A Method of Finding Shortest Paths for Bumblebees

Yu Guyang¹, Shi Huajun²

¹About the Author: Yu Guyang: Shanghai East—China Computer Co. Ltd Postgraduate student
²The 32nd Research Institute of China Electronics Technology Group Corporation, Shanghai, China

Research direction: artificial intelligence (AI)

¹The 32nd Research Institute of China Electronics Technology Group Corporation, Shanghai, China

Abstract: In October 2010, a new British study said that bumblebees fluttering from flower to flower showed the ability to easily solve the "traveler salesman problem." There is no good explanation for how the bumblebee solved the problem. In this paper, a simple and efficient algorithm is proposed to explain how bumblebee solves the traveling salesman problem.

1. Introduction

TSP, Traveling Salesman Problem, Traveling Salesman Problem, TSP Problem. The problem is to find the minimum path cost for a single traveler to go from the starting point, through all given points of demand, and finally back to the origin.

In October 2010, researchers at institutions such as royal holloway, university of London, reported that bumblebees have shown an ability to crack the problem with ease. So how does the bumblebee solve the TSP problem?

2. Introduction of algorithm

2.1 Proposal of a Problem

As shown in figure 1, starting from city A, if the path constructed according to the greedy algorithm is ABCDA, the total cost is 2+3+23+5=33, while the cost of path ACBDA is 4+3+7+5=19.

Figure 1 a simple diagram
We naturally ask: why choose edge AC over edge AB? How can we determine whether an edge belongs to the shortest Hamilton loop of the graph? Can we quantify the probability that an edge belongs to the shortest Hamilton loop?

2.2 Team competition and scoring rules:
For edge AB and AC in figure 1, it is difficult to judge the advantages and disadvantages if directly compared. Our solution is to race paths and paths.

Definition: Effective path $\text{EP}_k (K>=4)$: a path consisting of K nodes that may belong to the shortest Hamiltonian cycle of the graph.

Corollary: if $A_0A_1…A_{k-1}$ is an $\text{EP}_k$, it must be the shortest path from $A_0$ to $A_{k-1}$ passing $A_1, A_2, …, A_{k-2}$.

For example, CBDA is an $\text{EP}_4$. It is the shortest path from C to A passing nodes B and D. There are two possible paths from C to A through nodes B and D: CBDA and CDBA. Edges CB, BD, DA, CD, and BA participated in the competition, CB, BD, and DA won, while CD and BA failed.

We call this process a competition.

2.3 Link degree
We find out all the $\text{EP}_k (K>=4)$ in Figure 1, calculate the sum of the scores of each edge in all the competitions and the number of competitions participated of each edge. The sum of the score of edge x is $S_k(x) (K>=4)$, the sum of edge x which participated in competitions is $T_k(x) (K>=4)$.

We can calculate the link degrees of edge x, $L_k(x) = S_k(x) / T_k(x) (K>=4)$.

For example, in Figure 1, we take $K=4$, we can get: $S_4(AB)=1$, $S_4(AC)=3$, $S_4(AD)=5$, $S_4(BC)=5$, $S_4(BD)=3$, $S_4(CD)=1$.

The competition times of each edge are 5, we can get the link degrees of each edge $L_4(AB)=0.2$, $L_4(AC)=0.6$, $L_4(AD)=1$, $L_4(BC)=1$, $L_4(BD)=0$, $L_4(CD)=0.2$.

To some extent, $L_k(x)$ represents the probability of this edge belonging to shortest Hamiltonian cycle. The closer that K is to number of nodes N, the more precise $L_k(x)$ is.

Using the value of $L_k(x)$, we can easily get the shortest Hamiltonian cycle acbda in Figure 1 through the Greedy Algorithm.

3. How Bumblebees search for shortest Hamiltonian cycle?

3.1 Quadrilateral division
Bumblebees use $K=4$ to calculate the link degrees of each edge. A bumblebee divides the whole graph into several quadrilaterals. If the node number $N=6$, it divides into $\binom{6}{4}=15$ quadrilaterals.

In Euclidean space, quadrilaterals can be divided into convex quadrilaterals (Figure 2) and concave quadrilaterals (Figure 3).

There is a simple method to calculate the quadrilaterals.
For each quadrilateral, there are 6 EP3. Here is the method for the calculation: suppose d(x) is the length of the edge x, 

Convex quadrilateral (Figure 2): if \(d(AD)+d(BC)>d(CD)+d(AB)\), then the score of each edge is \(S_4(CD)=5\), \(S_4(AB)=5\), \(S_4(AD)=3\), \(S_4(BC)=3\), \(S_4(AC)=1\), \(S_4(BD)=1\). The number of competitions participated is \(T_4(x)=5\).  

Concave quadrilateral (Figure 3) if \(d(AD)+d(BC)<d(AC)+d(BD)<d(AB)+d(CD)\), then the score of each edge is \(S_4(AD)=5\), \(S_4(BC)=5\), \(S_4(AC)=3\), \(S_4(BD)=3\), \(S_4(AB)=1\), \(S_4(CD)=1\). The number of each edge competitions participated is \(T_4(x)=5\).  

In this way, the bumblebees only need to calculate twice in average to work out one quadrilateral. 

In this way, the bumblebee processing one quadrilateral at a time needs to make a quadratic judgment on average.

### 3.2 Link degree calculation

Bumblebees calculate each edge x, \(S_4(x)\) and \(T_4(x)\), then according to each edge \(L_4(x)\), using the Greedy Algorithm to calculate the shortest Hamiltonian cycle. 

As shown in Figure 4, we may get the total number of competitions participated for each edge.

- \(S_4(AB)=27\)  \(S_4(AC)=41\)  \(S_4(AD)=75\)  \(S_4(AE)=65\)
- \(S_4(AF)=41\)  \(S_4(AO)=43\)  \(S_4(AH)=23\)
- \(S_4(BC)=45\)  \(S_4(BD)=21\)  \(S_4(BE)=57\)  \(S_4(BF)=57\)
- \(S_4(BG)=35\)  \(S_4(BH)=73\)
- \(S_4(CD)=51\)  \(S_4(CE)=21\)  \(S_4(CF)=39\)  \(S_4(CG)=51\)
- \(S_4(CH)=67\)
- \(S_4(DE)=53\)  \(S_4(DF)=39\)  \(S_4(DG)=51\)  \(S_4(DH)=25\)
- \(S_4(EF)=43\)  \(S_4(EG)=37\)  \(S_4(EH)=39\)
- \(S_4(FG)=53\)  \(S_4(FH)=43\)
- \(S_4(GH)=45\)

Since \(T_4(x)\) is all 75, so we only need to compare \(S_4(x)\). According to Greedy Algorithm, we need to find ad, then bh, ch, ad, ed, fg, cg and then df. So the shortest Hamiltonian cycle is adfgchbea.
3.3 Search for shortest Hamiltonian Cycle

Bumblebees follow the following three rules when looking for the shortest Hamiltonian Cycle.

a) All the edges can not be intersected
b) In the shortest Hamilton Cycle, each continued 4 points can form an EP4.
c) If in the searching of Hamiltonian cycle, there is edge x, whose L4(x) is smaller than 0.5, then we shall exclude this edge and look for another Hamiltonian Cycle, and select the shorter cycle as the shortest Hamiltonian Cycle.

For example, in Figure 4, because S4(BE)=57, S4(BF)=57, if you choose bf, we can find the Hamiltonian Cycle: adchbfgaa. S4(GE)=37, L4(GE)=0.49, exclude edge GE, we can find EB. In this way, we find another Hamiltonian Cycle ADFGCHBE.

As to the rule c, strictly speaking, we have not found the example. Because we can also find the shortest Hamiltonian Cycle by calculating each edge separately in Figure 4. But we believe, in some extreme situation, bumblebees may use Rule c.

We believe bumblebees handle quadrilaterals and seek for paths at the same time; the more quadrilaterals they need to handle, the more accurate the link degrees of each edge is, the shorter the Hamiltonian Cycle is.

3.4 Effective amount of calculation

In order to look for the shortest Hamiltonian Cycle, bumblebees need to handle 15 quadrilaterals when the amount of node is 6, N=6, handle 70 quadrilaterals when the amount of node is 8, N=8. According to the statistical theory, bumblebees can already find the shortest Hamiltonian Cycle after a certain amount of quadrilaterals handled.

According to experiment, they only need to handle about 7-8 quadrilaterals when N=6, handle about 22 quadrilaterals when N=8. When bumblebees have handled a certain amount of quadrilaterals, they have already found the shortest Hamiltonian Cycle, the rest of the work does not mean a lot in the finding the shortest Hamiltonian Cycle.

3.5 The result of experiment

According to foreign experimental reports, most of the experiments about bumblebee are concentrated on 6 nodes, while only a small part of the experiments are 8 nodes. We have simulated the two situations respectively by computer.

For N=6, we have conducted hundreds of experiments, all of which can directly generate the shortest Hamilton loop based on the greedy algorithm.

For N=8, as a separate experiment, we randomly generated 20 shapes at once. Sixteen of them found the shortest loop directly according to greedy algorithm and rule 1. In the other four graphs, greedy algorithm and rule 1 were used to find a loop first, and then according to rule 2, four consecutive points need to form an EP4 for optimization, and the shortest Hamilton loop was also found successfully.

4. Bumblebee Algorithm and some thoughts

The challenge of TSP is how to find the shortest Hamiltonian Cycle when there are many nodes. We can improve this method applying in this “many nodes” situation.

I. Take an appropriate K (K>=4) to divide the graph, then find every EPk.
II. Calculate the winning odds of each edge Lk(x) based on the current solution group.
III. Add the biggest winning odd that is not in conflict with the edges from the solution group, into the current solution group.
IV. If in the solution group, there are already N edges, then END. If not, go back to II.

The computational complexity of this algorithm is O(N^K+4).

This Algorithm contains some interesting and very important characteristic, at the same time, it can evolve into many Algorithms. Next step is to experiment with many Nodes, we can discuss in detail in other articles.
This algorithm has some interesting and very important properties, and it can also produce many changes. Our next research work will mainly be carried out in the following aspects:

1. The core problem. If $L_K(X)$ of edge $X$ is equal to 1, does edge $X$ belong to the shortest Hamiltonian cycle? The answer is right when $K$ is equal to 4 or $K$ is equal to 5, and in general, so far we have neither proved nor found any counterexamples. Our guess is that this theorem holds.

2. With the increase of $K$, the question of whether $L_K(X)$ has monotony and its trend is raised.

3. Statistics of incomplete link-degree for each edge of different $K$.

4. The relationship between $K$ and $N$.

By calculating the winning odd of each edge to find the shortest Hamiltonian Cycle is the Algorithm that I would like to name as the Bumblebee Algorithm.

TSP is an NPC problem, and the research on it is of great significance. We hope to do some useful work on the TSP problem through the study of this algorithm. We would like to call this the bumblebee algorithm if we could.

Acknowledgment
It took me nearly two months to finish writing this paper. In this process, it brought me infinite passion and harvest in my student life. In the writing process of the paper encountered numerous difficulties and obstacles, are in the help of students and teachers through. In the school library to find information, the teacher gave me a lot of aspects of library's support and help, especially want to thank my thesis advisor - ShiHuaJun teacher, didn't he much guidance and help, for my selfless paper modifications and improvements for me, no I this paper finally completed. Here, I would like to express my heartfelt thanks to all the teachers who have instructed and helped me! At the same time, I would also like to thank the monographs of scholars quoted in this paper. Without the inspiration and help from the research results of these scholars, I would not be able to complete the final writing of this paper. So far, I would also like to thank my friends and classmates, who gave me a lot of useful materials in the process of writing my paper, and also provided enthusiastic help in the process of typesetting and writing my paper! No man is perfect without his faults. Due to my limited academic level, the paper will inevitably have shortcomings, I sincerely hope that you can criticize and correct me!

References
[1] Mathieu Lihoreau, Lars Chittka, and Nigel E.Raine. Travel Optimization by Foraging Bumblebees through Readjustments of Traplines after Discovery of New Feeding Location VOL.176, NO.6 THE AMERICAN NATURALIST DECEMBER 2010
[2] Zbigniew Michalewicz, and David B. Fogel(2000) . How to Solve It: Modern Heuristics
[3] S.Lin & B. W. Kernighan,An Effective Heuristic Algorithm for the Traveling-Salesman Problem,Operations Research 21,498-516(1973)
[4] Gutin and Punnen(eds),The Traveling Salesman Problem, Future Generation Computer System 11(1995) 175-182
[5] Golden B.,A Statistical Approach to the TSP, Network,1977,209~225
[6] Nurmi K. ,Traveling Salesman problem Tools for Microcomputers, Computers and Operations Research ,18,6,1991,537~547
[7] Colorni A, Dorigo M. Heuristics from nature for hard combinatorial optimization problems. International Trans-Operational Research, 1996, 3( 1): 1- 21
[8] Dorigo M, Gambardella L M. An t colony system: A cooperative learning approach to the traveling salesman problem. IEEE Trans Evolutionary Computation , 1997, 1( 1): 53- 66
[9] Bonabeau E,Do rig o M, Theraul az G. Inspirati on for optimization from social insect behaviour. Nature, 2000, 406( 6): 39- 42