# Performance Analysis of Hybridization of [PIO-GSO] Algorithms in Wireless Sensor Networks

K.THAMIZHMARAN, K.PRABU
Dept. of ECE, GCE, Bodinayakkanur, Theni, Tamilnadu, INDIA
PG & Research Dept. of CS, Sudharsan College of Arts & Science, INDIA.

*Abstract:* **-** In wireless sensor networks (WSN), clustering is treated as an energy efficient technique employed to achieve augmenting network lifetime. But, the process of cluster head (CH) selection for stabilized network operation and prolonged network lifetime remains a challenging issue in WSN. In this research, presents a novel Hybridization of Pigeon Inspired with Glowworm Swarm Optimization (HPIGSO) algorithm based clustering innovation in WSN. This innovative HPIGSO algorithm integrates the good characteristics of Pigeon Inspired Optimization (PIO) algorithm and Glowworm Swarm Optimization (GSO) algorithm. The proposed algorithm operates on three major stages namely initialization, cluster head selection and cluster construction. Once the nodes are deployed, the initialization process takes place. Followed by, Base Station (BS) executes the HPIGSO algorithm and selects the cluster heads effectively. Subsequently, nearby nodes joins the cluster head and becomes cluster members, thereby cluster construction takes place. Finally, the cluster members send the data to cluster heads which is then forwarded to the base station via inter-cluster communication. The performance of the proposed HPIGSO method has been evaluated and compared with QOGSO, PIOA-DS, ALO, GOA and FFOA. Finally the proposed HPIGSO algorithm provides prolonged the lifetime of WSN over the existing clustering techniques

*Keywords:* Clustering, Augmenting Network lifetime, PIO, GSO, optimization algorithm.

Received: December 27, 2021. Revised: October 24, 2022. Accepted: December 2, 2022. Published: December 29, 2022.

## 1. Introduction

Recently, WSN has become a predominant one which is highly efficient in real-time applications. WSN observes the atmosphere and predicts the modifications happening in target regions. Some of the physical changes in the environment were vibration, sound, pressure, humidity, intensity, temperature, and so forth. The domain of WSN is applied in diverse areas such as armed forces, habitat monitoring [1], bio-medical sector, health observation, smart home tracking as well as inventory management system [2]. As an inclusion, clustering [3] is developed which helps in dividing the geographical region into tiny sectors. The main purpose of applying clustering is to divide the load equally to all nodes as head of the cluster, called as CH. The election of CH is one of the major tasks which helps in better data transmission. Practically, the cluster can contain a CH with maximum number of CM. The key objective of CH is to modify the nodes within a cluster [4]. However, the proper CH selection [5] with best potential is essential to manage the network's power-efficiency. Thus, the meta-heuristic approaches were Computational intelligence (CI) methods like Artificial Bee Colony (ABC), Artificial Immune Systems (AIS), Reinforcement Learning (RL), and Evolutionary Algorithms (EA) have been applied to proceed clustering task and to resolve NP-hard optimization problem. Transmitting the data to a BS or sink from the sensor node via optimal CH [6] is a complicated operation. The optimal CH selection process results in minimum power consumption, latency, distance etc. When compared with all other methods, the optimal CH election process in WSN remains a challenging issue. Various studies have been developed to determine the optimal CH selection process in WSN. Mehra et al. [7] presented a Fuzzy-Based balanced cost CH Selection method (FBECS) which has been constrained with residual energy (RE), distance and node density are considered to be the input for Fuzzy Inference System (FIS). For the selection of optimal CH, the Eligibility index has been determined for each node. Priyadarshini and Sivakumar [8] applied load balancing by triggering the Adelson-Velskii and Landis (AVL) tree rotation clustering approaches. The developers have divided the unique area network into massive clusters by novel and improved K-means clustering methods. Mann and Singh [9] projected an improved ABC with optimal solution search function for improvising system efficiency. Furthermore, a population sampling algorithm has been employed for a student's distribution. It is mainly used for enhancing the global convergence of deployed meta-heuristic approaches. Elhabyan et al. [10] established a Pareto optimization-relied method for handling the problems involved in finding best network configuration. In order to estimate the efficiency, the proposed technique has assumed a few metrics such as the number of CHs, the number of the clustered nodes, and link supremacy over CMs. A fractional ABC dependent based multi-objective CH selection (FABCMOCHS) approach has been presented as an energy effective clustering model to expand the sensor nodes' duration with improved network power [11]. FABC-MOCHS is mainly utilized for managing the convergence present in ABC by adding fitness function (FF) along with latency, travelling distance and power application to reduce the problem. A combination of ACO and ABC model-based clustering scheme (ACO-ABCA-CS) was presented for effective CH election under the mutual prevention of limitations [12]. The problem of stagnation in ACO and delayed convergence of ABC can be solved by mutual modification in exploitation and exploration phases. A dynamic scour bee-based CS (DSB-CS) has been projected for increasing the scout bee and maintaining the count of active nodes as well as CH power in a system [13]. It is highly applicable due to the advantages of ABC and FABC for increasing the duration of a network and power by using the best CH election approach. The concatenated Simulated Annealing as well as differential evolution-based CH selection (SADE-CHS) model has been established to enhance the power effectiveness by using clustering [14]. The SADE-CHS is mainly used for eliminating the overload of sensor nodes which is related with CH, as it is a major reason for immediate death of sensor nodes that leads to improper CH election process. It highly focuses on the network extension by removing the possibility of premature death of CH. A combined PSO as well as HSA-based CH selection approach has been applied to retain the energy balance as well as network duration [15]. PSO-HSA-CHS method was presented by integrating the dynamic ability of PSO and the higher exploring potential of HSA meta-heuristic approach for selecting the best CH in a system. To maximize the network lifetime, this paper presents a new HPIGSO algorithm based clustering innovative in WSN.

# 2. The Proposed HPIGSO Algorithm

Fig. 1 depicts the workflow of the presented HPIGSO algorithm. Once the nodes are deployed, the initialization process takes place. Followed by, BS executes the HPIGSO algorithm and selects the CHs effectively. Subsequently, nearby nodes join the CH and become CMs, thereby cluster construction takes place. Finally, the CMs send the data to CHs which is then forwarded to BS via inter-cluster communication.

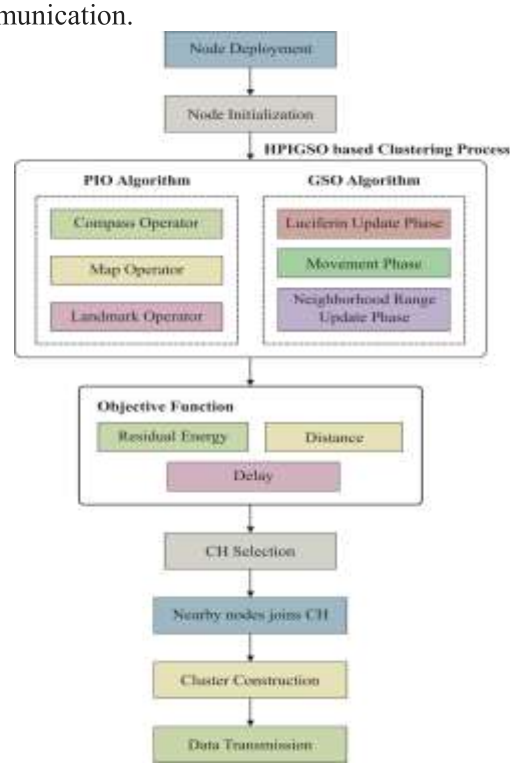


**Fig. 1.** Block diagram of HPIGSO algorithm

## 2.1. PIO Algorithm

The PIO algorithm is stimulated from the homing characteristics of pigeons. A pigeon is a familiar bird commonly employed for message passing by Egyptians, also by military forces. For idealizing the homing features of pigeons, 2 operators were developed by utilizing a few principles: Map and compass operator: the pigeon's predict the earth using magneto reception to design the map mentally. It considers the altitude of the sun as a range to alter the way. Landmark operator: If the pigeons fly near to a target, it relies on landmarks. When it is well-known with the landmarks, then it flies to the target.

When it is distant from the target and unknown to the landmarks, it is following the pigeons that are well-known through the landmarks.

#### 2.1.1. Map and compass operator

It determines the position $X_u$ and velocity $V_u$ of pigeon $u$ in a $D$- dimension search space which are updated for all iterations. A novel position $X_u$ and velocity $V_u$ of pigeon $u$ at the $z$ - th iteration is computed with the subsequent equations:

$$V_u(z) = V_u(z-1) \cdot e^{-Rz} + rand \cdot \left(X_g - X_u(-1)\right) \quad (1)$$

$$X_u(z) = X_u(z-1) + V_u(z) \quad (2)$$

where $R$ implies map and extent factor, *rand* is an arbitrary number, and $X_g$ is the present global optimal position, and that is attained by relating every the positions between each pigeon.

While an optimal position of each pigeon is assured by utilizing map and compass. With relating each domain position, it can be apparent which the right centred pigeon's location is optimized. All the pigeons alter the flying direction by subsequent and particular pigeons based on Eq. (1) that are illustrated by dark arrows. A thin arrow refers the former flying way relative to $V_u(z-1) \cdot e^{-Rz}$ in Eq. (2). A vector value of these 2 arrows is its subsequently flying way.

#### 2.1.2. Landmark operator

Here, maximum pigeons are reduced by $N_p$ in all generations. But, the pigeons are distant from the target, and it is different through the landmarks. Assume $X_c(z)$ be the center of any pigeon's position at the $z$th iteration, and assume all pigeons are flying directly to the target. A position updating rule to pigeon $u$ at the $z$th iteration is provided by:

$$N_P(z) = \frac{N_P(z-1)}{2} \quad (3)$$

$$X_c(z) = \frac{\sum \; X_u(z) \cdot fitness\left(X_u(z)\right)}{N_P \sum fitness\left(X_u(z)\right)} \quad (4)$$

$$X_u(z) = X_u(z-1) + rand \cdot \left(X_c(z) - X_u(z-1)\right) \quad (5)$$

where $fitness\,()$ is the efficiency of a pigeon. Followed by, it selects $fitness(X_u(z)) = \frac{1}{f_V(X_u(z))+\varepsilon}$. To maximize optimization problems, we have to select $fitness\,(X_u(z)) = f_{mx}(X_u(z))$. To all individuals pigeon, a better position of the Nc- th iteration is indicated with $X_p$ , and $X_p = \min\,(X_{u1}, X_{u2}, \dots, X_{uNc})$. The center of each pigeon is their aim in all iterations. The half of each pigeon which are distant from their target is following the pigeons which are near to their target which also implies that 2 pigeons can be at the similar position. A pigeon that is near to their target (a pigeon in encircle) would fly to the required place with sufficient speed.

### 2.2. Conventional GSO algorithm

Here, a swarm of glowworms are organized in a random manner on the solution space. A brighter individual implies an optimal position. Utilizing a probabilistic method, all agents are inspired by a neighbour with better luciferin intensity in local decision fields. A density of a glow worm's neighbors influences its decision radius and defines the size of its local decision field: if the neighbor density is minimum, a local decision field is extended; else, it may be limited to enable the swarms to divide into lesser groups. The above procedure is followed till reaching the termination criteria. Currently, maximum individuals collect brighter glowworms. Followed by, the GSO contains 5 important stages: luciferin-update stage, neighborhood select stage, moving probability computer stage, movement stage, and decision radius added stage.

#### 2.2.1. Luciferin Update stage

A luciferin update is based on the FF of preceding luciferin value, and regulation is provided by

$$l_u(z+1) = (1-\rho)l_u(z) + \gamma Fitness\,(x_u(z+1))\,. \quad (6)$$

where, $l_u(z)$ indicates the luciferin measure of glowworm $u$ at time $z, \rho$ represents the luciferin decompose constant, $\gamma$ implies the luciferin improvement constant; $x_u(z+1) \in R^M$ is the position of glowworm $u$ at time $z+1$ , and $Fitness\,(x_u(z+1))$ signifies the value of the fitness at glowworm $u$'s position location at time $z+1$.

#### 2.2.2. Neighborhood Select Phase.

The neighbors $N_u(z)$ of glowworm $u$ at $z$ time have the brighter ones and are expressed as

$$N_u(z) = \{v : d_{uv}(z) < r_d^u(z); l_u(z) < l_v(z)\}. \quad (7)$$

where, $d_{uv}(z)$ signifies the using Euclidean distance among glowworms $u$ and $v$ at time $z$, and $r_d^u(z)$ signifies a decision radius of glowworms $u$ at time $z$.

### 2.2.3. Moving Probability Computer Phase

The glowworm utilizes a possibility rule in order to send other glowworms containing superior luciferin level. A possibility $P_{uv}(z)$ of glowworm $u$ move towards the neighbor $v$ is expressed by:

$$P_{uv}(z) = \frac{l_v(z) - l_u(z)}{\sum_{k \in N_u(z)} l_k(z) - l_u(z)}. \quad (8)$$

### 2.2.4. Movement Phase

Assume glowworm $u$ chooses glowworm $v \in N_u(z)$ with $P_{uv}(z)$; the discrete- time method of glowworm $u$ is offered by (9)

$$x_u(z+1) = x_u(z) + s\left(\frac{x_v(z) - x_u(z)}{\|x_v(z) - x_u(z)\|}\right). \quad (9)$$

where, $\|\cdot\|$ signifies the Euclidean norm operator, and $s$ is the step- size.

### 2.2.5. Decision Radius Update Phase

During all updates, the decision radius of glowworm $u$ is provided as following:

$$r_d^u(z+1) = \min\{r_s, \max\{0, r_d^u(z) + \beta(n_z - |N_u(z)|)\}\}. \quad (10)$$

where, $\beta$ shows a constant, $r_s$ indicates the sensory radius of glowworm $u$, as well as $n_z$ refers to an attribute for balancing the adjacent value.

## 2.3. Hybridization of PIO and GSO algorithms

This section discusses the HPIGSO algorithm, which integrates PIO and GSO algorithms. Actually, the GSO algorithm has the ability to deal with non-linear, multimodal issues. But it gets stuck to solve high dimensional problems and fails to convergence faster. At the same time, the PIO algorithm has the ability of faster convergence.

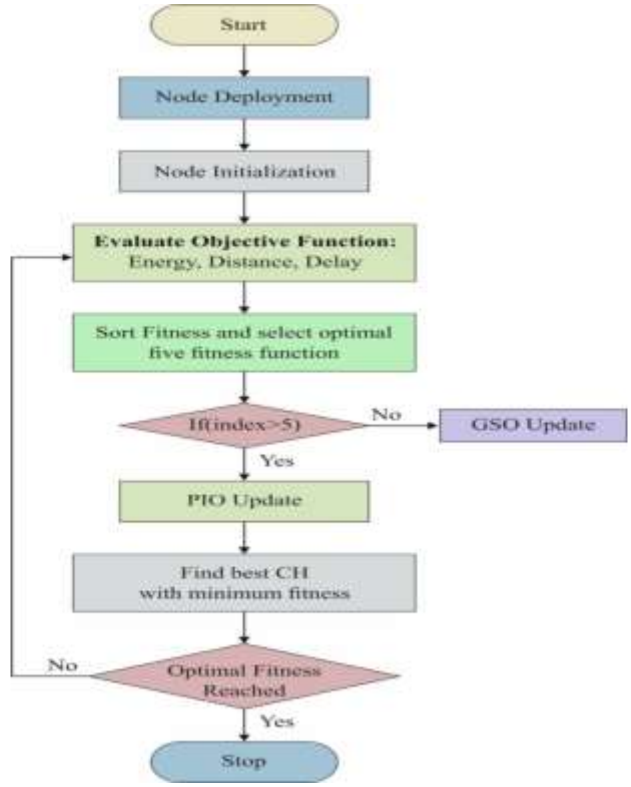


**Fig. 2.** Flowchart of HPIGSO algorithm

At this point, the validation of the objective function is carried out initially and the evaluated fitness gets sorted. Then, find the optimal five fitness values and choose an index. If it is higher than five, execute the PIO algorithm update, otherwise execute the GSO update. Through the hybridization, the multi-objective CHs election leads to minimum delay, and maximum energy saving. Besides, the negative searching ability is discarded by the HPIGSO technique, whereas the enhanced searching capability can be used for faster convergence. Therefore, the HPIGSO algorithm has achieved a better CH selection process. The processes involved in the HPIGSO method are illustrated in Fig. 2.

# 3. Performance Validation

This section investigates the experimental outcome of the HPIGSO technique under different dimensions. The presented HPIGSO technique has been simulated using MATLAB. Moreover, a set of measures applied to investigate the results of the network lifetime, network stability, count of active nodes as well as number of inactive nodes. The parameter settings involved in the experimentation are given in Table 1.

**Table 1** Parameter Settings

| **Parameters** | **Values** |
|---|---|
| Network Size | 100 x 100 $m^2$, 500 x 500 $m^2$ |
| No. of Nodes (N) | 100, 300, 500 |
| No. of BS | 1 |
| Initial energy ($E_0$) | 0.5 |
| Energy fraction for intermediate nodes ($\varphi$) and advanced nodes ($\omega$) | 1, 2 |
| No. of gateway nodes (m) and advanced nodes fraction ($m_0$) | m=0.1, $m_0$=0.2 |
| Energy need to transmit and receive $E_{elec}$ | 50nJ/bit |
| Threshold distance ($d_0$) | 80m |
| Amplifying power required for smaller distance $d \leq d_0$ ($E_{efs}$) | 10pJ/bit/$m^2$ |
| Amplifying power required for smaller distance $d \leq d_0$ ($E_{mp}$) | 0.0013 pJ/bit/$m^4$ |
| Energy utilization incurred when data aggregation ($E_{da}$) | 5 nJ/bit/signal |
| Data packet Size | 2000 bits |
| Population size (P) | 100 |
| Selection Method | Rank Selection Method |
| No. of generation | 30 |
| No. of runs | 20 |

### 3.1. Alive Nodes Analysis of HPIGSO technique on varying Node Count

Fig. 3 portrays the results analysis of HPIGSO technique interms of count of alive nodes within the node count of 100. The figure exhibited that the FFOA model has resulted in a least number of alive nodes over the compared methods. In line with that, the GOA and ALO algorithms have led to a higher and closer number of alive nodes. Continuing with this, the PIOA-DS algorithm has started to modify network lifetime with a somewhat supreme number in alive nodes. Simultaneously, the QOGSO algorithm has tried to exhibit near optimal results with the higher count of alive nodes. At last, the proposed HPIGSO technique has shown better outcome by attaining high alive nodes. For example, under the execution round of 1000, the HPIGSO technique has attained higher count of 51 alive nodes whereas the lowest of 42, 32, 16, 12 and 4 alive nodes are attained by QOGSO, PIOA-DS, ALO, GOA and FFOA algorithms. Likewise, under the execution round of 2000,the HPIGSO model has displayed a massive number of 18 alive nodes whereas the minimum of 14, 11, 6, 4 and 0 alive nodes are attained by QOGSO, PIOA-DS, ALO, GOA and FFOA algorithms.

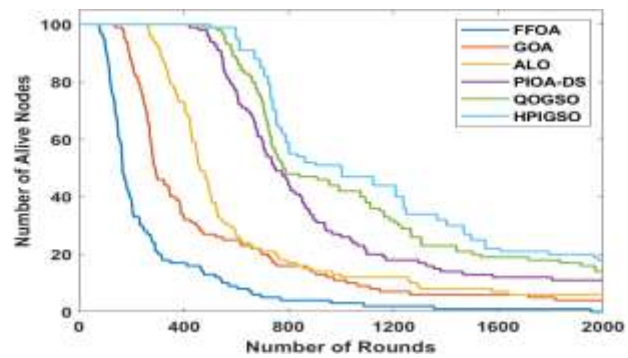


**Fig. 3.**Alive node analysis of HPIGSO technique under 100 nodes

### 3.2. Dead Nodes Analysis of HPIGSO technique on varying Node Count

Fig. 4 demonstrates the results analysis of the HPIGSO approach by means of number of inactive nodes within node value of 100. The diagram illustrates that the FFOA technique has provided a reasonable number of dead nodes than the previous modules. On continuing this, the GOA and ALO models have resulted in fewer and fewer dead nodes. Similarly, the PIOA-DS algorithm has been initialized to showcase better network lifetime with a lower normal number of inactive nodes. Meanwhile, the QOGSO approach has attempted to show near best results with a minimum number of dead nodes. Finally, the projected HPIGSO approach has exhibited moderate outcome by accomplishing lower count of dead nodes. For sample, under execution round of 1000, the HPIGSO model has achieved minimum number of 49 dead nodes while the maximum of 58, 68, 84, 88 and 96 dead nodes are reached by QOGSO, PIOA-DS, ALO, GOA and FFOA methodologies. In line with this, under the execution round of 2000,the HPIGSO scheme has depicted a lower number of 82 dead nodes and the maximum of 86, 89, 94, 96 and 100 dead nodes are achieved by QOGSO, PIOA-DS, ALO, GOA and FFOA algorithms.

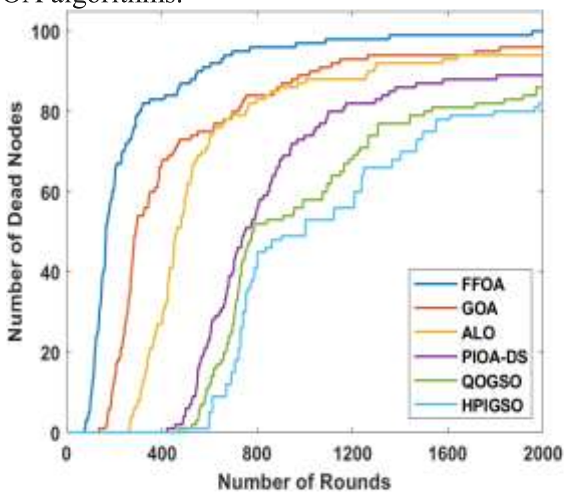


**Fig. 4.** Dead node analysis of HPIGSO technique under 100 nodes

### 3.3. Network Lifetime (Alive) Analysis of HPIGSO Algorithm on varying Node Count

Fig. 5 showcased the competing analysis of the HPIGSO method with respect to stability duration, HND and network lifespan. The figure illustrates that the HPIGSO model has achieved better network stability than the related technologies. Followed by, the HPIGSO technology has delayed the HND to a higher extent than previous models. Here, the HPIGSO scheme has demonstrated a maximum network lifetime. By seeking into the reached results, it is assured that the HPIGSO model has accomplished supreme function than alternate models.

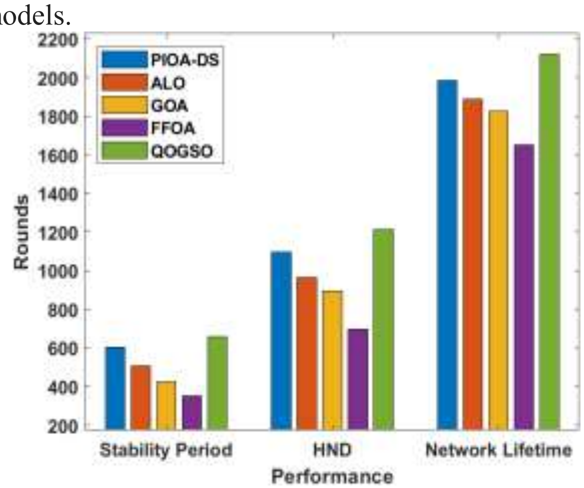


**Fig.5.** Network lifetime (alive) analysis of the HPIGSO method

The above mentioned figures indicated that the HPIGSO algorithm has achieved maximum network lifetime denoting the network stability, and stability period.

## 4. Conclusion

This paper has innovated a new HPIGSO algorithm based clustering technique in WSN, which integrates the characteristics of PIO and GSO algorithms. The proposed algorithm operates on three major stages namely initialization, CH selection and cluster construction. Once the nodes are deployed, the initialization process takes place. Followed by, BS executes the HPIGSO algorithm and selects the CHs effectively. Subsequently, nearby nodes joins the CH and becomes CMs, thereby cluster construction takes place. Finally, the CMs send the data to CHs which is then forwarded to BS via inter-cluster communication. The proposed HPIGSO algorithm involves an objective function using residual energy, distance and energy. The proposed method has the ability to select the CHs in an optimal way; thereby network (alive) lifetime can be maximized. An elaborate experimental validation takes place and the results of HPIGSO algorithm has attained maximum network lifetime compared to QOGSO, PIOA-DS, ALO, GOA and FFOA. In future, the network lifetime can be further increased by the use of data aggregation mechanisms.

*References*

[1] Madhuri, B., Aniruddha, G., Rahul, R., 2013. Identification and classification of flood prone areas using AHP, GIS and GPS. J. Disaster Adv. 6, 120–131.

[2] Sherifi, I., Baholli, I., 2015. Information Systems and Online Media in Albania: Challenges and Expectations, The 8th International Conference on Economic Sciences, Vienna, Austria, 8–15

[3] Zhu, E., Ma, R., 2018. An effective partitioned clustering algorithm based on new clustering validity index. Appl. Soft Comput. 71, 608–621.

[4] K.Thamizhmaran, M.Anitha and Alamelunachippan (2018) “Reduced End-To-End Delay for MANETS using SHSP-EA3ACK Algorithm”, Journal on Communication Engineering and System, Vol. 7, No. 3, pp. 8-15.

[5] K.Thamizhmaran (2018) "Performance Analysis of On-Demand Routing Protocols in MANETs Using (SHSP-EA3ACK) Algorithms", Ciit, International Journal of Digital Signal Processing, Vol. 10, No. 3, pp. 50-54.

[6] K.Thamizhmaran (2020), "EE-ATPSP – Evaluation Node Life Time for WSNs", i-manager's Journal on Wireless Communication Network Vol. 8, No. 4, pp. 27-35.

[7] Mehra, P.S., Alam, M.N.D., 2017. Zonal based approach for clustering in heterogeneous WSN. Int. J. Inf. Technol., 1–9

[8] Priyadarshini, R.R., Sivakumar, N., 2018. Cluster head selection based on minimum connected dominating set and bi-partite inspired methodology for energy conservation in WSNs. J. King Saud Univ. Comp. Inf. Sci. Available online 19.

[9] Kumar, R., & Kumar, D. (2015). Multi-objective fractional artificial bee colony algorithm to energy aware routing protocol in wireless sensor network. Wireless Networks, 22(5), 1461–1474.

[10] Sengathir, J. (2018). A hybrid ant colony and artificial bee colony optimization algorithm-based cluster head selection for IoT. Procedia Computer Science, 143(1), 360–366.

[11] K. Thamizhmaran, R. Santosh Kumar Mahto, and V. Sanjesh Kumar Tripathi (2012) "Performance Analysis of Secure Routing Protocols in MANET", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 1, No. 9, pp. 651-654.

[12] Akshayadevi Arivazhagan, K.Thamizhmaran and N.Thamilselvi (2015) "Performance Comparison of on Demand Routing Protocols under Back whole For MANET", Advance Research in Computer science and software Engineering, Vol. 5, No. 3, pp. 407 – 411.

[13] K.Prabu and K.Thamizhmaran (2016) "Cluster Head Selection Techniques and Algorithm for Mobile Ad-hoc Networks (MANETS)", Advance Research in Computer science and software Engg, Vol.6, No. 7, pp. 169-173.

[14] K.Thamizhmaran, M.Anitha and Alamelunachippan (2017) "Comparison and Parameter Adjustment of Topology Based (S-EA3ACK) for MANETs", International Journal of Control Theory and Application, Vol. 10, No. 30, pp. 423-436.

[15] K.Thamizhmaran, M.Anitha and Alamelunachippan (2017) "Performance Analysis of On-demand Routing Protocol for MANET Using EA3ACK Algorithm", International Journal of Mobile Network Design and Innovation (Inderscience), Vol. 7, No. 2, pp. 88-100.