Finding Shortest Path in Monster Hunter Games using Dijkstra Algorithm

Taufic Leonardo Sutejo
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung 40132, Indonesia
13514022@std.stei.itb.ac.id

***Abstract*—Dijkstra algorithm is a derived concept of greedy algorithm from strategy algorithm. It is used for solving the optimizing problem. One of the optimizing problem that be used by people nowadays is choosing the shortest path.**

Dijkstra algorithm has been used in many kinds of case such as in games, in transportation companies, in tours and so on. In games, for example Monster Hunter, a game that the writer was going to analyze. In transportation companies, for example Uber, grab taxi, easy taxi, Gojek.

The usual problem of using Dijkstra algorithm in games is how player to choose the shortest path from one place to another place. Each game has its own reason such as running of time, mission, and so on.

Keywords—Greedy, Dijkstra, algorithm, monster hunter, shortest path, graph.

I. INTRODUCTION

What is Action role-playing games? Before we go further there are 3 points that you need to know. First, action game is a game that controlling a character such as attack, defend, other character movement into action in a game controller in real time. It means that when a player press any button on the remote control, the character will move as exactly as what the button can do directly. For example, GTA, Call of Duty, and so on. Second, adventure game is an action game that is an action game where player can find items or weapons in various are in the game and allow them to explore any area in the game. Adventure game can also be called story game. Adventure game is one of the extension game from action game. For example, Assassin's Creed Syndicate, the Walking Dead, and so on. Finally, a strategy game is a command game that any step in battle or character need to be planned according to what story line or scenarios are carried out. For example, Star Craft, Command and Conquer, and so on.

RPG can be categorized into adventure, action and strategy games. RPG is a game which every character has its own role play in a game. RPG usually is a fantasy or science fiction background setting, an imaginary world. For example, Monster Hunter, Final Fantasy, Diablo, and the most popular online game World of Warcraft(WOW), and so on.

Source:
http://monsterhunter.wikia.com/wiki/Monster_Hunter_Wiki
Figure 1.1 One of the RPG examples

One of the role playing game that this paper is going to discuss is Monster Hunter. Monster Hunter release the first generation in PlayStation 2 since 2004 and now monster hunter has released until forth generation. Monster Hunter is fantasy and science fiction role playing game that is developed and published by one of most popular company in japan, Capcom. Besides Monster Hunter as a RPG, it also has its own game base on its anime.

Source:
http://venturebeat.com/2016/02/13/how-monster-hunter-finally-found-success-in-the-united-states/
Figure 1.2 Monster Hunter

The Endless Story begins from a character which is player in a village, given various quests such as gathering materials, hunting quests, urgent quests, and capturing quest. The player can choose various weapon such as, great sword, long sword, sword and shield, dual blades, bow, bowgun, and so on. The player can also choose various armor. Every weapon and armor are forged in blacksmith somewhere around the village the weapon and armor can have a material from the monster hunted by the player.

Source:
https://lunaticfisherman.wordpress.com/2011/04/02/
Figure 1.3 Player Village (Pokke Village second generation of Monster Hunter)

The Goals of this game are to complete all the quests that given by the chief by successfully do the quests. Every quests that succeed will be given rewards. Gathering materials such as player can gather minerals, plants, insects, honey, bones, and fishes and so on. Hunting quest is hunting big monster or hunting small monster, insects. The monster also have its own kingdom, family, and ordo. Beside it has its own hierarchy, the monster has a name. In this game, player does not level up but the player can improve the weapon and armor to get stronger. This game has also a solo mode and multiplayer mode. Solo mode, player can clear the quests by him/herself and multiplayer mode, player can clear the quests together with his/her friend using online mode. The multiplayer mode allows up to four player to play together.

Besides clearing the quest, single player or multiplayer, this game has a school in the game, to help beginner understand how to play and farm. Player can visit their farm and gather mineral, plants, honey, fishes, and insects. In that village, cat are assisting villagers doing their jobs even hunting.

Source:
http://monsterhunter.neoseeker.com/wiki/Pokke_Farm
Figure 1.4 Player Farm (Pokke farm second generation of Monster Hunter)

There are three things that make the quests failed. First, player must not die for more than 3 times. It is also including multiplayer mode, not each player can die 3 times but each quest allowed 3 player die. If the player die once then he or she will respawn and return to the camp. Second, player must clear the quest less than 50 minutes. In special condition, player must clear the quest less than 35 minutes. Third, player surrender from the quests.

II. BASIC THEORY

A. Graph

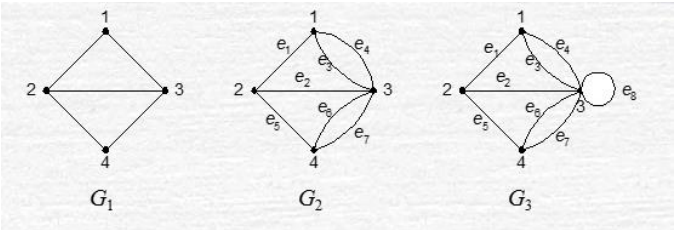
A graph consists of vertexes and edges that connected to the vertex. Graph can be defined as G = (V, E) which is

- V is a set of vertexes. It must not empty.
- E is a set of edges that connected to each node. E may be empty.

Graph can be grouped as several types. They are groups that have a direction, have a pair of edges, or looped graph.

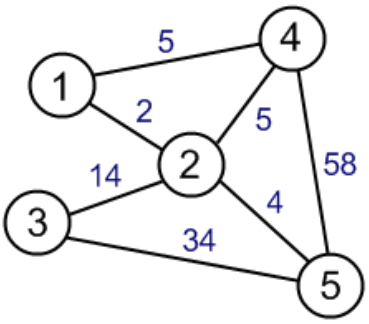
a. Graph consists of looped and not looped graph or have a pair of edges and not have a pair of edges.
   1. Unsimple graph, have at least one loop in any of its node. Look in picture, G2.
   2. Simple graph, does not have any loop in any of its node. Look in picture G1.

b. Graph consists of edges have a direction and no direction
   1. Directed graph, graph that have a direction. Look in picture G2.
   2. Undirected graph, graph that does not have any direction. Look in picture G2.



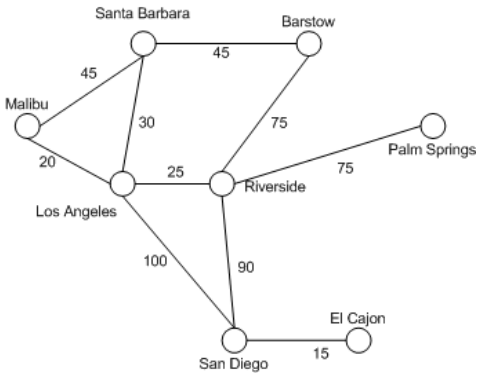
Source:
Diktat Matematika diskrit
Figure 2.1 Example of graph

A weighted graph is a graph that each of the edges has its own value or number. The value represents the distance between two nodes. Below is the example



Source: http://web.cecs.pdx.edu/~sheard/course/Cs163/Doc/Graphs.html

Figure 2.2 Example of weight graph



Source: https://msdn.microsoft.com/en-us/library/aa289152(v=vs.71).aspx

Figure 2.3 Example of City Graph

B. *Greedy Algorithm*

Greedy algorithm is an algorithm to solve optimizing problems. There are only two optimizing problems. They are maximization and minimization. Greedy algorithm for each step, there is always evaluation and looking for the best value (local optimum) and hoping that this local optimum can be the best solution (global optimum).

Here are the Greedy algorithm elements:

- Set of candidate, C
- Set of solution, S
- Selection function
- Feasible function
- Objective function

Here are the some examples using Greedy algorithm:

- The coin exchange problem
- Optimizing system time
- An activity selection problem
- Integer knapsack problem
- Fractional knapsack problem

There are advantages and disadvantages using Greedy algorithm. The advantages using Greedy algorithm:

- Because it is using selection function, this algorithm can find the smallest or the largest value in a problem.

The disadvantages using Greedy algorithm:

- It does not check all alternative that's possible like exhaustive search do so there is a possibility we do not get the most optimum value.

C. *Shortest Path Algorithm*

Shortest Path Algorithm is an algorithm that to find the shortest path from one point to another point.

There are several shortest path problems:

- A pair shortest path
- All pairs shortest path
- Single-source shortest path
- Intermediate shortest path

D. *Dijkstra Algorithm*

Dijkstra algorithm is one of the shortest path solving problem that derived from Greedy algorithm. Dijkstra algorithm had been found by Edsger Wybe Dijkstra. He is one of the most influencing in computer science. He won a lot of prizes and awards and one of the award is the ACM Turing Award.

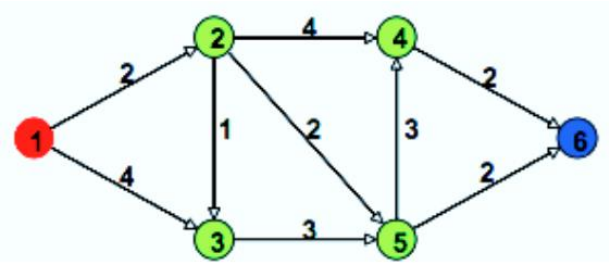
Dijkstra algorithm find shortest path using several steps. The steps is this following:

- Take the most minimum weighted path or value connected from one vertex to another vertex that has not been visited in every step.
- From the beginning vertex to the new vertex must be the minimum among the others vertex.

Dijkstra algorithm is a single-source shortest path problem that can solve both directed and undirected graph. All the edges values must not non-negative and must be connected for every nodes.

Following will explain how Dijkstra algorithm works:

1. Every node will be initiate by value infinity.
2. From node 1 will be initiate by value 0.
3. From node 1 to node 2 will make the path value into 2 and the shortest path for a while from 1 to 2 is 2.
4. From node 1 to node 3 will make the path value into 4 and the shortest path for a while from 1 to 3 is 4.
5. But from node 2 to node 3 will make the path value into 3 and the shortest path that before in node 3 is 4 become replaced by 3.
6. This algorithm will go on until meet the shortest path to 6
7. The shortest path from 1 to 6 is 1 – 2 – 5 – 6 and the value will be 6



Source:
http://math.mit.edu/~rothvoss/18.304.3PM/Presentations/1-Melissa.pdf
Figure 3.1 Dijkstra Algorithm example

The following below is the pseudo-code of Dijkstra algorithm.

```
dist[s] ←o                              (distance to source vertex is zero)
for all v ∈ V–{s}
    do dist[v] ←∞                       (set all other distances to infinity)
S←∅                                     (S, the set of visited vertices is initially empty
Q←V                                     (Q, the queue initially contains all vertices)
while Q ≠∅                              (while the queue is not empty)
do  u ← mindistance(Q,dist)             (select the element of Q with the min. distan
    S←S∪{u}                             (add u to list of visited vertices)
    for all v ∈ neighbors[u]
        do if  dist[v] > dist[u] + w(u, v)      (if new shortest path found)
           then    d[v] ←d[u] + w(u, v)         (set new value of shortest path)
                                                (if desired, add traceback code)
return dist
```

III. FINDING THE SHORTEST PATH IN MONSTER HUNTER

Time is important in this game in several condition such as hunting the elder dragons because the elder dragons will escape in 35 minutes and the quest is finished. Usually monster that difficult to be hunted down will be called slaying quest which is player just need to slay or slow the monster before reaching the village. The game in every quest usually gives 50 minutes to clear the quest. It seems long but for high rank monster and solo will need fast action. The player meet the monster in certain area, the monster will not stay still. They can escape to somewhere on the map. If the player just keep searching the monster and take too long, the player just wasting time and failed the quest. Although the monster can be pinpointed using certain item.

Because of time is important and time is money and items. To save the time we can find the shortest path to the monster. Using Dijkstra algorithm can solve the problem. Let's see the example of monster hunter's map below

Source:
http://monsterhunter.wikia.com/wiki/Sandy_Plains
Figure 3.1 Example of Monster Hunter Maps

This is the graph representation from transforming the map into the graph. We know that when the player enter to the next node, for example 11 to 7. The player will not walk through the line but directly send to 7. But the point is, for example the player are in the edge of 11. The player need to walk until the way in to 7. So in this paper what we count is start from where the player's position and represent it into the graph. The black point in the map will represent the node which the player will be.

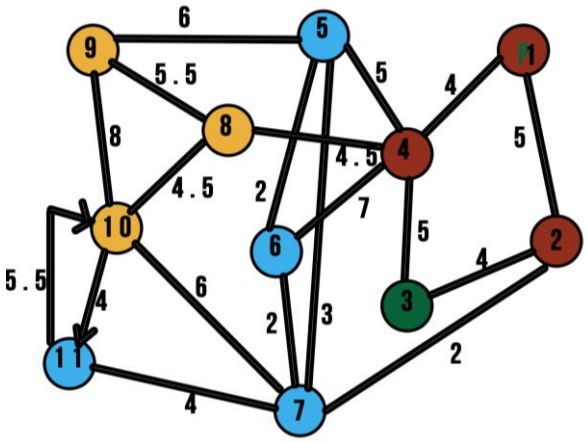


Figure 3.2 Graph representation from Monster Hunter Map

Perhaps reader will confuse about the graph. The writer will make the representative matrix. For once, this graph is representing the player position to the next node. Not exactly the distance between nodes.

	1	2	3	4	5	6	7	8	9	10	11
1	0	4.5	∞	3	∞	∞	∞	∞	∞	∞	∞
2	4.5	0	4	∞	∞	∞	3.5	∞	∞	∞	∞
3	∞	4	0	5	∞	∞	∞	∞	∞	∞	∞

4	3	∞	5	0	2	5.5	∞	4	∞	∞	∞
5	∞	∞	∞	2	0	4	3	∞	5	∞	∞
6	∞	∞	∞	5.5	4	0	3	∞	∞	∞	∞
7	∞	3.5	∞	∞	3	3	0	∞	∞	4	4.5
8	∞	∞	∞	3	∞	4	∞	0	4.5	4	∞
9	∞	∞	∞	∞	5	∞	∞	4.5	0	5	∞
10	∞	∞	∞	∞	∞	∞	4	4.5	5	0	4
11	∞	∞	∞	∞	∞	∞	4.5 s	∞	∞	3	0

Using Dijkstra algorithm, we know that from the node 1 where the player is and the detecting monster whereas using the item and it is in node 10. We can determine the shortest path as following. The illustration as following:

1. From node 1 there are 2 branches but the minimum value is 4 which is to node 4.
2. From node 4 there are 5 branches. The minimum value is 4.5 which is to node 8. Node 1 has been visited before.
3. From node 8 there are 2 branches. The minimum value is 4.5 which is to node 1

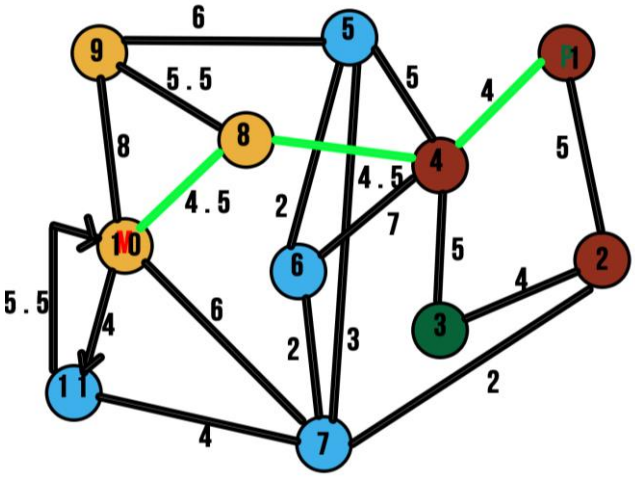


Figure 3.3 One of the Shortest Path Solution

From the above solution we can find the shortest path to the monster using Dijkstra algorithm. The shortest path is 1 – 4 – 8 – 10 and the total distance is 13. But the solution above if we know about where the monster is and bring the item. If we do not know where the monster is, better we find several locations that look big area that is possible for the monster location. For example, the monster can be in node 3, 4, 7, 8, 9, 10 and 11. If the monster in 4, you will find it immediately just taking 1 step from the illustration above. But how about 3, 7, 8, 9, and 11. Dijkstra algorithm defines that take the most minimum weighted path or value connected from one vertex to another vertex that has not been visited in every step. If we see the graph below, it tells we can go to 3 by taking 1 – 4 – 3 with total distance 9, we can go to 7 by taking path 1 – 2 – 7 with total distance 7, we can go to 8 by taking path 1 – 4 – 8 with total distance 8.5, we can go to 9 by taking 1 – 4 – 8 – 9 with total distance 14, we can go to 10 by taking 1 – 4 – 8 – 10 with total distance 13 or we can go to 11 by taking 1 – 2 – 7 – 11 with total distance 11.

IV. CONCLUSION

Using Dijkstra algorithm help a lot not only in reality but also in games to solve problem. Dijkstra algorithm help to find the shortest path from one place to another place when time is rushing. Using Dijkstra algorithm in RPG can also be found in one of the RPG such as Monster Hunter. Although in this game feel have plenty of time to hunt down the monster. But the writer believe there is a game out there that quite rushing time that you find the shortest path ask fast as you can in order to survive or to clear the mission. On the other hand, in this game sometimes is quite useful when time is running out and the player has not hunted down the monster or another case that the monster like to joke with the player for example we take figure 3.1. The player think that the monster will be going to area 6 from area 4 but it goes to are 10 that make you need to run the whole time or the monster is no where because of the tracking item has run out. The player find through all the area and cannot find it because the monster also mobile creature. Besides there are monsters that have ability to hide itself on the ground, invisible or maybe become one of the giant stones.

Dijkstra algorithm is not the only way to find the shortest path and not the best way to find it. There are still more such as A-star search, UCS (Uniform Cost Search), BFS (breadth first search), DFS (Depth First Search), IDS, Greedy Best First Search, Dynamic Programming searching shortest path. All above have their own advantages and disadvantages searching for the shortest path. But the writer choose Dijkstra algorithm because the writer want to know how it can solve games problem such as this game. The writer hope that Dijkstra algorithm can also solve any shortest path problem in a games.

V. ACKNOWLEDGEMENT

Taufic Leonardo Sutejo as the writer of this paper would like to say thanks to his parents that always support him, thanks to his both sisters that always give advice, thanks to his friends that always there when he needs the most. I want to thank to Dr. Ir. Rinaldi Munir, M.T. and Drs. Nur Ulfa as the lectures of IF2211 – "Strategi Algoritma". Because of them I can finish this paper.

REFERENCES

[1] https://lunaticfisherman.wordpress.com/2011/04/02/ Access in 7th May 2016

[2] http://www.rpgfan.com/editorials/old/1998/0007.html Access in 7th May 2016

[3] http://monsterhunter.wikia.com/ Access in 7th May 2016
[4] http://monsterhunter.neoseeker.com/wiki/Pokke_Farm Access in 7th May 2016
[5] http://whatis.techtarget.com/definition/role-playing-game-RPG Access in 7th May 2016
[6] http://math.mit.edu/~rothvoss/18.304.3PM/Presentations/1-Melissa.pdf Access in 7th May 2016
[7] Munir, Rinaldi. 2003. Matematika Diskrit Edisi Keempat. Bandung: Penerbit Informatika
[8] Munir, Rinaldi. 2009. *Diktat Kuliah IF2211 Strategi Algoritma*. Bandung: Institut Teknologi Bandung.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 7 May 2016

Taufic Leonardo Sutejo/13514022