2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)
Banff Center, Banff, Canada, October 5-8, 2017

A novel clustering algorithm for wireless sensor network based on search economics

Chun-Wei Tsai*, Zhen-An Li†, Ming-Chao Chiang† and Chu-Sing Yang‡

*Department of Computer Science and Engineering, National Chung-Hsing University, Taichung 40227, Taiwan
†Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan
‡Institute of Computer and Communication Engineering, Department of Electrical Engineering, National Cheng Kung University, Tainan 70101, Taiwan
E-mail:cwtsai@nchu.edu.tw, o2336151@gmail.com, mcchiang@cse.nsysu.edu.tw, csyang@ee.ncku.edu.tw

***Abstract*—One of the most well-known clustering methods for wireless sensor network is, no doubt, the so-called low energy adaptive clustering hierarchy (LEACH) because it is simple and easy to implement. Although LEACH tries to provide a *fair* selection mechanism by randomly selecting a number of sensors as the cluster-heads, it does not take into account the distribution of sensors, the main reason that LEACH is not able to allot transmissions to sensors efficiently, so it will make sensors far from the base station consume more energy in some cases. An efficient clustering algorithm to overcome this problem is presented in this paper. The proposed algorithm leverages the strength of a novel metaheuristics, search economics, and LEACH-centralized (LEACH-C) for wireless sensor network (WSN). Simulation results show that the search economics based clustering algorithm is capable of not only prolonging the lifetime of a WSN but also providing a balance strategy for the energy consumption of sensors in a homogeneous WSN.**

***Index Terms*—Wireless sensor network, search economics, energy efficiency, and metaheuristics.**

I. Introduction

Lifetime is a critical problem of wireless sensor network (WSN), for it is not easy to replace or recharge the battery on each wireless sensor that has a limited life span. Thus, energy efficiency has been a critical research topic for all kinds of applications [1], [2]. Improving the transmission device [3], reducing the data packet size [4], and controlling the duty cycle [5] are all possible solutions for reducing the energy consumption.

To avoid collision, to enhance energy efficiency, and to increase reachability of a WSN are the major goals in the design of the transmission protocol of a WSN. The flat and hierarchical protocols are the two representative protocols for energy efficiency of a WSN [6]. As a flat protocol, the energy-aware protocol based on temporally-ordered algorithm (E-TORA) [7] and reliable and energy efficient protocol [8] are the well-known examples. As a hierarchical protocol, the low energy adaptive clustering hierarchy (LEACH) [9], hybrid energy-efficient distributed clustering (HEED) [10], distributed hierarchical agglomeration clustering (DHAC) [11], and extending lifetime of cluster head (ELCH) [12] are the representative protocols. Although all these protocols can be used for reducing the energy consumption, there is plenty of room for improvement.

Several recent studies [13] have shown that metaheuristic algorithms provide a way to improve the performance of a WSN, e.g., to balance the energy consumption. For example, Heinzelman et al. [14] employed simulated annealing (SA) for clustering to select the cluster heads (CHs). Hoang et al. [15] used harmony search algorithm (HSA) with a fitness function that contains both distance and residual energy to select CHs, called harmony search algorithm cluster-based protocol (HSACP). The genetic algorithm (GA), of course, has also been applied to this research domain. In [16], GA is used to select CHs, called genetic algorithm based energy efficient clustering hierarchy (GAECH). The fitness function of GAECH takes into account several factors of a WSN, including total energy consumption, distance, standard deviation in energy consumption between clusters, and CH energy consumption. In our previous works [17], we observed that most metaheuristic algorithms did not attempt to depict the solution space during the convergence process; it, however, can be expected that the topography information of the solution space will be very useful to improve the search result of metaheuristic algorithms.

In this paper, a high performance metaheuristic algorithm, called search-economics-based clustering algorithm (SECA), is presented. One of the basic ideas of SE-based algorithms [17] is to depict the solution space to "avoid searching the same regions too many times" and to "search the potential regions that have not been searched as often as possible." The SECA is proposed for reducing the energy consumption of a WSN to prolong its lifetime. The remaining parts of the paper are organized as follows. Section II begins with a description of the clustering problem of WSN, followed by a brief introduction to the LEACH and LEACH-Centralized methods. Section III gives the details of the SE for the clustering problem of a WSN, including solution encoding, and the details of the proposed algorithm. Parameter settings and experiment results are given in Section IV to show the performance of the proposed algorithm. Finally, the conclusion and some future directions are drawn in Section V.

II. Related work

A. The Clustering Problem of WSN

The objective of the clustering problem of a WSN is to select a set of sensors as the CHs to provide an efficient way to transfer data between sensors, CHs, and base station (BS). Some essential properties for the objective of this problem are:

978-1-5386-1644-4/17/$31.00 ©2017 IEEE

- All the sensors are homogeneous.
- All the sensors can reach the furthest sensor and BS.
- The sensors are uniformly deployed or distributed.

The transmission model used in this paper is similar to LEACH-C [18]. For each round, all the sensor nodes will first send their energy information to the BS. Then, the BS will select CHs using SE based on the energy information received. The BS will determine which sensors will be elected as the CHs and then send this information back to all the sensors.

The first order radio model is used to compute energy consumption in this paper. The transmission energy (E_{Tx}) and receiving energy (E_{Rx}) can be computed as follows:

$$E_{Tx}(l,d) = E_{elec} \times l + \epsilon \times l \times d^{\beta} \tag{1}$$

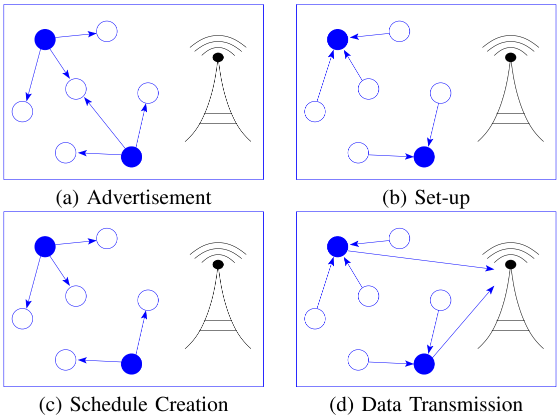
$$E_{Rx}(l,d) = E_{elec} \times l \tag{2}$$

where l is the data size in bits; d the distance between source and destination, E_{elec} the energy consumption per bit. The values of ϵ and β depend on the value of d. If $d \leq d0$, ϵ and β will be ϵ_{fs} and 2. Otherwise, they will be ϵ_{amp} and 4. Note that ϵ_{fs} and ϵ_{amp} are the amplifier cost. For an objective that is located at a distance longer than $d0$, the amplifier will have to consume much more energy to reach it. In summary, the goal of the clustering problem of a WSN is to maximize "the alive nodes" and "the remaining energies" of all the sensors.

B. LEACH

The low energy adaptive clustering hierarchy (LEACH) [9] is a distributed architecture for a WSN. Its basic idea is to randomly select a set of sensors as CHs (relay nodes), each of which will be responsible for collecting the data from other sensor nodes and then transferring the compressed data to the BS. The "random selection" strategy of LEACH is not a good one for balancing energy consumption in the sense that each node has the same probability of being elected as the CH, but if a CH is far away from its sensors, then it may waste energy for the transfer of data from sensors to CH and base station. This is one of the important reasons that LEACH is not suggested for a large-scale WSN [6]. Although LEACH has its disadvantages, it can be used as the touchstone in the development of a new clustering algorithm. Therefore, it still is a very well-known mechanism for clustering of a WSN.

Fig. 1 shows how sensors transmit data to the BS. A round represents a full transmission process. Each round consists of four steps, namely, advertisement phase, set-up phase, schedule creation, and data transmission. As shown in Fig. 1(a), in the advertisement phase, each node in the sensor network will randomly determine whether to be a CH with a predefined probability P, which guarantees that each node in the sensor network has a chance to become a CH. If there is no CH in a round, all the nodes in the network will each become a CH, thus directly transmitting the data it owns to the BS without relaying. LEACH also provides a threshold to prevent a sensor node from being a CH too frequently, which will cause some nodes to consume too much energy to transmit data and thus to be dead in an early round. The threshold is defined as


(a) Advertisement

(b) Set-up

(c) Schedule Creation

(d) Data Transmission

Fig. 1. Example LEACH. Note that arrowhead denotes the transmission direction of packet

$$T(n) = \begin{cases} \dfrac{P}{1 - P \times \left(r \times \bmod \left(\dfrac{1}{P} \right) \right)}, & \text{if } n \in G, \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

where P is the expected percentage of CHs, r is the current round, and G is a set of nodes, which have not become CHs in the last $1/P$ rounds. Set G will be reset every $1/P$ rounds so that every node will become a CH once in $1/P$ rounds. Once determined, each CH will broadcast a packet to nearby nodes to announce that it can relay their messages. As shown in Fig. 1(b), in the set-up phase, all the sensors (non-CH) will determine to which CH it belongs and then use that CH to relay its data. Cluster-head will then create schedules for its sensor nodes and broadcast them to the cluster members, as shown in Fig. 1(c). Finally, all the sensors will transmit data according to the schedules. Then, the CHs will aggregate data from members and send them to the BS, as shown in Fig. 1(d).

As observed by [19], [20], LEACH is the first protocol proposed for energy efficiency, which has a better performance than other earlier protocols in terms of the energy dissipation. Its distributed architecture also makes it easier to construct a network. However, the random rotation strategy makes nodes far from the base station consume more energy to relay data. Its distributed structure also makes it difficult for the sensor nodes to synchronize, thus causing more collisions when transferring.

C. LEACH-C

The low energy adaptive clustering hierarchy-centralized (LEACH-C) [18] was presented to deal with the problems of LEACH, which adjusts the advertisement phase to the BS; therefore, the BS is able to select the sensor nodes as CHs based on their remaining energies. As mentioned in [14], the CH selection can be regarded as a clustering problem, which is an NP-Hard problem. This explains that it is hard to find the optimum solution for this kind of problem in a reasonable time. For LEACH-C, simulated annealing is used for selecting

```
Initialization()
Resource_Arrangement()
While termination criterion is not met
  Vision_Search()
  Marketing_Research()
End While
```

Fig. 2. Outline of search-economics-based clustering algorithm.

the CHs. To make sure that the selected CHs have more remaining energy, the nodes that have a below average energy of all the nodes will not be elected as the CHs. Then, the fitness function of k-means is used as the objective function. After the selection process, the BS will inform CHs to continue with the remaining steps. The detail of LEACH-C can be found in [18].

Rather than a distributed architecture, LEACH-C was designed as a centralized structure, meaning that the selection of CHs is conducted on the BS. Geetha et al. [21] showed that LEACH-C is able to transmit data with a lower total energy dissipation and it can also receive more packets compared to LEACH because LEACH-C determines its CHs dynamically. However, LEACH-C will consume more energy in the set-up phase because all the nodes need to construct a connection to the BS. The "centralized structure" make it difficult for LEACH-C to modify its network settings after deployment.

III. The Proposed Method

A. The search-economics-based clustering algorithm

Fig. 2 gives an outline of SE. In this algorithm, the searchers (i.e., search agents) play the role of searching the possible solutions in the solution space. The resource_arrangement() operator will divide the solution space into subspaces, i.e., regions, each of which will use a certain number of samples to denote the searched information. Each region will keep one or more samples. The vision_search() operator will then generate candidate solutions from the solution of each searcher and the samples in each region. The proposed algorithm will then select the new candidate solutions based on the expected value of these candidate solutions, and it will also select regions that have a higher potential to get better solution or that have not been searched for a long time. The marketing_research() operator will then record the searched information of each searcher and each region to depict the solution space.

1) Solution Representation: The proposed algorithm encodes each of its solutions s_i as a binary string; i.e., $s_i = \{s_{i1}, s_{i1}, \ldots, s_{in}, \}$ where s_{ij} is the jth subsolution of solution s_i and n is the number of subsolutions of each solution. The first bit is the state of the first sensor, the second bit is the state of the second sensor, and so on. A value of 1 indicates that the corresponding sensor will become a CH whereas a value of 0 indicates that the sensor will not become a CH in this round.

2) Resource Arrangement: The resource arrangement operator will evenly distribute all the searchers to regions in the search space. This means that this operator will first divide the search space into h subregions, each of which will have w samples. The solution of each searcher is randomly generated with the first $\lfloor \log_2 h \rfloor$ bits being fixed to allow the searcher to be assigned to a particular region. For example, if the search space is divided into four regions, the first two bits of each solution will be fixed for these four regions, with $\{00\}$, $\{01\}$, $\{10\}$, and $\{11\}$ represent the first, second, third, and fourth regions, respectively. Note that the value of h and w are predefined. Then, the ith searcher will be assigned to the $(i \mod h)$-th region. The searchers are able to move from one region to another depending on the decision made by the vision_search() operator. Moreover, this operator will set the value of a bit corresponding to a dead sensor node to be 0, which cannot be changed anymore.

```
For each searcher
  Exchange information with samples
  Compute expected value
  Determine a region with higher potential
End For
```

Fig. 3. Outline of the vision search operator

3) Vision Search: As shown in Fig. 3, the main tasks of this operator is to exchange information, compute the expected value, and determine the search directions at later iterations. First, SECA will employ the crossover and mutation operators of genetic algorithm to *exchange information* between the searchers and samples; i.e., to crossover and mutate the solutions of searchers (i.e., s) and the samples of all the regions (i.e., m). Note that since SE is a framework of search algorithm, transition operators of other metaheuristic algorithms can also be used when applying it to an optimization problem. As far as this paper is concerned, the single point crossover is used to exchange information, and the bitwise mutation is used to improve the solution.

The *expected value* is calculated as follows:

$$e_{ij} = M_j \times V_j^i \times \rho_j \tag{4}$$

where e_{ij} is the expected value of the ith searcher in the jth region; and M_j is the ratio of the number of times it has been invested to the number of times it has not been invested defined as follows:

$$M_j = \frac{t_j^b}{t_j^a} \tag{5}$$

where t_j^a is the number of times it has been invested, and t_j^b is the number of times it has not been invested. V_j^i is defined as

$$V_j^i = \frac{\sum_{k=1}^{w} \sum_{l=1}^{2} f(v_{jk,l}^i)}{2w} \tag{6}$$

where f is the objective function; V_j^i, which is the experience of searcher i in region j, is the average objective value of each searcher in each region. For a minimization problem, $1 - (V_j^i - b)/b$, where b is the best solution so far, is used so that a smaller objective value implies a higher potential. The potential of region j is defined as

$$\rho_j = \frac{f(b_j)}{\sum_{i=1}^{h} \sum_{k=1}^{w} f(m_{ik})} \tag{7}$$

where b_j is the optimal solution so far of region j. For a minimization problem, $1 - \rho_j$ is used.

The *determination* operator is employed to select the potential regions to be invested. This means that for each searcher, SECA will randomly select some regions and compare their expected values, and then the region with the highest expected value will be invested; i.e., assigned the searcher.

4) Marketing Research: This operator is responsible for keeping track of the best solution and updating t_j^a and t_j^b, the values of which are set to 1 initially. t_j^b will be increased by 1 and t_j^a will be reset to the initial value 1 when a searcher is assigned to the jth region, and then t_j^a will be increased by 1 and t_j^b will be reset to the initial value 1, if no searcher is assigned to the jth region. For example, if region $\mathcal{B}$ is invested in this iteration, then t_1^b will be increased by 1 and $t_{\mathcal{B}}^a$ will be reset to 1. If region $\mathcal{B}$ is not invested, $t_{\mathcal{B}}^a$ will be increased by 1 and t_1^b will be reset to 1.

5) Objective Function: The objective function is a variant of that described in [22], which can be divided into two parts. The first part is the normalized energy consumption. The energy consumption for non-CHs is defined as

$$E_{\text{non-CH}} = \sum_{i=1}^{k} \sum_{j=1}^{N} E_{T_x}(l, d(\text{CH}_i, N_j)) \times x_{ij}, \tag{8}$$

where k is the number of CHs; N the number of nodes; l the packet size; and $d(\text{CH}_i, N_j)$ the distance between the ith CH and the jth node. The value of x_{ij} will be set to 1.0 if $N_j \in \text{CH}_i$; otherwise, it will be set 0.0. The energy consumption for CHs is defined as

$$E_{\text{CH}} = \sum_{i=1}^{k} l \times |C_i| \times (E_{\text{DA}} + E_{Rx}) + E_{T_x}(l, d(\text{CH}_i, \text{BS})), \tag{9}$$

where $|C_i|$ is the number of members belonging to the ith CH, and E_{DA} is the energy consumption for data aggregation. In this study, we assume that data from different nodes can be compressed and aggregated into l bits. The first part of the objective function is defined as follows:

$$f_1 = \frac{E_{\text{non-CH}} + E_{\text{CH}}}{\sum_{i=1}^{N} E_{T_x}(l, d(N_i, \text{BS}))}, \tag{10}$$

The denominator of f_1 is aimed to normalize the value of f_1 to an appropriate range. Also, the proposed algorithm uses direct transmission as the base so that if no solution is better than the direct transmission, it will use the direct transmission as the best result. If a solution better than the direct transmission is found, the value of f_1 will become smaller.

The second part is to compute in percentage the remaining energy of all the CHs with respect to the total remaining energy of each cluster. For a minimization problem, all you have to do is just to reverse the numerator and denominator. Symbolically, it is defined as

$$f_2 = \frac{\sum_{i=1}^{k} \left(\sum_{N_j \in C_i} E_{N_j}^{\text{init}} / E_{\text{CH}_i}^{\text{init}} \right)}{N \times 10^t}, \tag{11}$$

where $E_{N_j}^{\text{init}}$ is the remaining energy of the ith cluster of this round; $E_{\text{CH}_i}^{\text{init}}$ is the remaining energy of the ith CH of this round. The value t has to be adjusted when the region size or the number of sensors is altered. In general, the value should depend on the value of f_1. The value of f_2 is usually about 1.0% of f_1. In summary, the objective function of the proposed algorithm is defined as

$$f = f_1 + f_2. \tag{12}$$

TABLE I
PARAMETERS FOR ENERGY COMPUTATION

Parameter	Value
Initial energy	0.5 J/Node
Transferring energy	50 nJ/bit
Receiving energy	50 nJ/bit
Aggregating energy	5 nJ/bit
ϵ_{fs}	10 pJ/bit/m^2
ϵ_{amp}	0.0013 pJ/bit/m^4
$d0$	87.7 m

IV. EXPERIMENTAL RESULTS

A. *Environment Settings and Parameter Settings*

The programs were written in C++ and compiled with g++. The experimental environment is on a PC with 2.4GHz Intel Core2 Quad CPU and 4GB RAM. The operating system is Ubuntu 15.10 running Linux-4.4.0-62-generic x86_64. Three benchmarks are used to evaluate the performance of the clustering algorithm with different objective function. The first benchmark is 100 sensor nodes in a $100m \times 100m$ region, the second benchmark is 200 sensor nodes in a $200m \times 200m$ region, and the third benchmark is 100 sensor nodes in a $500m \times 500m$ region. The parameter settings for the sensor networks are shown in Table I.

In this paper, the proposes algorithm (SECA) is compared with two state-of-the-art clustering algorithms, which are LEACH [9] and GAECH [16]. All the experiments are carried out for 30 runs. The parameter settings of the clustering algorithms compared in this paper are as given below.

- LEACH: The percentage of CHs is set equal to 0.05.
- GAECH: The crossover and mutation rates are, respectively, 0.6 and 0.03 for $100m \times 100m$, 0.8 and 0.02 for the other experiments. The population size is set equal to 20. The parameters W_1, W_2, W_3, and W_4 for the objective function are 0.5, 0.3, 0.1, and 0.1, respectively.
- The proposed method: The number of searchers is 4, and so is the number of regions . Each region has two samples. The crossover rate is set equal to 1.0. The mutation rate is set equal to 0.02 for 100 nodes and 0.01 for 200 nodes. The parameter t for the objective function is 2.3 for $100m \times 100m$, 3 for $200m \times 200m$, and 2 for $500m \times 500m$.

B. *Results*

Fig. 4 show the results of these clustering algorithms for $100m \times 100m$ environment. The packet size in the first benchmark is set to 4,000 bits, and the BS is located at $(50, 175)$. Also, 200 iterations is performed for each round.

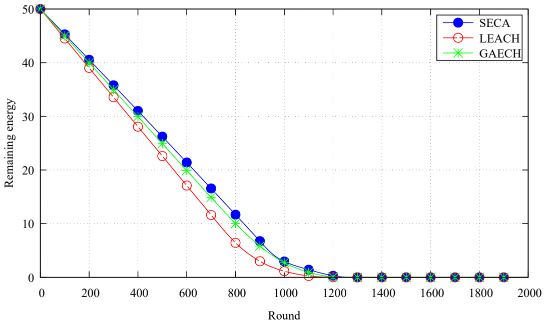


Fig. 4. Remaining energy for $100m \times 100m$.

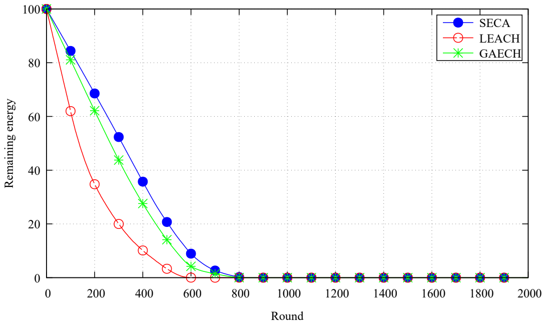


Fig. 5. Remaining Energy for $200m \times 200m$

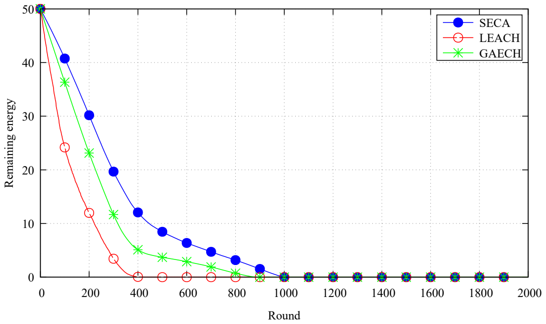


Fig. 6. Remaining Energy for $500m \times 500m$

TABLE II
ALIVE NODES FOR $100m \times 100m$

Percentage of dead nodes	Round		
	LEACH	GAECH	SECA
First node dead	694	624	**914**
10%	783	822	**929**
20%	810	871	**947**
30%	834	914	**956**
40%	862	953	**963**
50%	894	**985**	975
60%	923	**1015**	990
70%	959	**1046**	1015
80%	1023	**1086**	1065
90%	1102	1150	**1192**
Last node dead	**1260**	1237	**1260**

TABLE III
ALIVE NODES FOR $200m \times 200m$

Percentage of dead nodes	Round		
	LEACH	GAECH	SECA
First node dead	101	118	**265**
10%	136	328	**423**
20%	171	383	**458**
30%	205	428	**491**
40%	271	469	**531**
50%	330	507	**570**
60%	380	544	**601**
70%	442	580	**644**
80%	502	614	**687**
90%	541	707	**751**
Last node dead	627	**823**	**823**

TABLE IV
ALIVE NODES FOR $500m \times 500m$

Percentage of dead nodes	Round		
	LEACH	GAECH	SE
First node dead	41	69	**227**
10%	86	204	**308**
20%	104	247	**369**
30%	118	282	**398**
40%	142	314	**437**
50%	173	341	**510**
60%	224	371	**727**
70%	275	403	**912**
80%	309	588	**984**
90%	341	784	**993**
Last node dead	443	914	**1005**

TABLE V
IMPROVEMENT FOR $500m \times 500m$

Percentage of dead nodes	Round		
	LEACH	GAECH	SECA
First node dead	0	68.29	**453.66**
10%	0	137.21	**258.14**
20%	0	137.50	**254.81**
30%	0	138.98	**237.29**
40%	0	121.13	**207.75**
50%	0	97.11	**194.80**
60%	0	65.63	**224.55**
70%	0	46.55	**231.64**
80%	0	90.29	**218.45**
90%	0	129.91	**191.20**
Last node dead	0	106.32	**126.86**

Although Fig. 4 shows that the proposed algorithm can only improve a little bit of the energy consumption for the WSN, it further shows that the proposed algorithm can prolong the lifetime of the WSN before the first dead node appears. More precisely, the results show that the first dead node appeared after 914 rounds by using SECA, after 639 rounds by using LEACH, and after 623 rounds by using GAECH. SECA is able to prolong about 43% of the lifetime compared to the other clustering algorithms evaluated here in terms of the first node dead.

The detailed comparison between these clustering algorithms for the $100m \times 100m$ environment can also be found in Table II. The results further show that the proposed algorithm can significantly delay the sensor node from running out of its energy compared to the other clustering algorithms because it provides a balance plan to select the sensor nodes as the CHs.

Fig. 5 show the results of these clustering algorithms for the $200m \times 200m$ environment. The packet size for this benchmark is also set to 4,000 bits, the base station is located at $(100, 350)$, and all the clustering algorithms are carried out for 800 iterations each round. The results show that the proposed algorithm can provide a better plan to prolong the lifetime of a WSN in terms of the remaining energy of all the sensors node and the number of alive sensor nodes. The detailed comparison of these clustering algorithm can also be found in Table III, which shows that the proposed algorithm outperforms the other clustering algorithms compared herein, especially for large scale and complex benchmarks.

TABLE VI
p-VALUE OF t-TEST FOR $500m \times 500m$

Percentage of dead nodes	p-value	
	LEACH & SECA	GAECH & SECA
First node dead	1.93049E-38	1.03935E-28
10%	7.64792E-39	3.18984E-28
20%	4.02923E-40	2.36199E-34
30%	8.6221E-37	4.51272E-33
40%	4.03746E-34	3.10524E-36
50%	1.51895E-35	7.25208E-40
60%	2.31914E-36	2.4767E-44
70%	1.82286E-36	7.29176E-49
80%	2.3368E-38	2.27339E-24
90%	2.92847E-38	4.5232E-34
Last node dead	5.4442E-37	1.19321E-19

Fig. 6 and Table IV show the results of these clustering algorithms for the $500m \times 500m$ environment. Here, a 200-bit packet size is used to make sure that the network will not die in early round, the BS is located at $(250, 750)$, and each of the clustering algorithms is carried out for 200 iterations each round. Table V shows the improvement of GAECH and SECA with respect to LEACH in percentage, which is defined as

$$\frac{\mathcal{A}_{\text{new}} - \mathcal{A}_{\text{orig}}}{\mathcal{A}_{\text{orig}}} \times 100\%,$$

where $\mathcal{A}_{\text{new}}$ is either GEACH or SECA, and $\mathcal{A}_{\text{orig}}$ is LEACH. Table VI uses t-test to show that the results of the proposed algorithm are significantly better than those of LEACH and GAECH. This means that this experiment shows that the proposed algorithm can provide a better solution to prolong the lifetime of a WSN, and the differences are enlarged when the benchmarks become large and complex.

V. Conclusion

In this paper, an SE-based clustering algorithm is proposed to solve the CH selection problem of a WSN. Both GAECH and the proposed algorithm were carried out the same number of iterations. It can be easily seen that the proposed method can effectively extend the lifetime of a WSN before the first dead node appears and use less energy to transmit data than the other clustering algorithms. In the future, we will attempt to find a better way to define the objective function, and we will also try to improve the performance of the proposed algorithm.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions on the paper. This work was supported in part by the Ministry of Science and Technology of Taiwan, R.O.C., under Contracts MOST105-2221-E-005-091, MOST105-2221-E-110-067, MOST106-2221-E-197-001, MOST106-2221-E-110-023, and MOST106-3114-E-005-001.

References

[1] T. Rault, A. Bouabdallah, and Y. Challal, “Energy efficiency in wireless sensor networks: A top-down survey,” *Computer Networks*, vol. 67, pp. 104–122, 2014.

[2] T. Qiu, A. Zhao, F. Xia, W. Si, and D. O. Wu, “ROSE: Robustness strategy for scale-free wireless sensor networks,” *IEEE Transactions on Networking*, pp. 1–16, 2017.

[3] Q. Chi, H. Yan, C. Zhang, Z. Pang, and L. Da Xu, “A reconfigurable smart sensor interface for industrial wsn in iot environment,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1417–1425, 2014.

[4] X. Y. Liu, Y. Zhu, L. Kong, C. Liu, Y. Gu, A. V. Vasilakos, and M. Y. Wu, “Cdc: Compressive data collection for wireless sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 8, pp. 2188–2197, 2015.

[5] R. R. Rout and S. K. Ghosh, “Enhancement of lifetime using duty cycle and network coding in wireless sensor networks,” *IEEE Transactions on Wireless Communications*, vol. 12, no. 2, pp. 656–667, 2013.

[6] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, “Energy-efficient routing protocols in wireless sensor networks: A survey,” *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, pp. 551–591, 2013.

[7] F. Yu, Y. Li, F. Fang, and Q. Chen, “A new tora-based energy aware routing protocol in mobile ad hoc networks,” in *Proceedings of 3rd IEEE/IFIP International Conference in Central Asia on Internet*, 2007, pp. 1–4.

[8] F. Zabin, S. Misra, I. Woungang, H. Rashvand, N.-W. Ma, and M. A. Ali, “Reep: data-centric, energy-efficient and reliable routing protocol for wireless sensor networks,” *IET Communications*, vol. 2, pp. 995–1008, 2008.

[9] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” in *Proceedings of the 33th Annual Hawaii International Conference on System Sciences*, 2000, pp. 1–10.

[10] O. Younis and S. Fahmy, “Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, 2004.

[11] C.-H. Lung and C. Zhou, “Using hierarchical agglomerative clustering in wireless sensor networks: An energy-efficient and flexible approach,” *Ad Hoc Networks*, vol. 8, no. 3, pp. 328–344, 2010.

[12] J. J. Lotf, M. N. Bonab, and S. Khorsandi, “A novel cluster-based routing protocol with extending lifetime for wireless sensor networks,” in *Proceedings of IFIP International Conference on Wireless and Optical Communications Networks*, 2008, pp. 1–5.

[13] C. W. Tsai, T. P. Hong, and G. N. Shiu, “Metaheuristics for the lifetime of wsn: A review,” *IEEE Sensors Journal*, vol. 16, no. 9, pp. 2812–2831, 2016.

[14] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless microsensor networks,” *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[15] D. C. Hoang, P. Yadav, R. Kumar, and S. K. Panda, “Real-time implementation of a harmony search algorithm-based clustering protocol for energy-efficient wireless sensor networks,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 774–783, 2014.

[16] B. Baranidharan and B. Santhi, “GAECH: Genetic algorithm based energy efficient clustering hierarchy in wireless sensor networks,” *Journal of Sensors*, vol. 2015, 2015.

[17] C.-W. Tsai, “An effective WSN deployment algorithm via search economics,” *Computer Networks*, vol. 101, pp. 178 – 191, 2016.

[18] W. B. Heinzelman, “Application-specific protocol architectures for wireless networks,” Ph.D. dissertation, Massachusetts Institute of Technology, 2000.

[19] S. K. Singh, M. Singh, and D. Singh, “A survey of energy-efficient hierarchical cluster-based routing in wireless sensor networks,” *International Journal of Advanced Networking and Application*, vol. 2, no. 02, pp. 570–580, 2010.

[20] D. Goyal and M. R. Tripathy, “Routing protocols in wireless sensor networks: A survey,” in *Proceedings of Second International Conference on Advanced Computing & Communication Technologies*, 2012, pp. 474–480.

[21] V. Geetha, P. Kallapur, and S. Tellajeera, “Clustering in wireless sensor networks: Performance comparison of leach & leach-c protocols using NS2,” *Procedia Technology*, vol. 4, pp. 163–170, 2012.

[22] N. M. A. Latiff, C. C. Tsimenidis, and B. S. Sharif, “Energy-aware clustering for wireless sensor networks using particle swarm optimization,” in *Proceedings of International Symposium on Personal, Indoor and Mobile Radio Communications*, 2007, pp. 1–5.