Travelling Salesman Problem in Uncertain Environments

Ma Huiru[1,2,*], Jia Limin[1], Zhang Xingchen[2], Miao Jianrui[1] and Sun Jiandong[1]

[1]*State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, 100044, P.R. China;*
[2]*School of Traffic and Transportation, Beijing Jiaotong University, Beijing, 100044, P.R. China*

Abstract: In practice, due to the lack of information, imprecise variables which come from experts' empirical data usually appear. In order to deal with these imprecise variables, uncertainty theory is proposed and has been proved to be an efficient method. This paper introduces uncertainty theory into travelling salesman problem (TSP), in which the link travel times are assumed to be uncertain variables, and then a chance constrained programming model is proposed within the framework of uncertainty theory. The properties of the chance constrained programming model are investigated; furthermore, the uncertain model is proved to be equivalent to a deterministic model. To solve the problem, we design an algorithm based on genetic algorithm. Finally, a numerical example is given, the result of which verifies the effectiveness of the proposed chance constrained programming model and the algorithm.

1. INTRODUCTION

Travelling salesman problem (TSP) is a typical problem of Combinatorial Optimization, which can date back to Knight travel in the 18th century. Let N be the number of cities. TSP can be described as a scene that a salesman who goes to N cities to sell goods, starts at one city and visits each city (exactly once) in turn, finally goes back to the starting point. The task is to find the shortest possible path during travelling salesman iterate through all the cities. TSP is widely used in practical problems such as logistics distribution, vehicle routing, vehicle scheduling, pipe laying, robot control, circuit board drilling and so on. It is also the routing problem's research foundation of computer network, railway transportation, logistics and other fields, so it has become a burning issue which has been drawn tremendous attentionsby the researchers [1, 2].

TSP is easy to state, but is notoriously hard to solve. Garey & Johnson has proved this to be a non-deterministic polynomial-time hard (NP-hard) problem in combinatorial optimization [3]. Its research of computational complexity lays the foundation of NP complete theory and is a scale of testing the new algorithm. Currently, algorithm for TSP is mainly divided into two categories: exact algorithms and heuristics algorithms. The exact algorithms can get all solutions in the solution space, but computational complexity will grow exponentially when the number of cities N increases and network size reaches a certain level [4]. Heuristics algorithms cannot get all solutions in the solution space and can only find global optimal solution or approximate solution, but with the increase of the number of the city N, it cannot cause combinatorial explosion. Heuristics algorithms included greedy algorithm, tabu search algorithm, simulated annealing, neural network, ant colony algorithm, genetic algorithm, particle swarm optimization algorithm and hybrid intelligent algorithm [5], of which genetic algorithm (GA) is an adaptive method of search for global optimization with high efficiency and robustness [6]. Therefore, this paper adopts the GA to solve the TSP.

In the traditional TSP, travel time between cities is a deterministic numerical value. In the optimization problem, due to the lack of information and outside interference, decision-makers often encounter uncertain parameters. For example, the travel time cannot be expressed as an exact value in many cases. If there is more adequate historical data as a reference, we can get probability distribution function of non-deterministic parameters by the probability statistics method, and set up a stochastic programming model to solve the problem. In many cases of processing non-deterministic variables, we neither have historical data to refer, nor carry out measurements. We can only use expertise to estimate the distribution of non-deterministic variables. Characterizing imprecise variations which come from experts' empirical data from the angle of math, Liu proposed and improved uncertainty theory in many writings [7, 8]. Liu systematically studied mathematical programming problems in the uncertainty environment and developed the uncertain chance constrained programming model [9]. Many researchers also studied the problem of uncertain programming, and promote the development of the theory and application of uncertain programming [10-12].

This paper introduces uncertainty theory into TSP, assuming that the travel time of salesman visiting different cities is uncertain variables. We set the total travel time as chance constraints at a certain level of opportunity. We also construct the chance-constrained programming model of travelling salesman problem under uncertain environment and design genetic algorithm to solve. Finally, a numerical

example is given to verify that the algorithm proposed in this paper is effective.

2. THE CHANCE CONSTRAINED PROGRAMMING MODEL OF TSPNDER UNCERTAIN ENVIRON-MENTS

2.1. Uncertainty Theory Overview

This section provides a brief overview of the concepts involved uncertainty theory to ensure the integrity of the paper. The following definitions and theorems are referred from Liu [7, 8].

Definition 2.1 Let Γ be a nonempty set, and let $\mathcal{L}$ be an σ-algebra over Γ. Each element Λ in the σ-algebra $\mathcal{L}$ is called an event. If a set function M from $\mathcal{L}$ to the set of the real numbers $\mathcal{R}$ satisfies the following conditions:

Axiom 1. (Normality) $M(\Gamma) = 1$ for the universal set Γ.

Axiom 2. (Duality) $M\{\Lambda\} + M\{\Lambda^c\} = 1$ for any even Λ.

Axiom 3. (Subadditivity) For every countable sequence of events $\Lambda_1, \Lambda_2, ...$, we have

$$M\left\{\bigcup_{i=1}^{\infty} \Lambda_i\right\} \leq \sum_{i=1}^{\infty} M\{\Lambda_i\}. \tag{1}$$

Then, M is called an uncertain measure. The triple $(\Gamma, \mathcal{L}, M)$ is called an uncertainty space.

In order to study the product uncertain measure, Liu [9] gives the axiom 4:

Axiom 4: (Product Axiom) let $(\Gamma_i, \mathcal{L}_i, M_i)$, $i = 1,2, ...$ be uncertainty spaces. The product uncertain M is an uncertain measure satisfying

$$M\left\{\prod_{i=1}^{\infty} \Lambda_i\right\} = \sup_{i} M\{\Lambda_i\}, \tag{2}$$

where event $\Lambda_i \in \mathcal{L}_i, i = 1,2, ...$

Definition 2.2 An uncertain variable is a measurable function from an uncertainty space $(\Gamma, \mathcal{L}, M)$ to the set $\mathcal{R}$ of the real numbers. For any Borel set B of real numbers, the set $\{\xi \in B\} = \{\gamma \in \Gamma | \xi(\gamma) \in B\}$ is an event of $\mathcal{L}$.

Definition 2.3 The uncertainty distribution Φ of an uncertain variables ξ is defined by

$$\Phi(x) = M\{\xi \leq x\}, \forall x \in \mathcal{R}. \tag{3}$$

Definition 2.4 An uncertainty distribution Φ is said to be regular if its inverse function $\Phi^{-1}(\alpha)$ exists and is unique for each $\alpha \in (0,1)$. Then the inverse function $\Phi^{-1}(\alpha)$ is called the inverse uncertainty distribution of ξ.

The following theorem shows that regular uncertain variables have some good properties in the calculation. Many non-regular uncertain variables become regular by exerting the disturbance, which will greatly simplify operation between variables.

Theorem 2.1 Let $\xi_1, \xi_2, ..., \xi_n$ be independent regular uncertain variables with uncertainty distributions $\Phi_1, \Phi_2, ..., \Phi_n$, respectively, then $\xi = g(\xi_1, \xi_2, ..., \xi_n)$ is called regular uncertain variables. If $g(x_1, x_2, ..., x_n)$ is a strictly increasing about $x_1, x_2, ..., x_n$, then inverse uncertainty distribution of ξ is

$$\Psi^{-1}(a) = g\left(\Phi_1^{-1}(\alpha), \Phi_2^{-1}(\alpha), ..., \Phi_n^{-1}(\alpha)\right), \forall \alpha \in (0,1). \tag{4}$$

The strictly increasing function is defined as:

$g(x_1, x_2, ..., x_n)$ is a n-ary real valued functions. When $a_i \leq b_i$, $i = 1, ..., n$, then $f(a_1, a_2, ..., a_n) \leq f(b_1, b_2, ..., b_n)$; when $a_i < b_i$, $i = 1, ..., n$, then $f(a_1, a_2, ..., a_n) < f(b_1, b_2, ..., b_n)$.

The expected value of an uncertain variable ξ is defined by

$$E[\xi] = \int_0^{+\infty} M\{\xi \geq r\}dr - \int_{-\infty}^0 M\{\xi \leq r\}dr \tag{5}$$

An uncertain variable ξ is called normal if it has a normal uncertain distribution

$$\Phi(x) = \left(1 + exp\left(\frac{\pi(e - x)}{\sqrt{3}\sigma}\right)\right)^{-1}, \quad x \in \mathcal{R}, \tag{6}$$

Denoted by $\xi = N(e, \sigma)$ where e and σ are real numbers with $\sigma > 0$.

Fig. (**1**) Shows the uncertainty distribution of normal uncertain variables.
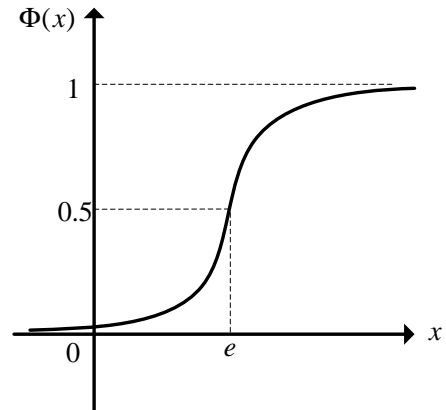

Fig. (1). Normal uncertainty distribution.

Obviously, $\xi = N(e, \sigma)$ is a regular uncertain variable. Its inverse distribution function is

$$\Phi^{-1}(\alpha) = e + \frac{\sigma\sqrt{3}}{\pi} ln \frac{\alpha}{1 - \alpha} \tag{7}$$

and its expected value is $E[\xi] = e$.

2.2. The Chance-constrained Programming Model of Travelling Salesman Problem

Assume a businessman wants to go to n cities A = $(a_1, a_2, ..., a_n)$ to sell commodities. He wants to select a route to visit each city (only once), so that he can go back to the starting point and hopes that the cost time is the least. Let x_{ij} represents whether salesman visits the city a_i and then visit the city a_j according to the order ($x_{ij} = 1$, represents salesman visits the city a_i and then visit the city a_j according to the order, otherwise $x_{ij} = 0$). In the classic TSP, the businessman's travelling time from city a_i to city a_j is determined value.

TSP can be built as the following mathematical model:

$$(a)\begin{cases} \min f(x,t) = \sum_{i,j} x_{ij} t_{ij} \\ s.t. \\ \sum_{j=1}^{n} x_{ij} = 1, \quad i = 1,2,\dots,n \quad (i) \\ \sum_{i=1}^{n} x_{ij} = 1, \quad j = 1,2,\dots,n \quad (ii) \\ \sum_{i=1}^{n} \sum_{j=1,j\neq i}^{n} x_{ij} = 1, \quad (iii) \end{cases} \quad (8)$$

where $X = (x_{ij})$, $t = (t_{ij})$.

Let objective function

$$f(x,t) = \sum_{i,j} x_{ij} t_{ij} \quad (9)$$

be total travel time. Constraints (i), (ii) and (iii) represent a route to make salesman to go back to the beginning after he visited each city. Obviously, it is an increasing function about variable t. With the travel time between cities increases, the total travel time will increases. In practice, travel time between cities is imprecise variables. This paper adopts uncertain variables to represent travel time from city a_i to citya_j. Then objective function is

$$f(x,\xi) = \sum_{i,j} x_{ij} \xi_{ij}. \quad (10)$$

where $f(x,\xi)$ is an uncertain variable, which can't simply judge the pros and cons of solution by the size of the value.

To build such model with non-precision variables, Charnes & Cooper propose chance-constrained model [13]. Liu further applies the chance constrained model to uncertain programming [7]. At the same time, he studies the equivalence class transformation of uncertain chance-constrained model, combined with properties of uncertain variables. Based on TSP, this paper proposed the following uncertainties chance constrained programming model:

$$(b)\begin{cases} \min T \\ s.t. \\ M\{f(x,\xi) \leq T\} \geq \alpha \quad (iv) \\ \sum_{j=1}^{n} x_{ij} = 1, \quad i = 1,2,\dots,n \quad (i) \\ \sum_{i=1}^{n} x_{ij} = 1, \quad j = 1,2,\dots,n \quad (ii) \\ \sum_{i=1}^{n} \sum_{j=1,j\neq i}^{n} x_{ij} = 1. \quad (iii) \end{cases} \quad (11)$$

Constraints (iv) is chance constraint, which requires the uncertainty measure of total travel time more than in the given level of opportunity condition.

Although the chance-constrained model effectively portrays the TSP under uncertainty environment, the solution is difficult due to the constraints with an uncertain chance. In the next section, this paper will study equivalence class conversion of uncertain model.

2.3. Equivalent Conversion Of Uncertain Chance-constrained Programming Model

Theorem 2.2 If the travel time ξ_{ij} between cities is regular uncertain variables in the TSP and is full of uncertainty distribution functionΦ_{ij}, equivalent conversion of uncertain chance-constrained programming model is the following deterministic model:

$$(c)\begin{cases} \min T \\ s.t. \\ \sum_{i,j} x_{ij} \Phi_{ij}^{-1}(\alpha) \leq T \quad (v) \\ \sum_{j=1}^{n} x_{ij} = 1 \quad i = 1,2,\dots,n \quad (i) \\ \sum_{i=1}^{n} x_{ij} = 1 \quad j = 1,2,\dots,n \quad (ii) \\ \sum_{i=1}^{n} \sum_{j=1,j\neq i}^{n} x_{ij} = 1. \quad (iii) \end{cases} \quad (12)$$

Proof: In the uncertain chance-constrained programming model, we conduct analysis of equivalence classes for constraint (iv).

At first, $f(x,\xi)$ is an uncertain regular variables, so its distribution function and inverse distribution functions are ψ_f, ψ_f^{-1}, respectively. In terms of a given confidence level, we can find a number $T_f(\alpha)$ that satisfies the following conditions:

$$M\{f(x,\xi) \leq T_f(\alpha)\} = \alpha. \quad (13)$$

On the other hand, if $T_f(\alpha)$ is not a large number, the uncertain measure of$f(x,\xi) \leq T_f(\alpha)$ will increase. Hence, we can know from the definition of regular uncertainty distribution

$$M\{g_j(x,\xi) \leq 0\} \geq \Phi\left(T_f(\alpha)\right) = M\{f(x,\xi) \leq T_f(\alpha)\} = \alpha. \quad (14)$$

We can deduce that $T_f(\alpha) = \Phi_f^{-1}(\alpha)$, because equation (10), the equivalence of conditions (iv) is

$$\psi_f(T) = M\{f(x,\xi) \leq T\} \geq \alpha \Rightarrow T_f(\alpha) \leq T$$

$$\Rightarrow \psi_f^{-1}(\alpha) \leq T. \quad (15)$$

According to Theorem 2.1,

$$\psi_f^{-1}(\alpha) = \sum_{i,j} x_{ij} \Phi_{ij}^{-1}(\alpha) \Rightarrow \sum_{i,j} x_{ij} \Phi_{ij}^{-1}(\alpha) \leq T, \quad (16)$$

So the chance constraints (iv) can be converted to constraint (v) equivalently. Theorem 2.2 shows that solving uncertain chance-constrained programming model (b) of TSP is equal to solve a deterministic model. This paper will adopt GA to obtain optimal solutions.

3. ALGORITHM

GA is a kind of random search algorithm based on the biological natural selection and genetic mechanism. In general GA begins with a group of randomly generated feasible solution and each solution use a string of coding to represented the individual. Fitness deter-mined by the optimization objective function evaluates individuals. The populations are combined to produce new individuals through the cross and mutation operator, and gradually make the optimized population evolution [14].

In order to solve the problem, the corresponding GA is designed. The basic steps of solving TSP are as follows:

(1)   Generation of urban coordinate points.

We will put coordinate points in the two-dimensional co-ordinate of the city ,and it can generated 200 set of coordinate points randomly to ensure the effectiveness of algorithms.

(2)   Determine the initial population

Population is made of multiple chromosomes. We randomly sort coordinate points of the search space, and make every possible path generated as a chromosome. The size of the initial population is 200 in the algorithm.

(3)   Calculation of fitness function

We use the inverse of an objective function that is the inverse of the total time used to compare account for fitness function of each chromosome.

(4)   Selection operation

We assign larger fitness of chromosome to new species and optimize the quality of chromosomes in the population. In the algorithm we use simulated roulette wheel operation to determine whether each individual will genetic to the next generation of population.

(5)   Crossover operation

We can regard adjacent two chromosomes as parental, section randomly crossover. Comparing the intersection distance of gene with the front between two parental chose, the shorter is the new chromosome genes. Finally, we see last gene of new chromosome as the initial gene of the two parental generations, in turn generate new chromosome.

(6)   Mutation operation

For each chromosome, select randomly mutation point, compare the distance between that point and the subsequent, select gene of the shortest distance, reverse mutation segment gene.

(7)   Specified number of iterations, output the best individual, terminate the entire program.

4. ALGORITHM EXAMPLES

To build chance-constrained programming model for the TSP under uncertain environment, we use the Matlab software to write GA codes. We take 200 cities for example to verify the accuracy of optimization models and algorithms. City coordinates are shown in Fig. (**2**).
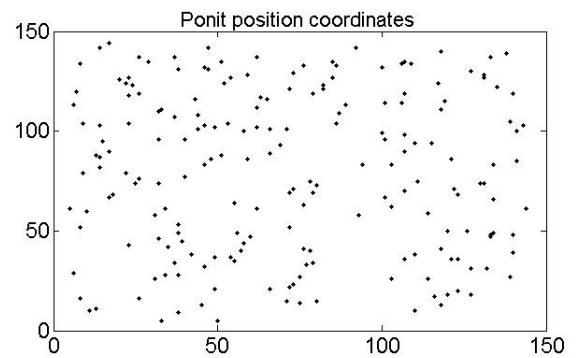

Fig. (2). Point position coordinates.

The GA defines that initial population size is 200, cross probability is 0.3, mutation probability is 0.3, maximal iteration is 200. For convenience, we assume that the cost time of salesman walking on the path is uncertain variables $\xi_{ij} = N\ (1, 0.25)$, and the chance level is α=95%. We can get optimal value of chance-constrained model $T = 1699.82$ by calculating. Optimal path is shown in Fig. (**3**).
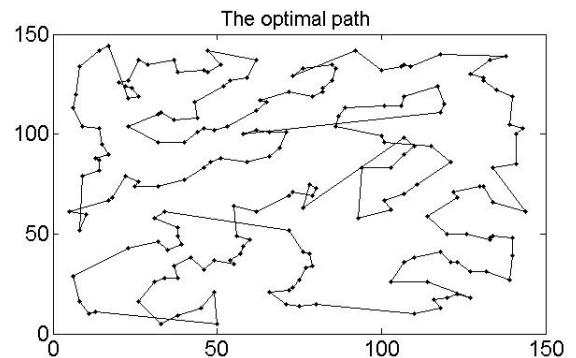

Fig. (3). The optimal path.

By adopting the strategies in Fig. (**3**), we can see that the chance of salesman's total travel time less than $T= 1699.82$ is at least 95%.

Average fitness of solutions in the genetic process is presented in Fig. (**4**). The average fitness of the chromosome has improved significantly during genetic process from one generation to 50 generations. At last, the average fitness of chromosome has a small -scale fluctuations.
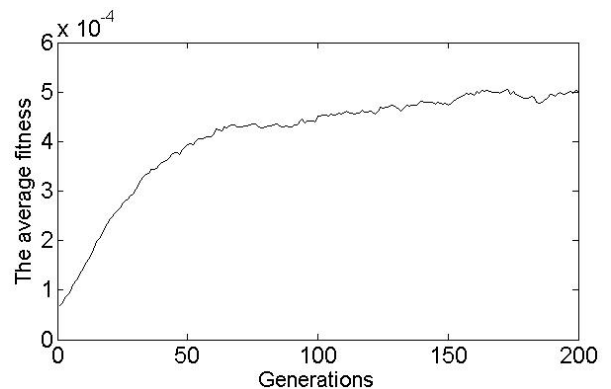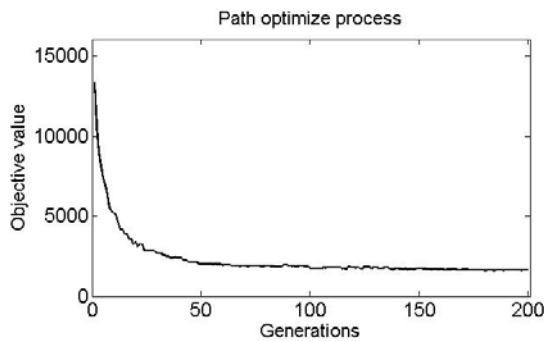

Fig. (4). The average fitness.

Fig. (5). Path optimization process.

The path optimization process is shown in Fig. (**5**). From Fig. (**5**) we can found that optimal solution will gradually tend to smooth after 100 generations in an iterative, which means that the algorithm has a certain robustness. The experimental results show that the proposed model is effective for solving the TSP in uncertain environments.

5. CONCLUSION

(1) Assuming uncertain variable is the travelling time of salesman visits different cities, optimization goal is the shortest travel time. It constructs chance-constrained model based on the theory of uncertainty and transforms the uncertainty model which is into an equivalent deterministic model.

(2) We design GA and write algorithm code and take a scale of cities for example to simulate. The results of theoretical analysis and simulation show that the algorithm can effectively get the optimal travel route by giving chance constrain.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

REFERENCES

[1]   D. Applegate, R. Bixby, V. Chvátaland, and W. Cook, "Implementing the Dantzig-Fulkerman-Johnson algorithm for large travelling salesman problems", *Mathematical Programming*, vol. 97, no. 1, pp. 91-98, 2003.

[2]   H.R. Zhou, W.S. Tangand, and H.L. Wang, "Optimization of multiple travelling salesman problem based on differential evolution algorithm", *Systems Engineering-Theory & Practice*, vol. 30, no. 8, pp. 1471-1476, 2010.

[3]   M.R. Garey and D.S. Johnson, *Computers and intractability: A guide to the theory of NP-Hardness*. W. H. Freeman, New York, pp. 85-126, 1979.

[4]   R. Germs, B. Goldengorin, and M. Turkensteen, "Lower tolerance-based Branch and Bound algorithms for the ATSP", *Computers and Operations Research*, vol. 39, no. 2, pp. 291-298, 2012.

[5]   W.Z. Rao, C. Jinand, and Y.Y. Huang, "Hybrid algorithm of the nearest neighbor and insertion for the travelling salesman problem ", *Systems Engineering-Theory & Practice*, vol. 31, no. 8, pp. 1419-1428, 2011.

[6]   C. Moon, J. Kim, G. Choi, and Y. Seo, "An efficient genetic algorithm for the travelling salesman problem with precedence constraints", *European Journal of Operational Research*, vol. 140, no. 3, pp. 606-617, 2002.

[7]   B. Liu, *Uncertainty Theory*. 2nded., Springer-Verlag, Berlin, 2007.

[8]   B. Liu, *Uncertainty Theory: A Branch of Mathematics for Modeling Human Uncertainty*. Springer-Verlag, Berlin, 2010.

[9]   B. Liu, *Theory and practice of uncertain programming*, 2nded., Springer-Verlag, Berlin, 2009.

[10]   Y. Gao, "Shortest path problem with uncertain arc lengths", *Computers and Mathematics with Applications*, vol. 62, no. 6, pp. 2591-2600, 2011.

[11]   B. Zhang and J. Peng, "Uncertain programming model for Chinese postman problem with uncertain weights", *Industrial Engineering & Management Systems*, vol. 11, no. 1, pp. 18-25, 2012.

[12]   S.W. Han, Z.X. Peng and S.Q. Wang, "The maximum flow problem of uncertain network, Information Sciences", *Information Sciences*, vol. 265, pp. 167-175, 2014.

[13]   A. Charnes and W.W. Coope, "Chance constrained programming", *Management Science*, vol. 6, no. 1, pp. 73-79, 1959.

[14]   J. Tian, J.S. Kou and M.Q. Li, "Genetic algorithms' application in operation research-status quo and prospect", *Journal of Systems Engineering*, vol. 13, no. 2, pp. 101-106, 1998.