



AI #1: Blind Search

- Agent คือตัวแทนใด ๆ ที่อยู่ภายในสิ่งแวดล้อมที่กำหนด เช่น หุ่นยนต์ดูดฝุ่น
- ประเภท Agent
- Intelligent ต้องมีมนุษย์เข้าไปควบคุม
- Autonomous สามารถทำงานได้เอง ไม่ต้องมีมนุษย์คอยควบคุมสิ่งแวดล้อม

- ประเภท Accessible

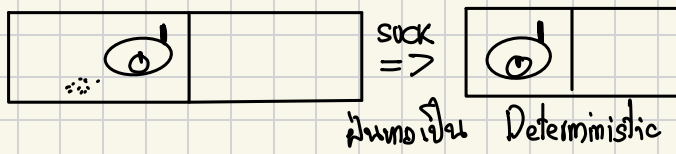
- local รู้แค่ที่อยู่ตำแหน่ง
- global รู้ทั้ง L, R สิ้น
- none

- ประเภทสภาพ env

- Static - สิ่งแวดล้อมไม่เปลี่ยน ฝุ่นหมด ไม่สิ่งใหม่เพิ่ม
- Dynamic
- Semi - สิ่งแวดล้อมไม่เปลี่ยน มาตราวัดเปลี่ยน เช่น เวลาผ่าน policy เปลี่ยน

- ประเภท Deterministic

- Deterministic - รู้ว่าทำ action หนึ่งแล้วได้ผลลัพธ์อย่างไร
- Non-Deterministic - ไม่รู้ผลด้านบวกจาก action หนึ่ง สิ่งสุดท้าย ฝุ่นไม่หมด

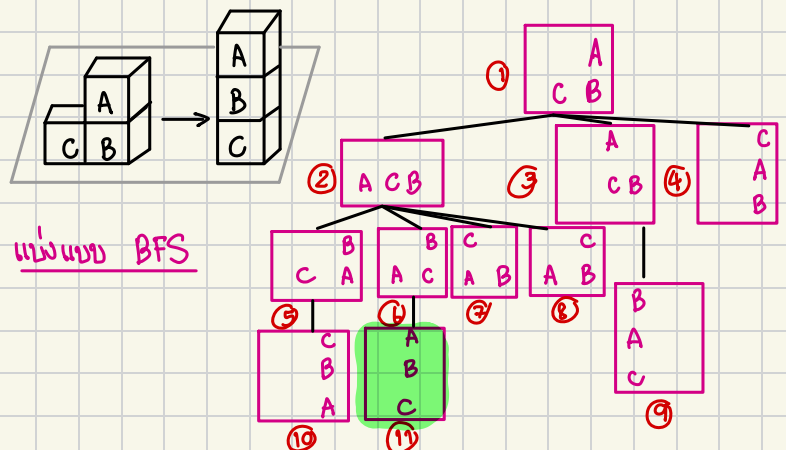


| Accessibility | Deterministic (สิ่งแวดล้อม) | Result |
|---------------------|-----------------------------|---|
| Y | Y | <p>Single State: L, S₁</p> |
| N (หากรู้ว่าไม่ได้) | N (สิ่งแวดล้อมไม่) | <p>while not in LEFT ROOM: LEFT while not CLEAN: SUCK</p> |

| Accessibility | Deterministic | Result |
|---------------|---------------|---|
| N | Y | <p>ไม่รู้ว่าจะอยู่ที่ไหน แต่ทำได้ (Multiple state)</p> <div> <div>1. </div> <div>2. </div> <div>3. </div> <div>4. </div> <div>5. </div> <div>6. </div> <div>7. </div> <div>8. </div> </div> <p>$[1, 2, 3, 4, 5, 6, 7, 8] \xRightarrow{S} [1, 2, 3, 6] \xRightarrow{L} [2, 4, 6] \xRightarrow{S} [2]$</p> |

- N
 - N
 - ไม่รู้ว่าจะอยู่ที่ไหน ต้องความน่า
- สิ่งที่ต้องทำในการแก้ปัญหา
 - นิยามปัญหาชัดเจน แสดง initial situation และ final situation
 - วิเคราะห์ปัญหา
 - หาความรู้ที่ใช้ในการแก้ปัญหา
 - เลือกเทคนิคแก้ปัญหาให้เหมาะสม

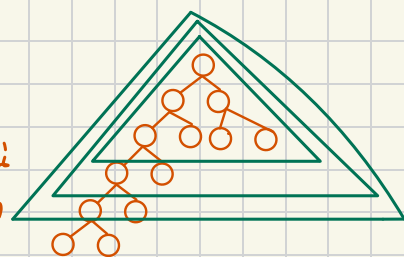
State Space Search : Block World



แบบ BFS

ข้อดี BFS

- ไม่ติด path ที่ลึกมาก ๆ โดยไม่พบคำตอบ
- ถ้ามีคำตอบ BFS ประเมินว่าพบแน่แล้วได้ path สั้นสุด เพราะ path ที่ยาวกว่าจะไม่ถูกตรวจสอบ ถ้า path ที่สั้นกว่าจะเจอคำตอบก่อน path



BFS code while True:

if fringe \emptyset return FAILURE

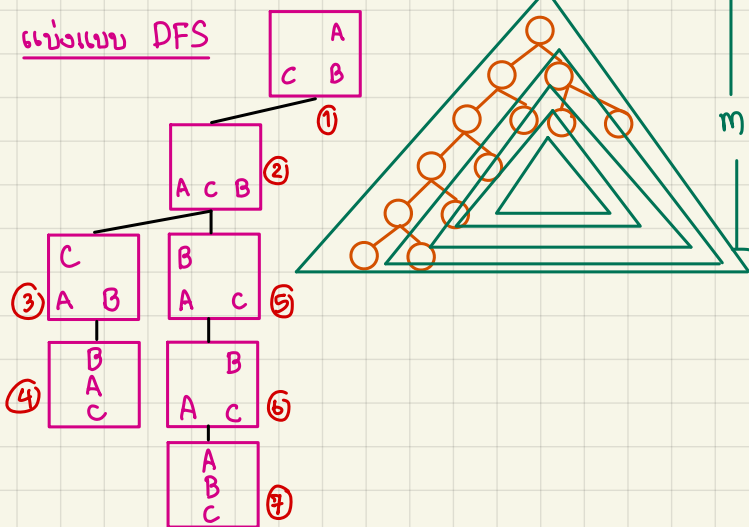
#ไม่เหลือ Leaf node

front \leftarrow removeFront (fringe)
(fringe.pop(0))

if is_goal (front) return solution

insert_all (gen_successor (front), fringe)

แนวทาง DFS



DFS code while True:

insert_all (gen_successor (front), fringe)

if fringe \emptyset return FAILURE

#ไม่เหลือ Leaf node

front \leftarrow removeFront (fringe)
(fringe.pop(0))

if is_goal (front) return solution

ข้อดี DFS

- ใช้ memory น้อยกว่า BFS เพราะหา state ในหน่วยปัจจุบันเท่านั้นที่ถูกลบ
- DFS อาจพบคำตอบโดยไม่ต้องค้นหา space มากเกินไป

การวิเคราะห์อัลกอริทึม

$b \rightarrow$ Branching Factor จำนวนลูกที่สามารถเกิดขึ้นสูงสุด

$d \rightarrow$ Depth ของคำตอบที่ดีที่สุด (Least cost solution)

$m \rightarrow$ Maximum depth

Optimality \rightarrow เจอคำตอบที่ดีที่สุดหรือไม่

Completeness \rightarrow เจอคำตอบหรือไม่ มีโอกาสวน loop ?

Time Complexity \rightarrow จัดจำนวน node ที่ถูก generate $O(?)$

Space Complexity \rightarrow จำนวน node สูงสุดที่ต้องเก็บ

Iterative Deepening Search

- ทำ DFS แบบกำหนด limit ความลึกแต่ละรอบ

$m=0$

$m=1$

$m=2$

$m=3$

...

$m=d$

•



10^0 วนรอบ 6 รอบ = 1×6

10^1 วนรอบ 5 รอบ = 10×5

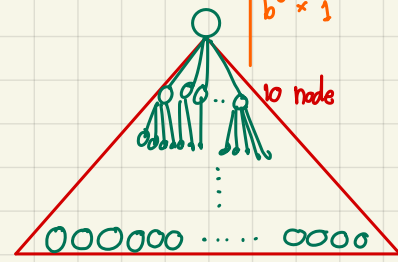
10^2 วนรอบ 4 รอบ = 100×4

10^3 วนรอบ 3 รอบ = $1,000 \times 3$

10^4 วนรอบ 2 รอบ = $10,000 \times 2$

10^5 วนรอบ 1 รอบ = $100,000 \times 1$

จำนวน node ที่ถูก expand 123,456 nodes



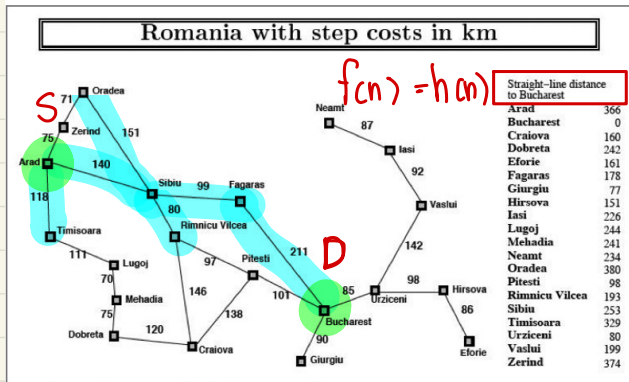
วิเคราะห์อัลกอริทึม

| | BFS | DFS | IDS |
|------------------|--------------|---|----------|
| Optimality | Y | N | N |
| Completeness | Y | N (Y เมื่อเก็บ repeat state และ maximum depth) | N |
| Time Complexity | $O(b^d)$ | $O(b^m)$ | $O(b^d)$ |
| Space Complexity | $O(b^{d+1})$ | $O(bm)$ | $O(bd)$ |

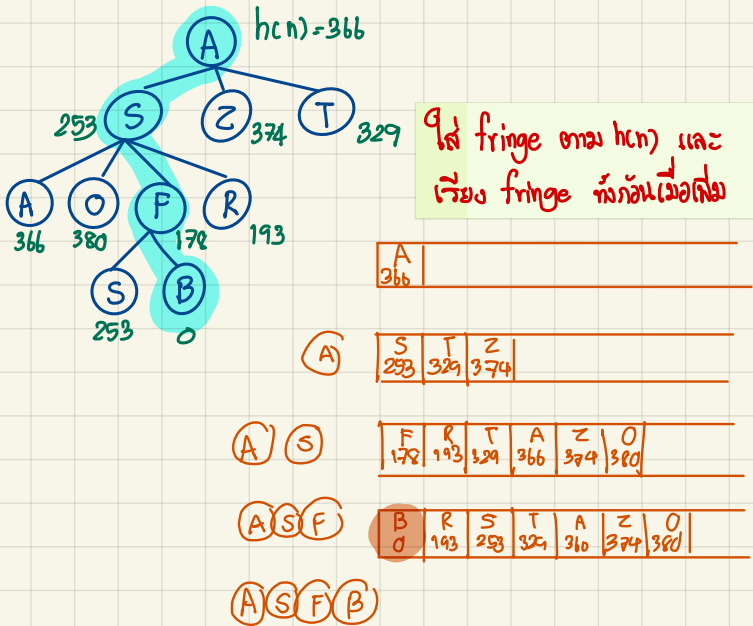


AI #2: Informed Search

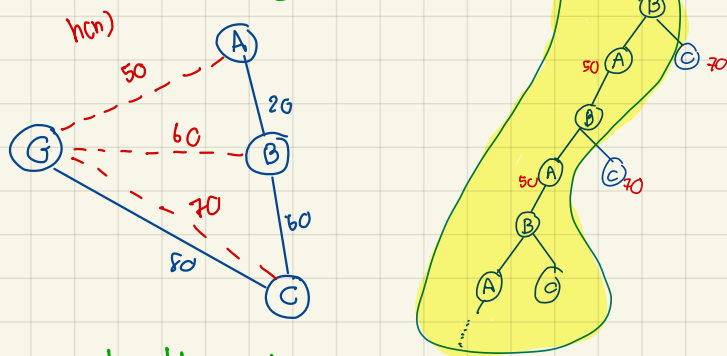
Greedy Search



$h(n)$ ไม่ใช่ค่าจริง เพราะวัดด้วยการกะจัดองๆ



วิเคราะห์ Greedy



Optimality : No

Completeness : No เพราะมี loop

Time Complexity } ขึ้นกับ $h(n)$
Space Complexity }

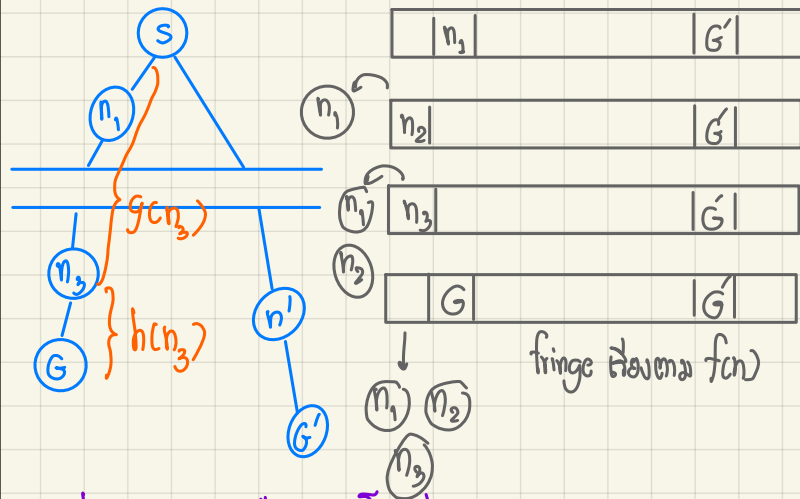
A* Search

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

$h(n)$ คือ admissible heuristic ที่ทำให้ A* ได้

คำตอบที่ดีที่สุด (แต่ไม่เสมอไป ขึ้นกับ $h(n)$)

Admissible heuristic ต้องวัด heuristic ไม่เกินจริง



$h(n) + g(n)$ คือระยะทางโดยประมาณจาก $S \rightarrow G$ โดยผ่าน node n

พิสูจน์

$g(G') > g(G)$ ระยะจริงวัดจาก $S \rightarrow G'$ ต้องมากกว่า $S \rightarrow G$ เสมอ

ถ้า $f(n) < f(G')$ เป็นจริง จะได้ Optimal Sol

จัดรูป ; $\cancel{h(G)} + g(G) < \cancel{h(G')} + g(G')$ - 7:07 $G \rightarrow G = 0$

$$f(n) \leq g(G) \leq g(G')$$

$$\text{แสดงว่า } h(n) + g(n) \leq h(G') + g(G')$$



AI #2: Swarm Intelligence

SI คือพฤติกรรมที่ทำงานร่วมกันระหว่าง agent กระทำ
กับสิ่งแวดล้อม

ประเภทการสื่อสาร

- Direct - agent สื่อสารตรงๆ เช่น แลกเปลี่ยน
สารเคมีของมด
- Indirect - สื่อสารผ่านสิ่งแวดล้อม โดย mark
ทิ้งไว้ เมื่อ agent เจอจะตามทำตามสิ่งที่เห็น
(stigmergy)

คุณลักษณะของ Artificial Stigmergy

- ① การสื่อสารแบบ Indirect
- ② Local Accessibility - access ที่สุดใกล้เคียน