applied sciences

MDPI

Article

Modelling and Optimization of Personalized Scenic Tourism Routes Based on Urgency

Xiangrong Xu [1], Lei Wang [1,2,*], Shuo Zhang [1], Wei Li [1] and Qiaoyong Jiang [1]

[1] The Key Laboratory of Network Computing and Security Technology of Shaanxi Province, Xi'an University of Technology, Xi'an 710048, China

[2] The Key Laboratory of Industrial Automation of Shaanxi Province, Shaanxi University of Technology, Hanzhong 723001, China

* Correspondence: leiwang@xaut.edu.cn

Abstract: Traditional route planning methods usually plan the "fastest" or "lowest cost" travel route for users with the goal of finding the shortest path or the lowest cost, but this method cannot meet the needs of tourism users for personalized and multifunctional travel routes. Given this phenomenon, this paper proposes a personalized route planning model based on urgency. First, the model uses the visitor's historical tourism data and public road network data to extract their preferences, POI (point of interest) relationships, edge scenic values and other information. Then, the planned route function is determined according to the urgency value, which provides users with travel routes that accommodate their interest preferences and urgency. Finally, the improved genetic algorithm based on gene replacement and gene splicing operators is used to carry out numerical experiments on the Xi'an and Wuhan road network datasets. The experimental results show that the proposed algorithm is not only capable of planning routes with different functions for diverse users but also performs personalized route planning according to their preferences.

Keywords: user preference; scenic route planning; POI relationship modelling; urgency; improved genetic algorithm

check for updates

Citation: Xu, X.; Wang, L.; Zhang, S.; Li, W.; Jiang, Q. Modelling and Optimization of Personalized Scenic Tourism Routes Based on Urgency. *Appl. Sci.* **2023**, *13*, 2030. https://doi.org/10.3390/app13042030

Academic Editor: Richard C. Millham

Received: 5 January 2023
Revised: 30 January 2023
Accepted: 2 February 2023
Published: 4 February 2023

1. Introduction

1.1. Background

Today, many enterprises (such as Google, Baidu, etc.) have launched map services (Google Maps, Baidu Maps) based on road networks and public transport networks to plan feasible routes between two points based on these services with the main aim to find the shortest path or the lowest cost to achieve the fastest or most provincial route [1]. However, in reality, tourism users not only pay attention to path length and travel cost but also usually consider the utility value of the path [2]. For example, they may want to take a quiet route to free themselves and enjoy their personal time, or some users may want to take a route with good quality scenery so that travel is not boring or burdensome. They can even stop and go, visit multiple favourite attractions along the way, and improve the overall travel experience. Therefore, the fastest or most economical route does not meet the diverse needs of tourism users [3].

At present, there are many studies on scenic tourism route planning. Most researchers describe the problem as an OP (orientation problem) or AOP (arc orientation problem). The OP [4] is a combination of selecting nodes and determining the shortest path between the selected nodes. This can be seen as a combination of two classical problems: the knapsack problem and the travelling salesman problem. The difference between AOP and OP is that the weight of AOP is assigned to an arc, and the search object in the path search process is an arc rather than a point. AOP and OP are NP (Non-deterministic Polynomial) hard problems [5]. In addition, due to the large scale of the road network, finding the exact solutions to these two problems is time-consuming. Therefore, in practical applications, in order to meet the needs of a rapid response, heuristic search methods are often used to find their approximate solutions [6–8].

1.2. Related Works

In recent years, in relation to the two basic problems of OP and AOP, researchers have conducted extensive and in-depth research on scenic route planning alone and by combining user preferences.

Scenic route planning aims to plan routes with a large scenic score for users without considering their preferences. Chen X et al. [1] believed that the more photos and check-in times that are distributed on the road section, the higher its scenic score. Hence, they used Open Street Map (OSM) to extract basic road network data, check-in data on Foursquare, and Flickr photo data, to calculate the scenic value of the road section and propose a scenic route planning system based on multisource heterogeneous crowdsourcing data to recommend the best travel route between two points for users. Chen et al. [9] proposed a two-stage scenic route planning framework. This framework first calculates the scenic score of the road section and then uses a memetic algorithm (MA) to search and plan travel routes with high scenic scores for users. Zheng et al. [10] proposed an enhanced GPS navigation system, GPSView, which takes landscape factors into account in the route planning process to plan a driving route with landscape and sightseeing properties for users so that travellers can sightsee while driving. Skoumas et al. [11] extracted the spatial relationship between POIs (points of interest) from tourism blogs, quantized the relationship with a probability model, and established a POI relationship diagram. Then, Bayesian inference was used to calculate the probability measure of spatial intimacy. Finally, the method was applied to the road network to plan a more attractive route for users to guide them through their favourite areas. Li et al. [12] proposed a new genetic algorithm based on a path network. Each chromosome represents a feasible path which avoids the search cycle. The mutation at a specific location can quickly plan the path. Demiryurek et al. [13] maintain that with real-world spatial networks the edge's travel times are time dependent, where the arrival time to an edge determines the actual travel time on the edge, based on which they proposed a time-dependent A* algorithm to accelerate the calculation speed of the online path. Chen et al. [14–16] assume that the scenic score and travel time cost of each arc in the road network are time dependent; that is, the scenic score of different POIs in various time periods changes, and thus the travel time cost associated with assorted departure times also differs. They defined this problem as a twofold time-dependent AOP and proposed an MA to solve this problem. Similarly [9], Chen et al. [17] also believe that the utility value and travel time cost on the arc in the road network are time dependent and constructed a twofold time-dependent path planner to solve the problem. The difference is that they modelled the utility value of the path so that it can be quantified according to specific needs. For example, for security patrol personnel, the utility value could be the danger degree of the path; for those taking exercise, the value could be the path's quietness; and for tourists, it could be its scenic value. This unified modelling method expands the application scope of the method. Verbeeck et al. [18] proposed a fast local search metaheuristic method based on the ant colony algorithm, which combines the principle of this system with the time-dependent local search method to quickly provide an effective solution. Experiments show that the algorithm can obtain high-quality route planning results with low computational time, ensure that the route can be updated swiftly when new available traffic information appears, and help tourists reach their destination quickly. In subsequent work, Lu et al. [19] proposed an efficient MA to model more specific details in the tourism route planning problem to improve the accuracy of recommendations for specific problems.

The scenic route planning method combined with user preferences has been used to plan scenic routes that meet user preferences. Quercia et al. [20] provided street view data of London to passersby, asking them to vote on which street view they considered more beautiful, happier, or quieter, and then quantitatively analysed the results to recommend more of these types of paths to users. After verification and analysis, they found that, compared with the shortest route, the recommended route did not increase the distance cost but the route attractiveness far exceeded that of the shortest route. On the basis that tourists' preferences (such as expected starting and ending POI, must-see POI subset, etc.) should be taken into account in route planning. Taylor et al. [21] defined the problem as a TourMustSee problem and proposed an LP + M algorithm to solve the problem as one of integer linear programming. The experimental results showed that the route recommended by LP + M is better in terms of POI popularity, total POI visits, total travel time spent, and POI(s). Liang et al. [22] considered a top-k route search problem, that is, the set of given interest points and the travel cost between each interest point in the set, to find k paths that meet the constraints and contain as many POIs as possible in the interest point set. Therefore, they used a submodule function to model personalized demand, trim the search space with user preferences and constraints, and obtain the optimal solution for the top-k path search problem. Jiang et al. [23] asked users to manually input their own preference information, set corresponding weights, and created a path search algorithm based on A* and an effective pruning strategy for users to plan travel routes more in line with their preferences in combination with starting and ending information and users' maximum cost constraints. Zhang et al. [24] proposed a new route planning method that comprehensively considered multiple factors (distance between impromptu interest points, initial travel location, initial departure time, travel duration, total cost, score of interest points, and popularity) and rated the route based on the comprehensive attractiveness index to plan a route with a high comprehensive attractiveness index for users. Huang et al. [25] proposed a multitask in-depth travel route planning framework, integrating rich auxiliary information (including POI characteristics, user preferences and historical tourism routes), and realizing more effective route planning methods. The framework realizes three kinds of route planning tasks (next point recommendation, general route planning, and must visit planning), which can simultaneously meet the diverse needs of users.

1.3. Motivations and Contributions

The scenic route planning method calculates the scenic value of arc or POI in the road network through relevant crowdsourced data and maximizes the scenic score in the process of path planning. Such methods can often plan travel routes with large scenic scores for users. Route planning methods based on user preferences consider the visitor's personalized needs, but most of them require users to manually input their preferences and then connect as many POIs as possible to form the optimal path. The above two types of route planning methods aim to plan the route with the maximum scenic score for users, but they do not consider the urgency of users' travel and explore users' interests and preferences. Therefore, this paper proposes a personalized scenic route planning model based on urgency. This model takes tourism users as the object to provide tourism users with a multifunctional and personalized scenic route planning scheme. The goal is to plan an optimal driving route that matches urgency and scenic features with users' interests and preferences. Therefore, the main contributions of this paper are summarized as follows:

1. A multifunctional and personalized time-dependent AOP is defined, which indicates different travel schemes at various departure times for several user preferences with specific travel urgency.
2. A personalized scenic tourism route planning model based on urgency is proposed to solve the above time-dependent AOP. Through the steps of user preference extraction, road network modelling and path optimization search, the model plans multifunctional and personalized scenic routes to meet the needs of users.
3. The test was carried out on the road network dataset of Xi'an and Wuhan. Compared with the three benchmark methods in the experiment, the proposed model can plan routes with various functions according to specific urgencies as well as plan routes with different neighbouring POIs for individual users. Therefore, the model has good performance in terms of effectiveness.

2. Problem Description and Modelling

For the convenience of description, the following definitions are given for the terms subsequently used and the research questions.

2.1. Basic Concepts

Definition 1 (Road network *G* [17]). *The road network is modelled as a directed graph* $G = (N, E, scenicScore, travelTime)$, *where N is a collection of nodes (intersections and dead ends),* $E \in N \times N$ *is a collection of edges, scenicScore is the scenic value of nodes and edges in the road network, and travelTime is the travel time of edges in the road network.*

Definition 2 (Path *R* [17]). *At time* t_0, *a path is formed by connecting multiple edges in the road network in turn. A path from the source point* n_0 *to the destination* n_k *is marked as* $R\langle\langle e_{0,1}, e_{1,2}, \ldots, e_{k-1,k}\rangle, t_0\rangle$, *where* n_0 *and* n_k *belong to the set N, e belongs to the set E,* $e_{0,1}$ *represents the edge of node* n_0 *to* n_1, *and* t_0 *is the departure time.*

Definition 3 (User preference *P*). *User preference is the preference vector for the POI feature type obtained by mining the historical behaviour data of users. The user's preference is expressed as* $P(u) = (x_1, x_2, \ldots, x_m)$, *where m is the embedding of the user preference into m-dimensional space. In this paper, the method described in [26] is adopted to obtain user preferences from users' historical check-in data, see Section 3.2 for details.*

Definition 4 (User query *Q* [17]). *The user's query is defined as a quad, expressed as* $Q\langle n_0, n_k, t_0, b\rangle$, *where* n_0 *is the starting point,* n_k *is the destination,* t_0 *is the departure time, and b is the time budget of user travel.*

Definition 5 (Effective area). *The area in the road network that the user may visit when starting from* n_0 *at time* t_0 *and arriving at destination* n_k *within travel time budget b.*

Definition 6 (Scenic edge). *In this paper, the edge with a scenic score greater than 0 is recorded as scenic edge.*

Definition 7 (Urgency). *The urgency is used to describe the urgency of the user's travel, that is, whether the user is anxious to reach the destination. At time* t_0, *the minimum travel time from starting point* n_0 *to destination* n_k *is determined. In this case, the smaller the travel time budget b, the more anxious the user is to reach the destination. Therefore, the urgency is defined as the ratio of the shortest travel time between* n_0 *and* n_k *at time* t_0 *to the time budget and is calculated as shown in Formula (1).*

$$urgency = \frac{\min travelTime(n_0, n_k, t_0)}{b} \tag{1}$$

where $mintravelTime(n_0, n_k, t_0)$ represents the shortest travel time from n_0 to n_k at time t_0.

Definition 8 (Quality ratio [6]). *The quality ratio is used to describe the cost performance of the path. That is, the higher the scenic score and the shorter the travel time is, the better the path is. Therefore, the quality ratio is defined as the ratio of the scenic path score to the travel time and is calculated as shown in Formula (2).*

$$qualityRatio(R) = \frac{scenicScore(R)}{travelTime(R)} \tag{2}$$

where $scenicScore(R)$ and $travelTime(R)$ are the scenic score and travel time cost of the path, respectively. The higher $qualityRatio(R)$ is, the better the path.

2.2. Problem Modelling

According to the above definition, the modelling of personalized route planning problem studied in this paper is as follows.

Definition 9 (Personalized scenic tourism route planning problem based on urgency Ω). *The problem is defined as $\Omega = (G, Q, P(u), urgency)$ for a given road network G, user query condition Q, user preference vector $P(u)$ and urgency, this paper can plan a personalized scenic tourism route R meeting the urgency for user u. Specifically, this paper plans the travel route R with different functions for users according to different values of urgency. That is, when users are anxious to reach a destination, it can plan the route with the shortest travel time; when the user is not anxious to reach the destination, it can plan the route with the largest scenic score for the user; if urgency lies between the two, the route with the best quality ratio will be planned for users. The simultaneous introduction of user preference information in the latter two path planning processes can make the scenic features of path R conform to user preferences as much as possible. Mathematically, the scenic tourism route planning problem based on urgency is a variant of the AOP, as shown below.*

$$f(R) = \begin{cases} \max\frac{1}{travelTime(R)} \\ \quad = \max\sum_{\mathrm{i}=0}^{k}\sum_{j=0}^{k}\frac{1}{travelTime(e_{i,j})} \times x_{i,j}, & urgency \geq 0.7 \\ \max qualityRatio(R) \\ \quad = \max\sum_{\mathrm{i}=0}^{k}\sum_{j=0}^{k}\frac{scenicScore(e_{i,j})}{travelTime(e_{i,j})} \times x_{i,j}, & 0.3 < urgency < 0.7 \\ \max scenicScore(R) \\ \quad = \max\sum_{\mathrm{i}=0}^{k}\sum_{j=0}^{k} scenicScore(e_{i,j}) \times x_{i,j}, & urgency \leq 0.3 \end{cases} \tag{3}$$

subject to

$$x_{i,j} \in \{0,1\} \tag{4}$$

$$\sum_{j=1}^{k} x_{0,j} = \sum_{i=0}^{k-1} x_{i,k} = 1 \tag{5}$$

$$\sum_{i=0}^{k} x_{i,0} = \sum_{j=0}^{k-1} x_{k,j} = 0 \tag{6}$$

$$\sum_{i=0}^{k} x_{i,j} = \sum_{j=0}^{k} x_{i,j} \leq 1 \tag{7}$$

$$\mathrm{dt}(n_0) = t_0 \tag{8}$$

$$travelTime(R) \leq b \tag{9}$$

$$dt(n_j) = \sum_{i=0}^{k} (dt(n_i) + travelTime(e_{i,j})) \times x_{i,j} \tag{10}$$

where $x_{i,j}$ represents whether the edge from node n_i to node n_j is included in the path, and $x_{i,j} = 1$ represents that the edge from node n_i to node n_j is included in the path; otherwise, the edges from node n_i to node n_j are not included in the path. $dt(n_j)$ is the departure time from node n_j, and $travelTime(e_{i,j})$ is the travel time on edge $e_{i,j}$, which is determined by the length of edge $e_{i,j}$ and the travel speed when starting from n_i.

Formula (3) is an objective function, which means minimizing the travel time of the path, maximizing the quality ratio of the path, or maximizing the scenic score of the path according to the difference values of urgency. Formula (5) ensures that the planned path starts from the starting point and ends at the end point. Formula (6) ensures that no edge enters the starting point and no edge leaves the end point. Formula (7) guarantees that any node connected with the edge on the path can pass through once at most. Formula (8) guarantees that the user starts at t_0. Formula (9) guarantees that the path travel time does not exceed the time budget specified by the user. Formula (10) calculates the departure time of node n_j.

3. Proposed Method

3.1. Model Overview

This paper proposes a personalized scenic tourism route planning model based on urgency. The model overview is shown in Figure 1.

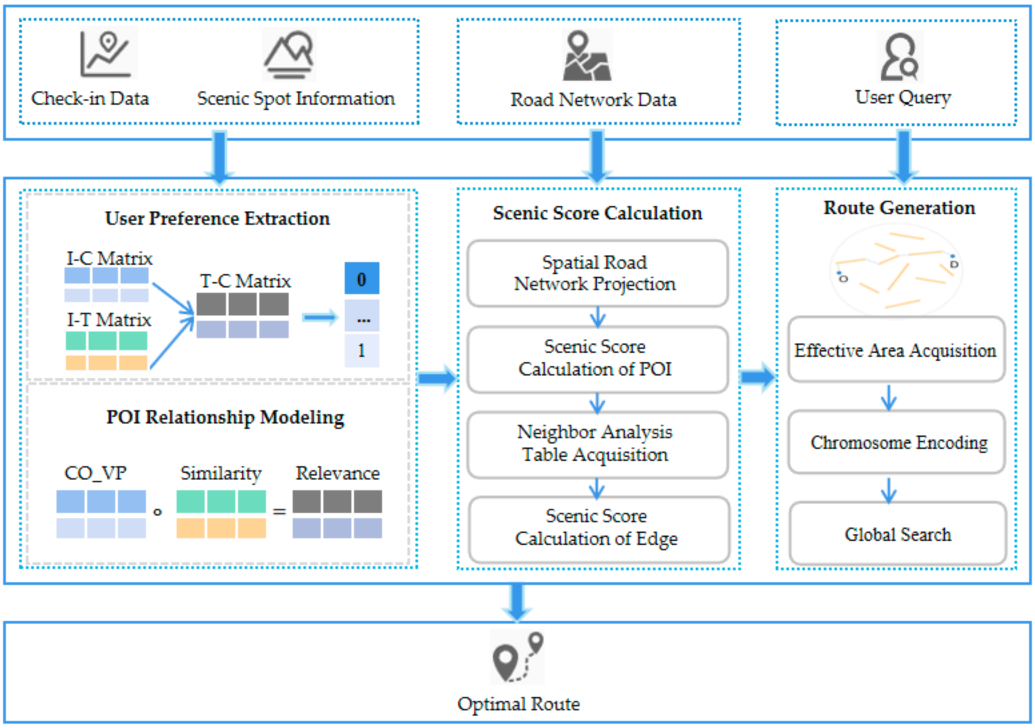


Figure 1. Overview of personalized scenic tourism route planning model.

As shown in Figure 1, this model takes tourist users as the object to plan an optimal travel route that meets the query conditions and whose scenic features are in line with user preferences. The main process of the model can be divided into three stages: user preferences extraction and POI relationship modelling, scenic score calculation, and route generation. The input data for user preference extraction and POI relationship modelling are historical check-in data and scenic spot information data, and the output is the correlation between the user preference vector and scenic spots. Scenic score calculation is the second stage. The input data are road network data, the relationship matrix of POIs, and preference vector of the user. The output is the effective edge scenic score. In the process of the calculation of the edge scenic score, it is necessary to perform the steps of the spatial road network projection, the scenic score calculation of POI, the neighbour analysis table acquisition, and the scenic score calculation of edge. Finally, the edge scenic score is calculated according to the POI scenic score of the edge neighbour. Considering that different users prefer different types of POI, the experience value obtained during sightseeing will also be different. Therefore, user preference information is combined with the scenic score calculation. Route generation is the last stage which uses the improved genetic algorithm to search the effective edge and obtain the best travel route from the starting point to the end point that meets the user's query conditions. This article will explain each stage separately in the following sections.

3.2. User Preference Extraction

In this paper, the method in reference [26] extracts user preferences based on POI-type information and user history check-in data. The details are as follows:

1. The user-POI sign-in matrix $I - C_{m \times n}$ is obtained from the user-POI historical sign-in data, where m is the number of users and n is the number of POIs. The values of each element in the matrix $I{-}C$ are shown in Formula (11):

$$Ic_{i,j} = \begin{cases} c_{i,j}, & \text{sign-in times of user } i \text{ to } POI_j \\ 0, & \text{user } i \text{ has not signed in to } POI_j \end{cases} \tag{11}$$

   where $Ic_{i,j}$ represents the number of sign-in times of user i to POI_j, and $i \in (1, 2, \cdots, m)$ and $j \in (1, 2, \cdots, n)$ are subscript variables.
2. The POI-type matrix $I - T_{n \times l}$ is obtained from the POI basic information data, where n is the POI number and l is the POI-type number. The expression of the values of each element in the matrix $I - T$ is shown in Formula (12):

$$It_{j,k} = \begin{cases} 1, & POI_j \text{ has type } T_k \\ 0, & POI_j \text{ does not have type } T_k \end{cases} \tag{12}$$

   where $It_{j,k}$ indicates whether POI_j has type feature T_k. $j \in (1, 2, \cdots, n)$ and $k \in (1, 2, \cdots, l)$ are subscript variables.
3. The user-type sign-in matrix $T - C_{m \times l}$ is obtained from the user-POI sign-in matrix $I - C_{m \times n}$ and the POI-type matrix $I - T_{n \times l}$, where m is the number of users and l is the number of POI types. The values of each element in the matrix $T - C$ are shown in Formula (13):

$$tc_{i,k} = \begin{cases} m_{i,k}, & \text{sign-in times of user } i \text{ for type } T_k \\ 0, & \text{user } i \text{ has not signed in type } T_k \end{cases} \tag{13}$$

   where $tc_{i,k}$ represents the number of sign-in times of user i for type feature T_k, that is, the number of times the user has been to POI with type feature T_k, and $i \in (1, 2, \cdots, m)$ and $k \in (1, 2, \cdots, l)$ are subscript variables.
4. After obtaining the $U{-}T$ matrix, take the value in the matrix as the parameter, use the linear regression model to obtain the type weight, and use the gradient descent to optimize the weight value. Finally, take the N types with the largest weight value as the user's preferred type, and code the user preference vector.

3.3. POI Relationship Modelling

This paper calculates the scenic score of the edge according to the POI scenic score of the adjacent edge. However, in the actual road network, there are often multiple POIs adjacent to the same edge. At the same time, due to the different characteristics of POIs, the scenic score will be lost when multiple POIs are combined. Therefore, to calculate the scenic score for the edge more accurately, the relationship between POIs is mined and modelled as follows:

$$r(i,j) = \sqrt{\text{Co\_}VP(i,j) \times Sim(i,j)} \tag{14}$$

where $r(i,j)$ represents the correlation between POI_i and POI_j, which is jointly determined by the user's common access probability $Co\_VP(i,j)$ to POIs and the feature similarity $Sim(i,j)$ between POIs. The larger $r(i,j)$ is, the closer the relationship between POI_i and POI_j, and the less the loss of scenic score when they are combined. The specific calculation of $Co\_VP(i,j)$ and $Sim(i,j)$ is shown in Formulas (15) and (16), respectively.

$$\text{Co\_}VP(i,j) = \alpha \frac{N_{i,j}}{N_i + N_j + N_{i,j}} + (1 - \alpha) \frac{\sum_{k=1}^{m} \min\left(uc_{k,i}, uc_{k,j}\right)}{\sum_{k=1}^{m} \left(uc_{k,i} + uc_{k,j}\right)} \tag{15}$$

where $Co\_VP(i,j)$ is the co-visit probability between POI_i and POI_j, $N_{i,j}$ is the number of users who have visited POI_i and POI_j at the same time, N_i is the number of users who have visited POI_i but not visited POI_j, $uc_{k,i}$ is the number of times user k has signed at

POI_i, m is the number of users, and α is the weighting coefficient, set at 0.5 here. The larger $Co\_VP(i,j)$, the more likely the user is to visit the two POIs at the same time, that is, the closer the relationship between the two POIs from the user's perspective.

$$Sim(i,j) = \cos\theta(T_i, T_j) = \frac{T_i \cdot T_j}{\|T_i\|\|T_j\|} \tag{16}$$

where $Sim(i,j)$ is the cosine similarity between the eigenvectors of POI_i and POI_j, T_i is the eigenvector of POI_i. The larger $Sim(i,j)$ is, the closer the relationship between POI_i and POI_j.

3.4. Scenic Score Calculation

The scenic score is used to measure the quality of the landscape of the POI or the edge. The larger the scenic score is, the higher the quality of the landscape of the POI or the edge, and the more popular the users. In this paper, the scenic value of POI and the scenic value of edge are calculated, respectively. The specific process is as follows.

1. The scenic score of POI. This is determined mainly by the corresponding score, level, pictures, and comments of POI. The score, pictures, and comments are contributed by the checking in of users, and the level is determined by the characteristics of the POI itself. The greater these values are, the greater the scenic score of the POI. The specific calculation is shown in Formulas (17) and (18).

$$s(i) = \log(\sqrt[3]{score(i) \times pictures(i) \times comments(i)} + level(i)) \tag{17}$$

$$scenicScore(i) = s(i) \times [1 + \max(s(j)) \times w_i)] \tag{18}$$

where $s(i)$ is the inherent scenic score of POI_i and $scenicScore(i)$ is the scenic score felt by different users. Here, combined with user preferences, *score*(*i*), *level*(*i*), *picture*(*i*) and *comments*(*i*) represent the data after POI_i's score, grade, number of photos, and positive comments, respectively, are normalized to [0, 5]. The higher the score of the POI, the more photos, the more positive comments, and the higher the star rating, the higher the scenic score corresponding to the POI. $max(s(j))$ is the highest scenic score of the POI. w_i is the cosine similarity between the feature vector T_i of POI_i and the user preference vector $P(u)$, which is called the reward factor here. The scenic score of a POI is inherent. However, from the user's perspective, those who like this type of POI actually have a much better viewing experience than those who do not. That is, POI will reward users. The more users like this type of POI, the greater the reward factor and, hence, the greater the reward value. When w_i is 0, the user does not like this POI or does not have a historical data linked to it. Then, the POI may be recommended to the user according to its inherent scenic score.

2. The scenic score of the edge. The scenic score of an edge is determined by the scenic score of the POI of its neighbours. The more neighbouring POIs there are, the greater the POI correlation, and the higher the POI scenic score, the higher the scenic score of the corresponding edge. The specific calculation is shown in Formula (19).

$$\text{s}cenicScore(e_{i,i+1}) = scenicScore(1) + \sum_{j=2}^{m} scenicScore(j) \times r(j-1,j) \tag{19}$$

where m is the number of POI adjacent to edge $e_{i,i+1}$, and $r(j-1,j)$ is the correlation between POI_{j-1} and POI_j.

3.5. Route Generation

3.5.1. Effective Area Acquisition

For a given query condition, when planning the path from the start point to the end point for users, the possible effective search area is only a small part of the entire road network, and most of the areas cannot appear in the road. However, if we do not process the road network and let the algorithm search the entire road network, the efficiency will be extremely low. In this case, it is necessary to obtain a suitable effective search area from the road network. Therefore, this paper adopts the method in [15] to cut the search area. Specifically, we take the starting point and the ending point as the centre of the circle and r as the radius to draw a circle. The intersecting part of the two circles is the effective area. r refers to the distance travelled at the average driving speed $\overline{v}(t_0)$ for b time starting at time t_0, i.e., $r = \overline{v}(t_0) \times b$. The edges within the valid region are valid edges. Here, the effective edge can be defined as: at time t_0, starting from the starting point, the end point can be reached within the time budget, and starting from the end point, the starting point can also be reached within the time budget.

3.5.2. Chromosome Encoding

Since this paper plans corresponding travel routes for users according to the value of urgency, different chromosome coding methods should be used for different values of urgency. Several effective edge selection strategies for generating different chromosomes are described below.

Strategy 1: The closest distance priority strategy. Priority is given to selecting the effective edge closest to the current starting point in space so that more edges can be added to the path.

Strategy 2: The shortest travel time priority strategy. Under time t_0, the effective edge with the shortest travel time is preferred so that more time budget can be reserved.

Strategy 3: The highest quality ratio priority strategy. Give priority to the effective edge with the largest quality ratio to ensure higher cost performance.

Strategy 4: The maximum scenic score priority strategy. Priority shall be given to the effective edge with the maximum scenic score to ensure a higher scenic score.

Strategy 5: Random priority strategy. An edge is randomly selected from the effective area to avoid the problem that the above strategies may search for effective edges in small areas of the road network.

Based on the above effective edge selection strategy, Algorithm 1 gives the pseudo code of chromosome encoding.

In Algorithm 1, first, initialize an empty chromosome set, and then execute lines 3–24 to return the chromosome set under the condition that user constraints are met. Line 3 obtains the valid search area according to the method in Section 3.5.1; Lines 4–14 acquire candidate edge sets, that is, using different effective edge selection strategies according to different urgencies to generate candidate edge sets. Specifically, when $urgency > 0.7$, the edge that is closer to the current starting point and shorter travel time is selected from the valid region each time to join the candidate edge set. When $urgency < 0.3$, the candidate edge is selected and added to the candidate edge set according to strategies 1, 2, 4, and 5. When $0.3 \leq urgency \leq 0.7$, the candidate edge is selected and added to the candidate edge set according to strategies 1, 2, 3, and 5. Lines 15 to 18 update the departure time, remaining time budget, and starting point for each edge in the candidate edge set. Lines 19–24 recursively call the chromosome encoding function encoding () to generate all the sub-chromosomes and carry out subsequent chromosome growth.

Algorithm 1: ***encoding*(n_0, n_k, *b*, *urgency*)**

Input: starting point n_0, ending point n_k, time budget b, $urgency$
Output: set of chromosomes $chrs$

```
Function $encoding(n_0,\ n_k,\ b,\ urgency)$
    $chrs \leftarrow \varnothing$
    obtaining the valid region
    $genes \leftarrow \varnothing$    //candidate edge set
    if$(urgency > 0.7)$    //plan the fastest route
        select the edges $en$ and $ef$ according to strategy 1 and 2 respectively
        $genes \leftarrow en \cup ef$
    else if$(urgency < 0.3)$    //plan the route with the maximum scenic value
        select edges $en, ef, em$ and $er$ according to strategies 1, 2, 4 and 5 respectively
        $genes \leftarrow en \cup ef \cup em \cup er$
    else if$(0.3 \leq urgency \leq 0.7)$    //plan the route with the maximum $qualityRatio$
        select edges $en, ef, eq$ and $er$ according to strategies 1, 2, 3 and 5 respectively
        $genes \leftarrow en \cup ef \cup eq \cup er$
    end if
    for$(gene$ in $genes)$
        $t_0 = t_0 + \frac{dist(n_0,gene)}{v(t_0)}$    //update departure time
        $b = b - \frac{dist(n_0,gene)}{v(t_0)}$    //update the remaining time budget
        update $n_0$    //update starting point
        $subbranchs = encoding(n_0,\ n_k,\ b,\ urgency)$
        for$(subbranch$ in $subbranchs)$
            $chrs \leftarrow chrs \cup (gene + subbranch)$
        end for
        $chrs \leftarrow chrs \cup gene$
    end for
    return $chrs$
```

3.5.3. Improved Genetic Algorithm

The genetic algorithm (GA) was designed according to the evolution law of organisms in nature. It is a computational model of the biological evolution process that simulates the natural selection and genetic mechanism of Darwinian biological evolution. It thus searches for the optimal solution by simulating the natural evolution process [27–30].

In this study, an improved GA was used to search effective edges, and travel routes satisfying constraint conditions and scenic features conforming to user preferences were planned according to different urgencies. The algorithm flow chart is shown in Figure 2. In the figure, the gene splicing operator and the gene replacement operator are two new operators compared with the traditional genetic operator, and the gene splicing operator is used to solve the problem of discontinuity of actual paths corresponding to chromosomes. The gene replacement operator is an improved part of the traditional GA according to the problem requirements. The purpose of adding this operator is to make a local optimization adjustment of all the current chromosomes before each iteration generates a new population to improve the performance of the final search results. b is the travel time budget, and C is the maximum number of replacements.

1. Selection. This paper uses roulette to select offspring, that is, the probability of each chromosome being selected is proportional to its fitness value. This method may not ensure the best individual inherits the next generation, but it can avoid the impact of super chromosomes on the overall evolution. At the same time, it is possible to pass on the worst chromosome to the next generation.
2. Crossover. In this paper, the method of single point crossing is used to cross chromosomes, that is, randomly select two chromosomes and determine the same edge of the path to cross that can ensure the continuity of the path. If two chromosomes have multiple identical edges, one of the same edges is randomly selected for crossing.

If two chromosomes do not have the same edge of the path, then randomly select one edge to cross, and the continuity of the path is guaranteed by the gene splicing operator. The schematic diagram of the crossover operation is shown in Figure 3a, in which the corresponding paths of the two chromosomes intersect at the edge *f*, so the chromosomes are crossed at a single point at *f* to form two new chromosomes.

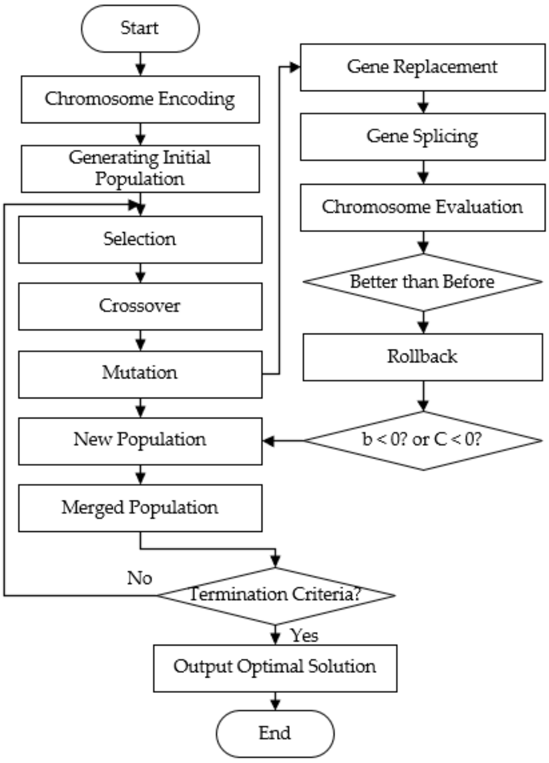


Figure 2. Flow chart of improved GA.

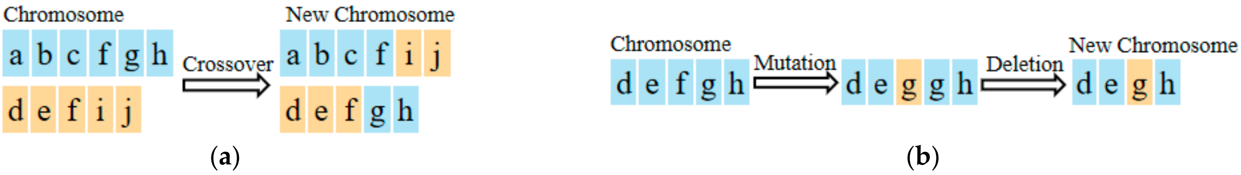


Figure 3. Schematic diagram of crossover and mutation operation. (**a**) Crossover; (**b**) Mutation.

3. Mutation. The mutation operation randomly selects an edge in a chromosome, randomly selects another effective edge in the effective region, and then replaces the selected edge on the chromosome with the effective edge. The variation here is adjusted with only a small probability. After the above crossover and mutation operators have been executed, the problem of gene duplication in chromosomes may arise. In practice, the user's route from the starting point to the end point should be unidirectional, and the user is not allowed to walk back and forth on a certain section of road repeatedly. This situation will also increase the cost of useless paths. Therefore, it is necessary to check the chromosomes and delete repeated edges. The schematic diagram of the mutation operation is shown in Figure 3b, in which the gene *f* of the chromosome mutates into gene *g*, resulting in the continuous occurrence of two gene *g* in the chromosome. At this time, a duplicate gene *g* is deleted through the deletion operation to form a new chromosome.
4. Gene Splicing. Gene splicing is mainly used to solve the problem of discontinuous paths corresponding to chromosomes. In the operation of chromosome coding and

cross mutation, the corresponding path of chromosome will be discontinuous, so this paper uses the shortest path to fill the gap between edges. The specific process is to map the chromosome to the actual path, fill the gap in the path by the shortest path, and then code the edge that fills the gap into a gene, which is spliced to the corresponding position of the chromosome. After filling the gap, the path corresponding to the chromosome is the continuous feasible path in the actual road network. The schematic diagram of the gene splicing operation is shown in Figure 4. The path corresponding to the chromosome in the figure has a gap between edge *c* and edge *e*, resulting in discontinuous paths. In this paper, the shortest path is used to find edge *d* (corresponding gene d), and gene *d* is spliced to the chromosome using the gene splicing operator to obtain a new chromosome. The corresponding path of this chromosome is a continuous and feasible path.

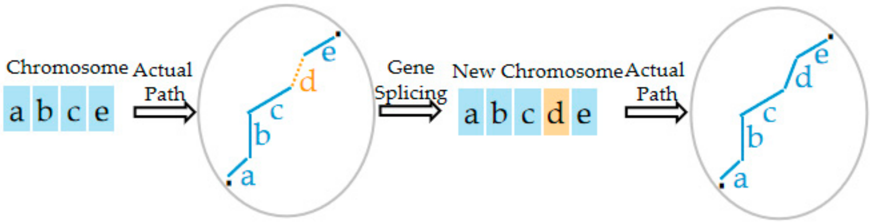


Figure 4. Schematic diagram of gene splicing operation.

5. Gene Replacement. The gene replacement operation is a specific mutation operation with close to high probability based on pattern theorem and random operation. The core is to replace the inferior gene of the current chromosome with the better gene not included in the current chromosome with a higher probability of Pr. Traditional GAs are prone to slow convergence and poor solutions when the amount of data in the search space is large. Therefore, this paper adds a gene replacement operator and, on the premise of not exceeding the travel time budget and the maximum replacement times, uses this operator many times to replace the inferior genes in the chromosome to improve the fitness value of the chromosome and accelerate algorithm convergence. After each execution of the gene replacement operator, the gene splicing operation needs to be performed again to ensure that the current chromosome mapping to the actual road network is continuous and feasible. Based on the above discussion, the pseudo code of the gene replacement operator is shown in Algorithm 2.

In Line 2, chromosomes before replacement are temporarily stored for rollback. Line 3 is to calculate $travelTime$, $qualityRatio$ and $scenicScore$ of genes in the current chromosome. Line 4 is to find the worst gene of $travelTime$, $qualityRatio$, and $scenicScore$ from the current chromosome, that is, to find the replaced gene. Lines 5–9 select K genes with $travelTime$ smaller than the replaced gene or $qualityRatio$ and $scenicScore$ larger than the replaced gene from the region close to the replaced gene, and add them to the candidate replacement gene set. Lines 10–11 randomly select a gene to be replaced from the candidate replacement gene set. Lines 12–13 were used for the gene splicing operation and new chromosome evaluation. Lines 14–18 indicate that when the new chromosome after replacement is better than that before, the replacement is successful; otherwise, the algorithm will return to the chromosome before replacement.

A schematic diagram of the gene replacement operation is shown in Figure 5. In the figure, the short blue line represents the gene added to the current chromosome, and the short yellow line represents the neighbouring gene of *c*.

Algorithm 2: ***geneReplacement*(*q*, *K*)**

```
Function geneReplacement(q, K)
    tempchromo ← currentchr   //temporarily save the current chromosome
    calculate travelTime, qualityRatio or scenicScore of genes
    select the worst gene in travelTime,qualityRatio or scenicScore   //look for the replaced gene in the current chromosome
    replacementgeneset ← ∅   //set of candidate replacement genes
    while (number of candidate replacement genes < K)   //search for candidate replacement genes
        select candidate replacement genes replacementgene
        replacementgeneset ← replacementgeneset ∪ replacement gene
    end while
    the replacement gene replacement gene was randomly selected from replacementgeneset
    genes are replaced to produce chromosomes newchromo
    gene splicing
    chromosome evaluation
    if (newchromo is better than tempchromo)
        currentchr ← newchromo
    else
        currentchr ← tempchromo
    end if
```

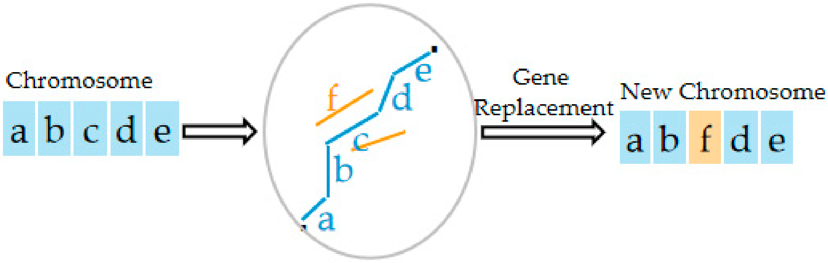


Figure 5. Schematic diagram of the gene replacement operation.

4. Experimental Results and Analysis

4.1. Experimental Setup

4.1.1. Data Preparation

The road network dataset used in this paper is the road network data of Xi'an and Wuhan downloaded from the OSM (https://www.openstreetmap.org/, (accessed on 10 September 2021)) platform (see Table 1 for details). The POI basic information data, user history check-in data and user history rating data were crawled from Ctrip (https://you.ctrip.com/, (accessed on 5 October 2021)) and Mafengwo (https://www.mafengwo.cn/ (accessed on 5 October 2021)). The basic POI information includes POI-type information, number of photos, number of favourable comments, rating, star rating, etc. Historical user check-in data and score data refer to the number of users' check-in times and POI scores. The crawled user data and POI data are shown in Table 2.

Table 1. Introduction to road network dataset.

City	Number of Nodes	Number of Edges	Number of POIs
Xi'an	25,431	33,010	4138
Wuhan	28,153	34,512	4350

Table 2. Introduction to other data.

City	Number of Users	Number of POIs	Number of POI Types
Xi'an	1161	1587	10
Wuhan	1583	2039	10

4.1.2. Benchmark Algorithm

In this paper, three benchmark algorithms are used for comparison, namely, the traditional GA, fastest algorithm, and MA. The details are as follows.

Fastest algorithm [13]. Ugur D et al. believed that the travel time was related to the departure time, so they introduced the time factor in modelling the road network and used the fastest path to plan the fastest travel route from the start point to the end point that met the time constraints.

MA [15]. Chao C et al. believed that the travel time and scenic score on each side of the road network are time dependent, so they defined the problem as a twofold time-dependent AOP and proposed an MA to solve the problem.

4.1.3. Parameter Setting

After many experiments and comparisons, the experimental parameters are set as follows: population size $N = 20$, iteration times $T = 30$, crossover probability $Pc = 0.9$, mutation probability $Pm = 0.15$, gene replacement probability $Pr = 0.95$, and candidate set size of replacement genes $K = 4$.

4.2. Experimental Result

4.2.1. Algorithm Gene Replacement Times and Convergence Analysis

To find the best times of gene replacement, this paper carried out several experiments on the road network datasets of Xi'an and Wuhan,. The results are shown in Figure 6. In the figure, when the number of gene substitutions is equal to 8, the chromosomal scenic score reaches the maximum and will not change with an increase in the number of gene substitutions. The reason for this phenomenon is that the selection conditions of replacement genes are relatively strict, and the chromosomes are not very long after the search area has been limited. Therefore, when replacement times reach a certain value, it is difficult to find a replacement edge that meets the conditions. Therefore, this paper considers that 8 is the best number of gene replacements. When the number of gene replacements is greater than 8, gene replacement will not be performed.

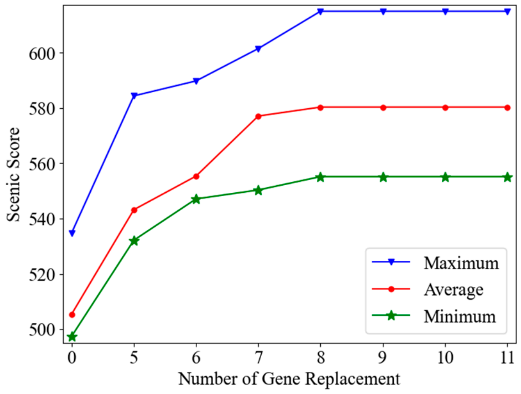


Figure 6. Number of gene replacements.

In addition, the convergence of several algorithms is experimentally analysed. The results are shown in Figure 7. The figure shows that the fastest algorithm converges first. When the number of iterations reaches 10, the improved GA converges. The MA and traditional GA reach the maximum scenic score after 15 iterations, and no change occurs in subsequent iterations.

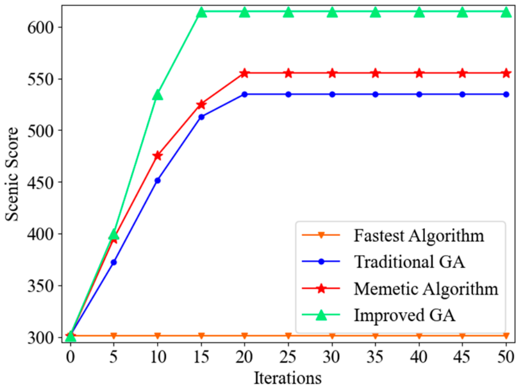


Figure 7. Convergence curves of the four algorithms.

4.2.2. Sensitivity Analysis of Algorithm to Starting and Ending Points

To prove that the proposed algorithm has the ability to plan routes, three pairs of starting and ending pairs (OD pairs) were selected on the road network datasets of Xi'an and Wuhan, and experiments conducted on three benchmark algorithms and improved genetic algorithms ($urgency = 0.2$, $t_0 = 9:00$). The average results are shown in Figure 8 (scenic score) and Figure 9 (running time). In Figure 8, for different starting and ending points, the four algorithms can plan routes with different scenic scores. The path scenic scores of the fastest algorithm and the improved GA are always the minimum and maximum. The path scenic score of the traditional GA is much lower than that of the improved GA, which shows that the improved genetic algorithm based on the gene replacement operator and the gene splicing operator has a good performance for this problem. The MA has a lower scenic score than the improved GA because when the MA encodes the chromosome, it randomly selects the edge close to the current origin, resulting in poor quality of the chromosome.

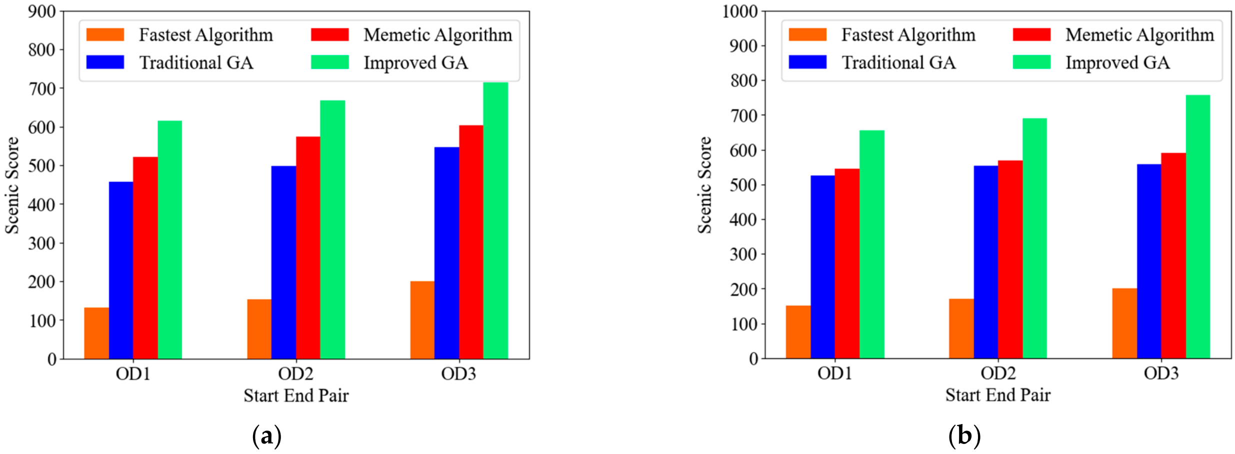
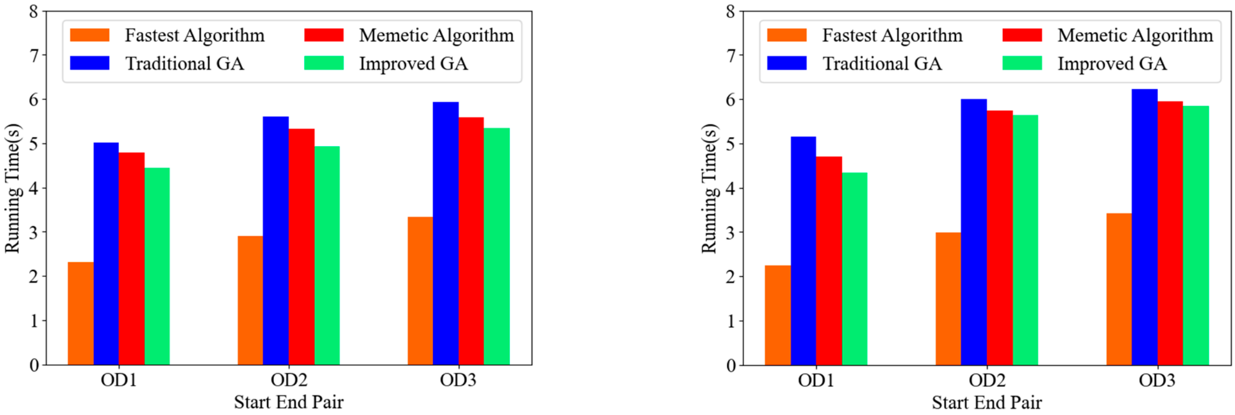


Figure 8. Comparison of scenic score under different start end pairs. (**a**) Xi'an; (**b**) Wuhan.

Figure 9 shows the running time of several algorithms for planning different starting and ending routes. The fastest algorithm has the shortest running time, while the traditional GA has the longest. The running time of the improved GA is shorter than that of the MA and the traditional GA. The reason for this is that the improved GA adds a gene replacement operator. The implementation of this operator speeds up algorithm convergence to a certain extent. At the same time, the chromosome encoding method of the improved GA will not produce too many chromosomes. Overall, on the road network datasets of Xi'an and Wuhan, the algorithms show identical performance, but the scenic score on the road network datasets of Wuhan is higher because Wuhan has more POIs to climb, which makes the density of scenic edges in the road network higher.



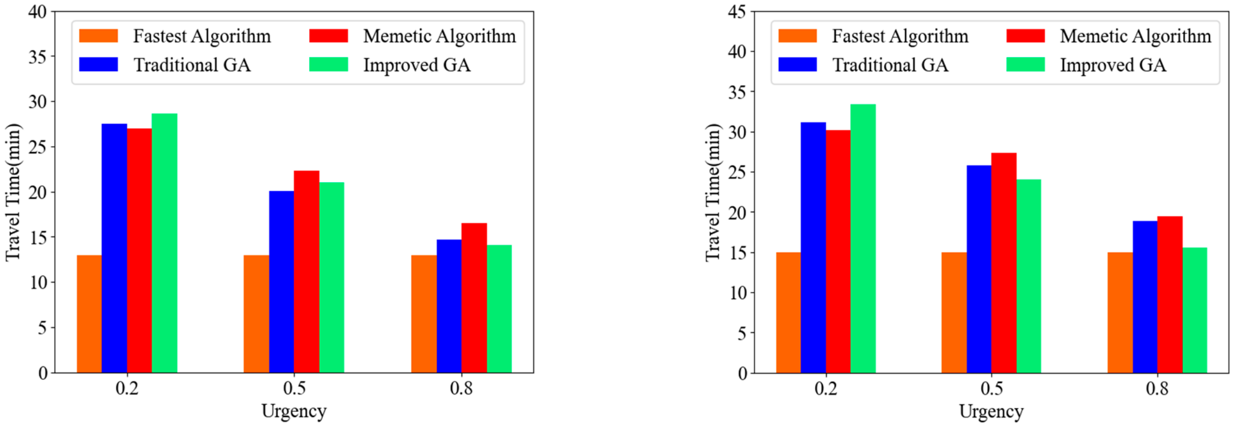
(**a**) (**b**)

Figure 9. Comparison of the running time of the algorithm under different start end pairs. (**a**) Xi'an; (**b**) Wuhan.

4.2.3. Sensitivity Analysis of Algorithm to Urgency

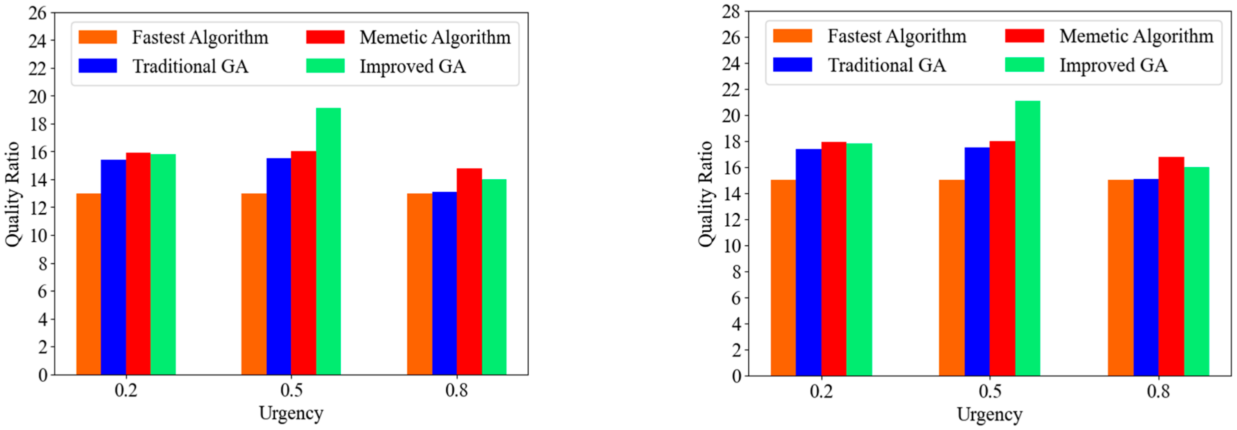
To verify that the algorithm in this paper can plan different travel routes according to different urgency levels, three urgency degrees were selected, namely, $urgency = 0.2$, $urgency = 0.5$, and $urgency = 0.8$. The travel time, quality ratio, scenic score, and running time were tested on the Xi'an and Wuhan road network datasets (departure time t_0 = 9:00). The average results are shown in Figure 10. Figure 10 shows that for different urgencies, the travel time of the fastest algorithm is always the same and the shortest because the fastest algorithm plans the shortest path between two points and is not affected by urgencies. When $urgency = 0.2$, the improved GA aims to plan the route with the highest scenic score for users. It can search more edges with high scenic scores, so the travel time is slightly higher than that of the MA and the traditional GA. When $urgency = 0.5$, the goal of the improved GA is to plan a route with a high-quality ratio for users, taking into account both the scenic score and travel time of the edges. Therefore, the travel time is lower than that of the MA. When $urgency = 0.8$, the improved GA and the traditional GA aim to plan the shortest travel time route for users, however, due to the weak search ability of traditional GA, the travel time is high. The MA is still used to plan the scenery route meeting time constraints for users. Therefore, the travel time of the improved GA is close to that of the fastest algorithm, while that of the MA is the highest.

The comparison results of the path quality ratio under different urgencies are shown in Figure 11. In Figure 11, for different urgencies, the quality ratio of the fastest algorithm is always the same and the lowest. When $urgency = 0.2$, the quality ratio of the MA is the highest because in this case, the improved GA will add more scenic edges to the path to maximize the scenic score of the path. The quality ratio of the improved GA is better than that of the traditional GA, because the gene replacement operator improves the search ability of the GA. When $urgency = 0.5$, the improved GA can maximize the quality ratio of the path, so the quality ratio is the highest. When $urgency = 0.8$, the quality ratios of several algorithms are close because there is not enough time budget to search for higher quality edges.



(**a**) (**b**)

Figure 10. Comparison of route travel time under different urgencies. (**a**) Xi'an; (**b**) Wuhan.



(**a**) (**b**)

Figure 11. Comparison of path quality ratio under different urgencies. (**a**) Xi'an; (**b**) Wuhan.

The comparison results of the scenic path score under different urgencies are shown in Figure 12. For different urgencies, the scenic score of the fastest algorithm is always the same and the minimum. When $urgency = 0.2$, the scenic score of the improved GA is the highest because when it is not urgent, the goal of the improved GA is to maximize the scenic score of the path, and compared with the MA and the traditional genetic, it can search more and better scenic edges. When $urgency = 0.5$, although the goal of the improved GA is to maximize the quality ratio of the path, the path scenic score is still the largest. When $urgency = 0.8$, the scenic score of the improved GA is close to that of the fastest algorithm and slightly lower than that of the MA. The reason for this result is that the MA still chooses the scenic edge when encoding chromosomes, and the running time and travel time of the algorithm will be relatively high. The scenic score of the traditional GA is slightly higher than that of the improved GA, because although the traditional GA also plans the fastest route for users at this time, its search ability is weaker than that of the improved GA, so the path travel time is higher.

Finally, the comparison results of the algorithm running time under different urgencies are shown in Figure 13. For different urgencies, the running time of the fastest algorithm is the same and always the shortest. With $urgency = 0.2$ and $urgency = 0.5$, the running time of the improved GA is slightly lower than that of the MA for two reasons. First, the improved GA's gene replacement operator has accelerated the convergence of the algorithm to a certain extent. Second, the chromosome encoding method of the MA may generate a large number of chromosomes, resulting in a slower search speed. When $urgency = 0.8$, the running time of the improved GA is close to the fastest algorithm, while the running time of the MA is much higher because the chromosome encoding strategy of the MA in this case is still to maximize the path scenic score, which wastes considerable time. On the whole, compared with the traditional GA, the execution time of the improved GA is much faster, which shows that the existence of both the gene replacement and gene splicing operators greatly improves the efficiency of the algorithm.

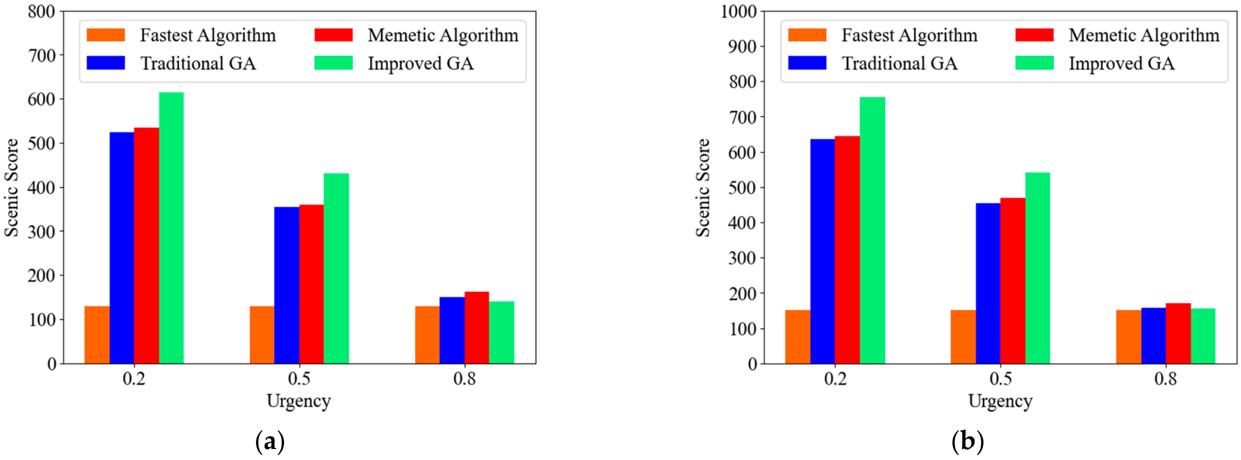


Figure 12. Comparison of scenic score under different urgencies. (**a**) Xi'an; (**b**) Wuhan.

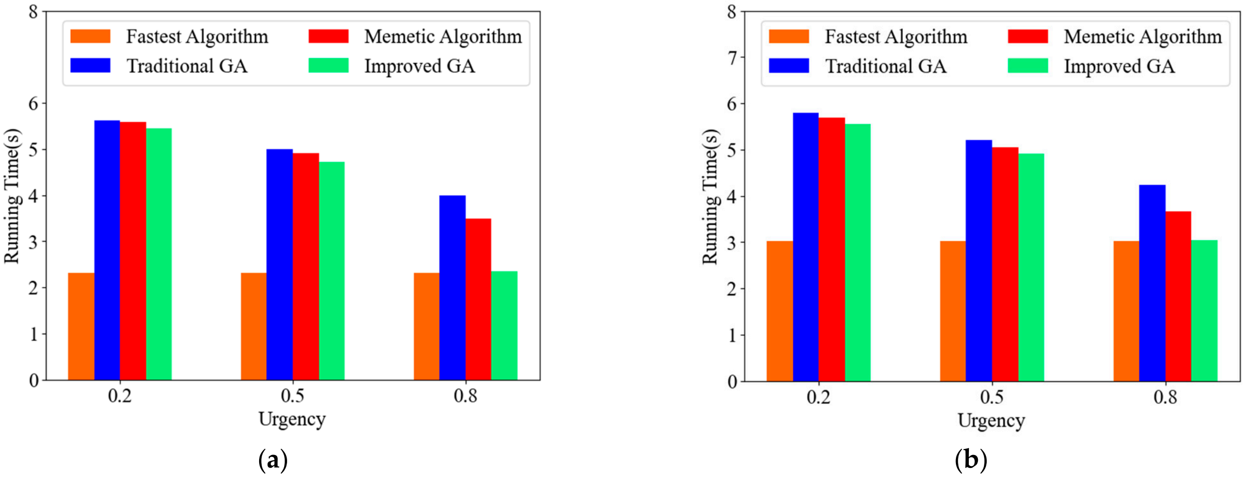


Figure 13. Comparison of algorithm running time under different urgencies. (**a**) Xi'an; (**b**) Wuhan.

In summary, it is not difficult to see that when $urgency = 0.2$, the improved GA can plan the route with the highest scenic score for users; when $urgency = 0.5$, the improved GA can plan the route with the highest quality ratio for users; and when $urgency = 0.8$, the improved GA can plan a route close to the fastest path for users. Therefore, the improved GA can plan routes of different functions for users according to different values of urgency.

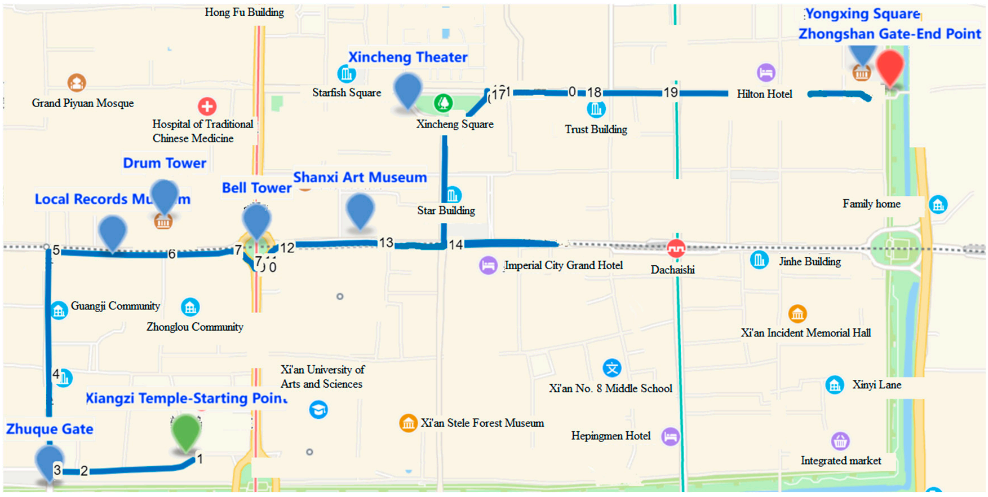
4.2.4. Sensitivity Analysis of Algorithm to Users

To prove that the proposed method can facilitate personalized scenic route planning, three different users (i.e., different preferences for POI types) were selected, and experiments conducted on three benchmark algorithms and the proposed algorithm many times. The results are shown in Table 3 (the experiment was conducted on the Xi'an road network dataset, and the departure time $t_0 = 9:00$). The table shows that for three different users, the route planned by the fastest algorithm is exactly the same because the fastest algorithm looks only for the fastest path between two given points, regardless of other conditions of the path. In contrast, the improved GA can plan different driving routes for different users. The reason for this is that here user preference information in the process of POI scenic score calculation is introduced, that is, users with different preferences can obtain different reward values when visiting a POI, thus ensuring that an edge with more POIs preferred by the user is more easily found during a path search. Compared with the MA and the traditional GA, the improved GA makes greater improvements to the scenic score of the path, and the running time is the shortest.

Table 3. Comparison of 30 running results of different users on four algorithms.

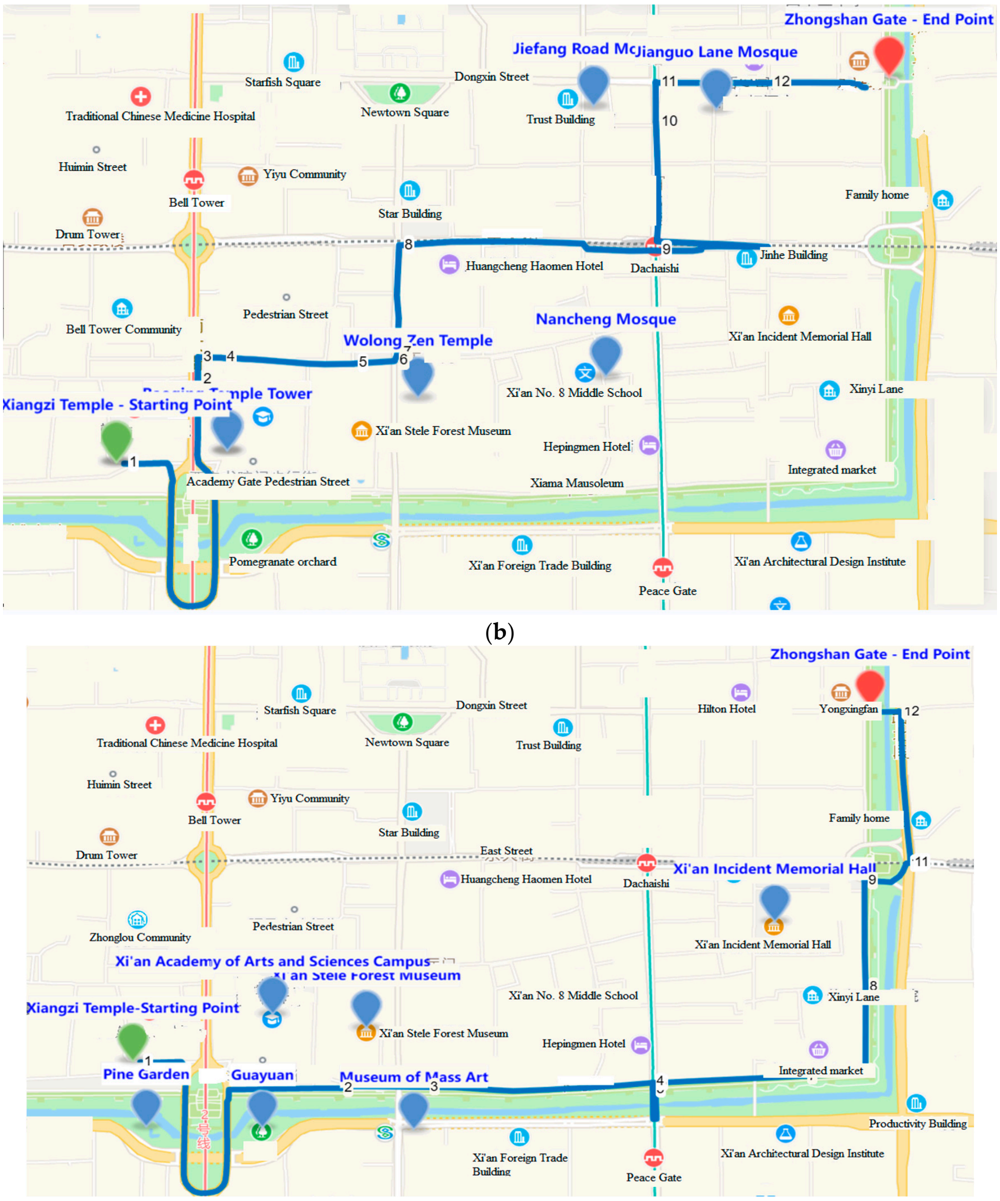
User			Fastest Algorithm	Traditional GA	Memetic Algorithm	Improved GA
u_1	scenic score	best value	301.25	534.82	555.38	615.03
		average value		505.43	512.41	580.35
	running time(s)	best value	2.31	6.80	6.78	6.61
		average value		7.02	6.96	6.75
u_2	scenic score	best value	301.25	475.42	496.21	584.15
		average value		451.23	472.63	560.24
	running time(s)	best value	2.31	5.51	5.47	5.35
		average value		5.62	5.59	5.48
u_3	scenic score	best value	301.25	392.84	412.25	454.51
		average value		370.51	395.32	428.46
	running time(s)	best value	2.31	5.51	5.36	5.25
		average value		5.63	5.41	5.33

The starting point $O(Xiangzi\ Temple)$, the ending point $D(Zhongshan\ Gate)$, $urgency = 0.2$, and t_0 = 9:00 are set here. The preferences of the selected three users are, respectively, $P(u1)$:($historical\ sites$, $cultural\ venues$, $characteristic\ buildings$), $P(u2)$:($religion$, $natural\ scenery$, $characteristic\ buildings$), and $P(u3)$:($cultural\ venues$, $natural\ scenery$, $red\ revolution$). The visualized results of the path data obtained from the experiment are shown in Figure 14a–c, respectively (the dark blue text in the figure describes the POI names that are close to the path and meet the user's preferences). In Figure 14a, user u1 can see the "Zhuque Gate", "Shaanxi Provincial Local Records Museum", "Drum Tower", "Bell Tower", "Shaanxi Art Museum", "Xincheng Theater", and "Yongxing Square" along the way. Among them, the "Zhuque Gate", "Bell Tower", and "Drum Tower" are historical sites and characteristic buildings, and the "Shaanxi Provincial Local Records Museum", "Shaanxi Art Museum", "Xincheng Theater", and "Yongxing Square" are cultural venues. These POIs not only improve the quality of path scenery but also make user u1's travel more fun.



(**a**)

Figure 14. *Cont.*



(**b**)

(**c**)

Figure 14. Visualization results of users u_1, u_2, and u_3 path data. (**a**) u1; (**b**) u2; (**c**) u3.

In Figure 14b, user u2 passes through the "Baoqing Temple Tower", "Wolong Zen Temple", "Nancheng Mosque", "Jiefang Road Mosque", and "Jianguo Lane Mosque" along the way. These POIs have religious, scenic, characteristic buildings, and other characteristics, which are in line with user u2's interests and preferences.

In Figure 14c, user u3 passes through the "Xi'an Academy of Arts and Sciences Campus", "Pine Garden", "Guayuan", "Xi'an Stele Forest Museum", "Museum of Mass Art", and "Xi'an Incident Memorial Hall" along the way. The "Pine Garden" is rich in vegetation, clean in environment, and adjacent to the moat and ancient city wall. The "Pomegranate Garden" displays the cultural characteristics of the "Silk Road". While exhibiting Xi'an's culture, it has also become a "city complex" integrating culture, tourism, commerce, and life.

5. Conclusions

In this paper, a personalized scenic tourism planning model based on urgency is proposed according to tourists' personalized needs for scenic tourism routes. A large number of experiments were conducted on the road network datasets of Xi'an and Wuhan. The results show that the proposed model can plan travel routes with different functions that meet the conditions of users, and the scenic features along the routes are in line with users' preferences.

In the future, we will deepen our research from the following aspects. First, the time factor will be introduced into the path planning process, that is, the change in POI scenic score with time will be considered so that the scenic score of the path can be calculated more accurately. Second, according to the obtained path, the nearest neighbour POI will be extracted, the POI scored according to user preferences and the best travel time, and a personalized guidance scheme generated to guide them to sign in to the POI along the way. Finally, we will add a personalized scenic tourism route planning module on the smart tourism platform, apply this method to the system, test the system by recruiting volunteers, and collect relevant suggestions to continuously improve the system.

Author Contributions: Conceptualization, methodology, L.W.; formal analysis, investigation, supervision, project administration, funding acquisition, L.W., X.X., Q.J., W.L. and S.Z.; resources, data curation, L.W. and X.X.; writing—original draft preparation, L.W. and X.X.; writing—review and editing, L.W., Q.J. and W.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by: (1) the National Natural Science Foundation of China, grant number 62176146, 62272384; (2) the National Social Science Foundation of China, grant number 21XTY012; (3) the National Education Science Foundation of China, grant number BCA200083; (4) Key Project of Shaanxi Provincial Natural Science Basic Research Program, grant number 2023-JC-ZD-34.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chen, X.; Chen, C.; Liu, K. Multi source heterogeneous crowdsourcing data landscape travel route planning. *J. Zhejiang Univ. Eng. Ed.* **2016**, *50*, 1183–1188.
2. Chen, C.; Jiao, S.; Zhang, S.; Liu, W.; Feng, L.; Wang, Y. TripImputor: Real-Time Imputing Taxi Trip Purpose Leveraging Multi-Sourced Urban Data. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 3292–3304. [CrossRef]
3. Ma, X.; Sun, M.; Gang, Z.; Liu, X. An Efficient Path Pruning Algorithm for Geographical Routing in Wireless Networks. *IEEE Trans. Veh. Technol.* **2008**, *57*, 2474–2488.
4. Vansteenwegen, P.; Souffriau, W.; Oudheusden, D.V. The orienteering problem: A survey. *Eur. J. Oper. Res.* **2011**, *209*, 1–10. [CrossRef]
5. Golden, B.; Levy, L.; Vohra, R. The orienteering problem. *Nav. Res. Logist.* **1987**, *34*, 307–318. [CrossRef]
6. Lu, Y.; Shahabi, C. An arc orienteering algorithm to find the most scenic path on a large-scale road network. In Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, Seattle, WA, USA, 3–6 November 2015; pp. 1–10.
7. Chen, R.; Hu, J.; Xu, W. An RRT-Dijkstra-Based Path Planning Strategy for Autonomous Vehicles. *Appl. Sci.* **2022**, *12*, 11982. [CrossRef]
8. Chai, Q.; Wang, Y. RJ-RRT: Improved RRT for Path Planning in Narrow Passages. *Appl. Sci.* **2022**, *12*, 12033. [CrossRef]
9. Chao, C.; Xia, C.; Wang, L.; Ma, X.; Zhu, W.; Kai, L.; Guo, B.; Zhen, Z. MA-SSR: A Memetic Algorithm for Skyline Scenic Routes Planning Leveraging Heterogeneous User-Generated Digital Footprints. *IEEE Trans. Veh. Technol.* **2017**, *66*, 5723–5736.
10. Zheng, Y.; Yan, S.; Zha, Z.; Li, Y.; Zhou, X.; Chuan, T.; Jain, R. GPSView: A scenic driving route planner. *ACM Trans. Multimed. Comput. Commun. Appl.* **2013**, *9*, 1–18. [CrossRef]
11. Skoumas, G.; Schmid, K.A.; Jossé, G.; Züfle, A.; Pfoser, D. Towards Knowledge-Enriched Path Computation. In Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Dallas, TX, USA, 4–7 November 2014; pp. 485–488.
12. Li, S.; Ding, M.; Chao, C.; Lei, J. Efficient Path Planning Method Based on Genetic Algorithm Combining Path Network. In Proceedings of the Fourth International Conference on Genetic and Evolutionary Computing, Shenzhen, China, 13–15 December 2010; pp. 194–197.

13. Demiryurek, U.; Banaei-Kashani, F.; Shahabi, C.; Ranganathan, A. Online Computation of Fastest Path in Time-Dependent Spatial Networks. In Proceedings of the Advances in Spatial and Temporal Databases: 12th International Symposium, SSTD 2011, Minneapolis, MN, USA, 24–26 August 2011; pp. 92–111.
14. Gao, L.; Chen, C.; Huang, H.; Xiang, C. A Memetic Algorithm for Finding the Two-fold Time-dependent Most Beautiful Driving Routes. In Proceedings of the 2019 IEEE Wireless Communications and Networking Conference, Marrakesh, Morocco, 15–18 April 2019; pp. 1–6.
15. Chen, C.; Gao, L.; Xie, X.; Wang, Z. Enjoy the Most Beautiful Scene Now: A Memetic Algorithm for Finding the Two-fold Time-dependent Arc Orienteering Problem. *Front. Comput. Sci.* **2020**, *14*, 364–377. [CrossRef]
16. Lu, Y.; Jossé, G.; Emrich, T.; Demiryurek, U.; Renz, M.; Shahabi, C.; Schubert, M. Scenic routes now: Efficiently solving the time-dependent arc orienteering problem. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; pp. 487–496.
17. Chen, C.; Gao, L.P.; Xie, X.; Wang, Y. 2TD Path-Planner: Towards a More Realistic Path Planning System over Two-Fold Time-Dependent Road Networks [Application Notes]. *IEEE Comput. Intell. Mag.* **2021**, *16*, 78–98. [CrossRef]
18. Verbeeck, C.; Sörensen, K.; Aghezzaf, E.; Vansteenwegen, P. A fast solution method for the time-dependent orienteering problem. *Eur. J. Oper. Res.* **2014**, *236*, 419–432. [CrossRef]
19. Lu, Y.; Benlic, U.; Wu, Q. A memetic algorithm for the Orienteering Problem with Mandatory Visits and Exclusionary Constraints. *Eur. J. Oper. Res.* **2018**, *268*, 54–69. [CrossRef]
20. Quercia, D.; Schifanella, R.; Aiello, L. The Shortest Path to Happiness: Recommending Beautiful, Quiet, and Happy Routes in the City. In Proceedings of the 25th ACM Conference on Hypertext and Social Media, Santiago, Chile, 1–4 September 2014; pp. 116–125.
21. Taylor, K.; Lim, K.; Chan, J. Travel Itinerary Recommendations with Must-see Points-of-Interest. In Proceedings of the Companion of the Web Conference, Lyon, France, 23–27 April 2018; pp. 1198–1205.
22. Liang, H.; Wang, K. Top-k route search through submodularity modeling of recurrent poi features. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 16 November 2018; pp. 545–554.
23. Jiang, Q.; Teng, W.; Liu, Y. ORSUP: Optimal Route Search with Users' Preferences. In Proceedings of the 2019 20th IEEE International Conference on Mobile Data Management (MDM), Hong Kong, China, 10–13 June 2019; pp. 357–358.
24. Zhang, Y.; Jiao, L.; Yu, Z.; Lin, Z.; Gan, M. A Tourism Route-Planning Approach Based on Comprehensive Attractiveness. *IEEE Access* **2020**, *8*, 39536–39547. [CrossRef]
25. Huang, F.; Xu, J.; Weng, J. Multi-Task Travel Route Planning with a Flexible Deep Learning Framework. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 3907–3918. [CrossRef]
26. Qiu, N.; He, Z.; Wang, P.; Li, Y. Research on Recommendation Algorithm Based on User Preference Optimization Model. *Appl. Res. Comput.* **2019**, *36*, 3579–3585.
27. Zheng, S. *Industrial Intelligent Technology and Application*; Shanghai Science and Technology Press: Shanghai, China, 2019; pp. 250–251.
28. Wang, T.F. *Grid Trust Model Based on Family Genes in Computer Genetics*; Intellectual Property Press: Beijing, China, 2016; pp. 93–94.
29. Damos, M.A.; Zhu, J.; Li, W.; Hassan, A.; Khalifa, E. A Novel Urban Tourism Path Planning Approach Based on a Multiobjective Genetic Algorithm. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 530. [CrossRef]
30. Janeš, G.; Ištoković, D.; Jurković, Z.; Perinić, M. Application of Modified Steady-State Genetic Algorithm for Batch Sizing and Scheduling Problem with Limited Buffers. *Appl. Sci.* **2022**, *12*, 11512. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.