Performance Analyses on Population Seeding Techniques for Genetic Algorithms

P. Victer Paul[*], A. Ramalingam[$], R. Baskaran[#], P. Dhavachelvan[*], K.Vivekanandan[@], R. Subramanian[*] and V.S.K.Venkatachalapathy[&]

[*]Department of Computer Science, Pondicherry University, Puducherry, India.
[$] Department of MCA, Sri Manakula Vinayagar Engineering College, Puducherry, India.
[#]Department of Computer Science and Engineering, Anna University, Chennai, India.
[@] Department of Computer Science and Engineering, Pondicherry Engineering College, Puducherry, India.
[&]Department of Mechanical Engineering, SMVEC, Puducherry, India.
{victerpaul, a.ramalingam1972}@gmail.com, baaski@annauniv.edu, dhavachelvan@gmail.com, k.vivekanandan@pec.edu, {subbur, vskvenkatachalapathy}@yahoo.com

***Abstract*—In Genetic Algorithm (GA), the fitness or quality of individual solutions in the initial population plays a significant part in determining the final optimal solution. The traditional GA with random population seeding technique is simple and proficient however the generated population may contain poor fitness individuals, which take long time to converge to the optimal solution. On the other hand, the hybrid population seeding techniques, which have the benefits of generating good fitness individuals and fast convergence to the optimal solution. Researchers have proposed several population seeding techniques using the background knowledge on the problem taken to solve. In this paper, we analyse the performance of different population seeding techniques for the permutation coded genetic algorithm based on the quality of the individuals generated. Experiments are carried out using the famous Travelling Salesman Problem (TSP) benchmark instances obtained from the TSPLIB, which is the standard library for TSP problems. The experimental results show the order of performance of different population seeding techniques in terms of Convergence Rate (%) and Error Rate (%).**

Keyword - Genetic Algorithm, Population Seeding Technique, Travelling Salesman Problem, Performance Analysis, MATLAB

I. Introduction

Optimization is a specialized field of theoretical computer science and applied mathematics. The prime objective of optimization functions is to find the optimal object, which may be an individual or set of individuals from within a finite set of objects. Sometimes the objective may be compromised with the findings of near optimal objects, which may not exactly optimal, but closer to optimal elements. Hence, the process of solving the optimization problems is a tricky one, for which heuristics may contribute evidently to find the either optimal or near optimal solutions as per the requirements. In this perspective, several meta-heuristic search techniques have been proposed to solve optimization problems, such as Tabu Search [9,10], Genetic Algorithm [11-17], Ant Colony Optimization [18-22], Particle swarm Optimization [23], Neural Networks [24,25,47], Simulated Annealing [26], Multi-Agent System [10,27,37] and Hybrid-Heuristics [28-30]. Among these search techniques, Genetic Algorithm (GA) is a well-known technique for optimization of complex problem with large search space. Some of the important features of GA which makes it perform well are GA operates on population of possible solutions rather than on a single solution, the range of genetic operators helps to search the unrevealed portions in the large search space.

The traditional GA consist of following steps population seeding (initial population), selection, reproduction, crossover, mutation and termination constraint in which first step occurs once and rest of the step are repeated until the stop condition is satisfied. The first step of any GA is to generate a set of feasible solutions randomly or hybrid as an initial population or population seeding [34]. The quality of individual solutions in initial population plays a critical role in determining the quality of final solution that can be obtained using GA [32,40]. Though random population initialization is often used [41], many hybrid population seeding techniques like Nearest Neighbor (NN) [3], Gene Bank (GB) [32], Selective Initialization (SI) [46], Sorted population (SP) [36] techniques are also used to improve the performance of permutation coded GA. From the literature study, it is found that the performance comparison among the population seeding techniques have not been performed in terms of their ability to converge the optimal solution. This motivates to perform an effective performance analyses on different population seeding techniques for the permutation coded genetic algorithm.

In order to perform the experiments, Traveling Salesman Problem (TSP) instances from the TSPLIB have been used. TSP is the most widely studied problem in combinational optimization [1-3]. In TSP, a salesman wants to visit each of a set of cities exactly once and return to the starting city with minimal distance travelled. Major applications of TSP are Vehicle routing [4], Drilling of printed circuit boards [5], Overhauling gas turbine engines [6], X-Ray crystallography [7], computer wiring [4], and the order-picking problem in warehouses [8]. Thus, TSP could be the suitable benchmark instances to evaluate the performance of the different population seeding techniques. The organization of the paper is as follows: Section 2 provides a comprehensive study on different population seeding techniques; Section 3 discusses about working principle various population seeding techniques and Section 4 details the experimentation and result analysis; finally Section 5 concludes the paper and proposes future works.

II. RELATED WORKS

In this section, a comprehensive study on different population seeding techniques has been carried out. Lawrence *et al.* [35] discussed about different GA configuration problems and claims that seeding the initial population using hybrid model can improve the effectiveness of the GA significantly. Tog˘an *et al.* [40] believes that ability to produce optimal solution is critically influenced by population seeding method and proposed two new self-adaptive member grouping strategies and a new strategy for population seeding to show the importance of the population seeding in GA. Random population seeding method [3,16,30-32,50,51] is widely used because of its simplicity, easy understanding and uncomplicatedness to implement. Nearest Neighbor (NN) tour construction heuristic is one of the familiar alternative for random population seeding in GA for TSP [3,16,39,42-45,49]. In NN technique, individuals in the population seeding are constructed with city nearest to the current city and such good individuals can refine the subsequent search in the next generations [3].

Yingzi *et al.* [32] proposed a Greedy GA (GGA) in which population seeding is performed using the Gene Bank (GB) technique. The GB is built by collecting the permutation of 'N' cities based on their distance. In GGA method, population of individuals is generated from the GB such that the individuals are of above-average fitness and short defining length. The Selective Initialization (SI) [46] population seeding technique formulates k-nearest-neighbor subgraph and its gives higher priority to edges that belong to the formulated subgraph. Olga *et al.* [36] proposes a modified GA with sorted initial population method based on theory of better parents would produce better offspring. In this approach, a large initial pool of population is generated, and then it is sorted in ascending order based on their quality and choosing a certain number of individual with above average fitness.

III. POPULATION SEEDING TECHNIQUES

In this section, a brief description on the working principle of different population seeding techniques such as Random, Nearest Neighbor, Gene Bank, Selective Initialization, Sorted population techniques has been studied.

A. Random Initialization Technique

In this technique, the successive cities of the initial solutions are chosen randomly. A variety of random number generation techniques have been proposed such as quasi random, sobol random and uniform random sequence [34,37,41,48].

B. Nearest Neighbor Technique

Nearest neighbor (NN) tour construction heuristic is a common choice, in alternative for random population initialization, to construct the initial population of solutions for solving TSP with GAs. In NN technique, individuals in the population seeding are constructed with city nearest to the current city and such good individuals can refine the subsequent search in the next generations [3].

C. Gene Bank Technique

The Gene Bank technique [32] is proposed to generate the initial population of solutions with quality and diversity. The working principle of GB technique is exemplified as follows: For a TSP of N cities, the 'N' cities are permuted and assembled to build a gene bank. The C cities that are closer to the city i are encoded to construct a gene bank, where C is a number less than $N-1$. Thus, the Gene bank is a matrix A whose size is $C \times N$. The element of $A[i][j]$ is the j^{th} closest city to city i. Therefore, the C closest cities constitute the whole i^{th} row of gene bank for the city i.

The initial city i for each solution is initialized randomly and from the i^{th} row of gene bank, city j is then selected where j is the nearby one in the unvisited elements of the i^{th} row. Then, city k is selected from the j^{th} row of gene bank as the next city. If all the city codes of the j^{th} row have been selected, then next city is chosen randomly from the set of unvisited cities. This method is repeated so that the solution is generated with the size of 'N'.

D. Selective Initialization Technique

In this technique [46], a k-nearest-neighbor subgraph is formulated, from the distance matrix, as a graph include all routes of cities c_i and c_j, such that city c_i is among the k-nearest neighbors of city c_j or city c_j is among the k-nearest neighbors of city c_i. (i.e) a list of k-nearest neighbors for each city is generated in advance. The value of 'k' is assigned based on the size of the problem taken. In selective initialization, higher priority is given to the routes with cities that belong to the k-nearest-neighbor subgraph. As an example, from a city c, first attempt to randomly select a next city from the *c*'s *k*-nearest neighbors list and if all cities of c in the k-nearest neighbors list have already been used, then the next city is chosen randomly.

E. *Sorted population Technique*

GA is motivated by the concept that the better parents would produce better offspring. Sorted population technique [36] generates a large initial solution of population, then sorting it in ascending order based on their fitness and choosing a certain percentage of population that has above average fitness, in case of TSP – short distance. It is assumed that with very large initial population there is a high probability of having good solutions in the population. This technique uses merge sort for sorting the initial solutions of population due to its stability.

IV. Experimentation and Analyses

In this section, the experiments are conducted to analyse the performance of the Random, Nearest Neighbour, Gene Bank, Selective Initialization and Sorted population seeding techniques.

A. *Experimentation Setup*

The different population seeding techniques are implemented using MATLAB tool on an Intel Dual Core PC with TSP datasets, obtained from TSPLIB [38]. The TSP instances that have been chosen for experimentation are eil51, kroA100, tsp225, d493, d657, u724, rat783, fl1577, d2103 and fnl4461. Path representation of individuals is used to represent the tour of cities in the TSP. The size of population is set as 100. The value of 'C' in Gene Bank technique is assigned as 3 and the value of 'k' in Selective Initialization Technique is assigned as 10 for the instances with less than 600 cities and as 20 for instances larger than 600 cities.

B. *Performance Factor*

Convergence Rate: Convergence Rate of a solution can be defined as the percentage of fitness attained by the solution w.r.t the known optimal solution for the problem. It can be given as,

$$Convergence\ Rate\ (\%) = 1 - \frac{Fitness\ -\ Optimal\ Fitness}{Optimal\ Fitness} \times 100$$

The Convergence rate factor is directly proportional to the quality or fitness of the solution generated.

Error Rate: Error Rate of a solution can be defined as the percentage of difference in the fitness value of the solution with the known optimal solution for the problem. It can be given as,

$$Error\ Rate\ (\%) = \frac{Fitness - Optimal\ Fitness}{Optimal\ Fitness} \times 100$$

The error rate factor is inversely proportional to the quality or fitness of the solution generated.

C. *Result Analyses*

In this section, the different population seeding techniques are investigated with similar GA configurations and the quality of individuals generated in the population initialization is examined. The Best solution of the population represents the solution with highest convergence rate and the worst solution of the population represents the solution with least convergence rate in the population. The initial solutions generated using each technique are inspected for the best and worst solutions in terms of their convergence rate and are tabulated in the Table I. For each technique, experimental outcomes have been collected for 10 runs and the average values of 10 runs are used for analyses.

TABLE I
The Convergence Rate (%) based evaluation for different population seeding techniques

Sl. No	Instance	Convergence Rate (%)									
		Random		NN		GB		SI		SP	
		Best	Worst	Best	Worst	Best	Worst	Best	Worst	Best	Worst
1	eil51	1.19%	-24.91%	81.27%	51.96%	69.52%	0.88%	64.77%	30.82%	11.47%	-26.83%
2	kroA100	-24.55%	-54.20%	83.95%	65.17%	16.33%	-10.84%	39.13%	19.24%	-2.48%	-34.45%
3	tsp225	-24.89%	-59.55%	81.69%	63.43%	5.40%	-27.37%	36.14%	10.25%	-8.69%	-41.36%
4	d493	-40.83%	-99.74%	81.44%	69.48%	-2.39%	-45.35%	38.52%	10.84%	-34.67%	-70.29%
5	d657	-62.37%	-87.70%	76.37%	61.90%	-8.88%	-44.21%	50.43%	31.08%	-47.50%	-70.19%
6	u724	-40.08%	-94.24%	78.06%	65.53%	-7.19%	-38.41%	51.65%	16.89%	-36.37%	-62.33%
7	rat783	-79.51%	-118.25%	78.39%	67.76%	-14.70%	-43.63%	54.63%	17.36%	-62.43%	-103.52%
8	fl1577	-98.54%	-153.18%	79.63%	67.10%	-20.76%	-34.26%	50.67%	10.59%	-72.85%	-130.68%
9	d2103	-77.10%	-155.70%	91.28%	80.03%	-18.29%	-44.49%	54.84%	6.41%	-61.51%	-127.72%
10	fnl4461	-86.16%	-169.41%	78.40%	72.69%	-9.71%	-61.30%	36.08%	9.09%	-71.70%	-143.83%

TABLE II
The Error Rate (%) based evaluation for different population seeding techniques

Sl. No	Instance	Error Rate (%)									
		Random		NN		GB		SI		SP	
		Best	Worst	Best	Worst	Best	Worst	Best	Worst	Best	Worst
1	eil51	98.81%	124.91%	18.73%	48.04%	30.48%	99.12%	35.23%	69.18%	88.53%	126.83%
2	kroA100	124.55%	154.20%	16.05%	34.83%	83.67%	110.84%	60.87%	80.76%	102.48%	134.45%
3	tsp225	124.89%	159.55%	18.31%	36.57%	94.60%	127.37%	63.86%	89.75%	108.69%	141.36%
4	d493	140.83%	199.74%	18.56%	30.52%	102.39%	145.35%	61.48%	89.16%	134.67%	170.29%
5	d657	162.37%	187.70%	23.63%	38.10%	108.88%	144.21%	49.57%	68.92%	147.50%	170.19%
6	u724	140.08%	194.24%	21.94%	34.47%	107.19%	138.41%	48.35%	83.11%	136.37%	162.33%
7	rat783	179.51%	218.25%	21.61%	32.24%	114.70%	143.63%	45.37%	82.64%	162.43%	203.52%
8	fl1577	198.54%	253.18%	20.37%	32.90%	120.76%	134.26%	49.33%	89.41%	172.85%	230.68%
9	d2103	177.10%	255.70%	8.72%	19.97%	118.29%	144.49%	45.16%	93.59%	161.51%	227.72%
10	fnl4461	186.16%	269.41%	21.60%	27.31%	109.71%	161.30%	63.92%	90.91%	171.70%	243.83%

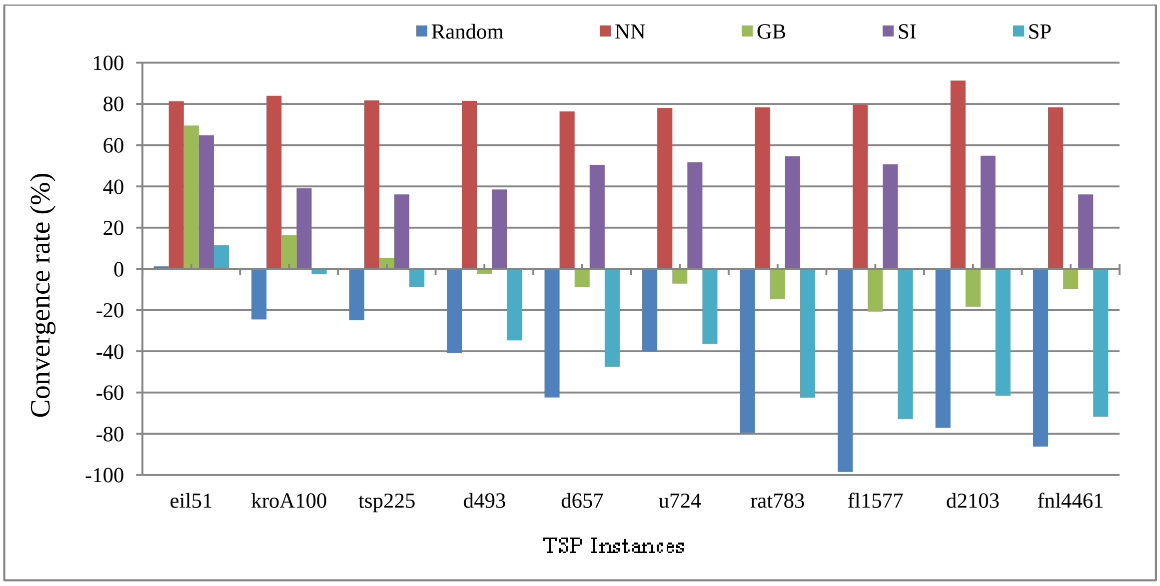

Figure 1. The Convergence Rate (%) of the best solutions generated by different population seeding techniques

From the Table I, it can be observed that the NN technique can generate the individual with high convergence rate for all the TSP instances taken. At maximum, NN generate a solution with 91.28% of convergence rate for the TSP instance d2103 at the initial stage of GA. The SI technique which performs better in next to NN technique can generate solution of 64.77% of convergence for the instance eil51. It can also be noted that the performance of SI diminishes with increase in the size of the TSP problem instance. The performance of GB technique degrades with increase in the size of the TSP instance and the negative value shows the bad quality of the individual generated. The random and SP techniques show their inability to produce high fitness individuals at the initialization stage of GA. The convergence rate (%) of best and worst individuals generated by different population seeding techniques is shown in the figure 1 and 2 respectively.

The error rate (%) of the best and the worst solutions generated by the different population seeding techniques are tabulated in the Table II. From the Table II, it can be derived that the NN technique performs better in terms of error rate (%) based assessments and the sequence of performance is same as that of convergence rate (%). Figures 3 and 4 correspond to the best and the worst individuals of the initial population generated by various population seeding methods in terms of error rate (%).

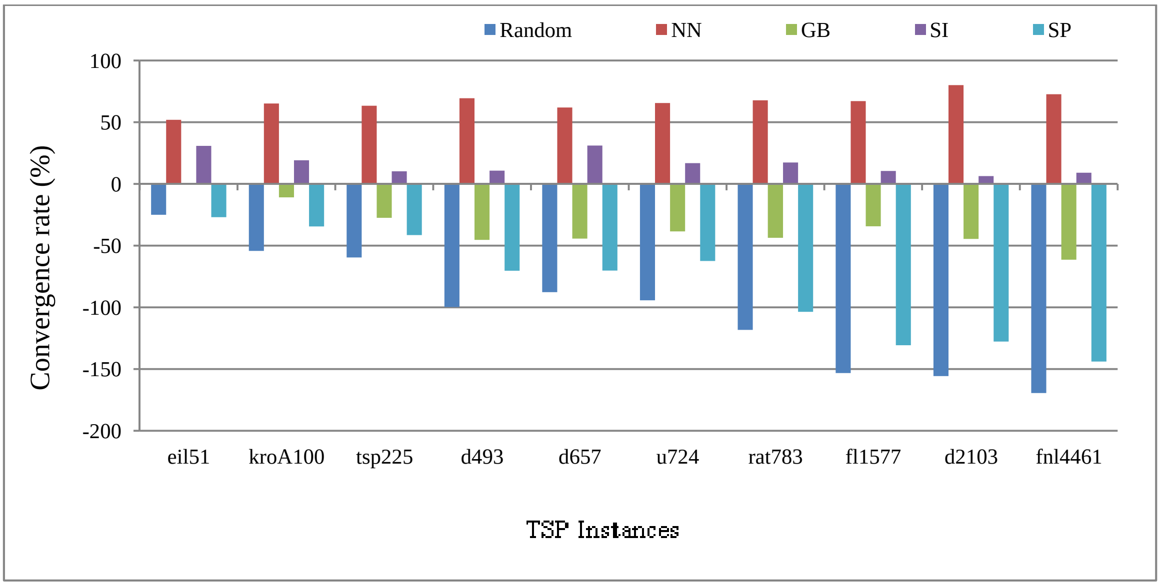

Figure 2. The Convergence Rate (%) of the worst solutions generated by different population seeding techniques

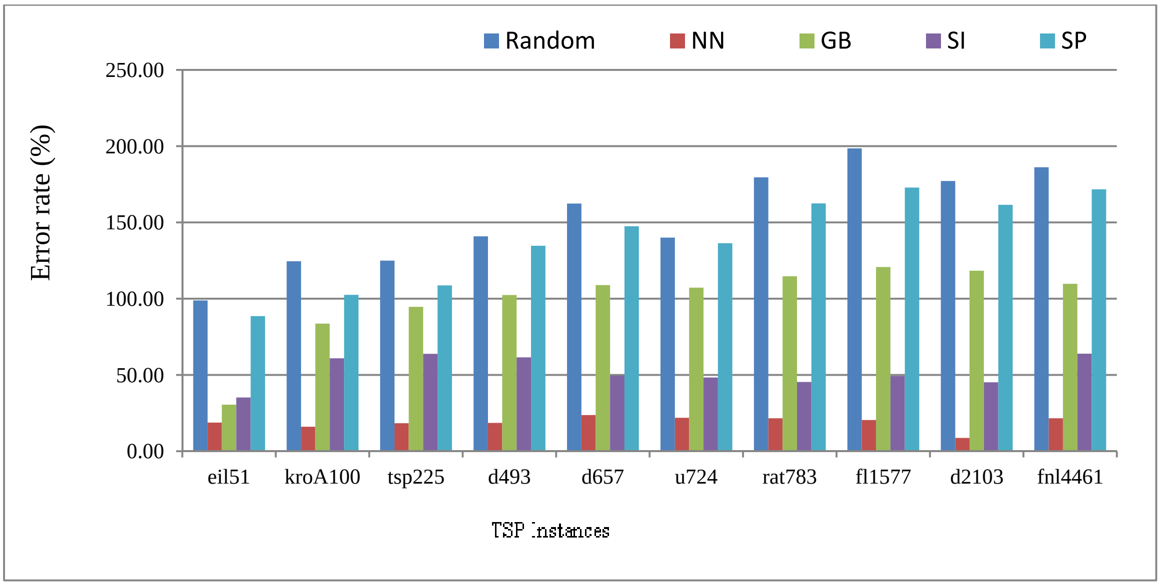

Figure 3. The Error Rate (%) of the best solutions generated by different population seeding techniques

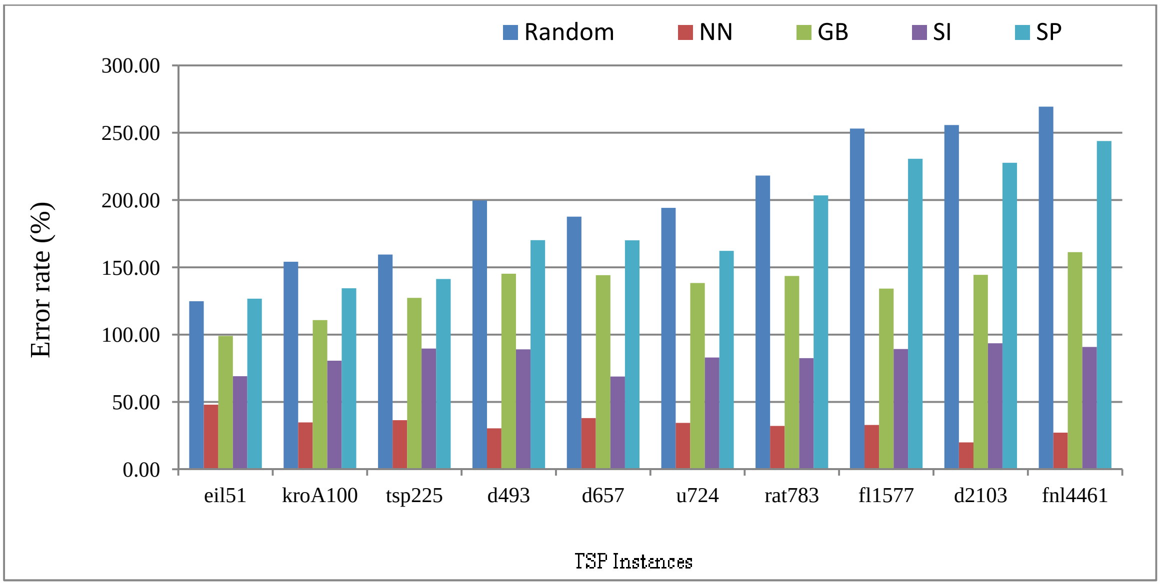

Figure 4. The Error Rate (%) of the worst solutions generated by different population seeding techniques

To summarize, the NN technique performs better than all other considered population seeding techniques followed by SI and then GB in terms of Convergence rate (%) and Error Rate (%). In case of Random and SP, the convergence rate (%) value for most of the instances are in negative which shows that the quality of solution generated are of bad quality.

V. Conclusion

In this paper, the performance of different population seeding techniques for the permutation coded genetic algorithm based on the quality of individuals generated has been analysed. Experiments are performed using the famous Travelling Salesman Problem (TSP) instances obtained from the TSPLIB, the standard library for TSP problems, using MATLAB tool. The experimental results show that the NN technique generates solutions with high fitness followed by SI and then GE techniques. The fitness of individuals generated by random and SP techniques are of bad quality in terms of convergence rate (%) and error rate (%). As a future work, analyses can also be performed in terms of performance factors such as Computation Time, Convergence Diversity and Average Convergence.

ACKNOWLEDGEMENT

This work is a part of the Research Projects sponsored under the Major Project Scheme, UGC, India, Reference Nos: F. No. 40-258/2011(SR), dated 29 June 2011 and F. No. 41-639/2012(SR)/Dt. 16-07-2012. The authors would like to express their thanks for the financial supports offered by the Sponsored Agency.

REFERENCES

[1] Murat Albayrak and Novruz Allahverdi, "Development a new mutation operator to solve the Traveling Salesman Problem by aid of Genetic Algorithms", Expert Systems with Applications, Issue 38, July 2011, pp.1313-1320, ISSN: 0957-4174.
[2] Helsgaun, K. (2000). An effective implementation of the Lin-Kernighan traveling salesman heuristic. European Journal of Operational Research, 126(1), 106–130.
[3] Shubhra S.R., Sanghamitra B, Sankar K.P.," Genetic operators for combinatorial optimization in TSP and microarray gene ordering", Journal of Applied Intelligence, Kluwer Academic Publishers, Volume 26 Issue 3, June 2007, pp.183 - 195.
[4] Lenstra, J.K, Rinnooy Kan, A.H.G. (1975). Some simple applications of the traveling salesman problem. Operational Research Quarterly, Vol. 26, pp. 717–33.
[5] Grötschel, M.; M. Jünger & G. Reinelt (1991). Optimal Control of Plotting and Drilling Machines: A Case Study. Mathematical Methods of Operations Research, Vol. 35, No. 1, (January, 1991), pp.61-84
[6] Plante, R.D.; T.J. Lowe & R. Chandrasekaran (1987). The Product Matrix Traveling Salesman Problem: An Application and Solution Heuristics. Operations Research, Vol. 35, pp. 772-783.
[7] Bland, R.E., & D.E Shallcross (1989). Large traveling salesman problem arising from experiments in X-ray crystallography: a preliminary report on computation. Operations Research Letters, Vol. 8(3), pp. 125-128.
[8] Ratliff, H.D. & A.S. Rosenthal (1983). Order-Picking in a Rectangular Warehouse: A Solvable Case for the Travelling Salesman Problem. Operations Research, Vol. 31, pp. 507-521.
[9] Fiechter, L. (1994) "A Parallel Tabu Search Algorithm for Large Travelling Salesman Problems". Discrete Applied Mathematics and Combinatorial Operations Research and Computer Science, 51 (3), 243–267.
[10] P. Dhavachelvan and G.V.Uma, "Complexity Measures For Software Systems: Towards Multi-Agent Based Software Testing Proceedings - 2005 International Conference on Intelligent Sensing and Information Processing, ICISIP'05 2005, Art. no. 1529476 , pp. 359-364.
[11] Freisleben, B., Merz, P., 1996. Genetic local search algorithm for solving symmetric and asymmetric traveling salesman problems. In: Proceedings of the IEEE Conference on Evolutionary Computation. IEEE Press, Nagoya, pp. 616–621.
[12] Potvin, J.Y., 1996. Genetic algorithm for the traveling salesman problem. Annals of Operations Research 63, 339–370.
[13] Tsai CF, Tsai CW, Yang T (2002) A modified multiple-searching method to genetic algorithms for solving traveling salesman problem. In: IEEE int conf systems, Man and cybernetics, vol. 3, pp 6–9
[14] Semya Elaoud, Jacques Teghem,Taıcir Loukil ,"Multiple crossover genetic algorithm for the multiobjective traveling salesman problem", Electronic Notes in Discrete Mathematics, Elsevier, Issue 36, pp.939-946, May 2010, ISSN:1571-0653.
[15] Ray SS, Bandyopadhyay S, Pal SK (2004) New operators of genetic algorithms for traveling salesman problem. Cambridge, UK, ICPR- 04 2:497–500
[16] Murat Albayrak and Novruz Allahverdi, "Development a new mutation operator to solve the Traveling Salesman Problem by aid of Genetic Algorithms", Expert Systems with Applications, Issue 38, July 2011, pp.1313-1320, ISSN: 0957-4174.
[17] Chang, P. C., Huang, W. S., & Ting, C. J. (2010). Dynamic diversity control in genetic algorithm for mining unsearched solution space in TSP problems. Expert Systems with Applications, 37(3), 1863–1878.
[18] Dorigo, M., Gambardella, L.M., 1997. Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Transactions on Evolutionary Computation 1 (1), 53–56.
[19] Stutzle T, DorigoM(1999) ACO algorithms for the traveling salesman problem, evolutionary algorithms in engineering and computer science. John Wiley and Sons
[20] Ellabib, I., Calamai, P., & Basir, O. (2007). Exchange strategies for multiple ant colony system. Information Sciences, 177(5), 1248–1264
[21] Li, L., Ju, S., & Zhang, Y. (2008). Improved ant colony optimization for the traveling salesman problem. In Proceedings of the 2008 international conference on intelligent computation technology and automation (Vol. 1, pp. 76–80)
[22] Shi, X., Wang, L., Zhou, Y., & Liang, Y. (2008). An ant colony optimization method for prize-collecting traveling salesman problem with time windows. In Proceedings of the fourth international conference on natural computation, Jinan, China (Vol. 7, pp. 480–484).
[23] Chen, S. M., & Chien, C. Y. (2010). A new method for solving the traveling salesman problem based on the genetic simulated annealing ant colony system with particle swarm optimization techniques. In Proceedings of the 2010 international conference on machine learning and cybernetics, Qingdao, Shandong, China (pp. 2477–2482).
[24] Potvin, J.Y., 1993. The traveling salesman problem: a neural network perspective. ORSA Journal on Computing 5 (4), 328–348.
[25] Paul P.V., T. Vengattaraman, P. Dhavachelvan, "Improving efficiency of Peer Network Applications by formulating Distributed Spanning Tree", Proceedings - 3rd International Conference on Emerging Trends in Engineering and Technology, ICETET 2010, Art. no. 5698439, pp. 813-818.
[26] Xiutang Geng, Zhihua Chen, Wei Yang, Deqian Shi, Kai Zhao,"Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search",Applied Soft Computing, Volume 11, Issue 4, June 2011, Pages 3680-3689.
[27] P. Dhavachelvan and G.V.Uma, "Reliability Enhancement in Software Testing: An Agent-Based Approach for Complex Systems", 7th ICIT 2004, Springer Verlag - Lecture Notes in Computer Science (LNCS), Vol. 3356, pp. 282-291. ISSN: 0302-9743.
[28] Marinakis, Y., &Marinaki,M. (2010). A hybrid genetic – Particle swarm optimization algorithm for solving the vehicle routing problem. Expert Systems with Applications, 37(2), 1446–1455
[29] Chen, S. M., & Chien, C. Y. (2010). A new method for solving the traveling salesman problem based on the genetic simulated annealing ant colony system with particle swarm optimization techniques. In Proceedings of the 2010 international conference on machine learning and cybernetics, Qingdao, Shandong, China (pp. 2477–2482).
[30] Shyi-Ming Chen and Chih-Yao Chien, "Solving the traveling salesman problem based on the genetic simulated annealing ant colony system with particle swarm optimization techniques", Expert Systems with Applications, Elsevier, Issue 38, May 2011, pp. 14439–14450, ISBN: 0957-4174.
[31] P. Victer Paul, R. Baskaran, P. Dhavachelvan, "A Novel Population Initialization Technique for Genetic Algorithm", IEEE International Conference on Circuit, Power and Computing Technologies (ICCPCT), March 2013, India, pp 1235 - 1238. ISBN: 978-1-4673-4921-5.

[32] P. Dhavachelvan and G.V.Uma, “Multi-agent based Framework for Intra-Class Testing of Object-Oriented Software”, International Journal on Applied Soft Computing, Elsevier, Vol-5, No.2, 2005, pp. 205-222.

[33] Pullan, W.," Adapting the Genetic Algorithm to the Travelling Salesman Problem", The 2003 Congress on Evolutionary Computation, CEC'03, Volume: 2, 2003. pp.1029-1035.

[34] K. Katayama, H. Sakamoto and H. Narihisa , "The Efficiency of Hybrid Mutation Genetic Algorithm for the Travelling Salesman Problem ", Mathematical and Computer Modelling, Elsevier, Issue 31, pp.197-203, October 2000.

[35] Lawrence J. Schmitt and Mohammad M. Amini, "Performance characteristics of alternative genetic algorithmic approaches to the traveling salesman problem using path representation: An empirical study", European Journal of Operational Research, Elsevier, Issue 108, April 1998, pp.551-570, ISSN: 0377-2217/98

[36] Olga Yugay, Insoo Kim, Beomjune Kim, Franz I.S. Ko ,"Hybrid genetic algorithm for solving Traveling Salesman Problem with sorted population", Third IEEE International Conference on Convergence and Hybrid Information Technology (ICCIT 2008), ISSN:978-0-7695-3407.

[37] S. Venkatesan, P. Dhavachelvan and C. Chellapan, “Performance analysis of mobile agent failure recovery in e-service applications”, International Journal of Computer Standards and Interfaces, Elsevier, Vol-32, No.1-2, 2005, pp. 38 43. ISSN: 0920-5489.

[38] Website: http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/. Last accessed on: 21st Sept. 2012.

[39] Lu, Y., Lu, S., Fotouhi, F., Deng, Y. and Brown, S, "Incremental Genetic K-means Algorithm and Its Application," Gene Expression Data Analysis, Submitted to BMC bioinformatics, 2004.

[40] Vedat Togan and Ayse T. Daloglu," An improved genetic algorithm with initial population strategy and self-adaptive member grouping", Journal of Computers and Structures, Volume 86, Issues 11–12, June 2008, pp. 1204-1218, ISSN 0045-7949.

[41] Liangsheng Qu and Ruixiang Sun, "A synergetic approach to genetic algorithms for solving traveling salesman problem", Information Sciences, Elsevier, Issue 117, pp.267-283, July 1999, ISSN: 0020-0255.

[42] P. Victer Paul, N. Saravanan, S.K.V. Jayakumar, P. Dhavachelvan and R. Baskaran (2012), “QoS enhancements for global replication management in peer to peer networks”, Future Generation Computer Systems 28 (3), pp. 573-582.

[43] Reinelt G (1994) The traveling salesman: computational solutions for TSP applications. Lecture notes in computer science, Springer Verlag 840

[44] Bentley JL (1992) Fast algorithms for geometric traveling salesman problems. ORSA J Computing 4(4):387–411

[45] Johnson D S, McGeoch LA(1996) The traveling salesman problem: a case study in local optimization. Local search in combinatorial optimization. Wiley and Sons, New York

[46] Rong Yang, "Solving Large Travelling Salesman Problems with Small Populations", Genetic Algorithms in Engineering Systems: Innovations and Applications, Conference Publication No. 446, 2-4 September 1997, pp.157-162.

[47] Saadatmand-Tarzjan, M., Khademi, M., Akbarzadeh-T., M.-R., & Moghaddan, H. A. (2007). A novel constructive-optimizer neural network for the traveling salesman problem. IEEE Transactions on Systems, Man, Cybernetics – Part B: Cybernetics, 37(4), 754–770.

[48] Ling-Ning Xing, Ying-WuChen, Ke-WeiYang, FengHou, Xue-ShiShen and Huai-PingCai, "A hybrid approach combining an improved genetic algorithm and optimization strategies for the asymmetric traveling salesman problem", Engineering Applications of Artificial Intelligence, Elsevier, Issue 21, April 2008, pp.1370-1380, ISSN:0952-1976.

[49] Tsai CF, Tsai CW, Yang T (2002) A modified multiple-searching method to genetic algorithms for solving traveling salesman problem. In: IEEE int conf systems, Man and cybernetics, vol. 3, pp 6–9.

[50] P. Victer Paul, D. Rajaguru, N. Saravanan, R. Baskaran and P. Dhavachelvan, "Efficient service cache management in mobile P2P networks", Future Generation Computer Systems, Elsevier, Volume 29, Issue 6, August 2013, Pages 1505–1521.

[51] Yingzi Wei, Yulan Hu, Kanfeng Gu," Parallel Search Strategies for TSPs using a Greedy Genetic Algorithm", Third International Conference on Natural Computation, ICNC 2007, Volume: 3, August 2007,pp.786-790.