ÇANKAYA UNIVERSITY

FACULTY OF ENGINEERING

COMPUTER ENGINEERING DEPARTMENT

CENG 466

Artificial Intelligence

FINDING OPTIMAL PATH

201514039 Merve Karakaya

Table of Contents

1. Introduction

Pathfinding or pathing is the plotting, by a computer application, of the shortest route between two points. The optimal path contains different features in addition to this definition. In optimal path, is exploring optimal algorithms in terms of time, space, and cost of solution path, and quantify their performance.

In this project, I will examine the optimal path finding problem in accordance with AI algorithms.

2. Definitions

Some general definitions in AI:

Agent: An agent is something that perceives and acts in an environment.
Ideal Agent: An ideal agent always takes actions that maximizes its performance. An agent adopts a goal and searches the best path to reach that goal.

Rational Agent: A rational agent is an agent which has clear preference, models uncertainty, and acts in a way to maximize its performance measure with all possible actions. A rational agent is said to perform the right things.

An AI system can be defined as the study of the rational agent and its environment.

Intelligent Agent: An intelligent agent is an autonomous entity which act upon an environment using sensors and actuators for achieving goals. An intelligent agent may learn from the environment to achieve their goals.
State: The set of all information items that describe a system at a given time.
State Space: The set of states that a problem can be in.

The set of states forms a graph where two states are connected if there is an action that can be performed to transform the first state into the second

Environment: An environment is everything in the world which surrounds the agent, but it is not a part of an agent itself. An environment can be described as a situation in which an agent is present. The environment is where agent lives, operate and provide the agent with something to sense and act upon it.

Action: Takes the agent from one state to another one.
Goal : It is the situation that the intelligent agent is trying to reach.
Goal Formulation: An intelligent agent should define its goals together with the limiting factors. Goal Formulation is the first step in problem solving.
Problem Formulation: To reach the goal, the agent should decide about the necessary actions.
Problem formulation is the process of deciding what actions and states to consider to reach the goal. Problem Formulation follows goal formulation.

3. Finding Optimal Path

3.1 Problem Solving

The goal in this project is to reach the routes between locations with the optimal path.
The locations show states, meanly nodes in graph.
The state space is defined as the locations and the roads between them that is links of the nodes.
The limiting factors in the project are:

Cost: The agent will want to find the lowest cost path from initial to goal state.

Distance: The agent will want to find the transition from the initial state to the goal state using the shortest distance.

Time: The agent will want to find out how to reach the goal state as soon as possible. Therefore, as the distance increases, the time and cost will increase.

Speed Limit: The speed limit on the roads to be chosen while reaching the goal state will affect the cost and time. This speed limit can vary within the city and between cities.

Road Condition: The condition of the roads to be chosen while going to goal state will affect the distance, cost and time, such as road construction works, dead-end roads, toll roads (expressways).

Present Location: To reach the target location, the present location must be known.

The environment changes dynamically as the agent detects the conditions and updates the environment according to the above conditions.

3.2 State Space Graph

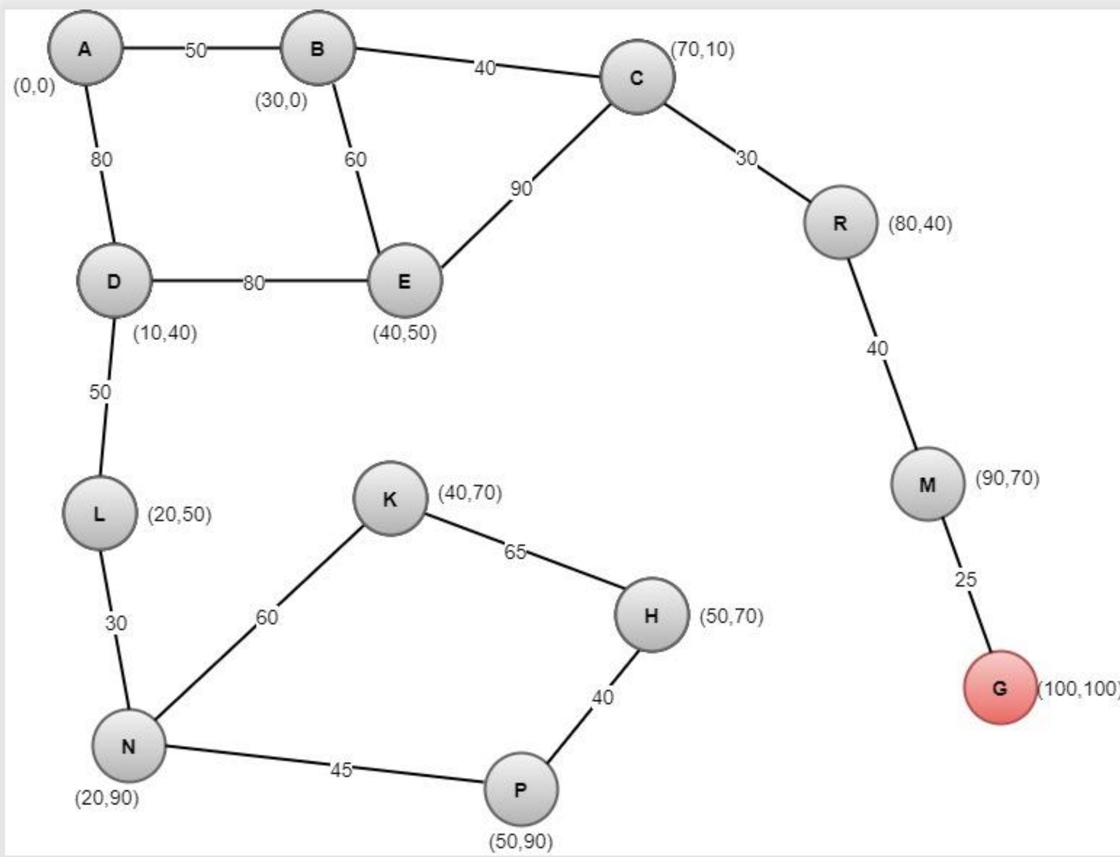

Figure 1: State Space Diagram

The nodes in the graphic above show different locations. Links indicate the routes between these locations. A location is the initial position, G location is the target location. The numerical values on the links represent the cost between two different locations. The values shown in parentheses show the distance between locations.

3.3 Search Algorithms

Some information about search algorithms is given below:

Greedy Search Algorithm

In using Greedy algorithm, choose the child with shorter distance to the goal node. Meanly, the greedy search which tries to expand the node closest to the goal.

Uniform-cost Algorithm

Uniform-cost search is a searching algorithm used for traversing a weighted tree or graph. This algorithm comes into play when a different cost is available for each edge. The primary goal of the uniform-cost search is to find a path to the goal node which has the lowest cumulative cost. Uniform-cost search expands nodes according to their path costs from the root node.

A* Search Algorithm

A* search is the most commonly known form of best-first search. It uses heuristic function h(n), and cost to reach the node n from the start state g(n). It has combined features of UCS and greedy best-first search, by which it solve the problem efficiently. A* search algorithm finds the shortest path through the search space using the heuristic function. This search algorithm expands less search tree and provides optimal result faster. A* algorithm is similar to UCS except that it uses g(n)+h(n) instead of g(n).

At each point in the search space, only those node is expanded which have the lowest value of f(n), and the algorithm terminates when the goal node is found.

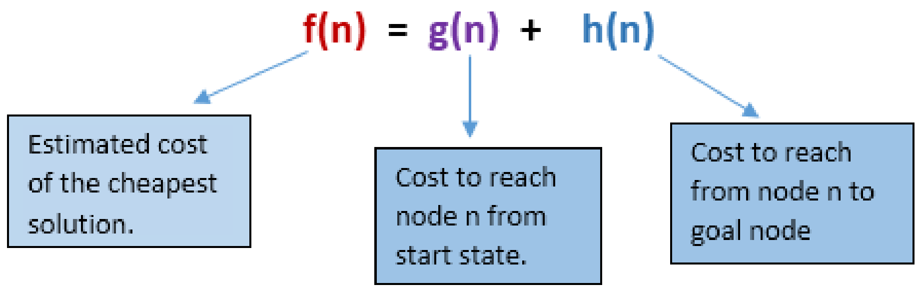

Figure 2: Estimated Cost of Solution

3.4 Searching with A* Algorithm

According to the search algorithms explained in the previous section, the most suitable algorithm for this project is the A* search algorithm because the Greedy search algorithm only chooses the shortest path close to the target, the Uniform-cost search algorithm simply proceeds through the paths with the lowest cost. However, for the problems of finding the most suitable path, the two properties just mentioned should be evaluated together.

In other words, while reaching the target, the nodes with the shortest path and the least cost should be expanded. Accordingly, a sample scenario would be as follows:

1- A* Search Example (According to Figure 1)

A (Initial Node) from to G (Goal Node)

B |100-30, 100-0| = B(70,100) distance is 70+100+**50** = 220

D |100-10, 100-40| = D(90,60) distance is 90+60+**80** = 230

220 is smaller than 230. Therefore, B should be chosen.

A → B → ?

E |100-40, 100-50| = E(60,50) distance is 60+50+**50**+**60** = 220

C |100-70, 100-10| = C(30,90) distance is 30+90+**50**+**40** = 210

210 is smaller than 220. Therefore, C should be chosen.

So path should be as follows:

A → B → C → R → M → G

2- A* Search Example (According to Figure 1)

D (Initial Node) from to H (Goal Node)

L |50-20, 70-50| = L(30,20) distance is 30+20**+50** = 100

E |50-40, 70-50| = E(10,20) distance is 10+20**+80** = 110

100 is smaller than 110. Therefore, L should be chosen.

D → L→ N→ ?

K |50-40, 70-70| = K(10,0) distance is 10+0+**50+30+60** = 150

P |50-50, 70-90| = P(0,20) distance is 0+20+**50+30+45** = 145

145 is smaller than 150. Therefore, P should be chosen.

So path should be as follows:

D → L→ N→ P→ H

3.5 Knowledge-Base Creation

- Assume a optimal_path_detector agent will find the most optimal path from an A location to G location. Our goal is to reach a location with the shortest distance and the least cost.

- The roads but, can be under construction, dead-end roads and toll roads (expressway) from time to time.

- For instance, the road under construction situtation traffic goes more slowly. In case the road is a dead end, the agent should choose another way. The agent should find a paid or free route according to preference.

- There is a copy of a graph for the agent with the locations and the paths connecting them, and the agent can update this graph based on the information it detects from the sensors.

- The agent can move forward, turn left or right and U turn and go backward.

- Speed conditions are that limit within the city is 82 km / h, between cities is 120 km / h.

3.6 Knowledge-Base Using Propositional Logic

List the locations:

- X1: M is a location
- X2: R is a location
- X3: B is a location
- …

Describe the roads connecting the locations:

- Y1: M is connected to G
- Y2: C is connected to R
- Y3: B is connected to E
- …

Describe the attributes of the locations and roads:

- Z1: The road connecting N to K is 60 km.
- Z2: The speed limit on road connecting N to K is 82 km/h.
- …

Describe the possible percepts:

- Sense (Time)
- Sense (Speed Limit)
- Sense (Road Condition)
- Sense (Present Location)

Describe the Actions:

- Move forward
- Go backward
- Turn right
- Turn left
- U turn

Conditional moves are decided about during inference:

For instance: Sense (Road Condition) = Dead-end → Go backward

Describe rules:

- ***Sense (Present Location)*** = Road connecting N to K **Λ** ***Sense (Speed Limit)*** = 82 km/h → Z1: The road connecting N to K within the city.
- ***Sense (Present Location)*** = Road connecting E to C **Λ** ***Sense (Speed Limit)*** = 120 km/h → Z3: The road connecting E to C between the cities.
- ***Sense (Present Location)*** = Road connecting P to H **Λ** ***Sense (Road Condition)*** = Under construction → Z5: The road connecting P to H is infinity km.
- ***Sense (Present Location)*** = Road connecting M to G **Λ** ***Sense (Time)*** = ½ hour → Z8: The road connecting M to G takes ½ hour.

3.7 Create The Knowledge-Base

Add all propositions as sentences (facts) to the knowledge-base.

Assume the knowledge-base is called KB.

Using TELL:

- TELL(KB, Z1: The road connecting N to K is 60 km)
- TELL(KB, ***Sense (Current Location)*** = Road connecting N to K***Λ Sense(Speed Limit)*** = 82 km/h → Z1: The road connecting N to K within the city)

Ask the agent to find the optimal path from A to G

- ASK (KB , what is the optimal path from A to G)

3.8 Inference Using Forward Chaining

- Knowledge Base:

  If [Road condition is dead-end road] Then [Go backward]

  If [Road condition is under construction] Then [U turn]

  If [Road condition is toll road (expressway)] Then [Go backward or Move forward]

  If [Speed limit is 82 km/h] Then [Road connecting is within the city]

  If [Speed limit is 120 km/h] Then [Road connecting is between the cities]

- Input Data:

  [Road is dead-end]

- Goal:

  [What action will be taken**?**] → **? = Go Backward**

4. Conclusion

The aim of our project is to reach the goal by following the optimal path. The search algorithm to be used for this purpose must find the minimum distance and cost. Then, it follows the optimal path. For this reason, A* search algorithm has been chosen.

The knowledge- base has been created for the next stage of the project. At this stage, the propositional logic mechanism was used and different rules were defined. For the inference mechanism, which is the last stage of the project, forward chaining is used.

In short, an AI system has been established that will find the optimal path throughout all these processes.

5.References

[1] https://www.javatpoint.com/ai-informed-search-algorithms

[2] www.drawio.com

[3] CENG466 Artificial Intelligence Lecture Notes, Engineering Faculty of Cankaya University