ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ONLINE MOBILE-BASED PERSONAL TOUR APPLICATION USING ARTIFICIAL BEE COLONY OPTIMIZATION ALGORITHM

[1]SAWSAN ALSHATTNAWI, [2]IYAD ABU DOUSH , [3]ANAS ALSOBEH
[1,2]Computer Science Department, Yarmouk University; Jordan
[3]Computer Information Systems Department, Yarmouk University; Jordan
E-mail: [1]sawsan_kh@yu.edu.jo, [2] iyad.doush@yu.edu.jo, [3]anas.alsobeh@yu.edu.jo

ABSTRACT

Tourists that visit multiple locations need to have prior knowledge about interesting and attractive destinations. Selecting the most suitable tour plan for any tourist is not an easy task. Building a schedule of proposed destinations may allow the tourist to navigate between these destinations efficiently based on his/her time and budget. This paper applies the Artificial Bee Colony optimization algorithm (ABC) to build a tour plan by selecting the Points of Interest (POIs). The algorithm is applied to the user mobile phone. Tourist must specify the start place, the end place, the stay period (number of days) and the preferred POIs, if any, and some other preferences. The objective is to maximize the tourist satisfaction by selecting several POIs which satisfy the provided time and budget. The system is built on top of Android for its simplicity, popularity, and availability of web services. To evaluate the efficiency of the proposed technique, we build a dataset of 50 attractions in Jordan. The algorithm fitness function is tuned based on comparing the resulted tour plan with a tour plan generated by an expert. The evaluation shows that the proposed technique gives the maximum number of POI within the provided time and budget.

Keywords: *Mobile Application, Artificial Intelligence, BEE Colony Algorithm, Web Service, Tour Plan.*

1. INTRODUCTION

The widespread use of smartphones makes it easier to access different kinds of information anywhere and anytime. Such availability can be used to help the tourism discipline by providing several web services for tourists. Therefore, several works in the literature have been proposed to help in guiding tourists to visit attractions using their smartphones. Such web services usually have a web back-end to get the desired information and a mobile application with the needed information [6][23][9].

Electronic tourist guide represents good support for the tourists which provides them with the whole information about all the Points of Interest (POIs). Such guide provides the tour information according to the user context: location, language, time and any other preferences [8].

Tourists, visiting cities for many days usually like to visit the most attractive locations. Selecting part of the available POIs is needed as the tourist will not be able to visit all the sites. The POIs selection depends on several criteria: the duration to visit each location, the time duration to travel from one location to another and the level of POI attractiveness. In addition, the selection must match all the user time and budget constraints. A challenging problem that emerges from this domain is designing a tour plan which is called Tourist Trip Design Problems (TTDP) [45].

Tour plan building is a generalization of the Traveling Salesman Problem (TSP). The objective when solving this problem is to maximize the total profit and to minimize the traveling cost within the time constraint [25]. Tour plan optimization includes maximizing user satisfaction, select the shortest path, and take the user time and budget preferences into consideration. Therefore, the problem could be seen as a multi-objective optimization with several constraints. A multi-objective algorithm can produce multiple solutions and the user can select the solution that satisfies his/her constraints.

Many heuristic-based methods and swarm intelligence techniques are pro- posed to solve such a problem. These algorithms need computational resources. A number of solutions use the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

client/server model **Error! Reference source not found.** or build different routing solutions **Error! Reference source not found.**.

Artificial Bee Colony (ABC) is an evolutionary algorithm which mimics the bees' behavior [24]. It has been used to solve many different combinatorial optimization problems such as: traveling salesman problem [46] [51] [59], shop scheduling problem [20]**Error! Reference source not found.**, transportation problem [34]**Error! Reference source not found.**, and path planning of mobile robots [18][33]. The POIs can be classified into several types such as historical, leisure, medical, etc. The work presented in [43] presents a decision support system based on selecting the POIs according to users' preferences by categorizing these points of interests such as culture, shopping, and leisure.

This paper introduces applying ABC to solve the problem of designing a schedule for a tourist trip. The proposed algorithm tries to select locations with maximum ratings as well as have a minimum distance between each pair of locations. The schedule consists of three phases: get user preferences, POI selection, and schedule customization. The proposed tour plan can be adapted by allowing users to add/delete any POI. This allows the user to select the suitable schedule that matches his/her preferences such as the total distance, arrival and stays time and the popularity of the POIs. The proposed technique is evaluated using a newly created dataset of 50 attractions in Jordan. The evaluation results show that the proposed technique is efficient and accurate as the fitness function is tuned based on comparing the generated tour plan with a tour generated by an expert. In addition, the proposed system is deployed on smartphones and the system response time to show the generated schedule is acceptable.

2. RELATED WORK

The internet provides access to a large amount of tourism information. This huge amount of information makes the selection of a customized trip from several alternatives, not an easy task for a decision-maker **Error! Reference source not found.**. Activities selection and sequencing is considered as a multi-objective problem. In this problem, we choose a specific activity for the tourist in each day from a set of different activities. It is a sequencing problem as we have to specify the activities order each day. It is considered a multi-objective problem as selecting each option could introduce a conflict between multiple competing objectives. The selection and routing problem can be divided into two problems. First, selecting the locations the tourist will visit. Second, developing the optimal route when visiting these locations **Error! Reference source not found.**.

Prize Collecting Traveling Salesman Problem (PCTSP) is a variation of this problem in which there is a cost when traveling to each pair of cities, a prize collected when visiting each city, and a penalty that must be paid when failing to visit a city. The objective is to minimize the cost and penalties while maximizing the collected prize [11]. This problem is tackled by several researchers (e.g., [17]**Error! Reference source not found.**). Another similar problem is the Orienteering Problem (OP) [16]. In this problem, each city has a score and the goal is to find the path with the smallest number of cities path with a maximum sum of the scores. Several algorithms are proposed to solve such problem [36][52][34]. Prize Collecting Traveling Salesman Problem with a time window [13][12] is a variation of this problem with a time window as a constraint at each visited location. The research on the tour planning problem can be classified into three groups [5]: Using exact approaches which relies on heuristics and meta-heuristics [26][31][49][50]. Dynamically planning an optimal itinerary by finding a path using multi-objective algorithm [7][19]**Error! Reference source not found.**. Designing intelligent tour planning based on the personalized tour recommender [37][47].

The problem of time-dependent tour planning in complex and large urban means specifying a chronological sequence of attractive points during a specific time interval using the available transportation system. Abbaspour and Samadzadegan [5] proposed an approach that uses two adapted genetic algorithms to generate an itinerary based on user preferences and restrictions of interesting points. The proposed process has been tested over Tehran city dataset and consists of the preparation of 400 tours with different initial points, start time, and tour duration. The proposed system uses three modes to travel between points of interest walking, bus, and subway. The experimental results and the optimality ratios show that the proposed algorithm can find the optimum tour according to the defined constraints. This algorithm solves the tour planning problem using only the time constraint. However, tourists may wish to have a tour that

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

satisfies multiple criteria (e.g., cost and attractiveness level).

Herath and Ratnayake [30] proposed a traveling guide system that can find a route according to users' preferences. The user can select his/her preferences such as the type of place, time duration, type of traveling. Then, the system will find a path that satisfies those criteria. The system can suggest places for users or display facilities based on the user current location. A new pathfinding algorithm is created for finding a solution while considering the type of the POIs (museums, parks, and historical places), the traveling method (by bus, bike), the cost, and the time window in each location. The implemented system is evaluated with Sri Lanka places dataset. The authors mentioned that the main deficiency of the existing tourist guide applications is that they do not have any suggestions when selecting the starting point and a destination point for a tour. They propose a system that have place suggestion facility, time management options, path selection and tour customization. The system usability is evaluated on twenty participants, but the results are not shown in the paper.

In another study [27], the researcher used a questionnaire about tourist's tour route. The purpose is to collect information about the tourists demand and tour characteristic. The results show that visit frequency and effective residence time are positively correlated with the satisfaction degree of the visited attraction. It shows that the real tour visiting time is less than the tour budget time for about 30 % of the tourists. The evaluation results show that crowded places are not preferred by the tourists.

Hasuike et al. [29] mentioned that existing mathematical models do not consider several important factors when planning the tour. For example, the required traveling times between two places and the traffic data to estimate the delay time in the case of traffic jam. In order to overcome these drawbacks and to construct a more general framework of tour planning the proposed mathematical model of tour planning is developed with time-dependent satisfactions. The authors used Time-Expanded Network (TEN) to represent the time-dependent model. Using the proposed TEN-based model, it is possible to construct various tours with flexible time, costs, and satisfaction. It can select optimal departure place and accommodations according to the tour route with tourists' favorite places and get the time scheduling of the tour route. The proposed model is formulated as a 0-1 integer programming problem. No real-time application is developed to test the efficiency of the model.

Souffriau et al. [45] solved the tourist trip design problem and developed an automatic approach to extract the scores associated with POI using the vector space model. The guided local search meta-heuristic algorithm is used to solve the tourist trip design problem. Rodrguez et al. [45] proposed a tool that generates itinerary based on tourist preferences. This problem is considered a multi-objective problem of activities selection and sequencing. A mathematical model is developed then Tabu Search procedure is used to solve the problem. The interactive method of g-dominance is used to represent the level of interest in each criterion according to users' preferences. The proposed method is evaluated on planning a personalized trip to visit Andalusia. They collected information about the activities and places in the city from the public and private Web Applications and saved this information in a database. The proposed model is tested using a computer to plan a personalized trip.

Claes and Holvoet [21] proposed an algorithm to find the shortest path in a road network. The algorithm is based on Ant Colony Optimization and on Ant Colony System (ACS). They use a new parameter γ to balance heuristic and pheromone values. This makes the algorithm uses the local update on pheromone (i.e., dynamic stopping criteria for the algorithm). The experiments on real-world traffic networks show that the proposed algorithm outperforms the static routing using the ACS algorithm. Joest and Stille [31] introduced a user adaptive tour framework that applies a local search algorithm. The tour information is modeled using a graph. Then they reduce the graph size to minimize the computation time by removing nodes that do not represent points of interest. The proposed framework uses two heuristics: where the first generates a feasible initial solution, and the second enhances successively that solution.

Zografos and Androutsopoulos **Error! Reference source not found.** developed a tour planning model in time-scheduled urban public transport network to determine the optimal departure time of a journey. A dynamic programming-based algorithm is developed to determine the lexicographical optimal tour route. The obtained route is optimized in terms of the number of transfers and the total walking and waiting time. The algorithm is tested by developing a standalone application on the PC and it takes around 9 seconds to present the journey planning.

Zhu et al. **Error! Reference source not found.** developed a framework using a mixed-integer and linear programming to solve the Tour Planning Problem (TPP). The proposed framework generates a tour with the most popular sites according to the specified budget and time constraints. The results show that the proposed method produces very good approximation solutions with lower computational complexity when compared with the existing methods. The proposed algorithm is evaluated on a PC by using 30 sites and the average processing time was 6.7 seconds.

The Team Orienteering Problem (TOP) is a special case of the tour planning problem in which both the starting and end points of the tour are given. Several researchers proposed different algorithms to solve TOP. Archetti et al. [10] developed two versions of tabu search and a variable neighborhood search algorithm to solve TOP. Ke et al. [31] applied ant colony optimization to propose four different methods to construct candidate solutions for such a problem. Vansteenwegen et al. [49] proposed two local search heuristics and applied a guided local search method for solving TOP. Multi-modal route planning is providing the traveler with optimal, feasible, and personalized route between start and end points using public and private transportation modes. Yu et al. **Error! Reference source not found.** proposed an improved genetic algorithm (GA) to solve this problem. The algorithm uses variable length chromosomes with different partitions to represent routes. The multi-objective ranking method is applied to obtain the optimal solutions.

Montemanni et al. [36] Applied Ant Colony System (ACS) algorithm to solve the Team Orienteering Problem with Time Windows (TOPTW). They enhanced the ACS algorithm to tackle the TOPTW using local search techniques. The evaluation results on benchmark instances available in the literature show that the enhanced ant colony algorithm outperforms the original ant colony algorithm. Clauss et al. [22] proposed a population-based Ant Colony Optimization (PACO) algorithm and applied it to solve the Combined Tours Traveling Salesperson Problem (CT-TSP). The problem is similar to TSP in which we have a set of cities with a cost between each pair of cities. The goal is to find k cyclic tours with each tour returns the traveler to the origin city with a minimum cost. The obtained results show the importance of using the heuristic when constructing the first tour with the heuristic for the second tour to obtain good results.

Zheng et al. **Error! Reference source not found.** introduced the use of a genetic algorithm and evolution algorithm to tackle the problem of tourist trip design. The obtained results show that the proposed system outperforms other existing systems with a route designed according to user preferences. Zehetner and Gutjahr **Error! Reference source not found.** applied an extension of the NSGA-II algorithm to solve the tour problem. The extension uses an approximation to improve the Pareto front. The algorithm is evaluated using a dataset from Senegal. The obtained results show that the proposed algorithm outperforms other algorithms when compared to a fixed solution. Cergibozan and Tasan [15] conducted a study using a genetic algorithm to solve the tourist route planning problem. The obtained results show the importance of including the popularity of the attraction on the selected points to visit on the generated route.

Several researchers introduce the use of Artificial Bee Colony (ABC) algorithm to solve the tour planning problem. Bhagade and Puranik [14] pro- posed Artificial Bee Colony (ABC) algorithm for vehicle routing optimization problem. Pandey and Kumar **Error! Reference source not found.** introduced an enhanced ABC Algorithm to solve the traveling salesman problem. Wong et al. **Error! Reference source not found.** presented an efficient ABC algorithm for the traveling salesman problem using frequency-based pruning. The Artificial Bee Colony (ABC) algorithm will propose a good solution for this type of problem, to the best of our knowledge such an algorithm is not used to construct a tour plan for a tourist. Our work is to analyze the work done by ABC to improve in a different area and to identify the strengths and weaknesses of the approach as well as mobile applications. The use of this algorithm is efficient from the run time complexity perspective which will be suitable for smartphones with their limited CPU capabilities. In addition, it different than other algorithms is the intelligence of selecting several POIs that are adapted to the user preferences and produce an efficient route to exploit the user's time and budget.

3. TOUR PLANNING PROBLEM MODELING

In this section, we present our modeling for the tour planning problem by applying the ABC algorithm. The tour planning problem can be presented by a graph G with V vertices (nodes here represent locations) and E edges connecting nodes which can be denoted as$G(V, E)$. The locations are categorized into five categories: leisure, medical, historical, ecological, and religious. Each node is associated with a quadric vector containing the necessary parameters: $\{Ri, Di, Ti, Ci\}$, where R_i is the rate which represents the degree of attractiveness of the node (POI). D_i is the time duration to visit this node. T_i represents the node category and C_i is the city ID for $node_i$.

The tour plan means selecting multiple nodes (v) from a start node (v0) to end node (vn−1) with the goal of maximizing the attractiveness rate of these nodes (equation 1) within the required time to visit all the nodes $v_0, ..., v_{n-1}$. The time to travel from each node to the next must be less than or equal to the total time (T_{max} in equation 2). Note that equation 2 is used as a constraint in the selection process, as each time we select a node (location) the visit time must be less than or equal to T_{max}. x_{ij} is a binary variable which equal to 1 if the visit from v_i to v_j is selected and 0 if not. The constraints in equation 3 implies that no POIs are visited more than once. Equation 4 guarantee that there is no loop in the tour where V is a set of POIs used in the solution. Note that when we search for the locations we pass from i to j until we reach the location to visit which is k.

$$max \sum_{i=0}^{nn-1} \sum_{j=1}^{n} R_i x_{ij} \quad (1)$$
$$\sum_{i=1}^{n} D_i + \sum_{t=1}^{n} \sum_{j=2}^{n} t_{ij} \leq T_{max} \quad (2)$$
$$\sum_{i\neq k} x_{ik} = \sum_{k\neq j} y_{kj} \leq 1 \quad (3)$$
$$\sum_{i\neq k} x_{ij} \leq |V| - 1 : \forall V \in D, |V| \geq 3 \quad (4)$$

After the selection of v nodes, a path will be built between these nodes according to several criteria such as the safety (road status) and the shortest distance. The overall distance can be calculated using equation 5 as:

$$D_{0,n-1} = \sum_{v=0}^{v=n-1} d_{i,i+1} \quad (5)$$

The total time may be calculated as equation 6 which represents the time to travel from one POI to another. The estimated duration to visit POI_i must be less than or equal T_{max}

$$t_{ij} = \sum_{v=i}^{v=j} W_k \quad (6)$$

Equation 6 is used after the location selection for the tour is complete. In the equation, W represents the estimated time to travel from one location to another. Equations 1, 2, 3, and 4 are the constraints applied during the selection of the tour locations, while equations 5 and 6 are applied after the selection.

4. OVERVIEW OF ARTIFICIAL BEE COLONY OPTIMIZATION ALGORITHM

This section provides an overview of the ABC which is used to solve the tourist tour planning. The ABC algorithm mimics the bees to solve optimization problems [32]. The bees are organized into three groups: employed bees, onlooker bees, and scout bees. The scout bees search for food sources around the hive, they return to their hive when they find food sources. They communicate with other bees by a waggle dance which give directions to food sources and the amount of the nectar [51]. Not all returning bees will dance but only the ones that find shorter tour length. The onlooker bees are waiting in the dance area to choose the food source. The bees that go to collect the food are called the employed bees. The number of employed bees is equal to the number of food sources. If the food source is exhausted by employed bees then the onlookers become scouts. The food sources positions represent the possible solutions. The nectar amount represents the solution quality. The number of employed bees equals the number of onlooker bees which is equal to the number of solutions. The algorithm steps are explained as follows:

1. Generate the initial population: The initial solution of size SN (food source) **Error! Reference source not found.**, where SN is half of the colony size. $X_i = \{x_{i,1}, x_{i,2}, ... , x_{i,D}\}$ represent the i_{th} solution where D is the number of generated solutions. Each bee X_i generates a new candidate solution x_{ij} from a neighborhood value based on its current position as in equation 7:

$$x_{ij}(t+1) = x_{ij}(t) + \emptyset_{ij}{:}(x_{ik} - x_{kj}) \quad (7)$$

where k is a randomly selected candidate solution ($i \neq k$), j is randomly selected from {1, 2,…,D} and $\emptyset_{ij}$ is a random number within [-1,1]. At each time (t) the solution x_{ij} is selected and the fitness function is applied. If the fitness value of $x_{ij}(t+1)$ is better than its parent $x_{ij}(t)$ then update X_i with x_{ij} otherwise keep X_i unchanged.

2. Send employed bees to food sources: the employed bees search for food sources and calculate the nectar amount. The bees then share the information of their food sources with the onlooker bees through a waggle dances. The nectar amount is calculated as follows:

$$p_i = \frac{fit_i}{\sum_{j=1}^{SN} fit_j} \quad (8)$$

3. Send onlooker bees to collect the nectar: the onlooker bees evaluate the nectar information taken from all employed bees and choose a food with the best food source probability. The onlooker bees collect the nectar amount and calculate the nectar amount. They collect the food until the food source is consumed.
4. Send scout bees to search for other food sources: looking for other sources is done if the food source is not improved over a predefined number of cycles. In this case, the food source is abandoned as it is replaced with a new food source. The scout bees can discover a new food source to be replaced with X_i as follows:

$$x_{ij} = lb_{ij} + rand(0,1).(ub_j - lb_j) \quad (9)$$

where $rand(0,1)$ is a random number within [0,1] and lb_j and ub_j are lower and upper boundaries of the j_{th} dimension. The lb_{ij} represents the lower bound for the i_{th} solution in the jth dimension.

5. ABC FOR TOUR PLANNING

In this section presents how to use ABC to solve the problem of tour planning. The following are the modeling steps:

1. Preference setting: some information is needed from the user to build an efficient schedule. The user has to specify the start point, the departure point, the stay period, the preferred POI, if any, and the preferred category.
2. POIs selection: the ABC algorithm is used for the location selection.
3. Schedule customization: the user may modify the schedule by adding, deleting, or re-ordering the POI in the schedule. The system then re-evaluates the schedule according to these modifications.

5.1. POIs SELECTION

As mentioned before, the POIs selection is done using the ABC algorithm. The initialized hive location is the same as the starting tourist point and the final location is the departure tourist point. The algorithm is repeated several times until reaching the final destination. Initially, all the bees are on the source node and the onlocker bees search for next food source (i.e., location). The probability to select the next location is computed using the equation 10. This equation depends on the location rating and the distance between the current location and the next selected location. The bees with a maximum p_{ij} will dance, which means that the location with the maximum rating value will be selected as the next location to be visited (equation 11). If equation 11 is not satisfied then the next higher value will be selected. The newly selected location now becomes a source and these steps are repeated until we have the final tour plan.

Step 1: Input the parameters start point which represent the arrival point (north, or south), and the number of days (T_{max}).
Step 2: Initialization step set the bee colony size to the total number of locations to be visited. This is because in the bee algorithm the population is distributed over the area of selection. Half of the colony size equal to the number of employed bees and the other half will be the onlooker bees. The bee colony individuals are initialized randomly as $\{x_1, x_2, \dots, x_n\}$ with a nectar value and an initial condition for each route. Initially, we set time counter to 0, set distance to 0, and no points of interest are selected yet ($POI_{set} = \{\}$).

Step 3: Start the tour

Place all the bees on the start node ($S_{original}$) that is entered by the user as a starting "arrival" point.

1. Send employed bees to the neighborhood.
2. The employed bees initiate a complete path by looking in the search space randomly
3. Compute the probability to move to *locationj* from the initial location using equation (10).

Step 4: Observe the dance sort the next locations based on the probability calculated in the previous step in descending order (start with the highest).

Step 5: Selection Select the location with maximum probability to be the next point.

$if\ the\ constraints\ are\ satisfied\ for\ the\ selected\ location\ then$

$POIset = POIset + selected\ location$

$tour\_time = tour\_time + D\ for\ the\ selected\ location$

$choose\ the\ next\ location$

$else\ goto\ step\ 4$

Step 6: $S_{original}$= selected location found in step 4, goto step 3 while tour time <= T_{max}

Step 7: compute the distance between the selected POIs,

output: $v_0,\ldots, v_{n-1}$ the selected POIs

$$P_{ij} = \frac{[R_j]^{\alpha} \cdot \left[\frac{1}{d_j}\right]^{\beta}}{\sum_{j \in A_i} [R_j]^{\alpha} \cdot \left[\frac{1}{d_j}\right]^{\beta}} \quad (10)$$

where Rj is the attractiveness rate of the location j and dj is the distance from location i to location j. α is a binary variable to switch on and off the rate value, and β controls the significant level of the location distance.

$$\sum_{i=1}^{n-1} [d_{ij} + D_j] \leq T_{max} \quad (11)$$

5.2. SCHEDULE CUSTOMIZATION

The proposed system is flexible as the user can modify the schedule by adding, deleting, or changing the order of the visited locations. In addition, it allows the user to change the route between two locations. The following is an explanation for each case:

1. Add Location: the tourist can choose a location to add, then the system can rebuild the schedule with the selected new location. The new location must be placed in the correct order. In this case, two possibilities may arise: adding the location without deleting any location by ignoring T_{max}. Another possibility is to delete another location to keep the constraint of T_{max} satisfied.
2. Delete Location: this option allows the tourist to delete a location and rebuild the schedule by replacing the deleted location with another location according to the user preference. In addition, the user can delete any location without replacing it with another.
3. Change locations order: the system allows the tourist to change the visiting order of the POIs.
4. Changing the route: if there are several routes between two locations, then the system allows the tourist to select a different path.

6. IMPLEMENTATION AND EVALUATION

Implementation and dataset: the system is implemented on top of the Android operating system. The database SQLite in Android is used for storing the dataset. The dataset contains 50 POIs that the tourist can visit in Jordan collected by the authors from several Websites such as the Web Application of the tourism ministry [4]. The 50 attractions can be used to generate a huge number of tours. The complete enumeration of all possible path for the 50 attractions is 50! which is 3.0414093e+64.

Jordan is a small country, its area is 89,342km2 and the estimated population is 9 million in 2016 [1]. The attractiveness rates for the locations are collected from [3]. In addition, we visited several tourism agencies to know the popularity of the location. The accommodations such as restaurants and hotels are collected and rated. A report from the tourism ministry [2] indicated that the maximum number of days to make a tour to visit the important locations in Jordan is 15 days. The proposed system takes this into

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

consideration and it allows the user to select the number of days for the tour (from 1 to 15 days).

The distance between two points of interest is retrieved using the Google Distance Matrix API. In addition, Geo Coder API is used to determine the current user's location. These web services retrieve the results in XML format. The SAXParser is used to get the information from the XML file.

Evaluation of Generated Schedule: Yan et al.[28] mentioned that the metrics used to evaluate a tourist route can be the distance between the visited sites, the amount of time given to each site and the tourist satisfaction which depends on his/her preference. In this study, we rely on comparing a fixed solution generated by an expert to compare against the obtained results and to tune the proposed algorithm. This technique is commonly used when developing expert systems. In real situations, tourists may wish a tour that satisfied multiple criteria such as the priority of the attraction based on the user preference, the staying time, and the cost of the route. The testing is performed during the system's development. The expert is Dr.Batoul Muhaisen from the department of travel and tourism at Yarmouk University. The used evaluation metric is to measure the matching between the expert selected POIs and the POI generated by the proposed system. During the development stages, the equations that are applied to get a valid result are verified and updated to give better results. This lead to several modifications to the fitness function. The development is based on two ideas to build a powerful schedule. First, we separate the locations into three regions: north, middle and south and apply the algorithm in each region separately. Second, we apply the algorithm by using the fitness function in equation 12 to get the maximum rate.

$$P_{ij} = \frac{R_j}{\sum_{j \in A_i}[R_j]} \quad (12)$$

The suggested algorithm is been evaluated on different problrms and compared in form of exteprimental resutls. The result shows that combined algorithm preforemed best among all algorithms.

During the initial evaluation, we use the system to schedule a plan for three days with the initial point is the north and the tourism category is historical. We found that there is a large mismatch between the expert selection and our system selection. The matching percentage is calculated as follows: find the intersection between the POIs generated by the expert and the POIs generated by the system. Then divide the intersection result by the total number of POIs selected by the expert. The selected POIs chosen by the

Table 1. Evaluation Process: A Comparison between the POIs chosen by the System

Phase no.	POIs Selected by the Expert	POIs Selected by the System	Matching Percentage
1	Petra, Om Qais, Ajloun Castle, Jerash, Sabeel Alhoriat, Roman amphitheatre, Aqaba	Um Qais, Ajloun Castle, Jerash, Temple of Hercules, Karak Castle, Makaor, Azraq Castle, Amra	42.9
2	Petra, Om Qais, Ajloun Castle, Jerash, Sabeel Alhoriat, Roman amphitheatre, Aqaba	Perta, Jerash, Sabeel Alhoriat, Temple of Hercules, Karak Castle, Alharrana	57.2
3	Petra, Om Qais, Ajloun Castle, Jerash, Sabeel Alhoriat, Roman amphitheatre, Aqaba	Perta, Om Qais, Ajloun Castle, Jerash, Sabeel Alhoriat, Temple of Hercules, Tabaget Fahael, Al-harrana	85.7

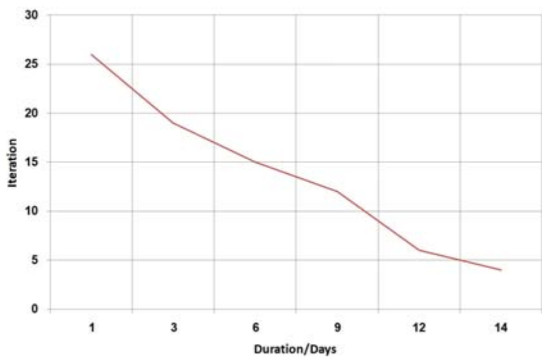

Figure 1. The number of iteration and duration

expert and chosen by the system are shown in Table 1.

The system chooses the highly rated locations in different regions, but it did not take into account the distance from the start point. This leads to choosing the highly rated POIs and ignoring the locations that have a very good rate which is near the tourist start location. The mismatch, in this case, was also large, and the expert advice to visit the highly rated locations that are around the tourist location. Therefore, the distance of the selected location needs to be taken into consideration by the algorithm. This leads us to the fitness function shown in equation 10. This change enhances the matching percentage by 28%. As it was 57.2% and by

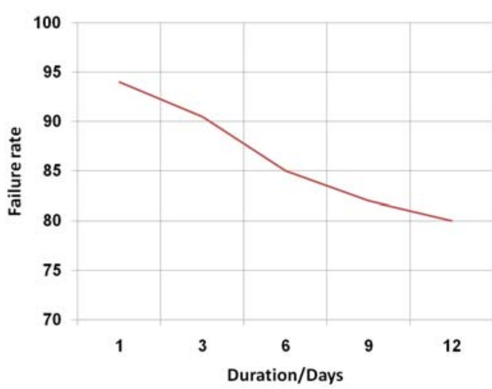

Figure 2. POIs failure rate

this change, it reaches 85.7%. After these two changes, the algorithm gives better results when compared to the expert's selection.

We experimentally tune the algorithm parameters. The colony size is defined to be 50 (total number of POIs). The Maximum Cycle Number (MCN) which represents the number of iterations depends on the tourist stay duration on the country. The number of iteration is increased according to the duration because if the number of stay days for the tourist increase then the possible enumerations of the tours will increase. Experimentally we found that such modification improves the obtained results. The relationship between the duration in days and the number of iterations is shown in Figure 1. As shown the number of iterations is decreased when the duration is increased. Therefore, we consider MCN value 30 which will give better results for all the tested durations. At each iteration, the number of selected POIs which did not appear in the final schedule is calculated and we call this the failure rate (see Figure 2).

As indicated by the expert, at each POI the tourist may take less time than the fixed time specified in the dataset. In this case, the tourists have extra time to visit other close POIs. The tourist guide normally gives a set of suggested POIs and the tourist select from the proposed list. The system is built to deal with this requirement by allowing the tourist to select a new POI that is not proposed in the schedule.

Experiment settings: The system is tested on Samsung S-advance and Galaxy tablet 7.0 with Android 3.4. Figures 3 and 4 show the system screens. Figures 4 shows the data that must be entered from the tourist such as the visit duration, the tourism type, the departure point, and the preferred city. The user can see the selected POIs to be visited as a list on the map or he can access the accommodations in these locations from the menu screen shown at the left of Figure 4. The generated route with POIs is shown at the right of Figure 3 on the map based on the user selection to stay for 3 days. Table 2 extends the Table 1 by computing the precision and the recall. The start city is the departure city. The tour duration represents how many days the tourist wants to stay in the country. The correct POI is the number of sites selected by the algorithm which matches the expert selection. The number of visited POI selected by the expert is based on the

Journal of Theoretical and Applied Information Technology
15th September 2019. Vol.97. No 17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

No.
1
2
3
4
5

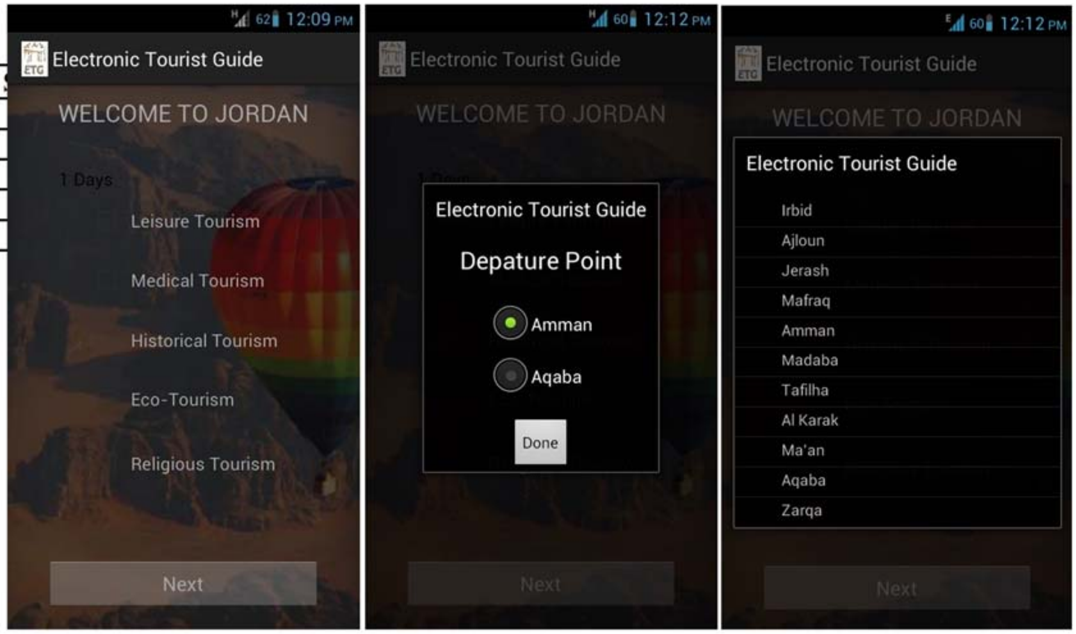

Figure 3. The system introductory screens

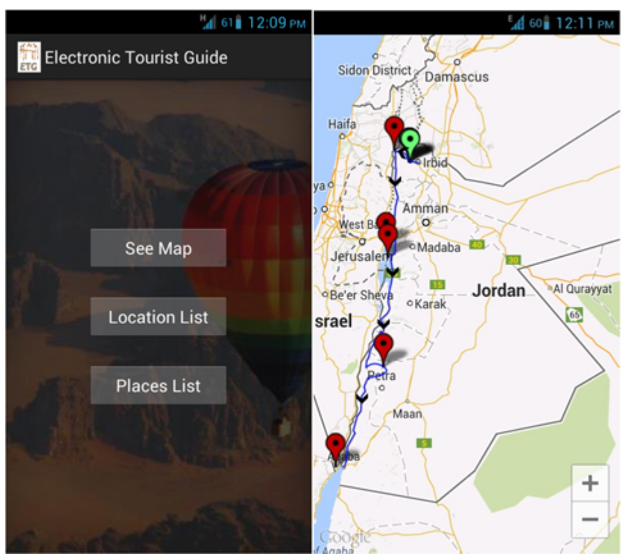

Figure 4. The Selected POIs and the Navigation Path

start city and the tour duration. The algorithm response time is in seconds and it shows how much time it takes the algorithm to generate the tour plan. The precision is the number of POI generated by the algorithm that matches the POI suggested by the expert over the total number of POI. The recall is the number of POI generated by the algorithm that matches the POI suggested by the expert over the total number of POI suggested b the expert. For example, line one represents a run of the tour plan with three days duration. The tourist may visit five POI; three POIs match the POI selected by the expert and two of them are not. Therefore, the precision is 0.06 and the recall is 0.6. In addition, the time required to display each tour plan is indicated in the table.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

7. CONCLUSION AND FUTURE WORK

Tourist route planning is considered an NP-complete problem which is not easy to solve. In this paper, we use the artificial bee colony algorithm for selecting POIs and then generating the route between POI's according to user preferences. We created a dataset for 50 points of interest in Jordan and this dataset is used for evaluating the proposed algorithm. The algorithm is evaluated in terms of the generated schedule accuracy. The proposed algorithm is tuned based on different experimental compared against routes generated by the expert. The proposed algorithm evaluation shows very good results. It provides better solution after the results were discussed. As a future work, the optimization may be done over the route. After the POI selection, the route can be built to obtain the best path. The route can have a rate based on several criteria. The ABC may be applied to connect the selected points by the highly rated paths. The factors that affect the path selection are path safety, path beauty, and the distance which reflect the cost. In addition, the departure point can be selected automatically based on user's current location via Google API or Geo Coder API. Also, the proposed method can be compared with a multi-objective variation of the proposed technique. Finally, user interactive tool can be suggested, as users can add the rating to the selected tours and add new tourism spots.

ACKNOWLEDGMENT

This research was supported by the Ministry of Higher Education and Scientific Research in Jordan (21,000 JD ~ 26600 USD).

REFERENCES:

[1] Alarabalyawm. http://alarabalyawm.net/?p=121052.

[2] Jordan. http://ar.visitjordan.com/Home.aspx.

[3] Ministry of Tourism. http://www.mota.gov.jo/ar/Default.aspx?tabid=120.

[4] Ministry of Tourism and Antiquities. http://www.mota.gov.jo/en/.

[5] Abbaspour, R. A., & Samadzadegan, F. (2011). Time-dependent personal tour planning and scheduling in metropolises. *Expert Systems with Applications*, 38(10), 12439-12452.

[6] Abu Doush, I., Alshatnawi, S., Al-Tamimi, A. K., Alhasan, B., & Hamasha, S. (2016). ISAB: integrated indoor navigation system for the blind. *Interacting with Computers*, 29(2), 181-202.

[7] Aifadopoulou, G., Ziliaskopoulos, A., & Chrisohoou, E. (2007). Multiobjective optimum path algorithm for passenger pretrip planning in multimodal transportation networks. *Transportation Research Record*, 2032(1), 26-34.

[8] Alshattnawi, S. (2013). Building mobile tourist guide applications using different development mobile platforms. *International Journal of Advanced Science and Technology*, 54, 13-22.

[9] Alshattnawi, S. (2013). Utilizing Cloud Computing in Developing a Mobile Location-Aware Tourist Guide System. *International Journal of Advanced Pervasive and Ubiquitous Computing (IJAPUC),* 5(2), 9-18.

[10] Archetti, C., Hertz, A., & Speranza, M. G. (2007). Metaheuristics for the team orienteering problem. *Journal of Heuristics*, 13(1), 49-76.

[11] Ausiello, G., Bonifaci, V., & Laura, L. (2008). The online prize-collecting traveling salesman problem. *Information Processing Letters*, 107(6), 199-204.

[12] Azar, Y., & Vardi, A. (2016, August). Dynamic Traveling Repair Problem with an Arbitrary Time Window. *In International Workshop on Approximation and Online Algorithms* (pp. 14-26). Springer, Cham.

[13] Bar-Yehuda, R., Even, G., & Shahar, S. M. (2005). On approximating a geometric prize-collecting traveling salesman problem with time windows. *Journal of Algorithms*, 55(1), 76-92.

[14] Bhagade, A. S., & Puranik, P. V. (2012). Artificial bee colony (ABC) algorithm for vehicle routing optimization problem. International Journal of Soft Computing and Engineering, 2(2), 329-333.

[15] Cergibozan, Ç., & Tasan, A. S. (2018). Tourist Route Planning with a Metaheuristic Approach. In *Closing the Gap Between Practice and Research in Industrial Engineering* (pp. 193-199). Springer, Cham.

[16] Chao, I. M., Golden, B. L., & Wasil, E. A. (1996). The team orienteering problem. *European journal of operational research*, *88*(3), 464-474.

[17] Chaves, A. A., & Lorena, L. A. N. (2008, March). Hybrid metaheuristic for the prize collecting travelling salesman problem. In *European Conference on Evolutionary Computation in Combinatorial Optimization* (pp. 123-134). Springer, Berlin, Heidelberg.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[18] Chen, C., Du, H., & Lin, S. (2017, June). Mobile robot wall-following control by improved artificial bee colony algorithm to design a compensatory fuzzy logic controller. In *2017 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)* (pp. 856-859). IEEE.

[19] Chong, C. S., Sivakumar, A. I., Low, M. Y. H., & Gay, K. L. (2006, December). A bee colony optimization algorithm to job shop scheduling. In *Proceedings of the 38th conference on Winter simulation* (pp. 1954-1961). Winter Simulation Conference.

[20] Pan, Q. K., Tasgetiren, M. F., Suganthan, P. N., & Chua, T. J. (2011). A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem. *Information sciences*, *181*(12), 2455-2468.

[21] Claes, R., & Holvoet, T. (2011, May). Ant colony optimization applied to route planning using link travel time predictions. In *2011 IEEE International Symposium on Parallel and Distributed Processing Workshops and Phd Forum* (pp. 358-365). IEEE.

[22] Clauß, M., Lotzmann, L., & Middendorf, M. (2016, May). A Population Based ACO Algorithm for the Combined Tours TSP Problem. In *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)* (pp. 128-135). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[23] Doush, I. A., Alshattnawi, S., & Barhoush, M. (2015). Non-visual navigation interface for completing tasks with a predefined order using mobile phone: a case study of pilgrimage. *International Journal of Mobile Network Design and Innovation*, *6*(1), 1-13.

[24] Doush, I. A., Hasan, B. H. F., Al-Betar, M. A., Al Maghayreh, E., Alkhateeb, F., & Hamdan, M. (2014). Artificial Bee Colony with Different Mutation Schemes: A comparative study. *Computer Science Journal of Moldova*, *22*(1).

[25] Feillet, D., Dejax, P., & Gendreau, M. (2005). Traveling salesman problems with profits. *Transportation science*, *39*(2), 188-205.

[26] Han, Y., Guan, H., & Duan, J. (2014). Tour route multiobjective optimization design based on the tourist satisfaction. *Discrete Dynamics in Nature and Society*, *2014*.

[27] Hasuike, T., Katagiri, H., Tsubaki, H., & Tsuda, H. (2013). Tour planning for sightseeing with time-dependent satisfactions of activities and traveling times. *American Journal of Operations Research*, *3*(03), 369.

[28] Herath, H. M. R. P. B., & Ratnayake, H. U. W. (2013, April). Multi agent system for trip planning. In *2013 8th International Conference on Computer Science & Education* (pp. 298-303). IEEE.

[29] Joest, M., & Stille, W. (2002, November). A user-aware tour proposal framework using a hybrid optimization approach. In *Proceedings of the 10th ACM international symposium on Advances in geographic information systems* (pp. 81-87). ACM.

[30] Karaboga, D., & Basturk, B. (2008). On the performance of artificial bee colony (ABC) algorithm. *Applied soft computing*, *8*(1), 687-697.

[31] Ke, L., Archetti, C., & Feng, Z. (2008). Ants can solve the team orienteering problem. *Computers & Industrial Engineering*, *54*(3), 648-665.

[32] Keshtkaran, M., Ziarati, K., Bettinelli, A., & Vigo, D. (2016). Enhanced exact solution methods for the team orienteering problem. *International Journal of Production Research*, *54*(2), 591-601.

[33] Lin, J. H., & Huang, L. R. (2009, March). Chaotic bee swarm optimization algorithm for path planning of mobile robots. In *Proceedings of the 10th WSEAS international conference on evolutionary computing* (pp. 84-89). World Scientific and Engineering Academy and Society (WSEAS).

[34] Lučić, P., & Teodorović, D. (2003). Computing with bees: attacking complex transportation engineering problems. *International Journal on Artificial Intelligence Tools*, *12*(03), 375-394.

[35] Maruyama, A., Shibata, N., Murata, Y., Yasumoto, K., & Ito, M. (2004, March). A personal tourism navigation system to support traveling multiple destinations with time restrictions. In *18th International Conference on Advanced Information Networking and Applications, 2004. AINA 2004.* (Vol. 2, pp. 18-21). IEEE.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[36] Montemanni, R., Weyland, D., & Gambardella, L. M. (2011, July). An enhanced ant colony system for the team orienteering problem with time windows. In *2011 International Symposium on Computer Science and Society* (pp. 381-384). IEEE.

[37] Nagata, M., Murata, Y., Shibata, N., Yasumoto, K., & Ito, M. (2006, October). A method to plan group tours with joining and forking. In *Asia-Pacific Conference on Simulated Evolution and Learning* (pp. 881-888). Springer, Berlin, Heidelberg.

[38] Pan, Q. K., Tasgetiren, M. F., Suganthan, P. N., & Chua, T. J. (2011). A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem. *Information sciences*, *181*(12), 2455-2468.

[39] Pandey, S., & Kumar, S. (2013). Enhanced artificial bee colony algorithm and it's application to travelling salesman problem. *HCTL Open International Journal of Technology Innovations and Research*, *2*.

[40] Paul, A., Freund, D., Ferber, A., Shmoys, D. B., & Williamson, D. P. (2017). Prize-collecting TSP with a budget constraint. In *25th Annual European Symposium on Algorithms (ESA 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[41] Peng, Z. R., & Huang, R. (2000). Design and development of interactive trip planning for web-based transit information systems. *Transportation Research Part C: Emerging Technologies*, *8*(1-6), 409-425.

[42] Rodríguez, B., Molina, J., Pérez, F., & Caballero, R. (2012). Interactive design of personalised tourism routes. *Tourism Management*, *33*(4), 926-940.

[43] Schilde, M., Doerner, K. F., Hartl, R. F., & Kiechle, G. (2009). Metaheuristics for the bi-objective orienteering problem. *Swarm Intelligence*, *3*(3), 179-201.

[44] Sonmez, M., Akgüngör, A. P., & Bektaş, S. (2017). Estimating transportation energy demand in Turkey using the artificial bee colony algorithm. *Energy*, *122*, 301-310.

[45] Souffriau, W., Vansteenwegen, P., Vertommen, J., Berghe, G. V., & Oudheusden, D. V. (2008). A personalized tourist trip design algorithm for mobile tourist guides. *Applied Artificial Intelligence*, *22*(10), 964-985.

[46] Tang, H., & Miller-Hooks, E. (2005). A tabu search heuristic for the team orienteering problem. *Computers & Operations Research*, *32*(6), 1379-1407.

[47] Ten Hagen, K., Kramer, R., Hermkes, M., Schumann, B., & Mueller, P. (2005). Semantic matching and heuristic search for a dynamic tour guide. In *Information and Communication Technologies in Tourism 2005* (pp. 149-159). Springer, Vienna.

[48] Tereshko, V. (2000, September). Reaction-diffusion model of a honeybee colony's foraging behaviour. In *International Conference on Parallel Problem Solving from Nature* (pp. 807-816). Springer, Berlin, Heidelberg.

[49] Vansteenwegen, P., Souffriau, W., Berghe, G. V., & Van Oudheusden, D. (2009). A guided local search metaheuristic for the team orienteering problem. *European journal of operational research*, *196*(1), 118-127.

[50] Wang, Q., Sun, X., Golden, B. L., & Jia, J. (1995). Using artificial neural networks to solve the orienteering problem. *Annals of Operations Research*, *61*(1), 111-120.

[51] Wong, L. P., Low, M. Y. H., & Chong, C. S. (2009, June). An efficient bee colony optimization algorithm for traveling salesman problem using frequency-based pruning. In *2009 7th IEEE International Conference on Industrial Informatics*(pp. 775-782). IEEE.

[52] Wong, L. P., Low, M. Y. H., & Chong, C. S. (2009, June). An efficient bee colony optimization algorithm for traveling salesman problem using frequency-based pruning. In *2009 7th IEEE International Conference on Industrial Informatics*(pp. 775-782). IEEE.

[53] Xu, Y., Fan, P., & Yuan, L. (2013). A simple and efficient artificial bee colony algorithm. *Mathematical Problems in Engineering*, *2013*.

[54] Yu, H., & Lu, F. (2012). A multi-modal route planning approach with an improved genetic algorithm. *Advances in Geo-Spatial Information Science*, *193*.

[55] Zehetner, M., & Gutjahr, W. J. (2018). Sampling-Based Genetic Algorithms for the Bi-Objective Stochastic Covering Tour Problem. In *Recent Developments in Metaheuristics* (pp. 253-284). Springer, Cham.

[56] Zheng, W., Liao, Z., & Qin, J. (2017). Using a four-step heuristic algorithm to design personalized day tour route within a tourist attraction. *Tourism Management*, *62*, 335-349.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[57] Zhu, C., Hu, J. Q., Wang, F., Xu, Y., & Cao, R. (2012). On the tour planning problem. *Annals of Operations Research*, *192*(1), 67-86.

[58] Zografos, K. G., & Androutsopoulos, K. N. (2008). Algorithms for itinerary planning in multimodal transportation networks. *IEEE Transactions on Intelligent Transportation Systems*, *9*(1), 175-184.

[59] Alsobeh, A. M., Magableh, & AlSukhni, E. M. (2018). Runtime Reusable Weaving Model for Cloud Services Using Aspect-Oriented Programming: The Security-Related Aspect. International Journal of Web Services Research (IJWSR), 15(1), 71-88.