Usage of Dijkstra's Algorithm to Optimize the Time Efficiency in Charm Grinding in Monster Hunter Portable 3rd

Hafizh Hananta Akbari - 13522132
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan 10 Ganesha Bandung
E-mail: 13522132@std.itb.ac.id

***Abstract*—Monster Hunter Portable 3rd is a game that was released on the PSP as part of the Portable series of the Monster Hunter game franchise, exclusive for the PSP. The game follows a similar structure and gameplay loop as past Monster Hunter games with some differences when compared to other Monster Hunter games released on the PSP. One of them is the addition of charms which are obtained through the act of mining in different areas within a locale. Due to the luck-based nature in obtaining charms, the act of charm grinding (which is an act done by players to repeatingly acquire charms) would require heavy amounts of time investment as every mining node not only has a small chance of getting charms, but every charm acquired would also have a very small chance of having a certain stat distribution according to the needs of a player. The amount of time required to obtain certain charms would make optimizing time efficiency an important factor in charm grinding. This paper researches the usage of Dijkstra's Algorithm in finding the most optimum and efficient route when obtaining charms to optimize time efficiency.**

Keywords—Monster Hunter Portable 3rd; Charm Grinding; Time Efficiency; Dijkstra's Algorithm

I. Introduction

Monster Hunter Portable 3rd is a game released on the Playstation Portable (also known as PSP) and Playstation 3 (also known as PS3) as a part of the 3rd generation games within the Monster Hunter franchise. It was released on December 1, 2010 for the PSP and August 25, 2011 for the PS3 with both releases only available in Japan. Although the game has never received a global release, the game saw massive success, especially on the PSP as it managed to be within the top 5 best-selling games for the PSP. Although official support for the game has ended, the game still has active communities that has since continued playing the game as well as providing community support for the game such as fan-made English patches to allow English-speaking players to play the game more easily.

As part of the 3rd generation of Monster Hunter games, Monster Hunter Portable 3rd introduces multiple different aspects within the Portable series of Monster Hunter games. One of the additions is the addition of charms which can be mined from mining nodes throughout different areas within a locale. Charms are used to increase the amount of skill points for certain skills depending on the stat distribution within the charm. Charms allows players to be more flexible in creating an armor set as it may provide certain skill points that would be important for the set as a whole. The skill points within a charm as well as the type of charm solely depends on luck which would make for a heavy time investment in order to obtain certain charms with certain stat distributions.

Due to the layout of locales which are divided into different areas for each locale, this creates the possibility of finding a route that would take the least amount of time in order to obtain certain charms. This means that in theory, it is possible to find such a route through mathematics as different areas have different layouts which would make every area unique in terms of dimensions. Finding the best route can be done in different ways which include the usage of Dijkstra's Algorithm. This paper researches the usage of Dijkstra's Algorithm to optimize the time efficiency in charm grinding in Monster Hunter Portable 3rd.

II. Theoretical Foundation

A. Monster Hunter Portable 3rd

Monster Hunter Portable 3rd is an action role-playing game released on both the PSP and PS3 that is developed and published by Capcom as one of the three games featured in the 3rd generation of games within the Monster Hunter franchise. The game was released only in Japan on December 1, 2010 for the PSP and August 25, 2011. It is the only 3rd generation Monster Hunter game to be released on Playstation consoles as opposed to Nintendo consoles and it also follows the Portable line of Monster Hunter games (Monster Hunter games that are exclusive to the PSP). Monster Hunter Portable 3rd features gameplay that follows the 3rd generation games with the inclusion of new aspects such as the introduction of new movesets for different weapons and charms while also omitting certain aspects such as underwater combat. Monster Hunter Portable 3rd also retains aspects from previous Monster Hunter games that are not available in the other 3rd generation games such as the static and non luck-based nature of mining nodes as well as the farm mechanic in the village hub world.

Monster Hunter Portable 3rd features a gameplay similar to past Monster Hunter games. The gameplay features a player controlling a character that was made during the character creation process. The character is a hunter that can hunt monsters with different weapons, gather materials from both monsters and environments, as well as interacting with other people in the different hub worlds. There are four hub worlds in

Monster Hunter Portable 3^{rd} that features multiple different things to offer. The following are the hub worlds featured on Monster Hunter Portable 3^{rd}.

1. Yukumo Village

Fig 2.1 Yukumo Village Hub World

Yukumo Village has a multitude of non-playable characters (also known as NPC). This is the main hub world that is physically connected to every other hub world with the back left corner being connected to the farm, the middle left being connected to the house, and the upmost front with the ladder being connected to the Gathering Hall. The NPC's in this hub world may serve different roles such as farm manager, equipment vendor, blacksmith for both the hunter and the cats following the hunter (known as palicos), palico vendor, the village chief who gives quests to the hunter, as well as several NPC's who make the hub world livelier.

2. Gathering Hall

Fig 2.2 Gathering Hall Hub World

The Gathering Hall has several NPC's as well as 2 features. The NPC's serve different roles such as the low rank quest counter, high rank quest counter, drink quest giver, drink vendor, as well as 2 other NPC's without a special purpose other than to make the Gathering Hall livelier. For players who engage in online play, there can also be the characters of other players within the Gathering Hall who have a chance of giving certain items. Other than the NPC's, there are 2 special features in this hub world which are the hot spring and drink vendor. The hot spring is used to increase the stats of a hunter during hunts and a drink vendor offers special drinks which grants the hunter special skills.

3. Yukumo Farm

Fig 2.3 Yukumo Farm Hub World

Yukumo Farm only has one unique NPC which is the farm felyne and the field felyne with the farm felyne only acts as a guide to the farm's features and the field felyne managing the field feature of the farm. The main features of the farm is the multitude amount of sections to acquire certain items. Those sections are divided into the mining point, field row, fishing pier, bug point, mushroom log, palico spot, BBQ spit, and a special wagon where palicos are able to gather materials.

4. Home

Fig 2.4 Home Hub World

Home acts as the hub world where a player can manage weapons, armors, and inventory space while also allowing the player to also manage the palico settings rather than forcing the player to go to the farm. The only NPC other than the palicos that are available within this hub world is the Poogie NPC. A player can interact with Poogie by changing it's clothes, picking it up, as well as petting it which results in a small minigame to time the length of the petting session.

Other than the hub worlds, Monster Hunter Portable 3^{rd} also features multiple amounts of locales and especially monsters. This ties into the main gameplay of this game. The main gameplay of Monster Hunter Portable 3^{rd} involves the player controlling the hunter with a third person perspective. Then, the player hunts down the monster in a 3D space within a certain area using the player's weapon of choice. The player is then rewarded with money and monster materials after the hunt. The following is the main gameplay of Monster Hunter Portable 3rd.

Fig 2.5 The Main Gameplay of Monster Hunter Portable 3rd

The main gameplay of Monster Hunter Portable 3rd features several design choices. Firstly, there is a quest timer which is located in the top left of the screen. Then, there are two bars which are the health bar which is colored green and the stamina bar which is colored yellow. Then, there are the weapon info which are signified by the colored sword icon and the red bar although this may differ depending on the player's weapon choice. Then, there are the names of the players and palicos within the hunt. Then, there is the minimap on the upper right of the screen which only fully shows up if a player picks up the map. Finally, there is the item slot in the bottom right of the screen to find and use items.

The gameplay loop of Monster Hunter Portable 3rd is similar to the previous Monster Hunter games which is preparing for a hunt, hunting monsters, acquiring materials from the monsters that have been hunted, create new weapons and armors, and repeat. Hunting monsters and acquiring materials from the monsters can be done by receiving and embarking on the quest that has been received. A player who has embarked on a quest is then put into a certain locale depending on the quest and is tasked to fulfill the quest's clear conditions. A player is then sent back to the hub worlds after completing a quest. This creates a connection between the hub worlds and the main gameplay of this game as both aspects are deeply connected to each other.

In the gameplay loop of Monster Hunter Portable 3rd, there are different types of quests. First is hunting quest which is a type of quest where a hunter is allowed to kill or capture the monster. This is the most common type of quests as this is the typical format for quests featuring large monsters. Second is slaying quest which is a type of quest where a hunter is only allowed to kill the monster. This type of quest is mostly reserved for small monsters and a type of monster called elder dragons. Third is gathering quest which is a type of quest where a hunter is tasked to acquire certain items and bring it to the base camp.

B. Volcano

Volcano is the name of an area or locale within Monster Hunter Portable 3rd. Volcano is unlocked as the last main locale in Monster Hunter Portable 3rd, not counting special locales used for certain major hunts. This locale is known for it's hot temperatures and fiery landscape. Due to this, Volcano is known to host certain kinds of monsters that have adapted to the harsh atmosphere of the locale with multiple different biological adaptations, no matter how big or small the monsters are. The large monsters in this area are known to be some of the more dangerous monsters that can be hunted down by the hunter.

Other than the harsh atmosphere of the locale, Volcano is also known as the locale with the most amount of mining nodes with the highest amount of drop rates for charms. This makes the Volcano locale the best locale to obtain charms and minerals. Volcano also has multiple other gathering points such as bug points and different kinds of plantation. Due to this, players tend to go to Volcano in order to acquire charms, different kinds of minerals, as well as other kinds of items that have a rarer chance of dropping in other locales.

C. Charm Grinding

Charm grinding is a term used to describe the act of a player repeatingly doing certain quests only for the sake of acquiring charms. Charms themselves are items in Monster Hunter Portable 3rd that can be appraised at the end of a quest to acquire talismans which is used to raise the skill points of certain skills. The concept of charms and talismans were introduced in the 3rd generation of Monster Hunter games and has been a mainstay since then. Charms and talismans offer a new random variable into the game with the type of charm a player would get depends completely on luck and the amount of skill points as well as the skill points that are provided with a talisman is random with the values of the talisman being set the moment a quest ended. This makes attempting to get certain talismans a long endeavor as it tests the player's patience and luck.

Charm grinding is done in order to get certain talismans that can substantially increase the amount of certain skill points depending on the weapon and armor set. This is usually done by mining in certain locales that tend to have a higher chance of dropping rare charms such as the Volcano as well as completing certain quests. Although it is possible to acquire talismans by only clearing certain quests, it is typically more accessible to mine in mining nodes instead. The act of mining is done by using a pickaxe on a mining node that tends to have a shape that looks like cracks on a wall or on a piece of rock. The following image is an example of a mining node.

Fig 2.6 A Mining Node in Monster Hunter Portable 3rd

Charm grinding is typically done by entering a gathering quest in the Volcano locale while bringing the necessary weapon, armor set, and items to maximize the total amount of items that can be acquired in a mining node. This is done in the Volcano locale due to the rates within the Volcano locale that are higher for rare charms compared to any other locales. Then, a player would typically consume an item called "Cool Drink" (this item negates the heat from the Volcano locale) and then starts to mine for charms with a good amount of pickaxes. After going through the entire map, a player would finish the quest and wait for the results. The following is an example of the reward screen for charms.

Fig 2.7 The Rewards Screen for Charms in Monster Hunter Portable 3rd

There are three distinct kinds of charms. First is a mystery charm signified by the color white. A mystery charm is known to give the lowest ranking talismans which are pawn talisman and bishop talisman. Second is a shining charm signified by the color yellow. A shining charm is known to give a middle tier talisman which are knight talisman and rook talisman. Third is a timeworn charm signified by the color red. A timeworn charm is known to give an upper tier talisman which are queen talisman, king talisman, and dragon talisman.

D. Graph

A graph is a data representation that is shown with a combination of vertices that are connected to edges. A graph is typically used to represent discreet objects and the relations between said objects. As a result, a graph has multiple real-life applications and uses. For example, a graph can be used to model computer networks with the computers being the vertices and the communication links being the edges. The following is the mathematical approach to the definition of graphs.

$$Graph\ G = (V, E)$$
$$V = Collection\ of\ Vertices = \{v_1, v_2, \dots, v_n\}$$
$$E = Collection\ of\ Edges = \{e_1, e_2, \dots, e_n\}$$

Historically, graph theory was born during the year 1736 with a well known mathematical problem known as the Seven Bridges of Königsberg. This problem is defined by an intricate set of bridges in the old Prussian city of Königsberg. These set of bridges are made to accommodate different landmasses and the existence of a river separating the landmasses apart. The following is the layout of the Seven Bridges of Königsberg.

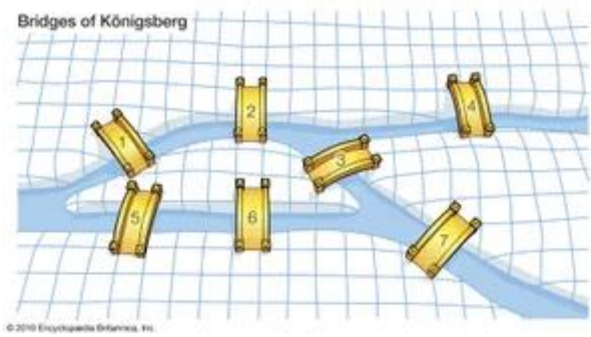


Fig 2.8 Seven Bridges of Königsberg
Source: https://www.britannica.com/science/Konigsberg-bridge-problem

The Seven Bridges of Königsberg problem is shown with seven bridges and four different landmasses. There are two central landmasses connected to each other with the third bridge, the top landmass that is connected to the first middle landmass with the first and second bridge and the rightmost middle landmass with the fourth bridge, and the bottom landmass that is connected to the first middle landmass with the fifth and sixth bridge and the rightmost middle landmass with the seventh bridge. The problem arises with the question of whether it is possible for a citizen to take a walk throughout the town of Königsberg while only crossing each and every bridge exactly once. This problem is then solved by a mathematician known as Leonhard Euler using what is now known as the graph theory.

There are multiple ways to model a graph. The different types of graphs are as follows.

1. Simple Graph

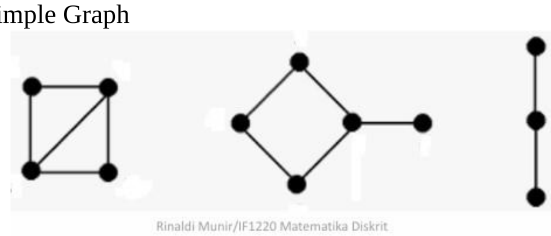


Fig 2.9 Simple Graph Examples
Source:
https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

A simple graph is a type of undirected graph without any loops nor multiple edges connecting the same pair or vertices.

2. Unsimple-Graph

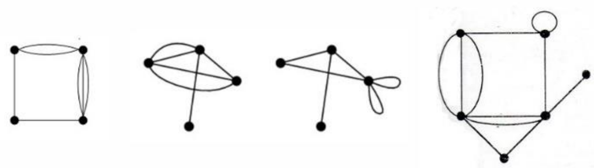


Fig 2.10 Unsimple Graph Examples
Source:
https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

An unsimple graph is a type of undirected graph that has loops or multiple edges that connect to the same pair of vertices. Unsimple graphs are the opposite of simple graphs.

3. Multi-Graph

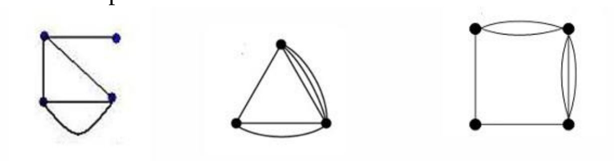


Fig 2.11 Multi-Graph Examples
Source:
https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

A multi-graph is a type of graph that specifically features at least one pair of vertices that have two edges connected to them.

4. Pseudo-Graph

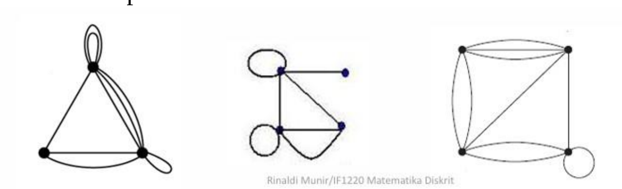


Fig 2.12 Pseudo-Graph Examples
Source: Charm grinding is a term used to describe the act of a player repeatingly doing certain quests only for the sake of acquiring charms. Charms themselves are items in Monster Hunter Portable 3rd that can be appraised at the end of a quest to acquire talismans which is used to raise the skill points of certain skills. The concept of charms and talismans were introduced in the 3rd generation of Monster Hunter games and has been a mainstay since then. Charms and talismans offer a new random variable into the game with the type of charm a player would get depends completely on luck and the amount of skill points as well as the skill points that are provided with a talisman is random with the values of the talisman being set the moment a quest ended. This makes attempting to get certain talismans a long endeavor as it tests the player's patience and luck.

A pseudo-graph is a type of graph that specifically features an edge that forms a loop. This means the edge that typically would connect to two different vertices would instead connect to a single vertex.

5. Undirected Graph

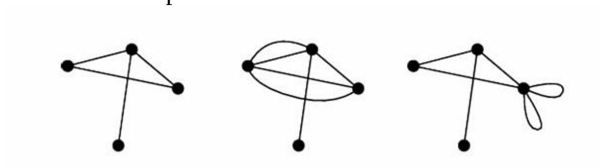


Fig 2.13 Undirected Graph Examples
Source:
https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

An undirected graph is a graph in which the edges that are in the graph lack a direction (for instance, a direction from a vertex to a different vertex).

6. Directed Graph

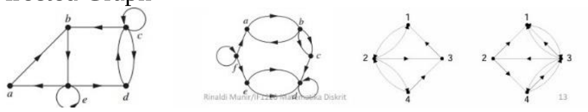


Fig 2.14 Directed Graph Examples
Source:
https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

A directed graph is a graph in which the edges that are in the graph have a direction signified by the usage of arrow lines to define the edges.

D. Dijkstra's Algorithm

Dijkstra's Algorithm is an algorithm used to solve single-source shortest path problems. To be more specific, Dijsktra's Algorithm is an algorithm used in tandem with graph theory, specifically using a weighted graph, in order to solve single-source path problems by finding the shortest distance between two vertices on a graph. In this context, single-source path means that one vertex within the graph is picked as the starting node and Dijkstra's Algorithm works to find the shortest path from the starting node to the other vertices. Dijkstra's Algorithm is created by a Dutch computer scientist by the name of Edsger W. Dijkstra in 1956.

The way Dijkstra's Algorithm works is by starting from the vertex that has been chosen. Then, initialize the distances of every other vertex as infinity and the starting vertex or node as zero. Then, check the distance between the current vertex and the neighboring vertex in order to find the shortest distance. Pick the vertex that has the smallest amount of weight before updating the distance. Then, mark the vertex that has been visited to avoid traversing through the same vertex. Lastly, repeat this process until all vertices have been visited.

IV. CALCULATIONS

A. Method of Research

This research that is covered by this paper is research to determine the most optimum route in charm grinding in Monster Hunter Portable 3rd. The most optimum route would be the route that would take the least amount of distance which would also be the route that would take the least amount of time. The proposed solution within this paper is to use Dijkstra's Algorithm in order to traverse through certain routes with the least amount of distance. Dijkstra's Algorithm is applied to a graph that is representative of the locale that is used in this research. Due to the amount of factors that may derail this research, some limitations are applied in order to stay within the topic of this research. The limitations are as follows.

1. This research will only cover the Volcano locale of Monster Hunter Portable 3^{rd}. The reason for this is due to the drop rates of rare charms which are higher in the Volcano area. This means the Volcano locale is the most optimal as well as the most well known area for players to obtain rare charms.
2. This research will not account for both small and large monsters that are available within a locale. High Rank Gathering Quests have a large monster designated for each locale and depending on where they are in the locale, they might affect the time efficiency when grinding for charms. Similar to large monsters, small monsters that are available within a locale might affect the time efficiency when grinding for charms depending on the nature of the small monsters.
3. This research will use the assumption that the stamina bar on a player will not decrease. This can be done with the usage of either Power Juice or Mega Juice (both items stop the depletion of the stamina bar for a certain amount of time). The reason for this is to ensure the charm grinding process would go as fast as possible.
4. This research will not account for the amount of time required to go to mining nodes within an area. The reasoning for this is that although certain mining nodes may take longer to reach from certain starting places within an area, the total amount of time would only amount to one to three seconds.
5. This research will not account for the amount of time required to mine in all the available mining nodes. The reason for this is that the time needed to mine within all the available mining nodes is a constant as a player will always attempt to mine in all the available mining nodes within an area. This is because the goal of charm grinding is to mine as much as possible to acquire the most amount of charms possible.
6. This research will use a modified version of Dijkstra's Algorithm. This modification allows the algorithm to have multiple different routes if there are two or more edges with the same amount of weight that are connected to one vertex or node. This modification also allows the algorithm to only traverse through areas that have been traversed if there is no other connection between the current area and the next area. The algorithm will still only traverse through vertices once as explained in the next section.

B. Graph Implementation

A graph is required to implement the Dijkstra's Algorithm. The graph is a graph that is representative of the Volcano locale in Monster Hunter Portable 3^{rd}. The following is a representation of the Volcano locale as a graph.

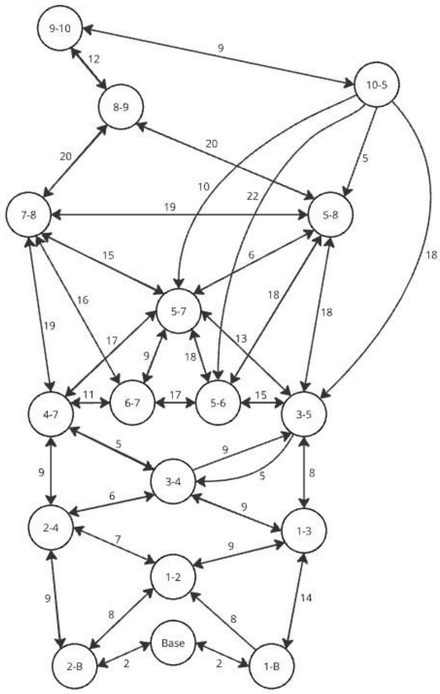


Fig 4.1 Volcano Locale as a Weighted Directed Graph

This graph uses the transition between areas as vertices and the areas as edges with the weight of the edges being the average time required to traverse through the area. This is because every area may have different area traversal times depending on which starting point a player starts from and which area a player is going to. For example, the time required to go from Area 2 to Area 4 may differ depending on the starting point a player starts from. For instance, the time from 2-B to 2-4 is different when compared to 1-2 to 2-4. Other than the weight, the graph also uses edges to define an area. For example, the edges 2-B, 1-2, and 2-4 create Area 2.

This graph uses a particular naming convention to represent the Volcano locale. The naming convention for the vertices uses the format of "Number 1 – Number 2" in which both numbers signifiy the area name. For instance, 2-B means the vertex represents the connection between area 2 and Base. The nodes also have a meaning as a starting point of an area depending on the last visited node. For instance, if a player visited the 1-2 node right before visiting the 2-4 node, this means the player traverses the route from Area 2 to Area 4 with the player starting in front of the transition between Area 2 and Area 1.

The weight values within this graph signifies the time spent traversing through an area in a certain manner. This is because the way a player traverses through an area may differ depending on the starting point and the end point. For instance, the time required to visit Area 4 from Area 2 may vary depending on the starting point. If a player starts from the front of the connection between Area 2 and Base, traversing to area 4 takes 9 seconds. Meanwhile, if a player starts from the front of the connection between Area 2 and Area 1, traversing to area 4 takes 7 seconds.

Certain vertices within the graph would logically not be visited by players. For instance, the node 3-4 will not be visited due to how a player would logically go for a different route when attempting to go to Area 3 rather than going to Area 4 in order to do so. Another example is both the nodes 6-7 and 5-6 with both nodes connecting Area 7 and Area 5 to Area 6. Area 6 lacks any mining nodes and due to how Area 7 and 5 are already connected with a different node, it is pointless to go to Area 6. Due to this logic, this graph can be simplified by not including certain areas that will not be visited by players. The following is the simplified representation of the Volcano locale in the form of a graph.

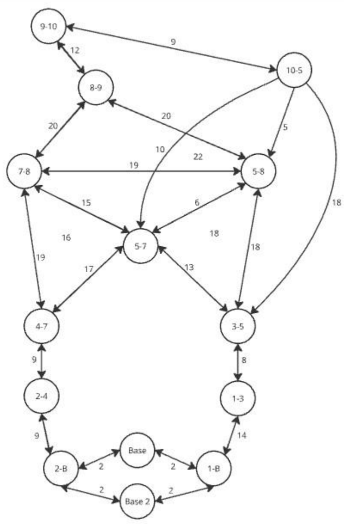


Fig 4.2 A Simplified Weighted Directed Graph of the Volcano Locale

This is the simplified graph that represents the Volcano locale. The connection between Area 1 and Area 2, Area 3 and Area 4, and the connections related to Area 6 have been deleted. This results in a more simplified graph that maintains the key area transitions needed to ensure a player traverses through all the important areas (areas that have at least 1 mining node). Due to how a player has to return to the base camp area in order to finish the Gathering Quest, an additional node is added which is Base 2 which represents the final node that needs to be traversed. With all the changes that have been made, this is the graph that will be used within the implementation of Dijkstra's Algorithm.

C. Algorithm Implementation

In order to optimize the efficiency in charm grinding, the most optimum route to traverse a locale must be found. This is because the most optimum route with the lowest amount of weight would amount to a lower overall time traversing through the locale. This allows for the usage of Dijkstra's Algorithm to traverse through areas within a locale with the lowest amount of weight possible. The algorithm used in this implementation is a modified version of Dijkstra's Algorithm as explained previously within the sixth limitation. The following is the implementation of the modified Dijkstra's Algorithm.

1. Start from Base as the starting node or vertex. Initialize the route with Base.
   Current route: {Base}
2. Traverse from Base to 1-B or 2-B. Create 2 different routes as both edges connecting to both vertexes have the same weight.
   Route 1: {Base, 1-B}
   Route 2: {Base, 2-B}
3. 2-B to 2-4 and 1-B to 1-3 due to both vertexes only having 1 edge moving forward.
   Route 1: {Base, 1-B, 1-3}
   Route 2: {Base, 2-B, 2-4}
4. 2-4 to 4-7 and 1-3 to 3-5 due to both vertexes only having 1 edge moving forward.
   Route 1: {Base, 1-B, 1-3, 3-5}
   Route 2: {Base, 2-B, 2-4, 4-7}
5. 4-7 to 5-7 and 3-5 to 5-7 since both edges are the edges with the smallest weight that is connected to each vertex.
   Route 1: {Base, 1-B, 1-3, 3-5, 5-7}
   Route 2: {Base, 2-B, 2-4, 4-7, 5-7}
6. 5-7 to 5-8 for both routes since the edge has the smallest weight.
   Route 1: {Base, 1-B, 1-3, 3-5, 5-7, 5-8}
   Route 2: {Base, 2-B, 2-4, 4-7, 5-7, 5-8}
7. 5-8 to 7-8 or 8-9 depending on the route. Route 1 traverses through 7-8 as the edge has the smallest amount of weight. For Route 2, it traverses through 8-9 due to the sixth limitation that has been previously mentioned.
   Route 1: {Base, 1-B, 1-3, 3-5, 5-7, 5-8, 7-8}
   Route 2: {Base, 2-B, 2-4, 4-7, 5-7, 5-8, 8-9}
8. 7-8 to 8-9 and 8-9 to 9-10 and 7-8 to 8-9 as both routes only have 1 edge moving forward.
   Route 1: {Base, 1-B, 1-3, 3-5, 5-7, 5-8, 7-8, 8-9}
   Route 2: {Base, 2-B, 2-4, 4-7, 5-7, 5-8, 8-9, 9-10}
9. 8-9 to 9-10 and 9-10 to 10-5 as both routes only have 1 edge moving forward.
   Route 1: {Base, 1-B, 1-3, 3-5, 5-7, 5-8, 7-8, 8-9, 9-10}
   Route 2: {Base, 2-B, 2-4, 4-7, 5-7, 5-8, 8-9, 9-10, 10-5}
10. 9-10 to 10-5 and 10-5 to 3-5 since both routes only have 1 edge moving forward.
    Route 1: {Base, 1-B, 1-3, 3-5, 5-7, 5-8, 7-8, 8-9, 9-10, 10-5}
    Route 2: {Base, 2-B, 2-4, 4-7, 5-7, 5-8, 8-9, 9-10, 10-5, 3-5}
11. 3-5 to 1-3 as route 1 no longer has a vertex to traverse to and route 2 only has 1 edge moving forward.
    Route 2: {Base, 2-B, 2-4, 4-7, 5-7, 5-8, 8-9, 9-10, 10-5, 3-5, 1-3}
12. 1-3 to 1-B as route 2 only has 1 edge moving forward.
    Route 2: {Base, 2-B, 2-4, 4-7, 5-7, 5-8, 8-9, 9-10, 10-5, 3-5, 1-3, 1-B}
13. 1-B to Base 2 as route 2 only has 1 edge moving forward.
    Route 2: {Base, 2-B, 2-4, 4-7, 5-7, 5-8, 8-9, 9-10, 10-5, 3-5, 1-3, 1-B, Base 2}

Due to how route 1 is incomplete as it failed to reach Base 2, the results only cover route 2. Route 1 is incomplete due to the concept of charm grinding which forces players to visit all areas that have mining nodes. Those areas are as shown in Fig 2.1 with the areas being Area 3, 5, 7, 8, 9, and 10. Within route 1, Dijkstra's Algorithm alone is incapable of traversing through more vertices as all other vertices that are connected to the 10-5 vertex have already been visited.

The following is the visualization of route 2 in the form of a graph.

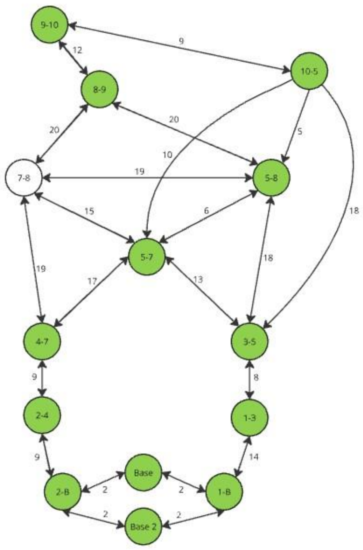


Fig 4.3 Graph Representation of the Route Taken

The following is the route, areas traversed, and time spent traversing through said areas using the modified Dijkstra's Algorithm.

Route: {Base, 2-B, 2-4, 4-7, 5-7, 5-8, 8-9, 9-10, 10-5, 3-5, 1-3, 1-B, Base 2}

Areas: {Base Camp, Area 2, Area 4, Area 7, Area 5, Area 8, Area 9, Area 10, Area 5, Area 3, Area 1, Base Camp}

Time: 126 seconds (2 minutes and 6 seconds)

V. RESULTS

The results that this research produce is that Dijkstra's Algorithm is capable of providing the best route to take in order to maximize time efficiency for charm grinding. However, Dijkstra's Algorithm alone will not provide an efficient solution as seen with the route taken by Route 2 within the previous section. The seventh step in the implementation of Dijkstra's Algorithm shows Route 2 taking 8-9 instead of 7-8 due to the limitations that have been imposed for the logical sake of this research. According to Dijkstra's Algorithm, the vertex 7-8 should have been taken due to the weight of the edge connecting 5-8 to 7-8 being smaller as opposed to 8-9. However, going to 8-9 is not a logical move as Route 2 has traversed through that area and especially has also passed by the mining node beforehand.

VI. CONCLUSION

In conclusion, Dijkstra's Algorithm is beneficial to find the shortest route in charm grinding. This is due to how it is possible to represent the locales in Monster Hunter Portable 3rd as weighted-directed graphs. However, Dijkstra's Algorithm should not be used by itself without any additions to the algorithm. The reason for this is as seen in the previous section, Dijkstra's Algorithm is incapable of handling the complexities of the graph that has been made from the Volcano locale in Monster Hunter Portable 3rd. The main issue is due to the way Monster Hunter map works, it is impossible to create a graph with the areas themselves as the vertices and the area transitions as the edges. The lack of a better graph structure means that Dijkstra's Algorithm becomes harder to use as the current graph structure may mislead the algorithm into traversing through unimportant vertices.

Fixing this issue would require a usage of Dijkstra's Algorithm in tandem with a different set of rules and algorithms. This is to ensure that Dijkstra's Algorithm will return the shortest path rather than a path that is misleading. For instance, Dijkstra's Algorithm can be used with a certain algorithm that would update the actual areas within the locale itself in order to help direct the algorithm to picking the more important vertices rather than blindly picking the route with the smallest amount of weight. These would ensure the routes picked by the algorithm will result in the path with the smallest total weight possible.

VII. ACKNOWLEDGMENT

First of all, the author would like to thank God for his blessings throughout the workings of this paper so that the author is capable of finishing this paper. The author would also like to thank Mr. Arrival Dwi Sentosa, S.Kom., M.T. as the lecturer of ITB Discreet Mathematics IF1220 for the materials that have been given throughout this entire semester. The author would also like to thank Mr Rinaldi Munir for the materials that have been shared on his website. Finally, the author would also like to thank his family and friends for their support throughout the process of writing this paper.

REFERENCES

[1] R. Munir, *Graf (Bag. 1)*, 2024. Retrieved 21:00, January 7, 2025, from https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

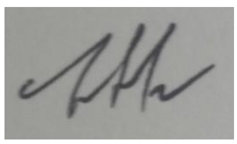
[2] DSA Dijkstra's Algorithm. Retrieved 21:00, January 7, 2025, from https://www.w3schools.com/dsa/dsa_algo_graphs_dijkstra.php

[3] What is Dijkstra's Algorithm? | Introduction to Dijkstra's Shortest Path Algorithm. Retrieved 21:00, January 7, 2025, from https://www.geeksforgeeks.org/introduction-to-dijkstras-shortest-path-algorithm/

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, 26 Desember 2024



Hafizh Hananta Akbari - 13522132