

Improving the Energy of Wireless Sensor Networks Using Genetic Algorithm

Alaa Al-Shdaifat, Khalid Batiha, and Wafa alsharafat

shdaifatalaa@yahoo.com

Summary

Energy efficiency is an essential issue for wireless sensor networks (WSNs), all sensor nodes equipped with a single or low-energy battery. The efficient method for saving energy is clustering network nodes to avoid long-distance communication with the base station. This action preserves energy of the sensor nodes for long and extends their lifetimes. Furthermore, selection of the cluster heads (CHs) for better, energy-efficient routing plays an important role in expanding the network lifetime. In this paper, a new centralized clustering algorithm based on the enhanced Genetic Algorithm (CCA-EGA) is used to find an adequate number of CHs in the network. New crossover methods are applied concurrently with the enhanced Genetic Algorithm (EGA) to improve CH selection in terms of energy consumption. These methods are called Simple Arithmetic Crossover (SMX) and Discrete Crossover (DX). The testing results suggest that the proposed CCA-EGA enhances energy consumption of the WSN by preserving CHs for long and extending their lifetimes.

Key words:

Genetic Algorithm; Crossover; SMX; DX; Cluster Head.

1. Introduction

The Wireless Sensor Network (WSN) was firstly designed and introduced by the military and defense industries. A wide range of applications is offered by the WSNs due to their affordability and multi-functionality. Some of the accustomed applications of these networks include:

- hazardous environment monitoring like the Real-Time Volcano Early Warning System [1];
- tactical military application for monitoring territories of the enemies [2];
- medical or healthcare applications [3];
- the Intelligent Traffic Management System (ITMS) [4].

One of the biggest advantages of the WSN is its ability to work in extreme and harsh environments where human presence is nearly impossible such as the environments of

remote active volcanoes or earthquakes. Mostly, such applications require thousands of sensor networks (SNs) and a single base station (BS) or very few such stations. In other respects, deploying hundreds or thousands of low-cost SNs imparts new benefits to these sensing applications like reduced cost of node, improved accuracy and reliability, and extended scope of sensing. In principle, having size and cost restrictions on SNs results in a corresponding restriction on resources like the energy, memory, computational speed, and the communication bandwidth required for highly-efficient resource utilization. In the WSN, energy and lifetime are of paramount importance. One of the major restrictions on the application areas of the WSN is the threat of limited energy resources [5].

In view of the foregoing discussion, this study develops and presents a new centralized-clustering algorithm based on the enhanced GA (CCA-EGA) to find an adequate number of CHs in the network for improved WSN energy efficiency.

The remainder of this paper is organized as follows: Section 2 provides a review of previous related works and Section 3 presents the proposed Genetic Algorithm for WSN Clustering. Then, Section 4 analyzes the experimental results of the Genetic Algorithm for WSN Clustering and discusses them. Lastly, Section 5 presents the conclusions of this study.

2. Related Works

The GA-based clustering technique, GABEEC, was proposed in wireless sensor networks [6]. This algorithm operates in two phases: a set-up phase and a steady-state phase over rounds. During the set-up phase, the cluster created throughout the network remains unchanged. So, after every round, there are static clusters with only changing CHs inside the formed clusters. In the stable state phase, all nodes communicate with CH. Once the CH receives all the data, it performs data aggregation. Then, it sends the compressed data in one packet to the BS. After completion of any round, the BS checks the energy status of the CHs with member nodes (MNs) for each formed

cluster. If the BS finds that energy of the CH is less than the average energy of all the MNs, then the CH is changed. Afterwards, the MN which has the highest residual energy becomes a CH.

Researchers often consider three parameters for selecting fitness function, which, in effect, are the three parameters taken into account by GABEEC for the fitness function. These parameters are:

- the round at which the first node dies (Rfnd),
- the round at which the last node dies (Rlnd),
- cluster distance (C), that is, the sum of the distances from the MNs to the CH and the distance from the CH to the BS.

The evaluation results revealed that GABEEC is more efficient than the classical LEACH. The results also showed that GABEEC extended the lifetime of the network. However, a limitation of GABEEC is that rotation of the CHs takes place, which results in a lack of efficiency because the clusters are not changed throughout the lifetime of the WSN [7, 8].

Abo-Zahhad et al. [9] developed a new energy-efficient clustering protocol called Genetic Algorithm-based Energy-Efficient adaptive clustering hierarchy Protocol (GAEEP) based on centralized clustering. The main concern of GAEEP is to efficiently maximize the network lifetime and improve the stable period of the WSNs in both the homogenous and heterogeneous networks. This protocol focused on prolonging the lifetime of the WSNs through finding the optimal number of CHs and their positions by using GA. Like the LEACH protocol, operation of GAEEP is divided into rounds, each comprising two phases; a set-up phase and a steady-state phase.

The steady-state phase of GAEEP operates like the steady-state phase in LEACH. During the set-up phase, the BS sends a short inquiry message that demands information about the network parameters for all the SNs in the sensor field. The parameters asked for include positions and energy levels of all SNs. Using this information feedback from the SNs, the BS employs GA to decide on the optimal number of CHs and their locations in the WSN based on minimization of energy consumption in the case of the communication process.

Consequently, the BS specifies the optimal selected CHs and their assigned members for each CH by sending an advertisement message to the SNs. When CHs are selected and identified members of each CH to form clusters, then each CH will allocate the time-division multiple access (TDMA) schedule to its MN for initiation of data transmission. The TDMA works on avoiding intra-cluster

collisions. Its operation is useful for reducing the energy consumption term. Essentially, the GAEEP selects the SN whose residual energy is equal to, or larger than, the average energy of all alive SNs in the sensor field (i.e., for any alive node to become CH it should satisfy the condition that $E_{residual} \geq E_{average}$).

In principle, the GAEEP applies binary representation in which each bit corresponds to one SN within a chromosome of the genetic population. The fitness function of GAEEP that was proposed to determine the optimal number of CHs and their locations is given by equation (1):

$$Fitnessfunction = w \times \left(\frac{E_{dissipated}}{E_{live}} \right) + (1 - w) \times \left(\frac{L}{N_{live}} \right) \quad (1)$$

where :

w is a pre-defined weighting factor that ranges from 0 to 1 ($0 \leq w \leq 1$)

Elive is the total energy of all alive SNs, which can be calculated using equation (2):

$$\begin{aligned} & \sum_{j=1}^{N_{live}} initial_{Energy}(j) \\ & = N_{live} \times initial_{Energy} \end{aligned} \quad (2)$$

Edissipated represents the total dissipated energy of all alive SNs, which is given by equation (3):

$$\begin{aligned} & E_{dissipated} \\ & = \sum_{i=1}^L [(E_{CH}(i) + (E_{CH_{CP}}(i) + \sum_{l=1}^{N_{live}-L} (E_{mem-i}(l) + E_{mem_{CP-i}(l)}))] \end{aligned} \quad (3)$$

ECH_CP and Emem_CP-i are the energies dissipated in the control packet for the CH and for the MN, respectively.

The GAEEP has been applied to the main GA processes to obtain the solution that best improves the lifetime of the WSN. The simulation results showed that the GAEEP protocol improves the lifetime of the network in terms of stability over the stability improvement achieved by LEACH.

Singh et al. [10] proposed another GA-based algorithm called Clustering and Energy Efficient Routing Protocol for Wireless Sensor Network using GA. This algorithm employs multi-hop communication among the CHs and from the CH to the BS for transmission of the data. In this algorithm, clustering and routing are performed in rounds like in LEACH. As well, each round consists of two phases; an initial phase and a steady phase. During the initial phase of each round, a genetic population and fitness function are used to choose the CH and MN, where the BS selects M initial CHs from the N nodes in the first round of this algorithm. Then, a MN will be selected for each CH. This proposed GA applies crossover and mutation operators in the fitness function to select the CH. The fitness function is defined as follows:

$$\text{Fitnessfunction} = (G(n) * E(r) * D) \quad (4)$$

where

- G(n): The probability for the node to become CH as defined in LEACH.
- E(r): The residual energy of the node.
- D: The distance between the node and the BS.

Based on the fitness function, the BS selects the particular chromosome that has the minimum difference in energy relative to the last round. Subsequently, the BS sends message for the selected CH. The after, the selected CH sends an advertisement to the normal node for it to join the closest CH and form a cluster. Once the initial phase ends, the steady-state phase for data transmission is initiated. It works as follows:

According to TDMA, the CH receives a data packet from the MN based on its allocated time slot (TS).

The CH aggregates and compresses the data received from the attached normal node.

Multi-hop communication is then applied for data transmission. Afterwards, the CH sends controlled messages to the nearer CHs. The perfect path will then be chosen. This process goes on until the BS receives all the due packets.

The perfect path is a route in which the CH has higher energy, a closer distance to the BS, and no traffic. Then, the data, after being collected by the head cluster, are sent to the same route. This process proceeds to the next CH until it reaches to the BS. The results of simulation of this algorithm indicated that it increases the network lifetime over the increase obtained from the LEACH protocol.

In [11], the researchers proposed an efficient energy CH selection technique based on GA to prolong the network

lifetime. The proposed GA is a BS-assisted, centralized clustering algorithm. Briefly, the SNs in this proposed technique send their information of energy and location to the BS. Then, the BS applies GA to optimize the number and selection of CHs. After that, the BS assigns CH to all nodes. Subsequently, a TDMA schedule is planned for each cluster and detailed information about clusters and TDMA are broadcasted to the network. Thus, all the SNs send their sensed data in a given time slot. This proposed GA works over rounds, each consisting of a set-up phase and a steady-state phase. To achieve load balancing, a re-clustering is used after the round-time is over by means of rotating the role of the CH among the nodes. The fitness function of the proposed GA focuses on selecting the CH according to the residual energy and pays good attention to tradeoff between the inter-, and intra-communication distances. Crossover and mutation operators are applied on representative chromosomes and the roulette wheel selection procedure is applied to select chromosomes for generation of a new population.

The proposed fitness function is defined as follows:

$$\text{Fitnessfunction} = E + (N - CH) + \frac{IC}{N} + \frac{BSD}{N} \quad (5)$$

Where:

E: Remaining Energy

CH: Number of CHs

IC: Total Intra-cluster Communication Distance

BCD: Total Distance from CHs to the BS

Outcomes of simulation of this proposed GA revealed prolongation of the network lifetime over the prolongation produced by LEACH and -LEACH-C. However, despite the fact that the proposed GA contributes to prolonged network lifetime that is comparable with the prolongations achieved by LEACH and LEACH-C with small enhancement in the node death rate, scaling of the fitness function still needs to be properly tuned.

An Optimized Genetic Algorithm (OGA) was proposed in [12]. It is an optimized, energy-efficient GA that is based on LEACH and which is intended to achieve increased WSN lifetime. This suggested OGA uses GA to optimize selection of the CHs for the network based on LEACH.

Much like the LEACH, this OGA proceeds in two phases per round for clustering: a set-up phase and a steady-state phase (for data transmission). In the set-up phase of the OGA, selection of the CHs is performed using GA. The normal nodes are associated with the respective nearest selected CHs in order to form a cluster. The OGA uses distributed clustering to create clusters in the network. It performs the main genetic operations (selection, crossover, and mutation) for optimizing selection of the CHs.

A new simple fitness function was designed in the OGA that considers the distances of the SNs from the BS and their energy status. This fitness function is calculated as follows:

$$\text{Fitnessfunction} = \frac{\text{sum}_{\text{Energy}} \times \sin(N_{\text{alive}})}{E_{\text{CH}} \times D_{\text{CH}}} \quad (6)$$

Where:

- $\text{sum}_{\text{Energy}}$: Sum of energies of the alive nodes
- N_{alive} : The number of the alive nodes
- E_{CH} : Sum of the energies of the candidate CHs
- D_{CH} : Sum of the distances of the candidate CHs

According to this fitness function, the OGA applies binary coding GA to select the most fitting chromosome for selection of the optimum number of CHs based on the maximum energy value. The simulation results showed that this proposed OGA increased network lifetime in terms of the stability period. Furthermore, the instability period decreased in OGA below that associated with the LEACH.

In [13], an energy-efficient protocol based on GA and distance-aware routing was proposed for WSNs. It is an improvement to LEACH that is called Genetic Algorithm-based Distance-Aware routing protocol (GADA-LEACH). This protocol employs GA to improve CH selection, considering different parameters for computation of the fitness function such as the energy of the node, distance between the node and the cluster head, and distance between the CH and the sink. In addition, this proposed protocol includes an intermediate rechargeable node known as a relay node that is placed between the BS and the CH to achieve distance-aware routing. The purposes of the relay node are to facilitate the communication between the CH and the BS and improve the energy efficiency by reducing the distances among them. The CH first transfers the

collected information to the relay node if the distance between the CH and the BS is high. Then, the relay node further transfers this information to the BS. However, if the distance is low, then the CH directly transfers the information to the BS. For the GADA-LEACH to select the best node as CH, the fitness function proposed for each node is computed according to equation (8):

$$\text{Fitnessfunction} = [(0.3 \times F1) + (0.35 \times F2) + (0.35 \times F3)] \quad (7)$$

Where:

$$F1 = \frac{\text{Energyofallnodes}}{\text{EnergyofCHs}}$$

$$F2 = \frac{\text{EuclidianDistanceofCHwithitsassociatednodes}}{\text{Numberofnodesincluster}}$$

$$F3 = \frac{\text{EuclidianDistanceofBSfromallCHs}}{\text{NumberofCHsformed}}$$

Initially, the population for CH selection is generated based on the percentage of CHs. After that, the fitness of each node is evaluated for best results so that the best nodes can be selected as CHs. Roulette wheel selection is used and crossover and mutation operations are applied. Fitness of each individual is then evaluated again and compared with the initial fitness value to be updated by the corresponding new generation and, then, the fitness of the new updated population is evaluated again and compared with the initial fitness of the population. Finally, if the preset stopping criteria are met, then the best nodes are chosen from that population as the CHs for that round. The simulation outputs uncovered that GADA-LEACH performs better than the LEACH and that it can prolong the lifetime of the network but balance of energy of the overall nodes is not realized because the nodes that are placed far away from the relay node die quickly.

A hybrid energy-efficient clustering algorithm has been developed for the WSNs in [14] by integrating two

effective algorithms (the k-means algorithm and the GA) together. This clustering algorithm has substantially improved the network lifetime. The algorithm consists of two major processes; a cluster formation process and a CH selection process. In this algorithm, the BS executes the algorithm to create the clusters and select the proper CH for them. The k-means algorithm assists in setting up the clusters based on the location of each SN. It then evaluates the distances between the various SNs in order to create the clusters through use of the mean value of their distances. Once the k-means algorithm assigns the clusters, the GA is run by using genetic operations such as selection, crossover, and mutation to select the chromosomes which best fit with the best CH for each cluster in the network. The fitness function of the GA is focused on two factors, namely, the energy level and the distance from the BS. The proposed fitness function is similar to the fitness function of GABEEC. It is, therefore, obtained from equations.

In principle, this algorithm corresponds to a binary representation for the genetic population to determine if the particular node is CH or not (that is, member node). Outcomes of performance evaluation of the proposed GA disclosed that it extends the network lifetime in terms of the stability period over the extension obtained from LEACH.

In 2019, Kim et al. [15] suggested using the GA approach for energy-efficient routing based on the clustering method. They applied GA to select the CH and the route discovery process in one phase based on a centralized methodology to extend the lifetime of the WSN. In their approach, the BS executes the proposed algorithm as follows:

- (i) the BS counts the number of alive SNs;
- (ii) the BS applies a fuzzy c-means algorithm in order to group the alive SNs and form cluster;
- (iii) the BS applies a GA to select CHs for each cluster and, then, constructs paths from the CHs to the BS;
- (iv) the BS creates TDMA schedule and broadcasts it with the path information;
- (v) the SNs collect sensed data and forward them through the CHs to the BS based on multi-hop routing.

The implemented GA is represented in two chromosomes called candidate cluster head (CCH) and CP, where the CCH chromosome is intended for CH selection

for each cluster while the CP chromosome is devoted for discovery of the path from the CHs to the BS. The proposed fitness function designed to compute the expected energy consumption is provided by equation 7:

$$\begin{aligned} \text{Fitness function } (i) &= \sum_{n=1}^N \{2 \times E_{elec} \times h_i(n) \\ &+ A_i(n)\} \end{aligned} \quad (8)$$

where “

$2 \times E_{elec} \times h_i(n)$: is the total energy of the transmitters and receivers that is dissipated in running the radio electronics,

$A_i(n)$: is the amplifier power of the transmitter for delivering data from n -th to the BS as in the radio model [16]

The simulation results showed that the proposed algorithm outperforms the LEACH, the Power Efficient Gathering in Sensor Information Systems (PEGASIS), and the Base Station Controlled Dynamic Clustering Protocol (BCDCP), especially in large networks.

3. The Genetic Algorithm proposed for WSN Clustering

Conventionally, the simplest way to represent solutions in GA is binary coding, where a string of binary digits/bits (0's and 1's) is employed to form the decision variables of the optimization problem in the genetic population. Basically, in Binary Coding-based Genetic Algorithm (BCGA), after generating the initial population and evaluating the possible solution, coding processes are carried out to represent the chromosomes in the reproduction process. After production of the next generation, the chromosome needs to be decoded. However, high precision solutions using BCGA require very long chromosomes.

In [17], Real Coding-based GA (RCGA) was applied to model the optimization problem via real-valued representation (or encoding) based on abstract of the genetic population that represents the possible solution so as to design a highly energy-efficient routing algorithm. A comparison of the RCGA feature with that of the BCGA points out that the RCGA representation of the solutions is very close to the natural formulation [17]. Due to that there is no need for mapping or converting the solution variables

to binary variables, the efficiency of the RCGA, which requires less memory for solution representation than the BCGA, is higher than that of the BCGA [18]. Furthermore, the RCGA proved to be able to solve optimization problems (i.e., continuous problems) with high robustness and accuracy, even after increasing the domain over that of the BCGA [17, 19]. Moreover, for effective genetic operation, the crossover and mutation operators, which can directly deal with real variables, are used, thus providing greater flexibility and freedom to employ different genetic operators [18]. This means that the reproduction operators applied using this type of GA can deal with the actual values of variables instead of their coded values.

The optimization algorithm proposed in the current study (CCA-EGA) is a centralized GA-based clustering strategy in which all executions are utilized in the BS that is responsible for clustering configuration of the network based on energy-efficient routing that aims at optimizing the process of cluster formation and CH selection. Ultimately, the GA operations are applied to achieve improvements in network lifetime and CH selection performance. In essence, the algorithm proposed herein considers dynamic clustering (or re-clustering) in the network, where routing configuration is performed to improve network lifetime. Similar to the LEACH version, the whole operation of the proposed algorithm is divided into rounds. Basically, each round can be separated into two main phases, namely, a set-up phase, which includes CH selection and clusters creation, and a steady-state phase, which includes data transmission. Essentially, the steady-state phase in clustering by CCA-EGA is similar to that of LEACH. The difference between this algorithm and LEACH is in the set-up phase, where in the proposed algorithm the BS performs calculations for optimization of CH selection for every node to become CH using GA-based methodology.

1) The Set-Up Phase

Based on a centralized methodology, the BS which possesses unlimited resources in terms of power, storage, and computation executes the GA to optimize all clustering configurations for the WSN. To our knowledge, the process of re-clustering configurations in all existing clustering protocols and algorithms is decided progressively, where the configuration for the round (r+1) is decided at the end of round r, without considering the configurations of further rounds. This phase can be subdivided into two steps; an information collection step and a clustering step. In the information collection step, the information of the network is collected only once. After that, the clustering step is

started, where the BS utilizes the received information to execute its central clustering based on GA to optimize the selection of the CHs throughout the overall network lifetime.

Cluster head selection is very important for energy-efficient clustering. In this section, we describe the structure of the GA that is intended to identify the optimum CHs which largely affect the network lifetime.

depending on the designed fitness function. The Euclidean distance $d(i, j)$ between nodes and the distance between node and BS are considered when selecting CH according to the following equation (9):

$$d(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (9)$$

In the foregoing equation, the variables x and y represent the x -coordinate and y -coordinate of the node, respectively.

The GA begins with a population of possible solutions by defining the initial population that has been generated randomly. Each solution in the population is linked with chromosomes in the GA that contains a sequence of genes. Each of these genes is assigned to CH in the network. Then, the objective function for every solution is evaluated to determine its optimality.

In every generation, the GA keeps the best-fit chromosome and replaces the others, depending on their fitness levels. Thereafter, the proposed GA is run so as to produce the next population for improving the network lifetime. Once the BS declares all clustering configurations by selecting the optimal set of CHs, the BS transmits a message (ADV_CONFIGURATION message) that informs the clustering configuration of the first round.

This message includes IDs of these CHs, the MN associated with each CH, and the TDMA schedule. In essence, the TDMA schedule that is allocated or arranged by the BS for each cluster member is proportional to the cluster size. Thus, to maintain energy using the TDMA schedule, each node turns off (OFF) its radio. It turns it on (ON) only for data transmission when its TS comes. On the other hand, the TDMA schedule organizes the intra-cluster communication, hence leading to reduction in interference within each cluster. Once formation of clusters is finished and the TDMA schedules are constructed and distributed, the steady-state phase is started. After completion of the first round in the network and during the set-up phase of the second round, only the BS needs to broadcast an ADV_CONFIGURATION message containing the pre-defined configuration for the incoming round that has been already computed. However, it is necessary to pay good

attention to the percentage of the optimal number of CHs, which is a crucial factor since large numbers and small numbers of CHs may drain the energy of the network.

Cluster head selection is very important for energy-efficient clustering. Here, we describe the structure of the GA that is intended to identify the optimum CHs which largely affect the network lifetime. The main GA processes are:

- 1- GA Encoding.
- 2- Population initialization.
- 3- Selection.
- 4- Crossover.
- 5- Mutation.
- 6- Termination process.

As follows, we will provide a brief overview of each GA processes

- The GA encoding

Each solution is represented by a chromosome with a specific length and every chromosome is subdivided into several genes, that is, every chromosome represents a set of CHs. Which represents a collection of feasible solution.

In general, the chromosome is a representation of a nest of SNs, in which every gene value of any chromosome indicates node ID (unique identifier ranging in value from 1 to n, where n represents the number of SNs in the network). Thus, each gene stands as a variable in the problem. Fig.1 describes the chromosome structure of the CCA-EGA proposed here. The fixed length of each chromosome is specified as follows (equation 9):

$$\text{ChrLength} = \text{percentageofCHs} \times r_{\text{max}} \quad (9)$$

Where:

r_{max} : the maximum number of rounds in the network.

ChrLength: the chromosome length.

However, any chromosome (i.e., solution vector) can be described as a set of CHs (genes) that govern clustering configurations for the overall assigned rounds (rmax) in the network. This will help in improving the clustering configurations with adapted round duration and eliminating the costs of periodicity of re-clustering. To demonstrate the

representation of a solution, consider a network that has 500 rounds and 5% of the nodes as CHs.

The ChrLength can be calculated as $500 \times 5 = 2,500$ genes. Accordingly, each gene within the chromosome represents the CH-ID of the network. Fig. 1 demonstrates the structure of the chromosome in the population.

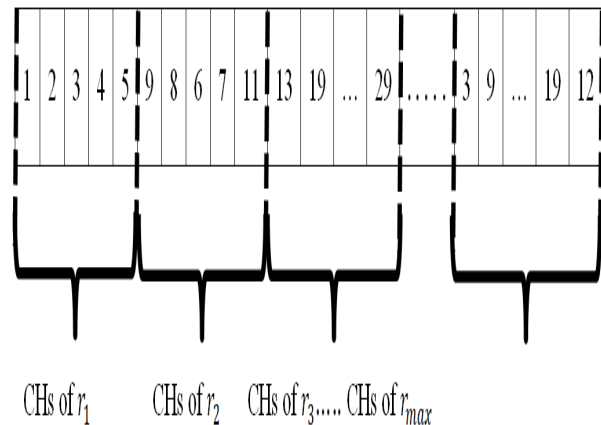


Fig. 1. Chromosome Structure

- Population initialization

The population (Pop) is initialized randomly and stored in the BS. Each solution (chromosome) has an evaluation value assigned to it and stored in a different vector-matrix f.

- Selection

The selection operation is mainly performed to pave the way for later mating or crossover for choosing chromosomes from the current population using a specific selection strategy. Essentially, the more fitting the chromosomes, the higher the probability for them to be passed on to the next generation.

- Crossover

Once the selection process is finished and the suitable chromosomes are prepared, the crossover (recombination) operator can be applied for mating. In GA, the crossover probability P_x commonly ranges from 0.5 to about 1.0 [20].

However, the crossover process ensures the exchange of genetic material between two members of the population

called parent chromosomes (PrA and PrB) and, thereupon, fabricates chromosomes that are more likely to be better than the parent chromosomes, which hopefully inherit their best characteristics to produce new children (new solutions).

In addition, we need to avoid premature convergence in GA as much as possible by selecting appropriate crossover operators. To combine (cross) any sub-structures of two-parent chromosomes.

$$Pr_A^{gene} = Pr_A^{gene 1}, Pr_A^{gene 2}, \dots, Pr_A^{genen}$$

and

$$Pr_B^{gene} = Pr_B^{gene 1}, Pr_B^{gene 2}, \dots, Pr_B^{genen}$$

to produce a child

$$Ch_i^{gene} = Ch_i^{gene 1}, Ch_i^{gene 2}, \dots, Ch_i^{genen},$$

Where:

Pr_A^{gene} is the *i*th gene value of the PrA chromosome.

Pr_B^{gene} is the *i*th gene value of the PrB chromosome.

Three crossover operators have been used in this work based on the real parameter. These are the Simple Arithmetic Crossover (SMX) operator and Discrete Crossover (DX) operator. In these operators, a specific value of the parameter α is specified and used to represent a random value in the range of 0 to 1. These operators were applied as described in [21]. It should be spotlighted that these operators can be used with the real-parameter GA.

Two crossover operators have been used in this work, these are:

- Simple Arithmetic Crossover (SMX) operator
- Discrete Crossover (DX) operator.

In these operators, a specific value of the parameter α is specified and used to represent a random value in the range of 0-1.

The SMX operator can be applied like the traditional one-point crossover as follows. First, a cross-point gene (*k*) is determined. Then, the value of the first *k* gene of a randomly-chosen parent is set by taking it and coping with the child. Afterwards, the rest of the gene value of a child is produced using the arithmetic average of the parents.

The DX operator produces a child from the parents with equal probabilities [21].

- Mutation

The mutation operation is done to prevent premature convergence of a new generation and to ensure a certain level of diversity in the genetic population through introducing random changes to the individuals who may provide a new individual.

In general, mutation is applied sequentially after crossover and is performed according to the mutation rate P_m , which represents the percentage of genes that are referred to as a mutation. Generally, the GA has a probability of mutation P_m that ranges from 0.001 to 0.05 [20].

A high mutation probability will interrupt the evolution process, whereas a low mutation probability does not correspond to a good likelihood for change.

In the herein proposed technique, mutation was performed according to P_m which operates by changing the gene value with a random number in the interval $[1, n]$, where *n* refers to node-ID in the network.

- Termination process

The algorithm stops looping and returns the best chromosome in the current population.

In the technique proposed in this paper, the process is terminated when the maximum number of generations is achieved.

2) The Steady-State Phase

The a steady-state phase is started after the set-up phase is completed. The steady-state phase which was applied in the CCA-EGA proposed in this study is identical to that used in LEACH. It is initiated once the clusters are organized. Data transmission will dictate when each node sends the sensed data to the corresponding CH. Once all the due data have been received by the CH, each CH performs aggregation of the received data so as to discard the redundant and un-correlated data and transmit the aggregated or fused data in one packet to the BS using single-hop communication.

4. Experimental results and Evaluation

A. Simulation Software (MATLAB)

All operations and simulation of the GA-based clustering were conducted using MATLAB software installed on a laptop with:

- Intel CORE i5

- 2.2 GHz CPU
- 4 GB RAM
- Windows 7 operating system.

MATLAB is a powerful high-level matrix-based programming language developed by Math Works. It can be used for numerical computation, visualization, and application development. It is an easy-to-use programming language that can be employed to write mathematical programs using a large library of mathematical functions. This software has a good built-in graphic tool that allows graphs to be plot in different dimensions (i.e., 2-D and 3-D). This feature allows programmers to visualize data using tools for creating custom plots. Moreover, MATLAB has several toolboxes for modeling and simulations, data analysis and processing, and algorithm development and optimization.

B. Performance Measurement

To evaluate the proposed CCA-EGA, we compared its performance with that of the well-known LEACH protocol based on the Network Lifetime.

There is no universally-agreed upon definition for the network lifetime but it can be described as the time when the First Node Dies (FND) [22].

C. The Simulation Setup

Simulation of the proposed CCA-EGA was based on a configuration of 100 SNs distributed randomly in a two-dimensional representation over a network area (i.e., sensor field) of 100m x 100m with far BS at the coordinates of 100 and 375. Additionally, in the simulations, the SNs were homogenous nodes. The parameters used in the simulations, as well as their values, are given in Table 1.

Table 1: Simulation Parameters and their values

Simulation Parameter	Value(s)	Simulation Parameter	Value(s)
Network Size (M×M)	100 m x 100 m	Energy to run radio electronics circuit (E_{elec})	50 nJ/bit
BS Location (X,Y)	(100, 375)	Threshold distance (d0)	87 m

Number of SNs	50, 100	Mutation probability	0.05
Percentage of CHs	5%	Crossover probability	0.8
Packet Size	4,000 bits	Primary population size	50
Control Packet Size	200 bits	Generation size	50
Max number of rounds	600	Elite	2
Initial energy (E_0) of SNs	0.5, 0.4 Joule/node	Crossover operator	SMX, DX
Data aggregation energy (E_{DA})	5 nJ/bit/signal		
Amplification energy for multipath (ϵ_{mp})	0.0013 pJ/bit/m ⁴		
Amplification energy for free space (ϵ_{fs})	10 pJ/bit/m ²		

D. Simulation Results and Evaluation

Simulation of the proposed CCA-EGA was based on a configuration of 100 SNs distributed randomly in a two-dimensional representation over a network area.

We discuss the results of simulations of the proposed CCA-EGA algorithm by analyzing its network lifetime, it should be highlighted that the results presented hereafter are averages of 10 independent runs of 600 rounds, each.

1) Comparison of network lifetimes

In essence, the GA operators of selection, crossover, and mutation are very important for achievement of the best performance. Thus, here we investigated two crossover operators; SMX and DX. Simulations were made to assess the performance of each crossover operator with each of the two studied selection operators. Table 2 and Table 3 show the efficiencies of the SMX and DX crossover operators in improving the network lifetime with or without fitness scaling under variant E0 values. The network lifetime metrics that were taken into account are:

- (i) FND
- (ii) certain fraction (percentage) of the dead nodes (10% to 90%).

Table 2: Comparison of network lifetimes without fitness scaling ($E_0 = 0.40$)

Crossover Operator	Rounds when Number of Dead Nodes								
	FND	10%	20%	30%	50%	70%	80%	90%	
SMX	24	59	78	92	118	153	173	211	
DX	29	58	73	88	114	147	174	213	

Table 5: Comparison of network lifetimes with fitness scaling ($E_0 = 0.50$)

Crossover Operator	Rounds when Number of Dead Nodes								
	FND	10%	20%	30%	50%	70%	80%	90%	
SMX	45	77	100	118	150	191	225	282	
DX	38	71	93	110	141	181	209	254	

Table 3: Comparison of network lifetimes without fitness scaling ($E_0 = 0.50$)

Crossover Operator	Rounds when Number of Dead Nodes								
	FND	10%	20%	30%	50%	70%	80%	90%	
SMX	26	54	71	87	115	159	191	286	
DX	27	54	73	90	117	153	175	208	

Tables 2 to Table 5 disclose that there are differences between the GA crossover operators with and without fitness scaling.

As to network lifetime, the results summarized in Tables 2 and 3 ($E_0 = 0.40$) lead to the conclusion that in the case of SMX, almost 70%, 80%, and 90% of the nodes died without fitness scaling. This took place in rounds 153, 173, and 211, respectively. By changing the value of ($E_0 = 0.50$) without fitness scaling, however, the foregoing percentages of node deaths took place in rounds 159, 191, and 286, respectively.

Furthermore, in the case of the SMX operator, Table 3 and Table 5 uncover that 70%, 80%, and 90% of the nodes died in rounds 159, 191, and 286, respectively. However, with fitness scaling these death percentages were reached in rounds 191, 225 and 282 respectively.

Table 4: Comparison of network lifetimes with fitness scaling ($E_0 = 0.40$)

Crossover Operator	Rounds when Number of Dead Nodes								
	FND	10%	20%	30%	50%	70%	80%	90%	
SMX	43	77	96	113	146	187	227	298	
DX	41	78	101	122	158	196	220	264	

2) Comparison of dead nodes

Fig. 2 and Fig. 3 make comparisons of incidences of node deaths in each round without fitness scaling under $E_0=0.50$. It can be seen that more nodes survive over rounds when using the different crossover operators without fitness scaling.

In addition, Fig. 2, Fig. 3, Fig. 4 and Fig. 5 make comparisons of incidences of node deaths in each round with and without fitness scaling under varying E_0 values. It can be seen in Fig. 4 and Fig. 5 that more nodes survive over rounds when using the different crossover operators without fitness scaling.

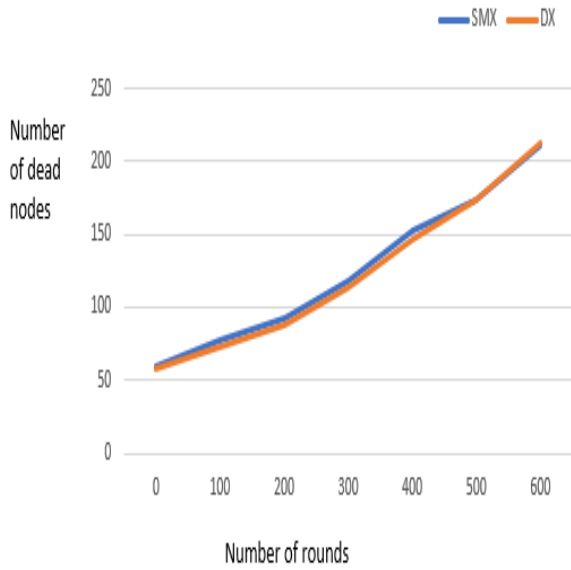


Fig.2. Comparison of dead nodes in each round without fitness scaling ($E_0 = 0.40$)

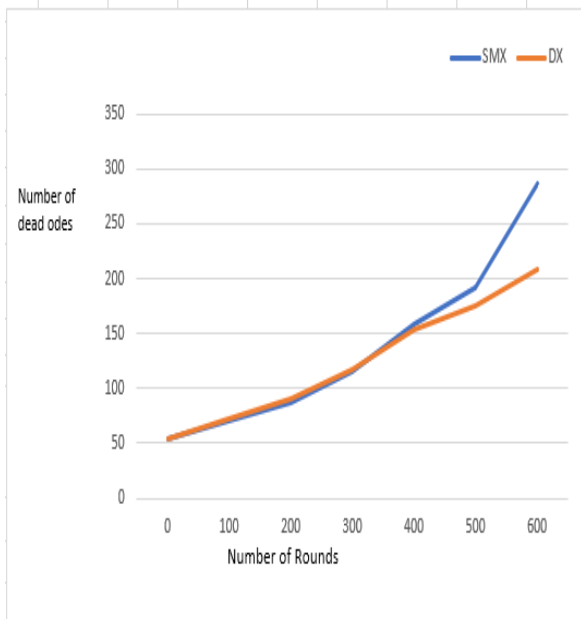


Fig.3. Comparison of dead nodes in each round without fitness scaling ($E_0 = 0.50$)

Fig.4. Comparison of network lifetimes with fitness scaling ($E_0 = 0.40$)

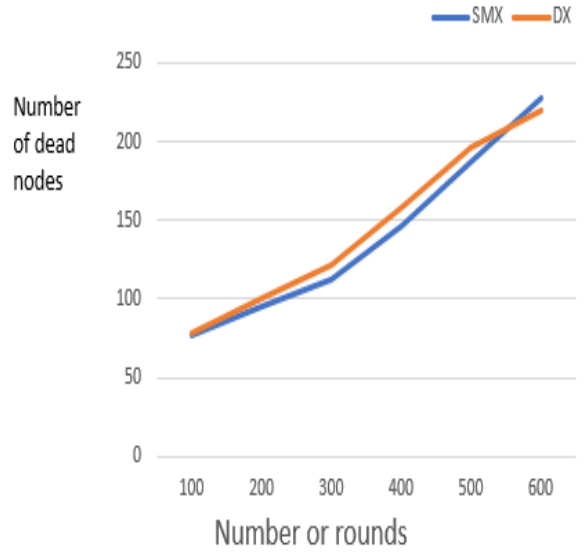
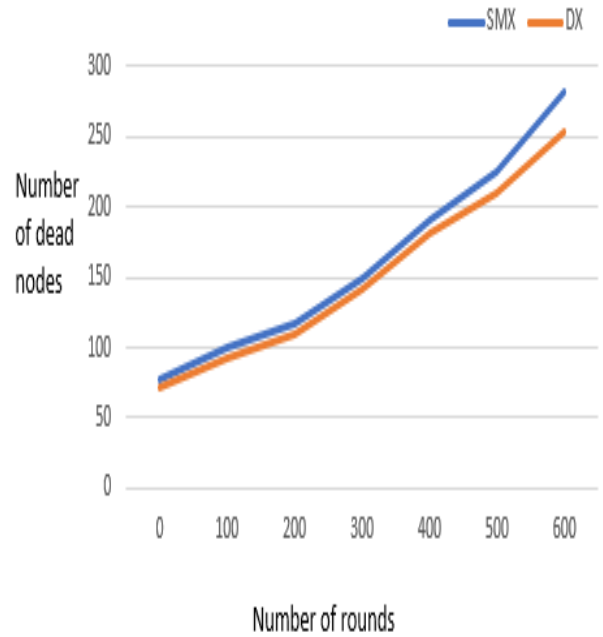


Fig.5. Comparison of dead nodes in each round with fitness scaling ($E_0 = 0.50$)



5. Conclusion

The main objective of this work was to find a way to prolong the lifetime of the WSN. Hence, we developed an energy-efficient clustering algorithm for the WSN that is based on GA (CCA-EGA).

For handling the clustering and CH selection problem, several decisions have been made to improve network lifetime. Thus, for optimizing network lifetime, two crossover operators were used and analyzed to assess their effects on network lifetime.

In view of the study findings, new crossover operators with will be investigated to obtain better WSN performance in terms of energy efficiency.

References

- [1] Lara, R., Benítez, D., Caamaño, A., Zennaro, M and Rojo-Alvarez, L. (2015) 'On real-time performance evaluation of volcano-monitoring systems with wireless sensor networks', *IEEE Sensors Journal*, 15(6), pp. 3514–3523. doi: 10.1109.
- [2] Jangam, K. B., S. V., Sunkara, C., Talluri, B. M., Shaik, H. Z. and Shaik, I. (2016) 'Wireless Sensor Network Design for Tactical Military Applications', *International Journal of Applied Sciences, Engineering and Management*, 5(2), pp. 5–8.
- [3] Elayan, H., Shubair, R. M. and Kiourti, A. (2017) 'Wireless sensors for medical applications: Current status and future challenges', in *2017 11th European Conference on Antennas and Propagation (EUCAP)*, pp. 2478–2482. doi: 10.23919/EuCAP.2017.7928405.
- [4] Sandhya, R. and Sengottaiyan, N. (2015) 'A Comparative Study on Sensor based Intelligent Rush-Hour Traffic Mechanism', 5(3), pp. 216–222.
- [5] Banerjee, R. and Bhattacharyya, C. K. (2014) 'Cluster based routing algorithm with evenly load distribution for large scale networks', *Computer Communication and Informatics (ICCCI)*, 2014 *International Conference on*, (1), pp. 1–6. doi: 10.1109/ICCCI.2014.6921779.
- [6] Bayrakli, S. and Erdogan, S. Z. (2012) 'Genetic Algorithm Based Energy Efficient Clusters (GABEEC) in Wireless Sensor Networks', *Procedia Computer Science*, 10, pp. 247–254. doi: 10.1016/j.procs.2012.06.034.
- [7] Das, S., Barani, S., Wagh, S. and Sanavane, S. (2015) 'An Exhaustive Survey on Nature Inspired MetaHeuristic Algorithm for Energy Optimization in Wireless Sensor Network', *ICTAT Journal on Communication Technology*, 6(4), pp. 1173–1181. doi: 10.21917/ijct.2015.0172.
- [8] Nimbalkar, N. B. and Das, S. S. (2014) 'Genetic Algorithm based Energy Aware Clustering Techniques: A Survey', *International Journal of Computer Applications*, 95(10), pp. 11–16.
- [9] Abo-Zahhad, M., Ahmed, S., Sabor, N. and Sasaki, S. (2014) 'A New Energy-Efficient Adaptive Clustering Protocol Based on Genetic Algorithm for Improving the Lifetime and the Stable Period of Wireless Sensor Networks', *International Journal of Energy, Information and Communications*, 5(3), pp. 47–72. doi: 10.14257/ijeic.2014.5.3.05.
- [10] Singh, A., Babbar, K. and Malik, A. (2015) 'Clustering and energy efficient routing protocol for wireless sensor network using genetic algorithm', *International Journal of Computer Applications*, 119(7), pp. 40–43.
- [11] Pal, V., Yogita., Singh, G. and Yadav. P. G. (2015) 'Cluster Head Selection Optimization Based on Genetic Algorithm to Prolong Lifetime of Wireless Sensor Networks', *Procedia Computer Science*, Elsevier Masson SAS, 57(Icrtc), pp. 1417–1423. doi: 10.1016/j.procs.2015.07.461.
- [12] Sharma, T., Tomar. S., Taneja, S. and Agrawal, K. (2015) 'Optimized Genetic Algorithm (OGA) for Homogeneous WSNs', *International Journal of Future Generation Communication and Networking*, 8(4), pp. 131–140.
- [13] Bhatia, T., Kansal, S., Goel, S. and Verma, A.K. (2016) 'A genetic algorithm based distance-aware routing protocol for wireless sensor networks', *Computers and Electrical Engineering*. Elsevier Ltd, 56, pp. 1–15. doi: 10.1016/j.compeleceng.2016.09.016.
- [14] Gbolagade, M. D., Jimoh, R. G., Gbolagde, K. A. (2017) 'Design of Hybridized Wireless Sensor Network using K-Means Clustering and Genetic Algorithm', 2(5), pp. 1–6.
- [15] Kim, J., Kim, Y., Yoo, Y. (2019) 'Energy-Efficient Routing Using Genetic Algorithm in Cluster-Based Wireless Sensor Networks', 2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall), 2019.
- [16] Kalyani, P. and Ramashri, D. T. (2017) 'A Comparative Study of Genetic Algorithm and the Particle Swarm Optimization', *International Journal of Innovative Technologies* 5(2), pp. 0318–0320.
- [17] Valarmathi, K., Devaraj, D. and Radhakrishnan, T. K. (2009) 'Real-coded genetic algorithm for system identification and controller tuning', *Applied Mathematical Modelling*. Elsevier Inc., 33(8), pp. 3392–3401. doi: 10.1016/j.apm.2008.11.006.
- [18] Oyama, A., Obayashi, S. and Nakahashi, K. (2000) 'Real-Coded Adaptive Range Genetic algorithm and its application to Aerodynamic Design', *JSME International Journal Series A Solid Mechanics and Material Engineering*, 43(2), pp. 124–129.
- [19] Anand, S., Afreen, N. and Yazdani, S. (2015) 'A Novel and Efficient Selection Method in Genetic Algorithm', *International Journal of Computer Applications*, 129(15), pp. 7–12.
- [20] Dey, D. K. (2014) 'Mathematical study of adaptive genetic algorithm (AGA) with mutation and crossover probabilities', *COMPUSOFT, An International Journal of Advanced Computer Technology*, 3(5), pp. 765–768.

- [21] Heinzelman, W. R., Chandrakasan, A. and Balakrishnan, H. (2000) 'Energy-efficient communication protocol for wireless microsensor networks', in Proceedings of the Hawaii International Conference on System Sciences. IEEE, pp. 1–10.

- [22] Pantazis, N. A., Nikolidakis, S. A. and Vergados, D. D. (2013) 'Energy-Efficient Routing Protocols in Wireless Sensor Networks: A survey', IEEE Communications Surveys & Tutorials, 15(2), pp. 551–591. doi: 10.1109/SURV.2012.062612.00084.