Available online at www.sciencedirect.com

SciVerse ScienceDirect

Systems Engineering Procedia 4 (2012) 226 – 235

Procedia Systems Engineering

The 2nd International Conference on Complexity Science & Information Engineering

Comparison of several intelligent algorithms for solving TSP problem in industrial engineering

Wang Hui[a]

North China Electric Power University No.2 Beinong Rd., Huilongguan, Beijing, 102206, P.R.China

Abstract

The paper presents three intelligent algorithms, namely, basic genetic algorithm, Hopfield neural network and basic ant colony algorithm to solve the TSP problem. Then different algorithms are compared in the perspectives of time complexity, space complexity, the advantages and disadvantages of the calculation results, and difficulty level of realization. We use the application of paired comparison matrix to make comprehensive evaluation, and then give the value of comprehensive evaluation in engineering.

Keywords: basic genetic algorithm, Hopfield neural network, ant colony optimization, paired comparison matrix, TSP

1. Introduction

The travelling salesman problem (TSP) is a problem in combinatorial optimization studied in operations research and theoretical computer science and in engineering. Given a list of cities and their pair wise distances, the task is to find a shortest possible tour that visits each city exactly once.This is an NP-hard problem, when a large number of nodes of G, if the use exhaustive search, the time complexity is $O(n!)$, if use the search of dynamic programming, the time complexity is $O(n^2 2^2)$, combinatorial explosion will occur in the both search methods. Therefore, the majority of domestic and foreign scholars began to study intelligence algorithms for TSP, since the basic genetic algorithm appears, they began to examine the use of genetic algorithm on solve TSP problems until present and proposed many improvements. Reference [1] presents a genetic algorithm based on common path, reference [2] proposed a new genetic algorithm through using multiple-search method. All these improved genetic algorithms are promising approach for TSP problem. Hopfield network was proposed in 1970s, and in 1985, Hopfield proposed to use CHNN for solving TSP problems, but Hopfield network prone to ineffective solutions and local solutions, so many scholars have been studying how to improve the algorithm, reference [3] analyzed the effectiveness of solving TSP with Hopfield, reference [4] through optimizing the Hopfield network and path of the initial steps to improve the Hopfield network to solve TSP and received good results. Ant colony algorithm which is effectiveness proposed a new computational intelligence algorithm for solving TSP problems recently. Because of its use of pheromone heuristic function, can greatly reduce the search

* Corresponding author. Tel.+086-010-51963765; fax: +086-010-51963765.
E-mail address: wanghuibd@163.com.

2211-3819 © 2011 Published by Elsevier Ltd. Selection and peer-review under responsibility of Desheng Dash Wu.
doi:10.1016/j.sepro.2011.11.070

space, so compared to other algorithms it have better time performance, but precisely because of this, ant colony algorithm is easy to fall into local optimal solution, reference [5] by dynamically adjusting the pheromone volatility to present a dynamic ant colony algorithm and get a more satisfactory results. All of these three algorithms for solving TSP problems have advantages and disadvantages. In this paper, these three kinds of algorithms are compared in time complexity, space complexity, the advantages and disadvantages of the calculation results, and difficulty level of realization, etc. We apply paired comparison matrix to make comprehensive evaluation, and then give the value of comprehensive evaluation.

2. The basic steps of three algorithms

2.1. The application of genetic algorithm for TSP

This paper uses a common framework of genetic algorithm to solve TSP. this section gives the general framework of genetic algorithm, then given the steps of genetic operators' algorithm.

Common framework for basic genetic algorithm

```
STEP0 Q=generateinitialpopulation( )；// Initialize population Q
STEP1 F=calculateobjectfitness(Q)； // calculated fitness(F) of population Q
STEP2 FOR i=1 to T BEGIN  // T is the iteration step
STEP3  selectoperator(Q,F)； // Selection operator
STEP4  crossoveroperator(Q,pc)； // crossover operator
STEP5  mutationoperator(Q,F,pm)；// mutation operator
STEP6  F=calculateobjectfitness(Q)；// calculated fitness(F) of population Q
STEP7 END
```

2.1.1. Encoding and decoding

This paper is using the decimal coding to encoding for the path. For example, a chromosome 193,264,785 that represent a path, the path from the starting city 1, followed city 9,3,2,6,4,7,8,5 and finish return to the city 1.

2.1.2. Fitness function

Fitness function is calculated by the reciprocal of the path distance, that is, the longer the path the smaller the fitness, and vice versa. Formula is: $f = \frac{1}{S}$, while $S = \sum_{i}^{n} d_{i,i+1}$, It represents path length.

2.1.3. Selection operator

We use classic roulette wheel method to select the operator. First we calculate the fitness of each individual. Then calculate the probability of individual to be selected and use roulette method to choice the next generation of individual.

2.1.4. Crossover operator

(1) Select two individuals which were recorded as X and Y from the population.