Accepted Manuscript

Highlights

- Clustered wireless sensor networks are more energy efficient than direct transmission
- Swarm Intelligence helps Mobile sensor nodes to move in a swarm bases
- Particle Swarm Optimization is a powerful technique to control the autonomous movements of energy efficient sensor networks

ENERGY EFFICIENT ALGORITHM FOR SWARMED SENSORS NETWORKS

Mohaned Al. Obaidy [1] and **Aladdin Ayesh** [2]

Abstract. In this work we are presenting the design of an intelligent hybrid optimization algorithm which is based on Evolutionary Computation and Swarm Intelligence to increase the life time of mobile wireless sensor networks (WSNs). It is composed of two phases; Phase-1 is designed to divide the sensor nodes into independent clusters by using Genetic Algorithms (GAs) to minimise the overall communication distance between the sensor-nodes and the sink-point. This will decrease the energy consumption for the entire network. Phase-2 which is based on Particle Swarm Optimization (PSO) is designed to keep the optimum distribution of sensors while the mobile sensor network is directed as a swarm to achieve a given goal. One of the main strengths in the presented algorithm is that the number of clusters within the sensor network is not predefined, this gives more flexibility for the nodes' deployment in the sensor network. Another strength is that sensors' density is not necessary to be uniformly distributed among the clusters, since in some applications constraints, the sensors need to be deployed in different densities depending on the nature of the application domain. Although traditionally Wireless Sensor Network have been regarded as static sensor arrays used mainly for environmental monitoring, recently, its applications have undergone a paradigm shift from static to more dynamic environments, where nodes are attached to moving objects, people or animals. Applications that use WSNs in motion are broad, ranging from transport and logistics to animal monitoring, health care and military. These application domains have a number of characteristics that challenge the algorithmic design of WSNs.

1 INTRODUCTION

Recent advances in micro-electro-mechanical systems, digital electronics, and wireless communications have led to the emergence of wireless sensor networks (WSNs), which consist of a large number of sensing devices each capable of detecting, processing and transmitting environmental information. A single sensor node may only be equipped with limited computation and communication capabilities; however, nodes in a WSN, when properly configured, can collaboratively perform signal processing tasks to obtain information pertaining to remote and potentially dangerous areas in an untended and robust way. Applications for wireless sensors networks include battlefield surveillance, environmental monitoring, biological detection, smart spaces, industrial diagnostics, etc. [10]. Any WSN is deeply involved in and related to the monitored environment, and any change occurring to the surroundings will significantly influence its performance; nevertheless, the network must be able to tolerate and 'survive'any change by implementing proper reactions and adaptation

[1] Gulf College, OMAN, email: mohaned@gulfcollegeoman.com
[2] De Montfort University, UK, email: Aayesh@dmu.ac.uk

mechanisms sustaining communications for both sensed data and commands [39].

Energy efficiency has been deemed to be the main challenge for Wireless Sensor Networks. Generally, the power supply of a single sensor node relies on a battery with limited energy (e.g., an AAA battery). Changing or recharging a nodes' battery is very difficult, if not impossible, after sensor nodes have been deployed. Therefore; it is desirable to design energy efficient protocols to run on individual nodes, to ensure that the operation time of the deployed WSN is as long as possible. However, some classical information processing approaches do not consider the energy efficiency issue and require re-examination when applied in resource constrained WSNs. Geographically distributed nodes in a WSN may have different views of the physical phenomenon in the sensor field and thus their measurements may have some points of correlation. A well designed algorithm should also exploit this to accomplish the information processing task via collaboration between nodes. In this work we propose to design an algorithm for a large scale mobile sensors network. This algorithm should provide a robust and energy-efficient communication mechanism which enables the swarms of sensors to move while keeping optimum distances between the sensor nodes.

The rest of this paper will be structured into the follwing sections; Section 2 describes the background and motivation for our work. In section 3, we are explaining the first phase of our proposed algorithm by using GAs to cluster the Sensors Network into independent groups. Section 4 shows the second phase of the proposed algorithm where we use the PSO technique to enable the clusters which are produced in phase-1 to move as Swarms while keeping the optimum distances. In section 5 the implementation of the proposed algorithm is explained by showing some snapshots of the simulation program. Section 6 shows the results discussion as well as comments for the output graphs are presented here including a critical review. Finally in section 7 we concluded our work and its objectives with possible future development and enhancments.

2 BACKGROUND AND MOTIVATION

As the Internet has revolutionized our life by the uncomplicated exchange of various forms of information among a large number of users, Wireless Sensor Networks (WSNs) may, in the near future, be equally significant in providing information regarding physical phenomena of interest; ultimately leading to detection and control, and where relevant enabling us to construct more accurate models of the physical world. WSNs have gained tremendous importance in recent years because of its potential use in a wide variety of applications. This, along with the unique characteristics of these networks, has spurred a significant amount of research for coming with network protocols specifically tailored for sensor networks [1]. Wireless

sensor networks are developing quickly and have been widely used in both military and civilian applications such as; target tracking, surveillance, and security management. It can be also used for monitoring of microclimates and wildlife habitats, the structural integrity of bridges and buildings, building security, location of valuable assets (via sensors placed on these valuable assets), traffic control, and so on. However, realizing the full potential of wireless sensor networks poses myriad research challenges ranging from hardware and architectural issues, to programming languages and operating systems for sensor networks, to security concerns, to algorithms for sensor network deployment, operation and management.

Since a sensor is a small, lightweight, un-tethered, battery-powered device, it has limited energy [27]. Therefore, energy consumption is a critical issue in sensor networks. We are interested in sensor networks in which a large number of sensors are deployed to achieve a given goal. All data obtained by member sensors must be transmitted to a sink or data collector. The longer the communication distance, the more energy will be consumed during transmission [14]. Direct transmission networks are very straightforward to design but can be very power-consuming due to the long distances from sensors to the sink. Alternative designs that shorten or minimize the communication distances can extend network lifetimes. The use of clusters for transmitting data to a base station leverages the advantages of small transmit distances for most nodes, requiring only a few nodes to transmit far distances to the base station. Clustering means to partition the network into a number of independent clusters, each of which has a cluster-head that collects data from all nodes within its cluster [17, 20]. These cluster-heads then compress the data and send it directly to the sink. The output of GA clustering will be assumed to be the initial population for the Swarms which will represent the dynamic WSN at the later stage of the proposed algorithm. Deployment of mobile swarms can enhance the sensor network in many ways. Firstly, the swarm nodes have much higher hardware capabilities than the sensor nodes. They can provide detailed information of the intended area (e.g. the hot spot). Secondly, the wireless radios of the swarm nodes usually have much longer range and higher channel bandwidth, which can support high quality and delay sensitive multimedia streams. Thirdly, the swarms are mobile [18]. They can be easily directed to the hot spots. A limited number of mobile swarms can easily cover a large scale sensor network. The sensor network can be deployed to cover a very large field due to the low cost of sensor nodes.

2.1 Energy-Aware Wireless Sensor Networks

Nodes in a WSN are usually highly energy-constrained and expected to operate for long periods from limited on-board energy reserves. To permit this, nodes and the embedded software that they execute must have energy-aware operation. Energy efficiency has been of significant importance since WSNs were first conceived but, as certain applications have emerged and evolved [23], a real need for ultra-miniaturized long-life devices has re-emerged as a dominant requirement. Because of this, continued developments in energy-efficient operation are paramount, requiring major advances to be made in energy hardware, power management circuitry and energy-aware algorithms and protocols.

The energy components of a typical wireless sensor node are shown in Figure 1. Energy is provided to the node from an energy source, whether this is a form of energy harvesting from sources such as solar, vibration or wind, or a resource such as the mains supply or the manual provision and replacement of primary batteries. Energy

obtained from the energy source is buffered in an energy store; this is usually a battery or super capacitor. Finally, energy is used by the node's energy consumers; these are hardware components such as; the microcontroller, radio transceiver, sensors and peripherals.
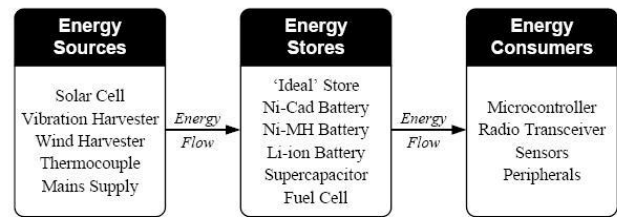

Figure 1. Energy components of a typical sensor node

With the increased usage of energy sources in nodes [50, 41], the need for energy stores other than batteries (many of which suffer from only offering a limited number of charging cycles) is increased. This can be seen in the researches that are now utilizing super capacitors (devices that are similar to standard electrolytic capacitors, but with capacities of many Farads) to store the node's energy [41, 22].

To be energy-aware, the embedded software executing on the node must be aware of the state of its energy components. This may be as advanced as monitoring the energy harvested from each source [44], inspecting the rate of consumption by different consumers [37], directing the flow of energy from and to different stores and managing the charging of rechargeable stores [22]. Alternatively, this may equate to simply being able to inspect the residual energy in a single store. Therefore, the embedded software must not only be capable of interfacing with energy hardware (this is generally a requirement of power management circuitry), but also interpreting the data that are obtained usually in the form of a sampled voltage into a remaining lifetime, power or energy. Based upon these values, the operation of the node is adjusted accordingly, usually to maximize the lifetime of the network.

2.2 GA Approach to Distance Optimization

In the past few decades, Genetic Algorithms have been used in science to derive solutions for many types of problems, from construction of wind turbines [3] to pattern-recognition systems [1]. Genetic Algorithm is an efficient search algorithm that simulates the adaptive evolution process of natural systems. It has been successfully applied to different problems such as multi-processor task scheduling, optimization, and traveling salesman problems [36].

Each individual in the GA population represents a possible solution to the problem. Finding individuals which are the best suggestions to our problem and combine these individuals into new individuals is an important stage of the evolutionary process. Using this method repeatedly, the population will hopefully evolve good solutions. Specifically, the elements of a GA are: selection (according to some measure of fitness), crossover (a method of reproduction, "mating" the individuals into new individuals), and mutation (adding a bit of random noise to the off-spring, changing their "genes").

Crossover and mutation provide exploration, compared with the exploitation provided by selection. The effectiveness of GA depends on the trade-off between exploitation and exploration. [36]

Crossover: The crossover operation takes place between two consecutive individuals with probability specified by crossover rate. These two individuals exchange portions that are separated by the crossover point. In this research we are using one-point crossover. The following is an example of crossover:

Indv1: 1 1 1 0 0 1 0 1
Indv2: 1 0 1 1 1 1 1 0
⇑
Cross over point

After crossover, two offspring are created as shown below:

Child1: 1 1 1 0 1 1 1 0
Child2: 1 0 1 1 0 1 0 1

If a regular node becomes a cluster-head after crossover, all other regular nodes should check if they are nearer to this new cluster-head. If so, they switch their membership to this new head. This new head is detached from its previous head. If a cluster-head becomes a regular node, all of its members must find new cluster-heads. Every node is either a cluster-head or a member of a cluster-head in the network.

Mutation: The mutation operator is applied to each bit of an individual with a probability of mutation rate. When applied, a bit whose value is 0 is mutated into 1 and vice versa. An example of mutation is shown below:

Indv: 1 1 1 1 1 1 0
⇓   ⇓
Indv: 1 1 1 0 1 1 1

In our research we are using GA algorithms to optimize the number of clusters and sensor connections for an arbitrary network. Algorithm (1) illustrate the basic process in GAs.

Initialization: Generate random population of n chromosomes
while *the stop condition is not satisfied* **do**
    Evaluate the fitness $g(x)$ of each chromosome x in the population;
    while *the new population is not complete* **do**
        Selection: Select two parent chromosomes from a population according to their fitness;
        Crossover: With a crossover probability, crossover the parents to form a new offspring (children);
        Mutation: With a mutation probability mutate new offspring;
        Accepting: Place new offspring in a new population;
    end
    Replace: Use new generated population for further runs;
end
Return: the best solution of the current population;

Algorithm 1: Basic Process in Genetic Algorithms

2.3 Swarm Intelligence

Swarm Intelligence (SI) indicates a recent computational and behavioural metaphor for solving distributed problems that originally took its inspiration from the biological examples provided by social insects (ants, termites, bees, wasps) and by swarming, flocking, herding behaviours in vertebrates [24, 15]. It is an attempt to design algorithms or distributed problem-solving devices inspired by the collective behaviour of social insects and other animal societies. The common behaviours in all kinds of swarms are [24, 6, 13];

- Control is fully distributed among a number of individuals;
- Communications among the individuals happen in a localised way;
- System-level behaviours appear to transcend the behavioural repertoire of the single individual; and
- The overall response of the system is quite robust and adaptive with respect to changes in the environment.

Swarm intelligence (SI) as defined by Bonabeau, Dorigo and Theraulaz is "any attempt to design algorithms or distributed problem-solving devices inspired by the collective behaviour of social insect colonies and other animal societies" [6]. The term "swarm" is used in a general sense to refer to any such loosely structured collection of interacting agents. The classic example of a swarm is a swarm of bees, but the metaphor of a swarm can be extended to other systems with a similar architecture. An ant colony can be thought of as a swarm whose individual agents are ants, a flock of birds is a swarm whose agents are birds, traffic is a swarm of cars, a crowd is a swarm of people, an immune system is a swarm of cells and molecules, and an economy is a swarm of economic agents. Although the notion of a swarm suggests an aspect of collective motion in space, as in the swarm of a flock of birds, all types of collective behaviour are considered here, not just spatial motion.

3 RELATED WORK

The use of Wireless Sensor Networks in the context of sustainable development and developing countries has been studied in many researches during the last decade [12]. Water quality is for example, a topic of interest for such technologies: system to monitor nitrate propagation through soils and ground water has been installed in California. Moving sensors using infrastructure-based robotics have already been implemented in order to collect data in relevant position of dynamical landscape [31]. Another use of sensor networks is search and rescue operations [16]. In such systems, the collective work of swarm robots can provide sensing coverage and deliver response action. Search and rescue systems impose particular requirements [11] that set at the heart of the justification in using swarming robots and swarm intelligence based control. These requirements centre on the need for robustness of the system against the loss of components. As robots search their way through damaged buildings or in fighting fire, they are likely to be damaged, lost in the rubble, or become inoperative for number of reasons. Also recently the sensors are increasingly introduced for intelligent homes use [8]. With the expansion in the availability of high-speed networks the idea of bringing home devices online is an active area.

Different approaches to combine PSO with the other evolutionary algorithms have been reported. Robinson et al. in [34] obtained better results by applying PSO first followed by applying GA in their profiled corrugated horn antenna optimization problem. In [25], either particle swarm optimization algorithm, genetic algorithm, or hill climbing search algorithm can be applied to a different sub-population of individuals which each individual is dynamically assigned to according to some predesigned rules. In [21], ant colony optimization is combined with PSO. A list of best positions found so far is recorded and the neighborhood best is randomly selected

s1	s2	s3	s4	s5	s6	s7	s8
1	0	0	1	0	1	0	0

Individual nodes s1, s4 and s6 are cluster-heads. The remaining nodes are regular sensors. The initial population consists of randomly generated individuals. GA is used to select cluster-heads. Each regular node uses a deterministic method to find its nearest cluster-head.

4.2 Modified GA Algorithm

In this research we have developed the basic GA in a way that in case of any cluster-head remain unconnected with any regular sensor then its state should be changed to be a regular node and linked to the nearest cluster-head available in the field. This process will eliminate inefficient clusters to survive. Decreasing the number of clusters will enhance the overall distance optimization of the sensors network [33]. As a result the optimization process will produce more energy efficient topology for the sensor network. The proposed algorithm is shown in Algorithm (2).

Initialization: Generate random population of n chromosomes
while *the stop condition is not satisfied* **do**
    if *cluster-head not connected to any sensor-node* **then**
        change cluster-head state into regular sensor;
        find the nearest cluster-head to be connected with;
    end
    Evaluate the fitness $g(x)$ of each chromosome x in the population;
    while *the new population is not complete* **do**
        Selection: Select two parent chromosomes from a population according to their fitness;
        Crossover: With a crossover probability, crossover the parents to form a new offspring (children);
        Mutation: With a mutation probability mutate new offspring;
        Accepting: Place new offspring in a new population;
    end
    Replace: Use new generated population for further runs;
end
Return: the best solution of the current population;

Algorithm 2: Modified GA Algorithms

---

4 PHASE-1: GA BASED CLUSTERS INITIALIZATION

Genetic Algorithms are used in phase-1 to generate the optimum clusters distribution for the sensor nodes before moving. In this stage the cluster-heads and its relative members are identified.

Once cluster-heads are selected, each regular node connects to its nearest cluster-head. Any node in a network is either a cluster-head or a "member" related to a cluster-head. Member-node can only belong to one cluster-head. Cluster-heads collects data from all sensors within its cluster and each head directly sends the collected data to the sink-point. Figure 2 shows an example of clustering.
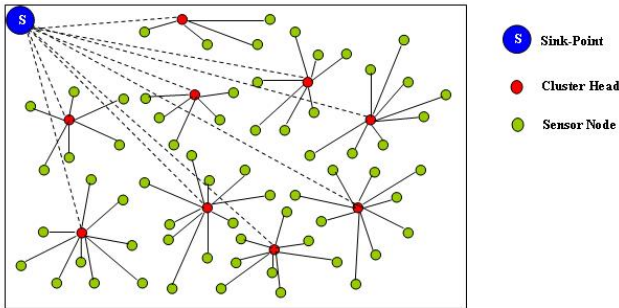

Figure 2. Clustered Sensors Network

4.1 Chromosome Representation of Distance-Head Problem

In order to find appropriate cluster-heads is critically important to minimizing the distance. We use binary representation in which each bit corresponds to one sensor. A "1" means that corresponding sensor is a cluster-head; otherwise, it is a regular node. In the following example:

4.3 Fitness Function: Distance-Number of Heads Rule

The total transmission distance is the main factor we need to minimize. In addition, the number of cluster heads can factor into the function. Given the same distance, fewer cluster heads result in greater energy efficiency as cluster heads drain more power than non-cluster-heads. Thus, each individual is evaluated by the following combined fitness components:

$$Fitness = w * (D - distance_i) + (1 - w) * (N - H_i) \quad (1)$$

where D is the total distance of all nodes to the sink, $distance_i$ is the sum of the distances from regular nodes to cluster-heads plus the sum of the distances from all cluster-heads to the sink; H_i is the number of cluster-heads; N is the total number of nodes; and w is a predefined

weight. Except for $distance_i$ and H_i, all other parameters are fixed values in a given topology. The shorter the distance, or the lower the number of cluster-heads, the higher the fitness value of an individual is. Our GA tries to maximize the fitness value to find a good solution. The value of w is between 0 and 1, and it is application-dependent. It indicates which factor is more important to be considered: distance or the cost incurred by cluster-heads. At one extreme, if $w = 1$, we optimize the network only based on the communication distance. If $w = 0$, only the number of cluster heads is considered.

5 PHASE-2: PSO BASED MOVABLE CLUSTERS

The second part of our algorithm is designed to provide the distance management by using Particle Swarm Optimization (PSO) which makes the wireless sensor network self organised while the sensors are moving on a swarm bases. In PSO, the potential solutions are called particles, fly through the problem space by following the current optimum particles. The particles are initialised randomly [5]. Each particle will have a fitness value, which will be evaluated by the fitness function to be optimised in each generation. Each particle knows its best position $pbest$ and the best position so far among the entire group of particles $gbest$. The particle will have velocities, which direct the flying of the particle. In each generation the velocity and the position of the particle will be updated. The velocity and the position update equations are given below as (2) and (3) respectively.

$$v_i^{k+1} = wv_i^k + c_1 rand_1 * (pbest_i - s_i^k) + c_2 rand_2 * (gbest - s_i^k) \quad (2)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (3)$$

The parameters used in equations 2 and 3 are described in Table 1.

Table 1. The parameters for PSO velocity and position update

Parameter	Description
v_i^k	velocity of particle i at iteration k
w	inertia weight
v_i^{k+1}	velocity of particle i at iteration $k+1$
c_j	acceleration coefficients j=1,2
$rand_i$	random number between 0 and 1 i=1,2
s_i^k	current position of particle i at iteration k
$pbest_i$	$pbest$ of particle i
$gbest$	$gbest$ of the group
x_i^{k+1}	position of the particle i at iteration $k+1$

The pseudo code for phase-2 of our proposed algorithm is shown in Algorithm (3).

PSO Initialization: Assume the initial population is the best solution generated by the previous stage of GAs;
while *the stop condition is not satisfied* **do**
    Evaluate the fitness value for each particle's position in the swarm;
    if $fitness(p)$ *better than* $fitness(pbest)$ **then**
        $pbest = p$;
        Set best of $pbest$ as $gbest$;
    end
    Update the particles' velocity v_i^{k+1};
    Update the particles' position x_i^{k+1};
end

Algorithm 3: Phase-2 PSO part of the Algorithm

In recent times, there has been a number of improvements to the original PSO [32]. In this paper we have explored two versions of PSO algorithms which are extension to the original PSO algorithm. These are discussed in the following sections.

5.1 PSO - Time Varying Inertia Weight (TVIW)

PSO-TVIW model is the same basic PSO algorithm with inertia weight parameter is varying with time from 0.9 to 0.4 and the acceleration coefficient is set to 2. This model is proposed by [35]. The time varying inertia weight is mathematically represented as follows:

$$w = (weight - 0.4) * \frac{(MAXITER - iter)}{MAXITER} + 0.4 \quad (4)$$

Where, MAXITER is the maximum iteration allowed, $iter$ is the current iteration number and $weight$ is a constant set to 0.9.

5.2 Particle Swarm Optimisation with Supervisor-Student Model (PSO-SSM)

The PSO-SSM model presented to achieve low computational costs as compared with the standard PSO algorithm. The algorithm introduces a new parameter called momentum factor (mc) to update the positions of particles as well as different velocity updating mechanism is presented [28]. In PSO-SSM model, velocity is updated only if each particle's fitness at the current iteration is not better than that of previous iteration. The velocity serves as a navigator (supervisor) by getting the right direction, while the position (student) gets a right step size along the direction. The velocity and the position are modified using the following equations:

$$v_i^{k+1} = v_i^k + c_1 rand_1 * (pbest_i - s_i^k) + c_2 rand_2 * (gbest - s_i^k) \quad (5)$$

$$x_i^{k+1} = (1 - mc) * x_i^k + mc * v_i^{k+1} \quad (6)$$

6 IMPLEMENTATION AND EXPERIMENTATION

6.1 Energy Model for Optimisation

We are studying the impact of the transmission range of sensor nodes and positioning of the sink in minimising the communication energy in a sensor network. The important components of each sensor are the data and control processing unit and the radio for communication. The microprocessor used in the processing unit should be energy efficient with less energy consumption. The energy dissipation in the radio depends on the different characteristics of the radio. The energy model used in this work is adopted from [20, 19, 42] and summarised here. The energy dissipation for transmitting b bits to d distance is shown in Equation 7.

$$E_{tx}(b, d) = E_{elec} \times b + E_{amp} \times b \times d^2 \quad (7)$$

Energy dissipation in a node to receive b bits of data is shown in Equation 8.

$$E_{rx}(b) = E_{elec} \times b \quad (8)$$

Where E_{elec} is the radio energy dissipation and E_{amp} is the transmition amplifier energy energy disipation. Energy consumption of a wireless sensor node transmitting and receiving data from another node at a distance d can be divided into two main components: Energy used to transmit, receive and amplify data and energy used for processing the data, mainly by the microcontroller. Leakage

current can be as large as a few mA for the microcontroller, and the effect of leakage current can be neglected for higher frequencies and lower supply voltage. Assuming the leakage current as negligible, the total energy loss for the sensor system due to the distance E_{dd} can be calculated according to Figure 3 using the following equation:

$$E_{dd} = \left(\sum_{j=1}^{k} \sum_{i=1}^{n_j} (d_{ij}^2 + \frac{D_j^2}{n_j}) \right) \qquad (9)$$

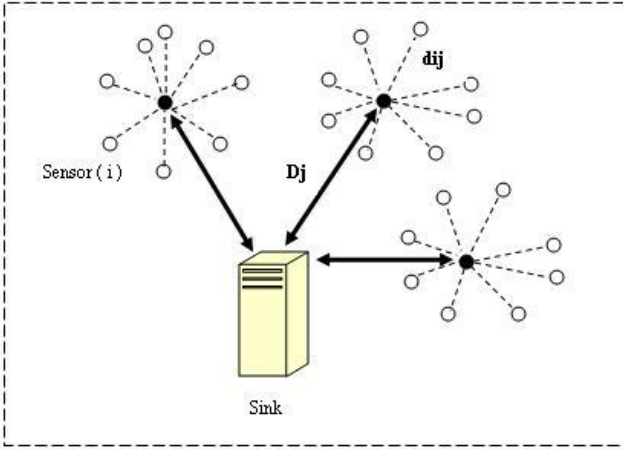For more details about the derivation and proof refer to [19].

Figure 3. Energy Model for distance based Sensor Network

6.2 Experiments and Simulation

In this section, we explore the use of GAs and PSO to solve the distance minimization problem for dynamic sensor networks.

Phase-1:
To implement our algorithm, we have used Java-Applet as a programming environment to simulate an experiment with 100 generated random nodes in a simulated 2-D environment with two different sink positions located at (0,0) and (100,100). As a tuning parameters for GAs, we used the parameters given in Table 2.

Table 2. The GA parameters settings

Parameter	Value
Population size	100
Selection type	Proportional
Crossover rate	0.7
Crossover type	one point
Mutation rate	0.005
Generation size	1000

We explored two case studies. They are:

- **case 1:** when the sink point is located at (0,0) (i.e. the upper left corner) and w is set 1.0, Figure 4. This network distribution is suitable when the application environment is inhospitable, which will

be not safe to allocate the sink-point (i.e. data collector) within the field area like some military applications or earthquake observations, etc.
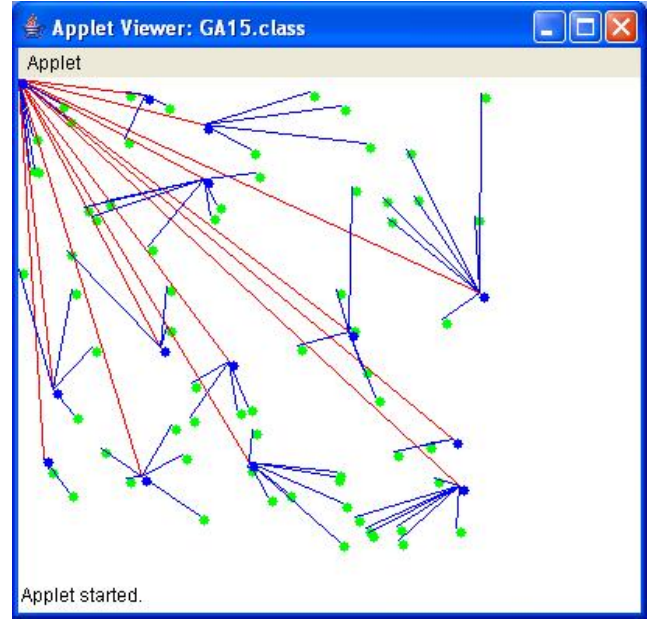

Figure 4. Clustered network when sink point at (0,0)

- **case 2:** when the sink point is located at (100,100) and w set to 0.8, Figure 5. This network distribution is more suitable when the sensor nodes are distributed around a centralized safe area where the sink-point can receive the data in a wider circular range and from different directions. For example the Mobile networks.

Phase-2:
The second phase of the algorithm enabling the sensors to move as a swarm using PSO while keeping the optimum distances between the sensor-nodes and their related cluster-head, avoiding any unnecessary movements. Refering to Equation (9), we can conclude that by reducing the distance from a node to the cluster-head and the cluster-head to the sink we can minimise the energy dissipation in a sensor network. In our simulation, we cluster the nodes taking into consideration that each node can transmit or receive data from all other nodes. Thus, nodes considered in this network do not have transmission range constraint. The fitness function used in this phase of our algorithm is based on Equation (9) described in the previous section. Using this fitness function, the sensors will be grouped on entirely distance base as shown in Equation (10) below:

$$Fitness = min \left(\sum_{j=1}^{k} \sum_{i=1}^{n_j} (d_{ij}^2 + \frac{D_j^2}{n_j}) \right) \qquad (10)$$

where,

$$\sum_{j=1}^{k} (n_j + k) = N$$

N is the number of nodes in a network. For our simulations, we used 100-node networks that are uniformly distributed in a 2-Dimensional problem space [0:100,0:100]. We have studied the impact of sink location on the fitness value of the PSO algorithms.
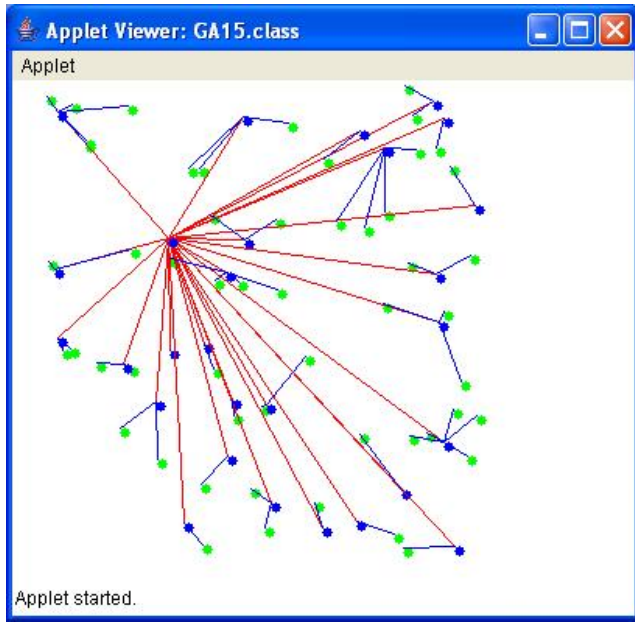
Figure 5. Clustered network when sink point at (100,100)

In one set of simulations we considered the sink-point to be located at the center of the network (50,50). In another set of simulations we considered the sink-point to be located remotely at (50,180). For both simulations we use the same set of nodes. The maximum number of generations we were running was 1000. The parameters used in the simulations are tabulated in Table 3. Snapshots for the

Table 3. Initialisation and Parameters Range

Parameter	Range
Population size	100
$MAXITER$	1000
v_{max}	100
x_{max}	100
v range	0-100
x range	0-100

mobile swarmed sensor-nodes are shown in Figure 6. Figure 6-a shows the initial distribution for sensor-nodes which is produced by GAs from the previous phase of our algorithm. It can be observed from this distribution that the WSN is clustered into 4-clusters, each one represents a swarm to be directed and controlled by the PSO when it will start running in the second phase of the algorithm. During PSO phase, clusters will be self-organised while they are moving within the experimentation boundaries. This will avoid the mobile sensors to make any unnecessary movements to reserve energy and enlarge the lifetime for each sensor. It is clear from the screen shots shown in Figure 6 - b, c, d, e and f respectively, that the mobile sensors in each cluster keep adjusting their positions during the movements to keep the distances between the sensor-nodes as much as possible the same as it was in the initial distribution.
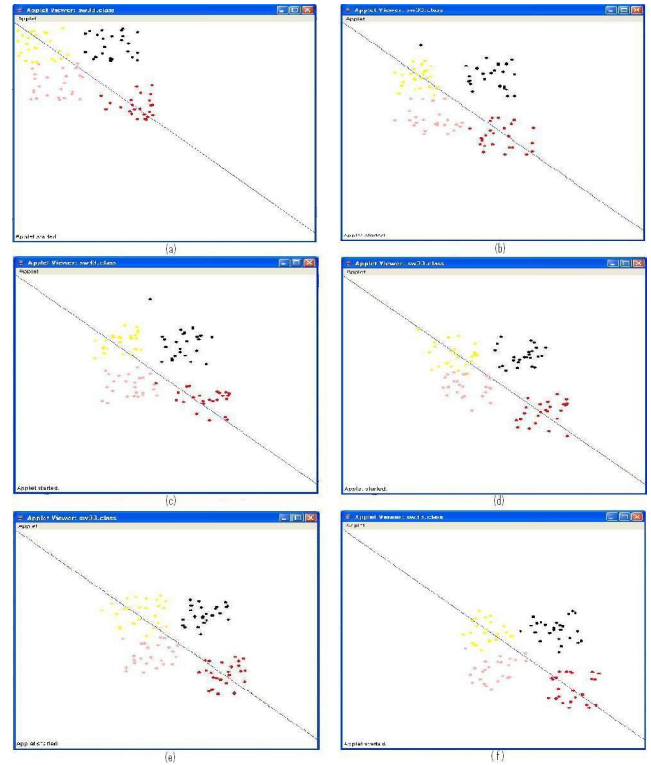
Figure 6. Snapshots of swarmed WSN with 4 clusters crossing the problem space

7 CRITICAL REVIEW AND RESULTS

Our proposed approach was able to find quickly the optimal solutions. For a 100-node problem, a good solution can be achieved after around 130 generations as shown in Figure 7 which is relatively a small number of generations in such applicatons. The fitness value

Figure 7. Fitness values over generations using modified GAs

is greatly enhanced after 100 generations due to the selection of the

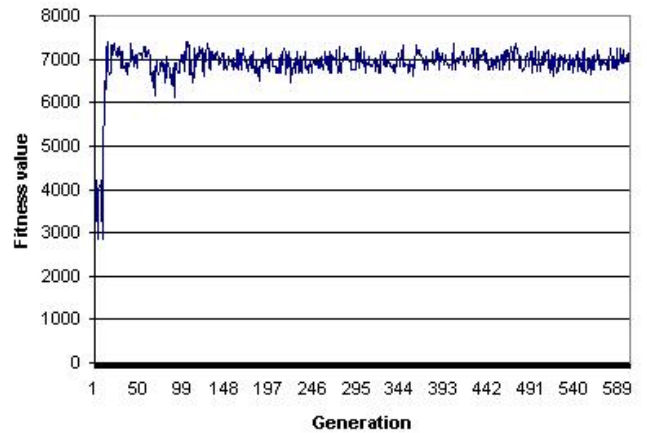best fitness chromosomes to be used in the next generation. In Figure 8, number of cluster heads decreases over generations to reach around 25% from the overall number of nodes in the network. This verifies the effectiveness of our algorithm because, as expected, the total distance will be minimized as the number of heads decreases. Experiments indicate that the scaling window plays an important role
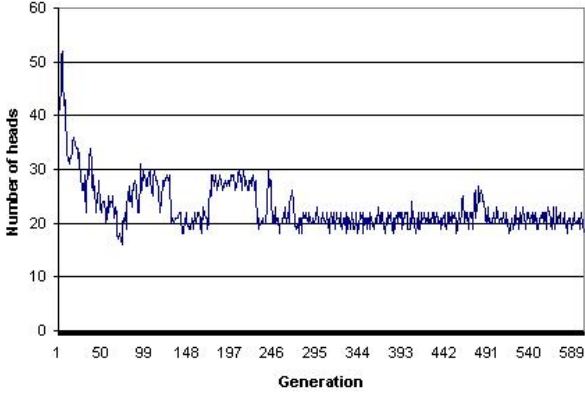

Figure 8. Number of Cluster heads over generations using GAs

in the quality of the solution found. When a single node is near to the sink, that node itself becomes a cluster-head and sends data directly to the sink. Experiments also show that nodes near to the sink are more likely become cluster-heads than those far away. More cluster-heads are needed when a sink is close to the center of a network than when it is located at a network corner. This observation is expected because when the sink is at the center, all regular nodes are located around the sink. As a result, cluster-heads tend to be distributed around the sink.

In this work we observed the performance in terms of quality of the average optimum value for 10 trials to the **PSO-SSM** and **PSO-TVIW** models which are described earlier. We chose these two methods for the following reasons; the **PSO-SSM** model is the only model which has the ability to stop particles from moving beyond the boundary of the problem space, that is under the influence of mc parameter in it. The **PSO-TVIW** model is almost similar to the basic PSO algorithm with just the inertia weight varying with time from 0.9 to 0.4. From the graph shown in Figure 9 we can conclude that **PS-TVIW** convergence is slower as compared to the **PSO-SSM** algorithm. This was due to constant acceleration co-efficients used in this model which affects the rate of convergence.

Table 4 shows a comparison of different clustering algorithms in terms of the purpose, decision neighbourhood rang, mobility, and finally whether the clusters are disjoint or not. By analysing this table, we can observe that most of the existing clustering algorithms are less suitable for mobile environment. The reasons for that are: Firstly, electing the cluster-heads based on information from nodes which are multiple hopes away leads to high overhead and slow reaction to topology changes. Secondly, maintaining complete intra-cluster information is an expensive task which results in a high traffic. Thirdly, the complexity of the multi-layer clustering algorithms leads to a lot
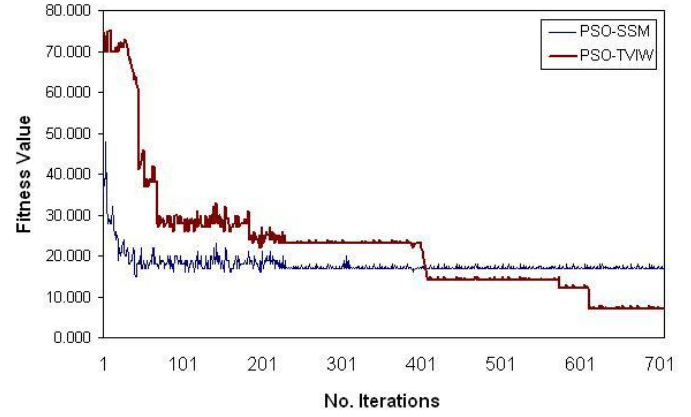

Figure 9. Convergence for the PSO-SSM and PSO-TVIW Models

of efforts in building and maintaining the desired structure.

Simulation results of our algorithm showed that our approach is an efficient and effective method with respect to distance minimization in mobile WSNs. For a scalable sensors network the average value of the communication distance is reduced by 84% using our approach as compared with the distance when direct transmission method is used.

Table 4. A Comparison of Clustering Algorithms

Algorithm	Purpose	Decision neighbour-hood range	Mobility	Disjoint Clusters
WCA [9]	MAC	Network wide	Mobile	Yes
LEACH [20]	Data collection	1-hop	Static	Yes
HEED [48]	Routing	1-hop	Quasi static	Yes
MOCA [49]	Data collection	k-hops	Static	No
Coyle et al. [2]	Data collection	k-hops	Static	Yes
EEMC [23]	Data collection	1-hop	Static	Yes
Bouhafs et al. [7]	Data collection	Network wide	Static	No
Tandem [29]	Collaborative processing	1-hop	Mobile	Yes
Smart clustering [38]	Routing	1-hop	Quasi static	Yes
Wang et al. [43]	Information decimation	Network wide	Quasi static	Yes

8 CONCLUSIONS AND FUTURE WORK

In this paper, we propose the use of GAs to minimize the communication distance in a sensor network by dividing it into clusters and the use of PSO to make this network moves as a Swarm keeping the optimum distances between the sensors while they are moving. Our proposed approach starts by taking random selecting nodes in a network to be used as a cluster-heads. The algorithm then starts to find an appropriate number of cluster-heads and their locations by

adjusting cluster-heads based on fitness function. We also explored the results of the performance evaluation of four extensions to the standard Particle Swarm Optimization algorithm in order to reduce the energy consumption in Wireless Sensor Networks. Communication distance is an important factor to be reduced in sensor networks. We have simulated two models; the Supervisor-Student Model (PSO-SSM) and the time varying Inertia Weight (PSO-TVIW) model. In the (PSO-SSM) model the new parameter introduced called the momentum factor mc to update the position of particles. Also here the velocity is updated only if each particle's fitness at the current iteration is not better than that of previous iteration. Hence the computational costs for this algorithm will be decreased. An important modification proposed is to use boundary checking for re-initialization of particle which moves outside the set boundary. We can also conclude that (PSO-TVIW) convergence is slower as compared to other algorithm. As a future work, our program can be upgraded to cover the two other models described in this paper, then a comprehensive comparison could be done to analyze the behavior of the particles within each case.

We plan to extend the problem on hand by considering a hierarchical structure where a cluster-head can have a super cluster-head which sends data directly to the sink.

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, 'A survey on sensor networks', *IEEE Communications Magazine*, **8**(40), 102–114, (August 2002).

[2] S. Bandyopadhyay and E. J. Coyle, 'An energy efficient hierarchical clustering algorithm for wireless sensor networks', *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, **3**, 1713–1723, (2003).

[3] E. Benini and A. Toffolo, 'Optimal design of horizontal-axis wind turbines using blade-element theory and evolutionary computation', *Journal of Solar Energy Engineering*, **124**(4), 357–363, (November 2002).

[4] F. Bergh and A. Engelbrecht, 'A cooperative approach to particle swarm optimisation', *IEEE Transactions on Evolutionary Computation*, 225–239, (2004).

[5] C. Bertelle, M. Obaidy, A. Ayesh, and R. Ghnemat, 'Intelligent land-use management and sustainable development: From interacting wireless sensors networks to spatial emergence for decision making', *Engineering of Autonomic and Autonomous Systems, IEEE International Conference and Workshops, Oxford, England*, **0**, 73–78, (2010).

[6] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, NY: Oxford Univ. Press, 1999.

[7] F. Bouhafs, M. Merabti, and H. Mokhtar, 'A semantic clustering routing protocol for wireless sensor networks', in *In Consumer Communications and Networking Conference*, pp. 351– 355. IEEE Computer Society, (2006).

[8] M. Hsieh C. Wu C. Liang, W. Hung and C. Luo, 'A multi-agent based architecture for an assistive user interface of intelligent home environment control', in *Eighth International Conference on Intelligent Systems Design and Applications ISDA 08*, volume 1, pp. 335–338, (2008).

[9] M. Chatterjee, S. Das, and D. Turgut, 'Wca: A weighted clustering algorithm for mobile ad hoc networks', *Journal of Cluster Computing (Special Issue on Mobile Ad hoc Networks)*, **5**, 193–204, (April 2002).

[10] W. Chu-Fu, S. Jau-Der, P. Bo-Han, and W. Tin-Yu, 'A network lifetime enhancement method for sink relocation and its analysis in wireless sensor networks', *IEEE Sensors Journal*, (December 2013).

[11] B. Boldt S. Skousen D. Stormont, A. Bhatt and M. D. Berkemeier., 'Building better swarms through competition: lessons learned from the aaai/robocup rescue robot competition', in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, volume 3, p. 28702875, (2003).

[12] W. Dargie and M. Zimmerling, 'Wireless sensor networks in the context of developing countries', in *The 3rd IFIP World Information Technology Forum (WITFOR)*, Addis Ababa, Ethiopia, (August 2007).

[13] M. Dorigo and T. Sttuzle, *Ant Colony Optimization*, MIT press, 2004.

[14] C.H. Duc, 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*, **10**(1), 774–783, (Feb 2014).

[15] A. Esmin and A. Coelho, 'Consensus clustering based on particle swarm optimization algorithm', in *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, pp. 2280–2285, (Oct 2013).

[16] M. Turner G. Al-Hudhud, A. Ayesh and H. Istance, 'Simulation and visualisation of a scalable real time multiple robot system', in *Theory and Practice of Computer Graphics, UKCG05*, (2005).

[17] M. Gerla, T. Kwon, and G. Pei, 'On-demand routing in large ad hoc wireless networks with passive clustering', in *In Wireless Communications and Networking Conference (WCNC)*, pp. 100–105, Chicago, IL, USA, (2000). IEEE Computer Society.

[18] Mario Gerla and Kaixin Xu, 'Multimedia streaming in larg-scale sensor networks with mobile swarms', *SIGMOD Record*, **32**(4), 72–76, (December, 2003).

[19] S. Guru, A. Hsu, S. Halgamuge, and S. Fernando, 'An extended growing self-organising map for selection of clustering in sensor networks', *International Journal of Distributed Sensor Networks*, **1**(2), 227–243, (2005).

[20] W. Heinzelman and A. Chandrakasan, 'An application-specific protocol architecture for wireless micro-sensor network', *IEEE Transactions on Wireless Communications*, **1**(4), 660–670, (2002).

[21] T. Hendtlass and M. Randall, 'A survey of ant colony and particle swarm metaheuristics and their application to discrete optimization problems', in *Proceedings of The Inaugural Workshop on Artificial Life (AL'01)*, pp. 15–25, Adelaide, Australia, (2001).

[22] X. Jiang, J. Polastre, and D. Culler, 'Perpetual environmentally powered sensor networks', in *4th Int'l Conf. Information Processing in Sensor Networks (IPSN'05)*, Los Angeles, CA, (2005).

[23] Y. Jin, L. Wang, Y. Kim, and X. Yang, 'Eemc: An energy-efficient multi-level clustering algorithm for large-scale wireless sensor networks', *Computer Networks*, **3**(53), 542–562, (2008).

[24] J. Kennedy and R. C. Eberhart, *Swarm Intelligence*, Morgan Kaufman Publishers, 2001.

[25] T. Krink and Lovbjerg, 'The life cycle model: combining particle swarm optimisation, genetic algorithms and hillclimbers', in *Proceedings of Parallel Problem Solving from Nature (PPSN)*, pp. 621–630, Granada, Spain, (September 2002).

[26] S. Kumar, F. Zhao, and D. Shepherd, 'Collaborative signal and information processing in microsensor networks', *Signal Processing Magazine, IEEE*, **19**, 13–14, (March 2002).

[27] C. Laurent, D. Helal, L. Verbaere, A. Wellig, and J. Zory, 'Wireless sensor networks devices: Overview, issues, state of the art and promising technologies', *ST Journal of Research*, **4**(1), 8–11, (June 2007).

[28] Y. Liu, Z. Qin, and X. He, 'Supervisor-student model in particle swarm optimization', in *Proceedings of CEC2004 Congress on Evolutionary Computation*, volume 1, pp. 542–547, USA, (2004).

[29] R. Marin-Perianu, C. Lombriser, P. Havinga, J. Scholten, and G. Troster, 'Tandem: A context-aware method for spontaneous clustering of dynamic wireless sensor nodes', in *Proceedings of Internet of Things, the International Conference for Industry and Academia*, pp. 342–360. Springer, (2008).

[30] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M.B. Srivastava, 'Coverage problems in wireless ad-hoc sensor networks', in *Proceedings of IEEE Infocom conference*, volume 3, pp. 1380–1387, Anchorage, AK, USA, (April 2001).

[31] D. Estrin M. Hansen T. Harmon J. Jay W. Kaiser N. Ramanathan, L. Balzano and G. Sukhatme, 'Designing wireless sensor networks as shared resource for sustainable development', in *International Conference on Information and Communication Technologies and Development - ICTD06*, (2006).

[32] M. Obaidy and A. Ayesh, 'Energy efficient pso-based algorithm for optimizing autonomous wireless sensor network', in *European Simulation and Modelling (ESM'2008) Conference. EUROSIS, Le Havre, France*, (October 2008).

[33] M. Obaidy, A. Ayesh, and A. Sheta, 'Optimizing the communication distance of an ad hoc wireless sensor networks by genetic algorithms', *Artificial Intelligence Review, Springer*, **29**(3), 183–194, (November 2009).

[34] J. Robinson, S. Sinton, and Rahmat Samii, 'Particle swarm, genetic algorithm, and their hybrids: optimization of a profiled corrugated horn

antenna', *IEEE International Symposium on Antennas and Propagation. San Antonio, Texas*, (June, 2002).

[35] Y. Shi and R. Eberhart, 'Empirical study of particle swarm optimization', in *Proceedings of the Congress on Evolutionary Computation, CEC 99*, volume 3, pp. 1945–1950, Washington, DC, USA, (July 1999).

[36] J. Shiyuan, Z. Ming, and W. Annie, 'Sensor network optimization using a genetic algorithms', in *7th World Multiconference on Systemics, Cybernetics, and Informatics*, Orlando, Forida, (July 2003).

[37] T. Stathopoulos, D. McLntire, and W. J. Kaiser, 'The energy endoscope: Real-time detailed energy accounting for wireless sensor nodes', in *Int'l Conf. Information Processing in Sensor Networks (IPSN'08)*, pp. 383–394, St. Louis, MO, (2008).

[38] M. Strohbach and H. Gellersen, 'Smart clustering - networking smart objects based on their physical relationships', in *Proceedings of the 5th IEEE International Workshop on Networked Appliances*, pp. 151– 155. IEEE Computer Society, (2002).

[39] Y. K. Tanand and S. K. Panda, 'Energy harvesting from hybrid indoor ambient light and thermal energy sources for enhanced performance of wireless sensor nodes', *IEEE Transaction, Industrial Electronics*, **58**(9), 4424–4435, (September 2011).

[40] D. Tian and N. Georganas, 'A coverage-preserving node scheduling scheme for large wireless sensor networks', in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pp. 32–41, Atlanta, USA, (2002).

[41] R. Torah, P. Glynne-Jones, M. Tudor, and S. Beeby, 'Energy aware wireless micro-system powered by vibration energy harvesting', in *PowerMEMS*, pp. 323–326, Freiburg, Germany, (2007).

[42] Chu-Fu Wang, Jau-Der Shih, Bo-Han Pan, and Tin-Yu Wu, 'A network lifetime enhancement method for sink relocation and its analysis in wireless sensor networks', *Sensors Journal, IEEE*, **2**(99), 1–12, (2014).

[43] K. Wang, S. Ayyash, T. Little, and P. Basu, 'Attribute-based clustering for information dissemination in wireless sensor networks', in *Proceedings of the Second Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, pp. 498–509. IEEE Computer Society, (2005).

[44] A. Weddell, N. Harris, and N. White, 'An efficient indoor photovoltaic power harvesting system for energy-aware wireless sensor nodes', in *Eurosensors'08*, pp. 1544–1547, Dresden, Germany, (2008).

[45] C. Wei-Peng, H. Jennifer, and C. Lui, 'Dynamic clustering for acoustic target tracking in wireless sensor networks', *IEEE Transactions on Mobile Computing*, **3**, 258–271, (July/September 2004).

[46] Z. Wensheng and C. Guohong, 'An energy efficient framework for mobile target tracking in sensor networks', *IEEE Proceedings - Military Communications Conference MILCOM, USA*, **1**, 597–602, (2003).

[47] H. Yang and B. Sikdar, 'A protocol for tracking mobile targets using sensor networks', *Proceedings of the First IEEE workshop on Sensor Network Protocols and Applications*, 71–81, (May 2006).

[48] O. Younis and S. Fahmy, 'Heed: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks', *IEEE Transactions on Mobile Computing*, **4**(3), 366–379, (2004).

[49] A. Youssef, M. Younis, M. Youssef, and A. Agrawala, 'Distributed formation of overlapping multi-hop clusters in wireless sensor networks', in *Proceedings of the Global Telecommunications Conference (GLOBECOM)*, pp. 1–6, (2006).

[50] B. Zhang, R. Simon, and H. Aydin, 'Harvesting-aware energy management for time-critical wireless sensor networks with joint voltage and modulation scaling', *IEEE Transaction on Industrial Inf.*, **9**(1), 514526, (February 2013).

[51] F. Zhao, J. Shin, and J. Reich, 'Information-driven dynamic sensor collaboration', *IEEE Signal Processing Magazine*, **9**, 61–72, (March 2002).