JEAS
Journal of Economics and Administrative Sciences
E-ISSN: 2148-1792 P- ISSN: 1302-2024
Volume 3, Supplement Issue 3
www.jeasweb.org

Provision of a Personalized Tourist Recommendation System to View the Scenic Areas of a City Using the Ant Algorithm

Liela Nazemi,
Master of Software, Pishtazan University College of Shiraz, Iran.

Saeed Mehrjoo
Assistant Professor, Department of Electrical and Computer Engineering, Dariun Branch, Islamic Azad University, Dariun, Iran.

ABSTRACT

Trip planning is one of the most important and time-consuming tasks in trip preparation. One has to choose some Point of Interest among the hundreds of possible places in planning to visit to a popular tourist city. Consider their visit order, the time needed to see each place, the time to move from one place to another, and even the time to meet a specific person in a given place. The study first defined a new trip planning problem, allowing the users to specify places to visit and the categories of optional locations, according to the trip time-budget. Moreover, we proposed a two-step program for personal, interactive, and traffic-aware trip planning. We proposed a new way to rate the itinerary that uses the popularity and personal preferences of places at simultaneously. The proposed system uses the ant algorithm to recommend the routes for maximizing the route score while simultaneously taking into account the time constraints of places and total trip time. Ultimately, we examined the efficiency of the proposed method with our evaluations using large real-world datasets. A case study on planning three trips with various start times and user preferences indicated that the proposed method can describe suitable routes that fully meet the needs of users and according to the traffic conditions throughout the routes selected at the specified time.
Key Words: Trip planning, ant algorithm, route points, places to visit

Introduction

Tourism is an important social, economic and cultural phenomenon involving the movement of millions of people around the world with a significant effect on the economies of many countries. Thus, producing the tools associated with tourism can have a significant effect on the economy of community [1]. In recent years, many studies have been proposed about the tourist recommendation systems [3-2]. In many countries, the lack of information systems for planning intra-city trips has had a very negative effect on their transportation and tourism industry. Hence, the need to use new technologies to provide intra-city trip planning services based on knowing the needs of the user and his knowledge of places to visit, is strongly felt in recent decades.

Nowadays, trip planning systems have become one of the famous solutions for organizing and planning trip [5, 4].

The objective of the trip planning system is the interaction with users to determine if the locations specified by the user can all be covered in the time budget. By using geographical and traffic information of places of interest, a framework for better planning of intra-city trips can be provided besides the existing parameters like popularity, number of visits, arrival and departure times, and so on.

The problem of trip planning in the studies has been referred to by other titles like tourist trip design problem (TTDP) [6], which are interested in meeting multiple POIs. One first has to get information about all POIs in the target city and connect them to build a POI network model to get a proper itinerary. In a POI network model, nodes contain information about POI, like location category, geographical locations, and so on, and edges usually contain information about the distance or transit time between POIs. Thus, finding an appropriate number of POIs and the optimal route by considering the constraints is the main problem of the study.

In trip planning problem, another issue is called multimodal tour planning [7] or multimodal transportation [8], where multimodal means using various modes of passage from one POI to another in the itinerary, like walking, or using various public vehicles like buses, subways, and so on.

In the framework presented in this thesis, two separate recommender and non-recommender systems are proposed given the various needs of the user - a one-day traffic-aware, interactive and personalized itinerary depending on the user needs and constraint discussed in [9]. Sequential pattern mining and meta-heuristic algorithms are used to enhance the accuracy of the proposal, to increase the route rank, and to reduce the calculation time.

Meta-heuristic algorithms are of the categories of approximate optimization algorithms that have solutions for exit from local optimal points and can be used in a wide range of problems. In the past, meta-heuristic algorithms used to be used to solve various trip planning problems [10, 7], but meta-heuristics have not been used in the trip planning problem defined in this thesis. The ant colony optimization algorithm is known as swarm intelligence (collective intelligence) algorithms and models the behavior of real ants. Ants are self-organizing creatures. Such a feature is at the core of the problem as it is exactly what makes insects quickly adapt to their changing environmental conditions to reach goals through low-level interaction.

Using the ant algorithm, one can recommend a suggested route that will end in passenger more satisfaction.

Various studies have been done regarding trip planning. Lim et al. (2017) have presented an algorithm for proposing personalized patrols that use POI popularity and user willingness to suggest suitable POI for meeting POIs. Test results show the effectiveness of the proposed algorithm [11]. Lim et al. (2017) present an algorithm for proposing personalized tours that use POI popularity and user willingness to suggest suitable POI meeting POIs. Their problem is formulated using a formulation from OP considering time budget constraints and the need for a start and end at specific POI. The proposed algorithm is strengthened using 1) weighted updates of users' interests based on the delay of meeting their POI, and 2) automatic weighting between POI popularity and user desire. The tests results show the effectiveness of the proposed algorithm [11]. The problem of recommending trip to travelers and tourists is a significant topic that has been the subject of many papers in recent years [13-12]. Hossain et al. (2019) [7] used graph theory to find the optimal tour and recommend to the audience. Padia et al. (2019) [14] proposed a method that personalizes the proposed route. To this end, they model the previous behaviors of users and use them to provide a recommended route. Sarkar et al. (2020) [15] proposed a system that recommends several routes to the audience instead of one, and the user can select one of these routes according to his personal taste.

Given the significance of the tourism industry in the social, cultural and especially economic status of a community, the need to produce tools related to tourism is defined well. Different trip planning systems have been proposed for tourists with various constraints, some of which are in the group of trip suggestion systems and recommend POI and suitable route to the user, and some of them do not belong to the group of recommending systems and the user tends to specify himself, places of interest or category (type) of places for the system. In both cases, the system determines the optimal routes between these locations. However, given the lack of a comprehensive framework that covers both of the above according to the needs of the user in the real world, it is necessary to provide such a framework: so that if the user is a recommender, the system given time constraints and the user's location, recommend him the best places and the optimal route between them, and if the user is a non-recommender system applicant and wants to visit pre-determined locations by himself, the system selects the appropriate POI from among the POI select the user to recommend the optimal route between those places. The framework provided for real-world use must be traffic-aware, interactive and personalized besides the above features. Thus, the purpose of the study is the problem of trip planning. According to a dynamic location network G in a target city and an IQ user itinerary request, our goal is to find the optimal valid route with maximum route points.

Material and methods

- **The framework of the proposed method [16]**

The proposed framework has three components: the dynamic location network model, the route search component, and the route enhancement component. While the dynamic location network model is pre-fabricated and maintained offline, route search and route enhancement components jointly respond to users' real-time trip requests.

Dynamic location network modeling

Node modeling

Each node has five characteristics in this model: operating time, category to which the location belongs, popularity, geographical location, and visit duration. The users specify the time they want to visit for each location, whereas information about the first four features is extracted from Foursquare data (Figure 1).

Figure 1: Information about a node in Foursquare

The operating time of a particular place may differ depending on the day of the week or even the time of year. A location can be identified by two or more category tags, which are multi-layered in Foursquare. For instance, use Nick's Crispy Tacos as shown in Figure 5. This label has three categories, of which "Food" is a Level 1 label, "Spot Breakfast" Level 2, and "Multiplex" Level 3.

We use two indices - total number of visitors (tvs) and total number of customers (tcs) (Equation 1) - to calculate the popularity of a specific location. The total number of visitors is usually less than the total number of visitors to the same place, as some users visit the site frequently during one visit.

$$Pop(v_i) = \frac{2 \times \frac{tvs(v_i)}{c_1} \times \frac{tcs(v_i)}{c_2}}{\frac{tvs(v_i)}{c_1} + \frac{tcs(v_i)}{c_2}}$$

(1)

Here, c_1 has the highest number of visitors from all locations in the target city, and similarly, c_2 has the highest number of visitors from all locations. One has to note that the most visited location may vary from the location with the most entry history.

Given the geographical location of a specific location, Foursquare provides latitude and longitude information along with its address. Although the exact time spent in a given location cannot be accurately derived from login data, one can approximate using the average length of stay of tourists. One has to note that the users may estimate the length of stay when planning a trip and adjust it to the actual visit.

Edge modeling

One must consider the nature of the traffic time between locations to estimate the dynamic displacement time by driving from one node to another (i.e. the value of an edge). In doing so, we use a real-world data set - the taxi GPS track. This data has two unique features: 1) spatial coverage: a certain number of city taxis can completely cover the entire road network. 2) Temporal coverage: the taxis usually work around the clock, in line with the visiting time of tourists. The two features of GPS Taxi data enable us to estimate the movement time between the two nodes at any given time interval.

For making calculating the time to move between nodes in the POI network easier, one first clusters the nodes that are near, among which walking is the best way to move. Intra-cluster movement time is calculated using the average walking speed, whereas inter-cluster transmission time is calculated based on driving speed at a specific time. Figure 6 indicates a simple dynamic POI network. Small circles of various colors refer to nodes (POI). Near nodes are clustered (oval shapes with striped lines). The directional edges within each cluster have walking time information between nodes independent of daylight hours. However, our directional edges between clusters specify the time information of movement between them, which varies with time. For instance, during the busy hours of the morning, the time from the upper right cluster to the lower cluster is more than twice the minimum daily time (Figure 2).

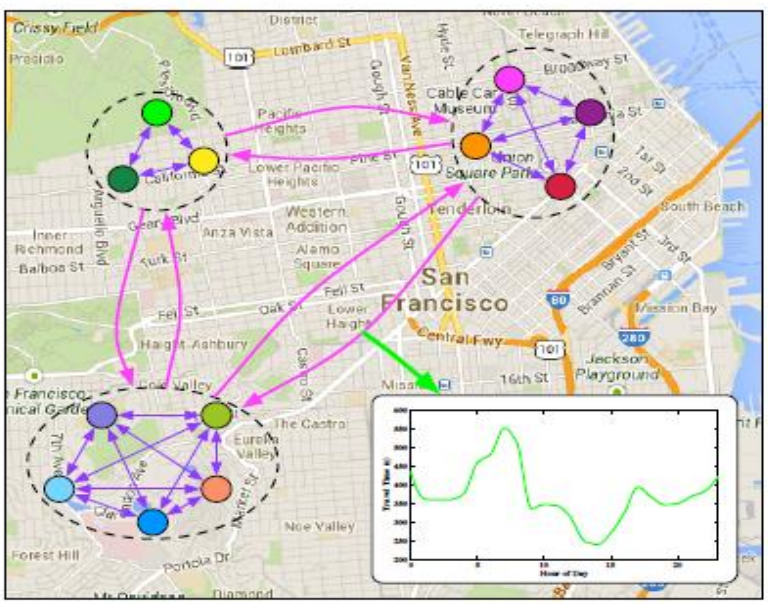

Figure 2: Dynamic network built using Foursquare and GPS trace of taxis [16]

Dual phases of the proposed method [75]

The proposed method [75] uses a two-phase approach for trip planning, route search and route reinforcement. The route search phase generates a series of candidate routes that visit all user-specified locations, considering time budgets and other constraints. The route reinforcement phase reinforces the candidate routes of the previous phase with the user's favorite locations from the user's suggested categories as long as time allows, and recommends the optimal routes with the highest scores to users.

A method is designed to strengthen the route, with two steps: location insert and maximum score. The former aims to find the right position on the candidate route to insert a selected location, whereas the latter is responsible for maximizing the score of the updated route. The purpose of the algorithms discussed in [16] is to reinforce the candidate routes received from the previous step (i.e. route search). Ranking based on route score selects the route with the highest route rating to respond to the user itinerary (IQ) request. Note that if several enhanced "optimal" routes have the same route score (routes with the same locations but in different order), the route will be higher with a smaller "trip time".

The proposed method based on the ant algorithm

The proposed method presented in [16] is a greedy method. Dynamic graph model, making nodes and edges, and so on and our proposed method are similar to the method presented in [16] and differ only in finding the optimal route of our task. The algorithm examined [16] had two general phases. Firstly, the algorithm is examined to see whether the selected locations could be visited within a specified time frame or not. Moreover, we used this phase at the beginning of our proposed method. Our proposed algorithm first examines whether the user's desired locations can be visited within the specified time period. If this is impossible, the user will be asked to remove a location from the list of places to visit. When the algorithm has ascertained that there is at least one possible route for the user in the specified time period, it moves on to the ant algorithm.

It releases N ants at the origin and executes the ant algorithm on the graph of the target city.

Producing ant populations

One of the main challenges in using ant algorithm for the intended problem is the structure used to create the ant population. In the previous phase, examining the possibility of visiting all the areas desired by the user, it is determined that, for instance, it is possible to visit k desired locations. Thus, all final proposed routes must have a minimum size of k + 2, k POIs of interest, an origin and the destination. However, it determines k number of minimums and the number of places visited may be much higher than k. Hence, a fixed-length structure cannot be used to show the proposed routes for each ant. Thus, we used the link list structure in our proposed method to structure the movement of each ant and the proposed routes. The linked list has a flexible structure that can easily increase or decrease its size; for instance, if an ant visits places v1, v2, v3, v4, respectively.

The list can easily increase the size and accommodate more visit places. We placed N number of ants at the origin of the tourist movement. This means that the population size is N. For all candidate routes, we consider the first house as the origin, and the ants move quasi-randomly, reviewing subsequent locations.

Fitness function

A value must be considered for each route produced by the ants in the ant colony algorithm. Our proposed method uses a fitness function similar to that of the proposed method in [16]. Indeed, the value of each route depends on the value of the places visited on that route. Hence, the value of each route is expressed as follows.

(2) Fitness (Ant_i) = $\{$ the route involves the POI of user $\mathcal{VS}_{obj}(v_i) + \mathcal{VS}_{sub}(u_j, v_i)$

The route does not involve POI of user 0

The fitness function above shows that if a route is generated by an ant that does not visit all of the user's POI, then the proposed route is undesirable and has no value. Thus, the fitness function sets the value to 0 for such a route. Otherwise, the value of the route depends on the value of the places visited on that route.

Results

This section presents the evaluation results aimed at (1) measuring the validity and efficiency of trip planning algorithms and (2) the usefulness and customizability of the trip planning system.

Starting the test

Data preparation

Similar to the study presented in [16] from April 2010 to October 2010, our study used from the San Francisco Four Score log data and taxis GPS traces in the same city from the CabSpotting project (http://cabspotting.org/ to build the San Francisco POI network. Four Score data contains 110214 login records by 15680 users. GPS taxi data includes 391938 passengers, which was developed in June 2008 by 536 taxis. We did not enter empty taxi data because they may not drive at normal speeds when searching for passengers. POI network building method is discussed in Section Four.

Evaluation environment

All evaluations in this study were performed with Python on an Intel core i7 computer with 8 GB of RAM and Windows 10 operating system.

Performance evaluation

This section compares our proposed method, based on the ant algorithm, with the best proposed method presented in [16], which is the gravity method. The efficiency of the algorithms depends on some parameters, like the total number of locations (N) in the target city, the number of location categories intended by user (k), the number of locations specified by the user (m) and the defined trip time budget defined by user (Δ). The first two variables specify the number of intended locations of the user (i.e. new candidate locations). The number of places specified by the user and the trip time budget affect the number of candidate routes produced in the first phase (route search phase) and the number of user locations that can be inserted in the second phase, especially the maximum m! The candidate route can be generated. The number of user-specified locations (m) is the same for both algorithms and affects the computation time in both the route search phase and the route amplification phase. For simplicity, we consider $m = 10$ in all evaluations. The next experiment shows how the choice of N affects the computation time of the two algorithms.

It has to be noted that all candidate routes are reinforced in parallel. In other words, the total calculation time in the route reinforcement stage is equal to the maximum calculation time among all candidate routes. Efficiency is measured by the total cost of time in both steps.

1) N changes: The relationship between the computation time of the two algorithms and the total number of city locations (N) is shown in Figure 3. The results show that the gravity algorithm presented in [16] consumes less time than our proposed algorithm. In this experiment, we have considered $k = 3$ and $\Delta = 10$ hours as constant. The ant algorithm is basically a time-consuming algorithm as a large number of ants have to build routes for each generation repeatedly. The reason for this noticeable difference is the inherent computational complexity of the ant algorithm.

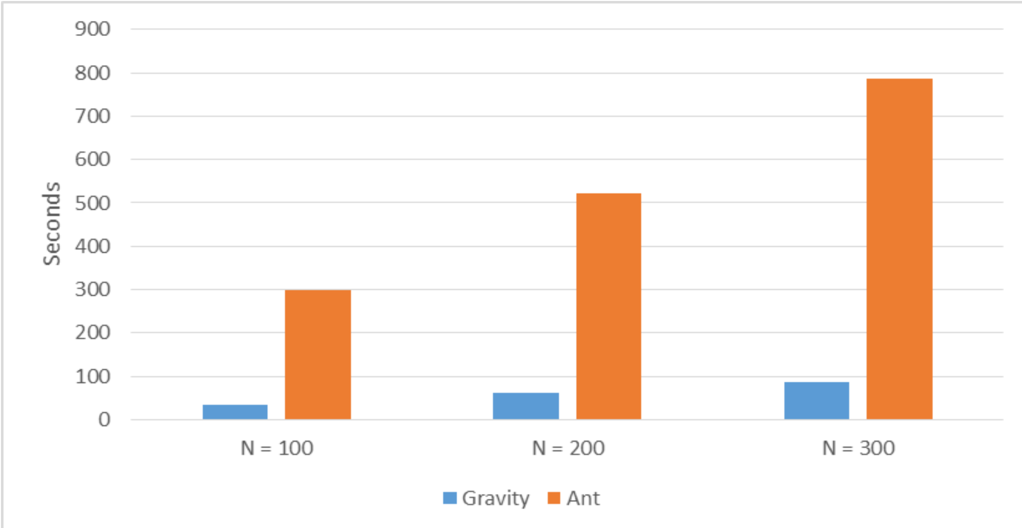

Figure 3: Time complexity for various algorithms

We performed a route score evaluation test to examine the performance of the algorithms. The optimal return route scores by the gravity algorithm are shown with N, K and Δ as 300, 3 and 10, respectively, and the proposed ant algorithm is shown in the figure. The proposed ant method outperforms the maximum gravity method in scoring the proposed optimal routes (Figure 4).

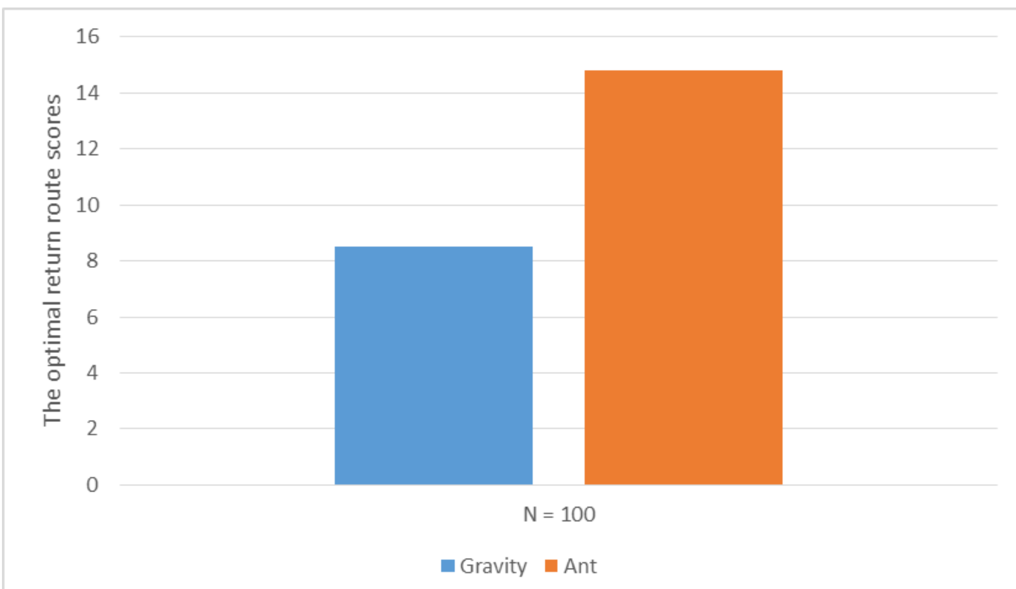

Figure 4: Optimal return route scores by the algorithm

The reason for the superiority of the proposed ant algorithm over the greedy gravitational method is quite clear. The ant algorithm is an optimal global heuristic algorithm, yet the greedy algorithm may not reach the optimal global solution. Although the ant algorithm is slower than the gravitational method, it gives more optimal solutions, making it superior.

Case study

We will test the customizability of the proposed system. The main places specified to visit on all routes are Golden Gate Bridge, Beach, Lombard Street, Fisherman's Wharf, and Museum. The train station is the beginning and end of the trip and another mandatory point. Thus, all the routes include these 6 places and the route points depend on the selected route and additional selected locations. In case two users with different personal interests send the same trip request (IQ1) to the system to be more precise, according to their login history, one user (u1) preferred large outdoor venues and restaurants, while the other user enjoyed art and entertainment venues and restaurants more. To show the ability to be aware of the proposed system traffic, we considered the latter case in which u1 has changed the trip request and set a different start time (IQ2). We consider the third case where the proposed route was compared to an intermediate route by u1 in response to IQ2 to confirm that the route recommended by the proposed system is optimized. The requests share the following information in all three modes: 1) the start and end of the trip of the users is Coltrane station, ii) the locations specified by the user are Museum, Golden Gate Bridge, Beach, Lombard Street and Fisherman's Wharf. iii) The total budget for the trip time is set at 11 hours. IV) Preferred categories for the user are restaurants, arts and entertainment, excellent outdoor space, and v) dining time for lunch [11:00 am, 12:59 pm] and for dinner [17:30 pm, 20:00 pm]. Table 1 shows the information we have designed, including the user, start time, and the recommended route results.

Table 1: Information about the design

	Users	Start time	Recommended route	Gravity	Ant
Mode 1	U1	10:00	R1	15.3	23.8
	U2	10:00	R2	15.6	24.6
Mode 2	U1	10:00	R1	15.3	23.8
	U2	08:30	R3	14.8	23.7
Mode 3	U1	08:30	R3	14.8	23.7
	U2	08:30	R4	13.5	21.8

Conclusion

The purpose of the study was to present an algorithm and trip planner that guides travelers to see the sights of a city. The study recommends some points out of hundreds of sights of a city based on the person and presents an optimal route for the user to see them. Increase in the number of sight areas, user satisfaction, the score of the proposed route, the optimal route to visit the sight areas, and so on are among the research objectives. The traffic parameter was used as a new parameter besides the previous parameters like popularity, number of visits, and so on.

Simultaneous application of street traffic along with restrictions like opening and closing times, start and end times of trip, appointment arrangements, optimal route were less studied in this problem. We tested this algorithm using the ant algorithm.

Sarkar et al. [15] suggested an algorithm that uses a greedy method whose optimal solutions may not be universal; thus, we used a method that could find optimal global solutions - the powerful ant algorithm.

The results of the experiments proved the superiority of the ant method over the greedy method. Although the ant method was worse compared to the greedy method concerning time complexity, it offered much better suggested routes in practice.

Overall, the study examined the simultaneous use of street traffic along with restrictions such as open and closed times, start and end times of trip, appointment arrangements, and the optimal route that are less examined. We studied this algorithm using the ant algorithm. However, if a dataset was available for these data in Iran, the problem would be much more significant.

We want to expand and deepen this work in several ways in the future. Firstly, we plan to extend the scenarios to multi-day trip planning. Secondly, we want to deploy our system on cellphone devices. Thirdly, we intend to test our system using real users in real-time, and gather feedback on how to enhance the services more. Ultimately, other swarm intelligence methods like bee algorithm, genetic algorithm, and so on could be used too.

References

[1] Cenamor, T. de la Rosa, S. Núñez, and D. Borrajo, "Planning for tourism routes using social networks," Expert Systems with Applications, vol. 69, pp. 1-9, 2017.

[2] T. Aoike, B. Ho, T. Hara, J. Ota, and Y. Kurata, "Utilising Crowd Information of Tourist Spots in an Interactive Tour Recommender System," in Information and Communication Technologies in Tourism 2019: Springer, 2019, pp. 27-39.

[3] F. Gasparetti, D. Gavalas, S. Ilarri, F. Ricci, and Z. Yu, "Mining social networks for local search and location-based recommender systems," Personal and Ubiquitous Computing, pp. 1-2, 2019.

[4] Moreno, A. Valls, D. Isern, L. Marin, and J. Borràs, "Sigtur/e-destination: ontology-based personalized recommendation of tourism and leisure activities," Engineering Applications of Artificial Intelligence, vol. 26, no. 1, pp. 633-651, 2013

[5] P. Vansteenwegen, W. Souffriau, G. V. Berghe, and D. Van Oudheusden, "The city trip planner: an expert system for tourists," Expert Systems with Applications, vol. 38, no. 6, pp. 6540-6546, 2011

[6] P. Vansteenwegen and D. Van Oudheusden, "The mobile tourist guide: an OR opportunity," OR insight, vol. 20, no. 3, pp. 21-27, 2007.

[7] D. Gavalas, V. Kasapakis, C. Konstantopoulos, G. Pantziou, N. Vathis, and C. Zaroliagis, "The eCOMPASS multimodal tourist tour planner," Expert Systems with Applications, vol. 42, no. 21, pp. 7303-7316, 2015.

[8] R. A. Abbaspour and F. Samadzadegan, "Time-dependent personal tour planning and scheduling in metropolises," Expert Systems with Applications, vol. 38, no. 10, pp. 12439-12452, 2011.

[9] C. Chen, D. Zhang, B. Guo, X. Ma, G. Pan, and Z. Wu, "TripPlanner: Personalized trip planning leveraging heterogeneous crowdsourced digital footprints," IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 3, pp. 1259-1273, 2015.

[10] D. Gavalas, C. Konstantopoulos, K. Mastakas, and G. Pantziou, "A survey on algorithmic approaches for solving tourist trip design problems," Journal of Heuristics, vol. 20, no. 3, pp. 291-328, 2014.

[11] K. H. Lim, J. Chan, C. Leckie, and S. Karunasekera, "Personalized trip recommendation for tourists based on user interests, points of interest visit durations and visit recency," Knowledge and Information Systems, pp. 1-32, 2017.

[12] M. S. Hossain, A. S. Tanim, N. Nawal, and S. Akter, "An Innovative Tour Recommendation System using Graph Algorithms," Journal of Information Systems Engineering and Business Intelligence, vol. 5, no. 1, pp. 32-39, 2019.

[13] P. Padia, B. Singhal, and K. H. Lim, "User-relative Personalized Tour Recommendation," in IUI Workshops, 2019.

[14] D. Herzog and W. Wörndl, "A Trip Recommender System for Combining Multiple Trip Regions to a Composite Trip," in CBRecSys@ RecSys, 2014, pp. 42-48.

[15] A. Gunawan, H. C. Lau, and K. Lu, "An iterated local search algorithm for solving the orienteering problem with time windows," in European Conference on Evolutionary Computation in Combinatorial Optimization, 2015: Springer, pp. 61-73.

[16] C. Chen, D. Zhang, B. Guo, X. Ma, G. Pan, and Z. Wu, "TripPlanner: Personalized trip planning leveraging heterogeneous crowdsourced digital footprints," IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 3, pp. 1259-1273, 2014.