Indian Journal of Science and Technology *Vol 9(17), 10.17485/ijst/2016/v9i17/92985, May 2016*

ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Optimized Cluster with Genetic Swarm Technique for Wireless Sensor Networks

M. V. Ramana Rao[1]* and T. Adilakshmi[2]

[1]CSE, University College of Engineering, Osmania University, Hyderabad - 500007, Telangana, India; mvramana72@gmail.com
[2]CSE, Vasavi College of Engineering, Hyderabad - 500031, Telangana, India; t_adilakshmi@rediffmail.com

Abstract

Background/Objectives: In the current work, a Berkeley-Media Access Control (B-MAC) clustering protocol with a hybridized Genetic Algorithm (GA) as well as Particle Swarm Optimization (PSO) methods for overcoming the clustering issue through discovery of quantity of clusters, Cluster Heads as well as cluster members is proposed. **Methods/Statistical Analysis**: Wireless Sensor Networks (WSNs) are comprised of several quantities of minute nodes with restricted capabilities. The primary problem with these kinds of networks is the energy constraints. Plenty of research has been carried out in this field, with clustering emerging as the most efficient solution to the problem. The aim of clustering is the division of networks into groups with every group possessing a Cluster Head (CH). The job of Cluster Head is gathering, aggregating and transmitting data to Base Stations. Simulations using OPNET has been carried out in this study. **Findings:** The proposed protocol performance is tested for packet delivery ratio, end to end delay, number of hops to destination and jitter with various node mobility levels. The outcome reveals that the Local Search Binary 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/Improvements:** Based on the performance of various MAC protocols, it is found that LSBPSO MAC clustering BMAC can be adopted for mobility based WSN applications like military recon operations, disaster management, security, healthcare systems, industrial mechanization and many others.

Keywords: Cluster-Head (CH), Genetic Algorithm (GA), Medium Access Control (MAC), Particle Swarm Optimization (PSO), Wireless Sensor Network (WSN)

1. Introduction

Because of several developments and incredible progress in Micro-Electro-Mechanical Systems (MEMS) technologies as well as wireless communication technologies, WSNs are turning out as an excellent tool for use in several applications like military recon operations, disaster managing, security, environment monitoring, healthcare systems, industrial mechanization and many others. Hence, Wireless Sensor Networks have accomplished excellent links between the physical world, computing space and society. Generally, WSNs comprise of a great quantity of small sensor nodes which are spread over a vast area with one or more efficient sinks or Base Stations (BS) which gather data from the sensor nodes. All nodes possess restricted energy supply and are capable of sensing data, processing information as well as communicating wirelessly.

MAC protocols are utilized for monitoring access to shared media for obviating several factors of wastage of energy and effectively share resources amongst several sensor nodes. Energy effective MAC protocols monitor duty cycles of sensor nodes on the basis of available traffic, reducing idle hearing resulting in decreased energy wastage. MAC protocols utilize effective schedulers for adapting to various traffic patterns of networks. Almost all schedulers are on the basis of sensor node traffic with no consideration to leftover energy in nodes. The utilization of the leftover energy in the nodes ought to be considered when

*Author for correspondence

defining node schedules and is significant in enhancing network performances. Huge scale employment of sensor nodes results in great transmission packets overheads in Wireless Sensor Networks because all nodes transmit the sensed information to Base Stations, which leads to energy wastage. For the alleviation of the issue, clustering has been utilized in designing Wireless Sensor Networks[1].

Clustering decreases the quantity of communications to Base Stations because Cluster Heads are in charge of the communications of all clusters. Clustering is famous for its scalable nature because it offers load balancing as well as effective resource usage through the clubbing of nodes in geographical proximity as clusters.

The features given below are to be regarded when designing excellent MAC protocols for WSNs[2]:

- Energy Efficiency: The nodes gain power through batteries and typically it is tedious to recharge the battery. Hence, it is more advantageous to substitute the nodes themselves rather than recharging their power supply.
- Latency: Latency needs fundamentally depend on the kind of application involved. With sensor network applications, the sensed events ought to be forwarded to sink nodes in real-time for adequate actions to be taken instantly.
- Throughput: Throughput needs also differ with varying applications. Few sensor network applications need sampling the data with fine temporal resolutions. With these applications, it is advantageous that sink nodes receive more information.
- Fairness: Bandwidths are severely restricted in several WSN applications and so it is required that sink nodes obtain data from all nodes in a fair manner. But amongst all those mentioned above, energy efficacy and throughputs are the primary ones. Energy efficacy may be improved through minimization of energy wastage.

MAC sub-layer protocols for Wireless Sensor Networks ought to handle the energy-related problems given below:

1.1 Collisions

Collisions happen in the event of two nodes transmitting concurrently. The packets may end up corrupt and it is necessary to retransmit them. Hence, great deal of time as well as energy is wasted because of these repetitive transmissions and receptions.

1.2 Overhead

Another significant issue is control packet overheads. The control packets do not possess any sort of application data but are necessary for transmissions. Transmissions as well as receptions of packets are overheads for WSNs[3].

1.3 Overhearing

Another issue is overhearing wherein sensor nodes may obtain packets for which they are not the targets. The nodes ought to turn off their radios to preserve energy.

1.4 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. This is certainly a huge issue in Wireless Sensor Networks because nodes utilize channels in an arbitrary manner. Schemes for turning nodes ON or OFF are significant for Wireless Sensor Networks.

1.5 Complexity

Complexity denotes the energy utilized because of needs to carry out operationally costly 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 explicitly linked to the main issue of optimization. Making lifecycle maximum as well as fulfilling Quality of Service (QoS) requisites as well as maintaining security is a difficult task. Frequently, the three goals counter one another. For instance, if energy efficacy is of the highest importance, quality of service as well as security is inferior. Otherwise, if quality of service is maintained, the other two are inferior. Hence, for optimizing Wireless Sensor Networks, the correct option for handling all the issues in the network is significant. The model that is selected for optimization relies on several factors such as kind of method, kind of the issue at hand, required quality of service, present resources, time restrictions and so many more. Optimizer's nature is the determinant of whether it is adequate for a certain kind of issue[4].

Several optimization algorithms owe their inspiration to nature. Evolutionary Algorithms (EAs) and Swarm Optimization Algorithms are two groups of algorithms that are bio-inspired. EAs attempt to mimic natural selection process, wherein every generation of species look for advantageous adaptations in a constantly dynamic environment. GA as well as Differential Evolution (DE) algorithms are examples of Evolutionary Algorithms. PSO, Ant Colony Optimizations (ACO) as well as Bee Colony Optimizations (BCO) are examples of Swarm Optimization Algorithms.

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.

In[7] 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.

In[8] 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 lifecycle 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.

2. Methodology

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 costs[10].

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 to 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 overheads. 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 T_p. It is to be remembered that T_p may vary from T_{lpl}. For waking up neighbours, nodes flood up messages with preamble T_p.

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 t_{up}. A node's energy expended on communication is:

$$P_l t_{cs} + P_s \left(T_p + t_{up}\right)$$

Nodes receive n packets from n neighbouring nodes. On average it overlistens $T_p/2$ preamble for one packet. Power it expends in reception is:

$$nP_l\left(T_p/2 + t_{up}\right)$$

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

$$P_{poll} t_p \, T_d/T_p$$

The final part of power utilized is sleep cost:

$$P_{slp}\left(T_p - t_p\right)T_d/T_p$$

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

$$\begin{aligned} E_{flood} = {} & P_l t_{cs} + P_s\left(T_p + t_{up}\right) \\ & + nP_l\left(T_p/2 + t_{up}\right) \\ & + P_{poll} t_p \, T_d/T_p \\ & + P_{slp}\left(T_p - t_p\right)T_d/T_p \end{aligned}$$

The formula above reveals a trade off with T_p. Incrementing T_p 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 E_{flood}, optimal T_p is required to be obtained from the formula below:

$$\frac{dE_{flood}}{dT_p} = 0$$

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 individuals wherein all individuals denote a generic solution for the issue it intends to resolve by following the steps given below[11]:

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 Algorithm[12].

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 mutations are executed, clusters ought to be reconfigured as Cluster Heads' 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 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)

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 gbest[13].

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, i^{th} particles of swarms are denoted by n-dimensional vectors:

$$X_i = (x_{i1}, x_{i2}, ..., x_{in})$$

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:

$$V_i^{(t+1)} = w * V_i^{(t)} + c_1 * rand1() * (P_i - X_i^{(t)})$$
$$+ c_2 * rand2() * (P_g - X_i^{(t)})$$

$$X_i^{(t+1)} = X_i^{(t)} + V_i^{(t+1)}$$

Wherein $X_i^{(t)}$ as well as $V_i^{(t)}$ are location and velocity of particle i in t^{th} iteration, correspondingly, while P_i is the earlier most optimal location of particle i and P_g 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, c_1 *and* c_2, are scaling constants which are typically $c_1 = c_2 = 2.0$.

2.4 Proposed Local Search Binary PSO (LSBPSO) MAC Clustering

The hybrid method 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 premises underlying PSO as well as GA are identical as search space is traversed to reduce prediction of errors. Originally, every node in WSNs is flooded with local temporal values 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 remaining using particle swarm optimization. If HC factor is between 0.486-0.789 then value is flooded on the basis of selective flooding method[14]. The Pseudo Code for the proposed LSBPSO method:

Function GA=PSO (F, fit, i, m, h)
Start
Set particle
Do
For every particle
Compute fitness function of particle i(m)
I i(m) is better than Ffit
Initialize current value as fresh Ffit
End_For
Initialize hfit to best fitness of ∀ particles
For ∀ particles
Compute particle rate as per
$V_{id} = V_{id} + n_1 r_1 (P_{id} - X_{id}) + n_2 r_2 (P_{id} - X_{id})$
Keep particle location updated using the formula
$X_{id} = X_{id} + V_{id}$

End_For
Ascertain ∀ particle
For ∀ iteration
Create Local condition for hfit
Initialize Ffit for maximum
Compute connection Matrix
Compute Fit ratio and prediction error
End_While if maximum recursions are reached
Stop

3. Results and Discussion

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 from the Table 1 to 4 and Figure 1 to 4 as shown as follows:

Table 1. Average packet delivery ratio

Node mobility	LSBPSO Cluster BMAC	BMAC with flooding	BMAC with cluster based routing
Static	0.9883	0.9366	0.9515
10 KMPH	0.9464	0.8912	0.9072
20 KMPH	0.9271	0.8703	0.8932
30 KMPH	0.8772	0.8337	0.8431
40 KMPH	0.8193	0.7705	0.7809

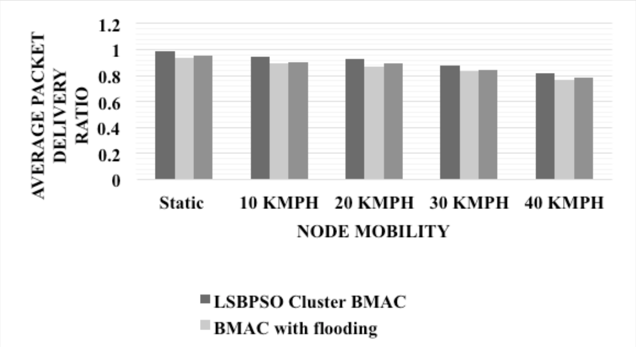

Figure 1. Average packet delivery ratio.

From the Figure 1, it can be observed that the BMAC with cluster based routing increased Average Packet Delivery Ratio by 3.79%, 4.22%, 3.72%, 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.

Table 2. Average end to end delays in second

Node mobility	LSBPSO Cluster BMAC	BMAC with flooding	BMAC with cluster based routing
Static	0.00107	0.001222	0.00118
10 KMPH	0.001	0.001121	0.00105
20 KMPH	0.00094	0.001112	0.00106
30 KMPH	0.00129	0.001453	0.00137
40 KMPH	0.00601	0.007154	0.00665

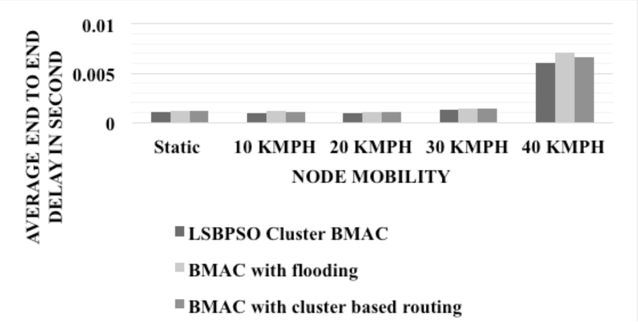

Figure 2. Average end to end delays in second.

In Figure 2, it can be seen that the BMAC with cluster based routing decreased Average End to End Delays in seconds by 9.77%, 4.87%, 12%, 6.01% and 10.11% compared for 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.

Table 3. Average number of hops to sink

Node mobility	LSBPSO Cluster BMAC	BMAC with flooding	BMAC with cluster based routing
Static	4.2	4.1	4.3
10 KMPH	4.4	4.2	4.7
20 KMPH	6.2	6	7.2
30 KMPH	7.7	7.6	8.1
40 KMPH	8.1	7.8	8.8

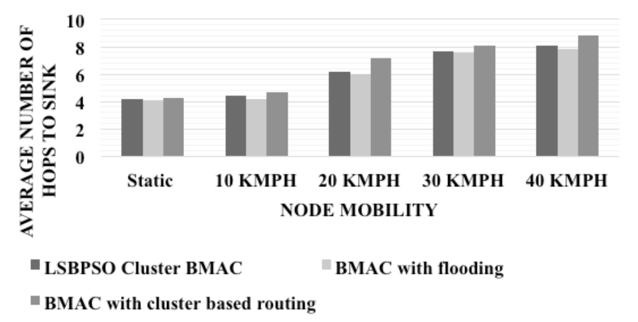

Figure 3. Average number of hops to sink.

From the Figure 3, it can be observed that the BMAC with cluster based routing decreased Average Number of hops to sink by 2.35%, 6.59%, 14.92%, 5.06% and 8.28% compared for LSBPSO Cluster BMAC and by 4.76%, 11.23%, 18.18%, 6.36% and 12.04% compared for BMAC with flooding when compared with various number of node mobility.

Table 4. Jitter

Node mobility	LSBPSO Cluster BMAC	BMAC with flooding	BMAC with cluster based routing
Static	0.000441	0.000478	0.000437
10 KMPH	0.001034	0.001179	0.001095
20 KMPH	0.001047	0.001183	0.001088
30 KMPH	0.001113	0.001274	0.001197
40 KMPH	0.001759	0.001932	0.001827

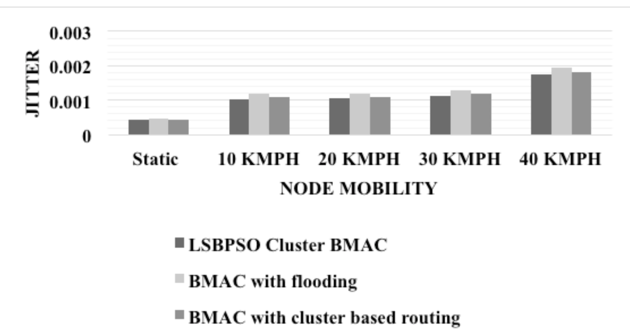

Figure 4. Jitter.

From the Figure 4, it can be observed that the BMAC with cluster based routing reduced jitter by 0.91%, 5.73%, 3.84%, 7.27% and 3.79% compared for LSBPSO Cluster BMAC and by 8.96%, 7.38%, 8.36%, 6.23% and 5.58% compared for BMAC with flooding when compared with various number of node mobility.

4. Conclusion

Clustering of the network relies on the CHs to send information to BS. 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. Outcomes are evaluated from the BMAC with cluster based routing increased Average Packet Delivery Ratio by 3.79%, 4.22%, 3.72%, 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.

5. 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 Networks. 2013 Jan; 57(1):85–98.
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. Njini I, Ekabua OO. Genetic Algorithm based Energy Efficient Optimization Strategies in Wireless Sensor Networks: A Survey. ACSIJ Advances in Computer Science: An International Journal. 2014 Sep; 3(5):1–9.
12. Hussain S, Matin AW, Islam O. Genetic Algorithm for hierarchical Wireless Sensor Networks. Journal of Networks. 2007 Sep; 2(5):87–97.
13. Sarangi S, Thankchan B. A novel routing algorithm for Wireless Sensor Network using Particle Swarm Optimization. IOSRJCE. 2012 Jan; 4(1):26–30.
14. Lakshmanan L, Tomar DC. Optimizing localization route using particle swarm - A genetic approach. American Journal of Applied Sciences. 2014; 11(3):520–7.