An improved multi-objective and colony algorithm

Jiang Fei[1, 2]

*[1]Laboratory of Intelligent Information Processing,
Suzhou University, Suzhou 234000, Anhui,China*
*[2]School of Information Engineering,
Suzhou University, Suzhou 234000, Anhui,China*

Abstract
Multi-objective optimization is to optimize multiple objectives, which usually conflict with each other. As one of the artificial intelligent algorithms, ant colony algorithm has been widely applied in solving combinational optimization problems. On account of the design concept of the basic ant colony algorithm, multi-objective ant colony optimization chooses the optimal solution by integrating Pareto dominance relation. This paper proposes an improved multi-objective ant colony algorithm, adopts the colony movement search and sets such strategies as the movement rules and the set update rules to search the set of the optimal solutions. Through the analysis of the experimental simulation, it can be seen that the Pareto front-end solutions obtained by the algorithm of this paper have excellent distributivity and similar convergence indexes and there is no big difference in the distributivity indexes of the solutions obtained, which further proves the performance and application value of multi-objective ant colony algorithm.
Key words: ANT COLONY ALGORITHM, MULTI-OBJECTIVE OPTIMIZATION, PARETO SOLUTION SET

Introduction

Plenty of optimization problems in scientific research, engineering practice, social production and economic development are multi-objective optimization problems. The idea of multi-objective optimization problem originates from the Utility Theory in the Economics in 1776. In 1896, the research of economic balance first brought forth the multi-objective optimization problem and introduced the concept of Pareto optimal solution set. T.Ckoopmans, a mathematical economist introduced the definition of efficient solution and he had made some fundamental achievements in 1951, laying a preliminary foundation for the discipline of multi-objective optimization[1]. Since the 1960s, people have designed numerous methods to solve multi-objective optimization problems, used these methods to settle various problems in reality and made some achievements. J.L.Cohrance and M.Zeleny edited and published the first “Multi-Objective Decision” in 1973, which greatly promoted the formation and development of multi-objective optimization. After that, the research on multi-objective optimization problems has rapidly developed in theory and application[2]. At present, the decision makers of different levels as well as the scientific and technical workers have paid their attention and interest to the application of multi-objective optimization. Besides, its application scope is becoming wider and wider and the problem scale to be solved is also becoming bigger and bigger. Multi-objective problem is also involved in Economics, Cybernetics and Systems Engineering[3].

Evolutionary algorithm is a random search algorithm, which searches the optimal solution by simulating the evolutionary theory of survival of the fittest in the natural world. The evolutionary algorithm is divided into single-objective and multi-objective evolutionary algorithms. The algorithm is quite simple when solving single-objective optimization problems. Multi-objective optimization involves minimizing or maximizing multiple objective functions subject to a set of constraints. Since the objectives in the Multi-objective Optimization Problem (MOP) can't be compared with and even conflict with each other, it may not be possible to obtain the optimal solution in all the objectives. In order to search the optimization of the overall objective, comprehensive consideration shall be taken to the conflicting objectives and a halfway method is adopted. Therefore, the Multi-Objective Evolutionary Algorithm (MOEA) appeared as a result for the optimization problems with different objectives[4]. People have designed evolutionary algorithms based on the rules of the biological evolution in the natural world. For example, such algorithms as genetic algorithm, simulated annealing algorithm, neural network, ant colony algorithm and particle swarm optimization have been continuously investigated by the scholars to solve the complicated optimization problems in more fields[5].

An important task in multiobjective optimization is to identify a set of optimal trade-off solutions (called a Pareto set) between the conflicting objectives, which helps gain a better understanding of the problem structure and supports the decision-maker in choosing the best compromise solution for the considered problem[6]. Because the ant-colony works on a very dynamic system, the ant colony algorithm works very well in graphs with changing topologies. Examples of such systems include computer networks, and artificial intelligence simulations of workers[7]. In the meanwhile, with better generality, ant colony algorithm is suitable to handle various types of objective functions and constraints and it is also easy to combine with traditional optimizations so as to improve its own limitations and solve the problems in a more efficient way. Therefore, it has huge advantages to apply ant colony algorithm to solve multi-objective optimization problem[8]. This paper proposes an improved multi-objective ant colony optimization. Firstly, it elaborates the basic principles of multi-objective optimization problem and ant colony algorithm. Then, it presents the implementation steps and strategies of multi-objective ant colony algorithm. Finally, it proves the convergence and distributivity of the algorithm of this paper through numerical experiment.

Description of Multi-Objective Optimization Problem

Optimization problems with multiple, often conflicting, objectives arise in a natural fashion in most real-world applications, such as aerodynamic design, financial decision making, and electronic circuit development[9]. Multi-objective optimization is to optimize multiple objectives, which usually conflict with each other. Example problems include analyzing design tradeoffs, selecting optimal product or process designs, or any other application where you need an optimal solution with tradeoffs between two or more conflicting objectives[10]. Literally, multi-objective optimization problem is formed by D decision variable parameters, N objective functions and $m+n$ constraint conditions and the decision variable has a functional relation with the objective function and the constraint conditions. In the non-inferior solution set, the decision makers can only choose one satisfactory non-inferior solution as the final solution according to the concrete problem[11].

Taking the minimization of multi-objective optimization problem as example, the multi-objective optimization problem usually involves the following aspects[12,13]:

(1) Pareto domination: if one decision vector x_1 dominates another decision vector x_2, mark them as: $x_1 \prec x_2$, when and only when $f_i(x_1) \le f_i(x_2), (i = 1, 2, \cdots, m)$ and $\exists i \in \{1, 2, \cdots, m\}$, $f_i(x_1) < f_i(x_2)$, in other words, when an objective vector f_1 dominates another objective vector f_2, if f_1 is not inferior to f_2 and there is at least one objective value, which is superior to f_2. Then mark the objective vector domination as $f_1 \prec f_2$.

(2) Pareto optimal solution: Reduces the values of a linear or nonlinear vector function to attain the goal values given in a goal vector. The relative importance of the goals is indicated using a weight vector. Goal attainment problems may also be subject to linear and nonlinear constraints. If x_1 is the Pareto optimal and when and only when there is no $x_2 \succ x_1$, namely there is no k, then $f_k(x_2) \le f_k(x_1)$.

(3) Pareto optimal solution set: Both goal attainment and minimax problems can be solved by transforming the problem into a standard constrained optimization problem and then using an active-set approach to find the solution. The set of all Pareto optimal solutions, namely $P_s = \left\{ x_1 \mid \bar{\exists} x_2 \succ x_1 \right\}$, is also called non-inferior optimal solution set, namely Pareto optimal solution set.

(4) Pareto optimal front end: Minimizes the worst-case values of a set of multivariate functions, possibly subject to linear and nonlinear constraints. P_F is the region formed by the objective functional values corresponding to all Pareto optimal solutions, namely $P_F = \{ f(x) = (f_1(x), f_2(x), \cdots, f_m(x)) \mid x \in P_S \}$.

Basic Principle of Ant Colony Algorithm

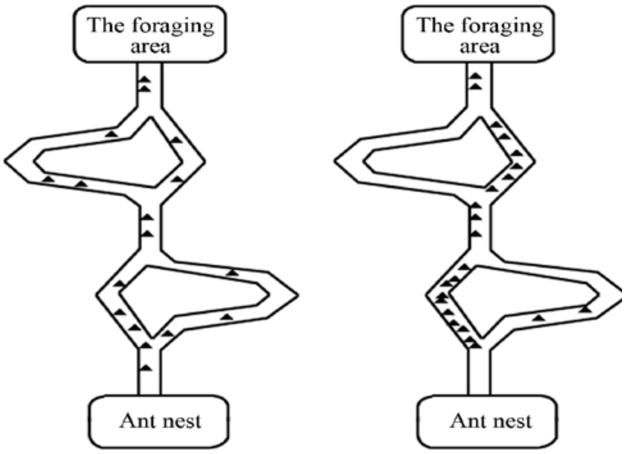


Figure 1. Equal probability selection Figure 2. The optimal path

The ant colony algorithm is an algorithm for finding optimal paths that is based on the behavior of ants searching for food. The ants leave a substance called pheromone along the path it passes in the foraging. They can feel the concentration of pheromone and guide their direction in the foraging. In fact, they usually move towards the direction with high pheromone concentration; therefore, the collective foraging formed by numerous ants displays a positive feedback of pheromone. If one path is short, more ants will pass along it, more pheromone will be left on it; the pheromone concentration is higher and the possibility for the ant to choose this path is higher. Therefore, a positive feedback is formed before it gradually approximates the optimal path[14].

The ants wander randomly. When an ant finds a source of food, it walks back to the colony leaving "pheromones markers" that show the path has food. When other ants come across the markers, they are likely to follow the path with a certain probability. If they do, they then populate the path with their own markers as they bring the food back. As more ants find the path, it gets stronger until there are a couple streams of ants traveling to various food sources near the colony. The Fig.1 and Fig.2 is ant colony’s equal probability selection and the optimal path[15].

The Procedures of Improved Multi-Objective Ant Colony Optimization

Different from single-objective optimization, multi-objective optimization has no absolute optimal solution but the relatively superior or inferior solutions, therefore, the pheromone amount released by the ants in this paper are determined by the superiority or inferiority of the solution and preserve the optimal solutions found by every loop to the optimal expenience set. The main strategies of the algorithm[16,17]:

(1)Colony movement search

Initially every path between points has some initial amount of pheromone. Each ant starts from a randomly assigned point and goes from a point to the next point until all points are visited exactly once. From the last visited point the ant returns to the start city.

Each ant located at point i hops to a point j selected among the point s that have not yet been visited according to the probability:

$$p_k(i,j) = \begin{cases} \dfrac{\tau_{ij}^{\alpha} * d_{ij}^{\beta}}{\sum_{g \in J_k^{(i)}} \tau_{ig}^{\alpha} * d_{ig}^{\beta}} \\ 0 \end{cases} \qquad (1)$$

Where $p_k(i,j)$ is the probability that ant k in point i will go to point j, $j \in J_k^{(i)}$ is the set of cities that have not yet been visited by ant k in point i, α is the relative importance of the pheromone trail, β is the relative importance of the distance between points.

Therefore the probability that a point is chosen is a function of how close the point is and how much pheromone already exists on that trail. It is further possible to determine which of these has a larger weight by tweaking with the α and β parameters. Once a tour has been completed (i.e. each point has been visited exactly once by the ant), pheromone evaporation the edges is calculated. Then each ant deposits pheromone on the complete tour by a quantity which is calculated from the following formula [18,19,20].

(2) Pheromone Update

The ants update pheromones on paths connecting the points according to the formula:

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^{k-m} \Delta\tau_{ij}^{k} \quad (2)$$

Where m is number of ants,

$\Delta\tau_{ij}^{k} = \frac{Q}{L_k}$ if the ant k traveled the path c_{ij}^{k} between points i and j, Q is some constant, and L_k is the length of the kth ant's travel, Q is a point's extent.

$\Delta\tau_{ij}^{k} = 0$, 0 otherwise.

(3) Evaporation

In terms of colony movement search method, after determining the movement direction, move the distance of $P_j * d_{ij}$ towards the objective ant. After all ants complete their nth trip, evaporation is applied to all paths between points:

$$\tau_{ij}^{n} \leftarrow (1-\rho) * \tau_{ij}^{n} \quad (3)$$

Here $\rho \subseteq (0,1]$ is the evaporation factor.

(4) Mutation operation

Randomly choose one solution (path) in the current solution group, determine mutation rate p_m whether undergo mutation. If meet the mutation rate , then randomly selecte a path ij as the mutation point, with randomly generated the value τ_{ij} that meet the conditions of $\tau_{\min} \le \tau_{ij} \le \tau_{\max}$, to be the pheromone concentration of the path. The mutation operation of pheromone concentration can provide new possibility for the path choice of search process, thus make the algorithm solution keep population diversity during search process, solve the problem of search stagnation. Fig.3 is multi-objective ant colony algorithm flowchart.

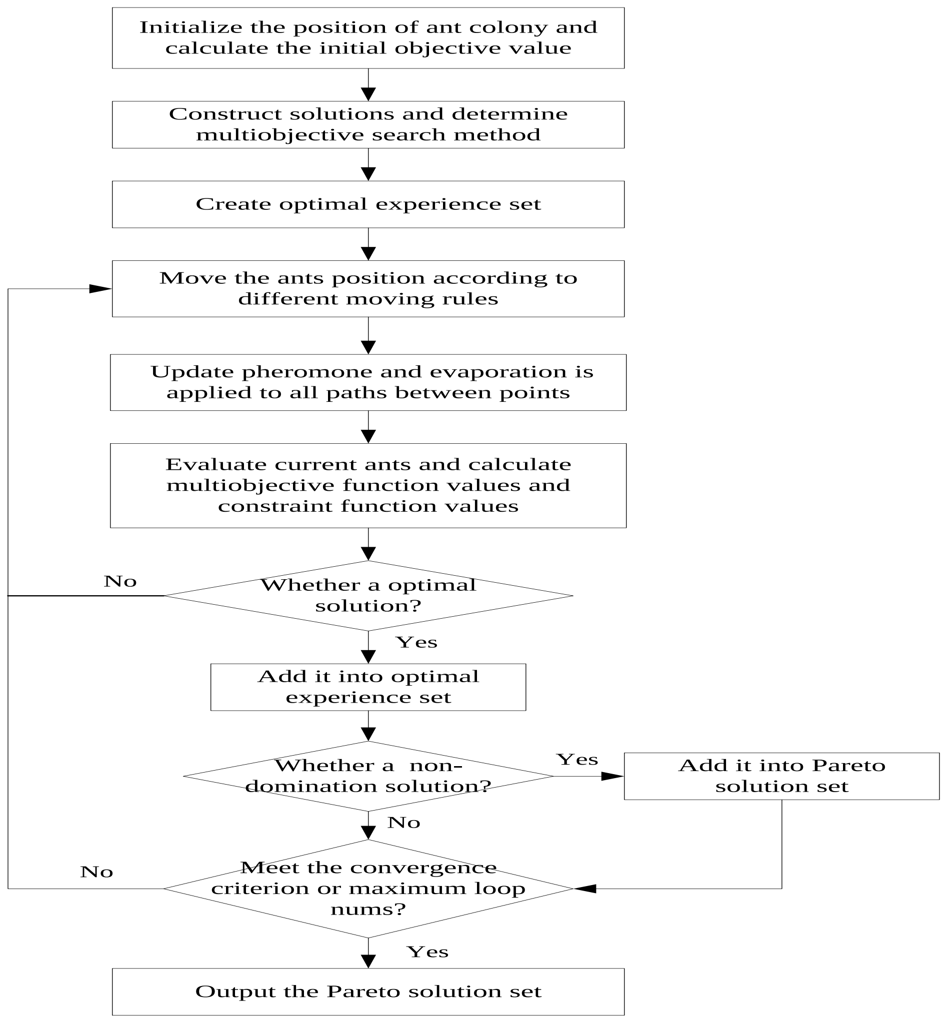


Figure 3. The flowchart of multi-objective ant colony algorithm

Experimental Simulation and Analysis

The following experiments are made on the three-objective standard test functions DTLZ-2 and DTLX-7. Implement the compilation of the algorithm through Matlab mathematical software.

(1) Test function 2: DTLZ-2

$$\begin{aligned} &\min f(x) = (f_1(x), f_2(x), f_3(x)) \\ &f_1(x) = \cos(x_1 \cdot \pi/2)\cos(x_2 \cdot \pi/2)(1+g(x_k)) \\ &f_2(x) = \cos(x_1 \cdot \pi/2)\sin(x_2 \cdot \pi/2)(1+g(x_k)) \\ &f_3(x) = \sin(x_1 \cdot \pi/2)(1+g(x_k)) \\ &g(x_k) = \sum_{i=2}^{d}(x_1 - 0.5)^2 \end{aligned} \quad (4)$$

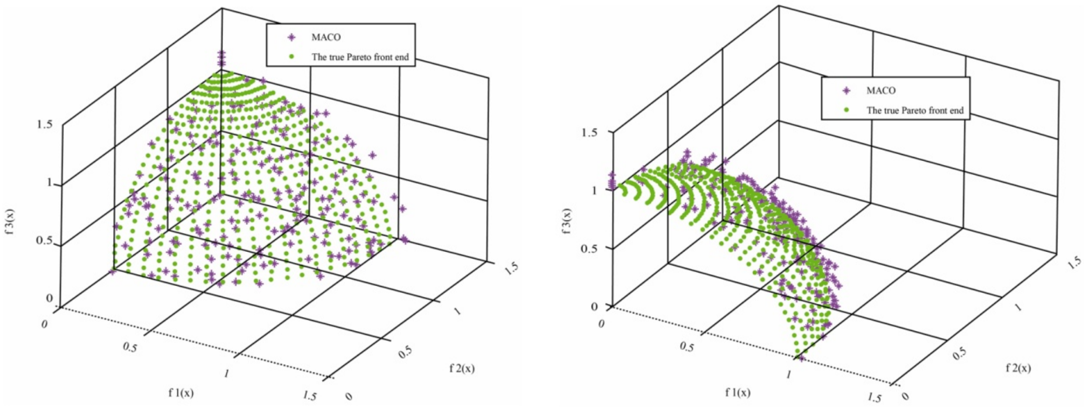
In here, $k = 3, d = 12, x \in [0,1]$.

(2) Test function 1: DTLZ-7

$$\begin{aligned} &\min f(x) = (f_1(x), f_2(x), f_3(x)) \\ &f_1(x) = x_1 \\ &f_2(x) = x_2 \\ &f_3(x) = (1+g(x_k))h(f_1, f_2, g) \\ &g(x_k) = 1 + \frac{9}{|x_k|}\sum_{x_i \in x_k} x_i \\ &h(f_1, f_2, g) = k - \sum_{i=1}^{k-1}\left[\frac{f_i}{1+g}(1+\sin(3\pi f_i))\right] \end{aligned} \quad (5)$$

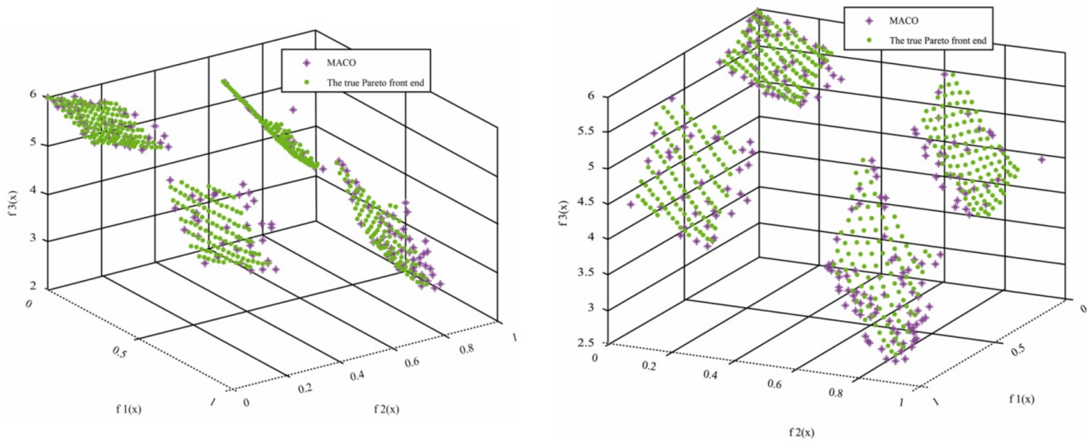
In here, $k = 3, d = 22, x \in [0,1]$.

The comparison of the qualitative experiments of the test functions of DTLZ-2 and DTLZ-7 is indicated in Figure 4 and Figure 5.



(a) (b)

Figure 4. The comparison between of pareto front-end of DTLZ-2 obtained by MACO and real front-end



(c) (d)

Figure 5. The comparison between of pareto front-end of DTLZ-7 obtained by MACO and real front-end

Through the computational results, it can be seen that the average distance between the Pareto front end obtained from the improved multi-objective ant colony algorithm and the real Pareto front end is small and that the solution has excellent convergence. Additionally, the algorithm of this paper is excellent in the distributivity of the solutions to the multi-objective optimization problems and it proves the effectiveness of the multi-objective ant colony optimization in this paper.

Conclusion

In the practical applications and the scientific research, multi-objective optimization is an extremely important research topic. Because many problems in real life have involved in simultaneous optimization of multiple objectives, the research topic of multi-objective optimization is receiving more and more extensive attention. The characteristics of superiority and robustness demonstrated in solving complicated problems make the multi-objective optimization an excellent solution in optimizing multi-objective problems. Based on the fact that the advantages and characteristics of ant colony algorithm still have a huge research space in solving multi-objective optimization problem, this paper has proposed an improved multi-objective ant colony optimization to enhance the multi-objective optimization performance of ant colony algorithm and make up for the deficiencies of the original ant colony algorithm through the combination with other methods and mechanisms. The experiment has verified that the algorithm of this paper is very effective in settling multi-objective problems and expanded the idea to solve multi-objective optimization problems.

Acknowledgements

The work described in this paper was sponsored in part by the Anhui Provincial Projects of Science Research in Colleges Universities (Grant No.KJ2014A247 and No.KJ2014ZD31) and Collaborative innovation center regional development of Suzhou open project (Grant No. 2014SZXTKF17) .

References

1. Juan Rada-Vilela, Manuel Chica, Óscar Cordón, Sergio Damas (2013) A Comparative Study of Multi-Objective Ant Colony Optimization Algorithms for The Time and Space Assembly Line Balancing Problem. *Applied Soft Computing*, 13(11), p.p. 4370-4382.
2. Yi-Jun He, Zi-Feng Ma (2014) Optimal Design of Linear Sensor Networks for Process Plants: A Multi-Objective Ant Colony Optimization Approach. *Chemometrics and Intelligent Laboratory Systems*, 135(15) , p.p. 37-47.
3. Duško Kančev, Marko Čepin, Blaže Gjorgiev (2014) Development and Application of A Living Probabilistic Safety Assessment Tool: Multi-objective Multi-dimensional Optimization of Surveillance Requirements in NPPs Considering Their Ageing. *Reliability Engineering & System Safety*, 131(11), p.p. 135-147.
4. Jan Hettenhausen, Andrew Lewis, Timoleon Kipouros (2014) A Web-Based System for Visualisation-Driven Interactive Multi-Objective Optimisation. *Procedia Computer Science*, 29(3), p.p. 1915-1925.
5. Samira Fazlollahi, Gwenaelle Becker, François Maréchal (2014) Multi-objectives, Multi-Period Optimization of District Energy Systems: II—Daily Thermal Storage. *Computers & Chemical Engineering*, 71(4), p.p. 648-662.
6. Matthias Ehrgott, Jonas Ide, Anita Schöbel (2014) Minmax Robustness for Multi-Objective Optimization Problems. European Journal of Operational Research, 239(1) , p.p. 17-31.
7. Anna Syberfeldt, Patrik Gustavsson. Increased Robustness of Product Sequencing using Multi-Objective Optimization. Procedia CIRP. 2014; 17(6), p.p. 434-439.
8. G.A. Vijayalakshmi Pai, Thierry Michel (2014) Metaheuristic Multi-Objective Optimization of Constrained Futures Portfolios for Effective Risk Management. *Swarm and Evolutionary Computation,* 19(12) , p.p. 1-14.
9. Koushik Venkata Narasimha, Elad Kivelevitch, Balaji Sharma, Manish Kumar (2013) An Ant Colony Optimization Technique for Solving Min–Max Multi-Depot Vehicle Routing Problem. *Swarm and Evolutionary Computation*, 13(12), p.p. 63-73.
10. Luis A. Moncayo-Martínez, Gustavo Recio (2014) Bi-criterion Optimisation for Configuring An Assembly Supply Chain using Pareto Ant Colony Meta-Heuristic. *Journal of Manufacturing Systems*, 33(1), p.p. 188-195.
11. Ali Roozbeh Nia, Mohammad Hemmati Far, Seyed Taghi Akhavan Niaki (2014) A Fuzzy Vendor Managed Inventory of Multi-Item Economic Order Quantity Model under Shortage: An Ant Colony Optimization Algorithm. *International Journal of Production Economics*, 155(9), p.p. 259-271.

12. Thirachit Saenphon, Suphakant Phimoltares, Chidchanok Lursinsap (2014) Combining New Fast Opposite Gradient Search with Ant Colony Optimization for Solving Travelling Salesman Problem. *Engineering Applications of Artificial Intelligence*, 35(10), p.p. 324-334.
13. R.F. Tavares Neto, M. Godinho Filho (2013) Literature Review Regarding Ant Colony Optimization Applied to Scheduling Problems: Guidelines for Implementation and Directions for Future Research. *Engineering Applications of Artificial Intelligence*, 26(1), p.p. 150-161.
14. A.A. Mousa, I.M. El_Desoky (2013) Stability of Pareto Optimal Allocation of Land Reclamation by Multistage Decision-Based Multipheromone Ant Colony Optimization. *Swarm and Evolutionary Computation*, 13(12) , p.p. 13-21.
15. T.A. Sebaey, C.S. Lopes, N. Blanco, J.A. Mayugo, J. Costa (2013) Two-Pheromone Ant Colony Optimization to Design Dispersed Laminates for Aeronautical Structural Applications. *Advances in Engineering Software,* 66(12) , p.p. 10-18.
16. Luis A. Moncayo-Martínez, David Z. Zhang (2013) Optimising Safety Stock Placement and lead Time in An Assembly Supply Chain using Bi-Objective MAX–MIN Ant System. *International Journal of Production Economics*, 145(1) , p.p. 18-28.
17. Poonam Savsani, R.L. Jhala, Vimal Savsani (2014) Effect of Hybridizing Biogeography-Based Optimization (BBO) Technique with Artificial Immune Algorithm (AIA) and Ant Colony Optimization (ACO). *Applied Soft Computing,* 21(8) , p.p. 542-553.
18. M. Dorigo, V. Maniezzo, A. Colorni (1991) *Positive Feedback as A Search Strategy*. Technical Report 91-016. Dipartimento di Elettronica, Politecnico di Milano, Milan, Italy.
19. M. Dorigo (1992) *Optimization, Learning and Natural Algorithms* (in Italian). PhD Thesis. Milan, Italy & Dipartimento di Elettronica, Politecnico di Milano.
20. M. Dorigo, V. Maniezzo, A. Colorni (1996) Ant System: Optimization by A Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics – Part B,* 26(1), p.p.29-41.