

Iterated Local Search Algorithm for Clustering Wireless Sensor Networks

Muyiwa Olakanmi Oladimeji

School of Engineering

London South Bank University

103 Borough Road, London SE1 0AA

oladimm2@lsbu.ac.uk

Mikdam Turkey

School of Engineering

London South Bank University

103 Borough Road, London SE1 0AA

turkeym@lsbu.ac.uk

Sandra Dudley, MIEEE

School of Engineering

London South Bank University

103 Borough Road, London SE1 0AA

dudleyms@lsbu.ac.uk

Abstract—In this paper, a new clustering protocol employing an iterated local search (ILS) to solve cluster head selection problem is proposed. ILS uses a perturbation operator to change an initial random solution to produce a new point in the vicinity of the solution. Using a combination operator, this new point is mated with the random solution producing a new solution. A move from the current solution to the new solution is considered acceptable only for higher fitness value. If a move is rejected after a predetermined search length, the change rate of the current solution is increased in order to explore a wider search space for quality solutions. In each round, this search process continues until good solution that ensures balanced energy consumption is obtained for the network. Furthermore, we propose a sleep scheduling scheme inspired by the Boltzmann Selection process in genetic algorithms. This mechanism stochastically considers coverage effect in the selection of nodes that are required to go into sleep mode in order to conserve energy of sensor nodes. The proposed mechanism of inactive node and cluster head selection protocols are performed sequentially at every round and they form part of the main algorithm proposed, namely the Dynamic Local Search-Based Algorithm for Clustering Hierarchy (DLSACH). The ultimate goal of the DLSACH protocol is to extend the network lifetime of wireless sensor networks by reducing and balancing the energy consumption among sensor nodes during communication processes. Our protocol shows an improved performance compared to state-of-the-art protocols such as LEACH, TCAC and SEECH in terms of improved network lifetime for wireless sensor networks deployment.

Keywords—Combinatorial Optimization, Heuristic, Local Search Operators, Wireless Sensor Networks.

I. INTRODUCTION

Recent progress in Micro-electromechanical Systems (MEMS) has enabled the development of self-configurable and spatially distributed autonomous sensors [1]. These sensor nodes can be networked and deployed randomly in remote and inaccessible areas, hence producing wireless sensor networks (WSNs). In large areas, WSNs are used for gathering data from the sensor field and transmitting data to a distant sink. The potential applications of WSNs are environmental monitoring, target field imaging, weather monitoring, event detection, security and battlefield surveillance etc. WSNs are characterized with severe energy, computation and memory constraints. Energy efficiency and network lifetime are major issues that require consideration in the design of protocols for WSNs. The network lifetime of WSNs can be defined as the

time interval between the start of network deployment and when the network begins to malfunction or become unstable. This term is application-dependent in the sense that in some applications the network becomes unstable when the first node dies or when a certain number of nodes die [2]. The energy constraint can be tackled by employing innovative design techniques that can efficiently manage the energy resource of sensor networks in order to extend the network lifetime of sensors.

The above challenges are often tackled using different routing protocols, which are classified into two broad categories namely flat and hierarchical networks. The two classical protocols under flat network is the Direct Transmission (DT) and Minimum Transmission Energy (MTE). The downside is that these protocols cannot guarantee a balanced energy distribution among sensor nodes in WSNs. In the DT protocol, each sensor communicates directly with the sink, thereby causing distant sensors to die first. Also for the MTE protocol, distant sensors communicate with the sink via a relay node but the drawback is that the relay node dies first due to overloading. Clustering is an efficient scheme under the hierarchical networks that helps in reducing energy consumption. Generally, clustering protocols can outperform flat protocols in balancing energy consumption and network lifetime by adopting data aggregation mechanisms. In the clustering protocol, the sensor nodes are grouped into clusters. Each cluster has a leader commonly referred to as the cluster head (CH), and this CH can be pre-assigned by the network designer or dynamically assigned by the WSNs protocols. Each sensor node that belongs to the CH is referred to as a member and its task is to sense and transmit data to the CH. Each CH aggregates the received data and sends it to the sink for processing [3], [4].

In the design of cluster based protocols, adopting a good CH configuration that can balance energy distribution is paramount to extend network lifetime. Generally, the decision of CH selection can either be done at the node itself or at the sink node. CH selection is preferably done at the sink because this node has enough energy and can perform complex calculations. In this work, the CH selection issue is viewed as an optimisation issue where we have used the local search, perturbation and combination operators to solve the problem.

These operators are applied on a set of random solutions in order to obtain the best solution within the search space. The quality of the solutions are accessed using a defined objective function. In this paper, a Dynamic Local Search-Based Algorithm for Clustering Hierarchy (DLSACH) protocol was developed for clustering WSNs. This protocol performs two major operations; (1.) the use of an iterated local search algorithm for cluster head selection (ILSACHS) protocol to select the best cluster configuration and (2.) our proposed Stochastic Selection of Inactive Nodes (SSIN) mechanism in order to send some nodes that have negligible effect on the WSNs coverage to sleep or inactive mode.

In our proposed ILSACHS protocol, the sink performs the CH selection task on active nodes using the iterated local search with perturbation operator. The action of perturbation operator resembles the traditional mutation operator, and is applied to a random solution at a specific change rate in order to generate a perturbed point in the neighbouring area of the random solution [5]. The perturbed point is mated with a random solution and a new solution is produced using the combination operator. The fitness of the new solution is evaluated by the objective function. A move to the new solution from the current solution is accepted if the fitness value of the new solution is greater than the current solution, otherwise the move is rejected. The total number of search lengths for a good solution is divided into four step sizes. In the case where a move to a new solution is rejected after the last attempt of each step size division, the change rate is increased to explore wider search space.

At each network operation round, the search continues until a move to a new solution is accepted. This new solution is applied to the WSNs and is expected to contain distributed CHs that balances energy consumption across the networks. The SSIN, a mechanism that mimics the Boltzmann selection process in genetic algorithm (GA) was employed to reduce the number of active nodes at the beginning of each network operation round by sending some nodes to sleep or into inactive mode to conserve energy and prolong network lifetime with minor effects on coverage. Both mechanism works collaboratively to maximize network lifetime by balancing the energy consumption among sensor nodes during communication processes. The balance in energy consumption is achieved by selecting spatially distributed nodes with higher energy as CHs and also sending some nodes to sleep mode without causing an adverse effect on the coverage. Our proposed DLSACH protocol is a more energy efficient protocol compared with other protocols.

The remainder of this study is organised as follows. Section II discusses related works on energy-efficient clustering approaches for wireless sensor networks. Section III describes the network and energy dissipation model underlying our protocol. Section IV describes the objective function for the proposed protocol. Section V describes the proposed sleep scheduling mechanism, clustering algorithm and energy consumption computation. Section VI discusses the experimental settings, performance measures, result and discussion. Finally,

a conclusion is presented in Section VII.

II. RELATED WORK

In recent times, much research work has been tailored towards energy efficient clustering protocols for WSNs [6], [7]. In the Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol [4], the energy load distribution is equally balanced among the CHs. At each round, the LEACH's operation is divided into the set-up and steady-state phase. During the set-up phases, each node makes decision of whether to become a CH or not. The CH selection is done probabilistically and the selected CH node broadcasts an advertisement message to the other nodes. Other nodes choose their CH based on the signal strength of the advertisement message and they reply back with a membership message. During the steady-state phase, the CH aggregates the received data from its members and transmits this data directly to the sink. After a certain round length, the network begins again at the set-up phase and enters into another round of CH selection. The drawback with the LEACH protocol is that the CH will die faster compared with other nodes if the distance between the CH and sink is too great.

Another variant of LEACH protocol was proposed in [4], which is called LEACH-centralized (LEACH-C). Unlike LEACH, LEACH-C employs the sink to perform the task of CH selection and formation. Each node sends their location and energy level to the sink. The sink employs a simulated annealing (SA) approach to determine the CH number and cluster configuration based on the received information. The energy and distance between CHs and non-CHs are considered for even load and cluster distribution. The sink optimizes global knowledge of the network to produce an improved network that requires less energy. However, it assumes that the CHs can send aggregated data streams directly to the sink which is a similar drawback to LEACH. A hybrid energy efficient distributive (HEED) protocol was proposed in [8]. CH selection is achieved by iteratively considering the residual energy and the proximity to member nodes. In this protocol, the energy consumption for communicating between the CHs and non-CHs is reduced considerably and each CH communicates with the sink using multi-communication approach. However, more CHs are generated than the expected number and this results in an unbalanced energy consumption. Also, HEED results into overhead since it does several iteration to select CHs.

The Topology-Controlled Adaptive Clustering (TCAC) protocol [9] provides a fair distribution of CHs. In this protocol, a large number of nodes consider themselves candidate CHs and inform this decision to other nodes. Each of the candidate CHs check the other CHs to see if their residual energy is greater than their own. If no other CH residual energy is greater than their own, it will consider itself the CH. The non-CH selects the CH with minimum cost distance between itself and corresponding CH node and the CH node distance to the sink. The TCAC balances the size of the cluster and sends the data directly to the sink via the CHs. In the scalable

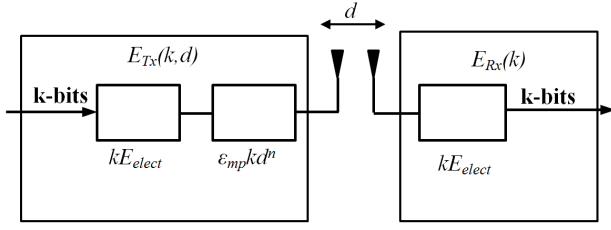


Fig. 1. Radio Energy Dissipation Model

energy efficient clustering hierarchy (SEECH) protocol [10], network nodes are divided into three layers namely member nodes, CH nodes and relays. Cluster formation is based on the centrality of the CH node with minimum intra-cluster energy distribution. The node that is closer to the sink in a cluster is selected as the relay node. The relay node assists the CH to send aggregated data to the sink using a hop or multi-hop communication approach.

III. NETWORK AND ENERGY DISSIPATION MODEL

To simplify the network model, a few reasonable assumptions have been adopted:

- All sensors are uniformly dispersed within a square sensor field. The sink device is placed a far distance from the sensor field.
- All sensors and sinks are stationary after deployment.
- All sensors have GPS or other location determination capabilities.
- Nodes located close to each other have correlated data.
- Nodes are capable of acting in inactive mode or a low power sleeping mode.
- The communication channel is symmetric (i.e. the energy required to transmit a message from sensor node s1 to sensor node s2 is equal to the energy required to transmit from node s2 to node s1 for a given signal to noise ratio).

In this paper, we employed a simplified model used in previous protocols [4], [11] for the radio hardware energy dissipation as shown in Figure 1. To transmit a k -bit message a distance d , the radio expends:

$$E_{Tx}(k, d) = \begin{cases} kE_{elect} + \varepsilon_{mp}kd^4, & \text{if } d > d_0 \\ kE_{elect} + \varepsilon_{fs}kd^2, & \text{if } d < d_0 \end{cases} \quad (1)$$

In Equation 1 above, the first term represents the energy that the transmitter dissipates to run the radio electronics while the second term represents the energy dissipation for signal amplification. The energy consumed for amplification depends on the distance between the transmitter and receiver. Both the free space (d^2 power loss) and the multipath fading (d^4 power loss) channel models were used. The power amplifier is appropriately set in such a way that if the distance is less than a given threshold, the free space (fs) model is used; otherwise, the multipath (mp) model is used.

The receiver also dissipates energy to run the radio electronics and to receive this message, the radio expends:

$$E_{Rx}(k) = kE_{elect} \quad (2)$$

where the $d_0 = \sqrt{\varepsilon_{fs}/\varepsilon_{mp}}$ denotes the threshold distance and the electronics energy, E_{elect} depends on factors such as the digital coding, modulation, filtering, and spreading of the signal, whereas the amplifier energy, ε_{mp} or ε_{fs} depends on the distance to the receiver and the acceptable bit-error rate).

IV. OBJECTIVE FUNCTION

Given a search space S for an instance ϕ of a problem Φ , an objective function F associated to each element $s \in S$ a value $F(s)$ is used to assess the quality of the solution. In optimisation problems, an objective function is employed to guide the search toward better solutions within the search space S and to select the appropriate step at each point of the search. In order to solve the CH selection problems, there is a need to develop an objective function since in the case presented here, CH selection is considered an optimisation issue. This objective function returns a fitness value which is used to assess the quality of a proposed solution. The objective function is derived by considering the weighted sum of the ratio of average energy of non-CH and CH and the Risk penalty R . The Risk penalty, R for CH selection is defined as:

$$R = \begin{cases} Lower - L, & \text{if } L < Lower \\ L - Upper, & \text{if } L > Upper \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where the $Lower$ and $Upper$ are calculated as the 4% and 6% of the total number all sensor nodes in the field (n) respectively. The risk penalty R imposes restrictions on the number of CHs (L).

Therefore, the objective function is computed as:

$$F(s) = 0.9 * \frac{AvgENCH}{AvgECH} + 0.1 * R \quad (4)$$

Where $\frac{AvgENCH}{AvgECH}$ is the average energy-ratio of non-CHs to CHs and R is the risk penalty with a weighting factor of 0.9 and 0.1 respectively. Higher weighting factor is assigned to the average energy ratio than the risk penalty because much emphasis is placed upon it. The average energy of the non-CHs set is defined as $AvgENCH = \frac{\sum_{i \in NCH} E_i}{n - L}$ and the CHs set is defined as $AvgECH = \frac{\sum_{i \in CH} E_i}{L}$.

V. THE PROPOSED DLSACH PROTOCOL

In this protocol, the sleep scheduling, clustering and energy consumption computations are performed in succession. We propose the stochastic selection of inactive node (SSIN), a sleep scheduling scheme is used to put some sensor nodes into sleep mode without harming the functionality of WSNs in terms of network coverage. The sleeping scheduling is performed at the beginning of every round during network

Algorithm 1 DLSACH Protocol

Let *AliveNodes* be the total number of sensor nodes
Compute the network total coverage.
while (*AliveNodes* > 0) **do**
 Use algorithm SSIN to select inactive nodes. (See Algorithm 2)
 Put selected nodes into sleep mode.
 Apply the proposed iterated local search algorithm for CHs selection. (See Algorithm 3)
 Compute the energy values of E_{CH} , E_{Mem} and E_{Res} . (refer to Section V-C2)
 Find out the number of dead nodes (node with energy equal or less than 0).
 Update *AliveNodes*
end while

operations. In the clustering process, our proposed ILSACHS protocol works in such a way that a local solution with a known fitness value is obtained from a set of random solutions. The perturbation operator mutates the selected local solution at a specified mutation rate in order to obtain a point. Using the combination operator, this point is mated with the local solution to produce a new local solution. The fitness value of the new local solution is obtained by evaluating the objective functions and compared with the previous solution. If the fitness value of the new solution is greater than the previous one, a move to the new local solution is accepted otherwise the new solution is discarded. The cycle continues until the moves reach the local optimal solution within the specified search length. In the case where no moves are accepted after a certain search length, the mutation rate of the perturbation operator is increased according to defined step sizes to widen the search length. At each network operation round, the final solution obtained in this protocol is expected to minimize the energy consumption due to well distributed CHs in the network. The energy consumption computation is performed at each network operation round as it moves from the set-up to steady state phase. At the setup phase, the sink transmits control packets to receive node information in terms of the nodes ID, location and energy. The residual energy of each sensor is computed at the end of each round of the steady state of network operation. The value of the residual energy computed in the current round is used for the next round as a parameter for sleep scheduling and cluster head selection process. This cycles continues until all nodes are dead; as shown in Algorithm 1.

A. Inactive Node Selection Using SSIN

In dense WSNs, it is common to put some nodes to sleep and use a set of active sensors for sensing and communication [12]. The SSIN makes decisions based on which nodes to put into inactive mode at the beginning of each network operation round. The candidate list of sleeping nodes is grown by checking the residual energy of nodes that are less than the computed average energy. This inactive node selection mechanism is synonymous to the Boltzmann selection process

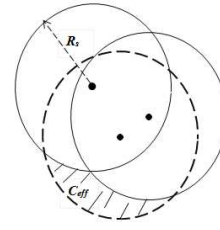


Fig. 2. Illustration of Nodes to Sleep on Coverage Area

whereby a method is adopted to control the selection pressure [13]. In Boltzmann selection, the temperature parameter is varied in order to control the selection pressure. In this work, the maximum coverage effect, Max_{eff} is used to control the effect of putting nodes to sleep within WSNs, and is defined as:

$$Max_{eff} = 2 \times \pi \times R_s^2 \quad (5)$$

where R_s is the sensing range of a sensor node (taking the coverage area as a circle with radius R_s), $(\pi \times R_s^2)$ signifies the coverage of one node and the value '2' represents coverage of two nodes.

The effect of putting a node to sleep based on the coverage is defined by the coverage effect C_{eff} as shown in Figure 2. The total coverage effect is calculated by capturing a matrix called the Coverage Matrix. The coverage matrix captures the overlapped areas of nodes coverage which allows the identification of nodes that can be sent into sleep mode without affecting the coverage since there will be other nodes covering part of the selected node's area. The accumulated Coverage effect Acc_{eff} is defined as the total effect on the coverage as a result of allowing some nodes to sleep. Our algorithm is designed so that the Acc_{eff} value is expected to be less than the Max_{eff} for optimum coverage ($Acc_{eff} < Max_{eff}$). The probability that a node will be added to the sleeping node list can be computed using:

$$P = e^{(-C_{eff}/Max_{eff})/(1-(Acc_{eff}/Max_{eff}))^2} \quad (6)$$

Algorithm 2 Proposed SSIN protocol

$Acc_{eff} = 0$;
Compute the residual energy, E_{Res} of each node. (refer to section V-C2)
Compute the average energy of all nodes, E_{Avg} .
Generate a candidate list for nodes with $E_{Res} < E_{Avg}$.
Compute Max_{eff} . (refer to equation 5)
while ($Acc_{eff} < Max_{eff}$) **do**
 Compute probability, P of adding nodes to the sleeping list. (See equation 6)
 if ($rand() < P$) **then**
 Create list of sleeping node from the candidate list.
 Compute the coverage effect, C_{eff} .
 $Acc_{eff} = Acc_{eff} + C_{eff}$
 end if
end while

A randomly generated number is compared with the computed probability, P . A candidate list of inactive nodes is created if the random number is less than the probability, P . The accumulated frequency Acc_{eff} is computed by adding its current value to coverage effect C_{eff} value. The operations of SSIN continues until the Acc_{eff} is greater than the maximum acceptable coverage effect, Max_{eff} as shown in Algorithm 2.

B. Proposed ILSACHS protocol

Clustering is an efficient way in which a WSN can balance its load, save energy and enhance the network lifetime. It is the grouping of sensor nodes into clusters and CH selection for all the clusters. The CH plays a vital role of gathering data from its associated nodes and forwarding the aggregated data to the sink for processing. In our proposed ILSACHS protocol, the perturbation operator is used to generate a new starting point for further local searches for the local optimal solution [5]. One of the major contributions of this paper is that the mutation rate of the perturbation operator changes dynamically according to a predetermined step size in order to search outside the local optimum. Using the combination operator, the local optimal solution is mated with the local solution in the same neighboring area, and a new solution is produced (See Algorithm 3).

1) *Iterated Local Search with Perturbation Operator*: In the proposed ILSACHS protocol, the iterated local Search algorithm improves a solution in the search space by starting from an initial random solution $s^* \in S^*$, and iteratively explores the search space for a local optimal solution. The fitness value $F(s^*)$ of the solution s^* is accessed using our proposed objective function in Equation 4. At the first step, the local solution s^* is mutated by a perturbation operator to generate an intermediate solution or point s' . The current solution s^* is combined with point s' , and a new solution $s^{*'}$ (See Section V-B2) is produced with fitness value $F(s^{*'})$. If the fitness value $F(s^{*'})$ is greater than $F(s^*)$, a move from solution s^* to $s^{*'}$ is accepted i.e. $s^{*'}$ replaces s^* . If the condition is not satisfied, solution $s^{*'}$ is discarded i.e. the current solution s^* remains unchanged.

In the case where local search failed to move to a new solution after some consecutive number of search attempts within the predetermined search length, the mutation rate of the perturbation operator is increased. This action of perturbation operators is similar to the term *kickers* used in special purpose local searches for intensification or diversification. It allows the search to escape from the attraction area of a local minimum [14]. The search length is defined as the total number of search attempts for our iterated local search operator. As shown in Figure 3, the search length is divided into four step sizes which denotes the number of search attempts before the mutation rate is increased to widen the search area.

2) *Combination Operator*: The point s' obtained by the perturbation operator is combined with the local solution s^* in order to obtain a new solution using a combination operator that uses heuristic crossover. The pioneer of this heuristic crossover operator is Lixin Tang [15], and proposed

Algorithm 3 Proposed ILSACHS protocol

```

i ← 0, searchlength = 100;
Generate a initial solution,  $s^*$ .
Compute the fitness value,  $F(s^*)$ .
while ( $i \leq searchlength$ ) do
    Mutate  $s^*$  to obtain a point  $s'$  using the perturbation
    operator (See Section V-B1 for details).
    Combine the solution  $s^*$  with the point  $s'$  to obtain a
    new solution  $s^{*'}$ . (refer to Algorithm 4)
    Compute the fitness value,  $F(s^{*'})$ .
    if ( $F(s^{*'}) < F(s^*)$ ) then
        Replace solution  $s^*$  with the new solution  $s^{*'}$ .
    else
         $i = i + 1$ 
    end if
end while

```

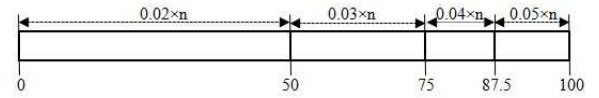


Fig. 3. Step Size division of Search Length

to utilize parents' implicit information to produce offspring. In the canonical crossover approach, parents mate to produce pairs of offspring that tend to substitute their parents with no guarantee that an offspring produced would be better than either of its mating parent [16]. The heuristic combination approach uses a special crossover that has a knowledge of a problem to combine two candidate solutions to produce an improved solution.

The solution produced by this heuristic combination operator represents CH configurations that are well distributed across the sensing field and favors those with higher energy. Our proposed crossover prohibits the selection of two CHs within the same region and higher priority is given to a CH with higher energy. The local solution $s^{*'}$ and perturb point s^* are selected from the iterative local search process and the CH position in the two solutions is computed and stored into the set r_a and r_b respectively. A threshold distance T is defined between two neighboring CH positions as; $T = \frac{\sqrt{(x_{max}-x_{min})^2+(y_{max}-y_{min})^2}}{n \times 0.04}$, where the (x_{min}, y_{min}) and (x_{max}, y_{max}) are minimum and maximum xy-coordinates of the sensor fields respectively, $(n \times 0.04)$ represent 4% of the total number of sensor nodes. The union of r_a and r_b is represented by $r_{a,b} = r_a \cup r_b$. By default, the first CH position $r_{a,b}(1)$ in the set $r_{a,b}$ is transferred to a newly created set r_{new} . Each subsequent CH position in the $r_{a,b}$ is compared with the r_{new} array set in order to make certain decisions which is based on distance between the CHs and their residual energy.

C. Energy Consumption Computation

Energy consumption at each round can be explained using two phases in the algorithm such as the setup phase and the

Algorithm 4 Proposed Combination Operator

Select the two solutions $s^{*'}$ and s^* .

Compute and store the CH position in solution $s^{*'}$ and s^* into set r_a and r_b respectively.

Compute the threshold distance, T (refer to Section V-B2)

Compute the union set $r_{a,b}$. (refer to Section V-B2)

Obtain the first cluster position $r_{a,b}(1)$ in the set $r_{a,b}$.

Create a new set r_{new} and transfer the $r_{a,b}(1)$ to it.

Compute the distance, D between CH positions in the sets r_a and r_b .

while ($D < T$) **do**

if (Energy in $CH_{a,b}$ node is less than r_{new} node) **then**

Discard the cluster head node. (i.e. do not add to r_{new} set)

end if

Replace the cluster head node in the r_{new} set

end while

Add to the cluster head node in the set $r_{a,b}$ into the r_{new} set.

steady state phase.

1) *Setup Phase*: The setup phase involves the transmission and reception of control packets k_{CP} from the sink to all nodes to initialise inter- and intra-communications. In the setup phase, the sink sends a control packet containing a short message to wake up and request IDs, positions and energy levels of all sensor nodes in the sensor field. Similar to Equation 2, the energy $E_{Rx}(k_{CP})$ is spent by each sensor to receive the control packets from the sink. All sensors report their IDs, positions and energy levels back to the sink and the transmitted energy $E_{Tx}(k_{CP}, d)$ consumed for the task is similar to Equation 1. The control packet received from all sensor nodes is processed by sink to make the following vital decisions; which nodes to keep active, CH selection, and the associated CH membership. Also, considered the energy $E_{Rx}(k_{CP})$ dissipated in receiving the membership status information from the sink. Elected CHs are required to transmit a TDMA schedules to their respective members and the energy dissipated to perform the task is computed using:

$$E_{Tx(ch_i)}(k_{CP}, d_{i-toMem}) = \sum_{i=1} ch_i \times E_{Tx}(k_{CP}, d_{i-toMem}) \quad (7)$$

Additionally, each member node also spends energy $E_{Rx}(k_{CP})$ to receive TDMA schedule from the CHs

2) *Steady Phase*: During the steady phase, the active sensor nodes begin sending data packets k . Each node sends the sensed data to its CH according to the TDMA schedule received. The CH node receiver must always be ready to receive packets from its nodes within its cluster. Data aggregation is performed on all received data at the CHs and all data are converted into a single data stream. This aggregated data stream is transmitted from the CHs to the sink. This process consumed some amount of energy by the sensor node transceiver as Equation 10. The total amount of energy spent

by all member nodes to transmit to their respective CHs is computed using:

$$E_{Tx(m_i)}(k) = \sum_{i=1} m_i \times E_{Tx}(k, d) \quad (8)$$

And the total amount of energy dissipated by CHs for receiving data packets from their member nodes is given as:

$$E_{Rx(m_i)}(k) = \sum_{i=1} m_i \times E_{Rx}(k) \quad (9)$$

Also, the energy dissipated by the CHs to aggregate the data received from all its members and itself can be calculated using:

$$E_{DA(m_i+1)}(k) = kE_{DA} * \left(\sum_{i=1} m_i + 1 \right) \quad (10)$$

Finally, the amount of energy spent by the CHs node for transmitting data packets to the sink is computed using:

$$E_{Tx(ch_i)}(k, d_{i-toSink}) = \sum_{i=1} ch_i \times E_{Tx}(k_{CP}, d_{i-toSink}) \quad (11)$$

Thus, the total energy consumed by all the CHs can be computed using:

$$\begin{aligned} E_{CHs} = & 2 * E_{Rx}(k_{CP}) + E_{Tx}(k_{CP}, d_{i-toSink}) + E_{Tx(ch_i)}(k_{CP}, d_{i-toMem}) \\ & + E_{Rx(m_1)}(k) + E_{DA(m_i+1)}(k) + E_{Tx(ch_i)}(k, d_{i-toSink}) \end{aligned} \quad (12)$$

And the energy dissipated by all the member nodes is computed as:

$$\begin{aligned} E_{Mem} = & E_{Tx}(k_{CP}, d_{i-toSink}) + E_{Tx}(k_{CP}, d_{i-toCH}) \\ & + 3 * E_{Rx}(k_{CP}) + E_{Tx(m_i)}(k) \end{aligned} \quad (13)$$

Therefore, the overall energy dissipated by all nodes is represented by $E_{TOTAL} = E_{CHs} + E_{Mem}$. Also, note that the residual energy of each node (either a CH or member node) at each round can be computed by subtracting the energy consumption from the current residual energy.

VI. EXPERIMENTAL RESULTS

In this work we evaluate the protocols from an energy efficiency perspective by examining the number of alive nodes versus rounds. The graphed results helps us to evaluate the lifetime of the sensor nodes using our proposed algorithm. The simulation models and programs are developed using the MATLAB tool. From our point of view the proposed technique is scalable and may lead to energy efficiency improvement in different network sizes. To assess this claim the performance of DLSACH is compared to three other protocols LEACH, TCAC and SEECH in five experiments; first three experiments are homogeneous networks and last two experiments are heterogeneous networks. TABLE I describes the parameter values for each experiment in details.

TABLE I
PARAMETER VALUES FOR EACH EXPERIMENT

Experiments	Parameter Settings		
	Sink Coordinates	Number of Sensors	Initial Energy
Experiment I	(50,175)	100	0.5J
Experiment II	(50,200)	400	0.5J
Experiment III	(50,50)	100	0.5J
Experiment IV	(50,50)	100	$\mu=0.5$ (All Nodes) $\sigma=0.05$
Experiment V	(50,50)	100	$\mu=0.5$ (75 Nodes) $\sigma=0.05$ 0.5J (25 Nodes)

A. Experimental Settings

The common communication parameters used for all the experiments are presented in Table I above with the sensor field having a dimension of 100×100 , electronic energy, $E_{elect} = 50$ nJ/bit, free space loss $\epsilon_{fs} = 10$ pJ/bit/m², multipath loss, $\epsilon_{mp} = 0.0013$ pJ/bit/m⁴, threshold distance (d_0)=87m, data aggregation energy $E_{DA} = 5$ nJ/bit/signal, packet size $k=4000$, and control packet size $k_{CP}=50$. (Note: μ and σ represent the mean and standard deviation of the sensor node energy distribution in Experiment IV & V).

B. Performance Measures

There are many metrics used to evaluate the performance of the clustering protocols [15]. These measures are used in this paper to evaluate the performance of DLSACH protocol:

- 1) **First Dead Node (FDN)**: This is the number of rounds at which the first node dies (FND). It can also be referred to as the operational lifetime or stability period of the network. Therefore, larger FND value signifies longer WSN stability periods.
- 2) **Last Dead Node (LDN)**: This is the number of rounds from the start of network operation until the last node dies (LND).
- 3) **Instability Period Length (IPL)**: The round difference between the round at which the last node dies and the first node dies (i.e. $IPL=LND-FND$).
- 4) **Average Energy at first node dies (AEFND)**: The average energy of all sensor nodes when the first node dies.

Clearly, the longer the stability period and the shorten the instability period are, the better the reliability of the clustering process of the WSN.

C. Results and Discussion

The average value of our performance measures are obtained from 100 simulation runs and presented in this section for analysis. For each simulation run, new sensor node are distributed in a sensor field area. Our proposed DLSACH protocol is compared with LEACH, TCAC and SEECH for a small and large scale network of 100 and 400 sensor nodes respectively and is shown in Figure 4, which depicts the number of alive nodes during simulation time versus the number of rounds.

TABLE II
COMPARISON OF LEACH, TCAC, SEECH AND DLSACH FOR FND,LND AND IPL

Experiments	Protocol	Performance Measure (Round)		
		FND	LND	IPL
Experiment I (100 Nodes)	LEACH	726	1209	483
	TCAC	933	1006	73
	SEECH	1028	1099	71
	DLSACH	1074	1166	92
Experiment II (400 Nodes)	LEACH	685	1274	589
	TCAC	948	1071	123
	SEECH	1016	1140	124
	DLSACH	1206	1280	74

TABLE III
PERFORMANCE MEASURES FOR EXPERIMENT III,IV AND V

Performance Measures	Experiments		
	Experiment III	Experiment IV	Experiment V
FND	1354	1246	1241
LND	1445	1336	1331
IPL	91	90	90
AEFND	0.014949	0.014518	0.016568

Also, the FND, LND and IPL values belonging to the graphs in Figure 4 are presented in Table II. The FND values presented in Table II show that the DLSACH protocol maintains the network operational lifetime of 46, 141 and 348 more than the SEECH, TCAC and LEACH respectively for Experiment I (100 nodes). In Experiment II, the FND value of DLSACH is higher compared with the other three protocols. This result of the experimentation shows that the energy of the sensor nodes is balanced and extend for a longer period. When the FND time is reached, most of the nodes begin to die due to

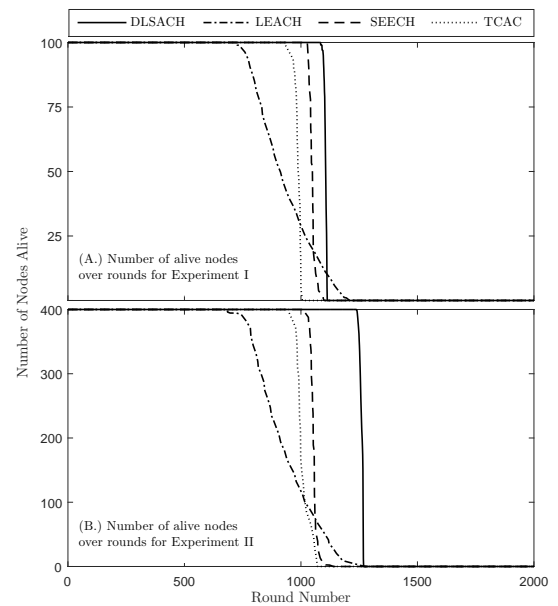


Fig. 4. Network Lifetime Comparison of DLSACH with LEACH, SEECH, TCAC

insufficient energy and this is represented by a sharp decline in the slope of TCAC, SEECH and DLSACH as shown in Figure 4.

For Experiment II, the instability period for our proposed DLSACH is 50, 49 and 515 less than the SEECH, TCAC and LEACH. This low value reveals that our protocol performs better in dense network. The results of three experiments which consist of one homogeneous (Experiment III) and two heterogeneous (Experiment IV & V) sensor networks is presented in Table III. Experiment III has a higher FND values of 1354 than Experiment IV & V, which are 1246 and 1241 respectively. This shows that the higher the complexity of the problem, the lower the FND value. Two interesting points from the three experiments is that we were able to keep the instability period almost constant for all the last three experiments and also keep the AEFND value very low number; which means we were able to use almost all of our initial energy till the death of first node

It can therefore be deduced that the proposed DLSACH protocol decreases energy consumption and optimizes energy balancing, thus increasing the network lifetime. At each round, our proposed DLSACH protocol conserves energy by selectively allowing some nodes to become inactive before network operation. Also, it uses an Iterated Local Search Algorithm to select the best CHs configuration ensuring that CHs are well distributed around the sensor field.

VII. CONCLUSION

In this paper, Dynamic Local Search-Based Algorithm for Clustering Hierarchy (DLSACH) protocol was proposed for energy management in wireless sensor networks. The two major operations in this protocol include sending some nodes into sleep mode and the cluster head selection. We proposed the Iterated Local Search Algorithm for Cluster Head Selection (ILSACHS), a mechanism that employs iterative local search with perturbation operator for cluster head selection. The perturbation operator mutates a random solution to produces a point within the local optimum. This point is combined with the selected random solution to produce a new solution using a heuristic based combination operator. A move to the new solution is accepted only at a higher fitness value, otherwise the move is discarded. After a complete search process, the new solution that guarantees optimally distributed cluster heads is applied to the network. Also, we employed a new mechanism namely the Stochastic Selection of Inactive Node (SSIN) inspired by Boltzmann selection process to stochastically select which nodes to send to sleep without adversely affecting coverage. The two proposed mechanisms work collaboratively to reduce and balance energy consumption by selecting well distributed nodes with higher energy as cluster heads in order to prolong network lifetime. Results shows that the network lifetime of DLSACH protocol is more than SEECH, TCAC and LEACH protocol; most especially within large network. Also, our DLSACH protocol shows good performance for heterogeneous sensor networks in terms of energy consumption and stability periods. In the future, we plan

to improve the energy efficiency under dense heterogeneous sensor networks.

REFERENCES

- [1] S. Naeimi, H. Ghafghazi, C.-O. Chow, and H. Ishii, "A survey on the taxonomy of cluster-based routing protocols for homogeneous wireless sensor networks," *Sensors*, vol. 12, no. 6, pp. 7350–7409, 2012.
- [2] A. Chakraborty, S. K. Mitra, and M. K. Naskar, "Energy efficient routing in wireless sensor networks: A genetic approach," *CoRR abs/1105.2090*, 2011.
- [3] M. A. Araghizadeh, N. Yazdani, S. Safari, and A. Hedayati, "Deed: Dynamic energy efficient distributed cluster forming algorithm for wireless sensors network," in *Telecommunications (IST), 2012 Sixth International Symposium on*. IEEE, 2012, pp. 776–781.
- [4] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *Wireless Communications, IEEE Transactions on*, vol. 1, no. 4, pp. 660–670, 2002.
- [5] Q. Zhang and J. Sun, "Iterated local search with guided mutation," in *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on*. IEEE, 2006, pp. 924–929.
- [6] L. YeMao, C. Fa, and W. Hai, "An energy efficient clustering scheme in wireless sensor networks," *Ad Hoc & Sensor Wireless Networks (to be published)*.
- [7] N. Dimokas, D. Katsaros, and Y. Manolopoulos, "Energy-efficient distributed clustering in wireless sensor networks," *Journal of parallel and Distributed Computing*, vol. 70, no. 4, pp. 371–383, 2010.
- [8] O. Younis and S. Fahmy, "Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *Mobile Computing, IEEE Transactions on*, vol. 3, no. 4, pp. 366–379, 2004.
- [9] D. P. Dahnil, Y. P. Singh, and C. K. Ho, "Topology-controlled adaptive clustering for uniformity and increased lifetime in wireless sensor networks," *IET Wireless Sensor Systems*, vol. 2, no. 4, pp. 318–327, 2012.
- [10] M. Tarhani, Y. S. Kaviani, and S. Siavoshi, "Seech: Scalable energy efficient clustering hierarchy protocol in wireless sensor networks," *Sensors Journal, IEEE*, vol. 14, no. 11, pp. 3944–3954, 2014.
- [11] K. G. Vijayvargiya and V. Shrivastava, "An amend implementation on leach protocol based on energy hierarchy," *International Journal of Current Engineering and Technology*, vol. 2, no. 4, pp. 427–431, 2012.
- [12] E. Bulut and I. Korpeoglu, "Sleep scheduling with expected common coverage in wireless sensor networks," *Wireless Networks*, vol. 17, no. 1, pp. 19–40, 2011.
- [13] D. Dumitrescu, B. Lazerzini, L. Jain, and A. Dumitrescu, *Evolutionary Computation*, ser. International Series on Computational Intelligence. Taylor & Francis, 2000.
- [14] L. Di Gaspero and A. Schaerf, "Multi-neighbourhood local search with application to course timetabling," in *Practice and theory of automated timetabling IV*. Springer, 2003, pp. 262–275.
- [15] T. Lixin, "Improved genetic algorithms for tsp," *JOURNAL OF NORTH-EASTERN UNIVERSITY (NATURAL SCIENCE)*, p. 01, 1999.
- [16] B. S. Hasan, M. Khamees, A. S. H. Mahmoud *et al.*, "A heuristic genetic algorithm for the single source shortest path problem," in *Computer Systems and Applications, 2007. AICCSA'07. IEEE/ACS International Conference on*, 2007, pp. 187–194.