

inferência em LPO

cláusulas de Horn

Cláusula de Horn \rightarrow Cláusula definida | Cláusula Objetivo

Cláusula definida $\rightarrow (\text{Símbolo}_1 \wedge \dots \wedge \text{Símbolo}_M) \Rightarrow \text{Símbolo}$

Cláusula objetivo $\rightarrow (\text{Símbolo}_1 \wedge \dots \wedge \text{Símbolo}_M) \Rightarrow \text{False}$

permitem dois tipos de **inferência em tempo linear**

- encadeamento adiante (*forward chaining*)
- encadeamento retrógrado (*backward chaining*)

em ambos se pretende verificar $KB \models q$? q decorre do conhecimento na KB?

encadeamento adiante

começa desde os factos na base de conhecimento
se todas as premissas de uma implicação são conhecidas
então a sua conclusão é adicionada ao conjunto de factos
continua o processo até a interrogação q ser adicionada
caso em que a resposta é positiva
ou até não se poderem fazer mais inferências
neste caso a resposta é negativa

**raciocínio orientado
pelos dados**

encadeamento retrógrado

começa desde a interrogação q

se já é conhecida na KB a resposta é desde logo afirmativa

caso contrário procura resolver as implicações na KB cuja conclusão é q

resolvendo sucessivamente as implicações que lhes dão origem

se consegue provar todas as premissas necessárias para concluir q , então a resposta é positiva

caso contrário a resposta é negativa

raciocínio orientado
pelo objetivo

encadeamentos

custo do encadeamento retrógrado

em geral é menor do que linear

porque só procura os factos relevantes para o objetivo

custo do encadeamento adiante

sendo linear, pode ainda assim gerar uma grande quantidade de conhecimento irrelevante

raciocínio humano usa os dois tipos, com controlo do 2º

para não gerarem demasiados detalhes desnecessários

cláusulas de Horn e grafo E-OU

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

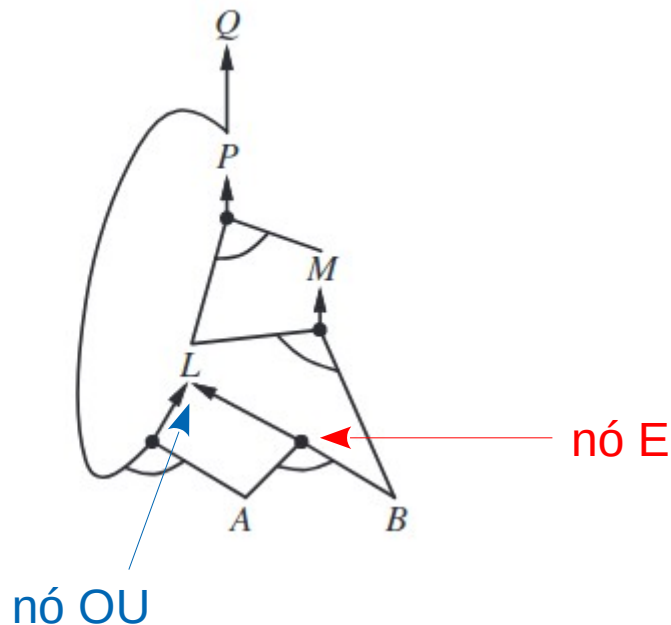
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

A

B



exemplo de encadeamento adiante

$KB \models Q?$

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

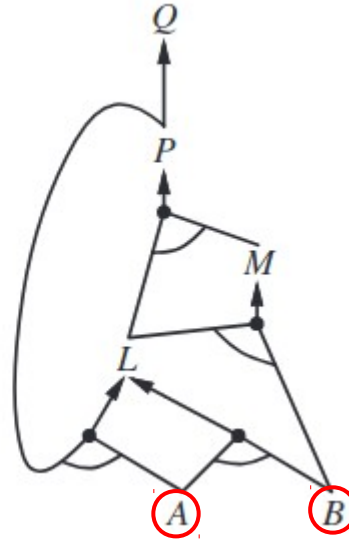
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

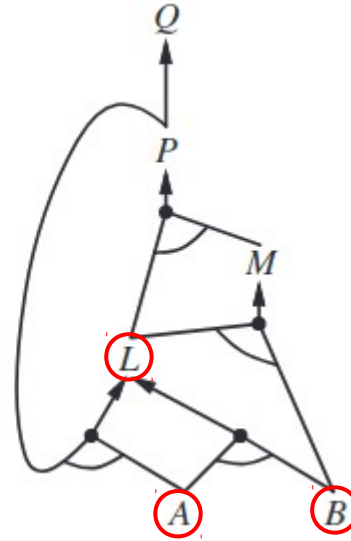
A

B



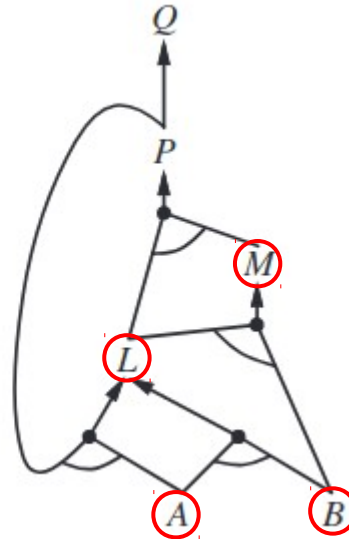
exemplo de encadeamento adiante

$P \Rightarrow Q$
 $L \wedge M \Rightarrow P$
 $\textcircled{B} \wedge L \Rightarrow M$
 $\textcircled{A} \wedge P \Rightarrow L$
 $\textcircled{A} \wedge \textcircled{B} \Rightarrow \textcircled{L}$
 \textcircled{A}
 \textcircled{B}



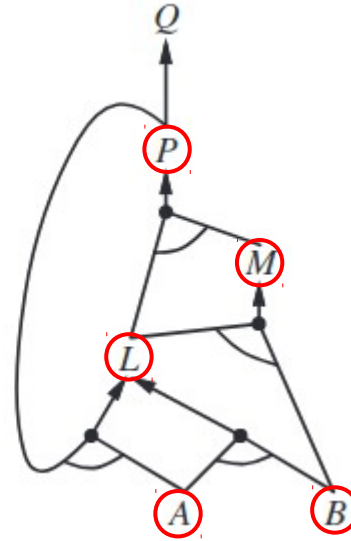
exemplo de encadeamento adiante

$P \Rightarrow Q$
 $(L \wedge M) \Rightarrow P$
 $(B \wedge L) \Rightarrow M$
 $(A \wedge P) \Rightarrow L$
 $(A \wedge B) \Rightarrow L$
 A
 B



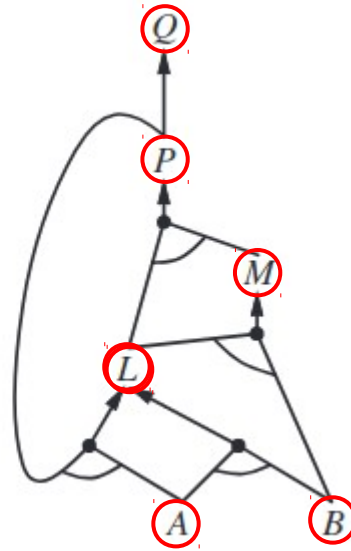
exemplo de encadeamento adiante

$P \Rightarrow Q$
 $(L \wedge M) \Rightarrow P$
 $(B \wedge L) \Rightarrow M$
 $(A \wedge P) \Rightarrow L$
 $(A \wedge B) \Rightarrow L$
 A
 B



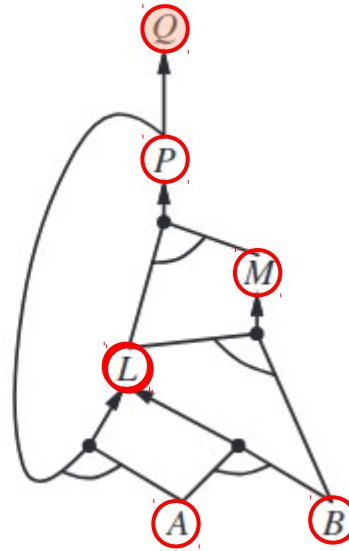
exemplo de encadeamento adiante

$P \Rightarrow Q$
 $L \wedge M \Rightarrow P$
 $B \wedge L \Rightarrow M$
 $A \wedge P \Rightarrow L$
 $A \wedge B \Rightarrow L$
 A
 B



exemplo de encadeamento adiante

$P \Rightarrow Q$
 $L \wedge M \Rightarrow P$
 $B \wedge L \Rightarrow M$
 $A \wedge P \Rightarrow L$
 $A \wedge B \Rightarrow L$
 A
 B



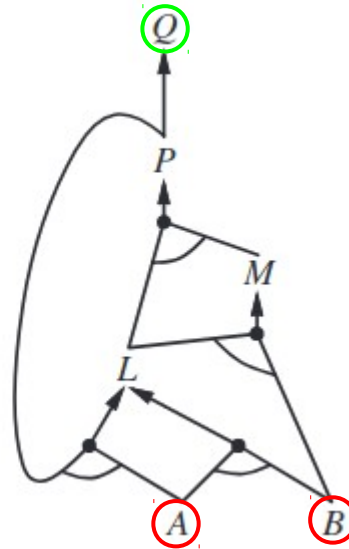
sim!

$KB \models Q$

exemplo de encadeamento retrógrado

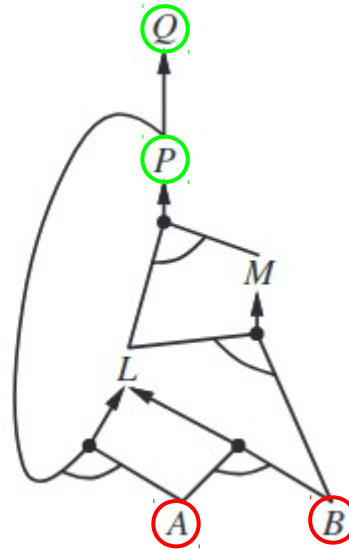
$KB \models Q?$

$P \Rightarrow Q$
 $L \wedge M \Rightarrow P$
 $B \wedge L \Rightarrow M$
 $A \wedge P \Rightarrow L$
 $A \wedge B \Rightarrow L$
 A
 B



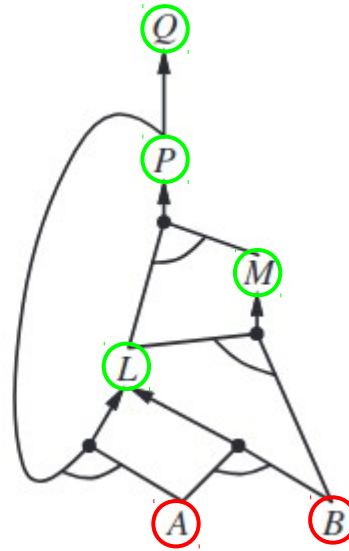
exemplo de encadeamento retrógrado

$\textcircled{P} \Rightarrow \textcircled{Q}$
 $L \wedge M \Rightarrow \textcircled{P}$
 $B \wedge L \Rightarrow M$
 $A \wedge P \Rightarrow L$
 $A \wedge B \Rightarrow L$
 \textcircled{A}
 \textcircled{B}



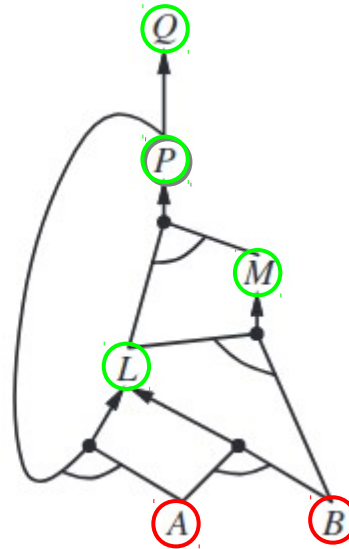
exemplo de encadeamento retrógrado

$P \Rightarrow Q$
 $L \wedge M \Rightarrow P$
 $B \wedge L \Rightarrow M$
 $A \wedge P \Rightarrow L$
 $A \wedge B \Rightarrow L$
 A
 B



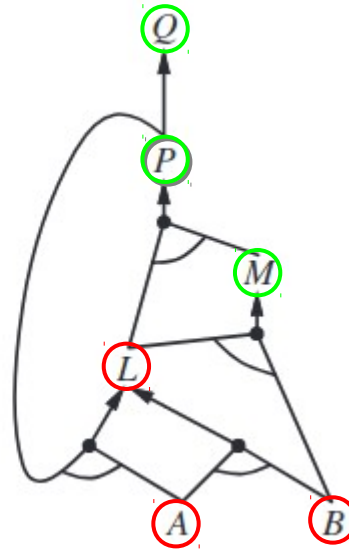
exemplo de encadeamento retrógrado

$P \Rightarrow Q$
 $L \wedge M \Rightarrow P$
 $B \wedge L \Rightarrow M$
 $A \wedge P \Rightarrow L$
 $A \wedge B \Rightarrow L$
 A
 B



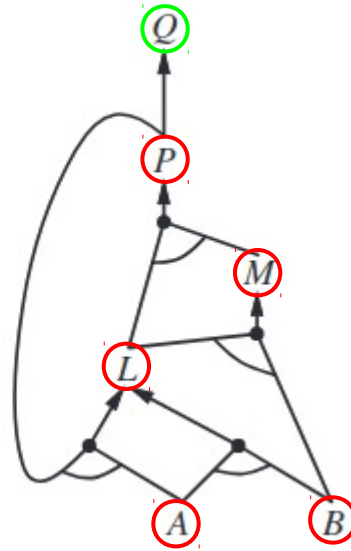
exemplo de encadeamento retrógrado

$P \Rightarrow Q$
 $L \wedge M \Rightarrow P$
 $B \wedge L \Rightarrow M$
 $A \wedge P \Rightarrow L$
 $A \wedge B \Rightarrow L$
 A
 B



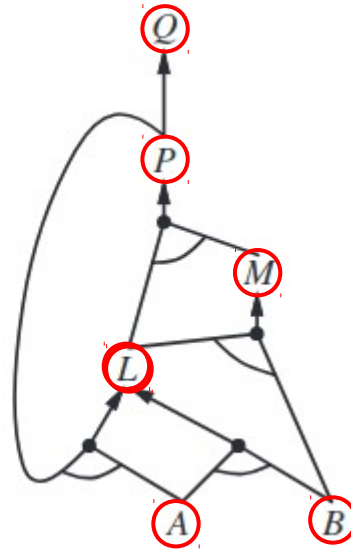
exemplo de encadeamento retrógrado

$P \Rightarrow Q$
 $L \wedge M \Rightarrow P$
 $B \wedge L \Rightarrow M$
 $A \wedge P \Rightarrow L$
 $A \wedge B \Rightarrow L$
 A
 B



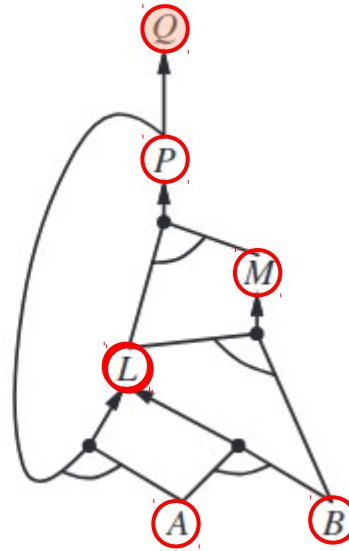
exemplo de encadeamento retrógrado

$P \Rightarrow Q$
 $L \wedge M \Rightarrow P$
 $B \wedge L \Rightarrow M$
 $A \wedge P \Rightarrow L$
 $A \wedge B \Rightarrow L$
 A
 B



exemplo de encadeamento retrógrado

$P \Rightarrow Q$
 $L \wedge M \Rightarrow P$
 $B \wedge L \Rightarrow M$
 $A \wedge P \Rightarrow L$
 $A \wedge B \Rightarrow L$
 A
 B



sim!

$KB \models Q$

eficiência do encadeamento adiante

problema: escolha da ordem das regras:

exemplo

$\text{iia2021}(X) \wedge \text{minorante}(50000, X) \Rightarrow \text{prim_insc}(\text{iia2021}, X).$

é mais eficiente procurar primeiro os alunos de iia2021 e só então procurar, desses, os que têm o nº maior do que 50000

solução:

semelhante à heurística MRV (*minimum remaining values*) de CSP

eficiência do encadeamento adiante

problema: unificações redundantes

solução: encadeamento adiante incremental

cada novo facto inferido na iteração t deverá resultar de, pelo menos, um facto inferido na iteração $t-1$

(caso contrário já teria sido inferido antes)

tipicamente há poucas novas regras alteradas por um novo facto → procura mais eficiente

eficiência do encadeamento adiante

problema: factos irrelevantes

solução:

integrar alguma forma de inferência retrógrada

para dirigir a procura das regras a avaliar

encadeamento adiante → sistemas de produção

sistemas periciais

modelos de raciocínio humano: ACT, SOAR

eficiência do encadeamento retrógrado

é uma procura em profundidade numa árvore E-OU

OU: um predicado ter várias cláusulas e ser provado por qualquer uma delas

E: todas as conjunções no antecedente têm de ser provadas
(para se poder inferir o consequente)

problemas: estados repetidos e incompletude

solução: ordenação das cláusulas de cada predicado

prolog

execução de programas em prolog: encadeamento retrógrado

exemplo: apender uma lista (2º arg) a outra (1º arg) com resultado no 3º arg.

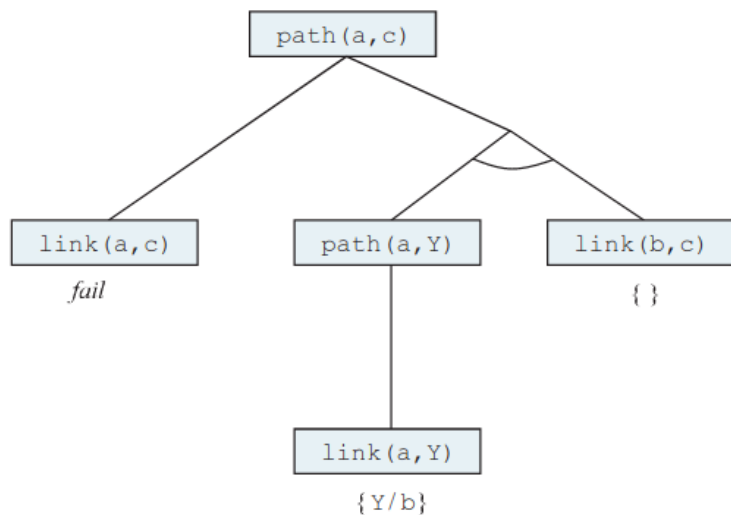
```
append([],Y,Y).
```

```
append([A | X],Y,[A | Z]) :- append(X,Y,Z).
```

ordem das cláusulas é importante!

$\text{path}(X,Z) \text{ :- link}(X,Z).$

$\text{path}(X,Z) \text{ :- path}(X,Y), \text{link}(Y,Z).$



$\text{path}(X,Z) \text{ :- path}(X,Y), \text{link}(Y,Z).$

$\text{path}(X,Z) \text{ :- link}(X,Z).$

