LÝ THUYẾT TÍNH TOÁN

BÀI 14: Quy dẫn

Phạm Xuân Cường
Khoa Công nghệ thông tin
cuongpx@tlu.edu.vn

Nội dung bài giảng

1. Giới thiệu

2. Các bài toán không quyết định được

3. Quy dẫn thông qua lịch sử tính toán

4. Bài toán PCP

5. Quy dẫn ánh xạ

Giới thiệu

Giới thiệu

- **Quy dẫn** là một kỹ thuật chứng minh sự không quyết định được của một ngôn ngữ
- Một **quy dẫn** là cách chuyển 1 bài toán (khó) thành bài toán khác (dễ hơn, có thể giải được)
- Có thể sử dụng lời giải của bài toán dễ để áp dụng cho bài toán khó
- Quy dẫn thường hay xuất hiện trong các bài toán về toán học
- Ví dụ:
  - Bài toán tìm đường đi trong một thành phố mới đến (khó) $\rightarrow$ Bài toán tìm bản đồ của thành phố đó (từ bản đồ $\rightarrow$ đường đi)
  - Bài toán tính diện tích hình chữ nhật $\rightarrow$ Bài toán đo chiều dài, chiều rộng

Logic ngược

- Quy dẫn: đưa một bài toán khó về một bài toán dễ hơn
- Nếu bài toán khó là không thể giải được $\rightarrow$ Bài toán dễ phải chắc chắn là không giải được
- Ví dụ:
  - Bài toán A: **Sống mãi mãi**
  - Bài toán B: **Trẻ mãi**
- Nếu ta tìm được lời giải cho bài toán **B** $\rightarrow$ Có thể giải được bài toán **A**
- Nhưng bài toán **A** là không thể xảy ra $\rightarrow$ Bài toán **B** cũng không thể xảy ra
- Tương tự trong LTTT, bài toán A là không quyết định được $\rightarrow$ bài toán B cũng không quyết định được

Logic

- Ta biết rằng A_{TM} là không quyết định được
- Xét bài toán P, P có quyết định được hay không?

Định lý 1

P là không quyết định được

Chứng minh

- Giả sử P là quyết định được
- Quy dẫn A_{TM} (Bài toán khó) về P (Bài toán dễ hơn)
- Sử dụng thuật toán quyết định P để giải A_{TM}
- Nhưng ta biết rằng không tồn tại bộ quyết định cho A_{TM}
  $\rightarrow$ Mâu thuẫn $\rightarrow$ P là không quyết định được

Các bài toán không quyết định được

Các bài toán không quyết định được

- **Bài toán dừng**: Kiểm tra xem một máy Turing có **dừng** trên một đầu vào w đã cho hay không

  $$\text{HALT}_{TM} = \{<\text{M, w}> \mid \text{M là một máy Turing và M dừng với đầu vào w}\}$$

- Vậy HALT_{TM} là quyết định được hay không? $\rightarrow$ **Không**

Bài toán dừng

Định lý 2

HALT_{TM} là không quyết định được

Chứng minh

Ý TƯỞNG:

- Giả sử HALT_{TM} là quyết định được
- Quy dẫn A_{TM} về HALT_{TM} $\rightarrow$ A_{TM} quyết định được
- Mâu thuẫn với định lý trong bài trước $\rightarrow$ Điều giả sử là sai $\rightarrow$ Vấn đề cốt lõi là làm sao để quy dẫn A_{TM} về HALT_{TM}

Bài toán dừng (2)

Chứng minh (Chi tiết)

Giả sử TM R quyết định HALT_{TM} $\rightarrow$ Xây dựng TM S quyết định A_{TM} như sau:

S với đầu vào là <M, w>

1. Chạy TM R trên đầu vào <M, w>
2. Nếu R bác bỏ thì bác bỏ
3. Nếu R chấp thuận, mô phỏng M trên w đến khi nó dừng
4. Nếu M chấp thuận w thì S chấp thuận, ngược lại S bác bỏ

Rõ ràng, R quyết định HALT_{TM} $\rightarrow$ S cũng phải quyết định A_{TM}

A_{TM} là không quyết định được $\rightarrow$ HALT_{TM} cũng không quyết định được

Bài toán kiểm tra rỗng

Định lý 3

E_{TM} = {<M> | M là một máy Turing và L(M)=Ø} là không quyết định được

Chứng minh (Tương tự $HALT_{TM}$)

- Giả sử máy Turing R quyết định E_{TM} → Sử dụng R để xây dựng máy Turing S quyết định A_{TM}
- S sẽ hoạt động như thế nào trên đầu vào <M, w>
- Nếu R chấp thuận xâu đầu vào <M> → L(M) = Ø → Bác bỏ w
- Nếu R bác bỏ xâu đầu vào <M> → L(M) ≠ Ø nhưng chưa chắc chấp thuận w → Chạy trên biến thể của M

Biến thể M_1 của M được mô tả như sau:

M_1 trên xâu đầu vào x:

1. Nếu x $\neq$ w thì kết luận là bác bỏ
2. Nếu x $=$ w thì chạy M trên đầu vào w và M_1 chấp thuận nếu M chấp thuận, ngược lại bác bỏ

Máy Turing S quyết định A_{TM}:

S= Trên đầu vào <M, w>:

1. Xây dựng M_1 từ M và w như trên
2. Chạy R trên xâu đầu vào M_1
3. Nếu R chấp thuận thì S bác bỏ, R bác bỏ thì S chấp thuận

Một số bài toán khác

Định lý 4

$\text{REGULLAR}_{TM} = \{<\text{M}> \mid$ M là một máy Turing và L(M) là ngôn ngữ chính quy$\}$ là không quyết định được

Định lý 5

$\text{EQ}_{TM} = \{<M_1, M_2> \mid M_1, M_2$ là máy Turing và $\text{L}(M_1) = \text{L}(M_2)\}$ là không quyết định được

Quy dẫn thông qua lịch sử tính toán

Quy dẫn thông qua lịch sử tính toán

- Lịch sử tính toán là một kỹ thuật quan trọng trong việc chứng minh A_{TM} có thể quy dẫn về ngôn ngữ nào đó
- Thường dùng để chứng minh bài toán kiểm tra sự tồn tại của một vấn đề
- Lịch sử tính toán C: abc$q_3$dac
- Lịch sử tính toán chấp thuận: C_1, C_2, ..., C_L với C_L là trạng thái chấp thuận
- Lịch sử tính toán bác bỏ: C_1, C_2, ..., C_L với C_L là trạng thái bác bỏ
- Nếu máy không dừng $\rightarrow$ Không có lịch sử tính toán

Ôtômat có biên tuyến tính

Định nghĩa

Ôtômat có biên tuyến tính (**Linear Bounded Automaton - LBA**) là một kiểu máy Turing có băng nhớ bằng đúng chuỗi đầu vào
$\rightarrow$ Đầu đọc không thể di chuyển ra ngoài đầu bên trái và phải của đoạn băng nhớ chứa chuỗi đầu vào

- Các bộ quyết định cho A_{DFA}, A_{CFG}, E_{DFA}, E_{CFG} đều là LBA
- Mọi ngôn ngữ phi ngữ cảnh CFL đều có thể quyết định được bởi một LBA

Bổ đề

Gọi M là một LBA có q trạng thái và g ký hiệu trong $\Sigma \rightarrow$ Có chính xác qng^n hình trạng phân biệt của M cho một băng chiều dài n

Định lý 6

$A_{LBA} = \{$<M, w> $|$ M là một LBA chấp thuận w$\}$ là quyết định được

Bài toán quyết định của LBA

Chứng minh

Ý Tưởng: Mô phỏng M trên w, nếu sau một số bước nhất định mà máy không dừng $\rightarrow$ Bác bỏ

Thuật toán quyết đinh A_{LBA} như sau:

L = Trên đầu vào <M, w>:

1. Mô phỏng M trên w cho qng^n bước cho tới khi nó dừng
2. Nếu M dừng, nếu M chấp thuận thì L chấp thuận, ngược lại bác bỏ.
   Nếu M không dừng thì bác bỏ

Định lý 7

$E_{LBA} = \{<M> \mid M \text{ là một LBA và } L(M) = \emptyset\}$ là không quyết định được

Chứng minh

Ý Tưởng: Quy dẫn về A_{TM}, nếu E_{LBA} quyết định được thì A_{TM} cũng quyết định được
Xây dựng một LBA B kiểm tra xem L(B) có rỗng hay không.
LBA B hoạt động như sau:

1. Nhận đầu vào là lịch sử tính toán của w trên M: $C_1\#C_2\#\ldots\#C_L \rightarrow$ Phân tách theo ký tự $\#$
2. Kiểm tra xem C_1 có đúng là cấu hình ban đầu của M và w không
3. Kiểm tra xem mỗi C_i có hợp lệ với C_1 không
4. Kiểm tra xem C_L có phải là cấu hình chấp thuận không

Chứng minh

Xây dựng máy Turing S quyết định A_{TM} như sau:

S = Trên đầu vào <M, w>

1. Xâu dựng LBA B từ M và w
2. Chạy R trên đầu vào <B>
3. Nếu R bác bỏ thì S chấp thuận, ngược lại thì S bác bỏ

Định lý 8

ALL_{CFG} là không quyết định được

Bài toán PCP

Bài toán PCP

- Bài toán PCP: Mô tả dưới dạng một trò chơi Domino
- Mỗi domino có dạng như sau:

$$\left[\frac{a}{ab}\right]$$

- Tập các domino có dạng:

$$\left\{\left[\frac{b}{ca}\right],\left[\frac{a}{ab}\right],\left[\frac{ca}{a}\right],\left[\frac{abc}{c}\right]\right\}$$

- Nhiệm vụ là tạo ra một danh sách các domino (cho phép lặp lại) sao cho xâu ở hàng trên giống hệt xâu ở hàng dưới $\rightarrow$ Danh sách đối xứng

$$\left\{\left[\frac{a}{ab}\right],\left[\frac{b}{ca}\right],\left[\frac{ca}{a}\right],\left[\frac{a}{ab}\right],\left[\frac{abc}{c}\right]\right\}$$

- Tồn tại một số tập domino không thể tìm được một đối xứng:

$$\left\{\left[\frac{abc}{ab}\right],\left[\frac{ca}{a}\right],\left[\frac{acc}{ba}\right]\right\}$$

Bài toán PCP

- Bài toán PCP là bài toán quyết định xem một tập các domino có đối xứng không
- Bài toán này **không** giải được bằng thuật toán
- Mô tả bài toán dưới dạng ngôn ngữ
  - Một thể hiện của PCP là một tập

$$P = \left\{ \left[\frac{t_1}{b_1}\right], \left[\frac{t_2}{b_2}\right], \ldots, \left[\frac{t_k}{b_k}\right] \right\}$$

  - Một sự đối xứng là một chuỗi $i_1, i_2, \ldots, i_l$ trong đó $t_{i_1} t_{i_2} \ldots t_{i_l} = b_{i_1} b_{i_2} \ldots b_{i_l}$

Bài toán PCP

PCP = {<P> | P là một thể hiện PCP có một đối xứng}

Chứng minh

Ý tưởng: Quy dẫn về A_{TM} thông qua các lịch sử tính toán chấp thuận $\rightarrow$ Chứng minh rằng $\forall$ TM M và đầu vào w ta có thể xây dựng một thể hiện P có một đối xứng là 1 lịch sử tính toán chấp thuận của M trên w

Để thuận tiện cho việc xây dựng P ta giả thiết:

- M chạy trên w không di chuyển đầu đọc quá ô cuối cùng bên trái trên băng
- Nếu w = $\varepsilon \rightarrow$ sử dụng xâu $\cup$ ở vị trí của w trong mô tả
- Đảm bảo rằng PCP một đối xứng sẽ bắt đầu với domino đầu tiên

PCP sửa đổi (MPCP-PCP)

PCP = {<P> | P là một thể hiện PCP có một đối xứng bắt đầu bằng domino đầu tiên}

Chứng minh

Gọi R là máy Turing quyết định PCP và xây dựng S quyết định A_{TM}

Đầu tiên S xâu dựng một thể hiện P' của MPCP như sau:

- Phần 1: Đặt $\left[\frac{\#}{\# q_0 w_1 w_2 \dots w_n \#}\right]$ vào P' như là domino đầu tiên
- Phần 2: Điều chỉnh đầu đọc ghi sang bên phải
  $\forall$ a,B $\in \Gamma$ và $\forall$ q, R $\in$ Q trong đó q $\neq q_{reject}$
  Nếu δ(q,a) = (r,b,R) thì đặt $\left[\frac{qa}{br}\right]$ vào P'
- Phần 3: Điều chỉnh đầu đọc ghi sang bên trái
  $\forall$ a, b, C $\in \Gamma$ và $\forall$ q, R $\in$ Q trong đó q $\neq q_{reject}$
  Nếu δ(q,a) = (r,b,L) thì đặt $\left[\frac{cqa}{rcb}\right]$ vào P'
- Phần 4: Điều khiển các ô không liền kề đối với đầu đọc ghi
  $\forall$ a $\in \Gamma$ đặt $\left[\frac{a}{a}\right]$ vào P'

Ví dụ

Cho $\Gamma = \{0, 1, 2, \sqcup\}$, w = 0100 và trạng thái đầu tiên của M là q_0, $\delta(q_0,0) = (q_7,2,R)$

- P1: Đưa domino $\left[\frac{\#}{\# q_0 0100 \#}\right] = \left[\frac{t_1}{b_1}\right]$ vào trong P'
- P2: Đưa domino $\left[\frac{q_0 0}{2 q_7}\right]$ vì $\delta(q_0,0) = (q_7,2,R)$
- P3: Bỏ qua do tại q_7 không đề cập tới dịch chuyển sang trái
- P4: Đưa các domino sau vào P': $\left[\frac{0}{0}\right], \left[\frac{1}{1}\right], \left[\frac{2}{2}\right], \left[\frac{\sqcup}{\sqcup}\right]$
- P5: Mở rộng match bằng cách đưa domino sau vào P': $\left[\frac{\#}{\#}\right]$ hoặc $\left[\frac{\#}{\sqcup \#}\right]$

Tiếp tục ta có $\delta(q_7,1) = (q_5,0,R)$ thì trong P' có domino
$\left[\frac{q_7 1}{0 q_5}\right], \left[\frac{\#}{\sqcup \#}\right]$
Tiếp tục ta có $\delta(q_5,0) = (q_9,2,L)$ thì trong P' có thể có domino
$\left[\frac{0 q_5 0}{q_9 0 2}\right], \left[\frac{1 q_5 0}{q_9 1 2}\right], \left[\frac{2 q_5 0}{q_9 2 2}\right], \left[\frac{\sqcup q_5 0}{q_9 \sqcup 2}\right]$

Ví dụ

- P6: $\forall$ A $\in \Gamma$, đặt $\left[\frac{aq_{accept}}{q_{accept}}\right]$ hoặc $\left[\frac{q_{accept}a}{q_{accept}}\right]$ vào P'
- P7: Ta thêm domino $\left[\frac{q_{accept}\#\#}{\#}\right]$ vào P' để hoàn thiện đối xứng

Quy dẫn ánh xạ

Quy dẫn ánh xạ

- Hàm f: $\Sigma^* \rightarrow \Sigma^*$ là một hàm tính toán được nếu tồn tại một TM trên $\forall$ w, dừng với đầu ra là f(w) trên băng
- Định nghĩa hình thức của quy dẫn ánh xạ

Định nghĩa

Ngôn ngữ A là quy dẫn ánh xạ (**mapping reducible**) sang ngôn ngữ B, ký hiệu $A \leq_m B$ nếu có một hàm f: $\Sigma^* \rightarrow \Sigma^*$, trong đó $\forall$ w, $w \in A \Leftrightarrow f(w) \in B$
Hàm f được gọi là **quy dẫn** từ A sang B

Quy dẫn ánh xạ

Định lý 9

Nếu $A \leq_m B$ và B quyết định được thì A cũng quyết định được

Chứng minh

Gọi M là bộ quyết định cho B, f là một quy dẫn từ A sang B. Bộ quyết định N cho A được mô tả như sau:

N = Trên đầu vào w:

1. Tính f(w)
2. Chạy M trên đầu vào f(w) và đầu ra chính là đầu ra của M

Hệ quả

Nếu $A \leq_m B$ và A không quyết định được thì B cũng không quyết định được

Quy dẫn ánh xạ

Định lý 10

Nếu A $\leq_m$ B và B nhận biết được bởi TM thì A cũng nhận biết được bởi TM

Chứng minh

Tương tự Định lý 9

Hệ quả

Nếu A $\leq_m$ B và A không nhận biết được bởi TM thì B cũng không nhận biết được bởi TM

Quy dẫn ánh xạ

Định lý 11

EQ_{TM} là không Turing-recognizable và cũng không là co-Turing-recognizable

Chứng minh

Đầu tiên chứng minh EQ_{TM} không là Turing-recognizable bằng cách quy dẫn A_{TM} về $\overline{EQ_{TM}}$. Hàm f thực hiện như sau:

F = Trên đầu vào <M, w>

1. Xây dựng hai máy TM M_1 và M_2 như sau:
   - M_1 trên đầu vào bất kỳ: Bác bỏ
   - M_2 trên đầu vào bất kỳ
     - Chạy M trên w
     - Nếu nó chấp thuận w thì M_2 chấp thuận
2. Đưa ra $<M_1, M_2>$

Quy dẫn ánh xạ

Chứng minh

Để chứng minh $\overline{\mathrm{EQ}_{TM}}$ không nhận biết được bởi TM ta quy dẫn A_{TM} về EQ_{TM}. Hàm G thực hiện như sau:

G = Trên đầu vào <M, w>

1. Xây dựng hai máy TM M_1 và M_2 như sau:
    - M_1 trên đầu vào bất kỳ: Chấp thuận
    - M_2 trên đầu vào bất kỳ
        - Chạy M trên w
        - Nếu nó chấp thuận w thì M_2 chấp thuận
2. Đưa ra $<M_1, M_2>$

Nội dung ôn thi cuối kỳ

- Nội dung các chương 2 $\rightarrow$ 5
- Cấu trúc đề thi 4 câu
- Hình thức thi **viết**, **không dùng tài liệu**
- Thời gian thi: 90 phút
- Tỷ lệ nội dung thi như sau:
  - Chương 1 + 2 + 3: 90%
  - Chương 4 + 5: 10%

Questions?