International Journal of Electronics Communication and Computer Engineering
Volume 7, Issue 5, ISSN (Online): 2249–071X

Local Search Binary Particle Swarm Optimization (LSBPSO) MAC Clustering With Genetic Algorithms for Wireless Sensor Networks

Lokesh.A
Assistant Professor, CSE Department, M S Engineering College, Bangalore, India

Abstract **– In the present effort, a Berkeley-Media Access Control (B-MAC) clustering protocol with a crossed Genetic Algorithm(GA) as well as Particle Swarm Optimization (PSO) methods for overwhelming the huddling issue through sighting of quantity of clusters, Cluster Heads as well as cluster members is projected Statistical Analysis: Wireless Sensor Networks (WSNs) are comprised of several magnitudes of minute nodes with constrained aptitudes. The main problem with these types of nets is the energy restraints with clustering emergent as the most resourceful resolution to the problem. The aim of bundling is the partition of networks into groups with every group retaining a Cluster Head (CH). The work of Cluster Head is meeting, accumulating and transferring data to Base Stations. Simulations using OPNET has been carried out in this study. The projected protocol enactment is experimented for packet delivery ratio, end to end delay, number of hops to destination and jitter with various node mobility levels. The result reveals that the Local Search Bi nary PSO (LSBPSO) MAC Clustering performs better when compared with BMAC with flooding and BMAC with cluster based routing in either static or dynamic scenarios. Application/Developments: Grounded on the enactment of innumerable MAC protocols, it is found that LSBPSO MAC clustering BMAC can be embraced for agility based WSN applications like military recon manoeuvres, tragedy management, security, healthcare systems, industrial mechanization and many others.**

Keywords – MEMS, Flooding, Cluster-Head (CH), Genetic Algorithm (GA), Medium Access Control (MAC), Particle SwarmOptimization (PSO), Wireless Sensor Network (WSN), wireless sensor and actuator networks (WSAN).

I. Introduction

Since of numerous developments and implausible development in Micro-Electro-Mechanical Systems (MEMS) technologies as well as wireless communication technologies, WSNs are revolving available as an brilliant tool for use in some presentations like military recon operations, tragedy managing, security, atmosphere watching, healthcare systems, industrial automation and many others. Hence, Wireless Sensor Networks have proficient admirable links between the physical world, calculating universe and civilisation. Generally, WSNs comprise of a excessive number of small sensor nodes which are blowout over a vast area with one or more effectual sinks or Base Stations (BS) which collect data from the sensor nodes. All nodes hold limited energy supply and are capable of sensing data, dispensation information as well as collaborating wirelessly.

MAC protocols are exploited for checking access to communal media for avoiding several issues of depletion of energy and successfully share resources between several sensor nodes. Energy operative MAC protocols monitor responsibility sequences of sensor nodes on the origin of accessible circulation, reducing indolent hearing resulting in reduced energy consumption. MAC protocols exploit actual schedulers for acclimatizing to numerous traffic outlines of networks. Virtually all schedulers are on the foundation of sensor node traffic with no consideration to discarded energy in nodes. The application of the leftover energy in the nodes should to be measured when describing node schedules and is substantial in attractive network enactments. Vast scale employ of sensor nodes results in great transmission packets overheads in Wireless Sensor Networks for all nodes transmit the sensed data to Base Stations, which frontrunners to energy wastage. For the mitigation of the issue, clustering has been exploited in manipulative Wireless Sensor Networks1.

Wireless sensor networks (WSN), sometimes called wireless sensor and actuator networks (WSAN) are spatially distributed autonomous sensorsto *monitor* physical or environmental conditions, such as temperature, sound,pressure,etc. and to cooperatively pass their data through the network to a main location. The more modern networks are bi-directional, also enabling *control* of sensor activity. The development of wireless sensor networks was motivated by military applications such as battlefield surveillance; today such networks are used in many industrial and consumer applications, such as industrial process monitoring and control, machine health monitoring, and so on.

The WSN is built of "nodes" – from a few to several hundreds or even thousands, where each node is connected to one (or sometimes several) sensors. Each such sensor network node has typically several parts: a radio transceiverwith an internal antennaor connection to an external antenna, a microcontroller,an electronic circuit for interfacing with the sensors and an energy source, usually a batteryor an embeddedform of energy harvesting.A sensor nodemight vary in size from that of a shoebox down to the size of a grain of dust, although functioning "motes" of genuine microscopic dimensions have yet to be created. The cost of sensor nodes is similarly variable, ranging from a few to hundreds of dollars, depending on the complexity of the individual sensor nodes. Sizeand cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, computational speed and communications bandwidth. The topology of the WSNs can vary from a simple star networkto an advanced multi-hop wireless mesh network.The propagation technique between the hops of the network can be routingor flooding.

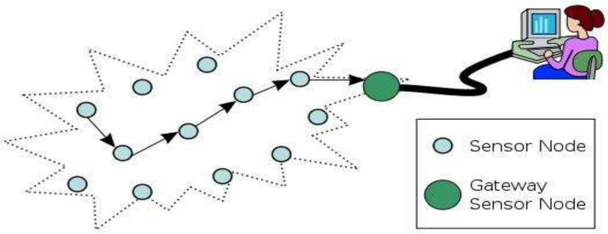

Fig.1. Wireless sensor networks

Clustering reductions the quantity of communications to Base Stations because Cluster Heads are in charge of the communications of all clusters. Clustering is well-known for its ascendable environment because it proposals load balancing as well as operative resource usage over the clubbing of nodes in geographical immediacy as clusters.

The features of designing excellent MAC protocols for WSNs2: Energy Efficiency, Latency, Fairness MAC sub-layer protocols for Wireless Sensor Networks ought to handle the energy-related problems specified below:

Collisions

Collisions occur in the event of two nodes transmitting concurrently.

Overhead

Additional important issue is switch packet overheads.

Overhearing

Alternative issue is overhearing where in sensor nodes may attain packets for which they are not the targets. The nodes should to turn off their radios to sanctuary energy.

Idle Listening

This denotes the energy utilized by nodes for keeping the circuits ON as well as being able to receive data even when no activity is present in networks.

Complexity

Complexity signifies the energy exploited because of desires to carry out operationally expensive algorithms and models. A major goal in designing Wireless Sensor Networks therefore is simplicity, while others are fairness, latency, throughput as well as bandwidth.

The problems presented previously are obviously linked to the main issue of optimization. Making lifecycle maximum as well as satisfying Quality of Service (QoS) basics as well as maintaining security is a problematic task. Recurrently, the three goals pledge one another. For example, if energy efficacy is of the highest importance, quality of service as well as security is inferior. Otherwise, if quality of service is preserved, the other two are substandard. Hence, for optimizing Wireless Sensor Networks, the precise option for conduct all the issues in the network is substantial

Numerous optimization procedures owe their stimulus to environment. Evolutionary Algorithms (EAs) and Swarm Optimization Algorithms are two groups of algorithms that are bio-inspired. EAs effort to mimic natural assortment method, wherein every generation of species look for beneficial adaptations in a constantly dynamic environment.

In 5 suggested a novel Cross-Layer MAC protocol (CL-MAC) for WSNs, for the efficient handling of multi-packet, multi-hops as well as multi-flow traffic patterns and at the same time adjusting to a great variety of traffic loads. It is different from other MAC protocols in that it does not support creation of multi-hop flows. Rather, CL-MAC regards all queued packets in the routing layer buffers and all flow setup demands from neighbours, for determining flows. This ensures that CL-MAC can take better informed scheduling decisions, with knowledge of current network status apart from dynamically optimizing scheduling technique correspondingly. During simulations, CL-MAC considerably decreases end-to-end latencies, improves delivery ratios and decreases average energy utilized per packet transmitted.

In 6 suggested an energy effective MAC protocol for WSNs which obviates overhearing and decreases contentions and delays through asynchronous scheduling of waking times of neighbourhood nodes. Energy utilization analysis for multi-hop networks may be provided. For validation of design as well as for analysis, the method was executed in TinyOS. Simulations revealed that AS-MAC significantly decreased energy utilization, packet losses as well as delays as opposed to other energy effective MAC protocols.

In7 studied several methods for node positioning to have decreased energy utilization with coverage conserved in Wireless Sensor Networks. Genetic Algorithms were utilized for creating energy effective node positioning in Wireless Sensor Networks. Simulations revealed that the suggested model expanded network lifecycle for several network positioning methods.

In8 suggested a clustering method for energy balance on the basis of genetic clustering path algorithms. The novel model combined GA as well as Fuzzy C-Means (FCM) for overcoming sensitivities of initial values of FCM. Optimal clusters may be formed and then Cluster Heads may be chosen for all groups. Simulations revealed that the suggested model outperformed Low Energy Adaptive Clustering Hierarchy (LEACH), in balancing energy costs of nodes as well as prolonging network life-cycle effectively.

In 9 suggested Linear/Nonlinear Programming (LP/NLP) formulations of the issues along with two new models grounded in PSO. The routing model was built with effective particle encoding strategy as well as multi-objective fitness functions. The clustering model was suggested with consideration of energy preservation of nodes by using load balancing. The suggested algorithms were run through experiments and the outcomes contrasted them with already present methods and revealed their improved performance with regard to network lifetime, energy usage, dead sensor nodes as well as delivery of all information packets to Base Stations.

In 14 reference The main contributions of this paper are (i) for-mulation of a multi-objective optimization problem for cluster head selection in a sensor network for UAV data acquisition (ii) considering realistic models and constraints on bit error rate, energy of the nodes, flight time of the UAV, and wind effects on the UAV (iii) comparing with existing LEACH-C algorithm in optimization for WSN data collection.

II. Design Methods

In this section, the GA-PSO BMAC clustering has been proposed and described.

2.1. *Berkeley MAC (B-MAC)*

B-MACs are malleable to configurations and may be executed with minimal code as well as memory size. B-MAC comprises Clear Channel Assessment (CCA), packet back-offs as well as link layer acknowledgement. For CCA, B-MAC utilizes weighted dynamic average of samples when channels are idle for assessing background noises and for better detection of permissible packets as well as collisions. Packet back-off times are configurable and are selected from linear ranges unlike exponential back-off strategies which are generally utilized in other distributed systems. This decreases delays and functions due to general transmission patterns discovered in WSNs. B-MAC furthermore, supports packet by packet link layer acknowledgements. Through this method, merely significant packets are required to pay the additional costs10.

B-MAC utilizes adaptive preambles for reducing idle hearing, which is a huge factor of energy wastage in several algorithms. When nodes have packets to transmit, they wait for a back-off time prior to checking channels. If the channels are clear, the node sends the data, else, it initiates another 'congestion' back-off. All nodes ought to check channels in a periodic manner utilizing Low-Power Listening (LPL); if channels are idle and nodes have no information to send, nodes switch back

Indian Journal of Science and Technologyto sleep mode. B-MAC preamble sampling strategy alters the interval within which channels are checked to be the same as frame preamble sizes. For instance, if mediums are ascertained every 100 ms, preambles of the packets ought to last 100 ms at least for receivers to sense the packets. Upper layers might alter preamble durations, as per application needs.

A benefit to utilizing B-MAC in Wireless Sensor Networks is that they do not utilize Request to Send (RTS), Clear to Send (CTS), ACK or other control frames automatically, but they may be appended if necessary. Furthermore, B-MAC is one of the very few specialized MAC protocols whose execution was evaluated in hardware. Synchronization is not needed and the protocol's performance may be fine-tuned by higher layers for fulfilling the requirements of different applications. The primary drawback is that preambles result in huge over-heads. For instance, 271 bytes of preamble are required for transmitting 36 bytes of information.

The first active node broadcasts control messages when reconfiguration terminates while remaining nodes flood one time to connect with neighbours in this method. A node expends energy to transmit one up message and on receiving of several up messages from remaining nodes, polls the channel and sleeps for the remainder of the time.

It presumes polling interval for LPL in the course of reconfiguration is *Tp*. It is to be remembered that *Tp* may vary from *T*. *lpl* For waking up neighbours, nodes flood up messages with preamble *Tp*.

In the process of flooding, nodes require to send up message once. It may be assumed that average carrier sense is t*cs*, and communication time for up packet is *tup*. A node's energy expended on communication is:

() + +*PtPTtlcs s p up*

Nodes receive n packets from n neighbouring nodes. On average it overlistens*T*2 *p* preamble for one packet. Power it expends in reception is:

() +*nPTt*2 *l p up*

Because nodes reboot in uniform distribution, average waiting period prior to flooding for nodes is *Td*. Hence LPL cost for all nodes is:

PtTTpollp d p

The final part of power utilized is sleep cost:

()−*PTtTTslp p p d p*

Through these equations, mean energy cost in the course of reconfiguration is obtained as:

( ) ( ) ( ) = + + + + ++ −*E Pt P T tnPT t P t TT P T tTT* 2
flood l cs s p up l p up poll p d p slp p p d p

The formula above reveals a trade off with *Tp*. Incrementing *Tp* decreases channel sampling frequencies and protects nodes from expending power on polling. However it increments preamble size, therein raising communication as well as overhearing costs. To decrease *Eflood*, optimal *Tp* is required to be obtained from the formula below:

= *dEdT* 0 *flood p*

On the basis of data rates, B-MAC suggests a similar method for the optimization of polling periods. However the analysis is on the basis of periodic data traffic and ensures no closed form equation. Rather in the course of LPL with flooding, networks do not formulate periodic data and flooding of up messages remains the sole cause of traffic.

2.2. *Genetic Algorithms (GA)*

Genetic Algorithms are effective stochastic optimization search processes which imitate the adaptive evolution procedure present in nature. They are employed with great success in several NP-hard issues like multi-processor designs, task scheduling and optimizations among others. GA is effective mostly in issues with non-regular search spaces wherein global optima are needed. Conventional gradient based mechanisms of optimizing encounter issues when search spaces are multi-modal because they get forced into local maxima. GAs is less vulnerable to this issue of premature convergence.

GA is an iterative method, with trial and error, that aims at discovering global optima. The equivalent of nature is the procedure of evolving over a long duration wherein several members are generated and every population changes for the better, adapting to its surrounding environments. This simulates an evolution procedure through the creation of original pool of chromosome 4individuals wherein all individuals denote a generic solution for the issue it intends to resolve by following the steps given below11:

Generate an arbitrary population of N chromosomes (potential solutions for population).Valuate fitness functions f(x) of all chromosomes x in population. Create novel populations through an iteration of the steps below till novel populations reach population N:Choose two parent chromosomes from population, providing preference to fitter chromosomes (high f(x) values). In an automatic manner, copy fittest chromosome to the subsequent generation (this is known as elitism).

With specified crossover probability, crossover the parent chromosomes to generate two new offsprings. (If no crossovers are carried out, off springs are exact copies of parents).

With specified mutation probability, arbitrarily switch two genes in offspring.

Replace the fresh population instead of the existing population.

If loop stopping criterion is met, return most optimal solution in current population.

Else go to Step 2.

The procedure typically continues for a specified set of generations or till standard deviations of fitness converge toward zero (when standard deviation begins to converts, chromosome individuals are becoming more fit and so it has reached the most optimal solution it can discover). Presuming the initial population is huge enough, with fitness well delineated, it ought to have reached an excellent solution.

GAs does not discover best or most ideal solution. But if simulated evolutions are run several times, they end up with very good solutions. But it is interesting to note the procedure through which more fit genes are evolved. Part of the evolutionary spirals toward fitness is due to mutations which bring in novel gene sequences to the population, but most of the successes of GAs are due to crossovers. Through the combination of bits of fit chromosomes in novel ways and arbitrary crossovers, GAs evolves with time to become fit chromosome individuals.

The variables of GA are elaborated on:

2.2.1 *Population*

Populations refer to sets of individuals known as chromosomes which denote a finished solution to a specified issue. All chromosomes are sequences of 0s and 1s. The original set of population is a randomly formulated group of individuals. Novel populations are created through two techniques: Steady-state Genetic Algorithm and generational Genetic Algorithm12.

2.2.2 *Fitness*

In real life, fitness refers to an individual's capacity to hand over genetic tissue, reproducing and ensuring survival for further reproduction. Within Genetic Algorithms, fitness is evaluated by the function describing the issue. The fate of individual chromosomes relies on fitness values. The rate of survival is greater when there is improved fitness value.

2.2.3 *Selection*

Choosing individuals is performed through Roulette-Wheel technique. Here, the likelihood of being chosen is raised with the fitness values of individual chromosomes.

2.2.4 *Crossover*

The kinds of crossovers as well as mutations are significant for the performing of Genetic Algorithms' optimizations.

For producing fresh generations from chosen parents, several crossover points are chosen. Crossovers are employed with certain particular probabilities. These are fine-tuned after adequate experiments.

2.2.5 *Mutation*

Mutations are exploratory procedures that arbitrarily mutate genes for overcoming the restrictions of crossovers. The operations enable searches for best chromosomes through the transformation of Cluster Heads to cluster members and cluster members into Cluster Heads, with a minute likelihood. The likelihood of changing from cluster individual to Cluster Head it set higher than that of the reverse case for prevention of anomalous increase in the number of Cluster Heads. After crossovers and mutationsare executed, clusters ought to be reconfigured as ClusterHeads' positions might have altered.

2.2.6 *Population Generation*

WSN nodes are denoted as bits of chromosomes. Cluster Heads and individual nodes are denoted as 1s and 0s correspondingly. Fitness values of chromosomes are defined by many variables like node density as well as power utilization. Populations comprise many chromosomes while 5 the most optimal chromosome is utilized for the generation of subsequent generation. For the first population, huge quantities of arbitrary CHs are chosen. Depending on survival fitness, populations transform into the subsequent generations.

2.3 *Particle Swarm Optimization (PSO)*

Many applications in WSN use PSO as an effective algorithm for making clusters of nodes The idea behind PSO is to simulate the social behavior of bird flocks and fish schools. The PSO algorithmis a computational method that iteratively tries to improve a swarm of candidate solutions or particles, based on a few basic rules.

xixi + vi ; (1) and

vi !vi + cprp(pi xi) + cgrg(g xi) ; (2)

PSO of particle-to-particle interactions, it retains the location of best solution reached by any particle so far and is also attracted toward that particular solution, termed gbest13.

The first and second factors are termed cognitive and social components, correspondingly. Once iterations are done, pbest and gbest are updated for every particle if an improved or stronger solution (with regard to fitness) is discovered. This procedure is continued, repetitively, till either anticipated outcome is reached through convergence or it is noted that an admissible solution may not be discovered inside operational limits. For n dimensional search spaces, with particles of swarms are denoted by n-dimensional vectors:

= *Xxxx*(,,....,) *i iiin* 1 2

PSO is the most recent population based evolutionary optimization method that has its basis in the activity of flocking/schooling of birds/fish. For instance there is a set of birds, which are searching for food with no information regarding the right place but know the distance to the source. All birds may be given the data regarding its own best earlier position as well as the flock's best position and how to reach those two positions.

When the particle solutions are encoded using binary values (0 and 1), then it is termed as Binary PSO (BPSO). Within PSO, all solutions behave like birds in search space. All particles possess velocity as well, that displays the direction of the flow as well as fitness value that reveals how excellent the particle is. The fitness is computed by a certain function. PSO generates the initial population arbitrarily and preserves the best discovered location by all particles as well as most optimal discovered location by particles in iterations. Candidate solutions may be reached by the particle which keeps location as well as velocity updated on the basis of:

= + − + − + *V w V c rand P X c rand P X* * * 1()*( ) * 2()*( ) *i t i t i it g it* (1) () 1 ( ) 2 ( )

= + + + *X X V i ii* (t 1) (t) (t 1)

Wherein *Xi*(t) as well as *Vi*(t) are location and velocity of particle i in *tth* iteration, correspondingly, while *Pi* is the earlier most optimal location of particle i and *Pg* is the earlier most optimal location of all particles which have been discovered as of yet. W is inertia factor which monitors trade-off between local and global location direction. rand1 () as well as rand2 () are two arbitrary numbers from interval [0, 1]. Lastly, *cand c*1 2, are scaling constants which are typically = *cc*= 2.01 2 .

A basic variant of the PSO algorithm works by having a population (called a swarm) of candidate solutions(called particles). These particles are moved around in the search-space according to a few simple formulae. The movements of the particles are guided by their own best known position in the search-space as well as the entire swarm's best known position. When improved positions are being discovered these will then come to guide the movements of the swarm. The process is repeated and by doing so it is hoped, but not guaranteed, that a satisfactory solution will eventually be discovered.

Formally, let f: $\mathbb{R}^n \rightarrow \mathbb{R}$ be the cost function which must be minimized. The function takes a candidate solution as argument in the form of a vectorof real numbersand produces a real number as output which indicates the objective function value of the given candidate solution. The gradient of f is not known. The goal is to find a solution **a** for which$f(\mathbf{a}) \le f(\mathbf{b})$ for all **b** in the search-space, which would mean **a** is the global minimum. Maximization can be performed by considering the function $h = -f$ instead.

Let S be the number of particles in the swarm, each having a position $\mathbf{x}_i \in \mathbb{R}^n$ in the search-space and a velocity $\mathbf{v}_i \in \mathbb{R}^n$. Let$\mathbf{p}_i$be the best known position of particle$i$andlet **g** be the best known position of the entire swarm. A basic PSO algorithm is then:[9]

- For each particle i = 1, ..., S do:
  - Initialize the particle's position with a uniformly distributed random vector:$\mathbf{x}_i \sim U(\mathbf{b}_{lo},\mathbf{b}_{up})$, where $\mathbf{b}_{lo}$ and $\mathbf{b}_{up}$ are the lower and upper boundaries of the search-space.
  - Initialize the particle's best known position to its initial position: $\mathbf{p}_i \leftarrow \mathbf{x}_i$
  - If ($f(\mathbf{p}_i) < f(\mathbf{g})$) update the swarm's best known position: $\mathbf{g} \leftarrow \mathbf{p}_i$
  - Initialize the particle's velocity: $\mathbf{v}_i \sim U(-|\mathbf{b}_{up}-\mathbf{b}_{lo}|, |\mathbf{b}_{up}-\mathbf{b}_{lo}|)$
- Until a termination criterion is met (e.g. number of iterations performed, or a solution with adequate objective function value is found), repeat:
  - For each particle i = 1, ..., S do:
    - For each dimension d = 1, ..., n do:
      - Pick random numbers: $r_p, r_g \sim U(0,1)$
      - Update the particle's velocity: $\mathbf{v}_{i,d} \leftarrow \omega\, \mathbf{v}_{i,d} + \varphi_p r_p (\mathbf{p}_{i,d}-\mathbf{x}_{i,d}) + \varphi_g r_g (\mathbf{g}_d-\mathbf{x}_{i,d})$
    - Update the particle's position: $\mathbf{x}_i \leftarrow \mathbf{x}_i + \mathbf{v}_i$
    - If ($f(\mathbf{x}_i) < f(\mathbf{p}_i)$) do:
      - Update the particle's best known position: $\mathbf{p}_i \leftarrow \mathbf{x}_i$
      - If ($f(\mathbf{p}_i) < f(\mathbf{g})$) update the swarm's best known position: $\mathbf{g} \leftarrow \mathbf{p}_i$
- Now **g** holds the best found solution.

The parameters ω, φ_p, and φ_g are selected by the practitioner and control the behaviour and efficacy of the PSO method.

2.4 *Proposed Local Search Binary PSO LSBPSO) MAC Clustering*

The hybrid technique suggested here is known as Local Search Binary PSO (LSBPSO) MAC Clustering that incorporates both PSO as well GA as local search for improved performance. The evidences fundamental PSO as well as GA are equal as search space is navigated to decrease prediction of errors. Initially, every node in WSNs is flooded with local temporal standards with Hybrid Coefficient (HC) factors.

The driving limit of LSBPSO algorithm is HC. It expresses the percentage of population every repetition has evolved using Genetic Algorithm: Hence HC = 0 implies the process is solely PSO (the entire population is evolved as per particle swarm optimization), HC = 1 implies solely Genetic Algorithm, whereas 0<HC<1 implies that the respective percentage of population is updated by Genetic Algorithm, the outstanding using particle swarm optimization.

International Journal of Electronics Communication and Computer Engineering
Volume 7, Issue 5, ISSN (Online): 2249–071X

III. IMPLEMENTATIONS

In this section, the LSBPSO Cluster BMAC, BMAC with flooding and BMAC with cluster based routing methods are used. The Average Packet Delivery Ratios, Average End to End Delays in seconds, Average Number of hops to sink and Jitter are evaluated Figure 2 to 5 as shown as follows:

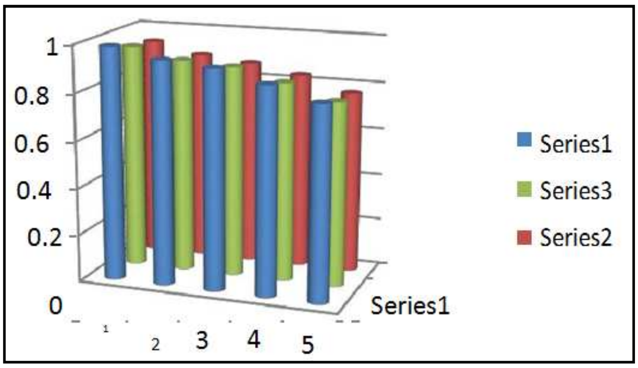

Fig. 2.Average packet delivery ratio

In the above chart the value 1 represents static, 2 represents 10KMPH,3 represents 20KMPH,4 represents30KMPH and 40KMPH standards.

From the Figure 2, it can be observed that the BMAC with cluster based routing increased Average Packet Delivery Ratio by 3.78%, 4.21%, 3.71%, 3.96% and 4.79% compared for LSBPSO Cluster BMAC and by 1.57%, 1.77%, 2.59%, 1.12% and 1.34% compared for BMAC with flooding when compared with various number of node mobility.

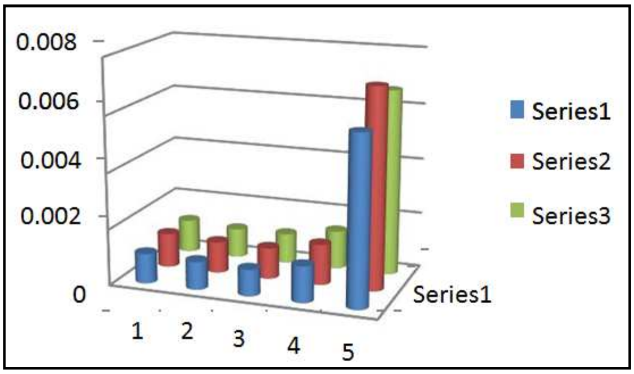

Fig. 3.Average end to end delays in second

In the above chart the value 1 represents static, 2 represents 10KMPH,3 represents 20KMPH,4 represents 30KMPH and 40KMPH standards.

LSBPSO Cluster BMAC and by 3.49%, 6.54%, 4.78%, 5.88% and 7.3% compared for BMAC with flooding when compared with various number of node mobility.

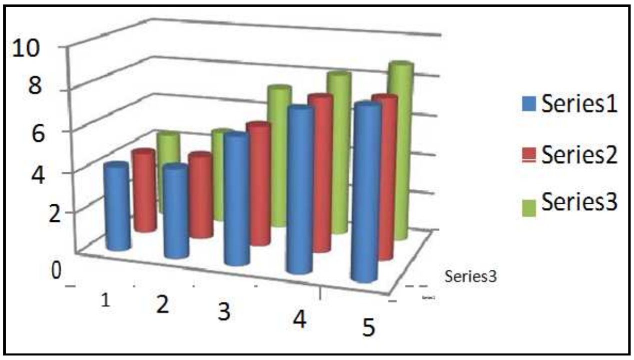

Fig. 4.Average number of hops to sink

In the above chart the value 1 represents static, 2 represents 10KMPH,3 represents 20KMPH,4 represents 30KMPH and 40KMPH standards.

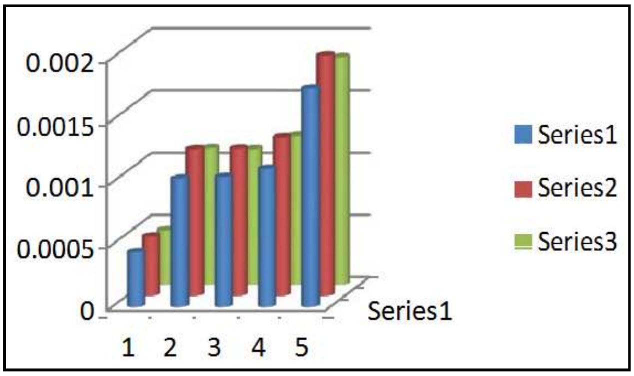

Fig. 5.Jitter

In the above chart the value 1 represents static, 2 represents 10KMPH,3 represents 20KMPH,4 represents 30KMPH and 40KMPH standards.

CONCLUSION

Clustering of the network relies on the Cluster Heads to send information to Base Stations. This reduces energy expended by sensor nodes to transmit information from other nodes to a Base Station, which potentially leads to improved network life as well as larger amount of data delivery during network life. In the current work, hybrid GA-PSO based clustering method which enhanced lifecycle of Wireless Sensor Networks efficiently was presented. Genetic Algorithm was used to select CHs and their quantity while Particle Swarm Optimization method was used for choosing cluster member nodes.

REFERENCES

[1] Sefuba M, Walingo T, Takawira F. Energy efficient medium access control protocol for clustered Wireless Sensor Networks with adaptive cross-layer scheduling. Journal of Sensors. 2015 Sep; 15(9):24026–53.

[2] Solehria SF, Jadoon S. Medium access control protocol for Wireless Sensor Network – A survey. International Journal of Video and Image Processing and Network Security IJVIPNS-IJENS. 2011 Jun; 11(03):16–21.

[3] Kaur S, Mahajan L. Power saving MAC protocols for WSNs and optimization of S-MAC protocol. International Journal of Radio

Frequency Identification and Wireless Sensor Networks. 2011; 1(1):1–8.

[4] Kaur L, Kumar D. Optimization techniques for routing in Wireless Sensor Network. IJCSIT. 2014; 5(3):4719–21.

[5] Hefeida MS, Canli T, Khokhar A. CL-MAC: A cross-layer MAC protocol for heterogeneous Wireless Sensor Networks. Ad Hoc Networks. 2013 Jan; 11(1):213–25.

[6] Jang B, Lim JB, Sichitiu ML. An asynchronous scheduled MAC protocol for Wireless Sensor Networks. Computer.

[7] Romoozi M, Ebrahimpour-Komleh H. A positioning method in Wireless Sensor Networks using Genetic Algorithms. Physics Procedia. 2012 Jan; 33:1042–9.

[8] He S, Dai Y, Zhou R, Zhao S. A clustering routing protocol for energy balance of WSN based on genetic clustering algorithm. IERI Procedia. 2012 Aug; 2:788–93.

[9] Kuila P, Jana PK. Energy efficient clustering and routing algorithms for Wireless Sensor Networks: Particle Swarm Optimization approach. Engineering Applications of Artificial Intelligence. 2014 Aug; 33:127–40.

[10] Patil UA, Modi SV, Suma BJ. A Survey: MAC layer protocol for Wireless Sensor Networks. International Journal of Emerging Technology and Advanced Engineering. 2013 Sep; 3(9):203–11.

[11] N. M. A. Lati_, C. C. Tsimenidis, and B. S. Sharif, Energy-aware clustering for wireless sensor networks using particle swarm optimization," in Proc. of the IEEE International Symposium on Personal, Indoor and Mo-bile Radio Communications, 2007, pp. 1{5}.

[12] B. Singh and D. Lobiyal, \A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks," Journal of Human-centric Computing and Information Sciences, vol. 2, pp. 1{18, 2012.

[13] J. Kennedy and R. Eberhart, \Particle swarm optimization,"in Proceedings of IEEE International Conference on Neural Networks, vol. 4, 1995, pp. 1942{1948}.

[14] Optimization of Wireless Sensor Network and UAV Data Acquisition