ISSN 1815-2066. Nauka innov., 2019, 15(2): 69—79 https://doi.org/10.15407/scin15.02.069

**Mohammad Javad Akbarirad [1] and Reza Ghaemi [2]**

[1] Department of Computer Engineering, Science and Research Branch of Khorasan Razavi,
Islamic Azad University, Neyshabur, Iran,
info.akbarirad@gmail.com
[2] Department of Computer Engineering, Quchan Branch,
Islamic Azad University, Quchan, Iran,
r.ghaemi@iauq.ac.ir

# PROVIDING AN EFFICIENT MODEL FOR WIRELESS SENSOR NETWORKS USING THE SCENARIO OF THE VARIABLE SINK COUNTS BASED ON THE PARTICLE SWARM ALGORITHM

***Introduction.*** *A wireless sensor network is a set of independent sensor nodes, which are dispersed in a distributed manner to monitor and collect data in a geographic environment. One of these problems is the manner of node division in a set of multi-sink sensors.*

***Problem Statement.*** *In fact, the main issue in this area is related to the division of sensor nodes between sinks so that reduced energy consumption and increased network life survival will be resulted. In this study, a solution has been provided to partition a multi-sink sensor network. Due to the nature of the problem of partitioning a multi-sink sensor network, the search space is very extensive and, on the other hand, proving that this issue is classified as NP-hard problems has made the presentation of a definitive solution very difficult.*

***Purpose.*** *To develop a solution for distribution of sensor network with a few sinks.*

***Materials and Methods.*** *Thus, given the broad search space of the problem ahead, particle swarm algorithm has been selected. In order to evaluate the proposed approach, MATLAB programming language has been applied.*

***Results.*** *The proposed approach has been developed using the criteria of hop counts to the sink and also the number of cluster heads plus the power of particle search in particle swarm algorithm.*

***Conclusions.*** *Study of these results in the form of two criteria of hop counts and the number of cluster heads using the scenario of the variable sink counts demonstrate that in the desired scenario, the proposed approach has been able to improve hop counts relative to the base method by 17% and the number of cluster heads by 59%.*

*Ключові слова: wireless sensor networks, multi-sink, and particle swarm algorithm.*

The ability to integrate knowledge in the field of microelectromechanical systems, digital electronics and wireless communications has enabled us to produce small and inexpensive devices, including different sensors with low power consumption and the capability of wireless communication. These devices which are referred to as sensor nodes, or briefly, nodes are able to collect different pieces of environmental information (based on the type of nodes) and transmit them through wireless communications. A sensor node, alone, is capable of collecting and processing a small amount of information.

But when a large number of sensor nodes work in sync with each other, they will be able to measure features of a specific physical environment with great details. This ability paves the way for the creation and development of networks called wireless sensor networks.

A wireless sensor network is a collection of sensor nodes that provide the ability for monitoring the environment and establishing wireless communications in order to collect and send information. Environmental information collected by the sensor nodes is sent to a sink through sending information in the form of multi-hops. The sink can process the received information locally or be connected through a gateway to another network (such as the Internet) and send information for a specific destination [1].

At the time of developing a wireless sensor network, the location of the sensor nodes is not already known. As a result, it is possible to leave sensor nodes in hazardous and inaccessible locations. Further, wireless sensor networks offer low cost solutions to many real-world challenges. These features have rapidly increased the popularity of these networks and made them gain various applications in military and civilian fields. But sensor nodes have different constraints in power supply, processing power, data storage and communications due to low cost, small size and single-functionality. Additionally, in most cases, an environment monitored by a wireless sensor network lacks an infrastructure for energy supply and communications. Hence, these networks are only dependent on limited resources of sensor nodes to collect, process, store and send information [2].

Resource constraints have made wireless sensor networks faced with many unique challenges, including the design of sensor nodes, the design of appropriate protocols, energy consumption, data transmission and information security. Numerous applications and special challenges of wireless sensor networks have provided many research areas for researchers in a variety of fields [3].

One of the challenging issues in wireless sensor networks occurs when there are several sinks to collect data in the network. In this state, the main question is for which sink each node should send information. To solve this challenge, a problem arose called how to partition a multi-sink sensor network, which attempts to solve how to divide network nodes among sinks. This problem attempts to cluster network nodes and then divide the clusters to a sink between the sinks in order to improve network parameters such as energy consumption and network lifetime while maintaining the connection between the nodes.

Table 1 presents a number of studies conducted in the field of the research subject.

Suppose that topology is a logical structure for network communications. In this paper, cluster-based topologies are considered in the WSN design. A cluster is a set of sensors, each playing the particular role of follower, cluster head or bridge sensors. A follower sensor collects data and sends it to the cluster head. The cluster head is responsible for directing cluster member data and, along with a bridge, is connected to at least two other cluster heads. Sinks manage the sensor network. They are different from other sensors and are usually stronger. Therefore, a cluster-based WSN relies on a specific topology in which the follower sensors send messages to the relevant cluster head and each message is sent to a sink. It should be noted that bridges are optional in cluster-based topologies. However, in general, inter-cluster relationships with bridges have less energy than a direct cluster head. Overall, the clusters are applied when the data is highly correlated because in this state, cluster heads can collect similar messages and perform data compression. Hence, in cluster-based topologies, the total number of messages in the network has decreased [4].

In this study, the issue of building cluster-based topologies for the WSN is considered with several sinks. To this end, an optimization algorithm based on the particle swarm optimization algorithm has been taken into account. Optimization relies on different levels of decision-making, including the selection of sensors as cluster heads and load balancing among cluster heads. The topology associated with each sink has been modeled as a set of independent nodes with specific internode connections (IDSC). Thus, the solution is a partition of a graph as a large number of

IDSCs so that in this way, network lifetime improves since the average number of messages is reduced using a cluster structure. By controlling hop counts, the average energy for sending messages is also improved.

Here, we provide the proposed approach in this paper. Flowchart of the proposed approach which is based on the particle swarm algorithm has been presented in Fig. 1.

As it is clear in this flowchart, the number of repetitions ($N$) and the number of particles in the particle swarm algorithm, i.e. $M$, are initially determined. Then, we begin to initialize the particles. Hence, according to the method provided in the particle swarm algorithm, the initial velocity of the particle is considered to be zero. Afterwards, for each particle, we randomly select a number of network nodes as cluster heads. Next, we randomly partition the selected cluster heads among network sinks. Indeed, each sink's share from the clusters is randomly determined. Given that the members of a follower cluster are cluster heads, we put the members of the cluster head in the partition to which the cluster head belongs. Now that particle $i$ partition was determined, it comes to determining the fit. Fit of particle $i$ is determined by a combination of the number of cluster heads and the number of each node hops to the sink. In the next step, considering that particle $i$ has gained its first experience, we consider

*Table 1*

**Comparison of Several Prominent Works**

| Reference | Method evaluation | Method | Year of publication |
|---|---|---|---|
| 4 | Use of multiple sinks to prevent traffic and use of fuzzy neural networks for more accurate location | Providing a hybrid method for intelligent particle swarm optimization to identify the optimal data storage location for the WSN | 2015 |
| 5 | Fuzzy inference system in the residual energy, node degree and distance to the base station | Providing a method for regulator load distribution in unequal clustering in wireless sensor networks using fuzzy logic | 2016 |
| 6 | The proposed algorithm acts faster than traditional DE and convergent GA. | Presenting a differential evolution method based on clustering algorithm in wireless sensor networks | 2016 |
| 7 | This evolutionary method is used based on calculations and data congestion to increase the lifetime of mobile wireless sensor networks. | Providing an efficient energy algorithm for congestion of network sensors | 2014 |
| 8 | It has better results in energy consumption, total network lifetime and the number of data packets received by the central station compared to LDC, PSO-C, LEACH-C, E-LEACH and LEACH protocols. | Providing an algorithm for choosing an effective energy cluster head based on PSO using efficient particle representation | 2015 |
| 9 | This method is raised based on discrete particle swarm optimization (DPSO) with local search and DPSO has achieved better results than GASP | Providing a robust topology control method for the problem of locating wells in WSNs | 2016 |
| 10 | To determine the dependent cluster and non-dependent cluster head, the status data of the location, the residual energy of the nodes and their neighbors must be taken into account. This algorithm balances energy consumption and can thus be effective in network lifetime | Providing a clustering algorithm by specifying a dual cluster (dependent and non-dependent cluster heads) using particle swarm optimization algorithm (PSO-DH) | 2015 |

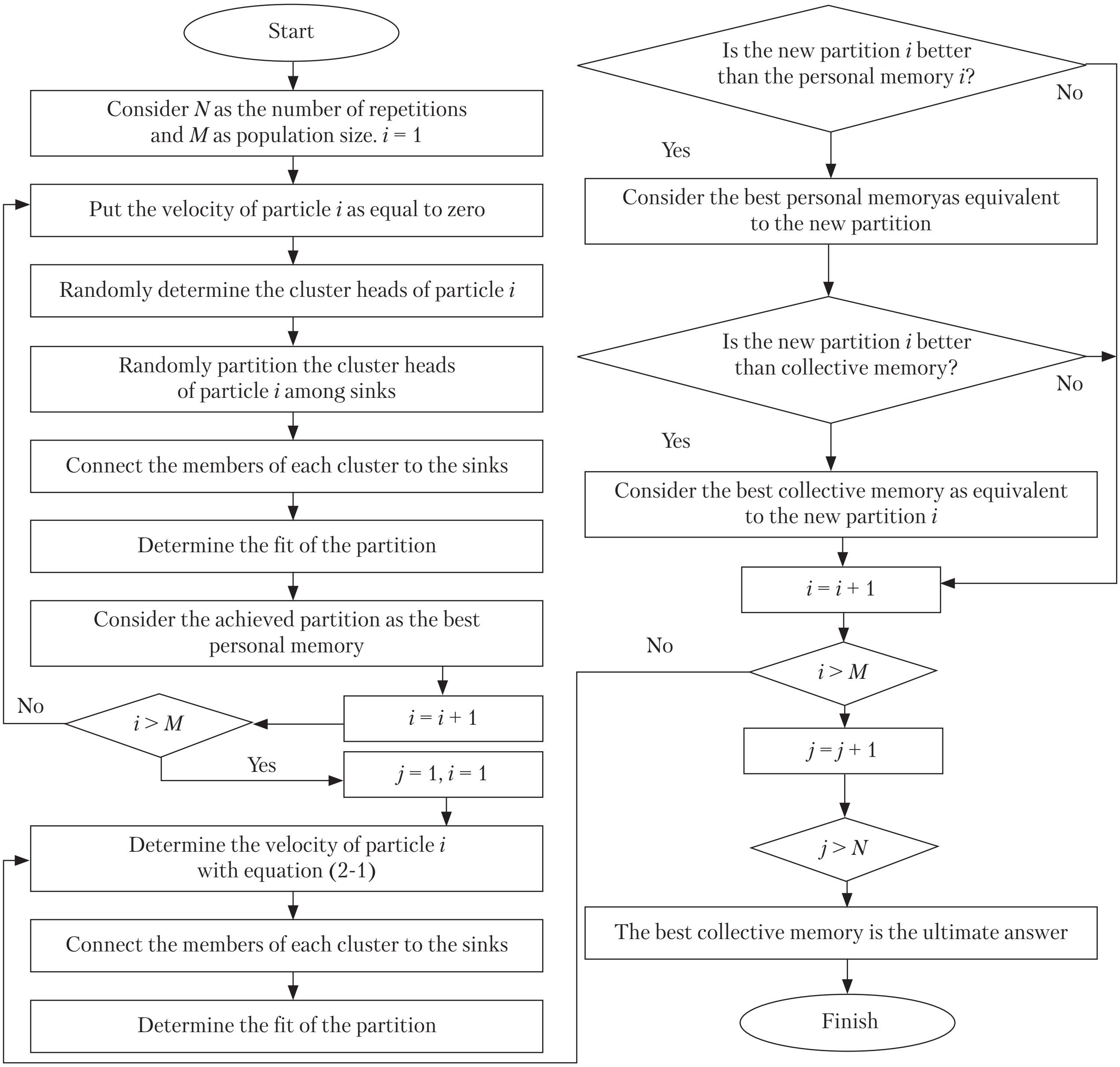


***Fig. 1.*** Flowchart of the proposed method

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | ... | $n$–1 | $n$ |
| ***S*** | ***M*** | ***S*** | ***M*** | **...** | ***S*** | *S* |

***Fig. 2.*** Proposed structure for particle clustering

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **3** | **3** | **1** | **2** | **1** | **2** | **2** | **3** | **1** | 3 |

***Fig. 3.*** Proposed structure for partitioning the particles

| Number | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Function | | | | | | | | | | |
| Partitioning | | | | | | | | | | |
| Velocity | | | | | | | | | | |

*Fig. 4.* Particle structure

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| M | S | M | M | S | S | S | S | M | S |
| 2 | 1 | 2 | 1 | 1 | 2 | 2 | 12 | 1 | 2 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

*Fig. 5.* Particle structure after initialization

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| M | S | M | M | S | S | S | S | M | S |
| 2 | 1 | 2 | 1 | 1 | 2 | 2 | 12 | 1 | 2 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

*Fig. 6.* Particle structure after determining the velocity

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| M | S | M | M | S | S | S | S | M | S |
| 2 | 1 | 1 | 1 | 2 | 1 | 2 | 12 | 2 | 2 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

*Fig. 7.* Particle structure after displacement

this experience as the best personal memory of particle *i*. This initialization process continues for all the particles. After initializing all the particles, it comes to displacement of particles in the problem space. For this purpose, the velocity of each particle is determined using equation (1). In the next step, each cluster head whose value corresponding to the velocity field is equal to 1 should be placed in the partition of another sink. After changing the sink of each cluster head, cluster members should change their category, following the cluster head. By changing the partition, fit of the new partition of particle *i* is calculated again. With regard to changing the fit of particle *i*, it should be examined whether or not the new partitioning can be the best personal memory for particle *i*. This partitioning may also be the best collective memory. Therefore, this issue has been checked in the following. By changing all members of the population to the specified number of repetitions, we are finished with the proposed algorithm and given that the best collective memory is the best state experienced by the whole set of particles in all repetitions, the best collective memory is the ultimate answer.

Considering that the structure used in the particles to determine the cluster heads and also the partitioning and velocity was not mentioned in the flowchart, we will outline these issues in the following.

#### PARTICLE STRUCTURE

In m-IDSC problem-solving, the goal is to calculate optimal clustering so that n nodes are placed in appropriate clusters. The nodes are divided into two groups of $M$ and $S$ which represent cluster head and follower/bridge nodes. Hence, in the proposed method, a structure for particle clustering is used in which each node will be in the role of $M$ or $S$. This structure can be seen in Fig. 2.

In this structure, each of the nodes 1 to $n$ will be in the role of $M$ or $S$. Besides, it is necessary to determine for each particle the structure used for partitioning. The proposed structure in the presented method for partitioning is very similar to the structure of Fig. 2, with the difference that in return for using $M$ and $S$, the numbers between 1 and sink counts have been used for each node. In fact, if $k$ value is stored in the cell related to node $s$, this means that node $s$ in the desired partition is located in the category of sink $k$. Fig. 3 shows this for a network with 10 nodes and 3 sinks.

It should be noted that the structure considered for the velocity of particles is exactly the same as the structure in Fig. 3, with the difference that the values of nodes will only be equal to 0 or 1. With this structure, if the velocity of a node is 1, it should change its partition and enter the partition of another sink. But if this velocity is zero, there is no need to change the category.

#### OBJECTIVE FUNCTION

In m-IDSC problem-solving, there are two objectives: First, reducing the number of clusters and second, reducing hop counts to send packets to the sink. Given that both objectives have the same direction, the purpose used in the proposed method is in the form of equation (1):

$$f = hop \times |clusters| . \tag{1}$$

In this equation, hop indicates hop counts necessary to send the packets in each section to the

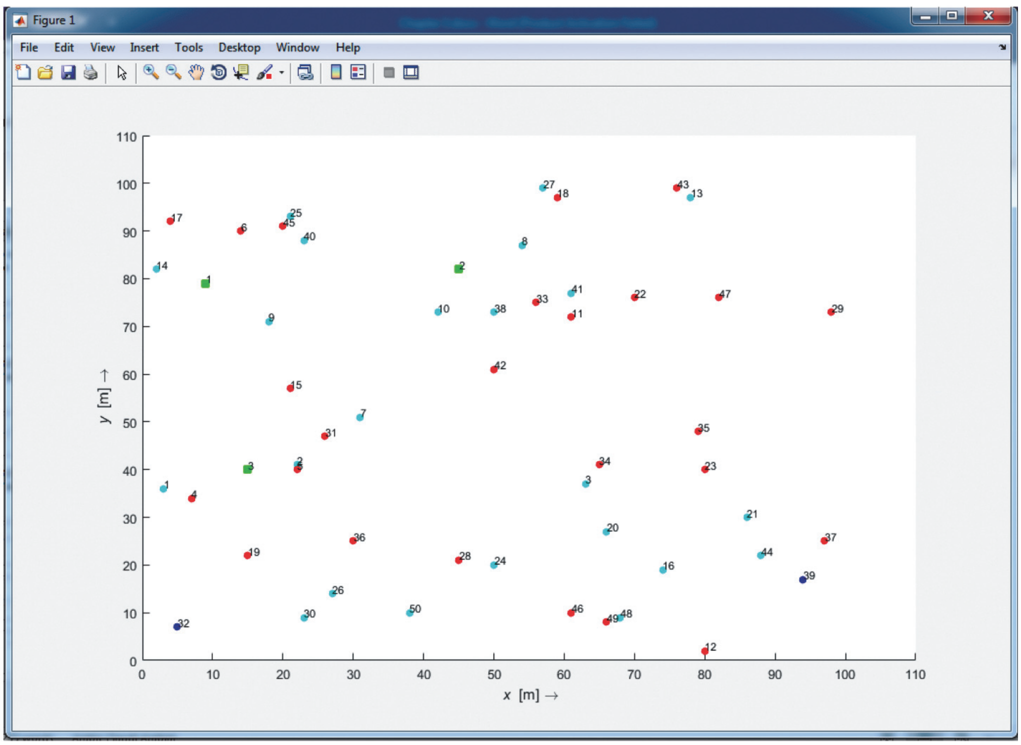


***Fig. 8.*** Dispersion of nodes in the simulation on the evaluation context

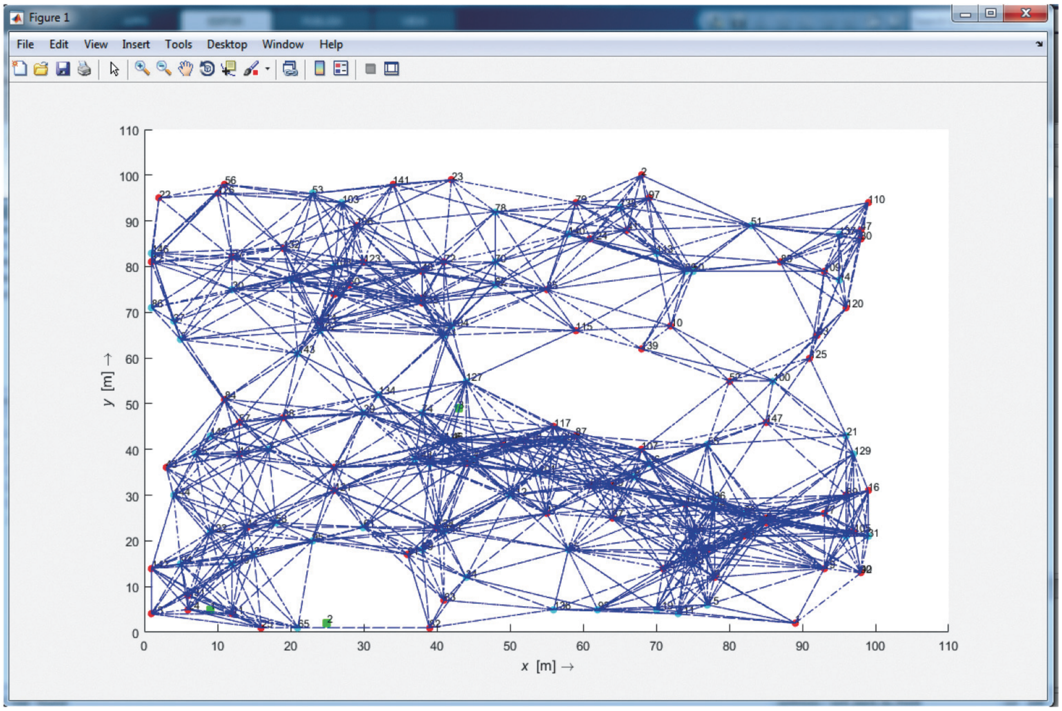


***Fig. 9.*** Communication links between nodes

relevant sink and cluster represents the number of clusters intended for problem-solving and partitioning.

## NUMERICAL EXAMPLES

In this section, because of more clarity, attempt is made to describe the proposed approach by a numerical example. To this end, the performance of the proposed method on a particle is explained. We assume that in the structure of the desired network, there are 10 nodes and 2 sinks. The initial structure of a particle is in the form of Fig. 4 such that the first row is the number of nodes, the second row is the function of nodes, the third row is the manner of partitioning and the fourth row is the velocity of nodes.

In initialization, the cluster heads of this particle are randomly determined. Then, the partition between them is also specified randomly. With respect to the partitioning of cluster heads, the members of each cluster are also partitioned among sinks and the particle velocity is considered to be zero. This has been displayed in Fig. 5.

After partitioning, it comes to determining the fit of the particle. In this particle, the number of cluster heads is 4 and it is assumed that the distance between the nodes and the connected sink is 27. The current status is regarded as the best personal memory of this particle (Fig. 6). After initialization of all particles, it comes to determining the best collective memory and repetition of the particle swarm algorithm. Suppose that the best collective memory is a partition with 4 cluster heads and 23 hops. Also, assume that in the first repetition, the velocity of the desired particle is calculated as follows.

With the calculation of the velocity, it is clear that cluster heads 3 and 9 should change their sink. Consequently, Fig. 7 will be the result of the displacement of this particle. It should be noted that the nodes belonging to cluster heads 3 and 9 also change their partition following the cluster heads.

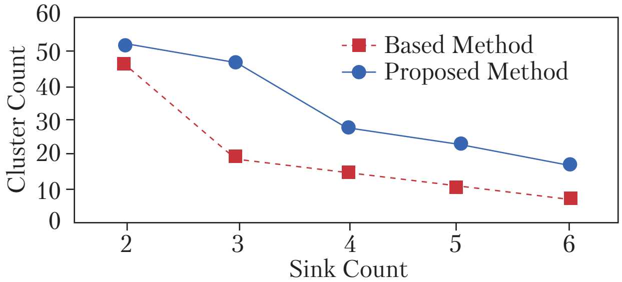


***Fig. 10.*** Cluster count for the variable sink count

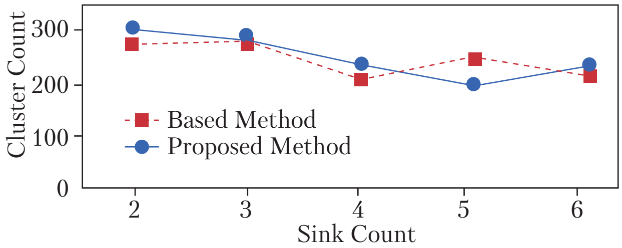


***Fig. 11.*** Hop count for the variable sink count

Now, it comes to determining the amount of the fit. The new structure has 4 cluster heads and we assume that it needs 20 hops to send the packets. Thus, the new position is considered as the best personal memory and since the new partition is better than the best collective memory, the best collective memory will be updated to the new partition. This trend continues for all particles and with the specified number of repetitions. Finally, the output of the algorithm is the best collective memory.

## INITIAL PARAMETERS

In the evaluations made, some parameters have been considered constant and some parameters have been regarded as variable according to the intended scenario. In Table 2, a list of parameters used in the evaluations is presented.

In Figs. 8 and 9, a view of the simulation environment is shown. Fig. 8 illustrates a view of the dispersion of nodes along with the number of each node. In this figure, blue nodes are normal nodes, red nodes are cluster heads and sinks are colored green.

In Fig. 9, in addition to a view of the nodes, neighborhood relations and communication links between them have also been displayed.

## ASSESSMENT SCENARIO: THE VARIABLE SINK COUNTS

In this section, results under the research scenario are provided. In this scenario, we consider the number of nodes as constant and the number of sinks as variable.

In evaluations of this section, the number of network nodes is equivalent to 100. To assess the two criteria, the cluster counts and hop counts necessary to send packets to the sink have been measured. It is clear that in both criteria, a method that achieves lower rates is superior. Fig. 10 shows the cluster count for the variable sink count in the range of 2 to 6.

In Fig. 10, better performance of the proposed method can be seen clearly. In this figure, both methods require fewer clusters with increased sink count. The reason is that with increased sink count, each node can deliver its packets to the nearest sink and there is less need for clusters to send packets. But despite reduced clusters in the base method, the proposed approach has managed to maintain its superiority throughout the entire range.

Fig. 11 exhibits the hop count necessary to send packets from each node to the related sink with the topology of the mentioned node.

Fig. 11 shows that the proposed approach, with the exception of the number of sink 5, always has performance equal or equivalent to the base method. The lower hop counts indicates the more appropriate approach of the proposed method because fewer hops mean less energy consumption to send data which is of great importance in

*Table 2*

**Parameters Used in the Experiment**

| Parameter | Value |
|---|---|
| Radio range | 20 m |
| Number of nodes | 50—150 |
| Environmental dimensions | 100 × 100 |
| Sink counts | 2—6 |
| Number of particles | 20 |
| Number of repetitions | 10 |

wireless sensor networks which are severely under pressure in terms of energy.

## CONCLUSION

In this research, attempt was made to provide a new approach for solving the IDSC problem or partitioning a wireless sensor network. In the proposed approach, an algorithm based on swarm intelligence called particle swarm algorithm has been used in order to partition the network since in the IDSC problem, the search space is extremely broad and the use of simple search algorithms has low performance. In the proposed method, each particle has a structure that represents a partition of the network. Particles are evaluated using a function that combines the two criteria of cluster counts and hop counts to send the packets to the destination. Better particles help to discover and approach the final answer by participating in the ongoing search process.

For assessment, the proposed approach along with the method presented in [4] have been implemented with MATLAB programming language and the results have been presented in the form of the scenario of the variable sink counts. To make evaluations, the criteria of cluster counts and hop counts were considered in which the purpose was to reduce these values. The results obtained from evaluations demonstrate the effectiveness of the proposed approach so that in the desired scenario which works on the variable sink count, the proposed approach has managed to make improvements even by 59 % and regarding the number of sent packets, the maximum improvement is 17 %.

## SUGGESTIONS

Given the fact that the IDSC problem is an NP-hard problem, it is very difficult to provide a definitive solution for it. Therefore, researchers interested in this field are recommended to attempt to provide a solution for this problem using evolutionary approaches because these algorithms, despite being simple, have high performance and good execution time. Interested researchers can look for desirable solutions using new evolutionary algorithms presented in this field, including grey wolf, cuckoo and firefly algorithms. Furthermore, due to the graph-like structure of wireless sensor networks and their similarity to social networks, it is suggested to develop the solutions applied in these fields for IDSC problem solution.

## REFERENCES


1. Zafar, S. (2011). A Survey of Transport Layer Protocols for Wireless Sensor Networks. *International Journal of Computer Applications*, New York, USA, November 2011.

2. Wang, Q., Balasingham, I. *Wireless Sensor Networks — An Introduction, Wireless Sensor Networks: Application — Centric Design*, ISBN: 978-953-307-321-7, 2010.

3. Hu, F., Cao, X. *Wireless sensor networks: principles and practice*. Boca Raton, FL, CRC Press, 2010.

4. Mohanasundaram, R., & Periasamy, P. S. (2015). Hybrid Swarm Intelligence Optimization Approach for Optimal Data Storage Position Identification in Wireless Sensor Networks. *The Scientific World Journal*, 2015.

5. Taruna, S., & Bhartiya, N. (2016). *A Survey Paper on Computational Intelligence Approaches*. In Proceedings of the International Conference on Recent Cognizance in Wireless Communication & Image Processing (pp. 609—617). Springer India.

6. Azharuddin, M., & Jana, P. K. (2016). Particle swarm optimization for maximizing lifetime of wireless sensor networks. *Computers & Electrical Engineering*, 51, 26—42.

7. Kaur, S., & Singh, J. (2014). *Optimization of Wireless Sensor Network Using PSO Algorithm. optimization*, 4, 5.

8. RejinaParvin, J., & Vasanthanayaki, C. (2015). Particle Swarm Optimization-Based Clustering by Preventing Residual Nodes in Wireless Sensor Networks. *IEEE sensors journal*, 15 (8), 4264—4274.

9. Obaidy, M. A., & Ayesh, A. (2015). Energy efficient algorithm for swarmed sensors networks. *Sustainable Computing: Informatics and Systems*, 5, 54—63.

10. Tsai, C. W., Tsai, P. W., Pan, J. S., & Chao, H. C. (2015). Metaheuristics for the deployment problem of WSN: A review. *Microprocessors and Microsystems*, 39 (8), 1305—1317.


**Received 07.10.18**

*Мохаммад Джавад Акбарірад* [1], *Реза Гаемі* [2]
[1] Кафедра комп'ютерної інженерії,
Департамент наукових досліджень у Хорасан Разаві,
Ісламський університет Азад, Нейшабур, Іран,
info.akbarirad@gmail.com
[2] Кафедра комп'ютерної інженерії, Філія в Кучані,
Ісламський університет Азад, Кучан, Іран,
r.ghaemi@iauq.ac.ir

## ЗАБЕЗПЕЧЕННЯ ЕФЕКТИВНОЇ МОДЕЛІ БЕЗДРОТОВИХ СЕНСОРНИХ МЕРЕЖ З ВИКОРИСТАННЯМ СЦЕНАРІЮ ЗМІННОЇ КІЛЬКОСТІ СТОКІВ НА ОСНОВІ АЛГОРИТМУ РОЮ ЧАСТИНОК

**Вступ.** Бездротова сенсорна мережа — це набір незалежних сенсорних вузлів, які розподілені певним чином для моніторингу та збору даних в географічному середовищі. Одним з їхніх функціональних завдань є спосіб розподілу вузлів у наборі датчиків з декількома стоками.

**Проблематика.** Основна проблема в цій галузі пов'язана з розділенням вузлів датчиків між стоками, що дозволить знизити споживання енергії та збільшити термін служби мережі. У зв'язку з природою проблеми розбиття сенсорної мережі з декількома стоками, пошуковий простір є надто великим і, з іншого боку, доведення того, що ця задача є NP-складною проблемою, зробило представлення остаточного рішення дуже складним.

**Мета.** Розробка рішення розподілу сенсорної мережі з декількома стоками.

**Матеріали й методи.** З огляду на широкий простір пошуку, в роботі використано алгоритм рою частинок. Для оцінки запропонованого підходу застосовано мову програмування *MATLAB*.

**Результати.** Запропонований підхід було розроблено з використанням критеріїв підрахунку кількості транзитних ділянок до стоку, а також кількості головок кластера сумарно з потужністю пошуку частинок в алгоритмі рою частинок.

**Висновки.** Вивчення отриманих результатів у вигляді двох критеріїв підрахунку кількості транзитних ділянок та кількості головок кластера з використанням сценарію змінної кількості стоків свідчить, що запропонований підхід дозволив поліпшити кількість транзитних ділянок відносно базового методу на 17 %, а кількість головок кластера — на 59 %.

*Ключові слова*: бездротові сенсорні мережі, мульти-сток, алгоритм рою частинок.

*Мохаммад Джавад Акбарирад* [1], *Реза Гаеми* [2]
[1] Кафедра компьютерной инженерии,
Департамент научных исследований в Хорасан Разави,
Исламский университет Азад, Нейшабур, Иран,
info.akbarirad@gmail.com
[2] Кафедра компьютерной инженерии, Филиал в Кучан,
Исламский университет Азад, Кучан, Иран,
r.ghaemi@iauq.ac.ir

## ОБЕСПЕЧЕНИЕ ЭФФЕКТИВНОЙ МОДЕЛИ БЕСПРОВОДНЫХ СЕНСОРНЫХ СЕТЕЙ С ИСПОЛЬЗОВАНИЕМ СЦЕНАРИЯ ПЕРЕМЕННОГО КОЛИЧЕСТВА СТОКОВ НА ОСНОВЕ АЛГОРИТМА РОЯ ЧАСТИЦ

**Введение.** Беспроводная сенсорная сеть — это набор независимых сенсорных узлов, которые распределены определенным образом для мониторинга и сбора данных в географической среде. Одной из их функциональных задач является способ распределения узлов в наборе датчиков с несколькими стоками.

**Проблематика.** Основная проблема в этой области связана с разделением узлов датчиков между стоками, что позволит снизить потребление энергии и увеличить срок службы сети. В связи с природой проблемы разбиения сенсорной сети с несколькими стоками, область поиска является слишком большой и, с другой стороны, доказательства того, что эта задача является NP-сложной проблемой, сделало представление окончательного решения очень сложным.

**Цель.** Разработка решения распределения сенсорной сети с несколькими стоками.

**Материалы и методы.** Учитывая обширную область поиска, в работе использован алгоритм роя частиц. Для оценки предложенного подхода применены язык программирования *MATLAB*.

**Результаты.** Предложенный подход был разработан с использованием критериев подсчета количества транзитных участков к стоку, а также количества головок кластера суммарно с мощностью поиска частиц в алгоритме роя частиц.

**Выводы.** Изучение полученных результатов в виде двух критериев подсчета количества транзитных участков и количества головок кластера с использованием сценария переменного количества стоков свидетельствует, что предложенный подход позволил улучшить количество транзитных участков относительно базового метода на 17 %, а количество головок кластера — на 59 %.

*Ключевые слова*: беспроводные сенсорные сети, мульти-сток, алгоритм роя частиц.