A Stamina-Aware Sightseeing Tour Scheduling Method

Bing Wu
Nara Institute of Science and Technology
Nara, Japan
bing-w@is.naist.jp

Weihua Sun
Nara Institute of Science and Technology
Nara, Japan
sunweihua@is.naist.jp

Yoshihiro Murata
Hiroshima City University
Hiroshima, Japan
yosihi-m@hiroshima-cu.ac.jp

Keiichi Yasumoto
Nara Institute of Science and Technology
Nara, Japan
yasumoto@is.naist.jp

Minoru Ito
Nara Institute of Science and Technology
Nara, Japan
ito@is.naist.jp

ABSTRACT

In general, a tour schedule is composed of multiple sightseeing spots taking into account the user' s preferences. However, during the tour, the stamina of the tourists may be exhausted. In this paper, we propose a sightseeing scheduling method that maximizes the degree of user satisfaction taking stamina into account. In our method, break times are allocated in the schedule to satisfy the stamina constraint. Since this problem implies a TSP and thus is NP-hard, it is difficult to solve in practical time. To calculate a semi-optimal solution in practical time, we propose a method that first composes a schedule visiting multiple sightseeing spots without considering stamina, and then, to recover stamina, allocates break times, based on a predatory search technique. To evaluate the proposed method, we compared our method through a simulation experiment with some conventional methods including a brute-force method. As a result, the proposed method composed a schedule in practical time whose expected degree of satisfaction was near the optimum.

Categories and Subject Descriptors

G.1.2 [**Approximation**]: Nonlinear approximation; G.1.6 [**Optimization**]: Constrained optimization; G.2.1 [**Combinatorics**]: [Combinatorial algorithms]

General Terms

Algorithms, Performance

Keywords

Time-tabling and scheduling, Metaheuristics, Local search

1. THE STAMINA-AWARE SIGHTSEEING TOUR PROBLEM

In recent years, various personal navigation systems which navigate users to their destinations and provide them with surrounding information have been proposed. In our previous work, we proposed a sightseeing navigation technique which considered changes in the weather [1]. However, this technique does not change tour schedules depending on a tourist's stamina and sightseeing method.

GECCO'11, July 12–16, 2011, Dublin, Ireland.
ACM 978-1-4503-0690-4/11/07.

Moreover, because there are various factors such as food, water, and different environments, that can decrease a tourist's stamina, the tourist can easily become sick during the sightseeing trip. Therefore, it is necessary to take the tourist's stamina into account when making the schedule. For scheduling, it is important to decide when and where to rest. How to set the rest properly under the restrictions of stamina and time, and how to make a sightseeing scheduling that maximizes satisfaction are a complex optimization problem.

To plan the sightseeing schedules, we define the attribute of tourists and sightseeing spots as follows. Tourist can input a degree of satisfaction degree for each sightseeing spot which has multiple sightseeing methods. According to the tourist's preferences, sightseeing methods are given different degrees of satisfaction. For each sightseeing method, the required stamina and time to stay are given by our system before sightseeing begins.

The object of our study is to find a sightseeing schedule that gives a tourist a good degree of satisfaction within the restriction of stamina. To simplify the problem, we propose a technique in which the tourist is able to rest only after finishing a sightseeing spot. Fig. 1 shows an example of a schedule. We can see that this schedule includes rests and the remaining stamina has a non-negative value during the whole schedule.

Start Time	*Destination*	*Sightseeing Method*	*Satisfaction Degree*	*Remained Stamina*	*Arrive Time*	*Return Time*
9:00	Hotel	By Car	0.0	900.0	9:00	
|___9:09	Spot10	Method1	106.327	644.0	10:39	
|___10:50	Spot1	Method2	97.432	303.0	11:50	
|___11:50	Spot1	Rest	0.0	343.0	11:54	
|___12:02	Spot9	Method2	94.134	2.0	13:02	
|___13:02	Spot9	Rest	0.0	72.0	13:09	
|___13:17	Spot3	Method4	90.714	4.0	15:17	
|___15:17	Spot3	Rest	0.0	214.0	15:38	
|___15:52	Spot8	Method6	105.317	9.0	16:52	16:56

Satisfaction Degree of Schedule : 493.924

Figure 1: An example of a schedule obtained by the proposed method

2. PROPOSED ALGORITHM

In this paper, to solve this problem, we utilize a predatory search method (hereafter, **PS**) [2] that is suitable to derive a solution consisting of multiple hierarchical and dependent factors.

Based on PS, we propose an algorithm consisting of two phases: (i) temporary solution search phase and (ii) neighborhood search phase. Here, (i) and (ii) correspond to the Extensive Search and

Area-restricted Search of PS, respectively. In the temporary solution search phase, we derive a temporary solution by generating a solution at random and improving the solution through local search. In the neighborhood search phase, we generate multiple solution sub-domains and intensively search the domains in the increasing order of the distance from the temporary solution to the domain. If a solution better than the temporary solution is found, we re-generate the sub-domain centered at the new solution and search the domain. These two phases are repeatedly applied as if a search-intensive predator seeks prey in the vicinity of some promising spots.

Below, we explain the details of these two phases.

2.1 Temporary solution search phase

The algorithm is as follows.

(1) For each sightseeing spot, one sightseeing method with the highest cost performance is chosen to get a set of candidate spots where each sightseeing spot contains just one sightseeing method. We define the cost-performance that determines the importance of the sightseeing method considering both required stamina and time to stay.

(2) Calculate the number of destinations to visit in the schedule and select destinations at random from the candidate spots.

(3) Select one destination in the sightseeing schedule at random and overwrite it with an unused spot randomly chosen from the candidate spots, or select another destination in the sightseeing schedule and swap both of them.

(4) Modify the sightseeing schedule so that it satisfies the stamina constraints by adding rest times and/ -or removing some of destinations. If the degree of satisfaction is improved, accepts the modification. Otherwise, cancel the modification.

(5) At last, return to (3). We decide a specified number of repetitions to do this search.

2.2 Neighborhood Search Phase

This phase improves the temporary solution derived by the previous phase by generating multiple solution sub-domains at random and thoroughly searching the domains one by one in the nearest domain first manner. We introduce 2 kinds of sub-domains: destination sub-domain and a visiting order sub-sub-domain for the sightseeing schedule. The algorithm is as follows.

(1) Initialize the solution by the temporary solution.

(2) Generate a specified number of destination sub-domains.

(3) Calculate the distance between a solution and the sub-domains. Each sub-domain contains destinations (here, each destination will be the center of a visiting order sub-sub-domain explained at (4)) selected at random. The solution is overwritten by the new one obtained by the search if its degree of satisfaction is higher than before, and then the algorithm returns to (2). At last, if it can-not find a higher one, the destination sub-domain search stops.

(4) We use the same method (1), (2), and (3) to generate visiting order sub-sub-domains for each destination sub-domain. Each sub-sub-domain contains a visiting order generated by swapping 2 destinations in the sightseeing schedule at random.

(5) Select a sightseeing method at random and search the neighborhood of the visiting order sub-sub-domain by local search, Finally, the algorithm returns the best solution found in the search.

3. EXPERIMENTS AND EVALUATIONS

In this section, we compare the proposed method with the following conventional methods to evaluate the degree of satisfaction of the schedule. **Greedy method**: This method selects the sightseeing spot of a sightseeing method with the highest user satisfaction degree one by one until the return time. If the remaining stamina becomes a negative value, a rest is inserted before the action. **TSS method**: This is the first phase of our proposed method (Temporary solution search). **The Brute-force search method**: This method makes all combinations of sightseeing spots and sightseeing mehtod, and searches the schedule exhaustively. This method can find the optimal schedule. However, a huge calculation time is needed.

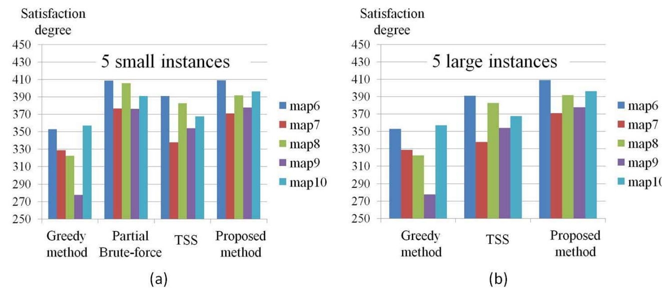

Figure 2: Results of a comparison of the degree of satisfaction - small instances(a) and large instances(b)

We used maps that contain some sightseeing spots in a field of 13500m $\times$ 9000m, where the positions of the sightseeing spots were given by random coordinates. The number of sightseeing methods for each sightseeing spot was assumed to be two, with a time to stay of each sightseeing method of 1, 1.5 or 2 hours. We conducted this comparison experiment for two kinds of instances (5 small instances containing 10 sightseeing spots and 5 large instances containing 20 sightseeing spots). The degree of satisfaction of each sightseeing spot was given randomly from 1-100. For the large instance, we compared the proposed method only with the Greedy method and TSS method, because the Brute-force search can not obtain the optimum schedule in practical time.

Fig. 2 shows the results of the experiments. They are the averages of 30 trials. In Fig. 2(a), we know that it took 109.75 minutes to obtain the optimum solution by the Brute-force search, however, for the same instance, the proposed method composed a schedule whose expected satisfaction is 95.53% of the optimum solution in 13.68 seconds on the average. Moreover, compared with the other three comparison methods, the proposed method obtained a good solution in practical time. In Fig. 2(b), the proposed method also showed a good result in a large map instance. From these results, we confirmed that the proposed method can obtain a schedule with a high degree of satisfaction in practical time.

4. FUTURE WORK

We think that it is important to consider the stamina in relation to the movement of a tourist. So we will extend our problem and approach to consider means of transportation.

5. REFERENCES

[1] Wu, B., Murata, Y., Shibata, N., Yasumoto, K., Ito, M., “A Method for Composing Tour Schedules Adaptive to Weather Change,” Proc. of 2009 IEEE Intelligent Vehicles Symposium (IV’09), pp. 1407-1412 (June 2009).

[2] Liu, C., Wang, D., “Predatory search algorithm with restriction of solution distance ,” Biological Cybernetics, Vol.92, No.5,pp.293-302 (2005).