

Makoto Kanazawa
National Institute of Informatics
Tokyo, Japan

This talk

Earley recognizer for multiple context-free grammars

- based on the Datalog representation of MCFGs (cf. Kanazawa 2007)
- with the correct prefix property
- yields an $O(n^6)$ recognizer for TAGs



This talk

Earley recognizer for multiple context-free grammars

- based on the Datalog representation of MCFGs (cf. Kanazawa 2007)
- with the correct prefix property
- yields an $O(n^6)$ recognizer for TAGs

Previous approaches:

- Matsumura et al. (1989), Harkema (2001), Albro (2002), de la Clergerie (2002a, 2002b)
- Nederhof (1999): $O(n^6)$ prefix-correct Earley-style recognizer for TAGs

My approach

tabular parsing

parsing schema

control algorithm

(grammar → deduction system)

My approach

tabular parsing

parsing schema — control algorithm

(grammar → deduction system)

Datalog program

$\langle A \rightarrow \alpha \bullet \gamma, i, j \rangle$ $\quad$ $[A \rightarrow \alpha \bullet \gamma](i, j)$

My approach

tabular parsing

parsing schema

(grammar → deduction system)

Datalog program

control algorithm

canonical chart parsing
(≈ seminaive bottom-up evaluation)

$\langle A \to \alpha \bullet \gamma, i, j \rangle$

$[A \to \alpha \bullet \gamma](i, j)$

My approach

tabular parsing

parsing schema

(grammar → deduction system)

Datalog program   Datalog program

control algorithm

canonical chart parsing
(≈ seminaive bottom-up evaluation)

$\langle A \to \alpha \bullet \gamma, i, j \rangle$ $[A \to \alpha \bullet \gamma](i, j)$

My approach

tabular parsing

parsing schema

program transformation

(grammar → deduction system)

Datalog program   Datalog program

$\langle A \to \alpha \bullet \gamma, i, j \rangle$

control algorithm

canonical chart parsing

(≈ seminaive bottom-up evaluation)

$[A \to \alpha \bullet \gamma](i, j)$

My approach

Earley parsing

parsing schema

program transformation

(grammar → deduction system)

Datalog program Datalog program

control algorithm

canonical chart parsing

(≈ seminaive bottom-up evaluation)

$\langle A \to \alpha \bullet \gamma, i, j \rangle$

$[A \to \alpha \bullet \gamma](i, j)$

My approach

Earley parsing

parsing schema

magic-sets rewriting

(grammar → deduction system)

Datalog program   Datalog program

control algorithm

canonical chart parsing

(≈ seminaive bottom-up evaluation)

$\langle A \to \alpha \bullet \gamma, i, j \rangle$ $[A \to \alpha \bullet \gamma](i, j)$

My approach

Earley parsing

parsing schema

magic-sets rewriting

(grammar → deduction system)

Datalog program   Datalog program

control algorithm

canonical chart parsing

(≈ seminaive bottom-up evaluation)

- uses well-established, general formalism/method
- avoids ad hoc techniques as much as possible
- easier to understand and easier to prove correct

Earley recognizer for CFGs

a	b	c	d

0 1 2 3 4

An item

- is of the form $\langle A \to \alpha \bullet \beta, i, j \rangle$

  dotted rule — input positions

- encodes the information

$$
\begin{aligned}
S &\Rightarrow^* a_1 \dots a_i A\gamma \\
&\Rightarrow a_1 \dots a_i \alpha\beta\gamma \\
&\Rightarrow^* \underbrace{a_1 \dots a_i a_{i+1} \dots a_j}_{\text{prefix of the input}} \beta\gamma
\end{aligned}
$$

Deduction system for Earley

$$\frac{}{\langle \text{start} \rightarrow \bullet S, 0, 0 \rangle} \text{ INITIALIZE}$$

$$\frac{\langle A \rightarrow \alpha \bullet B\gamma, i, j \rangle}{\langle B \rightarrow \bullet\beta, j, j \rangle} \text{ PREDICT} \qquad (B \rightarrow \beta \in G)$$

$$\frac{\langle A \rightarrow \alpha \bullet B\gamma, i, j \rangle \quad \langle B \rightarrow \beta \bullet, j, k \rangle}{\langle A \rightarrow \alpha B \bullet \gamma, i, k \rangle} \text{ COMPLETE}$$

$$\frac{\langle A \rightarrow \alpha \bullet b\gamma, i, j \rangle}{\langle A \rightarrow \alpha b \bullet \gamma, i, j+1 \rangle} \text{ SCAN} \qquad (b = a_j)$$

$a_1 \ldots a_n \in L(G)$ iff

$\langle \text{start} \rightarrow S \bullet, 0, n \rangle$ is derivable

Deduction system for Earley

$$\frac{}{\langle \text{start} \rightarrow \bullet S, 0, 0\rangle} \text{ INITIALIZE}$$

$$\frac{\langle A \rightarrow \alpha \bullet B\gamma, i, j\rangle}{\langle B \rightarrow \bullet\beta, j, j\rangle} \text{ PREDICT} \qquad (B \rightarrow \beta \in G)$$

$$\frac{\langle A \rightarrow \alpha \bullet B\gamma, i, j\rangle \quad \langle B \rightarrow \beta \bullet, j, k\rangle}{\langle A \rightarrow \alpha B \bullet \gamma, i, k\rangle} \text{ COMPLETE}$$

$$\frac{\langle A \rightarrow \alpha \bullet b\gamma, i, j\rangle}{\langle A \rightarrow \alpha b \bullet \gamma, i, j+1\rangle} \text{ SCAN} \qquad (b = a_j)$$

A SCAN move is made on $a_j \Rightarrow \exists w(a_1 \ldots a_j w \in L(G))$

correct prefix property

Deduction system = Datalog program

$$[B \rightarrow \bullet \beta](j, j) \text{ :- } [A \rightarrow \alpha \bullet B\gamma](i, j).$$

$$[A \rightarrow \alpha B \bullet \gamma](i, k) \text{ :- } [A \rightarrow \alpha \bullet B\gamma](i, j),\ [B \rightarrow \beta \bullet](j, k).$$

$$[A \rightarrow \alpha b \bullet \gamma](i, k) \text{ :- } [A \rightarrow \alpha \bullet b\gamma](i, j),\ b(j, k).$$

$a_1 \ldots a_n \in L(G)$ iff

$$\mathbf{P} \cup \{[\text{start} \rightarrow \bullet S](0, 0), a_1(0, 1), \ldots, a_n(n-1, n)\} \vdash [\text{start} \rightarrow S \bullet](0, n)$$

Chart parsing control algorithm

CHART-PARSE($a_1 \ldots a_n$)

  agenda $\leftarrow [seed, a_1(0,1), \ldots, a_n(n-1,n)]$
  chart $\leftarrow \varnothing$
  while *agenda* is not empty
    do *trigger* $\leftarrow$ POP(*agenda*)
      add *trigger* to *chart*
      new_items $\leftarrow$ IMMEDIATE_CONSEQUENCES(*chart*, *trigger*)

      foreach *item* $\in$ *new_items* $-$(*chart* $\cup$ *agenda*)
        do PUSH(*agenda*, *item*)
  return *chart*

Chart parsing control algorithm

$\textsc{Chart-parse}(a_1 \ldots a_n)$

$\quad agenda \leftarrow [seed, a_1(0,1), \ldots, a_n(n-1, n)]$

$\quad chart \leftarrow \varnothing$

$\quad$ **while** *agenda* is not empty

$\quad\quad$ **do** $trigger \leftarrow \textsc{pop}(agenda)$

$\quad\quad\quad$ add *trigger* to *chart*

$\quad\quad\quad new\_items \leftarrow \textsc{immediate\_consequences}(chart, trigger)$

$\quad\quad\quad$ **if** $trigger = a_j(j-1, j)$ and *new_items* is empty

$\quad\quad\quad\quad$ **then reject**

$\quad\quad\quad$ **foreach** $item \in new\_items - (chart \cup agenda)$

$\quad\quad\quad\quad$ **do** $\textsc{push}(agenda, item)$

$\quad$ **return** *chart*

CFGs as Datalog programs

$$S \rightarrow aSd \qquad S(i,l) \coloneq a(i,j), S(j,k), d(k,l).$$
$$S \rightarrow aTd \qquad S(i,l) \coloneq a(i,j), T(j,k), d(k,l).$$
$$T \rightarrow bTc \qquad T(i,l) \coloneq b(i,j), T(j,k), c(k,l).$$
$$T \rightarrow bc \qquad T(i,k) \coloneq b(i,j), c(j,k).$$

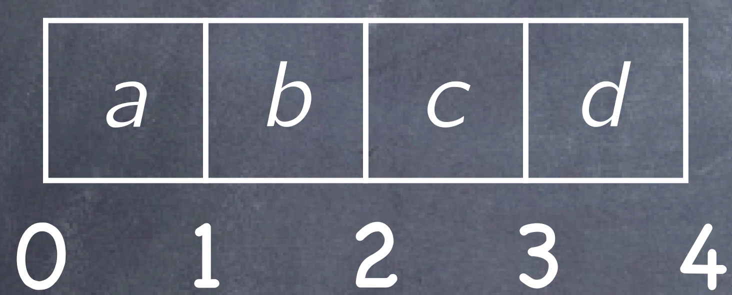
a	b	c	d

0 1 2 3 4

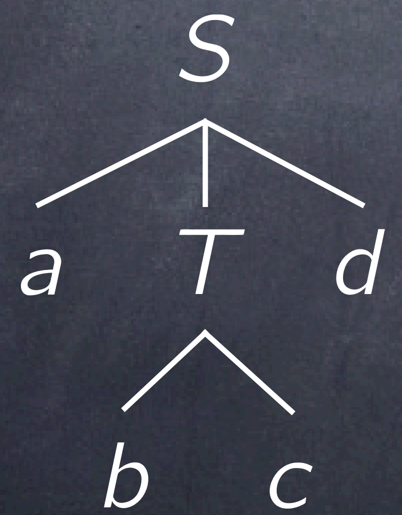
$a(0,1)$. $b(1,2)$. $c(2,3)$. $d(3,4)$.

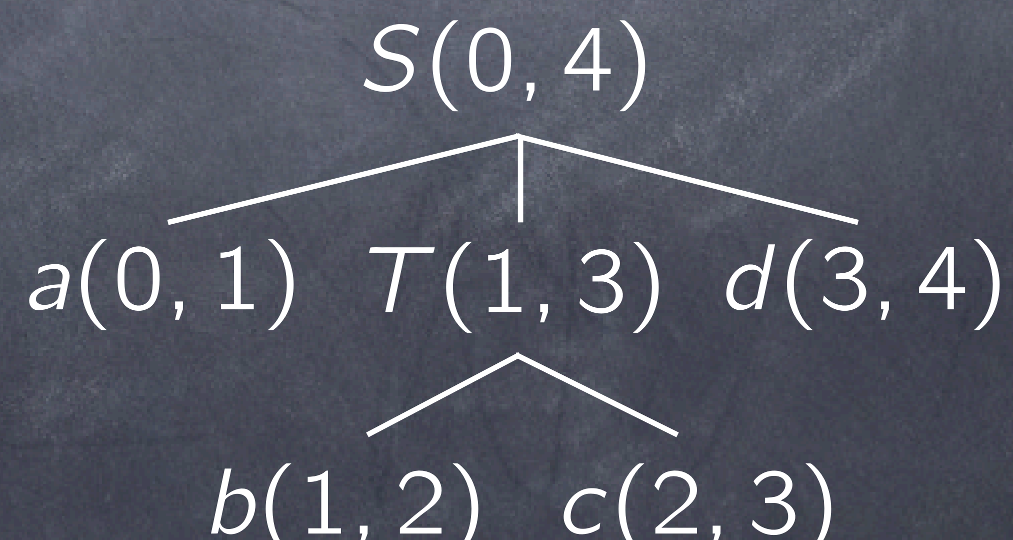
CFGs as Datalog programs

$$S \to aSd \qquad S(i,l) :- a(i,j), S(j,k), d(k,l).$$
$$S \to aTd \qquad S(i,l) :- a(i,j), T(j,k), d(k,l).$$
$$T \to bTc \qquad T(i,l) :- b(i,j), T(j,k), c(k,l).$$
$$T \to bc \qquad T(i,k) :- b(i,j), c(j,k).$$



$a(0,1). \quad b(1,2). \quad c(2,3). \quad d(3,4).$





Magic sets

Magic-sets rewriting

- equivalent to Earley deduction (OLDT resolution)
- introduces top-down prediction into bottom-up evaluation m_R magic predicates
- binarizes the rules $sup_{r,n}$ supplementary predicates
- the result essentially coincides with the Earley deduction system when applied to a Datalog program expressing a CFG

Magic sets

Magic-sets rewriting

- equivalent to Earley deduction (OLDT resolution)
- introduces top-down prediction into bottom-up evaluation m_R magic predicates
- binarizes the rules $sup_{r,n}$ supplementary predicates
- the result essentially coincides with the Earley deduction system when applied to a Datalog program expressing a CFG

$[A \rightarrow \bullet \alpha](i, i)$	$m\_A(i)$	magic predicates
$[A \rightarrow \alpha \bullet \beta](i, j)$	$sup_{r.n}(i, j)$	supplementary predicates $r =$ rule number, $n = \|\alpha\|$
$[A \rightarrow \alpha \bullet](i, j)$	$A(i, j)$	

Example

$1: S(i,l) :- a(i,j), S(j,k), d(k,l).$
$2: S(i,l) :- a(i,j), T(j,k), d(k,l).$
$3: T(i,l) :- b(i,j), T(j,k), c(k,l).$
$4: T(i,k) :- b(i,j), c(j,k).$

INITIALIZE

$m\_S(0).$

PREDICT

$m\_S(j) :- sup_{1.1}(i,j).$
$m\_T(j) :- sup_{2.1}(i,j).$
$m\_T(j) :- sup_{3.1}(i,j).$

SCAN/COMPLETE

$sup_{1.1}(i,j) :- m\_S(i), a(i,j).$
$sup_{1.2}(i,k) :- sup_{1.1}(i,j), S(j,k).$
$S(i,l) :- sup_{1.2}(i,k), d(k,l).$
$sup_{2.1}(i,j) :- m\_S(i), a(i,j).$
$sup_{2.2}(i,k) :- sup_{2.1}(i,j), T(j,k).$
$S(i,l) :- sup_{2.2}(i,k), d(k,l).$
$sup_{3.1}(i,j) :- m\_T(i), b(i,j).$
$sup_{3.2}(i,k) :- sup_{3.1}(i,j), T(j,k).$
$T(i,l) :- sup_{3.2}(i,k), c(k,l).$
$sup_{4.1}(i,j) :- m\_T(i), b(i,j).$
$T(i,k) :- sup_{4.1}(i,j), c(j,k).$

Example

1: $S(i,l) :- a(i,j), S(j,k), d(k,l).$
2: $S(i,l) :- a(i,j), T(j,k), d(k,l).$
3: $T(i,l) :- b(i,j), T(j,k), c(k,l).$
4: $T(i,k) :- b(i,j), c(j,k).$

INITIALIZE

$m\_S(0).$

PREDICT

$m\_S(j) :- sup_{1.1}(i,j).$
$m\_T(j) :- sup_{2.1}(i,j).$
$m\_T(j) :- sup_{3.1}(i,j).$

$sup_{1.1}(i,j) :- m\_S(i), a(i,j).$
$sup_{1.2}(i,k) :- sup_{1.1}(i,j), S(j,k).$
$S(i,l) :- sup_{1.2}(i,k), d(k,l).$
$sup_{2.1}(i,j) :- m\_S(i), a(i,j).$
$sup_{2.2}(i,k) :- sup_{2.1}(i,j), T(j,k).$
$S(i,l) :- sup_{2.2}(i,k), d(k,l).$
$sup_{3.1}(i,j) :- m\_T(i), b(i,j).$
$sup_{3.2}(i,k) :- sup_{3.1}(i,j), T(j,k).$
$T(i,l) :- sup_{3.2}(i,k), c(k,l).$
$sup_{4.1}(i,j) :- m\_T(i), b(i,j).$
$T(i,k) :- sup_{4.1}(i,j), c(j,k).$

SCAN/COMPLETE

$[T \rightarrow bT \bullet c](i,k) :-$
$\quad [T \rightarrow b \bullet Tc](i,j), [T \rightarrow bTc \bullet](j,k)$
$[T \rightarrow bT \bullet c](i,k) :-$
$\quad [T \rightarrow b \bullet Tc](i,j), [T \rightarrow bc \bullet](j,k)$

Example

$1: S(i,l) :- a(i,j), S(j,k), d(k,l).$
$2: S(i,l) :- a(i,j), T(j,k), d(k,l).$
$3: T(i,l) :- b(i,j), T(j,k), c(k,l).$
$4: T(i,k) :- b(i,j), c(j,k).$

$[T \to \bullet bTc](j,j) :- [T \to b \bullet Tc](i,j)$
$[T \to \bullet bc](j,j) :- [T \to b \bullet Tc](i,j)$

SCAN/COMPLETE

$[T \to bT \bullet c](i,k) :-$
$\quad [T \to b \bullet Tc](i,j), [T \to bTc \bullet](j,k)$
$[T \to bT \bullet c](i,k) :-$
$\quad [T \to b \bullet Tc](i,j), [T \to bc \bullet](j,k)$

$m\_S(0).$ INITIALIZE

$m\_S(j) :- sup_{1.1}(i,j).$
$m\_T(j) :- sup_{2.1}(i,j).$ PREDICT
$m\_T(j) :- sup_{3.1}(i,j).$

$sup_{1.1}(i,j) :- m\_S(i), a(i,j).$
$sup_{1.2}(i,k) :- sup_{1.1}(i,j), S(j,k).$
$S(i,l) :- sup_{1.2}(i,k), d(k,l).$
$sup_{2.1}(i,j) :- m\_S(i), a(i,j).$
$sup_{2.2}(i,k) :- sup_{2.1}(i,j), T(j,k).$
$S(i,l) :- sup_{2.2}(i,k), d(k,l).$
$sup_{3.1}(i,j) :- m\_T(i), b(i,j).$
$sup_{3.2}(i,k) :- sup_{3.1}(i,j), T(j,k).$
$T(i,l) :- sup_{3.2}(i,k), c(k,l).$
$sup_{4.1}(i,j) :- m\_T(i), b(i,j).$
$T(i,k) :- sup_{4.1}(i,j), c(j,k).$

Multiple context-free grammar

$S(x_1 y_1 x_2 y_2) \text{ :- } P(x_1, x_2), Q(y_1, y_2).$

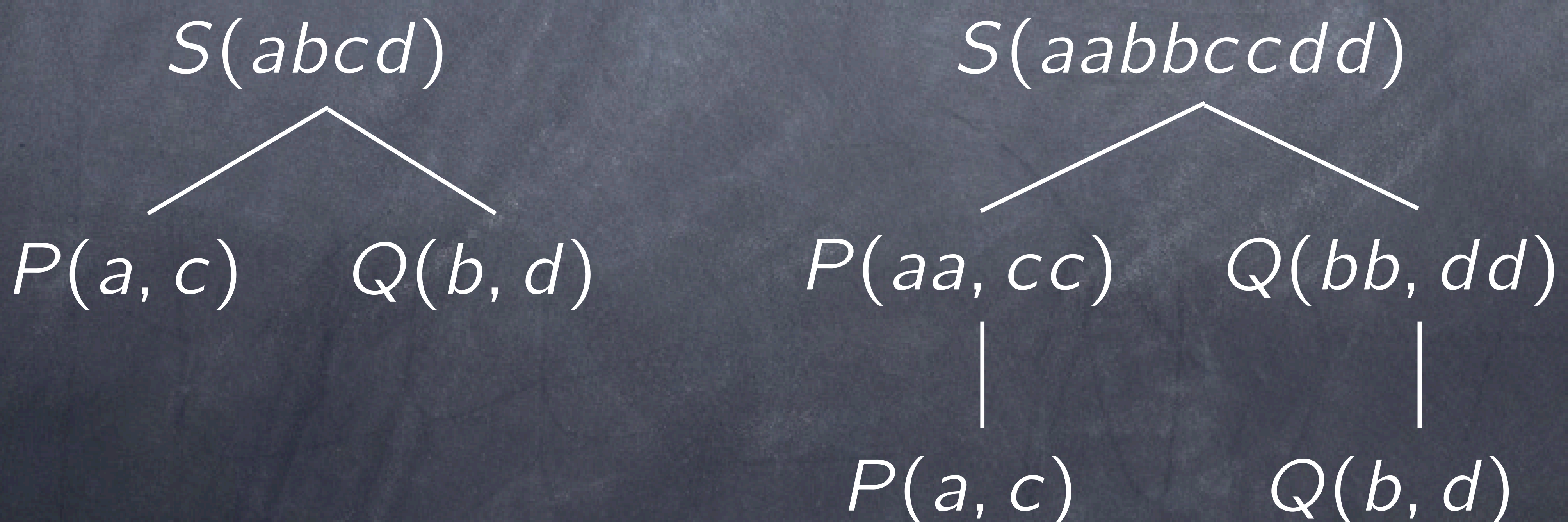
$P(a x_1, c x_2) \text{ :- } P(x_1, x_2).$

$P(a, c).$

$Q(b y_1, d y_2) \text{ :- } Q(y_1, y_2).$

$Q(b, d).$

$\{\, a^m b^n c^m d^n \mid m, n \geq 1 \,\}$



MCFGs as Datalog programs

$S(\ x_1\ y_1\ x_2\ y_2\ ) :- P(x_1, x_2), Q(y_1, y_2).$

$i\ \ j\ \ k\ \ l\ \ m$

$P(\ a\, x_1\ ,\ c\, x_2\ ) :- P(x_1, x_2).$

$i\ j\ \ k\ l\ m\ \ n$

$P(\ a\ ,\ c\ ).$

$i\ j\ k\ l$

$Q(by_1, dy_2) :- Q(y_1, y_2).$

$Q(b, d).$

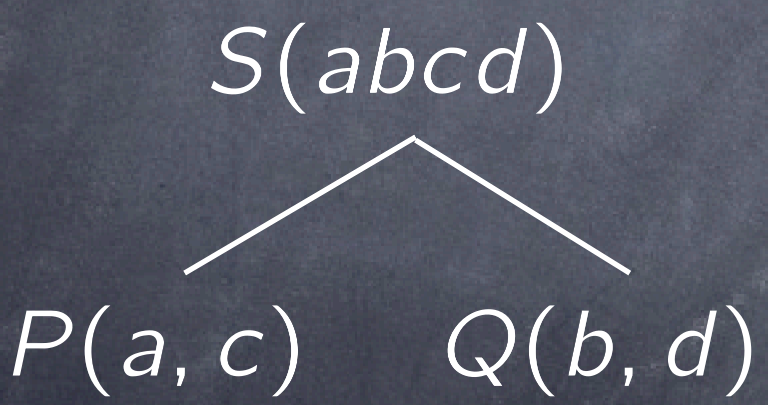
$S(i, m) :- P(i, j, k, l), Q(j, k, l, m).$

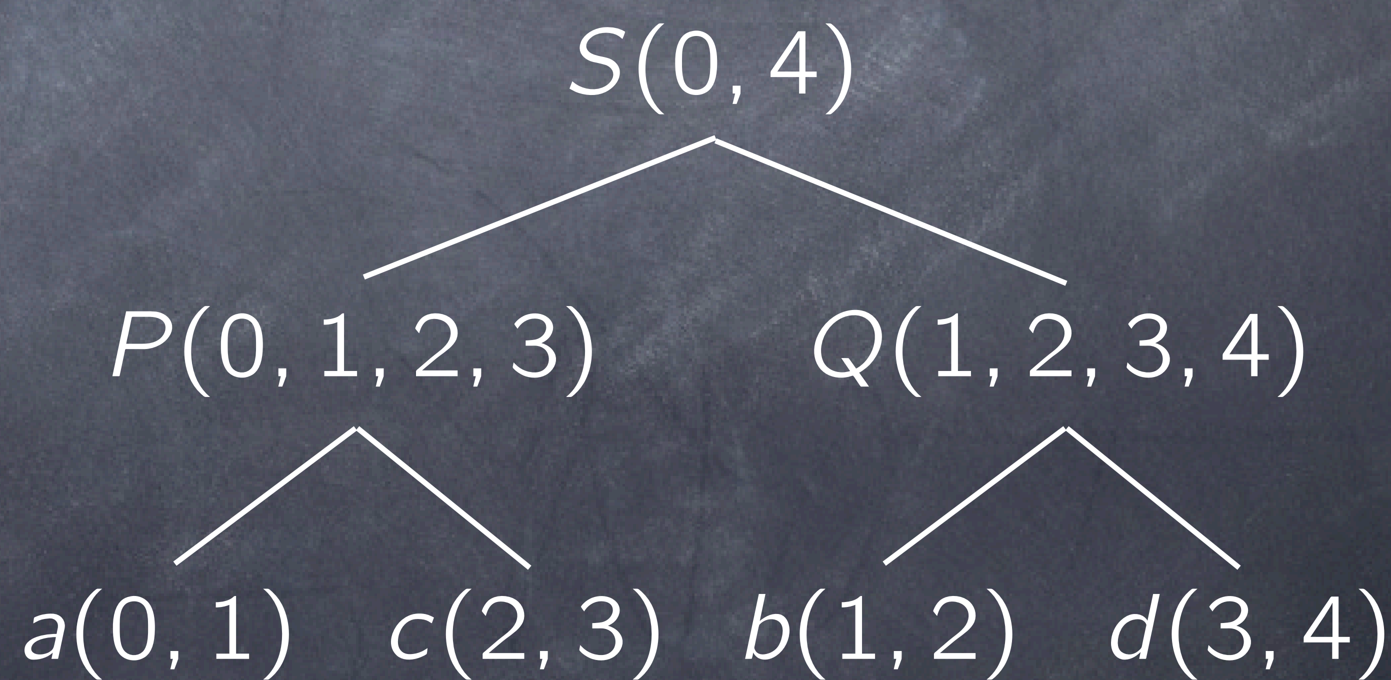
$P(i, k, l, n) :- a(i, j), P(j, k, m, n), c(l, m).$

$P(i, j, k, l) :- a(i, j), c(k, l).$

$Q(i, k, l, n) :- b(i, j), d(l, m), Q(j, k, m, n).$

$Q(i, j, k, l) :- b(i, j), d(k, l).$





Magic-sets rewriting

1: $S^{bf}(i, m) :- P^{bfff}(i, j, k, l), Q^{bbbf}(j, k, l, m).$

2: $P^{bfff}(i, k, l, n) :- a^{bf}(i, j), P^{bfff}(j, k, m, n), c^{fb}(l, m).$

3: $P^{bfff}(i, j, k, l) :- a^{bf}(i, j), c^{ff}(k, l).$

4: $Q^{bbbf}(i, k, l, n) :- b^{bf}(i, j), d^{bf}(l, m), Q^{bbbf}(j, k, m, n).$

5: $Q^{bbbf}(i, j, k, l) :- b^{bb}(i, j), d^{bf}(k, l).$

adornment

Magic-sets rewriting

$1: S^{bf}(i, m) :- P^{bfff}(i, j, k, l), Q^{bbbf}(j, k, l, m).$

$2: P^{bfff}(i, k, l, n) :- a^{bf}(i, j), P^{bfff}(j, k, m, n), c^{fb}(l, m).$

$3: P^{bfff}(i, j, k, l) :- a^{bf}(i, j), c^{ff}(k, l).$

$4: Q^{bbbf}(i, k, l, n) :- b^{bf}(i, j), d^{bf}(l, m), Q^{bbbf}(j, k, m, n).$

$5: Q^{bbbf}(i, j, k, l) :- b^{bb}(i, j), d^{bf}(k, l).$

adornment

$$?- S(0, x).$$

$$\stackrel{1}{\Longrightarrow} ?- P(0, j_1, k_1, l_1), Q(j_1, k_1, l_1, x).$$

$$\stackrel{3}{\Longrightarrow} ?- a(0, j_1), c(k_1, l_1), Q(j_1, k_1, l_1, x).$$

$$\stackrel{a(0,1)}{\Longrightarrow} ?- c(k_1, l_1), Q(1, k_1, l_1, x).$$

$$\stackrel{c(2,3)}{\Longrightarrow} ?- Q(1, 2, 3, x).$$

$$\stackrel{5}{\Longrightarrow} ?- b(1, 2), d(3, x).$$

$$\stackrel{b(1,2)}{\Longrightarrow} ?- d(3, x).$$

$$\stackrel{d(3,4)}{\Longrightarrow} ?- [].$$

SLD derivation

Magic-sets rewriting

1: $S^{bf}(i,m) :- P^{bfff}(i,j,k,l), Q^{bbbf}(j,k,l,m).$

2: $P^{bfff}(i,k,l,n) :- a^{bf}(i,j), P^{bfff}(j,k,m,n), c^{fb}(l,m).$

3: $P^{bfff}(i,j,k,l) :- a^{bf}(i,j), c^{ff}(k,l).$

4: $Q^{bbbf}(i,k,l,n) :- b^{bf}(i,j), d^{bf}(l,m), Q^{bbbf}(j,k,m,n).$

5: $Q^{bbbf}(i,j,k,l) :- b^{bb}(i,j), d^{bf}(k,l).$

adornment

$?- S(0,x).$ $m\_S(0)$

SLD derivation

$\overset{1}{\Longrightarrow} ?- P(0,j_1,k_1,l_1), Q(j_1,k_1,l_1,x).$ $m\_P(0)$

$\overset{3}{\Longrightarrow} ?- a(0,j_1), c(k_1,l_1), Q(j_1,k_1,l_1,x).$

$\overset{a(0,1)}{\Longrightarrow} ?- c(k_1,l_1), Q(1,k_1,l_1,x).$

$\overset{c(2,3)}{\Longrightarrow} ?- Q(1,2,3,x).$ $m\_Q(1,2,3)$

$\overset{5}{\Longrightarrow} ?- b(1,2), d(3,x).$

magic predicates

$\overset{b(1,2)}{\Longrightarrow} ?- d(3,x).$

$\overset{d(3,4)}{\Longrightarrow} ?- [].$

Magic-sets rewriting

$$1: S^{bf}(i,m) :- P^{bfff}(i,j,k,l), Q^{bbbf}(j,k,l,m).$$
$$2: P^{bfff}(i,k,l,n) :- a^{bf}(i,j), P^{bfff}(j,k,m,n), c^{fb}(l,m).$$
$$3: P^{bfff}(i,j,k,l) :- a^{bf}(i,j), c^{ff}(k,l).$$
$$4: Q^{bbbf}(i,k,l,n) :- b^{bf}(i,j), d^{bf}(l,m), Q^{bbbf}(j,k,m,n).$$
$$5: Q^{bbbf}(i,j,k,l) :- b^{bb}(i,j), d^{bf}(k,l).$$

$$r_1: m\_P(i) :- m\_S(i).$$
$$r_2: m\_Q(j,k,l) :- sup_{1.1}(i,j,k,l).$$
$$r_3: m\_P(j) :- sup_{2.1}(i,j).$$
$$r_4: m\_Q(j,k,m) :- sup_{4.2}(i,j,k,l,m).$$

$$r_5: sup_{1.1}(i,j,k,l) :- m\_S(i), P(i,j,k,l).$$
$$r_6: S(i,m) :- sup_{1.1}(i,j,k,l), Q(j,k,l,m).$$
$$r_7: sup_{2.1}(i,j) :- m\_P(i), a(i,j).$$
$$r_8: sup_{2.2}(i,k,m,n) :- sup_{2.1}(i,j), P(j,k,m,n).$$
$$r_9: P(i,k,l,n) :- sup_{2.2}(i,k,m,n), c(l,m).$$
$$r_{10}: sup_{3.1}(i,j) :- m\_P(i), a(i,j).$$
$$r_{11}: P(i,j,k,l) :- sup_{3.1}(i,j), c(k,l).$$
$$r_{12}: sup_{4.1}(i,j,k,l) :- m\_Q(i,k,l), b(i,j).$$

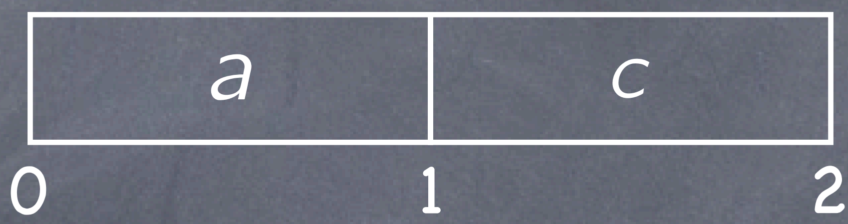
Magic-sets rewriting

$r_1: m\_P(i) :- m\_S(i).$
$r_2: m\_Q(j,k,l) :- sup_{1.1}(i,j,k,l).$
$r_3: m\_P(j) :- sup_{2.1}(i,j).$
$r_4: m\_Q(j,k,m) :- sup_{4.2}(i,j,k,l,m).$

$r_5: sup_{1.1}(i,j,k,l) :- m\_S(i), P(i,j,k,l).$
$r_6: S(i,m) :- sup_{1.1}(i,j,k,l), Q(j,k,l,m).$
$r_7: sup_{2.1}(i,j) :- m\_P(i), a(i,j).$
$r_8: sup_{2.2}(i,k,m,n) :- sup_{2.1}(i,j), P(j,k,m,n).$
$r_9: P(i,k,l,n) :- sup_{2.2}(i,k,m,n), c(l,m).$
$r_{10}: sup_{3.1}(i,j) :- m\_P(i), a(i,j).$
$r_{11}: P(i,j,k,l) :- sup_{3.1}(i,j), c(k,l).$
$r_{12}: sup_{4.1}(i,j,k,l) :- m\_Q(i,k,l), b(i,j).$
$r_{13}: sup_{4.2}(i,j,k,l,m) :- sup_{4.1}(i,j,k,l), d(l,m).$
$r_{14}: Q(i,k,l,n) :- sup_{4.2}(i,j,k,l,m), Q(j,k,m,n).$
$r_{15}: sup_{5.1}(i,j,k) :- m\_Q(i,j,k), b(i,j).$
$r_{16}: Q(i,j,k,l) :- sup_{5.1}(i,j,k), d(k,l).$

Magic-sets rewriting

$$r_1: m\_P(i) :- m\_S(i).$$
$$r_2: m\_Q(j,k,l) :- sup_{1.1}(i,j,k,l).$$
$$r_3: m\_P(j) :- sup_{2.1}(i,j).$$
$$r_4: m\_Q(j,k,m) :- sup_{4.2}(i,j,k,l,m).$$
$$r_5: sup_{1.1}(i,j,k,l) :- m\_S(i), P(i,j,k,l).$$
$$r_6: S(i,m) :- sup_{1.1}(i,j,k,l), Q(j,k,l,m).$$
$$r_7: sup_{2.1}(i,j) :- m\_P(i), a(i,j).$$
$$r_8: sup_{2.2}(i,k,m,n) :- sup_{2.1}(i,j), P(j,k,m,n).$$
$$r_9: P(i,k,l,n) :- sup_{2.2}(i,k,m,n), c(l,m).$$
$$r_{10}: sup_{3.1}(i,j) :- m\_P(i), a(i,j).$$
$$r_{11}: P(i,j,k,l) :- sup_{3.1}(i,j), c(k,l).$$
$$r_{12}: sup_{4.1}(i,j,k,l) :- m\_Q(i,k,l), b(i,j).$$
$$r_{13}: sup_{4.2}(i,j,k,l,m) :- sup_{4.1}(i,j,k,l), d(l,m).$$
$$r_{14}: Q(i,k,l,n) :- sup_{4.2}(i,j,k,l,m), Q(j,k,m,n).$$
$$r_{15}: sup_{5.1}(i,j,k) :- m\_Q(i,j,k), b(i,j).$$
$$r_{16}: Q(i,j,k,l) :- sup_{5.1}(i,j,k), d(k,l).$$



$m\_S(0)$	$sup_{2.1}(0,1)$	$P(0,1,1,2)$
$m\_P(0)$	$sup_{3.1}(0,1)$	$sup_{1.1}(0,1,1,2)$
	$m\_P(1)$	$m\_Q(1,1,2)$

Magic-sets rewriting

$r_1: m\_P(i) :- m\_S(i).$

$r_2: m\_Q(j,k,l) :- sup_{1.1}(i,j,k,l).$

$r_3: m\_P(j) :- sup_{2.1}(i,j).$

$r_4: m\_Q(j,k,m) :- sup_{4.2}(i,j,k,l,m).$

$r_5: sup_{1.1}(i,j,k,l) :- m\_S(i), P(i,j,k,l).$

$r_6: S(i,m) :- sup_{1.1}(i,j,k,l), Q(j,k,l,m).$

$r_7: sup_{2.1}(i,j) :- m\_P(i), a(i,j).$

$r_8: sup_{2.2}(i,k,m,n) :- sup_{2.1}(i,j), P(j,k,m,n).$

$r_9: P(i,k,l,n) :- sup_{2.2}(i,k,m,n), c(l,m).$

$r_{10}: sup_{3.1}(i,j) :- m\_P(i), a(i,j).$

$r_{11}: P(i,j,k,l) :- sup_{3.1}(i,j), c(k,l).$

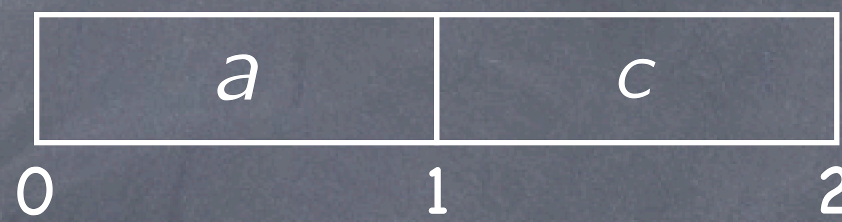
$r_{12}: sup_{4.1}(i,j,k,l) :- m\_Q(i,k,l), b(i,j).$

$r_{13}: sup_{4.2}(i,j,k,l,m) :- sup_{4.1}(i,j,k,l), d(l,m).$

$r_{14}: Q(i,k,l,n) :- sup_{4.2}(i,j,k,l,m), Q(j,k,m,n).$

$r_{15}: sup_{5.1}(i,j,k) :- m\_Q(i,j,k), b(i,j).$

$r_{16}: Q(i,j,k,l) :- sup_{5.1}(i,j,k), d(k,l).$



$m\_S(0)$	$sup_{2.1}(0,1)$	$P(0,1,1,2)$
$m\_P(0)$	$sup_{3.1}(0,1)$	$sup_{1.1}(0,1,1,2)$
	$m\_P(1)$	$m\_Q(1,1,2)$

not prefix-correct!

Magic-sets rewriting

$r_1 : m\_P(i) :- m\_S(i).$

$r_2 : m\_Q(j, k, l) :- sup_{1.1}(i, j, k, l).$

$r_3 : m\_P(j) :- sup_{2.1}(i, j).$

$r_4 : m\_Q(j, k, m) :- sup_{4.2}(i, j, k, l, m).$

$r_5 : sup_{1.1}(i, j, k, l) :- m\_S(i), P(i, j, k, l).$

$r_6 : S(i, m) :- sup_{1.1}(i, j, k, l), Q(j, k, l, m).$

$r_7 : sup_{2.1}(i, j) :- m\_P(i), a(i, j).$

$r_8 : sup_{2.2}(i, k, m, n) :- sup_{2.1}(i, j), P(j, k, m, n).$

$r_9 : P(i, k, l, n) :- sup_{2.2}(i, k, m, n), c(l, m).$

$r_{10} : sup_{3.1}(i, j) :- m\_P(i), a(i, j).$

$r_{11} : P(i, j, k, l) :- sup_{3.1}(i, j), c(k, l).$

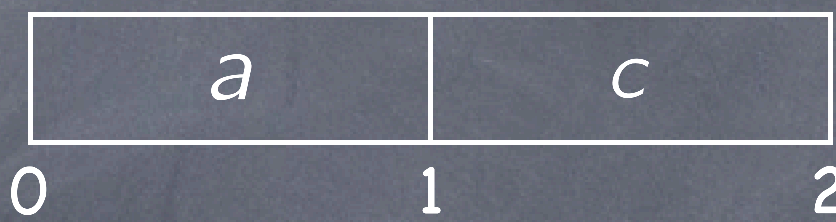
$r_{12} : sup_{4.1}(i, j, k, l) :- m\_Q(i, k, l), b(i, j).$

$r_{13} : sup_{4.2}(i, j, k, l, m) :- sup_{4.1}(i, j, k, l), d(l, m).$

$r_{14} : Q(i, k, l, n) :- sup_{4.2}(i, j, k, l, m), Q(j, k, m, n).$

$r_{15} : sup_{5.1}(i, j, k) :- m\_Q(i, j, k), b(i, j).$

$r_{16} : Q(i, j, k, l) :- sup_{5.1}(i, j, k), d(k, l).$



0	1	2
$m\_S(0)$	$sup_{2.1}(0, 1)$	$P(0, 1, 1, 2)$
$m\_P(0)$	$sup_{3.1}(0, 1)$	$sup_{1.1}(0, 1, 1, 2)$
	$m\_P(1)$	$m\_Q(1, 1, 2)$

not prefix-correct!

Cause of non-prefix-correctness

SLD derivation

$$?-\ S(0, x).$$

$$\overset{1}{\Longrightarrow}\ ?-\ P(0, j_1, k_1, l_1), Q(j_1, k_1, l_1, x).$$

$$\overset{3}{\Longrightarrow}\ ?-\ a(0, j_1), c(k_1, l_1), Q(j_1, k_1, l_1, x).$$

$$\overset{a(0,1)}{\Longrightarrow}\ ?-\ c(k_1, l_1), Q(1, k_1, l_1, x).$$

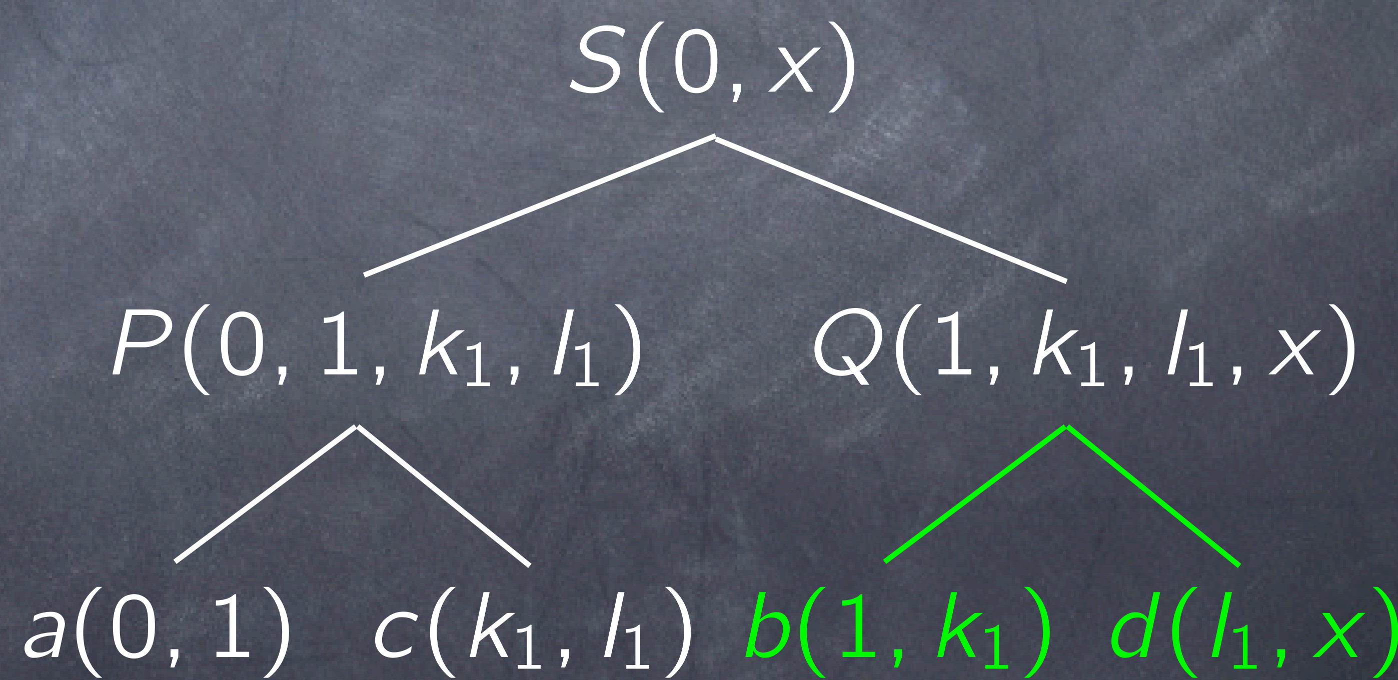
$$\overset{c(1,2)}{\Longrightarrow}\ ?-\ Q(1, 1, 2, x).$$

$$\overset{5}{\Longrightarrow}\ ?-\ b(1, 1), d(2, x).$$

Cause of non-prefix-correctness

SLD derivation

$$
\begin{aligned}
&?-\ S(0,x).\\
\overset{1}{\Longrightarrow}\ &?-\ P(0,j_1,k_1,l_1), Q(j_1,k_1,l_1,x).\\
\overset{3}{\Longrightarrow}\ &?-\ a(0,j_1), c(k_1,l_1), Q(j_1,k_1,l_1,x).\\
\overset{a(0,1)}{\Longrightarrow}\ &?-\ c(k_1,l_1), Q(1,k_1,l_1,x).\\
\overset{c(1,2)}{\Longrightarrow}\ &?-\ Q(1,1,2,x).\\
\overset{5}{\Longrightarrow}\ &?-\ b(1,1), d(2,x).
\end{aligned}
$$



incomplete derivation tree

Cause of non-prefix-correctness

SLD derivation

$$?\!-\ S(0, x).$$

$$\stackrel{1}{\Longrightarrow}\ ?\!-\ P(0, j_1, k_1, l_1), Q(j_1, k_1, l_1, x).$$

$$\stackrel{3}{\Longrightarrow}\ ?\!-\ a(0, j_1), c(k_1, l_1), Q(j_1, k_1, l_1, x).$$

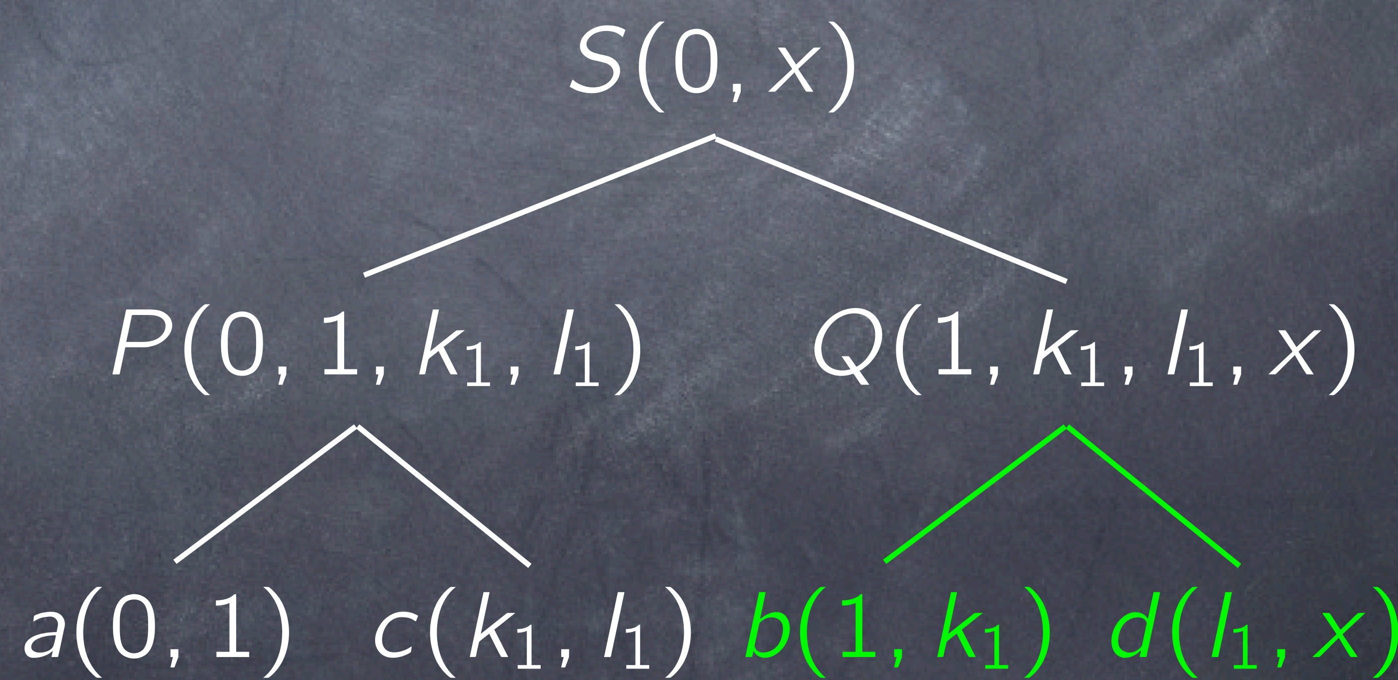
$$\stackrel{a(0,1)}{\Longrightarrow}\ ?\!-\ c(k_1, l_1), Q(1, k_1, l_1, x).$$

$$\stackrel{c(1,2)}{\Longrightarrow}\ ?\!-\ Q(1, 1, 2, x).$$

$$\stackrel{5}{\Longrightarrow}\ ?\!-\ b(1, 1), d(2, x).$$

adornment

$$3\colon\ P^{bfff}(i, j, k, l)\ :\!-\ a^{bf}(i, j), c^{ff}(k, l).$$



incomplete derivation tree

Securing prefix-correctness

$$S(\ x_1\ y_1\ x_2\ y_2\ ) :- P(x_1, x_2), Q(y_1, y_2).$$
$$\quad i \quad j \quad k \quad l \quad m$$

$$S(i, m) :- P(i, j, k, l), Q(j, k, l, m).$$

Securing prefix-correctness

$S(\ x_1\ y_1\ x_2\ y_2\ ) :-\ P(x_1, x_2), Q(y_1, y_2).$
$\quad i\ \ j\ \ k\ \ l\ \ m$

~~$S(i, m) :-\ P(i, j, k, l), Q(j, k, l, m).$~~

$S(i, m) :-\ P_1(i, j), Q_1(j, k), P(i, j, k, l), Q(j, k, l, m).$

Securing prefix-correctness

$$S(\ x_1\ y_1\ x_2\ y_2\ ) :- P(x_1, x_2), Q(y_1, y_2).$$
$$\quad i \quad j \quad k \quad l \quad m$$

~~$S(i, m) :- P(i, j, k, l), Q(j, k, l, m).$~~

$$S(i, m) :- P_1(i, j), Q_1(j, k), P(i, j, k, l), Q(j, k, l, m).$$

$$P(i, j, k, l) :- a(i, j), c(k, l).$$
$$P_1(i, j) :- a(i, j).$$

Securing prefix-correctness

$$S(\underset{i}{\;} x_1 \underset{j}{\;} y_1 \underset{k}{\;} x_2 \underset{l}{\;} y_2 \underset{m}{\;}) :- P(x_1, x_2), Q(y_1, y_2).$$

~~$S(i, m) :- P(i, j, k, l), Q(j, k, l, m).$~~

$$S(i, m) :- P_1(i, j), Q_1(j, k), P(i, j, k, l), Q(j, k, l, m).$$

~~$P(i, j, k, l) :- a(i, j), c(k, l).$~~
~~$P_1(i, j) :- a(i, j).$~~

$$P(i, j, k, l) :- aux(i, j), c(k, l).$$
$$P_1(i, j) :- aux(i, j).$$
$$aux(i, j) :- a(i, j).$$

Securing prefix-correctness

$S(\ x_1\ y_1\ x_2\ y_2\ ) :- P(x_1, x_2), Q(y_1, y_2).$
$\quad i \quad j \quad k \quad l \quad m$

~~$S(i, m) :- P(i, j, k, l), Q(j, k, l, m).$~~

$S(i, m) :- P_1(i, j), Q_1(j, k), P(i, j, k, l), Q(j, k, l, m).$

~~$P(i, j, k, l) :- a(i, j), c(k, l).$~~
~~$P_1(i, j) :- a(i, j).$~~

$P(i, j, k, l) :- aux(i, j), c(k, l).$
$P_1(i, j) :- aux(i, j).$
$aux(i, j) :- a(i, j).$

$S^{bf}(i, m) :- P_1^{bf}(i, j), Q_1^{bf}(j, k), P^{bbbf}(i, j, k, l), Q^{bbbf}(j, k, l, m).$
$P^{bbbf}(i, j, k, l) :- aux^{bb}(i, j), c^{bf}(k, l).$
$P_1^{bf}(i, j) :- aux^{bf}(i, j).$
$aux^{bf}(i, j) :- a^{bf}(i, j).$

New starting point

$$1: S^{bf}(i,m) :- P_1^{bf}(i,j), Q_1^{bf}(j,k), P^{bbbf}(i,j,k,l), Q^{bbbf}(j,k,l,m).$$
$$2: P_1^{bf}(i,j) :- aux_2^{bff}(i,j,k).$$
$$3: P^{bbbf}(i,k,l,n) :- aux_2^{bff}(i,j,k), c^{bf}(l,m), P^{bbbf}(j,k,m,n).$$
$$4: aux_2^{bff}(i,j,k) :- a^{bf}(i,j), P_1^{bf}(j,k).$$
$$5: P_1^{bf}(i,j) :- aux_3^{bf}(i,j).$$
$$6: P^{bbbf}(i,j,k,l) :- aux_3^{bf}(i,j), c^{bf}(k,l).$$
$$7: aux_3^{bf}(i,j) :- a^{bf}(i,j).$$
$$8: Q_1^{bf}(i,j) :- aux_4^{bff}(i,j,k).$$
$$9: Q^{bbbf}(i,k,l,n) :- aux_4^{bff}(i,j,k), d^{bf}(l,m), Q^{bbbf}(j,k,m,n).$$
$$10: aux_4^{bff}(i,j,k) :- b^{bf}(i,j), Q_1^{bf}(j,k).$$
$$11: Q_1^{bf}(i,j) :- aux_5^{bf}(i,j).$$
$$12: Q^{bbbf}(i,j,k,l) :- aux_5^{bf}(i,j), d^{bf}(k,l).$$
$$13: aux_5^{bf}(i,j) :- b^{bf}(i,j).$$

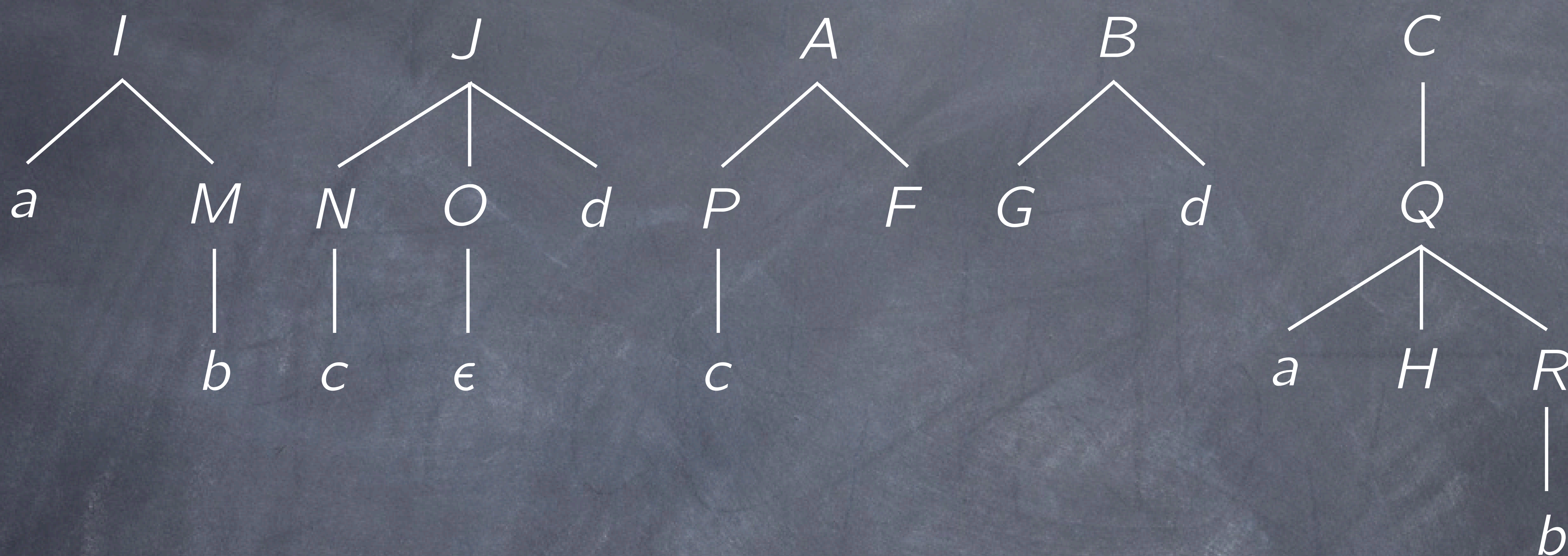
Magic-sets rewriting

$r_1: m\_P_1(i) :- m\_S(i).$
$r_2: m\_Q_1(j) :- sup_{1.1}(i,j).$
$r_3: m\_P(i,j,k) :- sup_{1.2}(i,j,k).$
$r_4: m\_Q(j,k,l) :- sup_{1.3}(i,j,k,l).$
$r_5: m\_aux_2(i) :- m\_P_1(i).$
$r_6: m\_aux_2(i) :- m\_P(i,k,l).$
$r_7: m\_P(j,k,m) :- sup_{3.2}(i,j,k,l,m).$
$r_8: m\_P_1(j) :- sup_{4.1}(i,j).$
$r_9: m\_aux_3(j) :- m\_P_1(i).$
$r_{10}: m\_aux_3(i) :- m\_P(i,j,k).$
$r_{11}: m\_aux_4(i) :- m\_Q_1(i).$
$r_{12}: m\_aux_4(i) :- m\_Q(i,k,l).$
$r_{13}: m\_Q(j,k,m) :- sup_{9.2}(i,j,k,l,m).$
$r_{14}: m\_Q_1(j) :- sup_{10.1}(i,j).$
$r_{15}: m\_aux_5(i) :- m\_Q_1(i).$
$r_{16}: m\_aux_5(i) :- m\_Q(i,j,k).$

$r_{17}: sup_{1.1}(i,j) :- m\_s(i), P_1(i,j).$
$r_{18}: sup_{1.2}(i,j,k) :- sup_{1.1}(i,j), Q_1(j,k).$
$r_{19}: sup_{1.3}(i,j,k,l) :- sup_{1.2}(i,j,k), P(i,j,k,l).$
$r_{20}: S(i,m) :- sup_{1.3}(i,j,k,l), Q(j,k,l,m).$

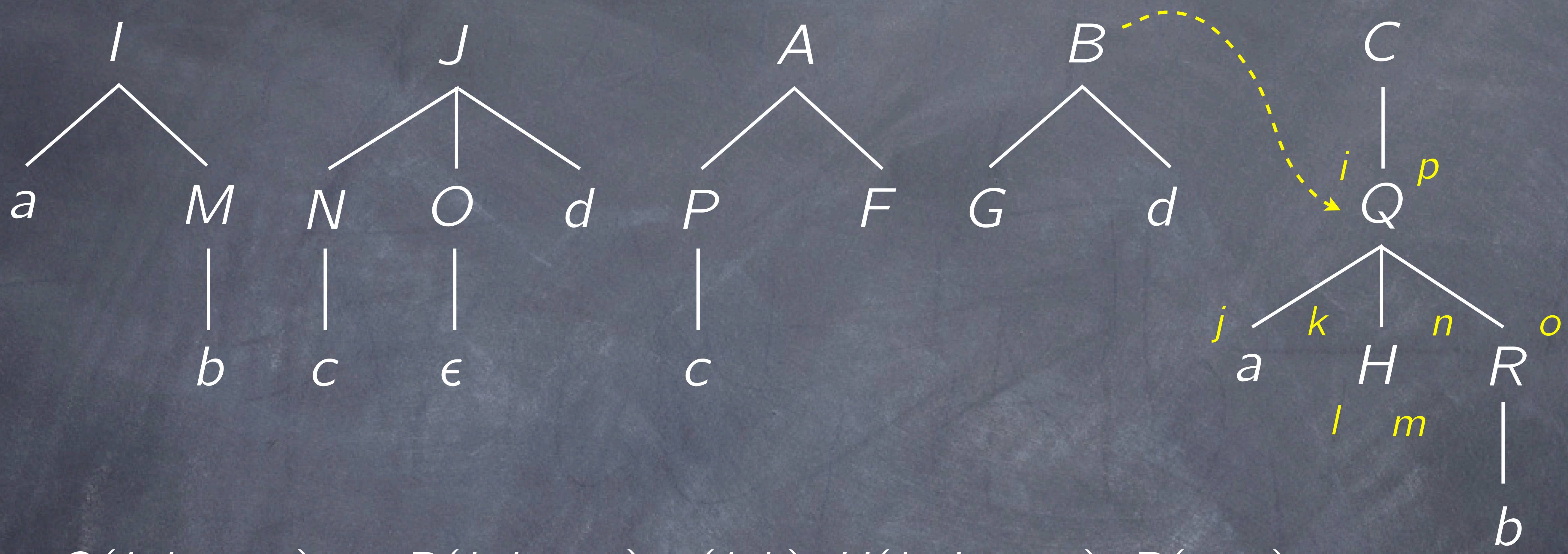
$r_{21}: P_1(i,k) :- m\_P_1(i), aux_2(i,j,k).$
$r_{22}: sup_{3.1}(i,j,k,l) :- m\_P(i,k,l), aux_2(i,j,k).$
$r_{23}: sup_{3.2}(i,j,k,l,m) :- sup_{3.1}(i,j,k,l), c(l,m).$
$r_{24}: P(i,k,l,n) :- sup_{3.2}(i,j,k,l,m), P(j,k,m,n).$
$r_{25}: sup_{4.1}(i,j) :- m\_aux_2(i), a(i,j).$
$r_{26}: aux_2(i,j,k) :- sup_{4.1}(i,j), P_1(j,k).$
$r_{27}: P_1(i,j) :- m\_P_1(i), aux_3(i,j).$
$r_{28}: sup_{6.1}(i,j,k) :- m\_P(i,j,k), aux_3(i,j).$
$r_{29}: P(i,j,k,l) :- sup_{6.1}(i,j,k), c(k,l).$
$r_{30}: aux_3(i,j) :- m\_aux_3(i), a(i,j).$
$r_{31}: Q_1(i,k) :- m\_Q_1(i), aux_4(i,_j,k).$
$r_{32}: sup_{9.1}(i,j,k,l) :- m\_Q(i,k,l), aux_4(i,j,k).$
$r_{33}: sup_{9.2}(i,j,k,l,m) :- sup_{9.1}(i,j,k,l), d(l,m).$
$r_{34}: Q(i,k,l,n) :- sup_{9.2}(i,j,k,l,m), Q(j,k,m,n).$
$r_{35}: sup_{10.1}(i,j) :- m\_aux_4(i), b(i,j).$
$r_{36}: aux_4(i,j,k) :- sup_{10.1}(i,j), Q_1(j,k).$
$r_{37}: Q_1(i,j) :- m\_Q_1(i), aux_5(i,j).$
$r_{38}: sup_{12.1}(i,j,k) :- m\_Q(i,j,k), aux_5(i,j).$
$r_{39}: Q(i,j,k,l) :- sup_{12.1}(i,j,k), d(k,l).$
$r_{40}: aux_5(i,j) :- m\_aux_5(i), b(i,j).$

Application to tree-adjoining grammar



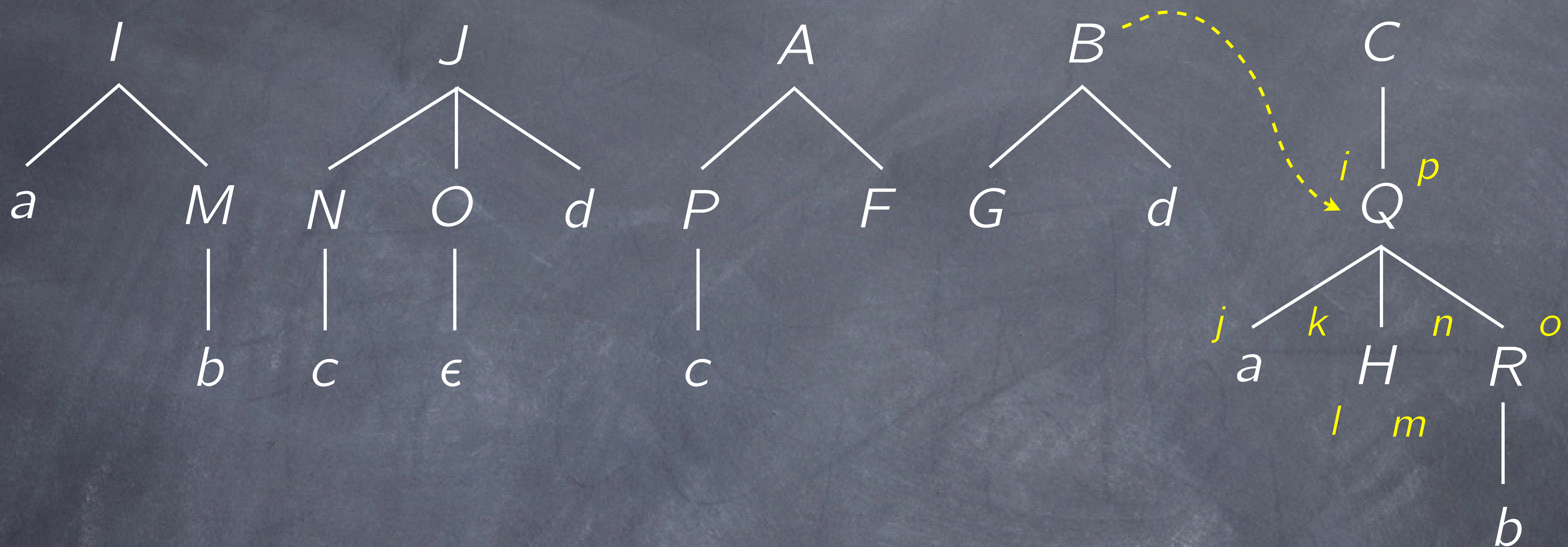
Application to tree-adjoining grammar

Application to tree-adjoining grammar



$$Q(i, l, m, p) :- B(i, j, o, p), a(j, k), H(k, l, m, n), R(n, o).$$

Application to tree-adjoining grammar



$$Q(i, l, m, p) :- B(i, j, o, p), a(j, k), H(k, l, m, n), R(n, o).$$

$$aux(i, j, k, l) :- B_1(i, j), a(j, k), H_1(k, l).$$

$$Q_1(i, l) :- aux(i, j, k, l).$$

$$Q(i, l, m, p) :- aux(i, j, k, l), H(k, l, m, n), R(n, o), B(i, j, o, p).$$

Magic-sets rewriting

$m\_B_1(i) :- m\_aux(i).$
$m\_H_1(k) :- sup_{1.2}(i, j, k).$
$m\_aux(i) :- m\_Q_1(i).$
$m\_aux(i) :- m\_Q(i, l, m).$
$m\_H(k, l, m) :- sup_{3.1}(i, j, k, l, m).$
$m\_R(n) :- sup_{3.2}(i, j, l, m, n).$
$m\_B(i, j, o) :- sup_{3.3}(i, j, l, m, o).$

$sup_{1.1}(i, j) :- m\_aux(i), B_1(i, j).$
$sup_{1.2}(i, j, k) :- sup_{1.1}(i, j), a(j, k).$
$aux(i, j, k, l) :- sup_{1.2}(i, j, k), H_1(k, l).$
$Q_1(i, l) :- m\_Q_1(i, l), aux(i, j, k, l).$
$sup_{3.1}(i, j, k, l, m) :- m\_Q(i, l, m), aux(i, j, k, l).$
$sup_{3.2}(i, j, l, m, n) :- sup_{3.1}(i, j, k, l, m), H(k, l, m, n).$
$sup_{3.3}(i, j, l, m, o) :- sup_{3.2}(i, j, l, m, n), R(n, o).$
$Q(i, l, m, p) :- sup_{3.3}(i, j, l, m, o), B(i, j, o, p).$

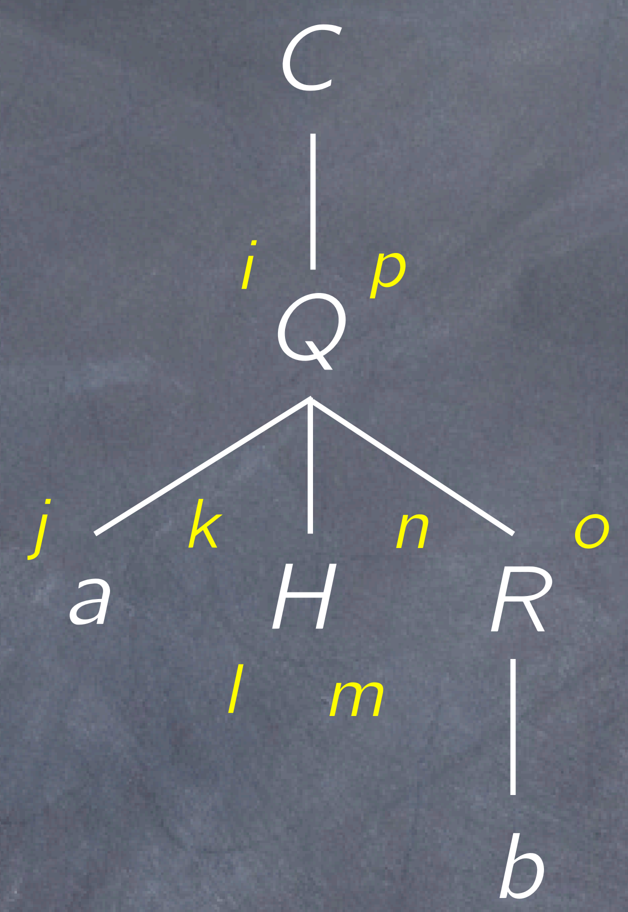
Magic-sets rewriting

$m\_B_1(i) :- m\_aux(i).$
$m\_H_1(k) :- sup_{1.2}(i, j, k).$
$m\_aux(i) :- m\_Q_1(i).$
$m\_aux(i) :- m\_Q(i, l, m).$
$m\_H(k, l, m) :- sup_{3.1}(i, j, k, l, m).$
$m\_R(n) :- sup_{3.2}(i, j, l, m, n).$
$m\_B(i, j, o) :- sup_{3.3}(i, j, l, m, o).$

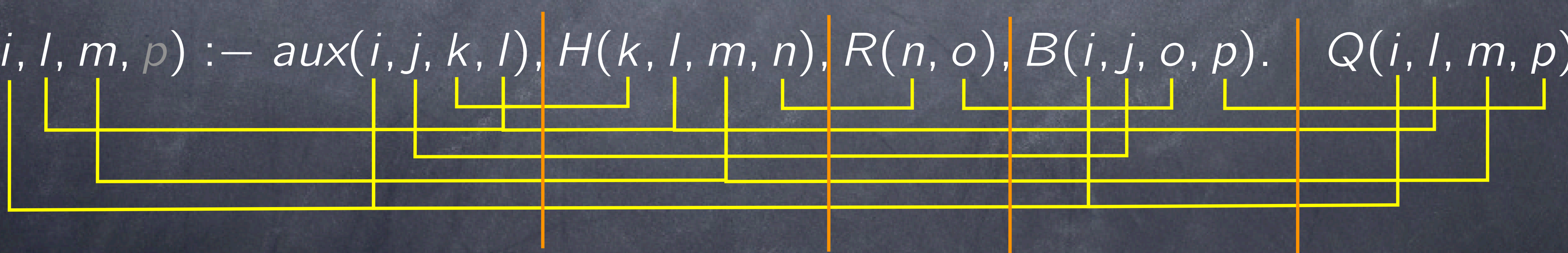
$sup_{1.1}(i, j) :- m\_aux(i), B_1(i, j).$
$sup_{1.2}(i, j, k) :- sup_{1.1}(i, j), a(j, k).$
$aux(i, j, k, l) :- sup_{1.2}(i, j, k), H_1(k, l).$
$Q_1(i, l) :- m\_Q_1(i, l), aux(i, j, k, l).$
$sup_{3.1}(i, j, k, l, m) :- m\_Q(i, l, m), aux(i, j, k, l).$
$sup_{3.2}(i, j, l, m, n) :- sup_{3.1}(i, j, k, l, m), H(k, l, m, n).$
$sup_{3.3}(i, j, l, m, o) :- sup_{3.2}(i, j, l, m, n), R(n, o).$
$Q(i, l, m, p) :- sup_{3.3}(i, j, l, m, o), B(i, j, o, p).$

$O(n^6)$

Time and space complexity



$$Q(i, l, m, p) :- aux(i, j, k, l), H(k, l, m, n), R(n, o), B(i, j, o, p). \quad Q(i, l, m, p)$$



Optimal complexity bounds are obtained without any fine-tuning