An Efficient Method on a Polynomial Time Evolution Algorithm for the Traveling Salesman Problem

Jian-wu Dang, Xiao-juan Liu, Zhen-hai Zhang

College Of Information and Electrical Engineering,
Lanzhou Jiaotong University, Lanzhou 730070

dangjw@mail.lzjtu.cn

Abstract

A genetic algorithm simulating evolution is proposed to yield near - optional solution to the Traveling Salesman Problem. Noting that Darwinian Evolution is itself optimization process, we propose a heuristic algorithm that incorporates the natural selection. The time complexity of this algorithm is equivalent to the fastest sorting scheme. The algorithm is used to solve the China - Traveling Salesman Problem, the shortest route is obtained in this paper. The adaptability of this technique to tackle other NP -complete problems such as the tree problem is also discussed.

Keyword: neural network; evolution; time complexity; traveling salesman problem

I. Introduction

The classic Traveling Salesman Problem (TSP) has been studied extensively in the past and regarded as a simple benchmark problem which is difficult to solve. Given n cities the objective is to visit each city exactly once, returning to the starting point, minimizing the total distance. Since the number of distinct paths, $n!/2n$ (there are $n!$ permutations of the n cities); the clockwise path is equivalent to the counter - clockwise one; The problem is, in principle, solvable by exhaustive search of the solution space. This brute force algorithm is highly impractical because its computing effort is $O(n!)$.

In fact, there is no known algorithm for determining the optimal path whose computational effort is bounded by any power of n. This lack of any polynomial time algorithm is characteristic of the diverse class of NP -complete (non - deterministic polynomial time complete) problem, of which the TSP is a classical example. For deeply understood reasons it is believed that such polynomial time algorithm exists.

Supported by ‘Qing Lan’Talent Engineering Funds by Lanzhou Jiaotong University

Genetic algorithms have been proposed to simulated evolutionary optimization through iterative mutation and selection. We propose a robust iterative improvement algorithm that simulates

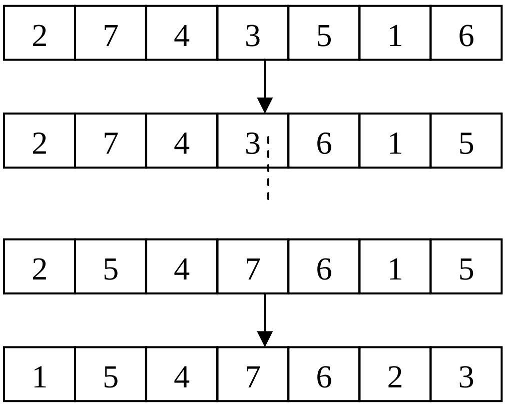
Darwinian evolution whose computation effort is $O(n \log n)$, no more than the effort of the fastest sorting scheme, and yields solutions that are typically less than 25% longer than the "expected" optimal tour. Selection of improved variants from the population that emerge as a resu1t of mutation is held to be the central principle underlying optimal evolution is clear when we observe that natural selection as formulated by Darwin is itself an optimization process whose objective is to foster the survival of fit species while inhibiting that of inferior ones.

II. Algorithm

Consider an n cities TSP. A feasible solution (path) is represented by an array of n distant elements of the set $\{1,2,...,n\}$. For example, the following array:

2	7	4	3	5	1	6

Represented a tour from $2 \to 7 \to 4 \to ... \to 6 \to 2$. Each such array is thought of as an organism. Each such organism will posses a score that represents the tota1 distance of its paths.
An initial popu1ation of NP organisms is randomly generated and each associated score is computed. From these NP, a survival set containing the organisms with the best N scores are chosen to reproduce. Reproduction occurs by repeated point mutation (exchanging 2 elements) as illustrated below:



Reproduction is as follows:
do exchange 2 elements
whi1e ($X < P$)
where X is a uniformly distributed random variable on (O) and $P \in$ (0, 1) determines the expected number of exchanges per reproduction. This variable of genetic drift is denoted by $E_{gd}(X)$, where X = 1,2, is a discrete variable. We claim:

Lemma 1 $P_x(m) = P^{m-1}(1-P)$.

Proof $P_x(m)$ denotes the probability that exactly m exchanges are performed. Obviously at least 1 exchange is preformed. To perform exactly m exchanges the loop must be repeated $m-1$times and stopped at the $m-th$ repetition. Each loop repetition occurs with a probability of P, while a loop stoppage occurs with a probability of $1-P$.

Lemma 2 $E_{gd}(X) = (1-p)^{-1}$ and $\sigma_{gd}^2(X) = P(1-P)^{-2}$.

Proof By definition

$$E_{gd}(X) = \sum_{m=1}^{\infty} m \cdot P^{m-1}(1-P) = \frac{1}{1-P} \tag{1}$$

Also by definition

$$\sigma_{gd}^2 = E_{gd}(X^2) - [E_{gd}(X)]^2 = \sum_{m=1}^{\infty} m^2 \cdot P_x(m) - [\sum_{m=1}^{\infty} m \cdot P_x(m)]^2 = \frac{1+P}{(1-P)^2} - \frac{1}{(1-P)^2} = \frac{P}{(1-P)^2} \tag{2}$$

Where σ_{gd}^2 is the variance.

Reproduction is therefore a stochastic process which can be thought of as searching a neighborhood of the solution space in order to better the population. It seems intuitive to search a larger neighborhood in the beginning when the population is not very good, and research a smal1er neighborhood later on when it has become substantially better. Thus we believe a graded perturbation of the organisms is efficient. Computer simulations indicate that by setting the initial $E_{gd}(X)$ proportional to $\log n$, a quick and effective search is conducted. Therefore, we choose

$$E_{gd}^*(i) = (k \log n - 1)e^{-\alpha i} + 1 \tag{3}$$

Where $E_{gd}^*(i)$ is the desired expected number of exchanges at the $i-th$ iteration and α and k are constant.

Setting $E_{gd}(X) = E_{gd}^*(i)$ yields

$$P(i) = \frac{(k \log n - 1)e^{-\alpha i}}{(k \log n - 1)e^{-\alpha i} + 1} \tag{4}$$

Note that since $E_{gd}^*(i)$ is a decreasing function, the overall expected number of exchanges per reproduction.

Lemma 3 Reproduction time $\le O(\log n)$ with probability $\theta \to 1$.

Proof Since the population space is constant, reproduction time is proportional to X, the number of exchanges.

For any fixed probability $\theta < 1$,let $X(\theta)$ be the smallest integer that satisfies $F(X(\theta)) = P(X \le X(\theta)) \ge 0$, the probability that the total number of exchanges does not exceed $X(\theta)$ is at 1east θ. We shall show that $X(\theta) \le O(\log n)$.

Since $\forall i[P(i, \alpha = 0) \ge P(i, \alpha > 0)$, it follows from (4) that $\forall \theta[X(\theta; \alpha = 0) \ge X(\theta, \alpha > 0)$. Hence, it suffices to show that $X(\theta; \alpha = 0) = O(\log n)$.

Chebgchev's inequality states that for any $\lambda > 0$, $P(|X - \mu| \le \lambda\delta) \ge 1 - 1/\lambda^2$, where the random variable X is distributed with mean μ and variable δ^2. Thus if $X(\theta; \alpha = 0) = E_{gd} + \lambda\delta$, where $\lambda = (1/\sqrt{1-\theta})$, then $F(X(\theta)) \ge \theta$.

Since $\delta \le E_{gd}$, $X(\theta; \delta > 0) \le X(\theta; \alpha = 0) = E_{gd} + \frac{\delta}{\sqrt{1-\theta}} = O(E_{gd}) = O(\log n)$ (5)

Theorem 4 Running time $\le O(n \log n)$ with probability $\theta \to 1$.

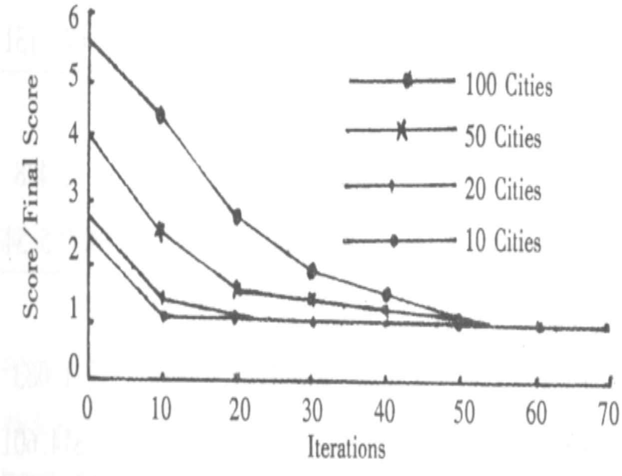


Fig.1 The logarithmic learning curves

Proof Creating the organisms and calculating their scores takes $Q(n)$ time. From (5) we know that reproduction takes $\leq O(n\log n)$ time. Computer simu1ations indicate that "good" solution to even 1arge city problems can be obtained in $O(\log n)$ iterations (See Fig.1). Therefore, composite analysis shows that the proposed algorithm runs, even in the Probabilistic worst case, $\leq [O(\log n)\cdot O(n+\log n)], \text{or} \leq O(n\log n)$ time. The expected running time is also $\leq O(n\log n)$.
From left to right, in Fig.1, the curved depict the convergence for a 10, 20, 50, and 100 cities TSP.

III. Conclusion

We have used the algorithm to China-Traveling Salesman Problem, the shortest route in Fig.2.

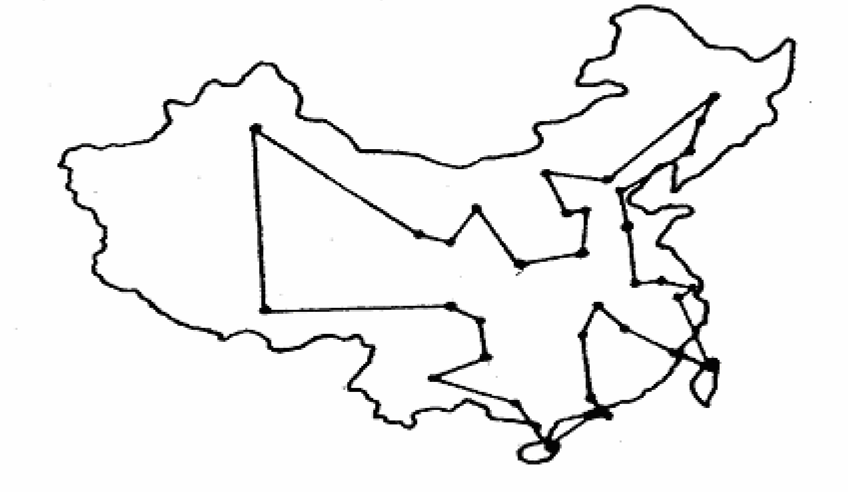


Fig.2 The route of the C-TSP

We present some results of computer simu1ations of proposed algorithm for various size TSP' s. Uniform random variables distributed on a Square of side length 100. The following result

$$\lim \frac{d_n^*}{s\sqrt{n}} \approx 0.749 \quad (n \to \infty) \tag{6}$$

Where d_n^* is the optional distance and s is the length of the square containing the cities, can been used to evaluate the asympotic efficiency of the algorithm.

Tab.1 The comparison of the algorithms

Methods	Number of Cities			
Evolution	10	20	50	100
Time	0.100	0.285	0.868	8.901
Distance	268.324	431.878	620.131	924.751
Greedy				
Time	0.053	0.210	1.428	6.470
Distance	268.324	1075.610	2565.541	4899.416
GA				
Time	0.113	4.351	1.083	2.411
Distance	268.324	431.878	814.601	1230.887
Expected Min Distance	268.324	431.878	529.623	749.00

The proposed algorithm was compared against the greedy algorithm and the genetic algorithm proposed by Fogel. The former algorithm, Perhaps the most intuitive one, consists of starting at a random city and proceeding to the closest city not already visited and finally returning to the starting city. The latter (hereafter referred to as GA) consists of mutating a population of organisms each of which competes against a fixed number of other organisms for survival. The probability of an organism's survival is inversely proportional to its score and directly proportional to that of its Competitor. Selection is probabilistic in GA, whereas it is deterministic in our algorithm.

The biological algorithm presented above has both theoretical and practical interest. The latent power of Darwin evolution as apply to combinatorial optimization, two seemingly unrelated subjects.

Most NP -comp1ete problems can easily be reformulated so that their feasible solution can undergo the process of mutation and reproduction as described above. The adaptability of this technique to tackle other NP -complete problems such as the tree problem demonstrates its robustness as a combinational optimization technique.

It is a1so appealing to consider hybrids of neural network learning algorithm with evolutionary search procedures, in view of Nature's success in this area. Genetic algorithm may be used to take advantage of the stochastic variation that often gradient techniques. This approach may also reduce the training times of neural networks. Further research will focus on investigating these interesting avenues.

References

[1] Fogel D B. An Evolutionary Approach to the Traveling Salesman Problem [M]. Cambridge Univ. Press Cambridge, UK,1998.

[2] Dang J W. Neural Network Theory Application [D]. Chengdu: Southwest Jiaotong University, 1996.

[3] Dang J W. Neural Network Technique [M]. Beijing: China Railway Press, 2000

[4] Hopfield J J. Neurons with graded response have collective computational properties like those of two –state neurons[J] Proc.Natl. Acad. Sci USA 81, 1984, 3088-3092.

[5] Xie X,Zhang W,Yang Z. Adaptive Particle Optimization on Individual Level[C].International Conference on Signal Processing(ICSP2002).Beijing:2002,1215-1218.

[6] Zhou Chun-guang, Liang Yan-chun. Computational Intelligence[M].Changchun: Jilin

University Press ,2001,269-277.
[7] Huang Lan, Wang Kang-ping, Liang Yan-chun.Hybrid Ant Colony Algorithm for Taveling Salsman Problem[J].*Journal of Jilin University*.2002,40(4):369-373.

Jian-wu Dang (1963-),doctoral supervisor, Lanzhou Jiaotong University, mainly researches in neural networks and artificial intelligence.