

Programming Assignment 1

การค้นหามีข้อมูลนำทางและไม่มีข้อมูลนำทาง (Uninformed and Informed Search)

เพื่อพัฒนา search algorithms พื้นฐานที่ใช้ในการค้นหาเส้นทางใน graphs หรือ grids

แนวคิดหลักที่ต้องนำไป implement:

Search Algorithms: Breadth-First Search (BFS), Depth-First Search (DFS) และ A* Search

Graph/Grid Representation: สร้าง search space เช่น เขาวงกตหรือผังเมือง

Heuristics: การกำหนดและการประยุกต์ใช้heuristics ที่ยอมรับได้(admissible heuristic) เช่น Manhattan distance หรือ Euclidean distance

คำอธิบายงาน:

พัฒนา search algorithms 3 แบบเพื่อค้นหาเส้นทางที่สั้นที่สุดจากจุดเริ่มต้นไปยังสถานะเป้าหมายบน grid ที่กำหนด (อาจจะเป็นเขาวงกตหรือผังเมือง)

1. Graph Setup: สร้างแบบจำลอง grid map ในรูปแบบกราฟ โดยที่ grid cells เป็น nodes และการเคลื่อนที่ระหว่าง cells ที่อยู่ติดกันเป็น edges

2. Uninformed Search: พัฒนา Breadth-First Search (BFS) และ Depth-First Search (DFS) เพื่อค้นหาเส้นทาง

3. Informed Search: พัฒนา A* Search algorithm โดยต้องกำหนดและใช้admissible heuristic อย่างน้อย 1 ตัว เช่น การคำนวณระยะทางแบบเส้นตรงไปยังเป้าหมาย

4. Evaluation: รันทั้ง 3 algorithms บนแผนที่หลายแบบที่มีความซับซ้อนต่างกัน เช่น แผนที่ขนาดเล็ก ขนาดใหญ่ และแผนที่ที่มีสิ่งกีดขวางหนาแน่น

5. Report: เปรียบเทียบประสิทธิภาพของ BFS, DFS และ A* ในประเด็นต่อไปนี้:

- Path Cost/Length: เส้นทางที่ค้นพบนั้นเหมาะสมที่สุด (optimal) หรือไม่?

AWS.

- BFS (ถ้า cost เท่ากันทุกก้าว) = optimal
- DFS = ไม่รับประกัน optimal
- A* = optimal

- Number of Nodes Expanded: จำนวน nodes ที่ถูกขยายออกไป

ซึ่งเป็นตัววัดประสิทธิภาพของการค้นหา

AWS. A* มักขยายโหนดน้อยกว่า BFS เพราะใช้ $f(n)=g(n)+h(n)$ เลือกโหนดที่ “น่าจะใกล้เป้าหมาย” ก่อน

- อธิบายว่าเพราะเหตุใด A* จึงมีประสิทธิภาพดีกว่าหรือแย่กว่า BFS/DFS ในบางสถานการณ์เฉพาะ

โดยอ้างอิงถึงคุณภาพของ heuristic

AWS.

- ถ้า heuristic “ดี” (ใกล้ค่าจริงแต่ไม่เกินจริง = admissible) จะนำทางได้แม่นยำ → expanded ลด
- ถ้า heuristic แย่/ข้อมูลน้อย (เช่น $h=0$) A* จะกลายเป็น Uniform Cost Search → ใกล้เคียง BFS ในพฤติกรรม