Australian Journal of Basic and Applied Sciences, 5(10): 540-543, 2011
ISSN 1991-8178

Using Imperial Competitive Algorithm for Solving Traveling Salesman Problem and Comparing the Efficiency of the Proposed Algorithm with Methods in Use

[1]Kourosh Nemati, [2]Siti Mariyam Shamsuddin and [3]Morteza Saberi Kamarposhti

[1]Department of Computer Science, Islamic Azad University Nour Branch, Nour, Iran.
[2]Faculty of Computer Science and Information System Universiti Teknologi Malaysia, (UTM) Skudai, 81310, Johor Bahru, Malaysia.
[3]Department of Computer Science Islamic azad university, Firoozkooh Branch, Firoozkooh, Iran.

Abstract: In this paper a new method for applying Imperial Competitive Algorithm on Traveling Salesman Problem is proposed. Traveling Salesman Problem is classified as combinational optimization problems. So its problem complexity is exponential that called NP-Complete problems. Imperial Competitive Algorithm is a new optimization method that commonly used for solving NP-Complete problems. In this paper Imperial Competitive Algorithm is used for solving Traveling Salesman Problems. Experimental results show the efficiency of proposed method.

Key words: Traveling Salesman Problem; Imperial Competitive Algorithm; Evolutionary Algorithm; Graph Theory.

INTRODUCTION

The goal of Traveling Salesman Problem is finding shortest tour between N cities. Traveling Salesman Problem is classified as combinational optimization problems.

Traditional method is not a good way for solving NP-Complete problems. So some new methods was proposed for solving this problem such as Simulated Annealing (SA) (Tian and Wang, 2000), Artificial Neural Networks (Saadatmand *et al.,* 2001), Genetic Algorithm (Jiao and Wang, 2000) and Ant Colony (Stützle and Dorigo, 1999). Each of these problems has some strength and weakness. For instance, Because of computational complexity evolutionary algorithm such as genetic algorithm is not suitable for online use and commonly it was used for offline applications. Instead of mentioned weakness, evolutionary algorithms have some good feature for solving complex and nonlinear problems (Boettcher and Percus, 2000). However, evolutionary algorithms are commonly used for solving NP-Complete problems.

In this paper Imperial Competitive Algorithm is used for solving Traveling Salesman Problems. Experimental results show the efficiency of proposed method. In next section, Traveling Salesman Problem is introduced. Then, Imperial Competitive Algorithm is defined and experimental results are shown at the final section.

Traveling Salesman Problem:

Graph is a good tool that is commonly used in several applications. One of the most important problems in the graph theory is Traveling Salesman Problem. Traveling Salesman Problem is generalized version of Hamilton Cycle. Suppose a complete graph that each edge of this graph has a nonnegative cost. On Traveling Salesman Problem, vertices refer to cities; edges refer to the path between cities and edges cost refer to the distance between cities. The salesman must to start from a city and visit all cities once only and return again to first city with minimized cost. However, the problem is Symmetric Traveling Salesman Problem if the graph is symmetric (undirected). Furthermore, If the graph is asymmetric (directed), the problem is Asymmetric Traveling Salesman Problem (Johnson *et al.,* 2002; Cirasella *et al.,* 2001; Grötschel and Holland, 1991; Padberg and Rinaldi, 1991).

Imperial Competitive Algorithm:

Like another evolutionary algorithms, Imperial Competitive Algorithm start with some random initial population that each of them called a “Country”. Some of the best country in the population is selected as “Imperialist”. Remained population is mentioned as “Colony”. Imperialist depending on their power attract colonies. The power of each empire depends on both parts: imperialist as a main core and the colonies. In mathematical model, it is modeled by the imperialist power in addition to few percent of colonies power (Esmaeil and Caro, 2007).

With formation of initial empires, imperialist competition is started. Each of imperialist will be removed if it can’t improve its power (at least prevent to decrease its power). So survival of each empire is depended on attracting other empires colonies. Thereupon, in imperialist competition, stronger empires gradually improve their

Corresponding Author: Kourosh Nemati, Department of Computer Science, Islamic Azad University, Nour Branch, Nour, Iran.
E-mail: Knms81@gmail.com

power and weaker empires will be removed. The empires must to develop their colonies to improve their power (Esmaeil and Caro, 2007).

Over time, colonies power will be closer to imperialist power and a convergence will be seen. When only one empire is existed, the algorithm is terminated. In this condition power of this empires colony is very close to empires power. The Imperial Competitive Algorithm flowchart can be seen in Figure 1.

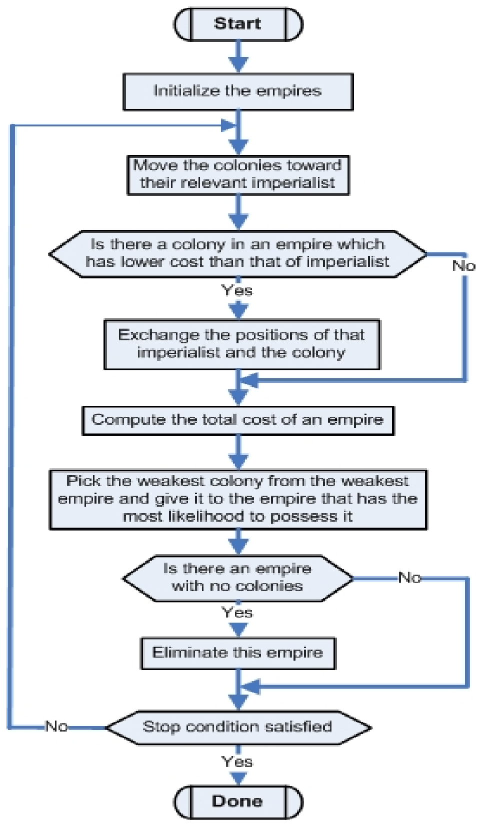

Fig. 1: The flowchart of Imperial Competitive Algorithm.

RESULTS AND DISCUSSIONS

In this section, TSPLIB data (Available online in http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/index.html) is used for evaluating of proposed method and comparing it with methods in use. The proposed method is applied on Symmetrical Traveling Salesman Problem. Simulating results of the proposed method is compared by Co-Adaptive Net (Cochrane and Beasley, 2003) and Memetic SMO. These results can be seen in Figure 2 and Table 1 and Table 2 respectively. This simulation is run on a computer with Intel Core i5 CPU and 4GB RAM. All the results are obtained over 10 runs of the simulation. The percentage deviation from the optimum of the mean solution value over the 10 runs is reported in column ‘‘%PDM’’. The percentage deviation from the optimum of the best solution value that was found is reported in column ‘‘%PDB’’.

Table 1: Comparing between proposed method and Memetic SMO.

			Memetic SOM			Proposed Method	
Data	Optimum	%PDM	%PDB	Time	%PDM	%PDB	Time
Eli51	426	2.51	1.64	0.11	2.22	1.39	0.09
Berlin54	7542	1.63	0.00	0.17	1.88	0.09	0.14
St70	675	1.13	0.59	0.20	1.01	0.44	0.16
Eli76	538	3.14	1.86	0.22	2.56	0.99	0.19
KroAloo	21282	1.41	0.18	0.31	1.01	0.10	0.25
KroBloo	22141	1.97	0.62	0.32	1.35	0.38	0.26
KroCloo	20749	1.02	0.30	0.31	1.25	0.47	0.25
Rd00	7910	2.66	0.43	0.32	2.05	0.15	0.26
Prl07	44303	2.07	0.28	0.37	1.89	0.22	0.30
Prl24	59030	2.04	0.00	0.39	2.28	0.14	0.32
Chl50	6528	3.16	1.81	0.49	2.79	1.36	0.39

Prl52	73682	2.53	1.07	0.52	2.07	0.95	0.41
Rat95	2323	6.77	6.03	0.72	4.87	3.98	0.58
Lin318	42029	4.95	3.48	1.25	5.23	3.94	1.09
Pcb442	50778	6.88	5.80	1.82	6.29	4.87	1.68
U574	36905	5.08	4.09	2.64	5.23	4.15	2.09
P654	34643	5.13	2.51	5.58	3.28	1.34	4.87
U724	41910	5.57	4.73	3.48	4.19	3.78	2.89

Table 2: Comparing between proposed method and Co-Adaptive Net.

		Co-Adaptive Net			Proposed Method		
Data	Optimum	%PDM	%PDB	Time	%PDM	%PDB	Time
Eli51	426	2.89	0.94	0.03	2.22	1.39	0.09
Berlin54	7542	7.01	0.00	0.05	1.88	0.09	0.14
St70	675	1.72	0.89	0.08	1.01	0.44	0.16
Eli76	538	4.35	2.04	0.07	2.56	0.99	0.19
KroAloo	21282	1.31	0.57	0.16	1.01	0.10	0.25
KroBloo	22141	2.20	1.53	0.16	1.35	0.38	0.26
KroCloo	20749	1.70	0.80	0.15	1.25	0.47	0.25
Rd00	7910	3.64	1.19	0.16	2.05	0.15	0.26
Prl07	44303	4.41	0.18	0.17	1.89	0.22	0.30
Prl24	59030	2.93	2.36	0.25	2.28	0.14	0.32
Chl50	6528	3.23	1.78	0.35	2.79	1.36	0.39
Prl52	73682	2.06	0.74	0.38	2.07	0.95	0.41
Rat95	2323	7.46	4.69	0.58	4.87	3.98	0.58
Lin318	42029	4.31	2.65	1.17	5.23	3.94	1.09
Pcb442	50778	7.58	5.88	1.87	6.29	4.87	1.68
U574	36905	4.92	3.78	2.62	5.23	4.15	2.09
P654	34643	4.56	3.69	4.09	3.28	1.34	4.87
U724	41910	5.59	4.95	3.68	4.19	3.78	2.89

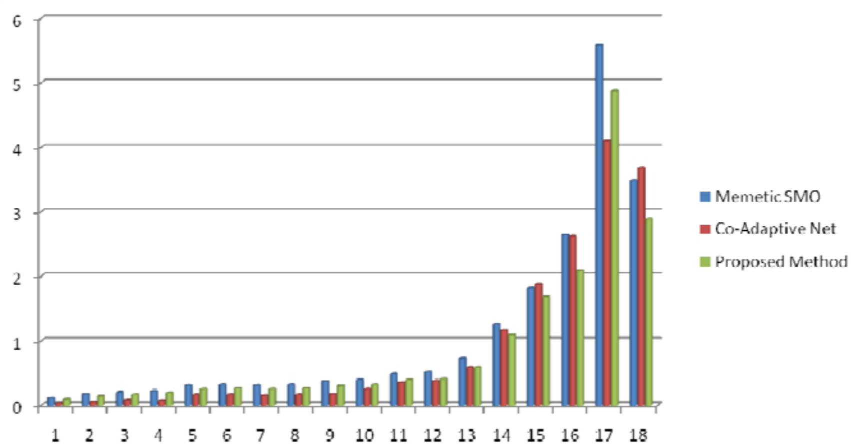

Fig. 2: Comparing between proposed method and Memetic SMO and Co-Adaptive Net.

ACKNOWLEDGMENT

This research was fully supported by Islamic Azad University, Nour Branch.

REFERENCES

Boettcher, S., A. Percus, 2000. "Research Note: Nature's way of optimizing," Artificial Intelligence, 119: 275-286.

Cirasella, J., D.S. Johnson, L.A. McGeoch and W. Zhang, 2001. "The Asymmetric Traveling Salesman Problem: Algorithms, Instance Generators, and Tests", in Algorithm Engineering and Experimentation, Third International Workshop, ALENEX 2001, Lecture Notes in Computer Science, Vol. 2153. Springer, Berlin, 32-59.

Cochrane, E.M., J.E. Beasley, 2003. The co-adaptive neural network approach to the Euclidean travelling salesman problem, Neural Networks., 16: 1499-1525.

Esmaeil Atashpaz-Gargari, Caro Lucas, 2007. "Imperialist Competitive Algorithm: An Algorithm for Optimization Inspired by Imperialistic Competition" 2007 IEEE Congress on Evolutionary Computation (CEC).

Grötschel M., and O. Holland, 1991. "Solution of Large-Scale Symmetric Traveling Salesman Problems", Mathematical Programming, 51: 141-202.

Jiao, L., L. Wang, 2000. "A novel genetic algorithm base on immunity," IEEE Trans. on Systems, Man, and Cybernetics-Part A: Systems and Humans, 30(5).

Johnson, D.S., G. Gutin, L.A. McGeoch, A. Yeo, W. Zhang and A. Zverovich, 2002. "Experimental Analysis of Heuristics for the ATSP", in the Traveling Salesman Problem and its Variations, G. Gutin and A. Punnen, Editors, Kluwer Academic Publishers, Boston, pp: 445-487.

Padberg, M. and G. Rinaldi, 1991. "A Branch-and-Cut Algorithm for the Resolution of Large-Scale Symmetric Traveling Salesman Problems", SIAM Review, 33: 60-100.

Saadatmand-T., M., M.-R. Akbarzadeh-T., M. Khademi, 2001. "A novel hybrid neural network for the traveling salesman problem (TSP)," 9th Iranian Conference on Electrical Engineering (ICEE2001).

Stützle, T., M. Dorigo, 1999. "ACO algorithms for the traveling salesman problem," In K. Miettinen, M. Makela, P. Neittaanmaki, J. Periaux, editors, Evolutionary Algorithms in Engineering and Computer Science, Wiley.

Tian, F., L. Wang, 2000. "Chaotic simulated annealing with augmented Lagrange for solving combinatorial optimization problems," 26th Annual Conference of the IEEE Industrial Electronics Society (IECON), 4: 2722 - 2725.