パルテノ遺伝的アルゴリズムに基づく時間依存型テーマパークルーティング問題の解法

著者	張 洋
出版者	法政大学大学院理工学研究科
雑誌名	法政大学大学院紀要．理工学・工学研究科編
巻	61
ページ	1-8
発行年	2020-03-24
URL	http://doi.org/10.15002/00022893

パルテノ遺伝的アルゴリズムに基づく時間依存型テーマパークルーティング問題の解法

TIME-DEPENDENT THEME PARK ROUTING PROBLEM BY PARTHENO-GENETIC ALGORITHM

張　洋
ZHANG YANG
指導教員　李　磊

法政大学大学院理工学研究科応用情報工学専攻博士前期課程

As the remarkable improvement of people's living levels and interesting for entertainment, the theme park has become one of the most popular places for people to enjoy life. However, since the high popularity of theme parks, based on the reality circumstance of tourists, enjoying more attraction projects and decreasing the fatigue greatly can largely improve the satisfaction of tourists. Therefore, based on the network of Traveling Salesman Problem (TSP), we propose a time-dependent theme park routing problem where the walking time is time-dependent under the consideration of congestion and fatigue degree. The primary objectives are to maximize the number of visited attractions, satisfaction and minimize the queuing time and walking time. In this study, the general model for time-dependent theme park problem is formulated and two different algorithms are used to solve the model. The numerical experiments are conducted to verify the feasibility and effectiveness.

Key Words *: Theme Park Problem, Routing Problem, Time-Dependent, Partheno-genetic Algorithm*

１． Introduction

With the improvement of people's living standards and material level, people's daily life is no longer solely focused on the pursuit of food and clothing but includes more spiritual pursuits. Holiday outings have become a common form of entertainment for modern people. Among them, theme parks have become the best place for short-term travel[1]. Some famous theme parks such as Disney World in Tokyo and Universal Studios in Osaka attract millions of tourists each year, which demonstrates that with the improvement of people's living standards, people's demands for entertainment industries have also surged. However, several problems with theme parks need to be solved. For instance, for popular attractions, visitors need to endure long queues that might take two or three hours. Furthermore, because most theme parks have extensive areas and a large flow of people, tourists spend a lot of time walking and queuing. Therefore, quickly planning the best way to enjoy the theme park, with limited time and energy, avoiding congestion, and enjoying as many attractions as possible, has become the most important concerns for tourists.

Moreover, for the managers of theme parks, the degree of satisfaction of tourists is fundamental to the sustainable development of the theme park. The higher the evaluation of tourists, the more other tourists will be attracted to the theme park. In contrast, if the degree of satisfaction of tourists is very low, the development of the theme park will be restricted. We can abstract this routing problem as follows: consider an individual starting from a specified starting point, trying to maximize their score by accessing the existing vertices and returning to the starting point within a given time range. Each of these vertices has a known score, and the goal is to maximize the score. Through the above description, we find that this kind of problem can be attributed to the Traveling Salesman Problem (TSP)[2]. Different from the traditional traveling salesman problem, not all vertices can be visited in the orienteering route due to the maximum travel time constraint. Subsequently, this could be called the Selective Travelling Salesman Problem (STSP)[3] or traveling salesman problem with profits[4].

Therefore, based on the network of Traveling Salesman Problem (TSP), we propose a Time-Dependent Theme Park Routing Problem (TDTPRP), where the walking time is time-dependent under the consideration of the degree of congestion and fatigue. The primary objectives are to maximize the number of visited attractions and satisfaction and minimize the queuing and walking time.

2. Literature Review

Kawamura et al. [5] defined the theme park problem in 2003, that is, in a theme park with multiple attractions, the visitors as individuals or groups visit the attractions to minimize congestion and maximize satisfaction. In addition, they developed a mass-user support-based coordination scheduling algorithm to solve this problem, proving that the average waiting time and congestion in theme parks can be reduced, and the average satisfaction of grouped visitors can be increased by guiding the visitor group and adjusting the schedule. In short, the problem of theme parks is to maximize the satisfaction of individuals or crowds under the use of the management rule and an arranged schedule plan. Considering the group users, Yasushi et al. [6] formulated networks such as small-world networks and scale-free networks for the theme park problem. The results of the simulation experiment indicated that congestion could be eased greatly, and satisfaction can be further enhanced.

Most previous studies involving the theme park problem focus on large-scale travel scheduling for group users. However, differently, in this paper, we aim to help individual users avoid congestion and improve the degree of satisfaction of the visitors in the theme park. We can call this kind of problem Theme Park Routing Problem (TPRP). So far, route planning for individual users is usually divided into two categories:

(1) One category aims to find an efficient route among the chosen attractions [7].

(2) The other is to select some attractions to visit and maximize the obtained total score for visited attractions during a limited time, where the origin and destination are appointed in advance [8].

The research in this paper belongs to the second category, which does not need to specify the attractions in advance. Tsai et al.[9] developed a route recommendation system where the recommended route satisfies visitor requirements using previous tourists' favorite experiences. Lee et al.[10] presented an ontological recommendation for a multi-agent for Tainan City travel, including a context decision agent and a travel route recommendation agent. Hirotaka et al.[11] proved the tour recommendation problem can be solved as the integer programming problem using a similar formulation as used in TSP. Matsuda et al.[12] established a simple model of the optimal sightseeing routing problem and solved the model with the exact algorithm and heuristic algorithm, respectively.

From the perspective of mathematical modeling, this problem can be described as follows. A set of points is given, along with associated scores and a connecting network. Under this assumption, a path needs to be found between the specified starting point and end point to maximize the total score at a given time. It should be noted that due to the limited time, it is impossible to select all points, and some points should be discarded. Subsequently, this problem is also called the Selective Traveling Salesperson Problem (STSP)[3], which is a generalized traveling salesman problem, in which profit is associated with each vertex and only some vertices can be visited due to time constraints[13]. The STSP is also known as the Orienteering Problem[14] and the Maximum Collection Problem[15]. As the STSP is an NP-hard problem, the exact algorithms are very time-consuming, so most researchers focus on heuristic algorithms, such as the Tabu Search (TS) heuristic algorithm[16][17] or the Ant Colony Optimization (ACO) approach[18].

The major weakness of the previous research is that only static route networks are constructed, but the change of time that will lead to a change in the next journey is not considered. Considering this problem, Bouzarth et al.[19] set the service time as time-dependence but did not consider that the travel times between two vertices are stochastic functions that depend on the department time from the first vertex. Moreover, the previous studies are all single objective functions, and there are few cases of solving multi-objective problems at the same time. The algorithm research of routing optimization is mainly carried out under the condition of static networks, and there is less research using dynamic networks.

3. Time-dependent Theme Park Routing Problem Based on Multi-Objectives

(1) Time-dependent Theme Park Routing Problem

a) Problem Description

Let $G= (N, A)$ be a connected digraph with node set $N = \{1,2...,n\}$ and arc set $A = \{(m,n) \mid m,n \in N,\ m \neq n\}$, where node 1 is the starting point, and n is the end point. In the Time-dependent Theme Park Routing Problem, each node represents an attraction in the theme park. What is associated with each node is a utility score and a function of dwelling time, which is related to arrival time, queuing time, and time at the attraction. When visitors enter from the designated entrance, they should find the most satisfactory route to the attraction, and cannot leave the exit after the designated time TD. In this process, the objective of visitors is to find a route that starts from node 1 and ends at node n before TD, such that the total utility collected by all visited nodes in the route is maximized and the number of nodes experienced is maximized but the dwelling time is minimized.

Assume that the distance between the two attractions (m,n) is d and the walking times that the visitor covers this section (m,n) in two connected time periods are T_1 and T_2 respectively, as well as t_B is the boundary time of the two time periods. The walking time of the visitor in the section (m, n) is different for different time periods. The arrival time at the node m is t_m, the queuing time at the node m is α_m, and the attraction playing time at the node m is β_m, the departure time for the node m is $r_m(t_m + \alpha_m + \beta_m)$. If $r_m \leq t_B - T_1$, the walking time to cover this section in the preceding period is T_1; if $r_m \geq t_B$, the whole walking process is completed in the following period, and the corresponding walking time is T_2. Only when $t_B - T_1 < r_m < t_B$, and the walking time is between T_1 and T_2. Assume that the walking distances of the visitor in the preceding and following time periods are d_1 and d_2 $(d_1 + d_2)$, the total walking time is $d_1\left(\frac{T_1}{d}\right) + d_2\left(\frac{T_2}{d}\right)$. Obviously, the time is $t_B - r_m$ for visitors to cover d_1, and the time for visitors to cover d_2 is:

$$\frac{d_2}{d}T_2 = \frac{d-d_1}{d}T_2 = \frac{d-\frac{d}{T_1}(t_B-r_m)}{d}T_2 = \left(1-\frac{t_B-r_m}{T_1}\right)T_2$$

Therefore, the total walking time from the node m to the node n is $t_B - r_m + (1 - \frac{t_B-r_m}{T_1})T_2$, and the arrival time for node n is:

$$\begin{aligned} t_n &= r_m + t_B - r_m + \left(1 - \frac{t_B - r_m}{T_1}\right)T_2 \\ &= t_B + \left(1 - \frac{t_B - r_m}{T_1}\right)T_2 \\ &= \frac{T_2}{T_1}r_m + \left(1 - \frac{T_2}{T_1}\right)t_B + T_2 \end{aligned}$$

As can be seen from the equation (1), the time t_n arriving at the node n is an increasing function of departure time r_m.

Assume that the playing time in one day is divided into P time periods, $[t_{B,p}, t_{B,p+1}]$ represents the time period $p(p \in [1, P])$. After the visitor arrives attraction m, the queuing time and playing time are allowed before the visitor departs from attraction m. let $T_{mn,p}$ be the walking time from the node m to the node n during the period p (regardless of the period crossing), and $u_{mn,p}$ be the time that the visitor walks from the nod m to the node n. For any section (m, n), there are two possibilities:

(1) The visitor walks from node m to node n without crossing the time period $p(p = 1,2, \ldots, P)$;

(2) The visitor walks from node m to node n, crossing from period p to period $p + 1(p = 1,2, \ldots, P - 1)$.

Corresponding mathematical formulations:

$$u_{mn,p} = \begin{cases} T_{mn,p}, \quad r_m < t_{B,\ p+1} - T_{mn,p}, \quad p = 1,2,\ldots,P \\ \left(\frac{T_{mn,p+1}}{T_{mn,p}} - 1\right) r_m + \left(1 - \frac{T_{mn,p+1}}{T_{mn,p}}\right) t_{B,p+1} \\ + T_{mn,p+1}, \qquad t_{B,p+1} - T_{mn,p} \leq r_m < t_{B,p+1}, p \\ \qquad = 1,2,\ldots,P-1 \end{cases}$$

b) Model

Based on the proposed problem, the following assumptions are considered:

(1) The queuing time of attractions is acquired according to the real queuing time of the attraction (e.g., Disney Resort and Universal Studios).

(2) The time spent at each attraction is given in advance.

(3) Routes exist between any two attractions.

(4) The preference of visitors for each attraction is pregiven.

The model of the time-dependent theme park routing problem contains the following parameters:

N: Set of attractions

m: Index of attraction m, $m \in N$

TE: Entrance time

TD: Departure time

P: The number of time periods in one day

p: Index of the time period, $p \in \left[1, \ P\right]$

c_1, c_2, c_3: Weights of the functions Z_1, Z_2, Z_3, in which the sum of three weights is equal to 1

X_m: If attraction m is selected, X_m=1; otherwise, X_m=0, which is a decision variable

$Y_{mn,p}$: The route from node m to node n with the visitor departs from node m in period p is selected, it takes 1; 0 otherwise

α_m: Queuing time for attraction m

β_m: Playing time spent at attraction m

$u_{mn,p}$: Walking time of visitor departing node m in time period p to node n

$T_{mn,p}$: Time for visitor walking from node m to node n in period p (regardless of the time period crossing)

r_m: Departure time for attraction m

t_m: Arrival time for attraction m

e_m: Utility that the customer obtained from node m

$\lambda_1, \lambda_2, \lambda_3$: The conversion factors of utility in term of cost

There are three objectives involved: maximize the number of attractions, maximize the satisfaction of the visitors, and minimize walking time and queuing time.

(1) Most visitors hope to enjoy as many attractions as possible in limited time at the theme park. Therefore, we considered the maximum number of attractions visited in one day, which is formulated as follows:

$$Z_1 = c_1 \sum_{m \in N} X_m$$

(2) Most visitors hope to get as much satisfaction as possible during their limited time at the theme park. Therefore, we considered the maximum the satisfaction of visitors in one day, which is formulated as follows:

$$Z_2 = c_2 \sum_{m \in N} e_m X_m$$

(3) For most visitors, they hope to spend as much time as possible at each attraction, rather than moving on and queuing for other attractions. Therefore, we considered the minimum walking time and queuing time among the attractions in one day, which is formulated as follows:

$$Z_3 = -c_3\left(\sum_{m \in N} \alpha_m X_m + \sum_{m,n \in N} \sum_{p=1}^{P} u_{mn,p} Y_{mn,p}\right)$$

Finally, the overall mathematical model is established as follows:

$$\text{Max} \qquad \lambda_1 Z_1 + \lambda_2 Z_2 + \lambda_3 Z_3 \qquad (3\text{-}1)$$

s.t.

$$TE + \sum_{m\in N}\alpha_m X_m + \sum_{m,n\in N}\sum_{p=1}^{P} u_{mn,p}\ Y_{mn,p} + \sum_{m\in N}\beta_m\ X_m \le TD \quad (3\text{-}2)$$

$$\sum_{\substack{m\in N\\ m\neq n}}\sum_{p=1}^{P} Y_{mn,p} = 1,\ \ n\in N \quad (3\text{-}3)$$

$$\sum_{\substack{n\in N\\ n\neq m}}\sum_{p=1}^{P} Y_{mn,p} = 1,\ \ m\in N \quad (3\text{-}4)$$

$$\sum_{\substack{m,n\in N\\ m\neq n}}\sum_{p=1}^{P} Y_{mn,p} \le |S|-1,\ \ \forall S\in N, 2\le |S|\le |N|-1 \quad (3\text{-}5)$$

$$X_m\ \in\{0,1\},\ \ m\in N \quad (3\text{-}6)$$

$$Y_{mn,p}\ \in\{0,1\},\ \ m,n\in N,\ \ p\in[1,P] \quad (3\text{-}7)$$

The objective function (3-1) maximizes the number of attractions and utility values (Degree of Satisfaction) and minimizes the queuing time and walking time. Constraint (3-2) ensures that the departure time from the theme park must not exceed the preset departure time. Constraints (3-3)-(3-5) are the general constraints of TSP, in which the former two constraints restrict that each attraction must be visited exactly once, and the last constraint is the subtour elimination constraint. Constraints (3-6)-(3-7) are the domain of the variables.

（2）Algorithm

a）Partheno-genetic Algorithm (PGA)

The Partheno-genetic Algorithm (PGA) is an improved GA which has been put forward by Li and Tong in 1999 [20]. Different from the classic GA, the PGA implements its operations based on a chromosome other than two chromosomes, and it does not include the crossover operator. Owing to its specialties, the PGA can solve specific problems successfully. While the PGA can overcome the "premature" problems of conventional GA, it is more suitable for handling chromosomes of different lengths.

This algorithm adopts new genetic operators, namely, the permutation operator, shift operator, and inversion operator, and they are characterized by a simple genetic operation, decreasing the requirement of the diversity of the initial population, and avoiding "premature convergence." Therefore, it is very suitable for solving optimization problems.

In PGA, each solution is represented by a chromosome, and each chromosome includes multiple variants, namely, genes. Here, each gene indicates a theme park attraction. Different permutations for attractions require different solutions.

(1) Permutation Operator. In this study, the permutation operator incorporates two modes: the first is a single-point transposition where the position of two genes on the same chromosome can be swapped, and the second is a multi-point transposition that swaps multiple genes on a chromosome. The above two modes for permutation operators are able to generate new chromosomes according to the permutation probability. An example of the single-point transposition and multi-point transposition is listed as follows. Note that, B and B' are generated by the single-point and multi-point transposition for chromosome A, respectively.

A = $(c_1, c_2, c_3, \dots,\ c_{i-1}, c_i, c_{i+1}, \dots, c_{j-1}, c_j, c_{j+1}, \dots, c_n)$

$\Downarrow$

B = $(c_1, c_2, c_3, \dots,\ c_{i-1}, c_j, c_{i+1}, \dots, c_{j-1}, c_i, c_{j+1}, \dots, c_n)$

B'= $(c_1, c_2, c_3, \dots,\ c_{i-1}, c_j, c_{i+1}, \dots, c_n, c_i, c_{j+1}, \dots, c_{j-1})$

(2) Shift Operator. The shift operator means that a substring is randomly selected in a chromosome according to the shifting probability, the genes of the substring are moved one bit back, and the last gene is placed in the first position of the substring. Similarly, the shift operator includes two shift modes: a single-point shift and a multi-point shift operation. In the following example, H is a chromosome including multiple genes $k_i (i \in \{1, 2, \dots n\})$. I and I' are the chromosomes obtained by implementing single-point shift and multi-point shift operations for H, respectively.

H =$(k_1, k_2, k_3, k_4, k_5, \dots, k_{i-2},\ k_{i-1}, k_i, k_{i+1}, k_{i+2}, \dots, k_n)$

$\Downarrow$

I = $(k_4, k_1, k_2, k_3, k_5, \dots, k_{i-2},\ k_{i-1}, k_i, k_{i+1}, k_{i+2}, \dots, k_n)$

I' = $(k_4, k_1, k_2, k_3, k_5, \dots, k_{i-2}, k_{i+2}, k_{i-1}, k_i, k_{i+1}, \dots, k_n)$

(3) Inversion Operator. The inversion Operator refers to the process of inverting genes at the head and end successively in a substring of a chromosome according to the inversion probability, in which the selected substring and its length are selected randomly. Inversion operators can also be divided into single-point inversion and multi-point inversion. For example, assume M is a chromosome including multiple genes $l_i (i \in \{1, 2, \dots n\})$. Based on the operations of single-point and multi-point inversions for chromosome M, we can obtain the chromosomes N and N'. The above procedures can be seen as follows.

M = $(l_1, l_2, l_3, l_4, l_5, \dots,\ l_{i-1}, l_i, l_{i+1}, \dots, l_{j-1}, l_j, l_{j+1}, \dots, l_n)$

$\Downarrow$

N = $(l_4, l_3, l_2, l_1, l_5, \dots,\ l_{i-1}, l_i, l_{i+1}, \dots, l_{j-1}, l_j, l_{j+1}, \dots, l_n)$

N'= $(l_4, l_3, l_2, l_1, l_5, \dots,\ l_{i-1}, l_j, l_{j-1}, \dots, l_{i+1}, l_i, l_{j+1}, \dots, c_n)$

The procedures of the partheno-genetic algorithm are summarized as follows:

Step 1: Encoding. A serial number encoding is adopted in the parthenogenetic genetic. Algorithm.

Step 2: Initialization. Generate feasible solutions as the initial population randomly.

Step 3: Fitness function. Fitness function is the evaluation criterion of the path scheme, which represents the survivability of genetic individuals. Here, we set the objective function as the fitness function.

Step 4: Selection. The generated population is equally divided into several groups. In. this study, every four individuals are composed as a group. The best individuals in each group are directly retained as the next generation of the population.

Step 5: Partheno-genetic. Implement the permutation, shift, and inversion operations. on the remaining individuals where the gene and gene strings are selected using a random approach.

After that, the newly generated three individuals for each group are inherited to the next generation.

Step 6: Calculate the fitness of the newly generated population.

Step 7: Determine whether the termination conditions are met. When the maximum. iteration is reached, go to Step 8. Otherwise, go to Step 4.

Step 8: Return the optimal solutions and stop the algorithm.

b) Annealing Partheno-genetic Algorithm(APGA)

The global optimization ability of Genetic Algorithm is strong , but the local optimization ability is insufficient. The Simulated Annealing Algorithm is strong in local optimization but weak in global optimization ability. Therefore, the evolution mechanism of simulated annealing algorithm can be integrated into genetic algorithm to enhance its local optimization ability[21].

The basic idea of Simulated Annealing Algorithm was originated from the physical annealing process in real life. The optimal solution is acquired by abstracting the process of cooling and heating isothermals from the real physical annealing process. The local optimization ability of the algorithm is ensured using a greedy strategy and its special Metropolis Criterion[22].

Both the Simulated Annealing Algorithm and the hill-climbing method use a greedy strategy. Different from the hill-climbing method, the Simulated Annealing method has a fault-tolerant ability and is able to accept inferior solutions with a certain probability. The probability is called the probability of the acceptance of the new solution, and its degree is influenced by the current temperature and fitness difference of new and old solutions[23]. The general trend is that the lower the temperature, the lower the probability of acceptance; the larger the difference, the lower the probability of acceptance. The above can be represented by a mathematical formula, as shown in Formula (3-8):

$$P=\begin{cases}1 & E(x_{new})<E(x_{old})\\ \exp\left(-\frac{E(x_{new})-E(x_{old})}{T}\right) & E(x_{new})\geq E(x_{old})\end{cases} \tag{3-8}$$

$E(x_{new})$ refers to the fitness of the new chromosome, and $E(x_{old})$ refers to the fitness of the individual parent chromosome. T is the current temperature, and an initial temperature and cooling coefficient are set at the beginning of the genetic operation. As the iteration proceeds, the initial temperature decreases continuously. At the end of the iteration, because the temperature is already very low, the probability of accepting the new solution is almost zero. A random number between 0 and 1 is generated after acquiring the probability of acceptance. A new solution is not accepted if the number is greater than the probability of acceptance P but accepted if it is smaller than P. The selection method of the probability of acceptance is also called the Metropolis Criterion.

Metropolis Criterion based on a Simulated Annealing Algorithm certainly accepts inferior solutions while accepting elegant solutions so as to ensure population diversity and further avoid the possibility that the algorithm is stuck in the optimal local solution.

As for the mathematical model proposed in this paper, we used this idea to propose the Annealing Partheno-Genetic Algorithm, the basic operations of which include:

Step 1: Encoding. A serial number encoding is adopted in the parthenogenetic genetic. algorithm.

Step 2: Initialization. Generate feasible solutions as the initial population randomly.

Step 3: Fitness function. Fitness function is the evaluation criterion of the path scheme, which represents the survivability of genetic individuals. Here, we set the objective function as the fitness function.

Step 4: Selection. The generated population is equally divided into several groups. In. this study, every four individuals are composed as a group. The best individuals in each group are directly retained as the next generation of the population.

Step 5: Partheno-genetic. Implement the permutation, shift, and inversion operations. on the remaining individuals where the gene and gene strings are selected using a random approach. Then the newly generated three individuals for each group are inherited to the next generation.

Step 6: Calculate the fitness of the newly generated population.

Step 7: Mix the new and original populations. First, reserve the best individuals in the. two populations and calculate the average fitness. Then select one among the mixed population (except for the optimal individual) at random. If the individual is better than the average value, reserve it; otherwise, remove it until the population of the quantity, which is the same as the original individuals, is reserved.

Step 8: Determine whether termination conditions are met. When the maximum. iteration is reached, go to Step 8. Otherwise, go to Step 4.

Step 9: Return the optimal solutions and stop the algorithm.

4. Computation Results

In the experiment, we use two kinds of theme parks with different scales as examples. First, we use the Tokyo Disney Sea with 28 attractions as a real-world problem to prove the effectiveness of the model and algorithm. Second, to prove the stability of the model and algorithm, we expand the scale of the experiment and randomly generate an example of a theme park with 60 attractions.

(1) Real World Problem Instances

First, the Tokyo Disney Sea with 28 attractions, was selected as the test object to verify the correctness and effectiveness of the proposed model and algorithms. The evolution result of the Partheno-Genetic Algorithm (PGA) is shown in the Figure 4.1. where the best solution in the experiment can be found when the iteration is 109, and the value of the fitness function is 36.0801. The obtained satisfaction route is 0-2-27-22-24-21-20-26-23-16-19-18-17-13-11-6-0.

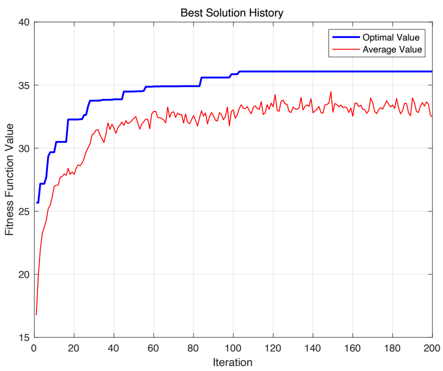

Figure 4.1. Evolution results of PGA with 28 attraction

Then, the evolution results of the improved Annealing Partheno-Genetic Algorithm (APGA) are illustrated in Figure 4.2. Here, we set the initial temperature to 90 and the cooling coefficient to 0.99. The best solution in the experiment can be found when the iteration is 156, and the value of fitness function is 36.0801. The result is the same as that of PGA, and the obtained satisfaction route is 0-2-27-22-24-21-20-26-23-16-19-18-17-13-11-6-0.

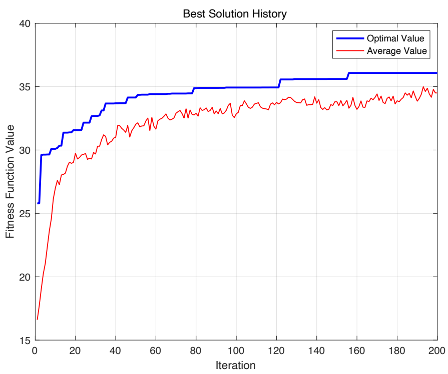

Figure 4.2. Evolution Results of APGA with 28 Attractions

By comparing the two results above, when there are 28 attractions, the best solution of the two different algorithms can be found in a short time. In addition, we tested the two algorithms 15 times. Table 4.1 illustrates the comparison results of the two different algorithms. The presented results are the best solution (Best), worst solution (Worst), average solution (Average), and the standard deviation (Std).

Table 4.1: Comparison of Different Methods for Solving TDTPRP (28 Attractions)

Method	Best	Worst	Average	Std
PGA	36.0801	34.9208	35.645	0.31
APGA	36.0801	35.611	35.931	0.12

（2）Real World Problem Instances

The results above demonstrate that both the PGA and APGA mentioned in this paper can be used to find the best solution within a short time when resolving the small and medium-sized theme park routing problem. Then, we expanded the test scale in the second test. We generated a theme park with one entrance and 60 attractions at random within a 2,000*2,000 test environment.

The evolution results of the PGA are illustrated in Figure 4.3, where the best solution in the experiment can be found when the iteration is 185, and the best value of fitness function is 34.0835. The obtained satisfaction route is 0-6-18-26-7-10-30-9-44-8-21-0.

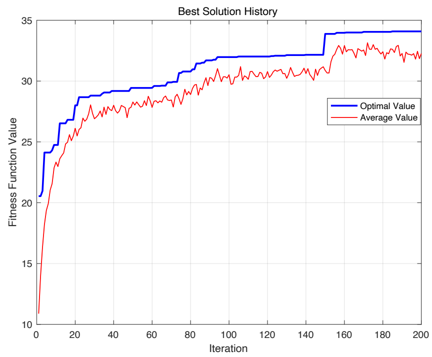

Figure 4.3. Evolution Results of PGA with 60 Attractions

The evolution result of the improved APGA is illustrated in Figure 4.4, where the best solution in the experiment can be found when the iteration is 160, and the best value of fitness function is 34.1492. The obtained satisfaction route is 0-6-18-26-24-7-10-30-9-44-8-0.

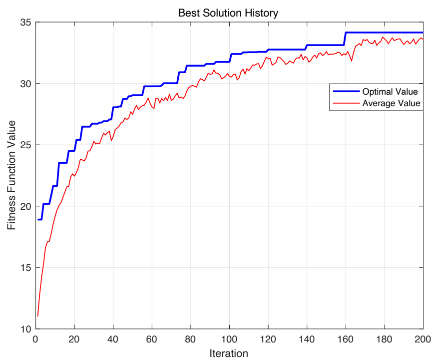

Figure 4.4. Evolution Results of APGA with 60 Attractions

If there are 28 attraction projects, the two different algorithms can be used to find the relative best solution within a short time successfully. However, when we increased the attractions to 60, the computation time of the PGA was longer, and it was inferior to the APGA in terms of optimizing ability. Similarly, we ran the two algorithms 15 times in the test. Table 4.2 illustrates the comparison of the two algorithms.

Table 4.2: Comparison of Different Methods for Solving TDTPRP (60 Attractions)

Method	Best	Worst	Average	Std
PGA	34.0835	31.3534	32.9746	0.86
APGA	34.1492	33.9915	34.0854	0.05

５． Conclusion

To solve the problem of the congestion and queuing problem in large scale theme parks, improve the satisfaction of visitors, and decrease congestion, a Time-Dependent Theme Park Routing Problem (TDTPRP) was proposed to maximize the utility of the visitors and minimize the queuing and walking time for selecting the optimal attractions under the framework of the Traveling Salesman Problem (TSP), where walking time was treated as time-dependent and changed according to different time periods. The model can provide more precise scheduling and plan for individual decisions. To solve the proposed model and verify the feasibility and effectiveness of the model, we proposed a Partheno-Genetic Algorithm and an Annealing Partheno-Genetic Algorithm. In the experimental stage, we conducted two experiments; the experimental data were divided into real-world problems and randomly generated problems. First, we used the Tokyo Disney Sea with 28 attractions as the real-world problem to prove the effectiveness of the model and algorithm. Second, to prove the stability of the model and algorithm, we expanded the scale of the experiment and randomly generated an example of a theme park with 60 attractions within a 2, 000*2, 000 test environment. The results demonstrate that when the experiment scale is small, the general Partheno-Genetic Algorithm and Annealing Partheno-Genetic Algorithm have the same excellent optimization ability, but the Annealing Partheno-Genetic Algorithm has better optimization ability than the general Partheno-Genetic Algorithm when the data scale was expanded.

For future research, there is space for further improvements. For example, the walking time is dynamic, but to make it comparable to a realistic problem, the queuing time should also be constantly changing. Moreover, at present, some theme parks offer tickets to skip queues, some of which are free and some of which involve a fee. These factors could be considered in future research to continue to develop the model, and with the improvement of the model, the design of the algorithm should also continue to improve.

Acknowledgements

I would like to express my gratitude to all those who helped me during the writing of this thesis. I gratefully acknowledge the help of my supervisor, Professor. Li Lei, who has offered me valuable suggestions in the academic studies. Without his patient instruction, insightful criticism and expert guidance, the completion of this thesis would not have been possible.

Second, I also owe a special debt of gratitude to all the professors in Hosei University, from whose devoted teaching and enlightening lectures I have benefited a lot and academically prepared for the thesis.

Lastly, my thanks would go to my beloved family for their loving considerations and great confidence in me all through these years. I also owe my sincere gratitude to my seniors who gave me their help and time in listening to me and helping me work out my problems during the difficult course of the thesis.

References

[1] Johns, Nick, and Szilvia Gyimothy. "Mythologies of a theme park: An icon of modern family life." *Journal of Vacation Marketing* 8.4 (2002): 320-331.

[2] Little, John DC, et al. "An algorithm for the traveling salesman problem." *Operations research* 11.6 (1963): 972-989.

[3] Laporte, Gilbert, and Silvano Martello. "The selective travelling salesman problem." *Discrete applied mathematics* 26.2-3 (1990): 193-207.

[4] Feillet, Dominique, Pierre Dejax, and Michel Gendreau. "Traveling salesman problems with profits." *Transportation science* 39.2 (2005): 188-205.

[5] Kawamura, H., K. Kurumatani, and A. Ohuchi. "A study on coordination scheduling algorithm for theme park problem with multiagent." *Institute of Electronics, Information and Communication Engineers Technical Report. Artificial intelligence and knowledge-based processing* 102.613 (2003): 25-30.

[6] 柳田靖, and 鈴木恵二. "テーマパークにおける巡回戦略の比較." *情報処理学会研究報告知能と複雑系 (ICS)* 2007.26 (2007-ICS-147) (2007): 15-22.

[7] 菊池新, and 大西啓介. "JAVA を用いた観光地経路案内システム." 全国大会講演論文集 ソフトウェア科学・工学 (1997): 431-432.

[8] Matsuda, Yoshitaka, et al. "An Optimal Routing Problem for Sightseeing and its Solution." *IEEJ Transactions on Electronics, Information and Systems* 124 (2004): 1507-1514.

[9] Tsai, Chieh-Yuan, and Shang-Hsuan Chung. "A personalized route recommendation service for theme parks using RFID information and tourist behavior." *Decision Support Systems*52.2 (2012): 514-527.

[10] Lee, Chang-Shing, Young-Chung Chang, and Mei-Hui Wang. "Ontological recommendation multi-agent for Tainan City travel." *Expert Systems with Applications* 36.3 (2009): 6740-6753.

[11] 新妻弘崇, 新井晃平, and 太田学. "観光ルート推薦のための効率的な制約条件." *情報処理学会論文誌データベース (TOD)* 9.2 (2016): 34-45.

[12] Matsuda, Yoshitaka, et al. "An optimal routing problem for sightseeing with fuzzy time-varying weights." *2004 IEEE*

International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583). Vol. 4. IEEE, 2004.

[13] Thomadsen, Tommy, and Thomas K. Stidsen. "The quadratic selective travelling salesman problem." (2003).

[14] Vansteenwegen, Pieter, Wouter Souffriau, and Dirk Van Oudheusden. "The orienteering problem: A survey." *European Journal of Operational Research* 209.1 (2011): 1-10.

[15] Ergezer, Halit, and Kemal Leblebicioglu. "Path planning for UAVs for maximum information collection." *IEEE Transactions on Aerospace and Electronic Systems* 49.1 (2013): 502-520.

[16] Gendreau, Michel, Gilbert Laporte, and Frédéric Semet. "A tabu search heuristic for the undirected selective travelling salesman problem." *European Journal of Operational Research* 106.2-3 (1998): 539-545.

[17] Qin, Hu, Andrew Lim, and Dongsheng Xu. "The selective traveling salesman problem with regular working time windows." *Opportunities and challenges for next generation applied intelligence*. Springer, Berlin, Heidelberg, 2009. 291-296.

[18] Ke, Liangjun, Claudia Archetti, and Zuren Feng. "Ants can solve the team orienteering problem." *Computers & Industrial Engineering* 54.3 (2008): 648-665.

[19] Bouzarth, Elizabeth L., et al. "A Comparison of Algorithms for Finding an Efficient Theme Park Tour." *Journal of Applied Mathematics* 2018 (2018).

[20] Maojun, Li, and Tong Tiaosheng. "A partheno-genetic algorithm and analysis on its global convergence." *Acta automatica sinica*25.1 (1999): 68-72.

[21] Gandomkar, M., M. Vakilian, and M. Ehsan. "A combination of genetic algorithm and simulated annealing for optimal DG allocation in distribution networks." *Canadian Conference on Electrical and Computer Engineering*, 2005. Ieee, 2005.

[22] TIAN, Qi-chuan, et al. "Research on Learning Algorithm of BP Neural Network Based on the Metropolis Criterion [J]." *Techniques of Automation and Applications* 5 (2003).

[23] Suppapitnarm, A., et al. "A simulated annealing algorithm for multiobjective optimization." *Engineering Optimization* 33.1 (2000): 59-85.