Pathfinding through Disneyland

Maximizing Time in a Park Using Route Planning and Decision Making Algorithms

Muhammad Atthaumar Rifqy - 13519148
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail: matthaumar@gmail.com

***Abstract*—As the popularity and size of an amusement park increases, the issue of how best to spend one's time in the limited frame of a park's open hours arises in tandem. In this paper, we will be implementing several route planning algorithms in order to maximize the amount of attractions an average person can experience in a single day within an amusement park, in this case Hong Kong Disneyland. Algorithms we will be using are branch and bound, uniform cost search, greedy best first search, A*, and dynamic programming. It is the hope of the author that this paper will serve as an interesting and entertaining analytical exercise for both the author and the reader and that this paper will leave the reader with an interest in the applications of various route pathing algorithms for a variety of different purposes, including the topic of this paper.**

***Keywords*—route planning, decision making, dynamic programming**

I. INTRODUCTION

Although it is difficult to imagine large crowds or public entertainment in a post-pandemic world, people and families from around the world – numbering in the thousands – often used to take the opportunity afforded by holidays to flock to centers of entertainment such as amusement parks. An amusement park is a park that features attractions such as rides or games as well as other events for entertainment. A specialization of amusement parks is the theme park, an amusement park that bases its attractions around a central theme and is often split into multiple areas with different subthemes. Unlike other attraction-offering locations such as carnivals which are mobile and temporary, amusement parks are stationary and built for long-lasting operation.

Before amusement parks, recreation centers such as fairs of the Middle Ages, pleasure gardens, and large picnic areas had existed for centuries. These, along with World's fairs and international expositions, influenced the emergence of the amusement park industry [1]. Many modern amusement parks evolved from earlier pleasure resorts that had become popular with the public for day trips or weekend holidays and in the United States, some amusement parks grew from picnic groves established along rivers and lakes that provided bathing and water sports, such as Lake Compounce which is considered the oldest, continuously-operating amusement park in North America. The first theme parks emerged in the mid-twentieth century with the opening of Santa Claus Land in 1946 [2], Santa's Workshop in 1949, and Disneyland in 1955 [3]. The topic of our paper, Hong Kong Disneyland was announced in 1998 by the Walt Disney Company and the government of Hong Kong as the second theme park in Asia and opened in 2005.

Fig. 1. Hong Kong Disneyland

Hong Kong Disneyland (also known as HKDL) consists of seven themed areas: Main Street, USA; Fantasyland; Adventureland; Tomorrowland; Grizzly Gulch; Mystic Point; and Toy Story Land, each with their own attractions and entertainment, ranging from nearly two dozen in areas like Adventureland to just three attractions in Mystic Point. Not all of these attractions are a "must-see", with many of them being smaller events. HKDL has a daily capacity of 34,000 and hosts 6 to 7 million visitors annually [4].

With such a wide variety of things to experience as well as the amount of people waiting to experience them, many attractions have large queues and wait times, increasing the amount of time one can spend at any given attraction. This has led many visitors unable to experience all that HKDL has to offer in a single trip. In the author's personal experience, he has still not experienced everything in HKDL despite having made two trips. This presents a unique problem: How do we optimize a trip HKDL, in this case by experiencing as many attractions and events as possible within the park's opening hours while also experiencing all the must-see attractions in the park? This is the problem that we will attempt to answer using route-planning and decision-making algorithms. Before we begin the analysis, we will first outline and explain the basics of each of the algorithms we will be using.

II. THEORETICAL FRAMEWORK

A. *Branch and Bound Algorithm*

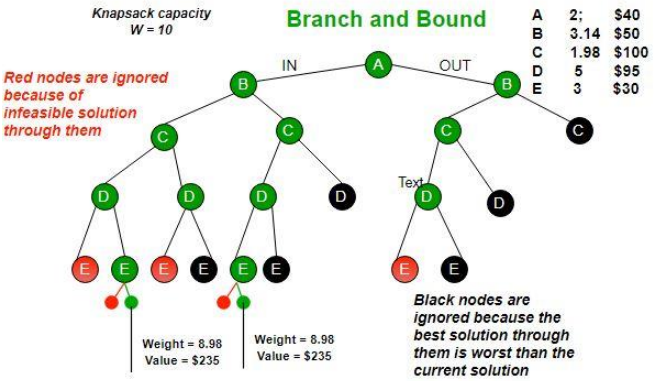

Fig. 2. Branch and Bound example

The Branch and Bound (BnB) algorithm is an algorithm design paradigm used for problems involving combinatorial and discrete optimization. This algorithm produces a solution *x* that minimizes or maximizes an objective function *f(x)* from a set of possible solutions *S* without violating a problem-specific constraint [5].

The algorithm envisions the problem as a dynamic state space tree, with each vertex representing a state (e.g. a path of nodes taken through a graph or a specific 1-0 configuration). Beginning at a singular state as a root with a beginning cost, the algorithm expands (branches) the root into all other possible states, assigning a cost to each. Before any other vertex is expanded, the algorithm compares the costs of each expanded state with a heuristic constraint defined by the problem (the bound). If a vertex violates this bound, that vertex cannot generate an optimal solution and is therefore "pruned" and will never be expanded. After the root has been pruned, the algorithm expands the vertex with the lowest cost and continues the process until a vertex containing state *x* which satisfies *f(x)* is expanded, which means that the solution is found.

The general algorithm for the branch and bound paradigm is as follows:

```
B <- problem_heuristic()
Q <- initialize_queue()
x <- root
while Q not empty and not solution(x) do
        for each Ni branch_of(x) do
                if (Ni) satisfies B then
                        enqueue(Ni)
                endif
        endfor
        x <- min_cost(Q)
        dequeue(x)
endwhile
        if solution(x) then
                write(x)
        endif
```

B. *Route-Planning Algorithms*

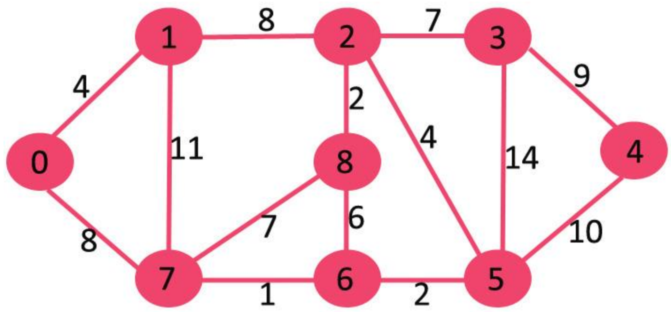

Fig. 3. Weighted graph commonly used to visualize route planning algorithms

Route/path planning algorithms are algorithms that are used to find the shortest route or path between two points. In general, route planning algorithms can be categorized as uninformed and informed searches. An uninformed search algorithm only makes decisions based on the paths that have been traveled, while an informed search uses a heuristic centered on the goal. We will be exploring three different route planning algorithms, which are the uniform cost search, the greedy best first search, and the A* algorithm. These algorithms will use weighted graphs as examples.

1) *Uniform Cost Search*

The Uniform Cost Search (UCS) is an uninformed route-planning algorithm which expands a node with the shortest path taken within that iteration of the search. This algorithm begins by expanding the start node and assigning a cost g(n) to each node n connected to the start node which is the value of the weighted edge connecting n to the expanded node. Each node is considered a live node and are queued in a priority queue in ascending order by the value g(n). The node in the head of the queue is always the node with the shortest path taken to a node from the start node so far. The head of the queue (N) is then expanded and each node n connected to it are assigned a g(n) value which is the value of the g(N) + weight of N to n. These live nodes are then placed in the priority queue as well and the process repeats until a solution is found.

The general algorithm for the UCS is as follows:
[FILL LATER]

2) *Greedy Best First Search*

The Greedy Best First Search (GBFS) is an informed route-planning algorithm which expands a node n based on a heuristic value h(n) which is an estimated cost for each node relative to the goal node. [EXPAND LATER]

The general algorithm for the GBFS is as follows:
[FILL LATER]

3) *A* Algorithm*

The A* Algorithm is an informed route-planning algorithm which can be viewed as an improvement and combination of both the UCS and GBFS algorithms. [EXPAND LATER]

The general algorithm for the A* Algorithm is as follows: [FILL LATER]

C. *Dynamic Programming*

III. Solution

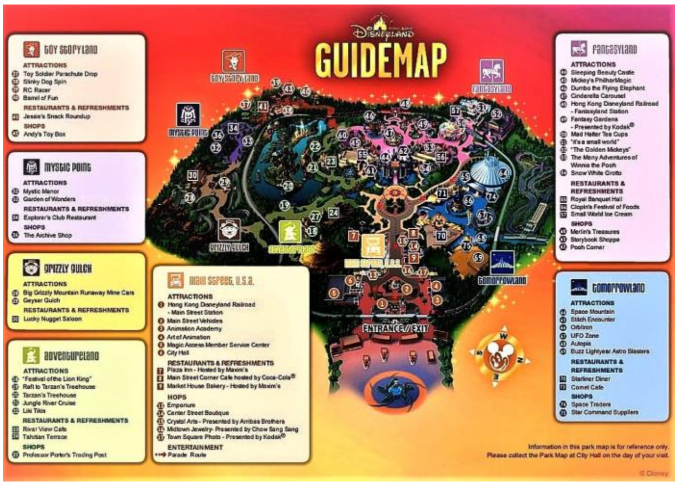

Fig. 4. Map of Hong Kong Disneyland

Before we utilize the algorithms, we must first break down the unique problems we must solve for our experiment to be a success:

1. A visitor must visit the major attractions. For this example, we will be using a list of recommended attractions as provided by timeout.com [6].
2. The total time within the park must not exceed the closing time of the park (9pm) but must not be less than a specified time before the Paint the Night parade (8.30pm). For this example, we will set this as 1 hour before the parade, therefore the total length of time within the park must end between 7.30pm to 9pm.
3. A visitor must visit and/or experience as many attractions and events as possible.

To solve these unique problems, we can use multiple algorithms. A simple breakdown of the solution is as follows:

1. Break down the amusement park into a simple estimated weighted graph of major attractions.
2. Determine the optimum actual path of each edge in the simple graph.

First, we can envision a graph with each major attraction as a node and the edges being the Euclidean distance between directly connected sections areas, i.e. two major attractions within the same area or within two adjacent areas. Included in this graph is the main entrance. A simplified graph of such is present in Figure 5 below.

[INSERT SIMPLE GRAPH]

Fig. 5. Simplified graph of Hong Kong Disneyland's major attractions

As this is a weighted connected graph, we can solve the first problem by viewing it as a simple Travelling Salesman Problem which starts at the Main Entrance. This problem can be solved using a Branch and Bound algorithm to determine the shortest possible path through the park.

However, each edge is not an actual path from one attraction to another. To find the actual path, we can use a route-planning algorithm.

We will use a heuristic to determine whether the chosen path is sufficient. First, we determine the fractional "weight" of the estimated path on the simple graph. Afterwards, we calculate the upper and lower bounds of the time spent in that path by multiplying said weight with the upper and lower total time a visitor can spend in the park. For example, if a visitor arrives at opening time, which is 10.30am, they have a lower bound of 10.30am – 7.30pm or 9 hours and an upper bound of 10.30am – 9pm or 10.5 hours. After multiplying the weight with these times, we will have an actual upper and lower bound.

[INSERT SIMPLE GRAPH]

Fig. 6. Example of an actual path from one attraction to another

Another alternative is to use dynamic programming. [EXPAND LATER]

IV. Conclusion

It is possible to determine an adequate path within an amusement park using various algorithms. However, this plan can be further improved using dynamic programming.

Acknowledgment

The author would like to thank

References

[1] J. Adams, The American amusement park industry. Boston: Twayne Publishers, 1991.

[2] "The History of Santa Claus, Indiana", www.hohoholdings.com. [Online]. Available: https://web.archive.org/web/20100127094024/http://www.hohoholdings.com/schistory.htm. [Accessed: 09- May- 2021].

[3] A. Onion, M. Sullivan and M. Mullen, "Disneyland opens", HISTORY, 2009. [Online]. Available: https://www.history.com/this-day-in-history/disneyland-opens. [Accessed: 09- May- 2021].

[4] C. Chan, "Disney ups intake for new year", thestandard.com.hk, 2006. [Online]. Available: https://web.archive.org/web/20071015072613/http://thestandard.com.hk/news_detail.asp?pp_cat=11&art_id=34507&sid=11427047&con_type=1&d_str=20061220&sear_year=2006. [Accessed: 09- May- 2021].

[5] R. Munir, "IF2211 Strategi Algoritma - Semester II Tahun 2020/2021", Informatika.stei.itb.ac.id, 2021. [Online]. Available: https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/stima20-21.htm. [Accessed: 10- May- 2021].

[6] [6]O. Lai, "Best things to do at Hong Kong Disneyland", Time Out Hong Kong, 2019. [Online]. Available: https://www.timeout.com/hong-

kong/things-to-do/best-things-to-do-at-hong-kong-disneyland. [Accessed: 11- May- 2021].

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 April 2021

Muhammad Atthaumar Rifqy - 13519148