UNIVERSITY OF STUDY OF CAMERINO

INTERNATIONAL SCHOOL OF ADVANCED STUDIES

Energy Efficient Clustering and Routing Protocols for Wireless Sensor Networks: a Simulation Approach

PhD Candidate (XXXI Cycle):
Matteo MICHELETTI

A thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosophy in

Computer Science

Supervisors:
Prof. Leonardo MOSTARDA
Prof. Diletta Romana CACCIAGRANO

Candidate:
Matteo MICHELETTI

JANUARY 2020

UNICAM

Abstract

Department or Computer Science

Doctor of Philosophy

Energy Efficient Clustering and Routing Protocols for Wireless Sensor Networks: a Simulation Approach

by Matteo MICHELETTI

This dissertation presents an approach to solve the Wireless Sensors Networks (WSNs) energy depletion problem. WSNs devices are usually battery-powered thus gathering data from a WSN in an energy efficient way is challenging. Each device is a node of the network. Clustering, an approach which organises nodes into sets (clusters), is a widely used solution proposed by researchers. Each cluster has a coordinator, called cluster head (CH), that gathers data from its neighbours and communicates with near CHs. Nodes elect to become CH basing on their hardware capabilities and some other network metrics. Most of the works in the literature use a cluster election that only considers the nodes battery capacity and residual energy. Such behaviour can cause a quick depletion of nodes energy level. CHs should be chosen considering other features in addition to residual energy. For this reason, Rotating Energy Efficient Clustering for Heterogeneous Devices (REECHD) has been conceived. REECHD is a clustering protocol which considers both node residual energy and node work for leader election. REECHD reduces the number of leader elections by rotating the CH role among nodes in the same cluster. REECHD also introduces the concept of intra-traffic rate limit (ITRL) which defines a limit on the communication inside the clusters and can be used to further improve energy efficiency. REECHD is then further enhanced by introducing a novel routing approach (CER-CH) which combines a novel rotation heuristic with a novel top-down CH routing tree definition in order to balance the node energy consumption and generate more energy efficient routing trees. CER-CH defines the rotation and the routing amongst CHs, regardless of the strategy chosen for the initial CHs election and cluster formation. Any clustering algorithm that produces a CH per cluster can be used. REECHD and CER-CH have been fairly compared with the state of art routing protocols through simulation. Simulations show that they ensure a gain with respect to the state of art clustering protocols.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

A wireless sensor network (WSN) is a network composed of spatially distributed sensing devices which monitor the physical environment and send their data through the network to a centralised base station (BS).

WSNs have a wide range of possible applications, including but not limited to industrial process monitoring and control, machine health monitoring, environmental detection, and habitat monitoring. A WSN is built of nodes, varying from few tens to several thousands. Each node is equipped with one ore more sensors, a radio transceiver with an antenna and a controller, electronic circuitry, and a battery. The size of a node can vary from centimetres down to dust-size. Battery and size are a constraint on nodes capabilities such as memory, computational power and communications range. A WSN topology can vary from a simple star to a more advanced multi-hop network. Propagation between hops can be done by flooding or by routing.

WSNs are referred to as homogeneous when all their nodes are equal i.e. they have the same hardware. A WSN which is not homogeneous is referred to as heterogeneous. Fig.1.1 shows a homogeneous WSN while fig.1.2 shows a heterogeneous WSN.

WNSs are the backbone of the Internet-of-Things.

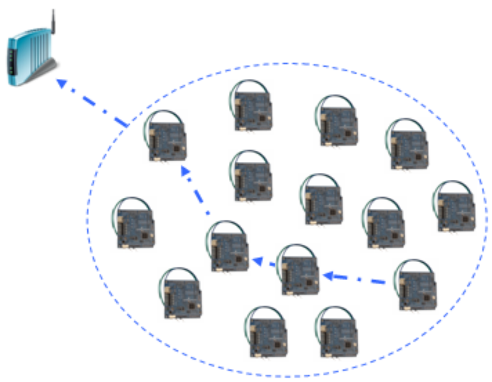

FIGURE 1.1: A homogeneous WSN

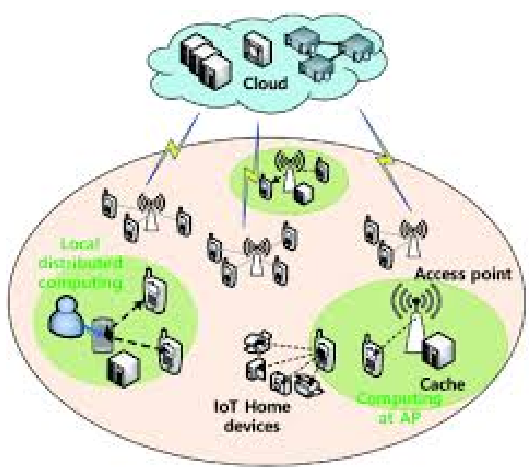

FIGURE 1.2: A heterogeneous WSN

Internet-of-things (IoT) is one of the modern technological revolutions that allows communication amongst a variety of different devices. Various studies predict that the global IoT market will grow from $157B in 2016 to $457B by 2020. Main applications [1, 2, 3, 4] include transportation, smart homes, smart supply chain, smart cities, connected cars, smart industry, and smart retails. Heterogeneous Wireless Sensor Networks (WSNs) are an essential part to implement the IoT vision. WSNs are used within an IoT system to collect and send data through a router as part of the infrastructure system. WSNs provide a virtual layer which can collect information about the physical world. Moreover, WSNs come with many benefits such as low cost, easy deployment and high fidelity sensing. Nevertheless, using WSN technology comes with great challenges. One of the most critical is the need to operate under severe resource constraints, which makes efficient design of WSNs very important.

1.2 Background and motivations

WSNs are networks composed of one or more BS and a variable number of sensor nodes. Data sensed from nodes is wirelessly transmitted to a BS which collects, analyses and stores it. Nodes in the Wireless Sensor Network can act as routers in order to transmit the information collected from their neighbours to the sink node (the BS). It is necessary to know the way nodes process sensed data before sending it. Most of the time, nodes can apply some kind of aggregation to data they sense or receive from near nodes. This is usually done by shrinking the size of the messages nodes send, or to decrease their number. Since the energy spent on communication is directly related to the amount of data nodes are sending, data aggregation reduces the energy spent from the nodes thus slowing down battery draining.

1.2.1 WSN features

WSNs can vary from having a low number of nodes and a very low network traffic to deploy thousands or nodes, leading to a huge increase in traffic. The number of nodes also affects the density of the network, where less nodes results in lower densities with respect to higher amounts of nodes. In WSNs devices are usually battery-powered thus gathering data from a WSN in an energy efficient way is a challenging research area [5]. Requirements and constraints of WSNs are challenging to design. Sensor nodes have a wide set of characteristics, ranging from power constraints (battery), resource constraints (CPU), hardware availability (boards), low communication range (antennas) to limited storage capabilities (no physical drives). Knowing all these capabilities is essential to understand and reproduce WSNs. In addition to this, it is important to know everything about the area in which nodes are deployed. This includes:

- the shape of the area
- the dimensions of the area
- the sink(s) position(s)
- the density of the area (or the number of sensors inside it)

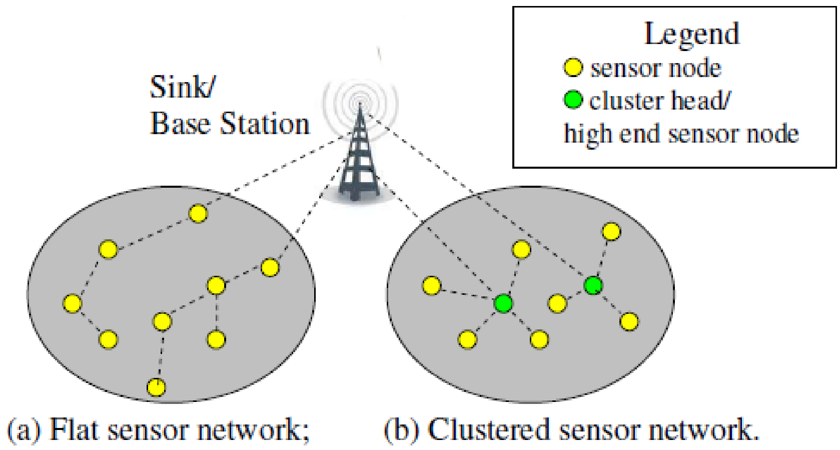

FIGURE 1.3: A clustered WSN

Since devices are battery-powered and they are often placed in spots which are difficult to reach, frequent battery replacements should be avoided. This raises the need for the energy efficiency of sensor nodes.

One effective way to reduce nodes energy consumption is multi hop communication. In multi hop communication, data is not sent from nodes to the BS through direct communication to the base station but is communicated in several hops from a source node through neighbouring nodes toward the base station, instead. Multi hop communication can also include node-level processing, such as data aggregation. Another possible solution proposed by the researchers to reduce energy consumption is Clustering (see fig.1.3). This organises the devices into sets (clusters). Each cluster has a cluster head (CH) that gathers data from its nodes (intra-cluster communication) and communicates with other CHs in order to report data to a centralised base station (BS) (inter-cluster communication). Quite a few clustering protocols have been proposed so far [6, 7, 8, 9]. Protocols can generate clusters with different features such as equal-sized, unequal-sized and grid based clusters. Some protocols make use of rotation techniques in order to better distribute energy consumption among the nodes composing the network. When rotation is applied the current CH designates the next CH without the need of any cluster election to be performed. This reduces the amount of overhead messages. Many approaches are used to elect CHs. Among the most used are assigning each node a probability to become CH based on some node features, defining the most fit node through machine learning techniques or assigning weights to nodes using analytic formulae and then use such weights for CH election. In these cases, cluster heads selection is usually based on the node residual energy. Few approaches may consider other nodes capabilities for cluster formation. In fact, most of the clustering approaches have their roots on the homogeneous WSN field which assumes devices with homogeneous capabilities [5, 10].

The WSN area can have various shapes such as circles and rectangles. Distribution of WSN nodes in the sensing area can be non-uniform. This can generate high-rate WSN subareas that produce a higher amount of traffic. The presence of high-rate subareas is realistic when heterogeneous WSNs are deployed in the context of IoT systems [11, 12, 13]. For instance, a nuclear reactor [11] is usually surrounded by temperature sensors that produce a higher volume of data when compared to radioactive ones which are farther away from the reactor. Sensors that produce a higher amount of data will eventually have less energy when compared to lower data sensors. The BS is located away from the reactor and collects data from the nearby radioactive sensors and the farther away temperature sensors. Such behaviour can originate the low-rate-high-energy node depletion problem [14]. This causes a quick energy depletion of nodes with high transmission rate and low energy.

Another variable which affect the energy consumption of WSNS is the CHs routing tree definition. Static clustering divides the network according to some virtual grids [9, 15, 16], virtual layers [7, 17, 18], concentric circles [18, 19]. In these cases rotation usually includes some predetermined scheduling that is calculated by using a node energy consumption model. This estimates the energy consumption in average conditions (e.g., average distance between a member and its CH) and simplified settings (e.g., virtual grids). This may lead to lifetime performance degradation when the average case is not representative. On the contrary, dynamic clustering have clusters that change over the WSN lifetime. Rotation is usually adaptive since considers the node residual energy [20, 21]; few approaches may add other node features such as the node rate and the node initial energy [14]. Although rotation based on residual energy is suitable for a wide range of WSN settings, it may be inefficient for heterogeneous networks [14]. After CH selection a CH routing tree is defined [22–24]. This allows the delivery of data from each node to the BS. Routing tree algorithms aim at ensuring connectivity and very few approaches consider the energy efficiency of the generated routing tree [22, 23]. These are limited at building energy efficient routing paths after the CH routing tree has been defined.

1.2.2 Radio models

WSNs nodes communicate with each other to forward data toward one ore more BS. The amount of energy they spend in the process depends on their equipped radios.

Nowadays, there exist several outdoor and indoor propagation models suitable for simulating WSNs. Each one of them has its own propagation environment, assumptions, and restrictions which make them more suitable to certain WSNs more than others. In the following, some of the most used RF-propagation models for WSNs are presented.

Free Space Model: considers only the line-of-sight path loss. No obstacles nearby that could cause reflections or diffraction. It is defined by equation:

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L}$$

where P_t is the transmitted signal power. G_t and G_r are the antenna gains of the transmitter and the receiver respectively. $L(L \geq 1)$ is the system loss, and λ is the wavelength.

Adapted Free Space Model: considers only the line-of-sight path loss. Uses an empirical value for the path loss exponent n according to the propagation environment. It is an empirical model, most used in urban environments, especially for safety applications in the field of Vehicular Ad Hoc Networks (VANETs) [25].

Two-Ray Ground Reflection Model: considers the line-of-sight path plus the ground reflected path and the real ground reflection coefficient. It is defined by equation:

$$P_r(d) = \frac{P_t G_t G_r {h_t}^2 {h_r}^2}{d^4 L}$$

where h_t and h_r are the heights of the transmit and receive antennas respectively. It is commonly used for localisation purposes in indoor and outdoor environments, e.g. in proximity detection involving large machines (mining, stevedoring and construction)[26].

Simplified Two-Ray Ground Reflection Model: considers the line-of-sight path plus the ground reflected path. Assumes a large distance d between the transmitter and the receiver, and a perfect reflection coefficient. It is frequently used in simulation-based performance evaluation of Inter-Vehicle Communication (IVC) protocols [27].

Shadowing Model: it is an extension to the free space model. The shadowing model consists of two parts. The first one is known as path loss model, which also predicts the mean received power at distance d.

$$\frac{P_r(d_0)}{\overline{P_r(d)}} = \left(\frac{d}{d_0}\right)^{\beta}$$

β is called the path loss exponent, and is usually empirically determined by field measurement. The second part of the shadowing model reflects the variation of the received power at certain distance. It is a log-normal random variable, that is, it is of Gaussian distribution if measured in dB. The overall shadowing model is represented by

$$\left[\frac{P_r(d)}{P_r(d_0)}\right]_{dB} = -10\beta \log\left(\frac{d}{d_0}\right) + X_{dB}$$

where X_{dB} is a Gaussian random variable with zero mean and standard deviation σ_{dB}. σ_{dB} is called the shadowing deviation, and is also obtained by measurement.

Finally, **MIT** developed a radio model they presented in LEACH article[28]. LEACH introduces a widely adopted radio model that utilises Friis free space and multi-path radio channel models. Transceiver circuitry of a sensor node consumes $E_{elec} = 50nJ/bit$. Sensor node amplification energy E_a depends on the distance between sender and receiver. When $d < d_0 = 75m$, E_a becomes $E_{fs} = 10pJ/bit/m^2$ and when $d \geq d_0 = 75m$, E_a reduces to $E_{mf} = 0.0013pJ/bit/m^4$. The transmission and reception energy consumed in sending and receiving a data packet of k bits over a distance d, can be computed as follows:

$$E_{Tx}(k,d) = k(E_{elec} + E_a d^n) \tag{1.1}$$

$$E_{Rx}(k) = k(E_{elec}) \tag{1.2}$$

where $n = 2$ for $d < d_0$, and $n = 4$ for $d \geq d_0$.

The **MIT** model is used to compute communication energy costs and determine WSN lifetime inside all simulations in the context of this study.

1.3 Research questions

What stated in the previous section leads to the following questions:

Q1: Is it possible to develop a clustering election strategy to reduce the energy consumption of Heterogeneous WSNs ?

Q2: Is it possible to avoid quick energy depletion of nodes with low transmission rate and high battery power in Heterogeneous WSNs?

Q3: Is it possible to define a CHs rotation strategy which considers the energy efficiency of the CH routing tree, regardless the CH election strategy ?

1.4 Thesis contributions

There exists a vast literature on clustering approaches proposed by researches to reduce the energy consumption of nodes in a WNS. These protocols only consider the residual energy for cluster head election and cluster formation. This can cause the low-rate-high-energy problem. For this reason, Rotating Energy Efficient Clustering for Heterogeneous Devices (REECHD) [14] has been conceived to consider both node residual energy and node work for leader election and rotation. The node work estimates the energy effort of a CH by considering the node transmission rate and the node residual energy. The novel election protocol of REECHD solves the low-rate-high-energy node depletion problem. REECHD also introduces the concept of intra-traffic rate limit (ITRL). This defines a limit on the intra-traffic communication that all WSN clusters must comply with. ITRL can be used to improve energy efficiency. The energy efficiency of REECHD has been compared with other existing protocols for heterogeneous WSNs through extensive simulations. Results prove that REECHD is more energy efficient when compared with well-known clustering protocols conceived for homogeneous and heterogeneous WSNs like ER-HEED[21], HEED [24], RUHEED [20], UHEED [29], FMUC [7] and M-LEACH [30].

It is also possible to reduce WSNs nodes energy consumption by optimising the inter-traffic communication that is how information is routed from CHs to the BS. Information is usually routed by using a CH-routing tree which is rooted at the BS. The CH selection and the CH routing tree definition are separated, more precisely, routing tree definition always follows the cluster head election [24]IEEEhowto:RUHEED. REECHD has been further enhanced by proposing a novel approach (CER-CH) which combines a novel rotation heuristic with a novel top-down CH routing tree definition in order to balance the node energy consumption and generate more energy efficient routing trees. CER-CH defines the rotation and the routing amongst CHs, regardless of the strategy chosen for the initial CHs election and cluster formation. Although REECHD [14] has been used for the CH election and cluster formation, any clustering algorithm that produces a CH per cluster can be used. REECHD enhanced with CER-CH has been fairly compared with the state of art routing protocols.

More precisely, well-known approaches for routing information amongst CHs have been implemented and simulated with various WSN settings and a widely accepted energy model. These settings are used to equally compare such protocols by considering the first node die lifetime measure. As a result, simulations show that CER-CH ensures a gain with respect to the state of art clustering protocols.

1.5 Methodology

The purpose of this section is to introduce the research methodology for this quantitative and (partially) qualitative simulation study regarding . This approach allowed for an extensive study of the behaviour of clustering and routing protocols on various WSNs with different configurations. In this section, the motivations, methodology, and methods used for this study are discussed.

1.5.1 Methodology selected

A simulation process is based on feeding data into a simulator to produce quantitative results in a structured sequential process. Discrete event simulation (DES) is widely known to be a typical quantitative research tool. However, it is also possible to consider a less structured approach more inclined towards judgemental and expert knowledge on a little amount of data, applying a qualitative approach to DES. In this study, DES has been mainly used for performance measurement, thus following a quantitative approach. However, by combining the results of different simulations, a certain amount of reasoning can be applied to understand their implications.

1.5.2 Motivations

Testing and evaluating protocols for WSNs is sometimes difficult to achieve through real experiments. WSNs benchmarking is the measurement and evaluation of wireless sensor protocols applied to devices and networks. To measure performance and characteristics of a network protocol, it needs to be benchmarked in multiple real life environments under various realistic conditions. However, such ad-hoc solutions lead to fast results only if they compare a fixed set of wireless protocols on some fixed networks once [31, 32]. In other words, the reproducibility obtained through these tests is generally limited. So, designing and implementing a flexible benchmarking for wireless sensor solutions on testbeds is not usually feasible.

Simulators can overcome the problems aforementioned. Simulators are flexible tools able to test and analyse the performances of protocols and algorithms when they are applied on a WSN.

1.5.3 Simulation method

This section introduces the simulation approach adopted in this study. The simulation procedure used to collect results of various clustering and routing protocols on different network configurations is presented.

The novel clustering protocol introduced in chapter3 and its main competitors are composed of the following phases: (i) cluster formation; (ii) CH routing tree definition; (iii) network operation; (iv) rotation in the role of CH (if any). While the cluster formation phase is a stochastic process the other phases are deterministic.

Cluster formation is a non-deterministic process where a clustering algorithm such as HEED[24] or UHEED[29] is used to elect CHs and form clusters. It is assumed that each member node can only join a cluster head, that is the node can be part of exactly one cluster. Cluster formation is non-deterministic because each sensor node can choose in a probabilistic way to be a CH. This probability depends on various features such as the node transmission rate, the residual energy, a random probability and the presence of surrounding nodes which are already CHs.

CH routing tree definition is a deterministic process that, once CHs are elected, builds a CH routing tree. This is used in order to deliver messages from CHs to the BS.

Network operation is a deterministic phase where sensor data is collected and delivered to the BS. More precisely, a round based model is used, that is each CH defines a TDMA schedule, i.e., one slot for each member. Each CH collects all data from its sensor members (intra-cluster communication), aggregates the data and forwards them to its father in the routing tree (i.e., either another CH or the BS). CHs forwards messages that are received from other CHs without any aggregation. We assume that no messages are lost or corrupted that is the MAC and physical layers are ideal. We only consider the energy consumption required to send the messages. A round starts with the collection of sensor data by the CHs and terminates when all CHs data reaches the base station.

Rotation in the role of CH rotation is a deterministic process where the current CH designates one of its member as the next CH, without the need of any cluster election to be performed.

Figure 1.4 shows the dynamic behaviour of a clustering protocol that has no rotation phase. Cluster formation, routing tree definition and network operation phases are repeated over time until one of the node dies. First node die (FND) is mainly used as lifetime measure to determine the lifetime of the WSNs, althought half of the nodes died (HND) is sometimes considered.

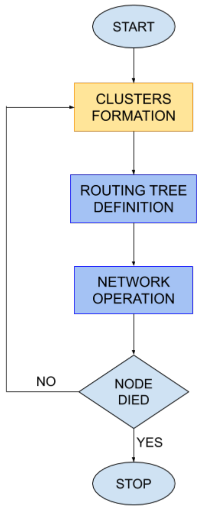

FIGURE 1.4: Phases of a clustering protocol. Cluster formation requires simulation, while the other phases are deterministic and are executed after clusters have been formed.

Clustering protocols have been simulated by using a discrete event driven simulator with a global tick (a similar approach has been used for the simulators used in UHEED and other simulations). At each global tick the simulator call a $run()$ procedure on each sensor. This defines a clustering algorithm, e.g., HEED and UHEED. At each tick T_i the run function of a sensor can:

- send messages to neighbouring sensors (e.g, a "leader_candidation" message). The sending is implemented by adding the message to the buffer of the receiving nodes. These messages can be used by the sensor run procedures in the next tick, i.e, T_{i+1}.
- use the messages that were sent at the time T_{i-1};
- put the sensor in a stop status (no action is anymore executed).

When all nodes are in a stop state the simulation ends and the clusters are formed. Afterwards all deterministic code is executed that is CH routing tree definition, network operation phase and rotation in the role of CH (if any).

Figure 1.5 shows the execution of the simulation for three sensors. Each sensor has a receiving buffer. Is it possible to see that at tick 2, sensor 2 can use the messages that were sent by sensor 1 and sensor 3.

1.5.4 Validation

The tool used for simulation has been validated simulating existing WSN configurations in the literature and comparing results. Configurations of LEACH[28], HEED[24] and UHEED[29]

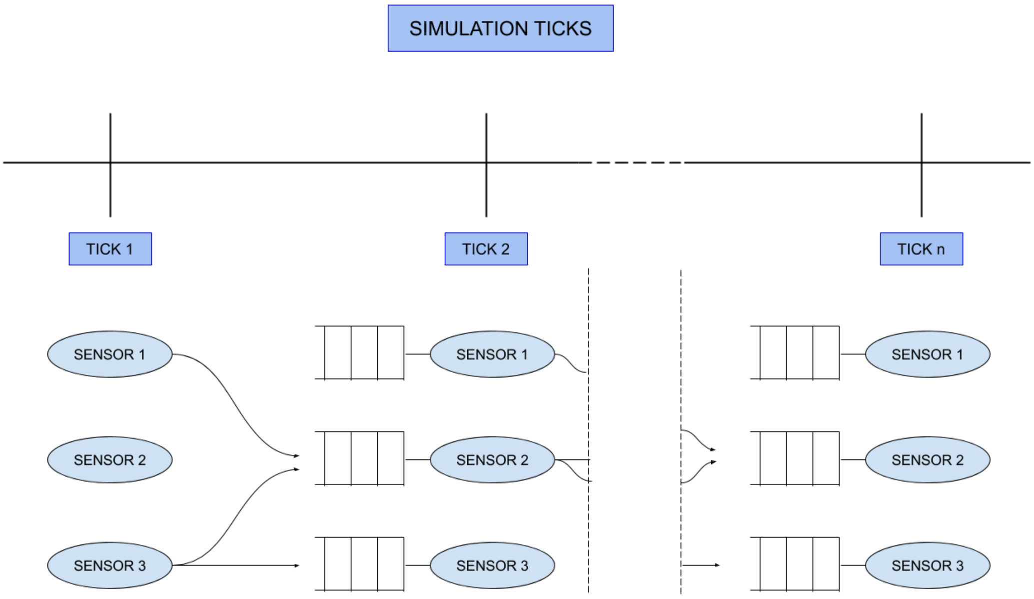

FIGURE 1.5: Global simulation ticks. Nodes can send a message to their neighbours incoming messages queues, which are consumed in the next tick.

has been simulated with the simulation tool, and then results have been collected and compared to assess they matched with the ones in the literature. In particular, it has been verified that the confidence interval of the region around the difference between results in the literature and the mean of several simulations in the simulation tool is within 95%, i.e. with a significance level of 5%, for the same configurations.

1.5.5 Perfomance

The performance of the simulation tool has been compared with NS2[33]. The same network has been simulated in both simulators and the real execution time has been compared. Figures1.6and1.7 show that the simulation tool is up to 10 times faster than NS2 on the same hardware when simulating the same network.

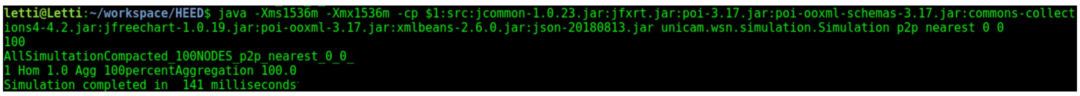

FIGURE 1.6: Performance of a simulation with the simulation tool.

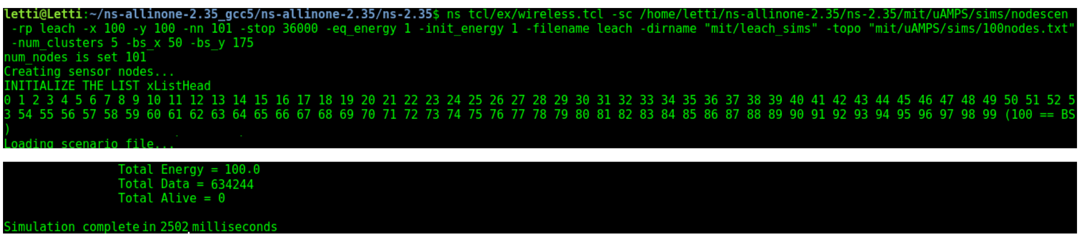

FIGURE 1.7: Performance of a simulation with NS2. The standard output of the simulation (in the centre) has been removed for the sake of space.

1.5.6 Tool architecture

A network simulator may include several components which models energy, power consumption, networking protocols and so on. Since WSNs are application specific, different WSNs have different environments and numerous other conditions which affects nodes interaction.

A WSN Simulator usually implement a subset of the following modules:

- Event
- Environment
- Node
- Transceiver
- MAC Protocol (ISO OSI level 2)
- Routing Protocol (ISO OSI level 3)
- Application Layer

Simulators could implement some modules in idealised way (i.e. a MAC protocol with no packet collisions) if they are focused on performing tasks which does not require to be studied.

Event: it describes how events should work, and changes the state of the simulator. Events are fired after every tick of the system.

Environment: it describes the physical properties of the model. It includes physical environment variables like coverage areas, positions of the nodes, and so.

Node: a single device in the network. This module defines all hardware and software properties of a sensor node, including: transceiver, initial energy, transmission rate and routing protocol.

Transceiver: models the hardware transceiver of sensor node (according to the chosen Radio Model). It covers the transceiver states i.e. Receive and Transmit along with their power consumption. When signals are fired, the transceiver generates events.

MAC Layer: allows nodes to send and receive packets. The simulation tool MAC layer is assumed to be collision-free and not affected from physical obstacles. So, the communication between nodes is not impeded, and simultaneous transmissions do not produce collisions. This is aligned to the simulation approaches of other work in the literature [7, 24, 28] to which the study presented in this Thesis want to compare.

Routing Protocol: allows WSNs nodes to communicate among themselves and to send data from sensor nodes to the BS. Routing Protocols can be classified in many ways (see fig.3.2). In particular, from a network point of view, they can be:

- point-to-point (direct communication)
- flat
- tree-based

different routing protocols are implemented in the simulation tool in order to study the effects of coupling routing and clustering protocols in a simulation.

1.5.7 Custom GUI

The simulation tool is designed to be used as a command line tool for parallel simulations. Even so, it comes with a GUI which has proven to be very useful (see figure 1.8). In fact, it can guide non-technical users during the set-up phase of the simulation (network creation, input of simulation parameters, clustering protocol choice). Moreover, it is also useful for debugging purposes during new protocols implementation. In particular, the GUI allows to see what is happening during every phase of the network clustering, the CH routing tree and which nodes are dead during a simulation. This is very useful to better interpret results in a qualitative study.

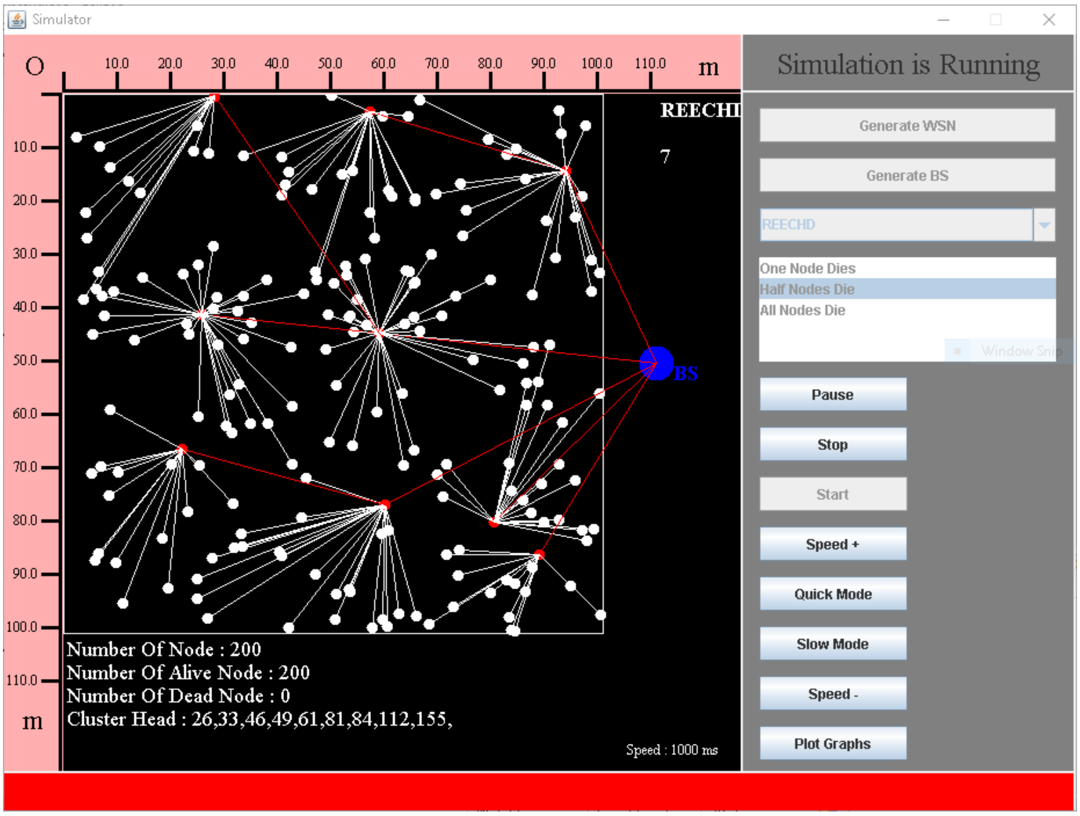

FIGURE 1.8: Simulation tool GUI. On the right there are the simulation network parameters, on the left the simulation area, in the bottom the current list of CHs

1.6 Organisation of Thesis

Chapter 2 reports the state of art on existing approaches to improve the energy efficiency and to manage communication inside WSNs with limited capabilities. In particular, existing solutions suitable for heterogeneous networks has been inspected.

Chapter 3 presents the work done to develop Rotating Energy Efficient Clustering for Heterogeneous Devices (REECHD), a novel Clustering protocol which take into accounts the heterogeneity of the nodes. REECHD works at network layer and includes novel clustering and routing algorithms to minimise the energy consumption, thus prolonging network lifetime, in heterogeneous WSNs.

Chapter 4 introduces CER-CH, a novel routing approach which combines CH rotation and CH routing tree formation to improve the network lifetime.

Chapter 5 compares REECHD and CER-CH with various clustering protocols. Comparison is performed by simulating all protocols with same case study and assumptions. This ensures a fair comparison. In addition, the impact on the network lifetime of different routing approaches is studied.

Chapter 6 concludes the Thesis and states possible future developments starting from current results.

Chapter 2

State of art on WSN

This chapter presents the state of art on existing approaches to improve the energy efficiency and to manage communication inside WSNs with limited capabilities, paying particular attention on existing solutions suitable for heterogeneous networks.

2.1 Clustering

Flooding all data from sensors to the sink directly it is highly inefficient for large scale energy-constrained sensor networks. An alternative, more efficient approach is to let intermediate nodes collect data from all the sensors within their scope and forward it to the sink node. Since such approach organises nodes into clusters, intermediate nodes are called cluster heads. The way in which cluster head election is done can greatly affect the energy saving. For this reason, many algorithms have been proposed for cluster head election to maximise the energy efficiency.

2.1.1 Clustering Homogeneous WSNs

Homogeneous WSNs are networks where all nodes share the same hardware. Initial battery capacity, rate of transmission, competition radius and many other nodes capabilities are equal. In this context, usually one of these capabilities is used to elect the next cluster head for a cluster.

Equal size clusters

Many of the first clustering protocol proposed by researchers organise nodes into sets of equal size.

Low Energy Adaptive Clustering Hierarchy (LEACH) [34] is one of the pioneering routing protocols that introduced the idea of clustering into the field of WSNs. Unlike most of the clustering protocols, which use the node residual energy for cluster election, LEACH uses a probabilistic function. All cluster heads can directly communicate with BS, i.e., multi-hop communication never takes place. Once a node has been elected as a CH it cannot take the same role in the next cluster election.

HEED [24] is a protocol composed of the following two phases: (i) clustering, and (ii) network operation. During the clustering phase, CHs get elected based on the residual energy, and member nodes join the closest CH[1]. During the network operation phase data messages get delivered from the members to the BS. Clustering and network operation phases are repeated over time. HEED generally prevents two nodes within the same transmission range from becoming CHs. As reported in [24], sensor nodes close to the BS deplete their energy faster with respect to nodes that are farther away. This problem is referred to as hot spot problem. In fact, while all CHs will have the same amount of average intra-traffic communication (i.e., the traffic inside a cluster) CHs close to the BS have a higher inter-cluster communication (i.e., relay traffic amongst CHs).

Distributed Weight-based Energy-efficient Hierarchical Clustering protocol (DWEHC) [35] is an equal size clustering based protocol for WSNs. It optimises intra-cluster communication by introducing multi-hop transmission within the clusters. All sensor nodes execute DWEHC individually to decide whether to be a cluster head or a member node. DWEHC clustering formation phase is based on HEED topology. Resultant clusters arrangement is well-balanced and leads to enhanced network lifetime.

Unequal size clusters

Voluminous literature have been developed on devising energy efficient unequal size clustering protocols for WSNs.

Energy Efficient Unequal Clustering (EEUC) algorithm [36] for WSNs is one of the first approaches to produce clusters of unequal size. EEUC is based on the idea that a larger cluster size should be used when the CH resides in zones farthest from BS whereas zones nearest to BS should be populated with a considerable amount of smaller clusters. This approach would minimise excessive overhead burden on cluster heads nearest to BS and should alleviate the energy hole or hot spot problem.

Unequal clustering algorithm based on HEED (UHEED) [29] is another unequal size clustering based protocol for WSNs. UHEED incorporates the idea of EEUC protocol into HEED in order to build unequal size clusters. The size of a cluster depends on its distance from the BS. The farther away CH is from the BS, the larger its competition radius is. In other words, clusters that are farther away from the BS have a larger radius with respect to clusters nearer to the BS. UHEED reduces the hot spot problem and increases network lifetime when compared to HEED and LEACH.

Rotated Unequal HEED (RUHEED) [20] uses an unequal size clustering based approach that not only alleviate the hot spot problem but also enhances the network lifetime. RUHEED is composed of three stages that are CH election; clusters formation; and CH rotation. HEED is used to elect CHs based on its residual energy and communication cost. EEUC concept, which is based on the sensor node distance from the BS, is used in order to establish unequal sized clusters. During CH rotation phase, current CH selects the member nodes with the highest energy and directly designates it as the next cluster head. Rotation strategy avoids re-clustering of the network thus network lifetime is improved. Re-clustering of the network takes place when any of the sensor nodes drain its entire energy. RUHEED preserves energy and minimises the number of cluster election and cluster formation phases.

[1]Communication costs can be considered to elect or join a CH.

ER-HEED [21] is a clustering protocol that enhances performance of HEED by introducing CHs role rotation inside clusters. ER-HEED is composed of three stages that are cluster head election, cluster formation using HEED and cluster head rotation. Like RUHEED, CHs nominate the next CHs that have the highest residual energies. This concept of CH selection within the cluster member nodes reduces the number of cluster elections. HEED based cluster head election is performed only when any of the sensor nodes depletes its energy completely. ER-HEED performance in terms of first node dies measure criteria is far superior to RUHEED, HEED and UHEED.

2.1.2 Clustering Heterogeneous WSNs

While WSNs have homogeneous nodes, heterogeneous WSNs introduce nodes that can have differences in the following features: (i) energy level; (ii) data rate; (iii) transmission range; (iv) aggregation performance; and (v) processing capabilities. Heterogeneity affects significantly the network lifetime and lessens network response time [37]. Different protocols can make different assumptions about the heterogeneity of the WSNs. This section describes various clustering algorithms that have been devised for heterogeneous wireless sensor networks. It is possible to split them in two groups basing on how they approach the heterogeneity problem, i.e. whether they work at MAC or at network level. Both groups include protocols producing clusters of equal and unequal size.

Network level

The following protocols work at network layer by clustering the network with different approaches.

Distributed energy-efficient clustering algorithm for heterogeneous WSNs (DEEC) [6] is an equal size clustering protocol. DEEC cluster head election is based on a probability that is calculated by considering the ratio of the residual sensor node energy and the network average energy. The CH role is rotated among sensor nodes on the basis of their residual energies. This ensures a uniform energy consumption over the entire network. Sensor nodes that have the highest residual and highest initial energies will more likely elect to become cluster heads. BS broadcasts the network average energy information to all wireless sensor network nodes.

Distributed energy balance clustering Protocol for heterogeneous WSNs (DEBC) [38] is a clustering protocol for heterogeneous WSNs. DEBC assumes that sensor nodes have heterogeneous energy levels. The cluster head election is based on the sensor node residual energy. Sensor nodes that have the highest initial energy and the highest residual energies are highly probable to elect to become cluster heads. Simulation results shows that the performance of DEBC is superior to LEACH and SEP[39].

Distributed Clustering Algorithm with Load Balancing (DCLB) [40] is able to form clusters which efficiently balance the energy load of inter-cluster communication. Size (range) is important in terms of energy efficiency and balancing load in multi-hop communication of CHs. This avoids energy inefficiency and produces balanced load of cluster. Balanced inter-traffic communication is achieved by using clusters with different sizes at each step.

In [41] Distributed Cluster Head Election (DCHE), a distributed election approach for Heterogeneous WSNs, is presented. The election of cluster heads is based on a weighted probability. Member nodes communicate with their CH and then CHs communicate the aggregated information to the base station. Three different types of nodes are considered and all have different threshold. The weight assigned to each node will decide the probability to elect as cluster head for each node type.

Energy efficient heterogeneous clustered scheme for wireless sensor networks (EEHC) [42] is a clustering protocol for heterogeneous WSNs. In EEHC, a percentage of sensor nodes are equipped with various levels of battery capacity. EEHC aims at enhancing network efficiency and reliability. Like DEEC and DEBC, the cluster head election probability of EEHC depends on sensor node residual energies.

A stable election protocol for clustered heterogeneous wireless sensor networks (SEP) [39] is a heterogeneous protocol and intents to enhance network lifetime according to the first node dies network lifetime measure. SEP assumes two different types of nodes that are normal and advanced sensor nodes. CH election is based on sensor node initial and residual energies. Simulation results show that SEP prolongs network lifetime and average throughput.

Feedback Mechanism–based Unequal Clustering (FMUC) [7] is a feedback mechanism based unequal heterogeneous protocol. FMUC is specifically designed to avoid the energy hole problem when balancing the energy load in application-based WSNs. Initially, FMUC divides the network into a given number of layers and assumes each cluster belongs to a certain level. Then, the energy consumption of each cluster is computed. As a consequence, the size of each cluster is set according to the ratio of the energy consumption of each layer. Finally, clusters send their sizes as a feedback to the sink which broadcasts the collected values into the network. All nodes of the WSN receives the feedback values but only the cluster heads change their competition radius according to received values.

Energy Efficient Clustering Protocol to Enhance Performance of Heterogeneous Wireless Sensor Network (EECPEP-HWSN)[43] is an energy efficient clustering protocol for heterogeneous WSNs. This is designed for WSNs that have three different energy level nodes that are normal, advanced, and super. Initial energy, hop count, and residual energy are used for cluster head selection. EECPEP-HWSN outperforms well-known protocols such as LEACH, DEEC, and SEP.

Energy-Coverage Ratio Clustering Protocol (E- CRCP) [44] is an adaptation of LEACH to heterogeneous WSNs. E-CRCP improves the cluster head selection of LEACH by considering energy consumption. More precisely, E-CRCP considers the lowest energy consumption in each communication iteration and the maximum cluster head coverage ratio. E-CRCP balances both the cluster head selection and the system energy load to extend the system's life cycle. This protocol considers nodes with different initial energy levels and homogeneous data transmission rates.

In [45] the authors propose Heterogeneous DEEC (HetDEEC), a 3-level heterogeneous network model for WSNs with 1-level, 2-level, and 3-level heterogeneity. The node energy level is then used as a weight in the election probability and threshold function to select cluster heads and their cluster members. The authors implement three version of DEEC, HetDEEC-1, HetDEEC-2, and HetDEEC-3 respectively according to the type of nodes in the network. HetDEEC improves network lifetime with respect to DEEC. HetDEEC lifetime performance increases for increasing energy levels.

The authors in [46] propose the Energy-Efficient TDMA (EC-TDMA) clustering protocol. This dynamically changes the length of TDMA frame according to the number of sensor nodes and the sensor node traffic load within a cluster. This reduces the idle listening and improves channel utilisation, thus prolongs networks lifetime. A similar approach is presented in [47] where nodes are assumed to have different rates. Sensor nodes forward data to the cluster head only if significant events are observed. This improves energy efficiency for low data rate sensors.

In the article Fuzzy-Based Dynamic Time Slot Allocation for Wireless Body Area Networks[48] the authors propose a wireless body area network for a real-time patient monitoring system. This consists of a sensor layer, body controller layer, and a central coordinator layer. Clustering is used for routing the data packets to the coordinator node. The slot allocation is dynamic and is designed by using a fuzzy logic that considers residual energy, buffer memory and node data rate.

MAC level

Some protocols use the concept of Time Division Multiple Access (TDMA) to assign the communication channel to one node at a time, for a certain time slot. Protocols which work at MAC level usually variate the TDMA slot size or the TDMA frames assignment to the nodes according to the network heterogeneity.

Authors in [49] build a variable TDMA slot size for optimising bandwidth usage and accommodate different data rate requirements for personal communication systems. They review different technologies that support Constant Bit Rate (CBR) and non-CBR traffic and propose an efficient intra-traffic communication mac layer that accommodates traffic with different requirements and different patterns.

Authors in [50] propose TDMA with variable frame size and extended Kalman filter to track the wireless channel quality and predict optimal frame size for data transmissions. The optimal size of a frame reduces transmission due to errors thus improving energy efficiency.

In [51], the authors use Dynamic Slot Assignment (DSA) to model a TDMA-based MAC protocol suited for bursty traffic environments under low traffic conditions with large clusters. The DSA scheme adapts to changing traffic loads. By considering the network activity a TDMA slot is dynamically assigned to each sensor node which has data to send. Their protocol proves a substantial improvement in the energy consumption and channel utilisation for low rate sensor networks.

The main topic of [52] is to evaluate the performance of WSNs under different logical topologies and different clustering protocols. The authors study the effect on the network lifetime for different packet sizes. More precisely, they measure the effects on energy consumption and throughput when the nodes packet size varies. They simulate packet size values ranging from 64 to 4098 bits. As a result, the authors observe how the TDMA maximum frame time influences the node packet size: in particular, the maximum TDMA frame time should increase as well as the packet size increase.

The authors in [53] present Dynamic Packet Length Control scheme for WSNs (DPLC). The authors argue that wireless link qualities can be greatly affected by environmental factors at different locations thus dynamic packet length control should be used in order to improve energy efficiency and throughput. DPLC allows to send messages of variable size. The link estimator within DPLC dynamically estimates the optimal packet length for transmission. Based on this, the DPLC module at the sender decides how many messages should be aggregated, or how many frames the message should be fragmented into.

The authors in [54] study the correlations amongst data from neighboring nodes in heterogeneous WSNs. Their aim is to improve the channel utility. For this reason, they present a Dynamic Time Division Multiple Access Control (DTDMA) protocol. DTDMA adjusts the allocation of time slots based on correlation to form the variable-length data frames. In other words, the time slots are allocated dynamically to improve the channel utility as well as decrease the latency and energy consumption. The protocol is divided in two phases: first, the BS dynamically chooses the most fitting nodes to be CH for a certain time period. Second, the TDMA time slot allocation is adjusted from the cluster heads and then the resulting schedules are broadcasted to the CHs cluster members.

In [55] the authors state that a good solution to cope with changes in network topology or traffic load is to adapt the TDMA frame size and time slot assignment to the nodes. For this reason they develop an energy-efficient TDMA protocol for clustered wireless sensor networks (EC-TDMA) which works at intra-cluster communication level. EC-TDMA is able to dynamically adapt the length of the TDMA frame according to the number of sensors and their traffic load within a cluster to reduce the idle listening times thus improving channel utility. This prolongs the networks lifetime thanks to the energy saved in reducing the communication.

In [13] the authors face the need of sampling an area with different types of loads. They adapt the CHs sensing rates in some specific regions that are triggered significantly less than other regions. To this ending the authors introduce the Adaptive Distributed Hierarchical Sensing (ADHS) algorithm. This changes the sampling rates of the CHs based on the variance of the sampled data without compromising the accuracy of the sensed area. The authors introduce different case studies that include WSN with high-rate subareas.

The authors in [56] propose Enhanced Threshold Distributed Energy Efficient Clustering (ETDEEC) which considers two-level and three-level heterogeneous networks. Nodes are considered heterogeneous when they have different sensing ranges, different packet size and different initial energy levels. They categorise the nodes into three types depending on their initial energy. The CH election probability is proportional to the type of node, the residual energy, and the distance of the node from the BS. The authors evaluate the protocol for non-uniform node distribution. They leave the problem of variable transmission rate as an important future work since WSNs are evolving into multimedia systems or the Variable Bit Rate requirements.

The authors in [57] propose Bandwidth Efficient Cluster-based Data Aggregation (BECDA) for the effective data gathering with in-network aggregation. BECDA considers WSNs with heterogeneous nodes in terms of energy and rate. The optimal approach is achieved by intra and inter-cluster aggregation on the randomly distributed nodes with the variable data generation rate. The proposed algorithm uses the correlation of data within the packet for applying the aggregation function on the variable data generated by nodes. The election probability is based on residual energy, node connectivity, density and number of rounds.

In [58] the authors propose a model where member nodes send a variable amount of data. The CH adjusts the allocation of time slots based on correlation to form the variable-length frames. This improves energy efficiency and achieves good performances.

In [59] the research problem is how to perform flow routing amongst the nodes when the bit rate from each source node is time-varying. When the average rate is known an optimum energy efficient flow routing can be obtained. When the average rate is unknown but can be estimated within a a certain error bound ε, then the network lifetime provided by the proposed flow routing solution is within $\frac{2\varepsilon}{1-\varepsilon}$ from the optimum.

The authors in [60] describe an energy-efficient sleep scheduling mechanism with similarity measure for WSNs (ESSM). This schedules sensors into active or sleep mode to improve energy efficiency. The optimal competition radius is estimated to organise all sensor nodes into several clusters. By considering the data collected by member nodes, a fuzzy matrix is obtained to measure the similarity degree. A correlation function that is based on fuzzy theory is defined in order to divide the sensor nodes into different categories. The redundant nodes will be selected to put into sleep state in the next round in order to ensure the data integrity for the entire network.

Multi-hop graph-based approach for an energy-efficient routing (MH-GEER) [61] aims at distributing energy consumption between clusters. The clustering is performed in a centralised way. The routing phase builds a dynamic multi-hop path between cluster heads and the base station. MH-GEER explores the energy levels in the entire network which are used to select the next hop in a probabilistic and intelligent way.

Finally, the surveys in [62–64] review the state of art of different clustering protocols.

It is worth mentioning that none of the protocols described in this section combine different nodes capabilities (i.e. data transmission rate and battery level) for cluster head election or cluster formation.

2.1.3 Clustering protocols with harvesting

Harvesting is the capability of sensor nodes to be able to harvest energy either from a dedicated or an opportunistic ambient source such as solar, thermal, wind, and vibration. If nodes energy consumption is lower than the amount of energy they are able to harvest, than perpetual network lifetime can be achieved.

The authors in [65] give a detailed list of various sources that can be used to harvest energy in WSNs. They can be categorised into ambient sources and external sources.

Ambient sources are:

- Radio Frequency-based energy harvesting where RF-based energy harvesting received radio waves are converted to DC power after conditioning;

- Solar-based energy harvesting where solar energy is an affordable and clean energy source useful to eliminate the energy problem in WSNs. The photo-voltaic effect converts solar rays into DC power when certain semiconductor materials are exposed to sunlight. We remark that it is not possible to perform solar energy harvesting during the night. Thus, developers have to ensure the highest possible efficiency during daylight hours to guarantee the viability of solar-power;
- Thermal-based Energy Harvesting where it is possible to convert heat energy into electrical energy exploiting the seebeck effect. This requires a load to be attached across the heated and cold faces of a ThermoElectric Generator (TEG) for thermal energy harvesting. This can be done at different scales, from large to small. In WSNs, we often need to keep the scale as small as possible. In this scenario, for instance, it can be interesting to generate power from human body temperatures;
- Flow-based energy harvesting generally uses turbines and rotors to convert rotational movement into electrical energy using electromagnetic induction principal.

External sources are:

- Mechanical-based energy harvesting that is performed with sources such as vibrations, pressure, and stress–strain. To do this, a suitable Mechanical-to-Electrical Energy Generator (MEEG) is needed. A MEEG uses either electromagnetic, electrostatic, or piezoelectric mechanisms to harvest energy.
- Human-based energy harvesting that is used in Wireless Body Area Network (WBAN). In these networks, sensor nodes are deployed on or inside of the human body to monitor physiological parameters continuously. These nodes need to be operational for long periods of time or even for the lifetime of the humans being monitored. Human-based energy harvesting can be categorized as activity based harvesters and inherent physiological parameters based harvesters. More precisely, energy can be harvested from humans in several ways, such as through locomotion, changes in finger position, body heat, and blood flow. Nevertheless, the main challenge still is to miniaturise sensors to make them easier for human adoption.

In [66] the following different energy harvesting approaches are investigated: (i) energy harvesting combined with simultaneous data decoding (a trade-off between the amount of energy that can be stored for future use and the amount of energy that should be spent for signal decoding); (ii) energy efficient operation of wireless sensor networks, making use of appropriate routing schemes or scheduled operation of sensor nodes; (iii) mobile chargers which stop at optimal locations to perform charging; and (iv) energy sharing. The approaches (iii) and (iv) are chosen in a 2 step protocol combining a mobile charger which moves inside the network toward the next discharged CH to overcharge it and an energy trading between overcharged CHs and other nodes. The energy trading takes place inside each cluster: a CH is chosen getting the node which has the highest number of neighbours inside its inclusion circle. In the first stage, the mobile charger follows the optimal path to reach the next discharged

CH then it stops (the mobile charger can "decide" to stop or move any time a time interval is passed) and it overcharges it. Then overcharged CHs sell their energy to their cluster members with no competition (the number of seller nodes is significantly larger than the number of buyer nodes thanks to the first stage). It is not CHs which offer the energy, but the nodes that broadcast a message (composed of an ID and the amount of energy needed) and wait for one or more CHs to give energy to them. CHs will serve near nodes first. Simulations produced with Omnet++ and Castalia show that EH-WSN protocol works best with greater nodes inclusion circle radius than smaller ones.

The authors in [67] uses a cross-layer cooperative TDMA scheme instead of a classical one to optimise the CHs relaying performance. The CH role is alternated between the nodes using duty cycling as a function of their individual energy harvesting capabilities. This protocol define the optimal number of clusters according to the intensity of the energy source (which is solar energy in the paper). The protocol is based on LEACH. The CH choice is based on a probability function that uses duty cycle mechanism where a node cannot become CH before n duty cycles are passed. This number is computed for each CH as the ratio of the CH required energy to its allocated energy rounded to the next integer value. The protocol can include cooperation (cooperative transmission protocol) ECO-LEACH or not (ENCO-LEACH). The cooperative transmission protocol makes use of the energy unconsumed in data transmission to relay undelivered packets from cluster members to CHs and also from CHs to the sink node.

2.1.4 Clustering protocols with Machine learning

Machine learning (ML) is a late 1950s technique for artificial intelligence (AI) and for the definition of computationally viable and robust algorithms. During the years, ML has been applied to different fields such as bioinformatics, speech recognition, spam detection, computer vision, fraud detection, and advertising networks. ML learning techniques have been used for many tasks like classification, regression and density estimation.

From the point of view of [68] and [69] machine learning can be defined as:

- The development of computer models for learning processes that provide solutions to the problem of knowledge acquisition and enhance the performance of developed systems.
- The adoption of computational methods for improving machine performance by detecting and describing consistencies and patterns in training data.

Applying ML to the field on WSNs routing protocols is a process which has both pros and cons. Some of the ML algorithms best properties are their ability to automatically calibrate according to newly acquired knowledge, their generally low complexity and their capability to uncover correlation between sensor data and improve sensor deployment for maximum data coverage. On the other hand, ML algorithms drawbacks lie in the high amount of computational power they need, which escalates when requiring more accuracy, and the large set of existing data and samples they require to achieve high generalisation capabilities.

There are several ML techniques that can be applied to WSNs to perform clustering. These techniques try to improve node clustering and data aggregation mainly in two ways:

- compress data locally at CHs by efficiently extracting similarity and dissimilarity (e.g., from faulty nodes) in different sensors' readings.
- CHs election, where appropriate cluster head selection will significantly reduce energy consumption and enhance the network's lifetime.

Some classic ML approaches have been investigated in the past to check their suitability for WSNs clustering and data aggregation [70]. Clustering can be performed based on (i) neural networks, (ii) decision trees, and (iii) role-free CHs selection while data can be aggregated using (i) self-organizing map (SOM), (ii) learning vector quantization, (iii) principal component analysis, (iv) k-means algorithm, and (v) decentralized learning. However, very few clustering protocols have been implemented through the approaches listed above. Moreover, most of them do not compare their results with well known adaptive clustering protocols such as HEED, ER-HEED and LEACH. One ML based protocol which makes a comparison with other existing protocols is LEACH Genetic Algorithm (LEACH GA)[71]. LEACH GA is a genetic algorithm based on LEACH [72]. LEACH GA modifies the LEACH algorithm, adding a preparation phase only once before the set-up phase of the first round. Initially, nodes perform cluster head selection, then they send messages including (i) whether they candidate as cluster head, (ii) their node IDs, and (iii) their geographical positions to the base station. At that point, the base station uses data received from nodes to determine the optimal probability p_{opt} by performing GA operations, then it broadcasts this value to all nodes. The following set-up and steady-state phases are performed in every round and are the same as LEACH. Recently, [73] has proposed a comparison of LEACH GA performance over LEACH and LEACH-C using MATLAB simulation tool. In the simulations, nodes are randomly distributed in an area of 100m x 100m with the base station located at the centre point (50, 50). According to the simulation results, LEACH-GA performs better when compared to LEACH and LEACH-C under different initial energy and probability thresholds. In particular, LEACH-GA increases the network lifetime of 54% over LEACH and 47% over LEACH-C, on the average. However, simulation results do not take into account novel clustering protocols proposed after LEACH and its variations [24][21][14], which are proved to perform better under various situations.

Table 2.1 shows a categorisation of the clustering protocols that are described in this section. Clustering protocols are categorised by considering several attributes.

2.2 Rotation

Rotation has been widely proposed in literature in order to balance the energy consumption and prolong the WSN lifetime. The authors in [18] propose an energy efficient protocol consisting of clustering, cluster head selection/rotation and data routing method to prolong the lifetime of sensor networks. The WSN area is organised in concentric circles. Nodes that are exactly midway between the two concentric circles have a higher probability of becoming CH

TABLE 2.1: Comparison of well known Clustering Protocols for WSNs

Protocols	Node Deployment	Heterogeneous	Homogeneous	Clustering Method Distributed(D) Centralized(C)	Equal Sized Clusters	Unequal Sized Clusters	Rotation	Location Awareness	Harvesting	Machine Learning	Probability	No Probability
LEACH[72]	Random		✓	D	✓			N			✓	
SEP[39]	Random	✓		D	✓			N			✓	
HEED[24]	Random		✓	D	✓			N			✓	
ERHEED[21]	Random		✓	D	✓		✓	N			✓	
UHEED[29]	Random		✓	D		✓		N			✓	
LEACH-GA[71]	Random		✓	D				N		✓	✓	
RUHEED[20]	Random		✓	D		✓	✓	N			✓	
DEEC[6]	Random	✓		D	✓			N			✓	
DWEHC[35]	Random		✓	D	✓			N			✓	
IEEUC[36]	Random		✓	D		✓		N			✓	
DEBC[38]	Random	✓		D	✓			N			✓	
DCLB[40]	Random		✓	C	✓			Y				✓
EEHC[42]	Random		✓	D	✓			N			✓	
DCHE[41]	Uniform	✓		D				N			✓	
FMUC[7]	Random	✓		D		✓		N			✓	
REECHD[14]	Random	✓		D	✓		✓	N			✓	
EHWSN[66]	Random			D	✓			N	✓		✓	

in the first election. Clusters are formed only once during the lifetime of the sensor network, then rotation takes place. This results in substantial saving of energy. Rotation takes place after a fixed amount of rounds which is calculated by using a mathematical energy model. This considers the average energy consumption for a node playing the CH and member roles. The authors in [19] propose an Area-Partitioned Clustering where the network is statically divided in concentric circles.

Rotation is used to provide a balanced consumption of energy. Three rotation strategies are proposed where rotation is based on a predetermined schedule or residual energy is used when the energy consumption is lower than expected. While a rotation based on a predetermined schedule may reduce overhead messages, energy consumption models usually estimate the energy consumption in average conditions (e.g., average distance between a member and its CH) and simplified settings (e.g., virtual grids). This may lead to lifetime performance degradation when the average case is not representative. This problem is solved by the adaptive rotation approach which incorporates node residual energy inside its energy consumption model.

Rotation is usually adaptive since it considers the node residual energy [20, 21, 74] In [21] the authors extend the HEED [24] protocol with rotation. They use HEED for the cluster formation tha rotation based on residual energy takes place. HEED cluster formation is performed every time a node depletes its energy completely. In [20] the authors extend the UHEED [29] protocol with rotation. They use UHEED for the clustering the WSN, then rotation based on residual energy takes place. UHEED clustering is performed every time a node dies. In [74] the authors use the CH current energy load to calculate a threshold for rotation. This reduces the premature death of cluster heads. A splitting policy divides the cluster when no cluster member can afford to be CH. Although rotation based on residual energy is suitable for a wide range of WSN settings, it may be inefficient for heterogeneous networks. The authors in[75] propose the Energy-Coverage Ratio Clustering Protocol (E-CRCP) for heterogeneous energy wireless sensor networks. They define an energy consumption model and an optimal number of clusters in order to minimise energy consumption. CHs that maximise the coverage are elected. CH nodes that consume a large amount of energy are replaced in the next iteration. Members join their nearest CH. Although this approach improves energy efficiency for various WSN simulation settings it does not consider node rate heterogeneity.

2.3 Routing WSNs

Clustering and routing are tightly connected. When building sets of neighbour nodes (i.e. clusters) and electing one of them as CH, it is fundamental to ensure that data sensed inside the cluster radius reach the BS. However, this is not as easy as it could seem. In fact, plenty of literature can be found on this topic. However, very few approaches consider the energy spent for the routing tree generation and the energy efficiency of the generated routing tree.

In [76], the authors perform a field study on existing routing techniques applied to WSN, highlighting the performance issues of each technique. They classify the routing techniques into the following three categories: flat, hierarchical, and location-based. Moreover, depending on the protocol operation they include, further classifications as multipath-based, query-based, negotiation-based, QoS-based, and coherent-based routing have been studied. A similar but more up-to-date work has been presented in [77]. In addition, they explore in details possible optimisation of existing routing algorithms as well as the unsolved issues and research gaps. An even more recent work [78] focuses on the factors that affect the energy aware routing in WSNs. In particular, the authors suggest various approaches to make existing routing techniques energy aware. In a recent work [79], the authors propose a Tree-Based Energy-Balance Routing in which each node selects its father amongst its neighbours on the basis of the communication distance between sensor nodes and the BS, the nodes residual energy level, the energy required to transmit the data to the BS, and the number of associated child nodes. The authors believe that this leads to a uniform energy utilisation and offers a better energy balance mechanism with respect to other general purpose routing algorithms.

HEED [24] and UHEED [29] build the routing tree between CHs in a peer to peer fashion. Each node sends a broadcast to search for neighbour CHs and connects to them. This approach only guarantees connectivity if CHs can increase their radius enough to reach the BS, and it doesn't allow to build the best possible routing tree. Other clustering protocols which implement rotation, like ER-HEED [21] and RUHEED [20], try to build a routing tree in a centralised way, starting from the BS. The BS sends a broadcast to find at least one CH, which will do the same to find neighbour CHs. This implies a lot of communication between the nodes. The only way to check for connectivity is that every CH sends a control message and waits for an ack from the BS (through the upper-level CHs). However, this synchronisation phase adds extra control messages. Finally, there are protocols like FMUC [7] which create clusters only between nodes at the same level. In this case, the only way to avoid the synchronisation phase is to choose the levels height accordingly to the nodes competition radius. Without this assumption, nodes would still need to find a way to understand if they are connected to the BS.

The authors in [62] survey the state of art clustering for Computational Intelligence (CI) and Machine Learning (ML). They compare the clusterings by using various parameters such as the data delivery rate; data aggregation; network lifetime; the scalability for an increasing number of nodes; centralised and distributed clustering; homogeneous and heterogeneous nodes (i.e., whether sensors have or not the same performance); the energy model and fault tolerance. Algorithms that are based on Swarm Intelligence (SI) seem the most energy efficient choice when artificial intelligence is considered. The authors in[63] provide a comparative analysis for sensor node deployment schemes and energy efficient clustering protocols. The state of art energy efficient cluster-based and grid-based techniques in WSN are evaluated by considering different parameters such as cluster formation metric, energy consumption, and lifetime. The authors also discuss the design issues and open research challenges. A comparative analysis is presented that helps in selecting the most appropriate technique for specific requirements. Although the surveys in [63] and [62] present a plethora of different clustering approaches, none of them seems to combine CH election and routing tree definition when heterogeneity in terms of rate and residual energy are considered.

Chapter 3

Rotating Energy Efficient Clustering for Heterogeneous Devices (REECHD)

This chapter introduces REECHD. This is an efficient HEED based clustering protocol that introduces a novel leader election protocol which, not only considers the node residual energy, but also the node transmission rate.

3.1 Notation

In the rest of this document the notation of table 3.1 is used. This describes the basic notation used for clusters, nodes and routing tree.

3.2 Radio Model

MIT radio model (see section 1.2.2), adopted inside the simulator, is recalled here.

Transceiver circuitry of a sensor node consumes $E_{elec} = 50nJ/bit$. Sensor node amplification energy E_a depends on the distance d between sender and receiver. When $d < d_0 = 75m$, E_a becomes $E_{fs} = 10pJ/bit/m^2$ and when $d \geq d_0 = 75m$, E_a reduces to $E_{mf} =$

TABLE 3.1: Notation and definition

Notation	Meaning
CR	a WSN cluster; $\{n_1, \ldots, n_k\}$ are elements in CR
CR_{CH}	the WSN cluster that contains the cluster head CH
M_{CH}	the member nodes of the cluster defined by CH; $M_{CH} = CR_{CH} \setminus \{CH\}$ and $\{m_1, \ldots, m_h\}$ are elements in M_{CH}
$T(n)$	the bits transmitted by a node n in a TDMA
$R(n)$	the bits received by a node n in a TDMA
E_{r_n}	the residual energy of a node n
E_{max_n}	the maximum energy of the node n (it is equal to a fully charged battery)
R_0	the transmission radius a CH uses to alert nodes of its presence. This defines the size of a cluster.
CH_f	the cluster head father of the cluster head CH in the routing tree
CDN_{CH}	the set of CH children of the cluster head CH in the routing tree
$\Re_{BS}$	the routing radius of the BS
$\Re_{CH}$	the routing radius of a CH

$0.0013 pJ/bit/m^4$. Eq. 1.1 defines the transmission energy that is needed in order to send k bits at distance d while Eq. 1.2 defines the reception energy that is spent for receiving k bits.

3.2.1 Network Model

In the chosen network model nodes are not mobile and are distributed in a two dimensional area. Nodes are heterogeneous. These have different initial energy and different data transmission rates within a defined maximum and minimum. Nodes have the same processing and aggregation capabilities, have a unique ID and can transmit at various power levels which depend on the distance of the receiver.

The BS is not mobile, has no energy constraints and has more communication and processing capabilities with respect to normal sensor nodes. The BS is located outside the WSN area. More precisely, the WSN is contained into a grid of size 100 to 100 meters and the BS can be placed at two different positions: (175, 50) and (125,50) (see Figure 3.1 for details)

REECHD clustering protocol includes the following phases: (i) cluster formation and CH election; (ii) CH routing tree definition; (iii) network operation phase; (iv) rotation in the role of CH within each cluster. Routing amongst CHs is defined in order to deliver data to the BS. Routing is usually achieved by defining a routing tree rooted at BS. During the network operation phase data gets delivered to the BS. A widely used data delivery model is based on the concept of TDMA [7, 14, 24, 20, 72]. A TDMA includes the following two activities: (i) each member node sends one message of variable size to its CH (i.e., intra-traffic communication); and (ii) all CHs send the gathered data to the BS via the defined routing tree (i.e., inter-traffic communication). In other words, a TDMA starts from the collection of data from the member nodes and ends when all the data reaches the base station. A round is composed of multiple TDMAs (five for all simulations).

In the following some formulae that are used to model intra-traffic and inter-traffic communication, and TDMA, are introduced.

Eq. 3.1 defines the intra-traffic communication $R_{intra}(CH)$. This defines the bits received by the cluster head CH from its member nodes in a TDMA. $T(m_i)$ is the transmission rate of the member node m_i in M_{CH}. This set contains all member nodes except CH.

$$R_{intra}(CH) = \sum_{i=1}^{|M_{CH}|} T(m_i) \tag{3.1}$$

Eq. 3.2 defines the *forwarded intra-traffic* that a cluster head CH forwards toward the BS.

$$T_{intra}(CH) = max(\sum_{i=1}^{|CR_{CH}|} T(m_i)(1-AR), M_{min}) \tag{3.2}$$

where $\sum_{i=1}^{|CR_{CH}|} T(m_i)$ is the total cluster rate that is the summation of the number of bits each node of the cluster CR transmits in a TDMA (this includes the cluster head CH). AR is the aggregation rate and M_{min} is a constant that denotes the minimum amount of intra-traffic (i.e., bits) CH must forward in a TDMA. When $AR = 0$, CH packs all messages received by the members (during a TDMA), adds its data and forwards them to the next hop. In this case no aggregation takes place. When $AR = 1$, then CH aggregates its data with all messages received by the members in a TDMA by producing a message of minimum size M_{min}. This size is set either to 1024 or 512 in simulations.

$R_{inter}(CH)$ denotes the total amount of bits received by CH from its children CH nodes in a TDMA (i.e., the inter-traffic received data). $R_{inter}(CH)$ can be calculated as follows:

$$R_{inter}(CH) = \sum_{i=1}^{|subtree_{CH}|} (T_{intra}(CH_i)) \tag{3.3}$$

where $T_{intra}(CH_i)$ is the intra-traffic that CH_i forwards to its father (see Eq. 3.2 for details), $subtree_{CH}$ contains all cluster heads that are in the CH routung subtree that is rooted at CH. The total amount of bits sent by a CH towards the BS (i.e., the inter-traffic data) is:

$$T_{inter}(CH) = T_{intra}(CH) + R_{inter}(CH) \tag{3.4}$$

It is worth noticing that a CH forwards the inter-traffic received $R_{inter}(CH)$ with no aggregation while the intra-traffic $T_{intra}(CH)$ is aggregated (see Eq. 3.2 for details).

The energy CH spends to forward data to its father (i.e., the inter-traffic transmission rate) is denoted with $E_{inter}(CH, CH_f)$ and can be calculated as follows:

$$\begin{aligned} E_{inter}(CH, CH_f) = E_{Tx}(T_{intra}(CH), d_{CH,CH_f}) +& \\ E_{Rx}(R_{inter}(CH)) +& \\ E_{Tx}(R_{inter}(CH), d_{CH,CH_f}) & \end{aligned} \tag{3.5}$$

where d_{CH,CH_f} is the distance between the CH and its father CH_f, the first adding is the energy CH spends for forwarding its intra-traffic to the its father, the second adding is the energy CH spends for receiving data from its children nodes while the last adding is the energy CH spends to send the data that are received from the children nodes to the father node.

The energy a node m spends when playing the member role can be computed as follows:

$$E_m(CH) = E_{Tx}(T(m), d_{m,CH}) \tag{3.6}$$

3.3 Low-rate-high-energy node depletion problem

Cluster head selection is usually based on the node residual energy, i.e., the node with the highest energy is usually elected as CH. This selection strategy can cause a quick energy depletion when a node that has high energy and low rate is elected. This is referred as low-rate-high-energy node depletion problem.

Table 3.2 summarises a simple heterogeneous WSN that is used to describe the low-rate-high-energy node depletion problem. The origin of the coordinate plane is set at the low-left corner of the WSN area which has a square shape (see Figure 3.1). The base station is at 125 meter on the x-axis and 50 meter on the y-axis. Three sensors populate the WSN area.

TABLE 3.2: Simple heterogeneous WSN to describe the low-rate-high-energy node depletion problem.

node	coordinates	T(n)	E_{max}	$E_m(CH)$	$E_{inter}(CH,BS)$
n_1	(95m,5m)	3000	1J	$198\mu J$	$90\mu J$
n_2	(64.9m,25m)	500	5J	$33\mu J$	$220\mu J$
n_3	(95m,45m)	500	5J	$33\mu J$	$220\mu J$

TABLE 3.3: Energy based CH election vs Rate based CH election

Election strategy	FND	Dead	E_r	WSN energy
Residual Energy	5050	n1	n1=0 n2=4.43J n3=4.43J	8.86J
Max Rate	11111	n1	n1=0 n2=4.63J n3=4.63J	9.26J

E_{max} denotes the initial energy of a node while $T(n)$ its transmission rate. $E_{inter}(CH,BS)$ is the energy the node spends in a TDMA while playing the cluster head role. This energy consumption is calculated by using Eq. 3.5. $E_m(CH)$ is the energy the node spends in a TDMA while playing the member role. This energy consumption is calculated by using Eq. 3.6. As an assumption, at each round (one round is composed of 1 TDMA) exactly one node is elected as CH; member nodes always connect to the CH and the CH directly connects to the BS.

Table 3.3 compares the first node die performance of two different election strategies that are “Residual Energy” and “Max Rate” based strategies. The former selects the node with the highest energy, while the latter selects the node with the highest transmission rate. The Residual Energy strategy would select n_2 and n_3 as CH while n_1 would play the member role. The nodes n_2 and n_3 would play the CH role in alternation until the node n_1 depletes completely its energy. A total amount of 5050 rounds would be performed before $n1$ dies with a total WSN energy left of $8.86J$. The Max Rate strategy would select n_1 as CH while $n1$ and $n2$ would play the member role. In this case a total amount of 11111 rounds would be performed before $n1$ dies with a total WSN energy left of $9.26J$. The Max Rate strategy outperforms the Residual Energy one when first node dies lifetime is considered. We can conclude that the selection of nodes with high energy and lower rate may reduce energy efficiency. A more elaborated heuristic that considers both residual energy and node transmission rate is needed.

3.4 REECHD: rotating energy efficient clustering for heterogeneous devices

In this section we describe the REECHD clustering protocol. This produces clusters with the same size. REECHD introduces a novel leader election and rotation algorithms that considers both node residual energy and node rate in order to solve the low-rate-high-energy node depletion problem. RHECEED includes the following phases: (i) CH election and cluster formation; (ii) CH routing tree definition;(iii) network operation phase; (iv) rotation; and (v) re-election and formation.

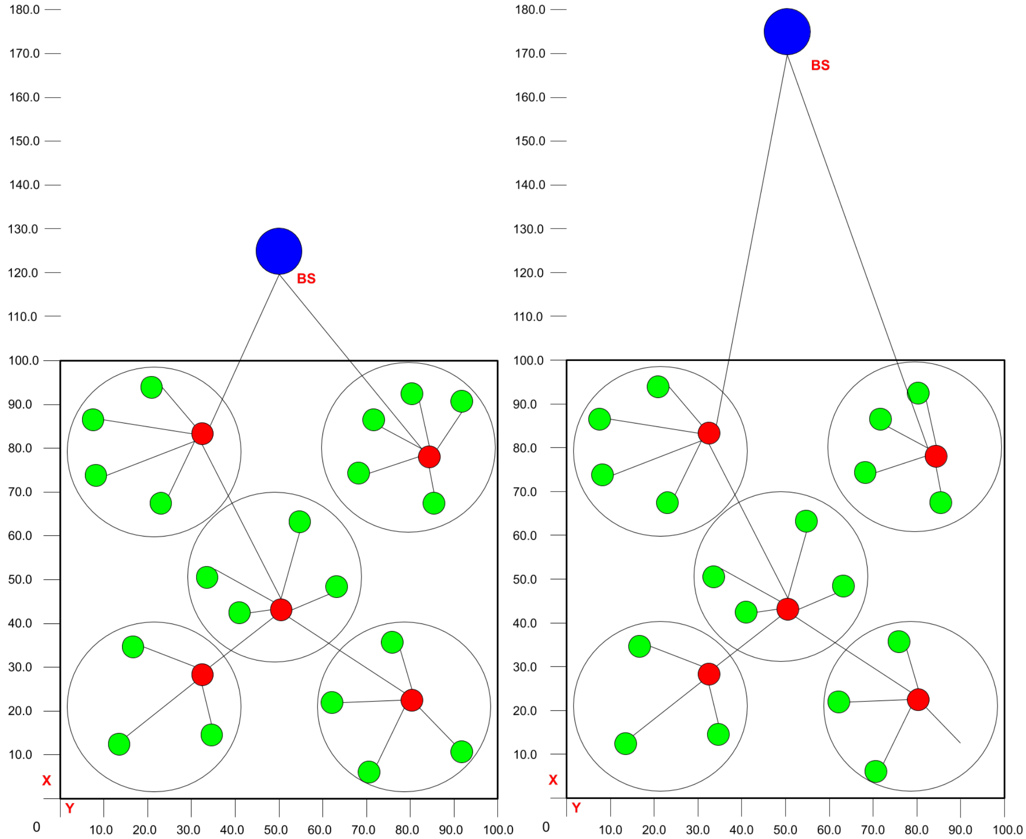

FIGURE 3.1: Network configurations: BS at (125,50) and (175,50)

3.4.1 REECHD leader election probability

The REECHD leader election combines the node residual energy and the node work. This estimates the cost of a node when is elected as CH and the cost that such selection induces on other nodes, e.g., the member nodes of CH. Nodes with low work and high residual energy should have a better chance of becoming CH when compared with nodes high work and low energy.

When the CH election process takes place a node set the probability of becoming CH by using Eq. 3.7 where:

- K is a constant (i.e., 2) s.t. the value of CH_{prob} is always between 0 and 1;
- C_{prob} is a predefined initial probability (e.g., 5%) that is useful to limit the first CH announcements, and does not impact the final clustering;
- P_{min} (e.g., 10^{-4}) is used to avoid that CH_{prob} falls below a minimum value[1];
- E_r is the node residual energy;
- E_{max} is the maximum energy of the node (it is equal to a fully charged battery);
- $T(CH)$ is the node transmission rate;
- T_{max} is the highest transmission rate of the WSN (it corresponds to the rate of the node which has the highest transmission rate in the WSN);
- IW is the node induced work;

$$CH_{prob} = max(\frac{C_{prob}}{K}(\frac{E_r}{E_{max}} + IW^{-1}), P_{min}) \tag{3.7}$$

$$IW = \frac{T_{max}}{T(CH)} \tag{3.8}$$

The use of the induced work has a twofold aim: to consider the energy effort of a node when is playing the CH role, and to take into account the energy cost induced on other nodes. REECHD induced work is the ratio between the node with the highest transmission rate and the node transmission rate (see Eq. 3.8). This work definition assigns to nodes with high transmission rate low induced work. Effectively, the selection of a CH with higher rate generates lower intra-traffic communication (see Eq. 3.1 for details). REECHD probability adds to the node induced work the residual energy, i.e., nodes with high energy and high rate have more probability of becoming CH.

The use of the REECHD probability CH_{prob} solves the low-rate-high-energy node depletion problem that is described in Section 3.3. REECHD FND lifetime measure is 10638 rounds and the energy left for node n_1, n_2 and n_3 are 0, $4.61J$ and $4.61J$. This performance doubles the residual energy based CH election when FND lifetime is considered.

[1] P_{min} is selected to be inversely proportional to E_{max} s.t. the algorithm terminates in $N_{iter} = O(1)$ iterations [24]

It is worth mentioning that the node induced work could be further refined by considering different sources of energy consumption such as the energy the node spends to run the sensor hardware.

3.4.2 REECHD intra-traffic rate limit

The ITRL defines a rate that each CH must use during cluster formation. More precisely, each CH must make sure that its intra-traffic data $R_{intra}(CH)$ (see Eq. 3.1 for a definition of $R_{intra}(CH)$) is less than the ITRL. This is defined by the following equation:

$$R_{intra}(CH) < ITRL$$

We can define a lower and upper bound for the ITLR:

$$\left[0, \sum_{i=1}^{|WSN\_nodes|} T(n_i)\right]$$

where $|WSN\_nodes|$ is the number of WSN nodes. We can have a flat routing (i.e., each node of the WSN is cluster head and has no member nodes) when the ITLR is equal to zero (or the minimum node rate of the WSN). We can have one single cluster that contains all nodes when the ITRL is the sum of all node sending rates.

The ITLR is a quite useful means to control the number of clusters inside the WSN. Low ITLR values can generate more clusters than high ITLR values. More clusters can lead to lower intra-traffic communication at the cost of higher inter-traffic communication. As we are going to show in Section 5.2 the use of the ITLR is also useful when nodes are not uniformly deployed. In this case areas where nodes are denser can get a higher number of clusters. This reduces the intra-cluster communication thus prolonging the WSN lifetime.

3.4.3 REECHD cluster head election and cluster formation

Cluster head election and cluster formation take place at the beginning and anytime a node dies. In this phase each node can become cluster head according to the probability that is defined by Eq. 3.7.

Figure 1 outlines the cluster head election algorithm that is executed by a node denoted by MYSELF. An INITIALISATION procedure is used to set some variables and is executed only once before the CLUSTER_HEAD_ELECTION procedure. The INITIALISATION procedure sets the following variables: (i) ITERATIONS; (ii) MAX_ITERATIONS; (iii) ITLR; and (iv) R_0. The variable ITERATIONS stores the number of times the node MYSELF has executed the CLUSTER_HEAD_ELECTION procedure. This is repeated when MYSELF is not able to join any cluster since all surrounding cluster heads reached the intra-traffic limit. The variable MAX_ITERATIONS defines the maximum amount of times a node repeats the CLUSTER_HEAD_ELECTION procedure before it elects itself as cluster head. The procedure SET_PARAMETER (ITRL) sets the ITRL to a predefined constant value. The competition radius R_0 sets the maximum size of a cluster, i.e., the transmission radius that is used by a CH to advert its presence to surrounding member nodes.

Algorithm 1 REECHD CH election at node MYSELF

```
procedure INITIALISATION( )
    ITERATIONS ← 0
    MAX_ITERATIONS ← k
    SET_PARAMETER(ITRL)
    R_0 ← TRANSMISSION_RADIUS
end procedure
procedure CLUSTER_HEAD_ELECTION( )
    NEIGHBOURS ← {n : d_{n,MYSELF} < R_0 ∧ n is alive}
    ALL_CH ← {n: n sent a tentative or final msg}
    CH_prob ← max((C_prob / 2)(E_r / E_max + T(MYSELF) / T_max), P_min)
    ITERATIONS ← ITERATIONS +1
    MYSELF_CH = FALSE
    repeat
        if ALL_CH ≠ ∅ then
            CH ← BEST_CH(ALL_CH)
            if CH = MYSELF then
                if CH_prob = 1 then
                    SEND(NEIGHBOURS,[FINAL])
                    MYSELF_CH = TRUE
                else
                    SEND(NEIGHBOURS,[TENTATIVE])
                end if
            end if
        else if CH_prob = 1 then
            SEND(NEIGHBOURS,[FINAL])
            MYSELF_CH = TRUE
        else if CH_prob >= random(0,1) then
            SEND(NEIGHBOURS,[TENTATIVE])
            ALL_CH = ALL_CH ⋃ MYSELF
        end if
        PREVIOUS_PROB ← CH_prob
        CH_prob ← min(CH_prob * 2, 1)
    until PREVIOUS_PROB = 1
end procedure
```

The CLUSTER_HEAD_ELECTION procedure initialises the set NEIGHBOURS with all neighbours of MYSELF that are still alive and are away at most R_0. The set ALL_CH contains the following two types of messages: (i) all nodes that sent a TENTATIVE message, i.e., all nodes attempting of becoming CH whose election is not finalised yet; and (ii) all nodes that sent a FINAL message, i.e., all nodes that are CH. The CLUSTER_HEAD_ELECTION procedure sets the probability of the node MYSELF of becoming cluster head (line 10 of the algorithm in Figure 1), increases the ITERATIONS counter and start the REPEAT loop (line $13-33$).

A cluster head CH is elected when the set ALL_CH is not empty (line $14-23$). This is done by using the BEST_CH function (line 15). We have experimented the following BEST_CH functions: (i) the selection of the CH which has the largest cluster set; (ii) the selection of the CH with the highest energy; (iii) the selection of the CH which has the largest value of CH_{prob}. This BEST_CH function has been proved to be the most energy efficient choice thus will be used in this paper. When the elected CH is the node itself (line $16-23$) it can broadcast either an election message or a tentative message. The election message is broadcast when CH_{prob} has reached 1 while the tentative message when CH_{prob} is less than 1. When no nodes proposed as CH and CH_{prob} is equal to one (lines $24-26$) the node itself proposes as cluster head. When no nodes proposed as CH and CH_{prob} is less than one (lines $27-29$) MYSELF decides whether or not to become tentative CH by considering its probability CH_{prob}; at the end of each repeat cycle the probability is doubled (line 32). This ensures that REECHD terminates in $\mathcal{O}(1)$ number of steps.

We emphasise that the BEST_CH function is used to break the tie and select a cluster head when two tentative nodes lies within the same communication range. This behaviour prevents two nodes within the same transmission range from becoming CHs.

Cluster formation and iteration

In Figure 2 we detail the cluster formation algorithm that is executed by the node MYSELF. This executes the $then$ branch of the if control structure (lines $3-7$) when is cluster head while executes the *else* branch (lines $9-21$) when is a member node.

Myself playing the CH role. The node MYSELF waits for all JOIN messages from surrounding member nodes. These nodes are stored inside the set ALL_MEMBER (lines 3) which is used together with the $ITRL$ to call the *pick_member* function. This returns a subset M of the ALL_MEMBER set that contains all nodes selected by MYSELF as cluster members. We recall that the intra-traffic communication generated by the nodes of M must be less than the $ITRL$ (see Section 3.4.2 for details). Various *pick_member* member selection strategies can be adopted. We list some in the following: (i) a random pick can be performed until the $ITRL$ is reached; (ii) the member nodes can be selected starting from the highest rate node until the $ITRL$ is reached; (iii) the member nodes can be selected so that the total rate is less than or equal to the $ITRL$ and is as large as possible; and (iv) the closest member nodes can be selected so that interference and transmission radius are reduced (this improves energy efficiency). This is the strategy that has been used in this paper. The node MYSELF sends an

Algorithm 2 REECHD cluster_formation MYSELF

```
procedure CLUSTER_FORMATION( )
    if MYSELF_CH then
        ALL_MEMBER ← {N: N SENT A JOIN MSG}
        M ← PICK_MEMBER(ALL_MEMBER,ITLR)
        SEND(M,[JOIN_ACK])
        REFUSED_MEMBER ← ALL_MEMBER-M
        SEND(REFUSED_MEMBER,[JOIN_NACK])
    else
        FINAL_CH ← {n: n sent a final msg}
        if FINAL_CH ≠ ∅ then
            while FINAL_CH! = ∅ do
                CH ← PICK_CH(FINAL_CH)
                FINAL_CH = FINAL_CH − CH
                SEND(CH,[JOIN])
                if JOIN_ACK IS RECEIVED then
                    return
                end if
            end while
        else
            SEND(NEIGHBOURS,[FINAL])
        end if
    end if
    if ITERATIONS ≤ MAX_ITERATIONS then
        CLUSTER_HEAD_ELECTION( )
    else
        SEND(NEIGHBOURS,[FINAL])
    end if
end procedure
```

acknowledgement message to all selected members nodes (lines 5) whereas sends an NACK message to all nodes that are not included in the *M* set (line 7).

Myself playing the member role. The set FINAL_CH contains all cluster heads that surround the node MYSELF and have been elected as CH (lines 9). MYSELF terminates the procedure and elects its self as CH when FINAL_CH is empty, i.e., no CHs surround MYSELF (lines $19-20$) otherwise the join algorithm of lines $11-18$ is executed. Here, the node MYSELF tries to join one after another all reachable CHs. More precisely, MYSELF selects all CHs in FINAL_CH starting from the one the maximises the *pick_ch* function down to the one that minimises it. MYSELF will join the first CH that replies with a JOIN_ACK message and terminates the cluster formation procedure. MYSELF iterates the procedure CLUSTER_HEAD_ELECTION when it is not able to join any CH that belongs to the set FINAL_CH. The cluster head election can be repeated a maximum number of times (i.e., $max\_iterations$).

We have experimented various PICK_CH functions. For instance the selection of the CH with the highest rate or the highest residual energy. In this paper we have used a PICK_CH function that selects the closest CH. This allows to reduce the communication costs for both member and its CH.

3.5 Routing data amongst cluster heads

Prior to describe REECHD routing tree defintion, it is worth to introduce different strategies for building the routing tree amongst cluster heads. In particular, two routing algorithms specifically developed to are presented. Figure 3.2 shows a classification of the most used routing approaches for WSNs.

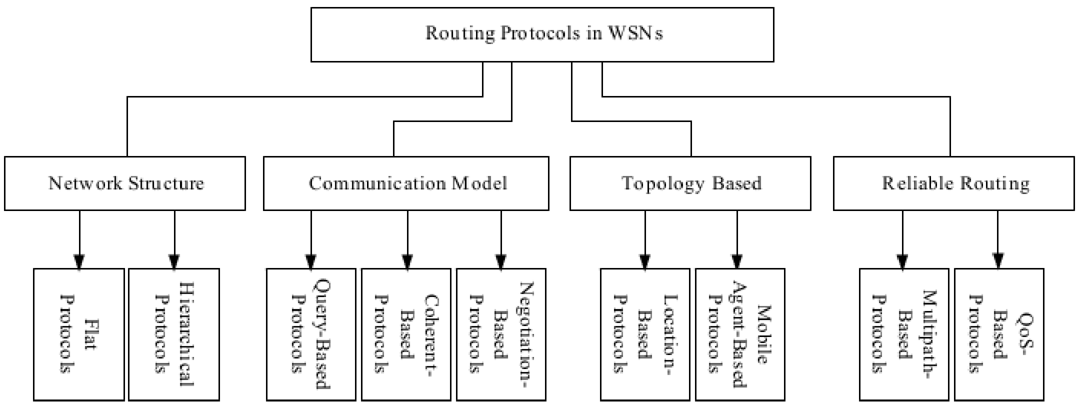

FIGURE 3.2: Classification of routing protocols in WSNs (from[80])

3.5.1 Peer to peer routing algorithm

Algorithm 3 shows the BS_p2p and CH_p2p procedures that are run by the BS and all CHs, respectively. These are used to build the routing tree in a peer to peer way. The BS_p2p is used by the *BS* to discover surrounding CHs. To this ending the routing radius $\Re_{BS}$ (line 2 of Algorithm 3) is initialised and the set BS_{nbr} is built (line 3 of Algorithm 3). This contains all

Algorithm 3 Peer to Peer routing

1: **procedure** BS_P2P
2: $\Re_{BS} \leftarrow$ INITIAL VALUE
3: $BS_{nbr} \leftarrow \{CH : d_{BS,CH} < \Re_{BS}\}$
4: **end procedure**
5: **procedure** CH_P2P
6: $\Re_{CH} \leftarrow$ INITIAL VALUE
7: $step \leftarrow \Re_{CH} \times 0.2$
8: **repeat**
9: $CH_{nbr} \leftarrow \{CH_s : d_{CH_s,CH} < \Re_{CH}\}$
10: $CH_f \leftarrow pick(CH_{nbr})$
11: $\Re_{CH} \leftarrow \Re_{CH} + step$
12: **until** CH_f == null $\wedge$ $\Re_{CH} < \Re_{CH_{max}}$
13: **end procedure**

CHs whose distance from the BS is less than $\Re_{BS}$. These CHs (if any) will always set the *BS* as father.

A CH uses procedure *CH_p2p* in order to find its father (this is denoted by CH_f) in the routing tree. To this ending a CH sets an initial value of the routing radius R_{CH} and a step value. The routing radius R_{CH} is used to discover surrounding *CHs* (if any). The step value is used to increase R_{CH} when CH finds no surrounding cluster heads. As we are going to see in the following, $\Re_{CH}$ needs to be carefully set in order to provide connectivity, generate an energy efficient routing tree and minimise the energy consumption during the routing tree definition. Procedure *CH_p2p* defines the set CH_{nbr} at line 9. This contains all cluster heads whose distance from CH is less than $\Re_{CH}$. A pick function selects a father from the set CH_{nbr}. When no father is found the routing radius is increased by a step quantity and the CH_{nbr} is defined again. Procedure *CH_p2p* terminates when the maximum transmission radius $\Re_{CH_{max}}$ is reached or a father is found. The definition of a suitable pick function is essential in order to ensure connectivity (i.e., all CHs can reach the BS) and to generate an energy efficient routing tree. In this respect we propose a $pick(CH_{nbr})$ function (running at the WSN node *CH*) that ensures the following three conditions:

$$(i)\ CH \neq CH_f$$

$$(ii)\ d_{BS,CH_f} < d_{BS,CH}$$

$$(iii)\ \forall CH_i \in father\_set, d_{BS,CH_f} \leq d_{BS,CH_i}$$

The condition (i) specifies that a cluster head cannot be selected as being its own father. The condition (ii) ensures connectivity since a node CH can always find a father closer to the BS or can directly communicate with the BS. The condition (iii) is used to break the tie when different cluster head fathers verify the condition (ii), i.e., more than one father that is closer to the BS is found. In this case simulations have shown that choosing the father closer to the BS is the most energy efficient choice. This choice implies that a CH always connects to the BS if this is directly reachable.

It is worth mentioning that such pick function ensures connectivity (i.e., conditions (i)-(iii)) when the WSN is dense or a node can always expand its transmission radius to reach a suitable father.

3.5.2 Top-down routing algorithm

Algorithm 4 Top-down routing

```
1: procedure BS_TOPDOWN
2:     ℜ_BS ← INITIAL VALUE
3:     L ← 0
4:     SEND(broadcast, [Join_msg, L+1])
5: end procedure
6: procedure CH_TOPDOWN
7:     ℜ_CH ← INITIAL VALUE
8:     MY_LEVEL ← ∞
9:     FATHER_SET ← ∅
10:    repeat
11:        RECEIVE(CH_s, [Join_msg, L])
12:        if L < my_level then
13:            MY_LEVEL ← L
14:            FATHER_SET ← {CH_s}
15:            SEND(broadcast, [Join_msg, L+1])
16:        end if
17:        if L==my_level then
18:            FATHER_SET ← FATHER_SET ⋃ {CH_s}
19:        end if
20:    until time_out_not_expired
21:    if FATHER_SET!=∅ then
22:        CH_f ← pick(father_set)
23:    else
24:        ℜ_CH ← MAX_RADIUS
25:        CH_nbr ← {father : d_{CH,father} < ℜ_CH}
26:        if CH_nbr! = ∅ then
27:            CH_f ← PICK(CH_nbr)
28:        end if
29:    end if
30: end procedure
```

Algorithm 4 shows a top-down routing strategy. The BS runs the *BS_topdown* procedure that sets the routing radius $\Re_{BS}$ to an initial value and the layer L equal to zero. The BS broadcasts a [*Join_msg*,L+1] message which contains the layer L equal to one (line 4); all nodes reachable from the BS are considered layer one nodes.

A cluster head runs the *CH_topdown* function. This sets the routing radius $\Re_{CH}$, the $father\_set$ to empty and the layer *my_level* to infinity. The $father\_set$ contains all nodes the CH can select as father while L is the length of the shortest path to reach the BS. A CH processes a $[Join\_msg, L]$ message sent by a cluster head CH_s according to the following strategy:

- When the CH receives a level L from CH_s that is smaller than the CH level my_level, the $father\_set$ is updated with CH_s, my_level is updated to L and the message $(Join\_msg, L_{i+1})$ is forwarded. A shorter path to the BS has been found.
- When L is equal to the CH level my_level. The cluster head CH_s is added to the $father\_set$.

The father can be picked with different strategies (these are implemented by the pick function of Algorithm 4, line 22). A CH performs the CH_p2p procedure of Algorithm 3 when it does not receive any $(Join\_msg, L)$ message (see Algorithm 4, line 24). This procedure is used by uncovered CHs to expand their routing radius and find a father. More precisely, the top-down routing has an hybrid approach: it builds the routing tree starting from the BS down to the CHs but uses a peer-to-peer approach for uncovered nodes.

When nodes receive and forward the *Join_msg* they are considered covered while uncovered nodes run the p2p routing. This requires the implementation of a neighbour discovery protocol which also needs acknowledge messages. As it will be shown by the experiments, the p2p approach would then consume more energy.

3.5.3 Grid based routing

Quite often the WSN area is organised in virtual layers [7, 17] or in a virtual grid [9, 15, 16]. In these cases each cluster has nodes that belong to the same layer or grid. Such approaches are used in order to balance the inter-traffic communication and ease the routing. Figure 5.12 shows a WSN clustering where virtual layers are defined as opposed to Figure 5.11 where no virtual layers are defined.

3.6 REECHD CH Routing tree definition

REECHD uses a top-down routing strategy for the CH Routing tree definition. The root of the tree is the base station and each node is a CH. This tree is used by the CHs to deliver data to the base station during the network operation phase. Figure 4 describes the algorithm that is used in order to build the CH Routing tree. The BS executes the BS_TOPDOWN procedure. This sets the BS routing radius $\mathfrak{R}_{BS}$ to an initial value and the layer L equal to zero. The BS broadcasts a [*Join_msg*, L+1] message that contains the layer L equal to one (line 4). All nodes that are reachable directly from the BS are considered layer one nodes.

The CH_TOPDOWN function is run by any cluster head of the WSN. The procedure sets the routing radius $\mathfrak{R}_{CH}$ to an initial value, the layer MY_LEVEL to infinity and the FATHER_SET to empty . The FATHER_SET contains all nodes the CH can select as father in the routing tree and L is the length of the shortest path to reach the BS. A CH processes a $[Join\_msg, L]$ message sent by a cluster head CH_s by using the following strategy:

- When the CH receives a level L from CH_s that is smaller than the CH level my_level, the $father\_set$ is updated with CH_s, my_level is updated to L and the message $(Join\_msg, L_{i+1})$ is forwarded. A shorter path to the BS has been found.

- When L is equal to the CH level my_level. The cluster head CH_s is added to the $father\_set$.

The father can be picked with different strategies that are implemented by the pick function (line 22). When nodes receive and forward the *Join_msg* they are considered covered. Any uncovered CH (line $24-27$) will try to reach a father that is connected to the tree. To this ending the CH sets its routing radius $\Re_{CH}$ to its maximum transmission range and discover all reachable CHs. These are added in the set CH_{nbr} that is used by the pick function to select a father of CH.

The following picks have been implemented: (i) the selection of the closest father; (ii) the selection of the father that is closer to the BS; and (iii) the selection of the father with the highest residual energy. The last strategy has been chosen since it has proved to give the best lifetime performance.

Connectivity is obtained when all CHs can reach the base station. In Top-down routing strategy connectivity depends on the WSN CH density, the maximum node transmission range and the initial values that are set for $\Re_{BS}$ and $\Re_{CH}$. In Section 5.2 the best values for $\Re_{BS}$ and $\Re_{CH}$ that ensures connectivity and energy efficiency of the generated CH routing tree are discussed.

3.6.1 Cluster rotation

Algorithm 5 REECHD rotation algorithm

```
1:  procedure REECHD_ROTATION
2:      max ← 0
3:      next_ch ← 0
4:      for each node n inside cluster CR_CH do
5:          w ← weight(n, CR_CH, WSN)
6:          if max < w then
7:              next_CH ← n
8:              max ← w
9:          end if
10:     end for
11:     SEND(CR_CH, [rotate_CH, next_CH])
12: end procedure
```

After a round (i.e., five TDMAs in the simulation settings) rotation takes place. The current CH designates the next CH by running the Algorithm of Figure 5. This evaluates a *weight* function on each node n that belongs to the cluster CR_{CH} (this includes the current CH plus its members). The nodes that maximises the function weight is elected as the next CH (line $5-9$). This is notified to all nodes by using a $rotate\_CH$ message (line 11). In its general form a *weight* function can take as input node, cluster, and WSN information. For instance node information can be its residual energy and its transmission rate; cluster information can be the total residual energy of the cluster; WSN information can be inter-traffic received by the node.

REECHD uses the election probability of equation 3.7 as a weight function. It is assumed that during the last TDMA data packet received by the CH also contains energy information of its member nodes. This is needed in order to calculate CH_{prob}. Unlike many clustering protocols such as [29, 20, 21] where rotation is based on residual energy, REECHD adds rate information.

The following weight function has also being tried:

$$\boxed{\begin{aligned} CH_{prob} &= max(E_r \times IW^{-1}, P_{min}) \quad (3.9) \\ IW &= E_{R_x}(R_{inter}(CH)) \quad (3.10) \end{aligned}}$$

where the work is the energy the node spends for receiving the intra-traffic communication $R_{inter}(CH)$ (see Eq. 3.1 for details) from its member nodes. Effectively, the ratio between the node residual energy and the intra-traffic communication energy tries to estimate the number of times the node can play the cluster head role. The performance obtained by using this weight function has been proved to be very poor when compared to the REECHD heuristic. In fact, the intra-traffic energy consumption fails to consider the energy spent for forwarding the intra-traffic plus the energy for receiving and forwarding the inter-traffic. These energy depletion sources are considered by the energy $E_{inter}(CH, CH_f)$ (see Eq. 3.5 for details). Unfortunately, the energy $E_{inter}(CH, CH_f)$ cannot be evaluated since at the time of rotation no CH routing tree is yet defined thus the receive inter-traffic by CH is unknown. Although there are few approaches that performs rotation by using $E_{inter}(CH, CH_f)$ these are limited to static clustering [18, 19] where the network is clustered once at the beginning according to some virtual grids [9, 15, 16], virtual layers [7, 17, 18], and concentric circles [18, 19]. In these case the estimates of $E_{inter}(CH, CH_f)$ can be given in average conditions (e.g., average distance between a member and its CH) and simplified settings (e.g., virtual grids). This may lead to lifetime performance degradation when the average case is not representative.

Chapter 4

Combining CH rotation and CH routing tree formation: CER-CH

As in REECHD, most of the clustering protocols are composed of the following phases: cluster head election (CH_Elect), cluster formation ($Clustering$), routing tree definition ($RoutingTree$), network operational phase ($Networking$), and rotation amongst CHs ($Rotation$), if any. These activities can be performed cyclically until the WSN depletes its energy. In particular, the routing tree definition always follows the cluster head election. Figure 4.1 outlines the novelty of CER-CH and its basic building blocks. An initial cluster head election and cluster formation must be performed. As already pointed out, any algorithm for CH election and cluster formation can be used as long as the algorithm produces one CH per cluster. After cluster formation, a routing tree definition phase is performed such as the peer to peer routing of Algorithm 3. The clustering and the routing tree are taken as input to the CER-CH rotation and routing definition components of Figure 4.1. This performs an initial network operational run (e.g., five TDMAs) and then combine the routing tree definition with the CH rotation phase (i.e., CH election). This combination allows the CH election and the routing tree definition to influence each other in order to elect CHs that produce an energy efficient routing tree. Rotation combined with routing tree definition plus network operational phases are performed in alternation until the first node dies. Although the CH is changed at each rotation and the routing three is updated amongst CHs, each CH will route information through the same cluster. In other words, considering the routing tree from the cluster prospective, it never changes. Note that, the initial clustering greatly affects the energy performance of the studied approach but, as discussed in the performance evaluation section, in the average case there is a gain of 20% with respect to the state of art clustering.

When the CER-CH rotation and routing tree definition takes place the BS starts the procedure $BS\_CER-CH\_rotation$ of Algorithm 6. This multicasts a rotate message $[rotate, CH_f]$ to all nodes in the CDN_{BS} set. This contains all CH children of BS in the routing tree.

A cluster head runs the procedure $CH\_CER-CH\_rotation$ when it receives the $[rotate, CH_f]$ message from the old father CH_{old} (line 8 of Algorithm 6). This message contains the new father CH_f that is elected by the previous father CH_{old}. The current cluster head CH designates as the new CH the node n that maximises a function $weight$. This can take as an input a node n and various inputs such as the features of the father CH_f and the cluster node set CR_{CH}. In the next section the implementation of a weight function is dscribed. The new cluster head

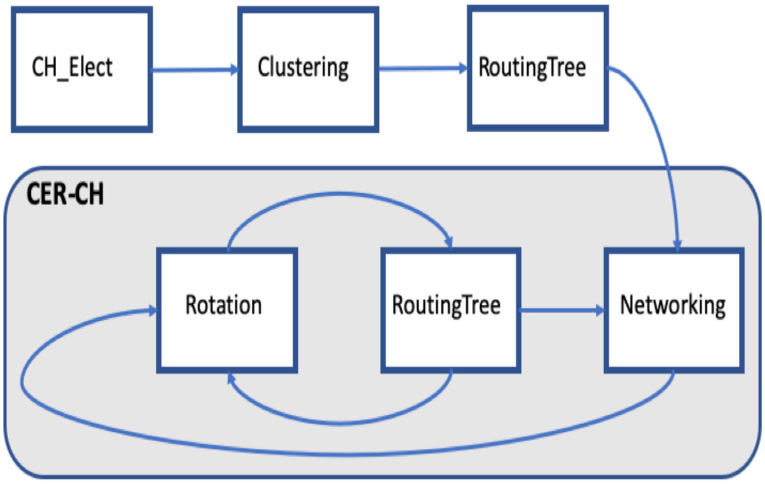

FIGURE 4.1: CER-CH main blocks.

is saved in the *next_CH* variable and the old CH sends a message to every cluster member to indicate *next_CH* as the new CH. Finally, the *next_CH* is communicated to all children of *CH* in the routing tree.

We emphasise that the *weight* function is used by a CH to build the routing tree and elect a CH at the same time. In contrast, a CH uses the *pick* function, which supports the implementation of the traditional approaches (see Algorithm 3 and Algorithm 4 for details), for selecting an already existing cluster head as father. Effectively, *pick* only builds the routing tree.

Algorithm 6 CER-CH routing

procedure BS_CER-CH_ROTATION
    $CH_f \leftarrow BS$
    SEND($CDN_{BS}, [rotate, CH_f]$)
end procedure
procedure CH_CER-CH_ROTATION
    $max \leftarrow 0$
    $next\_ch \leftarrow 0$
    RECEIVE($CH_{old}, [rotate, CH_f]$)
    for each node n inside cluster CR_{CH} **do**
        if $max < weight(n, CR_{CH}, CH_f)$ **then**
            $next\_CH \leftarrow n$
            $max \leftarrow weight(n, CR_{CH}, CH_f)$
        end if
    end for
    $CH_f \leftarrow next\_CH$
    SEND($CR_{CH}, members, CH_f$)
    SEND($CDN_{CH}, [rotate, CH_f]$)
end procedure

4.1 CER-CH Weight function

This section describes the CER-CH weight function that is used in order to implement the CH role rotation. This function requires some definitions and notation that are going to be introduced in the following.

We use $R_{intra}(CH)$3.1 to denote the total amount of bits received by the cluster head CH from its member nodes in a TDMA (i.e., the intra-traffic data).

We use $R_{inter}(CH)$ to denote the total amount of bits received by CH from its children CH nodes in a TDMA (i.e., the inter-traffic received data). $R_{inter}(CH)$ can be calculated as follows:

$$R_{inter}(CH) = \sum_{i=1}^{|subtree_{CH}|} (T_{intra}(CH_i)) \tag{4.1}$$

where $T_{intra}(CH_i)$ is the intra-traffic that CH_i forwards to its father (see Eq. 3.2 for details), a $subtree_{CH}$ contains all cluster heads that are in the sub-tree of the routing tree rooted at CH. Effectively, $subtree_{CH}$ contains all cluster heads that forward inter-traffic to CH. Inter-traffic is not aggregated.

We can now calculate the energy $E_{intra}(CH)$ a CH spends to receive messages from its member nodes. $E_{intra}(CH)$ can be defined by the following equation:

$$E_{intra}(CH) = E_{Rx}(R_{intra}(CH)) \tag{4.2}$$

where $R_{intra}(CH)$ is the intra-traffic that has been defined in Eq. 3.1. The energy CH spends to forward data to its father (i.e., the inter-traffic transmission rate) is denoted with $E_{inter}(CH, CH_f)$3.5.

The energy a node spends playing the CH role in a TDMA can be written as:

$$E_{TDMA}(CH, CH_f) = E_{inter}(CH, CH_f) + E_{intra}(CH) \tag{4.3}$$

The elected cluster head should reduce the E_{TDMA} energy consumption.

A good estimate for the intra communication energy consumption E_{intra} can be provided while it is extremely difficult to estimate the inter-traffic energy communication E_{inter}. This estimate requires to know the amount of traffic forwarded by the children nodes. This can be unknown since routing tree definition can occur after the cluster head election. The studied approach overcomes this problem by combining the routing tree definition and the CH election together.

$$CER-CH(CH, CH_f) = \frac{E_{r_{CH}}}{E_{TDMA}(CH, CH_f)} \tag{4.4}$$

Eq. 4.4 describes the weight function that is used by CER-CH. This estimates the number of times a node can play the cluster head role. This is obtained by dividing the residual energy of the candidate node CH by the energy that CH spends in a TDMA.

4.1.1 Weight function estimate

This section shows how CER-CH routing and rotation ease the estimate of the weight function CCE. This requires to estimate $E_{intra}(CH)$ and $E_{inter}(CH, CH_f)$ for the CH candidate node.

The intra-traffic energy $E_{intra}(CH)$ can be easily estimated by using Eq. 4.2. This only requires to know the transmission rate of the clustering nodes.

The inter-traffic communication $E_{inter}(CH, CH_f)$ is composed of the following addition:

1. $E_{Tx}(T_{intra}(CH), d_{CH,CH_f})$;
2. $E_{Rx}(R_{inter}(CH))$;
3. $E_{Tx}(R_{inter}(CH), d_{CH,CH_f})$

The energy (1) that is spent for forwarding the intra-traffic data T_{intra} to the father CH_f can be easily estimated by using Eq. 3.2. This requires to know the father of CH. Such rotation approach ensures the knowledge of the father since the cluster head election proceeds top-down. The energies (2) and (3) are difficult to estimate since the children nodes of CH have not been elected thus the received inter-traffic $R_{inter}(CH)$ is unknown. In the novel rotation Algorithm 6 is it possible to prove the received inter-traffic $R_{inter}(CH)$ is constant throughout the rotation phase. More precisely, rotation can change the CHs but clusters do not change. At each rotation the new CH always forwards data via the same cluster. Thus $R_{inter}(CH)$ (i.e., $T_{intra}(CH)$) is constant, no matter the CH which is designated during the rotation phase.

Chapter 5

A simulation comparison of REECHD and CER-CH

All the simulation aimed to collect results about running clustering protocols on a WSN are performed with a uni-purpose simulation tool (see section1.5.3), using the energy model presented in section3.2.1. Each simulation is the results of the average of various runs. Simulations are organised in the following two main parts: (i) simulation of well known protocols; (ii) simulation of the novel approach. Simulations in (i) are used to get performances of the most energy efficient competitors. Simulations in (ii) are used to compare the energy efficiency of the studied approach with the most efficient competitors.

5.1 REECHD simulation

5.1.1 Node features

We define two types of WSN nodes that are homogeneous and heterogeneous. Homogeneous nodes have identical initial energy and send messages of the same length. Heterogeneous nodes have an initial energy that falls within a fixed interval and send messages with sizes laying within certain bounds. Heterogeneity level is defined as the ratio between the number of the heterogeneous nodes and all WSN nodes. For instance an heterogeneous level of 20% means that 20% of the WSN nodes are heterogeneous.

5.1.2 WSN features

The network model has the following properties:

- nodes are not mobile and are distributed in a two dimensional area;
- energy is heterogeneous since nodes can have different initial energy;
- heterogeneous nodes have different data transmission rates within a defined maximum and minimum;
- homogeneous nodes have equal initial energy and data transmission rates;
- nodes have the same processing and aggregation capabilities;

- nodes have unique IDs and can transmit at various power levels which depend to the distance of the receiver.

The BS is not mobile, has no energy constraints and can be located either inside or outside the WSN area. The BS has more communication and processing capabilities with respect to normal sensor nodes. Each CH can aggregate the intra-traffic data in order to reduce the amount of bits that are forwarded to the BS. Inter-traffic is not aggregated that is a CH forwards (towards the BS) messages received from other CHs with no aggregation.

MIT radio model is adopted (see 1.2.2).

5.2 Clustering protocols simulation results and discussion

The WSN is composed of 100 nodes that are distributed in a two dimensional area of size 100x100 meters. There are the following two types of nodes:

- *High Energy and Low Rate (HELR)* that have an initial energy of 10J and a rate of 1024 bits per TDMA;
- *Low Energy and High Rate (LEHR)* that have an initial energy of 3J and a rate of 8192 bits per TDMA.

The WSN area is divided in two rectangles of equal size (see Figure 5.1 for details). The following two types of node distribution are considered:

- *70-HELR distribution* where 70 HELR nodes and 30 LEHR ones are uniformly distributed in the rectangle close and far from the BS, respectively;
- *50-HELR distribution* where 50 HELR nodes and 50 LEHR ones are uniformly distributed in the rectangle close and far from the BS, respectively;

These distributions ensure the formation of a high-rate WSN subarea that is the rectangle far from the BS. Simulation results where the high-rate WSN rectangle is close to the BS are not shown since they are in favour to the former approach. Finally, the case where heterogeneous nodes are uniformly distributed over the WSN area (i.e., no high-rate WSN area exists) has been already presented in [14].

The BS has no energy constraints and can be located located at different positions that are 175 and 125 meters away from the WSN area (see Figure 3.1 for details). The BS has more communication and processing capabilities when compared to normal sensor nodes. Each CH can aggregate the intra-traffic data in order to reduce the amount of bits that are forwarded to the BS. Inter-traffic is not aggregated that is a CH forwards (towards the BS) messages received from its CH children with no aggregation.

We have performed the following two set of experiments: (i) state of art protocol simulation; (ii) simulations of REECHD. Simulations in (i) are used to get the most energy efficient competitor of REECHD. Simulations in (ii) are used to compare REECHD with the most energy efficient competitor.

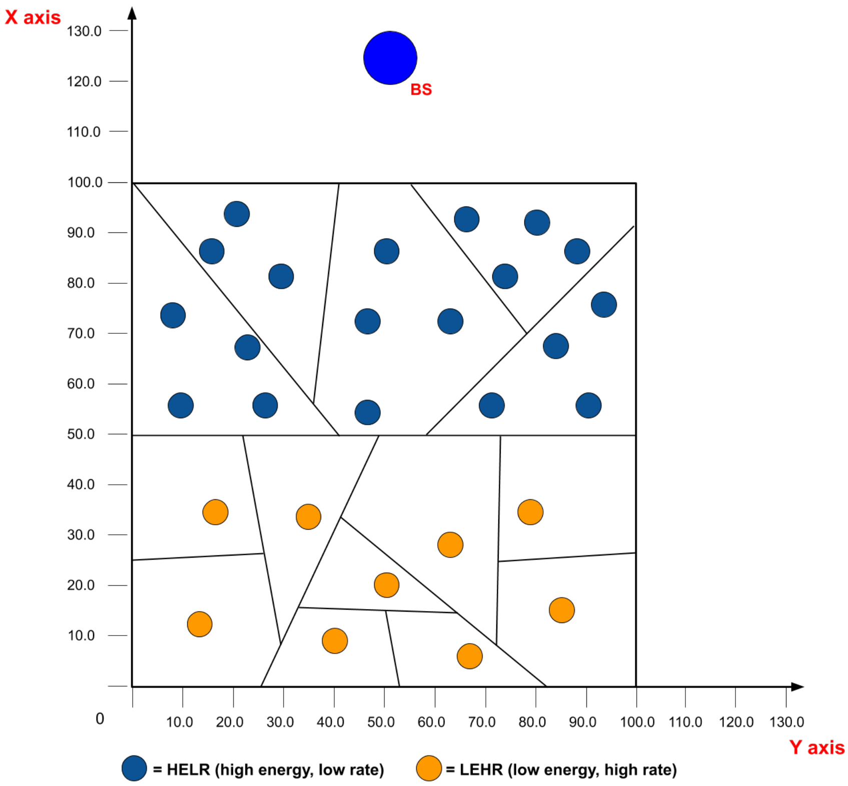

FIGURE 5.1: REECHD clustering

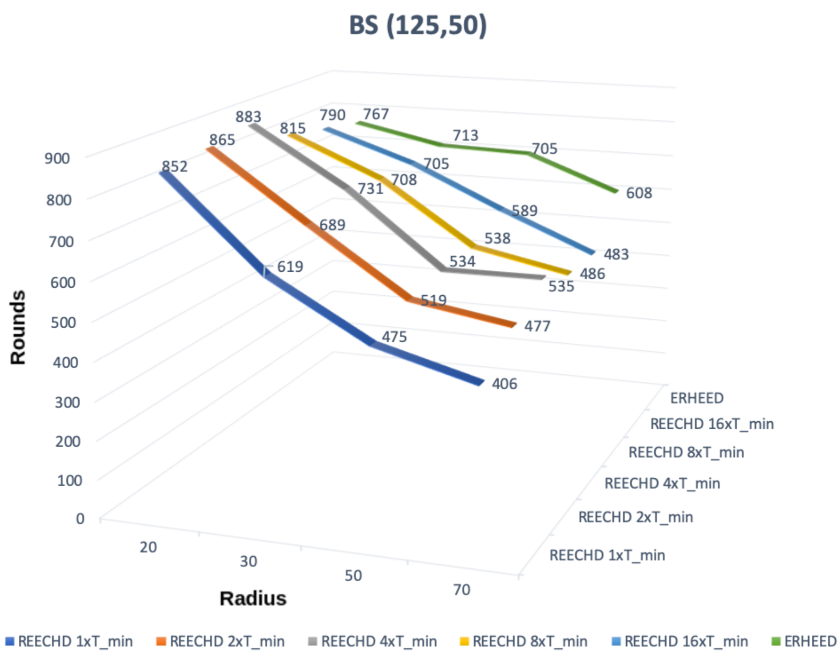

FIGURE 5.2: Network lifetime (FND) for 70-HELR distribution; BS Position= (125,50); $R_{BS} = R_{CH} = 1.7 \times R_0$; $M_{min} = 1024$

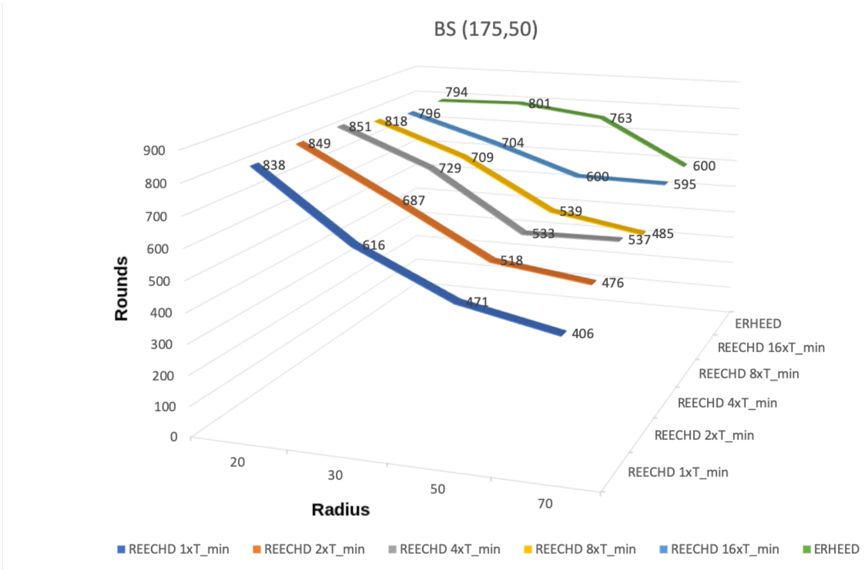

FIGURE 5.3: Network lifetime (FND) for 70-HELR distribution; BS Position= (175,50); $R_{BS} = R_{CH} = 1.7 \times R_0$; $M_{min} = 1024$

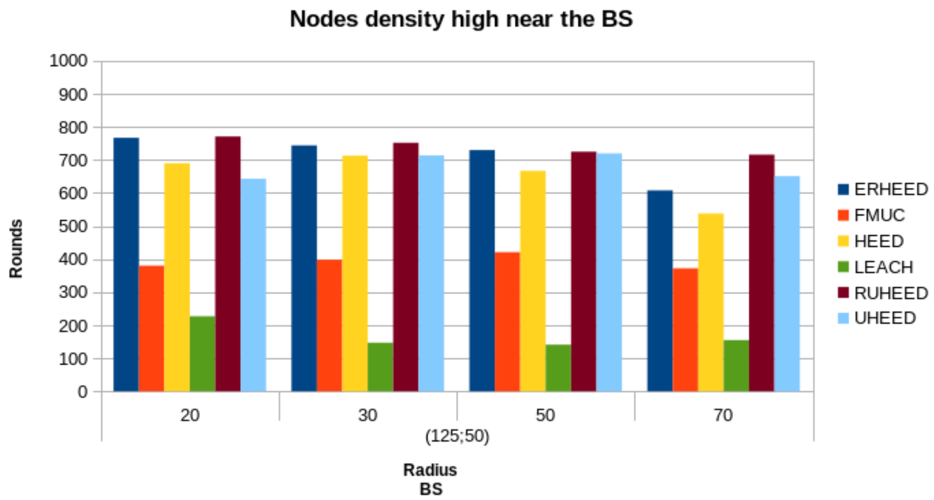

FIGURE 5.4: Network lifetime (FND) for 70-HELR distribution; BS Position= (125,50); $M_{min} = 1024$

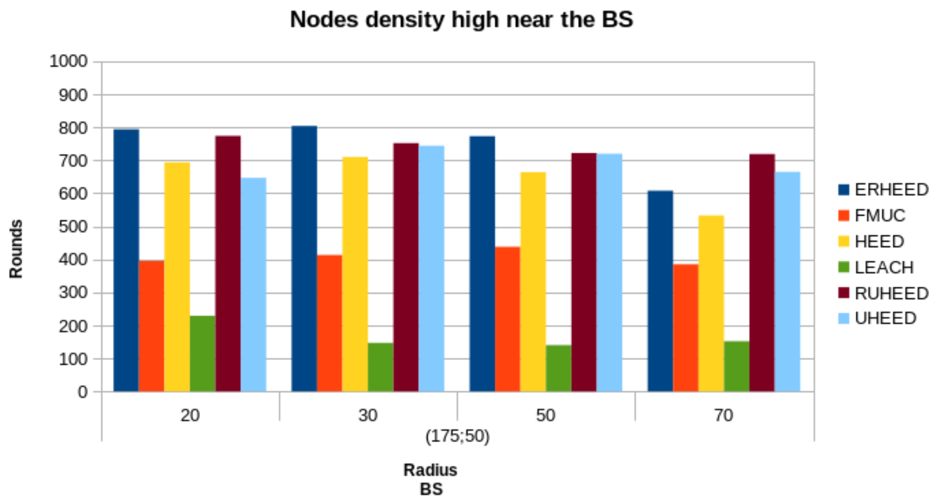

FIGURE 5.5: Network lifetime (FND) for 70-HELR distribution; BS Position= (175,50); $M_{min} = 1024$

5.2.1 State of art protocol simulation

Figures 5.4 and 5.5 show the simulation of the state of art protocols that are ER-HEED[21], HEED [24], FMUC [7], M-LEACH [30] (i.e., multi-hop LEACH), RUHEED [20] and UHEED [29] with control parameter c equals to 0.5. These figures have on the X-axis the competition radius R_0 which varies from 20 meters to 70 meters and on the Y-axis the FND lifetime measure. Figures 5.4 and 5.5 show the 70-HELR distribution for the BS positions (175,50) and (125,50) and a CH aggregation M_{min} of 1024 bits (see Eq. 3.2 for a definition of M_{min}). No matter the position of the base station ER-HEED will outperforms all the protocols. The same simulation results were observed for the 50-HELR distribution and a CH aggregation M_{min} of 512 bits thus ER-HEED will be selected as competitor for the REECHD protocol.

5.2.2 REECHD simulations

Figures 5.2 and 5.3 compare REECHD and ER-HEED for the 70-HELR distribution and a CH aggregation M_{min} of 1024 bits. These figures have on the X-axis the competition radius R_0 which varies from 20 meters to 70 meters, on the Z-axis the FND lifetime measure and on the Y-AXIS the ITLR values. These have been generated by using the equation $2^n \times T_{min}$ where T_{min} is the minimum node rate (1024 in this case) and n varies between zero and four. When n is zero the routing is flat. When n is four a maximum of sixteen HELR nodes can aggregate into one cluster while a maximum of two LEHR nodes can be part of the same cluster. REECHD CH routing tree has been defined by using the top-down Algorithm 4 whereas ER-HEED uses a peer-to-peer routing algorithm. The routing radiuses (i.e., $\Re_{BS}$ and $\Re_{CH}$) is set equal to $1.7 \times R_0$. This ensures connectivity (in all simulations) in 100% of the cases. No matter the position of the base station REECHD will always outperforms ERHEED for all the ITLR values. Both protocols have the best lifetime performance at radius 20. REECHD always gives the best lifetime results when the ITLR is set to $4 \times T_{min}$. In this case REECHD improves ERHEED of 20% in the average case.

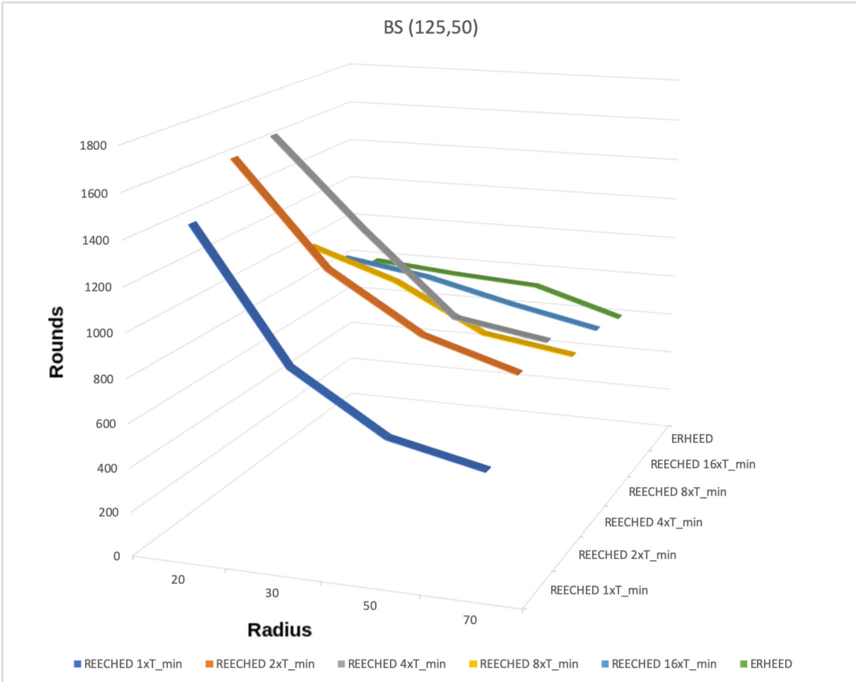

FIGURE 5.6: Network lifetime (FND) for 50-HELR distribution; BS Position= (125,50); $R_{BS} = 70$; $R_{CH} = 70$; $M_{min} = 512$

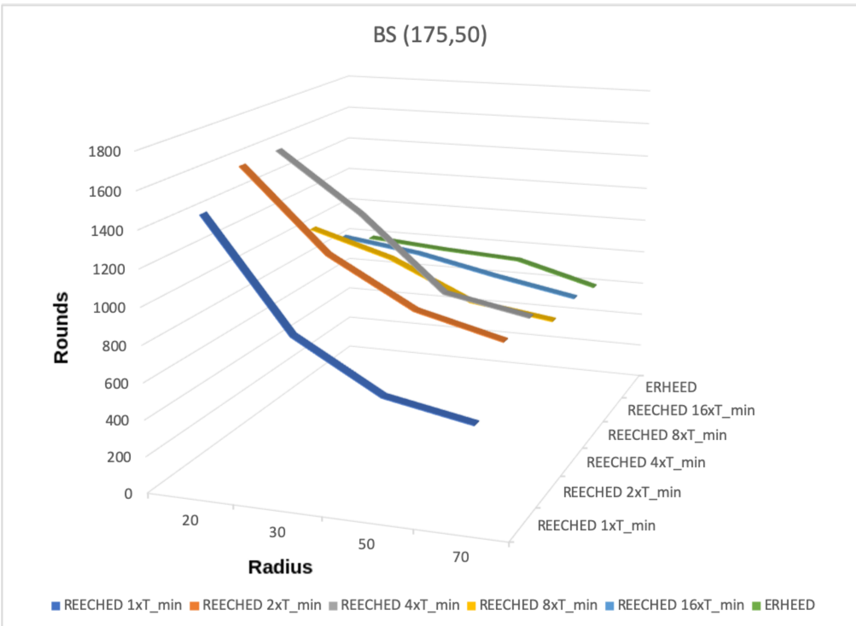

FIGURE 5.7: Network lifetime (FND) for 50-HELR distribution; BS Position= (175,50); $R_{BS} = 70$; $R_{CH} = 70$; $M_{min} = 512$

Figures 5.6 and 5.7 compare REECHD and ER-HEED for the 50-HELR distribution and a CH aggregation M_{min} of 512 bits . These figures have on the X-axis the competition radius R_0 which varies from 20 meters to 70 meters, on the Z-axis the FND lifetime measure and on the Y-AXIS the ITLR values. The routing radius (i.e., $\Re_{BS}$ and $\Re_{CH}$) is set to $70m$. As previously, this ensures connectivity (in all simulations) 100% of the time. No matter the position of the base station REECHD will always outperforms ERHEED for all the ITLR values. Both protocols have the best lifetime performance at radius 20. REECHD always gives the best lifetime results when the ITLR is set to $4 \times T_{min}$. In this case REECHD improves ERHEED of 220% in the average case.

5.2.3 Discussion

REECHD innovative heuristic makes use of both residual energy and transmission rate in order to mitigate the low-rate-high-energy problem. The use of the ITLR allows the generation of a higher amount of clusters for high-rate WSN subareas. For instance, Figure 5.1 shows an example of clustering that is created by REECHD using the simulation settings described in Section 5.2.2 that is the BS is located at (120,50) and the ITLR is equal to $4 \times T_{min}$. In this case the rectangle far away from the BS has a higher amount of clusters since it is the subarea with the highest rate. A higher amount of clusters reduces the intra-cluster communication at the price of a higher inter-cluster communication. Is it possible to see that if the rate that is generated in a WSN region keeps increasing there is a threshold after which the CH will take to mach intra-traffic and will die in few rounds (see Eq 3.3 for details). Increasing the number of clusters of the high-rate WSN subareas will reduce the energy for intra-cluster communication and will allow to benefit of the inter-cluster aggregation. In other words when aggregation produces a very small traffic compared to intra-traffic communication having more clusters can improve energy efficiency. This is confirmed by simulations where an aggregation message size of 512 bits (see Figures 5.6 and 5.7 for details) leads to better lifetime performance with respect to an aggregation message size of 1024 bits (see Figures 5.2 and 5.3 for details).

The base station position do not affect the the FND lifetime performance since the first node that dies is far from the BS (it is in the high-rate WSN subareas). This is profoundly different from uniformly distributed WSNs where nodes close to the BS die first [81] (this is often referred to as the sink energy hole problem).

Finally, it is worth mentioning that the routing radius can affect the first node die lifetime performance. Small values of the routing radius (i.e., Figures 5.2 and 5.3) can generate routing tree with short links where LEHR CHs are connected to other LEHR CHs. Although shorter link can improve energy efficiency LEHR nodes are equipped with little energy and die first. In contrast, large values of the routing radius (i.e., Figures 5.6 and 5.7) can generate routing trees where LEHR CHs connect to HELR ones which have a higher amount of energy.

5.3 Towards CER-CH: novel routing simulation settings and Results

In this section the simulation settings, the routing radius selection and the results of the novel routing strategy are presented.

5.3.1 Simulation settings

We assume nodes are uniformly distributed in a two dimensional area. Heterogeneous nodes have different data transmission rates and different initial energy within a defined range, while homogeneous nodes have equal initial energy and data transmission rates. More precisely, simulation assumes homogeneous nodes have an initial energy of 1 joule and send messages of 3000 bits per TDMA. A heterogeneous node has an initial energy that falls within the interval $[1,4]$ joules and sends messages of a size that falls within the interval $[500,3000]$ bits. This message size is drawn in the first setup phase and then is kept constant throughout the entire WSN simulation. Heterogeneity is set (i.e., the percentage of heterogeneous nodes) to 50% and 100%. Aggregation rate is 100% since lower values does not change the trend of simulations. Nodes have the same processing and aggregation capability, have a unique IDs and can transmit at various power levels which depend on the distance of the receiver.

The BS has no energy constraints and is located outside the WSN area. The BS has more communication and processing capabilities with respect to normal sensor nodes. Each CH can aggregate the intra-traffic data in order to reduce the amount of bits that are forwarded to the BS. Inter-traffic is not aggregated that is a CH forwards (towards the BS) messages received from its CH children with no aggregation.

5.3.2 Selection of the initial routing values $\Re_{BS}$ and $\Re_{CH}$

The initial routing radius value $\Re_{CH}$ (in the following referred to as v_{CH}) should be carefully set in order to ensure connectivity with few control messages and generate an energy efficient routing tree. Small values of v_{CH} can cause the following two problems: (i) high energy consumption for finding surrounding CHs; and (ii) inefficient routing tree generation. More precisely, the cluster head CH can fail to find surrounding cluster heads when v_{CH} is too small. In this case CH keeps sending control messages with an increasing $\Re_{CH}$ until at least a surrounding father is found. This causes a waste of energy and the selection of fathers that are closer to CH. This choice can generate routing paths with lots of hops, i.e., routing tree with lots of short connection links. Although short links require less energy for sending data, a high number of links can increase the inter-traffic communication. Large values of v_{CH} allow the CH to find a surrounding father with the first broadcast but can generate routing paths with few hops and links with long distance. More precisely, large values of v_{CH} can generate routing tree with a small number of long connection links. Although this keeps the inter-traffic communication low, CHs can require a lot of energy in order to send data over long distances.

Figures 5.8 and 5.9 show on the X-axis the routing initial values and on the Y-axis the FND measure. Such results have been obtained by using the following simulation parameters:

- uniformly distributed deployment of 100 nodes over a WSN square area of 200 by 200 meters;
- BS located at position (275,50);
- both heterogeneity and aggregation rate at 100%;
- control parameter c for UHEED and RUHEED equal to 0.5 (see [20] for the definition of c);
- REECHD ITLR parameter equal to 100% (see [14] for the definition of ITLR)

The choice of considering an area of 200 by 200 meters rather than 100 by 100 (which is widely used in literature) allows to better highlight the effect of the initial routing value. More precisely, as the network gets larger, the routing tree gets deeper and the effect of a badly chosen initial routing value becomes more visible. Figure 5.8 shows the FND for the p2p routing protocol where the competition radius is set to $R_0 = 70$. The FND measure is evaluated for the ER-HEED[21], REECHD [14] and RUHEED [20] protocols. REECHD uses the heuristic that is presented in [20] which is summarised by Eq. 3.7. Figure 5.9 shows the FND for the REECHD protocol for the top-down routing protocol where the competition radius is set to $R_0 = 70$ and $R_0 = 50$ (red and blue lines respectively). For each routing initial value the number of rounds it takes for the first node to die is considered. Each value is obtained by averaging 50 simulation runs.

For all experiments and all routing protocols small values of v_{CH} produce a decrease in performance up to 20% when compared to the most energy efficient initial routing value. This same behaviour can be observed when the initial value of v_{CH} is large. It is also possible to observe that the unequal protocol RUHEED is less affected by the initial value v_{CH}. This is a consequence of the RUHEED clustering strategy that has a dense amount of small clusters next to the BS. In this case small values of v_{CH} still allow to find a father with no extra messages.

We have performed an extensive set of experiments for different WSN settings that show $\Re_{CH} = 1.6 \times R_0$ to be the most energy efficient routing initial value. This initial value allows a CH to find a father with one broadcast message (for both p2p and the top-down routing) and to build an energy efficient routing tree. This can be easily proved for the equal variations of the HEED (e.g., ER-HEED and REECHD) which produce clusters of equal radius R_0. HEED prevents two nodes within the same transmission range from becoming CHs. Thus, the distance between two CHs must be greater than R_0. Figure 5.10 shows a network model where the clusters form a grid with cells R_0 by R_0. In this case a routing radius $\Re_{CH}^{min}$ equal to $\sqrt{2} \times R_0$ would allow a CH to discover all surrounding cluster heads. During the rotation phase a communication radius of $2\sqrt{2} \times R_0$ is the longest distance to be covered amongst two communicating CHs.

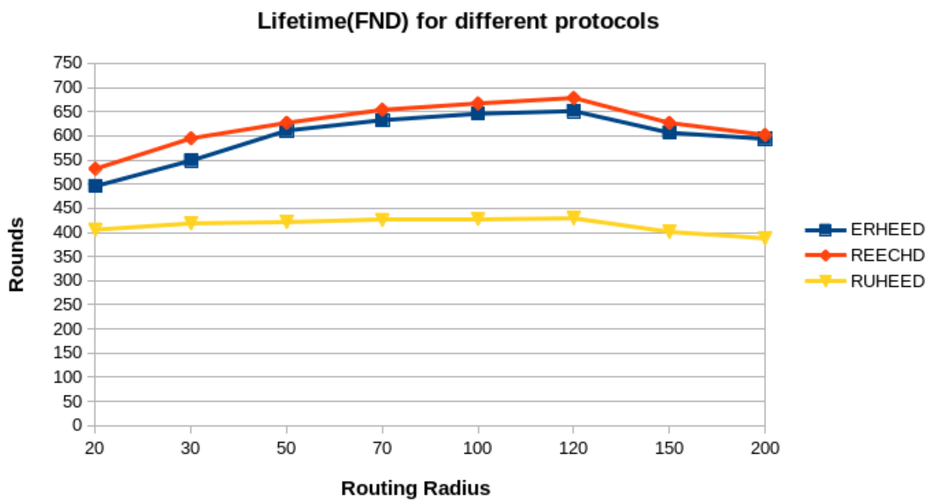

FIGURE 5.8: Network lifetime (FND) for a WSN of 100 nodes, WSN grid 200x200, BS(275,50) Heterogeneity 100%, Aggregation 100%, Routing p2p, $R_0 = 70$.

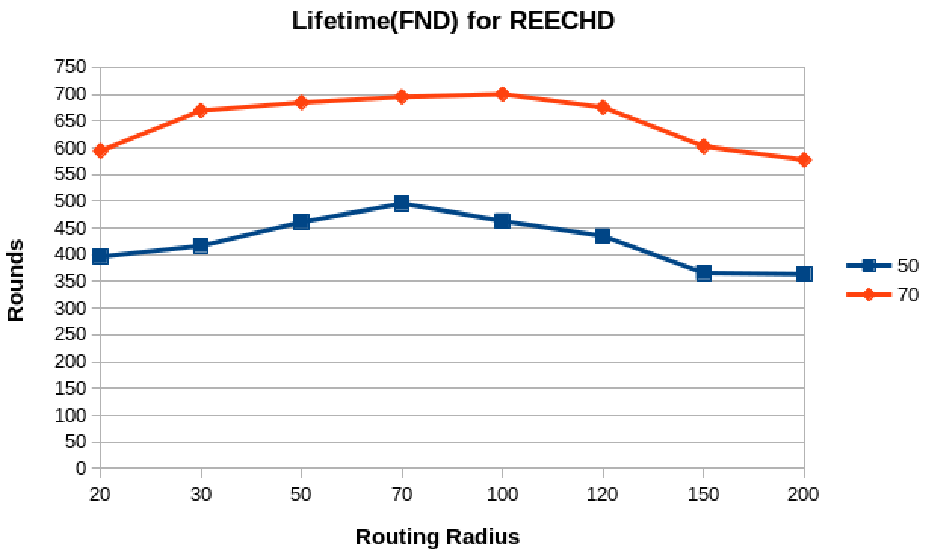

FIGURE 5.9: Network lifetime (FND) for a WSN of 100 nodes, WSN grid 200x200, BS(275,50) Heterogeneity 100%, Aggregation 100%, Top-Down routing, $R_0 = 70$

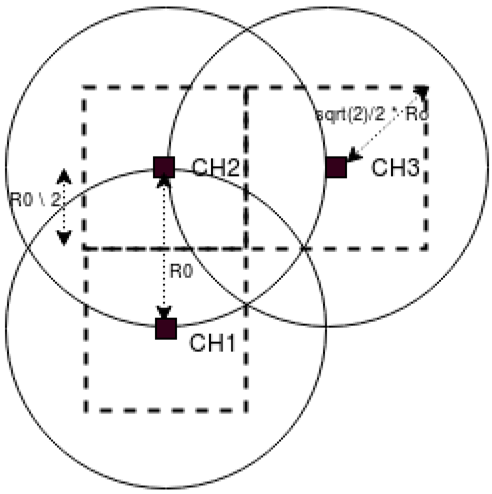

FIGURE 5.10: A network model where the clusters form a grid with cells of dimensionR_0 by R_0

$$\Re_{CH}^2 \times \pi \times \delta_{CH} \geq 2 \tag{5.1}$$

Eq. 5.1 can be also used to approximate the initial value of the routing radius $\Re_{CH}$ where δ_{CH} is the cluster head density. This equation ensures the initial routing radius always intercept a surrounding cluster head. This approach could be not viable since the density function δ_{CH} can be difficult to estimate. In fact, cluster heads are not always uniformly distributed since δ_{CH} could depend on the position. This is the case of unequal clustering protocols such as RUHEED [20] where regions that are closer to the BS have more clusters when compared to regions located farther away. The density δ_{CH} can be also difficult to estimate for protocols that produce equal-sized clusters since different protocols can produce different overlapping of clusters or the same cluster can have more cluster heads.

The radius $\Re_{BS}$ determines the percentage of cluster heads that are directly connected to the BS. Extensive simulations have been performed in order to verify that for the chosen setting the value of $\Re_{BS}$ does not significantly affect the lifetime. Suppose that the *BS* does not reach any surrounding CHs. Then CHs will be still able to reach the BS with respect to both routing Algorithms 3 and 4

5.3.3 Routing protocols simulation

We have performed the following two set of experiments: (i) simulations of the state of art protocols; (ii) simulations of the novel approach. All simulation results have been obtained by using the following parameters:

- a uniformly distributed deployment of 100 nodes over a WSN square area of 100 by 100 meters;
- a BS that is located at position (175,50);
- an heterogeneity and aggregation rate of 100% and 50%;

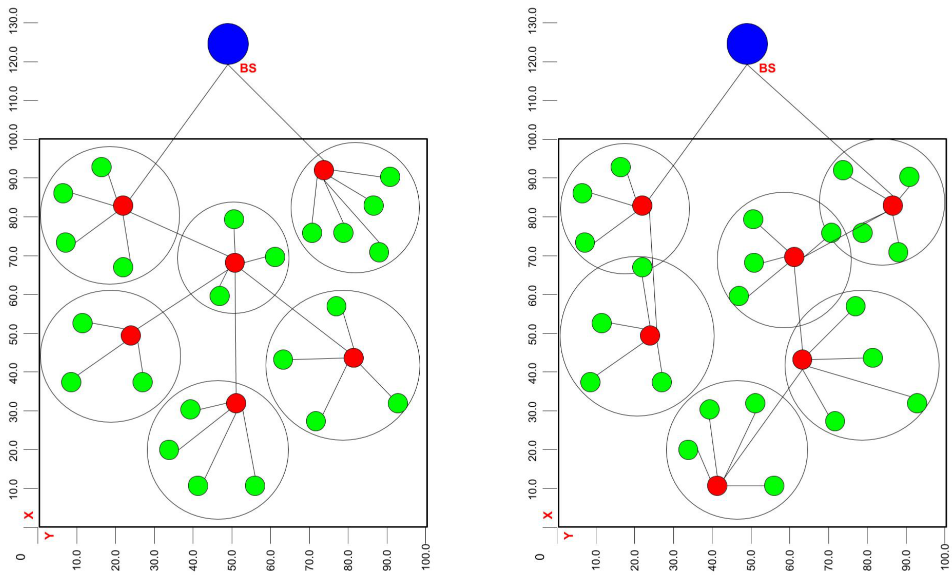

FIGURE 5.11: A WSN clustering without virtual layers. After the rotation phase, some CH can route information via a different cluster

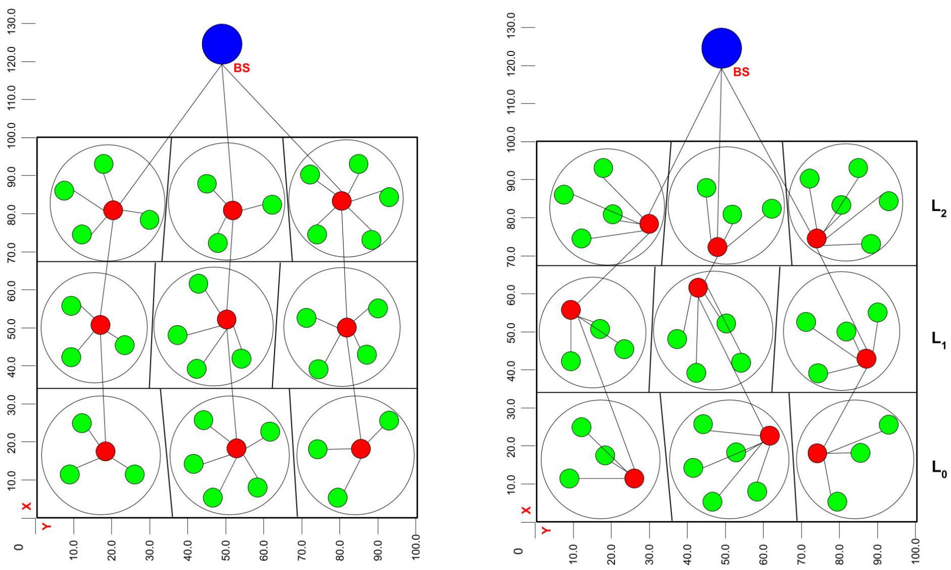

FIGURE 5.12: A WSN clustering with virtual layers. After each rotation phase, the routing tree between the clusters is unchanged

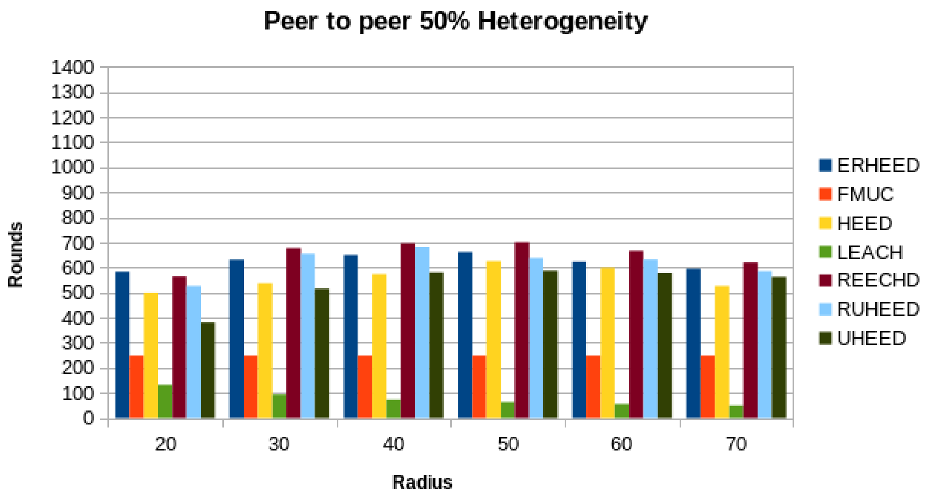

FIGURE 5.13: Simulation of several clustering protocols with Peer2Peer routing, lifetime measure FND. Aggregation is 100%, Heterogeneity level is 50% and BS Position is at (175,50).

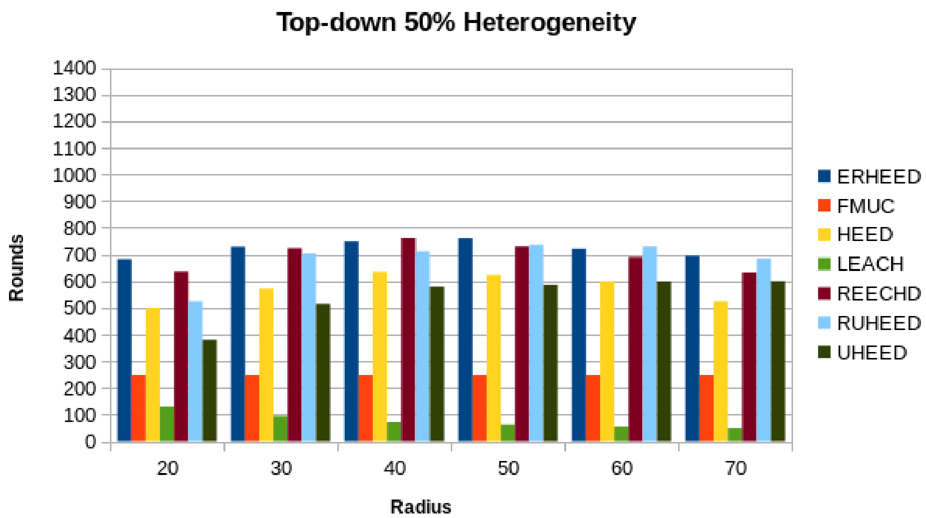

FIGURE 5.14: Simulation of several clustering protocols with Top-Down routing, lifetime measure FND. Aggregation is 100%, Heterogeneity level is 50% and BS Position is at (175,50)

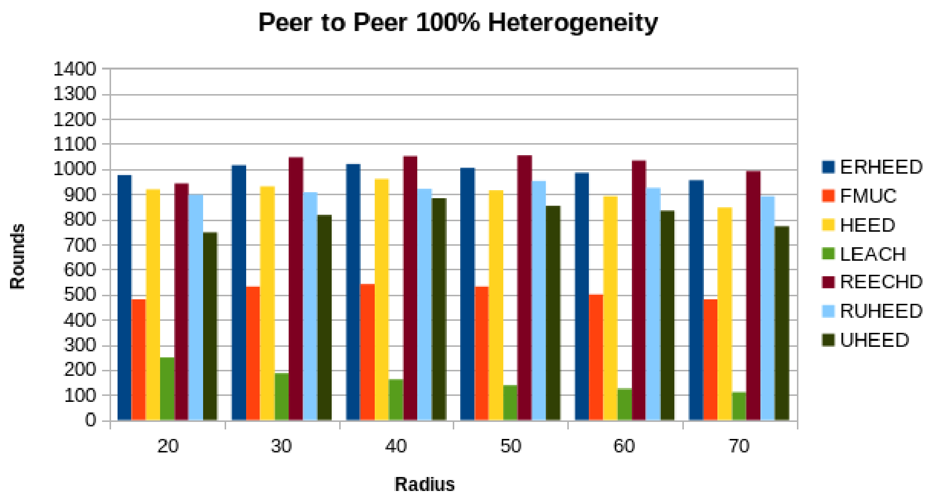

FIGURE 5.15: Simulation of several clustering protocols with Peer2Peer routing, lifetime measure FND. Aggregation is 100%, Heterogeneity level is 100% and BS Position is at (175,50).

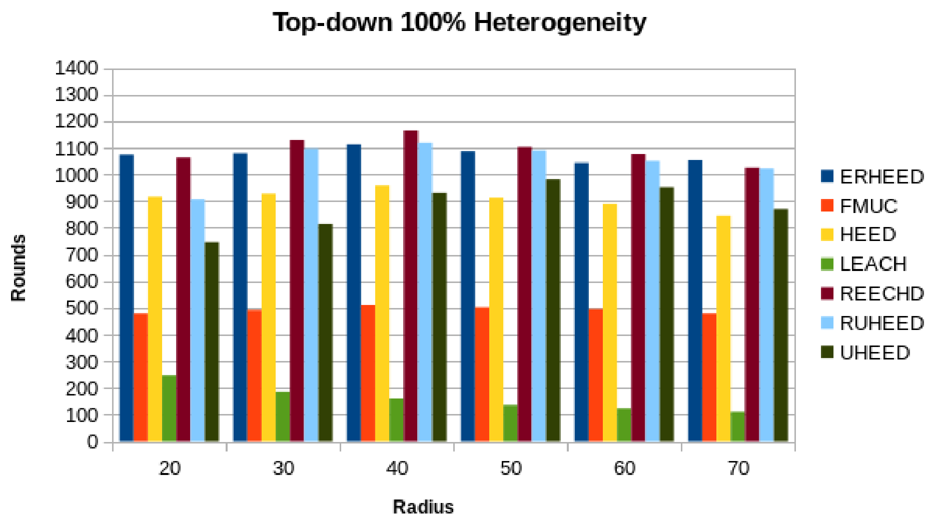

FIGURE 5.16: Simulation of several clustering protocols with top down routing, lifetime measure FND. Aggregation is 100%, Heterogeneity level is 100% and BS Position is at (175,50).

- a control parameter c for UHEED and RUHEED equal to 0.5 (see [20] for the definition of c);
- a REECHD ITLR parameter equal to 100% (see [14] for the definition of ITLR)
- each round is composed of 5 TDMAs

We emphasise that the WSN square area of 100 by 100 meters is widely used in literature. This setting penalises the studied approach since the generated routing tree are usually deep at most three. Higher tree would better show the efficiency of the tree definition.

State of art protocols simulation: the effects of p2p and top-down approaches

Figures 5.13, 5.14, 5.15 and 5.16 show the simulation of the state of art protocols that are ER-HEED[21], HEED [24], REECHD [14], RUHEED [20] and UHEED [29], FMUC [7] and M-LEACH [30] (i.e., multi-hop LEACH). These figures have on the X-axis the competition radius R_0 which varies from 20 meters to 70 meters and on the Y-axis the FND measure. Figures 5.13, and 5.14 show the result for the p2p and top-down routing protocols of Algorithms 3 and 4 when the heterogeneity and aggregation levels are set to 50% and 100%, respectively. REECHD and ERHEED have the best lifetime performance. More precisely, REECHD performance is sightly better than ERHEED and the best lifetime for both protocols is obtained at competition radius $R_0 = 40$. Both p2p and top-down routing protocols have a similar performance. Figures 5.15 and 5.16 show the result for the p2p and top-down routing protocols, respectively, when both the heterogeneity and aggregation levels are set to 100%. Again, REECHD and ERHEED have the best lifetime performance. More precisely, REECHD performance is sightly better than ERHEED and the best lifetime for both protocols is obtained at competition radius $R_0 = 40$. Is it possible to conclude that REECHD is the most energy efficient protocol for the WSN settings considered (this is confirmed from the relevant research literature [14]). Top-down routing performs better than p2p. Hence, in what follows REECHD with its top-down implementation will be the competitor of CER-CH when REECHD is used to set the initial set of clusters. This shows how CER-CH can improve the performance of a protocol which has been proved to be better than many other state of art protocol.

CER-CH: enhancing REECHD with top-down routing combined with rotation

In this section REECHD [14] is compared with REECHD enhanced with CER-CH novel rotation approach. The REECHD simulation makes use of the top-down routing and uses the weight function that is defined by the Eq. 3.7. CER-CH uses REECHD for the initial clustering, the novel top-down routing combined with rotation approach of Algorithm 6 and the Eq. 4.4 for rotation.

Figures 5.17 and 5.18 show simulation results. The X-axis has the competition radius R_0 which varies from 20 meters to 70 meters while the Y-axis the FND measure. CER-CH always outperforms the REECHD protocol. The lifetime performance has an average gain of

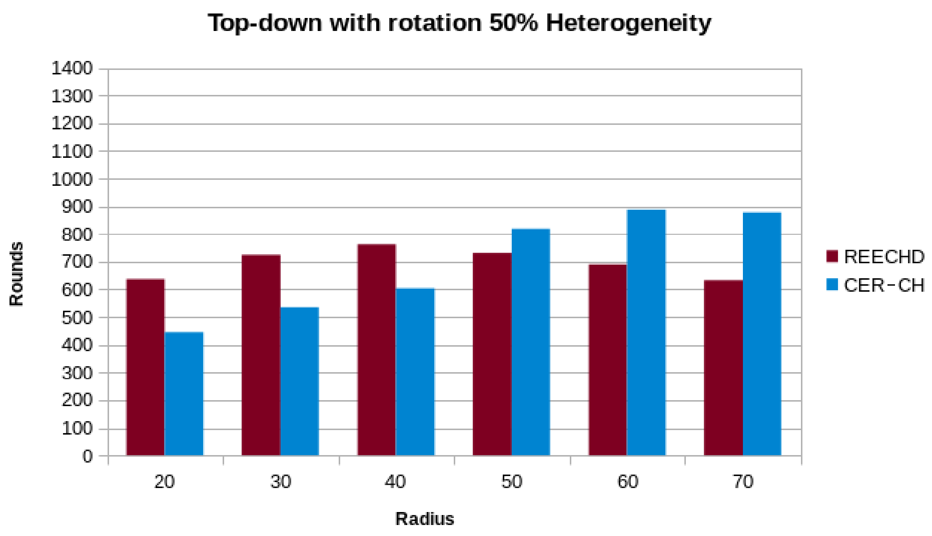

FIGURE 5.17: Simulation of several clustering protocols with Top-Down with rotation routing, lifetime measure FND. Aggregation is 100%, Heterogeneity level is 50% and BS Position is at (175,50).

20% when CER-CH is compared with REECHD. It is worth to notice that as the heterogeneity level increases the lifetime of both REECHD and CER-CH increases. This is consequence of the higher energy that heterogeneous nodes have when compared to homogeneous ones.

5.3.4 Discussion

Although CER-CH starts with the same set of clusters dictated by REECHD, it still has significant performance differences with respect to REECHD. The obtained improvements can be explained by considering their different rotation and routing strategies. REECHD CH rotation: (i) uses node rate and node residual energy; (ii) performs routing by selecting the closest father. Figure 5.11 shows an example of the REECHD routing strategy where (after the rotation) some CHs route information via different clusters. CER-CH rotation: (i) considers node rate, node residual energy and the energy cost for communicating with an already selected father (i.e., local routing path cost); (ii) CHs of a cluster *CR* always select a father from the same cluster (i.e., routing amongst clusters does not change). Figure 5.12 shows an example of CER-CH routing strategy where (after rotation) the routing tree between clusters is unchanged.

CER-CH rotation has some advantages over REECHD one. During CER-CH rotation only multicast messages flow from an old CH to its children while REECHED requires the top-down routing algorithm of Figure 4 to be performed from scratch. CER-CH rotation considers the cost for inter traffic communication while REECHD does not. This allows the selection of a CH that reduces the inter-traffic energy cost.

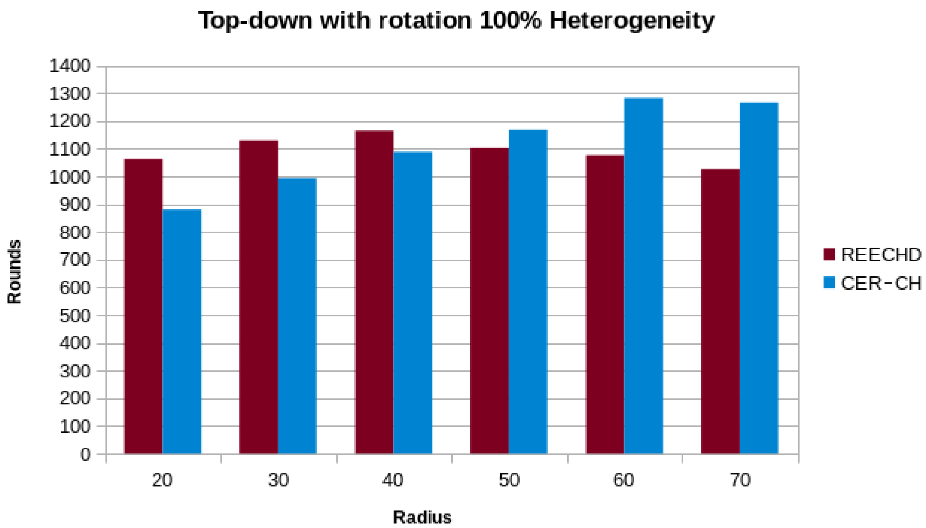

FIGURE 5.18: Simulation of several clustering protocols with Top-Down with rotation routing. Aggregation is 100%, Heterogeneity level is 100% and BS Position is at (175,50).

Chapter 6

Conclusions and future work

In this thesis REECHD, a novel clustering protocol for Heterogeneous WSNs, and CER-CH, a novel routing approach, are proposed.

In chapter 1, the motivations behind this Thesis work as well as the research methodology followed to contribute to the existing body of knowledge with novel work are introduced. The simulation strategy to compare novel clustering and routing protocols with existing work is then discussed.

In chapter 2, the state of art of the most prominent energy efficient clustering for homogeneous and heterogeneous WSNs is revised. Clustering protocols that consider the introduction of harvesting into WSNs and clustering protocols that are based on machine learning techniques are also listed and commented.

In chapter 3 REECHD, a novel clustering protocol for Heterogeneous WSNs, is presented. The low-rate-high-energy problem for non-uniformly distributed WSNs is identified. Non-uniform distribution causes low lifetime performance when well-known clustering protocols that are based on residual energy for CH election are applied. For this reason, REECHD is proposed. REECHD CH election not only considers the nodes residual energy but also their transmission rate. REECHD also introduces the concept of intra-traffic rate limit (ITRL). ITRL defines a limit on the intra-traffic communication that all WSN clusters must comply with. This makes REECHD more suitable to cluster heterogeneous networks in which the communication rates of the devices are heterogeneous and devices are not uniformly distributed in the WSN area.

Chapter 4 introduces CER-CH. CER-CH is a novel rotation approach where the routing tree definition and the CH rotation are combined together. More precisely, a novel rotation heuristic is combined with a novel top-down CH routing tree definition in order to balance the node energy consumption and generate more energy efficient routing trees. CER-CH can be considered as a plug-in defining the rotation and the routing amongst CHs, regardless the strategy chosen for the initial CHs election and cluster formation. Any clustering algorithm that produces a CH per cluster can be used. In this context, REECHD is used for the cluster election and cluster formation.

In chapter 5 REECHD and CER-CH are compared with existing protocols for Homogeneous and Heterogeneous WSNs trough simulation. The simulation tool introduced in chapter1 is used to get results. Simulations validate the initial hypothesis **Q1**, proving that REECHD is more energy efficient when compared with existing clustering protocols and that it is particularly suited for Heterogeneous WSNs. In addition, applying certain ITRL values actually improves the network lifetime for some network configurations, validating the hypothesis **Q2**. Finally, applying CER-CH to REECHD further improve its performance for the considered set of network configurations which validate the last hypothesis **Q3**.

However, while the proposed clustering and routing protocols behave better with respect to existing work on certain network configurations, there are others for which REECHD and CER-CH do not perform well. REECHD novel election probability function considers two nodes properties, residual battery level and data transmission rate. While this is an improvement with respect to existing election functions which only consider nodes battery level, other nodes properties may be taken into account, such as radio power level or core processing power. Moreover, while REECHD ITRL allows to obtain smaller clusters thus reducing communication in the network, it is a global value which does not change for regions when the deployment of the nodes inside the WSN grid is non-uniform. Hence, there is still a need for future research, and a future variation of REECHD may include a local ITRL value which adapts to variable network grid size and density.

Another planned work is to apply CER-CH to other clustering algorithms, e.g. FMUC [7] or DEEC [6], in order to evaluate whether the enhancement induced by CER-CH would be as relevant as the one obtained for REECHD.

Bibliography

[1] C. Vannucchi, M. Diamanti, G. Mazzante, D. R. Cacciagrano, F. Corradini, R. Culmone, N. Gorogiannis, L. Mostarda, and F. Raimondi. "virony: A tool for analysis and verification of ECA rules in intelligent environments". In: *2017 International Conference on Intelligent Environments (IE)*. IEEE. 2017, pp. 92–99.

[2] F. Corradini, R. Culmone, L. Mostarda, L. Tesei, and F. Raimondi. "A constrained ECA language supporting formal verification of wsns". In: *2015 IEEE 29th International Conference on Advanced Information Networking and Applications Workshops*. IEEE. 2015, pp. 187–192.

[3] G. Russello, L. Mostarda, and N. Dulay. "Escape: A component-based policy framework for sense and react applications". In: *International Symposium on Component-Based Software Engineering*. Springer. 2008, pp. 212–229.

[4] L. Mostarda and A. Navarra. "Distributed Intrusion Detection Systems for Enhancing Security in Mobile Wireless Sensor Networks". In: *IJDSN* vol. 4, no. 2, (2008), pp. 83–109.

[5] D. Miorandi, S. Sicari, F. D. Pellegrini, and I. Chlamtac. "Internet of things: Vision, applications and research challenges". In: *Ad Hoc Networks* vol. 10, no. 7, (2012), pp. 1497 –1516. ISSN: 1570-8705. URL: `http://www.sciencedirect.com/science/article/pii/S1570870512000674`.

[6] L. Qing, Q. Zhu, and M. Wang. "Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks". In: *Computer Communications* vol. 29, no. 12, (2006), pp. 2230 –2237. ISSN: 0140-3664. URL: `http://www.sciencedirect.com/science/article/pii/S0140366406000727`.

[7] T. Liu, J. Peng, J. Yang, G. Chen, and W. Xu. "Avoidance of energy hole problem based on feedback mechanism for heterogeneous sensor networks". In: *International Journal of Distributed Sensor Networks* vol. 13, no. 6, (2017), p. 1550147717713625. eprint: `http://dx.doi.org/10.1177/1550147717713625`. URL: `http://dx.doi.org/10.1177/1550147717713625`.

[8] A. A. Papadopoulos, A. Navarra, J. A. McCann, and C. M. Pinotti. "VIBE: An energy efficient routing protocol for dense and mobile sensor networks". In: *J. Network and Computer Applications* vol. 35, no. 4, (2012), pp. 1177–1190.

[9] A. Navarra, C. M. Pinotti, and A. Formisano. "Distributed colorings for collision-free routing in sink-centric sensor networks". In: *J. Discrete Algorithms* vol. 14, (2012), pp. 232–247.

[10] A. R. Sfar, E. Natalizio, Y. Challal, and Z. Chtourou. “A Roadmap for Security Challenges in Internet of Things”. In: *Digital Communications and Networks* (2017). ISSN: 2352-8648. URL: `http://www.sciencedirect.com/science/article/pii/S2352864817300214`.

[11] C. RJ. “Wireless Sensor Network deployment inside RA-6 Argentine Nuclear Research Reactor for environmental measurement.” In: *4o Simposio Argentino de Informática Industrial. (SII)*. IEEE. 2015, pp. 245–248.

[12] ViswanathanS and JayakumarD. “The Role of Wireless Sensor Network in Tracking Wild Animals Crossing Forest Boundaries.” In: *Programmable Device Circuits and Systems.* 2016, pp. 105–107.

[13] G. Oren, L. Barenboim, and H. Levin. “Adaptive Distributed Hierarchical Sensing algorithm for reduction of wireless sensor network cluster-heads energy consumption”. In: *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*. June 2017, pp. 980–986.

[14] M. Micheletti, L. Mostarda, and A. Piermarteri. “Rotating Energy Efficient Clustering for Heterogeneous Devices (REECHD)”. In: *32nd IEEE International Conference on. Advanced Information Networking and Applications (IEEE AINA 2018), Pedagogical University of Cracow, Poland, May 16-18, 2018*. 2018.

[15] A. W. Khan, A. H. Abdullah, M. A. Razzaque, and J. I. Bangash. “VGDRA: A Virtual Grid-Based Dynamic Routes Adjustment Scheme for Mobile Sink-Based Wireless Sensor Networks”. In: *IEEE Sensors Journal* vol. 15, no. 1, (Jan. 2015), pp. 526–534. ISSN: 1530-437X.

[16] A. Navarra, C. M. Pinotti, V. Ravelomanana, F. B. Sorbelli, and R. Ciotti. “Cooperative training for high density sensor and actor networks”. In: *IEEE Journal on Selected Areas in Communications* vol. 28, no. 5, (2010), pp. 753–763.

[17] F. Barsi, A. A. Bertossi, C. Lavault, A. Navarra, S. Olariu, C. M. Pinotti, and V. Ravelomanana. “Efficient Location Training Protocols for Heterogeneous Sensor and Actor Networks”. In: *IEEE Trans. Mob. Comput.* vol. 10, no. 3, (2011), pp. 377–391.

[18] A. Kumar, V. Kumar, and N. Chand. “Energy Efficient Clustering and Cluster Head Rotation Scheme for Wireless Sensor Networks”. In: *International Journal of Advanced Computer Science and Applications* vol. 3, no. 5, (2012).

[19] H. Ferng and J. Chuang. “Area-partitioned clustering and cluster head rotation for wireless sensor networks”. In: *2017 International Conference on Machine Learning and Cybernetics (ICMLC)*. vol. 2,. July 2017, pp. 593–598.

[20] N. Aierken, R. Gagliardi, L. Mostarda, and Z. Ullah. “RUHEED-Rotated Unequal Clustering Algorithm for Wireless Sensor Networks”. In: *29th IEEE International Conference on Advanced Information Networking and Applications Workshops, AINA 2015 Workshops, Gwangju, South Korea, March 24-27, 2015*. 2015, pp. 170–174. URL: `https://doi.org/10.1109/WAINA.2015.86`.

[21] Z. Ullah, L. Mostarda, R. Gagliardi, D. Cacciagrano, and F. Corradini. "A Comparison of HEED Based Clustering Algorithms – Introducing ER-HEED". In: *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*. Mar. 2016, pp. 339–345.

[22] W. Liu and J. Yu. "Energy efficient clustering and routing scheme for wireless sensor networks". In: *2009 IEEE International Conference on Intelligent Computing and Intelligent Systems*. vol. 3,. 2009, pp. 612–616.

[23] E. Babaee, S. Zareei, and R. Salleh. "Best Path Cluster-Based Routing Protocol for Wireless Sensor Networks". In: *2013 UKSim 15th International Conference on Computer Modelling and Simulation*. 2013, pp. 663–667.

[24] O. Younis and S. Fahmy. "HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks". In: *IEEE Transactions on Mobile Computing* vol. 3, no. 4, (Oct. 2004), pp. 366–379. ISSN: 1536-1233. URL: http://dx.doi.org/10.1109/TMC.2004.41.

[25] C. Sommer, D. Eckhoff, R. German, and F. Dressler. "A computationally inexpensive empirical model of IEEE 802.11p radio shadowing in urban environments". In: *2011 Eighth International Conference on Wireless On-Demand Network Systems and Services*. Jan. 2011, pp. 84–90.

[26] G. Kloos, J. E. Guivant, E. M. Nebot, and F. Masson. "Range Based Localisation Using RF and the Application to Mining Safety". In: *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Oct. 2006, pp. 1304–1311.

[27] C. Sommer and F. Dressler. "Using the Right Two-Ray Model? A Measurement-based Evaluation of PHY Models in VANETs". In: (May 2019).

[28] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan. "An application-specific protocol architecture for wireless microsensor networks". In: *IEEE Transactions on Wireless Communications* vol. 1, no. 4, (Oct. 2002), pp. 660–670. ISSN: 1536-1276.

[29] E. Ever, R. Luchmun, L. Mostarda, A. Navarra, and P. Shah. "UHEED - An Unequal Clustering Algorithm for Wireless Sensor Networks". In: *SENSORNETS*. 2012.

[30] C. T. Sony, C. P. Sangeucetha, and C. D. Suriyakala. "Multi-hop LEACH protocol with modified cluster head selection and TDMA schedule for wireless sensor networks". In: *2015 Global Conference on Communication Technologies (GCCT)*. Apr. 2015, pp. 539–543.

[31] M. Hempstead, M. Welsh, and D. Brooks. "TinyBench: The case for a standardized benchmark suite for TinyOS based wireless sensor network devices". In: *29th Annual IEEE International Conference on Local Computer Networks*. IEEE. 2004, pp. 585–586.

[32] L. Nazhandali, M. Minuth, and T. Austin. "SenseBench: Toward an accurate evaluation of sensor network processors". In: *IEEE International. 2005 Proceedings of the IEEE Workload Characterization Symposium, 2005.* IEEE. 2005, pp. 197–203.

[33] UCB/LBNL/VINT. *Network Simulator - ns*. 1999. URL: https://www.isi.edu/nsnam/ns/ (visited on 03/30/2019).

[34] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. "Energy-Efficient Communication Protocol for Wireless Microsensor Networks". In: *Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 8 - Volume 8*. HICSS '00. Washington, DC, USA: IEEE Computer Society, 2000, pp. 8020–. ISBN: 0-7695-0493-0. URL: http://dl.acm.org/citation.cfm?id=820264.820485.

[35] P. Ding, J. Holliday, and A. Celik. "Distributed Energy-efficient Hierarchical Clustering for Wireless Sensor Networks". In: *Proceedings of the First IEEE International Conference on Distributed Computing in Sensor Systems*. DCOSS'05. Marina del Rey, CA: Springer-Verlag, 2005, pp. 322–339. ISBN: 3-540-26422-1, 978-3-540-26422-4. URL: http://dx.doi.org/10.1007/11502593_25.

[36] P. Liu, T. l. Huang, X. y. Zhou, and G. x. Wu. "An improved Energy Efficient Unequal Clustering algorithm of wireless sensor network". In: *2010 International Conference on Intelligent Computing and Integrated Systems*. Oct. 2010, pp. 930–933.

[37] M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu, and S. Singh. "Exploiting heterogeneity in sensor networks". In: *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.* vol. 2,. Mar. 2005, 878–890 vol. 2.

[38] C. Duan and H. Fan. "A Distributed Energy Balance Clustering Protocol for Heterogeneous Wireless Sensor Networks". In: *2007 International Conference on Wireless Communications, Networking and Mobile Computing*. Sept. 2007, pp. 2469–2473.

[39] G. Smaragdakis, I. Matta, and A. Bestavros. "SEP: A Stable Election Protocol for clustered heterogeneous wireless sensor networks". In: *Second International Workshop on Sensor and Actor Network Protocols and Applications (SANPA 2004)*. Boston, MA, Aug. 2004.

[40] F. Ishmanov and S. W. Kim. "Distributed Clustering Algorithm with Load Balancing in Wireless Sensor Network". In: *2009 WRI World Congress on Computer Science and Information Engineering*. vol. 1,. Mar. 2009, pp. 19–23.

[41] D. Kumar, T. C Aseri, and R. Patel. "Distributed Cluster Head Election (DCHE) Scheme for Improving Lifetime of Heterogeneous Sensor Networks". In: vol. 13, (Sept. 2010).

[42] D. Kumar, T. C. Aseri, and R. Patel. "EEHC: Energy efficient heterogeneous clustered scheme for wireless sensor networks". In: *Computer Communications* vol. 32, no. 4, (2009), pp. 662 –667. ISSN: 0140-3664. URL: http://www.sciencedirect.com/science/article/pii/S0140366408006087.

[43] S. V. Purkar and R. S. Deshpande. "Energy Efficient Clustering Protocol to Enhance Performance of Heterogeneous Wireless Sensor Network: EECPEP-HWSN". In: *Journal of Computer Networks and Communications* (2018). https://doi.org/10.1155/2018/2078627.

[44] M. Zeng, X. Huang, B. Zheng, and X. Fan. “A Heterogeneous Energy Wireless Sensor Network Clustering Protocol”. In: *Wireless Communications and Mobile Computing* (2019). https://doi.org/10.1155/2019/7367281.

[45] S. Singh, A. Malik, and R. Kumar. “Energy efficient heterogeneous DEEC protocol for enhancing lifetime in WSNs”. In: *Engineering Science and Technology, an International Journal* vol. 20, no. 1, (2017), pp. 345 –353. ISSN: 2215-0986.

[46] M. Xie and X. Wang. “An Energy-Efficient TDMA Protocol for Clustered Wireless Sensor Networks”. In: *2008 ISECS International Colloquium on Computing, Communication, Control, and Management*. vol. 2,. Aug. 2008, pp. 547–551.

[47] G. Y. Lazarou, J. Li, and J. Picone. “A Cluster-based Power-efficient MAC Scheme for Event-driven Sensing Applications”. In: *Ad Hoc Netw.* vol. 5, no. 7, (Sept. 2007), pp. 1017–1030. ISSN: 1570-8705.

[48] S. Pushpan and B. Velusamy. “Fuzzy-Based Dynamic Time Slot Allocation for Wireless Body Area Networks”. In: *Sensors* vol. 19, no. 9, (2019). ISSN: 1424-8220. URL: https://www.mdpi.com/1424-8220/19/9/2112.

[49] H. Zhang and P. Gburzynski. “A Variable Slot Length TDMA Protocol for Personal Communication Systems”. In: *Wirel. Pers. Commun.* vol. 22, no. 3, (Sept. 2002), pp. 409–432. ISSN: 0929-6212. URL: https://doi.org/10.1023/A:1020232331191.

[50] Song Ci, H. Sharif, and K. Nuli. “Study of an adaptive frame size predictor to enhance energy conservation in wireless sensor networks”. In: *IEEE Journal on Selected Areas in Communications* vol. 23, no. 2, (Feb. 2005), pp. 283–292. ISSN: 0733-8716.

[51] S. U. Hashmi, J. H. Sarker, H. T. Mouftah, and N. D. Georganas. “An Efficient TDMA Scheme with Dynamic Slot Assignment in Clustered Wireless Sensor Networks”. In: *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*. Dec. 2010, pp. 1–6.

[52] N. A. M. Alduais, L Audah, A Jamil, and J Abdullah. “The performance evaluation of different logical topologies and their respective protocols for wireless sensor networks”. PhD thesis. Universiti Tun Hussein Onn Malaysia, 2015.

[53] W. Dong, X. Liu, C. Chen, Y. He, G. Chen, Y. Liu, and J. Bu. “DPLC: Dynamic Packet Length Control in Wireless Sensor Networks”. In: *2010 Proceedings IEEE INFOCOM*. Mar. 2010, pp. 1–9.

[54] L. Bing. “A dynamic TDMA protocol based on correlation in wireless sensor networks”. In: *2016 6th International Conference on Electronics Information and Emergency Communication (ICEIEC)*. IEEE. 2016, pp. 245–248.

[55] M. Xie and X. Wang. “An energy-efficient TDMA protocol for clustered wireless sensor networks”. In: *2008 ISECS International Colloquium on Computing, Communication, Control, and Management*. vol. 2,. IEEE. 2008, pp. 547–551.

[56] S. Dutt and O. Khanna. “An Enhanced Energy Efficient Clustering Scheme for Prolonging the Lifetime of Heterogeneous Wireless Sensor Networks”. In: *International Journal of Computer Applications* vol. 76, no. 8, (2013), pp. 27–32.

[57] D. S. Mantri, N. R. Prasad, and R. Prasad. "Bandwidth Efficient Cluster-based Data Aggregation for Wireless Sensor Network". In: *Comput. Electr. Eng.* vol. 41, no. C, (Jan. 2015), pp. 256–264. ISSN: 0045-7906.

[58] L. Bing. "A dynamic TDMA protocol based on correlation in Wireless Sensor Networks". In: *2016 6th International Conference on Electronics Information and Emergency Communication (ICEIEC)*. June 2016, pp. 245–248.

[59] Y. T. Hou and Y. Shi. "Variable bit rate flow routing in wireless sensor networks". In: *IEEE transactions on wireless communications* vol. 6, no. 6, (2007), pp. 2140–2148.

[60] R. Wan, N. Xiong, and N. T. Loc. "An energy-efficient sleep scheduling mechanism with similarity measure for wireless sensor networks". In: *Human-centric Computing and Information Sciences* vol. 8, no. 1, (2018), p. 18.

[61] "A multi-hop graph-based approach for an energy-efficient routing protocol in wireless sensor networks". In: *Human-centric Computing and Information Sciences* vol. 8, no. 1, (2018), p. 30.

[62] D. Wohwe Sambo, B. Yenke, A. Förster, and P. Dayang. "Optimized Clustering Algorithms for Large Wireless Sensor Networks: A Review". In: *Sensors* vol. 19, (2019). https://doi.org/10.3390/s19020322.

[63] B. Jan, H. Farman, H. Javed, B. Montrucchio, M. Khan, and S. Ali. "Energy Efficient Hierarchical Clustering Approaches in Wireless Sensor Networks: A Survey". In: *Wireless Communications and Mobile Computing* (2017). https://doi.org/10.1155/2017/645794.

[64] I. Akila, S. Manisekaran, and R. Venkatesan. "Modern Clustering Techniques in Wireless Sensor Networks". In: *Wireless Sensor Networks - Insights and Innovations*. InTech, Oct. 2017. URL: `https://doi.org/10.5772/intechopen.70382`.

[65] F. K. Shaikh and S. Zeadally. "Energy harvesting in wireless sensor networks: A comprehensive review". In: *Renewable and Sustainable Energy Reviews* vol. 55, (2016), pp. 1041 –1054. ISSN: 1364-0321. URL: `http://www.sciencedirect.com/science/article/pii/S1364032115012629`.

[66] M. S. Bahbahani and E. Alsusa. "A Cooperative Clustering Protocol With Duty Cycling for Energy Harvesting Enabled Wireless Sensor Networks". In: *IEEE Transactions on Wireless Communications* vol. 17, no. 1, (Jan. 2018), pp. 101–111. ISSN: 1536-1276.

[67] C. Moraes and D. Har. "Charging Distributed Sensor Nodes Exploiting Clustering and Energy Trading". In: *IEEE Sensors Journal* vol. 17, no. 2, (Jan. 2017), pp. 546–555. ISSN: 1530-437X.

[68] A. H. Duffy. "The "What" and "How" of Learning in Design". In: *IEEE Intelligent Systems* vol. 12, (May 1997), pp. 71–76. ISSN: 1541-1672. URL: `doi.ieeecomputersociety.org/10.1109/64.590079`.

[69] P. Langley and H. A. Simon. "Applications of Machine Learning and Rule Induction". In: *Commun. ACM* vol. 38, no. 11, (Nov. 1995), pp. 54–64. ISSN: 0001-0782. URL: `http://doi.acm.org/10.1145/219717.219768`.

[70] M. A. Alsheikh, S. Lin, D. Niyato, and H. P. Tan. “Machine Learning in Wireless Sensor Networks: Algorithms, Strategies, and Applications”. In: *IEEE Communications Surveys Tutorials* vol. 16, no. 4, (Aug.–Dec. 2014), pp. 1996–2018. ISSN: 1553-877X.

[71] J.-L. Liu and C. Ravishankar. “LEACH-GA: Genetic Algorithm-BasedEnergy-Efficient Adaptive Clustering Protocolfor Wireless Sensor Networks”. In: vol. 1, (Jan. 2011), pp. 79–85.

[72] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. “Energy-efficient communication protocol for wireless microsensor networks”. In: *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*. Jan. 2000, 10 pp. vol.2–.

[73] P. Sivakumar and M. Radhika. “Performance Analysis of LEACH-GA over LEACH and LEACH-C in WSN”. In: *Procedia Computer Science* vol. 125, (2018). The 6th International Conference on Smart Computing and Communications, pp. 248 –256. ISSN: 1877-0509. URL: `http://www.sciencedirect.com/science/article/pii/S1877050917327989`.

[74] R. Pachlor and D. Shrimankar. “LAR-CH: A Cluster-Head Rotation Approach for Sensor Networks”. In: *IEEE Sensors Journal* vol. 18, no. 23, (Dec. 2018), pp. 9821–9828. ISSN: 1530-437X.

[75] M. Zeng, X. Huang, and X. Fan. “A Heterogeneous Energy Wireless Sensor Network Clustering Protocol”. In: *Wireless Communications and Mobile Computing* (2019). https://doi.org/10.1155/2019/7367281.

[76] J. Al-Karaki and A. Kamal. “Routing Techniques in Wireless Sensor Networks: A Survey”. In: *Wireless Communications, IEEE* vol. 11, (Jan. 2005), pp. 6 –28.

[77] A. Sarkar and T. S. Murugan. “Routing protocols for wireless sensor networks: What the literature says?” In: *Alexandria Engineering Journal* vol. 55, no. 4, (2016), pp. 3173 –3183.

[78] S. Jabbar, M. A. Habib, A. A. Minhas, A. Mudassar, R. Ashraf, S. Khalid, and K. Han. “Analysis of Factors Affecting Energy Aware Routing in Wireless Sensor Network”. In: *Wireless Communications and Mobile Computing, Hindawi* vol. 2018, (Feb. 2018), p. 21.

[79] V. K. Arora, V. Sharma, and M. Sachdeva. “A distributed, multi-hop, adaptive, tree-based energy-balanced routing approach”. In: *International Journal of Communication Systems* vol. 32, no. 9, (2019), e3949.

[80] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados. “Energy-Efficient Routing Protocols in Wireless Sensor Networks: A Survey”. In: *IEEE Communications Surveys Tutorials* vol. 15, no. 2, (Feb. 2013), pp. 551–591. ISSN: 1553-877X.

[81] X. Liu. “A novel transmission range adjustment strategy for energy hole avoiding in wireless sensor networks”. In: *Journal of Network and Computer Applications* vol. 67, (2016), pp. 43 –52. ISSN: 1084-8045. URL: `http://www.sciencedirect.com/science/article/pii/S1084804516300066`.