Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 6.017

IJCSMC, Vol. 7, Issue. 12, December 2018, pg.313 – 327

Comparative Analysis of Trustworthiness in Swarm Intelligence Techniques using Mobile Adhoc Network Environment

Mogaji Stephen Alaba[1], Alese Boniface Kayode[2], Thompson Aderonke F.[3], Ayeni, O Abiodun[4]

[1] *School of Computing, Federal University of Technology, Akure, Nigeria*

[2] *School of Computing, Federal University of Technology, Akure, Nigeria*

[3] *School of Computing, Federal University of Technology, Akure, Nigeria*

[4] *School of Computing, Federal University of Technology, Akure, Nigeria*

[1] alabadepoola@yahoo.com, [2] bkalese@futa.edu.ng, [3] tonyfav24@gmail.com, [4] oaayeni@futa.edu.ng

Abstract: Swarm Intelligence is a technique that deals with collective behaviour of decentralized and self-organized systems. For algorithm with complex problems, Swarm Intelligence is one of the successful paradigms.

In this paper, an elaborate comparative study on trustworthiness is evaluated between two techniques of swarm intelligence and its hybrid: Ant Colony Optimization, Particle Swarm Optimization and hybridized Particle Swarm Optimization using a trust model in a simulated Mobile Adhoc Network environment.

The effect of evaluating trustworthiness and discovering misbehaving nodes prior to interactions, as well as their influence on the network performance would be investigated. The results of investigating the trustworthiness evaluation and the network performance even in the presence of malicious nodes showing improvement of a hybridized Swarm Intelligence technique over other existing convectional swarm intelligence systems are shown.

Keywords - Swarm Intelligence, Trustworthiness, Ant Colony Optimization, Particle Swarm Optimization, Hybridization, Pheromone, Simulation, Mobile Adhoc Network

I. INTRODUCTION

Optimization is ubiquitous and spontaneous process that forms an integral part of our day-to-day life. In the most basic sense, it can be defined as an art of selecting the best alternative among a given set of options. Swarm Intelligence, an artificial intelligence discipline has become an interesting and exciting development in computer industry. Inspired by the collective behaviours of social insects and animal societies, swarm intelligent techniques are used to solve complex real-world optimization problems.

Colonies mechanism of social insects have is fascinating and it remains unknown for a long time.

Complex tasks can be achieved by cooperation. Each cluster is similar among themselves and dissimilar to objects of other groups. Homogeneity, locality, collision avoidance, velocity matching and flock centering

are some of the main properties of collective behaviour. It has a decentralized way of working. That is, the flock moves without any leader. The movement of each bird is influenced by the nearest block mates.
For flock organization, vision is an important sense. Collision should be avoided with the nearest flock mates and also speed should match with them. They should stay close to each other without collision. There is limited communication and has no explicit model of the environment. They have perception of environment (that is sensing) and has the ability to react to environment changes.
Swarm intelligence techniques are robust and relatively simple. This paper analyzes the two most successful methods of optimization techniques inspired by Swarm Intelligence (SI): Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO). A Comparative analysis of effective node evaluation in the presence of misbehaving nodes for Ant (ACO) Based model standard PSO model, and Hybridized PSO model are presented.

II. PARTICLE SWARM OPTIMISATION (PSO)

Particle swarm optimization (PSO) is inspired by the social behavior among individuals, a stochastic search technique considered as one of the modern heuristic algorithms for optimization, introduced by Kennedy and Eberhart [19], [20], & [21] . It is based on the social behaviour metaphor of bird flocking and it is a population-based optimization technique.

According to Hazem and Janice [17]. The emergence of flocking and schooling in groups of interacting agents (such as birds, fish, penguins, etc.) have long intrigued a wide range of scientists from diverse disciplines including animal behaviour, physics, social psychology, social science, and computer science for many decades. Bird flocking can be defined as the social collective motion behaviour of a large number of interacting birds with a common group objective. The local interactions among birds (particles) usually emerge the shared motion direction of the swarm. Such interactions are based on the "nearest neighbour principle" where birds follow certain flocking rules to adjust their motion (i.e., position and velocity) based only on their nearest neighbours, without any central coordination The pioneering work of Reynolds [31] proposed three simple flocking rules to implement a simulated flocking behaviour of birds:

(i) flock centering (flock members attempt to stay close to nearby flockmates by flying in a direction that keeps them closer to the centroid of the nearby flockmates),

(ii) Collision avoidance (flock members avoid collisions with nearby flockmates based on their relative position), and

(iii) Velocity matching (flock members attempt to match velocity with nearby flockmates).

In PSO, candidate solutions of a population, called particles, coexist and evolve simultaneously based on knowledge sharing with neighbouring particles. While flying through the problem search space, each particle generates a solution using directed velocity vector. Each particle modifies its velocity to find a better solution (position) by applying its own flying experience (that is, memory having best position found in the earlier flights) and experience of neighbouring particles that is, best found solution of the population. Finally, all particles fly towards the best.

The standard PSO model consists of a swarm of particles, moving interactively through the feasible problem space to find new solutions. Each particle has a position represented by a position vector where ***n*** is the index of the particle and a velocity represented by a velocity vector. Each particle remembers its own best position so far in the vector, ***pbest*** and the best position vector among the swarm, ***gbest.***

The search for the optimal position (solution) advances as the particles' velocities and positions are updated. A particle's velocity and position are updated as follows:

$$v_{n+1} = wv_n + c_1 r_1 (p_{best} - x_n) + c_2 r_2 (g_{best} - x_n) \quad (1)$$

$$x_{n+1} = x_n + x_{n+1} \quad (2)$$

Where

- v_{n+1} = Velocity of the particle at n+1th iteration
- w = Particle inertia weight
- v_n = Velocity of particle at nth iteration
- c_1 = acceleration factor related to g_{best}, the cognitive scaling parameter
- c_2 = acceleration factor related to l_{best}, the social scaling parameter
- r_1 = random number between 0 and 1
- r_2 = random number between 0 and 1
- g_{best} = global best position on the swarm
- p_{best} = personal best position of the particle

The position of each particle in the swarm is affected both by the most optimist position during its movement (individual experience) and the position of the most optimist particle in its surrounding (near experience). Each solution vector can be confined to a vector range to control excessive roaming of particles outside the search space.

The particle weight inertial is reduced dynamically to decrease the search area in gradual fashion, using the equation below:

$$\mathrm{w} = (\mathrm{w}_{\max} - \mathrm{w}_{\min}) \times \frac{(\mathrm{t}_{\max} - \mathrm{t})}{\mathrm{t}_{\max}} + \mathrm{w}_{\min} \tag{3}$$

- $\mathrm{w}_{\max}$ = Maximum particle weight inertia
- $\mathrm{w}_{\min}$ = Minimum particle weight inertia
- $\mathrm{t}_{\max}$ = Given maximum number of iterations

Particle flies toward a new position using equation (1) and (2). All particles of the swarm find their new positions and apply these new positions to update their individual best position and global best position of the swarm. This process is repeated until maximum number of iteration count $\mathrm{t}_{\max}$ is reached.

III. ANT COLONY OPTIMISATION (ACO)

In the 1990's, Ant Colony Optimization was introduced as a novel nature inspired method for the solution of hard optimization problems [10].

Ants, like many other social insects, communicate with each other using volatile chemical substances known as pheromones, whose direction and intensity can be perceived with their long, mobile antennae. The term "pheromone" was first introduced by Karlson and Lüscher [19], based on the Greek word pherein (means to transport) and hormone (means to stimulate). There are different types of pheromones used by social insects. One example of pheromone types is alarm pheromone that crushed ants produce as an alert to nearby ants to fight or escape dangerous predators and to protect their colony [25].

ACO, being a multiagent has an approach that simulates the foraging behavior of ants for solving difficult combinatorial optimization problems, such as, the traveling salesman problem and the quadratic assignment problem. Ants, being social insects have a behavior is directed more toward the survival of the colony as a whole than that of a single individual of the colony. An important and interesting behavior of an ant colony is its indirect co-operative foraging process. While walking from food sources to the nest and vice versa, ants deposit a substance, called pheromone on the ground and form a pheromone trail. Ants smell pheromone, when choosing their way; they tend to choose, with high probability, paths marked by strong pheromone concentrations (shorter paths). Also, other ants can use pheromone to find the locations of food sources found by their nest mates. In fact, ACO simulates the optimization of ant foraging behavior. In Recent past, there are few adaptations of ACO for solution of continuous optimization problems.

For an ant, the probability of moving from position *i* to position *j* depends upon two factors:-

(1) The attractiveness of the edge: It is the prior desire of the move and is calculated by some heuristic. Normally, it is the reciprocal of the distance between *i* and *j*.

(2) Pheromone density on the edge: It is the amount of the pheromones on the edge of *i* and *j*.

An ant moves from *i* to *j* with probability as follows:-

$$P_{ij}^{k} = \begin{cases} \frac{T_{ij}^{\alpha}}{\sum_{l \in N_i^k} T_{ij}^{\alpha}}, & if\ j \in N_i^k, \\ 0 & if\ j \notin N_i^k \end{cases} \tag{4}$$

- N_i^k: Neighbour of nodes around ant k
- $\propto$ produces better results when = 1 *as opposed to 2 which mimic better results for the stochastic equation for the bridge problem*

The amount of pheromone laid down on return trip is equivalent to the quality of the solution, consequently the shorter the path, the more pheromone that is laid down.

After each ant, k, has moved to the next node, the global pheromone trail is updated by evaporating the existing pheromone:

$$T_{ij} \leftarrow (1 - \rho)T_{ij} \tag{5}$$

Where ρ is a parameter relating to the rate of evaporation (higher values of ρ leads to faster evaporation).

At end of iteration, pheromone is laid down on the nodes that ant k has visited, which is equal to one divided by the length of its trip.

$$\Delta T^{k} = \frac{1}{L^{k}} \tag{6}$$

This method reduces the number of artificial ants required to converge to best quality solution.

In this work, a simple pheromone-guided search mechanism of ant colony is implemented which acts locally to synchronize positions of the particles of PSO to quickly attain the feasible domain of objective function.

IV. Hybridization of Particle Swarm Optimisation Using Ant Colony Optimisation

Swarm intelligence meta-heuristics, namely, particle swarm optimisation and ant colony optimisation are proven to be successful approaches to solve complex optimization problems. PSO algorithm, whose concept began as a simulation of a simplified social environment, is a powerful optimization technique for solving multimodal optimization problems [31], [6] & [30]. ACO imitates foraging behaviour of real life ants, and are known to be efficient and robust for solution of combinatorial optimization problems [38], [9], [47], & [42].

The implementation of this algorithm comes in two stages. In the first stage, PSO is applied while ACO is implemented in the second stage. ACO works as a local search, wherein, ants apply pheromone-guided mechanism to update the positions found by the particles in the earlier stage, to attain rapid convergence on a feasible solution space. The implementation of ACO in the second stage of this model is based on the studies of Angeline (1998) which shows that:

i. PSO discovers reasonable quality solutions much faster than other evolutionary algorithms
ii. If the swarm is going to be in equilibrium, the evolution process will be stagnated as time goes on. Thus, PSO does not possess the ability to improve upon the quality of the solutions as the number of generations is increased.

In this model, a simple pheromone-guided mechanism of ACO is proposed to apply as local search.
The proposed ACO algorithm handles *P* ants equal to the number of particles in PSO. Each ant *i* generate a solution $\boldsymbol{z_t}$ around $\boldsymbol{g_{best}}$ the global best-found position among all particles in the swarm up to iteration count t as [39].

$$z_t = N(g_{best}, \sigma) \tag{7}$$

The components of the solution vector z_t which satisfies the Gaussian distribution with mean g_{best} and standard deviation σ is generated, where, initially at t = 1 value of σ = 1 and is updated at the end of each iteration as

$$\sigma = \sigma \times d \tag{8}$$

d is a parameter in (0.25, 0.997) and if $\sigma < \sigma_{min}$ then $\sigma = \sigma_{min}$, where, σ_{min} is a parameter in (10^{-2} , 10^{-4}). The objective function around z_t, $f(z_t)$ is the computed and replaces the current position of the particle swarm if $f(z_t) < f(x_t)$ the $x_t = z_t$

This simple pheromone-guided mechanism considers there is highest density of trails (single pheromone spot) at the global best solution ***gbest*** of the swarm at any iteration t + 1 in each stage of ACO implementation and all ants *P* search for better solutions in the neighbourhood of the global best solution. In the beginning of the search process, ants explore larger search area in the neighborhood of ***gbest*** due to the high value of standard deviation r and intensify the search around ***gbest*** as the algorithm progresses [39]. ACO pheromone mechanism helps PSO process, not only to efficiently perform global exploration for rapidly attaining the feasible solution space, but also to effectively reach optimal or near optimal solution. The pseudo-code for the Hybridized PSO model algorithmic is given in Fig. 1

BEGIN Algorithm

1. Initialise, ACO and PSO design variables, search space (P) and maximum epoch (t_{max})

2. Initialise random particles positions and velocities

 # initialise optimisation

3. For each particle in the search space

4. Calculate the fitness value (by evaluate the objective function)

5. If the fitness value (pbest) is better than the best fitness value in history

a. Set current value as the new pbest

6. End If

7. Find the particle with best (min) fitness value of all particles as the gbest

8. End For

 # perform optimisation

9. While (t <= maximum epoch)

10. Update particle velocity according to the particle velocity equation

11. Update particle position according to position update equation

12. Calculate the fitness value by evaluating the objective function

13. Update particle best (min) by comparing previous and current fitness, as pbest

14. Find the best (min) fitness value of all particles as gbest

15. Generate P solutions, z, from gbest value according to zsolution equation

16. Calculate fitness value by evaluating objective function on generated z solution

17. Update particle best position by comparing zsolution and particle objective solution

18. Find the best (min) fitness value of all particles pbest and gPbest, as gbest

19. Increment epoch count t=t+1

20. End While

21. Report the best solution gbest of the swarm

END Algorithm

Fig. 1 Hybridized PSO Model Algorithmic Pseudo-code.

V. Trust Modelling in MANETS

This subsection outlines the parameters that would be used as a basis to measure trust values in a MANET environment using the swarm intelligence techniques.

Trust is defined as the probability by which an entity is capable of performing a given action at a specific level of quality in a given timeslot and within a specified context, considering the risks and incentives involved. This definition intends to deal with trust as the composition of multiple attributes to reflect the trust features of subjectivity, uncertainty, and unpredictability.

Social and QoS properties are considered to evaluate trustworthiness of nodes. Metrics associated with the formulation of nodes trustworthiness are described below.

A. Honesty

Honesty is a social property and a friendship-based trust model metric, which is defined as the way in which nodes behave in terms of acting to favour themselves or the communities of which they are a part of [2]. Honesty is an important social trust factor in the proposed model and it refers to the degree of honesty of the evaluating node *j* about the evaluated node *i*. It is a measure of successful or failed interactions.

Negative and positive behaviours of nodes are indicators of the honesty of nodes in detecting irregular behaviour. The value of Honesty, $T_{ij}^{honesty}$ is computed by using the number of successful interactions α_{ij} between node *i* and *j* over the maximum number of successful and failed interactions$\alpha_{ij} + \beta_{ij}$.

$$T_{ij}^{honesty} = \frac{\alpha_{ij}}{\alpha_{ij}+\beta_{ij}} \qquad (9)$$

Where α_{ij} positive interaction (or feedback) is a social factor, referred to as the accumulated number of forwarding packets successfully delivered by a node in the network, This value is represented as ρ. Accumulated positive interaction is calculated as

$\alpha = \rho + 1.$

β_{ij}. is negative interaction, an important social factor described as the number of packets dropped by a node on the network. This value is represented as n. Accumulated negative interaction, β is calculated as

$\beta = n + 1.$

The initial value of $T_{ij}^{honesty}$ is 0.5 at time *t = 0*, which means that node *i* is a stranger to node *j* and no previous interaction has been observed. $T_{ij}^{honesty}$develops over time also, and its value is between 0 and 1.

B. Confidence Metric

Confidence is another friendship-based trust metric [33] and an important social property that is used to indicate how strong a tie is between two interacting nodes. It measures how frequently nodes interact with one another to evaluate relationship strength between interacting nodes. Basically, it evaluates the number of interaction between nodes. A high number of interactions can be translated into the idea that the evaluating node has a strong relationship with the evaluated node. Consequently, it improves the ability of the evaluating node to judge the trustworthiness of the node under evaluation.

Confidence $T_{ij}^{confidence}$ is expressed as the variance value of all past experiences between two interacting nodes. Its value is measured by using the beta standard deviation σ

$$T_{ij}^{confidence} = 1 - \sqrt{12\sigma_{ij}} \qquad (10)$$

$$\sigma_{ij} = \frac{\alpha_{ij} \times \beta_{ij}}{(\alpha_{ij}+\beta_{ij})^2 + (\alpha_{ij}+\beta_{ij}+1)} \tag{11}$$

α_{ij} and β_{ij} represents the positive and negative interaction observed by node *i* and *j*.

C. Energy Level

Energy is a critical Quality of Service (QoS) factor in trust systems. All nodes are energy-constrained and the lifetime of each node depends on its energy consumption. In the proposed model, the Energy Level, EV_{ij} factor indicates the remaining energy level of the node after each trust update interval *t* performed by the evaluating node *i* about the evaluated node *j*. The energy factor is calculated as

$$EV_{ij} = \frac{EV_{ij}^{Current} - EV_{ij}^{Consumed}}{EV_{ij}^{Initial}} \tag{12}$$

Where

- $EV_{ij}^{Consumed}$ is the level of energy consumed by node *j* in performing interactions
- $EV_{ij}^{Current}$ is the previous current energy of node *j*
- $EV_{ij}^{Initial}$ is the initial level of energy of node *j* to start with.

Energy is initially at the same level for all nodes in the network. Receiving and transmitting packets are the only types of communications, which are considered for energy consumption. Over time, the level of energy is adjusted based on each node's interactions. The value of the energy factor is defined in the interval (0, 1). It starts at 1, which refers to a situation where nodes have a full battery, and gradually decreases over time as nodes involve themselves in more communications. Nodes continue to be effective in performing interactions so long as the energy factor is not reduced below the threshold.

D. Trust Model Objective Function

The objective function, also referred to as the fitness function or cost function, is the performance index of particles in the population. In this work, the objective function represents the evaluated trustworthiness in generating a trusted routing path.

To generate a routing path between a source node and a destination node, trustworthiness of intermediate nodes are evaluated to ascertain that constraints are met, social trust and quality of service requirements are fulfilled. When the trust value or cost value is bigger, the performance is better.

The cost function is evaluated using multidimensional social trust and QoS metrics describes above. The social trust metrics are positive interaction, negative interaction, honesty and confidence level, while the energy factor represents an important QoS metric. The trustworthiness is therefore evaluated by combining all metrics. The confidence factor,$T_{ij}^{confidence}$, measures the level of experience between interacting nodes. The honesty factor, $T_{ij}^{honesty}$, measures selfishness or maliciousness of nodes. The energy factor, EV_{ij}, measure if a node is capable of performing an intended task or not. The threshold for all multidimensional metrics introduced in this work is 0.5 [27].

$$T_{ij} = \frac{T_{ij}^{honesty} + T_{ij}^{confidence} + EV_{ij}}{3} \tag{13}$$

E. Evaluating Trust in MANETS using the Model

In order to generate a routing path between a source and the destination node, trustworthiness of intermediate nodes are evaluated.

The trustworthiness T_{ij} is therefore evaluated by combining all metrics setting the threshold for all multidimensional metrics to 0.5, any node whose Trust value or cost value is evaluated to be less than the set threshold which is 0.5 is considered to be malicious or misbehaving node.

VI. The Network Simulator

There are many tools available for simulations of network topologies. But for this work all simulations are carried out using the Network Simulator 2 (NS-2) tool.

A. The Experimental Setup

A network with 50 randomly placed nodes is simulated. Several nodes were randomly selected to be misbehaving by dropping packets by different rates. Table1 shows the parameters used in configuring the network for the experiment. Badly behaving nodes (selfish nodes) amounting to up to 50% were simulated in the network and were responsible for dropping packets. Results from the experiment used to evaluate the proposed model are based on summarized multiple runs, and negligible variation is noticed.

TABLE 1
SIMULATED NETWORK CONFIGURATIONS

Parameter	**Value**
Number of Nodes	50
Speed	10 m/s
Routing Protocol	AODV
MAC	802.11
Source-destination Pairs	15
Transmitting Capacity	2Kbps
Packet Size	512B
Simulation	500s
Trust Threshold	0.5
Number of Particles	50
Inertial Weight (w)	1.0
Cognitive Parameter (c1)	2.0
Social Parameter (c2)	2.0
Iteration Count	100

VII. THE EXPERIMENTAL RESULTS ANALYSIS

In this section, results and analysis on the computation of the components of the trust model, namely the social Trust metrics and QoS trust metric values, are presented. The values and their relationship to the overall trustworthiness value are also provided. The simulation results over a number of iteration are shown in subsequent tables and figure.

TABLE 2
RELATIONSHIP OF ALL TRUST METRICS TO THE OVERALL TRUSTWORTHINESS OF THE ANT BASED (ACO) TECHNIQUE WITH A NUMBER OF ITERATION

Number of Iteration	Social Trust Metrics		QoS Trust Metric	Overall Trustworthiness
	Honesty	Confidence	Energy Level	
10	0.51051	0.5987	0.9657	0.6916
20	0.6425	0.6854	0.9795	0.7691
30	0.8461	0.8141	0.9681	0.8761
40	0.8621	0.8314	0.9611	0.8849
50	0.87528	0.9017	0.9505	0.9092
60	0.8987	0.9097	0.9378	0.9154
70	0.9021	0.9173	0.9243	0.9146
80	0.9354	0.9357	0.9189	0.9300
90	0.9482	0.9669	0.9025	0.9392
100	0.9615	0.9698	0.8816	0.9376

TABLE 3
RELATIONSHIP OF ALL TRUST METRICS TO THE OVERALL TRUSTWORTHINESS OF THE PSO TECHNIQUE WITH A NUMBER OF ITERATION

Number of Iteration	Social Trust Metrics		QoS Trust Metric	Overall Trustworthiness
	Honesty	Confidence	Energy Level	
10	0.5121	0.6132	0.9724	0.6992
20	0.6515	0.7054	0.9845	0.7805
30	0.8611	0.8481	0.9751	0.8948
40	0.8771	0.8583	0.9654	0.9003
50	0.89128	0.9077	0.9545	0.9178
60	0.9049	0.9147	0.9413	0.9203
70	0.9071	0.9213	0.9316	0.9200
80	0.9494	0.9469	0.9216	0.9393
90	0.9642	0.9742	0.9069	0.9484
100	0.9705	0.9795	0.8902	0.9467

TABLE 4:
RELATIONSHIP OF ALL TRUST METRICS TO THE OVERALL TRUSTWORTHINESS OF THE HYBRIDIZED PSO TECHNIQUE WITH A NUMBER OF ITERATION

Number of Iteration	Social Trust Metrics		QoS Trust Metric	Overall Trustworthiness
	Honesty	Confidence	Energy Level	
10	0.5618	0.6447	0.9892	0.7319
20	0.6725	0.7454	0.9865	0.8015
30	0.8822	0.8521	0.9781	0.9041
40	0.887	0.8603	0.9714	0.9062
50	0.9008	0.9197	0.9655	0.9287
60	0.9119	0.9267	0.9573	0.9320
70	0.9151	0.9423	0.9446	0.9340
80	0.9693	0.9509	0.9336	0.9513
90	0.9702	0.9822	0.9219	0.9581
100	0.9785	0.9895	0.9056	0.9579

TABLE 5
SUMMARY TRUSTWORTHINESS CHART OF THE THREE SWARM INTELLIGENCE TECHNIQUES WITH A NUMBER OF ITERATION

Number of Iteration	Trustworthiness		
	Ant Based	**PSO**	**Hybridized PSO**
10	0.6916	0.6992	0.7319
20	0.7691	0.7805	0.8015
30	0.8761	0.8948	0.9041
40	0.8849	0.9003	0.9062
50	0.9092	0.9178	0.9287
60	0.9154	0.9203	0.9320
70	0.9146	0.9200	0.9340
80	0.9300	0.9393	0.9513
90	0.9392	0.9484	0.9581
100	0.9376	0.9467	0.9579

VIII. Comparative analysis of effective node evaluation in the presence of misbehaving nodes for Ant (ACO) based, standard PSO, and Hybridized PSO techniques

This section shows performance of system in choosing trusted nodes while sending packets from the source to the destination in the presence of misbehaving nodes. It also shows performance comparison between the Ant (ACO) based, the standard PSO and the hybridized PSO techniques

The result of the comparative analysis is shown in Table 6 and Fig. 2

TABLE 6
COMPARATIVE ANALYSIS OF EFFECTIVE NODE EVALUATION IN THE PRESENCE OF MISBEHAVING NODES FOR ANT (ACO) BASED, STANDARD PSO, AND HYBRIDIZED PSO TECHNIQUES

Number of Malicious nodes	% choosing good nodes		
	Ant (ACO) Based	**Standard PSO**	**Hybridized PSO**
10	88.5741	92.7254	99.3110
20	86.5326	90.8521	99.1433
30	83.8746	89.2416	99.0247
40	82.0628	88.6954	98.7744
50	79.9845	88.0215	98.7513

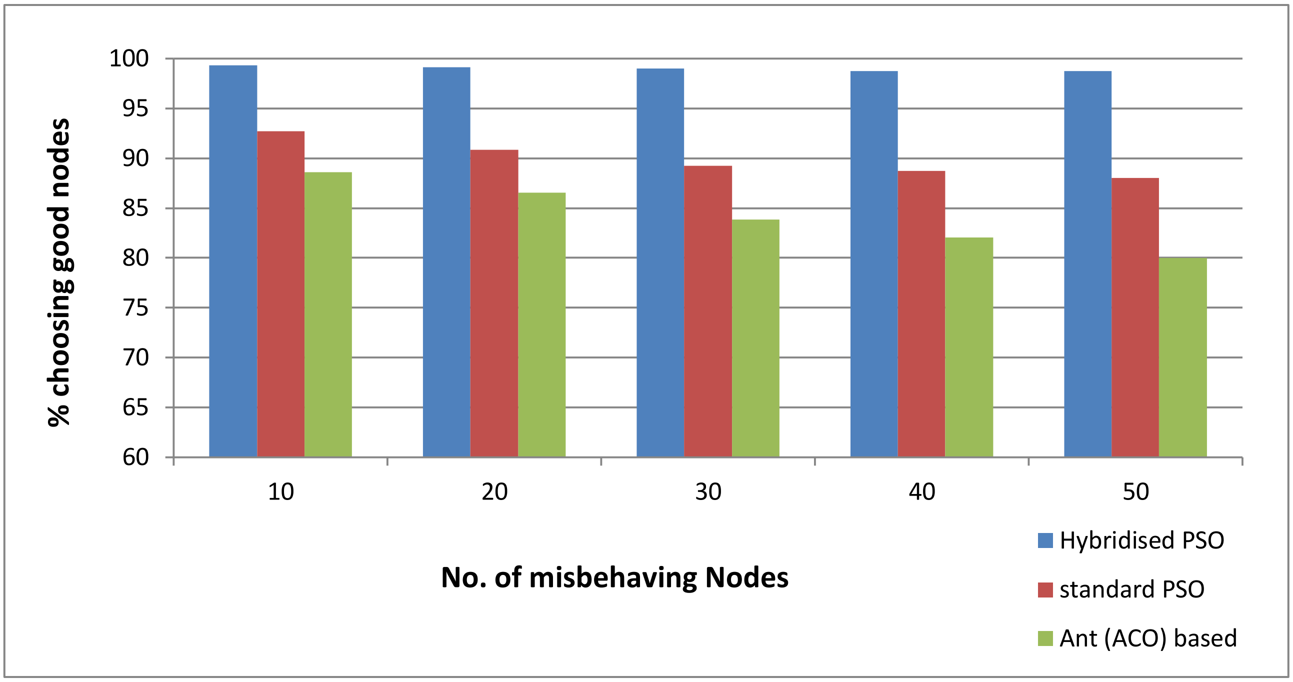

Fig. 2 Comparative analysis of effective node evaluation in the presence of misbehaving nodes for Ant (ACO) based, standard PSO, and Hybridized PSO techniques

IX. SUMMARY AND CONCLUSIONS

This Paper presented the application of some of swarm intelligence techniques to establish trust in Ad-hoc Network environment. It also hybridized Particle Swarm Optimization (PSO) with Ant Colony Optimization (ACO) techniques using pheromone-guided local search to improve the performance of Particle Swarm Optimization algorithm.

These techniques were incorporated into a simulated Mobile Ad-hoc Network (MANET) environment using specific configured parameters.

Table 6 and fig. 2 present the comparative evaluation of the swarm intelligence techniques in accurately evaluating trusted node in the network environment and choosing good nodes for communication path generation. It shows that the as the number of misbehaving nodes increase in a network environment the accuracy slightly reduces but at a very minute rate. When 5 malicious nodes existed in the network, the hybridized system model was at an accuracy of 99.31%. Increasing the misbehaving nodes to 50, which is the half of the total nodes in the network, the accuracy level dropped slightly to 98.75%, representing a mere 0.56% difference.

It show that ACO helps PSO process not only to efficiently perform global exploration for rapidly attaining the feasible solution space but also to effectively reach optimal or near optimal solution.

Consequently implies that trustworthiness of the hybridized PSO system is an improvement over some of the existing convectional Swarm Intelligent techniques.

These comparisons of numerical results show that there is a opportunity of research in hybridizing swarm intelligence methods to solve difficult continuous optimization problems.

References

[1] Angeline, P.J. (1998). Evolutionary optimization versus particle swarm optimization: philosophy and performance difference, in: V.W. Porto et al. (Eds.), Proceedings of 7th Annual Conference on Evolutionary Programming, Lecture Notes in Computer Science, vol. 1447, Springer, Berlin, 1998, pp. 601–610.

[2] Bao, F., Chen I.R., Chang M., and Cho, J.H. (2011). Hierarchical trust management for wireless sensor networks and its application to trust-based routing. Proceedings of the 2011 ACM Symposium on Applied Computing, pp. 1732- 1738.

[3] Bao, F., Chen, R., Chang, M., and Cho, J. H. (2012). Hierarchical trust management for wireless sensor networks and its applications to trust-based routing and intrusion detection, Network and service Management, IEEE Transactions on, vol. 9, pp. 169-183.

[4] Beni, G., W.J. (1989). Swarm intelligence in cellular robotics systems: NATO Advanced Workshop on Robots and Biological System.

[5] Bin, Y., Zhong-Zhen, Y., and Baozhen, Y. (2009). An improved ant colony optimization for vehicle routing sensor networks, IEEE Communications Surveys & Tutorials, pp: 2-28.

[6] Dong, Y., Tang, J., Xu, B., and Wang D., (2005). An application of swarm optimization to nonlinear programming, Computers & Mathematics with Applications 49 (11–12) pp1655–1668.

[7] Dong, P., Wang H., and Zhang H. (2009). Probability-based trust management model for distributed e-commerce, Network Infrastructure and Digital Content. IC-NIDC. IEEE International Conference, pp. 419-423.

[8] Dorigo, M. (1992). Optimization, Learning and Natural Algorithms (in Italian). Ph.D. thesis, Dipartimento di Elettronica, Politecnico di Milano, Italy.

[9] Dorigo, M., Blum C. (2005). Ant colony optimization theory: a survey, Theoretical Computer Science 344 (2–3) (2005) 243–278

[10] Dorigo, M., Maniezzo, V., and Colorni, A. (1996). The ant system: Optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics - Part B 26(1):29–41

[11] Dorigo, M., Gambardella, L.M. (1997). Ant colony system: A cooperative learning approach to the travelling salesman problem. IEEE Transactions on Evolutionary Computation

[12] Dorigo, M., Di Caro, G. (1999). The ant colony optimization meta-heuristic, McGraw-Hill Ltd.,UK, Maidenhead, UK, England, pp 11–32 1:53–66

[13] Dorigo, M., Caro, G.D., Gambardella, L.M. (1999). Ant algorithms for discrete optimization. Artificial Life 5(2):137–172

[14] Dorigo, M,. and Stützle, T. (2004). Ant Colony Optimization. MIT Press, Cambridge, ISBN: 978-0-262-04219-2.

[15] Goss, S., Aron, S., Deneubourg, J., and Pasteels, J. (1989). Self-Organized Shortcuts in the Argentine Ant, Naturwissenschaften, Vol. 76, pp. 579-581,

[16] Hao, Y., et al. (2004). Security in mobile ad hoc networks: challenges and solutions. Wireless Communications, IEEE, 11(1): p. 38-47.

[17] Hazem, A., and Janice, G. (2012). Swarm Intelligence: Concepts, Models and Applications. Technical Report School of Computing Queen's University Ontario Canada.

[18] He, Q., Wu, D., and Khosla, (2004) P. "SORI: A Secure and Objective Reputation-based Incentive Scheme for Ad-Hoc Networks," Proc. IEEE Wireless Communications and Networking Conf., vol. 2, pp. 825- 830.

[19] Karlson, P., and Lüscher, M. (1995). Pheromones: a new term for a class of biologically active substances. Nature, Vol. 183, pp. 55–56.

[20] Kennedy, J., and Eberhart, R.C. (1995). Particle swarm optimization. In Neural Networks, 1995. Proceedings. IEEE International Conference on. IEEE.

[21] Kennedy, J., and Eberhart, R.C (1995). Particle Swarm Optimization. In Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, pp. 1942–1948.

[22] Kennedy, J., and Eberhart, R.C (1995). A new optimizer using particle swarm theory. In Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, pp. 39–43.

[23] Li, W., Parker, J., and Joshi A. (2012). Security through collaboration and trust in MANETs, Mobile Networks and Applications, vol. 17, pp. 342-352.

[24] Li, R., and Li, J. (2013). Requirements and design for neutral trust management framework in unstructured

[25] Lloyd, C. (2003). The alarm pheromones of social insects: A review, Technical report, Colorado State University.

[26] Luo, J., Liu, X., and Fan, M. (2009). A trust model based on fuzzy recommendation for mobile ad-hoc networks. Computer Networks, vol. 53, pp. 2396-2407.

[27] Mogaji, S.A. Alese, B.K and Adetunmbi A.O. (2018) "Validation of Hybridized Particle Swarm Optimization (PSO) algorithm with the pheromone mechanism of Ant Colony Optimization (ACO) using Standard Benchmark Function" Communications on Applied Electronics (CAE) – ISSN : 2394-4714 Foundation of Computer Science FCS, New York, USA Volume 6 – No.1, www.caeaccess.org

[28] Nagalakshmi, S., and Rakesh, P. (2016). Performance Comparison and Evaluation of Efficient Routing Protocols for MANETs: Ant Inspired Adaptive Routing. *International Journal of Advanced Research in Computer and Communication Engineering* ISO 3297:2007 Certified Vol. 5, Issue 9.

[29] Nekkantim R.K., and Lee, C. (2004) "Trust-based Adaptive On Demand Ad Hoc Routing Protocol," Proc. 42th Annual ACM Southeast Regional Conf., Huntsville, Alabama, 2004, pp. 88-93.

[30] Ourique, C.O., Biscaia, E.C., and Pinto, J.C.(2002). The use of particle swarm optimization for dynamical analysis in chemical processes, Computers & Chemical Engineering 26 (12) 1783–1793.

[31] Paterlini, S., and Krink, T. (2006). Differential evolution and particle swarm optimisation in partitional clustering, Computational Statistics & Data Analysis" 50 (5) 1220–1247.

[32] Reynolds, C.W. (1987). Flocks, herds, and schools: A distributed behavioural model, Computer Graphics (ACM SIGGRAPH '87 Conference Proceedings), Vol. 21, No. 4, pp. 25–34.

[33] Shabut, A.R.M. (2015). Trust Computational Models for Mobile Ad Hoc Networks, a Ph.D. thesis submitted to School of Computing Faculty of Engineering and Informatics University of Bradford

[34] Shabut, A., Dahal, K.P., and Awan, I. (2013). A Recommendation-Based Trust Model for MANETs to Enhance Dynamic Recommender Selection Using Multiple Rules, Seventh International Open Conference HET-NETs 2013, UK, Ilkely.

[35] Shabut, A., Dahal, K.P., Awan, I. (2013). A Trust-Based Monitoring Model to Secure Routing Protocol in MANETs Using Enhanced Trust Metric", Seventh International Open Conference HET-NETs 2013, UK, Ilkely, 2013.

[36] Shabut, A., and Dahal, K.P. (2011). Trust and Security Management in Distributed Systems, University of Bradford school of computing, School Research seminars.

[37] Sharvani G.S. (2012) "Development of Swarm Intelligent Systems for MANET" A Ph.D Thesis submitted to the Department of Computer Science and Engineering, Faculty of Engineering, Avinashilingam University for Women, Coimbatore

[38] Shelokar, P.S., Jayaraman, V.K., and Kulkarni, B.D. (2004). An ant colony classifier system: application to some process engineering problems. Computers & Chemical Engineering 28 (9) (2004) 1577–1584

[39] Shelokar, P., Siarry, P., Jayaraman, V. K., & Kulkarni, B. D. (2007). Particle swarm and ant colony algorithms hybridized for improved continuous optimization. Applied Mathematics and Computation, 188(1), 129–142.

[40] Shi, Y (2004) "Feature article on particle swarm optimization", IEEE Neural Network Society, Feature Article, pp. 8–13.

[41] Shyu, S.J., Yin, P.Y., Lin, B.M.T. (2004). An ant colony optimization algorithm for the minimum weight vertex cover problem. Annals of Operations Research 131:283–304

[42] Shyu, S.J., Lin, B.M.T., and Hsiao, T.S. (2006). Ant colony optimization for the cell assignment problem in PCS networks, Computers & Operations Research 33 (6) (2006) 1713–1740.

[43] Theodorakopoulos and J.S. Baras, "On Trust Models and Trust Evaluation Metrics for Ad-hoc Networks," IEEE Journal on selected Areas in Communications, Vol24, No. 2, Feb 2006, pp. 318-328

[44] Wang Y.D., and Emurian, H .H. (2005). An overview of online trust: Concepts, elements, and implications, Computers in human behavior, vol. 21, pp. 105-125, 2005.

[45] Wang, Y.F., Hori, Y., and Sakurai, K. (2008). Characterizing economic and social properties of trust and reputation systems in P2P environment, *Journal of Computer Science and Technology*, vol. 23, pp. 129-140, 2008.
[46] Wang, Y. and Vassileva, J. (2003). Trust and reputation model in peer-to-peer networks. Peer-to-Peer Computing, 2003. (P2P 2003). Proceedings. Third International Conference on, 2003, pp. 150-157
[47] Yin, P.Y., and Wang, J.Y., (2006). Ant colony optimization for the nonlinear resource allocation problem. Applied Mathematics & Computation 174 (2)
[48] Yunfang, F (2007) “Adaptive Trust Management in MANETs,” Proc. 2007 Int’l Conf. on Computational Intelligence and Security, Harbin, China, pp 804-808.

