

Intelligent Taxi Dispatching Based on Improved Artificial Fish Swarm Algorithm

Zhiwei Luo¹, Rong Xie²(✉), Wangyi Huang², and Yiwei Shan²

¹ School of Automation, Huazhong University of Science and Technology,
1037 Luoyu Road, Wuhan 430074, China
zhiweiluo@hust.edu.cn

² International School of Software, Wuhan University, 39 Luoyu Road, Wuhan 430079, China
{xierong, wangyihuang, shanyiwei}@whu.edu.cn

Abstract. Taxi plays an important role in public transportation. However, in real life, it usually takes a long time for passengers to wait for an unoccupied taxi, while number of no-load taxi is very high. Optimized scheme is required to handle global taxi dispatching. The paper presents a novel method of taxi dispatching based on Artificial Fish Swarm Algorithm (AFSA). To avoid slow convergence and falling into local optimum, standard AFSA is improved using strategies of optimizing visual range, step size and swimming behavior. We compare our improved algorithm with standard AFSA and Particle Swarm Optimization through the evaluation with real data of GPS trajectory of 12,000 taxis in Beijing and make performance analysis of effects of parameter configuration to dispatching algorithm. The experimental results show good effectiveness of our method.

Keywords: Taxi · Taxi dispatching · Improved Artificial Fish Swarm Algorithm (AFSA) · Intelligent Transportation

1 Introduction

Nowadays, taxis provide people with fast and comfortable service. However, there are still some problems in taxi dispatching. Lack of a good solution to global taxi dispatching, which bringing about asymmetric information between taxi drivers and passengers, it usually costs a long time for passenger to wait for an unoccupied taxi; while simultaneously taxi drivers wander in streets, which leads to severe waste of fuel and traffic resources. In recent years, under the help of software for taxi reservation, for example, “DiDi” in China, unoccupied taxi nearby passenger can be quickly assigned to passenger who calls for a taxi by mobile phone. Though this method indeed reaches some achievements in alleviating the contradiction between passenger and taxi driver, it fails to make global optimization on taxi resources. Efficiently taxi dispatching is urgently required in dealing with this issue.

Methods for taxi dispatching are studied in the past years, which are mainly divided into two stages, manual scheduling and intelligent scheduling. With burgeon of taxis amount, due to low effectiveness, manual scheduling cannot meet passenger’s demand.

Intelligent taxi scheduling shows promise. Gilbert et al. pioneered improved taxi dispatching method by using global positioning system (GPS). In their simulation model with 16 teams and 300 cars in each team, the time wasted on taxi dispatching was decreased by 50–60%. Rosenschein and Zlotkin [1] proposed a method with association structure to allocate taxi for passengers. Aiming at minimum time waste on taxing arriving at passenger's position, Lee et al. [2] proposed intelligent taxi scheduling system based on road traffic information. Dorer and Calisti [3] proposed a method based on dynamic traffic information for the optimization for taxis' driving route. Seow et al. [4, 5] proposed the idea of distributed taxi dispatching. Sung and jong [6] predicted cost for transportation on each road for next period of time based on real-time and historical GPS data, and then used optimized Floyd-Washa algorithm for minimized path length. Sen et al. [7] proposed a path selection algorithm for time-varying traffic networks based on minimum expectations and variance. For practical applications, US Module Company once developed a vehicle dispatch management system which could achieve real-time GPS positioning, monitored the current state of taxis, and sent cars in a reasonable manner. In the recent, some researches tend to develop a recommender system for searching for passengers and vacant Taxis [8].

Differing from traditional methods mentioned above, learning from the thought of swarm intelligence in artificial intelligence, this paper presents a novel method to deal with the issues of taxi dispatching through AFSA. Optimization is made by dispatching taxis to an area with high density of passengers calling for taxis to realize the maximum efficiency of traffic resource utilization and minimum of waiting time.

2 AFSA and Its Improvements

2.1 Artificial Fish Vision Model

Under natural environment, as we can observe, fishes tend to go some areas with rich food resources. Such phenomenon can be interpreted with some behaviors as follows. (1) Foraging behavior. Fish usually swims in water without a certain target. But once food is found, fish will quickly swim to the food. (2) Clustering behavior. Individual fish tends to join with some other fishes to become swarm without certain guidance or organization but gather together in water to gain strengthened acuity to predators and resistance to disasters. (3) Following behavior. When a fish finds food in some areas, other fishes nearby would immediately change direction towards their companions. (4) Random behavior. If fish does not perform one of the above behaviors, it will search for food randomly in water.

In order to simulate these characteristics of natural fish, we can build an artificial fish vision model as follows. $X = (X_1, X_2, \dots, X_n)$ is the current state of artificial fish, where X_i ($i = 1, 2, \dots, n$) is each variable to be optimized. Food concentration of the current location of X is $Y = f(X)$, representing the value of objective function. *vision* is the current visual range of artificial fish. X_v is location of artificial fish's viewpoint at a certain moment. *step* is the maximum movement length for artificial fish's one step. X_{inext} is the next position after artificial fish makes a movement, which can be calculated according to Eq. (1).

$$X_{next} = X_i + rand() \times step \times (X_i - X_j) / \|X_i - X_j\| \quad (1)$$

where, $rand()$ generates a random number between $[0, 1]$. X_i and X_j represent the state of two companion artificial fishes respectively, and distance between them can be calculated as $d_{ij} = \|X_i - X_j\|$.

2.2 Standard AFSA Improvement

Although standard AFSA [9] has many advantages, some shortcomings can be found in actual applications. Neshat et al. [10] once made a survey of the existing artificial fish swarm algorithm. In the paper, we make improvements on it in the aspects of optimizing visual range and step size, as well as swimming behaviors as follows.

Improvement 1: Visual range and step size. Vision range $vision$ has a great influence on the algorithm for both behavior and convergence performance. In general, scale of visual range will impact convergence and search ability. With rather larger visual range, artificial fish's behavior mainly embodied in following behavior and clustering behavior, and the algorithm will have quick convergence and strong global search ability. Adversely, with rather smaller visual range, artificial fish's behavior mainly embodied in foraging behavior and movement pattern show more randomness, and algorithm will be with slow convergence and cost more computing resources, but has strong local search ability. On the other hand, step size $step$ will also impact on algorithm convergence. When step size is small, increment of step size will accelerate convergence of algorithm, but when it exceeds a certain value, the increase of step size will reduce convergence, and in the even worse case, leading to oscillation and failure to find optimal solutions. So, at the primary iterations of the algorithm, rather larger scale of visual range and step size should be chosen in order to accelerate convergence and enhance search ability of global optimal result. When algorithm gradually converge, the given result is the optimal solution, step length and visual range should be adjusted to rather small size to avoid oscillation and make sure that the result will approach the optimal solution progressively. The specific parameter configuration can be determined by Eq. (2).

$$\begin{cases} a = \exp(-30 \times (t / t_{max})^s) \\ vision = vision \times a + vision_{min} \\ step = step \times a + step_{min} \end{cases} \quad (2)$$

where, t is the current number of iteration. t_{max} is the maximum number of iteration. The minimum value of visual field is $vision_{min} = 0.001$. $step_{min} = 0.0002$. s is an integer which is greater than 1, between $[1, 30]$ in the paper. Using the method, vision range and step size are the maximum at the beginning, and then gradually become smaller and finally reach the minimum.

Improvement 2: Swimming behaviors. In the standard AFSA, when artificial fish tries $trynum$ times, if it still does not satisfy with forwarding condition, it will perform

random behavior. However, if artificial fish has been already in the vicinity of the optimal solution, it may swim far away from the optimal position which will affect the results of the algorithm. In the implementation of cluster behavior and following behavior, if the algorithm has approached to global optimal solution, then forging behavior may cause fish to be deviated from the global optimal solution. The same problem will be encountered for performing clustering behavior and following behavior. Therefore, the improvements on swimming behaviors can be performed in the following aspects. (1) Forging behavior. When artificial fish has tried $trynum$ times, if food concentration in the current state is greater than $1/n$ of optimal food concentration Y_{best} , then current state is maintained; otherwise random behavior is performed. (2) Clustering behavior. Clustering behavior will be continued only if food concentration Y_c at center of all companion fishes found in the vision region is greater than $1/n$ of optimal food concentration Y_{best} ; otherwise, foraging behavior is performed. (3) Following behavior. The following behavior will be continued only if the maximum food concentration Y_{max} of all companion fishes found in vision range is greater than $1/n$ of optimum food concentration max ; otherwise foraging behavior is performed.

3 Taxi Dispatching

3.1 Taxi Dispatching Problem (TDP)

The basic idea of taxi dispatching is to dispatch taxi to an area with high density of passengers. Similarly, AFSA makes artificial fishes search for an area with highest concentration of food. Taxi dispatching model can be described as follows. The scope that taxi dispatching can be divided into R areas. Suppose there are P passengers calling for taxi in this scope at a moment, with amount of passengers P_i in area i ($i = 1, 2, \dots, R$), then $P = \sum_{i=1}^R P_i$. The amount of empty taxis in each area is represented by M_i . Suppose that a taxi T_j can accept the maximum no-load distance being d_{T_j} with the above conditions. TDP, that taxi dispatching is to make taxis go to the area with highest density and also no-load distance for taxi should be as short as possible, is described as $max Opportunity(P_i, T_j) \wedge (Distance(P_i, T_j) \leq d_{unloading_threshold})$. The possible dispatching results may be as follows. (1) The area where passengers' density is the largest will be satisfied first, and followed by the area with the second largest density of passengers, and so on. (2) Assuming that there are N areas with a relatively larger density, number of passenger of every taxi's currently area is P_i . Crowding factor cf is used to determine whether number of taxi gathered in a certain area exceeds a certain threshold. Set the area with the maximum density of passengers to N_{max} , number of passenger in this area to P_{max} , and number of free taxi in range to M_{max} . So, if $P_{max}/M_{max} > CP_i$, then the taxi will be distributed to N_{max} .

3.2 Taxi Dispatching Model

The waste of taxi resource is mainly caused by inappropriate dispatching. A feasible solution is to quickly dispatch available taxi to passenger nearby according to dynamic

passenger's order. With terminal platform on taxi and client of mobile communication by passenger's mobile phone, passengers' information, such as location and time, can be captured from client and uploaded to server. Then they will be used as feedback for conducting proper guidance for taxi drivers. In this way, reasonable dispatching of taxi resources can be performed according to dynamic information.

Applying AFSA to taxi dispatching, we can regard passengers calling for taxi as food resource in a virtual pond, and taxi as an artificial fish changing its behaviors according to change of environment. Let X_i be location of taxi's visual point at a certain time, $step$ be the maximum step of taxi to move at one time, and X_{next} be location of taxi after one step forward. Distance between two taxis X_i and X_j can be calculated by $d_{ij} = \|X_i - X_j\|$. Then movement of taxi can be modeled as Eq. (3).

$$X_{next} = X_i + rand() \times step \times (X_c - X_i) / \|X_c - X_i\| \quad (3)$$

In real-life environment, passengers usually gather around some hotspot area, such as shopping malls, schools, railway stations and etc. So, in some certain scope, there may simultaneously exist several areas with high passenger density. When simulating this situation, area with different density of passengers is abstracted as a spatial local maximum point, where there is a unique global optimal maximum point. Finally, through performing ASFA, artificial fish distributed randomly will be dispatched to the final position. We can compare number of artificial fish near the global maximum point and the average distance of taxis from the optimal location, and use their final location to determine the algorithm's performance and effect to solve the problem.

3.3 Implementation

The steps of taxi dispatching based on our improved AFSA are described as follows.

Step 1: Initialize setting of the algorithm, including initial number of taxi, random distribution of initial position of taxi, location of passenger, visual range $vision$, length of one step forward $step$, crowding factor cf etc.

Step 2: Calculate size of passengers' density at location of all initial locations of taxis and record them in an array of the optimal state ($bestx_i, besty_i$).

Step 3: Implement evaluation function, the results of it will be used for taxi to choose one of the four basic behaviors of artificial fish, i.e. foraging behavior, clustering behavior, following behavior, and random behavior. The optimal behavior will be selected to implement.

Step 4: Update optimal state array ($bestx_i, besty_i$).

Step 5: Check whether termination condition is satisfied or not. If termination condition is satisfied, output the result; otherwise go back to execute Step 3.

4 Experiment and Performance Test

4.1 Raw Data and Test Function

The original data used in our experiment came from GPS big data which was generated by 12,000 taxis in Beijing in November 2012.

This experiment is performed by MATLAB R2013a in Windows 7 system. Experimental parameters are initial number of artificial fish *fishnum*, vision range of artificial fish *vision*, step size of one step forward *step*, passenger's crowding factor *cf*, number of attempt *trynum* and maximum number of iteration *maxgen*.

The essence of TDP is how to optimize allocation of taxi resources to regional centres with the highest density of passengers through an effective dispatching method. The purpose of this experiment is to test whether artificial fishes can be eventually clustered in the region with the largest concentration of food using the improved AFSA, i.e. in a certain spatial range, centre of region with global optimal maximum. Simulation test function, i.e. $z(x, y) = \sin(x) \times \sin(y)/(x \times y)$, is selected in our experiment.

As shown in Fig. 1, the test function, having a unique global maximum, with the size of 1 located at (0, 0), and many local optimal extremes near the global optimal maximum point, can be used to test effectively whether the results of the algorithm are in globally optimal feature or not. In addition, as standard AFSA or other optimization algorithms is often easy to fall into local optimal or shock near the local optimal region, but the improved AFSA through theoretical analysis that the searching results can be well gathered around global optimal maximum point, this test function is more suitable for verifying optimization performance of the algorithm in the experiment. In the simulation experiment, the initial position of taxi is randomly distributed at a position that maybe far from the global maximum point. After executing the improved AFSA, we can compare the result of the optimization algorithm with the result of the simulation function, and to check whether the distribution results can be clustered near the global optimal maximum point.

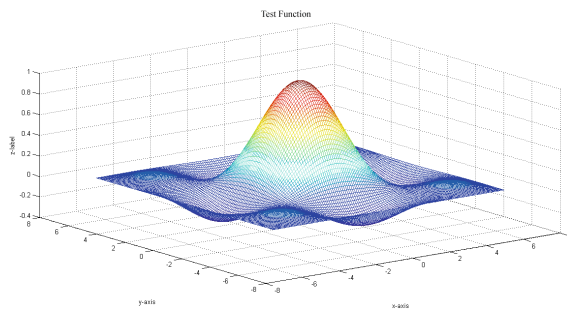


Fig. 1. Simulation of test function.

4.2 Comparison Analysis

Figure 2 gives comparison results of distribution map of taxi dispatching among standard AFSA, Particle Swarm Optimization (PSO) and improved AFSA. Black represents

distribution-intensive area of passengers, red asterisk as the initial distribution of taxi, and blue circle as the final distribution of taxis using the algorithms. The horizontal coordinate indicates distance x between taxi and the area with the maximum density of passengers, and the vertical coordinate indicates distance y between taxi and the area with the maximum density of passengers. Figure 2 is a partial enlarged view of central area, number of taxi is $fishnum = 216$.

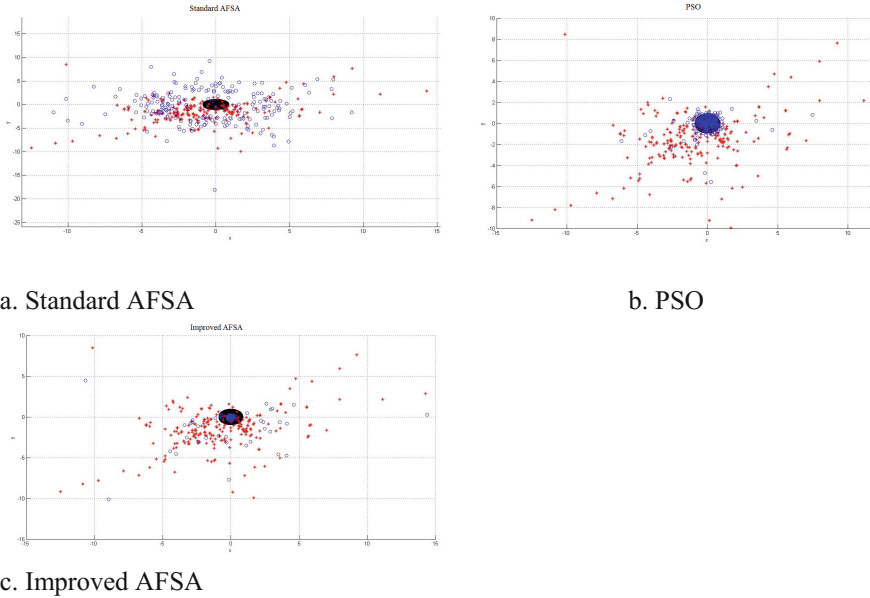


Fig. 2. Comparison of taxi dispatch distribution map among standard AFSA, PSO and improved AFSA.

As seen from Fig. 2a, only few taxis can be gathered in the area where passengers are most densely distributed under the standard AFSA, and number of taxi near optimal location is 2. The average distance between the assigned taxi and the optimal location is 8.33346 (Due to distance in the original data is represented with latitude and longitude, this experiment is carried out at a certain scale for magnification processing, so there is no specific physical unit). Under the PSO, shown in Fig. 2b, part of taxis can be gathered in the area where passengers are most densely distributed, and the overall distribution is close to the optimal position. Number of taxi near the optimal location is 123, average distance of the taxi from the optimal location is 0.900788. Under the improved AFSA, shown in Fig. 2c, most of taxis can be gathered to the area where passengers are most densely distributed, with number of taxi near the optimal location is 183, the average distance of taxi from the optimal location is 0.820658. Through the experiment, the improved AFSA represents its better optimization performance and better optimization effect comparing with standard AFSA and PSO.

4.3 Performance Analysis

We also analyze the influence of parameter setting on optimization performance of our improved algorithm as follows.

The impact of number of attempt *trynum* of taxi on optimal performance of the algorithm is shown in Fig. 3. As we can see, *trynum* has little effect on the optimization performance, if other parameters are set to the same. When *trynum* increases, artificial fish can search for more locations, but there is no guarantee that it can find a better location.

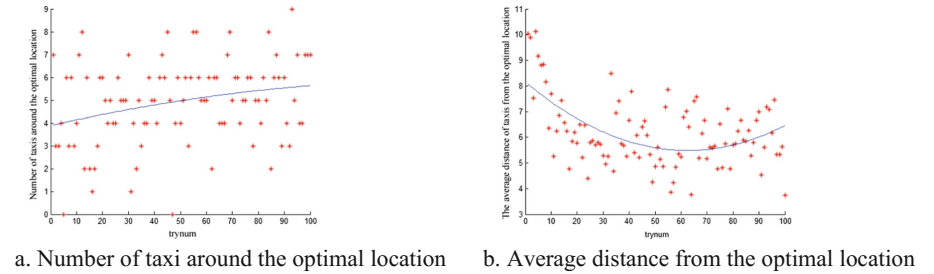


Fig. 3. Influence of *trynum* on optimal performance of the improved algorithm.

The effect of moving step *step* of taxi (artificial fish) on optimization performance of the algorithm is shown in Fig. 4. Under the condition that the other parameters are set to the same, the effect of improved AFSA is very good when *step* is about 50–65% of *vision*. Within a certain region of view, the greater the *step* is, the faster is to reach the area with the optimal maximum point, and the more potential position will be found when random behavior is performed. However, when *step* exceeds a certain range, it causes artificial fish to shock in the vicinity of optimal value, that optimization performance of the algorithm will be greatly weakened.

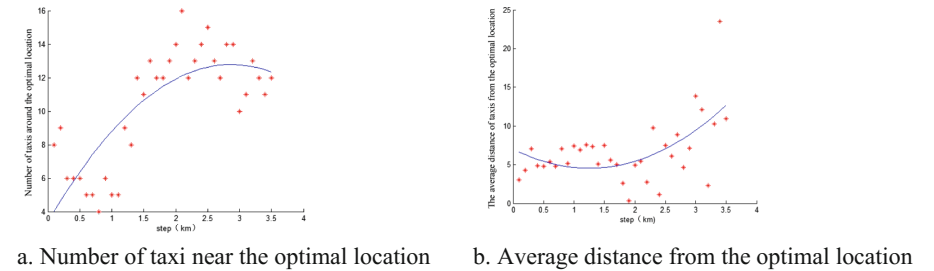
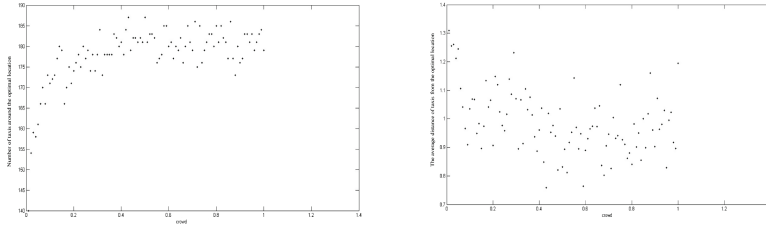


Fig. 4. Influence of *step* on optimal performance of the improved algorithm.

The influence of crowding factor *cf* on the optimization result of the algorithm is shown in Fig. 5. Under the other parameters are set to the same, the effect of *cf* on the experimental results is obvious when it is small. The greater *cf* is, the greater number of taxi near the optimal position is, and the smaller the average distance of taxi from the

optimal position is. It is shown that the introduction of cf has obvious effect on the optimization experiment and is better to avoid the local optimum. For the case of cf is large, the effect on number of taxi near the optimal location and on the average distance of taxi from the optimal position are both less. This is because clustering behavior and following behavior are all optimized. Fish will implement following behavior only if concentration of food at the center of all companion fishes (or the maximum food concentration of all companion fishes) is greater than $1/n$ of optimal food concentration Y_{max} ; otherwise foraging behavior is performed.



a. Number of taxi near optimal location b. Average distance of taxi from optimal location

Fig. 5. The influence of cf on optimal performance of the improved algorithm.

The influence of $maxgen$ on optimal performance of the algorithm is shown in Table 1. Under the other parameters are set to the same, when $maxgen$ reaches 1,500 and above, there is no great difference in the final results. It is proved that when $vision$, $step$, $maxgen$ and cf is fixed, we need to adjust $maxgen$ to about 4000 in order to assure fast convergence of the algorithm and avoidance of shocking in the vicinity of all optimal maximum area.

Table 1. Influence of $maxgen$ on optimal performance of the improved algorithm.

Iteration times ($maxgen$)	Number of taxi near optimal location	Average distance of taxi from optimal location
5	0	13.3674
10	0	10.5342
50	0	9.2456
100	2	7.6742
500	5	5.9894
1000	7	4.8273
1500	9	3.5927
4000	14	1.4283
5000	12	2.5856
10000	8	4.4769

5 Conclusion

In the paper, we propose a novel solution to taxi dispatching on the basis of improving the standard AFSA in the aspects of an adaptive visual field and step size, as well as swimming behaviors to optimize behavior of artificial fish. Some simulation experiments are conducted using MATLAB, showing that our improved AFSA has a better performance on taxi dispatching than standard AFSA and PSO.

In the paper, our experiment stays at the level of MATLAB simulation. Further work shall be combined with map to achieve more convincing experimental results.

Acknowledgment. This work is supported by Suzhou Science and Technology Bureau - International Cooperation Programme of Science and Technology under grant no. SH201213. The authors would like to thank Wei Pan, Maohui Lu and Yingfei Xiang at Wuhan University for their helpful algorithm development.

References

1. Rosenschein, J.S., Zlotkin, G.: Rules of Encounter: Designing Conventions for Automated Negotiation Among Computers, pp. 25–46. MIT Press, Cambridge (1994)
2. Lee, D.H., Wang, H., Cheu, R.L., et al.: A taxi dispatch system based on current demands and real-time traffic information. *Transp. Res. Rec.* **1882**, 193–200 (2004)
3. Dorer, K., Calisti, M.: An adaptive solution to dynamic transport optimization. In: Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems, Industry Track, Utrecht, The Netherlands, pp. 45–51 (2005)
4. Seow, K.T., Dang, N.H., Lee, D.H.: Towards an automated multiagent taxi-dispatch system. In: IEEE International Conference on Automation Science and Engineering, Scottsdale, AZ, USA, 22–25 September 2007, IEEE Xplore, pp. 1045–1050 (2007)
5. Seow, K.T., Dang, N.H., Lee, D.H.: A collaborative multiagent taxi-dispatch system. *IEEE Trans. Autom. Sci. Eng.* **7**(3), 607–616 (2010)
6. Sung, S.K., Jong, H.L.: A study on design of dynamic route guidance system using forecasted travel time based on GPS data and modified shortest path algorithm. In: IEEE/IEEJ/JSAI International Conference on Intelligent Transportation Systems, Tokyo, Japan, IEEE Xplore, pp. 44–48 (2002)
7. Sen, S., Pillai, R., Joshi, S., et al.: A mean-variance model for route guidance in advanced traveler information systems. *J. Transp. Sci.* **35**(1), 37–49 (2001)
8. Yuan, J., Zheng, Y., Zhang, L., et al.: Where to find my next passenger. In: Proceedings of the 13th International conference on Ubiquitous computing (UbiComp 2011), New York, pp. 1–10. ACM (2011)
9. Li, X.: A new intelligent optimization method-Artificial fish school algorithm. Hangzhou: Zhejiang University (2003)
10. Neshat, M., Sepidnam, G., Sargolzaei, M., et al.: Artificial fish swarm algorithm: a survey of the state-of-the-art, hybridization, combinatorial and indicative applications. *Artif. Intell. Rev.* **42**(4), 965–997 (2014)