Journal of Theoretical and Applied Information Technology
31st July 2014. Vol. 65 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A MULTI-OBJECTIVE APPROACH FOR ENERGY EFFICIENT CLUSTERING USING COMPREHENSIVE LEARNING PARTICLE SWARM OPTIMIZATION IN MOBILE AD-HOC NETWORK

[1]A.KARTHIKEYAN, [2]SANYUKTA, [2]SHEPHALI GUPTA, [3]T.SHANKAR
[1]Assistant Professor (Senior), School of Electronics Engineering, VIT University
[2]Student, School of Electronics Engineering, VIT University
[3]Assistant Professor (S.G), School of Electronics Engineering, VIT University
E-mail: [1]akarthikeyan@vit.ac.in , [3]tshankar@vit.ac.in

ABSTRACT

A mobile ad-hoc network (MANET) faces various challenges including limited energy, limited communication bandwidth, computation constraint and cost. Therefore, clustering of sensor nodes is adopted which involves selection of cluster-heads for each cluster. This enhances system performance by enabling bandwidth reuse, better resource allocation and improved power control. The various existing clustering techniques provide a single optimized solution in a single simulation run. Therefore, a multi-objective approach is used to optimize the number of clusters and to manage the energy dissipation issues. The proposed algorithm is a multi-objective variant of Particle Swarm Optimization (PSO) called multi-objective comprehensive learning particle swarm optimization (MOCLPSO) which reduces the time-complexity and increases the speed of the algorithm. In this technique, the best position of a randomly selected particle from the population is used to update the velocity of particle in each dimension, rather than using the personal or global best positions. The parameters taken into consideration in the proposed algorithm includes degree of nodes, transmission range and battery power consumption of the nodes. This technique provides multiple trade–off solutions in a single run of the algorithm. The performance of the proposed algorithm is compared with various clustering techniques: LEACH, PSO, WCA, CLPSO and MOPSO.

Keywords: *Comprehensive Learning Particle Swarm Optimization (CLPSO), Multi-objective Particle Swarm Optimization (MOPSO), Multi-objective Comprehensive Learning Particle Swarm Optimization (MOCLPSO), Particle Swarm Optimization (PSO), Weighted Clustering Algorithm (WCA)*

1. INTRODUCTION

A mobile ad-hoc network (MANET) is a network of free or wirelessly connected nodes that operate without a fixed infrastructure. It is a self-adapting network with no centralized control. There exists issues of cost, speed, computational constraints, limited energy and limited communication bandwidth in these networks. Since the nodes are battery-powered; thus energy is a precious resource, that has to be carefully used by the nodes in order to avoid an early termination of their activity, and hence the study and implementation of energy-efficient algorithms for mobile ad-hoc network constitutes a vast area of research in the field of ad-hoc networks. The network performance is enhanced by meaningful organization of nodes in clusters. For each of these clusters, a cluster-head is elected and the technique is called clustering. Clustering enables scalability, bandwidth reuse and better resource allocation resulting in enhanced system capacity and improved power control.

Cluster-heads are responsible for managing data aggregation and topology of the network. But the frequent selection of cluster-heads adversely affects the network performance. Hence, optimal cluster-head selection procedure is adopted, which is a NP-hard problem. This clustering technique based on optimal cluster-head selection helps in achieving energy-efficiency in mobile ad-hoc network.

Optimization refers to finding alternative that best fits the situation, employs resources in a most effective and efficient manner, and yields the result corresponding to the extreme values of one or more objectives. In the real world, most of the problems have several conflicting objectives that need to be optimized simultaneously. Such problems are

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

referred to as multi-objective problems (MOPs). The traditional mathematical programming techniques used for solving MOPs generate a single solution in a single run. This is overcome by adopting evolutionary algorithm paradigm that generates a set of solutions in one run. The set of non-dominated solutions generated by MOP form a Pareto optimal front.

Particle swarm optimization (PSO) is a well-known evolutionary algorithm that has been implemented to find solutions for single as well as multi-objective problems. The algorithm uses local best position (*lbest*) and global best position (*gbest*) to update the positions and velocities of a particle. PSO based on *lbest* or *gbest* restricts the social learning aspect and may easily get trapped in the local optimum. To avoid this problem, comprehensive learning particle swarm optimization (CLPSO) technique is used, which uses *lbest* for updating the position, preventing the swarm to go into premature convergence.

In this paper, handling of multi-objective using CLPSO is presented and this technique is called multi-objective comprehensive learning particle swarm optimization (MOCLPSO). In the proposed algorithm, the factors like degree of nodes, transmission range and battery power consumption are optimized. This provides diversity of solutions, hence increasing the flexibility of choosing a solution according to the requirement. This not only reduces the time-complexity but also searches more promising areas of the search space.

The rest of the paper is divided into following sections: Section 2 describes the already existing algorithm – LEACH, PSO, WCA and CLPSO. Section 3 deals with multi-objective clustering. Section 4 discusses MOPSO. Section 5 describes our proposed algorithm. The experimentation and results are given in Section 6 and Section 7 provides conclusion.

2. RELATED WORK

2.1 LEACH

LEACH stands for Low-Energy Adaptive Clustering Hierarchy and was one of the first cluster-based hierarchical protocol introduced in [2]. It is the simplest routing protocol in wireless sensor network (WSN) whose main aim is to distribute the energy load equally among all the sensor nodes in the network and prolong network lifetime [3][4]. LEACH protocol runs with many rounds [5]. Each round begins with a cluster set-up phase. The nodes organize themselves into local clusters, with one node acting as the cluster head (CH). The decision of being a CH is made by the node *n* by selecting a random number between 0 and 1. If the number is less than threshold T(n), node becomes a cluster head for the current round.

$$T(n) = \begin{cases} \dfrac{p}{1-(p*r*\mathrm{mod}(1/p))} & ;if \quad n \in G \\ 0 & ;otherwise \end{cases} \qquad (1)$$

where, p is the desired percentage of cluster heads, r is the current round and G is the set of nodes that have not been cluster heads in last 1/p rounds. The selection of CH is followed by an advertisement message sent, using CSMA-MAC protocol, to the rest of the nodes. The received signal strength helps the nodes to decide the cluster to which it belongs. CH nodes creates a TDMA schedule for each node in the cluster to transmit data.

The cluster set-up phase is followed by a steady state phase, where transfer of data from nodes to CH and finally to base station takes place. The CHs fuse the data they receive reducing the number of packets to be transmitted. This protocol minimizes the energy consumption by turning of the non CH nodes till its turn appears.

2.2 Particle Swarm Optimization

Particle swarm optimization (PSO) is a centralized, energy aware cluster based protocol insinuated by James Kennedy and Russel Eberhart in [6]. This algorithm, motivated by the social behavior, is a population based optimization procedure. PSO uses high energy node as CH and produces clusters that are evenly distributed throughout the network. The main idea is to reduce the intra-cluster distance and energy optimization of the entire network.

A single solution of the problem is called a particle and the group of all these particles becomes a swarm. Initially the positions and velocities of the particles are generated randomly. Each particle will have a fitness value which will be evaluated by the fitness function to be optimized in each generation [7] [8]. Later the algorithm proceeds iteratively and updates all velocities and positions of the particles using personal best and global best positions. The velocity update formula is as follows:

$$v_{id}(t+1) = w.v_{id}(t) + c_1.\alpha.(p_{id} - x_{id}(t)) + c_2.\beta.(p_{gd} - x_{id}(t)) \qquad (2)$$

The position update formula is as follows:

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \qquad (3)$$

Journal of Theoretical and Applied Information Technology
31st July 2014. Vol. 65 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

where w is the inertia weight, α and β are two positive constants in the range [0,1], c_1 and c_2 are learning factors, p_{id} is particle's best position, p_{gd} is global best position. The performance of the optimal solution is directly related to the fitness function. The fitness function is specified as follows [9] :

$$f = \varepsilon.f1 + (1-\varepsilon).f2 \tag{4}$$

$$f_1(i) = E(i) / \sum_{k=1,k\neq i}^{m} E(k) \tag{5}$$

$$f_2(i) = (m-1) / \sum_{k=1,k\neq i}^{m} d(i,k) \tag{6}$$

In the above quoted equations, function $f_1(i)$ is the ratio of node i's energy to the total energy of cluster, m is the number of nodes within the cluster, $E(k)$ is the energy of node k. Function $f_2(i)$ refers to total Euclidean distance of cluster nodes to node i; $d(i, k)$ being the distance between node i & node k. ε is a user defined constant which determines the contribution of each of the functions used. The node with the maximum value of $f(i)$ is chosen as the cluster head.

2.3 Weighted Clustering Algorithm

Chatterjee et al. [10] proposed an on-demand distributed clustering algorithm called the weighted clustering algorithm (WCA). The on-demand characteristic of the algorithm reduces the information exchange by less system updates and thus reduces the computational cost. This algorithm takes into account the following factors: ideal degree, transmission power, mobility and battery power of mobile nodes[11]. Each node calculates its weight based on these parameters. The node with the highest weight is elected as the CH. The CH election procedure is not periodic and is invoked as rarely as possible.

The combined weight W_v for each node v is given by

$$W_v = w_1\Delta_v + w_2D_v + w_3M_v + w_4P_v \tag{7}$$

where w_1, w_2, w_3 and w_4 are the weighing factors such that $w_1 + w_2 + w_3 + w_4 = 1$. Δ_v is the degree difference given by

$$\Delta_v = |d_v - \delta| \tag{8}$$

Here, d_v is the neighbors of each node within the transmission range and δ is the pre-defined threshold number of nodes a CH can handle. D_v is the sum of distances of a node with all its neighbors:

$$D_v = \sum_{v'\in N(v)} \{dist(v,v')\} \tag{9}$$

M_v denotes mobility, which is the running average of the speed for every node till current time T.

$$M_v = \frac{1}{T}\sum_{t=1}^{T}\sqrt{(X_t - X_{t-1})^2 + (Y_t - Y_{t-1})^2} \tag{10}$$

where (X_t,Y_t) and (X_{t-1},Y_{t-1}) are coordinates of v at t and $(t-1)$. P_v implies how much battery power has been consumed.

2.4 Comprehensive Learning Particle Swarm Optimization

W. Shahzad et al. [12] insinuated comprehensive learning particle swarm optimization (CLPSO) based clustering to optimize the number of clusters and efficiently manage the resources of the network. CLPSO is a weighted clustering algorithm that selects CH based on the combined weight of each node. CLPSO can be distinguished from classical PSO based on the position and velocity updating procedure. In CLPSO, the *lbest* position of a particle is used for updating instead of using the *gbest* and *lbest* of the particle. The velocity update equation is given as:

$$v_{id}(t+1) = w.v_{id}(t) + c_1.rand.(lbest_{fi(d)} - x_{id}(t)) \tag{11}$$

where, $f_i = [f_i(1), f_i(2), \ldots\ldots, f_i(d)]$ describes which particles *lbest* the particle i will use and $lbest_{fi(d)}$ is the dimension of any particle's *lbest* including its own *lbest*.

A tournament selection procedure is used in this algorithm for updating the particle. First, two particles are randomly selected from the population, excluding the particle whose velocity is to be updated. The fitness value of the *lbest* of these two particles are compared and the particle with the better fitness is used as the exemplar vector for updating, given the fitness of both the particles are not equal. This algorithm provides a higher convergence speed and consistency in the results.

3. MULTI-OBJECTIVE CLUSTERING

The conventional clustering algorithms optimize only a single objective and are thus limited in their scope of application. But the real world problems have multiple conflicting objectives and highly complex search space. This is handle by multi-objective clustering which give rise to a set of compromise solutions, known as Pareto optimal. The vector corresponding to the solutions in the Pareto optimal set is called non-dominated solution.

Journal of Theoretical and Applied Information Technology
31st July 2014. Vol. 65 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A general MOP consists of *n* conflicting objectives that are functions of decision variables. These objectives can be maximized or minimized in order to fulfill the requirement. A MOP has two search spaces that includes decision variable space and objective space, unlike single objective problem that has only decision variable space. A solution s_1 is said to dominate the other solution s_2 if and only if following two conditions are true: 1) the solution s_1 is no worse than s_2 in all objectives, and 2) the solution s_1 is strictly better than s_2 in at least one objective. If any of the two condition is not satisfied, the solution s_1 does not dominate solution s_2. When two solutions are compared with respect to all objective functions and if none of them is better than the other then the two solutions are called non-dominated solutions.

4. MULTI- OBJECTIVE PARTICLE SWARM OPTIMIZATION

Evolutionary algorithm (EA) is a generic population-based metaheuristic optimization algorithm inspired by biological evolution. These algorithm have been used for solving MOP since they are used to obtain multiple solutions in a single run. Coello Coello et al. [13] [14] introduced an algorithm to extend the existing heuristic, particle swarm optimization, in order to deal with MOP. This technique is called multi-objective particle swarm optimization (MOPSO) and is a geographical-based approach to maintain diversity. It uses an external memory called repository, which stores the previously obtained non-dominated vectors that are later used by other particles to guide their own flight.

Table 1: MOPSO Algorithm

(1) Randomly initialize the positions of all the nodes.
(2) Initialize the velocity of each node
(3) Initialize all the general parameters of MOPSO
(4) For each particle X
    DO
    a) WHILE whole network is not covered
        DO
        i. Select cluster-head X_i
        ii. Find the neighbors of the cluster head
    b) Remove the cluster-head '*i*' and its neighbors for next cluster-head selection process.
    c) END WHILE
(5) END FOR
(6) Evaluate each of the fitness of each particle.
(7) Find the Pareto front with non-dominated sorting.
(8) Store the particles having non-dominated vectors in the repository.
(9) Find the personal best *lbest* and global best *gbest* vectors from the repository.
(10) WHILE maximum number of cycles are not reached
    DO
    a) FOR each particle X
        DO
        i. Update the velocity and position of each particle

$$v_{id}(t+1) = w.v_{id}(t) + c_1.\alpha.(p_{id} - x_{id}(t))$$
$$+ c_2.\beta.(p_{gd} - x_{id}(t))$$
$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1)$$

        ii. Keep the particles within the boundaries.
        iii. Evaluate the fitness of each particle
        iv. Update the *lbest* and *gbest* of each particle if required.
    b) END FOR
    c) Update the content of repository.
(11) END WHILE

MOPSO starts with random generation of population P_o consisting of particles with unique ID. Each particle in MOPSO has a characteristic of being unique and complete in its form. First, the CHs are chosen based on weights assigned to each objective and this is followed by selection of neighbors for each CH resulting in formation of clusters. The Pareto optimal solutions are generated using non-dominated sorting based on the sum of objectives and these solutions are stored in the repository [15]. The *lbest* and *gbest* are calculated for each particle which are later used for updating positions and velocities of each particle. When the current combination of particles in the solution is better than that stored in repository, the particle's position and repository content is updated.

5. PROPOSED TECHNIQUE

Although there exists many variants of PSO, premature convergence is still the main deficiency of the algorithm causing a breakdown in the global search capacity of the system. [16].The search behavior of PSO is driven by changes in the particles' local variables. In MOPSO, *lbest* and *gbest* of the particle is used for updating the position and velocity of each particle thus inhibiting the social learning behavior to only *gbest*. This results in fast convergence of the algorithm after which all the future positions of a particle are either very close to their historic best position or are of greater cost. Thus, the particles are attracted towards the *gbest* and are trapped in the local optimum. To overcome this problem a novel technique called multi-objective comprehensive

Journal of Theoretical and Applied Information Technology
31st July 2014. Vol. 65 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

learning particle swarm optimization (MOCLPSO) is proposed. In this technique, instead of using the *lbest* and *gbest* for modifying the position and velocity of a particle, *lbest* of a randomly selected particle from the population is used to modify the particle. This method not only increases the speed of the algorithm but also searches more promising areas of the search space.

In MOCLPSO, a learning probability P_c is used to determine the exemplar vector. For each particle, a random number is generated and is compared to P_c. If the random number is larger than P_c then the particle will learn from its own *lbest* else it selects an exemplar vector from the population. The procedure of selecting an exemplar vector is as follows:

1. Two particles are randomly chosen from the population. This excludes the particle under consideration for updating.
2. The fitness of *lbest* of these two particles are compared and select the winner.
3. The winner's *lbest* acts as the exemplar vector for the particle.

It uses a good exemplar vector to update the velocity of a particle until the fitness of the exemplar vector is better than the fitness of the particle being updated. If the fitness of the exemplar vector is no longer better than that of the particle, a new exemplar vector is opted. This learning procedure employs the following velocity update equation:

This algorithm also uses a repository for storing the non-dominated solutions.

$$v_{id}(t+1) = w.v_{id}(t) + c_1.rand.(lbest_{fi(d)} - x_{id}(t)) \quad (12)$$

In this algorithm a repository is added to keep a historical record of the non-dominated solutions obtained during the search process. After adding the new solutions to the external archive, non-dominated sorting is performed on the repository.

Table 2: MOCLPSO Algorithm

(1) Randomly initialize the positions and velocity of all the nodes.
(2) Initialize all the general parameters of MOCLPSO
(3) Initialize P_c for each particle.
(4) For each particle X
DO
  a) WHILE whole network is not covered
  DO
    i. Select cluster-head X_i
    ii. Find the neighbors of the cluster head
  b) Remove the cluster-head '*i*' and its neighbors for next cluster-head selection process.
  c) END WHILE
(5) END FOR
(6) Evaluate each of the fitness of each particle.
(7) Find the Pareto front with non-dominated sorting.
(8) Store the particles having non-dominated vectors in the repository.
(9) Find the personal best *lbest* from the repository.
(10) WHILE maximum number of cycles are not reached
DO
  a) FOR each particle X
  DO
    i. Select the exemplar vector as described.
    ii. Update the velocity and position of each particle

$$v_{id}(t+1) = w.v_{id}(t) + c_1.rand.(lbest_{fi(d)} - x_{id}(t))$$
$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1)$$

    iii. Keep the particles within the boundaries.
    iv. Evaluate the fitness of each particle
    v. Update the *lbest* of each particle if required.
  b) END FOR
  c) Update the content of repository.
END WHILE

In the proposed algorithm, the factors like degree of nodes, transmission range and battery power consumption are optimized. This provides diversity of solutions, hence increasing the flexibility of choosing a solution according to the requirement. The random selection approach to find multiple trade-off solutions in a single run is fast and has low computational complexity. This new strategy provides multiple exemplars to learn from and hence resulting in geographical diversity and increased potential search space. It also reduces time complexity.

6. SIMULATION RESULTS AND ANALYSIS

This section deals with the performance analysis of the proposed algorithm and its comparison with the existing techniques. The implementation is done using MATLAB. The network size taken into consideration is 400mx400m for 300 nodes. The simulations are run for two scenarios by varying the sink positions: 1) sink is within the sensor network area, and 2) sink is outside the sensor network area. The comparison of performance of the proposed approach with other algorithms is done using two performance metrics: 1) number of nodes alive, and 2) total residual energy of the network.

Journal of Theoretical and Applied Information Technology
31st July 2014. Vol. 65 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

6.1 Number of Nodes Alive

Fig. 1(a). considers the scenario in which the sink is located inside the sensor area whereas in Fig. 1(b). it is located outside. However, in both the scenarios the network life time, in terms of number of nodes alive vs rounds, is better in case of MOCLPSO and MOPSO as compared to WCA, CLPSO, PSO and LEACH. Also, it is to be noted that MOCLPSO has faster execution owing to its lesser computational complexity. Thus the performance of the proposed algorithm supersedes other algorithms in terms of both network lifetime and speed.

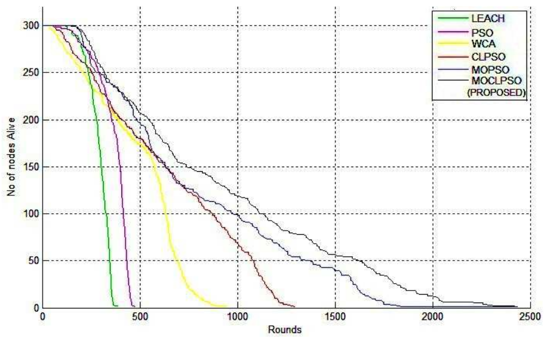

Figure 1(A): Number Of Nodes Alive Vs Rounds For Sink Within The Sensor Area

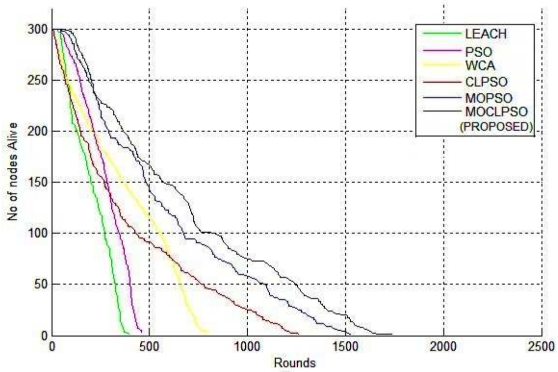

Figure 1(B). Number Of Nodes Alive Vs. Rounds For Sink Position Outside The Sensor Area

6.2 Residual Energy

The radio model is used according to which the energy loss of a node varies as the distance between the transmitter and receiver (r) and is proportional to r^2. Hence the energy spent by the transmitter to transmit l bit data over a distance d is given by

$$E_{TX}(l,d) = \begin{cases} l.E_{elec} + l.E_{FS}.d^2 & ,if \quad d < d_o \\ l.E_{elec} + l.E_{TR}.d^4 & ,if \quad d \geq d_o \end{cases} \qquad (13)$$

where, E_{elec} is the energy dissipated per bit to run the transmitter or receiver circuit. E_{FS} and E_{TR} depend on transmitter amplifier model. d_o is the threshold transmission distance. To receive this message the radio expends energy,

$$E_{RX}(l) = l.E_{elec} \qquad (14)$$

Fig. 2 represents total residual energy vs. rounds for two different sink positions. The total residual energy of the network in case of MOCLPSO and MOPSO is found to be greater at any instant of time as compared to CLPSO. WCA, PSO and LEACH. The reduction in amount of energy consumption for MOCLPSO and MOPSO ascertains these algorithm to be energy efficient and hence enhancing the network lifetime.

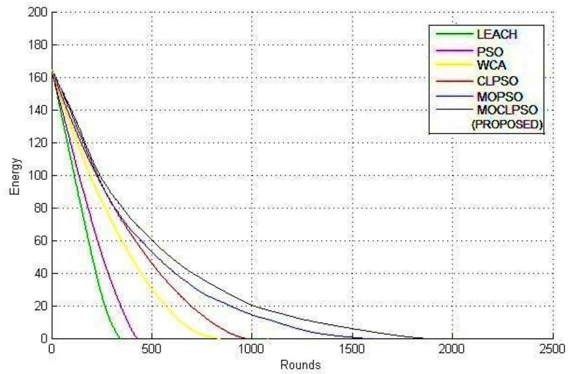

Figure 2(a). Total Residual Energy Vs Rounds For Sink Position Within The Sensor Area

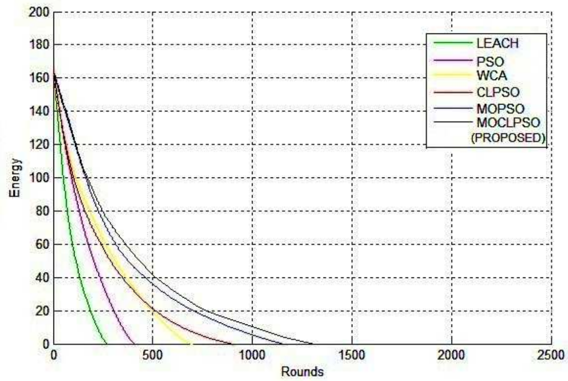

Figure 2(B). Total Residual Energy Vs. Rounds For Sink Position Outside The Sensor Area

7. CONCLUSION

In this paper, a novel approach for energy efficient clustering is presented. All the major network challenges and constraints are taken into consideration while proposing the technique. The

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

major challenge faced in mobile ad hoc networks is due to limited battery power. The proposed algorithm provides an energy efficient solution, thus increasing the network lifetime. The new strategy gave the particles larger potential search space. Thus, the new MOCLPSO algorithm has better global search ability. The algorithm provides not just a single solution (as is the case with classical approaches), but a set of Pareto-optimal solutions (i.e. a set of non-dominating solutions) to a decision maker. This set of non-dominating solutions are stored using an external repository. It is obvious that a set of Pareto-optimal solutions provides diversity and flexibility of the solutions. The users can choose the solution as per their need.

Also, this novel technique minimizes the number of clusters, and hence helps reduce the routing cost of a packet. It also makes the routing energy-efficient because less number of nodes are involved for routing a packet. The simulation results show that it is an effective and flexible approach. The speed of execution for the proposed algorithm is faster, thus reducing the time-complexity. The results of the proposed MOCLPSO-based approach is compared to well-known clustering techniques: MOPSO, CLPSO, WCA, PSO and LEACH. The results are found to outperform these techniques in finding optimal number of clusters, providing multiple options for users and also increasing the network lifetime.

In future, a scheme similar to the one used in the presented work can be proposed where instead of using a single repository from which the *lbest* particle is selected to guide the search; *n* repositories can be used which will reflect specialized knowledge in different and particular areas of exploration of the Pareto front. This work can also be extended to solve dynamic problems that involves detection of changes in the environment setup, like changes in the number of nodes. Further, adding more objectives and optimizing different parameters can be implemented using the algorithm.

REFRENCES:

[1] Hamid Ali, Waseem Shahzad, Farrukh Aslam Khan. "Energy-efficient clustering in mobile ad-hoc networks using multi-objective particle swarm optimization". *Applied Soft Computing*, Vol. 12, no. 7 (2012): 1913-1928.

[2] W. Heinzelman, H.Balakrishnan, A. Chandrakasan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks", *Proceedings of 33rd Hawaii International Conference on System Sciences*, 2000, pp. 1-10.

[3] Meenakshi Sharma, Kalpana Sharma, "An Energy Efficient Extended LEACH (EEE LEACH)", *Communication Systems and Network Technologies (CSNT),* 2012 International Conference on Communication Systems and Network Technologies, IEEE, 2012, pp. 377-382.

[4] N. Fan and Y. L. Song, "Improvement on LEACH Protocol of Wireless Sensor Networks", *Proceedings of 2007 International Conference on Sensor Technologies and Applications*, *Valencia*, 2007, pp. 260-264.

[5] Akanksha Mohan Gupte, Suryalok Sarkar, "A. Karthikeyan, Deploying a Single or a Double Cluster Head Particle Swarm Optimization Technique based on WSN scenarios", *International Journal of Engineering and Technology (IJET)*, Vol. 5, No. 2, May 2013, pp. 1887 – 1896

[6] James Kennedy, Russell Eberhart. "Particle swarm optimization", *Proceedings of IEEE international conference on neural networks*, vol. 4, no. 2, 1995, pp. 1942-1948.

[7] James Kennedy, "The particle swarm: social adaptation of knowledge", *In Evolutionary Computation,* 1997. IEEE International Conference on IEEE, 1997, pp. 303-308.

[8] Raghavendra V. Kulkarni, Ganesh Kumar Venayagamoorthy. "Particle swarm optimization in wireless-sensor networks: A brief survey"". *Systems, Man, and Cybernetics, Part C: Applications and Reviews*, IEEE Transactions on 41, no. 2 (2011): 262-267.

[9] Zhang Ruihua, Jia Zhiping, Li Xin, Han Dongxue, "Double Cluster-Heads Clustering Algorithm for Wireless Sensor Networks Using PSO", In *Industrial Electronics and Applications (ICIEA),* 2011 6th IEEE Conference on IEEE, 2011, pp. 763-766.

[10] Mainak Chatterjee, Sajal K. Das, Damla Turgut, "WCA: A weighted clustering algorithm for mobile ad hoc networks", *Cluster Computing,* Vol. 5, no. 2 (2002), pp. 193-204.

[11] An, Jing, Chang Li, and Bin Li, "A improved weight based clustering algorithm in mobile ad hoc networks", In *Information, Computing and Telecommunication, 2009. YC-ICT'09.* IEEE Youth Conference on IEEE, 2009, pp. 220-223.

[12] Waseem Shahzad, Farrukh Aslam Khan, Abdul Basit Siddiqui. "Weighted Clustering using Comprehensive Learning Particle Swarm

Journal of Theoretical and Applied Information Technology
31st July 2014. Vol. 65 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Optimization for Mobile Ad Hoc Networks", *International Journal of Future Generation Communication & Networking*, Vol. 3, no. 1 (2010), pp. 342-349.
[13] Carlos A. Coello Coello, Maximino Salazar Lechuga. "MOPSO: A proposal for multiple objective particle swarm optimization". In *Evolutionary Computation, 2002. CEC'02.* Proceedings of the 2002 Congress on IEEE, 2002, vol. 2, pp. 1051-1056.
[14] Carlos A. Coello Coello, Gregorio Toscano Pulido, and M. Salazar Lechuga. "Handling multiple objectives with particle swarm optimization". *Evolutionary Computation,* IEEE Transactions on 8, no. 3 (2004): 256-279.
[15] S. Mostaghim, J. Teich, "Covering pareto-optimal fronts by subswarms in multi-objective particle swarm optimization", *In Congress on Evolutionary Computation*, vol. 2, 2004, pp. 1404–1411.
[16] Jing J. Liang, A. Kai Qin, Ponnuthurai N. Suganthan, S. Baskar. "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions". *Evolutionary Computation,* IEEE Transactions, Vol. 10, no. 3 (2006): 281-295.
[17] Shankar T., Shanmugavel S., Karthikeyan A., "Hybrid approach for energy optimization in wireless sensor networks using PSO", *International Review on Computers and Software,* Vol. 8, no. 6, 2013, pp. 1454-1459.
[18] Karthikeyan, A., Sarkar S., Mohan Gupte A. , Srividhya V., "Selection of cluster head using fuzzy adaptive clustering for energy optimization in wireless sensor network", *Journal of Theoretical and Applied Information Technology*, Vol. 53, no. 1, 2013, pp. 6 – 12.
[19] Shankar, T., Shanmugavel, S., Karthikeyan, A., Mohan Gupte, A., Sarkar, S., "Load balancing and optimization of network lifetime by use of double cluster head clustering algorithm and its comparison with various extended leach versions", *International Review on Computers and Software,* Vol. 8, no. 3, 2013, pp. 795 – 803.
[20] Nadeem, A., Shankar, T., Shanmugavel, S., Karthikeyan, A., "Multivariate Gaussian mean distributed network topology for energy efficient communication in wireless sensor networks", *International Review on Computers and Software,* Vol. 8, no. 5, 2013, pp. 1203 – 1208.
[21] Shanmugavel, S., Shankar, T., Karthikeyan, A., "Modified harmony search algorithm for energy optimization in WSN", *International Review on Computers and Software,* Vol. 8, no. 6, 2013, pp. 1469 – 1475.
[22] Shankar, T., Shanmugavel, S., Karthikeyan, A., Dhanabal, R., "Selection of cluster head using neural network in wireless sensor network", *European Journal of Scientific Research,* Vol. 83, no. 3, August 2012, pp. 320-337.
[23] Dhanabal, R., Shankar, T., Karthikeyan, A., Shanmugavel, S., Bharathi, V., "Improvement of spatial reuse and throughput using OCSMA protocol in wireless ad hoc networks", *Journal of Theoretical and Applied Information Technology*, Vol. 35, no. 2, 2012, pp. 215 – 224.
[24] Maizate, A., El Kamoun, N., "A new metric based cluster head selection technique for prolonged lifetime in wireless sensor networks", *International Review on Computers and Software,* Vol. 8, no. 6, 2013, pp. 1346-1355.