TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO Lab02.Logic

MSSV: 19120686
Họ tên: Trần Văn Tình
Ngày sinh: 19/03/2001

HỒ CHÍ MINH, ngày 18 tháng 12 năm 2021

I. Mô tả về thuật toán:

```
function PL-RESOLUTION(KB, α) returns true or false
  inputs: KB, the knowledge base, a sentence in propositional logic
          α, the query, a sentence in propositional logic

  clauses ← the set of clauses in the CNF representation of KB ∧ ¬α
  new ← { }
  loop do
    for each pair of clauses C_i, C_j in clauses do
      resolvents ← PL-RESOLVE(C_i, C_j)
      if resolvents contains the empty clause then return true
      new ← new ∪ resolvents
    if new ⊆ clauses then return false
    clauses ← clauses ∪ new
```

Figure 7.12 A simple resolution algorithm for propositional logic. The function PL-RESOLVE returns the set of all possible clauses obtained by resolving its two inputs.

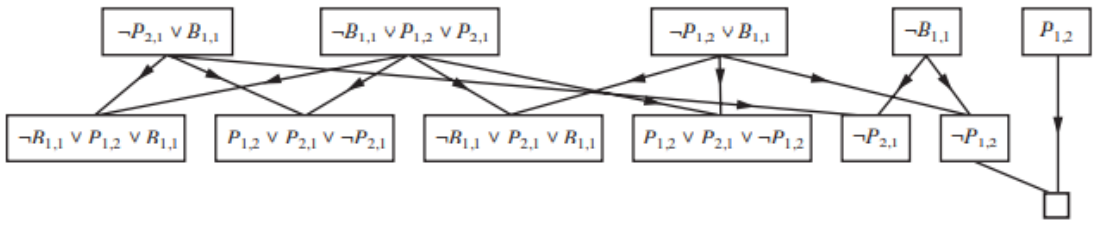

Figure 7.13 Partial application of PL-RESOLUTION to a simple inference in the wumpus world. $\neg P_{1,2}$ is shown to follow from the first four clauses in the top row.

Thuật toán được tham khảo theo Artificial Intelligence, A modern Approach (3rd edition)

Các hàm chính trong sourcecode đã nộp bao gồm:

PL_Resolve được cài đặt trong class Clause, và PL_Resolution là thuật toán chính

Cùng với các hàm bổ trợ như:

Check_clause (để kiểm tra một clause có phải là chân lí hay không)

Normalize nằm trong class Clause: Chuẩn hóa clause về tập hợp unique literal và theo thứ tự alphabet

II. Cách chạy chương trình:

+ Bước 1: nhập Input vào file Input.txt theo tiêu chuẩn của bài tập lưu ý file kết thúc ngay tại dòng cuối cùng (không nên có các khoảng trắng và \n sau những dòng cuối cùng trong file)
+ Bước 2: Chạy main.py và kết quả sẽ xuất ra file Output.txt

III. Kiểm thử

STT	Input	Output	Ghi chú (Theo VD notebook)	Nhận xét
1	K 4 A OR B OR C D OR K P -P	2 D {} Yes	(D OR K) AND (-K) (P) AND (-P)	Ta thấy mệnh đề cần chứng minh là **K**. Tuy nhiên giả thuyết sử dụng để chứng minh K lại là: P hợp giải với -P (không liên quan tới K) Điều này cho thấy tầm quan trọng của các cặp phủ định **(P,-P)** trong Input. Nếu đây là các cặp **"nhiễu"** ta thu thập **"nhầm"** thì lúc đó bất kì kết luận nào ta kiểm tra cũng cho kết quả **đúng**.
2	C 5 A OR B OR C -B C OR -C E OR -A -E	4 A OR C B OR C OR E A OR B -A 6 B OR C C OR E A B OR E C B 2 E {} Yes	(A OR B OR C) HG (-B) (A OR B OR C) HG (E OR -A) (A OR B OR C) HG (-C) (E OR -A) HG (-E) (A OR B OR C) HG (-A) (-B) HG (A OR C) (-C) HG (A OR C) (-C) HG (B OR C OR E) (A OR C) HG (-A) (A OR B) HG (-A) (-B) HG (B OR E) (-B) HG (B)	Thuật toán cần hợp giải rất nhiều Clause khác nhau. Tuy nhiên trên thực tế chỉ cần hợp giải những clause cần thiết là có thể đi đến kết luận
3	-A OR B 2 B OR C -C	2 B C 1 {} Yes	(B OR C) HG (-C) (B OR C) HG (-B) (-C) HG (C)	Với clause cần suy luận là -A OR B ta sẽ phủ định (-A OR B) Được hai clause mới là (A) AND (B) thêm vào knowledge base và thực hiện quá trình hợp giải đi đến kết luận
4	C 4 A OR B OR K -B D OR E OR -A F OR -E	3 A OR K B OR D OR E OR K -A OR D OR F 3 B OR D OR F OR K D OR E OR K D OR F OR K 0 NO	Không suy ra được C do không có mệnh đề rỗng.	Khuyết điểm:Thuật toán chạy nhiều vòng lặp, hợp giải nhiều clause khác nhau Trong khi KB ban đầu **không** tồn tại cặp đối kháng (P,-P) đồng thời cũng **không** có biến C trong đó nên rõ ràng không thể suy dẫn đến C được.
5	B OR C 4 -B OR C -C OR D OR E D OR F -F OR C	4 -B OR D OR E D OR E OR -F C OR D -F 2 D OR E D 0 NO	Không thể suy ra được (B OR C) do không có mệnh đề rỗng	Quá trình phủ định (B OR C) sẽ được hai clause mới là (-B) AND (-C) . Qua các

IV. Nhận xét chung về thuật toán:

Tích cực (Advantage):

+Thuật toán luôn cho ta kết quả liệu rằng mệnh đề alpha có thể chứng minh được từ knowledge base hay không

+Từ hình 7.13 ở trên ta thấy rằng thuật toán sẽ xét hết các mệnh đề có thể hợp giải làm cho bài toán xét được mọi trường hợp có thể xảy ra.

Hạn chế (Drawback):

+ Từ phần nhận xét trên II. Kiểm thử

Ta có một số kết luận như sau:

Thuật toán chứng minh mệnh đề bằng cách vét cạn dẫn đến dư thừa quá nhiều hợp giải không liên quan

Có thể bị "nhiễu" từ Input các cặp "nhiễu" này làm quá trình suy luận sai

(Bởi vì nếu tồn tại cặp đối kháng (P,-P) trong KB thì mệnh đề alpha lúc nào cũng đúng)

Chiến lược chứng minh mệnh đề alpha (Strategic Proof):

+ Ưu tiên chọn những hợp giải có cặp literal đối kháng hay những hợp giải cho kết quả clause ngắn hơn (vì kết quả muốn đạt được là kết quả {} rỗng)

+ Ưu tiên chọn những hợp giải có liên quan tới alpha.

+ Xem xét kĩ Input và Output những mệnh đề đối kháng (P,-P) đề phòng **"nhiễu"**

Reference:

1/ https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-825-techniques-in-artificial-intelligence-sma-5504-fall-2002/lecture-notes/Lecture7FinalPart1.pdf

2/ https://cs50.harvard.edu/ai/2020/notes/1/

3/ Artificial Intelligence, A modern Approach (3rd edition)

4/ 6.034 Artificial Intelligence. Copyright © 2004 by Massachusetts Institute of Technology

(Tài liệu logic trên Moodle môn học)