Energy Restoration of Sensor Networks by Mobile Robots

by

Eman Omar

Thesissubmitted to the
Faculty of Engineering
In partial fulfillment of the requirements
For the Ph.D. degree in
Computer Science

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

In this thesis, a variety of different approaches are proposed to study the energy restoration problem in wireless sensor networks by one or more robots.

First, we introduce an on-demand decentralized strategy performed by a robot that visits the sensors in a predefined circular order. We study it both analytically and experimentally analyzing the impact of various network parameters on network coverage, disconnection time, and time sensors have to wait to be served. We then introduce an optimal centralized approach as a benchmark to assess how close to optimal our on-demand strategy is, and we discover that, for sufficiently large networks, the on-demand strategy is indeed optimal. We then propose an even simpler mechanism where the robot simply moves blindly along the circular order, which is experimentally shown to be as efficient as the other two. The results above apply to arbitrary sensor network; we then consider a common special topology: a linear arrangement of sensors, were we propose three restoring mechanisms. We compare them experimentally discovering, once again, that the simplest approach is also the best, in most cases. We finally consider the case of multiple robots. We propose two strategies where the network is portioned among the robots and each robot takes care of a portion, and we compare those with a collaborative strategy where all robots work on the global network.

The main general result of this study is that simple solutions are often as good as more sophisticated ones. In fact, a totally blind strategy where a robot simply moves around restoring energy on its way turns out to be as efficient as the best possible centralized solution for most networks.

Acknowledgements

I would like to take this opportunity to express my thanks to those who helped me with various aspects of conducting research and the writing of the thesis.

First and foremost, I would like to express my deepest appreciation and sincere thanks to my supervisor Prof. Paola Flocchini, who provided me an opportunity to join her team, and provided me with a remarkable scientific and moral support that most students envy. Her ideas, guidance, and recommendations always guided me to the right direction, and her knowledge lightened the dark corners of the route to PhD. As the words come short in expressing my gratitude, there is no doubt that without her precious support it would not be possible to conduct this research. Moreover, I would like to thank my co-supervisor Prof. Nicola Santoro that he shared his knowledge, and comments with me during this research. He patiently provided me with directions of this research.

My sincerest gratitude to my parents for unlimited support. I have no enough expressions that describe my thankfulness for your love and care. This success and all successes are yours.

Special warmest thanks goes to my husband Abdullah Aref, and to our kids Mohammad, Abedalaziz, Maryam and sarah for supporting me spiritually throughout this thesis and throughout my life. You are my sweet heart, and to whom I dedicate my whole life to endless care and love

I am grateful to my brothers and sisters who have provided me through moral and emotional support. I am also grateful to my other family members and friends who have supported me along the way.

Dedication

I dedicate my dissertation work to my husband Abdullah Aref, whose words of encouragement and push for tenacity has been the fuel to my enthusiasm for research.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Wireless Sensor Networks (WSNs)

A Wireless Sensor Network (WSN) is a network composed of a large number of low-cost tiny sensor nodes deployed on a target field and able to communicate wirelessly in short distances [3].

A typical node normally contains four basic components: a sensing unit, a processing unit, a communication unit, and a power unit. A block diagram of a typical node is shown in Figure 1.1. Sensing units have sensors that measure environmental conditions and provide row data in the form of analogue signals, which are digitized by analogue-to-digital converters before being sent to processing units. The processing unit manages the

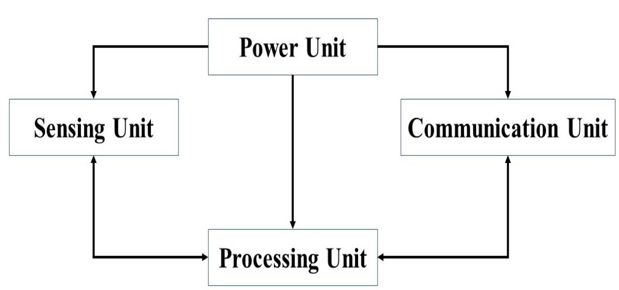


Figure 1.1: A block diagram of a typical node

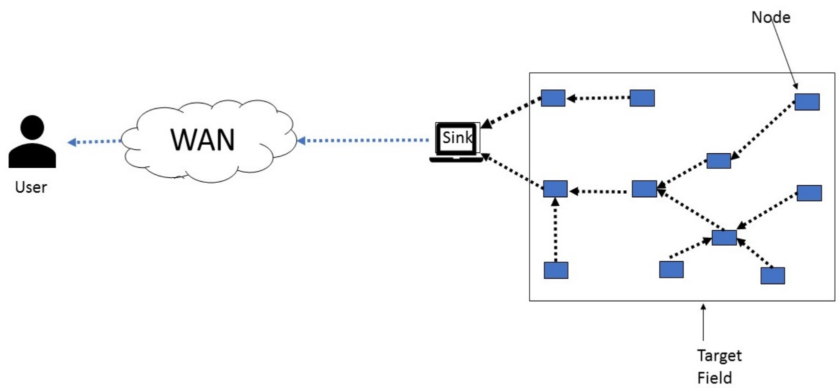


Figure 1.2: Sketch of the general architecture of a WSN

behavior of the node; it can also manipulate data slightly before sending it. A transceiver can be used used to exchange messages with other nodes. A power source (e.g., a battery) provides the power necessary for the node to operate [3]. Other application specific units can be added when necessary.

In a typical deployment, sensors monitor their surrounding and report to special purpose nodes, called sinks, for processing [15]. A sketch of the general architecture of a WSN is shown in Figure 1.2. Since the nodes communicate data wirelessly, WSN is considered easy to install. Simplicity of deployment and low cost have made WSNs attractive, and WSNs have been applied to hundreds of applications in a variety of domains [14, 82]. The main application domains have been identified as: military, environment, habitat, public, industrial, health, and business [100]. Regardless of the domain, the main use of WSNs is for monitoring the conditions of the area in which they are deployed (e.g., [24, 100]).

1.2 Energy Restoration

1.2.1 Energy in WSN

Regardless of the application, the ultimate goal of any sensor network is to achieve accurate sensing and maximize lifetime while maintaining an acceptable level of coverage. Sensing, processing, and communication are the main reasons for power consumption in WSNs; the last of these three activities is responsible the largest portion of power use [3, 37]. As most existing sensors are powered by batteries, their lifetime is limited by the storage capacity of the battery used [22]. The time interval between the starting time of a WSN until it stops functioning is refereed to as the lifetime or the spanning time of the WSN [85]. Each WSN has a different lifespan depending on sensor types, energy sources, and use [37].

Once the battery becomes depleted, the node is no longer sensing and, in absence of redundant coverage, this *sensing hole* creates a *coverage hole* in the monitored area. Indeed, the coverage provided by the network degrades over time and eventually disappears if no action is taken. This represents a main challenge for environments where sensors are deployed to work for long term, for example for surveillance tasks, such as road condition monitoring [85].

The most simplistic approach to cope with the eventual loss of coverage has been to deploy a very large number of sensors to compensate for the loss of the depleted ones. However, for obvious environmental reasons, these kind of solutions are not really sustainable; furthermore, regardless of the number of spares, a loss of coverage over time is inevitable since there are no provisions to recharge or replace sensors in the long run.

Extensive research has being carried out on energy in WSNs, concentrating on *energy management* strategies, whose goal is to prolong the lifetime of the network and delay the progressive coverage decay by balancing the energy levels among the sensors (e.g., see [4, 49]). This however only extends the life of the system but is not going to prevent the network from eventually dying.

A very different line of research has been on *energy restoration*, with the ambitious goal to maintain the network operating perpetually. In this line are proposals to enhance the sensors with (radically different) additional capabilities.

For instance, to overcome the finite lifetime of the battery powered sensor nodes, in recent years, researchers investigated ways to have the sensors extracting energy from the environment (such as solar energy, thermoelectric energy, wind energy, waves energy, mechanical vibration, and body motion, wireless power transmission, RF radiation energy, and inductive coupling) and to convert it to electrical energy, enabling them to recharge their batteries (e.g., see [70, 83]). [37]. However, the dependence on uncontrollable environmental conditions limits the use of such schemes for providing reliable power supply [74].

A different direction is to add mobility to the sensors, enabling them to move to recharge facilities deployed throughout the sensing area (e.g., [57, 75, 76]). The drawback of this approach is the increased complexity, and thus cost, of the sensor nodes; this at a time when technology trends are scaling sensors to be smaller and cheaper.

1.2.2 Energy Restoration by Mobile Robot

An alternative way to achieve energy restoration without adding complexity to the nodes is by using one or more external mobile devices, typically called *robots*, which would go around to restore energy to nodes with (near) depleted energy. The restoration can take place by either *recharging* the depleted batteries or by *replacing* them entirely with fully charged one. Fueled by the recent evolution in wireless power transfer techniques [93], the research on sensor recharging by mobile robots has been quite intensive (e.g., see [7, 16, 46, 53, 68, 90, 93]).

The alternative of replacement has been considered in the literature (e.g., see [58, 86]), albeit with less intensity. For a comparison between these two alternatives see [58].

Notice that the idea of using robots in sensor networks is not new, as it has been proposed for data gathering and aggregation, for network repairs where the special maintenance entities replace depleted sensors (e.g., [44, 56, 81, 84]), as well as for other network maintenance tasks [42]. Other examples of these approaches can be found in research papers in robotics [9, 18, 78]; in all cases, a robot is responsible for delivering energy to a swarm of sensors.

Regardless of whether restoration is by recharging or replacement, after servicing a node, the robot must choose where to move next. The algorithm followed by the robot to make this decision, here called *energy restoration strategy*, may prescribe the acquisition of information from the sensors (e.g., energy level, location, etc) and require possibly complex computations by the robot.

Most of the existing work is based on offline approaches, where the decisions about how the recharging has to be done is made *offline*, that is, when all the data about the sensor network is available and known a priori. Among the *online* approaches (where data is not a-priori available but arrives as events occur), the majority of the work relies on the availability of a *centralized* unit that controls and manage the charges requests (e.g., see [28, 34, 39, 45, 68, 86, 103] is communicated to the robot that then computes where to go next; alternatively, the information is communicated to the base station, which takes the decisions and provides the mobile robot with instructions. In addition to the high communication costs required, the optimization requirements to be met by the decision are typically accompanied by high computational complexity, which grows non-linearly with the number of sensor nodes; these factors imply a difficulty to scale for these strategies.

The focus of this thesis is instead on fully *decentralized online* energy restoration strategies. Moreover, unlike most other work in this setting, our focus is on *network coverage*. In fact, we would like to design strategies that guarantee as much as possible continuous operations of the sensor network, limiting the number of sensing holes created by batteries depletion and their duration.

1.2.3 Effectiveness and Costs

The fact that some sensors might be inactive at some times is not a problem if an energy restoration strategy is in place. Indeed, the goal of an energy restoration strategy should be to maintain as much as possible of the network operational at any time.

The *effectiveness* of an energy restoration strategy towards this goal finally rests with the number of sensors it is able to maintain operational at any given time, in spite of battery depletions. We shall call this measure *operational size* or, with an abuse of notation, *coverage*. Note that effectiveness can be equivalently measured by the number of sensing holes in the network at any given time; interestingly, unless the deployment has $k > 1$ redundant coverage, this measure corresponds to the number of coverage holes at that time.

The other effectiveness measure of interest is the time from the moment a sensor becomes no longer operational to the time when the robot arrives to serve it; i.e., for how long a sensing hole lasts. We shall call this measure *disconnection time*.

The effectiveness of any strategy depends on many factors, first and foremost the battery capacity of the sensors and how the restoration is physically achieved (e.g., by battery replacement, by wireless transmission, etc.). More precisely, the two crucial associated system parameters are the amount of time Δ a fully charged battery lasts under normal operations and the amount of time ρ the charging/replacement requires. Observe that, if restoration is by recharging, ρ generally depends on the current energy level of the battery; on the other hand, if restoration is by battery replacement, ρ is the same regardless of the battery level (i.e., it is a constant). We denote by $\mathcal{N}(\Delta, \rho)$ the set of networks whose components and restoration technology has these parameters.

Clearly, an important variable is the size of the network, i.e.,number n of deployed sensors. Notice that, as sensor technology becomes more affordable and reliable, the possibility of deploying hundreds or thousands of sensor nodes gets closer to reality.

Associated with each energy restoration strategy are also the computation and communication *costs* incurred when the robot operates in the network according to that strategy. These costs determine the *efficiency* of a strategy.

1.3 Contributions

In this thesis we focus on fully decentralized online energy restoration strategies by a robot and study their effectiveness. We then extend the study to multiple robots.

Our main contributions are indicated below:

1. We first consider a very simple on-demand recharging strategy, which we shall call ON-DEMAND. In this strategy, the robot visits the sensors in a predefined circular order, moving in a "clockwise" direction when aware of a pending request. A node whose battery is about to become depleted (e.g., its battery level falls below a predefined threshold) originates a recharging request and waits for the robot; the request is forwarded in a "counter-clockwise" direction until it reaches either the robot or another node waiting for the robot. In other words, each node communicates only locally: with the neighboring sensors in the circular order (to send or receive a request), or with the robot if currently there (to communicate the presence of a pending request); the robot moves only from one node to the next in the cyclic order, is aware only of whether or not there is a pending request, and has no need of memory or calculation.

   In contrast to the existing centralized strategies, this simple on-line strategy is fully distributed and decentralized, uses only local communication, requires no computations, and it is highly scalable. In other words, ON-DEMAND is highly efficient.

   We study the effectiveness of ON-DEMAND, both analytically and experimentally. We establish several analytical results related to the *stability* of the system, whereas the network is seemed to be in a stable state if the order in which the nodes are charged in a round is the same in every round ON-DEMAND the nodes are always visited in the same order in every round. In particular, we prove that, for almost all networks in $\mathcal{N}(\Delta, \rho)$, once the system becomes stable, although the set of operational nodes changes in time, its size remains unchanged; furthermore this value is the same regardless of the initial network size. We also determine the disconnection time for such stable networks, providing a precise characterization of the effectiveness of ON-DEMAND when the threshold is minimal. These results are established assuming that ρ is constant; this is always the case when energy restoration is by battery replacement, as well in some cases when restoration is by recharging (e.g., robots only recharges depleted sensors).

We then run extensive simulations considering both constant and variable ρ. The most interesting experimental results are that, regardless of Δ, ρ, n:

(a) the system stabilizes very quickly (within two rounds);

(b) both the coverage size and the disconnection time match the analytical values established for them when ρ is constant.

In other words, the experimental results show that the analytical results on effectiveness of ON-DEMAND hold regardless of whether restoration is by recharging or by replacement.

2. To put the results on the effectiveness of ON-DEMAND into perspective, we consider the "ideal" optimal (from the effectiveness point of view) on-line strategy, which we shall call simply OPTIMAL, where the robot at any time knows the current status of all sensors, and it computes on-line which request it must satisfy next so to minimize the number of sensing holes and/or their duration. Observe that, even more that any other centralized on-line strategy, in addition to a high computational complexity, OPTIMAL requires constantly up-to-date global information; hence the communica-

tion costs required to implement it severely limit its feasibility. On the other hand, its effectiveness is by definition the best possible and we study it to provide a benchmark for assessing the other strategies.

The effectiveness and the costs of the strategy employed by the robot to service the sensors depends on many factors, a crucial one being the amount of information about the network status available at any given time to the robot. Now, On-Demand provides the robot only with the awareness of the existence of at least one current request.

We show the perhaps unexpected result that, in most networks, the effectiveness of On-Demand is equivalent to that of Optimal. In other words, in most cases, in spite of its simplicity and its extremely small (communication and computation) costs, the decentralized strategy is as effective as the ideal optimal one which is instead highly expensive, both computationally and communication-wise, and hardly feasible.

This is particularly interesting because it counters the intuitive notion that providing the robot with complete up-to-date information should give a critical advantage with respect to providing the robot only with the awareness of the existence of at least one current request.

3. We then consider an even simpler energy restoration strategy that requires no information at all. In this strategy, that we call Blind, the charger visits the sensors in a fixed circular order, like in On-Demand; if the current sensor has a battery level below the threshold, the robot restores it and then moves to the next sensor; otherwise, it moves directly to the sensor next in the order. Nodes in need of charge do not send requests; they simply wait for the robot to pass by. That is, no requests are issued; the robot always operates in the same way, unaware of whether there are pending requests at some node other than the current one, without need of memory

or calculation. Note that, while in the On-Demand strategy the robot moves only if there is a request, in the Blind strategy it might move uselessly several times without performing any work before encountering a sensor in need.

We show that, surprisingly, the Blind strategy performs almost as well as the On-Demand strategy. We analyze experimentally the effectiveness of this strategy and show that:

(a) Like for On-Demand, the system reaches stability after a small transient (two rounds); in most networks, after a transient, the coverage size does not change.

(b) In most networks, Blind has approximately the same coverage size and disconnection time as On-Demand.

As a consequence, by our previous results, our experiments show (even more counterintuitively) that the optimal on-line effectiveness of Optimal can be achieved in most networks without any knowledge and without any communication or processing overhead.

4. We then start the investigation of energy restoration strategies for *linear networks*, and focus on three decentralized on-line strategies with increasingly simpler operational requirements.

   In the first strategy, which we shall call On-Request, a sensor whose battery is about to become depleted originates a restoring request, and waits for the robot; the request travels towards the robot until it reaches either the robot or another sensor waiting for the robot; in absence of requests, the robot does not move. In this strategy, each sensor communicates only locally: with the neighboring sensor in the line (to send a request), or with the robot if currently there (to communicate the presence and direction of a pending request).

   Also in the second strategy, which we shall call Straight, a node requiring recharge issues a request that travels until it reaches either the robot or another sensor waiting for it, and the robot moves only if there are pending requests. Unlike the previous strategy, here the moving direction of the robot does not change until it reaches the end of the line. Also this strategy requires only local communication.

The third strategy, SLIDE, is extremely simple and requires no communication at all: no requests are sent, and the robot automatically and continuously moves from one end of the line to the other, servicing any sensor found needing restoring. Note that, while in the ON-REQUEST and STRAIGHT strategies the robot moves to serve a sensor only when needed, in the SLIDE strategy it might move uselessly along the line several times without performing any work before encountering a sensor in need.

We study these three strategies running extensive simulations to assess their effectiveness.

Interestingly, we make the counter-intuitive discovery that, among the two on-demand strategies, the simpler STRAIGHT performs at least as well as the standard ON-REQUEST; furthermore, the SLIDE strategy, which requires absolutely no communications, performs as well as the two on-demand strategies with respect to both metrics.

We also provide strong analytical support to these experimental findings. In fact we prove that, starting with initially empty batteries, the SLIDE strategy has better coverage performance that the other two strategies for almost all network sizes. Indeed, for some network sizes, we prove that SLIDE is optimal: no other strategy, even if centralized and off-line, can do better.

5. Finally, we investigate how the effectiveness of the simple efficient decentralized strategies can be improved by using *multiple robots*. We investigate and compare two different approaches.

   The first approach (OVERPASS) is quite intuitive: the robots are initially placed at roughly equal intervals along a circular ordering of the sensors, and they perpetually and blindly move in the same direction, stopping to service any node needing restoring. In case a robot encounters another robot at work on a node, it just overpasses it and continues. In the second approach the circular ordering is partitioned into k regions (where k is the number of available robots), each containing approximately the same number of nodes. Each region is assigned to a different robot that becomes responsible for energy restoration for that region, independently of the rest of the network and of the other robots. Within this approach, we consider the choice of each robot using a fixed circular order of the nodes in its region (SUB-RING), or a fixed linear order of the nodes in its region (SUB-SEGMENT). Clearly, when considering the robot operating in its region, we could have used any of the charging strategies studied in the rest of the thesis. Given the excellent results offered by BLIND (for the circular ordering) and SLIDE (for the linear arrangement), we use those within our SUB-RING and SUB-SEGMENT strategies.

We study the two strategies running extensive simulations to assess their effectiveness, varying several network parameters. With respect to *minimum coverage size*, our experiments indicate is that both SUB-RING and SUB-SEGMENT are always better (i.e., it has larger coverage) than OVERPASS. With respect to maximum disconnection time, SUB-RING always outperform OVERPASS, but this is not the case for SUB-SEGMENT, where the results depend on the number of robots. In fact, SUB-SEGMENT outperforms OVERPASS only when a sufficient number of robots is deployed; when fewer robots are available, OVERPASS achieves a better (i.e., smaller) maximum disconnection time.

Both SUB-RING and SUB-SEGMENT achieve a near *optimal speed-up* in effectiveness: the coverage size increases, and the disconnection time decreases, by a factor of k with respect to the near-optimal strategy for a single robot.

1.4 Organization

The rest of the theses is organized as follows:

- In Chapter 2, we present some background information reviewing the research efforts on the use of mobile robots.
- Chapter 3 describes the setting and introduces the performance measures we intend to evaluate.
- In Chapter 4 we introduce the On-Demand strategy. We study its effectiveness in terms of various performance measures: *charging latency* to evaluate the time it takes for a requesting node to be served; *coverage size* and *disconnection time* to evaluate the extent and duration of coverage holes.
- In Chapter 5 we introduce the optimal Centralized strategy to be used as benchmark and we compare the efficacy of On-Demand with the optimal one, showing that for most network they coincide.
- Chapter 6 describes an even simpler decentralized approach: Blind. We study its efficacy experimentally and we compare the results with the ones obtained by On-Demand. We show that, in spite of the extreme simplicity of this approach, its efficacy is the same. The results of this chapter appeared in [64].
- In Chapter 7 we turn to a particular topology (that will also be useful in the next chapter): a linear arrangement of sensors. We present three strategies: (On-Request, Straight, and Slide). We study their effectiveness and we compare them showing, once more, that the simplest and less demanding strategy is equivalent to the others for sufficiently large networks. The results of this chapter appeared in [65].

- In Chapter 8 we focus on situations when multiple mobile robots are available. We design three strategies (Overpass, Sub-ring, Sub-segment), based on different collaborative approaches. The different strategies are analyzed and compared in terms of disconnection time and coverage size.

- Chapter 9 draws some conclusions and discusses potential directions for future research.

Chapter 2

Related Work

2.1 Introduction

Being a low cost and a small device, sensors can only be equipped with limited energy sources which provides limited functioning time; in fact, most existing sensors are powered by batteries. [2] . Consequently, deployed sensors eventually deplete the energy storage; once the battery becomes depleted, the node is no longer sensing and, in absence of redundant coverage, this *sensing hole* creates a *coverage hole* in the monitored area. Indeed, the coverage provided by the network degrades over time and eventually disappears if no action is taken.

The most simplistic approach to cope with this problem has been to deploy a very large number of sensors, those not needed for the coverage acting as *spares* ready to compensate for the loss of the depleted ones. However, for obvious environmental reasons, these kind of solutions are not really sustainable; furthermore, regardless of the number of spares, a loss of coverage over time is inevitable since there are no provisions to recharge or replace sensors in the long run.

Recently, the scientific research for enhancing the lifetime of wireless sensor networks considerably increased [38] and resulted in a variety of schemes that can be used in a wide range of situations [17]. The extensive research has mainly concentrated on *energy management* strategies, whose goal is to prolong the lifetime of the network and delay the progressive coverage decay by balancing the energy levels among the sensors (e.g., see [4, 49]).

A very different line of research has been on *energy restoration*, with the ambitious goal to maintain the network operating perpetually. This is precisely the line of investigations to which this thesis belong.

In this chapter, after briefly describing some of the approaches used to prolong the lifetime of wireless sensor networks, we review the recent work on energy restoration of sensor networks by mobile robots.

2.2 Energy Conservation

Schemes used to prolong the lifetime of wireless sensor networks can be analyzed from different perspectives. Various dimensions to classify and characterize such schemes have been presented in different surveys, reviews, and theoretical evaluation frameworks exist in the literature. Some of them are based energy harvesting techniques such as [50, 87, 83, 1, 69], others based on energy conservation such as [4, 40]. Some reviews focus on concrete aspects or functionalities like power aware routing techniques [66] and power aware clustering techniques [31]. Others are more general like [38].

Many approaches has been proposed in the literature to prolong the lifetime of WSN based on enhancing energy storage, using topology control and reducing energy consumption.

Topology control schemes that adapt dynamic network topology in accordance to the application requirements. The aim is to set some nodes in sleep mode while keeping the network operational, hence prolonging network lifetime [38].

Considerable efforts have been conducted to prolong the life time of WSNs based on reducing energy consumption. Such schemes can be typically categorized as duty cycling based, data-driven based, and mobility based schemes. Sensors alternate between sleep and wake-up modes in the first approach aiming to achieve efficient energy utilization [96, 106, 13, 12, 30, 89, 63]. Several schemes are adopted to reduce data or to predict data in the second approach. A detailed survey for about in-network processing is presented in [21]. On the other hand, mobility based schemes use a mobile sink or a mobile relay depending on its behavior, which can be used as part of the environment or part of the network [38]. Several algorithms are proposed for efficient energy utilization in each case[10, 98, 104, 77, 35]

In WSN, different entities can be mobile to prolong the life time of WSN in different ways including reducing power consumption (beyond duty cycling and data-driven techniques) [4].

The Mobile relay model for data collection in multi-hop ad hoc networks has already been explored in the context of opportunistic networks [5]. One of the most well-known approaches is given by the message ferrying scheme [36] where a set of mobile nodes pass by sensors and carry data toward the sink. When the cost of deploying a mobile robot or mobilizing sensor nodes is unjustified, sensors can be attached to entities that are moving in the sensing area, such as vehicles or animals. Unfortunately, in such cases; nodes mobility can be uncontrollable and unpredictable [38].

2.3 Energy Restoration

2.3.1 Energy Harvesting

Being equipped with limited energy sources, deployed sensors can eventually deplete the energy storage and loss of coverage can occur [88]. One way to address this challenge is extracting energy from the environment [2].

Depending on the environment, different sources of energy exists which can be harvested and used to extend battery life of a sensor node. The harvested energy varies from different sources, of which, typical examples are solar, wind energy, thermal power, etc [67, 83]. The authors in [70] studied the feasibility of energy harvesting from ambient environment [38].

Although energy harvesting is a promising solution, its success in WSNs may be limited in by the time-varying nature of energy-harvesting sources, which can be a great challenge in sensor energy management [46]. Furthermore, the cost of energy-harvesting sensors will be significantly increased, especially for large-scale WSN. The size of an energy-harvesting device is also a concern for such deployment, particularly when the size of a solar panel is of a much larger scale than the sensor itself [46].

2.3.2 Node Mobility

A proactive scheme to energy redistribution and restoration presented in [88, 76] for situations where mobility becomes a sensor's attribute and energy service stations are static. Sensors tend to move toward a charging station before they reach a critical energy level, and coordinate with other neighboring sensors to avoid coverage holes due to this movement. Such an approach found to be more efficient than simply waiting until the energy level is critical to move toward, possibly far away, charging station and create a coverage hole.

2.3.3 Restoration by Mobile Robots

An alternative to adding more complexity to the nodes has been the proposal of using one or more external mobile devices, typically called *robots*, which would go around to restore energy to nodes with (near) depleted energy. The restoration can take place by either *recharging* the depleted batteries or by *replacing* them entirely with fully charged one.

2.3.3.1 Recharging by Mobile Robots

Fuelled by the recent evolution in wireless power transfer techniques [41], the research on sensor recharging by mobile robots has been quite intensive (e.g., see [7, 16, 46, 53, 68, 90, 93]).

Wireless power transfer by employing inductive coupling has been well-refined recently. It has been a popular technology to transfer power without wires due to its simplicity, convenience, and safety [92]. It can be easily integrated in WSN for extending sensor node lifetime due to its reasonable size [2]. However, it requires close contact as the charging efficiency starts to drop as distance increases [2, 92].

Magnetic resonance generates and transfers electrical energy between two resonant coils through varying or oscillating magnetic fields [50]. Compared to inductive coupling, magnetic resonant coupling can achieve higher transfer efficiency while extending the charging distance [97], therefore, can be employed in mid-range applications with a suitable efficiency [32]. This technology creates no interference with neighboring environments, and can be applied between one transmitting resonator and many receivers and enables concurrent charging of multiple devices[2].

Radio waves can be used to transfer energy. Besides longer transmission distance, this technology offers the advantage of compatibility with existing communications system, where energy and data can be transferred using the same technology at the same time [50]. This method does not require a line of sight between transmitter and receiver as power is radiated from the transmitting antenna in all directions. This also means that power can be transferred simultaneously to multiple devices [2]. Unfortunately, such technology for power transmission has low efficiency, and can be harmful to humans [50].

Laser can also be used for power transfer, by converting the power source into a laser beam which can be then converted to electricity using photo cells by the receiver [2]. Compared to previously mentioned technologies, laser can gain a high transfer efficiency for relatively long distance. However, laser beam requires an uninterrupted line of sight between two devices [2]. Furthermore, an intense beam can prove to be severely harmful for humans [50].

2.3.3.2 Battery Replacement by Mobile Robots

The alternative of replacement has been considered in the literature (e.g., see [59, 86]), albeit with less intensity. For a comparison between these two alternatives see [59]. The case of multiple robots has also been extensively studied (e.g., see [16, 46, 53, 78, 90]).

Notice that the idea of using mobile robots in sensor networks is not new, as it has been proposed for data gathering and aggregation, for network repairs, as well as for other network maintenance tasks (e.g., see [42, 43, 60]). Other examples of these approaches can be found in research papers in robotics (e.g., [9, 18]).

2.3.3.3 Restoration Strategies

Regardless of whether restoration is by recharging or replacement, after servicing a node, the robots must choose where to move next; this decision is crucial for the efficiency and the effectiveness of the resulting energy restoration strategy.

In most published work this decision is made *off-line*, thus in a totally centralized way. Fewer investigations have concentrated on making the decision *on-line*, where the data of the system arrives as events (e.g., battery becoming empty) occur: in other words, what happens next is not (usually) known a priori. These can be further distinguished between *centralized*, in which the information is continuously gathered at a single entity (e.g., the base station) where the decision is made; and *decentralized* (or *distributed*) where each robot makes its decision based solely on local information.

In the next sections we review the principal existing work. See the comprehensive study [50].

2.3.3.4 Offline

In [80, 93], the authors propose an off-line centralized strategy for a single robot; this strategy consist in computing the shortest Hamiltonian cycle of the sensor network; the robot then traverses the cycle charging each sensor nodes battery wirelessly. Once the robot visits all the sensor nodes, it returns to its service station to be serviced (e.g., replacing or recharging its battery). The goal is of maximizing the ratio of the robot's vacation time (i.e., the time spent by the robot, at the end of a cycle, to replacing or recharging its battery at its service station) over the cycle time. Since the optimal traveling path in each period is the shortest Hamiltonian cycle, the heuristic solution is near-optimal for both total cycle time and individual charging time; experimentally they analyze it in randomly generated networks consisting of $n = 50$ nodes, showing that they can remain operational forever.

The static routing of [80] has been extended in [79] to a dynamic multi-hop routing scheme for the robot to visit all the sensor nodes, leading to a non-linear optimization problem. Simulation results demonstrated that compared with static routing, the proposed strategy yields much larger objective value and incurs lower complexity.

The work of [94] considers multiple robots. They perform a off-line centralized computation aimed to minimize the sum of traveling distance of all q robots: the goal is to find q closed tours covering all locations such that the total length of the q tours is minimized. Due to the NP-hardness of this problem, the authors proposed a sequential approximation algorithm with a provable 2-approximation ratio under the assumption that the energy consumption rates of all devices are fixed. The basic idea is to find q-root trees with the minimum distance and to transform each tree into a closed tour with the length of each tour being not more than twice of the corresponding tree. Then, for the case with heterogeneous energy consumption rates, a sequential heuristic algorithm was developed. The simulation demonstrated the superiority of the proposed algorithm over a greedy baseline algorithm.

In [11] the authors investigated an offline scheme for scheduling the minimum number of mobile chargers to meet the recharging frequency of each sensor in *line* and *ring* networks. They assumed the chargers to have infinite battery capacity. Additionally, the proposed solutions were examined in a small network up to 10 devices. They first provided an optimal algorithm to determine the minimum number of chargers and corresponding dispatch planning when the charging frequency for all devices is identical. Then the authors designed a near-optimal greedy algorithm when charging frequencies

2.3.3.5 On-line Centralized

Since the decision can be made based solely on current and older information, the feasible strategies are necessarily on-line algorithms. Almost all existing on-line strategies are *centralized* (e.g., see [28, 34, 39, 45, 86, 103]): the information about all the requests is communicated to the robots that then compute where to go next; alternatively, the information is communicated to the base station, which takes the decisions and provides the mobile robots with instructions.

Centralized online energy restoration strategies can further be categorized into single robot and multi-robot strategies.

In the literature, several schemes have been proposed to prolong the lifetime of sensor networks using a centralized approach with a single robot. In these strategies, a node whose battery is (about to become) depleted issues a request for the robot; the robot moves to service the requests so to optimize some cost parameters, based on the information currently available.

The authors of [68] study centralized on-line strategies for a single robot in special topologies: grid and line. In this strategy the nodes estimate their remaining lifetime periodically and an aggregated report of the shortest lifetime nodes is delivered to the Sink which computes the strategy to be followed by the robot, The centralized computation uses heuristic algorithms with the aim to find optimal travel paths for the mobile charger. The simulation results show that the proposed routing strategy achieves longer network lifetime when the charging efficiency is low or the amount of energy carried by the mobile charger is small.

In [26] the authors introduce a centralized on-line tree-based schedule for a single robot. Their objective is to minimize the travel distance without causing robot energy depletion. Both analytical and simulation results indicated that the tree-based strategy approaches the optimal solution when the robot speed increases and when the battery lifetime increases.

In [28, 29] the focus is the centralized evaluation of the on-demand strategy based on the nearest-job-next with preemption. The studied metrics are throughput (i.e., number of requests the robot can serve in a given time period) and charging latency (i.e., the time since the request is sent by a node to the time it is fully charged. Their solution does not guarantee that all sensors will be charged prior to their energy expiration [46], and the use of preemption in [28] may result in a low efficiency of the scheme [47].

A centralized joint routing and charging scheme for a single robot has been proposed in [45]; it uses a charging-aware routing metric to estimate routing costs and make routing decisions, with the objective of prolonging the sensor network lifetime.

On-demand centralized schemes to maximize the network charging throughput per travel tour has been presented in [71]. The authors first propose the naive strategy of always serving the request with the smallest processing time (defined as the sum of traveling time and charging time). Then they introduce a cluster-based algorithm that groups the requesting sensors into different clusters according to their locations; the robot then evaluates the clusters using a newly defined metric and serves the best cluster according to that metric.

The work presented in [27] constructs a collection of nested Traveling Salesman tours of sensors with low residual energy. Energy synchronization is adopted to harmonize the charging sequence of sensors and the construction of travel tours is dynamically adjusted based on the request sequence to synchronize sensors in each charging round [46].

Other on-line centralized investigations for a single robot include [95], [23], [99], [6].

In the context of *multiple robots*, the computation of the shortest routes to be followed by the robots so to minimize their movements becomes a Multiple Traveling Salesmen Problem. In addition, the (periodic) coordination of the robots becomes an important problem. Such coordination can be achieved either in a fully centralized or partially centralized fashion [52], where a central entity performs such coordination using information from all chargers in the first approach [53] and each charger can be informed about the status of its neighboring chargers in the second one [52].

In addition to the high communication costs required, the optimization requirements to be met by the decision are typically accompanied by high computational complexity, which grows non-linearly with the number of sensor nodes; these factors imply a difficulty to scale for these strategies.

In [20] the authors compute in a centralized fashion an on-line efficient clustering of the nodes, based on their energy and time constraints with the objective of reducing the charging latency and the number of robots.

A centralized on-demand scheme for a set of robots with limited energy capacity has been proposed in [48], for networks of homogeneous sensors with identical energy capacity and energy consumption rates. A sensor send a charging request to the base station when the sensor's energy drops below a threshold; the base assigns mobile chargers to service regions. Each mobile charger decides on future movements, within its region, to maximize profits of the energy it provides, taking into account contribution degree of sensors (and potentially other chargers) and charging priority.

The authors in [52] addressed how multiple mobile chargers can periodically coordinate and partition the sensor nodes in a balanced manner, according to their energy and adapt to network energy consumption in order to improve energy efficiency, expand the lifetime of the network and also improve important network properties, such as the quality of network coverage and the robustness of data propagation. They presented a centralized variation where the coordination is performed using information from all chargers.

In [46] the authors study on-demand recharging focuing on the minimum number of robots so that no sensor runs out of energy. They study this problem introducing a tree decomposition; assuming that the energy consumption of each sensor during each charging tour is neglected, they propose a heuristic and evaluate the proposed algorithms using experiments by simulations.

The special case of *line networks* with multiple robots with limited energy capacity (the payload) has been examined in [102]. Their objective of maximizing the ratio of the amount of payload to total energy, as well as minimizing the number of robots; they propose a scheduling scheme which is optimal for homogeneous line network, and several greedy scheduling solutions for heterogeneous line networks.

2.3.3.6 On-line Decentralized

Few *decentralized* strategies have been proposed in the literature, and none of them based on an on-demand approach.

In [59] the charging robot collects information from nearby sensors and decides which one to serve in a on-line fashion. Variants for selecting the next sensor to be recharged are evaluated, among them: random selection and local greedy choices (sensor with minimum energy, or sensor with minimum expected life). Multiple chargers are employed, each making its own local selection. The concern is to maximize the time until the first interruption of the sensing activity of a sensor to achieve self-sustainability.

in [7], a tour is computed (both in a centralized manner and on-line) and the robot follows it blindly. The computation of the tour is either based on global knowledge of the sensors' status and position, or based on random walk. The authors also investigate adaptive strategies to adjust the tour depending on changes in the sensors' conditions.

Several authors considered the case when chargers have limited energy and thus also they need to be recharged, and studied the scenarios where chargers can transfer energy from one to another as well as recharging at a fixed charging location.

Collaborative robots' charging strategies have also been studied in [102, 101] with the objective to improve energy usage and charging coverage. Similar work has been done in [91] in order to achieve the maximum ratio of energy consumed for charging and for traveling. In [54, 55] the authors investigated the use of a hierarchical structure among the chargers. They proposed to partition the chargers into two groups: lower chargers, that are responsible for transferring energy only to sensor nodes, and the higher chargers, that are responsible for transferring energy to chargers. Among the metrics they focused on are coverage and the number of nodes with enough residual energy to operate during the progress of the experiment.

2.4 Relationship with our Approach

The main focus of the existing body of work on energy restoration using mobile robot(s) is on the selection of optimal tour(s) to be taken by the robot(s) so to efficiently perform the charging, where efficiency is measured in various ways but is typically related to the full operational maintenance of the network.

Optimization is sought in a centralized way and parameters to be minimized include the distance to be traveled by the robots, as well as the charging delay to be incurred by the sensor nodes. Another concern typically found in the literature is how to periodically replenish the robot's power so to minimize the disruption of the charging activity. Once a tour is established, the robot usually move along it in a decentralized way. All on-demand approaches found in the literature, on the other hand, are based on a centralized authority that calls the robot upon receiving requests.

In the thesis, our concerns are different as we wish to study a *fully decentralized on-demand* approach without being concerned with the choice of an optimal tour nor with the power of the robot. To do that, we assume a tour has been already chosen (we actually assume homogenous distance among the sensors in the tour), as well as unlimited charging power by the robot, and we devise an on-demand approach based on such conditions.

Shifting the attention from the optimization problems of determining a "best" circular tour and the "best" re-charging pattern for the robot, we can fully focus our attention on the online decentralized strategies themselves. We do so making the additional simplifying assumption of homogeneous sensors' batteries (all batteries have the same capacity and discharging rate).

Since we did not find any distributed on-demand strategy, we decided to compare our results with the ideal *optimal* strategy where the robot is assumed to have full knowledge of the status of the sensors at any given time and is therefore able to serve them directly as needed.

Chapter 3

The Model

Let $\mathcal{X} = \{x_0 \ldots, x_{n-1}\}$ be the set of *sensor nodes*, or simply *nodes*, forming the network. Each node has sensory equipment that allows it to monitor its surroundings; it also has provision for wireless communication. Normal operations require sensing and occasional communication; both operations consume energy, which is provided by an on-board battery of limited capacity.

Each node monitors the energy level of its battery and determines whether it is below a fixed restoration threshold, at which point energy restoration is allowed. After this point, when the battery is nearly (but not completely) depleted, the node becomes *inactive* and stops its sensing activity, thus creating a *sensing hole* in the network; once this occurs, the residual energy is used only for the communication necessary for restoring its energy.

In the thesis, we assume that all batteries have the same characteristics and, in particular, under normal operations they deplete at the same rate. More precisely, let Δ denote the amount of time it takes for a node with a fully charged battery to stop its sensing activity under normal operations. We denote by τ the amount of time, under normal operations, elapsed from the moment the battery falls below the restoration threshold to the time the node stops its sensing activity; hence $\hat{\Delta} = \Delta - \tau$ denotes the amount of time before a fully charged battery falls below the restoration threshold. In the following, for ease of discussion, we will sometimes refer to Δ as the battery life or capacity, and to τ as the threshold.

Let π denote a cyclic order of the nodes where successive nodes in the order (e.g., $x_{\pi(i)}$ and $x_{\pi(i+1)}$, where all operations on the indices are modulo n) are called neighbors, and can communicate (possibly by multiple hops). Figure 3.1 shows an example of one such a circular ordering.

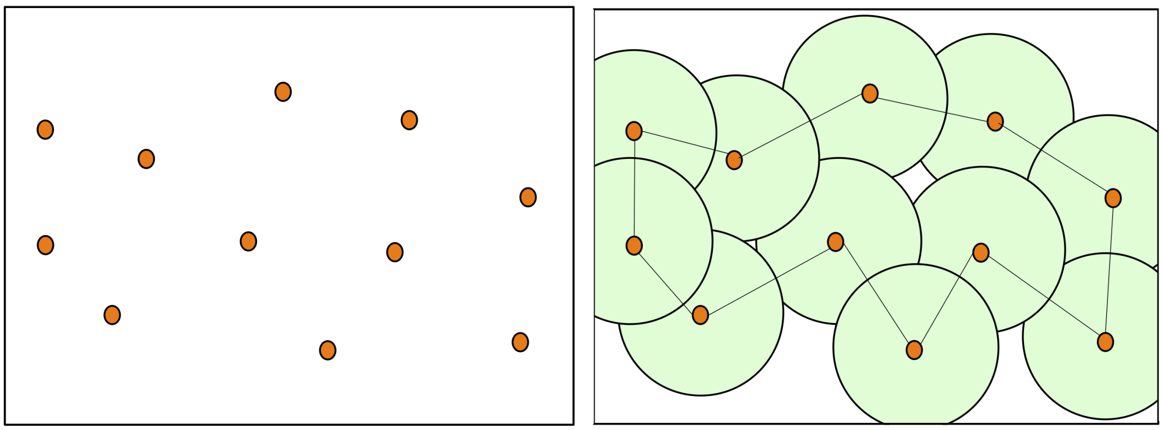


Figure 3.1: *a*) A deployed sensor network. *b*) A cyclic order π of the sensors.

A mobile *robot* R, which we assume having unlimited charging power, is available in the system to restore energy to the sensors. Once R reaches a node, if the energy level of the node is below the threshold, the robot (also called "charger") will restore the energy of that node.

In case of energy restoration by recharging, the charging time depends on the sensor's remaining energy. More precisely, the time $\gamma(\ell)$ it takes for recharging depends on the time $\ell \leq \tau$ left before sensing is stopped, and is given by $\gamma(\ell) = \frac{(\Delta-\ell)\rho}{\Delta}$ where ρ, called *charging time*, denotes the amount of time it takes for an empty battery to become recharged (i.e., when $\ell = 0$). In case of energy restoration by replacement, the time required is the same regardless of the level i.e., $\gamma(x) = \rho$ for all $\ell \leq \tau$. A node resumes its sensing activities as soon as the robot starts restoring its energy.

The robot can move from node to node; we denote by $d_{i,j}$ the time it takes the robot to travel from node x_i to x_j. Sometimes, we will improperly call such a value, the "distance" between x_i and x_j.

After servicing a node, the robot must choose where to move next. The algorithm followed by the robots to make this decision is called *energy restoration strategy*; it may prescribe the acquisition of information from the sensors (e.g., energy level, location, etc) and require computations by the robot.

The goal of an energy restoration strategy is to maintain as much as possible of the network operational at any time. Its *effectiveness* towards this goal is measured in terms of two parameters: the number of sensors it is able to maintain operational at any given time, in spite of battery depletions; and how long sensing holes last. More precisely the effectiveness of an energy restoration strategy is evaluated in terms of:

1. The *Operational Size* at time t (denoted by $Coverage(t)$) is the number of operational nodes (i.e., nodes with active sensing) at that time; with an abuse of notation, we will refer to this measure *Coverage*. Note that the coverage size implicitly measures the number $Holes(t) = (n - Coverage(t))$ of sensing holes in the network at time t.

2. The *Disconnection Time* for a node x is the time from the moment x becomes inactive to the time when the charger arrives to serve x. Disconnection time is, of course, zero if the node is charged before the battery is empty (i.e., before it becomes inactive). This measures indicates for *how long a coverage hole lasts*.

We will sometimes denote by $Coverage(\texttt{Strategy}, t)$ the coverage size of some strategy Strategy at time t, and by $Disconnect(\texttt{Strategy}, t, x)$ the disconnection time of node x at time t using Strategy.

Another interesting parameter is the *Charging Latency* for a node x, defined as the time since the energy of x drops below the threshold to the time the charger reaches it.

Chapter 4

On-Demand Strategy

In this Chapter, we introduce a decentralized on-demand charging strategy. We then study its effectiveness both theoretically and experimentally.

4.1 The Strategy

We now describe a simple decentralized on-demand strategy (On-Demand), which uses only local information and requires only local communication.

Starting from its initial position at an arbitrary node, the robot visits the sensors according to a circular order π, moving in a "counter-clockwise" direction (i.e., from $x_{\pi(i)}$ to $x_{\pi(i+1)}$) when aware of a pending request. A sensor whose battery is about to become depleted originates a recharging request and waits for the robot; the request is forwarded in a "clockwise" direction (i.e., from $x_{\pi(i)}$ to $x_{\pi(i-1)}$) until it reaches either the robot or another sensor waiting for the robot, creating in this way a trail to be followed by the robot when it becomes available. Note that a request contains no specific information (e.g., id, location, etc) about the node issuing or forwarding it. All nodes have the ability to receive and forward a single request even if their sensor is no longer operational.

Summarizing, (1) each sensor communicates only locally: with the neighboring sensors in the cyclic order (to send or receive a request), or with the robot if currently there (to communicate the presence of a pending request); (2) the robot moves only from one node to the next in the cyclic order, is aware only of whether or not there is a pending request, and has no need of additional memory or calculation.

Algorithm ON-DEMAND prescribing the behavior of the sensor nodes and the robot is shown below. With respect to the protocol, a sensor node x can be in one of two protocol states, REGULAR or WAITING, and keeps track of whether or not it has received a pending request from its predecessor in the order (Boolean variable $Q(x)$). Initially all nodes are in state REGULAR, $Q(x) = 0$ for all $x \in \mathcal{X}$, and the robot is at an arbitrary node.

Algorithm ON-DEMAND **for node** x**.**

<u>REGULAR</u>

when battery level reaches threshold

 if robot is not here **then**

  send request **to** clockwise neighbor;

 become WAITING;

receiving request

 $Q(x) := 1$;

 send request **to** clockwise neighbor;

 become WAITING

<u>WAITING</u>

receiving request:

 $Q(x) := 1$;

receiving Robot: (robot is here)

if battery level at or below threshold

`Be Charged`;

if $Q(x) = 1$ **then**

send Robot **to** counter-clockwise neighbor;

$Q(x) := 0$;

become REGULAR;

4.2 Theoretical Analysis

In the following, we make the simplifying assumption that the charging time ρ is constant (as when the batteries where replaced rather than charged).

4.2.1 Properties: Tours and Weakness

A *tour* from node x is defined as the visit of all the nodes by the robot starting from and ending in x. Let $\hat{\Delta} = \Delta - \tau$ denote the amount of time before a fully charged battery falls below the threshold.

Lemma 1. *Let $x_{\pi(i)}$ require recharging both at the beginning and at the end of a tour from it; then also $x_{\pi(i-1)}$ requires recharging when reached by the robot in this tour.*

Proof. Let $x_{\pi(i)}$ be found to be needing recharging at time t_0, fully recharged at time $t_1 = t_0 + \rho$, and found again needing recharging at the end of this tour, at time t_2.

By contradiction, let the previous node $x_{\pi(i-1)}$ be found not needing recharging when visited by the robot in this tour. Let $k \geq 0$ be the number of the nodes (other than $x_{\pi(i)}$) that have been recharged in this tour. In other words, the robot has spent in this tour $k\rho$

time units to charge them before reaching x_0 at time t_2; that is, $t_2 \geq t_1 + k\rho + nd$. Since $x_{\pi(i)}$ is found needing recharging at time t_2, we must have

$$k\rho + nd > \Delta - \tau = \hat{\Delta}.$$

On the other hand, the time elapsed between the time $t' = t_0 - d$ the robot left $x_{\pi(i-1)}$ and the time t'' it reached it again is at least $(k+1)\rho + nd$, since $x_{\pi(i)}$ was recharged in this interval. Since $x_{\pi(i-1)}$ is assumed to be found non needing recharging, we must have

$$(k+1)\rho + nd \leq \hat{\Delta}$$

a contradiction. □

Analogously,

Lemma 2. *Let $x_{\pi(i)}$ require recharging both at the beginning and at the end of a tour from it; then also $x_{\pi(i+1)}$ will need to be recharged when when reached by the robot in the next tour from $x_{\pi(i)}$.*

We say that x is *weak* if there exists a tour from x where x needs recharging both at the beginning and at the end of the tour. The two Lemmas, 1 and 2, together prove the following important property:

Theorem 1. *If there is a weak node, then within finite time every node visited by the robot is found to need recharging. This holds regardless of the threshold τ.*

This, in turns, has important consequences on the size of the coverage of the network:

Theorem 2. *Let $n > m = \lceil \frac{\Delta}{(\rho+d)} \rceil$. If there is a weak node, then there exists a time t, such that, $\forall t' > t$:*

$$m \leq Coverage(\mathtt{On-Demand}, t') \leq m + 1$$

and all sensors have the same disconnection time:

$$\forall x \in \mathcal{X}, Disconnect(\texttt{On} - \texttt{Demand}, t', x) = (n-1)(\rho + d) + d - \Delta.$$

Proof. Let $x_{\pi(i)}$ be weak; thus, there is a tour where $x_{\pi(i)}$ will be recharged both at the start and the end of that tour; let t be the time when the recharging at the end of that tour will be completed. By Theorem 1, from this time on the robot finds only nodes needing recharging; since it takes ρ time units for the robot to recharge a sensor and d time units to move to the next sensor, by time $t' = t + \Delta$ the robot has recharged the consecutive nodes $x_{\pi(i+1)}, x_{\pi(i+2)}, ..., x_{\pi(i+m-1)}$ where $m = \lceil \frac{\Delta}{(\rho+d)} \rceil$ and all operations on the indices are modulo n; if Δ is not a multiple of $\rho + d$, then it is currently recharging $x_{\pi(i+m)}$, otherwise also $x_{\pi(i+m)}$ has been fully recharged. But, at this time $t' = t + \Delta$, $x_{\pi(i)}$'s battery is totally depleted, and so obviously is the battery of all the sensors after $x_{\pi(i+m)}$ up to and including $x_{\pi(i)}$. This means that, at time $t_0 = t + m(\rho + d)$, exactly $n - m$ nodes have their battery completely empty; hence $Coverage(\texttt{On} - \texttt{Demand}, t_0) = m$.

Observe that, when $x_{\pi(i+m+1)}$ is reached and fully recharged, at time $t_1 = t_0 + (\rho + d)$, $x_{\pi(i+1)}$'s battery is depleted; that is $Coverage(\texttt{On} - \texttt{Demand}, t_1) = m$. More generally, when $x_{\pi(i+m+j)}$ is reached and fully recharged, at time $t_j = t_0 + j(\rho + d)$, $x_{\pi(i+j)}$'s battery is depleted; that is, $Coverage(\texttt{On} - \texttt{Demand}, t_j) = m$. On the other hand, at any time $t_j < t' < t_{j+1}$ $x_{\pi(i+j)}$'s battery might not yet be depleted; that is, $Coverage(\texttt{On} - \texttt{Demand}, t') \leq m+1$.

Since after time t the robot keeps charging every node it encounters, it will spend $(n-1)(\rho+d)+d$ time units to complete a round; hence, every node will have disconnection time $(n-1)(\rho + d) + d - \Delta$. □

4.2.2 Properties: Rounds and Stability

Let us call *round* from node x any sequence of consecutive tours from x where at the beginning of the first tour and at the end of the last x's battery needs to be recharged, and in all others it does not. Clearly a round from x might include several tours from x.

We will denote by $r(x, j)$ the j-th round from x, $j \geq 1$; by $t_s(x, j)$ and $t_e(x, j)$ the starting and ending time of $r(x, j)$, respectively. When j is clear from the context, we will indicate a round from x simply by $r(x)$ and the corresponding starting and ending time by $t_s(x)$ and $t_e(x)$ respectively.

Let $\sigma(x, j)$ denote the ordered sequence of the nodes charged during $r(x, j)$; notice that a node may appear more than once in $\sigma(x, j)$ while some be absent.

We say that the system is *stable* if $\exists j \geq 1$ such that $\forall x \in \mathcal{X}, \forall j' > j$, $\sigma(x, j) = \sigma(x, j')$. That is, in a stable system, the order in which the nodes are charged is the same in every round; hence, in a stable system, we can omit the indication of the round and denote $\sigma(x, j)$ simply as $\sigma(x)$.

When a system is (or has become) stable, it enjoys particular properties. Among them:

Lemma 3. *Let the system be stable. Then*

(i) $\sigma(x)$ is a permutation of the elements of $\mathcal{X}$.

(ii) $\forall x, y \in \mathcal{X}$, $\sigma(x)$ is a cyclic shift of $\sigma(y)$

(iii) Every round from any node $x \in \mathcal{X}$, is composed of the same number s of tours.

Theorem 3. *If the system is stable and $n > 2\frac{\Delta}{\rho} + 1$, each round is composed by a single tour.*

Proof. Let the system be stable; then, by Lemma 3 (iii), every round from any node x is composed of the same number s of tours. We want to show that $s = 1$ if $n > 2\frac{\Delta}{\rho} + 1$.

By contradiction, let $s > 1$. Consider a round from node x and let $f(x, s)$ be the number of nodes charged in the last tour of this round. We will consider three cases depending on the value of $f(x, s)$.

Case 1 : $f(x, s) < \frac{n}{2}$. In this case, the number of nodes charged in the first $s - 1$ tours is $k = n - f(x, s) \geq \frac{n}{2} + 1$.

Consider the amount of time T elapsed from the moment x has been charged at the beginning of the round, to the beginning of the last tour of this round.

By definition, and since $k \geq \frac{n}{2}+1$ and $s > 1$, we have

$$T = (k-1)\rho + (s-1)d\ n \geq \frac{1}{2}n\rho + (s-1)d\ n \geq \frac{1}{2}n\rho + d\ n.$$

Since, by definition of round, x is found by the robot not to need recharging at this time, we have $\Delta - \tau = \hat{\Delta} \geq T$, that is:

$$\Delta \geq \hat{\Delta} \geq T = \frac{1}{2}n\ \rho + d\ n > \frac{1}{2}n\ (\rho+1)\ .$$

But $n > 2\frac{\Delta}{\rho} + 1$ by hypothesis; that is, $\frac{1}{2}(n-1)\rho > \Delta$; a contradiction.

Case 2 : $f(x,s) > \frac{n}{2}$. In this case, there must exist a node y such that $f(y,s) < \frac{n}{2}$. By considering the round from y (instead than from x), by Case 1 the contradiction occurs.

Case 3 : $f(x,s) = \frac{n}{2}$. If $s > 2$ then there must exist a node y such that $f(y,s) < \frac{n}{2}$; by considering the round from y (instead than from x), by Case 1 the contradiction occurs.

Finally, let $s = 2$. In this case, n is even, the round $r(x)$ is composed of two tours, and $f(x,1) = f(x,2) = \frac{n}{2}$. Note that the nodes charged in the second tour of $r(x)$ are the complement of the ones charged in the first tour.

Consider now the node y, next in the cycle, visited by the charger right after x. We have two cases (see Figure 4.1) depending on whether or not y needs to be recharged at this time. Case $(3a)$: y needs to be charged. In this case, consider the round $r(y)$ starting after charging y (see Figure 4.1 (3a), bottom). By definition, we must have $f(y,1) = f(y,2) = \frac{n}{2}$. However, the nodes in need of charge in the first tour of $r(y)$ are the same as the ones in the first tour of $r(x)$ except for y itself; thus $f(y,1) = \frac{n}{2} - 1$, a contradiction. Case $(3b)$: y does not need to be charged. In this case, consider the round $r(y)$ starting from y the last time it was charged before time $t_s(x)$. By definition, the nodes in need of charge in the first tour of $r(y)$ are the complement of the ones charged in the first tour of $r(x, 1)$ excluding y, thus $f(y, 1) = \frac{n}{2} - 1$, a contradiction. □

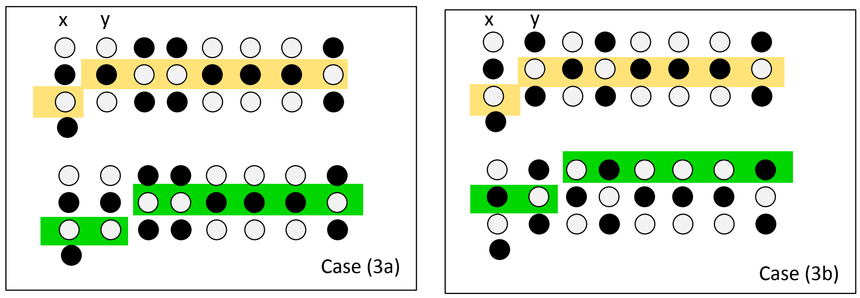


Figure 4.1: Cases (3a) and (3b) of Theorem 3 in a stable system with $n = 8$. Each row corresponds to a tour starting from x; black (white) circles represent sensors charged (not charged) in that tour. A round $r(x)$ consists of two consecutive tours starting after a black x. Highlighted in the top (resp. bottom) is the first tour of $r(x)$ (resp. $r(y)$) in the two cases.

By bringing the observations on weakness and stability together, we have:

Theorem 4. *Let the system be stable and let $n > \frac{2\Delta}{\rho} + 1$. Then there exists a time t, such that, for all $t' > t$ we have*

$$m \leq Coverage(\texttt{On} - \texttt{Demand}, t') \leq m + 1,$$

where $m = \lceil \frac{\Delta}{(\rho+d)} \rceil$; moreover, for all $x \in \mathcal{X}$

$$Disconnect(\texttt{On} - \texttt{Demand}, t', x) = (n-1)(\rho + d) + d - \Delta.$$

All the analytical results we have established on the effectiveness hold once the network has become stable.

It is not difficult to prove that all networks, regardless of their size, become stable when starting from specific initial battery levels. This is for example the case when all nodes start with a fully charged battery, or when initially all batteries are empty:

Theorem 5. *Let $\delta(x)$ denote the amount of time elapsing before the battery of node x*

reaches the threshold for the first time. If $\delta(x) = \delta(y)$ *for all* $x, y \in \mathcal{X}$*, then the network become stable after one round.*

This is also true if the initial battery levels are different but increasing with respect to the cyclic order starting from the initial position of the robot:

Theorem 6. *Let initially the robot be at node* $x_{\pi(i)}$*. If* $\delta(x_j) \leq \delta(x_{j+1})$ *for all* $i \leq j \leq i + n - 1$*, then the network becomes stable after one round.*

We conjecture that, in all networks with $n > \frac{2\Delta}{\rho} + 1$, stability is inevitably achieved; furthermore this occurs with a small number of rounds.

Conjecture. Let $\frac{2\Delta}{\rho} + 1$; then the system becomes stable within a constant number of rounds.

As we will see, this conjecture is supported by the strong experimental evidence.

4.3 Effectiveness Analysis

The results of the previous section describe the behavior of the system once it stabilizes in time when the restoration time ρ is constant (as if the batteries where replaced rather than charged). In this section, we run extensive simulation to determine the stability of the system and to observe its behavior in terms of Coverage Size and Disconnection Time (already studied theoretically), as well as Charging Latency. The experimental results are performed in the more general case, when the restoration time ρ might depend on the current level of energy. Interestingly, the theoretical and experimental results match quite closely.

4.3.1 Simulation Environment

The experiments were implemented in the simulator discrete-event MAS toolkit MASON [51].

The variable parameters involved in the experiments are: the number of nodes n, the battery life Δ, the charging time ρ, and the travel time d from a node to the next.

The following table shows the values considered for each parameter:

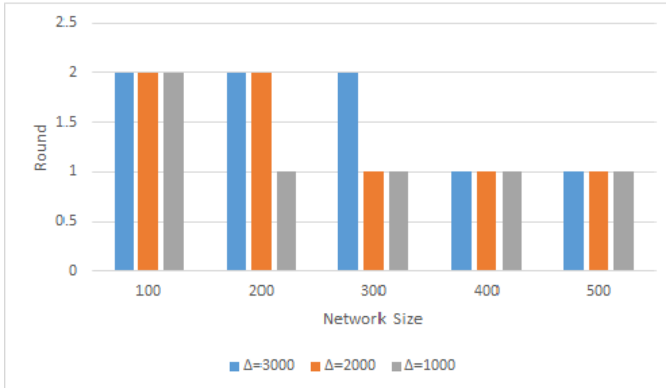
Parameters	*Values*
Number of nodes n	100, 200, 300, 400, 500
Battery Lifetime Δ	2000, 3000, 4000
Threshold τ	0%,10%,20%,30% of Δ
Charging Time ρ	1, 10, 20, 30, 40, 50
Travel Time d	1, 5, 10, 20, 30

Each sensor x has initially an amount of energy level chosen uniformly at random in the range $[\tau, \Delta]$; the robot's initial placement is at a node chosen uniformly at random.

The average value of the parameters under consideration has been taken over 10 different executions. All results are obtained with a 98 % level of confidence.

4.3.2 Stability

Starting with arbitrary initial charge levels, the experimental results show that, for all values of the parameters, the charging order becomes periodic and the system stable; moreover stabilization occurs within two rounds. In other words, *under* ON-DEMAND *the network becomes stable within two rounds.* See Figure 4.2 where stability is shown for some choices of the parameters; the results for all the other parameters' combinations are consistent with these. Note that, for smaller capacities, as well as for larger networks, stabilization occurs even sooner, within one round.



(a) stabilization

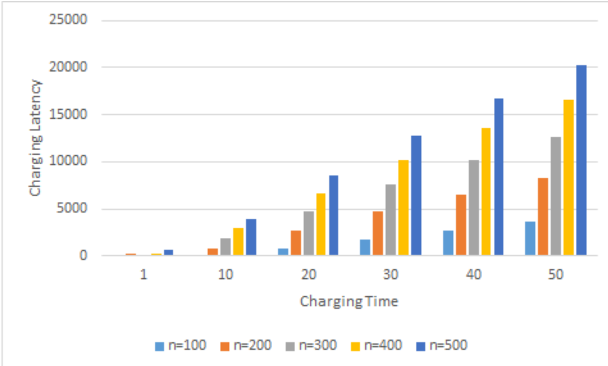
Figure 4.2: stabilization n=300, d=1

Once the system stabilizes, the theoretical bounds on coverage size and disconnection time established analytically (Theorem 4) hold; indeed, the simulation results of the next section confirm all these bounds.

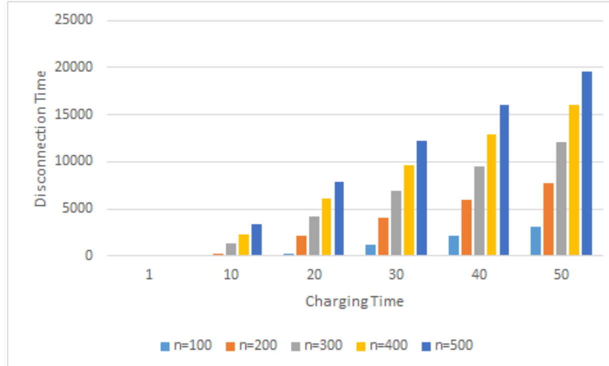
4.3.3 Charging Latency and Disconnection Time

We now study the impact of varying the various network parameters on Charging Latency and Disconnection Time. The simulations show expected results without revealing any surprises. Generally, Charging Latency and Disconnection Time increase with the increase of the number of nodes, of charging time and of travel time between successive sensors, while they decrease with the increase of capacity.

Effects of Charging Time ρ. In order to determine the impact of the charging time ρ on the effectiveness of the proposed algorithm, we start by analyzing the effects of various charging time on charging latency and disconnection time for different network sizes using $n = 100, 200, 300, 400, 500$ nodes. In these scenarios, we fix the node capacity $\Delta = 2000$ and we consider unitary travel time between neighboring nodes (the case of $d \neq 1$ will be treated separately). The results obtained varying Δ are consistent with the ones displayed here and are not shown. In all cases the results are as one would have expected.



(a) Effect of Charging Time on Charging Latency



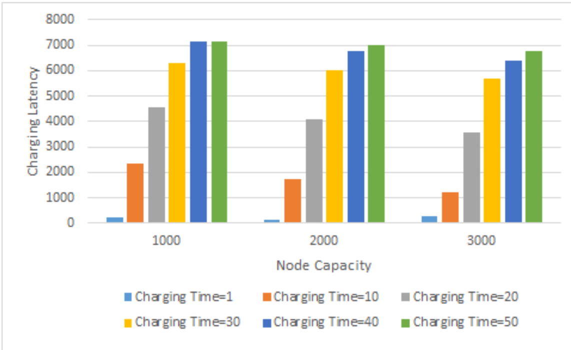
(b) Effect of Charging Time on Disconnection Time

Figure 4.3: Effect of Charging Time ρ; Δ=2000, d=1

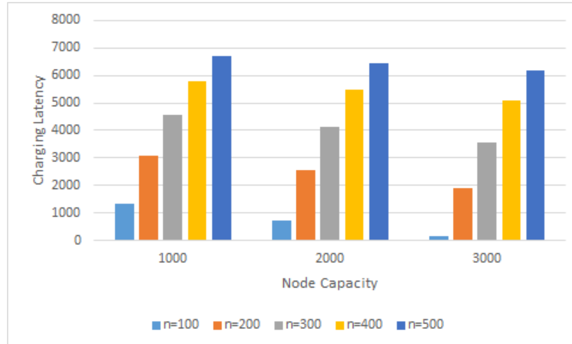
The graph in Figure 4.3(a) shows the relationship between charging latency and charging time. As expected, for a given network size, the charging latency increases as the charging time increases. As the size of the system gets larger, the number of requested nodes increases and so charging latency increases too.

The graphs in Figure 4.3(b) shows the trend in the disconnection time for the considered network sizes. In general, we can see a similar pattern in the graph: for a given number of nodes, the disconnection time increases as the charging time increases because the waiting time for the robot increases.

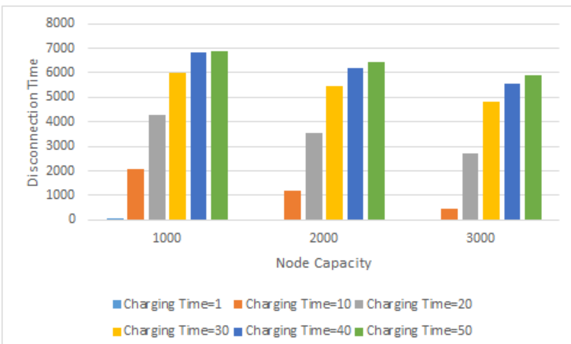
Effects of Node Capacity Δ. We now analyze the effect of node capacity Δ. In the graphs below we show the impact of Δ by fixing n and varying ρ, as well as by fixing ρ and varying n. Also here, the travel time between sensors is considered unitary (the case of $d \neq 1$ will be treated separately). Figure 4.4 shows the impact on charging latency and disconnection Time (a), (b): when $n = 300$ nodes while ρ varies, and (c),(d) when $\rho = 20$ and n varies. The results reveal no surprises: for a given network size and charging time, the charging latency and the disconnection time decrease as the node capacity increases.



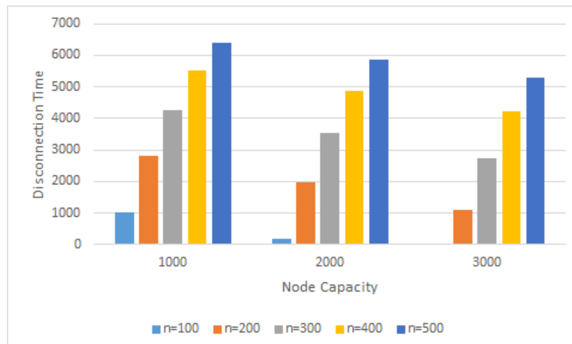
(a) Effect of Node Capacity on Charging Latency; n= 300



(b) Effect of Node Capacity on Charging Latency.ρ=20



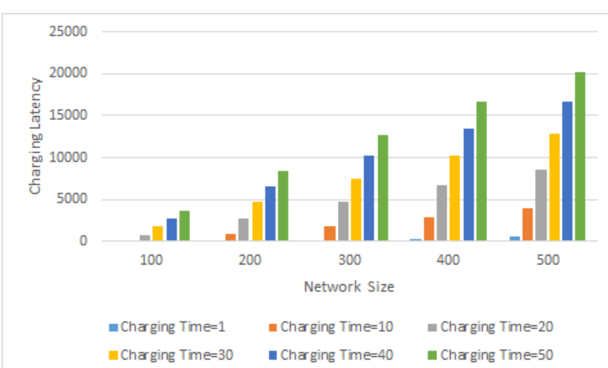
(c) Effect of Node Capacity on Disconnection Time. n= 300



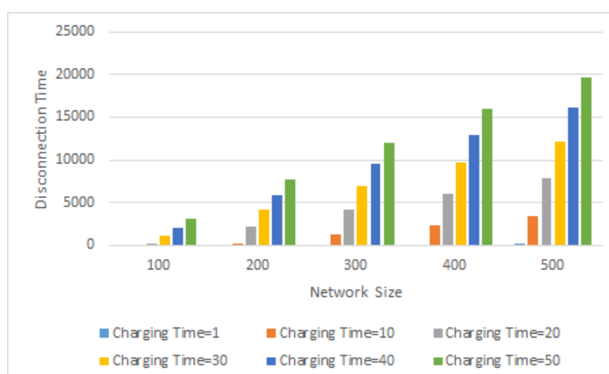
(d) Effect of Node Capacity on Disconnection Time.ρ=20

Figure 4.4: Effect of Node Capacity Δ on Charging Latency and Disconnection Time

Effects of Network Size n**.** We now observe the effect of network size on charging latency and disconnection time, fixing the node capacity $\Delta = 2000$ and travel time $d = 1$, and varying the value of the charging time ρ. As Figures 4.5(a) and 4.5(b) show, charging



(a) Effect of Network Size on Charging Latency

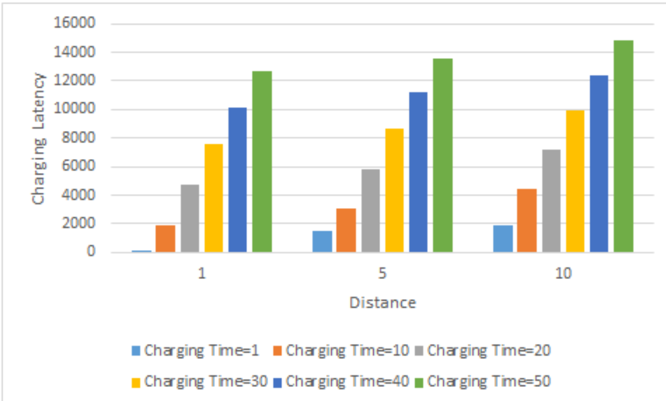


(b) Effect of Network Size on Disconnection Time

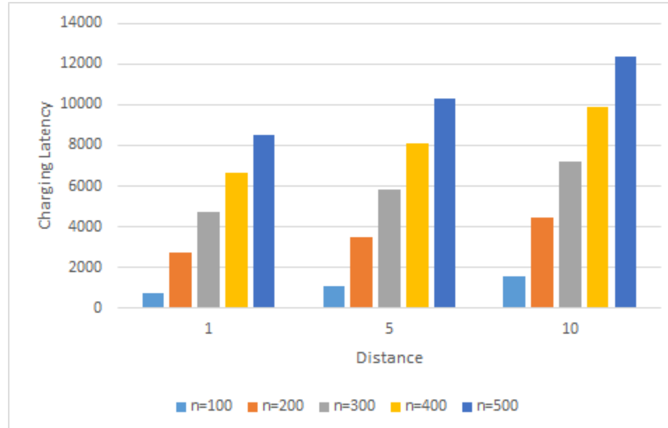
Figure 4.5: Effect of Network Size. Δ=2000, d=1

latency and disconnection time increase, as expected, while the network size increases, and they do so also when varying ρ.

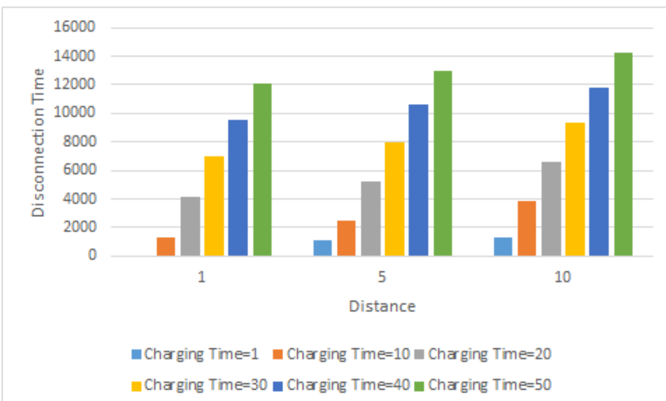
Effects of Travel Time. We now study the impact of the time it takes the robot to go from a node to the next in the cyclic order on charging latency and disconnection time. In the experiments, we set node capacity $\Delta = 2000$ unit of energy, and we vary the travel time between nodes to be uniform, but not necessarily unitary: we consider $d = 1, 5, 10$ units. We consider various network sizes $n = 100, 200, 300, 400, 500$, and various charging latencies $\rho = 1, 10, 20, 30, 40, 50$. Figures 4.6(a) and 4.6(b) show the trend of charging latency with different travel times, whereas figures 4.6(c) and 4.6(d) depict the effect of the travel time on the disconnection time. We conclude that as the travel time between nodes increases, charging latency and the disconnection time increase too. In general, most nodes get charged before their battery becomes empty, which means a lower disconnecting time. As the number of inactive nodes increases due to the increase in travel time, the disconnecting time increases.



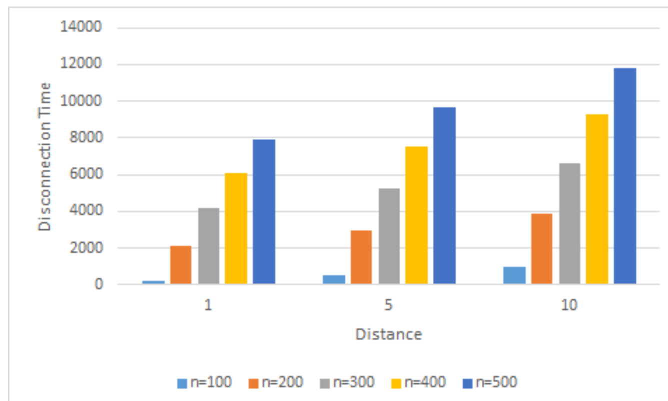
(a) Effect of Travel time on Charging Latency.n=300



(b) Effect of Travel time on Charging Latency.ρ=20



(c) Effect of Travel time on Disconnection Time.n=300



(d) Effect of Travel time on Disconnection Time.ρ=20

Figure 4.6: Effect of Travel time on Charging Latency and Disconnection Time. Δ=2000

4.3.4 Coverage Size

While the impact on disconnection time and charging latencies are as expected, the observation of coverage size has revealed much more interesting behaviors.

Effects of Charging Time ρ. From Figure 4.7 we observe that when $\rho = 1$ the increase in size corresponds to an increase of coverage size up to $n = 400$, at which point the coverage size seems to stabilize to a constant value. The same behavior can be observed for larger ρ, where however the stabilization occurs earlier; in fact, for $\rho \geq 10$ we observe it when $n = 200$. This behavior is quite interesting and will be discussed further later. From

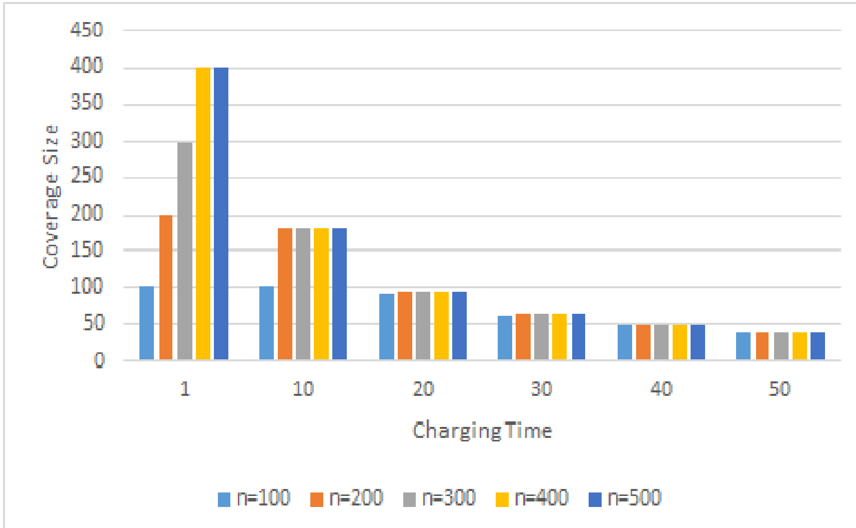
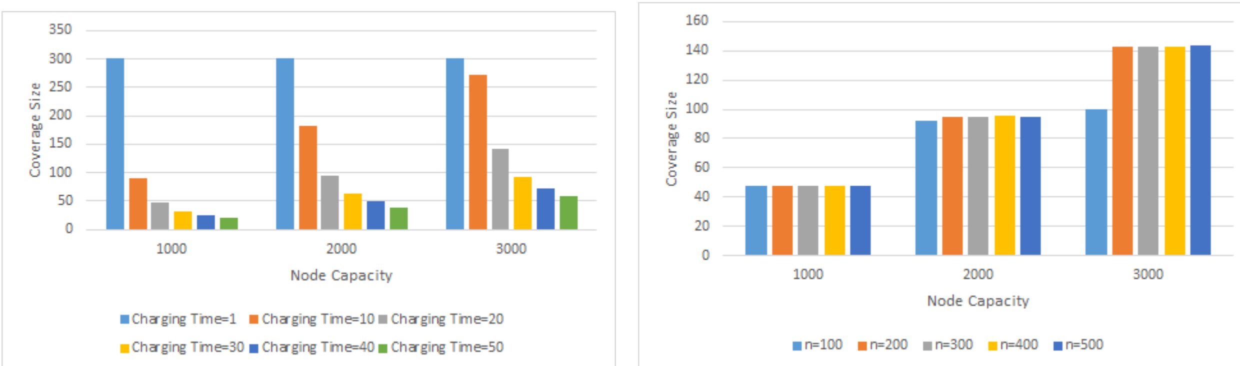


Figure 4.7: Effect of Charging Time ρ; Δ=2000, d=1

the graph, we can also see that coverage size decreases varying ρ, when observing a fixed value of n, which is to be expected.

Effects of Node Capacity Δ. Figure 4.8 (a) shows that, as one would expect, when node capacity increases, also the coverage size increases (when n is fixed). On the other hand, fixing a given capacity, the increase in the network size correspond to a decrease of coverage size. Interestingly, from Figure 4.8 (b) we observe again that the coverage size does not change when fixing a given node capacity, even when n changes. In other words, the node capacity is independent of the network size and confirms the theoretical derived in Theorem 4, even for n smaller that $\frac{2\Delta}{\rho} + 1$.

Effects of Network Size n. The observation just made can be clearly seen also in Figure 4.9, where we note that the coverage size increases with the network size but eventually stabilizes. The stabilization occurs earlier for smaller values of charging time ρ still confirming the theoretical bounds established in the previous Section.

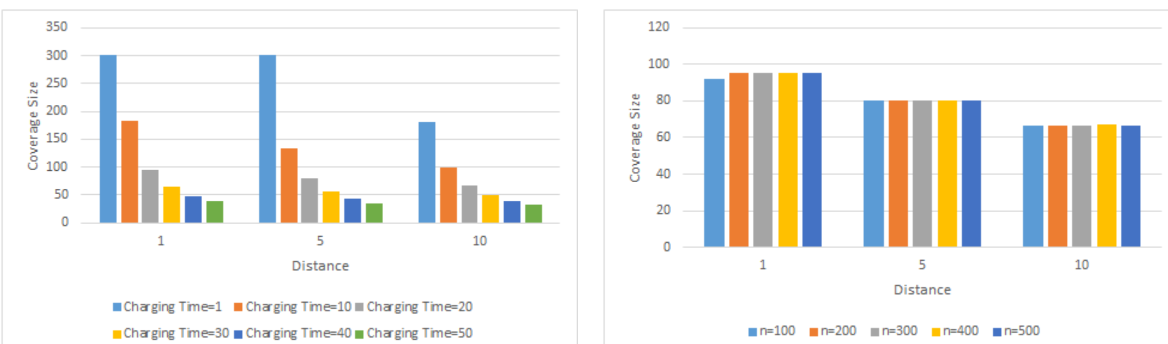


(a) Effect of Node Capacity on Coverage Size.n=300

(b) Effect of Node Capacity on Coverage Size.ρ=20

Figure 4.8: Effect of Node Capacity on Coverage Size

Effects of Travel Time. When the travel time between two nodes is not unitary, but still constant, we make observations that are consistent with what already seen (see Figure 4.10), which makes us conclude that, as long as the travel time between two nodes is the same, the actual value does not influence the general behavior of the strategy.



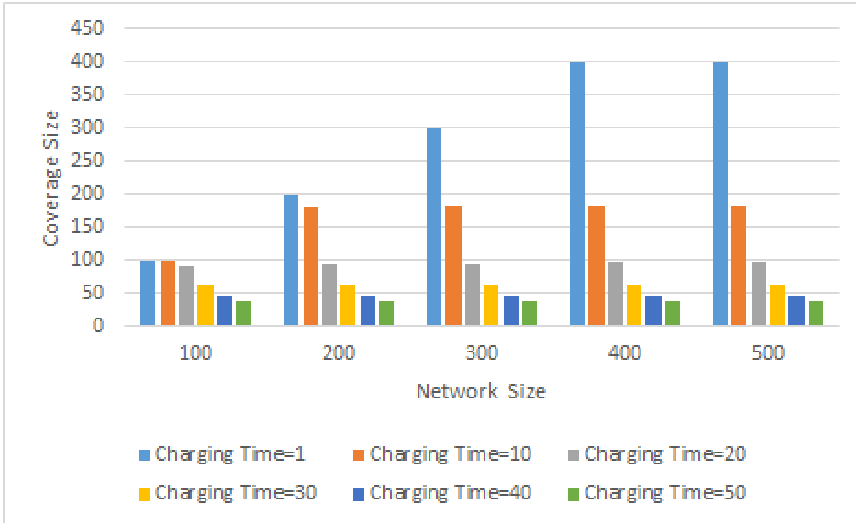
(b) Effect of Travel Time on Coverage Size.n=300

(c) Effect of Travel Time on Coverage Size.ρ=20

Figure 4.10: Effect of Travel Time on Coverage Size

More detailed analysis. To better analyses the observed phenomena concerning coverage size, we now focus on some specific scenarios to analyze how the coverage size changes in time.

Starting with arbitrary initial charge levels, the experimental results show that the first few tours of the ring performed by the robot follow a schedule that highly depends on the



(a) Effect of Network Size on Coverage Size

Figure 4.9: Effect of Network Size on Coverage Size. Δ=2000, d=1

initial distribution of charges; however, we confirm our conjecture by observing that, for all choices of parameters:

- the charging pattern stabilizes becoming periodic;
- stabilization occurs within two rounds.

In other words, the networks become stable within two rounds. Figure 4.7, for example, shows coverage size varying ρ confirming that whenever we have $n > \frac{2\Delta}{\rho} + 1$, the average coverage size stabilizes, that coincides with the theoretical value shown in Theorem 4. The same is observed when varying the node capacity Δ.

Figures 4.11 and 4.12 also display a clear evidence of stabilization by showing coverage in time for different system sizes under different choices of Δ and ρ. When executing each scenario, computing coverage size and disconnection time, we have also recorded the charging order of the nodes. We have seen that, after at most two rounds, regardless of the size, the system stabilizes. Stability occurs for any value of n, we then can observe that for $n > \frac{2\Delta}{\rho} + 1$ (i.e., $n = 200, 300, 400, 500$ in Figure 4.11 and all values of n in Figure 4.12),

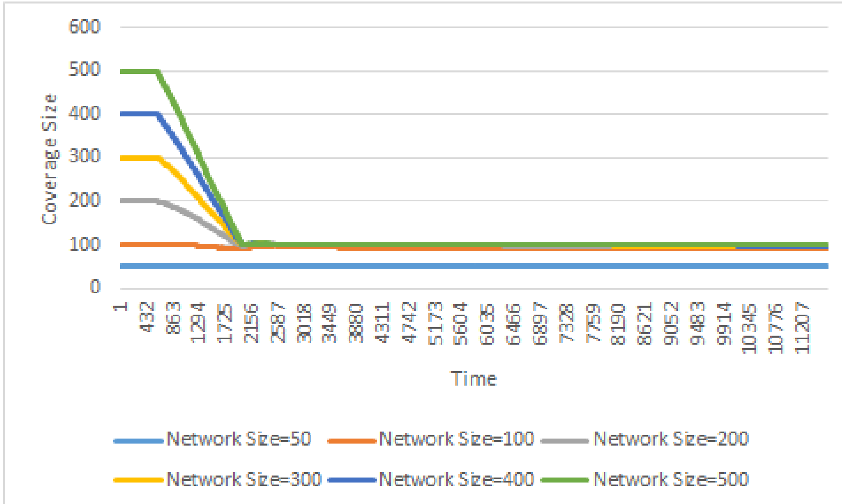


Figure 4.11: Coverage Size for Δ=2000,d=1,ρ=20.

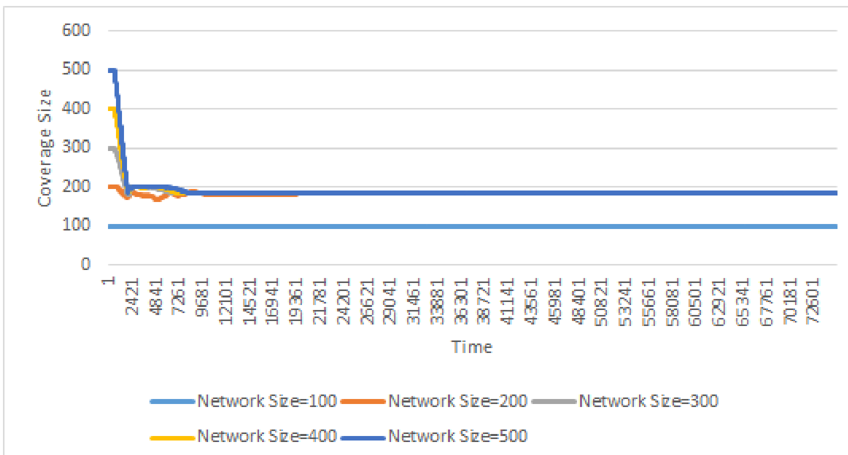


Figure 4.12: Coverage Size for Δ=2000,d=1,ρ=10.

coverage does not depend on n and coincides with the theoretical value which is shown in the next section.

Finally, we observe that these phenomena hold in all experiments regardless of the initial distribution of charges; that is, the same phenomenon has been observed not only with random distributions, but also with specific ones, such as when the charges are increasing in a clockwise order (see Figure 4.13) or in counterclockwise order (see Figure 4.14).

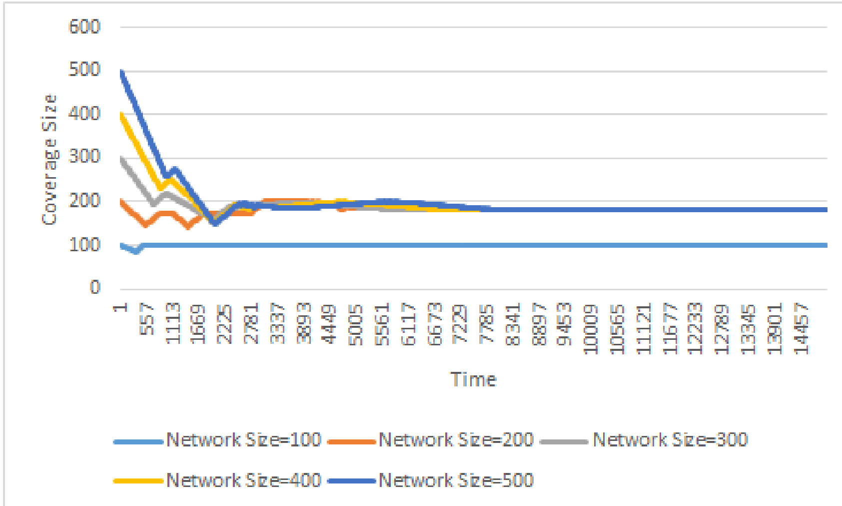


Figure 4.13: Increasing initial charges

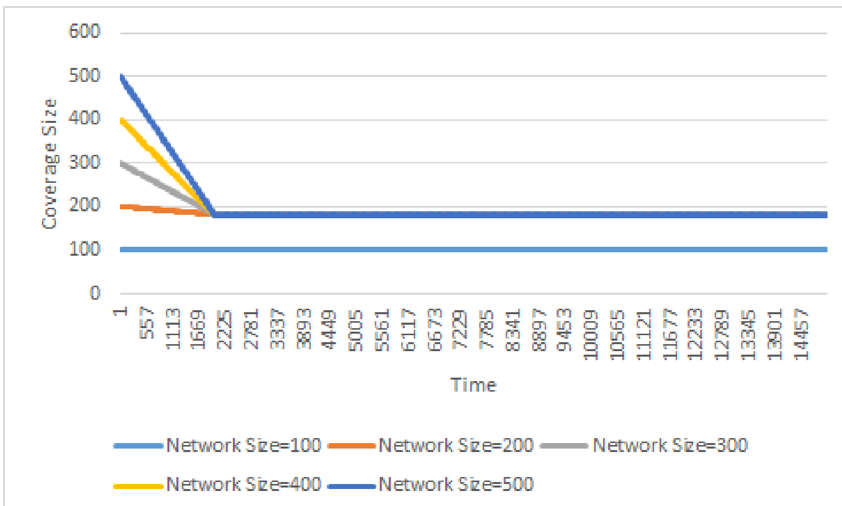


Figure 4.14: Decreasing initial charges

4.4 Summary

In this Chapter, we introduced an on-demand charging strategy for sensor networks. We evaluated disconnection time, and coverage both theoretically and through experiments. We observed both theoretically and experimentally rather expected results concerning disconnection time when varying the other network parameters. We instead made interesting discoveries regarding the coverage size, which quickly stabilizes to a value independent of the network size. In fact, we observed that, for small networks, the coverage is total; for sufficiently large networks, the coverage size depends solely on the ratio between battery capacity and charging time.

Chapter 5

Ideal Centralized Strategy

From the effectiveness point of view, the "ideal" optimal on-line strategy, which we shall call simply CENTRALIZED, is clearly when the robot knows at any time the current status of all sensors, and it computes on-line which request it must satisfy next so to minimize the number of sensing holes and/or their duration. Even more that any other centralized on-line strategy, in addition to a high computational complexity, CENTRALIZED assumes full knowledge and communication requiring constantly up-to-date global information; hence the communication costs required to implement it severely limit its feasibility. This unrealistic protocol will be used as a benchmark to compare the optimal ideal performance with the one of the decentralized ON-DEMAND strategy presented and analyzed in the previous section.

5.1 Description

In CENTRALIZED, each request message is sent by the sensor to the charger; the robot processes all the current request messages, and it computes which request to satisfy next so to minimize the number of sensing holes and their duration. We are actually going to consider the ideal cost settings for CENTRALIZED: every request from every node reaches the robot directly and immediately, regardless of its current location; the robot can reach any node from any node in the same amount d of time, regardless of its distance; and the robot's processing time is negligible regardless of the complexity of the computation.

Notice that the behavior of the robot under Centralized in this setting is easy to describe: the robot just processes and services the request messages in the order they arrive; if two or more requests arrive at the same time, ties are broken by a fixed rule (e.g., by Ids).

5.2 Analysis

We show that the coverage size and the disconnection time stabilize and we compute their value considering the case of constant recharging time ρ.

Theorem 7. *If $n > (\Delta + \rho + d)/(\rho + d)$ then, under the* Centralized *strategy, there exists a time t such that, for all $t' > t$ and all $x \in \mathcal{X}$ we have:*

$$\lceil \frac{\Delta}{(\rho + d)} \rceil \leq Coverage(\texttt{Centralized}, t') \leq \lceil \frac{\Delta}{(\rho + d)} \rceil + 1;$$

$$Disconnect(\texttt{Centralized}, t', x) = (n - 1)(\rho + d) + d - \Delta.$$

Proof. Let $x_0, x_1, \dots x_{n-1}$ be the sensors ordered by their initial battery charge; the robot will thus initially services the nodes in this order. Let $t_{i,1}$ be the time when x_i is fully charged for the first time; thus, $t_{i,1} + \rho + d \leq t_{i+1,1}$. This implies that the time $t_{i,j}$ when the node x_i will been fully charged for the j-th time is $t_{i,j} + \rho + d \leq t_{i+1,j}$. In other words, the nodes will be always serviced according to the initial order.

After fully charging node x_i for the j-th time at time $t_{i,j}$, the robot will have fully charged all other nodes after at least $(\rho + d)(n - 1)$ time units.

By hypothesis, $n \geq \frac{\Delta+\rho+d}{\rho+d}$; thus, $(\rho+d)(n-1) \geq \hat{\Delta}$. In this case, by time $t_{i,j} + (\rho + d)(n-1)$, the $(j+1)$-th request by x_i is already in the queue and thus the robot goes to service x_i next. In other words, after a full round, the robot continuously charges the nodes in the same order without ever stopping.

The coverage size in this case is easy to compute. Let X be the total number of nodes that are alive at any given time. This means that $\rho(X-1) + (X-1)d < \Delta$, but $\rho X + Xd \geq \Delta$ and thus $\frac{\Delta}{(\rho+d)} < X < \frac{\Delta+\rho+d}{(\rho+d)}$.

Since after time $t_{0,2}$ the robot keeps charging every node it encounters, it will spend $(n-1)(\rho+d)+d$ time units to complete a round; hence, every node will have disconnection time $(n-1)(\rho+d)+d-\Delta$.

□

This theorem has an immediate very strong consequence for the effectiveness of On-Demand:

Theorem 8. *Let the system become stable under* On-Demand. *If $n > \frac{2\Delta}{\rho} + 1$ then there exists a time t such that for all $t' > t$ and all $x \in \mathcal{X}$*

$$Coverage(\texttt{On}-\texttt{Demand}, t') = Coverage(\texttt{Centralized}, t'),$$

$$Disconnect(\texttt{On}-\texttt{Demand}, t', x) = Disconnect(\texttt{Centralized}, t', x)$$

Proof. Since $\frac{2\Delta}{\rho} + 1 > \frac{\Delta+\rho+d}{\rho+d}$, the claim follows directly from Theorems 4 and 7. □

In other words, for all networks with $n > \frac{2\Delta}{\rho} + 1$, the recharging strategy On-Demand, with its low communication and computations costs, performs as well as the optimal strategy.

5.3 Experimental Comparison with On-Demand

Under the same simulation environment employed in the previous chapter, we now compare average disconnection time and coverage size of On-Demand with Centralized.

We already established that, when $n > \frac{2\Delta}{\rho} + 1$, the two strategies are equivalent both in terms of coverage size and disconnection time (Theorem 8). Extensive experimental results, varying Δ, ρ, d, and n, confirm the theoretical findings. We show below only an example for a specific setting, all the other choices of the parameters are consistent with this behavior.

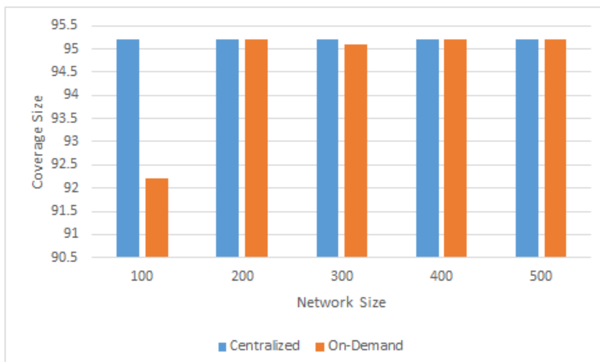


Figure 5.1: On-Demand vs Centralized: Coverage. ($\Delta = 2000$, $\rho = 20$, $d = 1$).

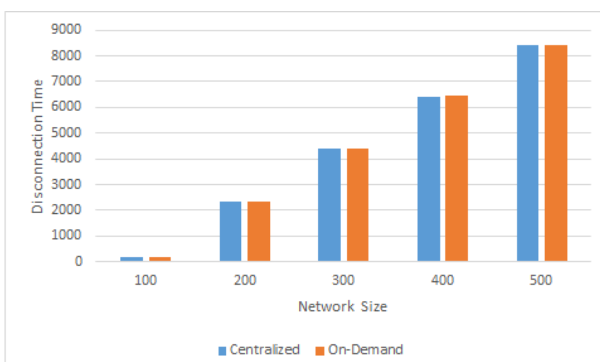


Figure 5.2: On-Demand vs Centralized: Disconnection time. ($\Delta = 2000$, $\rho = 20$, $d = 1$).

Figures 5.1 and 5.2 show coverage and disconnection time of the two strategies for different network sizes when $\Delta = 2000$, $\rho = 20$ and $d = 1$. Notice that Theorem 8 does not hold when $n < \frac{2\Delta}{\rho} + 1$; in fact, as shown in Figure 5.1, Centralized has a much better coverage than On-Demand for the case $n = 100$.

5.4 Summary

In this Chapter, we analyzed the ideal optimal centralized strategy, which is clearly very demanding in terms of resources, and we compared both analytically and experimentally its disconnection time and coverage size with the strategy described in the previous Chapter (On-Demand), which is fully decentralized and requires very little memory and communication. We conclude that, for large enough networks, On-Demand achieves the same, optimal, effectiveness.

Chapter 6

Blind Strategy

In this Chapter, we describe a very simple decentralized strategy, called BLIND, which is even simpler than ON-DEMAND. We compare its performance with ON-DEMAND observing that it achieves the same results in spite of its extreme simplicity.

6.1 Decentralized Blind Strategy

In this strategy, the robot visits the sensors in a fixed circular order; if the current sensor has a battery level below the threshold, the robot recharges it and then moves to the next sensor, otherwise it moves directly to the next sensor. That is, no pending recharging requests are being communicated; the robot always operates in the same way, unaware of whether there are pending requests at some node other than the current one, and without need of memory or calculation. Because of its nature, we call this decentralized approach BLIND.

6.2 Experimental Results

The On-Demand strategy uses only local communication and provides the charger with the awareness of the existence of a current request; the Blind strategy does not use any communication but does not provide any information to the charger.

In this Section we examine experimentally the effectiveness of these two recharging strategies. The experimental results show the perhaps unexpected result that, in most cases, *the Blind strategy is as efficient as the On-Demand one.* All results are derived with 98% confidence level.

6.2.1 Effects of Charging Time

Consider first the effect of charging time ρ on charging latency, disconnection time, and coverage size for the On-Demand and Blind strategies. For both strategies, as the charging time increases, both charging latency and disconnection time increase, whereas coverage size decreases. Interestingly, for each of those measures, the costs of the two strategies are very close in all experiments. More precisely, they are within 2.7% for the Charging Latency, 4% for the Disconnection time, and 0.4% for the Coverage Size.

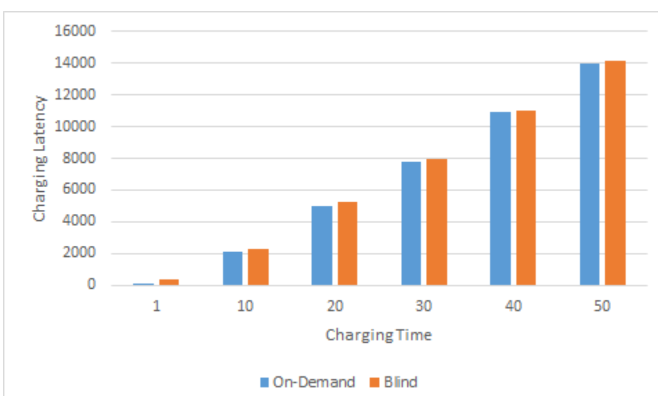


Figure 6.1: Effect of ρ on Charging Latency (n=300,Δ=2000,d=1)

Shown in Figures 6.1-6.3 are the results for $n = 300, \Delta = 2000, d = 1$.

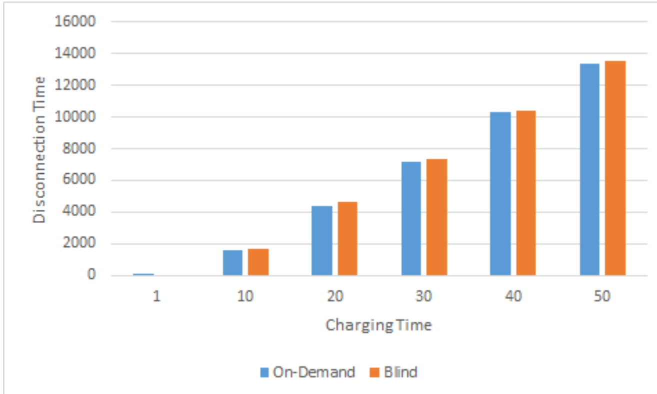


Figure 6.2: Effect of ρ on Disconnection Time (n=300,Δ=2000,d=1)

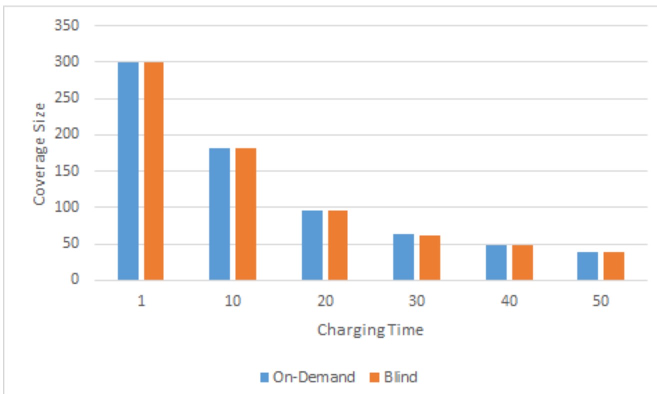


Figure 6.3: Effect of ρ on Coverage Size (n=300,Δ=2000,d=1)

6.2.2 Effects of Node Capacity

We then analyze the effect of the node capacity Δ on charging latency, disconnection time, and coverage size for the ON-DEMAND and BLIND strategies. For both strategies, as the node capacity increases, both charging latency and disconnection time decrease, whereas coverage size increases. Also in this case, for each of those measures, the costs of the two strategies are very close in all experiments. More precisely, they are within 2% for the Charging Latency, 3% for the Disconnection time, and 0.45% for the Coverage Size.

Shown in Figures 6.4-6.6 are the results when $n = 300, \rho = 20, d = 1$.

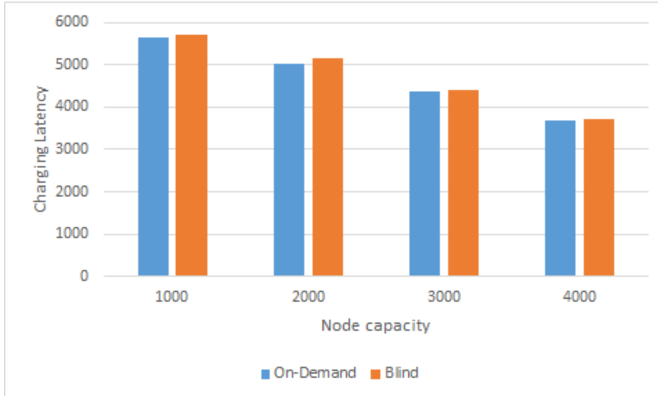


Figure 6.4: Effect of Δ on Charging Latency (n=300,ρ=20,d=1)

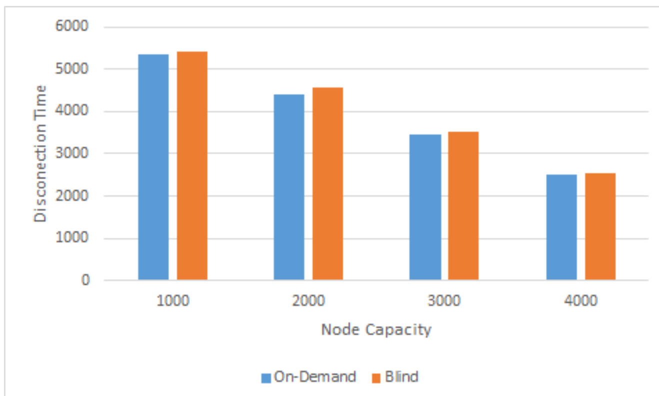


Figure 6.5: Effect of Δ on Disconnection Time (n=300,ρ=20,d=1)

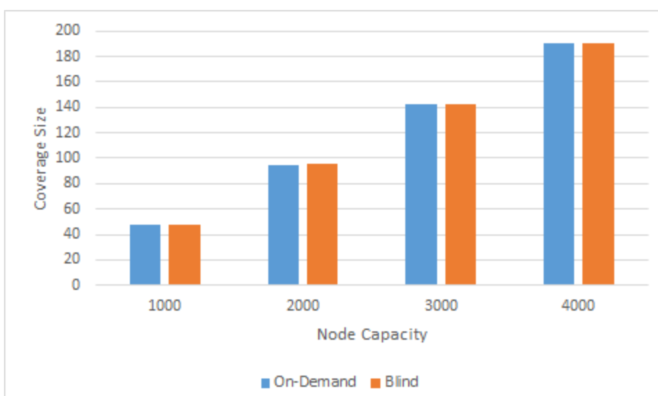


Figure 6.6: Effect of Δ on Coverage Size (n=300,ρ=20,d=1)

6.2.3 Effects of Network Size

The next parameter analyzed is the network size n. For each of the performance measures (charging latency, disconnection time, and coverage size) the impact of n is almost the same in both the On-Demand and Blind strategies.

For both strategies, as the network size increases, the charging latency and disconnection time increase.

The impact of the network size on the coverage size is unlike the one observed for the other parameters. In fact, as the network size increases, from a certain value of n on, the coverage size remains the same. In other words, increasing the number of sensors would actually increase the number of coverage holes. This unexpected behavior holds with both strategies.

Also in this case, for each of those measures, the performance of the two strategies is very close in all experiments. More precisely, they are within 2.5% for the Charging Latency, 3.5% for the Disconnection time, and 0.4% for the Coverage Size.

Shown in Figures 6.7-6.9 are the results when $\Delta = 2000, \rho = 20, d = 1$.

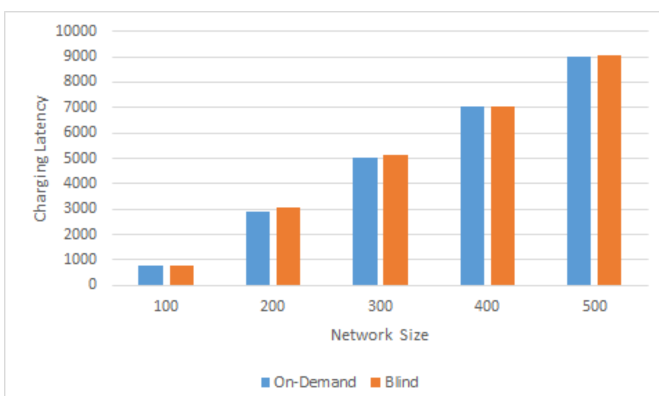


Figure 6.7: Effect of n on Charging Latency ($\Delta = 2000$,$d = 1$,$\rho = 20$)

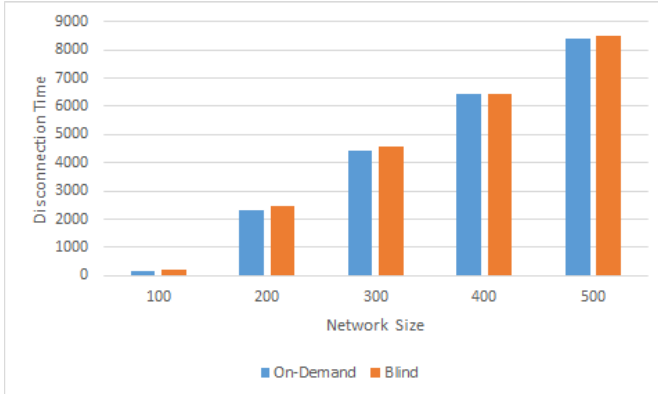


Figure 6.8: Effect of n on Disconnection Time (Δ=2000,d=1,ρ=20)

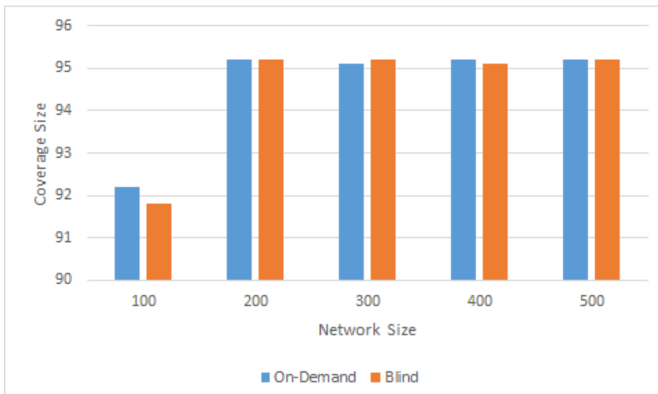


Figure 6.9: Effect of n on Coverage Size ($\Delta = 2000$,$d = 1$,$\rho = 20$)

6.2.4 Effects of Travel Time

Finally, we investigate the impact of the time spent by the robot has on the charging latency, disconnection time and coverage size in the two strategies. As expected, for both strategies, as the travel time between nodes increases, both charging latency and disconnection time increase, whereas coverage size decreases. Also in this case, for each of those measures, the costs of the two strategies are very close in all experiments. More precisely, they are within 1% for all three measures. Shown in 6.10-6.12 are the results when $n = 300, \Delta = 2000, \rho = 20$.

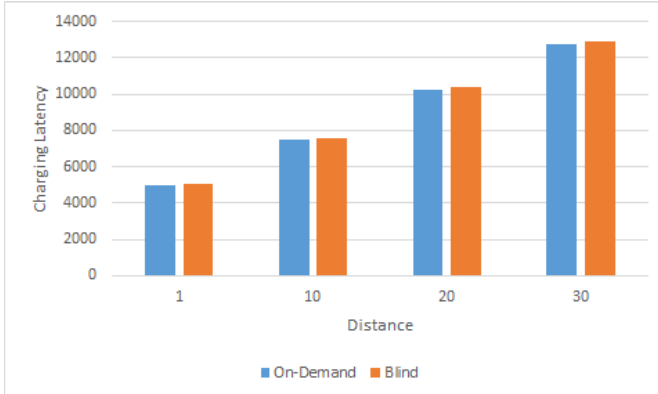


Figure 6.10: Effect of d on Charging Latency (n=300,Δ=2000,ρ= 20)

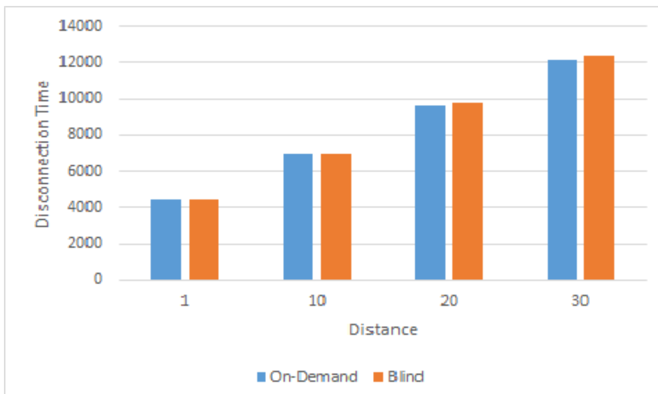


Figure 6.11: Effect of d on Disconnection Time (n=300,Δ=2000,ρ= 20)

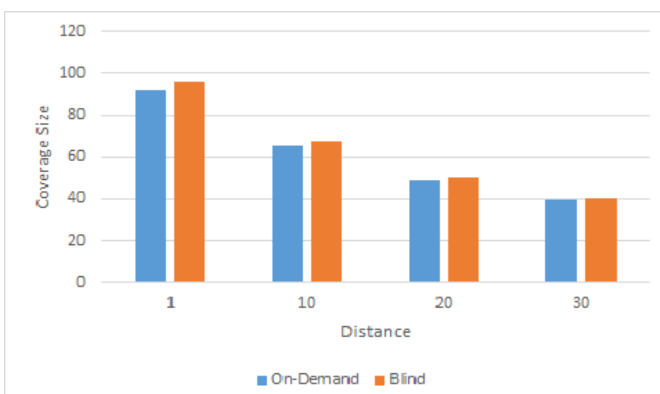


Figure 6.12: Effect of d on Coverage Size (n=300,Δ=2000,ρ= 20)

6.2.5 Discussion

The experiments show the surprising result that, in spite of its small overhead and its simplicity, the decentralized Blind strategy is in most cases as effective as On-Demand (and thus equivalent to the optimal centralized one, which however uses full information and communication).

In the following we summarize the relationship between the performance of the three strategies through a series of properties.

Stability

All the experiments conducted in this and in the previous Chapter confirm that all three strategies become stable after a short transient, as we summarize in the following Property.

Property 1. (Stability and Transient)
Under Centralized*,* On-Demand *and* Blind*, after a transient, Coverage Size, Disconnection Time and Charging Latency become stable for any choice of the values of the parameters. The transient is at most two complete rounds by the robot.*

OnDemand vs. Blind

As seen from the experiments, for large networks, after a transient, the coverage size does not change under both On-Demand and Blind. That is:

Property 2. (Bounded Coverage)
Both under the On-Demand *and the* Blind *strategies, there exists a* $\hat{n}$ *such that, in every network of size* $n > \hat{n}$*, after a transient* $\hat{t}(n)$*, for all* $t \geq \hat{t}(n)$

$$|CoverageSize(\mathtt{On-Demand}, t) - CoverageSize(\mathtt{On-Demand}, \hat{t}(n))| \leq 1$$

$$|CoverageSize(\mathtt{Blind}, t) - CoverageSize(\mathtt{Blind}, \hat{t}(n))| \leq 1$$

This phenomenon holds in all experiments regardless of the initial distribution of charges; that is, the same phenomenon has been observed not only with random distributions, but also with specific ones, such as when the charges are increasing in a clockwise order or in counterclockwise order.

Property 3. (Distribution-Independence)

Properties 1-2 hold regardless of the initial distribution of the charges and regardless of the threshold.

The examination of the experimental results shows that for all each performance measure (Charging Latency, Coverage Size, and Disconnection Time) the resulting value in the same network for the On-Demand and the Blind strategies is approximately the same:

Property 4. *For all values of n, Δ and ρ, the difference between* Blind *and* OnDemand *is the following: i) for Charging Latency, at most* 4.5%*; ii) for Disconnection Time, at most* 3.5%*; iii) for Coverage Size, at most* 0.4%*; For all values of d, the difference between* Blind *and* OnDemand *is at most* 1% *for all three measures.*

That is, with respect to all three metrics, the Blind strategy is as efficient as the On-Demand one.

Conclusion

Combining the above properties with the results of the previous Chapter we can conclude:

Property 5. *For sufficiently large networks,* Blind*,* OnDemand*, and* Central *provide the same charging latencies to sensors, and reach the same disconnection time, and the coverage size obtained by them are the same.*

6.3 Summary

In this Chapter, we briefly introduced an extremely simple decentralized strategy Blind, that requires very little resources and, in particular, no memory nor communication. We run experiments to evaluate charging latency, disconnection time, and coverage size of Blind. Combining this analysis with the results of the previous Chapters, we conclude with a comparison between the three strategies and we make the unexpected observation that, for large enough networks, they all have the same effectiveness. We can then conclude that the Blind strategy, which demands very little resources, is the most convenient to employ.

Chapter 7

Linear Sensor Network

In this chapter, we focus on decentralized on-line strategies for energy restoration by a robot in *linear sensor networks*, i.e. whose topology is modeled as a line $L = < x_0, x_1, \ldots, x_{n-1} >$ where each node x_i has two neighbors (x_{i-1} and x_{i+1}), except for the end nodes x_0 and x_{n-1} that have only one neighbor. Line-based shape is an important formation for many practical applications of WSN, such as intrusion detection, border surveillance, routing overlay formation, and information collection and has been extensively studied in the literature (e.g., [8, 19, 25, 33, 49, 61, 62, 72, 73, 105]).

In this chapter, we investigate three simple decentralized on-line energy restoration strategies for linear networks based on different approaches. We propose strategy On-Request, where a sensor whose battery is about to become depleted originates a recharging request and waits for the robot; the request is forwarded along the line until it reaches either the robot or another sensor waiting for the robot; when aware of a pending request, the robot moves in that direction. We examine also a simpler variant of this strategy, Straight, in which the direction of movement of the robot along the line cannot be changed until it reaches the end of the line. We finally consider an even simpler on-line strategy, Slide, where no requests are sent and the robot automatically and continuously moves from one end of the line to the other, servicing any sensor found needing recharging.

7.1 Strategies

7.1.1 On-Request Strategy

Every node maintains a local variable, *direction*$\in$ {*left,right,here*}, indicating (the direction of) the current location of the robot; the value of *direction* at a node x_i is updated when the robot arrives at x_i, and again when it leaves x_i. Initially, the robot is at one end of the line and all nodes are aware of its location.

Every node maintains three local Boolean variables, *pending-left*, *pending-right*, and *pending-here*, indicating whether or not there is an unserviced request from that direction.

When the sensor battery of a node x_i reaches the predefined threshold, the process of requesting a recharge by the robot is started. If the robot is not already there (i.e., *direction*$\neq$*here*), then: (i) in absence of other pending requests (i.e., *pending-left* = *pending-right* = *false*) x_i sends a `Request` message to its neighbor in *direction*; (ii) it sets *pending-here* = *true*; and (iii) it waits for the robot. Should the robot be already at x_i, is just notifies the robot, which will charge it.

Node x_i, upon receiving a `Request` message from its neighbor x_{i-1} (respectively, x_{i+1}) does as follows: If no pending request has been originated by x_i (i.e. *pending-here* =*false*), then it sets *pending-left*= *true* (resp., *pending-right* = *true*). If the robot is not already there (i.e., *direction*$\neq$*here*) then it sends a `Request` message to x_{i+1} (resp., x_{i-1}), otherwise it notifies the robot.

In other words, each request is forwarded along the line until it reaches either the robot or another node waiting for the robot. Consequently, not all requests reach the robot; rather they create a trail of requests to be served by the robot when it becomes available.

At any time, the robot has a current direction of movement, *current*$\in$ $\{left, right\}$. When arriving at a node x_i, if x_i is an end point of the line (i.e., $i = 0$ or $i = n - 1$), then the robot changes the value of *current* to the other direction. If informed that x_i needs recharging, it will charge it. Once x_i has been charged or if it never needed recharging, the robot determines if there are pending requests known to x_i.

If there are no pending requests (i.e., at x_i *pending-left* = *pending-right* = *false*) then R waits until notified by x_i that a request is present.

If there is a pending request from only one direction, then R sets *current* to that direction and moves there; when this happens, x_i updates the value of the corresponding *pending* variable (to *false*), as well as the value of *direction*.

If there are requests from both directions (i.e., at x_i *pending-left* = *pending-right* = *true*), it choses to service *current* and moves there; when this happens, x_i updates the value of the corresponding *pending* variable (to *false*) and the value of *direction*; it then forwards[1] the still pending request to *direction*.

Note that each node communicates only locally, either with the node next in the line (to send a request), or with the robot if currently there to communicate its need to be recharged and/or the presence of a still pending request.

7.1.2 Straight Strategy

The Straight strategy is just a simplification of On-Request, where the robot, in presence of requests, keeps on moving in the same direction, charging needing nodes it encounters, but changing direction only when reaching an end node of the line.

The main difference with On-Request is when the robot at node x_i is informed of a request that it is not originating locally (i.e., not by x_i) nor coming from its current direction of movement (*current*). In this case, the robot keeps *current* unchanged (unlike On-Request where, under these circumstances, the value of *current* would be changed) and moves there; when this happens, x_i updates the value of the corresponding *pending*

[1]The transmission of this message can be avoided; e.g., the robot can "carry" this information in its move, and inform the node once there.

variable (to *false*) and the value of *direction*; it then forwards[1] the still pending request to *direction*.

7.1.3 Slide Strategy

The last strategy we consider, Slide, is extremely simple and requires no communication. In this strategy, no requests are sent and the robot automatically and continuously moves from one end of the line to the other, servicing any node it finds needing recharging. Note that, while in the On-Request and Straight strategies the robot moves to serve a sensor only when needed, in the Slide strategy it might move uselessly along the line several times without performing any work before encountering a sensor in need.

7.2 Experimental Analysis

The On-Request and the Straight strategies use local communication and provide the charger with the awareness of the existence of a current request. On the other hand, the blind strategy Slide does not use any communication but does not provide any information to the charger.

In this Section we examine experimentally the effectiveness of the three recharging strategies for linear sensor networks in terms of sensing holes (the reverse measure of coverage size) and disconnection time. The experimental results show that the simplest strategy, Slide, is the most efficient under all the considered scenarios

7.2.1 Simulation Environment

For the experiments in this setting we use the same environment as in the previous sections.

The parameters involved in the experiments are indicated in the table below, together with the values that have been considered:

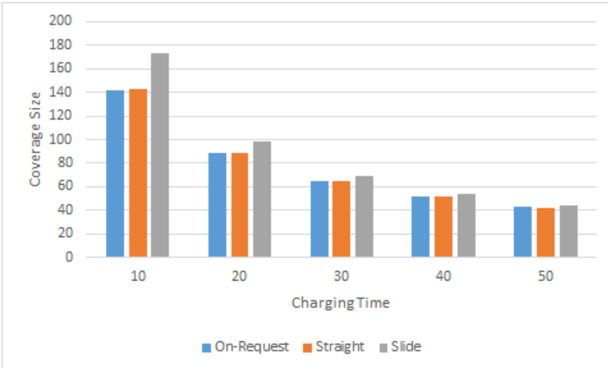
Parameters	Values
Number of nodes n	100; 200; 300; 400; 500
Node Capacity δ	1000,2000, 3000,4000
Threshold τ	30% of Δ
Charging Time ρ	10,20,30,40
Travel time d	1,10,20,30

Initially, each sensor x has a random amount of energy level in the range $[\tau\ ,\ \Delta]$, and the charger is initially located at one end node of the line. For each possible choice of the values of the parameters, ten experiments are carried out, each for a number of steps sufficient to ensure the system stability. Averages of the effectiveness measures are then calculated. The results on the observed impact of each parameters on the effectiveness measures are discussed in the following. All results are obtained with 98 % confidence level.

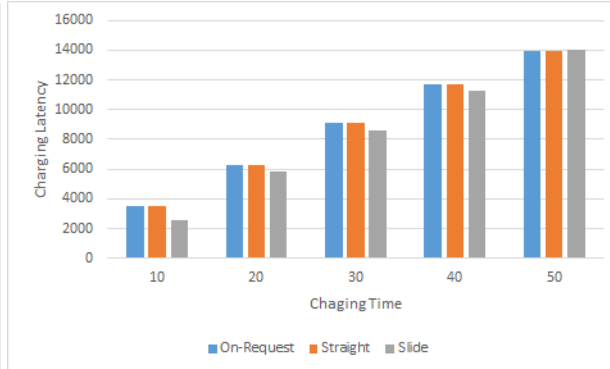
7.2.2 Effects of Charging Time ρ

We study the impact of charging time ρ for the three strategies, analyzing its impact on disconnection time, charging latency and coverage size for different network sizes.

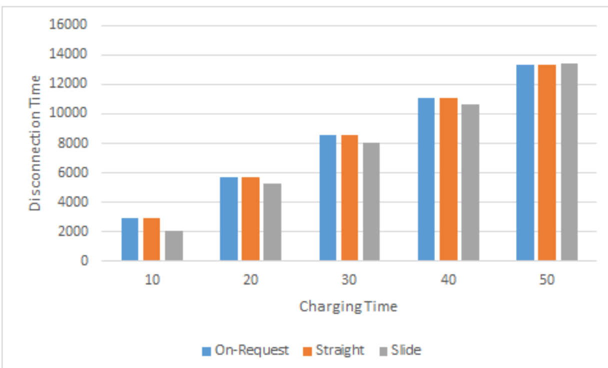
As expected, the coverage size decreases when the charging time increases, while the disconnection time decreases, for all the strategies. Interestingly, SLIDE is always the most efficient of the three strategies in terms of all performance measures; see Figure 7.2. This is more evident for smaller values of ρ while the difference becomes negligible for large values of ρ.



(a) Effect of Charging Time on Coverage Size



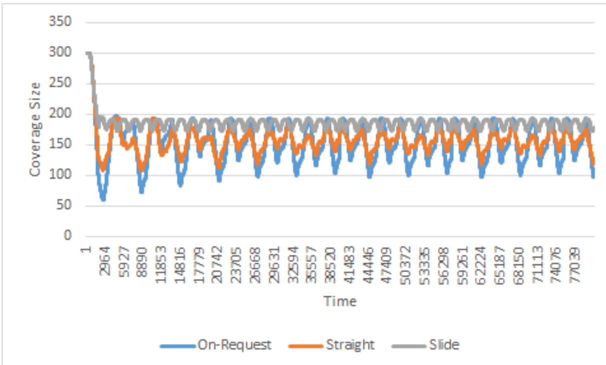
(b) Effect of Charging Time on Charging Latency



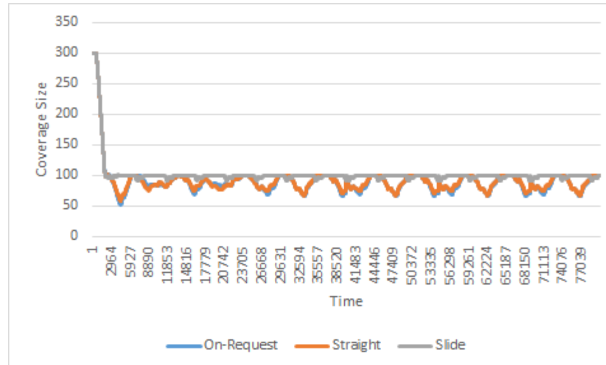
(c) Effect of Charging Time on Disconnection Time

Figure 7.1: Effect of Charging Time on average (a) Coverage Size (b) Charging Latency and (c) Disconnection Time (n=300, Δ= 2000, d=1).

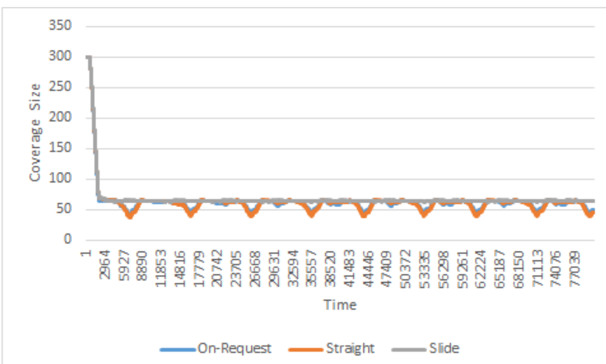
Consider now how coverage size changes during the execution of the three algorithms for different values of ρ (when fixing Δ and n). Interestingly, coverage size oscillates between a minimum and a maximum coverage, and the oscillations's amplitude tend to decrease while increasing the charging time. An example of this behavior is given in Figure 7.2 for $n = 300$, $\Delta = 2000$, and $d = 1$. In this case, we can observe that, for $\rho = 10$ the oscillations are quite evident for all three strategies; on the other hand, for $\rho = 40$, SLIDE provides a constant coverage, while ON-REQUEST and STRAIGHT behave in the same way with very small oscillations. In all cases, SLIDE is confirmed to be the strategy that offers the best coverage with smallest oscillations.



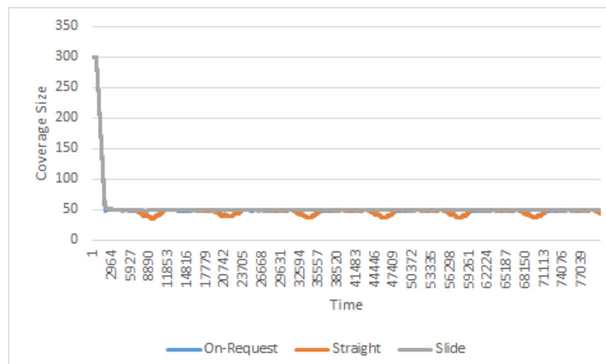
(a) $\rho = 10$



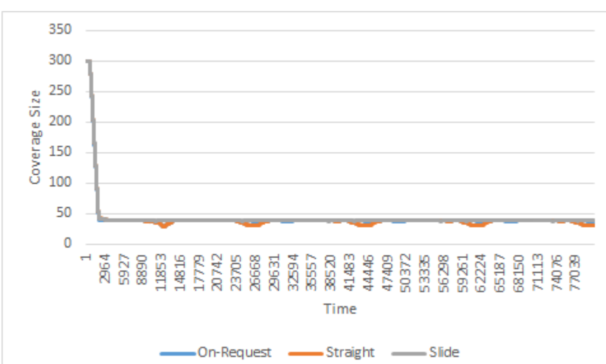
(b) $\rho = 20$



(c) $\rho = 30$



(d) $\rho = 40$



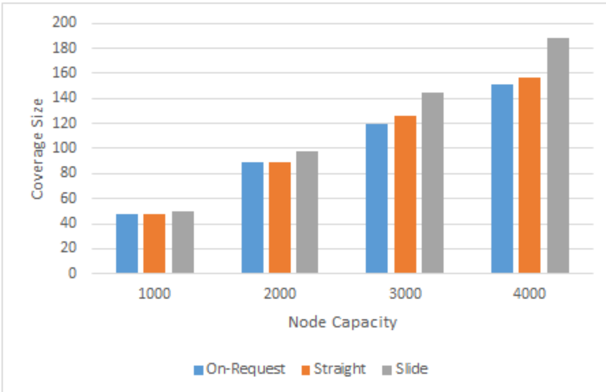
(e) $\rho = 50$

Figure 7.2: Effect of ρ on Coverage Size for the three strategies ($n = 300$, $\Delta = 2000$,$d = 1$).

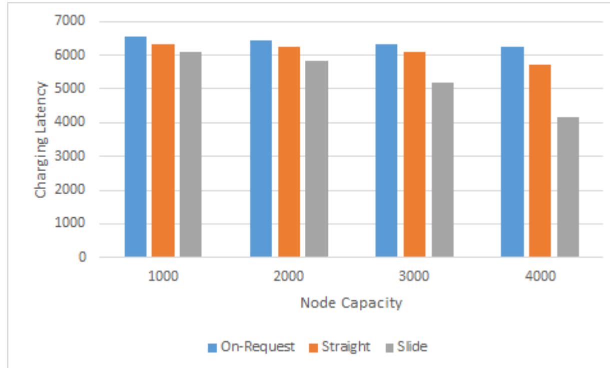
7.2.3 Effects of Node Capacity Δ

The effect of the node capacity Δ on average disconnection time, charging latency and coverage size for the three strategies, is analyzed fixing the other parameters. For all three strategies, when Δ increases coverage size increases while disconnection time and charging latency decrease (as expected). Again we observe that SLIDE has significantly better performances than ON-REQUEST and STRAIGHT , and this becomes increasingly more evident when the node capacity increases. Shown in Figure 7.3 are the results when n= 300; ρ= 20; d= 1.

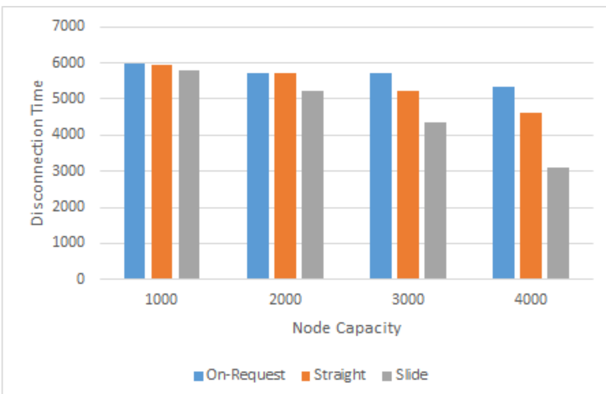
When studying the coverage size varies in time for different values of Δ, we observe again an oscillating behavior in all three strategies, similar to the behavior observed when changing ρ. Not surprisingly, ON-REQUEST generally displays oscillations of higher amplitude, which is probably due to the fact that the robot can change direction quite frequently when following the requests. This behavior is especially clear for larger values of Δ, when the difference among the three strategies becomes also more evident. Consistently with the results obtained when varying ρ, we can observe that, after a short transient, SLIDE outperforms the other strategies producing small oscillations, followed by STRAIGHT, which is better than ON-REQUEST especially for larger values of Δ. Figure 7.3 shows the variations of coverage size in time for the case n = 300, $\rho = 20$, d = 1.



(a) Effect of Node Capacity on Coverage Size



(b) Effect of Node Capacity on Charging Latency



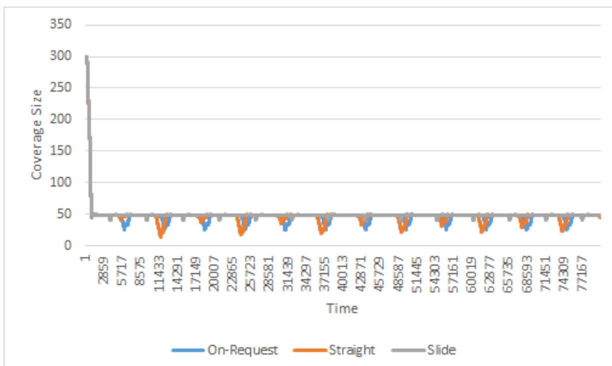
(c) Effect of Node Capacity on Disconnection Time

Figure 7.3: Effect of Node Capacity on average (a) Coverage Size (b) Charging Latency and (c) Disconnection Time ($n = 300$, $d = 1$, $\rho = 20$).

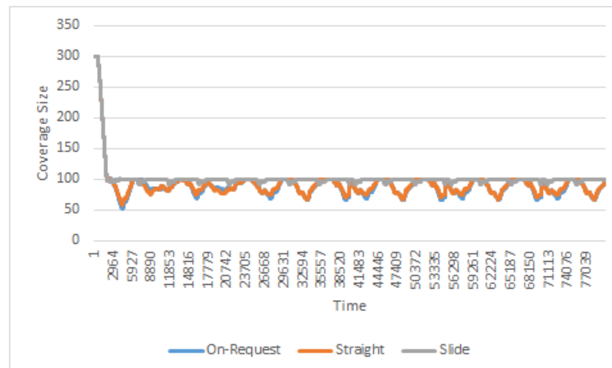
7.2.4 Effects of Network Size

We now observe average disconnection time, charging latency and coverage size for the three strategies, considering different network sizes and fixing the other parameters. As expected, disconnection time and charging latency increase for all three strategies as the network size increases, roughly at the same rate. Coverage size, on the other hand, increases at a much smaller rate for SLIDE than for the other two strategies. In any case, SLIDE is the best strategy; the difference between the three becomes less evident as the network size increases. Shown in Figure 7.5 are the results when $\Delta = 2000$, $\rho = 20$ and $d = 1$.

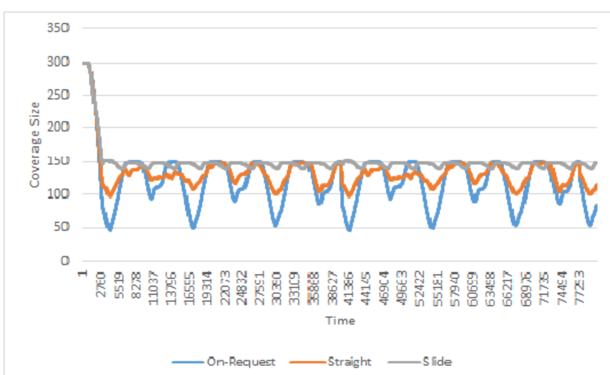
The very slight increase of coverage size when varying n is noticeable also from the



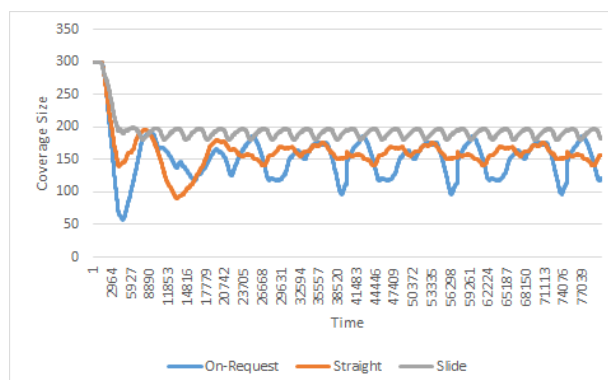
(a) $\Delta = 1000$



(b) $\Delta = 2000$



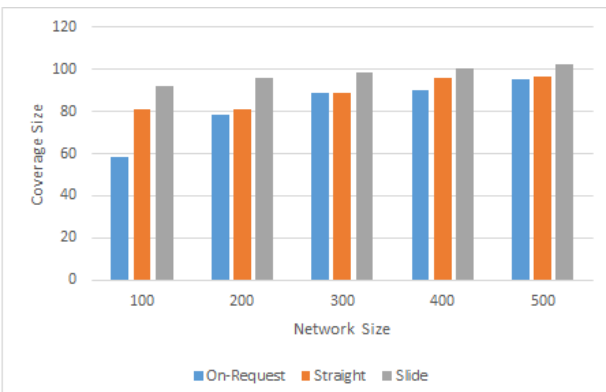
(c) $\Delta = 3000$



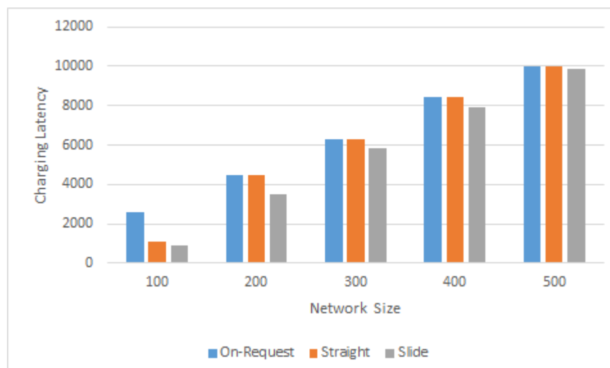
(d) $\Delta = 4000$

Figure 7.4: Effect of Δ on Coverage Size for the three strategies ($n = 300$, $d = 1$, $\rho = 20$).

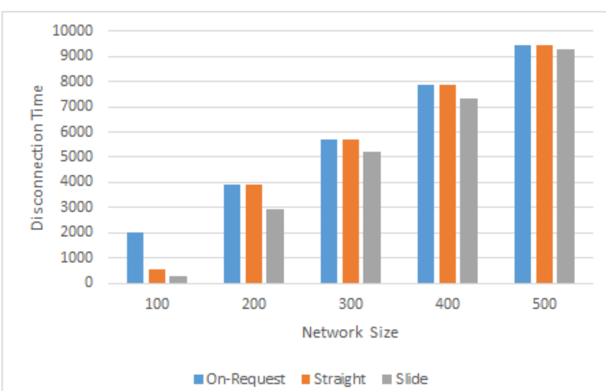
analysis of the coverage size behavior in time, where we observe that both On-Request and Straight display an oscillating periodic behavior with the axis of the oscillations increasing only slightly increasing n; the oscillation have a decreasing amplitude while the network size grows. On the other hand, Slide seems to stabilize around a constant coverage size. In next Section we will provide some theoretical support to these observations by analyzing our strategies from particular initial conditions of the battery charges. Shown in Figure 7.6 is the variation in time of the coverage size for $\Delta = 2000$, $\rho = 20$ and $d = 1$. Note that On-Request and Straight become almost identical when n = 300 and n = 400. Also, the coverage size for Slide seems to stabilize around the value of 100, regardless of the network size.



(a) Effect of Network Size on Coverage Size

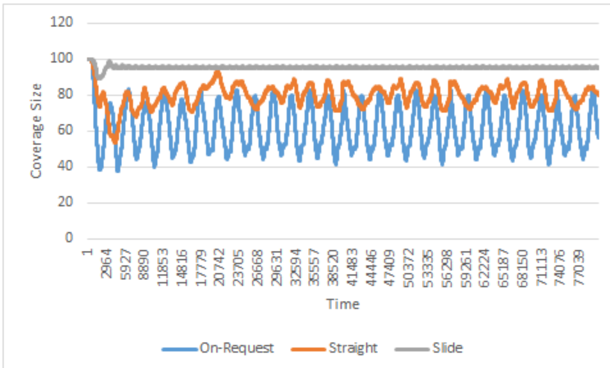


(b) Effect of Network Size on Charging Latency

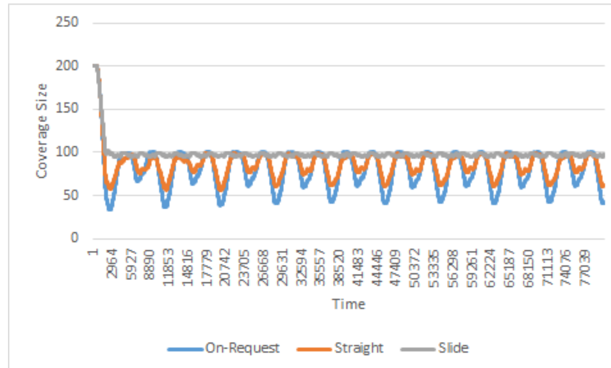


(c) Effect of Network Size on Disconnection Time

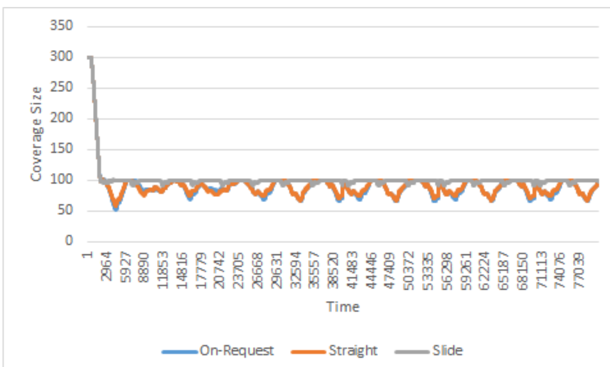
Figure 7.5: Effect of n on average Coverage Size, Charging Latency and Disconnection Time ($\Delta = 2000$, $d = 1$, $\rho = 20$).



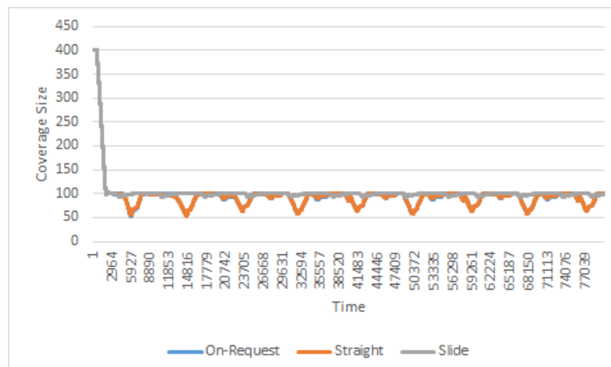
(a) $n = 100$



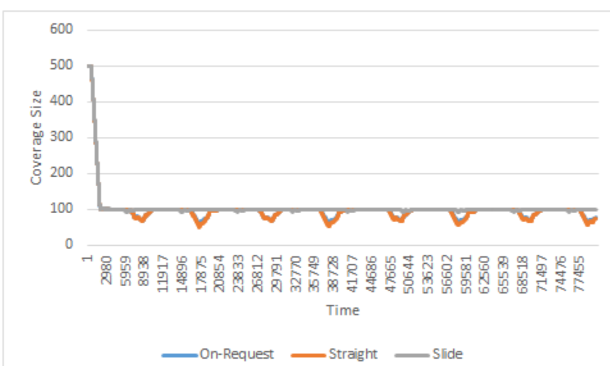
(b) $n = 200$



(c) $n = 300$



(d) $n = 400$

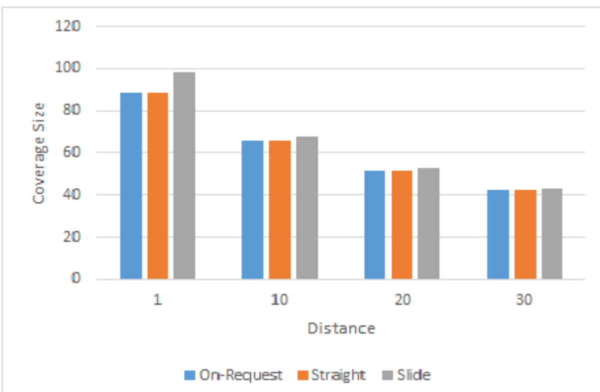


(e) $n = 500$

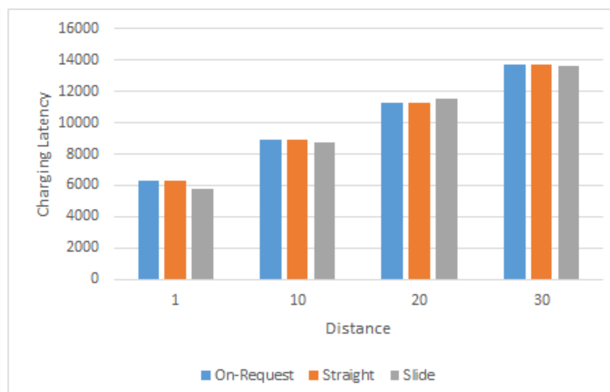
Figure 7.6: Effect of n on Coverage Size for the three strategies ($\Delta = 2000$, $d = 1$, $\rho = 20$).

7.2.5 Effects of Travel Time d

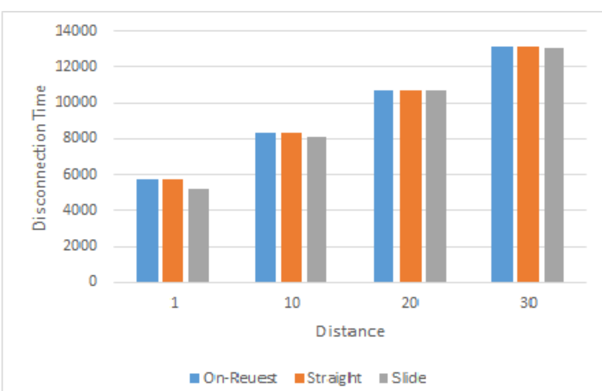
We investigate the impact of the travel time between nodes on the average disconnection time, charging latency and coverage size in the three strategies. As expected, for all strategies, as the travel time between nodes increases, disconnection time and charging latency increase and coverage size decreases. The SLIDE strategy outperforms the others, especially when the travel time is small, while the difference between the three strategies becomes less evident when the travel time becomes larger. Shown in Figure 7.7 are the results when $\Delta = 2000$,$n = 300$ and $\rho = 20$.



(a) Effect of Travel Time on Coverage Size



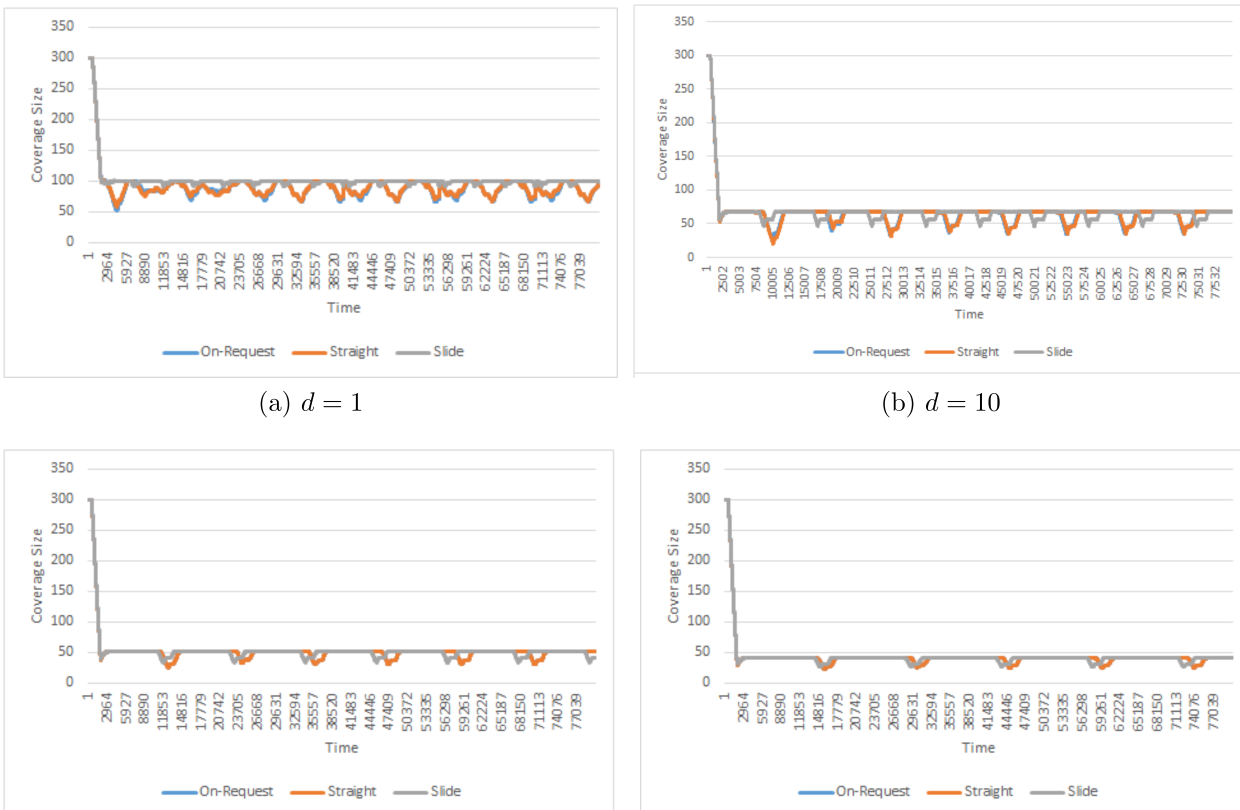
(b) Effect of Travel Time on Charging Latency



(c) Effect of Travel Time on Disconnection Time

Figure 7.7: Effect of d on average Coverage Size, Charging Latency and Disconnection Time ($\Delta = 2000$, $n = 300$, $\rho = 20$).

The analysis of how the coverage size varies in time shows that, as the travel time between nodes increases, coverage size decreases in all strategies with the oscillating behavior becoming less and less pronounced (see Figure 7.8 where $\Delta = 2000$,$n = 300$ and $\rho = 20$).



(a) $d = 1$

(b) $d = 10$

(c) $d = 20$

(d) $d = 30$

Figure 7.8: Effect of d on Coverage Size for the three strategies (Δ=2000, n=300, ρ=20)

7.2.6 Effects of Threshold

We observe coverage size and average disconnection time, for the three strategies, considering different threshold and fixing the other parameters. Disconnection time slight increases for all three strategies as the threshold increases, roughly at the same rate. Coverage size, on the other hand, increases at a much smaller rate for On-Request and straight strategies than for the Slide . In any case, Slide is the best strategy. Shown in Figure 7.9 are the results when $\Delta = 2000$, $n = 300$ and $d = 1$.

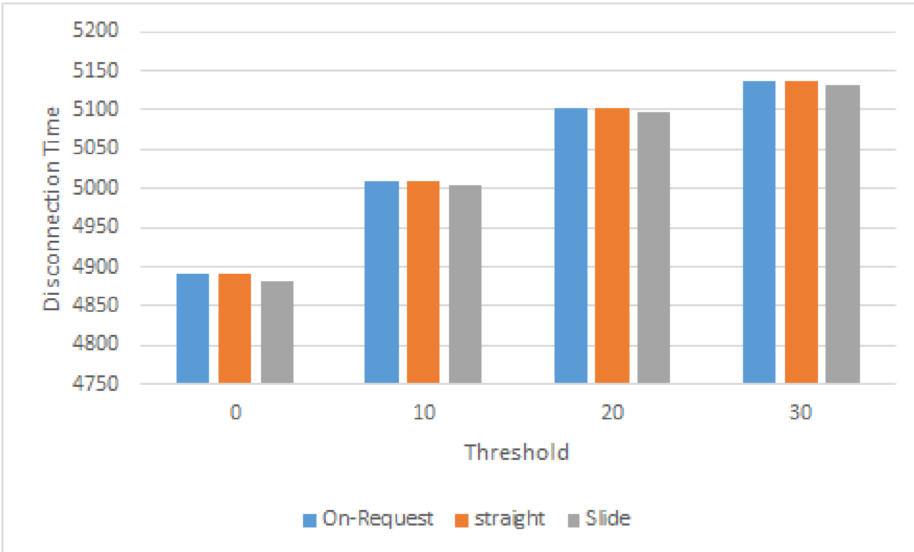
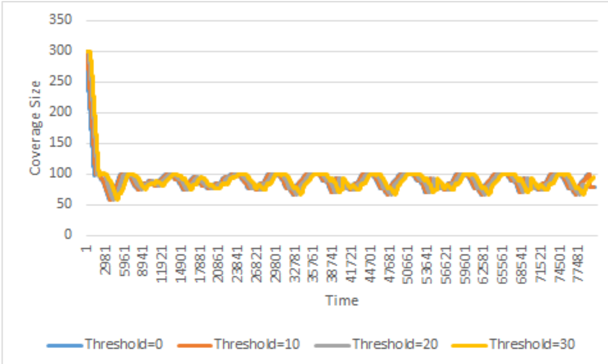
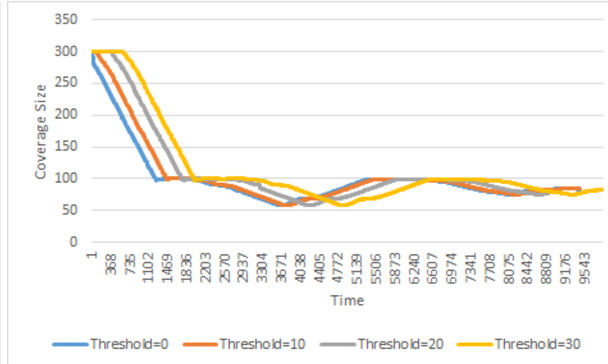


Figure 7.9: Effect of Threshold on Disconnection Time for the Three Strategies (Δ=2000, n=300, $d = 1$).

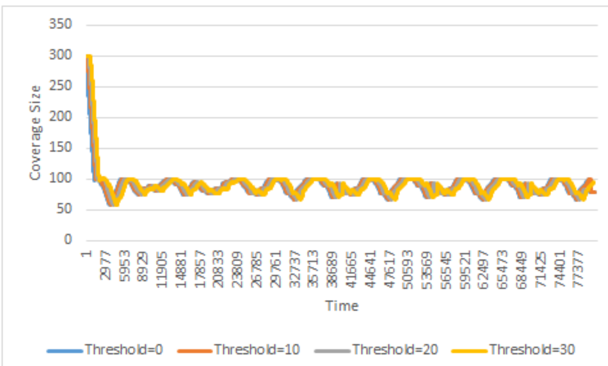
The very slight difference of oscillating coverage size when varying threshold is noticeable, also from the analysis of the coverage size behavior in smaller part of the time, where we observe that both On-Request and Straight display the same oscillating periodic behavior with the axis of the oscillations regardless of the threshold. On the other hand, Slide seems to stabilize around a constant coverage size. Shown in Figure 7.10 is the variation in time of the coverage size for $\Delta = 2000$, $n = 300$ and $d = 1$. Note that On-Request and Straight become almost identical. Also, the coverage size for Slide seems to stabilize around the value of 100, regardless of the threshold.



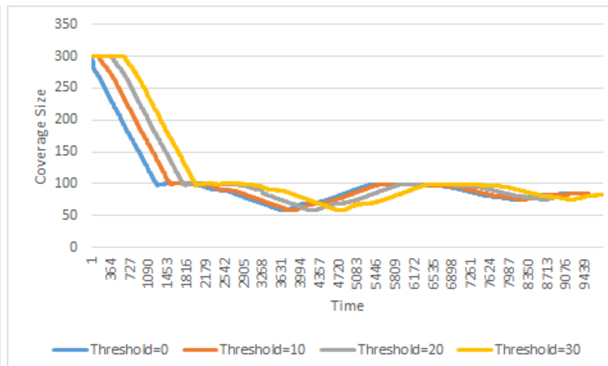
(a) Effect of Threshold on Coverage Size in On-Request Strategy



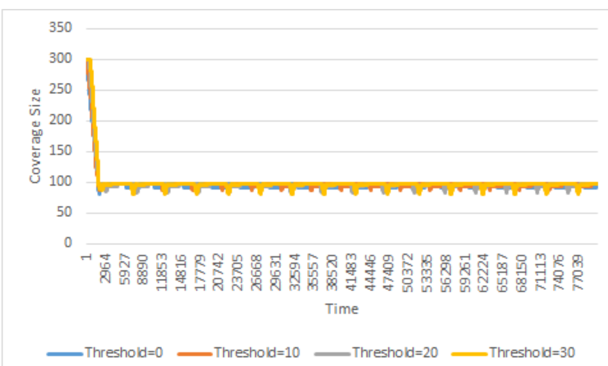
(b) Effect of Threshold on Coverage Size in On-Request Strategy



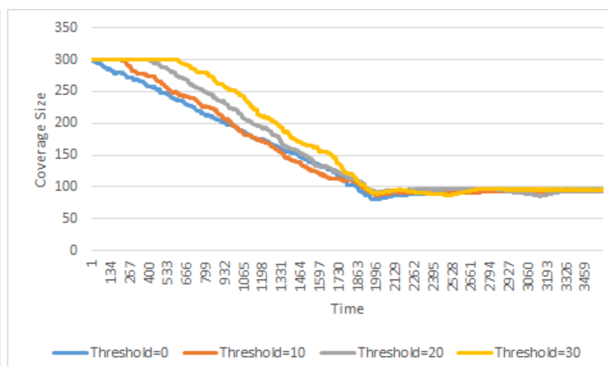
(c) Effect of Threshold on Coverage Size in Straight Strategy



(d) Effect of Threshold on Coverage Size in Straight Strategy



(e) Effect of Threshold on Coverage Size in Slide Strategy



(f) Effect of Threshold on Coverage Size in Slide Strategy

Figure 7.10: Effect of τ on Coverage Size ($\Delta = 2000$, $n = 300$, $d = 1$).

7.3 Analytical Support

The experimental analysis has unexpectedly shown that the performance of the Slide strategy is as good as, if not better, than that of the two, more sophisticated, on-demand strategies. In this section we provide some analytical support to these findings.

We study analytically the performance of the three strategies in the "bootstrap" scenario where initially (at time $t = 0$) all batteries are empty and the robot is at one end of the line, say x_n. In this initial scenario, all three strategies have the same *initialization* phase, in which the robot sequentially charges all the nodes, starting from x_n. For all three strategies, initialization is completed at time $t_I = n\rho + d(n-1)$, with the robot at x_1. From this moment on, the robot will behave in possibly different ways, depending on the strategy. We examine now their performance when $\tau = 0$; that is, a node sends a request (in the on demand strategies) and is recharged only when the battery is depleted. Let $\gamma = \lceil \frac{\Delta}{\rho+2d} \rceil$. Let $Hole_{max}(t) = \mathrm{Max}\{Hole(\texttt{Slide}, t') : t' \geq t\}$, be the maximum coverage hole from time t on. Clearly we have: $Coverage_{min}(\texttt{Slide},, t) = n - Hole_{max}(\texttt{Slide},, t)$. Let $Dis_{max}(t) = \mathrm{Max}\{Disconnect(\texttt{Slide}, t') : t' \geq t\}$ be the largest disconnection interval experienced in the network from time t on.

Theorem 9. *Let $n \geq 2\gamma$. Then, under the* Slide *strategy, for any $t \geq t_I + d\ \gamma$,*

$$Hole_{max}(\texttt{Slide}, t) = n - \gamma, Coverage_{min}(\texttt{Slide}, t) = \gamma$$

$$Dis_{max}(\texttt{Slide}, t) = (2n - 3\gamma)(\rho + d) + (\gamma - 1)d$$

Proof. After initialization, the robot starts moving along the line starting from x_1; let x_{j+1}, $j \geq 1$ be the first node encountered by R needing recharging. From the time x_{j+1} was charged during initialization to the end of the initialization precisely $\rho j + d\ j$ time units have elapsed; the robot will reach x_{j+1} again (and find it depleted) after an additional $d\ j$ time units. In other words, R finds x_{n-j} empty $\rho j + 2\ d\ j$ time units after it charged it during initialization. This means that $\rho j + 2\ d\ j \geq \Delta$. That is $j \geq \frac{\Delta}{\rho+2d}$. This also implies that R reaches $x_j + 1$ at time $t' = t_I + \lceil \frac{\Delta}{\rho+2\ d} \rceil = t_i + \gamma$; but at this time every x_i with $i \geq j+1$ is no longer operational. Notice that from this time on, starting with x_{j+1}, once R has finished recharging x_{j+l} and move to the next one, x_{j-1} becomes depleted. In other words, from time t' to the time R reaches and charger x_n the number of coverage holes is $n - \gamma$. It is not difficult to verify that the behavior of the system is at this point symmetric and periodic: the robot, starting at x_n will move along the line, and the first node encountered by R needing recharging is precisely $x_{n-(j+1)}$, and once it is charged, the number of coverage holes is again $n - \gamma$ and it stays like that until R reaches x_1 again.

Consider now disconnection time. It can be shown that, after the system becomes periodic, node x_γ is among the nodes experiencing the highest disconnection time. Let x_γ become depleted at some time $t > 2t_I$. At this time, the robot moves to $x_{\gamma+2}$ (from $x_{\gamma+1}$). Before the robots passes again by x_γ it will have to charge all the nodes until the end of the line spending $(n-\gamma-1)(\rho+d)$ time units, it will change direction and it will pass by $(\gamma-1)$ charged nodes spending $(\gamma-1)d$ time units, at which point it will encounter again nodes in need of charge. Before reaching x_γ, the robot will have to charge $(n-2\gamma+1)$ such nodes, for an additional $(n-2\gamma+1)(\rho+d)$ units of time. The result then follows. □

Examining the other two strategies under the same bootstrap scenario, we obtain the following:

Theorem 10. *Let $n \geq 2\gamma$ and $\rho > 2$. Then, the robot's behavior under the* On-Request *and* Straight *strategies is the same as under the* Slide *strategy.*

Proof. In the initialization, the robot clearly acts in the same way under all three strategies: starting from x_n, after charging the current node, it immediately moves to the next. From the time t' the robot R leaves x_n fully charged to the time it completes charging x_1, precisely $(n-1)\rho + (n-1)d$ time units have elapsed. On the other hand, x_n becomes again depleted after Δ time units and its requests would reach x_1 after an additional $n-1$ time units. Observe that, since $\rho > 2$, we have $2(\rho+d-1) > \rho+2d$; hence

$$(\rho+d-1)\lceil\frac{2\Delta}{\rho+2d}\rceil > \Delta$$

that is $(\rho+d-1)\gamma > \Delta$ Since $n > 2\gamma$, this implies that $(\rho+d-1)\ n > \Delta$; that is, $(n-1)(\rho+d) \leq \Delta + (n-1)$.

In other words, the request from x_n has reached the robot when it ends charging x_1. Hence, also in ON-REQUEST and STRAIGHT strategies: (1) the robot will start immediately moving from x_1 towards x_n; (2) the first $\gamma-1$ nodes it encounters do not need to be recharged while $x_{\gamma+1}$ does: (3) once R finishes charging $x_{\gamma+1}$, every node x_i, $n \geq i > \gamma$ is depleted; hence R will continue to move towards x_n charging every node it visits. Summarizing, R will behave during all this time as under the SLIDE strategy. Furthermore, once R finishes charging $x_{\gamma+1}$, also node $x_{\gamma-1}$ becomes depleted and originates a request that reaches R before or when it reaches x_n; note that, even if it reaches R before the robot reaches x_n, according to both STRAIGHT and ON-REQUEST strategies, R will still continue to move towards x_n. That is, by the time R has finished charging x_n, the system conditions and its behavior will be exactly like in the previous travel from x_1 to x_n, only specular (i.e., from x_n to x_1); in other words, the behavior of the system is at this point symmetric and periodic and exactly like under SLIDE. □

Hence all three strategies have the same performance.

Interestingly, for the special case $n = \gamma - 1$ we have the following.

Theorem 11. *Let $n = \gamma - 1$. Then, under the* SLIDE *strategy, for any $t \geq t_I$, $Hole_{max} = 0, Dis_{max}(t) = 0$.*

That is, for $n = \gamma - 1$, as soon as a node needs a recharge, the robot is there at that time starting a recharge. In other words, no hole is ever created and the network provides full coverage at all times. This is the *optimal* behavior and no other strategy, even if centralized and off-line, can do better for this network size.

7.4 Summary

We study three energy restoring strategies in linear arrangements running extensive simulations to assess their performance.

Interestingly, we make the counter-intuitive discovery that, among the two on-demand strategies (On-Request and Straight), the simpler Straight performs at least as well as the standard On-Request; furthermore, the Slide strategy, which requires absolutely no communications nor computations, performs as well as the two on-demand strategies with respect to all three metrics.

We also provide some analytical support to these experimental findings by studying a specific situation when the sensors start with initially empty batteries and threshold $\tau = 0$. In this case, we prove that the Slide strategy has better coverage performance that the other two strategies for almost all network sizes.

Chapter 8

Multiple Robots

To overcome some of the limitation of single chargers approaches, multiple mobile robots can be used to maximize the coverage size and reduce the disconnection time. In this chapter we study effective energy restoration strategies for $k > 1$ mobile robots in a sensor network and we investigate three different energy restoration strategies. The different strategies (Overpass, Sub-Segment and Sub-Ring), which are all simple, decentralized, and do not require neither communication nor computations, will be analyzed and compared.

8.1 Strategies

We propose two approaches: one consists of a collaborative restoration of the network, the other is based on the idea of partitioning the network and giving each robot the responsibility for one portion.

- **Overpass Strategy**

  The first approach (Overpass) is quite intuitive: the robots are initially placed at roughly equal intervals along a predefined circular ordering of the sensors, and they perpetually and bindlindly move in the same direction, stopping to service any node whose energy level falls below the threshold. In case a robot encounters another robot that is recharging a node, it just overpasses it and continues in the same direction looking for another sensor in need.

- **Partitioning Strategies**

  In the second approach the circular ordering is partitioned into k regions (where k is the number of available robots), each containing approximately the same number of nodes. Each region is assigned to a different robot that becomes responsible for energy restoration for that region, independently of the rest of the network and of the other robots. Within this approach, we consider the choice of each robot using a fixed circular order of the nodes in its region (SUB-RING), or a fixed linear order of the nodes in its region (SUB-SEGMENT). Clearly, when considering the robot operating in its region, we could have used any of the restoring strategies studied in the rest of the thesis. Given the excellent results offered by BLIND (for the circular ordering) and SLIDE (for the linear arrangement), we use those within our SUB-RING and SUB-SEGMENT strategies.

  The fact that SUB-RING is more efficient than SUB-SEGMENT is evident by the results of Chapters 6 and 7. In fact, it is clear and intuitive that having the robot perpetually move on a ring has benefits with respect to having it move back and forth on a segment of the same size. However, both types of partitions are meaningful because in certain situations the SUB-RING strategy might not be feasible This is the case, for example, when the original topology is indeed a ring (i.e., the perimeter of a closed region) and the partitioning of this ring results in a collection of segments. Thus the need to study also the SUB-SEGMENT one.

Let us stress again that all strategies are simple and blind; that is, they do not require communication, memory, nor global information. The OVERPASS is a blind collaborative strategy along a ring where the robots collectively and perpetually traverse the network, the Sub-Segment (resp. Sub-Ring) is a blind strategy where each robot moves independently and autonomously along a predefined linear arrangement (resp. circular arrangement) that constitutes a partition of the network.

8.2 Experimental Analysis

In this Section we examine experimentally the effectiveness of these three recharging strategies for circular sensor networks.

8.2.1 Simulation Environment

The environment is the same as the one studied in the previous chapters.

Table 8.1 presents the values considered for each parameter.

Table 8.1: Environmental Parameters

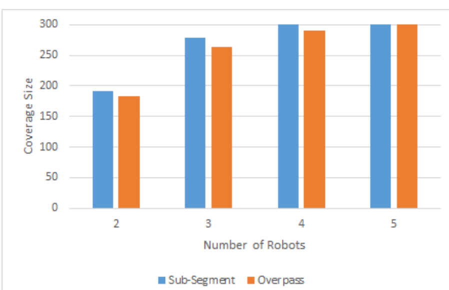
Parameters	Values
Number of nodes n	100, 200, 300, 400, 500
Node Capacity Δ	1000,2000, 3000, 4000
Threshold τ	30% of Δ
Recharging Time ρ	10, 20, 30, 40, 50
Number of robots k	2,3,4,5

Each sensor x has initially an amount of energy level randomly chosen in the range [τ, Δ]. In each run, we record the average disconnection time of the sensors, and the coverage size achieved by the strategy, and we average these values over 10 executions. We do not consider charging latency since, in all the results obtained in the rest of the thesis, we clearly observed that its behavior is very similar to the one of disconnection time and its analysis would not bring any new observation. Every execution is let to run for a sufficient amount of time that allows the system to stabilize (stabilization will be discussed later). All results have been obtained with 98 % confidence level.

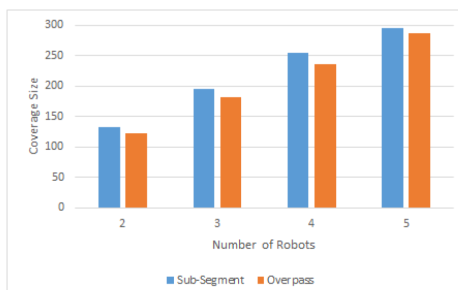
Clearly, the use of more robots improves the performance of the system: coverage increases and disconnection time decreases. This happens until using k robots would result in perfect coverage (and thus no disconnection time). At this point, the use of $k+1$ robots would obviously have the same result. Let us call such a k *saturation value.* We are interested in determining which of the two techniques has a better effectiveness speedup (with respect to the near-optimal strategy with one robot) using $k > 1$ robots and under what conditions of the system parameters.

8.2.2 Comparison between Overpass and Sub-segment

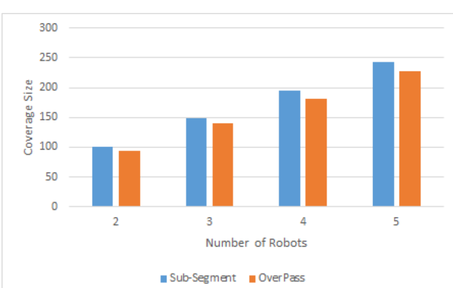
In the following we analyze Overpass and Sub-Segment with respect to coverage size and disconnection time, when varying the usual network parameters.



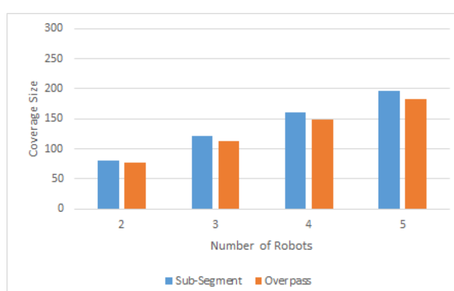
(a) $\rho = 20$



(b) $\rho = 30$

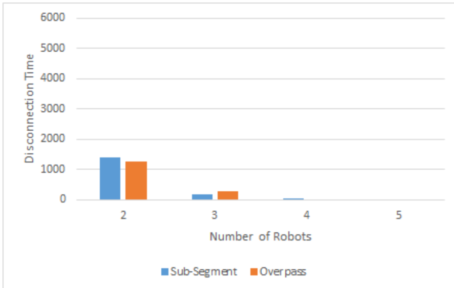


(c) $\rho = 40$

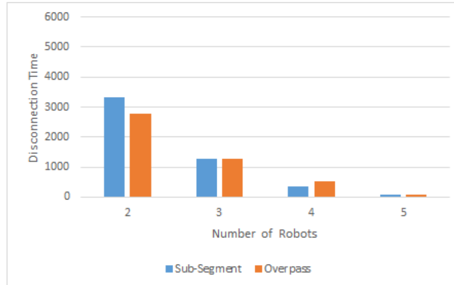


(d) $\rho = 50$

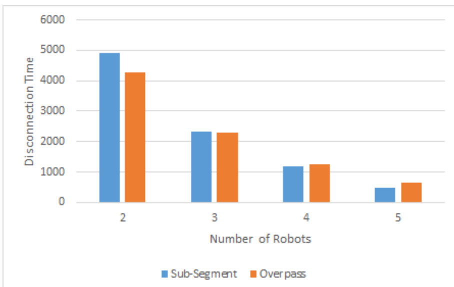
Figure 8.1: Effect of ρ on Coverage Size for the two strategies ($n = 300$, $\Delta = 2000$).



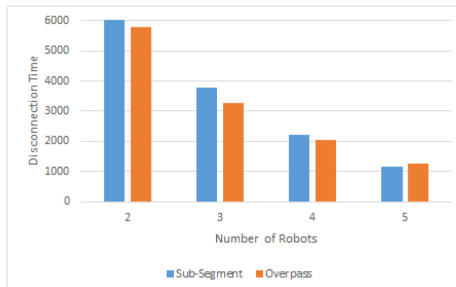
(a) $\rho = 20$



(b) $\rho = 30$



(c) $\rho = 40$



(d) $\rho = 50$

Figure 8.2: Effect of ρ on Disconnection Time for the two strategies ($n = 300$, $\Delta = 2000$).

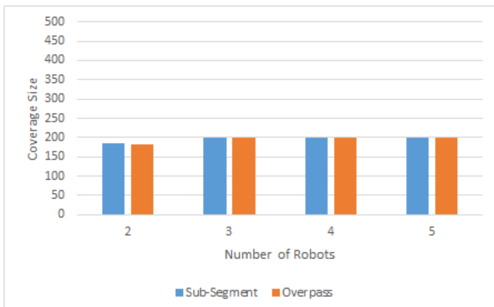
8.2.2.1 Effects of ρ

Consider first the impact of charging time ρ on disconnection time and coverage size for both strategies varying the number of robots, when fixing n (see Figures 8.1 and 8.2, where results are shown for some choices of the parameters; the results for all the other parameters' combinations are consistent with these).

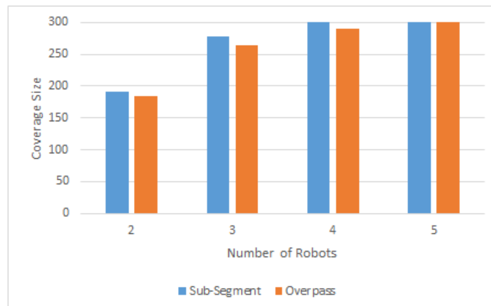
As expected, a higher charging time implies a decrease in both performance measures. What is interesting to note is that, in terms of coverage size, Sub-segment always outperforms Overpass. Furthermore, the advantages of Sub-segment are more and more evident as the number of robots increases.

On the other hand, we observe a different phenomenon when looking at disconnection time. In fact, Overpass reaches a better disconnection time than Sub-segment when few robots are employed, but the behavior reverses when the number of robots increases. See for example, Figure 8.2 and, in particular, observe the case $\rho = 30$. In this setting with

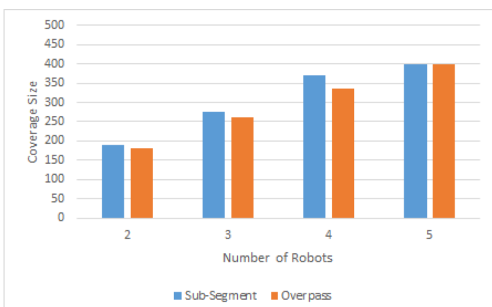
2 robots Overpass clearly achieves a better disconnection time, with three robots the two strategies seem to be equivalent, with four robots Sub-segment becomes better, and with five robots we are close to the saturation value for both. Similar behavior occurs for different values of ρ. We call the value of k corresponding to this behavior, the *switching value*.



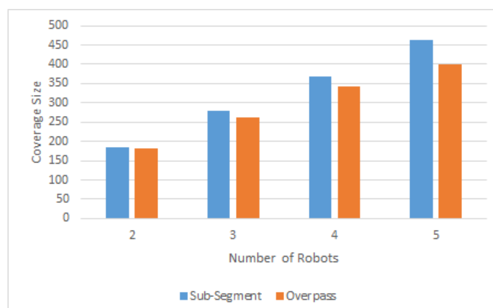
(a) $n = 200$



(b) $n = 300$



(c) $n = 400$



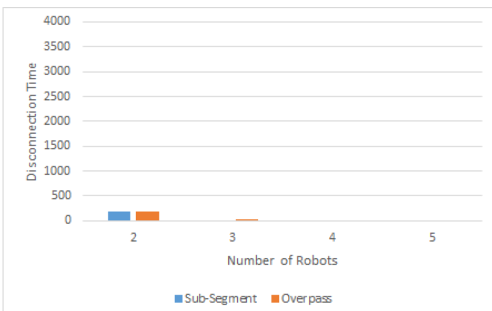
(d) $n = 500$

Figure 8.3: Effect of n on Coverage Size for the two strategies ($\Delta = 2000$, $\rho = 20$).

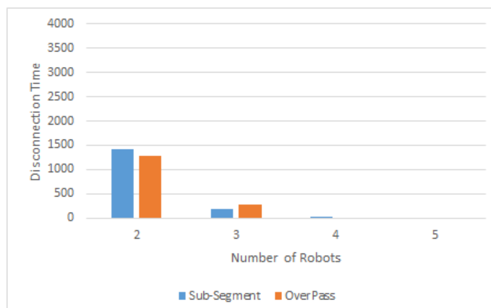
8.2.2.2 Effects of Network Size n

We investigate the effect of network size with various number of robots on disconnection time and coverage size for both strategies. Figures 8.3 and 8.4 show the trend in the disconnection time and coverage size for some choice of the parameters. The results for all the other parameters' combinations are consistent with these.

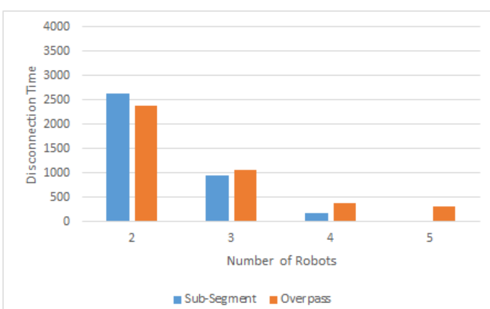
Again, we observe that Sub-segment is always better than Overpass in terms of coverage size, until they become equal at the saturation value. The behavior of disconnection time, however, is quite different and confirms what already observed when varying the charging time: Overpass outperforms Sub-segment when fewer robots are employed and the reverse behavior occurs when the number increases. With the growth of the network, this phenomenon becomes more and more evident and the switching value increases. In fact, for $n = 200$ the two strategies seem to be equivalent, the saturation value is reached already with 3 robots, and clearly there is no evidence of a switching behavior; for $n = 300$ and $n = 400$ Overpass is better than Sub-segment when employing 2 robots, but already with 3 robots the behavior is reversed; finally, with $n = 500$ Sub-segment becomes better than Overpass when using 4 robots.



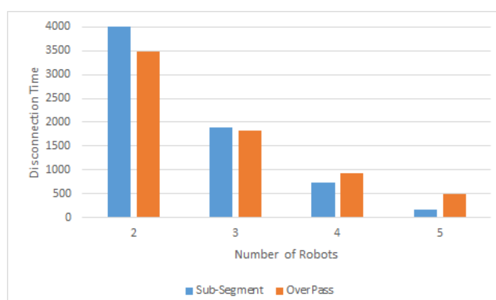
(a) $n = 200$



(b) $n = 300$



(c) $n = 400$

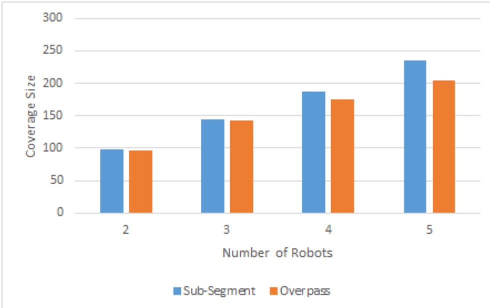


(d) $n = 500$

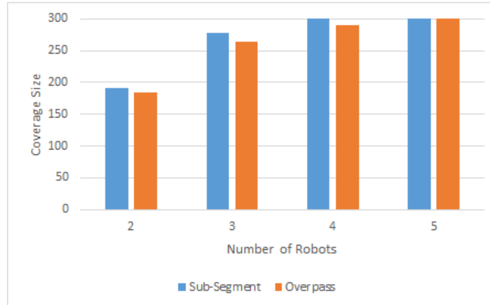
Figure 8.4: Effect of n on Disconnection Time for the two strategies ($\Delta = 2000$, $\rho = 20$).

8.2.2.3 Effects of Node Capacity Δ

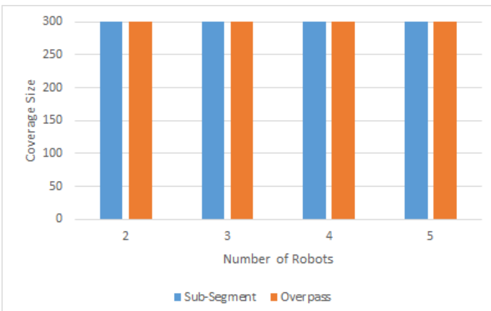
We finally analyze the effect of the node capacity Δ on disconnection time and coverage size for the Sub-segment and Overpass strategies, varying the number of robots. Figures



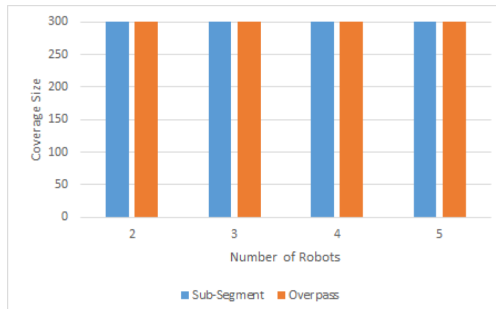
(a) $\Delta = 1000$



(b) $\Delta = 2000$



(c) $\Delta = 3000$



(d) $\Delta = 4000$

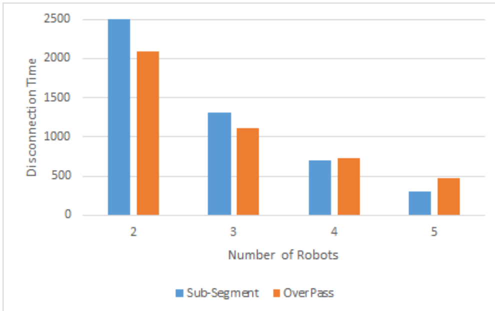
Figure 8.5: Effect of Δ on Coverage size for the two strategies ($n = 300$, $\rho = 20$).

8.5 and 8.6 are obtained when $n = 300$, $\rho = 20$, and the distance between nodes is unitary. The results for all the other parameters' combinations are consistent with these.

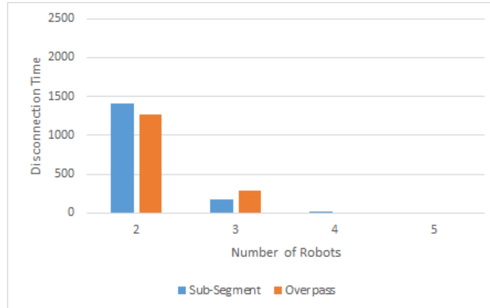
Clearly, for both strategies, as the node capacity increases, disconnection time decrease, whereas coverage size increases. As already observed earlier, also in this case we clearly see that SUB-SEGMENT is always better than OVERPASS for coverage size, while OVERPASS initially outperforms SUB-SEGMENT until the *switching value*, when the opposite becomes true.

8.2.2.4 Discussion

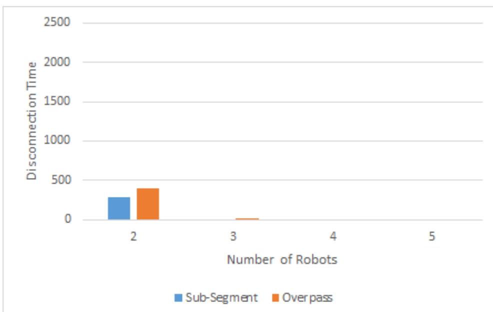
The goal of the experimental study was to compare the two techniques and to understand under what conditions of the parameters one is preferable than the other. The main observation we can draw from our analysis are the following:



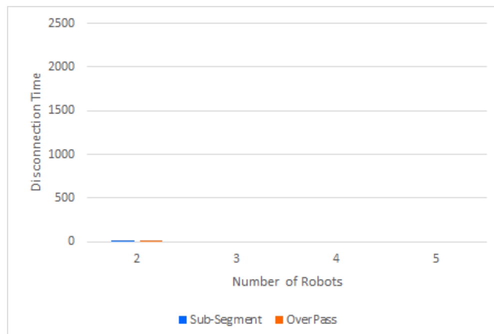
(a) $\Delta = 1000$



(b) $\Delta = 2000$



(c) $\Delta = 3000$



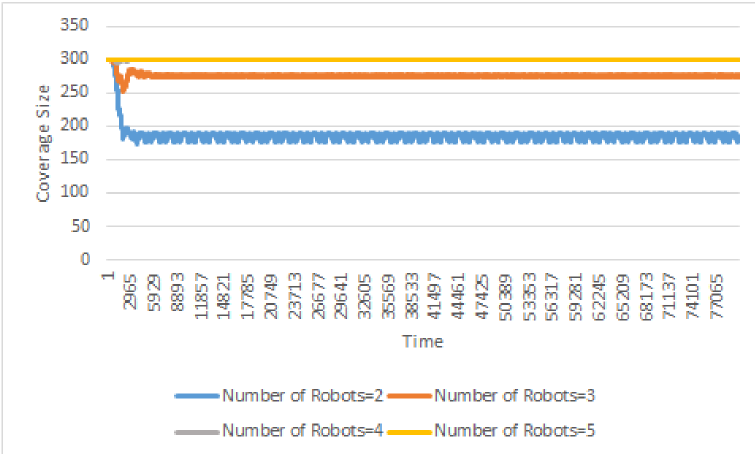
(d) $\Delta = 4000$

Figure 8.6: Effect of Δ on Disconnection Time for the two strategies ($n = 300$, $\rho = 20$).

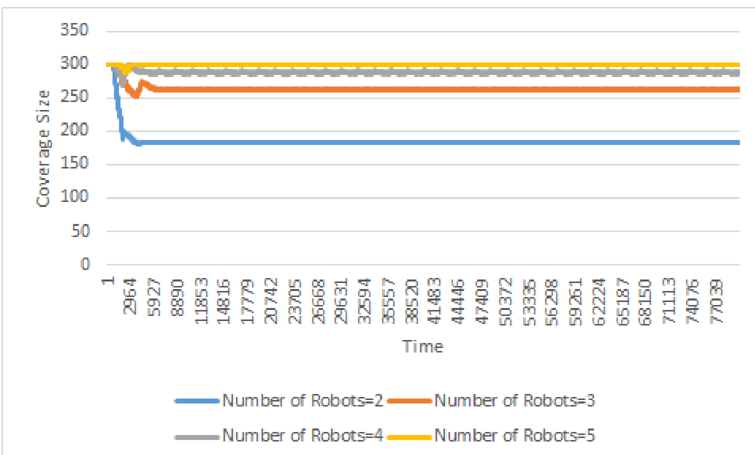
Stability Under both OVERPASS and SUB-SEGMENT, the coverage size quickly reaches an almost-periodic behavior, oscillating within a very small range. Figure 8.7 and 8.8 display the behavior of coverage size in time for particular scenarios, showing that, after a short transient, the system stabilizes and the amplitude of the oscillation is quite small with both strategies. As we can see from the figures, fixing the other parameters, the transient is longer as n grows. Figure 8.7 shows the case of for $n = 300$ (for $\Delta = 2000$, $\rho = 20$), Figure 8.8 considers $n = 400$ for the same choice of the other parameters. The general behavior is quite consistent for all other choices of the parameters.

In correspondence to the formation of this periodic behavior, the average disconnection time stabilizes as well.

Saturation and Coverage Rate Let $cov(k)$ indicate the coverage size obtained with k robots. The saturation value is defined as the minimum k when $cov(k) = cov(k + 1) = n$.



(a) Sub-Segment

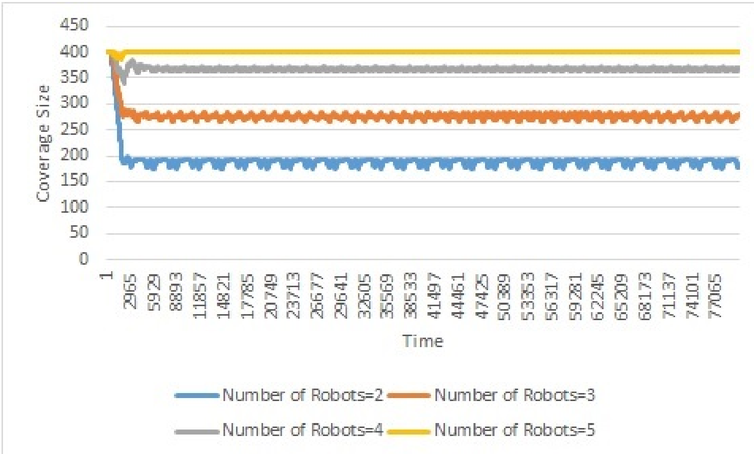


(b) Overpass

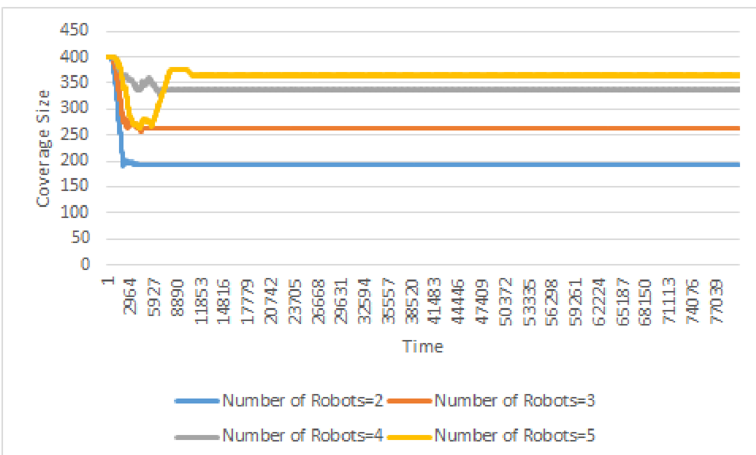
Figure 8.7: Oscillating behaviour of coverage size under the two strategies for $n = 300$, $\Delta = 2000$, $\rho = 20$.

On the basis of extensive simulation, we noticed that, with the SUB-SEGMENT strategy, saturation occurs when $\frac{n}{k}$ approaches $\frac{\Delta}{\rho}$, more precisely, this happens when $k \approx \frac{n\rho}{\Delta}+1$. This is clearly in line with the results obtained for the SLIDE strategy in linear arrangements, here applied to each partition.

The saturation value for OVERPASS is always slightly higher than the one for SUB-SEGMENT. This can be seen also observing the rate at which full coverage is reached by OVERPASS and SUB-SEGMENT as shown in Figure 8.9, where the percentage of the ring covered once the system stabilizes is shown under the two strategies, varying the number



(a) Sub-Segment



(b) Overpass

Figure 8.8: Oscillating behaviour of coverage size under the two strategies for $n = 400$, $\Delta = 2000$, $\rho = 20$.

of robots. We can observe again the crucial role played by the relationship between size of the ring, number of robots, charging time, and node capacity. As we can see from the figure, when $\frac{n}{k}$ is smaller than $\frac{\Delta}{\rho}$, the difference between SUB-SEGMENT and OVERPASS is null or rather small, when $\frac{n}{k}$ approaches $\frac{\Delta}{\rho}$ it is at its maximum, it then decreases again going towards saturation.

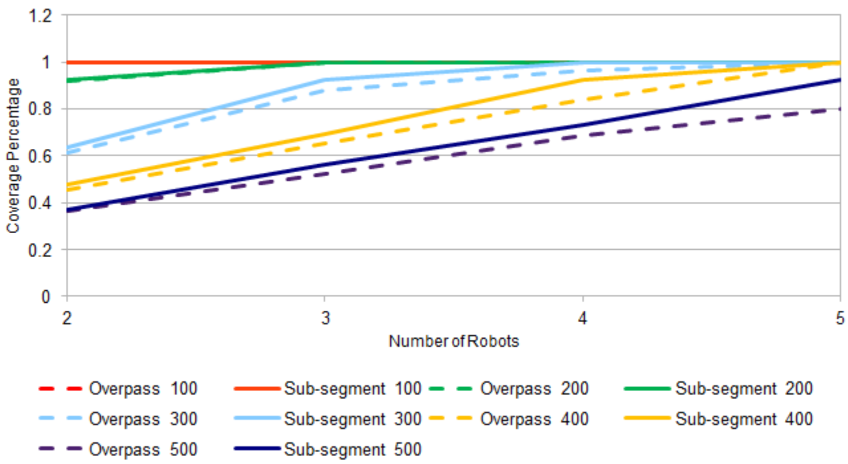


Figure 8.9: Rate of coverage varying n for $\Delta = 2000$, $\rho = 20$.

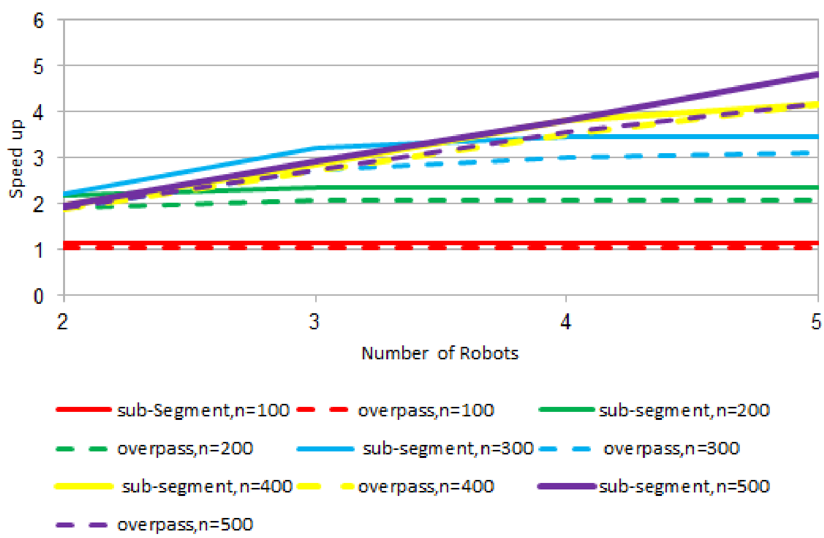


Figure 8.10: Speedup of the two strategies varying n, for $\Delta = 2000$, $\rho = 20$.

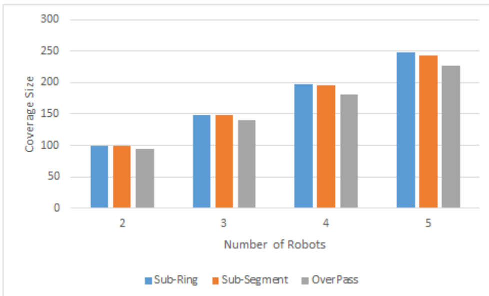
Speedup We measured the ratio: $\frac{cov(k)}{cov(1)}$ to evaluate the effect on coverage size of using our strategies with k robots rather than the near-optimal strategy of Chapter 7 for a single robot. We observe that, with SUB-SEGMENT, the coverage size increases, by a factor of k with respect to the near-optimal strategy for a single robot; this continues with the increasing of k until saturation. Disconnection time decreases accordingly. If the ring is small with respect to $\frac{\Delta}{\rho}$, the saturation value is reached quickly and using more robots does not bear any effect. The extreme case in Figure 8.10 is for $n = 100$ (and $\frac{\Delta}{\rho} = 100$), where a single robot would have already achieved full coverage and zero disconnection time; in this case, obviously, the speedup ratio is 1 regardless of the number of robots employed. For larger values of n (see, for example, $n = 500$ in the figure), the speedup is close to optimal, that is: $\frac{cov(k)}{cov(1)} \approx k$. Note that the speedup of SUB-SEGMENT with respect to coverage is always better than the one of OVERPASS.

Switching value and Clustering Effect While it is very clear from our experiments that SUB-SEGMENT always outperforms OVERPASS in terms of coverage size, this is not always the case for disconnection time. In fact, as observed earlier, it seems that OVERPASS achieves better disconnection time than SUB-SEGMENT when few robots are available, while the opposite happens when more robots are used.

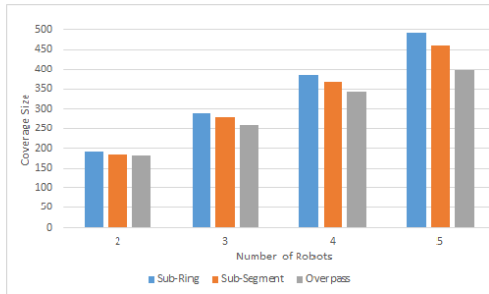
Our simulations always start with the robots being equi-distributed along the ring, with distance between any two being approximately n/k (where k is the number of robots). For SUB-SEGMENT this equi-distribution is maintained by construction (each segment is independently served by a robot); on the other hand, in OVERPASS the robots do not generally preserve this nice feature and they end up at variable distance from each other, possibly rendering the size of the maximum segment between any two robots greater than n/k. Moreover, robots sometimes get close to each other forming consecutive groups (*Clustering Effect*). When several robots find themselves in adjacent nodes, they tend to maintain a cluster formation, creating even larger segments.

Intuitively, the larger is a stretch of sensors between any two robots, the higher is the worst case disconnection time of a node waiting for a robot. Since the clustering phenomenon intensifies as the number of robots grow, it is then reasonable to speculate that the switching behavior is to be attributed to the clustering effect. In fact, when fewer robots are employed, in OVERPASS they preserve an almost equi-distribution and they achieve a better disconnection time than in SUB-SEGMENT; when however more robots are used, they tend to form clusters (in the limit case a single one) significantly decreasing the performance of OVERPASS in terms of disconnection time.

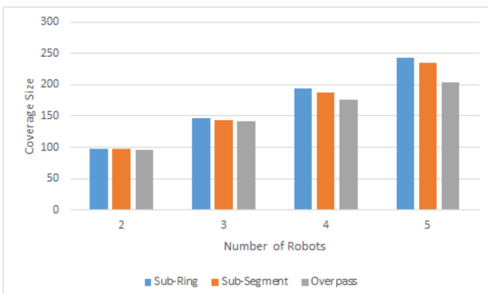
While the phenomenon is clearly observed in our experiments, we leave for future studies a precise analysis of this behavior, that seems to be quite complex.



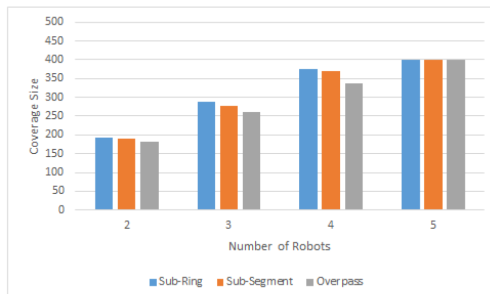
(a) $\rho = 40,\ n = 300, \Delta = 2000$



(b) $\rho = 20,\ n = 500, \Delta = 2000$



(c) $\rho = 20,\ n = 300, \Delta = 1000$



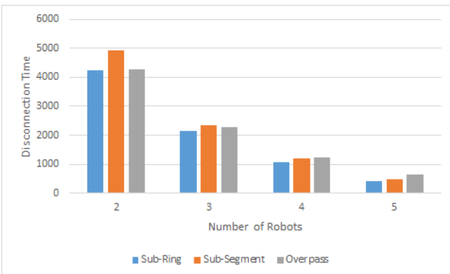
(d) $\rho = 20,\ n = 400, \Delta = 2000$

Figure 8.11: Coverage Size of the three strategies for some scenarios.

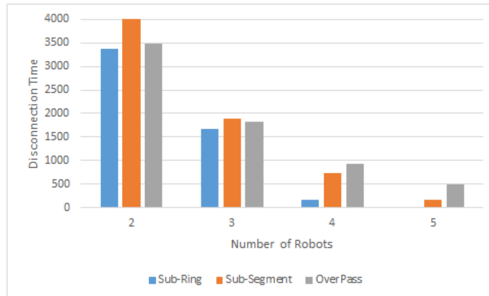
8.2.3 Sub-ring vs Overpass and Sub-Segment

In the Sub-ring strategy, we still partition the network into k regions but, instead of identifying a linear arrangement in each region and applying the Slide strategy for each robot, as we do with Sub-Segment, we define a circular arrangement in each region and we use the Blind strategy in each of them (Sub-Ring). As already mentioned, we know from the results of Chapters 6 and 7 that using Blind we will never achieve a worse performance than using Slide. So, clearly, whenever feasible, Sub-ring is preferable to Sub-segment. Still, there might be situations where such an approach is not feasible (e.g., when the topology is an actual ring: that is, the perimeter of a closed region). We are now interested in observing the relationship between the three strategies and, in particular, how

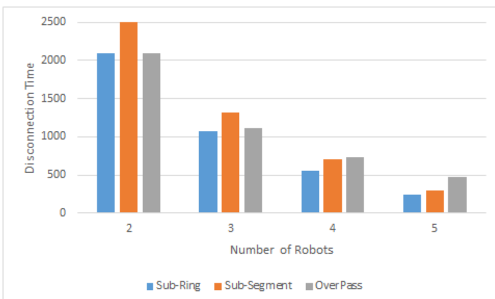
Sub-Ring behaves compared to Overpass.



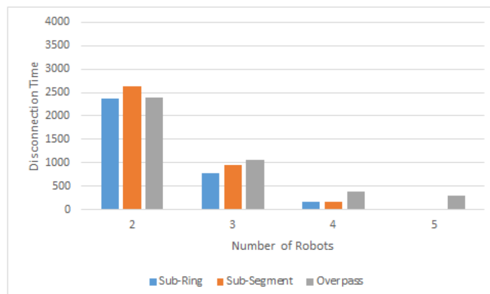
(a) $\rho = 40$, $n = 300$,$\Delta = 2000$



(b) $\rho = 20$, $n = 500$,$\Delta = 2000$



(c) $\rho = 20$, $n = 300$,$\Delta = 1000$



(d) $\rho = 20$, $n = 400$,$\Delta = 2000$

Figure 8.12: Disconnection Time of the three strategies for some scenarios.

The comparison turns out to be rather simple; in fact, under all choices of the parameters we always observe that Sub-Ring outperforms Overpass both in terms of disconnection time and coverage size.

Since the results are very consistent and do not present any interesting deviation from this behavior, we show in Figures 8.11 and 8.12 the comparison of the three only for some settings. Speedup, saturation, and coverage rate of Sub-ring behaves very similarly to the ones of Sub-segment.

What we observe is that Sub-Ring is always better than both Overpass and Sub-segment under all choices of the parameters. It becomes equivalent only when they both reach saturation (in Figure 8.11, we observe coverage saturation for the combination of parameters $\rho = 20$, $n = 400$,$\Delta = 2000$).

8.3 Summary

In this Chapter we studied three strategies for energy restoration which employ multiple robots: a collaborative one (Overpass), and two based on portion of work (Sub-segment and Sub-ring). We first compared Sub-segment with Overpass. With respect to *minimum coverage size*, our experiments indicate is that Sub-segment is always better (i.e., it has larger coverage) than Overpass. With respect to maximum disconnection time, the results depend on the number of robots: Sub-segment outperforms Overpass when a sufficient number of robots is deployed; when fewer robots are available, Overpass achieves a better (i.e., smaller) maximum disconnection time. More importantly, the Sub-segment strategy achieves a near *optimal speed-up* in effectiveness: the coverage size increases, and the disconnection time decreases, by a factor of k with respect to the near-optimal strategy for a single robot; this continues with the increasing of k until the coverage becomes and stays complete (i.e., no sensing hole ever occurs). We determine the smallest number k of robots that makes this happen. In other words, the effectiveness of Sub-segment is near optimal in spite of its simplicity and of not requiring communication or computations.

The comparison between Overpass and Sub-ring, on the other hand, has rather different results. In fact, Sub-ring is always better than both Sub-segment and Overpass both in terms of disconnection time and coverage size.

Chapter 9

Conclusions

9.1 Summary

The purpose of this research is to promote effective and efficient energy restoration strategies in wireless sensor networks to maintain an effective coverage. In this thesis, a variety of different approaches have been proposed to study this problem employing one or more robots to charge (or replace) the sensors' batteries.

First, we introduced an on-demand decentralized strategy (ON-DEMAND), where a robot visits the sensors in a predefined circular order. We studied it both analytically and experimentally analyzing the impact of various network parameters on coverage size, disconnection time, and charging latency. We then introduced an optimal centralized approach (CENTRALIZED) as a benchmark to assess how close to optimal our on-demand strategy is, and we discovered that, for sufficiently large networks, ON-DEMAND is actually optimal. We then proposed an even simpler strategy (BLIND), and we experimentally showed that it is as efficient as the other two, and thus preferable. The results above apply to any sensor network; we then turned our attention to a common special topology: a linear arrangement of sensors. We proposed three strategies: SLIDE, STRAIGHT, ON-REQUEST and we compared them experimentally discovering, once again, that the simplest approach is also the best, in most cases. We finally considered the availability of multiple robots and we compared two strategies based on division of work among the robots (Sub-Segment and Sub-Ring), where each robot employs one of the techniques studied in the rest of the thesis on a portion of the network, with a global collaborative approach (Overpass) and we experimentally showed under what conditions each is preferable.

The main general finding that springs from our investigation is that simple solutions are often as good as more sophisticated ones. In fact, a totally blind strategy where a robot simply moves around restoring energy on its way turns out to be as efficient as the best possible centralized solution for most networks.

9.2 Future Work

In this thesis, a range of issues have been explored. The results of our investigation open several problems and many potential directions for future research. Here, we identify several noteworthy directions to expand our current research, both in extending mechanisms and metrics we are building and in developing more extensive validation to demonstrate the value of our models. Some interesting directions are indicated below

Our studies have been carried out in an abstract setting, with simplifying assumptions. The next step would be to remove some of these simplifications making the environment more realistic. For example, when considering a circular (or linear) ordering of the sensors, we assumed that the travel time between neighbors is always the same. The issue of different travel times (e.g., proportional to the number of hops) required by the robot between different pairs of neighbors in the cyclic order is an interesting and important topic for further study.

Another interesting and realistic direction would be to assume batteries with different capacities and different charging rates, as well as to assign priorities to batteries depending on their importance. All these are interesting directions for future study.

All results have been derived through experimental analysis. For some of them we also provided a full theoretical validation; this is the case, for example, of the On-Demand strategy. For others, we only indicated some theoretical support in special situations (e.g., Blind). The full theoretical analysis of all the strategies is left for future work.

While from a practical point of view large networks represent the most realistic setting, particularly interesting from a theoretical perspective is the case of small networks, which seems to be more difficult to study analytically. All our theoretical results are valid for sufficiently large networks; as part of the future work, it would be interesting to understand the behavior of the proposed strategies on small ones.

Although in stationary networks topological structures have been often the object of investigation as they reveal interesting results in sensor networks, the investigation of dynamical topological structures and their effects on the sensing holes and disconnection time is an open area of research.

References

[1] S. Akbari, "Energy harvesting for wireless sensor networks review," in *2014 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2014, pp. 987–992.

[2] F. Akhtar and M. H. Rehmani, "Energy replenishment using renewable and traditional energy resources for sustainable wireless sensor networks: A review," *Renewable and Sustainable Energy Reviews*, vol. 45, pp. 769–784, 2015.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[4] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad Hoc Networks*, vol. 7, no. 3, pp. 537–568, 2009.

[5] G. Anastasi, M. Conti, A. Passarella, and L. Pelusi, "Mobile-relay forwarding in opportunistic networks," *Adaptation and Cross Layer Design in Wireless Networks*, vol. 21, p. 389, 2008.

[6] C. M. Angelopoulos, S. Nikoletseas, T. P. Raptis, C. Raptopoulos, and F. Vasilakis, "Efficient energy management in wireless rechargeable sensor networks," in *Proceedings of the 15th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2012, pp. 309–316.

[7] C. M. Angelopoulos, S. Nikoletseas, T. P. Raptis, C. Raptopoulos, and F. Vasilakis, "Improving sensor network performance with wireless energy transfer," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 20, no. 3, pp. 159–171, 2015.

[8] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda *et al.*, "A line in the sand: a wireless sensor network for target detection, classification, and tracking," *Computer Networks*, vol. 46, no. 5, pp. 605–634, 2004.

[9] F. Arvin, K. Samsudin, and A. R. Ramli, "Swarm robots long term autonomy using moveable charger," in *2009 International Conference on Future Computer and Communication*, 2009, pp. 127–130.

[10] S. Basagni, A. Carosi, E. Melachrinoudis, C. Petrioli, and Z. M. Wang, "Controlled sink mobility for prolonging wireless sensor networks lifetime," *Wireless Networks*, vol. 14, no. 6, pp. 831–858, 2008.

[11] R. Beigel, J. Wu, and H. Zheng, "On optimal scheduling of multiple mobile chargers in wireless sensor networks," in *Proceedings of the 1st International Workshop on Mobile Sensing, Computing and Communication*, 2014, pp. 1–6.

[12] A. Cerpa and D. Estrin, "Ascent: Adaptive self-configuring sensor networks topologies," *IEEE Transactions on Mobile Computing*, vol. 3, no. 3, pp. 272–285, 2004.

[13] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," *Wireless networks*, vol. 8, no. 5, pp. 481–494, 2002.

[14] G. A. Cohn, "Snupi: Sensor network utilizing powerline infrastructure," Ph.D. dissertation, 2015.

[15] H. Dai, L. Jiang, X. Wu, D. K. Yau, G. Chen, and S. Tang, "Near optimal charging and scheduling scheme for stochastic event capture with rechargeable sensors," in *2013 IEEE 10th International Conference on Mobile Ad-Hoc and Sensor Systems (MASS)*, 2013, pp. 10–18.

[16] H. Dai, X. Wu, L. Xu, G. Chen, and S. Lin, "Using minimum mobile chargers to keep large-scale wireless rechargeable sensor networks running forever," in *2013 22nd International Conference on Computer Communication and Networks (ICCCN)*, 2013, pp. 1–7.

[17] A. Dewan, S. U. Ay, M. N. Karim, and H. Beyenal, "Alternative power sources for remote sensors: A review," *Journal of Power Sources*, vol. 245, pp. 129–143, 2014.

[18] A. Drenner and N. Papanikolopoulos, "Docking station relocation for maximizing longevity of distributed robotic teams," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2006, pp. 2436–2441.

[19] W. Elgenaidi, T. Newe, E. OConnell, D. Toal, and G. Dooly, "Secure and efficient key coordination algorithm for line topology network maintenance for use in maritime wireless sensor networks," *Sensors*, vol. 16, no. 12, p. 2204, 2016.

[20] I. Farris, L. Militano, A. Iera, A. Molinaro, and S. C. Spinella, "Tag-based cooperative data gathering and energy recharging in wide area rfid sensor networks," *Ad Hoc Networks*, vol. 36, pp. 214–228, 2016.

[21] E. Fasolo, M. Rossi, J. Widmer, and M. Zorzi, "In-network aggregation techniques for wireless sensor networks: a survey," *IEEE Wireless Communications*, vol. 14, no. 2, 2007.

[22] L. Fu, P. Cheng, Y. Gu, J. Chen, and T. He, "Optimal charging in wireless rechargeable sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 1, pp. 278–291, 2016.

[23] L. Fu, L. He, P. Cheng, Y. Gu, J. Pan, and J. Chen, "Esync: Energy synchronized mobile charging in rechargeable wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7415–7431, 2016.

[24] C. F. García-Hernández, P. H. Ibarguengoytia-Gonzalez, J. García-Hernández, and J. A. Pérez-Díaz, "Wireless sensor networks and applications: a survey," *International Journal of Computer Science and Network Security*, vol. 7, no. 3, pp. 264–273, 2007.

[25] X. Gong, J. Zhang, D. Cochran, and K. Xing, "Barrier coverage in bistatic radar sensor networks: Cassini oval sensing and optimal placement," in *Proceedings of the 14th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2013, pp. 49–58.

[26] L. He, P. Cheng, Y. Gu, J. Pan, T. Zhu, and C. Liu, "Mobile-to-mobile energy replenishment in mission-critical robotic sensor networks," in *Proceeding of IEEE International Conference on Computer Communications (INFOCOM)*, 2014, pp. 1195–1203.

[27] L. He, L. Fu, L. Zheng, Y. Gu, P. Cheng, J. Chen, and J. Pan, "Esync: An energy synchronized charging protocol for rechargeable wireless sensor networks," in *Proceedings of the 15th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2014, pp. 247–256.

[28] L. He, Y. Gu, J. Pan, and T. Zhu, "On-demand charging in wireless sensor networks: Theories and applications," in *IEEE 10th International Conference on Mobile Ad-Hoc and Sensor Systems (MASS)*, 2013, pp. 28–36.

[29] L. He, L. Kong, Y. Gu, J. Pan, and T. Zhu, "Evaluating the on-demand mobile charging in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 14, no. 9, pp. 1861–1875, 2015.

[30] J. Hsu, S. Zahedi, A. Kansal, M. Srivastava, and V. Raghunathan, "Adaptive duty cycling for energy harvesting systems," in *Proceedings of the 2006 International Symposium on Low Power Electronics and Design*, 2006, pp. 180–185.

[31] B. Jan, H. Farman, H. Javed, B. Montrucchio, M. Khan, and S. Ali, "Energy efficient hierarchical clustering approaches in wireless sensor networks: A survey," *Wireless Communications and Mobile Computing*, vol. 2017, 2017.

[32] A. M. Jawad, R. Nordin, S. K. Gharghan, H. M. Jawad, and M. Ismail, "Opportunities and challenges for near-field wireless power transfer: A review," *Energies*, vol. 10, no. 7, p. 1022, 2017.

[33] I. Jawhar, N. Mohamed, and D. P. Agrawal, "Linear wireless sensor networks: Classification and applications," *Journal of Network and Computer Applications*, vol. 34, no. 5, pp. 1671–1682, 2011.

[34] L. Jiang, X. Wu, G. Chen, and Y. Li, "Effective on-demand mobile charger scheduling for maximizing coverage in wireless rechargeable sensor networks," *Mobile Networks and Applications*, vol. 19, no. 4, pp. 543–551, 2014.

[35] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet," *ACM SIGARCH Computer Architecture News*, vol. 30, no. 5, pp. 96–107, 2002.

[36] H. Jun, W. Zhao, M. H. Ammar, E. W. Zegura, and C. Lee, "Trading latency for energy in wireless ad hoc networks using message ferrying," in *Third IEEE International Conference on Pervasive Computing and Communications Workshops.*, 2005, pp. 220–225.

[37] T. Kamsuvan, "Power management in wireless sensor networks (wsns)," Master's thesis, Brunel University London, 2016.

[38] J. A. Khan, H. K. Qureshi, and A. Iqbal, “Energy management in wireless sensor networks: A survey,” *Computers & Electrical Engineering*, vol. 41, pp. 159–176, 2015.

[39] L. Khelladi, D. Djenouri, M. Rossi, and N. Badache, “Efficient on-demand multi-node charging techniques for wireless sensor networks,” *Computer Communications*, vol. 101, pp. 44–56, 2017.

[40] C. K. S. Kumar, R. Sukumar, and M. Nageswari, “Sensors lifetime enhancement techniques in wireless sensor networks a critical review,” *International Journal of Computer Science and Information Technology & Security*, vol. 3, no. 2, 2013.

[41] A. Kurs, A. Karalis, R. Moffatt, J. D. Joannopoulos, P. Fisher, and M. Soljačić, “Wireless power transfer via strongly coupled magnetic resonances,” *Science*, vol. 317, no. 5834, pp. 83–86, 2007.

[42] A. LaMarca, W. Brunette, D. Koizumi, M. Lease, S. B. Sigurdsson, K. Sikorski, D. Fox, and G. Borriello, “Making sensor networks practical with robots,” in *International Conference on Pervasive Computing*. Springer, 2002, pp. 152–166.

[43] X. Li, I. Lille, R. Falcon, A. Nayak, and I. Stojmenovic, “Servicing wireless sensor networks by mobile robots,” *IEEE Communications Magazine*, vol. 50, no. 7, 2012.

[44] X. Li, A. Nayak, and I. Stojmenovic, “Exploiting actuator mobility for energy-efficient data collection in delay-tolerant wireless sensor networks,” in *Fifth International Conference on Networking and Services, 2009 (ICNS’09)*, 2009, pp. 216–221.

[45] Z. Li, Y. Peng, W. Zhang, and D. Qiao, “J-roc: A joint routing and charging scheme to prolong sensor network lifetime,” in *19th IEEE International Conference on Network Protocols (ICNP)*, 2011, pp. 373–382.

[46] W. Liang, W. Xu, X. Ren, X. Jia, and X. Lin, "Maintaining large-scale rechargeable sensor networks perpetually via multiple mobile charging vehicles," *ACM Transactions on Sensor Networks (TOSN)*, vol. 12, no. 2, p. 14, 2016.

[47] C. Lin, Z. Wang, D. Han, Y. Wu, C. W. Yu, and G. Wu, "Tadp: Enabling temporal and distantial priority scheduling for on-demand charging architecture in wireless rechargeable sensor networks," *Journal of Systems Architecture*, vol. 70, pp. 26–38, 2016.

[48] C. Lin, Y. Wu, Z. Liu, M. S. Obaidat, C. W. Yu, and G. Wu, "Gtcharge: A game theoretical collaborative charging scheme for wireless rechargeable sensor networks," *Journal of Systems and Software*, vol. 121, pp. 88–104, 2016.

[49] L. Liu, B. Hu, and L. Li, "Energy conservation algorithms for maintaining coverage and connectivity in wireless sensor networks," *IET Communications*, vol. 4, no. 7, pp. 786–800, 2010.

[50] X. Lu, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Wireless charging technologies: Fundamentals, standards, and network applications," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1413–1452, 2016.

[51] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. Balan, "Mason: A multiagent simulation environment," *Simulation*, vol. 81, no. 7, pp. 517–527, 2005.

[52] A. Madhja, S. Nikoletseas, and T. P. Raptis, "Efficient, distributed coordination of multiple mobile chargers in sensor networks," in *Proceedings of the 16th ACM International Conference on Modeling, Analysis & Simulation of Wireless and Mobile Systems*, 2013, pp. 101–108.

[53] A. Madhja, S. Nikoletseas, and T. P. Raptis, "Distributed wireless power transfer in sensor networks with multiple mobile chargers," *Computer Networks*, vol. 80, pp. 89–108, 2015.

[54] A. Madhja, S. Nikoletseas, and T. P. Raptis, “Hierarchical, collaborative wireless charging in sensor networks,” in *2015 IEEE Wireless Communications and Networking Conference (WCNC)*, 2015, pp. 1285–1290.

[55] A. Madhja, S. Nikoletseas, and T. P. Raptis, “Hierarchical, collaborative wireless energy transfer in sensor networks with multiple mobile chargers,” *Computer Networks*, vol. 97, pp. 98–112, 2016.

[56] Y. Mei, C. Xian, S. Das, Y. C. Hu, and Y.-H. Lu, “Sensor replacement using mobile robots,” *Computer Communications*, vol. 30, no. 13, pp. 2615–2626, 2007.

[57] F. Michaud and E. Robichaud, “Sharing charging stations for long-term activity of autonomous robots,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002, p. 27462751.

[58] L. Militano, M. Erdelj, A. Molinaro, N. Mitton, and A. Iera, “Recharging versus replacing sensor nodes using mobile robots for network maintenance,” *Telecommunication Systems*, vol. 63, no. 4, pp. 625–642, 2016.

[59] L. Militano, M. Erdelj, A. Molinaro, N. Mitton, and A. Iera, “Recharging versus replacing sensor nodes using mobile robots for network maintenance,” *Telecommunication Systems*, vol. 63, no. 4, pp. 625–642, 2016.

[60] N. Mitton and D. Simplot-Ryl, *Wireless Sensor and Robot Networks From Topology Control to Communication Aspects*. World Scientific, 2014.

[61] N. Mohamed, J. Al-Jaroodi, I. Jawhar, and A. Eid, “Reliability analysis of linear wireless sensor networks,” in *2013 12th IEEE International Symposium on Network Computing and Applications (NCA)*, 2013, pp. 11–16.

[62] N. Mohamed, J. Al-Jaroodi, I. Jawhar, and S. Lazarova-Molnar, “Failure impact on coverage in linear wireless sensor networks,” in *2013 International Symposium on*

Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2013, pp. 188–195.

[63] D. Niyato, E. Hossain, and A. Fallahi, "Sleep and wakeup strategies in solar-powered wireless sensor/mesh networks: Performance analysis and optimization," *IEEE Transactions on Mobile Computing*, vol. 6, no. 2, pp. 221–236, 2007.

[64] E. Omar, P. Flocchini, and N. Santoro, "Online energy restoration by a mobile robot in a ring of sensors." in *14th International Wireless Communications & Mobile Computing Conference (IWCMC)*, 2018, pp. 654–659.

[65] E. Omar, P. Flocchini, and N. Santoro, "Online energy restoration by a mobile robot in a ring of sensors." in *15th IEEE/ACS International Conference on Computer Systems and Applications (AICCSA)*, 2018, pp. 1–8.

[66] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-efficient routing protocols in wireless sensor networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 551–591, 2013.

[67] J. A. Paradiso and T. Starner, "Energy scavenging for mobile and wireless electronics," *IEEE Pervasive Computing*, vol. 4, no. 1, pp. 18–27, 01 2005.

[68] Y. Peng, Z. Li, W. Zhang, and D. Qiao, "Prolonging sensor network lifetime through wireless charging," in *IEEE 31st Real-Time Systems Symposium (RTSS)*, 2010, pp. 129–139.

[69] R. V. Prasad, S. Devasenapathy, V. S. Rao, and J. Vazifehdan, "Reincarnation in the ambiance: Devices and networks with energy harvesting," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 195–213, 2014.

[70] M. Rahimi, H. Shah, G. S. Sukhatme, J. Heideman, and D. Estrin, "Studying the feasibility of energy harvesting in a mobile sensor network," in *Proceedings of IEEE In-*

ternational Conference on Robotics and Automation, 2003 (ICRA'03)., vol. 1, 2003, pp. 19–24.

[71] X. Ren, W. Liang, and W. Xu, "Maximizing charging throughput in rechargeable sensor networks," in *23rd International Conference on Computer Communication and Networks (ICCCN)*, 2014, pp. 1–8.

[72] A. Saipulla, C. Westphal, B. Liu, and J. Wang, "Barrier coverage of line-based deployed wireless sensor networks," in *Proceeding of IEEE International Conference on Computer Communications (INFOCOM)*, 2009, pp. 127–135.

[73] A. Saipulla, C. Westphal, B. Liu, and J. Wang, "Barrier coverage with line-based deployed mobile sensors," *Ad Hoc Networks*, vol. 11, no. 4, pp. 1381–1391, 2013.

[74] H. Salarian, "Mitigating energy holes in wireless sensor actuator networks using a mobile sink and charger," Ph.D. dissertation, School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, 2013.

[75] N. Santoro and E. Velazquez, "Energy restoration in mobile sensor networks," *Wireless Sensor And Robot Networks: From Topology Control To Communication Aspects*, p. 113, 2013.

[76] N. Santoro and E. Velazquez, "*Energy Restoration in Mobile Sensor Networks*," in *Wireless Sensor and Robot Networks: From Topology Control to Communication Aspects*. World Scientific, 2014, pp. 113–142.

[77] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: Modeling and analysis of a three-tier architecture for sparse sensor networks," *Ad Hoc Networks*, vol. 1, no. 2-3, pp. 215–233, 2003.

[78] M. Sharifi, S. Sedighian, and M. Kamali, "Recharging sensor nodes using implicit actor coordination in wireless sensor actor networks," *Wireless Sensor Network*, vol. 2, no. 02, p. 123, 2010.

[79] L. Shi, J. Han, D. Han, X. Ding, and Z. Wei, "The dynamic routing algorithm for renewable wireless sensor networks with wireless power transfer," *Computer Networks*, vol. 74, pp. 34–52, 2014.

[80] Y. Shi, L. Xie, Y. T. Hou, and H. D. Sherali, "On renewable sensor networks with wireless energy transfer," in *Proceeding of IEEE International Conference on Computer Communications (INFOCOM)*, 2011, pp. 1350–1358.

[81] D. Simplot-Ryl, I. Stojmenovic, and J. Wu, "Energy efficient backbone construction, broadcasting, and area coverage in sensor networks," *Handbook of Sensor Networks*, pp. 343–380, 2005.

[82] K. Sohraby, D. Minoli, and T. Znati, *Wireless Sensor Networks: Technology, Protocols, and Applications*. John Wiley & Sons, 2007.

[83] S. Sudevalayam and P. Kulkarni, "Energy harvesting sensor nodes: Survey and implications," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 3, pp. 443–461, 2011.

[84] Y. Tirta, B. Lau, N. Malhotra, S. Bagchi, Z. Li, and Y.-H. Lu, "Controlled mobility for efficient data gathering in sensor networks with passively mobile nodes," *Sensor Network Operations*, vol. 3, pp. 92–113, 2006.

[85] B. Tong, "Mobility-assisted energy replenishment for sensor networks," Ph.D. dissertation, 2009.

[86] B. Tong, G. Wang, W. Zhang, and C. Wang, “Node reclamation and replacement for long-lived sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 9, pp. 1550–1563, 2011.

[87] S. Ulukus, A. Yener, E. Erkip, O. Simeone, M. Zorzi, P. Grover, and K. Huang, “Energy harvesting wireless communications: A review of recent advances,” *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 3, pp. 360–381, 2015.

[88] E. Velazquez and N. Santoro, “Mobility-based strategies for energy restoration in wireless sensor networks,” in *Sixth International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, 2010, pp. 161–168.

[89] C. M. Vigorito, D. Ganesan, and A. G. Barto, “Adaptive control of duty cycling in energy-harvesting wireless sensor networks,” in *2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, 2007, pp. 21–30.

[90] C. Wang, J. Li, F. Ye, and Y. Yang, “Multi-vehicle coordination for wireless energy replenishment in sensor networks,” in *Parallel & distributed processing (IPDPS)*, 2013, pp. 1101–1111.

[91] J. Wu, “Collaborative mobile charging and coverage,” *Journal of Computer Science and Technology*, vol. 29, no. 4, pp. 550–561, 2014.

[92] L. Xie, Y. Shi, Y. T. Hou, and A. Lou, “Wireless power transfer and applications to sensor networks,” *IEEE Wireless Communications*, vol. 20, no. 4, pp. 140–145, 2013.

[93] L. Xie, Y. Shi, Y. T. Hou, and H. D. Sherali, “Making sensor networks immortal: An energy-renewal approach with wireless power transfer,” *IEEE/ACM Transactions on Networking*, vol. 20, no. 6, pp. 1748–1761, 2012.

[94] W. Xu, W. Liang, X. Lin, G. Mao, and X. Ren, “Towards perpetual sensor networks via deploying multiple mobile wireless chargers,” in *43rd International Conference on Parallel Processing (ICPP)*, 2014, pp. 80–89.

[95] W. Xu, W. Liang, X. Ren, and X. Lin, “On-demand energy replenishment for sensor networks via wireless energy transfer,” in *IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC)*, 2014, pp. 1269–1273.

[96] Y. Xu, J. Heidemann, and D. Estrin, “Geography-informed energy conservation for ad hoc routing,” in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, 2001, pp. 70–84.

[97] Y. Yang, C. Wang, and J. Li, “Wireless rechargeable sensor networks-current status and future trends,” *Journal of Communications*, vol. 10, no. 9, pp. 696–706, 2015.

[98] F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang, “A two-tier data dissemination model for large-scale wireless sensor networks,” in *Proceedings of the 8th Annual International Conference on Mobile Nomputing and networking*, 2002, pp. 148–159.

[99] X. Ye and W. Liang, “Charging utility maximization in wireless rechargeable sensor networks,” *Wireless Networks*, pp. 1–13.

[100] J. Yick, B. Mukherjee, and D. Ghosal, “Wireless sensor network survey,” *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[101] S. Zhang and J. Wu, “Collaborative mobile charging,” in *Wireless Power Transfer Algorithms, Technologies and Applications in Ad Hoc Communication Networks*. Springer, 2016, pp. 505–531.

[102] S. Zhang, J. Wu, and S. Lu, "Collaborative mobile charging for sensor networks," in *IEEE 9th International Conference on Mobile Adhoc and Sensor Systems (MASS)*, 2012, pp. 84–92.

[103] M. Zhao, J. Li, and Y. Yang, "Joint mobile energy replenishment and data gathering in wireless rechargeable sensor networks," in *Proceedings of the 23rd International Teletraffic Congress*, 2011, pp. 238–245.

[104] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2004, pp. 187–198.

[105] M. Zimmerling, W. Dargie, and J. M. Reason, "Energy-efficient routing in linear wireless sensor networks," in *2007 IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS)*, 2007, pp. 1–3.

[106] M. Zorzi and R. R. Rao, "Geographic random forwarding (geraf) for ad hoc and sensor networks: energy and latency performance," *IEEE Transactions on Mobile Computing*, vol. 2, no. 4, pp. 349–365, 2003.