HEURISTIC SEAKCH

Date: ....../......./.......

⊛ Mở rộng khái niệm Thuật giải toán: Thuật giải

- Tính xác định (Đơn định)
- Tính Dừng (Hữu hạn)
- Tính đúng đắn (Kết quả) – Tính hiệu quả
  - Tính phổ dụng
  - I/O

⊛ Định nghĩa Heuristic 2 đặc tính sau:

- Nó thường tìm được lời giải tốt mặc dù không phải là tốt nhất.
- Nó thực hiện nhanh và dễ hơn bất kỳ một giải thuật tối ưu.

⊛ Một cách tiếp cận chung để thiết kế 1 Heuristic

- Liệt kê tất cả các y/c cầu ~~chung~~ của một giải thuật chính xác và phân chia chúng thành 2 lớp.
- Chẳng hạn:
  + Nhg y/c cầu dễ dàng thoả mãn. (phải thoả mãn)
  + Nhg y/c cầu k/o dễ dàng thoả mãn (có thể sẵn lòng thoả hiệp)

⊛ Greedy Traveling Salesman

(Bài toán người bán hàng)

1. Phát biểu bài toán

_ Cho N thành phố, trong đó hai thành phố bất kỳ đều có nối với nhau.

_ Một n xuất phát tại 1 thành phố, đi qua tất cả các tp còn lại 1 lần duy nhất và trở về thành phố xuất phát

_ Hãy xác định lộ trình sao cho tổng chi phí là MIN

⊛ ~~Thuật giải GTS1~~ 2. Thuật giải GTS1

_ Ý tưởng: Có đạt được lời giải tốt nhất ở mỗi bước thực hiện bằng cách chọn đường đi có chi phí thấp nhất tại đường đi hiện tại và tiếp tục đi.

→ Không tối ưu, tối ưu tại thời đ' chọn đ' mới ≠ tổng quát

→ K° đảm bảo đc lộ trình ngắn nhất

3. Thuật giải GTS2 Meta

_ Các giá trị Input: N, P, ma trận chi phí C và P thành phố khởi tạo: $\{V_1, V_2, \ldots, V_p\}$

_ Thuật giải:

B1: { Khởi tạo }

$K := 0$, $Best := \emptyset$, $Cost := \infty$

B2: { Bắt đầu 1 chu trình mới }

Chuyển qua B3 khi $K \leq p$, ngược lại STOP.

B3: { Tạo ra chu trình mới }

Đặt $K := K+1$. Call GST1 (V_K) Trả về: chu trình $T(K)$ với chi phí $C(K)$.

B4: { Cập nhật chu trình tốt nhất }

Nếu $C(K) < Cost$ thì: $BEST := T(K)$ và $Cost := T(K)$

→ bắt đầu từ p thành phố riêng biệt.

⊛ Hill-Climbing Search

(Thuật giải leo đồi)

_ Trong tìm kiếm ~~leo đồi~~ trên đồ thị, chiến lược của việc thử đạt tới đích bằng cách lựa chọn đỉnh mà được dự đoán trước là gần tới đích nhất thì được gọi là Hill-climbing

_ Đặc biệt là thử lựa chọn lựa đỉnh sau n. Tính $\hat{h}(n_i)$ các đỉnh con: $(n_1, n_2, \dots, n_m)$ của n. Sau đó chọn đỉnh có $\hat{h}(n_i)$ là nhỏ nhất làm đỉnh kế.

_ Thuật giải Hill-Climbing:

Key points

Date: ....../....../......

$n :=$ Start node

Loop: If Goal(n) Then Exit (Success)

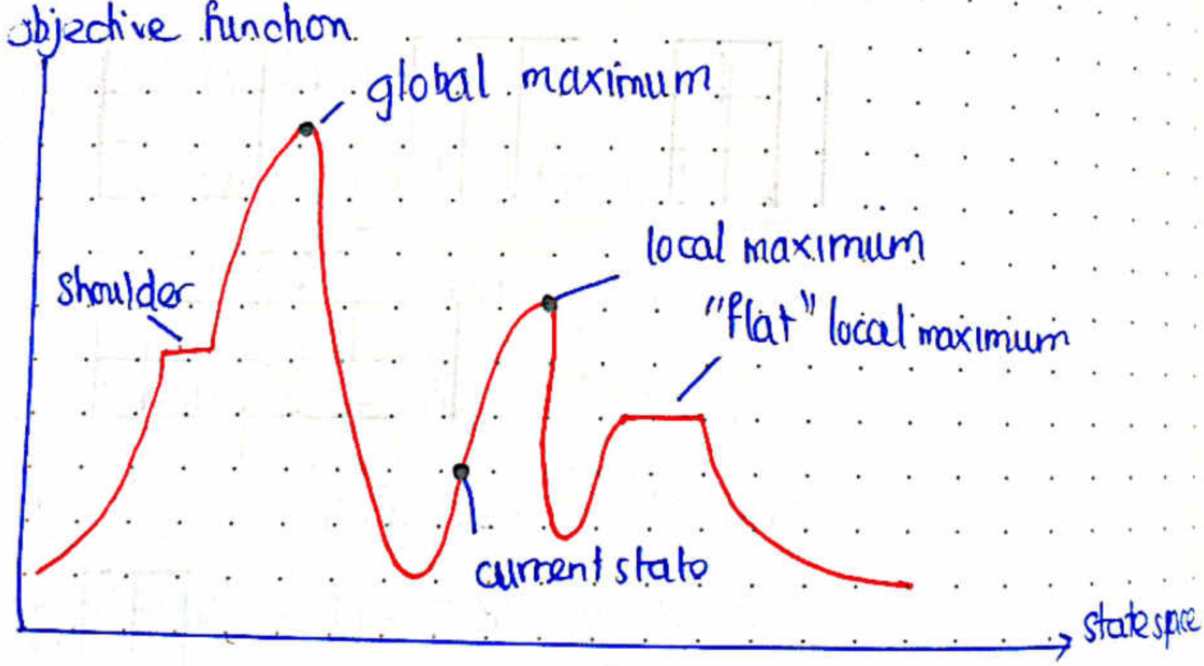
Expand n. Compute $\hat{h}(n_i)$ for all Child node n_i and take the child node which gives minimum value as next n.

If $\hat{h}(n_i) \in < \hat{h}(\text{next } n)$ Then Exit (Fail)

$n :=$ next n

Goto Loop

_ Vấn đề của Leo đồi:



* Hàm Heuristic

$g(A) = g(s) + g(S \to A) \quad g(s) = 0$

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

$f(D) = g(A) + g(D) \to \min \begin{cases} f(A) \\ f(D) \end{cases}$

ary

⊛ Bài toán N-Puzzle

Date: ....../....../......

- N-puzzle là 1 bảng gỗ kích thước N×N với N-1 ô số và 1 ô trống. Ô số nằm cạnh ô trống có thể trượt đến ô trống.

- Mục tiêu là đẩy các ô số sao cho đạt được trạng thái đích mong muốn.

VD: $h_2(a, b) = \sum_{i=1}^{8} \eta(a_i, b_i)$ (Chuyển ô 1 lần tính)

$\eta(a_i, b_i) = \sum_{i=1}$ (số lần ít nhất để ~~đt~~ di chuyển ô a_i ở (a) theo c ngang/dọc để về đúng vị trí b_i ở (b)

2	8	3
1	6	4
7		5

1	2	3
8		4
7	6	5

2	8	3
1	6	4
7		5

g = 0
h = 5, f = ~~6~~5

2	8	3
1	6	4
	7	5

h = 6, f = 7

2	8	3
1		4
7	6	5

h = 4, f = 5

2	8	3
1	6	4
7	5	

h = 6, f = 7

g = 1

2	8	3
	1	4
7	6	5

h = ~~4~~5, f = 7

2		3
1	8	4
7	6	5

h = 3, f = 5

2	8	3
1	4	
7	6	5

h = 5, f = 7

g = 2

	2	3
1	8	4
7	6	5

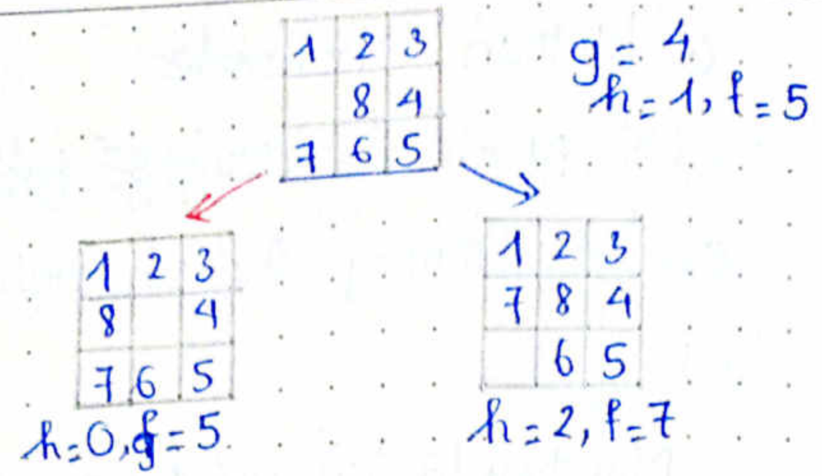
h = 2, f = 5

2	3	
1	8	4
7	6	5

h = 4, f = 7

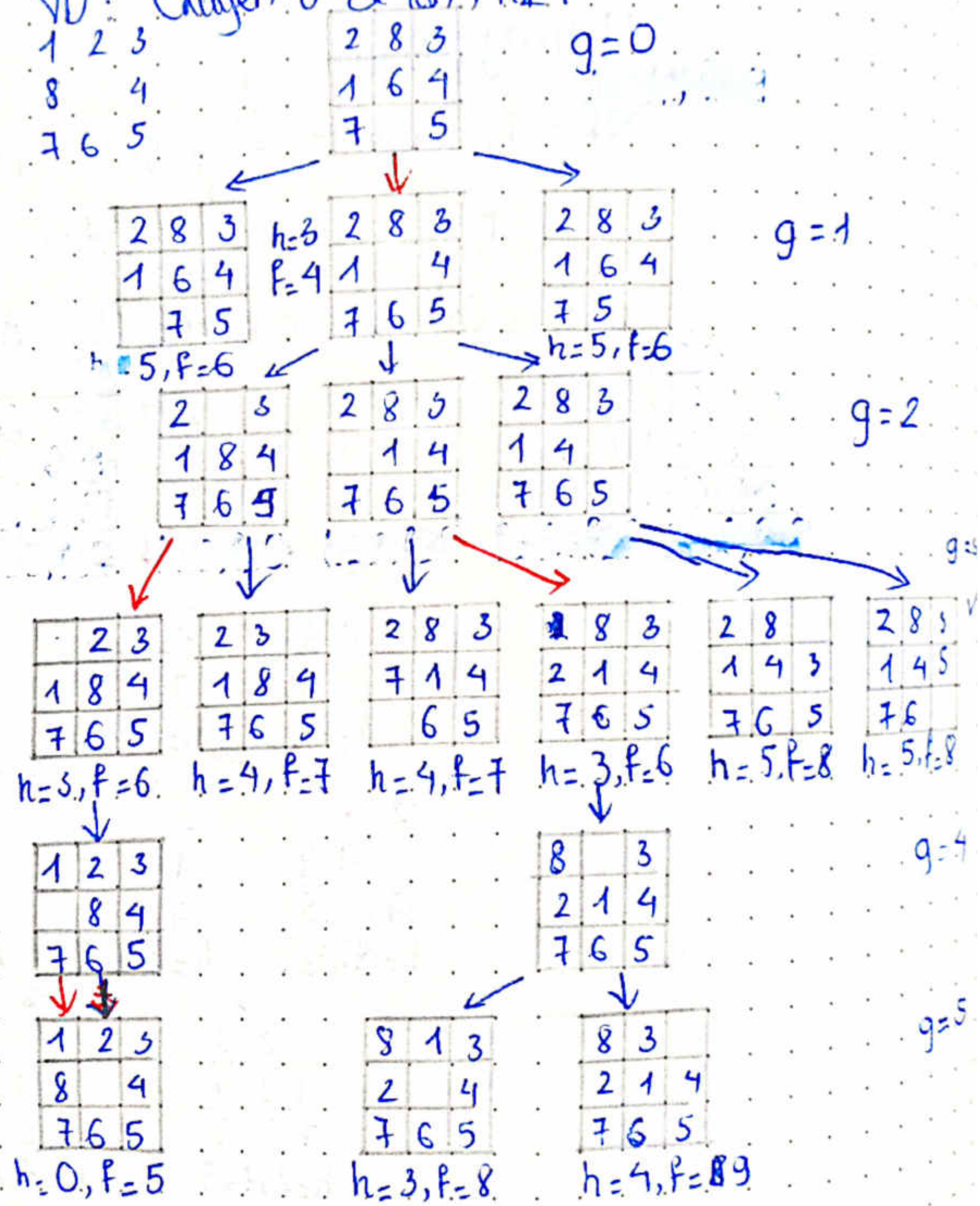
g = 3

Date: ....../....../......

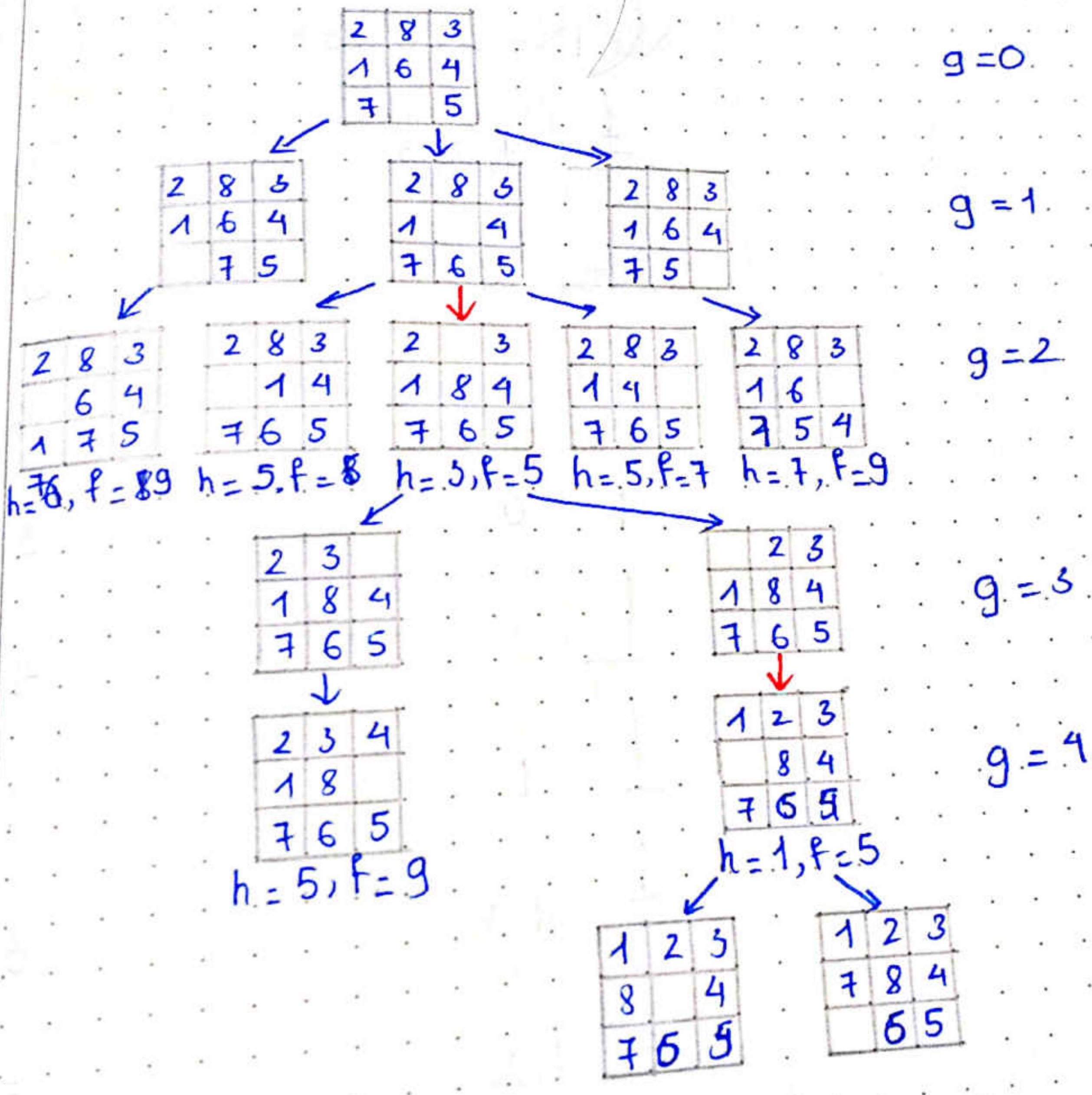


VD: Chuyển ô 2 lần, h≥1

1 2 3
8   4
7 6 5



VD: Chuyển ô 2 lần, k>2



→ Không hình, không thể tới đích được.

Date: ..../......./

VD : Bài toán Tháp Hà Nội

Xét $N = 3$ $2^3 = 8$

1, 2, 3	0
2, 3	3
3	2
0	3
2	4
1	4
1, 3	3
1, 2	5

Quiz 3: Admissible heuristics : $h(n) \le h^*(n)$

$h_1(n)$ = total number of misplaced tiles

$h_2(n)$ = total Manhattan distance

$h_3(n) = 0$ $\quad$ $h_4(n) = 1$

$h_5(n) = h^*(n)$ $\quad$ $h_6(n) = \min(2, h^*(n))$

$h_7(n) = \max(2, h^*(n))$

Để di chuyển 1 ô về đúng vị trí luôn cần ít nhất 1 phép biến đổi, giả sử có $h(n)$ vị trí sai thì cần $>= h(n)$ phép biến đổi, do đó $h(n) \le h^*(n)$ ($h_1(n)$ thoả)

- Tương tự $h_2(n)$ cũng thoả
- $h^*(n)$ luôn không âm $\geq h_3(n)$ ($h_3(n)$ thoả)
- $h_4(n) = 1$ k° thoả vì $h(n)$ (n là Goal) $= 0$
- $h_5(n)$ thoả nhưng k° đạt đc 1 cách hiệu quả
- $h_6(n)$ thoả $\quad$ - $h_7(n)$ k° thoả

* Heuristic thông tin $\quad$ ta nói h_2

- Nếu tại $\forall n$ $h_2(n) \geq h_1(n)$ thông tin h_1 (h_2 k° bao giờ mở nhiều nút hơn h_1)
- Relaxed problem (Gỡ bớt ràng buộc): Chi phí giải ít đi so với giải tối ưu, tuân theo bất thức tam giác → nhất quán.