

Chương 3

Kỹ thuật giải quyết vấn đề

Lê Thanh Hương
Khoa CNTT – ĐHBKHN

1

3.1. Khoa học TTNT

- TTNT quan tâm đến việc tạo ra các đối tượng có thể...
 - Hành động đúng
 - trên cơ sở hoàn cảnh cụ thể và những thứ mà nó đã biết

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

2

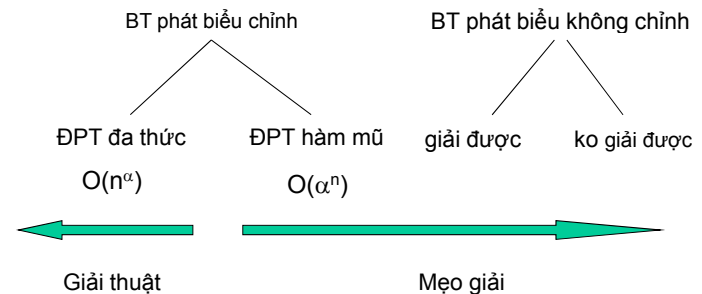
3.2. Phân loại vấn đề

- GQVĐ là quá trình xuất phát từ **hình trạng đầu**, tìm kiếm trong không gian bài toán để tìm ra **dãy toán tử** hay **dãy hành động** cho phép dẫn tới **đích**.
- **BT phát biểu chính**: là BT biết rõ đầu vào, đầu ra và với mỗi lời giải giả định nào đó, có thể áp dụng thuật toán để xác định xem đó có phải là lời giải của BT ban đầu hay không.
- **BT phát biểu không chính**: ngược lại

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

3

3.2. Phân loại vấn đề



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

4

Ví dụ 1. Bài toán đổ chữ

- Hãy thay các chữ cái bằng các chữ số từ 0 đến 9 sao cho không có hai chữ cái nào được thay bởi cùng 1 số và thỏa mãn ràng buộc sau:

$$\begin{array}{r} \text{SEND} \\ + \text{MORE} \\ \hline \text{MONEY} \end{array} \quad \begin{array}{r} \text{CROSS} \\ + \text{ROADS} \\ \hline \text{DANGER} \end{array}$$

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

5

Ví dụ 2. Bài toán rót nước

- Cho 2 bình A(m lít), B(n lít). Làm cách nào để đo được k lít ($k \leq \max(m,n)$) chỉ bằng 2 bình A, B và 1 bình trung gian C.

- Các thao tác rót (how):

$$C \rightarrow A; C \rightarrow B; A \rightarrow B; A \rightarrow C; B \rightarrow A; B \rightarrow C$$

- Điều kiện: không tràn, đổ hết

- Ví dụ: m = 5, n = 6, k = 2 (what)

- Mô hình toán học:

$$(x, y) \rightarrow (x', y')$$

$$A \ B \quad A \ B$$

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

6

Ví dụ 3. Bài toán trò chơi $n^2 - 1$ số

- Trong bảng ô vuông n hàng, n cột, mỗi ô chứa 1 số nằm trong phạm vi từ $1 \rightarrow n^2 - 1$ sao cho không có 2 ô có cùng giá trị. Còn đúng 1 ô bị trống. Xuất phát từ 1 cách sắp xếp nào đó của các số trong bảng, hãy dịch chuyển các ô trống sang phải, sang trái, lên trên, xuống dưới để đưa về bảng:

7	2	4
5		6
8	3	1

Start State

	1	2
3	4	5
6	7	8

Goal State

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

7

Ví dụ 4. Bài toán tháp Hà Nội

- Cho 3 cọc 1,2,3. Ở cọc 1 ban đầu có n đĩa, sắp theo thứ tự to dần từ trên xuống dưới. Hãy tìm cách chuyển n đĩa đó sang cọc 3 sao cho:

- Mỗi lần chỉ chuyển 1 đĩa
- Ở mỗi cọc không cho phép đĩa to nằm trên đĩa con



Bài toán tháp Hà Nội với n = 3

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

8

Ví dụ 5. Bài toán đồ: Quan tòa - Hề - Trộm

- Có 3 người ngồi quanh 1 bàn tròn. Một người qua đường nghe thấy ba người này nói chuyện với nhau:
 - người 1 nói 2 là quan tòa
 - người 2 nói 3 là hề
 - người 3 nói 1 là trộm
- Biết rằng:
 - hề luôn nói đùa
 - quan tòa nói thật
 - trộm nói dối
- Hỏi ai là ai?

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

9

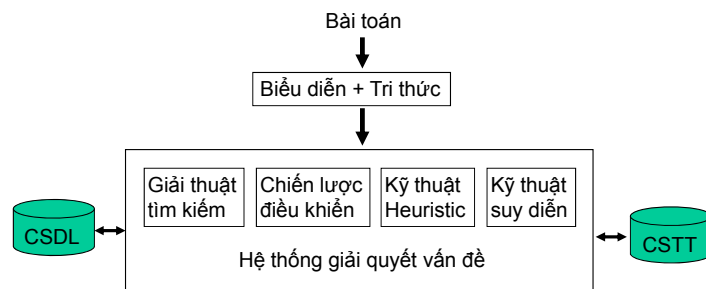
Các đặc trưng cơ bản của vấn đề

- Bài toán có thể phân rã?
- Không gian bài toán có thể đoán trước?
- Có tiêu chuẩn xác định lời giải tối ưu?
- Có cơ sở tri thức phi mâu thuẫn?
- Tri thức cần cho quá trình tìm kiếm hay để điều khiển?
- Có cần tương tác người – máy?

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

10

3.3. Những yếu tố cơ bản trong GQVĐ



Cấu trúc các hệ thống giải quyết vấn đề

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

11

3.4. Các phương pháp biểu diễn vấn đề

1 Biểu diễn nhờ KGTT

- Mỗi hình trạng của bài toán tương ứng với 1 **trạng thái** (state)
- Mỗi phép biến đổi từ hình trạng này sang hình trạng khác tương ứng với các **toán tử** (operator)

2 Quy bài toán về bài toán con

- Phân chia bài toán thành các bài toán con, các bài toán con lại được phân rã tiếp cho đến khi gặp được các bài toán sơ cấp cho phép xác định lời giải của bài toán ban đầu trên cơ sở lời giải của các bài toán con
- VD: phương pháp tìm đường từ bước trong công nghệ lập trình

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

12

3.4.Các phương pháp biểu diễn vấn đề

③ Sử dụng logic hình thức

Khi giải quyết bài toán, phải tiến hành phân tích logic để thu gọn quá trình tìm kiếm, nhiều khi chứng minh được không có lời giải.

- logic mệnh đề
- logic vị từ cấp 1

cho phép:

- kiểm tra điều kiện kết thúc trong tìm kiếm đối với KGTT
- kiểm tra tính áp dụng được của các toán tử
- Chứng minh không tồn tại lời giải
- Mục đích: CM 1 phát biểu nào đó trên cơ sở những tiền đề và luật suy diễn đã có.

3.4.Các phương pháp biểu diễn vấn đề

④ Lựa chọn phương pháp biểu diễn thích hợp nhằm:

- chia để trị
- tinh lọc thông tin
- tận dụng các phương pháp giải đã có
- phát biểu mới có thể thể hiện 1 vài tương quan nào đó giữa các yếu tố trong bài toán nhằm thu gọn quá trình giải

3.4.Các phương pháp biểu diễn vấn đề

⑤ Biểu diễn trong máy

- dùng bảng/mảng (array): ví dụ, trò chơi n^2-1 số

Trạng thái đầu

11	14	4	7
10	6		5
1	2	13	15
9	12	8	3

Trạng thái đích

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

$$A = (a_{ij}) = \begin{cases} 4(i-1) + j & (i, j) \neq (4, 4) \\ 0 & (i, j) = (4, 4) \end{cases}$$

3.4.Các phương pháp biểu diễn vấn đề

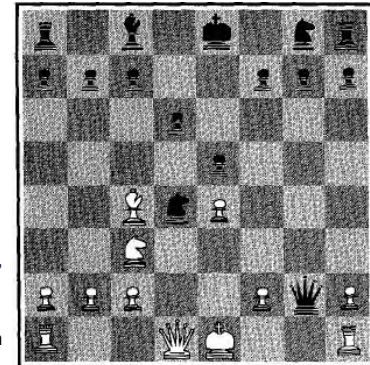
⑤ Biểu diễn trong máy

- dùng chuỗi ký hiệu

Ví dụ: bàn cờ Châu Âu

"T, XD, x, TgD, x, VD, x, MD, XD, ToD, ToD, ToD, x, x, ToD, ToD, ToD, x, x, x, ToD, x, x, x, x, x, x, ToD, x, x, x, x, x, x, TgT, MD, ToT, x, x, x, x, x, x, MT, x, x, x, x, x, x, ToT, ToT, ToT, x, x, ToT, HD, ToT, XT, x, x, HT, VT, x, x, XT."

x: ô trống, T: quân trắng đến lượt đi
XD: xe đen, TgD: tượng đen, VD: vua đen
MD: mã đen, ToD: tốt đen, HD: hậu đen
TgT: tượng trắng, ToT: tốt trắng, MT: mã trắng,
XT: xe trắng, HT: hậu trắng, VT: vua trắng



3.4. Các phương pháp biểu diễn vấn đề

❖ Biểu diễn trong máy

- dùng cấu trúc danh sách

Ví dụ: nghiệm của phương trình bậc 2

$$x_1 = \frac{-b + (b^2 - 4ac)^{1/2}}{2a}$$

Lê Thanh Hương – Khoa CNTT – ĐHBKHN

17

3.5. Giải quyết vấn đề

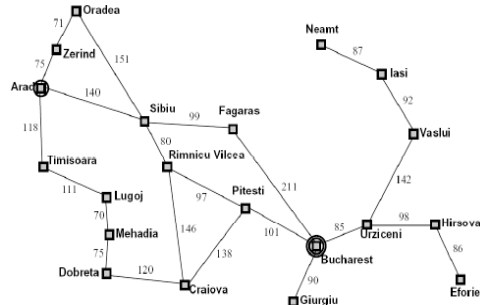
Để xây dựng các tác tử biết suy luận, ta cần sử dụng lý thuyết logic, xác suất, và tính hữu dụng. Các kỹ thuật tìm kiếm được nghiên cứu trước hết vì:

- Tìm kiếm là vấn đề quan trọng trong TTNT:
 - Tìm chuỗi hành động nhằm tối đa kết quả trong tương lai (lập kế hoạch)
 - Tìm kiếm trong CSTT để tìm chuỗi các hành động có thể thực hiện trong tương lai (suy luận logic, xác suất)
 - Tìm các mô hình phù hợp với các quan sát (trong học máy)
- Tìm kiếm là 1 trong những thành công của các nghiên cứu về TTNT giai đoạn đầu



Lê Thanh Hương – Khoa CNTT – ĐHBKHN

Bài toán tìm kiếm: Lập kế hoạch đường đi



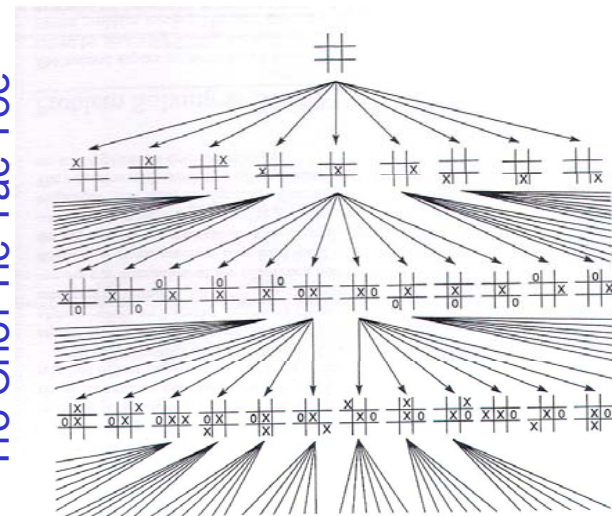
Kết quả: đi từ Arad đến Bucharest trong thời gian ngắn nhất

Môi trường: bản đồ với các thành phố, đường, và thời gian đi giữa 2 thành phố

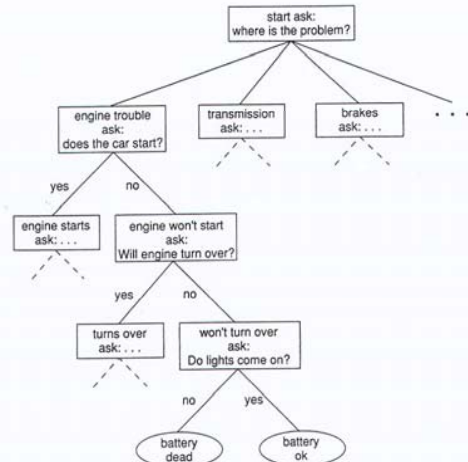
Lê Thanh Hương – Khoa CNTT – ĐHBKHN

19

KGTT của Trò Chơi Tic-Tac-Toe



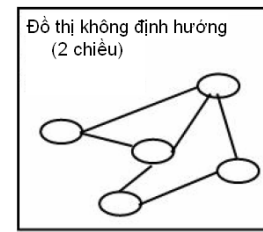
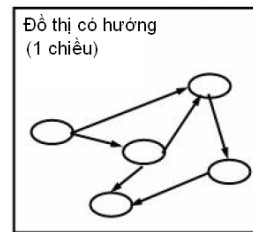
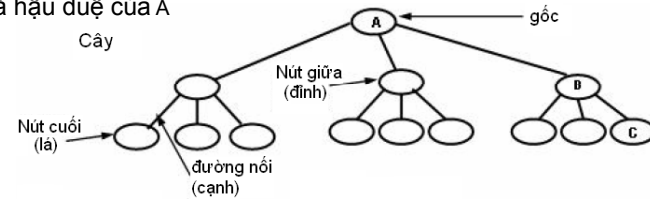
Chẩn đoán trục trặc máy móc trong ô tô



21

Cây và đồ thị

B là cha của C
C là con của B
A là tổ tiên của C
C là hậu duệ của A

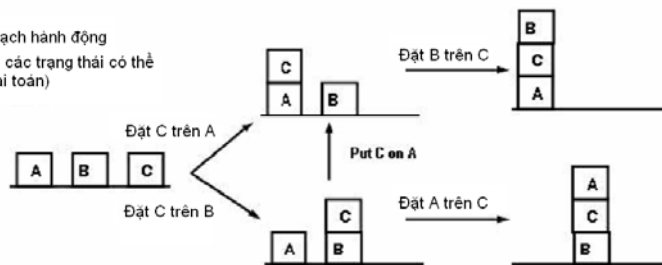


22

Ví dụ về đồ thị



Kế hoạch hành động
(đồ thị các trạng thái có thể
của bài toán)



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

3.5.1 Biểu diễn bài toán trong không gian tìm kiếm

Phát biểu bài toán P1:

- Cho trạng thái đầu s_0
- Cho tập trạng thái đích ĐÍCH
- Tìm dãy trạng thái s_0, s_1, \dots, s_n sao cho
 - $s_n \in \text{ĐÍCH}$ và
 - $\forall i: s_i \rightarrow s_{i+1}$ nhờ áp dụng toán tử biến đổi
- Giá đường đi: (cộng gộp)
 - ví dụ, tổng khoảng cách, số lượng hành động đã thực hiện, ...
 - $c(x, a, y)$ là giá 1 bước, ≥ 0
- Để biểu diễn phép biến đổi trạng thái, có 2 cách viết:
 1. Cách viết dùng luật sản xuất,
 - VD: $VT \rightarrow VP$
 - Bài toán Tháp Hà Nội
 2. Cách viết dùng ký hiệu hàm
 - VD: $B = f(A)$
 - Bài toán n^2-1 số

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

24

3.5.1 Biểu diễn bài toán trong KGTK

Phát biểu lại bài toán P1 (bài toán P2):

1. Tìm dãy trạng thái s_0, s_1, \dots, s_n sao cho
 - $s_n \in \text{ĐÍCH}$ và
 - $\forall i: s_i \rightarrow s_{i+1}$ (hay \exists toán tử biến đổi O : $O(s_i) = s_{i+1}$)
2. Tìm dãy toán tử O_1, \dots, O_{n-1}, O_n sao cho:

$$O_n(O_{n-1}(\dots O_1(s_0)\dots)) = s_n \in \text{ĐÍCH}$$
 hay tìm dãy sản xuất p_1, \dots, p_n sao cho

$$s_0 \Rightarrow^{p_1} s_1 \Rightarrow \dots \Rightarrow^{p_n} s_n \in \text{ĐÍCH}$$

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

25

Các chiến lược tìm kiếm lời giải

VD1: Bài toán rót nước

What: $A(m), B(n)$. Đầu: $(0,0)$. Đích $(k,*) \cup (*,k)$

How: Thao tác rót: $A \rightarrow B, \dots$

Điều kiện: không tràn, đổ hết

Biểu diễn sản xuất: $(x,y) \rightarrow (x', y')$

$m = 6, n = 5, k = 2$:

$6 - 5 = 1$

$2*6 - 2*5 = 2; \quad 4*5 - 3*6 = 2.$

USCLN(m,n)= d . Nếu k không chia hết cho $d \rightarrow$ not OK

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

26

Các chiến lược tìm kiếm lời giải

VD2: Bài toán Tháp Hà Nội, $n=3$

(i, j, k) Nếu i, j, k là 3 cọc riêng biệt
C B A $i + j + k = 6$

Procedure Thap(n,i,j : integer);

//nhắc n đĩa từ cọc i sang cọc j

Var k : integer;

Begin $k = 6 - i - j$;

if $n=1$ then Nhac(i,j)

else begin Thap($n-1,i,k$);

Nhac(i,j);

Thap($n-1,k,j$);

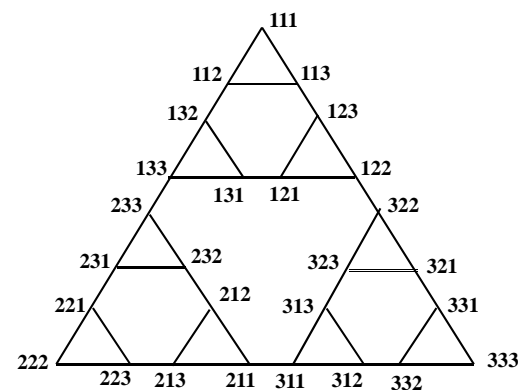
end;

End;

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

27

Không gian trạng thái của bài toán Tháp Hà Nội



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

28

Biểu diễn bằng đồ thị

Đồ thị G là cặp $G = (N, A)$ với N - tập các nút, A - tập các cung và với $\forall n \in N: \Gamma(n) = \{m \in N \mid (n, m) \in A\}$

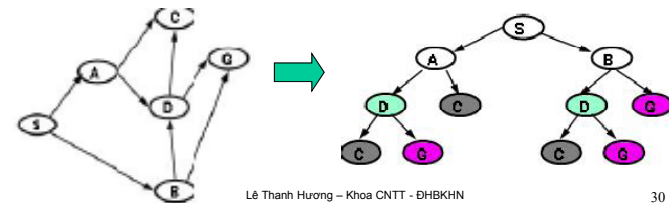
KGTT	Đồ thị
<ul style="list-style-type: none"> Trạng thái (đầu, đích) S, ababb Toán tử (sản xuất) $S \rightarrow Sa$ Dãy trạng thái liên tiếp Dãy toán tử $S \rightarrow Sa \rightarrow aBa$ Bài toán P_1, P_2 	<ul style="list-style-type: none"> nút (đầu, đích) cung đường đi Tìm đường đi trên đồ thị từ đỉnh đầu n_0 (tương ứng với s_0) tới đỉnh ĐÍCH

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

29

Chuyển bài toán tìm kiếm trên đồ thị về tìm kiếm trên cây

- Cây là đồ thị có hướng không có chu trình và các nút có ≤ 1 nút cha.
- Chuyển TK trên đồ thị về TK trên cây:
 - thay các liên kết không định hướng bằng 2 liên kết có hướng
 - tránh các vòng lặp trên đường (sử dụng biến tổng thể để lưu vết các nút đã thăm)



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

30

Các đặc tính tìm kiếm

- Tính đầy đủ**
 - Khi bài toán có lời giải thì giải thuật tìm kiếm có thể tìm thấy lời giải không?
- Thời gian**
 - Thời gian cần thiết để tìm thấy lời giải
- Không gian**
 - Dung lượng nhớ cần thiết để tìm thấy lời giải
- Sự tối ưu**
 - Khi có hàm giá, giải thuật có đảm bảo tìm được lời giải tối ưu không?

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

31

3.5.2 Các phương pháp tìm kiếm

Lớp	Tên	Thao tác
Bất kì 0 biết giá	TK sâu TK rộng	Khám phá có hệ thống toàn bộ cây đến khi tìm thấy đích
Tối ưu Biết giá	TK cực tiểu	Sử dụng độ đo là độ dài đường đi, tìm đường đi ngắn nhất
Tối ưu Biết giá	TK cực tiểu *	Sử dụng độ đo là độ dài đường đi và mệo, tìm đường đi ngắn nhất

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

32

Thuật toán tìm kiếm cơ bản

Xây dựng tập Mở - tập các đỉnh sắp duyệt

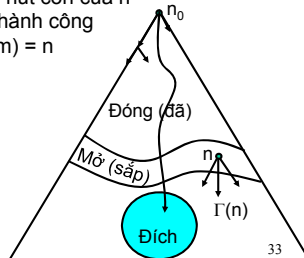
Đóng - tập các đỉnh đã duyệt

n_0 - trạng thái đầu

1. $Mở = \{n_0\}$; $Đóng = \emptyset$
2. Chọn $n \in Mở$:
 $Đóng = Đóng \cup \{n\}$
 $Mở = Mở \cup \Gamma(n)$ // $\Gamma(n)$: tập các nút con của n
3. Lặp (2) đến khi gặp $n^* \in Đích \Rightarrow$ thành công
4. Với mỗi $m \in \Gamma(n)$, thực hiện: $cha(m) = n$
 $p' = g, cha(g), cha^2(g), \dots, n_0$
 $p = inverse(p')$
 In đường đi

Các quyết định quan trọng:

- Lấy $n \in Mở$
- Bổ sung $\Gamma(n)$ vào Mở



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

33

Cài đặt các chiến lược tìm kiếm

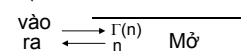
Các quyết định quan trọng:

- Lấy $n \in Mở$
- Bổ sung $\Gamma(n)$ vào Mở

- **Tìm kiếm sâu (Depth-first):**

Vào sau ra trước

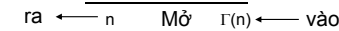
(LIFO – Last In First Out)



- **Tìm kiếm rộng (Breadth-first):**

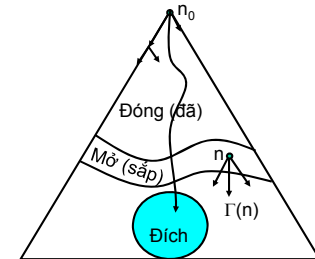
Vào trước ra trước

(FIFO – First In First Out)



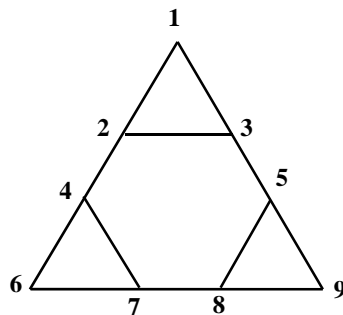
- **Tìm kiếm cực tiểu (Uniform-cost):**

Lấy phần tử có giá nhỏ nhất dựa trên hàm giá



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

34



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

35

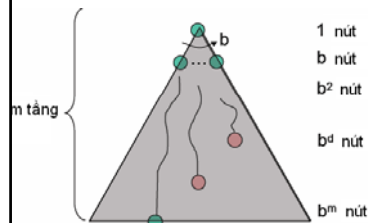
Tìm Kiếm Sâu hay Rộng?

- Có cần thiết tìm một *đường đi ngắn nhất* đến mục tiêu hay không?
- Sự *phân nhánh* của không gian trạng thái
- Tài nguyên về *không gian* và *thời gian* sẵn có
- *Khoảng cách trung bình* của đường dẫn đến trạng thái mục tiêu.
- Yêu cầu đưa ra *tất cả các lời giải* hay chỉ là lời giải tìm được đầu tiên.

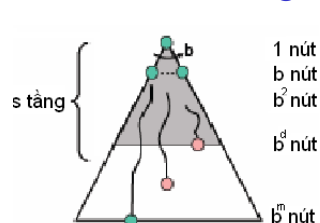
Lê Thanh Hương – Khoa CNTT - ĐHBKHN

36

Tìm kiếm sâu



Tìm kiếm rộng



Thuật toán	Hoàn thiện	Tối ưu	Thời gian	Không gian
TKS	không	không	$O(b^m)$	$O(bm)$
TKR	có	không	$O(b^{d+1})$	$O(b^{d+1})$

$$= 1 + b + b^2 + \dots + b^d + b(b^d - 1) = O(b^{d+1})$$

37

Tìm kiếm sâu dần

- TKS có thể cho kết quả nhưng đường đi không phải là ngắn nhất
- Tuy có \exists 1 đường đi đến Đích nhưng TKS có thể không dừng.

\Rightarrow chọn ngưỡng sâu D, mỗi đỉnh được gán một ngưỡng sâu $d(n)$

Lấy $n \in \text{Mở}$, $d(n) \leq D$

- Vấn đề
 - Nếu điểm đích n^* có $d(n^*) > D$?

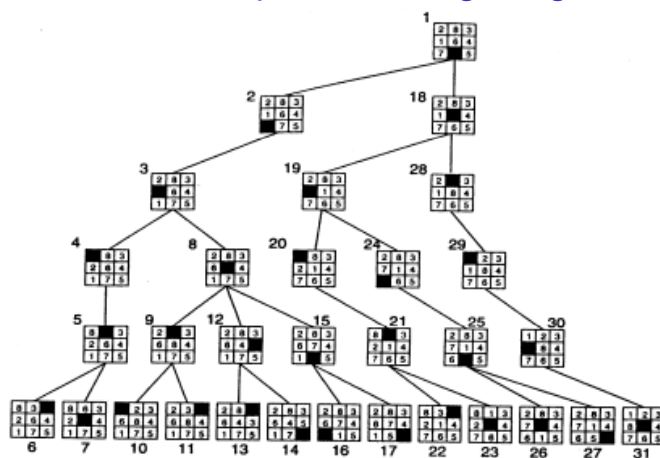
\Rightarrow Tìm kiếm sâu dần

Thuật toán	Hoàn thiện	Tối ưu	Thời gian	Không gian
TKSD	có	không	$O(b^d)$	$O(bm)$

Lê Thanh Hương – Khoa CNTT – ĐHBKHN

38

Trò chơi ô đố 8-puzzle với ngưỡng sâu 5



Lê Thanh Hương – Khoa CNTT – ĐHBKHN

Goal

Tìm kiếm cực tiểu

$c(n_i, n_j)$: chi phí đi từ n_i đến n_j

Xét $p = n_0, n_1, \dots, n_k$

Hàm đánh giá

$$c(p) = c(n_0, n_1) + c(n_1, n_2) + \dots + c(n_{k-1}, n_k)$$

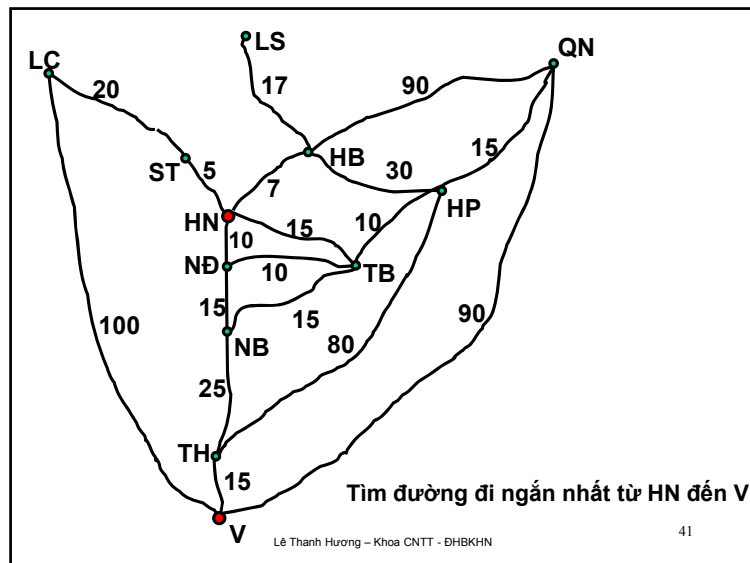
Lấy $n \in \text{Mở}$: $g(n) = c(p(n_0, n))$ min

Nếu $\forall c(n_i, n_j) > \epsilon$, C^* là chi phí của lời giải tối ưu

Thuật toán	Hoàn thiện	Tối ưu	Thời gian	Không gian
TKCT	có	có	$O(b^{\text{ceiling}(C^*/\epsilon)})$	$O(b^{\text{ceiling}(C^*/\epsilon)})$

Lê Thanh Hương – Khoa CNTT – ĐHBKHN

40



Tìm kiếm cực tiểu với tri thức bổ sung (A^*)

$c(n_i, n_j)$ = chi phí đi từ n_i đến n_j

$g(n)$ = chi phí thực tế đường đi từ n_0 đến n

$h(n)$ = chi phí ước lượng đường đi từ n đến đích, do chuyên gia cung cấp

- $h(n)$ chấp nhận được nếu với $\forall n, 0 \leq h(n) \leq h^*(n)$, trong đó $h^*(n)$ là chi phí thực để tới trạng thái đích từ n .
- $h(n)$ càng sát với $h^*(n)$ thì thuật toán càng mạnh

$$f(n) = g(n) + h(n)$$

$$f(n-1) = g(n-1) + h(n-1)$$

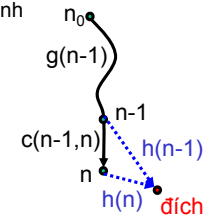
$$g(n) = g(n-1) + c(n-1, n)$$

$$f(n) = g(n-1) + c(n-1, n) + h(n)$$

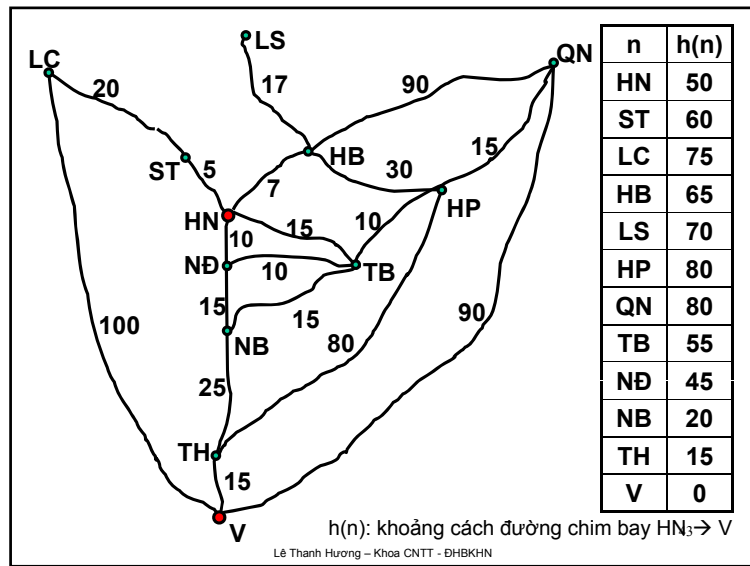
$$= f(n-1) - h(n-1) + c(n-1, n) + h(n)$$

Lấy $n \in \text{Mở}$: $f(n)$ min

Lê Thanh Hương – Khoa CNTT - ĐHBKHN



42



3.5.3 Một số dạng heuristic trong bài toán tìm kiếm

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

44

Bài toán đồ 8 số

5	4	
6	1	8
7	3	2

Start State

1	2	3
8		4
7	6	5

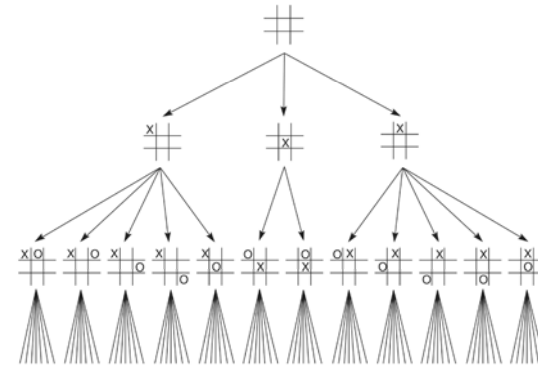
Goal State

- Ví dụ về heuristic
 - Số viên sai vị trí
 - Khoảng cách Manhattan. (Khoảng cách Manhattan giữa (x_1, y_1) và (x_2, y_2) là $|x_1 - x_2| + |y_1 - y_2|$.)
- $H1(S) = 7$
- $H2(S) = 2 + 3 + 3 + 2 + 4 + 2 + 0 + 2 = 18$

Lê Thanh Hương – Khoa CNTT – ĐHBKHN

45

Trò chơi Tic-tac-toe

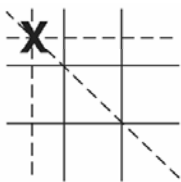


KGTT của tic-tac-toe được thu nhỏ nhờ tính đối xứng của các trạng thái

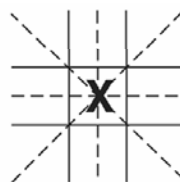
Lê Thanh Hương – Khoa CNTT – ĐHBKHN

46

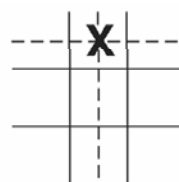
Phép đo heuristic



Chiếm 3 đường



Chiếm 4 đường



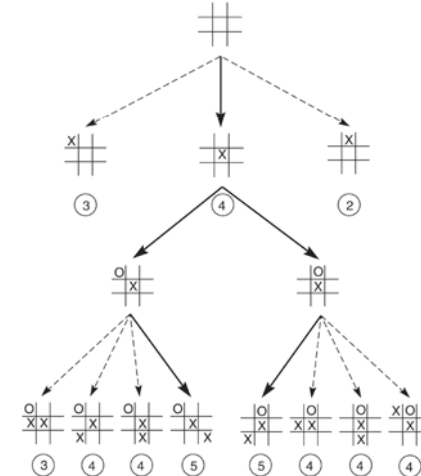
Chiếm 2 đường

Heuristic “Số đường thẳng nhiều nhất” áp dụng cho các nút con đầu tiên trong tic-tac-toe.

Lê Thanh Hương – Khoa CNTT – ĐHBKHN

47

Phép đo heuristic



48

Trò chơi đối kháng MINIMAX

Có 2 đối thủ MAX và MIN

- MAX tìm cách làm cực đại 1 hàm ước lượng nào đó: Chọn nước đi ứng với GTLN
- MIN tìm cách làm cực tiểu và chọn nước đi ứng với GTNN

Ở mỗi thời điểm:

- Nếu 1 đỉnh ứng với nước đi của MAX thì giá trị của nó là GT cực đại của các đỉnh con.
- Nếu 1 đỉnh ứng với nước đi của MIN thì giá trị của nó là GT cực tiểu của các đỉnh con.

Áp dụng vào chơi cờ caro trên bảng ô vuông (Tictactoe), kích thước 3x3. MAX đặt dấu x, MIN đặt dấu o. Ở mỗi nước đi, mỗi đối thủ xem trước 2 nước.

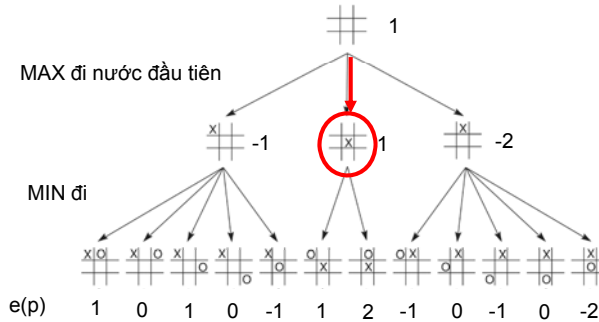
Ước lượng $e(p)$ đối với mỗi thế cờ p:

$E(p)$ = (số dòng, số cột, số đường chéo còn mở đối với MAX)
- (số dòng, số cột, số đường chéo còn mở đối với MIN)

- Nếu p là thế thắng đối với MAX, $e(p) = +\infty$
- Nếu p là thế thắng đối với MIN, $e(p) = -\infty$
- MAX đi mọi đường không có o; MIN đi mọi đường không có x

49

KGTT của tic-tac-toe được thu nhỏ nhờ tính đối xứng của các trạng thái

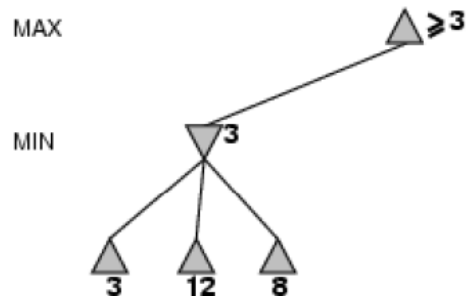


→ Tìm kiếm theo kiểu depth-first

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

50

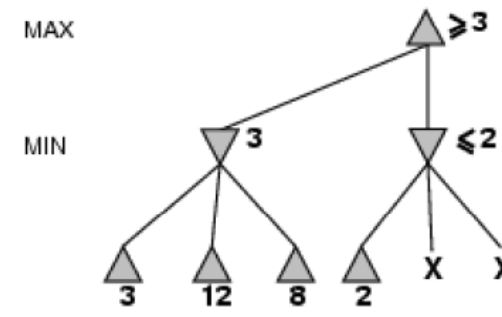
Phương pháp cắt tỉa α - β trong trò chơi minimax



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

51

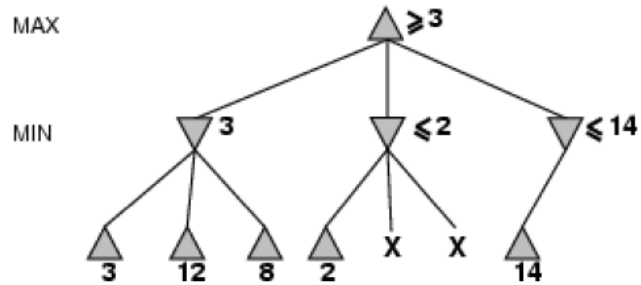
Phương pháp cắt tỉa α - β



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

52

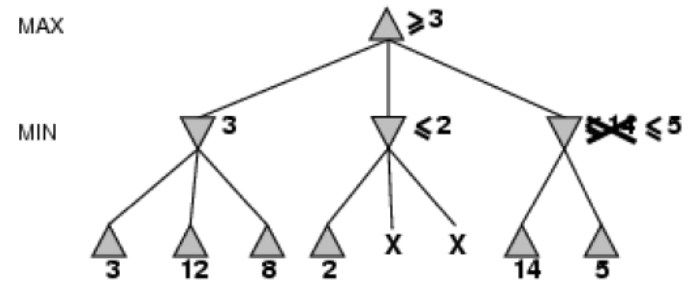
Phương pháp cắt tỉa α - β



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

53

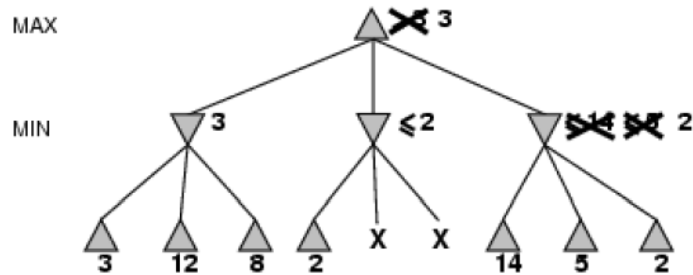
Phương pháp cắt tỉa α - β



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

54

Phương pháp cắt tỉa α - β



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

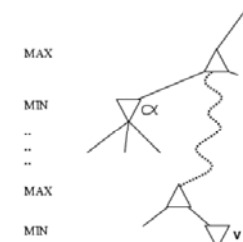
55

Phương pháp cắt tỉa α - β

- Cắt tỉa **không làm ảnh hưởng** tới kết quả cuối cùng
- Sắp xếp thứ tự duyệt tối ưu sẽ nâng cao hiệu quả của quá trình cắt tỉa
- Trong trường hợp tốt nhất, độ phức tạp thời gian = $O(b^{m/2})$

Tại sao gọi là α - β ?

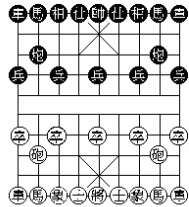
- α là giá trị của lựa chọn tốt nhất được tìm thấy ở thời điểm hiện tại trên đường đi của max
- Nếu v tồi hơn α , max sẽ không duyệt nó \rightarrow cắt tỉa nhánh đó
- Định nghĩa β tương tự đối với min



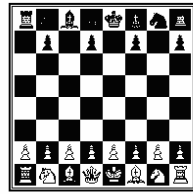
Lê Thanh Hương – Khoa CNTT - ĐHBKHN

56

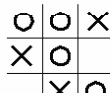
Một số trò chơi đối kháng (minimax)



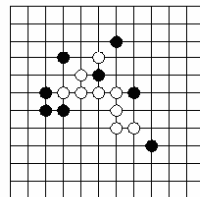
a) Cờ Tướng



b) Cờ Vua (cờ Quốc Tế)



c) Tic-tac-toe



d) Go-moku (cờ caro)

57

3.5.4 Tìm kiếm lời giải trên đồ thị Và/Hoặc

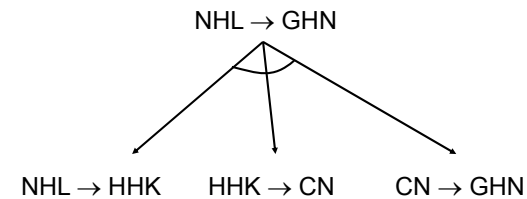
Phân rã bài toán thành bài toán con

VD1: Tìm đường đi từ Nhà hát lớn đến Ga Hà Nội.

BT1. Đi từ Nhà hát lớn đến Hồ Hoàn Kiếm

BT2. Đi từ Hồ Hoàn Kiếm đến Cửa Nam

BT3. Đi từ Cửa Nam đến Ga Hà Nội



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

58

Tìm kiếm lời giải trên đồ thị Và/Hoặc

VD2: Tháp Hà Nội $n = 3$

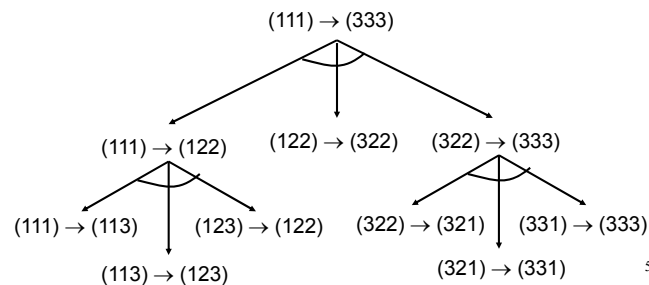
Bài toán đầu $(111) \rightarrow (333)$ được qui về 3 bài toán con:

BT1. $(111) \rightarrow (122)$: chuyển 2 đĩa AB từ cọc 1 sang cọc 2

BT2. $(122) \rightarrow (322)$: chuyển đĩa C từ cọc 1 sang cọc 3

BT3. $(322) \rightarrow (333)$: chuyển 2 đĩa AB từ cọc 2 sang cọc 3

BT2 giải được ngay, BT1 và BT3 tiếp tục phân rã



59

Qui bài toán về BT con

- Bài toán
- Qui về BT con
 - Phát biểu lại BT
 $P \rightarrow P_1, \dots, P_n$
 - Phân rã P thành
 P_1, \dots, P_n
- BT xuất phát
- BT sơ cấp (nguyên tử)
(\exists thuật giải để giải quyết)
- Giải BT P

Đồ thị V/H

- Đỉnh
- Cung
 - Cung Hoặc $P \rightarrow P_1, P_2, \dots, P_n$
 - Cung Và $P \rightarrow P_1, \dots, P_n$
- Đỉnh gốc n_0
- Đỉnh kết thúc
- XD đồ thị lời giải

Lê Thanh Hương – Khoa CNTT - ĐHBKHN

60

Đỉnh giải được

1. Đỉnh kết thúc (\Leftrightarrow bài toán sơ cấp) giải được

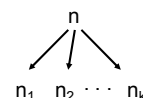
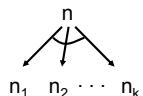
2. Giả sử n có con n_1, \dots, n_k

– $n_1, \dots, n_k \in N_V$

n giải được $\Leftrightarrow \forall n_i$ giải được

– $n_1, \dots, n_k \in N_H$

n giải được $\Leftrightarrow \exists n_i$ giải được



Lê Thanh Hương – Khoa CNTT – ĐHBKHN

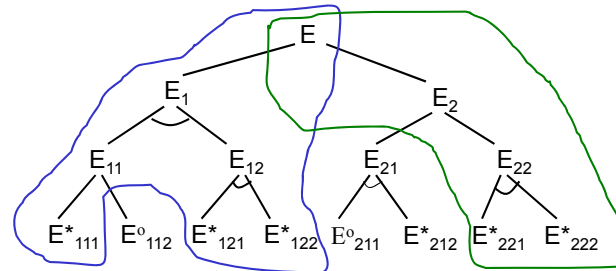
61

Cây lời giải T

• là cây con của đồ thị G

– $n_0 \in T$

– \forall đỉnh $n \in T$, n giải được



Lê Thanh Hương – Khoa CNTT – ĐHBKHN

62

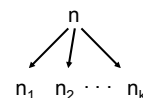
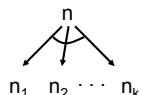
Đỉnh không giải được

1. n là lá ($\Gamma(n) = \emptyset$), n không kết thúc

2. n có con n_1, \dots, n_k

– $n_i \in N_V$, $n_{kgd} \Leftrightarrow \exists n_i$ không giải được

– $n_i \in N_H$, $n_{kgd} \Leftrightarrow \forall n_i$ không giải được



Qui ước: $n \in N$ là 1 bài toán nào đó

$nhan(n) = \text{true}$ nếu đỉnh n giải được

false nếu đỉnh n không giải được

kxd nếu đỉnh n không xác định

Với bài toán P , cần xác định $nhan(n_0)$, kéo theo đồ thị lời giải

Lê Thanh Hương – Khoa CNTT – ĐHBKHN

63

Thuật toán gán nhãn đỉnh giải được

procedure GD(n)

/* n gd hay $nhan(n) = \text{true}$ tùy thuộc vào thông tin gd của các đỉnh con của n */

{₁ if $kt(n)$ then $nhan(n) = \text{true}$

else {₂

if n có các đỉnh con là đỉnh VÀ n_1, \dots, n_k then

{₃ $gd(n_1); \dots; gd(n_k);$

if ($nhan(n_1)$ and ... and $nhan(n_k)$) then $nhan(n) = \text{true}$ }₃

if n có các đỉnh con là đỉnh HOẶC n_1, \dots, n_k then

{₄ $gd(n_1); \dots; gd(n_k);$

if ($nhan(n_1)$ or ... or $nhan(n_k)$) then $nhan(n) = \text{true}$ }₄

}₂

}₁

Lê Thanh Hương – Khoa CNTT – ĐHBKHN

64

Thuật toán gán nhãn đỉnh K giải được

procedure KGD(n)

```

{1 if not kt(n) then nhan(n)=false
  else {2
    if n có các đỉnh con là đỉnh VÀ  $n_1, \dots, n_k$  then
      {3 gd( $n_1$ ); ...; gd( $n_k$ );
        if(not nhan( $n_1$ ) or ... or not nhan( $n_k$ )) then nhan(n) = false }3
      if n có các đỉnh con là đỉnh HOẶC  $n_1, \dots, n_k$  then
        {4 gd( $n_1$ ); ...; gd( $n_k$ );
          if(not nhan( $n_1$ ) and ... and not nhan( $n_k$ )) then nhan(n) = false }4
        }2
  }1

```

Lê Thanh Hương – Khoa CNTT – ĐHBKHN

65

Thuật toán gán nhãn đỉnh giải được

procedure GD(n)

/ n gd hay nhan(n) = true tùy thuộc vào thông tin gd của các đỉnh con của n */*

```

{1 if kt(n) then nhan(n)=true
  else {2
    if n có các đỉnh con là đỉnh VÀ  $n_1, \dots, n_k$  then
      {3 gd( $n_1$ ); ...; gd( $n_k$ );
        if(nhan( $n_1$ ) and ... and nhan( $n_k$ )) then nhan(n) = true }3
      if n có các đỉnh con là đỉnh HOẶC  $n_1, \dots, n_k$  then
        {4 gd( $n_1$ ); ...; gd( $n_k$ );
          if(nhan( $n_1$ ) or ... or nhan( $n_k$ )) then nhan(n) = true }4
        }2
  }1

```

Lê Thanh Hương – Khoa CNTT – ĐHBKHN

66

TK rộng mới

Vào: Cây V/H $G=(N,A)$ với đỉnh đầu n_0

Ra: cây lời giải

PP: */* sử dụng 2 d/s queue MO, DONG */*

**TK sâu mới:
MO là stack**

```

{1 MO = { $n_0$ }; DONG =  $\emptyset$ ;
  while MO  $\neq \emptyset$  do {2
    n  $\leftarrow$  get(MO); DONG  $\leftarrow$  DONG  $\cup$  {n};
    if  $\Gamma(n) \neq \emptyset$  then {3 MO  $\leftarrow$  MO  $\cup$   $\Gamma(n$ );
      if trong  $\Gamma(n)$  có đỉnh m kết thúc then {4
        nhan(m) = true; gd( $n_0$ );
        if nhan( $n_0$ ) then exit('thanh cong')
      }4
      else Loại khỏi MO các đỉnh có tổ tiên là đỉnh giải được
    }3
  }2
  else {5 nhan(n) = false; kgd( $n_0$ );
    if not nhan( $n_0$ ) then exit('không thanh cong')
    else Loại khỏi MO các đỉnh có tổ tiên là đỉnh không giải được }5 }1

```

Lê Thanh Hương – Khoa CNTT – ĐHBKHN

67

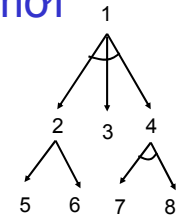
Tìm kiếm cực tiểu mới

Đặt vấn đề: Tìm cây T^* để chi phí $C(T^*)$ min

Thực tiễn đưa ra 2 mô hình về giá cây T

• Giá tổng cộng: $C_{\Sigma}(T) = \sum_{a \in T} c(a)$

• Giá max: $C_{\max}(T) = \max_{p: n_0 \rightarrow \text{leaves}} (c(p))$



Giá tối ưu để giải bài toán là $h(n)$, $h(n)$ có các tính chất sau:

• Nếu n là đỉnh kết thúc, $h(n) = 0$

• Nếu n có con là n_1, \dots, n_k

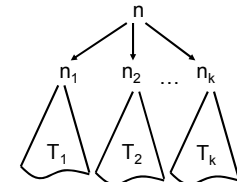
• $\forall n_i \in N_H: h(n) = \min[h(n_i) + c(n, n_i)]$

• $\forall n_i \in N_V:$

– Giá Σ : $h(n) = [\Sigma h(n_i) + \Sigma c(n, n_i)]$

– Giá max: $h(n) = \max[h(n_i) + c(n, n_i)]$

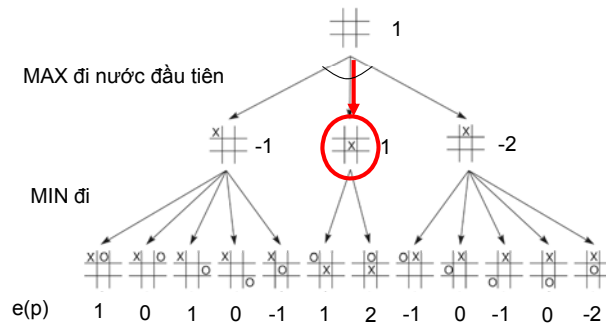
• Nếu n là đỉnh không giải được, $h(n) = \infty$



Lê Thanh Hương – Khoa CNTT – ĐHBKHN

68

KGTT của tic-tac-toe



Lê Thanh Hương – Khoa CNTT – ĐHBKHN

69

Tìm kiếm cực tiểu mới

- Trong quá trình tìm kiếm, tại mỗi bước có **1 tập các cây con gốc n_0** sao cho chúng có thể thành phần trên của cây lời giải cuối - cây lời giải tiềm tàng T_0 .
- Xây dựng cây T_0 dựa theo h'

Lê Thanh Hương – Khoa CNTT – ĐHBKHN

70

Tìm kiếm cực tiểu mới

Cách xác định cây lời giải tiềm tàng T_0

- Đỉnh đầu $n_0 \in T_0$
- Nếu $n \in T_0$ có các đỉnh con $n_1 \rightarrow n_k$ là:
 - đỉnh HOẶC:
 - chọn $n_i: c(n, n_i) + h'(n_i) \min$, $\text{nhan}(n_i) \neq \text{kgd}$
 - đỉnh VÀ:
 - chọn n_1, \dots, n_k vào T_0 nếu $\forall n_i: \text{nhan}(n_i) \neq \text{kgd}$

Nhận xét:

- Nếu cây V/H chỉ chứa đỉnh hoặc \rightarrow TKCT
- Nếu $c = 1$ và $h' = 0$ với \forall nút và sử dụng giá max \rightarrow TKRM
- Nếu $h'(n) \leq h(n)$ với $\forall n$ và $\exists \delta: \forall a \in A, c(a) \geq \delta$ thì TKCTM dừng và cho kết quả là cây lời giải tối ưu

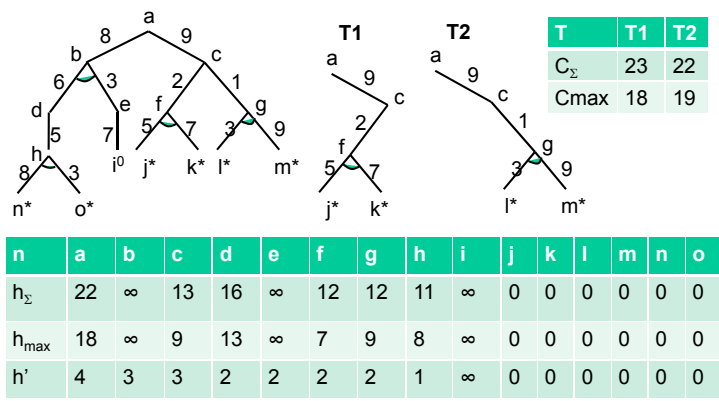
Lê Thanh Hương – Khoa CNTT – ĐHBKHN

TKCT mới

- $MO = \{n_0\}$; $DONG = \emptyset$; $T_0 = n_0$
- Chọn $n \in MO \cap$ lá (T_0):
 - $DONG \leftarrow DONG \cup \{n\}$;
 - if $kt(n)$ then { $\text{nhan}(n) = \text{true}$; $gd(n_0)$;
 - if $\text{nhan}(n_0)$ then exit('thanh cong')
 - else Loại khỏi MO các đỉnh có tổ tiên là đỉnh giải được
 - } else { // n không kết thúc
 - if $\Gamma(n) \neq \emptyset$ then { $MO \leftarrow MO \cup \Gamma(n)$;
 - Với mỗi $m \in \Gamma(n)$, tính $h'(m)$**
 - Với mỗi $m \in MO \cup DONG$, tính $h'(m)$**
 - else { // n không kết thúc và không có con
 - $\text{nhan}(n) = \text{false}$; $kgd(n_0)$;
 - if not $\text{nhan}(n_0)$ then exit('không thanh cong')
 - else Loại khỏi MO các đỉnh có tổ tiên là đỉnh không giải được
 - }
 - }
- Lặp lại 2 đến khi $MO = \emptyset$

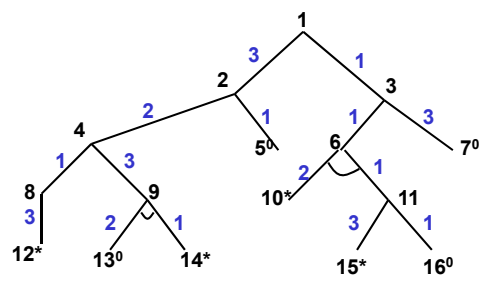
72

TKCT mới



Lê Thanh Hương – Khoa CNTT - ĐHBKHN

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
h'	4	3	3	2	∞	2	∞	1	1	0	1	0	∞	0	0	∞



Lê Thanh Hương – Khoa CNTT - ĐHBKHN