# Route Planning in a New Tourist Recommender System: a Fireworks Algorithm Based Approach

Hui Ding, Liangjun Ke
State Key Laboratory for Manufacturing System Engineering
Xi'an Jiaotong University
Email: dh.redmaple@stu.xjtu.edu.cn

Zihao Geng
School of Computer and Information Technology
Beijing Jiaotong University

***Abstract*—With the development of online tourist, Tourist Recommender System (TRS) has become a hot research topic. This paper presents a TRS and considers a new tourist trip design problem (TTDP), taking into account the compactness of the trip. To solve this problem, fireworks algorithm (FWA) is adopted. As a meta-heuristic method, FWA has been widely used in continuous problems, while TTDP is a discrete optimization problem with various constraints, the key difference between them lies in the definition of distance. In this paper, operators of the conventional FWA are redesigned to handle with TTDP. Experimental results demonstrate the effectiveness of the proposed FWA.**

***Keywords—fireworks algorithm; tourism recommendation; TTDP;***

## I. Introduction

As a major part of the modern service industry, tourism has experienced a rapid growth over the past decade. With the expansion of the Internet, online tourism has been enormously developed. Statistics in fig.1 shows that the market scale of online tourism in China has ballooned with an average annual growth rate of 30 percent since 2012, and will be doubled before 2018. Compared with the time-consuming traditional approaches, online tourism allows users to plan their trip and complete the booking whenever they want, which greatly fosters their passion to travel.

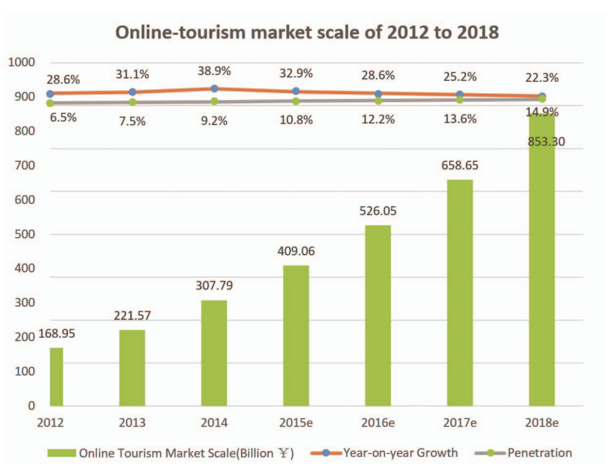


Fig. 1. Online-tourism market scale of China

[a] Source: Report@ www.iresearch.cn

Nevertheless, with the assistance of Internet, everyone has become information producer, which has caused severe information overload. Namely, it's hard to locate the required information from the big data. So far, two approaches have been utilized to solve this problem. The first one is information retrieval technique of which the representative is search engine, and the second one is information filtering technique of which the representative recommender systems. The most significant difference between them is that the result of search engine relies heavily on the accuracy of the problem description, while recommender system generates recommendations by automatically exploiting users' profile data and historical activities, thus it's widely applied in many fields.

For most tourists, they may have limited time and budget available, thus only the most interesting point-of-interests (POIs) can be visited. In that case, the planning problem of online tourism can be divided into two independent processes: a) Selection of the POIs, i.e. to choose the most favorite POIs to visit; b) trip design process, i.e. to design the traveling routes according to user's preferences and constraints of the POIs. In this paper, we mainly focus on trip design problem.

In the literature, the generation of personalized traveling routes is also called Tourist Trip Design Problem (TTDP) [1], which aims to generate the optimal traveling routes under the constraints of time and cost. Authors in [2] surveyed the information considered in the formulation of the problem:

a) A set of candidate POIs, each is associated with a number of attributes (e.g. type, location, opening days/hours, etc.).
b) The travel time among POIs.
c) "Profit" or score of each POI, according to user preference.
d) Visiting time of each POI.
e) The number of routes to be generated.
f) The daily time limit that a tourist wishes to spend on visiting POIs.

The TTDP is NP-hard [3], hence, it is difficult to solve exactly. Authors in [4] applied Guided Local Search (GLS) to solve the simplest form of TTDP, in which the result outperforms the Dynamic Tour Guide systems for problem instances with over 50 POIs. They extended their approach by taking opening hours of POIs into account on a mobile device, with limited computational resources in [5]. In [6], the authors employ the greedy randomized adaptive search procedure (GRASP) to solve TTDP, which is a metaheuristic originally proposed by Feo and Resende in [7]. Authors in [8] presented CSCRatio and CSCRoutes, two cluster-based approaches for the TTDP, and they employed the algorithms to solve the time-dependent TTDP (TD-TTDP), and designed TD_CSCR and TD_SlCSCR [9].

Mostly, the designed trip is affected by some other factors, such as the reserved flights, or the physical strength of visitors. However, to the best of our knowledge, it is seldom considered that the schedule need to be more tight or relaxed. Besides, result of the questionnaire in [Han et.al 2014] shows that the stay time around a POI is positively related to the preference degree of it, namely, visitors are likely to spend more time around their favorite POIs. To further enhance the satisfaction of the trip, the metric of compactness is introduced in this paper, which is defined as the degree of relaxation, based upon the number of POIs per day and ratio between visiting time with that of transportation. Accordingly, the new "profit" of the POI will be redefined and the stay time for the specified POI will be recalculated. In this way, we propose a new model for the problem of trip plan.

The other goal of this paper is to employ a new heuristic to solve the new model. As a swarm intelligence algorithm, Fireworks Algorithm (FWA) has been designed for continuous optimization problems [10]. It can deal with linear or non-linear and multi-model test functions and is suitable to implement in parallel. Most important of all, FWA has a fast convergence speed and can find the global optimal solutions. These virtues make it be recognized as one of the best intelligent optimization algorithms. The application of FWA includes non-negative matrix factorization [11], image identification [12], Spam Detection [13] etc.

However, FWA is seldom applied on discrete optimization problems [14]. This mainly due to the difference between the definition of distance for discrete space with that of continuous space, which plays an important role in the conventional FWA. For example, the explosion operator needs to search the local space within an explosion amplitude, the mutation operator needs to choose several dimensions to scale, and the selection strategy needs to reserve the fireworks according to the density. To apply FWA in the TTDP, all these operators need to be redesigned.

The rest of this paper is organized as follows: in Section II , the proposed recommender system is introduced including the framework and the recommendation algorithm, in Section III, the new TTDP model, taking into account the compactness of the trip is presented, and a discrete FWA with new operators is proposed. The experimental setting and results are displayed in Section IV. The conclusion of this paper and the system demo is presented in Section V.

## II. OUR RECOMMENDER SYSTEM

### *A. Framework of Our System*

The framework of the proposed recommender system is shown in Fig.2. Before recommendation, information of users are collected and stored in the user database, then the score of each POI is estimated by the recommender module, according to the user's preference and the features of POIs. The score of a POI can be viewed as its reward when visited. After that, the tourist trip design module will design detailed routes for the user, satisfying all the restrictive conditions. In the end, the detailed routes is displayed to the users in a web page, and they can edit the routes as they wish.

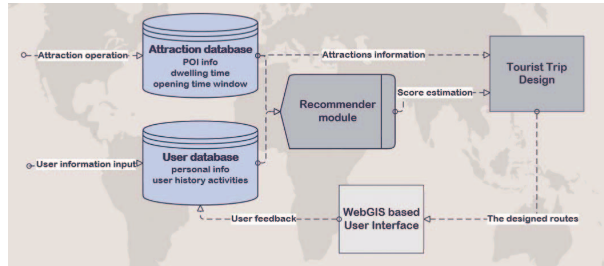


Fig. 2. Framework of our recommender system

### *B. User Feature Selection*

In this paper, the information of users are collected in an explicit way, which includes: a) age and gender, as visitors of different age or gender often have difference taste for the POI and different physical strength; b) income, which has much effect on the accommodation, transportation, and other price-sensitive aspects; c) preference feature, which determines the preference degree to specified POI categories. Besides, other constraints are also obtained such as scheduled flights or other transportation, budget of this trip, etc.

### *C. User Feature Selection*

The accuracy of the classification for POIs has a serious influence on the recommendation results. In this paper, the POIs are classified according to the categories in the National Standards of PRC: GB/T18972-2003, which includes topography, water area landscape, cities and Ruins, architecture and facilities, etc. More often, a POI may have many features and can't be entirely sorted into a single category, thus we classify the POIs based upon their membership towards each category, and the classification is represented by a vector of eight dimensions.

Our system is tested upon the 264 POIs around Xi'an city, Shaanxi province of China. To improve the classification accuracy, five tourism enthusiasts are invited to classify these POIs. After classification, final score of each POI is a weighted sum of the following scores:

a) Preference score: calculated by the Knowledge-Based recommendation algorithm;

b) Popularity score: the number of visitors who have been there before;

c) History score: the average scores given by the visitors who have been there before.

## III. Tourist Trip Design Problem

After the score estimation, the tourist trip design problem can be formulated as a searching procedure within a complete undirected graph, which consists of a set of vertices and a set of arcs. Each vertex corresponds to a POI, described as a five-tuple $\{S_i, s_i, C_i, c_i, T_i\}$ where $S_i$ is the final score of each POI, $s_i$ and $c_i$ are the open time and close time, $C_i$ denotes the entrance fee of the POI, and $T_i$ is the recommended stay time; each arc $d_{ij}$ represents the traveling time between every two POIs. This optimal trip should also meet the following constraints: a) the travel time constraint (i.e. the time corresponds to each route should be no more than a time limit), b) The time window, c) the budget constraint. The target of the problem is to maximize the total "profit" or satisfaction.

### A. Problem formulation

As mentioned above, δ is introduced as a parameter to describe the compactness of the trip. It is defined in the interval of [0, 1], 0.5 is marked as the normal state. The stay time of each POI depends on both the preference score and the compactness, which can be depicted as follows:

$$T_i^{'} = F(T_{\max}, T_{\min}, \delta, S_i) \qquad (1)$$

$T_{\max}, T_{\min}$ are the upper bound and lower bound of recommended visiting time respectively.

In the literature, the problem is often modeled as TOPTW (team orienteering problem with time window) with service time, and the objective function is defined as the sum of the "profit" obtained. Herein, a tradeoff of the compactness and the comfort is applied in the objective function.

$$\max \sum_{d=1}^{m}\sum_{i=2}^{n-1} S_i y_{id} + 10*\left(comp-0.5\right)*\left(\sum_{d=1}^{m}\sum_{i=2}^{n-1} y_{id} + \frac{T_{visit}}{T_{total}}*10\right) \qquad (2)$$

$$T_{visit} = \sum_{d=1}^{m}\sum_{i=2}^{n-1} T_i y_{id}$$

Where n is the number of the POIs, m is the scheduled days to visit, $y_{id}$ is 1 if the $i_{th}$ POI is visited on day d, and 0 otherwise.

Besides, the final trip plan should meet the constraints caused by budget $\mathrm{cost}_{\lim it}$, daily traveling limit $T_{\lim it}$, and the opening windows of each POI, as follows:

$$\sum_{d=1}^{m}\sum_{i=2}^{n-1} C_i y_{id} \le \cos t_{\lim it} \qquad (3)$$

$$\sum_{i=1}^{n-1}(T_i y_{id} + \sum_{j=2}^{n} c_{ij} x_{ijd}) \le T_{\lim it}$$

$$s_{id} + T_i + d_{ij} - s_{jd} \le M*(1 - x_{ijd})$$

### B. FWA

The mechanism of FWA is composed of three process: initialization, local search, selection strategy.

- Initialization

  Randomly select N positions in the solution space, set off firework at each position;

- Local search

  After the fireworks are set off, a shower of sparks will fill the local space around the firework, which can be seen as the local search within the neighborhood. In this process, two kinds of sparks will be generated, the explosion spark and the mutation spark. The former one is designed to search the local space, while the latter one is to enhance the diversity of the fireworks. The detailed operators are defined as below:

a) Explosion Operator

the FWA incorporates an automatic procedure to balance exploration and exploitation capabilities. Fireworks with better fitness will have a smaller explosion amplitude $A_i$ and a larger number of explosion sparks $S_i$, for each firework $X_i$, the explosion amplitude and number of explosion sparks is calculated as follows:

$$A_i = \hat{A} * \frac{f(X_i - y_{\min}) + \varepsilon}{\sum_{i=1}^{N}(f(X_i) - y_{\min}) + \varepsilon} \qquad (4)$$

$$S_i = M_e * \frac{y_{\max} - f(X_i) + \varepsilon}{\sum_{i=1}^{N}(y_{\max} - f(X_i)) + \varepsilon}$$

$y_{\min} = \min(\mathrm{f}(X_i))$, $y_{\max} = \max(f(X_i))$, $\hat{A}$ and $M_e$ are two constants to control the explosion amplitude and the number of explosion sparks respectively, and $\varepsilon$ is a tiny parameter to avoid zero divisor. Besides, a strategy is applied to restrict the number of explosion sparks within the upper bound $aM_e$ and lower bound $\mathrm{b}M_e$.

b) Mutation Operator

After explosion, $M_g$ Gaussian mutation sparks are generated, by randomly select k dimensions from the $M_g$ fireworks to multiple by e.

$$\widehat{X_{ik}} = X_{ik} * e \qquad (5)$$

Where $e \sim N(1,1)$, which is a Gaussian distribution in which both the mean value and the variance are 1. When the result is out of boundary, a mapping rule is applied.

- Selection Strategy

A new population of fireworks is selected from the candidate fireworks set K at the end of each iteration. The candidate set K includes the current fireworks, explosion sparks, and mutation sparks. The current best solution is always kept down to the next generation. For other candidates, the selection probability $P(X_i)$ is calculated by:

$$P(X_i) = \frac{R(X_i)}{\sum_{X_j \in K} X_j} \tag{6}$$

$$R(X_i) = \sum_{X_j \in \mathrm{K}} d(X_i - X_j) = \sum_{X_j \in \mathrm{K}} \| X_i - X_j \|$$

Where $R(X_i)$ is the sum of the distance from current firework to all the fireworks in the candidate set K. Thus the probability would decrease when the density of the fireworks is too large.

### C. FWA in Tour Planning

As mentioned in the Section I, to apply the FWA in TTDP, the operators need to be redesigned. We mark a feasible solution as $\pi = \{\pi_{11}, \pi_{12}, \ldots, \pi_{1k_1}, \pi_{21}, \ldots, \pi_{mk_m}\}$, where $\pi_{ip} \neq \pi_{jq}$, $\pi_{dk}$ denotes the $k_{th}$ POI planned on day d. For solutions with same POI set, they can be transformed by changing the sequence of POIs, thus the distance is defined as follows:

*Definition*: the distance between two solutions is the number of different nodes regardless of order.

According to this definition, we adopted the same framework of conventional FWA, and redesigned the operators.

a) Explosion operator

According to the new definition of distance, to explore the local space within the explosion amplitude $A_i$, we need to replace at most $A_i$ nodes in the sequence, and rearrange it. In this paper, 2-opt move is applied to improve the solution. The pseudocode is presented in Algorithm 1.

As the problem is a maximization problem, the explosion amplitude and the number of explosion sparks is calculated equation (7).

$$A_i = \hat{A} * \frac{y_{\max} - f(X_i) + \varepsilon}{\sum_{i=1}^{N} (y_{\max} - f(X_j)) + \varepsilon} \tag{7}$$

$$S_i = M_e * \frac{f(X_i) - \mathrm{y}_{\min} + \varepsilon}{\sum_{j=1}^{N} (f(X_j) - \mathrm{y}_{\min}) + \varepsilon}$$

b) Mutation operator

In the optimization process, 2-opt move can only optimize the tour of a single day, it can't optimize the tours of different days when reached a local optimum, in that case, the cross move is introduced as mutation operator, to improve the efficiency, 2-opt move is also adopted in the mutation procedure.

To maintain the diversity of the population, this operation has the ability to accept a worse solution with probability $P_a$. The pseudocode is given in Algorithm 2.

**Algorithm 1** Explosion Procedure

Set Parameters $N$, $\hat{A}$, $M_e$
Calculate the amplitude $A_i$ and number of sparks $S_i$
according to equation (7)
**for** $i$ = 1 to $S_i$ **do**
    *rand* = *Rand* ($A_i$);
    **for** $j$ = 1 to *rand* **do**
        Randomly remove a node from the current solution
        Greedy insert a new node at the best place
        Iteratively choose a daily tour to operate the 2-opt
        move
    **end for**
**end for**
Store all the explosion sparks into candidate set K

c) Selection strategy

It plays an important role in FWA. Similar to the conventional fireworks algorithm, the best solution is kept to the next generation all the time, for the other solutions, we adopt a strategy similar to roulette strategy. The selected probability of each solution is:

$$P_{sel}(X_i) = \frac{(f(X_i) - y_{\min})^2}{\sum_{X_j \in K} ((\mathrm{f}(\mathrm{X}_j) - y_{\min}) + \varepsilon)^2} \tag{8}$$

According to equation (8), solution with higher fitness is more likely to be preserved into the next generation. Comparing with the random selection strategy mentioned in the Enhanced Fireworks Algorithm [15], the preliminary results demonstrate that the performance of random selection strategy is not very satisfactory.

**Algorithm 2** Mutation Procedure

```
Set Parameters N, M_g, P_a
for i = 1 to M_g do
    Randomly select two nodes from two tours,
    operate cross move
    L = old profit, L' = new profit;
    if (L < L')
        Implement operation;
    else
        rand = Random (1);
        if (rand < P_a)
            Implement operation
        endif
    endif
    Iteratively choose a daily tour to operate the 2-opt move
end for
Store all the mutation sparks into candidate set K
```

## IV. Algorithms and Experiments

In this section, experiments are performed to illustrate the validity of FWA, after that, numerical analysis are carried out to compare FWA with other algorithms applied in the literature. The algorithms were compiled in Visual Studio 2015, and the experimental results were obtained on a PC with an Intel core i5-3470 3.2GHz processor and 4.0GB of memory.

### A. The problem instances

Four instances are tested in this paper, which is shown in table I. Coordinates of each POI is obtained from Baidu Map, together with the transferring time between POIs. To better illustrate the performance of the algorithms, the starting point and the end point is both chosen as the very center of the City wall in Xi'an.

TABLE I. The User-related Information

| Instances | User Information | | |
|---|---|---|---|
| | *user preference vector* | *Compactness(δ)* | *days* |
| Instance 1 | [1,1,2,3,5,2,3,1] | 0.25 | 4 |
| Instance 2 | [2,1,2,3,4,5,3,3] | 0.5 | 3 |
| Instance 3 | [3,2,2,1,1,3,5,2] | 0.5 | 3 |
| Instance 4 | [4,5,3,1,2,2,1,4] | 0.75 | 2 |

As mentioned in section II, rates of each POI is a weighted sum of preference score, popularity score and history score. The rates are the basis of trip planning process, the top 20 POIs for Instance 1 are listed in Table II as an example.

TABLE II. Rates of the Top 20 for Instance 1

| POI index | User Information Exploitation | |
|---|---|---|
| | *Name* | *Score* |
| 1 | Hua Mountain | 35.2 |
| 2 | the Big Goose Pagoda | 33.2 |
| 3 | Expo Park of Xi'an | 31.0 |
| 4 | Taiping National Forest Park | 31.0 |
| 5 | Muslim Snack Street | 30.4 |
| 6 | Qujiang Pool site garden | 30.0 |
| 7 | Terra-Cotta Warriors | 29.8 |
| 8 | Tang Paradise | 29.6 |
| 9 | Lishan Mountain | 29.0 |
| 10 | Bell tower of Xi'an | 28.8 |
| 11 | Xiaoyan Pagoda | 28.6 |
| 12 | City wall of Xi'an | 28 |
| 13 | Town of Ci'en | 27.8 |
| 14 | Xingqing Palace Park | 26.8 |
| 15 | Tang City Wall Relic Park | 26.2 |
| 16 | Xi'an Great Mosque | 26.0 |
| 17 | Shaanxi Provincial History Museum | 25.6 |
| 18 | Huaqing Palace | 24.8 |
| 19 | Academy door of Xi'an | 24.8 |
| 20 | Qinglong Temple | 24.0 |

### B. Performance analysis

FWA is applied to four instances mentioned above. An experiment is also carried out to compare with the performance of Iterated Local Search (ILS), Greedy Randomized Adaptive Search Procedure (GRASP), which is also has been applied in TTDP. Since the time efficiency is quite important for the on-line recommender system, to compare the performance of these algorithms, the runtime is set to a constant value, and each algorithm is run 20 times for different instances respectively.

Two runtime is tested in this paper: 10 seconds and 20 seconds. The best and average objective values of the solutions are presented in table III and table IV respectively.

We can conclude the following points from the numerical results. Firstly, the results generated by FWA have found six optimal results out of the eight conditions, same to ILS and better than GRASP. Secondly, FWA outperforms both ILS and GRASP with a limited running time, with better efficiency in finding global optimal solution.

TABLE III. Test Results of the Instances in 10 Seconds

| Instance | ILS | | GRASP | | FWA | |
|---|---|---|---|---|---|---|
| | *best* | *mean* | *best* | *mean* | *best* | *mean* |
| Instance 1 | 365.75 | 363.37 | 365.65 | **364.66** | **365.92** | 363.49 |
| Instance 2 | **441.00** | 438.6 | 427.00 | 425.90 | **441.00** | **439.65** |
| Instance 3 | 390.00 | 386.00 | 381.00 | 376.15 | **391.00** | **388.25** |
| Instance 4 | **370.25** | **367.64** | 365.21 | 352.96 | **370.25** | 367.23 |

TABLE IV. Test Results of the Instances in 20 Seconds

| Instance | ILS | | GRASP | | FWA | |
|---|---|---|---|---|---|---|
| | *best* | *mean* | *best* | *mean* | *best* | *mean* |
| Instance 1 | **366.11** | 364.28 | 365.71 | **365.34** | 365.94 | 363.87 |
| Instance 2 | **444.00** | **440.1** | 427.00 | 426.15 | **444.00** | 439.75 |
| Instance 3 | **392.00** | **388.85** | 384.00 | 377.95 | 391.00 | 388.45 |
| Instance 4 | **370.25** | **368.90** | 365.21 | 356.71 | **370.25** | 368.37 |

## Conclusion and Demo

### A. Conclusion

In this paper, a new TTDP model is proposed, taking into account the compactness of the designed trip. To apply FWA in TTDP, operators of the conventional FWA were redesigned based on the distance definition. Compared with ILS and GRASP, FWA can provide the most competitive results with outperformed efficiency. In the future work, to testify the efficiency of the proposed discrete FWA, comparisons will be implemented with more evolutionary algorithm, such as GA, ACO, etc.

### B. System Demo

The system is designed as a JSP webpage, composed of information collecting module, trip design module, and the routes displaying module. We take the instance 2 as example, the main process is as the demos below.


## Acknowledgment

This work was supported by National Natural Science Foundation of China (No. 61573277), and Natural Science Basic Research Plan in Shaanxi Province of China (Program No. 2015JM6316) .


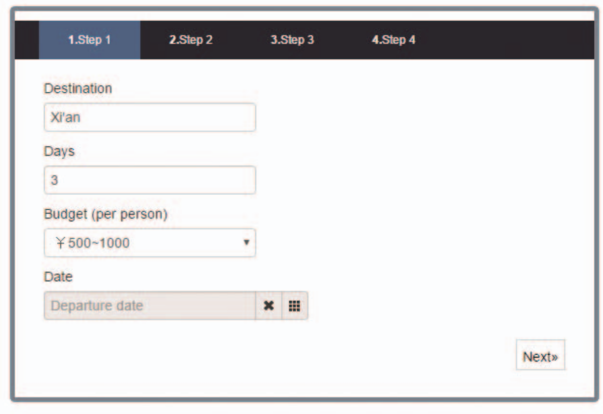


Fig. 3. Preference vector collection


Fig. 4. Routes display interface

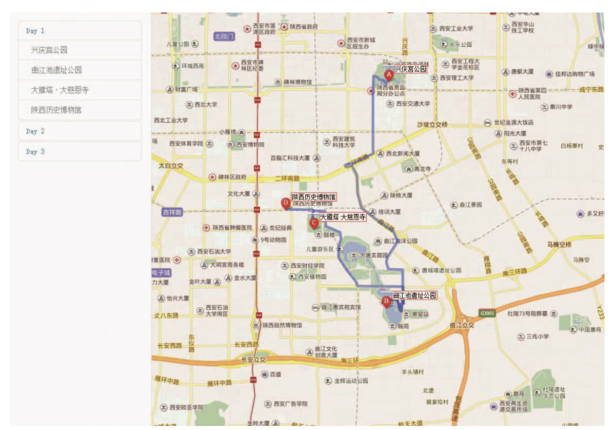

Fig. 5. Time and budget collection

## References


[1] P. Vansteenwegen, and D. Van Oudheusden, "The mobile tourist guide: an OR opportunity," OR Insight, 2014, 20(3), 21-27.

[2] D. Gavalas, C. Konstantopoulos, K. Mastakas, and G. Pantziou, "A survey on algorithmic approaches for solving tourist trip design problems," Journal of Heuristics, 2014, 20(3), 291-328.

[3] M. Fischetti, J. J. Salazar-González, and P. Toth, "The generalized traveling salesman and orienteering problems," In The traveling salesman problem and its variations, Springer US, 2007, pp. 609-662.

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

[5] W. Souffiau, J. Maervoet, P. Vansteenwegen, G. V. Berghe, and D. Van Oudheusden, "A mobile tourist decision support system for small footprint devices," In Bio-Inspired Systems: Computational and Ambient Intelligence, Springer Berlin Heidelberg, 2009, pp. 1248-1255.

[6] P. Vansteenwegen, W. Souffriau, G. V. Berghe, and D. Van Oudheusden, "The city trip planner: an expert system for tourists," Expert Systems with Applications, 2011, 38(6), pp. 6540-6546.

[7] T. A. Feo, and M. G. Resende, "A probabilistic heuristic for a computationally difficult set covering problem," Operations research letters, 1989, 8(2), pp. 67-71.

[8] D. Gavalas, C. Konstantopoulos, K. Mastakas, G. Pantziou, and Y. Tasoulas, "Cluster-based heuristics for the team orienteering problem with time windows," In Experimental Algorithms, Springer Berlin Heidelberg, 2013, pp. 390-401.

[9] D. Gavalas, C. Konstantopoulos, K. Mastakas, G. Pantziou, and N. Vathis, "Heuristics for the time dependent team orienteering problem: Application to tourist route planning," Computers & Operations Research, 2015, 62, pp. 36-50.

[10] Y. Tan and Y. Zhu, "Fireworks algorithm for optimization," in Advances in Swarm Intelligence, vol. 6145. Springer Berlin Heidelberg, 2010, pp. 355–364.

[11] A. Janecek and Ying Tan, "Swarm Intelligence for Non-Negative Matrix Factorization," International Journal of Swarm Intelligence Research, vol. 2, pp. 12–34, Oct.-Dec. 2011.

[12] S. Zheng and Y. Tan, "A unified distance measure scheme for orientation coding in identification," in 2013 International Conference on Information Science and Technology (ICIST), March 2013, pp. 979–985.

[13] W. He, G. Mi, and Y. Tan, "Parameter optimization of local-concentration model for spam detection by using fireworks algorithm,"in Advances in Swarm Intelligence, vol. 7928. Springer Berlin Heidelberg, 2013, pp. 439–450.

[14] Y. Tan, "Fireworks Algorithm (FWA)," In Fireworks Algorithm, Springer Berlin Heidelberg, 2015, pp. 17-35.

[15] S. Zheng, A. Janecek, and Y. Tan, "Enhanced fireworks algorithm," in Evolutionary Computation (CEC), 2013 IEEE Congress on, June 2013, pp. 2069–2077.