Genetic Algorithm Based Energy Efficient Optimization Strategies in Wireless Sensor Networks: A Survey

Innocent Njini[1] and Obeten O. Ekabua[2]
[1]Department of Computer Science, North-West University, Mafikeng Campus,
Private Bag X2046, Mmabatho 2735, South Africa
Innocent.njini@gmail.com

[2]Department of Computer Science, North-West University, Mafikeng Campus,
Private Bag X2046, Mmabatho 2735, South Africa
obeten.ekabuao@nwu.ac.za

Abstract

The past decade has witnessed tremendous growth in research in various issues of concern in wireless sensor networks (WSNs) such as energy conservation, node deployment, routing protocols, Quality of services (QoS) management, security, energy harvesting etc. Most of the issues involved in WSNs research are conflicting in nature and hence require optimization strategies that are capable of mitigating the conflicting objectives such as life time maximization, node coverage and reliability among others. In this survey paper, we stimulate new research initiatives by reviewing how a more holistic view to optimization can be achieved through the use of genetic algorithms (GAs) in sensor network optimization. We review how genetic algorithms have been used to model sensor communication, in clustering and routing problems. We also provide a performance evaluation of various GA-based optimization strategies. Our observations shows that while a number of algorithms try to select the best cluster headers or routing path based on some metric, the process normally introduces overheads in communication which in turn leads to more energy dissipation. We propose that future research should focus more on the use of Stochastic Network State Model to model the behavior of sensor nodes and then predict energy consumption by a sensor node with minimum overheads in communications to base station.

Key words: *wireless sensor networks, Genetic algorithms, optimization strategies.*

1. Introduction

Wireless sensor networks are composed of hundreds or thousands of sensor nodes deployed in the environment for the purposes of detecting and transmitting information of interest. According to Chen etal [1], a wireless sensor network (WSN) is a collection of wireless sensor nodes forming a temporary network without the aid of any established infrastructure or centralised administration [1]. In such an environment, there is limited range therefore it is necessary for one sensor node to collaborate for the help of another node in forwarding packets to its base station. Usually the device nodes consists of CPU for data processing, memory for data storage, battery for energy and a transceiver for receiving and sending signals or data from one node to another. WSN are generally characterized by short range radio communications, limited computational capacity and limited generally irreplaceable battery power [2].

1.1 Research Challenges in WSNs

One of the major concerns in the design and deployment of wireless sensor networks is the issue of maximizing network life time. The problem is how do we minimize the energy utilized by the sensors in order to extend the life of the sensor network?

Network life span is decided by network connectivity and network connectivity in turn depends on network bottleneck nodes. This problem is further exacerbated by the limited communication range of radio communication used by the sensor nodes. For most practical implementations, multi-hop transmission is thus necessary.

In a multi-hop wireless sensor networks, nodes that are close to sink node transmit much more data, and then exhaust their energy while other nodes in the same network remain with energy. It has been shown in literature that the main source of energy consumption in wireless sensor networks is signal transmission/reception, which consumes 50 times more than the energy required to process 1000 instructions [3]. This means that the more hops we have the more number of transmissions we have and the more energy consumed. In [4], it is estimated that the energy consumed to transmit k bits of data over a distance d is:

$$E(k,d) = E_{elec} * k + E_{amp} * k * d^2 \qquad (1)$$

Where E_{elec} is the radio energy dissipation and E_{amp} is transmitting amplifier energy dissipation. Now consider a situation where we have to send a large chunk of data. This means we have to break it down into smaller chunk sizes that can be accommodated by the channel capacity and sent them over the several hops discussed above resulting in a lot of energy consumption. It is there for necessary to research on the various ways in which energy can be conserved as sensor nodes transmit data from source to sink within a WSN. Literature reveals that genetic algorithms are playing an increasing important role in the design and deployment of wireless sensor networks (WSNs). Recent advances in wireless sensor networks have led to many new routing protocols and clustering methods using genetic algorithms specifically designed for energy awareness.

The rest of this paper is organized as follows: Section 2 discusses genetic algorithms. In section 3 we discuss the main approaches to clustering using genetic algorithms and other optimization approaches. Section 4 discusses multi-objective genetic algorithms based optimization strategies.

2. Genetic Algorithms

Genetic algorithms are efficient stochastic optimization search procedures that mimic the adaptive evolution process of natural systems. They have been successfully applied to in many NP-hard problems such as multiprocessor deign, task scheduling, optimization and travelling salesman problem. Genetic algorithms are most useful in problems with large irregular search space where a global optimum is required [5]. Traditional gradient based methods of optimizing encounter problems when the search space is multimodal as they tend to become stack at local maxima. Genetic algorithms tend to suffer less from this problem of premature convergence [6].

A genetic algorithm is an iterative approach, involving trial and error, which aim to find a global optimum. Nature's equivalent is the process of evolution over time, where many members are created, and each population becomes better adapted to its environment. We may simulate an evolution process by creating an initial pool of chromosomes, where each chromosome represents a typical solution to the problem we intend to solve and taking the following steps [7]:

Create a random population of *N* chromosomes (Candidate solutions for the population). Evaluate the fitness function *f(x)* of each chromosome x in the population. Generate a new population by repeating the following steps until the new population reaches population N:

1. Select two parent chromosomes from the population, giving preference to the fitter chromosomes (high *f(x)* values). Automatically copy the fittest chromosome to the next generation (this is called 'elitism').
2. With a given crossover probability, cross over the parent chromosomes to form two new offspring. (If no crossover was performed, offspring is exact copy of the parents).
3. With a given mutation probability, randomly swap two genes in the offspring.
4. Copy the newly generated population over the existing population.

5. Copy the newly generated population over the existing population.
6. If the loop termination condition is satisfied, return the best solution in current population.
7. Otherwise go to step2.

We generally let this process go on for a predetermined number of generations, or until the standard deviation of the fitness converges towards zero (When the standard deviation starts to converge, the chromosomes are generally getting fitter, so we have arrived at the best solution we can find). Assuming that the initial population is large enough, and the fitness is well defined, we would have arrived at a good solution [8].

Genetic algorithms do not find the best solution or the ideal solution. However, if we run a simulated evolution many times, they do tend to find a very good solution. So how does this process evolve fitter genes? Some of the evolutionary spiral towards fitness comes from mutations that introduce new gene sequences to the population, but the majority of Genetic Algorithm success comes from crossover. By combining portions of fit chromosomes in new ways through random crossover, Genetic algorithms will over time evolve even fitter chromosomes [9].

2.1 Research Issues in Evolutionary algorithms

When implementing evolutionary algorithms (EAs), it is necessary to specify the method of parent selection, crossover, mutation and control parameters such as population size and number of generations. These are briefly discussed below:

2.1.1 Parent Selection Schemes

One needs a method for identifying good parents to select for mating to produce the next generation. The following parental selection schemes have been predominantly used in the implementation of evolutionary algorithms:

Proportionate reproduction: in this scheme, individuals are chosen for birth in proportion to their fitness value. The probability that an individual from the i^{th} class (having common fitness value f_i is chosen for selection in the t^{th} generation is [10]:

$$m_{it} = \frac{f(x_i)}{\sum_{i=1}^{n} f(x_i)} \qquad (2)$$

Where m_{it} is the number of individuals in the population at time t with fitness i. Proportionate reproduction is usually implemented with a Monte Carlo or roulette wheel selection.

Ranking Selection

In ranking selection the population is selected from best to worst. The number of copies that an individual should receive is given an assignment function, and it’s proportional to the rank assignment of an individual.

Tournament selection: In tournament selection, a random number of individuals are chosen from the population (with or without replacement) and the best individual from the group is chosen as a parent for the next generation. This process is repeated until the mating pool is filled. There are a variety of other selection methods including stochastic remainder and universal selection [11].

2.1.2 Encoding

Encoding of chromosomes is a very important question to ask when starting to solve a problem with genetic algorithms. Encoding depends heavily on the problem. There are many ways of encoding and they depend mainly on the problem to be solved. Binary encoding is the most common one, mainly because the first research on genetic algorithms used this type of encoding and its relative simplicity. In binary encoding, every chromosome is a string of bits, either 0 or 1. A chromosome should in some way contain information about the solution that it represents [12].

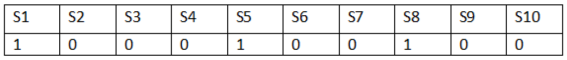
Section 3.0 discusses binary encoding for cluster based problems and section 3.1 discusses binary encoding for routing problems in wireless sensor networks

3. Clustering

Clustering partitions the network into groups of sensors nodes which are geographically close to each other. Each cluster will have a cluster head which is responsible for controlling all the activities of the group like transmission, aggregation, management and maintaining structure. With clustering in WSNs, energy consumption, lifetime of a network and scalability can be improved. Currently, the accepted and mostly used topology for clustering in WSNs is where each cluster has a cluster head. The sensor nodes transfer their data directly to their associated cluster head nodes (relay nodes) and then cluster head nodes perform the initial data aggregation and send it to the designated route [13].

3.1 Energy Efficient clustering using Genetic algorithms

Genetic algorithms Partition, the sensor network into independent clusters using GA, to minimize the total communication distance, and thus prolong the life time of a network [14]. In [15], an intelligent clustering approached in wireless sensor networks is proposed. The approach uses a genetic algorithm to minimize the total communication distance by using a binary representation in which a bit corresponds to a sensor node. A “1” corresponds to a cluster-header, while a “0”, and corresponds to an ordinary sensor node. Consider the following example, representing a typical chromosome, or GA solution.

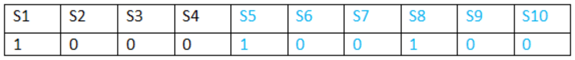


S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
1	0	0	0	1	0	0	1	0	0

Figure 1

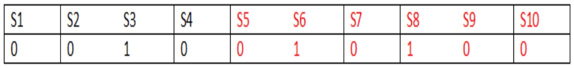
Nodes s1, s5, s8 are the cluster headers and the rest are ordinary nodes. Two parent chromosomes, parent 1 and parent 2 representing two different clustering solutions can then be engaged in a process called crossover to generate two new solutions, child 1 and child2 as indicated below.

Parent 1



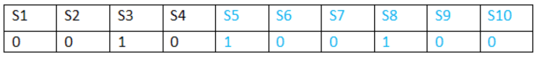
S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
1	0	0	0	1	0	0	1	0	0

Parent 2



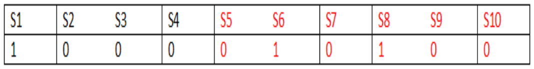
S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
0	0	1	0	0	1	0	1	0	0

Child 1



S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
0	0	1	0	1	0	0	1	0	0

Child2



S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
1	0	0	0	0	1	0	1	0	0

Figure 2

This implies that initially we had s1, s5 and s8 as the cluster headers for one typical solution, and s3, s6 and s8 as cluster headers for another solution. After the crossover, two solutions are generated with s3,s5 and s8 as the new cluster headers in one solution set and s1, s6 and s8 as the cluster headers in another solution set. After a new population is generated the performance of the clusters generated was evaluated using a Fitness function metric which depends on the total distance of nodes to the sink, and a weighted ratio of the number of cluster heads to ordinary nodes. The limitation of the approach by Jin etal[15] is that model does not take into consideration the fact that sensor nodes may alternative between sleeping mode and active modes.

3.2 Cluster Based Optimisation Strategies

An Energy Efficient Clustering Scheme Based on Grid Optimisation using a genetic algorithm which dives the network area into virtual grids with each grid representing a cluster in presented in [16]. The genetic algorithm is used to divide nodes equally among the grinds to ensure load balancing and thus enhancing the network life time. The model does not consider the fact that energy consumption pattern vary as one moves from source to sink, with nodes closer to sink transmitting more, and hence spending more energy. In [17], an algorithm which does cluster head selection using fuzzy logic and chaotic based genetic algorithm based on fussy logic is presented. Each node calculates its chance based on its energy, density and centrality. Nodes having high energy inform the base station as potential cluster header candidates. The base station uses the genetic algorithm based on chaotic reasoning to select the cluster headers. Although this approach tries to use information on residual energy as well as node density and centrality to ensure prolonged network life time, it suffers the drawback that this increases communication between the sensor nodes and the base station is another form of energy wastage.

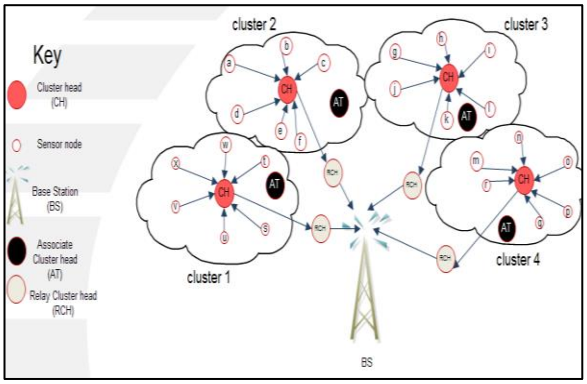


Figure 3 Network configuration in the Set-up Phase

In [18] a Genetic Algorithm Based Energy Efficiency Clusters (GABEEC) algorithm is proposed. This approach consists of two phases, the set-up phase and the steady-state phase. In the set-up phase the cluster heads are selected and other nodes are assigned to the cluster heads as ordinary nodes based on distances. In the steady-state phase, the nodes send their data to the cluster heads which in turn send forward it to the base station. The Base station then checks the energy levels of nodes and CH, and if a cluster head is having low energy, then an associate CH is selected from the population. This selection is done using Roulette-wheel selection method.

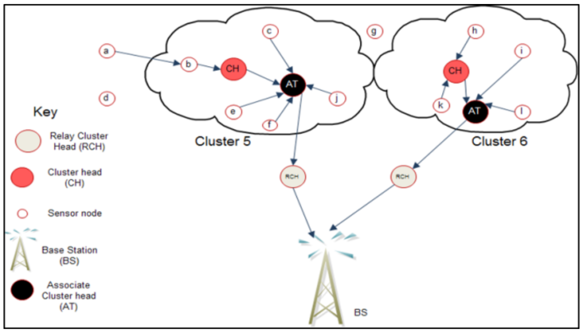


Figure 4 Network configuration in the Steady-state phase

While this approach attempts to maximize the network life time by minimizing communication distance, it also increases communication overheads as to send information about the residual energy to base station. This increased communication leads to depletion of energy and hence reducing the network life time.

4. Multi-Objective Genetic Algorithm based Optimisation Strategies.

The broad application of wireless sensor networks has resulted in the development of a wide variety of techniques which are NP hard and most of them difficult to obtain high precision solutions by traditional methods. Thus while employing genetic algorithms to solve problems in WSNs, it is important to form a broad review of the current research and future trends in the use of genetic algorithms and multi-objective genetic algorithms in particular in WSNs. The characteristics of wireless sensors networks determine a different kind of design problem with different requirements for detailed applications.

There is need for good routing protocols that should make comprehensive considerations of multiple factors to satisfy the transmission requirements of different data with Quality of services (QoS) parameters that may be conflicting and different such as end-to-end delay, energy efficient routing, node placement and layout optimization etc.

4.1 Multi-Objective Optimization Problem

The general multi-objective optimization can be modeled as

$$\min\ y = f(x) = (f_1(x), f_2\,(x), \dots, f_m(x))$$

$$subject\ to\colon \{x = (x_{1,}x_2, \dots, x_n\,) \in \mathcal{F}_x\ \ \subset X \qquad (3)$$

$where\ \ x\ \in X$ is called the decision vector, X is the optimization space, $f\ \in Y$ is the objective vector, Y is the objective space, and the set $\mathcal{F}_x$ is the feasible set composed of the solutions which satisfy the problem constraints. Let the vector f be described component-wise by $\boldsymbol{f} = (f_1\,, f_2, \dots, f_m)$, and let $\mathcal{F}_y\ \subset Y$ represent the image set of region $\mathcal{F}_x$ for the mapping $\boldsymbol{f}(.)\colon \boldsymbol{X} \longmapsto \boldsymbol{Y}$[19]. The set of solutions of a multi-objective problem consists of all decision vectors in which the corresponding objective vector can be improved in any dimension without degradation in another one. This set of solutions is known as the Pareto-Optimal set. Each element of the Pareto optimal set constitutes a non-inferior solution to the multi-objective optimization problem. The problem has usually no unique, perfect solution, but a set of equally efficient, non-inferior, alterative solutions (Pareto-optimal set). Each point in this set is optimal in the sense that no improvement can be achieved in one vector component that does not lead to degradation in at least one of the remaining components. The set of non-dominated solutions lie on a surface known as the Pareto optimal frontier [19].

In most cases, there will be several optimal solutions in the Pareto sense, and we have to look to the values of the objective function in order to decide which values seem most appropriate.

4.2 Multi-Objective Optimization Algorithms Approaches

A multi-objective optimization algorithm is one that deals directly with a vector objective function and seeks to find multiple solutions offering different, optimal tradeoffs of multiple objectives. There are basically three approaches to tackling multi-objective optimization problems which are as follows [20]:

1. Ignore some of the attributes entirely and just optimize one that looks most important.
2. Lump all attributes together by just adding them up or multiplying them together and then optimize the resulting function.
3. Apply a multi-objective algorithm that seeks to find all the solutions that are non-dominated. Non-dominated solutions are those that are optimal under some/any reasonable way of combining the different objective functions into a single one. A non-dominated individual is one where an improvement in one objective results in deterioration in one or more of the other objectives when compared with the other individuals in the population.

Thus in this paper we argue that (3) seeking multiple, distinct solutions to a problem, conferring different tradeoffs of objectives, is the essence of true multi-objective optimization (MOO). In the next section we discuss various implementations of MOO in WSNs.

4.3 Multi-Objective Optimisation Strategies

In [21] a Dynamic Multi-objective Hybrid Approach for designing WSNs is presented. The approach proposes a multi-objective hybrid approach for solving Dynamic Coverage and Connectivity problems (DCCP)is a network with no clusters heads and with nodes subject to node failures. The rationale behind this approach is to maximize the network life time by minimizing the number of active nodes in each time period, while complying with the network requirements. The presented approach using a local online algorithm intended to restore the network coverage when one or more node failures occur. The limitation of this approach lies in the assumption that there are online resources that may be available to a WSNs and this may not always be the case.

4.3.1 Quality of Service Routing

In [22] Ekbatanifard etal, proposed a Multi-objective Genetic Algorithm based Approach for Energy Efficient Qos-Routing in Two tiered Wireless sensor Networks. The approach optimizes the network by routing data from source to sink in such a way that the three conflicting objectives, end-to-end delay, transmission reliability, and node residual energy are optimised. Thus these QoS parameters are used to form a multi-objective functions that serves as a performance criteria for identifying optimal routes. We briefly discuss how the Qos parameters can be modeled for clarity.

4.3.2 End to end Delay

Consider a network in which n source nodes transmits data to relay nodes which in term perform fusion and retransmits the data to the sink node through relay nodes via a multi-hop WSN. Then the networks of relay nodes form a routing tree, with sources, several intermediate nodes and a single sink node. The routing tree can be modeled as a graph [23] [24],

$G = (V, E)$, where V is the set of relay nodes and E is the set of edges. A path between source node (V_d)

and relay node (V_r) can be represented as a sequence, v_r, v_1, v_2, …v_d, where $v_i \in V$.

The delay over a particular route from source to sink follows a Weibull distribution with parameter μ [25]. The Weibull distribution gives the probability distribution of lifetimes of objects. It was originally proposed to quantify fatigue data, but it is also used in analysis of systems involving a "weakest link." Thus, the Weibull distribution can be used to model devices with decreasing failure rate, constant failure rate, or increasing failure rate. This versatility is one reason for the wide use of the Weibull distribution in reliability.

The Erlang distribution can be used to model the time to complete *n* operations in series where each operation requires an exponential period of time to complete. The probability that a delay d_p over an individual path *k* is less than *t* is estimated by the Erlangen distribution,

$$P_r\,(d_p < t) = \frac{(\frac{\mu}{\alpha})^{k\beta}\, t^{k\beta-1} e^{-(\mu t/\alpha)\beta}}{(k-1)!} \qquad (4)$$

Where α >0 is a scale constant, and β is the shape parameter. The multi-objective function seeks to minimise this objective.

4.3.3 Reliability

Path reliability can be defined as the expected number of successful end-to-end forwarding transmissions (and retransmissions) of data for a successful end-to-end delivery of and hop-by-hop acknowledgement (ETX). For a paths *p* consisting of links $v_{1,}$ …,v_n with *forward delivery ratio* fd_{vi}, and *reverse delivery ratio of* rd_{vi} for link *vi*, the reliability metric EXT may be computed as:

$$etx_{vi} = \frac{1}{(fd_{vi} \times rd_{vi})}$$

$$ETX(p) = etx_{v1} + etx_{v2} + \cdots + etx_{vn} \qquad (5)$$

The reliability of the entire routing tree is then computed by maximizing the whole routing tree reliability given by [26]:

$$R_{tree} = \left(\frac{\sum_{p\in Tree} EXT(p)}{\sum_{p\in Tree} 1}\right) - 1 \qquad (6)$$

4.3.4 Energy

Energy consumption by a relay node may be estimated by the following equation:

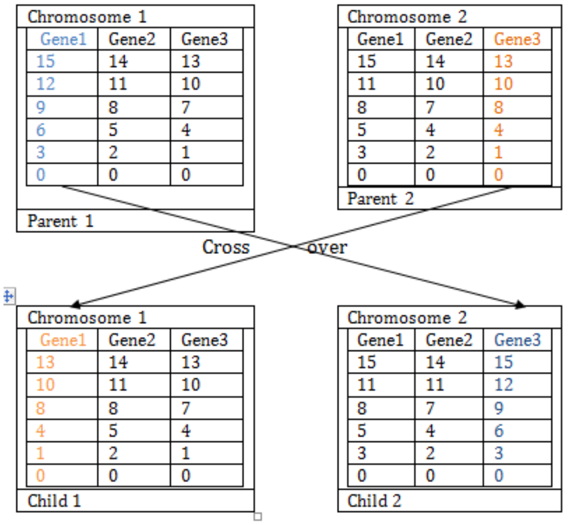
$$E_T\big(b, d_{ij}\big) = \theta_1 b + \gamma b d_{ij}^m \qquad (7)$$

$$E_R(b) = \theta_2 b$$

Where d_{ij} is the Euclidian distance between node i and *j*, θ_1 is the transmit energy coefficient, γ is the amplifier coefficient, m is the path loss exponent, $2 \leq m \leq 4$, and $\theta_2$is the receive energy coefficient. *b*, represents the traffic bit-rate in relay nodes which depends on current bandwidth.

4.3.5 Routing Algorithm based Optimization

The approach by EkbataniFard etal [21][27][28] uses then uses a genetic algorithm to find routes from source to sink that optimizes the QoS parameters, discussed above. An initial population is first constructed using depth first search algorithm. Using this population, an initial set of routing trees is constructed. Each of these routing trees is then encoded into chromosomes that represent typical routes, with integer representing a sensor node. These chromosomes then participate in mating (crossover) to generate new routes in the network. Figure 5 shows how two typical parents participate in a crossover to generate two child chromosomes typifying the generation of new routes from ones combinations of old ones.



Chromosome 1

Gene1	Gene2	Gene3
15	14	13
12	11	10
9	8	7
6	5	4
3	2	1
0	0	0

Chromosome 2

Gene1	Gene2	Gene3
15	14	13
11	10	10
8	7	8
5	4	4
3	2	1
0	0	0

Chromosome 1

Gene1	Gene2	Gene3
13	14	13
10	11	10
8	8	7
4	5	4
1	2	1
0	0	0

Chromosome 2

Gene1	Gene2	Gene3
15	14	15
11	11	12
8	7	9
5	4	6
3	2	3
0	0	0

Figure 5 Crossover operation

With a given probability *p*, mutation is carried out on carefully chosen nodes to ensure feasibility of the new paths. A new population $P_t = P_t \cup Q_t$ is formed where t is the number of generation.

5. Conclusion

This research reveals how genetic algorithms have been used to model sensor communication, in clustering and routing problems. We also provide a performance evaluation of various GA-based optimization strategies. Our observations shows that while a number of algorithms try to select the best cluster headers or routing path based on some metric, the process normally introduces overheads in communication which in turn leads to more energy dissipation. There is there for need for probabilistic approaches to predict the energy consumption in WSNs. Stochastic Network State Model may be can be used to model the behavior of sensor nodes and then predict energy consumption by a sensor node. The rationale behind this approach is that in many instances the node can predict its energy dissipation rate on its own past history and also on the past history of its neighbors. If a sensor can efficiently predict the amount of energy it will dissipate in future, then it won't be necessary to transmit available energy often.

References

[1] Nasser, Nidal, and Yunfeng Chen. "SEEM: Secure and energy-efficient multipath routing protocol for wireless sensor networks." *Computer Communications* 30, no. 11 (2007): 2401-2412.

[2] Akyildiz, Ian F., Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. "Wireless sensor networks: a survey." *Computer networks* 38, no. 4 (2002): 393-422.

[3] Olariu, Stephan, and Ivan Stojmenovic. "Design Guidelines for Maximizing Lifetime and Avoiding Energy Holes in Sensor Networks with Uniform Distribution and Uniform Reporting." In *INFOCOM*, pp. 1-12. 2006.

[4] Patil, Prakashgoud R., and Umakant P. Kulkarni. "Energy-Efficient Cluster-Based Aggregation Protocol for Heterogeneous Wireless Sensor Networks." In*Intelligent Computing, Networking, and Informatics*, pp. 129-141. Springer India, 2014.

[5] Talbi, El-Ghazali. *Metaheuristics: from design to implementation*. Vol. 74. John Wiley & Sons, 2009.

[6] Knowles, Joshua D. "Local-search and hybrid evolutionary algorithms for Pareto optimization." Department of Computer Science, University of Reading, 2002.

[7] Bäck, Thomas, David B. Fogel, and Zbigniew Michalewicz, eds. *Evolutionary computation 1: Basic algorithms and operators*. Vol. 1. CRC Press, 2000.

[8] Michalewicz, Zbigniew. *Genetic algorithms+ data structures= evolution programs*. springer, 1996.

[9] Haupt, Randy L., and Sue Ellen Haupt. *Practical genetic algorithms*. John Wiley & Sons, 2004.

[10] Larrañaga, Pedro, and Jose A. Lozano, eds. *Estimation of distribution algorithms: A new tool for evolutionary computation*. Vol. 2. Springer, 2002.

[11] Eiben, Agoston E., and James E. Smith. *Introduction to evolutionary computing*. springer, 2003.

[12] Falkenauer, Emanuel. "A new representation and operators for genetic algorithms applied to grouping

problems." *Evolutionary computation* 2, no. 2 (1994): 123-144.

[13] Stankovic, John A., Tarek F. Abdelzaher, Chenyang Lu, Lui Sha, and Jennifer C. Hou. "Real-time communication and coordination in embedded sensor networks." *Proceedings of the IEEE* 91, no. 7 (2003): 1002-1022.

[14] Jin, Shiyuan, Ming Zhou, and Annie S. Wu. "Sensor network optimization using a genetic algorithm." In *Proceedings of the 7th World Multiconference on Systemics, Cybernetics and Informatics*, pp. 109-116. 2003.

[15] Jia, Jie, Xueli Wu, Jian Chen, and Xingwei Wang. "Exploiting sensor redistribution for eliminating the energy hole problem in mobile sensor networks." *EURASIP Journal on Wireless Communications and Networking*2012, no. 1 (2012): 1-11.

[16] Loo, Jonathan, Jaime Lloret Mauri, and Jesús Hamilton Ortiz, eds. *Mobile Ad Hoc Networks: Current Status and Future Trends*. CRC Press, 2012.

[17] Kim, Jong-Myoung, Seon-Ho Park, Young-Ju Han, and Tai-Myoung Chung. "CHEF: cluster head election mechanism using fuzzy logic in wireless sensor networks." In *Advanced communication technology, 2008. ICACT 2008. 10th international conference on*, vol. 1, pp. 654-659. IEEE, 2008.

[18] Bayraklı, Selim, and Senol Zafer Erdogan. "Genetic Algorithm Based Energy Efficient Clusters (GABEEC) in Wireless Sensor Networks." *Procedia Computer Science* 10 (2012): 247-254.

[19] Fonseca, Carlos M., and Peter J. Fleming. "Genetic Algorithms for Multiobjective Optimization: FormulationDiscussion and Generalization." In*ICGA*, vol. 93, pp. 416-423. 1993.

[20]Deb, Kalyanmoy. "Introduction to evolutionary multiobjective optimization." In*Multiobjective Optimization*, pp. 59-96. Springer Berlin Heidelberg, 2008.

[21] Martins, Flávio VC, Eduardo G. Carrano, Elizabeth F. Wanner, Ricardo HC Takahashi, and Geraldo Robson Mateus. "A dynamic multiobjective hybrid approach for designing wireless sensor networks." In *Evolutionary Computation, 2009. CEC'09. IEEE Congress on*, pp. 1145-1152. IEEE, 2009.

[22] EkbataniFard, G. Hossein, Reza Monsefi, M-R. Akbarzadeh-T, and Mohammad Yaghmaee. "A multi-objective genetic algorithm based approach for energy efficient QoS-routing in two-tiered wireless sensor networks." In *Wireless Pervasive Computing (ISWPC), 2010 5th IEEE International Symposium on*, pp. 80-85. IEEE, 2010.

[23] Krishnamachari, Bhaskar. *Networking wireless sensors*. Cambridge University Press, 2005.

[24] Ezhumalai, Periyathambi, S. Manoj Kumar, Chokkalingam Arun, and D. Sridaharan. "Tree Based Aggregation Algorithm Design Issues in Wireless Sensor Networks." *Wireless Sensor Network* 1, no. 4 (2009).

[25] Malada, Awelani. "Stochastic reliability modelling for complex systems." PhD diss., 2007.

[26] EkbataniFard, GholamHossein, and Reza Monsefi. "A Fast Multi-objective Genetic Algorithm based Approach for Energy Efficient QoS-Routing in Two-tiered Wireless Multimedia Sensor Networks." *Modern Applied Science* 4, no. 6 (2010).

[28] Yen, Yun-Sheng, Yi-Kung Chan, Han-Chieh Chao, and Jong Hyuk Park. "A genetic algorithm for energy-efficient based multicast routing on MANETs."*Computer Communications* 31, no. 4 (2008): 858-869.