Maejo Int. J. Sci. Technol. **2016**, *10*(03), 294-303

Maejo International Journal of Science and Technology

ISSN 1905-7873
Available online at www.mijst.mju.ac.th

Full Paper

Internal-genetic-algorithm-based, energy-efficient clusters

Selim Bayrakli[1], Senol Zafer Erdogan[2,*] and Turgut Kilinc[3]

[1] Faculty of Engineering and Natural Sciences, Maltepe University, Maltepe Istanbul, Turkey
[2] Faculty of Engineering and Architecture, Konya Food and Agriculture University, Konya, Turkey
[3] Faculty of Engineering, Trakya University, Edirne, Turkey

* Corresponding author, e-mail: senol.erdogan@gidatarim.edu.tr

Received: 6 March 2015 / Accepted: 18 October 2016 / Published: 3 November 2016

Abstract: We propose an internal-genetic-algorithm-based, energy-efficient clustering (INGABEEC) method to enhance the lifetime of wireless sensor networks (WSNs). The proposed method generates energy-efficient clusters by considering several aspects of the WSNs. The INGABEEC generates energy-efficient clusters by genetic algorithm, where cluster heads are changed dynamically. Furthermore, the INGABEEC is improved by using an intra-cluster communication between non-cluster head nodes.

Keywords: wireless sensor networks, energy-efficient clustering, genetic algorithm

INTRODUCTION

Wireless sensor networks (WSNs) consist of sensors that can sense biophysical events such as light, sound, humidity and pressure, and a network infrastructure enables these sensed values to be transmitted without any wire. The receiver-transmitter circuits providing the wireless communication are positioned on these sensor nodes.

Although WSNs have a wide range of application areas such as target tracking, habitat monitoring, surveillance and security [1, 2], the sensor nodes and communication capabilities of the network are required to be used at an optimum level due to the limited availability of energy resources. Using clusters for WSNs reduces the energy consumption by lessening communication distance among sensor nodes. Therefore, for environmental monitoring it may be energetically advantageous to use cluster-based methods [3]. The issue of efficient use of energy in WSNs is among the topics to be researched extensively. Other issues related to reduced energy consumption that need further attention are: compressing data to be sent to minimum size, improvement of data processing process, and routing techniques in the step of transmitting and delivery of data to a targeted destination at minimum cost.

In this study the architecture and design criteria of WSNs as well as the requirements for routing are investigated, and a new method that will extend the lifespan of the network and reduce the amount of energy spent in the data transmission phase is presented.

RELATED WORK

The low-energy adaptive clustering hierarchy (LEACH) is one of the oldest and best-known clustering algorithms [4]. Sensors are divided into clusters by organising themselves using the LEACH algorithm. In doing so they evaluate the distance to the cluster head, where the leader of the cluster sends the cluster's whole data to base station. They calculate this distance by checking the strength of the broadcast signal received from the cluster head. Cluster heads are also used as routers for the base station. All data-processing operations are performed in the clusters. In LEACH algorithm, the communication process is divided into rounds and each round consists of two phases. The first phase is the set-up phase and the second phase is the steady-state phase. LEACH uses adaptive clusters, which is different from conventional clustering methods, and changes the cluster heads by rotation. In this way energy requirements of the network are distributed between the sensors. When cluster heads are changed, new clusters are created. Besides, local calculating is performed in the LEACH algorithm. Thus, the cluster heads do not send the data received from the members' nodes to the base station directly. Instead, they compress all the data received from all the member nodes and send them as a single package. This reduces the energy consumption significantly and processing the data requires much less energy compared to the direct communication.

A two-level hierarchy for LEACH (TL-LEACH) protocol is an enhanced form of LEACH [5]. Two layers are formed between the cluster heads (primary and secondary). In each cluster the primary cluster heads communicate with the secondary ones and the respective secondary heads communicate with the member nodes. The selection process of primary and secondary cluster heads is the same as in the LEACH. The probability of selecting the primary cluster heads is less than that of selecting the secondary cluster heads. Communication takes place in two steps from the source node to the carrier: collecting the data from member nodes of the secondary cluster heads, and the primary cluster heads collecting the data from relevant secondary cluster heads.

The candidate cluster heads compete with each other for a certain round to be the cluster head by meeting some specifications in an energy-efficient clustering scheme [6]. In this competition the candidates notify the others of their remaining energy. The node with the highest remaining energy becomes the cluster head. The cluster formation is different from the LEACH in that the energy-efficient clustering scheme dynamically changes the cluster size based on the distance between the cluster and base station. In this way the algorithm tries to prevent the energy consumption problem of the clusters far away from the base station. This improves the energy distribution and increases the lifespan of the network. In the cluster head selection process, well-distributed cluster heads can be selected by a small control support process. In the cluster head formation process, a new weighting function for load balancing in the clusters has been developed [6].

In the hierarchical clustering method the base station is included in the operation [7]. In this routing scheme the principle of 'energy spent for long distance messaging are much more than energy spent for short distance messaging' is taken into account. The LEACH protocol has been expanded by using a set of cooperative called head-set rather than using a single cluster head. In addition, the base station is informed about energy status of the nodes when result packages are sent. Thus, the base station generates energy-efficient clusters using energy status of the nodes. The members of the head-set are in charge of sending the messages to a remote base station. Only one of the members of the head-set is active at a time while the others are in the sleeping mode. The transmitting task to the base station is distributed equally among members of the head-set. In the radio model used in this study, in addition to LEACH, the energy consumed for remote and close distances is considered by different equations. In the selection phase of the study the base station is aware of the number of active nodes and makes a decision regarding the suitable number of clusters by using the energy levels of nodes. In contradistinction to LEACH, the base station selects a group of head-sets and publishes this information in the network. Then the selection of members (grouped into the clusters) is the same as in LEACH. In the last step of the selection stage each cluster head forms a collaborative cluster (head-set) by checking the signal level of notification message received from member nodes.

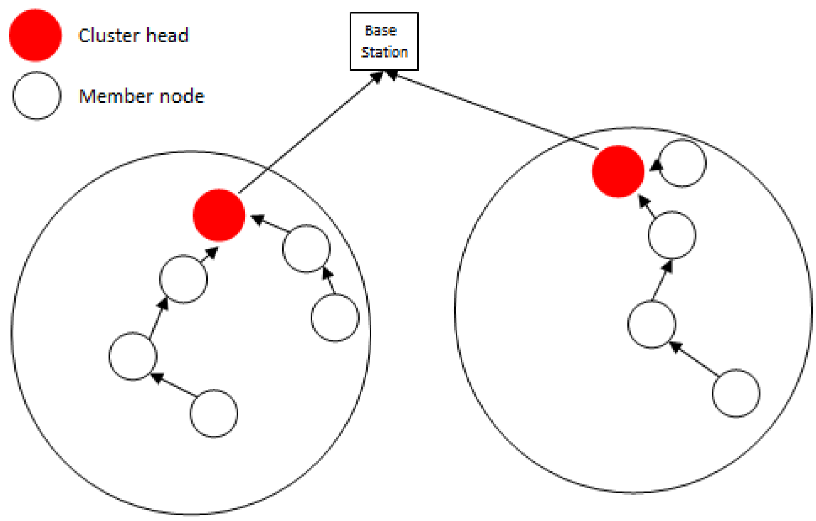
In the LEACH-central method it is aimed to create clusters that can generate better results using a central control algorithm by completely changing the set-up phase in LEACH [8]. The data transmission phase is the same as in LEACH. In the set-up phase of LEACH-central, each node transmits its location information and energy level to the base station, which calculates the average energy levels of the nodes in order to balance the load between nodes. The nodes with an energy level lower than average cannot be the cluster head in that round. The base station uses a simulated annealing algorithm to find the remaining possible cluster heads and optimal clusters. This algorithm aims to use a minimum amount of energy spent while sending the data of member nodes to the cluster heads. It can achieve this by minimising the distances between all member nodes and cluster heads.

The main differences of the present study from those summarised above are twofold: (1) a better genetic algorithm is used to increase efficiency, and (2) sensor nodes in every cluster are organised as a chain to perform a multi-hop intra-cluster communication to reduce energy consumption and maximise the lifetime of the network.

INTERNAL-GENETIC-ALGORITHM-BASED, ENERGY-EFFICIENT CLUSTERS (INGABEEC)

The INGABEEC algorithm is a new method that improves on the genetic-algorithm-based, energy-efficient clusters (GABEEC) algorithm [9], which was developed to increase the network lifetime of WSNs. The integration of a model similar to the 'power-efficient gathering in sensor information systems' algorithm [10] was suggested in order to minimise the packet communication process in GABEEC. The two stages used for GABEEC are modified in the INGABEEC method to three stages as follows: set-up, creation of intra-chain and data communication.

The first stage is the set-up and it is performed only once. In this stage some nodes become the cluster heads of some clusters and the remaining ones become the members of these clusters. The selection of cluster heads was pre-defined and the total cluster head number shows the total number of clusters that can be created in a network at the same time. The nodes which are not cluster heads are assigned to the cluster with the nearest cluster head based on their distances to these cluster heads. After the determination of the clusters, the nodes in each cluster are programmed to perform communication with the nearest node in the same cluster. The approach of INGABEEC addresses and applies the principle of the 'power-efficient gathering in sensor information systems' method [10], which has a key idea of creating a chain by cluster nodes with the nearest-neighbour nodes and transmitting the data to the base station by transferring them from one node to another. In the chain creation of each cluster, which is the second stage, the endless message spread is prevented by sorting the nodes from farther to nearer. This process is shown in Scheme 1.



Scheme 1. Topology of INGABEEC protocol

The third stage is the stage where the communication takes place. All members of sensor nodes in the cluster begin to communicate with the cluster heads. Each node transmits the data to the target node in the chain in order to finally deliver the data to the cluster head. In this way all the packets are transmitted to the cluster head through the target nodes.

The cluster head collects all the data received from the closest node in a single packet and transmits this packet to the base station. At the end of transmitting of the data collected from cluster heads to the base station, one single round is completed. Upon completion of each round, the base station checks the energy levels of the member nodes and cluster heads. If the energy of a cluster head drops to a level lower than the average energy level of the cluster, a new cluster head should be selected. To accomplish this, the member node with the highest energy is assigned as cluster head. The current cluster head acts as a member node [9]. After an exchange between nodes per cluster, the chain between nodes is re-established.

The genetic-algorithm-based application of the proposed method is described in the following sub-headings.

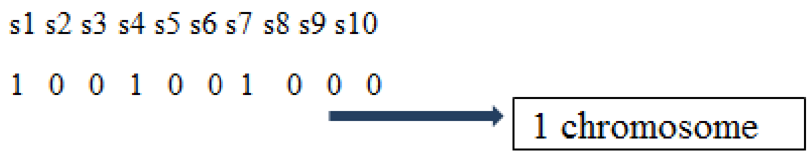
Representing the Network as a Chromosome

The binary system is used in genetic algorithm in order to express the nodes in the network. A network is represented by the bit group consisting of 0's and 1's. Each bit within this representation refers to the normal node or cluster head. The cluster head node takes the value of 1, whereas the normal node is 0. This 10-bit network is named as chromosome in the genetic algorithm literature. Each bit in the representation refers to a gene. The structure of a chromosome composed of 10 bit-gene is shown in Scheme 2.

The sensor nodes (s1, s4 and s7) given in Scheme 2 are the cluster heads, whereas s2, s3, s5, s6, s8, s9 and s10 are normal nodes. Each chromosome of the genetic algorithm represents a network topology. Chromosomes are composed of random genes resulting in different network topologies. Genetic algorithm is used to select the cluster heads; each node makes a distance calculation to find the nearest cluster head and connects with the nearest one. After this step, the

Maejo Int. J. Sci. Technol. **2016**, *10*(03), 294-303

determination of the fitness function and calculation of this fitness function for the chromosome representing this network must be performed.



Scheme 2. Bit representation of a wireless sensor network

Fitness Function

The fitness function, which is a component of the genetic algorithms, is used in the evaluation of sensor networks. The data obtained by calculating the fitness value present the achievement level of the fitness function.

According to the literature, one of the most important factors in the fitness function is distance. Reducing the distance variable decreases the amount of energy consumed. The fitness function used by Jin et al.[11] is given in equation 1:

$$\text{Fitness} = w*(D-\text{Distance}_i)+(1-w)*(N-H_i) \quad (1)$$

In equation 1 D refers to the total distance of all sensor nodes to the base station/data collector. The Distance_i value is calculated by summing up the total distance from the sensor nodes to the cluster head in each cluster and the total distance of each cluster head to the base station. H_i refers to the number of cluster heads while N refers to total number of nodes. The value of w can be between 0 and 1. In a sensor network the other parameters are constant except Distance_i and H.

The value of fitness function is higher as the distance becomes shorter [11]. In the present study we attempt to maximise the value of the fitness function. The value of w can vary depending on the application; it is possible to have the most suitable w value by considering the distance and cost of being the cluster head. If the value of w is 1, only the distance of the network is taken into account, whereas if the value of w is 0, then only the number of cluster heads is taken into account.

In some cases the fitness function values are close to each other so it is very difficult to select the best value in terms of genetic algorithm. A proportional distribution must be performed in order to increase the rate of making a better selection [11]. For this, scaling must first be done, in which the value of chromosome with minimum fitness is subtracted from the fitness value of each chromosome in the population.

In the present study the fitness function in the GABEEC algorithm is considered as a reference to be used in the proposed method. The fitness function (F) is given in equation 2:

$$F = \sum_i (R_{FND}, R_{LND}, -C) \quad (2)$$

where R_{FND} is the dying time of the first node in rounds, R_{LND} is the dying time of the last node in rounds, and C is the cluster distance.

The equation used for calculating the cluster distance (C) is

$$C = \sum_{i=1}^{k} d_{ih} + d_{hs} \quad (3)$$

where d_{ih} refers to the distance from h cluster head to cluster head, and d_{hs} refers to the distance from h cluster head to s base station.

Maejo Int. J. Sci. Technol. **2016**, *10*(03), 294-303

The selection process is performed after the fitness value is calculated. Through this process the chromosomes are determined to create a new population. The roulette-wheel selection method is used in the selection process [12].

Selection

In the genetic algorithm good chromosomes are determined in the selection process. Poor chromosomes are discarded in this process. The fitness values of chromosomes are used to make the selection.

In the sensor networks the fitness values of each network topology are calculated and the roulette-wheel selection method is used after the values are obtained. According to this method, each chromosome is placed into a slot. The areas covered by slots are in accordance with the fitness values of the chromosomes. Thus, since the chromosomes with higher fitness values cover larger areas on the wheel, their probability to be transferred to the next generation is increased. The roulette-wheel method is used in both the GABEEC and the approach proposed in this study.

Crossover

A single-point crossover is applied for the approach proposed. In the crossover process a gene interchange is performed between two parents based on values specified with the crossover ratio between two chromosomes selected. In Scheme 3 the structure of the child created as a result of the crossover is shown. Last four bits of parent 1 and parent 2 are interchanged to produce child 1 and child 2. As a result of crossover, a normal node can be formed as a cluster head in the sensor network. The other nodes close to the cluster head should connect to the nearest cluster head. When a cluster head becomes a normal node, then it should act as a normal node and connect to the new closest cluster head. As indicated before, a node in a cluster is either a cluster head or a normal node.

Parent 1	**1110 | 0101**
Parent 2	**1011 | 1110**
Child 1	**1110 | 1110**
Child 2	**1011 | 0101**

Scheme 3. Crossover process on chromosomes in WSNs

Mutation

A crossover results in a new combination of chromosomes, and then some genes are selected to be inverted based on the mutation probability. For example, the sequences of the nodes that are mutated are shown in Scheme 4.

New pre-mutation chromosome	**1 1 1 1 0 0 1**
New post-mutation chromosome	**1 1 0 1 0 1 1**

Scheme 4. The change of two bits as a result of mutation

Maejo Int. J. Sci. Technol. **2016**, *10*(03), 294-303

Steps of INGABEEC Algorithm

The steps of the INGABEEC algorithm are described as follows:

1. The sensor network nodes per cluster are selected at random.
2. The nodes create the chromosomes.
3. Selected chromosomes are encoded by binary coding and this coding represents a network.
4. In the chromosomes sequence some nodes are assigned as cluster heads (1) and others as normal nodes (0), based on the ratio specified before.
5. Normal sensor nodes connect to the nearest cluster head.
6. The fitness value of each chromosome is calculated in accordance with the fitness function.
7. The roulette wheel is applied to the selection process of the chromosomes after calculating their eligibility.
8. The chromosome with the highest fitness value is selected for the next generation.
9. Chromosomes perform the crossover and mutation on the sensor networks selected, and a new chromosome is created for the next population.
10. In the last phase a new generation is created with new network structures (chromosomes).
11. The fitness value of each chromosome is calculated by taking the network structure into a simulation. All of these steps are executed based on generation numbers specified before [8].

IMPLEMENTATION AND RESULTS

In this study Microsoft Visual C#, which is an integrated development environment, was employed to develop a WSN simulator in order to simulate the approach proposed.

In the simulation a total of 100 sensors were placed in a 50x50-metre area at random and the base station was positioned 100 metres from the network. The radio model used in the LEACH algorithm was also used in this application. The aspects of the radio model are shown in Table 1.

Table 1. Radio aspects

Activity	Energy consumed
Transmitter electronics (E_{Tx}) Receiver electronics (E_{Rx}) $E_{Tx} = E_{Rx} = E_{elec}$	50 nJ/bit
Transmit amplifier (ϵ_{amp})	100 pJ/bit/m^2

The energy consumption function is shown in equation 4. Dissipated energy (E_{elec} = 50 nJ/bit) was consumed to run either the radio transmitter or radio receiver of the wireless sensors, and the transmission amplifier also consumed 100 pJ/bit/m^2 of energy. With this radio model, the amount of energy consumed by the radio in order to transmit an n-bit message to a distance d is given as

$$E_{Tx}(n,d) = E_{elec} * n + \epsilon_{amp} * n * d^2 \tag{4}$$

The amount of energy consumed to receive the message is given as

$$E_{Rx}(n) = E_{elec} * n \quad (5)$$

The parameters and their values used in the genetic algorithm during simulation are given below:

- Number of population: 20
- Number of generations: 120
- Crossover rate: 6 %
- Mutation rate: 1 %
- Number of nodes: 100
- Message length: 2000 bits
- Network size: 50 x 50 metres
- Cluster head rate: 10 %

The simulation was run 3 times with 3 different starting energies. The results are shown in Table 2. According to the simulation in which the starting energy of the nodes was 0.25 joule, the lifetime of the first node in the INGABEEC algorithm was 1.27 times longer compared to GABEEC and 2.06 times longer compared to LEACH. For the second simulation in which the starting energy of the nodes was 0.5 joule, the two lifetime values were 1.09 and 1.43 times longer respectively, and for the third simulation with 1-joule starting energy, the two lifetime values were 1.19 and 1.82 times longer respectively. There was also a significant improvement in the dying round number of the last node. According to the simulation in which the starting energy of the nodes was 0.25 joule, the lifetime of the last node in the INGABEEC algorithm was 1.34 times longer compared to GABEEC and 1.53 times longer compared to LEACH. For the second simulation in which the starting energy of the nodes was 0.5 jould, the two lifetime values were 1.37 and 1.62 times longer respectively, and for the third simulation with 1-joule startng energy, the two lifetime values were 1.5 and 1.65 times longer respectively.

Table 2. Simulation results obtained by using different initial energies and methods

Initial energy (J/node)	Method	Round no. where first node dies	Round no. where last node dies
	LEACH	394	665
0.25	GABEEC	640	760
	INGABEEC	815	1024
	LEACH	932	1312
0.5	GABEEC	1218	1553
	INGABEEC	1335	2132
	LEACH	1820	2608
1	GABEEC	2776	2880
	INGABEEC	3316	4324

In the next simulation the network configuration was re-configured based on the parameter values given in Figure 1. There were 100 nodes with a starting energy of 0.25 joule in a 100x100-metre area and the base station was positioned 200 metres from the network. In this simulation GABEEC and INGABEEC were compared and the program was run 10 times to obtain the values. Simulation results are shown in Table 3. According to these results, it is clear that the INGABEEC method shows slightly poorer performance compared to the GABEEC during the time until 40% of

Maejo Int. J. Sci. Technol. **2016**, *10*(03), 294-303

the nodes are dead. Considering the round numbers, however, it can be observed that the INGABEEC gives better results in the case of 60-100% dead nodes.

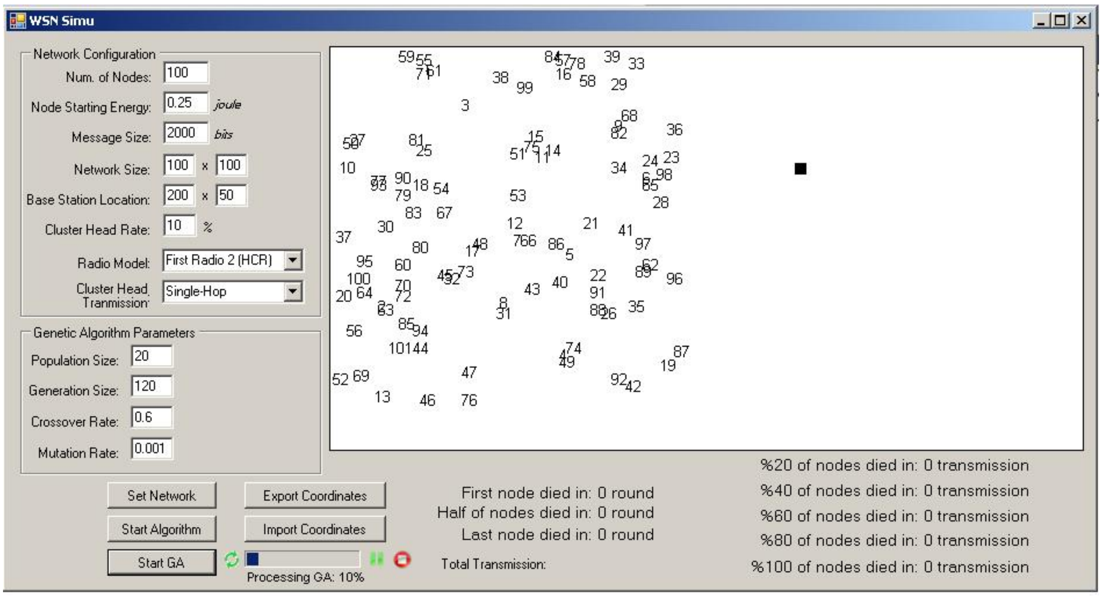


Figure 1. A screenshot from the simulator representing a specific simulation setting.

Table 3. Comparison of GABEEC and INGABEEC algorithms

	GABEEC						INGABEEC					
Simulation	First Dead	20% Dead	40% Dead	60% Dead	80% Dead	100% Dead	First Dead	20% Dead	40% Dead	60% Dead	80% Dead	100% Dead
1	770	782	788	818	962	1032	609	622	622	829	1051	1217
2	685	699	814	835	925	1049	774	950	1067	1166	1247	1281
3	840	872	882	896	908	1045	707	736	869	914	1094	1228
4	838	851	861	893	959	1020	633	762	683	706	792	1220
5	827	843	852	906	1021	1032	718	731	708	999	1011	1188
6	839	854	879	890	901	1044	706	714	716	872	950	1127
7	796	808	837	848	977	990	718	643	939	968	948	1176
8	812	823	918	925	932	1052	680	689	817	937	952	1218
9	774	790	836	935	948	1004	727	739	947	960	1089	1200
10	845	859	870	903	961	1030	565	708	755	857	978	1292
Average	802.6	818.1	853.7	884.9	949.4	1029.8	683.7	729.4	812.3	920.8	1011.2	1214.7
Min	685	699	788	818	901	990	565	622	622	706	792	1127
Max	845	872	918	935	1021	1052	774	950	1067	1166	1247	1292
Average Deviation	37.08	38.68	28.3	30.74	26.6	15.08	49.56	54.2	115.5	85.2	87.24	33.56
Standard Deviation	49.67	51.61	37.09	38.63	35.32	20.13	62.96	88.93	140.89	120.79	120.41	47.97

Note: Figures in Table indicate number of rounds.

CONCLUSIONS

In contrast to most studies, which focus on increasing the network lifetime by reducing the energy consumption of wireless sensors working with batteries, in other words by creating efficient solutions, in this study a genetic-based routing method is suggested in order to ensure the conservation of energy consumed during the data transmission in the sensor nodes of WSNs.

In the proposed INGABEEC method, which is based on an optimisation of GABEEC algorithm, a chain between nodes is established rather than communication of intra-cluster nodes with cluster heads. By using a genetic algorithm, the proposed approach can find an applicable number of cluster-heads and their locations, and also the optimum intra-cluster chains. Simulation results have showed that this chain-based cluster approach can extend the lifetime of the sensor network.

ACKNOWLEDGEMENTS

The authors are grateful to the editor of this paper, who helped to clarify and improve the presentation.

REFERENCES

1. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "Wireless sensor networks: A survey", *Comput. Networks*, **2002,** *38*, 393-422.
2. D. Estrin, D. Culler, K. Pister and G. Sukhatme, "Connecting the physical world with pervasive networks", *IEEE Pervasive Comput.*, **2002**, *1*, 59-69.
3. S. Bandyopadhyay and E. J. Coyle, "An energy-efficient hierarchical clustering algorithm for wireless sensor networks", Proceedings of 22nd Annual Joint Conference of IEEE on Computer and Communications, **2003**, San Francisco, USA, pp.1713-1723.
4. W. R. Heinzelman, A. Chandrakasan and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks", Proceedings of 33rd Hawaii International Conference on System Sciences, **2000**, Hawaii, USA, pp.8020-8030.
5. V. Loscri, G. Morabito and S. Marano, "A two-levels hierarchy for low-energy adaptive clustering hierarchy", Proceedings of 62nd IEEE Vehicular Technology Conference, **2005**, Dallas, USA, pp.1809-1813.
6. M. Ye, C. Li, G. Chen and J. Wu, "EECS: An energy-efficient clustering scheme in wireless sensor networks", Proceedings of 24th IEEE International Performance, Computing, and Communications Conference, **2005**, Phoenix, USA, pp.535-540.
7. S. Hussain and A. W. Matin, "Base station assisted hierarchical cluster-based routing", Proceedings of International Conference on Wireless and Mobile Communications, **2006**, Bucharest, Romania, pp.171-177.
8. W. B. Heinzelman, A. P. Chandrakasan and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks", *IEEE Trans. Wireless Commun.*, **2002**, *1*, 660-670.
9. S. Bayrakli and S. Z. Erdogan, "Genetic algorithm based energy-efficient clusters in wireless sensor networks", *Procedia Comput. Sci.*, **2012**, *10*, 247-254.
10. S. Lindsey and C. S. Raghavendra. "PEGASIS: Power-efficient gathering in sensor information systems", Proceedings of IEEE Aerospace Conference, **2002**, Big Sky, USA, pp. 3-1125 – 3-1130.
11. S. Jin, M. Zhou and A. S. Wu, "Sensor network optimization using a genetic algorithm", Proceedings of 7th World Multiconference on Systemics, Cybernetics and Informatics, **2003**, Orlando, USA, pp.109-116.
12. D. E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley Professional, Boston, **1989**.