Using Ant Colony Optimization for Tourist Route Construction Automation

Boris Skripal
Peter the Great St. Petersburg Polytechnic University
29 Polytechnicheskaya st.
195251 St. Petersburg Russia
skipalboris@gmail.com

Evgeny Pyshkin
University of Aizu
Software Engineering Lab
Tsuruga, Ikki-machi, Aizu-Wakamatsu
Fukushima, Japan 965-8580
pyshe@u-aizu.ac.jp

ABSTRACT

In this paper an ant colony optimization (ACO) adoption for tourist itinerary planning automation is presented. We introduce a model describing a set of features used in our component for itinerary planning automation and study how an ACO algorithm could be used for the purposes of constructing an attractive tourist route.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Automation, Human Factors

Keywords

Information systems, Travel itinerary, ACO (Ant Colony Optimization)

1. INTRODUCTION

Tourist route planning automation is often considered as a generalization of shortest path algorithms, however there are many competitive factors making difficult to rely only on the minimization algorithms straightforwardly. Many connected aspects have to be taken into consideration including but not limited to the following ones:

- Tourists usually don't want to reach a point of interest in a shortest possible time, they rather expect to arrange their leisure time in the best way.
- Departure and arrival points are often the same (so, you would probably like to be back to your hotel after discovering a certain set of attractions).
- Besides walk distance, there are other factors affecting an itinerary score: object interestingness, ticket prices and related expenses, probability of being too crowded, opening hours, etc.
- A route shouldn't include only historical or cultural attractions, food, relaxation and sanitary points are also important.
- A route might be strongly focused (in a case when it includes a selection of objects of similar type, e.g. mostly churches, or monuments, or art exhibitions, etc.) or less focused (if there is a balance between attractions of different type).

More aspects we take into consideration, more accurate planning we can expect. There are at least three major problems to be resolved while implementing route planning systems including a problem of mining data required for a future route [5], a problem of constructing an algorithm for itinerary planning, and a problem of implementing a good presentation of the proposed route [4].

In this paper we pay a particular attention to developing an algorithm for planning tourist routes on the base of an ant colony optimization modification aimed at taking into accounts some competitive factors listed above.

Based on [2] we use a model of a tourist route [1] represented as a hierarchy of concepts describing tourist-related information (such as expenses, speed and energy), attraction-related information (such as name, type, visiting fees, walking time, location and attractiveness score) and visit-related information (such as a list of included attractions, visit time and duration).

2. RELATED WORK

The standard task of tourist route generation is formalized by Souffriau [7] as follows:

Assume there are N points of interest (POIs), each point can be connected to other ones (but a departure point isn't connected with a destination point). Each POI i has a score $S_i \geqslant 0$, where for a departure point $i = 1$, for the destination point $i = N$. The shortest path between points i to j requires time t_{ij}, the total score S_{total} has to be maximized under a time limit constraint T_{max}. This model sets the future route boundaries and determines criteria for the best tourist route such as a best attraction selection-best path combination.

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

---

[1] The proposed model is not complete, but provides a set of basic features which are enough to illustrate our vision.

In this study we investigate a possibility to adopt an ACO algorithm for automated itinerary construction. A basic ACO algorithm (usually used for finding shortest paths in graphs [8]) is based on a probability of ant moves from state x to state y:

$$p^k_{xy} = \frac{(\tau^\alpha_{xy})(\eta^\beta_{xy})}{\sum_{z\in allowed_z}(\tau^\alpha_{xz})(\eta^\beta_{xz})} \tag{1}$$

In equation (1) α and β are coefficients used to manage τ_{xy} and η_{xy}, τ_{xy} representing the amount of pheromone deposited for transition from state x to y and η_{xy} – the desirability for xy transition.

A pheromone update conditioned by a transition is as follows:

$$\tau_{xy}(t+1) = (1-p)\tau_{xy}(t) + \sum_k \Delta\tau^k_{xy}(t) \tag{2}$$

In equation (2) $0 < p < 1$ is the pheromone trail evaporation, $\Delta tau^k_{xy}(t)$ is the amount of pheromone that an ant k puts on the arcs it visits (defined in the formula 3).

$$\Delta\tau^k_{xy}(t) = \begin{cases} 1/L^k(t), & \text{if the path \{xy\} is used by the ant } k \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Due to the randomization and a kind of feedback based schema, ACO allows discovering good solutions rapidly without loosing local maximums. This is a reason to use ACO as a foundation for our purposes.

One known modification of ACO for creating tourist routes was described by Huang Han-Chen [3]. Its general idea is to extend a pheromone update function 2 by an additional parameter $\Delta\tau^*_{xy} = \sigma\frac{Q}{L^*}$, where σ is the number of elite ants, L^* is the route length of the determined optimal solution.

Another modification was introduced in [1] by Claes and Holvoet. Their idea is to extend the list of basic ACO parameters by adding transition speed from i-th vertex to j-th, as well as the path time and some others.

3. ADOPTING ACO FOR ROUTE CONSTRUCTION AUTOMATION

In order to pay attention to the POI types, we defined a table of preferences containing the preference coefficients for different types of objects (see Table 1 as an example).

Table 1: Preference Coefficients

Museums	Monuments	Parks	Restaurants	Cafes
1	0.8	1.2	0.6	0

There is a multiplication coefficient $\acute{p}$: $0 \leq \acute{p}$ for each type of objects, so as 0 corresponds to the lowest preference level (such an object will never be selected).

In order to improve algorithm performance we make preliminary selection of the objects considered for inclusion to the route. As we described in [6], for object selection we propose to use an elliptical area so as to allow selecting reachable objects which are inside the ellipse. The reason of using an elliptical area is that an ellipse has such a nice property that the sum of distances from every point on the curve to two focal points is constant. Hence, the sum of distances to the two focal points is set as a maximum possible distance: any point inside the curve can be reached while points outside the curve are unreachable. Semi-major axis (a) and semi-minor axis (b) are defined as follows:

$$a = \sqrt{(X_{start} - X_{finish})^2 + (Y_{start} - Y_{finish})^2} \tag{4}$$

$$b = \frac{Vt}{2} - \frac{a}{2} \tag{5}$$

In equation (5) V is the tourist speed, t is the planned route duration.

For a point P_i represented by its coordinates x and y, "being inside the ellipse" condition is defined as follows:

$$\frac{(\acute{x}\cos\alpha + \acute{y}\sin\alpha)^2}{a^2} - \frac{(-\acute{x}\sin\alpha + \acute{y}\cos\alpha)^2}{b^2} \leq 1 \tag{6}$$

In equation (6) $\acute{x} = x - x_0$, $\acute{y} = y - y_0$ (where x_0 and y_0 are the coordinates of the ellipse center).

The points which are outside the elliptic area are automatically excluded from the searching process.

In addition to standard ACO parameters (see equation (1)), each ant is associated with an energy reserve value E which allows taking into account such human (not ant!) properties as being hungry or/and being tired.

Effectively, in i-th step, there is a probability e_i that the next selected point is a food place:

$$e_i = \frac{E_{init} - E}{E_{init}} \tag{7}$$

In equation (7) E_{init} is an initial energy reserve, E is a current energy level. Each time unit (for example, each one minute of walking) the energy reserve score decreases by some value (now we use a fixed value, but it could be also a function of a distance). After visiting a food place, the energy level is restored to E_{init}.

Total score of each route is calculated by using the following equation:

$$S = \frac{t_{see}}{t_{moving}}(-f_e t_{ze} + \sum_{i\in visited} S_i\acute{p}) \tag{8}$$

As you see from equation (8), if E becomes zero, we decrease ant score S_{total} by f_e (f_e being a penalty per time unit).

To calculate the total score, the pheromone level on the best routes is updated after each iteration.

4. A PROCESS OF ROUTE GENERATION

There are some preliminary actions before generating a route. First, we prepare a table of preference (see Table 1). Second, some parameters have to be set:

- Route speed (we use 5 kph as a default value for average pedestrian speed);
- Excursion departure and destination points (they may be the same);
- Excursion departure time and expected arrival time to the destination [2];

---

[2] A set of possible destinations could also be an option. For example, in a big city a traveler might prefer to reach a metro station: sometimes it doesn't matter which one.

- List of filtered POIs (due to being closed or not fitting well user preferences);
- Tourist balance (a non-negative value representing how much money a traveler plans to spend for an excursion);
- Initial energy value and an energy unit used while decreasing the energy level.

Figure 1 illustrates the process of selecting points with the scope of elliptic area and with paying attention to preference constraints.

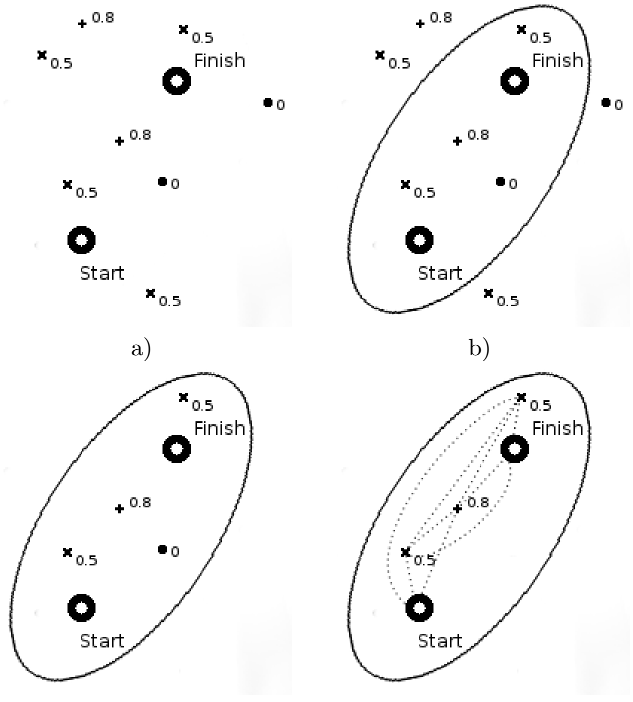

Figure 1: Selecting POIs: a) initial set of POIs (three POI types shown), b) reachable area, c) filtering unreachable POIs, d) removing POIs with zero preference level or those which aren't open for visiting (dashed lines show possible ways between points).

We start a route generation procedure from creating some ants and setting required parameters (initial time, departure point, etc.). For each ant there are the following steps:

1. If there are POIs which are unreachable within the limits of the planned period of time (for example, they might be closed or they might delay significantly the total route duration due to a long waiting time at the entrance) such POIs are filtered.
2. If there is no more POIs to consider, move to the finish and stop searching.
3. If there is an option to include a food point (and there are some available), select a food point with probability e_i calculated by using equation (7) (otherwise, an attraction point is selected, respectively, with probability $1 - e_i$.
4. Proceed with a standard ACO iteration: select the next POI to visit by using equation (1) and move to the selected POI.
5. Update current ant time ($t_{new} = t_{prev} + t_{moving} + t_{visit}$), the energy level, as well as moving time and visiting time (if we visit a food place, we don't include object visiting time), balance and pheromone on this point by using formula (2).
6. Go to the step 1.

When all ants achieved the finish point, we select N best routes (each route is evaluated by using formula (8)) and move "elite" ants on these routes, for update best routes (increase τ of all POIs in the best path).

Then a next set of ants is generated and the above procedure is repeated. The final route is selected among the best routes obtained during the above described iterations.

5. CONCLUSIONS

In this paper we described an ACO modification to be used for tourist route generation. Further investigations are required in order to evaluate whether such an approach is appropriate for creating interesting routes under a selection of constraints and route properties. In the future we plan to consider such parameters as season dependency, visiting time periods (to avoid crowds), leveraging information about tourist preferences achieved from thematic web sites. An interesting issue is generating multi-day or thematically connected routes considered as the parts of one journey.

6. REFERENCES

[1] R. Claes and T. Holvoet. Ant colony optimization applied to route planning using link travel time predictions. *IEEE International Parallel and Distributed Processing Symposium*, 2011.

[2] D. Gavalas, M. Kenteris, C. Konstantopoulos, and G. Pantziou. An efficient algorithm for recommending personalized mobile tourist routes. *IET Software*, 2011.

[3] H. Huang. The application of ant colony optimisation algorithm in tour route planning. *Journal of Theoretical and Applied Information Technology*, 2013.

[4] Y. Huang, C. Liu, C. Lee, and Y. Huang. Designing a personalized guide recommendation system to mitigate information overload in museum learning. *Educational Technology and Society, 15*, 2012.

[5] M. Karathiya. Data mining for travel and tourism. *Journal of Information and Operations Management*, 2012.

[6] B. Skripal and E. Pyshkin. Automated leisure walk route generation for an interactive travel planner. In *Proceedings of the International Workshop on Applications in Information Technology (IWAIT-2015)*, pages 29–32. The University of Aizu, The University of Aizu Press, Oct 2015.

[7] W. Souffriau. *Automated Tourist Decision Support.* Katholieke Universiteit Leuven, 2010.

[8] T. Stutzle and M. Dorigo. Ant colony optimization: Overview and recent advances. *IRIDIA, Institut de Recherches Interdisciplinaires et de Developpements en Intelligence Artificielle*, 2009.