Theory and Practice of Science and Technology
2025, VOL. 6, NO. 3, 40-48
DOI: 10.47297/taposatWSP2633-456907.20250603

Solving TSP based on the Hybrid Algorithm of GA and FLS

Yuhan Sun

Liaoning Technical University, Faculty of Electrical and Control Engineering, 125105

ABSTRACT
About implementing a hybrid algorithm of genetic algorithm (GA) and fuzzy logic to solve the traveling salesman problem (TSP). Fuzzy logic controller was used to dynamically adjust the running parameters of genetic algorithm to improve the speed of path optimization and avoid falling into the local optimal path. The simulation results show that the hybrid algorithm is more effective than GA in solving the TSP problem.

KEYWORDS
Genetic Algorithm (GA); Fuzzy Logic; Traveling Salesman Problem (TSP)

1 Introduction

The TSP problem is one of the famous problems in the field of mathematics. Suppose a traveling businessman wants to visit n cities, he must choose the path he wants to take. The limit of the path is that each city can only be visited once, and finally, he has to return to the original starting point. city. The goal of path selection is that the required path distance is the minimum value among all paths. The TSP problem is a combinatorial optimization problem and an NP-complete problem. It often takes a lot of time to use the usual solution method. At present, the common methods of path planning include (1) Artificial potential field method, A* algorithm, Dijkstra algorithm, etc. Traditional algorithms; (2) Artificial neural network, particle swarm algorithm, ant colony algorithm, and other intelligent optimization algorithms.

Genetic algorithms originated from computer simulation studies of biological systems. It is a random global search and optimization method developed by imitating the biological evolution mechanism in nature, drawing on Darwin's theory of evolution and Mendel's genetic theory. Its essence is an efficient, parallel, and global search method, which can automatically acquire and accumulate knowledge about the search space during the search process, and adaptively control the search process to obtain the best solution. Inspired by Darwin's theory of evolution "Natural selection, survival of the fittest", the genetic algorithm simulates the biological evolution process of natural selection, performs multi-directional searches by maintaining a group of potential solutions, and supports information in these directions composition and exchange.

The search in units of planes can find the global optimal solution better than the search in units of points. Genetic Algorithm is a kind of optimization search algorithm that imitates biological genetics and natural selection mechanism through artificial structure. It is a mathematical simulation of the biological evolution process and is one of the most important forms of evolutionary calculation.

To dynamically adjust the search speed of the genetic algorithm when solving the TSP problem and improve the performance of the algorithm, a Hybrid Algorithm of GA and Fuzzy Logic is proposed. The fuzzy logic controller is used to dynamically adjust the crossover probability and mutation probability of the offspring population, and adjust the evolution process of the population. The diversity and convergence of individuals in the medium can shorten the time for GA to solve a better path.

2 Literature Review

Genetic algorithms[1] and fuzzy logic systems are optimization techniques commonly used in the field of computer science. Genetic algorithms are inspired by the natural evolutionary process and use a set of genetic operations to optimize the solution to a given problem. The goal is to find optimal solutions in a large search space by simulating the process of natural selection and genetic variation.

Fuzzy logic systems, on the other hand, are based on fuzzy set theory and use fuzzy logic to model and analyze uncertainty and imprecision in real world problems. Fuzzy logic systems provide a way to represent and manipulate knowledge in the form of fuzzy rules and membership functions. This allows for a decision-making process that is more in line with human reasoning than traditional binary logic systems.

Combining genetic algorithm and fuzzy logic system [2-3] has been proved to be a powerful tool for solving complex optimization, hybrid genetic algorithm and fuzzy logic system approach has been used to solve multi-objective scheduling problems and obtain better performance than either technique alone A better solution. The selection of crossover probability and mutation probability in genetic algorithm will directly affect the evolution rate of the algorithm, the diversity of the population and the quality of the final solution. Fuzzy logic control can transform expert knowledge and accumulated experience into clear control strategies in genetic evolution, making it possible to dynamically calculate the control parameters of genetic algorithms. In the algorithm proposed in this paper, a two-input and two-output fuzzy logic controller is designed [4], and the information reflecting the population evolution rate ES(t) and diversity D(t) is used as the input variable of the fuzzy logic control, and the crossover probability and mutation probability as output variables, the convergence speed and solution diversity of the dynamic optimization algorithm.

In conclusion, both genetic algorithms and fuzzy logic systems are robust optimization techniques that have been successfully applied in various fields[4]. The combination of these techniques offers a promising approach to solving complex problems and provides a way to model uncertainty and imprecision while enabling system optimization.

3 Problem Modeling and Formulation

Using fuzzy logic to dynamically adjust the genetic algorithm parameters, the TSP path planning problem formula is as follows:

$$\min f(X,Y)$$
$$x_{\text{start}} \le x_i \le x_{\text{goal}}$$
$$y_{\text{start}} \le y_i \le y_{\text{goal}}$$

F(x) is the objective function of the optimization problem, which is the path length, represents the abscissa and ordinate of a series of trajectory coordinate points moving in the map, that is, the decision variable of the optimization problem. Choose the decimal encoding method with a fixed length of n to represent a complete path, and use the path length as the evaluation factor to obtain the fitness function. The formula is as follows:

$$f(X,Y)=\sum_{i=1}^{n}\sqrt{(x_i-x_{i-1})^2-(y_i-y_{i-1})^2}$$

n is the number of cities included in the route, is the x-axis coordinate value of the i-th city, and is the y-axis coordinate value of the i-th city. The Hybrid Algorithm design process is shown in Figure 1.

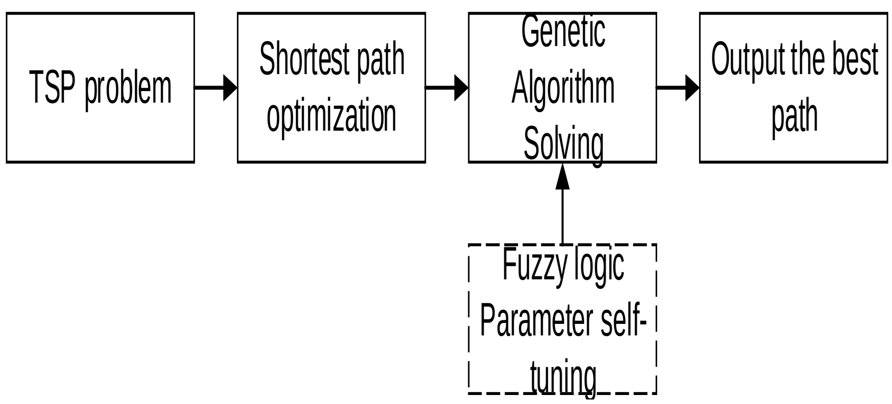

Figure 1 Hybrid Algorithm design process

4 Proposed Algorithm

The selection of the crossover probability pc and the mutation probability pm in the genetic algorithm will directly affect the evolution rate of the algorithm, the diversity of the population, and the quality of the final solution. Fuzzy logic control can transform expert knowledge and accumulated experience into clear control strategies in genetic evolution, making it possible to dynamically calculate the control parameters of genetic algorithms. In the proposed algorithm, a two-input and the two-output fuzzy logic controller is designed, and the rate of change between the average fitness of the previous generation population and the average fitness of the current generation population is used as the evaluation index ES(t) of the algorithm convergence speed. When ES(t) is negative, set ES(t) to 0.001, and when ES(t) is greater than 1, set ES(t) to 1.

The variance change rate D(t) of individual fitness in the population is used as an evaluation index to measure the diversity of the population. The information reflecting population evolution speed ES(t) and diversity D(t) are used as input variables of fuzzy logic control, and crossover probability pc and mutation probability pm are used as output variables to dynamically optimize the convergence speed and solution diversity of the algorithm.

4.1 Fuzzy output parameters

The crossover probability of the population in the genetic algorithm usually takes [0, 1], but in actual operation, if the value of the crossover probability is too low, it will lead to the stagnation of the population evolution and the inability to continuously generate individuals. Therefore, the output parameters pc and pm are set to use the triangle Type membership function, limit the value range of the crossover probability pc to [0.4, 1], and the value range of the mutation probability pm to [0, 0.4], as shown in Table 1.

Table 1 Crossover and mutation probability

	pc	pm
Lower	[0.4,0.6]	[0,0.04]
Low	[0.5,0.7]	[0.02,0.1]
Medium	[0.6,0.8]	[0.08,0.16]
High	[0.7,0.9]	[0.14,0.3]
Higher	[0.8,1.0]	[0.28,0.4]

4.2 Fuzzy input parameters

The values of the input parameters progress speed ES and fitness variance D are shown in Table 2.

Table 2 Values of progress speed and fitness variance

	pc	pm
Low	[0,0.1]	[0,0.8]
Medium	[0.08,0.25]	[0.7,1.5]
High	[0.2,1]	[1.5,2]

4.3 Fuzzy Reasoning Knowledge Base

For the output variable crossover probability pc and mutation probability pm, formulate the fuzzy inference rules shown in Table 3 and Table 4.

Table 3 pc fuzzy inference rules

ES	D		
	Low	Medium	High
Low	Higher	High	Medium
Medium	High	Medium	Low
High	Medium	Low	Lower

Table 4 pm fuzzy inference rules

ES	D		
	Low	Medium	High
Low	Higher	Medium	Low
Medium	High	Medium	Low
High	Medium	Low	Lower

Fuzzy inference rules can be understood as:

(a) When the evolution rate ES(t) of a certain generation population is small during the evolution process, the probability of crossover and mutation should be increased to promote evolution to produce more excellent individuals;

(b) When the evolution rate ES(t) is large, the crossover probability and mutation probability should be appropriately reduced to avoid the algorithm from falling into the local optimum;

(c) When the fitness variance D(t) of a certain generation population is small, the crossover probability should be increased to increase the diversity of the population and avoid premature maturity;

(d) When the fitness variance D(t) is large, it means that the individual differences in the population are large, and the crossover probability should be reduced to speed up the convergence of the algorithm;

(e) Especially, in the middle and late stages of population evolution, there may be evolution When both the speed ES(t) and the fitness variance D(t) are low, in order to avoid the prematureness of the algorithm, the output value of the crossover probability should be adjusted to the maximum to further enrich the chromosomes of the population.

4.4 TSP path planning process

The flow of Hybrid Algorithm of GA and Fuzzy Logic is shown in Figure 2.

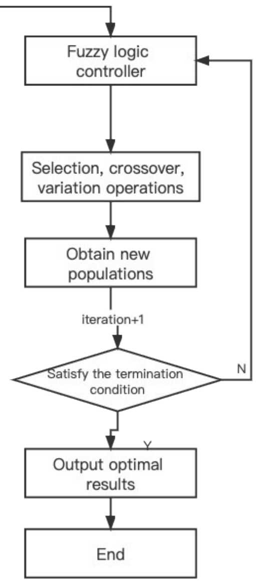

Figure 2 Path planning process

5 Experimental Setting

In full consideration of computing power and data set size, in addition to the required data sets of 4 countries, we also downloaded 2 simulated data sets XQF131 and XQG237 from the website to verify the effectiveness of the proposed algorithm. The proposed algorithm is implemented in Python language. The experimental memory size is 16GB, the processor is M1 chip CPU with 10 cores, the highest frequency is 3.35GHz, and the operating system is Mac OS X Lion.. The dataset statistics are shown in Table 5.

Table 5 Dataset Statistics

data set	points	best result	URL
XQF131	131	564	VLSI Data Sets (uwaterloo.ca)
XQG237	237	1019	
Greece	9882	300899	National Traveling Salesman Problems (uwaterloo.ca)
Japan	9847	491924	
Kazakhstan	9976	2061882	
Ireland	8246	206171	

The number of populations and iterations when the algorithm is running, as well as the crossover probability and mutation probability of GA are shown in Table 6. Except for whether to add fuzzy logic control, other parameter settings of GA and Hybrid Algorithm are completely consistent.

Table 6 Algorithm parameter settings

data set	The number of points actually used	total group number	iterations	GA_pm	GA_pc
XQF131	131	300	1000	0.5	0.1
XQG237	237	500	1000		
Greece	9882	2000	50		
Japan	50	300	1000		
Kazakhstan	100	500	1000		
Ireland	200	600	1000		

Due to the limited computing power, we adjusted the scale of the data set. In the case where the proposed fuzzy control logic can be verified to be effective, we used all the city points for the German data, but only used the city size of 2000 and 50 Obviously, this is far from enough to effectively solve the TSP problem of the current scale. We mainly explore whether fuzzy logic control plays a role.

After running for one day in this case, although the optimal solution cannot be found, we can get the iterative curve of the algorithm and the running process of whether the Hybrid Algorithm is better than GA. For the data from Japan, Kazakhstan, and Ireland, we selected 50, 100, and 200 city points for simulation experiments, and correspondingly used a relatively reasonable population size and the number of iterations to verify whether Fuzzy Logic works. For the two simulation data XQF131 and XQG237, we used all the data for experiments.

6 Experimental Results and Comparison Study

6.1 Sensitivity analysis of crossover rate parameters

In order to explore the impact of the crossover rate on the results of the genetic algorithm, we randomly generated data with city sizes of 10 and 20 to conduct crossover rate impact experiments. The crossover rate setting range is [0,1], the interval is 0.05, the population size is 150, 300, the number of iterations is 300, and the experimental results are shown in Figure 3.

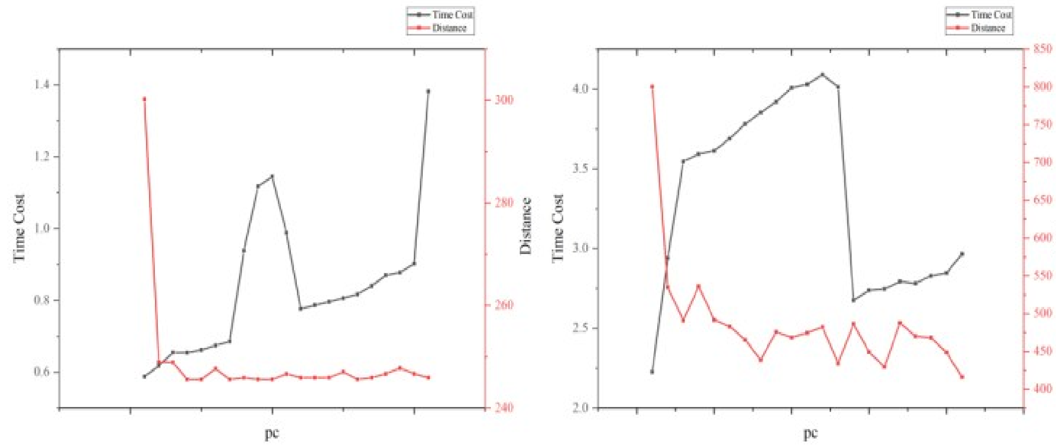

Figure 3 The crossover rate affects the experimental results (the scale is divided into data sets of 10 and 20)

6.2 Sensitivity analysis of variation rate parameters

In the same way, we also used the same parameters to conduct experiments on the mutation rate.

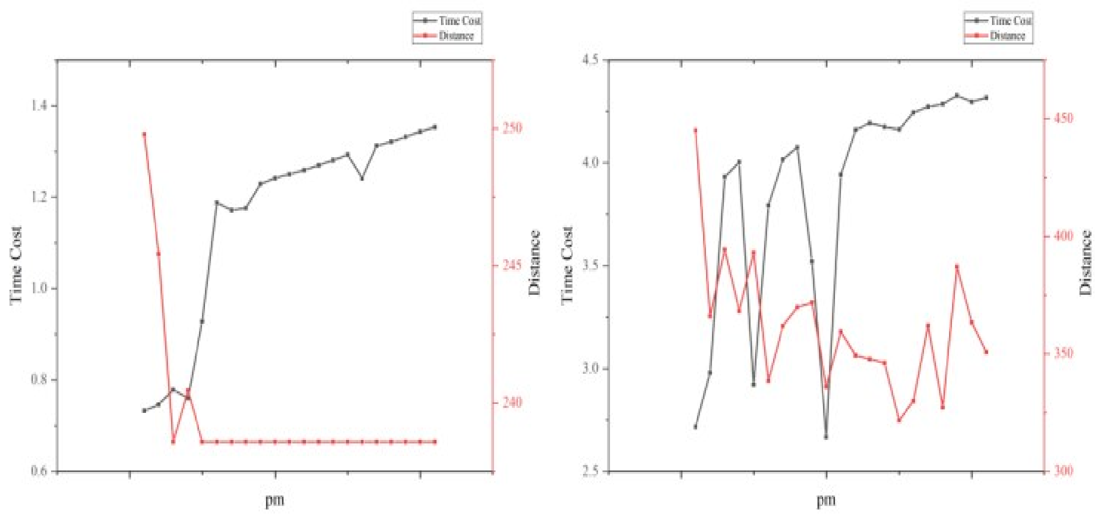

Figure 4 The mutation rate affects the experimental results (the scale is divided into data sets of 10 and 20)

From the experimental results, the crossover rate and mutation rate have an impact on the experimental results, that is, the performance of the genetic algorithm. Data sets of different scales show different trends in the experimental results, and the performance of the algorithm fluctuates. It is impossible to determine which specific value is better. On the other hand, as the crossover rate and mutation rate continue to increase, the running time of the algorithm basically shows an increasing trend. The determination of the crossover rate and mutation rate needs to comprehensively consider the time factor and the algorithm performance.

6.3 Compare the experimental results

In addition to the Greece data set, the two algorithms were run 10 times to take the average value. On different data sets, the results of GA and Hybrid Algorithm and the iteration curve of one operation are shown in Table 7 and Figure 5 respectively (the results are rounded).

Table 7 Comparative experiment results

data set	GA	Hybrid Algorithm
XQF131	1822	2013
XQG237	14894	13179
Japan	51334	46806
Kazakhstan	19033	18231
Ireland	34185	32430

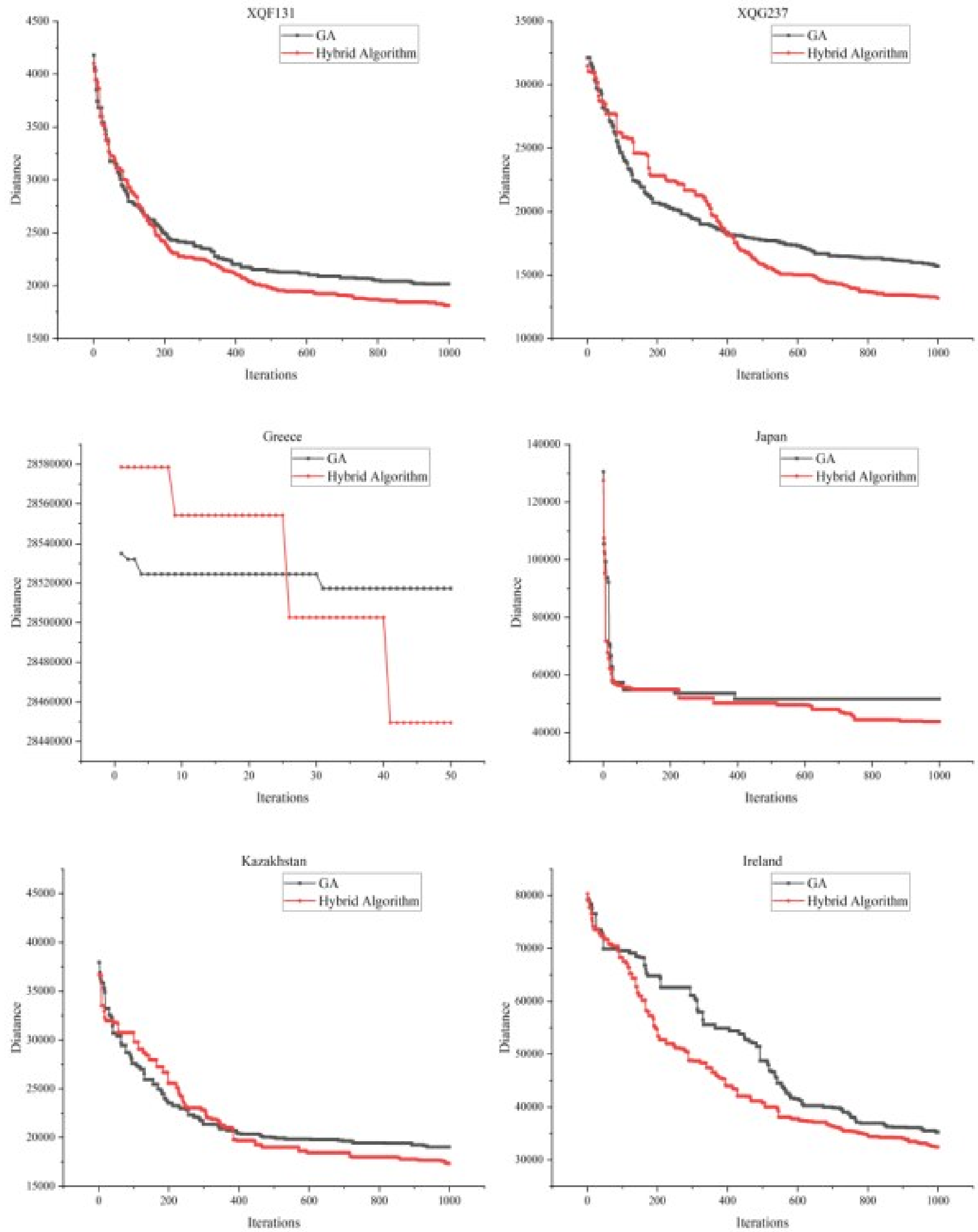

Figure 5 Iteration curves of different data sets

It can be seen from the chart that for most data sets, the basic algorithm GA is more likely to fall into the local optimal solution than the Hybrid Algorithm iteration, and then falls into multiple local optimal solutions, and does not search for the global optimal solution. Hybrid Algorithm of GA and Fuzzy Logic can dynamically adjust the crossover probability and mutation probability according to the evolution state of the population, which improves the evolution ability of the algorithm in the early stage and expands the global search range in the middle and late stages of evolution. , a better solution can be found.

Taking the Kazakhstan dataset as an example, Figure 6 records the change process of the crossover probability and mutation probability in the algorithm solution process respectively. It can be seen that the improved algorithm has a regulating effect on the crossover probability and mutation probability. When there is a risk, the fuzzy controller actively responds and adjusts the parameter values. The improved algorithm has strong self-regulation and self-adaptation capabilities.

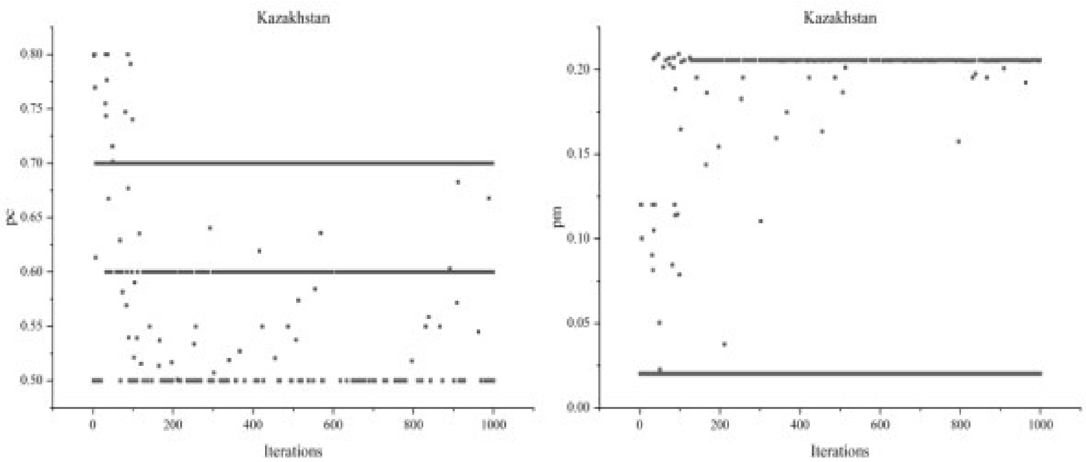

Figure 6 Changes of pc and pm

6.4 TSP optimization results

In order to show the effectiveness of the algorithm more clearly, we randomly generate data with a scale of 20, 40, and 60 to visually display the results of TSP optimization. The number of iterations and population size are set to 300, and other parameters are consistent with those in the comparison algorithm. . Similarly, except whether to introduce fuzzy control logic, other parameter settings are completely consistent between the two algorithms. The experimental results are shown in Table 8.

Table 8 Comparison of different scales randomly generated

scale	GA	Hybrid Algorithm
20	469	435
40	810	698
60	1349	1311

The corresponding paths and iteration curves are shown below.

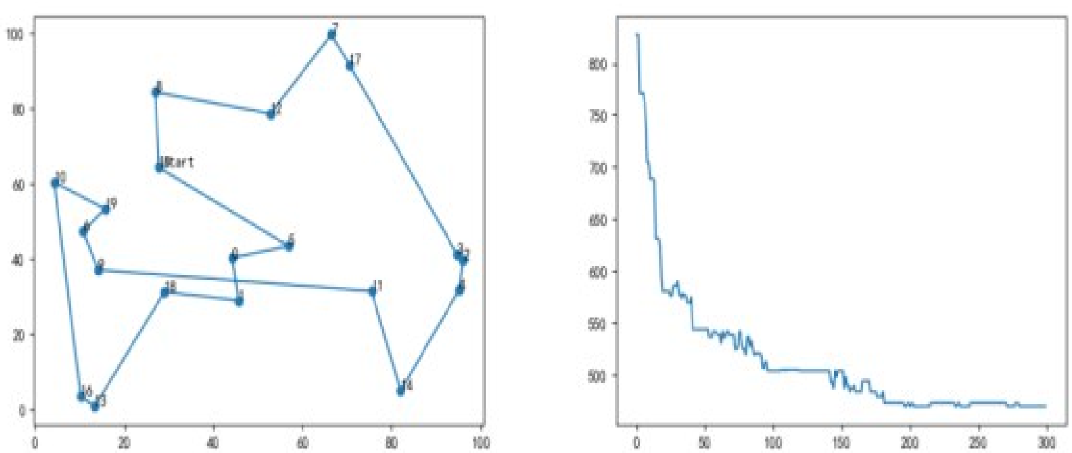

Figure 7 GA (scale 20)

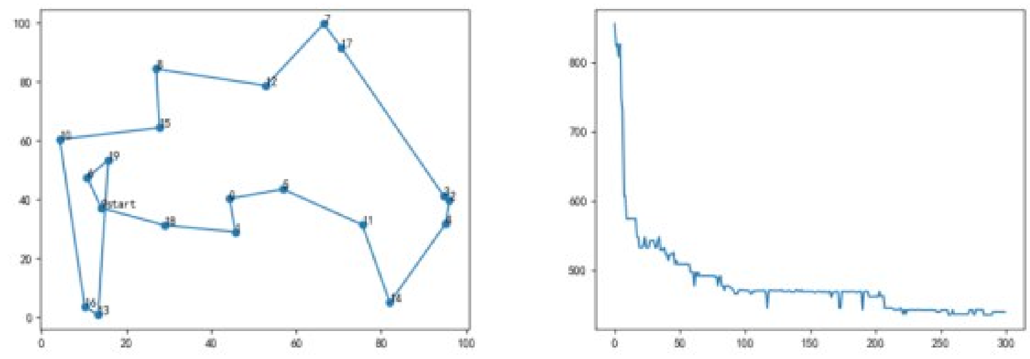

Figure 8 Hybrid Algorithm (scale is 20)

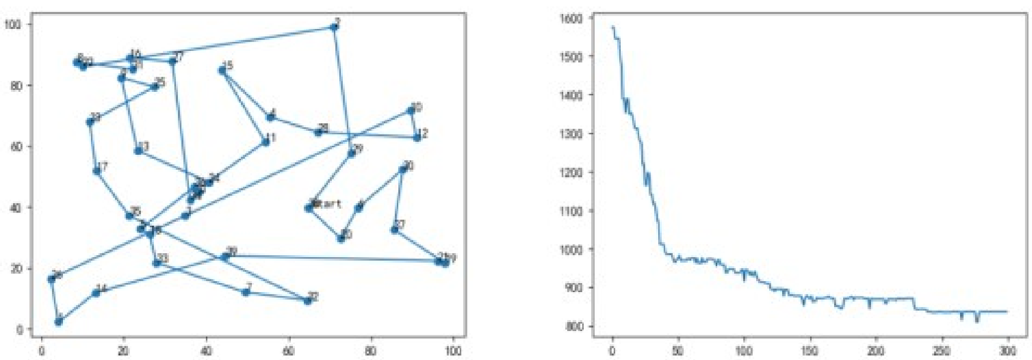

Figure 9 GA (scale 40)

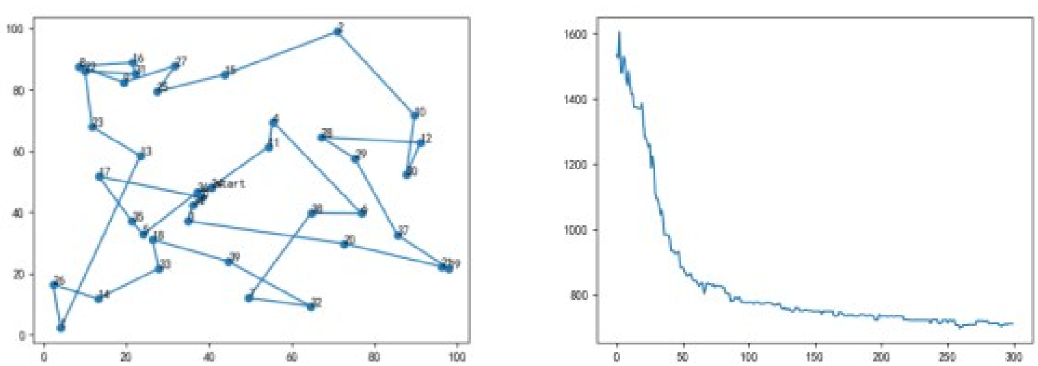

Figure 10 Hybrid Algorithm (scale is 40)

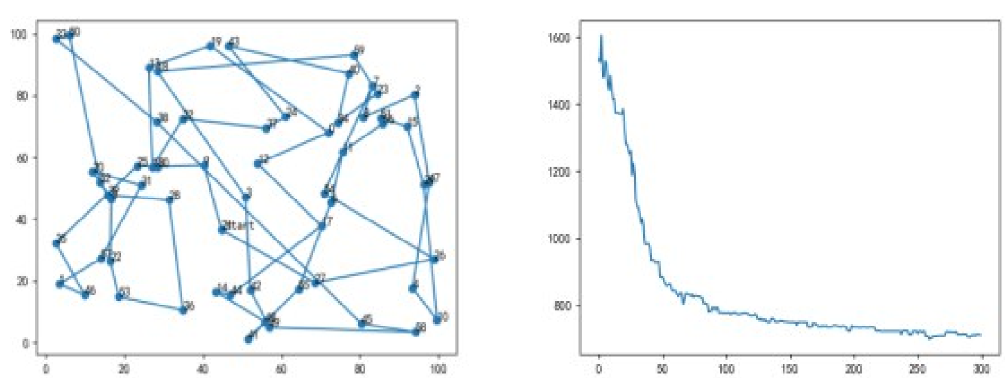

Figure 11 GA (scale 60)

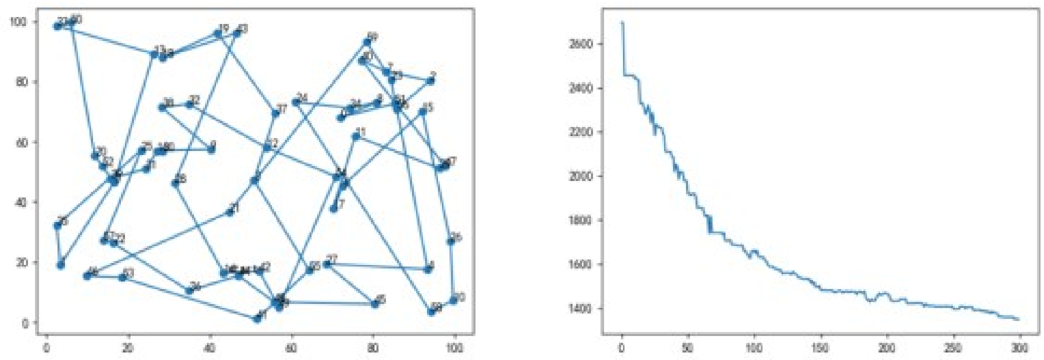

Figure 12 Hybrid Algorithm (scale is 60)

7 Conclusion and Future Work

For the genetic algorithm to solve the TSP problem, for a relatively large-scale problem, the algorithm can find an acceptable solution in a short period of time, but it is not necessarily the optimal solution, and with the increase of the scale, the size of the genetic algorithm population It also needs to be increased to match. The accuracy of genetic algorithm results will be affected by parameters such as crossover probability, mutation probability, number of iterations, and population size. Whether it is for GA or the Hybrid Algorithm of GA and Fuzzy Logic, increasing the number of populations will improve the accuracy of the algorithm results to a certain extent, but the running time is much longer than before. Both the crossover probability and the mutation probability will affect the final result. If the crossover probability is too low, it will be difficult to obtain the optimal solution. The higher the crossover probability, the better the average fitness, but it should not be too high, otherwise it will affect the iteration time. How to balance the relationship between time, result, and efficiency and find the optimal parameters is an important issue, which is very important for the performance of the genetic algorithm.

Fuzzy logic control is adopted to realize the dynamic adaptive adjustment of crossover probability and mutation probability in the process of population evolution, which expands the search range of the population, avoids premature algorithm and falling into local optimum, and improves the search efficiency of the algorithm and the search success of the optimal solution Rate. However, for the process of fuzzy logic control genetic algorithm parameter change, the design of its input is very important, whether ES and D can directly and correctly reflect the convergence rate and diversity of the algorithm is an important issue, we also try to find more suitable indicators to evaluate the running process of the algorithm. For some data sets, the genetic algorithm based on fuzzy control theory does not perform well. The possible reason is that on the one hand, the two evaluation indicators of ES and D selected may not fully reflect the performance of all data sets at runtime. On the other hand, Its range definition or membership function definition may not be applicable to the current dataset.

About the Author

Yuhan Sun, undergraduate reading, research direction: automation.

References

[1] Sorma, Ripon & Abdul Wadud, Md & Karim, S. M. & Ahamed, F. A. Sabbir. (2020). Solving Traveling Salesman Problem By Using Genetic Algorithm. 10.5923/j.eee.20201002.02.

[2] LAN L K,ZHU J W,LIU W L,et al.New method of fuzzybased genetic algorithms [J].Computer Engineering and Design,2008,29(14) : 3714-3718.

[3] El-Shazly, A., & Abdel-Salam, T. (2002). A hybrid genetic algorithm for solving engineering optimization problems. Engineering Optimization, 34(1), 1-18.

[4] WANG Jidai, WANG Xindong, TIAN Qunhong, et al. Mobile robot path planning based on improved fuzzy adaptive ge-netic algorithm[J]. Machine Tool&Hydraulics,2021,49(23):18-23.