



TRƯỜNG ĐẠI HỌC  
BÁCH KHOA HÀ NỘI  
HANOI UNIVERSITY  
OF SCIENCE AND TECHNOLOGY

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG  
SCHOOL OF INFORMATION AND COMMUNICATIONS TECHNOLOGY

CLOUD  
COMPUTING

SOFTWARE

HARDWARE

TRANSACTIONS

COMMUNICATIONS  
TECHNOLOGY

01  
10  
01

# IT3160

## Nhập môn Trí tuệ nhân tạo

*Artificial Intelligence*

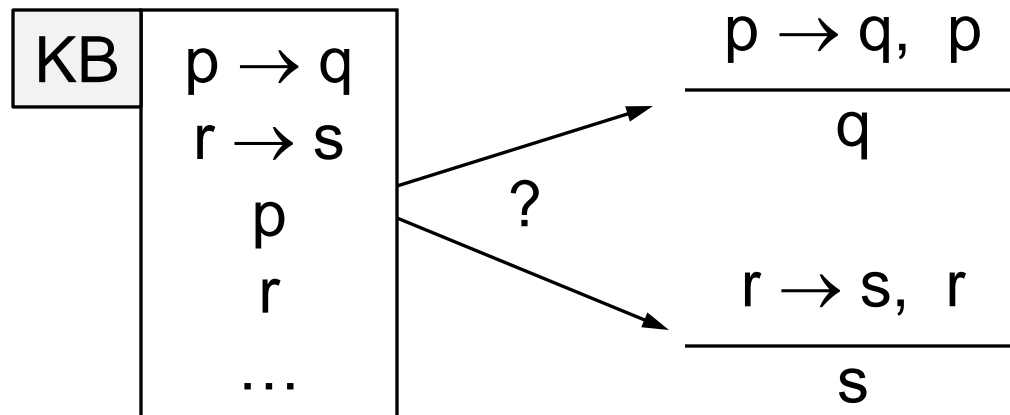
PGS.TS. Lê Thanh Hương  
Trường Công nghệ thông tin và Truyền thông  
Đại Học Bách Khoa Hà Nội

ONE LOVE. ONE FUTURE.

- Chương 1 - Giới thiệu về Trí tuệ nhân tạo
- Chương 2 - Tác tử
- Chương 3 - Giải quyết vấn đề
- **Chương 4 - Logic và suy diễn**
- Chương 5 - Học máy

# Suy diễn logic và Tìm kiếm

- Để chứng minh định lý  $\alpha$  là đúng đối với tập giả thiết  $KB$ , cần áp dụng một chuỗi các luật suy diễn đúng đắn
- **Vấn đề:** Ở mỗi bước suy diễn, có nhiều luật có thể áp dụng được
  - **Chọn luật nào để áp dụng tiếp theo?**
- Đây là vấn đề của bài toán tìm kiếm (search)



# Chuyển đổi các biểu thức logic

- Trong logic định đề
  - Một biểu thức có thể bao gồm nhiều liên kết:  $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
  - Một biểu thức có thể bao gồm nhiều biểu thức con (lòng) khác
- Chúng ta có cần sử dụng tất cả các liên kết logic để biểu diễn một biểu thức phức tạp?
  - Không.
  - Chúng ta có thể viết lại (chuyển đổi) một biểu thức logic định đề thành một biểu thức tương đương *chỉ chứa các liên kết*  $\neg, \wedge, \vee$

# Các dạng chuẩn

- Các biểu thức trong logic định đề có thể được chuyển đổi về một trong các dạng chuẩn (Normal forms)
  - Giúp đơn giản hóa quá trình suy diễn
- **Dạng chuẩn kết hợp** (Conjunctive normal form – CNF)
  - Là kết hợp (liên kết VÀ) của các mệnh đề (clauses)
  - Mỗi mệnh đề (clause) là một liên kết HOẶC của các ký hiệu định đề đơn
  - Ví dụ:  $(p \vee q) \wedge (\neg q \vee \neg r \vee s)$
- **Dạng chuẩn tuyển** (Disjunctive normal form – DNF)
  - Là liên kết HOẶC của các mệnh đề (clauses)
  - Mỗi mệnh đề (clause) là một liên kết VÀ của các ký hiệu định đề đơn
  - Ví dụ:  $(p \wedge \neg q) \vee (\neg p \wedge r) \vee (r \wedge \neg s)$

# Chuyển đổi về dạng chuẩn CNF: Ví dụ

Chuyển đổi về dạng chuẩn CNF:  $\neg(p \rightarrow q) \vee (r \rightarrow p)$

1. Loại bỏ các liên kết  $\rightarrow, \leftrightarrow$

$$\neg(\neg p \vee q) \vee (\neg r \vee p)$$

2. Sử dụng các phép biến đổi tương đương (vd: luật De Morgan và phép phủ định 2 lần)

$$(p \wedge \neg q) \vee (\neg r \vee p)$$

3. Sử dụng các luật kết hợp (associative rules) và phân bố (distributive rules)

$$(p \vee \neg r \vee p) \wedge (\neg q \vee \neg r \vee p)$$

$$(p \vee \neg r) \wedge (\neg q \vee \neg r \vee p)$$

# Bài toán chứng minh thỏa mãn (SAT)

- Mục đích của bài toán chứng minh thỏa mãn (Satisfiability – SAT) là xác định một biểu thức ở dạng chuẩn kết hợp (CNF) có thể thỏa mãn được hay không
  - Tức là chứng minh biểu thức đó là đúng hay không
  - Ví dụ:  $(p \vee q \vee \neg r) \wedge (\neg p \vee \neg r \vee s) \wedge (\neg p \vee q \vee \neg t)$
- Đây là một trường hợp của bài toán thỏa mãn ràng buộc (CSP)
  - Tập các biến
    - Các ký hiệu định đề (ví dụ:  $p, q, r, s, t$ )
    - Các giá trị (hằng) logic *đúng*, *sai*
  - Tập các ràng buộc
    - Tất cả các mệnh đề (được liên kết bởi phép VÀ) trong biểu thức phải đúng
    - Với mỗi mệnh đề, ít nhất một trong các định đề đơn phải đúng

# Giải quyết bài toán SAT

- Phương pháp **Backtracking**
  - Áp dụng chiến lược tìm kiếm theo chiều sâu (Depth-first search)
  - Xét một biến (một định đề đơn), xét các khả năng gán giá trị (đúng/sai) cho biến đó
  - Lặp lại, cho đến khi tất cả các biến được gán giá trị, hoặc việc gán giá trị cho tập con của tập tất cả các biến, làm cho **biểu thức là sai**
- Các phương pháp **tối ưu hóa lặp (Iterative optimization methods)**
  - Bắt đầu với một phép gán ngẫu nhiên các giá trị đúng/sai cho các ký hiệu định đề
  - Đổi giá trị (*đúng* thành *sai* / *sai* thành *đúng*) đối với một biến
  - Heuristic: ưu tiên các phép gán giá trị làm cho nhiều mệnh đề (hơn) đúng
  - Sử dụng các phương pháp tìm kiếm cục bộ: Simulated Annealing, Walk-SAT



# Bài toán suy diễn vs. Bài toán thỏa mãn được

- **Bài toán suy diễn logic**

- Cần chứng minh: biểu thức logic (định lý)  $\alpha$  được bao hàm bởi tập các mệnh đề  $KB$
- Nói cách khác: với mọi phép diễn giải mà trong đó  $KB$  đúng, thì  $\alpha$  có đúng?

- **Bài toán thỏa mãn được (SAT)**

- Có tồn tại một phép gán giá trị đúng/sai cho các ký hiệu định đề (một phép diễn giải) sao cho biểu thức  $\alpha$  là đúng?

- Mỗi quan hệ:

$KB \models \alpha$	nếu và chỉ nếu:
$(KB \wedge \neg \alpha)$	là <b>không thể thỏa mãn được</b>
	<b>(unsatisfiable)</b>

# Luật suy diễn hợp giải

- Luật suy diễn **hợp giải (Resolution)**

$$\frac{p \vee q, \quad \neg q \vee r}{p \vee r}$$

- Luật suy diễn hợp giải áp dụng được đối với các biểu thức logic ở dạng chuẩn CNF
- Luật suy diễn hợp giải có tính *đúng đắn (sound)*, nhưng *không có tính hoàn chỉnh (incomplete)*
  - Tập giả thiết (cơ sở tri thức) *KB* chứa biểu thức  $(p \wedge q)$
  - Cần chứng minh:  $(p \vee q)$  ?
  - Luật suy diễn hợp giải không thể suy ra được biểu thức cần chứng minh!

# Giải thuật hợp giải

- Chuyển đổi tất cả các biểu thức trong KB về dạng chuẩn CNF
  - Áp dụng liên tiếp luật suy diễn hợp giải (Resolution rule) bắt đầu từ:  $(KB \wedge \neg\alpha)$ 
    - KB là kết hợp của các biểu thức ở dạng chuẩn CNF
    - Do đó,  $(KB \wedge \neg\alpha)$  cũng là một biểu thức ở dạng chuẩn CNF!
  - Quá trình áp dụng luật suy diễn hợp giải dừng lại khi:
    - Có mâu thuẫn xảy ra
      - Sau khi hợp giải, thu được (suy ra) biểu thức rỗng (mâu thuẫn)
- $$\frac{p, \neg p}{\{}}$$
- Không có biểu thức mới nào được sinh ra nữa

# Chứng minh bằng hợp giải: Ví dụ (1)

- Giả sử có tập giả thiết KB
  - $p \wedge q$
  - $p \rightarrow r$
  - $(q \wedge r) \rightarrow s$
- Cần chứng minh định lý  $s$
- Bước 1. Chuyển đổi KB về dạng chuẩn CNF
  - $(p \rightarrow r)$  được chuyển thành  $(\neg p \vee r)$
  - $((q \wedge r) \rightarrow s)$  được chuyển thành  $(\neg q \vee \neg r \vee s)$
- Bước 2. Phủ định biểu thức cần chứng minh
  - $\neg s$
- Bước 3. Áp dụng liên tiếp luật hợp giải đối với  $(KB \wedge \neg \alpha)$ :  
 $\{p, q, \neg p \vee r, \neg q \vee \neg r \vee s, \neg s\}$

# Chứng minh bằng hợp giải: Ví dụ (2)

- Bắt đầu quá trình hợp giải, ta có tập các mệnh đề:

1)  $p$

2)  $q$

3)  $\neg p \vee r$

4)  $\neg q \vee \neg r \vee s$

5)  $\neg s$

- Hợp giải 1) và 3), ta thu được

6)  $r$

- Hợp giải 2) và 4), ta thu được

7)  $\neg r \vee s$

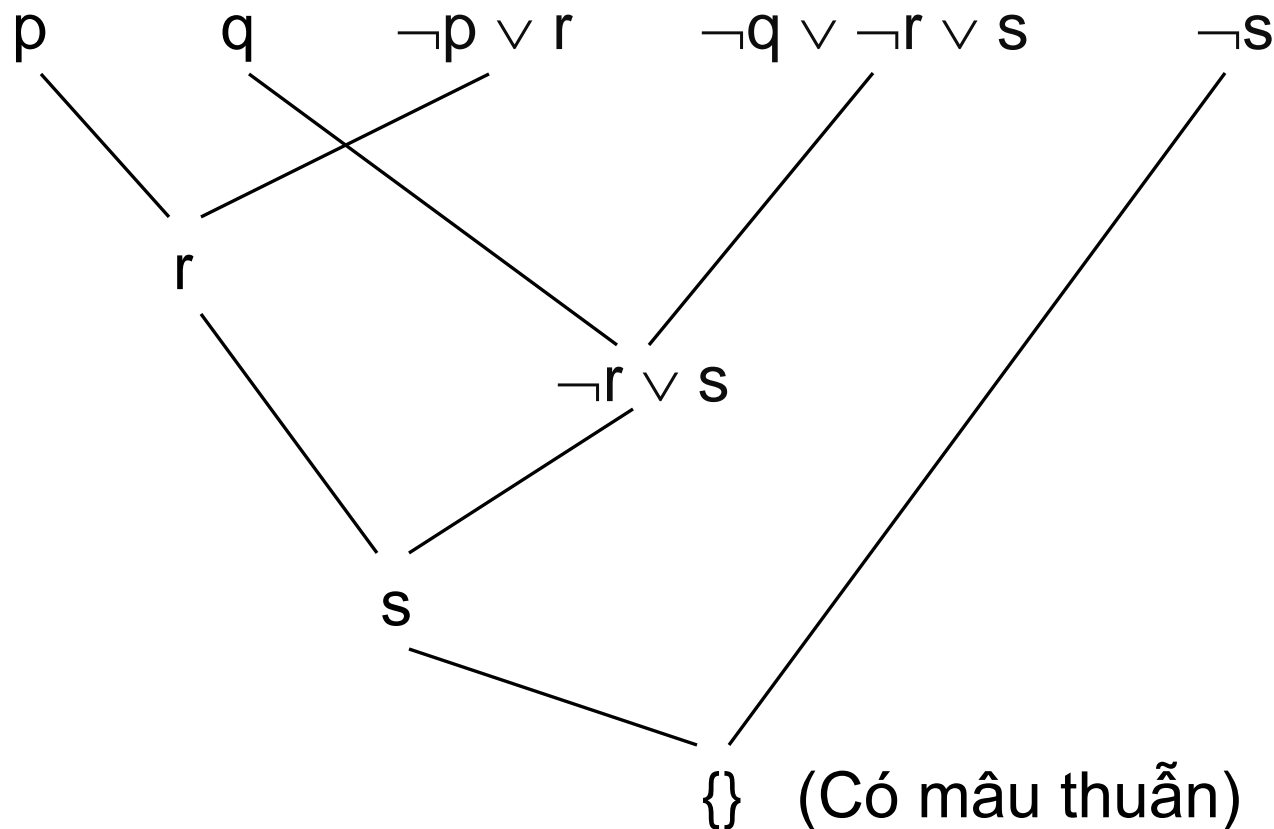
- Hợp giải 6) và 7), ta thu được

8)  $s$

- Hợp giải 8) và 5), ta thu được mâu thuẫn ( $\{\}$ )

- Tức là biểu thức ban đầu ( $s$ ) được chứng minh là đúng

# Chứng minh bằng hợp giải: Ví dụ (3)



# Dạng chuẩn Horn

- Một biểu thức logic ở dạng chuẩn Horn nếu:
  - Biểu thức đó là một liên kết VÀ của các mệnh đề
  - Mỗi mệnh đề là một liên kết HOẶC các ký hiệu (literals), và có tối đa là 1 ký hiệu khẳng định (positive literal)
  - Ví dụ:  $(p \vee \neg q) \wedge (\neg p \vee \neg r \vee s)$
- Không phải mọi biểu thức logic định đề đều có thể được chuyển về dạng chuẩn Horn!
- Biểu diễn tập giả thiết KB ở dạng chuẩn Horn
  - **Các luật (Rules)**
    - $(\neg p_1 \vee \neg p_2 \vee \dots \vee \neg p_n \vee q)$
    - Tương đương với luật:  $(p_1 \wedge p_2 \wedge \dots \wedge p_n \rightarrow q)$
  - **Các sự kiện (Facts)**
    - $p, q$
  - **Các ràng buộc toàn vẹn (Integrity constraints)**
    - $(\neg p_1 \vee \neg p_2 \vee \dots \vee \neg p_n)$
    - Tương đương với luật:  $(p_1 \wedge p_2 \wedge \dots \wedge p_n \rightarrow \text{sai})$

# Luật suy diễn Modus Ponens tổng quát

$$\frac{(p_1 \wedge p_2 \wedge \dots \wedge p_n \rightarrow q), p_1, p_2, \dots, p_n}{q}$$

- Luật suy diễn Modus Ponens có tính *đúng dẫn (sound)* và *hoàn chỉnh (complete)*, đối với các ký hiệu định đề và đối với tập các biểu thức KB ở dạng chuẩn Horn
- Luật suy diễn Modus Ponens có thể được sử dụng với cả 2 chiến lược suy diễn (suy diễn tiến và suy diễn lùi)



# Suy diễn tiến (forward chaining)

- Với một tập các mệnh đề giả thiết (cơ sở tri thức)  $KB$ , cần suy ra mệnh đề kết luận  $Q$
- Ý tưởng: Lặp lại 2 bước sau cho đến khi suy ra được kết luận
  - Áp dụng các luật có mệnh đề giả thiết được thỏa mãn trong  $KB$
  - Bổ sung kết luận của các luật đó vào  $KB$

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

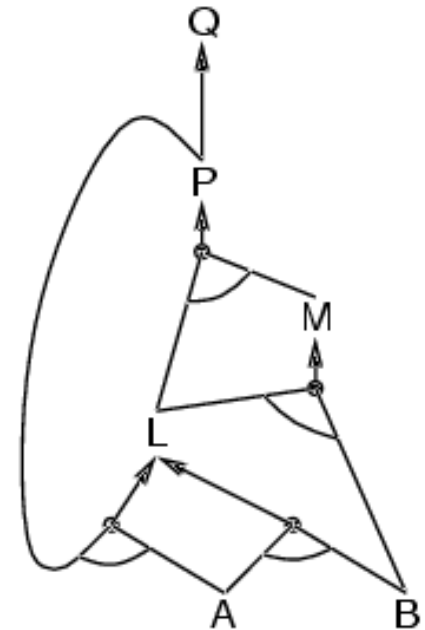
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn tiến: Ví dụ (1)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

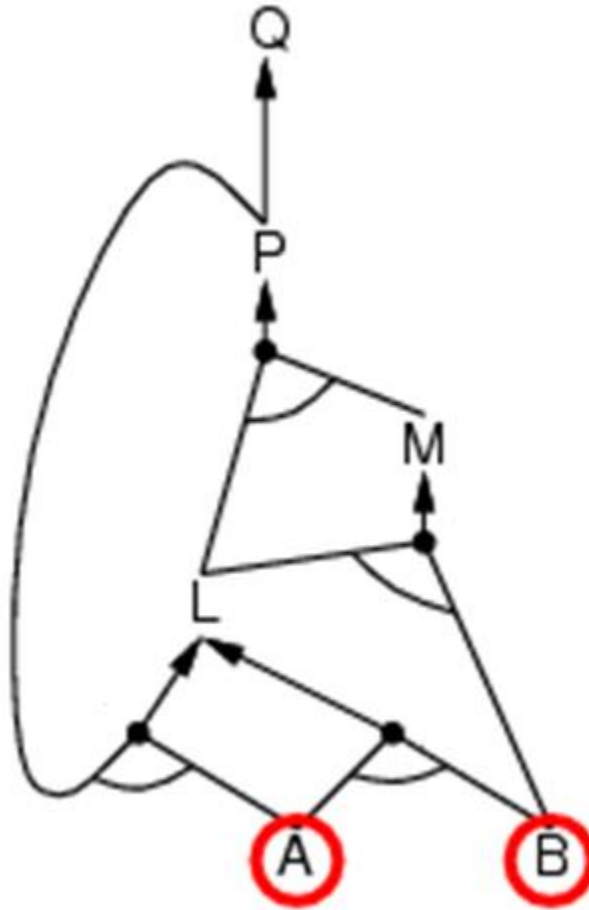
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn tiến: Ví dụ (2)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

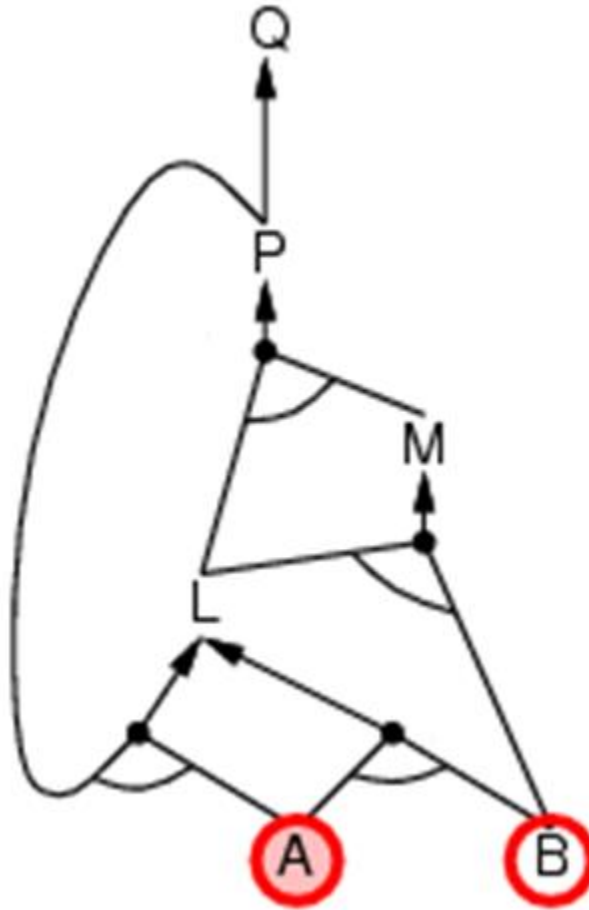
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn tiên: Ví dụ (3)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

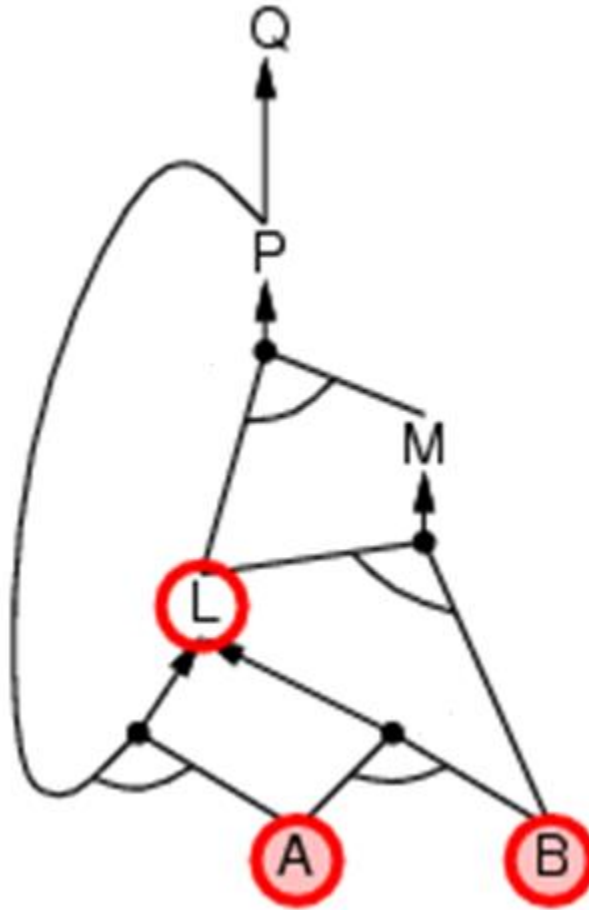
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn tiến: Ví dụ (4)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

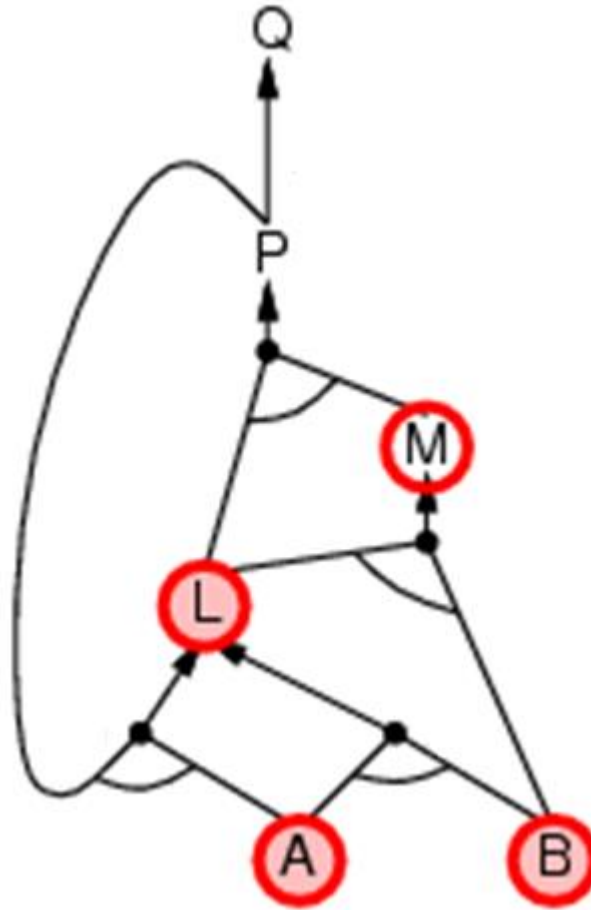
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn tiến: Ví dụ (5)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

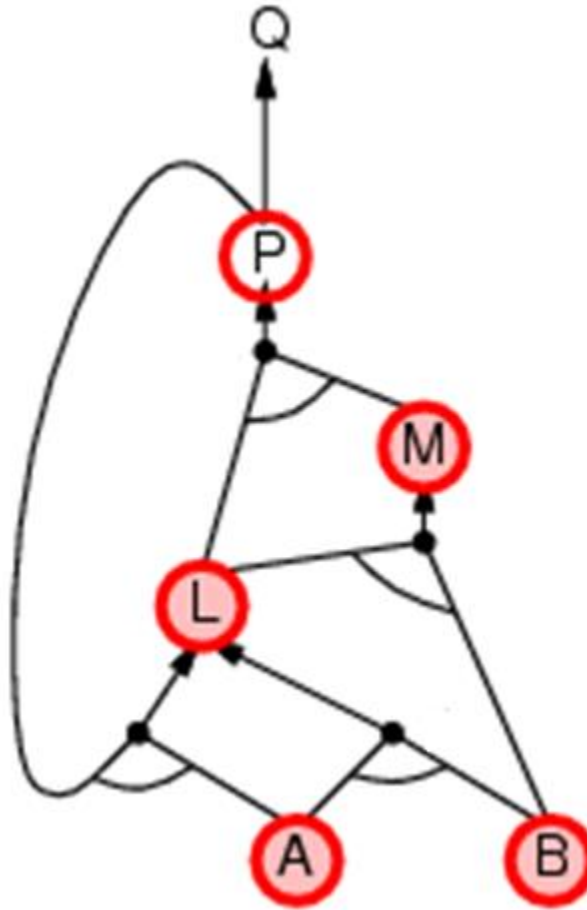
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn tiến: Ví dụ (6)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

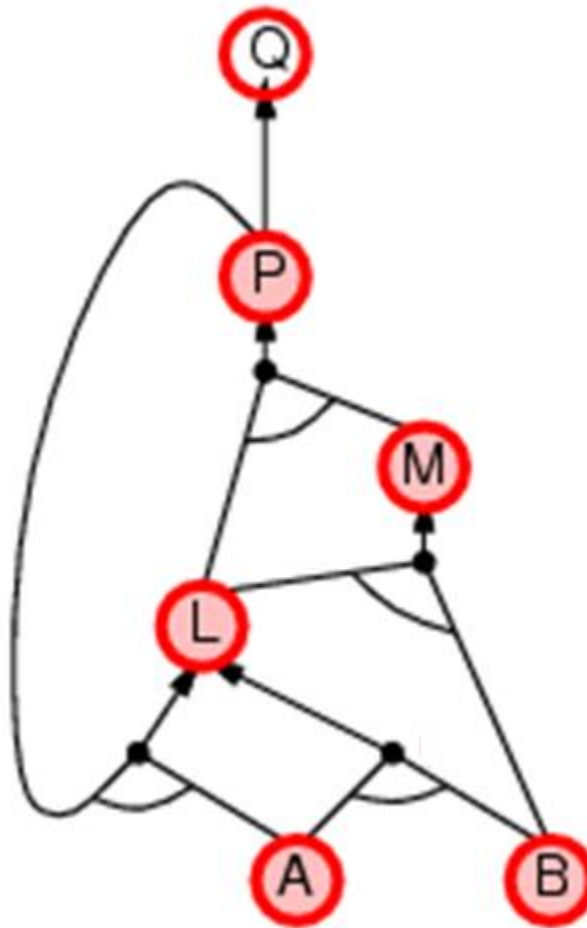
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn tiến: Ví dụ (7)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

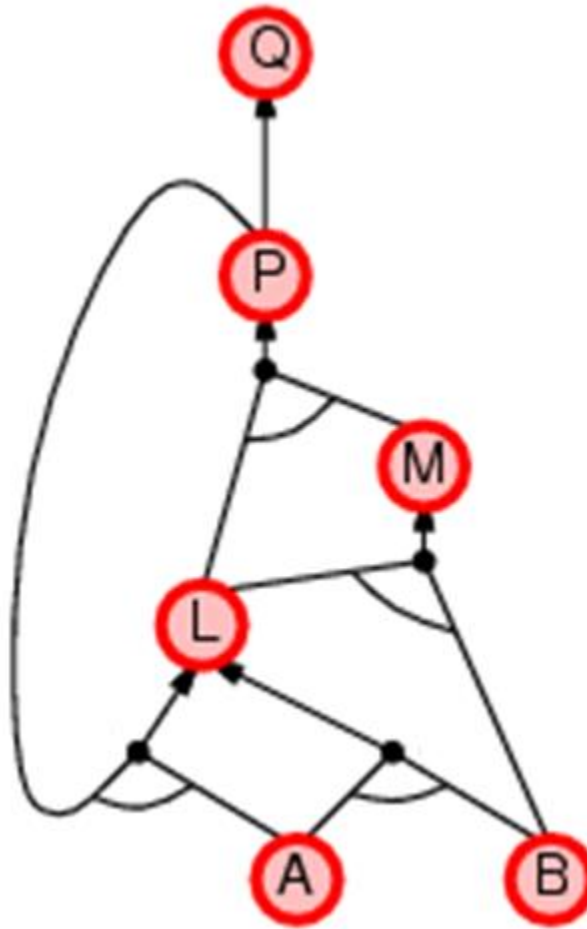
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$





# Suy diễn lùi (backward chaining)

- Ý tưởng: Quá trình suy diễn bắt đầu từ mệnh đề kết luận  $Q$
- Để chứng minh  $Q$  bằng tập mệnh đề (cơ sở tri thức)  $KB$ 
  - Kiểm tra xem  $Q$  đã được chứng minh (trong  $KB$ ) chưa,
  - Nếu chưa, tiếp tục **chứng minh tất cả các mệnh đề giả thiết của một luật nào đó (trong  $KB$ ) có mệnh đề kết luận là  $Q$**
- Tránh các vòng lặp
  - Kiểm tra xem các mệnh đề mới đã có trong danh sách các mệnh đề cần chứng minh chưa? – Nếu rồi, thì không bổ sung (lại) nữa!
- Tránh việc chứng minh lặp lại đối với 1 mệnh đề
  - Đã được chứng minh (trước đó) là đúng
  - Đã được chứng minh (trước đó) là không thể thỏa mãn được (sai) trong  $KB$

# Suy diễn lùi: Ví dụ (1)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

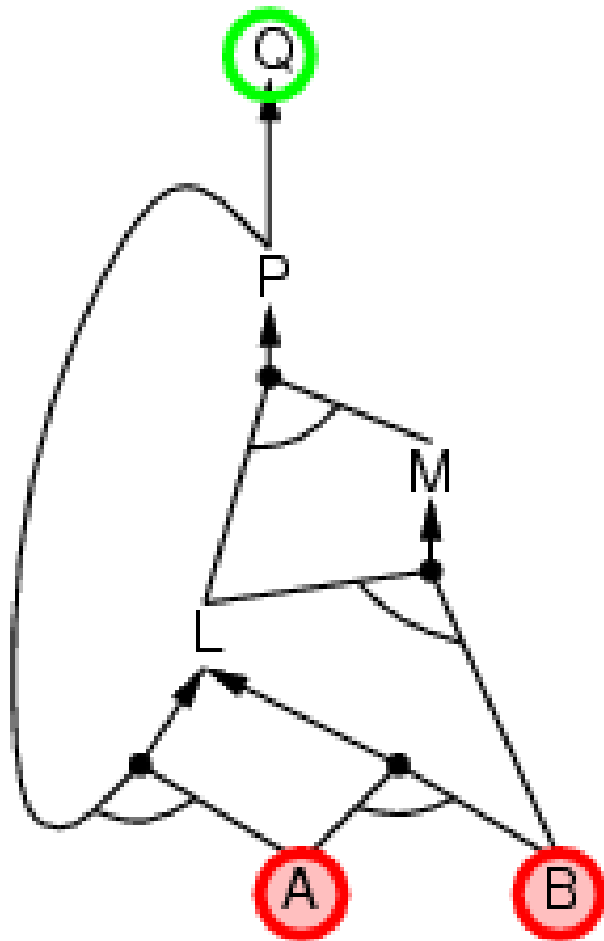
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn lùi: Ví dụ (2)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

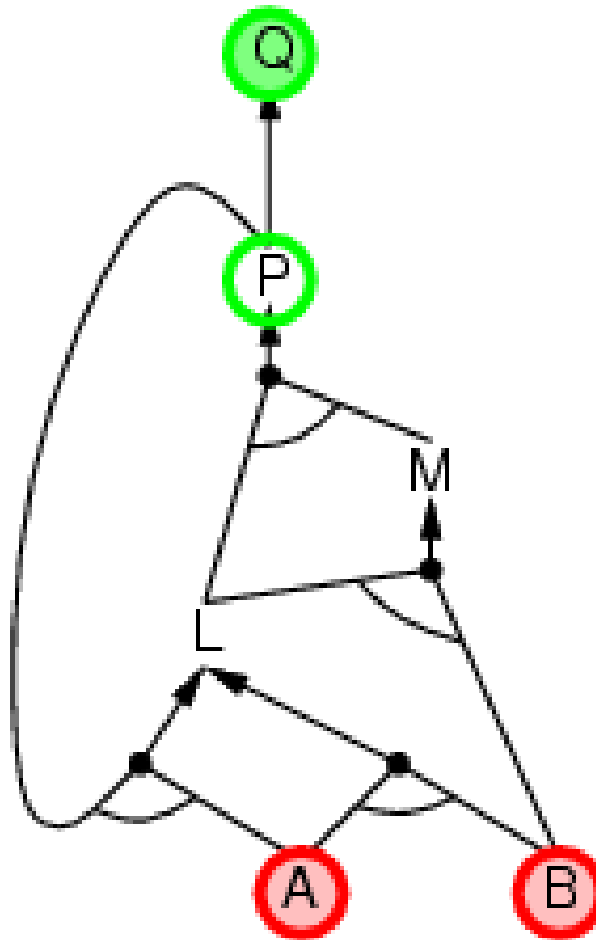
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn lùi: Ví dụ (3)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

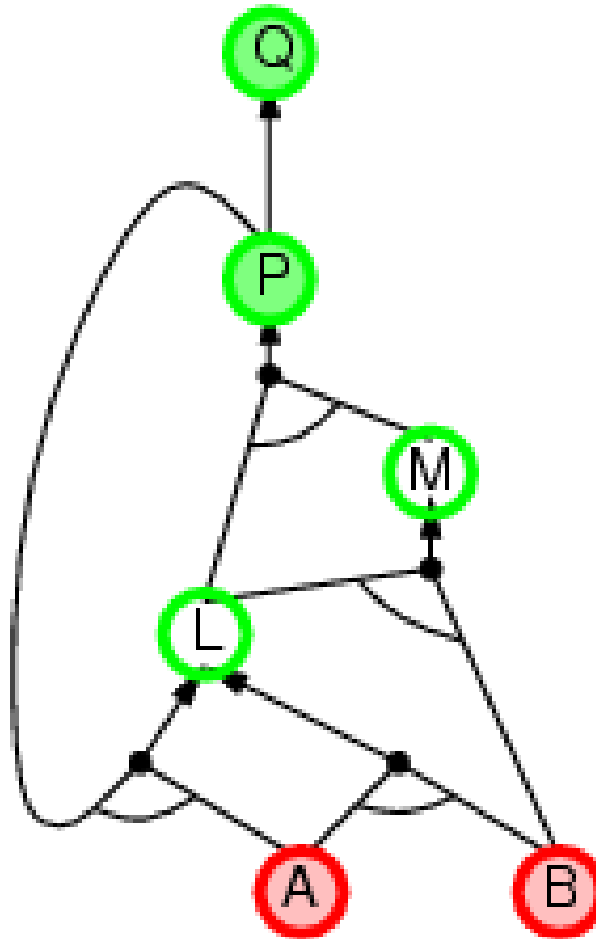
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn lùi: Ví dụ (4)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

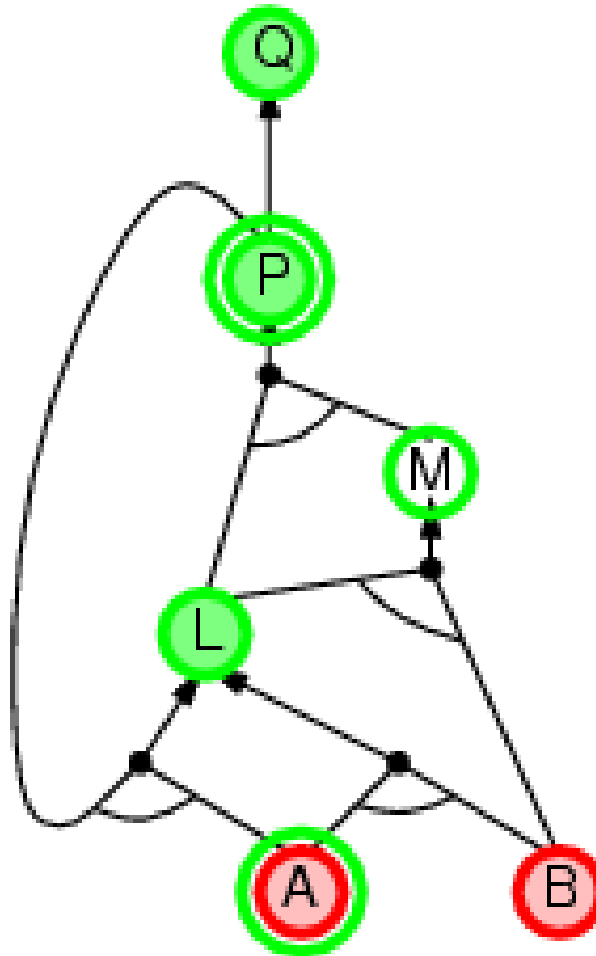
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn lùi: Ví dụ (5)

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

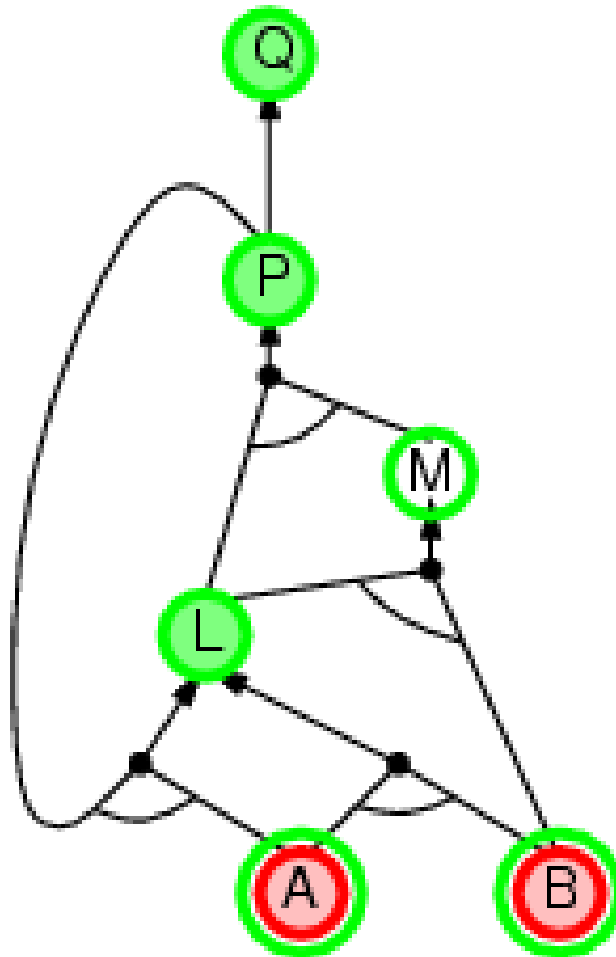
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

$A$

$B$



# Suy diễn tiến hay Suy diễn lùi?

- Suy diễn tiến là quá trình dựa trên dữ liệu (data-driven)
  - Ví dụ: việc nhận dạng đối tượng, việc đưa ra quyết định
- Suy diễn tiến có thể thực hiện nhiều bước suy diễn dư thừa – chẳng liên quan tới (cần thiết cho) mục tiêu cần chứng minh
- Suy diễn lùi là quá trình hướng tới mục tiêu (goal-driven), phù hợp cho việc giải quyết vấn đề

# Logic định đề: Ưu và nhược điểm

- (+) Logic định đề cho phép dễ dàng phát biểu (biểu diễn) cơ sở tri thức bằng tập các mệnh đề
- (+) Logic định đề cho phép làm việc với các thông tin ở dạng phủ định, dạng tuyển (disjunctive)
- (+) Logic định đề có tính cấu tạo (kết cấu)
  - Ngữ nghĩa của mệnh đề ( $S_1 \wedge S_2$ ) được suy ra từ ngữ nghĩa của  $S_1$  và ngữ nghĩa của  $S_2$
- (+) Ngữ nghĩa trong logic định đề không phụ thuộc ngữ cảnh (context-independent)
  - Không như trong ngôn ngữ tự nhiên (ngữ nghĩa phụ thuộc vào ngữ cảnh của các câu nói)
- (-) Khả năng diễn đạt (biểu diễn) của logic định đề là rất hạn chế
  - Logic định đề không thể diễn đạt được (như trong ngôn ngữ tự nhiên): “Nếu X là cha của Y, thì Y là con của X”
  - Logic định đề phải liệt kê (xét) mọi khả năng gán giá trị chân lý (đúng/sai) cho X và Y



# Giới hạn của Logic định đề

- Hãy xét ví dụ sau đây:
  - Tuấn là một sinh viên của HUST
  - Mọi sinh viên của HUST đều học môn Đại số
  - Vì Tuấn là một sinh viên của HUST, nên Tuấn học môn Đại số
- Trong logic định đề:
  - Định đề  $p$ : “Tuấn là một sinh viên của HUST”
  - Định đề  $q$ : “Mọi sinh viên của HUST đều học môn Đại số”
  - Định đề  $r$ : “Tuấn học môn Đại số”
  - Nhưng: (trong logic định đề)  $r$  không thể suy ra được từ  $p$  và  $q$ !