**Pokémon Go Maximal Resource Collecting and Routing Problem**

**Final Project Report**

Student id: R78051012

Name:蕭維慶

1. Introduction

With the mobile gaming market growing a lot over these years, people would most likely expect to paly the games as well as possible. Due to the busy life, people can not just get to play all day to achieve their expectation. They need to think how to play.

Pokémon Go is a most popular game recently. It has achieved a number of records since its debut – the most downloaded app in its first week ever, the fastest to reach 50 million installs on Google Play and surpassed $500 million in worldwide customer spending across iOS and Android in just over 60 days[1]. The game developed by [Niantic](https://en.wikipedia.org/wiki/Niantic,_Inc.) company. In the game, players use a mobile device's [GPS](https://en.wikipedia.org/wiki/Global_Positioning_System) capability to locate, capture, battle, and train virtual creatures ([Pokémon](https://en.wikipedia.org/wiki/List_of_Pok%C3%A9mon)), who appear on the screen[2]. Whatever locate, capture battle or train need resources. The player need to have the resources like poke ball to catch the [Pokémon](https://en.wikipedia.org/wiki/List_of_Pok%C3%A9mon) or make the lower health [Pokémon](https://en.wikipedia.org/wiki/List_of_Pok%C3%A9mon) cured by medicine resource. The only way for player to have the resource is close to the PokéStops which is the attraction on the map and provide players with resource items, such as eggs, [Poké Balls](https://en.wikipedia.org/wiki/Pok%C3%A9_Balls), berries, and potions. Previous research investigated how to route all the given PokéStops. The problem is, even player pass all the given PokéStops, the player will not get enough resource in time. If the player can just pass some of the PokéStops near by and achieve the objective, player can save time. The difficulty is when player go through the PokéStop, it will refresh and is no longer accessible in 5 minutes. This research is going to investigate how to get more resources in time constrained according those game rule. We use the gurobi and python to model the problem and optimize.

The rest of this paper is organized as follows. In section 2, there are some realted investigation. In section 3 we propose the research method. In section 4, we report computational results on test problems. In section 5, we give our conclusions.

1. Related Research

Mathematicians at the University of Waterloo have been calculating the fastest tour of various metropolitan areas like Boston, Cincinnati, and San Francisco, hitting every PokeStop in the city in the shortest possible time. Many of these PokeTours are very long. Cincinnati for instance has 551 PokeStops, and it takes 223 miles of walking to hit them all. Boston has 518 Pokestops, which can be done in 218 miles. San Fransisco's tour is much smaller, at 65 miles, but that will still take a very long time and your backpack has an upper limit on Pokeballs anyway.

Currently, the page also has maps of Denver, Roanoke, the University of Missouri, Kansas State University, and Champaign, Illinois. Presumably more maps will be added soon, so be patient if your city isn't on the list. Or just go outside and find a tree to appreciate, or something[3].

1. Modeling of [Pokémon](https://en.wikipedia.org/wiki/List_of_Pok%C3%A9mon) Go Resource Collecting

There are several game rules will influence player collect the resource in between those PokeStops. First is delay time: when the player visited the PokeStop, PokeStop would be inaccessible in 5 minutes for the player. Second, space liminted resource bag, player’s bag have limited space to put the resource and the default is 250. Third is bonus point, when player go through 10 difference PokeStops in 30 minutes and must pass a new PokeStop in each 10 minutes, player would get much bonus resource(in my experience, default is 2 to 6, bonus is 6 to 10).

Notation:

* N: Given N location, i = 1~N
* T: Given T time, t = 1~T
* : Award on location i
* , almost 5
* : travel time from i to j

This project is going to solve the Maximal resource routing in time problem and consider the delay time rule. We formulate the it as an MIP, and we are able to use this formulation to solve problems. Let N to be the given location number and T to be the number of time unit. is the award player will get on i location after player visited i. Set randomly given from 2 to 6. Set to be delay time on each location i, it’s almost be 5. Let equal the travel time from location i to location j (i = 0 is the depot). Set = 1 if player collected award on location i at time t and = 0 otherwise. The decision variable equals 1 if player depart location from I at time t to location j at time and equals 0 otherwise. Using this notation, the objective function and constraints of the MIP are as follows:

|  |  |  |
| --- | --- | --- |
| Objective: |  | 3. 1 |
| Subject to: | = 1 | 3. 2 |
|  | = 1 | 3. 3 |
|  | i, t | 3. 4 |
|  |  | 3. 5 |
|  |  | 3. 6 |
|  |  | 3. 7 |
|  |  | 3. 8 |

The objective function 3. 1 maximize the total resource get from each visited and collected location. In 3. 2 and 3. 3, we require that the depot be visited at time 0 and end to depot at time T. the time is not necessarily from time T-1 to time T. Constraints 3. 4 are classical flow conservation constraints. In 3. 5, we require that there are only one flow out at each time t.. Constraints 3. 6 determine the edges have been arrived location i, otherwise cannot collect. Constraints 3. 7 determine that if collect the reward on location i at time t, player would not collect reward on location I before delay time t+d[i]. Constraint 3. 8 determine that the collected resource can not exceed 250 units.

1. Computational Experiments
2. Dataset and preprocessing

There are several datasets include ncku, nctu, nthu, and thu. Those datasets include Latitude and longitude degree.

Table 1 Dataset number

|  |  |
| --- | --- |
| Dataset name | Number |
| Ncku | 198 |
| Nctu | 78 |
| Nthu | 99 |
| Thu | 124 |

We use the Euclidean distance to calculate the Latitude and longitude degree from each node and multiply 110 for converting degree to kilometer. We consider that each person walks 70 meter per minutes. Therefore, 1 unit = 1 minute = degree \* 110 \* 1000 / 70. Calculate the travel time between each node to create adjacency matrix. If the distance unit less than 1, let it be 1.

1. Small dataset:

I use the small ncku dataset which include time = 10 and n = 3 to test whether the model is correct or not. Table 2 show the award on each node. Table 3 is the travel time matrix between each location. Figure 1is the routing result. Figure 2 is the routing collected point path on google map

Table 2 award on each node

|  |  |
| --- | --- |
| location | Award number |
| 0 | 2 |
| 1 | 2 |
| 2 | 4 |

Table 3 travel time matrix between each location

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 0 | 0 | 3 | 3 |
| 1 | 3 | 0 | 1 |
| 2 | 3 | 1 | 0 |

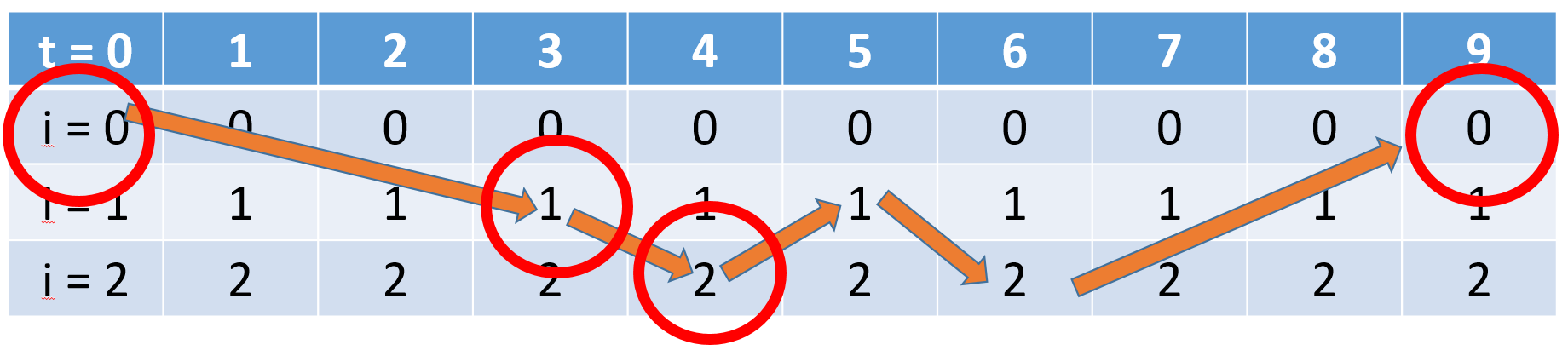


Figure 1 Routing Result

Objective value: 10

Edge result(X): [(0, 1, 0), (1, 2, 3), (2, 1, 4), (1, 2, 5), (2, 0, 6)]

Routing point: [[0, 1], [1, 2], [2, 1], [1, 2], [2, 0]]

Collected point(Y): [(0, 0), (1, 3), (2, 4), (0, 9)]

Routing collected point: [[0, 1], [1, 2], [2, 0]]



Figure 2 Routing collected point path on google map

1. Big dataset

We use the ncku dataset and set Time = 30 and location number = 30. Figure 3 routing collected point path on google map.

Calculated time: 100 seconds

Objective value: 148

Edge result(X): [(0, 15, 0), (15, 28, 1), (28, 24, 2), (24, 6, 3), (6, 18, 4), (18, 19, 5), (19, 16, 6), (16, 3, 7), (3, 6, 8), (6, 18, 9), (18, 19, 10), (19, 16, 11), (16, 3, 12), (3, 6, 13), (6, 18, 14), (18, 19, 15), (19, 16, 16), (16, 3, 17), (3, 6, 18), (6, 18, 19), (18, 19, 20), (19, 16, 21), (16, 3, 22), (3, 14, 23), (14, 24, 24), (24, 22, 25), (22, 21, 26), (21, 15, 27), (15, 0, 28)]

Collected point(Y): [(0, 0), (15, 1), (28, 2), (24, 3), (6, 4), (18, 5), (19, 6), (16, 7), (3, 8), (6, 9), (18, 10), (19, 11), (16, 12), (3, 13), (6, 14), (18, 15), (19, 16), (16, 17), (3, 18), (6, 19), (18, 20), (19, 21), (16, 22), (3, 23), (14, 24), (24, 25), (22, 26), (21, 27), (15, 28), (0, 29)]

Routing collected point: [[0, 15], [15, 28], [28, 24], [24, 6], [6, 18], [18, 19], [19, 16], [16, 3], [3, 6], [6, 18], [18, 19], [19, 16], [16, 3], [3, 6], [6, 18], [18, 19], [19, 16], [16, 3], [3, 6], [6, 18], [18, 19], [19, 16], [16, 3], [3, 14], [14, 24], [24, 22], [22, 21], [21, 15], [15, 0], [0, 0]]



Figure 3 big dataset routing collected point path on google map

1. Conclusion

The content of this article was initially motivated by playing the mobile game – Pokemon Go and colleting the resource. So as to collect resources as soon as possible. We not only consider the routing distance but also the time consistency measure, based on time-classes, was defined. In this project, we introduced the model and solve the problem.

Experiments tend to demonstrate the great difficulty of this new problem, but efficiency of the modeling or solving still need to improve. An important result is that, at least for the (realistic) instances used for the computations, very consistent solutions could always be found with a relatively small impact on travel costs or reward collected number.

A clear development of this work is to provide exact model solve the Pokemon Go resource collecting problem. The integer programming formulation proposed in this article is not solved efficiently by the solver so it would be worth investigating more evolved modeling or combine other heuristic approach. From the application perspective, it can developed the app by using the model for player to use. Player can start the app and set the free time they have, then the app will calculate the optimal collecting path for player. In addition, other possible models of routing Pokmon Go PokeStop just consider about the routing path but this project might be more interesting. It considers more about the time class and time constrained.

Finally, we introduced a new routing problem for Pokemon Go in the paper. Given the inherent interest and difficulty of this problem, further research regarding the bonus problem and more scenario constrained like in practice the location service time is important and waiting time must be considered.

1. Reference

[1] S. Perez. (2016). *Pokémon Go becomes the fastest game to ever hit $500 million in revenue*. Available: <https://techcrunch.com/2016/09/08/pokemon-go-becomes-the-fastest-game-to-ever-hit-500-million-in-revenue/>

[2] Wikipedia. (2016). *Pokémon Go*. Available: <https://en.wikipedia.org/wiki/Pok%C3%A9mon_Go>

[3] A. Thompson. (2016). *A Wonderfully Over-Engineered Solution to the Problem of Pokemon Go*. Available: <http://www.popularmechanics.com/culture/gaming/a21843/traveling-salesman-problem-pokemon-go/>