Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 173 (2020) 245–253

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on *Smart Sustainable Intelligent Computing and Applications* under ICITETM2020

Comparative Analysis of Ant Colony and Particle Swarm Optimization Algorithms for Distance Optimization

Arushi Gupta[a], Smriti Srivastava[b]

[a]*Accenture Technologies, Pune - 411028, India*
[b]*Division of Instrumentation and Control, Netaji Subhas University of Technology, New Delhi-110078, India*

Abstract

In this paper, we have solved one of the widely popular problems in the domain of control systems – distance optimization using two separate swarm intelligence algorithms – particle swarm and ant colony optimization. The problem is successfully solved with both algorithms and subsequently quantitative comparison between their performances is done. Simulated results show the more recently developed ant colony optimization algorithm to be better and more robust of the two. The requisite simulations are carried out and results are obtained in the MATLAB environment.

Keywords: Ant colony optimization; Particle swarm optimization; Pheromone; Swarm intelligence; Distance optimization.

1. Introduction

Swarm intelligence (SI) basically deals with coordination using decentralized control and self-organization of many individuals of any natural and artificial systems. Or in other words these techniques are based on collective behaviours decided by the individuals with each other (local interactions) and with their environment. To name a few swarm intelligence techniques - colonies of ants and termites, schools of fish, flocks of birds, herds of land animals [1, 2]. Among these Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO) are the most popular optimization problems. Every algorithm has its own advantages and disadvantages and is best suited for particular

1877-0509
10.1016/j.procs.2020.06.029

type of problems. Any optimization algorithm is said to be effective if it converges to global minima in less number of iterations.

In this paper a classic problem of control theory – distance optimization is done using the swarm intelligence techniques. Objective is to find shortest path or distance between two vertices. This can be further extended to traveling sales man problem.

Here comparative analysis of two different SI algorithms – ant colonization (ACO) and particle swarm optimization (PSO) is done for distance optimization in terms of plots for square of error. We aim to find shortest route from the initial starting point of (3,3) to the final destination of (5,5) on the coordinate axis. We know shortest distance between two points is a straight line Comparative analysis is done for ACO and PSO to find this shortest route.

ACO is a probabilistic method based on the behaviour of ants to find shortest path to their food. Ants are not so intelligent animals but the tend to find the shortest path by following the paths set by their fellow ants. Optimal path is found on which maximum number of ants travel. PSO is based on how a group of birds adapt their paths to find their food. It basically works in iterations, finding local minima in the group (pbest) and global minima attained so far (gbest).

The paper is organized as follows: Section 2 and 3 brief about PSO and ACO algorithms. Section 4 describes the problem and discusses the simulated results. Finally, section 5 concludes the paper followed by references.

2. Particle Swarm Optimization Algorithm

Particle swarm optimization (PSO), originally developed by Kennedy and Eberhart. In PSO each possible solution is called particle and group of these possible solutions is called population. PSO is based on how a group of birds will randomly search for food. Birds don't know where exactly the food is, so they follow the bird which is nearest to the food. Each bird is known as particle and each particle has its fitness function (here it is square of error). A group of particles is known as swarm. In each iteration, first the best solution found in the swarm is stored and is known as pbest. Another best solution gbest is also stored, which is the best solution found so far. Values of pbest and gbest are updated according to equations 1 and 2 known as velocity and distance equation respectively.

$$v(t+1) = wv(t) + c_1 r_1 \left[\hat{x}(t) - x(t)\right] + c_2 r_2 \left[g(t) - x(t)\right] \tag{1}$$

The distance equation is defined as

$$x(t+1) = x(t) + v(t+1) \tag{2}$$

Here, x is the position of particle at time t, v is the velocity of particle at time t, c1 is the acceleration constant for cognitive component, c2 is the acceleration constant for social component and r is the stochastic random constant [9-11].

The algorithm of PSO for finding the shortest path is described below [3-11]:

1) A swarm is initialized with N number of particles. Each particle has random position and velocity constants.

2) The paths of the particles are optimized after comparing them with the pbest and gbest paths.

3) If the path of the particle is shorter than its local best path, then the path of particle is updated as the new local best path. Then the comparisons are made with global best path.

4) Positions and velocities of particles are updated according to equations 1 and 2.

5) The steps are followed until the minimum required path is obtained.

3. Ant Colony Optimization Algorithm

Ant Colony Optimization (ACO) was first used by M. Dorigo et al. to solve discrete optimization problems, in the late 1980s. ACO is based on behavior of ants for finding food. Ants deposit pheromone as they walk and find their route by walking along the pheromone deposition. Density of pheromone deposition increases as ants walk back to source with food. Pheromone deposition on way back is dependent on quality and quantity of food taken to the source point. Pheromone deposition /evaporation is directly related to number of ants traveling on that path. Ants find the optimal path by following maximum pheromone deposition [12,13]. The general steps in ACO are explained in figure1.

Figure 2 shows the general behavior followed by the swarm based approach as far as distance optimization is considered. In our distance optimization problem pheromone, model decides the choice of solutions or paths in each iteration. These possible solutions are known as transition probabilities and are found as in equation 3.

$$p\left(\frac{c_i}{s}\right) = \frac{[\tau_i]^{\alpha}.[\eta(c_i)]^{\beta}}{\sum_{c_j \in N(s)}[\tau_j]^{\alpha}.[\eta(c_j)]^{\beta}} \qquad \forall c_i \in N(s) \tag{3}$$

Where τ_i is pheromone deposition at i^{th} node,
η is optional weighing function,
c_j is each feasible solution,
α and β are positive parameters.

Pheromone updation which includes pheromone evaporation and deposition is governed equation 4.

$$\tau_i \leftarrow (1-\rho).\tau_i + \rho.\sum_{\left\{s \leftarrow \frac{S_{upd}}{c_i E_s}\right\}} w_s \,.\, F(s) \tag{4}$$

For $i=1 \dots\dots n$
Where S_{upd} is set of solution used for pheromone updation,
W_s is weight of solution s,
ρ is evaporation constant,
$F: S \to R^{+}$ is called quality function such that $f(s) < f(s') \Rightarrow F(s) \geq F(s')$.

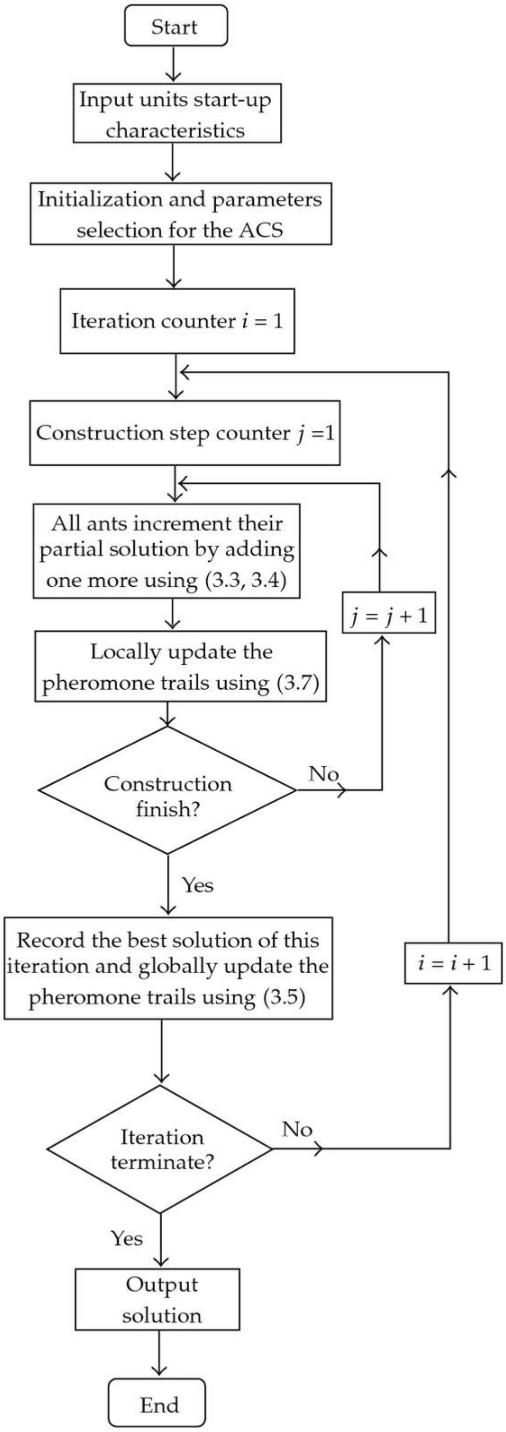

Fig. 1. Flow Chart for ACO to find shortest path.

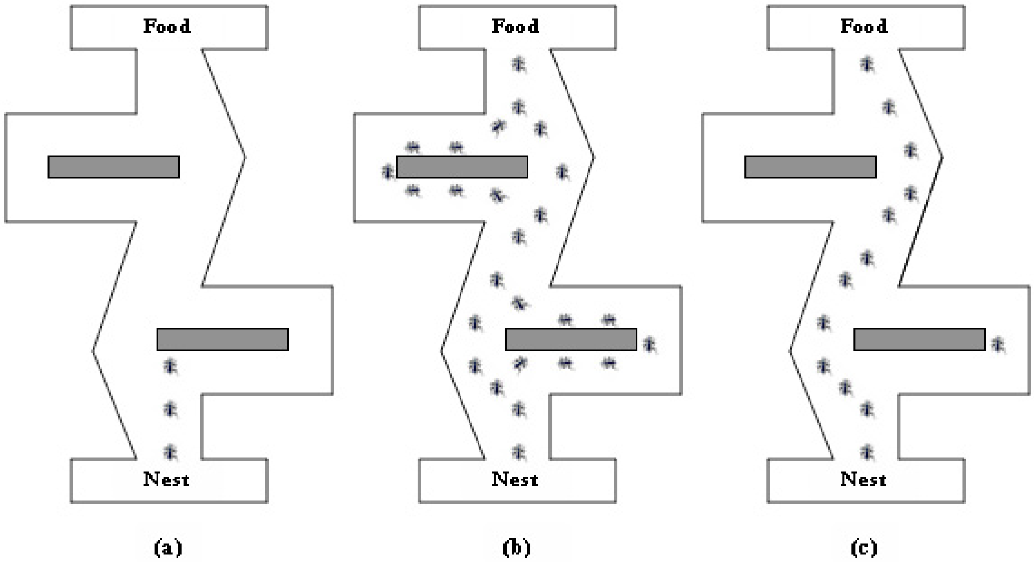

Fig. 2. The general behaviour followed by a swarm during distance optimization.

4. Problem Formulation and Simulated Results

The simulated results for both the PSO and the ACO algorithms are demonstrated in this section. Figure 3 documents the coordinate axis and demonstrates the progress of the ants/swarm as they move from the origin point of (3,3) to the final destination (5,5). The green dots signify the ants /swarm particles and the red dots / red path signifies the actual path taken at the end of all iterations.

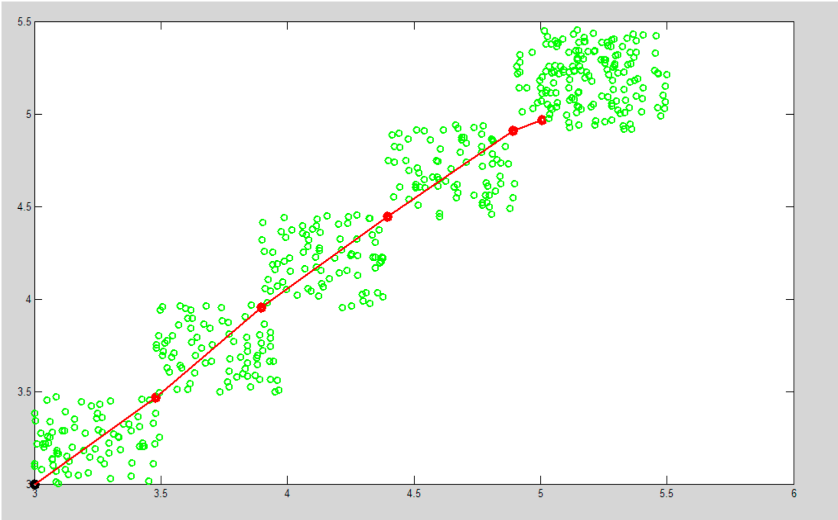

Fig. 3. Plot for coordinate axis showing the distance travelled.

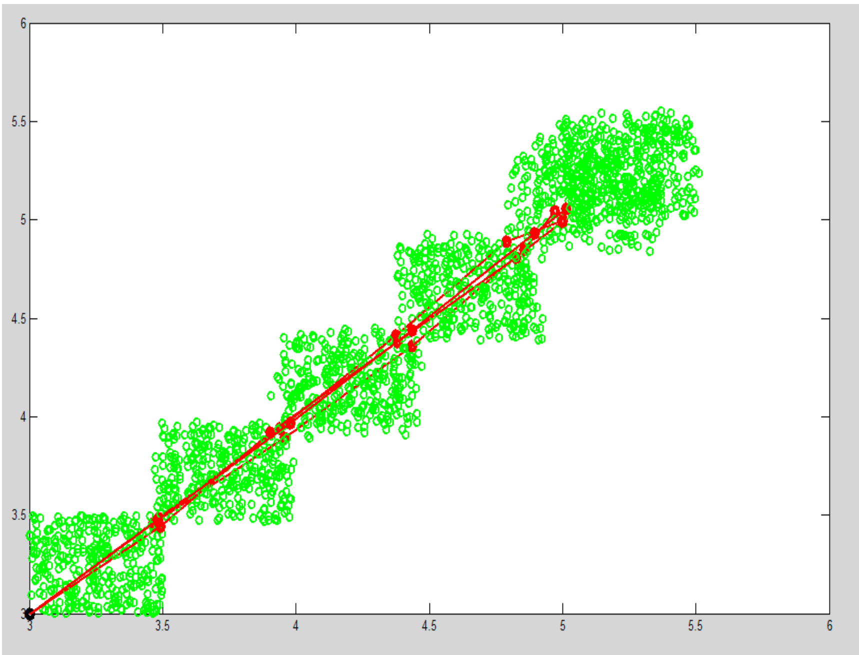

Fig. 4. Plot for the distance travelled in multiple iterations.

Figure 4 shows the multiple iterations taking place on the coordinate axis. To check that global minima was achieved simulations were run for very large number of iterations (10,000). Similarly, the swarm size has also to maintained accordingly because both PSO and ACO are after all a swarm based algorithms and only work in an optimum way if there is an appropriately large enough swarm size.

Figure 5 and 6 are plots for square of error for PSO and ACO. In figures 5 and 6 initially the square of error has higher values which decrease as number of iterations increase. This because initially parameters of the algorithms are randomly initialized and as iterations proceed these parameters get trained. From figure 5 and 6 it is seen that square of error plot attains global minima in 400 for PSO and in 80 iterations for ACO. So we can say that ACO is an overall more robust and more accurate algorithm for this particular problem.

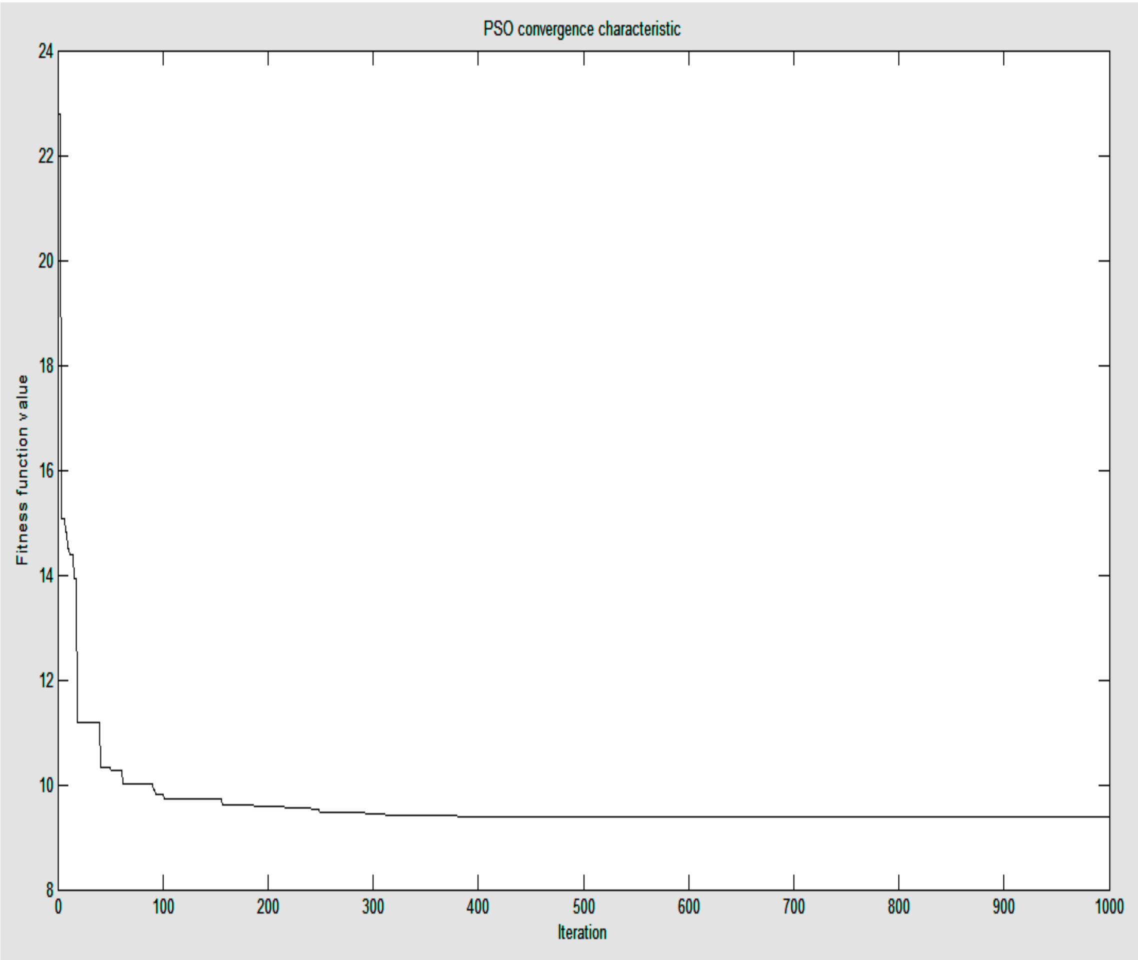

Fig. 5. Graph for square of error Vs iterations with PSO.

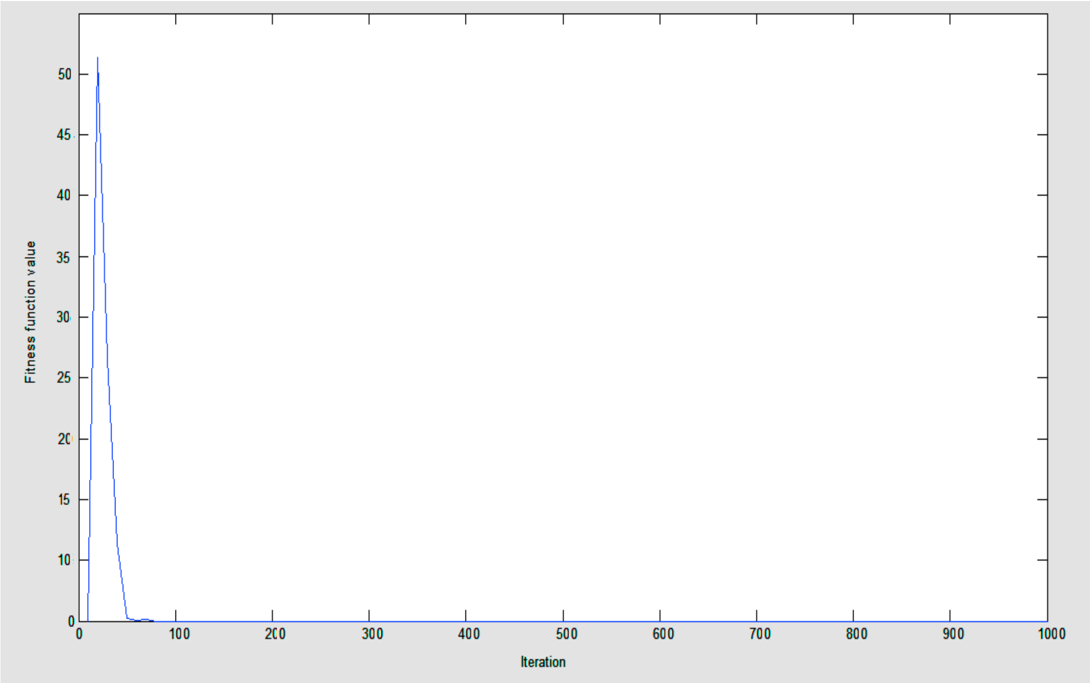

Fig. 6. Graph for square of error Vs iterations with ACO.

From table 1 it can be observed that the ACO algorithm has a lower square of error (0.2 at 80 iterations) and an overall better performance.

Table 1. Comparative Analysis of ACO Vs PSO

Algorithm Used	Minimum value of Square of error
PSO	6.8 in 400 iterations
ACO	0.2 in 80 iterations

5. Conclusion

This paper compares ACO and PSO algorithms for finding shortest distance between two vertices. Both the algorithms were tested on a model requirement where the blind ants /swarm particles had to go in the most effective manner and taking the shortest route from the initial starting point of (3,3) to the final destination of (5,5) on the coordinate axis. Simulation results show that ACO attains global minima in lesser number of iterations and to a lower value when compared with PSO. Also there is lesser number of transients in the graph for square of error for ACO. Thus it can be concluded that ACO gives better performance than PSO for distance optimization problems.

References

[1] Eberhart, R., and Shi, Y., ‘Comparing inertia weights and constriction factors in particle swarm optimization,’ Proceedings of the 2000 IEEE congress on evolutionary computation, pp. 84–88, New York: IEEE Press, 2000.

[2] Van den Bergh, F., Analysis of particle swarm optimization. Ph.D. thesis, University of Pretoria, South Africa, 2002.

[3] Van den Bergh, F., and Engelbrecht, A. P., ‘A study of particle swarm optimization particle trajectories,’ Information Sciences, 176, pp.937–971, 2006.

[4] Mendes, R., Kennedy, J., and Neves, J.,’The fully informed particle swarm: simpler, maybe better,’ IEEE Transactions on Evolutionary Computation, 8, pp.204–210.

[5] M. Geethanjali, , S. Mary Raja Slochanal, and R. Bhavani; ‘PSO trained ANN-based differential protection scheme for power transformers,’ Neurocomputing Volume 71, Issues 4–6, pp. 904–918, January 2008.

[6] Li, Y., Tong, Y., Bai, B., and Zhang, Y., ‘An improved particle swarm optimization for SVM training,’ In Proceedings of the third international conference on natural computation, pp. 611–615, Los Alamitos: IEEE Computer Society, 2007.

[7] Li, X., and Engelbrecht, A. P., ‘Particle swarm optimization—an introduction and its recent developments,’ In Proceedings of genetic and evolutionary computation conference, pp. 3391–3414, New York: ACM, 2007.

[8] Samanta, B., Al-Balushi, K. R., and Al-Araimi, S. A., ‘Artificial neural networks and genetic algorithm for bearing fault detection,’ Journal of Soft Computing, 10, pp.264–271, 2006.

[9] Yisu, J., J. Knowles, L. Hongmei, L. Yizeng and D.B. Kell, ‘The Landscape Adaptive Particle Swarm Optimizer,’ Applied Soft Computing, 8, pp.295- 304, June 2008.

[10] D. Erdogmus, and J. C. Principe, ‘Comparison of Entropy and Mean Square Error Criteria in Adaptive System Training Using Higher Order Statistics,’, Proc. ICA, 2000.

[11] Kennedy, J., and R.C. Eberhart, ‘Particle Swarm Optimization’ Proceeding of The IEEE International Conference On Neural Networks, Perth, Australia, Vol. 4, pp. 1942- 1948, 1998

[12] Roberta De Santis, Roberto Montanari. Giuseppe Vignali, ‘An adapted ant colony optimization algorithm for the minimization of the travel distance of pickers in manual warehouses’, European Journal of Operational Research, Vol. 267, Issue 1, 16 May 2018, pp 120-137.

[13] B.Booba. T.V.Gopal, ‘Comparison of Ant Colony Optimization & Particle Swarm Optimization in Grid Environment’ International Journal of Advanced Research in Computer Science and Applications Vol.1, Issue 5, November 2013.