A Pareto optimization-based approach to clustering and routing in Wireless Sensor Networks

Article *in* Journal of Network and Computer Applications · April 2018

DOI: 10.1016/j.jnca.2018.04.005

A Pareto optimization-based approach to Clustering and Routing in Wireless Sensor Networks

Riham Elhabyan, Wei Shi, Marc St-Hilaire

School of Information Technology, Faculty of Engineering and Design
Carleton University, Ontario, Canada
{*riham.elhabyan, wei.shi, marc.sthilaire*}*@carleton.ca*

Abstract

Clustering and routing in WSNs are two well-known optimization problems that are classified as Non-deterministic Polynomial (NP)-hard. In this paper, we propose a single multi-objective problem formulation tackling these two problems simultaneously with the aim of finding the optimal network configuration. The proposed formulation takes into consideration the number of Cluster Heads (CHs), the number of clustered nodes, the link quality between the Cluster Members (CMs) and CHs and the link quality of the constructed routing tree. To select the best multi-objective optimization method, the formulated problem is solved by two state-of-the-art Multi-Objective Evolutionary Algorithms (MOEAs), and their performance is compared using two well-known quality indicators: the hypervolume indicator and the Epsilon indicator. Based on the proposed problem formulation and the best multi-objective optimization method, we also propose an energy efficient, reliable and scalable routing protocol. The proposed protocol is developed and tested under a realistic communication model and a realistic energy consumption model that is based on the characteristics of the Chipcon CC2420 radio transceiver data sheet. Simulation results show that the proposed protocol outperforms the other competent protocols in terms of the average consumed energy per node, number of clustered nodes, the throughput at the BS and execution time.

Keywords: WSN, Clustering, Multi-hop routing, Pareto optimization, RSSI, CC2420

1. Introduction

Wireless Sensor Networks (WSNs) have emerged as a key technology in realizing many applications in a wide range of contexts including military operations, environmental monitoring, surveillance systems, healthcare, environmental monitoring and public safety. In order to realize the existing and potential applications for WSNs, sophisticated and extremely efficient routing protocols are needed. However, it is a challenging task to select or propose a new routing protocol for a specific WSN application due to the inherent properties of the individual sensor nodes such as the limited power supply and the limited transmission range [1].

Cluster-based routing (clustering) protocols can solve some of those challenges due to their scalability, energy-efficiency, and data delivery reliability [2, 3]. In a clustering protocol, the network operating time is divided into rounds and each round is usually divided into three phases: Cluster Head (CH) selection, cluster formation, and data transmission. The CH selection algorithm is responsible for selecting the optimal set of CHs according to some predefined objectives. After selecting the optimal set of CHs, the clusters are usually formed by associating each regular node to its nearest CH. The data transmission phase can either be intra-cluster or inter-cluster. Intra-cluster communication comprises the data transmission between the member nodes and their respective CH while Inter-cluster communication includes transmission of data between the CHs or between a CH and the Base Station (BS).

Data delivery reliability is considered a key requirement in WSNs [4]. In order to realize this requirement, clustering protocols should adopt a multi-hop inter-cluster communication model as it is considered a more realistic approach due to the limited transmission range of the sensor nodes. On the other hand, using a single-hop inter-cluster communication model, in which the CHs send the data directly to the BS, can adversely affect the network performance due to collisions and communication interference [5]. Therefore, clustering protocols should ensure high-quality links between the cluster members and their associated CHs.

Several link quality-based clustering protocols proposed so far use the distance between two nodes as a metric of their link's quality. However, several studies have shown that distance is not necessarily correlated with link quality as it ignores the link asymmetry characteristic of WSNs [6]. Two other prominent link-quality metrics are the received signal strength indicator (RSSI) and link quality indicator (LQI). These two metrics are provided by most of the wireless sensor chips [7]. The RSSI is a parameter that represents the signal strength observed at the receiver at the moment of reception of the packet. The LQI is described as the characterization of the strength and quality of the received packets. Several studies prove that RSSI can provide a quick and accurate estimate of whether a link is of very good quality [8].

Another main problem in most of the proposed clustering protocols is the use of the first order radio model [9] to model the energy consumption of the sensor nodes. This model assumes that all the sensor nodes can communicate regardless of the distance between them. Moreover, it ignores the energy consumption due to the different states of the sensors especially the listening energy consumption, which is known to be the largest contributor to expended energy in WSNs. Therefore, this energy model is considered an ideal [10] and is fundamentally flawed for modeling the energy consumption in sensor networks. A discrete radio model should be used for more accurate and realistic calculation of the power consumption and to determine which links between sensor nodes are available for transmission.

Both the problems of selecting the optimal set of CHs and finding the optimal inter-cluster routing tree have been proved to be Non-deterministic Polynomial (NP)-hard optimization problems [11]. Moreover, they consist of multiple conflicting objectives that need to be optimized simultaneously. Pareto-based optimization techniques can be used to solve the CHs selection problem especially when the number of CHs is not fixed. For example, clustering can provide an energy-efficient solution if only a small number of CHs is involved in doing the main operations in the network such as routing, management, and data aggregation. However, minimizing the number of CHs may lead to a reduced number of clustered nodes and hence hindering the clustering protocol scalability. Another objective to consider concurrently is the inter-cluster communication cost which greatly affects the data delivery reliability.

1.1. Contributions

In this paper, we adopt a centralized Pareto optimization-based approach to design and develop an energy-efficient, scalable and reliable clustering and routing protocol. The objective of the proposed approach is to assign each network node to its respective CH and to find the optimal routing tree that connects the CHs to the BS. The novelty of the proposed protocol lies in proposing a new individual encoding scheme that allows for the mutual optimization of both the problems of clustering and routing in WSNs. Moreover, The protocol is designed and tested under realistic network settings, in terms of both the energy consumption model and the communication model. The main contributions of this paper include:

- Formulating jointly the clustering and routing problems in WSNs as a multi-objective minimization problem rather than dealing with them as two separate problems. The proposed formulation uses a non-predefined number of CHs and aims at determining an energy efficient, reliable and scalable clustering and routing protocol. In order to realize such formulation, a new individual encoding scheme that represents a joint solution for both the clustering and routing problems in WSNs is proposed. Moreover, a repair function is proposed to correct any invalid routing tree and to guide the search towards the optimal routing tree. In order to choose the best multi-objective optimization method, the formulated problem has been solved by two state-of-the-art Multi-Objective Evolutionary Algorithms (MOEA), and their performance has been compared using some quality indicators.
- Developing the protocol under realistic network settings. No assumptions are made about the location awareness or transmission range capabilities of the sensor nodes. In this paper, the RSSI value for the link between any two nodes is used to assess the quality of that link. The protocols are also tested using a realistic discrete energy consumption model that is based on the characteristics of the Chipcon CC2420 radio transceiver's data sheet.

According to the initial energy of the sensors, we consider two types of networks: homogeneous and heterogeneous networks. Extensive simulations on 25 homogeneous as well as 25 heterogeneous WSN networks are conducted. Results are evaluated and compared against several well-known clustering protocols. Simulation results show that the proposed protocol outperforms the other competent protocols in terms of average consumed energy per node, number of clustered nodes, throughput at the BS and execution time.

1.2. Paper Organization

The remainder of this paper is organized as follows. In Section 2, we review the related work about clustering protocols. A brief introduction of the concept of Pareto-based multi-objective optimization is given in Section 3. Section 4 presents the system model and the problem formulation. Section 5 gives a detailed description of the newly proposed encoding scheme. A detailed comparative evaluation of the proposed protocol against other works is reported in Section 6. Finally, Section 7 concludes this research work and highlights few future directions.

2. Related Work

Clustering techniques have been studied extensively in the literature to improve the performance of WSNs [2].

The Low Energy Adaptive Clustering Hierarchy (LEACH) protocol [12, 9] is the first and one of the most common cluster-based routing protocols for WSNs. LEACH is a completely distributed protocol where each node decides whether to become a CH for the current round based on its local information. In order to take this decision, each sensor node uses a probabilistic calculation which is based on the suggested percentage of CHs for the network and the number of times the node has been a CH so far. This is done to equalize the energy load distribution among the CHs. The basic idea of LEACH has been an inspiration for many subsequent clustering protocols [13] that are designed for heterogeneous networks such as the Energy Efficient Heterogeneous Clustered (EEHC) scheme [14], the Enhanced Heterogeneous LEACH Protocol for Lifetime Enhancement (EHE-LEACH) [15], and the Single-hop and Multi-hop Energy-Efficient Clustering Protocols (S-EECP) and (M-EECP) [16]. The Energy Efficient and Dynamic Clustering (EEDC) protocol [17] is a distributed protocol. In EEDC, each sensor estimates the number of active nodes in the network by monitoring the received signal power from its neighboring. Based on that estimation, each sensor computes its optimal probability of becoming a CH. In addition to that, an Energy Efficient and Power-Aware (EEPA) routing protocol is proposed to connect the CHs to the BS. In EEPA, each CH uses flooding to request a route to the BS. The BS then computes many candidate routes and uses flooding to send these routes to the CH. The CH then chooses one of these routes based on its current battery level. All these protocols adopt a probabilistic approach for electing the CHs and yield a different number of CHs each round. However, in these protocols, each sensor decides whether it will become a CH based on its local information. This, in turn, does not guarantee the selection of the optimal set of CHs [18], mainly due to the lack of global knowledge about the network.

The literature is also rich with studies that propose EA-based clustering protocols to achieve the optimal set of CHs. In such protocols, the BS utilizes an EA algorithm and its global knowledge of the network to select the optimal set of CHs, based on a set of predefined objective functions.

A centralized version of LEACH (LEACH-C) is proposed in [9]. LEACH-C employs a Simulated Annealing (SA) approach to find a predetermined number of CHs and to configure the network into clusters. The objective function is defined to minimize the amount of energy for the non-CH nodes to transmit their data to the CH, by minimizing the total sum of squared distances between all the non-CH and the closest CH.

A centralized Particle Swarm Optimization (PSO)-based clustering protocol (PSO-C) is proposed in [19]. The objective function is defined to minimize both the maximum average Euclidean distance of nodes to their associated CHs and the ratio of the total initial energy of all nodes to the total energy of the CH candidates. PSO-C assumes that the CHs can communicate with the BS directly and takes into consideration the cost of both the inter-cluster communication and the network's energy efficiency. However, the sub-objectives of PSO-C are not scaled, hence it is hard to determine the optimal weight coefficient for each sub-objective. This may lead the optimal solution to be biased toward one of the sub-objectives than the other. The Genetic Algorithm (GA)-based clustering protocol (GA-C) [20] is similar to PSO-C in terms of the objective function. However, GA-C adopts a GA approach to find the optimal set of CHs.

Particle Swarm Optimization Protocol for Cluster Heads (PSO-CH) protocol [10] is another centralized PSO-based protocol that is used to find the optimal set of CHs. The PSO-CH protocol considers the following properties: the network's energy efficiency, data transmission reliability, and the protocol's scalability. The objective function used to evaluate each individual particle is defined as the weighted sum of three sub-objectives, each of which is related to the aforementioned properties. After calculating the sub-objectives, they are scaled to avoid any bias. PSO-CH is designed and implemented under realistic networks settings and realistic energy consumption model. No assumptions are made about the nodes' location awareness or transmission range capabilities. However, PSO-CH does not solve the inter-cluster communication problem.

Two-tier Particle Swarm Optimization for Clustering and Routing (TPSO-CR) protocol [21] is a PSO-inspired protocol proposed mainly to solve the routing tree construction problem for clustered WSNs. The protocol runs in two tiers: the first tier finds the best CHs and their associative clusters using the PSO-CH protocol [10] while the second tier solves the problem of the inter-cluster communication by finding the optimal routing tree. To achieve an energy-efficient routing tree, two sub-objectives need to be met: the first one minimizes the total number of active nodes including both the CHs and relay nodes while the second one favors the nodes with higher residual energy to act as relay nodes. To maximize the PDR, the protocol maximizes the link quality between the relay nodes in the routing tree. TPSO-CR is a centralized weighted-sum PSO-based protocol that is proposed mainly for finding the optimal inter-cluster routing tree. This protocol is appropriate when the CHs are predetermined in advance. It uses a particle encoding scheme and defines an objective function to find the optimal routing tree. The objective function is used to build the trade-off between the energy-efficiency and data delivery reliability of the constructed tree.

Improved Cuckoo Search and Harmony Search (iCSHS) [22] is an integrated clustering and routing protocol for WSN. Similar to TPSO-CR, the iCSHS protocol runs in two tiers. The first tier uses a cuckoo search-based optimization algorithm to find the optimal set of CHs. This algorithm uses an objective function that considers the following parameters: the sensor: remaining energy, the number of neighbors, the intra-cluster distance and the number of unclustered sensors. In the second tier, an improved harmony search based routing algorithm is proposed to find the optimal inter-cluster routing tree.

Many of the aforementioned distance-based protocols assume that the sensor nodes are aware of their position. However, to find the distance between two nodes, each node should be equipped with self-locating hardware such as a global positioning system (GPS). The resulting cost of attaching a GPS to each node renders such a solution inefficient and unrealistic [23]. In addition to the previously mentioned problems, and to the best of our knowledge, most of the clustering protocols that are proposed so far use the first order radio model [9] to calculate the energy consumption of the sensor nodes. However, this energy model is very idealized [10, 24, 25]. It assumes that all the sensor nodes communicate regardless of the distance between them. Moreover, it ignores the listening energy consumption, which is known to be the largest contributor to the overall energy consumption in WSNs. A discrete radio model should be used for more accurate and realistic calculation of the power consumption and to determine which links between sensor nodes are available for transmission [24, 25, 26].

Although the TPSO-CR protocol does not suffer from these problems, the process of finding the optimal set of CHs and the routing tree takes a long time because it is divided into two tiers. Moreover, the inter-cluster communication is not limited to the CHs only. Other relay nodes may be added to the inter-cluster routing tree which may lead to more energy consumption as the number of active node increases. In this paper, we improve the

TPSO-CR protocol by jointly finding the optimal set of CHs and the optimal inter-cluster routing tree in one tier instead of two tiers to minimize the execution time. Moreover, in order to minimize the energy consumption, we limit the inter-cluster communication to the CHs only.

Table 1 provides a comparison of the clustering protocols mentioned above with respect to different clustering properties.

3. Pareto-based Multi-objective Optimization

Assuming a minimization problem for convenience, a Multi-objective Optimization Problem (MOP) with n decision variables and M objective functions can be expressed as follows: given an n-dimensional decision variable vector $x = \{x_1, ..., x_n\}$ in the solution space X find a vector x^* which yields the optimum value for a given set of M objective functions $z(x^*) = \{z_1(x^*), ..., z_M(x^*)\}$ where $M \geq 2$.

However, due to the conflicting nature of the objective functions, it is rare that the global optimum for all of the individual objective functions occurs simultaneously at one single point of the search space. Instead, we are interested in finding a set of trade-off solutions. The most commonly adopted notion of optimality is the so-called Pareto optimality.

A feasible solution x is said to dominate another feasible solution y if and only if the following two conditions are true:

- Solution x is no worse than a solution y in all objectives.
- Solution x is strictly better than a solution y in at least one objective.

If any of the conditions mentioned above is false, then solution x does not dominate solution y. If solution x dominates solution y, then solution x is better than solution y.

Solution x^* is a *Pareto optimal solution* if there exists no feasible vector of decision variables $x \in X$, which would decrease some objective value without causing a simultaneous increase in at least one other objective value. There are no superior solutions to the problem than x^*, although there may be other equally good solutions. The set of solutions that satisfies this condition is known as the *Pareto optimal set*. A Pareto optimal set is a set of solutions that are non-dominated with respect to each other. The vector corresponding to the solutions included in the Pareto optimal set is called *non-dominated vector*. The plot of the objective functions whose non-dominated solutions are in the Pareto optimal set is called the *Pareto optimal front* [27] which corresponds to the trade-off surface in objective space.

The literature hosts several interesting approaches for tackling MOPs, with multi-objective evolutionary algorithms, posing all the desired characteristics for obtaining a set of non-dominated solutions, in a single run. These approaches work with two main goals:

- Convergence: find a set of Pareto-optimal solutions, and
- Diversity: find a set of diverse solutions in order to prevent premature convergence and achieve a well-distributed trade-off Pareto front.

The first goal guides the solutions towards the Pareto-optimal region and the second goal guides along the Pareto-optimal front.

In this paper, two different types of MOEAs are considered as the optimization tools to solve the joint problem of clustering and routing in WSNs:

- The Non-dominated Sorting Genetic Algorithm II (NSGA-II) and
- Speed-constrained Multi-objective Particle Swarm Optimization (SMPSO)

There are extensive applications of these two algorithms in different fields of WSNs [28]. They provide the most to the needs of practical optimization problems known to date [29]. These algorithms are popular also because of their ease of hardware implementation [29].

3.1. Non-dominated Sorting Genetic Algorithm II

NSGA-II [30] is a popular non-domination-based genetic algorithm for multi-objective optimization. It starts with producing a population that consists of random solutions (chromosomes). In each generation, the population in NSGA-II is sorted into several non-dominated fronts using a ranking algorithm first (non-dominated sorting). Then, individual solutions are selected from these non-dominated fronts by calculating the crowding distance. The crowding distance measures the distance between the individual solutions and the rest of the solutions in the population. If two individual solutions are in the same non-dominated front, the solution with a higher value of crowding distance will be selected. The crowding distance calculation is used to preserve the diversity among non-dominated solutions in the later stage of the run in order to obtain a good spread of solutions. After that, the algorithm applies the standard crossover and polynomial operators to combine the current population and its offspring generated as next generation. At last, the best individuals in terms of non-dominance and diversity are selected as solutions.

3.2. Speed-constrained Multi-objective Particle Swarm Optimization

Recently, Particle Swarm Optimization (PSO) is playing a very important role in MOPs because of its convergence speed and simple operators. The SMPSO algorithm [31] is based on the PSO theory.

An experimental comparison is conducted in [32] to assess the performance of SMPSO against six of the state of the art Pareto-based MOPSO representatives. Experimental results prove that SMPSO outperforms the other protocols in terms of the quality of results. Furthermore, SMPSO shows a remarkable performance in terms of other different assessment criteria [33]: convergence towards the optimum solutions [34], and scalability with the problem size [35].

Similar to NSGA-II, SMPSO selects best solutions by calculating crowding distance and also stores the selected individual solutions in an archive. SMPSO applies a polynomial mutation

Table 1: Comparison of clustering protocols with respect to clustering attributes

Clustering Protocol	Clustering Method	Clustering Approach	Location Awareness	Number of Cluster Heads	Connectivity to the BS	Energy Model	Link Quality Metric	Network Type	Protocol's Objectives EE	DDR	SC
LEACH [9]	Distributed	Prob./Random	No	Variable	One-hop	First Order	None	Homogeneous	✓	✗	✗
EEHC [14]	Distributed	Prob./Energy	No	Variable	One-hop	First Order	None	Homogeneous/ Heterogeneous	✓	✗	✗
EHE-LEACH [15]	Distributed	Prob./Energy	No	Variable	One-hop	First Order	None	Homogeneous/ Heterogeneous	✓	✗	✗
S-EEP [16]	Distributed	Prob./Energy	No	Variable	One-hop	First Order	None	Heterogeneous	✓	✗	✗
M-EEP [16]	Centralized	Greedy	Yes	Variable	One-hop/ Multi-hop	First Order	Distance	Homogeneous/ Heterogeneous	✓	✓	✗
LEACH-C [9]	Centralized	SA	Yes	Fixed	One-hop	First Order	Distance	Homogeneous	✓	✗	✗
PSO-C [19]	Centralized	PSO	Yes	Fixed	One-hop	First Order	Distance	Homogeneous	✓	✗	✗
GA-C [20]	Centralized	GA	Yes	Fixed	One-hop	First Order	Distance	Homogeneous	✓	✗	✗
PSO-CH [10]	Centralized	PSO	No	Fixed	One-hop	Discrete (CC2420)	RSSI	Homogeneous/ Heterogeneous	✓	✓	✓
TPSO-CR [21]	Centralized	PSO	No	Fixed	Multi-hop	Discrete (CC2420)	RSSI	Homogeneous/ Heterogeneous	✓	✓	✓

operator [36] to 15% of the population to accelerate the speed of convergence. In addition, SMPSO incorporates a velocity constriction procedure [37] to produce new effective particle positions in those cases in which the velocity becomes too high and hence avoid the swarm explosion problem [37].

3.3. Determining Best Compromise Individual

MOEAs provide a set of Pareto optimal solutions. Therefore, a mechanism is needed to determine the best compromise solution among those solutions. Due to the imprecise nature of the decision maker's judgement, it is assumed that there is fuzziness in the goal for each objective. This fuzziness is defined by membership functions that represent the degree of fuzziness of some fuzzy sets using values in the range [0, 1].

The fuzzy mechanism looks at the way the solutions are contributing to each objective and assigns a fuzzy variable. It shows a possible way of finding a compromise solution in case solutions are very close to each other. In this paper, a fuzzy-based mechanism [38] is used to find out a compromise solution on the Pareto front. This mechanism has been successfully used in numerous applications of MOEAs [29].

In the fuzzy-based mechanism, a membership value for i^{th} objective of j^{th} solution in the Pareto-front is calculated using the membership function as:

$$\mu_i^j = \begin{cases} 1 & \text{if } F_i \leq F_i^{min}. \\ \frac{F_i^{max} - F_i}{F_i^{max} - F_i^{min}} & \text{if } F_i^{min} < F_i < F_i^{max} \\ 0 & \text{if } F_i \geq F_i^{max}. \end{cases} \tag{1}$$

μ_i^j indicates how well the j^{th} solution in the Pareto optimal set can satisfy the i^{th} objective. The sum of membership values for all objectives of the j^{th} solution suggests how well it satisfies all the objectives.

Given N solutions in the Pareto-optimal set and M objective functions for each solution, the achievement of each non-dominated solution with respect to all non-dominated solutions can be calculated using:

$$\mu^j = \frac{\sum_{i=1}^{M} \mu_i^j}{\sum_{j=1}^{N} \sum_{i=1}^{M} \mu_i^j} \tag{2}$$

The solution with the maximum value of μ^j is a compromise solution that can be accepted by the decision maker.

In this paper, we employ a fuzzy-based mechanism to find the best compromise solution.

3.4. Performance Assessment of Different MOEAs

With the existence of different MOEAs, it is necessary to quantify the performance of each algorithm. A number of quality indicators are proposed in the literature for measuring both the convergence and the diversity of the obtained set of non-dominated solutions. The quality indicator method is the dominant method in the literature to assess the performance of different MOEAs [39]. It maps each Pareto set approximation to a number and performs statistics on the resulting distributions of numbers [39]. Some quality indicators require the knowledge of the true Pareto-optimal front that is unknown in this application. Instead, an approximation set to the optimal Pareto-front of the problem is computed. Taking this into account, the hypervolume indicator and the Epsilon indicator are adapted to access the performance of SMPSO and NSGA-II in this paper.

The Hypervolume Indicator. The hypervolume (HV) indicator is introduced in [40]. It has gained increasing interest in recent years and has become a popular indicator of the performance of different MOEAs [41, 42]. If solutions are considered as points in objective space, hypervolume is the n-dimensional space that is contained within a solution set, i.e. the n-dimensional volume of the set relative to some reference point, usually the anti-optimal point or worst possible point for the space. In other words, the hypervolume of a set is the total size of the space dominated by the solutions in the set. A set with a larger hypervolume is likely to represent a better set of trade-offs than sets with lower hypervolume. Algorithms with larger values of HV are desirable [43]. The hypervolume indicator measures both the convergence and diversity of the obtained Pareto-optimal front solutions simultaneously [44, 45].

The Epsilon Indicator. The Epsilon indicator is proposed in [46]. Given two sets of non-dominated solutions A and B, this indicator computes the minimum factor by which objectives of solutions in B can be multiplied so that the transformed set of non-dominated solutions is still weakly dominated by A. For the Epsilon indicator, the lower the value the better the computed fronts [45]. The Epsilon indicator takes into account measuring the convergence properties of the obtained Pareto-optimal front [45].

4. System Model and Problem Formulation

4.1. System Model

We consider a WSN with N sensor nodes, K CHs and one BS. Each sensor node has a unique ID and the BS ID is 0. We assume that all nodes are stationary after deployment and the locations of both sensor nodes and CHs are unknown.

A realistic energy consumption model which is based on the characteristics of the Chipcon CC2420 radio transceiver data sheet [47] is used. The total energy consumed by sensor S with ID n, $E(S_n)$, is calculated as follows [48]:

$$E(S_n) = \sum_{state\,j} P_{state\,j} \times t_{state\,j} + \sum_{tr} E(transitions) \tag{3}$$

The index $state\,j$ refers to the energy states of the sensor: sleep, reception, or transmission. $P_{state\,j}$ is the power consumed in each $state\,j$, $t_{state\,j}$ is the time spent in the corresponding state, and tr is the number of transitions for n_d. The energy spent in transitions between states, $E_{transitions}$, is also added to the node's total energy consumption. The different values of $P_{state\,j}$ and $E_{transitions}$ can be found in [47].

4.2. Problem Formulation

The main goal of the protocol is to find the optimal set of CHs such that the following objectives are achieved concurrently:

- Maximize the protocol's energy efficiency by minimizing the average consumed energy per node which in turn maximizes the network lifetime.
- Maximize the protocol's scalability by maximizing the number of clustered nodes.
- Maximize the protocol's data delivery reliability by maximizing the network throughput.

The joint clustering and routing problem is formulated as a multi-objective minimization problem. The objective functions are constructed to simultaneously evaluate each candidate solution, based on the following objective functions. The main notations used in this problem formulation are presented in Table 2.

Table 2: Table of Notations

Notation	Description
CH_k	CH number k
S_n	Sensor number n
K	Total number of elected CHs
$NextHop_{CH_k}$	Next hop for CH_k
$E(CH_k)$	Remaining energy of CH_k
V	The vector containing the elected CHs
N	Total number of sensors in the network
$cm_{m,k}$	Cluster member number m of cluster k
NCH_{THR}	Number of CHs threshold
RSS_{THR}	Link quality threshold based on the RSSI value
$w_{s \to d}$	Weight of the link from the sender sensor s to the destination sensor d
C_k	The vector containing the cluster members in the cluster that corresponds to CH_k

4.2.1. Energy Efficiency

To save more energy, fewer sensor nodes need to be active during each round. Our main approach to achieving that is to minimize the number of elected CHs, given by Eq. 4.

$$K = |V| \tag{4}$$

Constraint 5 ensures that the total number of CHs should not exceed a prespecified threshold, NCH_{THR}. In our experiments, we set NCH_{THR} = 10% of the total number of sensors in the network.

$$K < NCH_{THR} \tag{5}$$

Furthermore, a sensor node with a higher level of energy is a better CH candidate to both aggregate the data and to act as a relay node towards another CH or BS. This is achieved by minimizing Eq. 6.

$$K \Big/ \sum_{k=1}^{|V|} E(CH_k) \tag{6}$$

Constraint 7 ensures that only the sensors with sufficient remaining energy are selected as CHs.

$$\forall \quad CH_k \in V, \quad E(CH_k) > \sum_{n=1}^{N} E(S_n) \Big/ N \tag{7}$$

4.2.2. Scalability

To increase the protocol's scalability, the clustering process should cluster as many sensor nodes as possible. This, in turn, will reduce the chance of creating clusters with one node only. This is achieved by minimizing Eq. 8.

$$N - \sum_{k=1}^{|V|} |C_k| \tag{8}$$

4.2.3. Data Delivery Reliability

To increase the data delivery reliability, two objectives need to be considered simultaneously:

- Minimize the cost of the intra-cluster communication.
- Minimize the cost of the inter-cluster communication.

It should be noted that the cost of the link between any two nodes is given as link weights in the adjacency matrix, D_t. The next section gives a detailed description of how D_t is constructed. The intra-cluster communication cost is defined as the total cost of the links between all the cluster members and their correspondent CHs. The total cost of the constructed tree, the inter-cluster communication cost, is defined as the sum of the costs of the links between the CHs forming that tree. Both the intra-cluster communication cost and the inter-cluster communication cost are optimized using Eq. 9 and Eq. 10 respectively.

$$\sum_{k=1}^{|V|} \sum_{m=1}^{|C_k|} w_{cm_{m,k} \rightarrow CH_k} \tag{9}$$

$$\sum_{k=1}^{|V|} w_{CH_k \rightarrow NextHop_{CH_k}} \tag{10}$$

It should be noted that in the case that any two CHs are not connected, the constructed tree is assigned a high penalty value to narrow the search to optimal valid tree solutions only.

5. Individuals Encoding/Decoding Scheme

In this section, we present a new individual encoding/decoding scheme to represent a joint solution for both the clustering and routing problems in WSN.

5.1. CHs Selection and Clusters Formation

The individuals are presented in such a way that each individual provides the optimal set of CHs and the route from each CH to the BS. The dimension of an individual is equal to the number of sensor nodes in the network (i.e., N). Let, $I_i = [X_{i,1}, X_{i,2}, X_{i,3}, ..., X_{i,N}]$ be the i_{th} individual of the population where each component, $X_{i,d}, 1 \leq d \leq N$ maps the assignment of the sensor node n_d to a CH. Each component is initialized with a randomly generated number in the range $[0.0, 1.0]$ based on a uniform distribution. Let $Nbrs(n_d)$ be the list of all n_d neighbours. Then, for individual I_i, the CH of node n_d is encoded as follows: $CH_{n_d} = \lceil (X_{i,d} \times |Nbrs(n_d)|) \rceil$

To illustrate how this encoding scheme works, consider a WSN with 20 sensor nodes, i.e., $N = \{n_0, n_2, ..., n_{19}\}$ where n_0 is the BS as shown in Figure 1. Therefore, the dimension of the individual is same as the number of sensor nodes minus the BS, i.e., 19. The edge $u \rightarrow v$ indicates that node v is within communication range of node u hence node u can send to node v but not necessarily vice versa.

Now, for each $X_{i,d}, 1 \leq d \leq 19$ of individual I_i , a random number is generated to initialize it. Let us assume that an individual $I_i = [1.00, 0.79, 0.20, 0.43, 0.71, 0.62, 0.61, 0.74, 0.11,$ $0.29, 0.29, 0.33, 0.18, 0.60, 0.46, 0.47, 0.24, 0.57, 0.77]$ has been randomly generated as shown in column 4 (i.e., $X_{i,d}$) of Table 3.

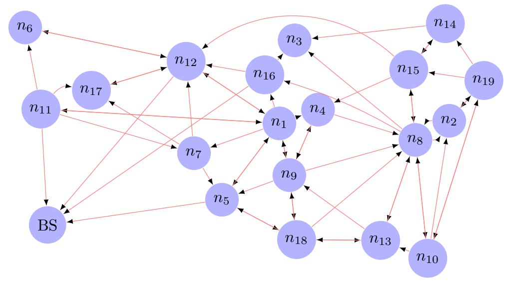

Figure 1: A WSN with 20 sensor nodes

We show that this individual actually represents a candidate solution to both the clustering and routing problems as follows.

Let's consider the generated random number for the first component, 1.00, i.e., $X_{i,1} = 1.00$ as shown in row 1 of Table 3. Hence, $\lceil (X_{i,1} \times |Nbrs(n_1)|) \rceil = 7$, therefore the 7th neighbour from $Nbrs(n_1)$, i.e., n_{12} is selected as a CH for n_1 as shown in Table 3. In the same way, each sensor node is assigned to a CH using the randomly generated particle. Then, the CH candidates that result from decoding I_i is $V_i = \{n_{12}, n_{15}, n_8, n_0\}$. Table 3 summarizes the decoding process for individual I_i.

Table 3: Particle decoding process

n_i	$Nbrs(n_i)$	$\lvert Nbrs(n_i)\rvert$	$X_{i,d}$	$\lceil (X_{i,d} \times \lvert Nbrs(n_i)\rvert) \rceil$	CH
n_1	$\{n_9, n_7, n_{16}, n_4, n_{11}, n_5, n_{12}\}$	7	1.00	7	n_{12}
n_2	$\{n_{19}, n_8\}$	2	0.79	2	n_8
n_3	$\{Null\}$	0	0.20	0	None
n_4	$\{n_8, n_9\}$	2	0.43	1	n_8
n_5	$\{n_{18}, n_0, n_1\}$	2	0.71	2	n_0
n_6	$\{n_{12}\}$	1	0.62	1	n_{12}
n_7	$\{n_5, n_{12}, n_{17}\}$	3	0.61	2	n_{12}
n_8	$\{n_3, n_2, n_{10}, n_{16}, n_{15}, n_{13}\}$	6	0.74	5	n_{15}
n_9	$\{n_8, n_1, n_4, n_5, n_{18}\}$	5	0.11	1	n_8
n_{10}	$\{n_2, n_8, n_{13}, n_{19}\}$	4	0.29	2	n_8
n_{11}	$\{n_6, n_{17}, n_7, n_0\}$	4	0.92	4	n_0
n_{12}	$\{n_1, n_0, n_{17}, n_6\}$	4	0.33	2	n_0
n_{13}	$\{n_8, n_9, n_{18}\}$	3	0.18	1	n_8
n_{14}	$\{n_3, n_{15}\}$	2	0.60	2	n_{15}
n_{15}	$\{n_{14}, n_8, n_4, n_{12}\}$	4	0.46	2	n_8
n_{16}	$\{n_3, n_{12}, n_0\}$	3	0.47	2	n_{12}
n_{17}	$\{n_{12}\}$	1	0.24	1	n_{12}
n_{18}	$\{n_{13}, n_9, n_8, n_5\}$	4	0.57	3	n_8
n_{19}	$\{n_{14}, n_{10}, n_2, n_{15}\}$	4	0.77	4	n_{15}

The final assignment of each node to its next hop, for individual I_i, and the respective clusters are shown in Figure 2.

5.2. Routing Tree Construction

The inter-cluster communication is used to carry data from the CHs to the BS. In the proposed approach, a multi-hop model where the CHs form a network among themselves, with each CH node using a multi-hop route for routing data towards the BS.

It should be noted that using the proposed individual encoding scheme also results in the routing tree construction by assigning each CH to its next hop. However, the constructed routing tree is considered not valid if any of the following conditions is violated:

- The constructed routing tree is loop-free.

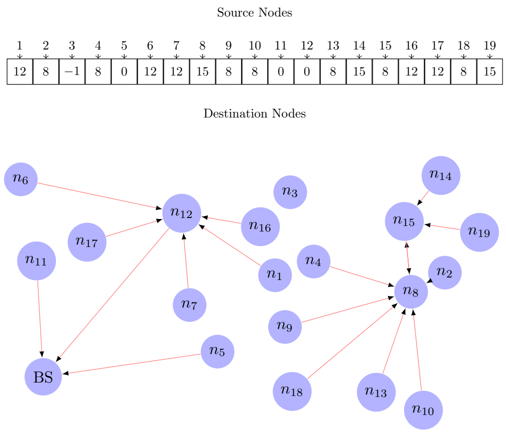

Figure 2:
The assignment of the nodes to their respective next hop

- Each route from each CH should terminate at the BS.

Otherwise, the constructed routing tree is valid and can be used for the inter-cluster communication. An example of a non valid routing tree is illustrated below.

Let's consider the final nodes assignments shown in Figure 2 and the generated CHs set, $V_i = \{n_{12}, n_{15}, n_8, n_0\}$. We notice that CH n_{12} can send to the BS node directly. However, there is no route from CH n_{15} to the BS and there is a loop between CH n_8 and CH n_{15}. Hence, the constructed routing tree is considered not valid and should be assigned a high penalty fitness value to exclude this solution from further consideration.

Now, let's suppose that the generated random number for n_{15} is 0.96, i.e. $X_{i,15} = 0.96$ in Table 3, then node n_{15} will be assigned to node n_{12} instead of n_8. In this case, the final valid nodes assignment to their next hop and the generated clusters are shown in Figure 3. This assignment will result in the following routes from each CH, $n_{12} \rightarrow n_0$, $n_{15} \rightarrow n_{12} \rightarrow n_0$ and $n_8 \rightarrow n_{15} \rightarrow n_{12} \rightarrow n_0$. This routing tree is considered valid since there are no loops between the CHs and each route from each CH terminates at the BS. The corresponding routing tree is illustrated in Figure 3.

Throughout different experimentation, it is found that, regardless of the network density, the proposed approach results in a large number of non-valid routing trees due to existing loops. Hence, there is a need for a repair function to repair the constructed routing tree.

Repair Function based on Dijkstra Algorithm. In order to construct a loop-free routing tree, the Dijkstra algorithm is used to find the shortest path tree (SPT) that connects the CHs to the BS.

Let's assume that the network is presented as a weighted directed graph, $G = (V, E)$, where V represents the set of CHs in addition to the BS and E represents the set of edges between them. It is assumed that the best link quality between two nodes represents the shortest path. An edge e from node u to v has

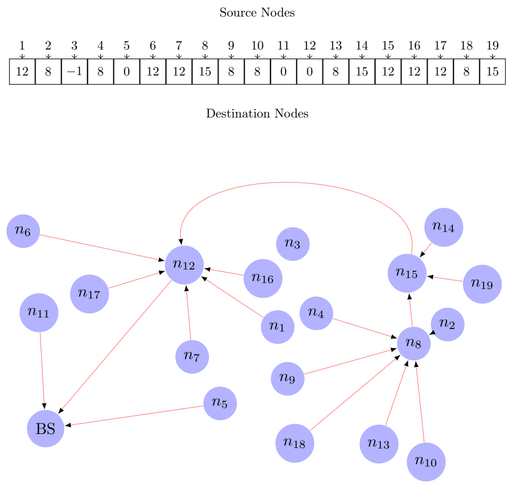

Figure 3:
Valid assignment of the nodes to their respective next hop

weight $w_{u\rightarrow v}$, given by:

$$w_{u\rightarrow v} = \begin{cases} \frac{RSSI_{u\rightarrow v}}{worstRSSI} & \text{if } v \text{ is neighbor of } u. \\ 0 & \text{if } u{=}v \\ INF & \text{otherwise} \end{cases} \tag{11}$$

where $RSSI_{u\rightarrow v}$ represents the $RSSI$ for the link from u to v. $worstRSSI$ represents the worst $RSSI$ value between any to nodes, and is set to -100. INF represents a very high weight value.

The link quality for the route R from CH ch to the BS is calculated as follows:

$$LQ_{ch\rightarrow BS} = \begin{cases} \sum_{e\in R} w_e & \text{If BS is reachable from } ch \\ INF & \text{otherwise} \end{cases} \tag{12}$$

It should be noted that the link quality from sensor u to sensor v is different from the link quality from sensor v to sensor u. Therefore, at iteration t, the BS generates a dynamic adjacency matrix D_t as follows:

$$D_t = \begin{pmatrix} 0 & w_{1\rightarrow 2} & \cdots & w_{1\rightarrow n} \\ w_{2\rightarrow 1} & 0 & \cdots & w_{2\rightarrow n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n\rightarrow 1} & w_{n\rightarrow 2} & \cdots & 0 \end{pmatrix}$$

The Dijkstra algorithm that is used to find the SPT that connects the CHs to the BS is shown in Algorithm 1.

6. Simulation Results

In this section, the results of the simulation that are employed to evaluate the proposed approach are presented. The goals of the simulation are to:

Algorithm 1: The Dijkstra algorithm used to find the SPT

Input: The directed graph $G = (V, E)$ and the positive edge lengths $\{w_e : e \in E\}$ given by D_t.
Output: The SPT and its associated cost

```
// For each the CH, ch ∈ V, lq[ch] is the link
   quality for the route from ch to the BS
   and calculated by 12.  The SPT cost is
   calculated using 10.
// Q: Set of unvisited vertices
lq[BS] = 0
prev[BS] = null
foreach ch ∈ V do
    if ch! = BS then
        lq[ch] = ∞
        prev[ch] = null
    end
    add ch to Q
end
while Q is not empty do
    u ← vertex in Q with minimum rssi(u) value
    remove u from Q
    foreach neighbour v of u do
        alt ← lq[u] + w_{v→u}
        if alt < lq[v] then
            lq[v] ← alt
            prev[v] ← u
        end
    end
end
return prev[], Σ_{ch∈V} lq[ch]
```

- Evaluate the performance of applying both NSGA-II and SMPSO on the formulated joint clustering and routing problem.
- Evaluate the performance of the proposed protocol against the well-known clustering protocols LEACH, EHE-LEACH, EEHC, LEACH-C, PSO-C, and GA-C.
- Evaluate the performance of the proposed protocol against TPSO-CR. We provide a separate comparison between our proposed protocol and TPSO-CR because, similar to our protocol, TPSO-CR is implemented under realistic network settings and both provide a dedicated routing tree to deliver the data to the BS.

Simulations are carried using the Castalia Simulator. We test the performance of the proposed protocol for both homogeneous and heterogeneous networks. According to the initial energy of the sensors, the simulations are performed on two groups of WSNs, $WSNs\#1$ and $WSNs\#2$, each with 25 different playground topologies. The first case assumes homogeneous sensor networks referred to as $WSNs\#1$ while the second experiments referred to as $WSNs\#2$ assume heterogeneous sensor networks. There are three types of nodes in $WSNs\#2$: a normal node in which the initial energy is set to 6240 joules, a super node in which the initial energy is set to 12480 joules, and an advanced node in which the initial energy is set to 18720 joules. $WSNs\#2$ consists of 10% advanced nodes, 10% super nodes and the rest of the nodes are normal nodes. Each WSNs group consists of 5 different network sizes ranging from 100 to 500 sensor nodes. Overall, the simulation results presented here are the average over five simulation runs for each network size, for a total of 50 different networks. The sensor nodes are deployed randomly in a sensor field of $100m \times 100m$. The BS is located at the field's corner at position $(0, 0)$. TMAC that is known for its energy efficiency is used as a medium access control because it adopts a variable sleep schedule that increases the battery utilization [49].

6.1. Performance Evaluation of NSGA-II and SMPSO

In this subsection, the performance results of applying both NSGA-II and SMPSO, on the formulated joint clustering and routing problem, are compared. To evaluate the performance of both algorithms, fifty independent runs using different random seeds are performed for a random round of $WSNs\#2$. The parameters setting of NSGA-II and SMPSO is given in Table 4.

Table 4: Parameters setting of NSGA-II and SMPSO

Parameter	**Value**
Problem dimension	$NetworkSize - 1$
NSGA-II Parameters Settings	
Population size	100
Number of iterations	250
Crossover probability	0.9
Crossover distribution index	20
Mutation probability	1.0/Problem dimension
Mutation distribution index	20
SMPSO Parameters Settings	
Swarm Size	100
Archive Size	100
Number of iterations	250
Mutation probability	1.0/Problem dimension
Mutation distribution index	20

It is necessary to increase the population's size and to run additional iterations when we want to solve the same problem with an increased dimension. However, it is rather difficult to predict the population's size and the number of evaluations required to solve a problem of known dimension [50]. Besides, it is of minor importance to tune this parameter based on the problem at hand [51]. Full analysis and determination of the optimal population size are beyond the scope of this paper. The aforementioned parameters are the default parameters which are provided by the adopted multi-objective framework (the JMetal Framework).

The capability of NSGA-II and SMPSO in comparison to each other is measured using two quality indicators, namely, the hypervolume indicator and the Epsilon indicator, explained in Section 3.4. Tables 5 and 6 show the comparisons of the hypervolume and Epsilon indicators respectively, for different network sizes. The results are in the form $Mean_{StandardDeviation}$. It can be observed that SMPSO clearly outperforms NSGA-II, in terms of the hypervolume and Epsilon indicators, for all network sizes. Hence, it is concluded that SMPSO outperforms NSGA-II in terms of the diversity of the non-dominated solutions and the convergence towards the true approximated Pareto-front.

Table 5: Mean and standard deviation for the HV indicator

Network Size	NSGA-II	SMPSO
100	$4.92e-02_{3.4e-01}$	$\mathbf{2.41e+02_{9.3e+02}}$
200	$1.18e-02_{5.0e-02}$	$\mathbf{1.71e+01_{5.2e+00}}$
300	$1.80e-02_{9.1e-02}$	$\mathbf{2.08e+01_{1.1e+01}}$
400	$3.07e-03_{3.4e-04}$	$\mathbf{2.07e+01_{1.2e+01}}$
500	$1.58e-02_{9.4e-02}$	$\mathbf{2.56e+01_{1.5e+01}}$

Table 6: Mean and standard deviation for the Epsilon indicator

Network Size	NSGA-II	SMPSO
100	$9.27e+01_{2.5e+00}$	$\mathbf{6.65e+00_{2.7e+00}}$
200	$2.04e+02_{4.0e+00}$	$\mathbf{7.67e+00_{2.6e+00}}$
300	$3.16e+02_{4.6e+00}$	$\mathbf{5.34e+00_{1.7e+00}}$
400	$4.32e+02_{5.7e+00}$	$\mathbf{6.20e+00_{2.0e+00}}$
500	$5.47e+02_{6.0e+00}$	$\mathbf{6.02e+00_{1.8e+00}}$

The number of non-dominated solutions (NNDS) is another widely used performance metric with larger value representing better performance [52]. Table 7 shows the average number of non-dominated solutions per run for both NSGA-II and SMPSO. The computational results show that for the NNDS metric, the SMPSO algorithm significantly outperforms the NSGA-II algorithm.

Table 7: The average number of non-dominated solutions per run

Network Size	NSGA-II	SMPSO
100	24.22	**95.76**
200	26.26	**94.5**
300	25.52	**85.1**
400	24.52	**93.62**
500	25.56	**94.26**

Table 8 illustrates the minimum values, among all the simulation runs, for the different objective functions. It is clearly shown that SMPSO obtains the best values for all the objective functions. Both algorithms are able to cluster all the sensor nodes.

Table 8: Minimum objective functions values for NSGA-II and SMPSO

Network Size	NSGA-II					SMPSO				
	CH	SC	LQ	EE	TC	CH	SC	LQ	EE	TC
100	50	0	0.853	2.438	109.71	**10**	0	**0.843**	**1.652**	**24.11**
200	107	0	0.861	2.521	218.36	**15**	0	**0.851**	**1.672**	34.55
300	167	0	0.858	2.545	341.03	**18**	0	**0.854**	**1.797**	**42.69**
400	227	0	0.861	2.544	458.91	**21**	0	**0.856**	**1.674**	**43.54**
500	285	0	0.862	2.562	571.815	**23**	0	**0.855**	**1.813**	**48.89**

6.2. *Performance Evaluation of the Proposed Protocol against well-known Clustering Protocols*

In the previous subsection, it is shown that SMPSO performs better than NSGA-II. Therefore, we adopt the SMPSO algorithm in our proposed protocol. The proposed protocol is named SMPSO-CR, from the initials of the words ***SMPSO** for **C**lustering and **R**outing*. SMPSO-CR is evaluated and compared to the well-known protocols LEACH, EHE-LEACH, EEHC, LEACH-C, PSO-C, and GA-C. To execute SMPSO-CR, an initial population of 100 particles is considered, and they evolve for 250 iterations. The values of the other SMPSO parameters are the same as in Table 4.

The results in Table 9 record the average number of CHs per round for $WSNs\#1$, for different network sizes. It can be observed that as the network density increases, SMPSO-CR achieves a lower number of CHs per round. LEACH-C, GA-C, and PSO-C always use a fixed number of CHs (which is equal to 5% of the network size) regardless of the network density. Similar results are also observed for $WSNs\#2$.

Table 9: Average number of cluster heads per round for $WSNs\#1$

Network Size	EEHC	EHE-LEACH	LEACH	LEACH-C	GA-C	PSO-C	SMPSO-CR
100	18.4	17.22	**4.84**	5	5	5	5.6
200	28.08	28.56	**9.6**	10	10	10	9.72
300	36.72	35.8	15.3	15	15	15	**14.58**
400	42.14	41.18	20.08	20	20	20	**19.16**
500	48.86	46.02	24.46	25	25	25	**24.06**

Next, the protocols are compared in terms of their scalability by varying the number of sensor nodes from 100 to 500 on both of the network scenarios, $WSNs\#1$ and $WSNs\#2$. The produced results represent the average of 5 different runs, for each network size, with a confidence level of 0.99. Figure 4 shows the comparison of SMPSO-CR against the other competent protocols in terms of the number of unclustered nodes per round in $WSNs\#2$. It can be observed that SMPSO-CR has better scalability than the other competent protocols, especially in the case of densely deployed networks. This result is due to the clustering phase of SMPSO which takes care of minimizing the number of unclustered nodes, whereas the other protocols do not deal with that problem. Similar results are also observed for $WSNs\#1$.

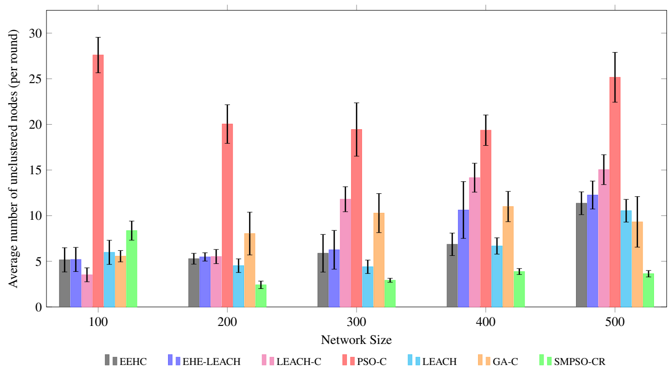

Figure 4: Average number of unclustered nodes per round for $WSNs\#2$

In order to judge the energy efficiency of SMPSO-CR, Table 10 records the mean and standard deviation for the average consumed energy per node for $WSNs$#1, for different network sizes. It is noted that as the network density increases, SMPSO-CR generally records lower energy consumption. This is because it also used less number of CHs (and hence less number of active nodes), as illustrated in Table 9. It is also noted that SMPSO-CR shows minor improvement (about 3% lower energy consumption) in terms of the average consumed energy, compared to other EA-based competent protocols. On the other side, the LEACH, EHE-LEACH, and EEHC protocols show around 50% higher levels of energy consumption because there are many unclustered nodes that are left unattended without any sleeping schedule. Although, PSO-C have the worst performance in terms of the number of unclustered nodes; it shows lower energy consumption in comparison to LEACH, EHE-LEACH and EEHC protocols. This is because PSO-C virtually clusters all the network nodes and hence gives each node a sleeping schedule. Similar results are also observed for $WSNs$#2.

Table 10: Mean and standard deviation for the average consumed energy per node for $WSNs$#1

Protocols	**100 Sensor nodes**		**200 Sensor nodes**		**300 Sensor nodes**		**400 Sensor nodes**		**500 Sensor nodes**	
	Mean	**SD**	**Mean**	**SD**	**Mean**	**SD**	**Mean**	**SD**	**Mean**	**SD**
LEACH	271.87	5.425	140.56	5.933	122.35	3.914	124.19	5.747	120.66	3.748
EHE-LEACH	176.87	7.484	160.15	2.556	146.38	3.725	139.68	1.602	138.01	2.764
EEHC	179.05	7.393	160.98	2.654	148.19	3.240	141.45	1.490	141.55	2.660
PSO-C	**71.492**	0.131	**71.469**	0.085	71.509	0.038	71.394	0.083	71.460	0.045
GA-C	74.499	0.074	72.660	0.305	71.824	0.304	71.602	0.121	71.336	0.386
LEACH-C	74.554	0.008	73.056	0.005	72.558	0.002	72.309	0.004	72.161	0.002
SMPSO-CR	76.478	3.921	71.521	0.337	**71.026**	0.519	**70.398**	0.348	**70.386**	0.330

Figure 5 shows the comparison of SMPSO-CR and other protocols, in term of the network throughput, for $WSNs$#1. Throughput is defined here as the number of data packets successfully received at the BS during the simulation time. Using the number of aggregated packets delivered to the BS is not accurate, since many packets result from the aggregation process of many raw packets collected from the cluster members. In this paper, the number of raw packets is used to calculate the throughput at the BS. The produced results represent the average of 5 different runs, for each network size, with a confidence level of 0.99. It can be observed that SMPSO-CR outperforms the other competent protocols in terms of network throughput. This is mainly due to using a dedicated routing tree for the inter-cluster communication. Similar results are also observed for $WSNs$#2.

6.3. *Performance Evaluation of the Proposed Protocol against TPSO-CR*

In addition to the previous experiments, a comparison between SMPSO-CR and TPSO-CR is conducted, in terms of their execution time, the number of active sensors, average energy consumed per node, throughput and average number of unclustered nodes per round. All the produced results represent the average of 5 different runs, for each network size, with a confidence level of 0.99.

Table 11 records the execution time in seconds, for one round of operation for $WSNs$#2, for different network sizes. We

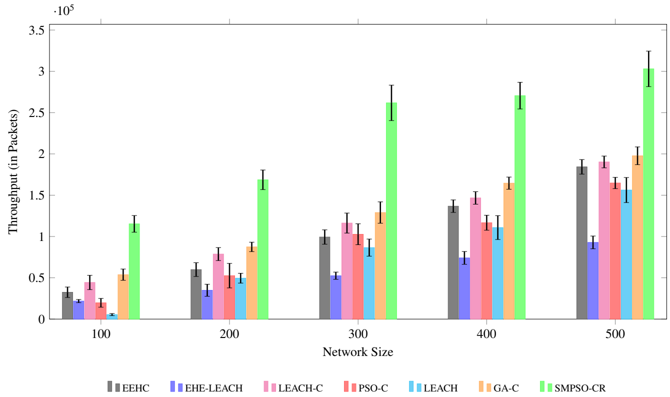

Figure 5: Throughput for $WSNs$#1

measure the execution time for the network configuration step. These results clearly show that SMPSO-CR has significantly lower execution time than TPSO-CR. In TPSO-CR, the network configuration is done over two tiers. For each tier, a set of particles evolve to find the optimal configuration. On the contrary, SMPSO-CR configures the network in one tier only which leads to much lower execution time. Similar results are also observed for $WSNs$#1.

Table 11: Execution time, in secs, for $WSNs$#2

Network Size	TPSO-CR	SMPSO-CR
100	0.78143	**0.0021**
200	3.51217	**0.00206**
300	8.82514	**0.00218**
400	16.86192	**0.0022**
500	32.22664	**0.00238**

Table 12 record the number of active nodes, for one round of operation for $WSNs$#2, for different network sizes. These results clearly show that SMPSO-CR always achieves a lower number of active nodes. In TPSO-CR, active nodes consist of both CHs and relay nodes. In SMPSO-CR, there are no relay nodes because the inter-cluster communication is limited to the CHs only. Similar results are also observed for $WSNs$#1.

Table 12: Number of active nodes for $WSNs$#2

Network Size	TPSO-CR	SMPSO-CR
100	6.8	**5.8**
200	12	**9.7**
300	17.6	**14.6**
400	22.4	**19.14**
500	27.8	**24.24**

Figure 6 shows the average energy consumed per node and their 99% confidence intervals, for $WSNs$#2. It is clearly shown that SMPSO-CR has lower energy consumption than TPSO-CR. This is because SMPSO-CR uses a smaller number of active node per round and it limits the inter-cluster communication to the CHs only. While in TPSO-CR, extra relay nodes can be added in addition to the CHs in order to construct the inter-cluster communication tree. These results are also confirmed by Table 12. Similar results are also observed for $WSNs\#1$.

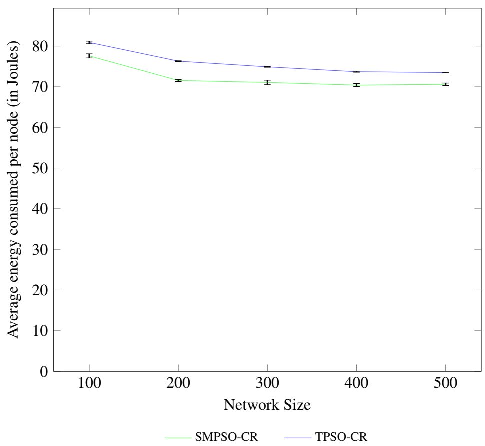

Figure 6: Average consumed energy per node for, $WSNs\#2$

Figure 7 shows the average network throughput and the 99% confidence interval for these results, for $WSNs\#2$. While SMPSO-CR has a higher throughput average for 60% of the cases, the confidence intervals show that these results are not statistically significant. Similar results are also observed for $WSNs\#1$.

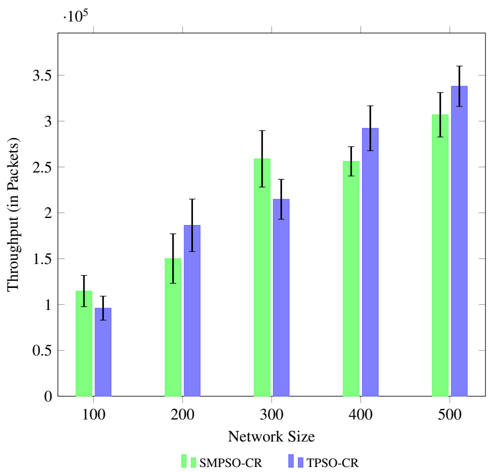

Figure 7: Throughput for $WSNs\#2$

Figure 8 shows the average number of unclustered nodes per round for $WSNs\#2$, for different network sizes. The results show that TPSO-CR outperformed SMPSO-CR for most of the cases. TPSO-CR showed better scalability in more than 90% of the networks under test. This is because TPSO-CR uses a larger number of CHs that are able to cover the whole network. Similar results are also observed for $WSNs\#1$.

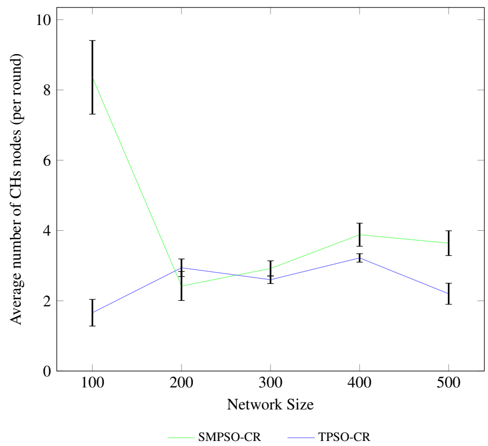

Figure 8: Average number of unclustered nodes per round, for $WSNs\#2$

7. Conclusions and Future work

7.1. Conclusions

In this paper, a centralized multi-objective Pareto optimization approach is adapted to find a joint solution for both the clustering and routing problems in WSNs. A new individual encoding scheme that represents a complete solution for this joint problem is proposed. The problem is formulated as a single multi-objective minimization problem aiming at determining an energy efficient, reliable and scalable clustering and routing scheme. Moreover, a repair function is proposed to correct any invalid routing tree and to guide the search towards the optimal routing tree. The proposed protocol is designed and tested under realistic network settings.

Two different types of MOEAs are used to solve the formulated problem: SMPSO and NSGA-II. Simulation results show that SMPSO outperforms NSGA-II in terms of the number of non-dominated solutions, the objective functions values, the convergence toward the true Pareto-front and the diversity of the obtained solutions.

Furthermore, the performance of the SMPSO-based approach (SMPSO-CR) is evaluated and compared to the following well-known protocols: LEACH, EHE-LEACH, EEHC, LEACH-C, PSO-C, and GA-C. Simulation results show that the SMPSO-CR protocol outperforms the other protocols in terms of the average consumed energy per node, number of clustered nodes and the throughput at the BS. Simulation results also confirm that using a smaller number of active nodes (CHs) and restricting the inter-cluster communication to the CHs only enhances the energy efficiency of WSNs. Moreover, using a dedicated routing tree enhances the data delivery reliability by maximizing the throughput at the BS.

In addition, the performance of the proposed protocol is evaluated and compared to TPSO-CR. Performance results show that TPSO-CR has better scalability than SMPSO-CR because TPSO-CR uses a larger number of CHs (5% of the network size). However, SMPSO-CR shows better performance in terms of the execution time. Moreover, SMPSO-CR outperforms TPSO-CR in terms of the energy efficiency property because

SMPSO-CR tends to minimize the number of CHs per round, and it limits the inter-cluster communication to the CHs only. However, in TPSO-CR, more nodes in addition to the CHs may be added to construct the routing tree. As for the throughput, SMPSO-CR has a higher average throughput for almost 60% of the cases. However, statistical analysis shows no significance in the obtained throughput results.

In summary, it has been demonstrated that the number of active nodes has a great impact on the network's energy efficiency. Minimizing the number of active CHs leads to minimizing the average of energy consumed per node and in turn maximizing the network's energy efficiency. However, increasing the number of CHs and taking link quality measures into consideration could result in more compact clusters and hence increase the PDR. Clustering protocols that ignore minimizing the number of un-clustered nodes lead to leaving those nodes unattended, and hence deplete their energy quickly. A sleep scheduling mechanism should be employed to minimize the energy consumption of such nodes. We also showed that using a dedicated routing tree results in higher network throughput and hence enhances the network's data delivery reliability. Moreover, limiting the inter-cluster communication to the CHs results in fewer active nodes, and this minimizes the average consumed energy per node and hence enhances the network's energy efficiency.

7.2. Future Work

A method to significantly reduce the energy consumption in WSNs is to apply Transmission Power Control (TPC) techniques to adjust the transmission power dynamically. In the protocol proposed in this paper, a node always transmits packets at the same power level, that is, normally the maximum possible power level. When a node transmits packets at a high power level, it may generate too much interference in the network and consume more energy than necessary. In the case of two nodes that are close to each other, low transmission power is enough to communicate with each other. Therefore, the power level should be high enough to guarantee the transmission, but should be as low as possible to save energy. As a future research direction, a cross-layer clustering protocol can be proposed such that it takes into consideration finding the optimal transmission power for each sensor node while clustering.

References

[1] R. R. Selmic, V. V. Phoha, A. Serwadda, Introduction, Springer International Publishing, Cham, 2016, pp. 1–22. doi:10.1007/978-3-319-46769-6_1.

[2] V. Sucasas, A. Radwan, H. Marques, J. Rodriguez, S. Vahid, R. Tafazolli, A survey on clustering techniques for cooperative wireless networks, Ad Hoc Networks 47 (2016) 53 – 81. doi:https://doi.org/10.1016/j.adhoc.2016.04.008.

[3] M. M. Afsar, M.-H. Tayarani-N, Clustering in sensor networks: A literature survey, Journal of Network and Computer Applications 46 (2014) 198 – 226.

[4] M. A. Mahmood, W. K. Seah, I. Welch, Reliability in wireless sensor networks: A survey and challenges ahead, Computer Networks 79 (0) (2015) 166 – 187. doi:http://dx.doi.org/10.1016/j.comnet.2014.12.016.

[5] S. Galzarano, C. Savaglio, A. Liotta, G. Fortino, Gossiping-based AODV for wireless sensor networks, in: IEEE International Conference on Systems, Man, and Cybernetics, 2013, pp. 26–31. doi:10.1109/SMC.2013.12.

[6] K. Srinivasan, P. Dutta, A. Tavakoli, P. Levis, An empirical study of low-power wireless, ACM Transactions on Sensor Networks 6 (2) (2010) 16:1–16:49. doi:10.1145/1689239.1689246.

[7] S. Ji, D. Liu, J. Shen, Localization Technology in Wireless Sensor Networks Using RSSI and LQI: A Survey, Springer Singapore, Singapore, 2017, pp. 342–351. doi:10.1007/978-981-10-3023-9_53.

[8] R. D. Gomes, D. V. Queiroz, A. C. L. Filho, I. E. Fonseca, M. S. Alencar, Real-time link quality estimation for industrial wireless sensor networks using dedicated nodes, Ad Hoc Networks 59 (2017) 116 – 133. doi:https://doi.org/10.1016/j.adhoc.2017.02.007.

[9] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, An application-specific protocol architecture for wireless microsensor networks, IEEE Transactions on Wireless Communications 1 (4) (2002) 660–670. doi:10.1109/TWC.2002.804190.

[10] R. S. Elhabyan, M. C. E. Yagoub, Particle swarm optimization protocol for clustering in wireless sensor networks: A realistic approach, in: Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration (IEEE IRI 2014), 2014, pp. 345–350. doi:10.1109/IRI.2014.7051910.

[11] M. Sharawi, E. Emary, Clustering Optimization for WSN Based on Nature-Inspired Algorithms, Springer International Publishing, 2016, pp. 111–132. doi:10.1007/978-3-319-30235-5_6.

[12] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, in: Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Vol. 2, 2000. doi:10.1109/HICSS.2000.926982.

[13] X. Liu, A survey on clustering routing protocols in wireless sensor networks, Sensors 12 (8) (2012) 11113–11153. doi:10.3390/s120811113.

[14] D. Kumar, T. C. Aseri, R. Patel, Eehc: Energy efficient heterogeneous clustered scheme for wireless sensor networks, Computer Communications 32 (4) (2009) 662 – 667. doi:http://dx.doi.org/10.1016/j.comcom.2008.11.025.

[15] S. Tyagi, S. Gupta, S. Tanwar, N. Kumar, Ehe-leach: Enhanced heterogeneous leach protocol for lifetime enhancement of wireless sns, in: 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2013, pp. 1485–1490. doi:10.1109/ICACCI.2013.6637399.

[16] D. Kumar, Performance analysis of energy efficient clustering protocols for maximising lifetime of wireless sensor networks, IET Wireless Sensor Systems 4 (1) (2014) 9–16. doi:10.1049/iet-wss.2012.0150.

[17] M. Yu, K. K. Leung, A. Malvankar, A dynamic clustering and energy efficient routing technique for sensor networks, IEEE Transactions on Wireless Communications 6 (8) (2007) 3069–3079.

[18] Z. Cui, Y. Cao, X. Cai, J. Cai, J. Chen, Optimal leach protocol with modified bat algorithm for big data sensing systems in internet of things, Journal of Parallel and Distributed Computingdoi:https://doi.org/10.1016/j.jpdc.2017.12.014.

[19] N. Latiff, C. Tsimenidis, B. Sharif, Energy-aware clustering for wireless sensor networks using particle swarm optimization, in: IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMR C'07), 2007, pp. 1–5. doi:10.1109/PIMRC.2007.4394521.

[20] A. Rahmanian, H. Omranpour, M. Akbari, K. Raahemifar, A novel genetic algorithm in leach-c routing protocol for sensor networks, in: 24th Canadian Conference on Electrical and Computer Engineering (CCECE), 2011, 2011, pp. 001096–001100. doi:10.1109/CCECE.2011.6030631.

[21] R. S. Elhabyan, M. C. Yagoub, Two-tier particle swarm optimization protocol for clustering and routing in wireless sensor network, Journal of Network and Computer Applications 52 (2015) 116 – 128. doi:http://dx.doi.org/10.1016/j.jnca.2015.02.004.

[22] G. P. Gupta, S. Jha, Integrated clustering and routing protocol for wireless sensor networks using cuckoo and harmony search based metaheuristic techniques, Engineering Applications of Artificial Intelligence 68 (2018) 101 – 109. doi:https://doi.org/10.1016/j.engappai.2017.11.003.

[23] G. Molina, E. Alba, Location discovery in wireless sensor networks using metaheuristics, Applied Soft Computing 11 (1) (2011) 1223 – 1240. doi:10.1016/j.asoc.2010.02.021.

[24] K. Xu, I. Howitt, Realistic energy model based energy balanced optimization for low rate wpan network, in: IEEE Southeastcon, 2009. SOUTHEASTCON '09, 2009, pp. 261–266.

[25] M. Mallinson, S. Hussain, J. H. Park, Investigating wireless sensor network lifetime using a realistic radio communication model, in: International Conference on Multimedia and Ubiquitous Engineering, IEEE Computer Society, Los Alamitos, CA, USA, 2008, pp. 433–437. doi:http://doi.ieeecomputersociety.org/10.1109/MUE.2008.62.

[26] S. K. Mitra, M. K. askar, Comparative study of radio models for data gathering in wireless sensor network, International Journal of Computer Applications 27 (4) (2008) 49–57.

[27] C. Coello, G. Pulido, M. Lechuga, Handling multiple objectives with particle swarm optimization, IEEE Transactions on Evolutionary Computation 8 (3) (2004) 256–279. doi:10.1109/TEVC.2004.826067.

[28] Z. Wu, C. Kwong, R. Aydin, J. Tang, A cooperative negotiation embedded nsga-ii for solving an integrated product family and supply chain design problem with remanufacturing consideration, Applied Soft Computing 57 (2017) 19 – 34. doi:https://doi.org/10.1016/j.asoc.2017.03.021.

[29] P. M. Pradhan, G. Panda, Connectivity constrained wireless sensor deployment using multiobjective evolutionary algorithms and fuzzy decision making, Ad Hoc Networks 10 (6) (2012) 1134 – 1145. doi:10.1016/j.adhoc.2012.03.001.

[30] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: Nsga-ii, IEEE Transactions on Evolutionary Computation 6 (2) (2002) 182–197. doi:10.1109/4235.996017.

[31] A. Nebro, J. Durillo, J. Garcia-Nieto, C. Coello Coello, F. Luna, E. Alba, Smpso: A new pso-based metaheuristic for multi-objective optimization, in: IEEE symposium on Computational intelligence in multi-criteria decision-making, 2009. mcdm '09., 2009, pp. 66–73. doi:10.1109/MCDM.2009.4938830.

[32] J. Durillo, J. Garca-Nieto, A. Nebro, C. Coello, F. Luna, E. Alba, Multiobjective particle swarm optimizers: An experimental comparison, in: M. Ehrgott, C. Fonseca, X. Gandibleux, J.-K. Hao, M. Sevaux (Eds.), Evolutionary Multi-Criterion Optimization, Vol. 5467 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2009, pp. 495–509. doi:10.1007/978-3-642-01020-0_39.

[33] A. Nebro, J. Durillo, C. Coello, Analysis of leader selection strategies in a multi-objective particle swarm optimizer, in: 2013 IEEE Congress on Evolutionary Computation (CEC), 2013, pp. 3153–3160. doi:10.1109/CEC.2013.6557955.

[34] J. J. Durillo, A. J. Nebro, F. Luna, C. A. Coello Coello, E. Alba, Convergence speed in multi-objective metaheuristics: Efficiency criteria and empirical study, International Journal for Numerical Methods in Engineering 84 (11) (2010) 1344–1375. doi:10.1002/nme.2944.

[35] J. Durillo, A. Nebro, C. A. Coello Coello, J. Garcia-Nieto, F. Luna, E. Alba, A study of multiobjective metaheuristics when solving parameter scalable problems, IEEE Transactions on Evolutionary Computation 14 (4) (2010) 618–635. doi:10.1109/TEVC.2009.2034647.

[36] D. Kalyanmoy, Multi-Objective Optimization Using Evolutionary Algorithms, 1st Edition, Wiley, 2001.

[37] M. Clerc, J. Kennedy, The particle swarm - explosion, stability, and convergence in a multidimensional complex space, IEEE Transactions on Evolutionary Computation 6 (1) (2002) 58–73. doi:10.1109/4235.985692.

[38] B. Panigrahi, V. R. Pandi, R. Sharma, S. Das, S. Das, Multiobjective bacteria foraging algorithm for electrical load dispatch problem, Energy Conversion and Management 52 (2) (2011) 1334 – 1342. doi:http://dx.doi.org/10.1016/j.enconman.2010.09.031.

[39] E. Zitzler, J. Knowles, L. Thiele, Quality assessment of pareto set approximations, in: Multiobjective Optimization, Vol. 5252 of Lecture Notes in Computer Science, 2008, pp. 373–404.

[40] E. Zitzler, L. Thiele, Multiobjective optimization using evolutionary algorithms a comparative case study, in: Parallel Problem Solving from Nature, Vol. 1498 of Lecture Notes in Computer Science, 1998, pp. 292–301. doi:10.1007/BFb0056872.

[41] M. Li, S. Yang, X. Liu, Diversity comparison of pareto front approximations in many-objective optimization, IEEE Transactions on Cybernetics 44 (12) (2014) 2568–2584. doi:10.1109/TCYB.2014.2310651.

[42] Y. Cao, B. J. Smucker, T. J. Robinson, On using the hypervolume indicator to compare pareto fronts: Applications to multi-criteria optimal experimental design, Journal of Statistical Planning and Inference 160 (0) (2015) 60–74. doi:http://dx.doi.org/10.1016/j.jspi.2014.12.004.

[43] A. Nebro, F. Luna, E. Alba, B. Dorronsoro, J. Durillo, A. Beham, Abyss: Adapting scatter search to multiobjective optimization, IEEE Transactions on Evolutionary Computation 12 (4) (2008) 439–457. doi:10.1109/TEVC.2007.913109.

[44] H.-C. Cheng, T.-C. Chiang, L.-C. Fu, A two-stage hybrid memetic algorithm for multiobjective job shop scheduling, Expert Systems with Applications 38 (9) (2011) 10983 – 10998. doi:http://dx.doi.org/10.1016/j.eswa.2011.02.142.

[45] J. J. Durillo, A. J. Nebro, jmetal: A java framework for multi-objective optimization, Advances in Engineering Software 42 (10) (2011) 760 – 771. doi:http://dx.doi.org/10.1016/j.advengsoft.2011.05.014.

[46] E. Zitzler, L. Thiele, M. Laumanns, C. Fonseca, V. da Fonseca, Performance assessment of multiobjective optimizers: an analysis and review, IEEE Transactions on Evolutionary Computation 7 (2) (2003) 117–132. doi:10.1109/TEVC.2003.810758.

[47] Texas Instruments, Chipcon CC2420 radio transceiver data sheet, http://www.ti.com/lit/ds/symlink/cc2420.pdf, Last access: September 25, 2014 (2013).

[48] A. Barberis, L. Barboni, M. Valle, Evaluating energy consumption in wireless sensor networks applications, in: 10th Euromicro Conference on Digital System Design Architectures, Methods and Tools, 2007. DSD 2007., 2007, pp. 455 –462. doi:10.1109/DSD.2007.4341509.

[49] S. Khatarkar, R. Kamble, Wireless sensor network mac protocol: Smac and tmac, Indian Journal of Computer Science and Engineering 4 (4) (2013) 304–310.

[50] S. Piccand, M. O'Neill, J. Walker, On the scalability of particle swarm optimisation, in: IEEE Congress on Evolutionary Computation, 2008, pp. 2505–2512.

[51] D. Bratton, J. Kennedy, Defining a standard for particle swarm optimization, in: Swarm Intelligence Symposium, 2007. SIS 2007. IEEE, 2007, pp. 120–127. doi:10.1109/SIS.2007.368035.

[52] M. Rabiee, M. Zandieh, P. Ramezani, Bi-objective partial flexible job shop scheduling problem: Nsga-ii, nrga, moga and paes approaches, International Journal of Production Research 50 (24) (2012) 7327–7342. doi:10.1080/00207543.2011.648280.