HAL
open science

# Formal Probabilistic Verification of Wireless Sensor Networks

Maissa Elleuch

## To cite this version:

Maissa Elleuch. Formal Probabilistic Verification of Wireless Sensor Networks. Networking and Internet Architecture [cs.NI]. Ecole Nationale d'Ingénieurs de Sfax (ENIS), 2015. English. NNT : . tel-02964355

**HAL Id: tel-02964355**

**https://theses.hal.science/tel-02964355v1**

Submitted on 21 Oct 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

République Tunisienne
Ministère de l'Enseignement Supérieur,
de la Recherche Scientifique

Université de Sfax
École Nationale d'Ingénieurs de Sfax

*Ecole Doctorale*
*Sciences et Technologies*

**Thèse de *DOCTORAT***

***Nom du Doctorat***
**N°d'ordre: 568/2014**

# THESE

***Présentée à***

## L'École Nationale d'Ingénieurs de Sfax

***En vue de l'obtention du***

# DOCTORAT

**Informatique**
***Ingénierie des Systèmes Informatique***

***Par***

**Maissa ELLEUCH SAHNOUN**

**(Mastère NTSID)**

# FORMAL PROBABILISTIC VERIFICATION OF WIRELESS SENSOR NETWORKS

***Soutenu le 26 Février 2015, devant le jury composé de :***

| | | |
|---|---|---|
| **M.** | **Mohamed JMAIEL** (Professeur à l'ENIS) | Président |
| **M.** | **Abderrazak JEMAI** (Maître de conférences à l'INSAT) | Rapporteur |
| **M.** | **Lamia CHAARI** (Maître de conférences à l'ISIMS) | Rapporteur |
| **M.** | **Sofiène TAHAR** (Professeur à l'Univ. Concordia, Canada) | Examinateur |
| **M.** | **Mohamed ABID** (Professeur à l'ENIS) | Directeur de Thèse |

# ABSTRACT

Formal Probabilistic Verification of Wireless Sensor Networks

Maissa Elleuch

Sfax University, 2014

In the context of Wireless Sensor Networks (WSN), energy efficiency is considered as the most critical requirement. To preserve energy and thus extend the network lifetime, the randomized node scheduling approach is one of the most widespread solutions. Traditionally, the performance of the proposed scheduling algorithms for WSN is usually analyzed using simulation or paper-and-pencil proof methods. Formal methods, in particular model checking, have been less frequently explored. However, these methods either are not scalable or do not ensure accurate results, which are serious drawbacks given the mission-critical WSN applications.

To cope with these intrinsic limitations, this thesis advocates the usage of higher-order-logic theorem proving to formally analyze the probabilistic performance properties of randomly-deployed WSN using the $k$-set randomized node scheduling. Based on the recently developed probability theory, available in the HOL theorem prover, we present the foundational higher-order-logic formalizations of the randomized node scheduling algorithm. Then, we build upon these foundations to formalize the key performance attributes, namely the expected coverage intensity of the network, the detection probability of an intrusion and the delay of detection for an occurring event. Using the achieved formalizations, we present the formal verification of the optimal network lifetime problem under Quality of Service (QoS) constraints associated to coverage and detection. Due to the wide applicability of the $k$-set randomized node

scheduling, these formalizations allow us to tackle the formal analysis of various WSN applications. For illustration purposes, the thesis thus provides the formal performance analysis of different randomly-scheduled wireless sensor networks deployed for forest fire detection and border security monitoring.

# Résumé

Formal Probabilistic Verification of Wireless Sensor Networks

Maissa Elleuch

Université de Sfax, 2014

Dans le cadre des Réseaux de Capteurs sans Fil (RCSF), l'efficacité en énergie est considérée comme la contrainte la plus critique. Pour économiser l'énergie et étendre ainsi la durée de vie de ces réseaux, l'approche d'ordonnancement aléatoire des noeuds est communément utilisée dans ce contexte. Traditionnellement, les performances des algorithmes d'ordonnancement des noeuds proposés pour les RCSF sont analysées en utilisant la simulation ou les modèles analytiques. Les méthodes formelles, en particulier la vérification de modèle, ont été moins souvent explorées. Toutefois, étant donné le caractère probabiliste inhérent aux algorithmes d'ordonnancement aléatoire de noeuds, ces méthodes ne peuvent, en aucun cas, fournir une analyse complètement correcte, ce qui constitue une limitation majeure étant donné l'aspect critique des applications de RCSF.

Pour surmonter les limitations majeures des techniques existantes, cette thèse préconise l'utilisation de la logique d'ordre supérieur, à travers la technique de démonstration de théorèmes, pour analyser formellement diverses propriétés probabilistes de performance de RCSF utilisant l'ordonnancement aléatoire de noeuds. En se basant sur la théorie des probabilités, récemment disponible dans le prouveur de théorèmes HOL, nous développons les formalisations fondamentales de l'algorithme d'ordonnancement aléatoire de noeuds en k-partitions. Ensuite, nous construisons sur ces fondations pour formaliser les attributs clés de performance, à savoir l'intensité moyenne de la

couverture réseau, la probabilité de détection d'une intrusion et le délai de détection d'un évènement. En se basant sur les formalisations obtenues, nous présentons aussi la vérification formelle du problème de la durée de vie optimale du réseau sous des contraintes de Qualité de Service (QoS) liées à la couverture et à la détection. En raison de la large applicabilité de l'algorithme d'ordonnancement aléatoire de noeuds, ces formalisations nous permettent de s'attaquer à l'analyse formelle de diverses applications de RCSF. A titre d'illustration, la thèse fournit l'analyse formelle des performances de réseaux de capteurs sans fil déployés pour la détection des feux de forêts et la surveillance de la sécurité des frontières.

**To My Parents and My Sister, My Husband, My Son and My Daughter.**

# ACKNOWLEDGEMENTS

It is with a great emotion and sincerity that I would like to thank all those who, through their participation and encouragements, helped me to complete this work.

Foremost, I would like to thank both of my supervisors Pr. Mohamed ABID and Pr. Sofiène TAHAR for their support and assistance during my Ph.D studies, providing me with an excellent atmosphere for doing research. I sincerely thank Pr. Sofiène TAHAR for all his precious guidance, caring and patience, from the very beginning and all over my Ph.D studies. His valuable advices have, each time, significant impacts on my research progress. I would like also to thank Pr. TAHAR, who gave me the opportunity to experience the research in his group through two internships. I'm deeply grateful to Dr. Osman HASAN, who was the closest supervisor of my Ph.D, for his outstanding support of my Ph.D research, even though remotely. His practical recommendations, his patience, motivation and enthusiasm helped me to effectively progress in my thesis. For sure, without the close and insightful support of Pr. TAHAR and Dr. Osman, my thesis would not have been completed with such valuable results.

My deepest thanks go to the members of my thesis committee who agreed to evaluate my thesis. Thank you to Mr. Mohamed JMAIEL, Professor at ENIS and Director of the research center, for honoring me by accepting to be the committee chair of this thesis. I am also thankful to Mr. Abderrazak JEMAI, lecturer at INSAT and director of the National Centre of Computing, and Mrs. Lamia CHAARI, Lecturer at ISIMS, for their interest in judging this work.

I am also grateful to my parents, who are always encouraging me to pursue my research and providing me with too much love to progress in my studies. I would like

also to deeply thank my sister for her great support and her care especially regarding my children. I'm also very thankful to my husband for his continuous encouragements and his unconditional patience to complete my doctoral studies. I also want to thank my son, Mohamed Aziz, and my daughter Mariam who brought me with too much love and happiness that helped me overcome hard times during my Ph.D studies. Besides, Mohamed Aziz has been growing up with this thesis and had usually to endure the constraints of a busy student mom.

I finally extend my thanks to all my friends and all the members of the CES laboratory for the good atmosphere. I would also acknowledge all my friends in the Hardware Verification Group (HVG) in Concordia University, where I spent two internships, for their warm welcome and their kindness helping me to quickly adapt myself in the group. I really appreciated working in such friendly team.

# TABLE OF CONTENTS

LIST OF TABLES . . . . . 11

LIST OF FIGURES . . . . . 12

LIST OF ACRONYMS . . . . . 13

**1 Introduction** **1**

1.1 Motivation . . . . . 1

1.2 Wireless Sensor Networks . . . . . 5

1.3 Analysis Approaches for Wireless Sensor Networks . . . . . 7

1.3.1 Theoretical Analysis . . . . . 7

1.3.2 Model Checking . . . . . 9

1.3.3 Theorem Proving . . . . . 12

1.4 Problem Statement . . . . . 14

1.5 Proposed Methodology . . . . . 15

1.6 Thesis Contributions . . . . . 19

1.7 Thesis Organization . . . . . 20

**2 Preliminaries** **22**

2.1 The $k$-set Randomized Scheduling Algorithm for WSN . . . . . 22

2.1.1 Design Assumptions . . . . . 23

2.1.2 Description . . . . . 25

2.1.3 Performance Metrics . . . . . 27

2.2 Probabilistic Analysis in HOL . . . . . 28

2.2.1 HOL Theorem Proving . . . . . 28

2.2.2 Measure Theory . . . . . 30

2.2.3 Probability Theory . . . . . 31

**3 Coverage Analysis** **37**

3.1 System Model . . . . . 37

3.2 Formalization of the $k$-set Randomized Scheduling . . . . . 39

3.3 Formalization of the Coverage Intensity of a Specific Point . . . . . 46

3.4 Formalization of the Network Coverage Intensity . . . . . 51

3.5 Application: Forest Fire Detection . . . . . 56

3.5.1 Formal Analysis based on the Number of Nodes . . . . . 58

3.5.2 Formal Analysis based on the Number of Subsets . . . . . 61

3.5.3 Formal Analysis based on Uniform Partitions . . . . . 63

3.6 Summary and Discussions . . . . . 66

**4 Detection Analysis** **71**

4.1 Formalization of the Intrusion Period . . . . . 71

4.2 Formalization of the Detection Probability . . . . . 74

4.2.1 Detection Probability for Short Events . . . . . 76

4.2.2 Detection Probability for Long Events . . . . . 81

4.3 Formalization of the Average Detection Delay . . . . . 82

4.4 Application: Formal Analysis of WSN for Border Surveillance . . . . . 91

4.4.1 Formal Analysis based on the Number of Nodes . . . . . 93

4.4.2 Formal Analysis based on Uniform Partitions . . . . . 95

4.5 Summary and Discussions . . . . . 98

**5 Lifetime Analysis** **101**

5.1 Problem Formulation . . . . . 101

5.2 Mathematical Analysis of the Optimal Lifetime . . . . . 104

5.3 Formalization of the Optimal Lifetime . . . . . 109

5.4 Summary and Discussions . . . . . 113

**6 Conclusions and Future Work** **117**

6.1 Conclusions . . . . . 117

6.2 Future Work . . . . . 118

**Bibliography** **120**

**Biography** **133**

# LIST OF TABLES

2.1 Variable Notations for the $k$-set Randomized Scheduling . . . . . 26

2.2 HOL Symbols . . . . . 30

3.1 Variable Notations for Coverage . . . . . 48

3.2 Coverage Analysis of the Forest Fire Application . . . . . 66

4.1 Detection Analysis of the Border Surveillance Application . . . . . 98

5.1 Verified Properties for the Lifetime Analysis . . . . . 113

# LIST OF FIGURES

1.1 The Network Architecture of a Wireless Sensor Network. . . . . 6

1.2 Proposed Methodology . . . . . 17

2.1 An example of the $k$-set randomized scheduling for 8 nodes and 2 subsets. 26

2.2 Illustration of Performance Attributes. . . . . 27

3.1 An example of the $k$-set randomized scheduling for $n$ nodes and $k = 3$. 40

4.1 Detection Analysis [89]. . . . . 73

# LIST OF ACRONYMS

| | |
|---|---|
| WSN | Wireless Sensor Network |
| HOL | Higher-Order Logic |
| HOL4 | HOL4 Theorem Prover |
| PMF | Probability Mass Function |
| PRISM | PRobabilistIc Symbolic Model checker |

# Chapter 1

# Introduction

## 1.1 Motivation

The stunning progress in Micro-electromechanical Systems (MEMS) technology has led to very small tiny sensors, so that their deployment on a wireless network over an area is at once fast, reliable and cheap. Currently, Wireless Sensor Networks; called also WSN, are being increasingly used to ensure a continuous and automated monitoring of different kind of environments and serving thus limitless applications including, home automation, external environmental monitoring and object tracking [94]. Due to their inherent features, wireless sensor networks have attracted a great deal of attention in the research community. Indeed, although these networks are direct descendants of traditional wireless networks, their multiple resource constraints make the existing algorithms for classical wireless networks, inappropriate. Furthermore, wireless sensor networks are highly dynamic networks in nature. Such networks can thus commonly exhibit a lot of probabilistic behavior whose mainly due to the random nodes deployment, the hostile environment and the unpredictable traffic patterns. A wide variety of protocols and algorithms, more frequently probabilistic, have

been thus specifically designed to meet the WSN requirements.

Due to their restricted size, sensors are basically battery-powered and thus have very limited energy resources. This feature makes energy saving as one of the most critical requirements within a wireless sensor network. Consider the example of a WSN deployed for forest fire detection, in which the sensor nodes are randomly distributed with a high density. Once deployed, the network is expected to keep functional for a sufficiently long period while efficiently ensuring the monitoring of the whole forest area. Such expectation will never be reached without appropriately scheduling the energy of each of the sensor node to extend the lifetime of the whole network. In fact, replacing or recharging the sensor batteries in such harsh environmental conditions would be obviously hard. In addition, the heavy node deployment results in further energy losses since it is highly probable that the same region would be simultaneously covered, i.e., monitored by many nodes. On the other hand, monitoring every point of the forest by keeping every node at the active state will surely lead to a huge waste of energy and seems hence to be completely unrealistic [83]. Since a wild fire occurs occasionally, some sensor nodes can be intuitively deactivated to save the network energy. By having a smaller number of sensors active at any given time, the lifetime of the overall system increases, at the cost of lower performances. Based on this idea, various sensor scheduling algorithms have been explored in the open literature [59, 73, 77, 1, 42, 87, 13, 83, 68, 46].

Scheduling sensor nodes to save energy is surely a simple and intuitive approach, however it is very important to keep good monitoring performances of the area. For the same forest fire application, the deployed WSN should be also able to detect the outbreak of fires at any point with a high probability and report it within a small delay. Consequently, besides the network lifetime, the coverage and detection performance

equally arise as critical performance requirements. These application requirements are called Quality of Service (QoS) constraints. According to the target applications, different design goals have been taken into account in the design of sensor scheduling algorithms. Early solutions have been conceived to extend the network lifetime while preserving the coverage quality [59, 77, 1, 42, 13, 87] whereas other consider network connectivity [15, 29, 14]. More recent scheduling solutions consider both constraints of coverage and connectivity [78, 72, 97]. In this context, the $k$-set randomized scheduling [52] considers the joint problem of coverage and connectivity. The main idea consists in organizing a set of nodes by randomly subdividing them into a partition with "k" sets. The formed subsets of nodes work alternatively within their allocated time slot so that the overall network energy is preserved. Such algorithm, suitable for a wide range of WSN applications, has shown good performance results in extending the network lifetime while keeping acceptable performance.

Randomized algorithms are usually much more efficient [61], but more difficult to validate. More particularly, the random feature of the $k$-set randomized scheduling makes it very challenging to analyze for all possible cases. The random assignment of the sensor nodes to the $k$ sub-networks may lead to some sub-networks which are completely empty. In this case, it is highly probable that unacceptable delays will be made for the detection of a critical intrusion. Moreover, due to the random deployment of nodes coupled with the randomized scheduling, it may happen that certain parts of the area are not monitored at all or simultaneously monitored by many sensors. Traditionally, the $k$-set randomized scheduling has been extensively analyzed using paper-and-pencil based probabilistic technique [59, 77, 1, 42, 51, 87, 97]. The reliability of the obtained analytical models is consolidated through simulation using the Monte Carlo method [55]. Although based on mathematic as very powerful tool,

the complete correctness of analytical models is apparently hard to assert. Such statement can be unbelievable but unfortunately true! The paper-and-pencil based proof can be prone to human-errors regarding the set of assumptions or even the mathematical steps. It is very common that missing a mathematical step or even a very small sign error will result in faulty models. Evidently, all these limitations, if not carefully spotted, lead to models which remain wrong forever, even till the design stage. Added to that, the simulation approach, used to validate the analytical results, usually produces very incredible results for various reasons such as the "bugs" that may stem from the underlying computer programs. These analysis limitations can have detrimental consequences especially in case of safety-critical applications like forest fire detection, e.g., a fire threat may be ignored due to an undetected bug.

In order to overcome the common drawbacks of simulation, formal methods [31] have been recommended as an efficient solution to validate a wide range of hardware and software systems. Using mathematical techniques, such methods provide the possibility to rigorously analyze the mathematical model for the given system to check if it meets a given property. In recent years, there was a growing interest in applying formal methods in the context of analyzing wireless sensor networks to assess their functional correctness or analyze their quantitative performance [58, 7, 64, 98]. Nevertheless, wireless sensor networks pose many challenges in their analysis especially because of their inherent randomness which imply that most of their properties are probabilistic. Examples of such properties include the probability that an intrusion event occurs or the expected coverage quality. More recent progress in the formal methods area has presented efficient solutions to correctly include the probabilistic feature in the system analysis. Probabilistic model checking is very commonly adopted in this context. Such technique surely provides valuable understandings of the system

behavior, however, as soon as WSN of large scale are concerned, state explosion problems [16] are shortly observed. In addition, many inaccuracies arise when modelling the probabilistic metrics and in the reasoning support about statistical quantities like expectation and variance.

## 1.2 Wireless Sensor Networks

A wireless sensor network can be basically defined as a collection of small tiny sensors [94] that collaborate together to monitor a given area (see Figure 1.1). In general, a sensor node is composed of four units: the sensing unit (sensors), the processing unit (processors), the transmission unit (wireless transceivers) and the energy control unit (battery). All of these four units are within the size of several cube millimeters [94]. Thanks to its sensing unit, a sensor can individually take different measurements of the monitored area such as light, temperature, humidity, pressure and acceleration. The communications between the sensor nodes is a short-range wireless communication, which is made possible thanks to their transmission unit. The gathered data is either transmitted to a specific sensor called sink, whose main goal is to collect data from different sensors, or directly to the gateway sensors (see Figure 1.1). Finally, the data transmitted to the end user is analyzed and sent to a remote user so that appropriate decisions can be taken. Depending on the application domain, a sensor node may integrate optional modules such as a positioning system (GPS), or an energy harvesting system (solar cell). More recently, sensor nodes can be even equipped with a movable system for mobility purposes [86].

- **Structured vs. unstructured WSN:** Wireless sensor networks can be roughly classified into structured and unstructured according mainly to the kind of the area of interest; indoor or outdoor [94]. In structured WSN, a given number

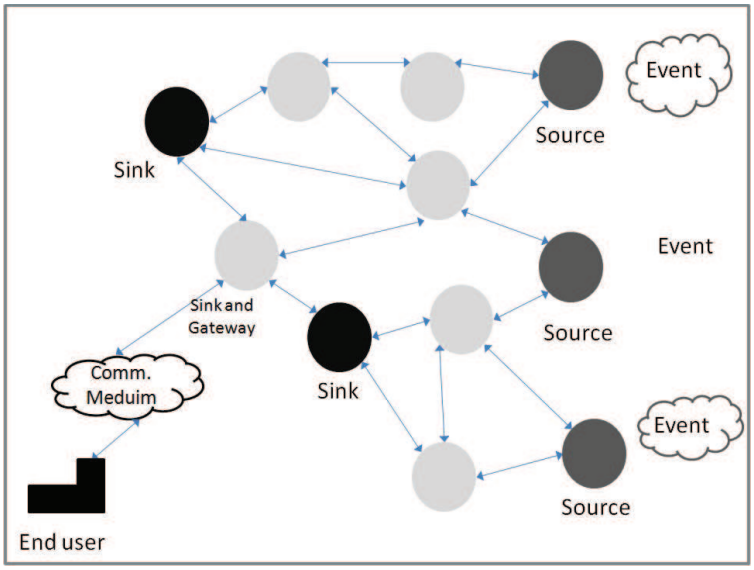


Figure 1.1: The Network Architecture of a Wireless Sensor Network.

of nodes are usually placed at specific locations in a pre-planned way within a closed environment, e.g., a building. For unstructured WSN, a greater number of nodes is deployed in a completely ad-hoc manner into an open area like a mountain. Therefore, a structured WSN is generally characterized by a lower node density and requires thus little network maintenance. On the other hand, several design issues regarding for example connectivity, detection and coverage, are raised in unstructured WSN because of their ad-hoc feature.

- **WSN applications:** Wireless sensor networks have a wide variety of real-world applications that can be mainly classified into two main categories: monitoring and tracking [94]. Monitoring applications include monitoring environments, health, power and manufacturing process. Environmental applications, for example, enable the prevention from natural disasters through measuring environmental indicators, e.g., earthquakes, forest fires and floods. Moreover, health monitoring is also very useful in the biomedical domain to take care of a patient

through in-body sensors. For power surveillance purposes, a WSN can be also deployed to detect chemical or biological attacks. On the other hand, tracking applications include tracking different kind of objects such as animals and humans. Military applications are also an example of tracking applications where it is possible to make the detection and the identification of enemy intrusion through a wireless sensor network. As examples of successful WSN applications, we can mention the WSN deployed to monitor the bird "Leach's Storm Petrel" in the Great Duck island in the USA [94], and the ZebraNet application [94] for analyzing the wild animals over a harsh area of $1000m^2$ in Kenya.

## 1.3 Analysis Approaches for Wireless Sensor Networks

In what follows, we survey the existing approaches for the performance analysis of randomized scheduling algorithms for WSN. While theoretical analysis is the most commonly used approach, we extend our state-of-the-art to include also the formal approaches applied in the general context of WSN.

### 1.3.1 Theoretical Analysis

Theoretical analysis, also known as paper-and-pencil based probabilistic technique, has been widely used to validate randomized scheduling algorithms for WSN. Such analysis consists in building a pure theoretical model by first identifying the required random variables and the associated performance attributes. After that, a rigourous analysis based on the foundations of probability theory is achieved. To validate the analytical results, simulation, using the Monte Carlo method [55], is finally performed.

Based on repeated random sampling, the Monte Carlo simulation method, build a picture of the probability distribution over which estimates of some statistical quantities, e.g., expectation and variance, can be then made.

Several works report on the analysis of the randomized scheduling using paper-and-pencil based probabilistic technique. In [87], a variant of the randomized scheduling, called Lightweight Deployment-Aware Scheduling (LDAS), is proposed and studied via analytical modeling. Such schema deactivates redundant nodes using a random weighted voting method.The corresponding performance analysis has examined some metrics associated to redundancy, where many statistical quantities like the expectation of non-covered area, have been proved. The resulting analytical model has been validated through extensive simulations on a WSN deployed over a region of $150m \times 150m$ with sensors whose detection range is $10m$. The problem of scheduling nodes in low-duty WSN has been also considered in [42]. The coverage extensity and intensity of the network have been mathematically studied, then validated through simulation using Matlab [80]. In [13, 52], a variant of the randomized scheduling, based on uniform partitions, is presented. To show the practical effectiveness of this algorithm, theoretical analysis, using probability theory, has been done to evaluate pertinent performance metrics, namely, the network coverage, the detection probability and delay, and the network lifetime [52, 89]. The resilience of the same algorithm regarding clock asynchrony has been also mathematically investigated in [52]. In [92], the coverage performance of the same scheduling algorithm has been mathematically analyzed under different nodes deployment schemas while considering the size and the shape of the intrusion objects. The detection probability under different scenarios has been examined as well. The detection accuracy of WSN for forest fire detection has been analyzed using paper-and-pencil analysis in [95]. Experimentation has been

done to validate the forest fire system on a real prototype of 5 nodes. More generally, theoretical analysis has been conducted to validate the coverage performance of the randomized scheduling algorithm, proposed in [13, 52], in the context of an hybrid surveillance framework for environmental monitoring [92]. Results have been validated through simulation on a circular surface of a radius $R = 10000$, where up to $n = 2000$ nodes are uniformly deployed. Very interesting also is the recent work of Quazi who has proposed a new kind of randomized nodes scheduling based on some sensor information about neighbours and residual energy [56]. The coverage performance has been considered through mathematical analysis while simulations have been run with specific network sizes and different sensing ranges.

Clearly, the accuracy of a paper-and-pencil based proof heavily depends on the human-error factor. Probabilistic models usually rely on a lot of intuition where most of the assumptions are either not explicit or not so accurate. In addition, the simulation technique applied to analyze such models is usually subject to many imprecisions. Indeed, computer simulation relies on computer models which consists in some coded algorithms coupled with numerical data to simulate the system. These models are frequently prone to many coding errors, i.e., "bugs". It is thus hard to completely assert their correctness. Finally, the produced results through simulation can never be generic, i.e., they are usually specific to given settings, e.g., the number of nodes, their range, and the size of the sensing area. Such inherent inaccuracies are clearly very compromising given the safety-critical feature of most of WSN applications.

### 1.3.2 Model Checking

Model checking is one of the most used formal methods for the probabilistic analysis of wireless systems [70]. Traditional model checking is primarily based on building a

mathematical model of the system which is exhaustively tested to check if it meets a set of properties. The system to be verified is hence usually modelled as a finite state automate, then the property of interest is formalized into a logical formula. To verify the satisfiability of the property, the state space is exhaustively explored through dedicated tools. Taking into account the probabilistic feature of the target system, probabilistic model checking has recently emerged as a promising alternative enabling thus a more realistic analysis.

Both model checking techniques; classical and probabilistic, have been successfully used to validate various aspects in the WSN context. In [64], the formal analysis of the Optimal Geographical Density Control (OGDC) algorithm [97] has been performed. The OGDC algorithm is a kind of randomized scheduling algorithm which saves energy by switching nodes while maintaining network connectivity. The formal analysis has been achieved in the RT-Maude rewriting tool [69] where common performance metrics, such as network coverage intensity and lifetime, have been successfully verified. Several other works have also reported on the use of the model checker Uppaal [8] for the analysis of various protocols for WSN [25, 81]. Also, the probabilistic model checker PRISM [84] has been used quite frequently for the verification of Medium Access Control (MAC) protocols designed for WSN [7, 28, 96]. General transmission properties for specific network configurations have been thus formally checked. Some statistical measures such as expected communication latency and energy consumption, have been formally analyzed as well. The state-based formal verification method, model checking [16], has been also the basis of many formal frameworks proposed for the validation of WSN. In [32], the model checker SPIN [41] is used within the SLEDE framework to verify WSN security aspects for NesC implementations. Similarly, a model checking based framework, called NesC@PAT

[98], has been also used for verifying WSN implementations in the NesC language. In this work, based on a formal semantics of the NesC language, the sensor behaviors are captured through Labelled Transition Systems (LTSs) while the model checker of the PAT tool [75] is used to analyze the WSN model. The target properties are deadlock-freeness, state reachability and some temporal properties.

In addition to its accuracy, the main advantage of model checking method is its mechanization. However, it suffers from the common problem of state space explosion [16]. Indeed, once the state graph of the verified system becomes too large, its exploration for a given property turns out to be 'painful' and even impossible. Hence, during the verification of the OGDC algorithm [64], only networks of up to 6 nodes has been handled within a monitored surface of $15m \times 15m$. Similarly, in [7], the network hops have been restricted to 3 and the number of schedules to 2 to keep tractable model in PRISM. For the verification of ECO-MAC [96], the authors have been obliged to readjust some parameters by a reduction factor to avoid a state explosion problem which was completely unpredictable. Furthermore, the work of [98] has reported over 1 million generated states for the verification of a single property. In [32], some additional simplifying assumptions including some temporal abstractions and parameters reduction have been applied to carry out the analysis. On the other hand, while probabilistic model checkers have been proposed to cope with the probabilistic limitations of classical ones, these tools still lack of a sound probability support. For example, in [64], a random function, which is assumed to be 'good', has been used to model probabilistic behavior. For Uniform distributions, a sampling value generated by the same random function on a given interval is selected. Such kind of analysis is not exhaustive and thus cannot be termed as formally verified. The

authors of [64] have besides suggested the use of PMaude [2] to enhance their probabilistic analysis results. Finally, the reasoning support for statistical quantities in the PRISM model checker suffers from many shortcomings. Indeed, expected values of the performance attributes are usually given through running several experiments on the built model [7, 96]. These values have been usually specific to the chosen configuration and can not be considered as general in any way.

Some other attempts for building formal simulation frameworks for WSN can be found in [71, 26, 57, 54]. The main idea of these frameworks is to first describe the different components of wireless sensor networks in a single formalism, e.g, timed automata [71], process algebra [57], then provide the possibility of formal analysis of some properties using formal tools [84, 67, 12]. Nevertheless, besides the fact that most of the proposed frameworks have been restricted to the specification stage, no interesting WSN case study has been made to show their effectiveness at the formal verification side.

### 1.3.3 Theorem Proving

Unlike the many works based on model checking found for the analysis of wireless sensor networks, very few works based on theorem proving exist in the open literature. In general, theorem proving [30] consists in formalizing a given system and the properties of interest into logical statements of first or higher-order logic. Using existing axioms and inference rules, a proof that the system satisfies these properties is then built.

In [36], a clock synchronization protocol for WSN, has been analyzed using the Isabelle/HOL theorem prover [62]. More specifically, the correctness of the strict set of constraints on the required parameters for full connectivity has been formally checked.

A second work reports on the use of the PVS system [65] to build a theorem proving based framework for WSN algorithms [9]. Within such formal framework, a WSN algorithm can be formally specified using a library of mathematically specified sub-blocks like the nodes, the network structure, communication primitives and protocols. The communication primitives are functionalities for communication between nodes like the forwarding, injection and dropping of packets, while the protocols are specified so that they can use services installed on nodes. Services include the packet logger, the receive buffer, the node scheduler and the clock. Each component is described by a PVS model taken from the corresponding theory. Different versions of the same component are available so that it is possible to analyze the same WSN algorithm under several perspectives and at the desired level of abstraction. The resulted framework is then extended to include some probabilistic scenarios such as nodes mobility and link quality changes. The practical effectiveness of the whole PVS framework for WSN, has been illustrated by manually analyzing the trace execution of the Surge algorithm [9]. The authors have hence evaluated the receive queue size, the energy consumption and the robustness to topology changes on a network of maximum 25 nodes with different topologies. By inspecting the execution traces, they have been able to detect a potential problem of infinite loops of routed packets in the algorithm specification. As a second case study, the correctness of the message delivery for the Reverse Path Forwarding (RPF) algorithm has been formally analyzed in [10].

Despite the guaranteed advantages of the theorem proving technique, many limitations have been recognized in the works mentioned above. Effectively, the formal analysis, done in [36], has been performed for the required set of constraints on the parameters and not for the properties of interest. This clearly restricts the scope of the verification work. In addition, while the PVS framework [9] is supposed to be extended

with some "dynamic" scenarios in [10], the randomness aspect has been characterized by a pseudo-random generator. The nodes mobility, specified by the random walk pattern, has been also specified through a simple recursive function. Furthermore, considering the link quality changes for lossy channels with a uniform probability $P_c$, this probability has been instantiated by a given value throughout the analysis. The corresponding routing tables have been thus generated using the concrete value of $P_c$. Given all these restrictions, it is unlikely that the analysis results using the PVS framework can be considered as accurate regarding the probability modelling.

## 1.4 Problem Statement

Although the need of formal methods has been pointed out in many papers, effective attempts at using them in the WSN context are not very common. On the other hand, since wireless sensor networks are being increasingly explored for deployment in many safety-critical applications, there is a great need to accurately assess their correctness.

In summary, previous research techniques, used to validate these networks, are not:

- reliable in capturing randomness of WSN into account. Although considering the randomized aspects increases the confidence in the obtained results, the existing techniques, such as simulation and probabilistic model checking, usually suffer from imprecisions in the probabilistic modelling and inabilities in reasoning about statistical quantities like expectation and variance.

- scalable to handle WSN of large size. Neither simulation, nor model checking can provide an exhaustive analysis regardless of the design parameters values.

Hence, the analysis results usually lack the generic property. This is a significant limitation, especially that WSN are commonly deployed in applications of large scale like environmental monitoring.

- accurate in their analysis. For example, the paper-and-pencil based probabilistic technique is based on analyzing a theoretical model on paper by a human, which makes it error prone. Similarly, the non-exhaustive nature of simulation makes it inaccurate as well.

- practical enough to address the analysis of a wide range of WSN applications. The usefulness of previous approaches is mostly limited to specific case studies in the WSN domain.

In this thesis, we propose to use the HOL4 theorem prover to tackle the formal analysis of wireless sensor networks. Through the choice of the HOL4 theorem prover, we aim at providing trustworthy analysis of the target problem while developing solid and scalable analysis results. Thanks to a rich probability library, this prover offers promising abilities to reason about a wide range of randomness, including the formal reasoning about WSN models with multiple continuous random variables, which has not been addressed before. On the other hand, due to the usage of theorem proving, the analysis results are guaranteed to be effectively sound, generic, and of a wide applicability.

## 1.5 Proposed Methodology

We are interested to provide, in this thesis, a completely rigorous performance analysis of wireless sensor networks using the $k$-set randomized scheduling algorithm, which is a widely used probabilistic algorithm to save energy in the context of WSN. The

basic building blocks of the proposed methodology are depicted in Fig. 1.2, while the formalization requirements are represented by shaded boxes at the left side.

To achieve the formal performance analysis of a given WSN application using the $k$-set randomized scheduling, the first step is to formalize the description of the WSN application as a system model in higher-order logic. This step mainly requires the foundational formalizations of the randomized scheduling. Appropriate probabilistic variables are thus needed to model the inherent randomness of the algorithm as higher-order-logic functions. The second step consists in expressing the properties of interest as higher-order-logic goals based on the formal system model developed in the first step. This step is made possible due to the formalizations of the key performance attributes including the network coverage, the detection probability, the detection delay and the network lifetime. Our fundamental work is to develop the formalizations of these performance attributes based on the paper-and-pencil probabilistic models of the $k$-set randomized algorithm available in the open literature [13, 52, 89]. Each of these performance attributes is hence formally specified taking into account its probabilistic feature and verified afterwards utilizing the foundations of probability theory. The foremost requirement here is to be able to correctly model the probabilistic aspect of the performance properties within a theorem prover. The resulting formalizations for each of the attributes, shown by the three rectangular boxes at the left side of Fig. 1.2, are made available in distinct higher-order-logic library or theory in order to facilitate the formal reasoning about WSN systems using the randomized scheduling. The third step is to formally provide the proofs of the goals, developed in the previous step as theorems, in a theorem prover using the pre-verified theorems. The output of the theorem prover, annotated by the dashed edge rectangular box, certifies that the given performance properties are valid for the

given WSN application using the $k$-set randomized nodes scheduling algorithm.

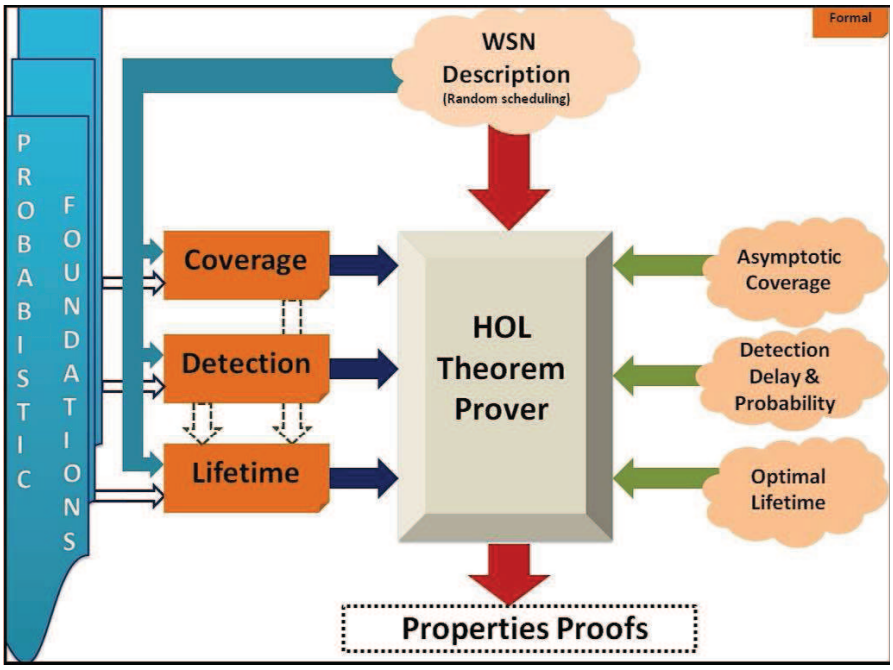


Figure 1.2: Proposed Methodology

It is important to note that the formalizations steps, mentioned above, are mainly founded on the paper-and-pencil probabilistic analysis of the $k$-set randomized algorithm available in [13, 52, 89]. Though, our main work is to formalize them in higher-order-logic to handle the formal performance analysis of WSN within the sound core of the HOL theorem prover. Finally, due to the wide applicability of the $k$-set randomized algorithm in various WSN applications, the practical effectiveness of our methodology is possible through formally analyzing various real-world applications, such as, environmental outdoor monitoring [91] and enemy intrusion detection [63].

Although there are many propositions of node scheduling algorithms in the open literature [1, 42, 59, 77, 87, 97], the $k$-set randomized scheduling, presented in [13], is effectively considered as one of the most interesting. While certain scheduling algorithms have been focused either on coverage or connectivity, the $k$-set randomized

scheduling is distinguishable by considering the joint problem of coverage and connectivity without too constraining assumptions. In addition, such algorithm is the only found so far which provides rigorous paper-and-pencil probabilistic models in terms of various performance metrics which are the network coverage, the detection probability, the detection delay and the network lifetime. Compared to the other proposed node scheduling schemas, we therefore strongly believe that the $k$-set randomized scheduling probabilistic algorithm is worth formalizing within a theorem prover.

Based on [34], we developed some formalizations of the $k$-set randomized scheduling algorithm and the network coverage [20, 21]. Recently, a more generic formalization of probability theory has been developed in the HOL theorem prover [60]. Aiming at providing more solid and scalable formalizations, we decided thus to migrate our previous higher-order-logic formalizations into the new HOL probability theory. Our whole work, in this thesis, is primarily built on the most recent and generic probability theory in the HOL4 theorem prover [39].

To the best of our knowledge, the formal analysis of the OGDC algorithm, done in [64], constitutes the only one dealing with the formal analysis of a variant of randomized scheduling algorithms through model checking within the RT-Maude rewriting tool [69]. Besides the well-known limitations of model checking, limits in the probability modelling have been clearly recognized by the authors themselves who suggested the use of the PMaude tool [2] to enhance their results. The work of [9, 10] can be also considered as related to ours in the general context of the formal analysis of WSN through theorem proving. Although such work is intended to formalize dynamic probabilistic scenarios, it has been largely limited by the probability support of the PVS system which gave very inaccurate formalizations.

## 1.6 Thesis Contributions

The main contribution of this thesis is an approach for formally analyzing the performance of wireless sensor networks using the $k$-set randomized nodes scheduling, which is a widely used energy conservation algorithm in this context. For that purpose, we build the foundational formalizations of the $k$-set randomized scheduling algorithm for WSN and its key performance attributes within the sound core of the higher-order-logic theorem prover. Our approach has the merits to provide accurate and generic results while allowing modular reasoning about the different performance attributes. Based on the proposed approach, the formal performance analysis of a wide range of WSN applications is hence possible. Some of the key contributions of this thesis are listed as follows.

- Formal specification of the $k$-set randomized nodes scheduling algorithm for randomly deployed wireless sensor networks according to the given design assumptions.

- Formalization of the coverage performance attributes which are the coverage of a specific point of the monitored area and the coverage of the whole network [20]. We successfully utilize these formalizations to provide the formal analysis of the coverage behavior of a randomly-scheduled wireless sensor network deployed for forest fire detection [21]. The same higher-order-logic developments of the the coverage property have been applied to formally perform some asymptotic analysis on a real-world WSN for volcanic earthquake detection as well [22].

- Formalization of the detection performance metrics including the detection probability and the detection delay of an intrusion event within the deployed WSN [23]. These formalizations are primarily built upon some formal reasoning about

the intrusion period of any occurring event. For illustration purposes, we formally evaluate the detection performances of a WSN deployed for border security monitoring [23]. The developed formalizations regarding the detection probability can be built upon to formalize other performance attributes, such as the impact of clock asynchrony on the network coverage in randomly-scheduled WSN [52].

- Formal verification of the optimal network lifetime based on both formalizations of the coverage and detection properties. More particularly, we formally analyze the optimal lifetime problem under Quality of Service (QoS) constraints associated to coverage and detection.

## 1.7 Thesis Organization

The rest of the thesis is organized as follows.

In Chapter 2, we first present the $k$-set randomized scheduling algorithm for wireless sensor networks. We also provide a brief description of the main theories required to conduct the probabilistic analysis in the HOL4 theorem prover. The theorem proving technique and the HOL4 theorem prover are also introduced in this chapter.

In Chapter 3, we present our fundamental formalizations of the $k$-set randomized scheduling algorithm for wireless sensor networks according to a given system model. We exploit these foundations to formally reason about the key coverage performance attributes: the coverage intensity of a specific point and the expected value of the network coverage intensity. To show the practical interest of our higher-order-logic developments, we formally evaluate the coverage behavior of a real-world WSN

application for forest fire detection.

In Chapter 4, we describe the higher-order-logic formalization of the detection probability and the detection delay of an intrusion event in randomly-scheduled WSN. Both of these probabilistic characteristics are built upon the formal analysis of a statistical property associated to the intrusion period of any occurring event. Using the resulting theoretical developments of detection, the detection performance of a WSN deployed for border security monitoring is formally checked where various detection properties are analyzed including the asymptotic detection behavior of the given application.

Chapter 5 shows how useful are the formalizations of coverage and detection, developed in Chapters 3 and 4, for the formalization of the optimal network lifetime under Quality of Service (QoS) constraints.

Finally, Chapter 6 provides concluding remarks and summarizes perspective insights.

# Chapter 2

# Preliminaries

In this chapter, we first present the $k$-set randomized scheduling algorithm for wireless sensor networks, in particular, the main design assumptions and performance metrics are described. After that, we introduce the higher-order-logic theorem proving technique and the HOL theorem prover. We finally provide an overview of the main theories required to conduct the probabilistic analysis within the HOL theorem prover.

## 2.1 The $k$-set Randomized Scheduling Algorithm for WSN

In the open literature, a wide variety of node scheduling algorithms have been proposed for use in the context of wireless sensor networks [83]. The main idea of such approach is to deactivate nodes by rounds so that the overall energy can be preserved. While the common objective is to maximize the network lifetime, the nodes to deactivate are usually chosen according to some selection criteria that can be deterministic or completely random. Moreover, the proposed solutions fundamentally differ in their

assumptions regarding the sensors (detection model, transmission range, location information, etc.) and the whole network (deployment strategy, network structure, time synchronization, etc.) [83]. In this thesis, we are interested in a variant of node scheduling which is completely random, that is the $k$-set randomized scheduling. This algorithm has been separately proposed by [1] and [51]. The main idea of such approach is to randomly organize the nodes into alternatively working subsets of nodes. Hence, during a given time slot, only the nodes belonging to the current active subset are turned on and may report an occurring event while all the other nodes are inactive and thus enables a whole energy saving of the overall system. Subsequently, we give a detailed description of the $k$-set randomized scheduling algorithm. The most relevant design assumptions and the performance metrics of interest are also surveyed.

### 2.1.1 Design Assumptions

The main design assumptions of the $k$-set randomized scheduling algorithm are as follows [51, 13].

- **Sensor sensing range:** Every sensor can only sense the environment and detect events within its circular sensing area. Hence, a sensor can never sense an event being out of his detection range. The typical values of the detection range are in the order of meters. It is important to note here that no relationship is assumed between the detection range of a sensor and its transmission range which is associated to the radio of the transmission unit.

- **Network deployment:** Deployment can be defined as how to physically put the sensor nodes over the area of interest. This deployment can be either deterministic or random. In a deterministic deployment, nodes are manually placed at specific locations according to a given model such as grid. On the contrary, a

random deployment relies on dropping the sensors from a boat or an helicopter according to a given random model, which can Uniform, Poisson, etc. In [79], it has been shown that random deployment is much cheaper and reliable than the deterministic one. Moreover, scheduling deterministic network is simple and may not be worth exploring. For the $k$-set randomized scheduling, a random deployment is much more relevant. This random deployment is assumed to be Uniform, i.e., the sensor nodes are fairly distributed over the monitored area. Moreover, the deployment of one node is completely independent of the other ones. In other words, the location of a given sensor does not have an impact on the placement of another one.

- **Communication architecture:** Once the sensors are deployed over the area, they have to organize themselves into a given communication architecture which can be flat or hierarchical. In a flat structure, all the sensors have identical role and communicate together via multi-hop radio communication to transmit the data till the sink node. Whereas the hierarchical architecture relies on organizing the sensors into clusters, where each sensor communicate first with the corresponding cluster head, then this cluster head will directly communicate with the sink node. For the $k$-set randomized scheduling, a flat structure is simply assumed.

- **Network density:** The sensor density is the number of deployed sensors by unit square. According to the application requirements, this density can be average or high. Here, the network is assumed to be enough dense, which is suitable for monitoring large area.

### 2.1.2 Description

Consider a wireless sensor network that is formed by randomly deploying a set $Sn = \{s_0, s_1, ..., s_{(n-1)}\}$ of $n$ sensor nodes over a field of interest of size $a$. Each sensor has a sensing area of size $r$. During the initialization stage, the $k$-set randomized scheduling is run in parallel on every node as follows [51]. Each node starts by randomly picking a number, denoted by $i$, ranging from 0 to $(k-1)$, where $k$ is the number of subsets or partitions. A node $s_j$ is thus assigned to the $i^{th}$ sub-network, designated by $S_i$, and will activate itself only during the scheduling round of that subset. At the end of the algorithm, $k$ disjoint sub-networks are created. These subsets will be working independently and alternatively in a round-robin fashion. In other words, during a given working round $T_i$, only the nodes belonging to the subset $S_i$ are turned on to detect a potential event. Whereas, during all the other scheduled rounds, the nodes of the subset $S_i$ will fall asleep. The main steps of the $k$-set randomized algorithm can be summarized as follows.

| **The $k$-set randomized algorithm on sensor $s_j$** |
|---|
| ***1.*** *Pick a random number* $i \in [0..(k-1)]$ |
| ***2.*** *Assign* $s_j$ *to subset* $S_i$ |

Intuitively, when the wireless sensor network is quite dense, each subset alone can cover most of the area. The $k$-set randomized algorithm has the merits to be a purely distributed algorithm, thus scalable for large networks. Table 2.1 lists the main variable notations used for the $k$-set randomized scheduling, as well as their significations.

For illustration purposes, Fig. 2.1 shows how the $k$-set randomized scheduling algorithm splits arbitrarily a small WSN of eight sensor nodes to two sub-networks. The eight nodes, randomly deployed in the monitored region, are identified by IDs

| Var. | Signification |
|---|---|
| $n$ | The total number of deployed nodes |
| $k$ | The number of sub-networks or partitions |
| $a$ | The size of the monitored field |
| $r$ | The size of the sensing area of each sensor |
| $Sn$ | The initial set of sensors in the network whose cardinality is $n$ |
| $s_j$ | The sensor node number $j$ such that $0 \leq j \leq (n-1)$ |
| $S_i$ | The sub-network number $i$ such that $0 \leq i \leq (k-1)$ |
| $T_i$ | The scheduling or working round of subset $S_i$ with $0 \leq i \leq (k-1)$ |

Table 2.1: Variable Notations for the $k$-set Randomized Scheduling

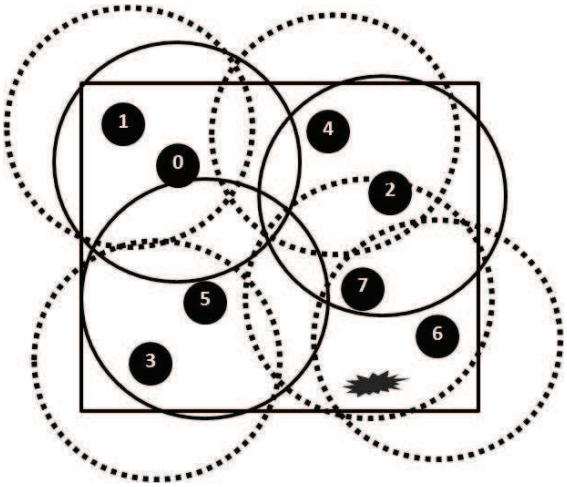


Figure 2.1: An example of the $k$-set randomized scheduling for 8 nodes and 2 subsets.

ranging from 0 to 7. The two sub-networks are called $S_0$ and $S_1$. Each node randomly chooses a number 0 or 1 in order to be assigned to one of these two sub-networks. Suppose that nodes 0; 2; 5, select the number 0 and join the subset $S_0$ and nodes 1; 3; 4; 6; 7, choose the number 1 and join the subset $S_1$. These two sub-networks will work alternatively, i.e., when the nodes 0; 2; 5, with sensing ranges denoted by the solid circles, are active, the nodes 1; 3; 4; 6; 7, illustrated by the dashed circles, will be idle and vice-versa.

### 2.1.3 Performance Metrics

The randomness in the $k$-set randomized scheduling algorithm, presented above, makes it very challenging to analyze for all possible cases. Furthermore, the number of subsets $k$ has a great impact on the overall network performance. The design assumptions may have also different effects on the network performance attributes. In addition to the network lifetime, the main relevant metrics for the performance analysis of the $k$-set randomized scheduling algorithm, are the network coverage, the detection delay and the detection probability [83, 52, 89]. These are the more related metrics to the lifetime performance.

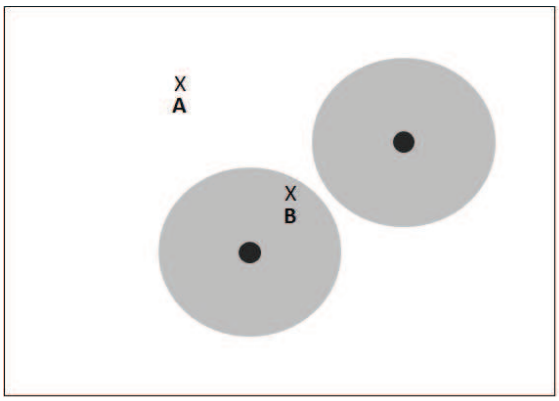


Figure 2.2: Illustration of Performance Attributes.

- **Network coverage:** Called also sensing coverage, the network coverage is a spatial performance attribute which measures how well the area of interest is monitored or tracked by the sensor nodes [83, 52]. In Figure 2.2, the point $B$ is covered by the sensor at the bottom whereas point $A$ is uncoverd since it does not belong to any of the sensing range of the two sensors. In general, the sensor network is said to provide $k$-coverage when every point of the area is monitored by at least $k$ active sensors. If there are uncovered points, then the coverage is said to be partial.

- **Detection probability:** It is the probability that an occurring event can be detected by one or more sensor nodes. It is clear that an event should be detected with a high probability.

- **Average detection delay:** It is the average time spent from the occurrence of an event to the time when the event is detected by some sensor nodes. We require that an occurring event is detected with the smaller possible delay.

- **Network lifetime:** It is the time duration from which the network is no longer alive. It depends basically on the lifetime of the individual sensors. In general, the network should keep its operation as long as possible.

## 2.2 Probabilistic Analysis in HOL

In this section, we start by an overview of the higher-order-logic theorem proving which includes a description of the HOL4 theorem prover. Then, we present the probabilistic foundations available in this prover.

### 2.2.1 HOL Theorem Proving

The theorem proving based method consists in showing that a given assertion can be deduced as a logical consequence of a set of statements (the axioms and assumptions). Basically, an axiom designates an unprovable proposition to admit as is, whereas all statements have to be written in the logical language of the proof assistant, which is commonly propositional, first-order logic and higher-order logic. Each logic has its own syntax that is used to describe the informal description. The general process behind theorem proving is composed of three main steps: formally specifying the system to be verified by functions in the target logic, formalizing the properties of interest as

proof goals in the same logic and finally verifying these goals as theorems within the proof assistant, using the existing axioms and inference rules. The proof procedure is based on various techniques such as rewriting, simplification by repeated substitution, decision procedures and mathematical induction. The theorem proving based method offers a sound support for mathematical reasoning about systems using computers. The proof of a given theorem is possible using only the existing axioms, primitive inference rules and previously proved theorems. A wide variety of theorem provers exist in the open literature. An overview of systems implementing mathematics in the computer is available at [50]. Examples of the most successful higher-order-logic provers include Isabelle [44], HOL4 [39], HOL Light [38], PVS [85], and Coq [17].

The HOL theorem prover [39] is a proof assistant of higher-order logic which includes a very rich library of theories. A theory can be defined as a set of pre-verified theorems for a given domain, function or operation. When needed, a HOL theory can be loaded and used, which greatly aids the verification process. Additionally, users may be assisted by automatic proof procedures [30], which are a collection of steps in a single command. Despite the existence of all these theories and automatic procedures, most of the time, proofs in HOL are interactive and require the intervention of user. Various proof techniques, such as rewriting, simplification, specialization, generalization and mathematical induction, are available in HOL to aid the verification process. In Table 2.2, we summarize some of the HOL symbols used throughout this thesis and their corresponding mathematical interpretation [30].

Several higher-order-logic provers include the formalization of probability theory (See e.g. [43, 49, 34, 4, 40, 60]). In this thesis, we utilize the recently developed and most generic probability theory developed by Mhamdi [60], within the HOL4 theorem prover. The work of Mhamdi [60] has the merit of generalizing the previous HOL

Table 2.2: HOL Symbols

| **HOL Symbol** | **Standard Symbol** | **Meaning** |
|---|---|---|
| $\wedge$ | *and* | Logical *and* |
| $\vee$ | *or* | Logical *or* |
| $\sim t$ | $\neg t$ | Not $t$ |
| $\lambda x.t$ | $\lambda x.t$ | Function that maps $x$ to $t(x)$ |
| SUC $n$ | $n+1$ | Successor of a *num* |
| count $n$ | $\{m \mid m < n\}$ | Set of all $m$ strictly less than $n$ |
| PREIMAGE $f$ $s$ | $\{x \mid f\ x \in s\}$ | The inverse image of the subset $s$ |
| $\{x \mid P(x)\}$ | $\{\lambda x.P(x)\}$ | Set of all $x$ that satisfy the condition $P$ |
| $x$ *pow* $n$ | $x^n$ | *real* $x$ raised to *num* power $n$ |
| *exp* $x$ | $e^x$ | Exponential logarithm on $x$ |
| SIGMA $f$ $s$ | $\sum_s f$ | Sum of the sequence $f(x)$ where $x \in s$ |
| $lim(\lambda n.f\ n)$ | $\lim_{n\to\infty} f(n)$ | Limit of the *real* sequence $f$ |

formalization of measure theory by including a Borel space. After specifying the extended real numbers in HOL, he formalized measure, Lebesgue, probability and information theories.

### 2.2.2 Measure Theory

In general, a measure can be considered as a generalization of the concepts of length, area, volume, etc. It consists in assigning a number to each suitable subset of a given set. Two widely common examples are the Lebesgue measure on an Euclidean space and the probability measure on a Borel space. A measure function is defined over a class of subsets, called the measurable sets, and assigns a non-negative real number to every measurable set. Some of the important definitions of measure theory [11], formalized in [60], are given below.

- Sigma algebra: It contains the empty set $\varnothing$, is closed under countable unions and complementarity within the space $\chi$.

- A triplet $(\chi, A, \mu)$ where $(\chi, A)$ is a measurable space and $\mu : A \to \mathbb{R}$ is a

measure.

- Measurable functions: A function $f : X1 \rightarrow X2$ is called measurable if the inverse image of a measurable set is also measurable, i.e., $f^{-1}(A) \in A1$ for all $A \in A2$, where $A1$ and $A2$ are measurable sets.

- Borel sigma algebra: The Borel sigma algebra is the smallest sigma algebra generated by the open sets of $X$.

### 2.2.3 Probability Theory

The formalization of probability theory in HOL is based on the Kolmogorov axiomatic definition of probability. Hence, by building upon the measure theory, this formalization has the advantage to provide a unified framework for discrete and continuous probability measures.

A probability measure $P$ is basically a measure function on the sample space $\Omega$ and an event is a measurable set within the set $F$ of events which are subsets of $\Omega$. Thus, $(\Omega, F, P)$ is a probability space iff it is a measure space and $P(\Omega) = 1$. A random variable is by definition a measurable function. A real random variable is thus specified in HOL in the following definition [60].

**Definition 2.1.**

```
⊢ ∀X p.  real_random_variable X p = prob_space p ∧
                    ∀x ∈ p_space p ⇒ X x ≠ NegInf ∧ X x ≠ PosInf) ∧
                    X ∈ measurable (p_space p,events p) Borel.
```

where $X$ designates the random variable, $p$ is a given probability space, $NegInf$ and $PosInf$ are the higher-order-logic formalizations of negative infinity or positive infinity, and $Borel$ is the HOL definition of the Borel sigma algebra.

The probability distribution of a random variable is specified as the function that accepts a random variable $X$ and a set $s$ and returns the probability of the event $\{X \in s\}$. It has been formalized in HOL [60] in Definition 2.2.

**Definition 2.2.**

```
⊢ ∀X p.  distribution p X = (λs.  prob p (PREIMAGE X s ∩ p_space p)).
```

The expectation of a random variable $X$ is defined in HOL as its Lebesgue integral with respect to the probability measure $p$ [60].

$$E[X] = \int_{\Omega} X dp. \tag{2.1}$$

which has been formalized in HOL as follows.

**Definition 2.3.**

```
⊢ expectation = integral.
```

For a discrete random variable, the expectation has been verified in HOL in Theorem 2.1.

**Theorem 2.1.**

```
⊢ ∀X p.  FINITE (IMAGE X (p_space p)) ∧ (real_random_variable X p)
  ⇒ (expectation p X = SIGMA (λr.
    r × Normal (distribution p X {r})) (IMAGE X (p_space p))).
```

where `(IMAGE X (p_space p))` designates the list of values taken by the function $X$ over the sample space `(p_space p)`. In the discrete case, this list has to be finite.

- **Conditional probability in HOL**

The conditional probability has been also formalized in HOL [53] according to the following mathematical definition.

$$Pr(A \mid B) = \frac{Pr(A \cap B)}{Pr(A)}. \tag{2.2}$$

where $A$ and $B$ are two events of the set $F$ of events.

Accordingly, the following useful results have been formally verified in HOL [53].

- If the events $A$ and $B$ are independent such that $(Pr(B) \neq 0)$ , then

$$Pr(A \mid B) = Pr(A). \tag{2.3}$$

- The conditional probability of the event $(A \cup B)$, given the event $C$ is

$$Pr(A \cup B \mid C) = Pr(A \mid C) + Pr(B \mid C) - Pr(A \cap B \mid C). \tag{2.4}$$

- If $A$ and $B$ are disjoint, then the above equation becomes

$$Pr(A \cup B \mid C) = Pr(A \mid C) + Pr(B \mid C). \tag{2.5}$$

- The conditional probability of the event $(A \cap B)$ given the event $C$ is

$$Pr(A \cap B \mid C) = Pr(A \mid B \cap C) \times Pr(B \mid C). \tag{2.6}$$

- Given that $\{B_i, i \in s\}$, is a finite partition of the entire sample space $\Omega$, the law

of total probability states that

$$Pr(A) = \sum_{i \in s} Pr(A \mid B_i) \times Pr(B_i). \tag{2.7}$$

The above equation has been formalized in HOL as follows.

**Theorem 2.2.**

```
⊢ ∀p B A s.  (prob_space p) ∧ FINITE s ∧ (A ∈ events p) ∧
  (∀x.  x ∈ s ⇒ B x ∈ events p) ∧
  (∀a b.  a ∈ s ∧ b ∈ s ∧ (a ≠ b) ⇒ DISJOINT (B a) (B b)) ∧
  (BIGUNION (IMAGE B s) = p_space p)
  ⇒ (prob p A = ∑_s (λi.  (prob p (B i)) ×
     (cond_prob p A (B i))) s).
```

where

- The assumption `(∀x.  x ∈ s ⇒ B x ∈ events p)` specifies a finite partition of the whole outcome space $\Omega$, i.e., a collection of events, which is pairwise disjoint `(∀a b.  a ∈ s ∧ b ∈ s ∧ (a ≠ b) ⇒ DISJOINT (B a) (B b))`, and whose union is $\Omega$ `(BIGUNION (IMAGE B s) = p_space p)`.
- `cond_prob` is the HOL formalization of the conditional probability.

- **Conditional Expectation**

Based on the above probability formalizations, we next describe our higher-order-logic developments of further probabilistic notions required for the work described in this thesis, and which are not available in the HOL4 theorem prover.

- Conditional independence: Two events $A$ and $B$ are conditionally independent given the event $C$, iff:

$$Pr(A \cap B \mid C) = Pr(A \mid B) \times Pr(A \mid C). \tag{2.8}$$

- The conditional independence is also equivalent to

$$Pr(A \mid B \cap C) = Pr(A \mid C). \tag{2.9}$$

- Discrete conditional expectation: The conditional expectation of the discrete random variable $X$ given the event $(Y = y)$, denoted by $E(X \mid Y = y)$, is the expected value of $X$ with respect to its conditional probability distribution, and is mathematically specified as follows

$$E(X \mid Y = y) = \sum_{x} x \times Pr(X = x \mid Y = y). \tag{2.10}$$

The concept of conditional expectation can be also extended to multiple events. In the current work, we will basically require the conditional expectation of $X$ given two events, i.e., $E(X \mid Y = y, Z = z)$, which is mathematically defined as

$$E(X \mid Y = y, Z = z) = \sum_{x} x \times Pr(X = x \mid Y = y \cap Z = z). \tag{2.11}$$

where $Z$ is a discrete random variable. Definition 2.4 gives the higher-order-logic formalization of the conditional expectation $E(X \mid Y = y, Z = z)$.

**Definition 2.4.**

```
⊢ ∀X Y Z y z p sx.  cond_expec_2 X Y Z y z p sx =
  ∑_space sx (λx.  x ×
  Normal (cond_prob p (PREIMAGE X {x} ∩ p_space p)
  (PREIMAGE Y {y} ∩ p_space p ∩ (PREIMAGE Z {z} ∩ p_space p)))).
```

where the HOL function `Normal` is used to convert a real value to its corresponding value in an extended real. Based on the above definition, we can easily verify, in HOL, that $E(X \mid Y = y) = E(X \mid Y = y, \mathbb{1}_\Omega = 1)$, where $\mathbb{1}_\Omega$ is the indicator function on the probability space $\Omega$.

- The conditional expectation of a function of a random variable is formally verified in HOL as

$$E(g(X) \mid Y = y) = \sum_x g(x) \times Pr(X = x \mid Y = y) \tag{2.12}$$

- The law of total expectation: By analogy to the law of total probability (Equation (2.7)), we formally verify that

$$E(X) = \sum_y E(X \mid Y = y) \times Pr(Y = y) \tag{2.13}$$

# Chapter 3

# Coverage Analysis

After deployment, a wireless sensor network is expected to cover the whole area of interest, i.e., any point of the monitored area should be monitored with at least one sensor. In this chapter, we first formally develop the foundational higher-order-logic formalizations of the randomized nodes scheduling algorithm for WSN, using the recently developed probability theory, available in the HOL4 theorem prover. Then, we build upon these foundations to formally reason about the key coverage performance attributes: the coverage intensity of a specific point and the expected value of the network coverage intensity. The coverage performance behavior of a real-world WSN for forest fire detection is then formally analyzed illustrating thus the practical interest of our higher-order-logic developments.

## 3.1 System Model

We consider a wireless sensor network formed by deploying $n$ nodes over a field of interest of any shape with size $a$. Every sensor in this WSN can only sense the environment and detect events within its sensing range $r$. To preserve energy, the

$k$-set randomized scheduling [13, 52, 89] is applied to partition the $n$ nodes into $k$ subsets. In the following, we give provide the main set of assumptions required for our higher-order-logic formalizations. More details about the algorithm and each of its assumptions can be found in Section 2.1.

- The area of interest can have any shape.
- The node deployment is random following a Uniform distribution.
- The deployment of nodes is independent. This means that sensor nodes are independently distributed of each other over the area of interest.
- The sensor density can be high or normal.
- The communication structure is flat.
- The sensing range of each sensor is uniform.
- The transmission range of each sensor is fixed.
- No hard time synchronization between nodes is required.
- Location information of each sensor are not needed.

Compared to other energy-efficient scheduling mechanisms [83], we believe that the above set of assumptions are sufficiently realistic, so that the formalization of the $k$-set randomized scheduling and its key performance metrics, within the HOL4 theorem prover, has significant contributions. These higher-order-logic formalizations will be primarily based on the existing paper-and-pencil analysis available in the open litterature [13, 52, 88, 93, 89].

## 3.2 Formalization of the $k$-set Randomized Scheduling

Given the description of the $k$-set randomized scheduling algorithm, presented in Chapter 2, each sensor node randomly selects a unique number $i$ out of the $k$ available options. The $k$ generated subsets of nodes $\{S_i, 0 \leq i \leq (k-1)\}$ are thus disjoint, i.e., a given node belongs to one subset at once. Afterwards, these node subsets are scheduled to work alternatively within their scheduling time slots $\{T_i, 0 \leq i \leq (k-1)\}$.

To emphasize on the impact of the random feature inherent to the $k$-set randomized scheduling algorithm, we consider the example of a randomly-scheduled WSN where the set of $n$ nodes is partionned into $(k = 3)$ subsets: $S_0$, $S_1$ and $S_2$ (see Fig. 3.1). Let $t_0$ be any reference time while an intrusion event $e$, which lasts $L$ time units and starts at time $t_z$. Due to the probabilistic feature of the scheduling algorithm, the sub-network $S_2$ does not contain any node. Since the subsets are working by rounds, a complete time slot is allocated to the subset $S_2$ at every turn, but there are no active nodes to detect the event $e$ during the whole time slot. In an other scenario, all the $n$ nodes may randomly be assigned into the same partition giving only one subset, which is non-empty. In this case, there will be a single round during which an event is likely to be covered. The empty sub-networks of nodes, generated by the randomized scheduling, hence have a significant effect on the overall network performance.

Subsequently, we are first interested in formally verifying the probability that the $k$-set randomized node scheduling produces an empty partition or sub-network. As previously mentioned, the basic idea of the randomized scheduling of nodes consists in randomly assigning each of the node to one of the $k$ sub-networks. This assignment is done uniformly so that the random organization of the nodes into several sub-networks

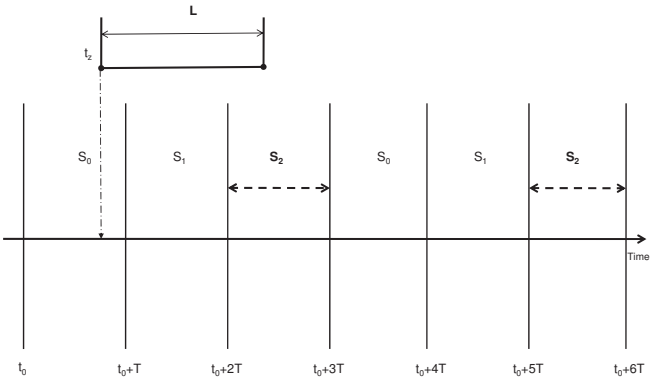


Figure 3.1: An example of the $k$-set randomized scheduling for $n$ nodes and $k = 3$.

is potentially fair over the whole network. Each node intuitively joins a single subset with the same probability $\left(\frac{1}{k}\right)$. The appropriate random variable, required in this formalization, should uniformly distribute the nodes over the $k$ sub-networks, i.e., a Uniform random variable, which we formally specify as follows:

**Definition 3.1.** *(The Uniform random variable)*

```
⊢ ∀X p k.  uniform_distr_rv X p k =
        (real_random_variable X p) ∧
        (IMAGE X (p_space p) = IMAGE (λx.&x) (count (SUC k))) ∧
        (∀m.  m ∈ IMAGE X (p_space p) ⇒ (distribution p X {m} = 1/&k)).
```

where $X$ is a real-valued random variable; `real_random_variable`, which takes values on the integer interval $[0..(k-1)]$, i.e., `(IMAGE (λx.&x) (count (SUC k)))` with the probability distribution; `distribution`, equals $\left(\frac{1}{\&k}\right)$. The operator `&`, used in the above definition, allows the conversion of the natural number $m$ into its extended real number counterpart.

Using the output information of the Uniform random variable on the whole set

of nodes $n$, we can identify if a given scheduled subset of nodes; $S_j$, is empty. Indeed, a subset $S_j$ is empty if the randomized scheduling does not assign any of the nodes to that subset. In other words, none of the $n$ nodes selects the number $j$. To model an empty sub-network, we first consider the $n$ Uniform random variables, generated as a list, and then determine if the index of the sub-network; $j$, belongs or not to this list.

For that purpose, we start by specifying, in Definition 3.2, the HOL function `rd_subsets` which recursively generates a list of $n$ elements.

**Definition 3.2.** *(General list of n elements)*

```
⊢ (∀x.  rd_subsets 0 x = []) ∧
  (∀n x.  rd_subsets (SUC n) x = x::(rd_subsets n x)).
```

where the input parameter $n$ denotes the number of nodes which is a natural number, and $x$ is an extended real number.

Next, we formally specify, in Definition 3.3, a recursive HOL predicate, which looks for a specific index $j$ in a given list . The corresponding function `subset_empty` takes as inputs an extended real $j$ and a list $L$ having the format $(h :: t)$, and returns true only if $j$ is not in the list $L$.

**Definition 3.3.** *(Predicate for an empty subset)*

```
⊢ (∀j.  subset_empty [] = T) ∧
  (∀j h t.  subset_empty j (h::t) = (h ≠ j) ∧ (subset_empty j t)).
```

The set of $n$ nodes is uniformly partitioned into $k$ sub-networks, a node hence joins a given subset $S_j$ with the uniform probability $\left(\frac{1}{k}\right)$. The same node will miss the same subset with the complement probability $\left(1 - \frac{1}{k}\right)$. Consequently, a given subset $S_j$ is empty if and only if the $n$ sensors do not join, i.e., miss this subset. More

formally, lets consider the event $T_{i,j}$: "The sensor $i$ does not join the subset $S_j$", we have then

$$\begin{aligned} Pr(S_j\, is\, empty) &= Pr(n\, sensors\, do\, not\, join\, S_j) \\ &= Pr(T_{0,j} \cap ... \cap T_{(n-1),j}) \end{aligned} \tag{3.1}$$

where

$$Pr(T_{i,j}) = \left(1 - \frac{1}{k}\right) \tag{3.2}$$

Since the $n$ sensor nodes miss the subset $S_j$ independently, the events $T_{0,j}$,...,$T_{(n-1),j}$ will be mutually independent, which means that any given event is completely independent of the intersection of any other events [27]. Based on that, the probability that a given subset $S_j$ is empty (Equation 3.1), can be obtained by applying the mutual independence rule, which gives $\left(1 - \frac{1}{k}\right)^n$.

Accordingly, we successfully verify, in Theorem 3.1, the probability that a given subset $S_j$ is empty in a randomly-scheduled WSN.

**Theorem 3.1.** *(The basic probability of an empty subset)*
*The $k$-set randomized scheduling algorithm applied in a WSN of $n$ nodes, may generate an empty subset with the following probability:*

$$Pr(T_{0,j} \cap ... \cap T_{(n-1),j}) = \left(1 - \tfrac{1}{k}\right)^n$$

```
⊢ ∀X p k n j.  (prob_space p) ∧ (1 < k) ∧
  (uniform_distr_rv (X k) p k) ∧ (j ∈ IMAGE (X k) (p_space p)) ∧
```

```
(∀s m.  indep p ({x | (X k) x ≠ m} ∩ (p_space p))
({x | subset_empty m (rd_subsets s ((X k) x))} ∩ (p_space p)))
⇒ (prob p ({x | subset_empty j (rd_subsets n ((X k) x))}
∩ p_space p) = (1 − 1/&k)^n).
```

where

- The assumption `(1 < k)` ensures that the number of sub-networks is greater than 1 since the randomized scheduling would be meaningless for $(k = 1)$.
- `(uniform_distr_rv (X k) p k)` is the Uniform random variable, given in Definition 3.1.
- The event `({x | subset_empty j (rd_subsets n ((X k) x))} ∩ p_space p)` formally models the event of the probability given in Equation 3.2, i.e., the event "The subset $S_j$ is empty". The function `rd_subsets` (Definition 3.2) hence generates the output values of the Uniform random variable $(X\ k)$ ordered as a list of length $n$ in which the predicate `subset_empty` (Definition 3.3) looks for the index $j$.
- The last assumption ensures the mutual independence over the set of the $T_{i,j}$ events (Equation (3.1) using the HOL function `indep`.

*Proof.* The proof of the above theorem is based on induction and the multiplication rule, which switches the probability of a set of independent events to the product of their respective probabilities, i.e., $Pr(\bigcap_{i=0}^{(n-1)} T_{i,j}) = \prod_{i=0}^{(n-1)} Pr(T_{i,j})$. To complete the proof, the verification of the probability distribution of the Uniform random variable, $Pr(T_{i,j})$, and its complement, along with set theoretic analysis was required.

Since a sub-network is either empty or not, we can model such behavior by simply a Bernoulli random variable `Y`, with the success probability `(prob p ({x`

| subset_empty j (rd_subsets n ((X k) x))} ∩ p_space p). We describe the higher-order-logic formalization of an empty sub-network in HOL as follows.

**Definition 3.4.** *(The basic empty subset random variable)*

```
⊢ ∀Y n X p k j.  subset_empty_rv1 Y n X p k j =
  (bernoulli_distr_rv Y p (prob p ({x | subset_empty j (rd_subsets n
  ((X k) x)) = T} ∩ p_space p))).
```

where we specify the higher-order-logic Bernoulli random variable with success probability *pr* in the following definition.

**Definition 3.5.** *(The Bernoulli random variable)*

```
⊢ ∀X p pr.  bernoulli_distr_rv X p pr =
              (real_random_variable X p) ∧
              (IMAGE X (p_space p) = {0;1}) ∧
              (distribution p X {1} = pr).
```

Based on the above formalization, we can easily reverify the probability distribution of an empty sub-network, already verified in Theorem 3.1, as follows.

**Theorem 3.2.** *(The probability of an empty subset)*
*Given $n$ empty subsets, generated by the $k$-set randomized scheduling, each modelled as a Bernoulli random variables, the probability that the $k$-set randomized scheduling algorithm may generate an empty subset is reverified to be equal to $\left(1-\frac{1}{k}\right)^n$.*

```
⊢ ∀X Y p j n k.
  (prob_space p) ∧ (1 < k) ∧ (uniform_distr_rv (X k) p k) ∧
  (j IN (IMAGE (X k) (p_space p))) ∧ (subset_empty_rv1 Y n X p k j) ∧
  (∀s m.  indep p ({x | (X k) x ≠ m} ∩ (p_space p))
```

```
({x | subset_empty m (rd_subsets s ((X k) x))} ∩ (p_space p)))
⇒ (prob p ({x | Y x = 1} ∩ p_space p) = (1 − 1/&k)^n).
```

*Proof.* The proof is based on some rewriting together with the proof of the probability distribution of a Bernoulli random variable (Definition 3.5).

According to Theorem 3.2, we can notice how the probability distribution of an empty sub-network, generated by the randomized node scheduling, depends only on the input parameters $k$ and $n$. For the sake of simplicity, we opt to abstract the Uniform random variable by directly modelling an empty subset using a Bernoulli random variable, with success probability, the resulting probability value, i.e., $\left(1 - \frac{1}{k}\right)^n$. The new higher-order-logic function, denoted `sbst_empty_rv`, is shown in Definition 3.6.

**Definition 3.6.** *(The empty subset random variable using Bernoulli)*

```
⊢ ∀X p k n.
        sbst_empty_rv X p k n = bernoulli_distr_rv X p (1 − 1/&k)^n.
```

The higher-order-logic formalizations, presented so far, constitute our foundations towards the formalization of the probabilistic performance properties of the randomized node scheduling. While it would have been much simpler to directly model an empty sub-network by a Bernoulli random variable (Definition 3.6), the above analysis has been useful to concretely show the logical reasoning while justifying the origin of the associated probability. In what follows, we will simply make use of Definition 3.6, whereas the complete HOL code for this part is available at [19].

## 3.3 Formalization of the Coverage Intensity of a Specific Point

Within a wireless sensor network, any point of the deployment area should be monitored by at least one active sensor, so that an occuring event, at any time, can be detected (see Figure 2.2). The coverage ability of each point of the monitored area is hence characterized by an intensity, whereas the coverage behavior of the whole network is the average among all the nodes. The network coverage is thus a widely used performance metric [94].

Consider the same example of a WSN deployed for forest fire detection, where the randomized scheduling is applied to save energy over the whole network. The outbreak of a fire at any point of the forest area should be covered with the highest probability in order to alarm the user. Besides, the coverage characteristic may not be correctly ensured, if for example, because of the unpredictable deployment of sensors, there are no nodes deployed in the close area of the fire. On the other hand, it may happen that there exist nodes in this area, but they are inactive due to the scheduling rounds. However, such a typical application of WSN is considered as very critical where missing an intrusion event can be really disastrous.

In the next analysis, we assume a wireless sensor network where the $k$-set randomized scheduling is applied as energy conservation mechanism. Based on the reference paper-and-pencil probabilistic analysis [13, 52], we are first interested in formalizing the coverage intensity of a specific point of the monitored area, which we build upon to develop the higher-order-logic formalization of the network coverage of the whole WSN.

We suppose that a given point of the area is monitored by $c$ sensors which form

a set $S$. Note that the variable $c$ corresponds to the variable $s$ used in the initial specification [13, 52]. According to the randomized scheduling of nodes, each of the node in set $S$ belongs to only one scheduled sub-network, $S_i$, where $0 \leq i \leq (k-1)$ (cf. Table 3.1). Let $Sc_i$ denote the set of sensors that belongs to the sub-network $S_i$ and covers a specific point inside the field, i.e., $Sc_i \subseteq S_i$. The set $S$ hence consists in the union of the subsets $Sc_i$, where $\{0 \leq i \leq (k-1)\}$, and is specified in the following equation.

$$S = Sc_0 \cup Sc_1 \cup Sc_2 \cup ... \cup Sc_{(k-1)} \tag{3.3}$$

The coverage intensity of a specific point inside the monitored area, denoted by $C_p$, is mathematically defined as [52] the average time during which the point is covered in a whole scheduling cycle of length $k \times T_i$. Since the WSN is randomly scheduled, a given point would be covered if the current active subset, $S_i$, contains at least one node in the set of covering nodes, i.e., $Sc_i$. In other words, the subset $Sc_i$ is not empty. The term "empty", used here, refers to a subset empty of covering nodes since we are now reasoning on the set $S$. Consequently, the coverage metric of a specific point depends on the scheduled non-empty subsets regarding the point of interest, $Sc_i$, within a whole scheduling cycle.

Table 3.1 contains a summary of the variables notation that will be used throughout the coverage part.

Let $X$ be a random variable describing the total number of non-empty subsets, i.e,

$$X = \sum_{j=0}^{k-1} X_j \tag{3.4}$$

| Var. | Signification |
|---|---|
| $a$ | The size of the monitored field |
| $r$ | The size of the sensing area of each sensor |
| $q$ | The probability that each sensor covers a given point, equals $r/a$ |
| $Sc_i$ | The set of sensors that belongs to the sub-network $S_i$ and covers a specific point inside the field |
| $T_i$ | The working round of subset $S_i$ |
| $S$ | The set of nodes covering a specific point inside the field |
| $c$ | The cardinality of $S$ |

Table 3.1: Variable Notations for Coverage

where $X_j$ is the Bernoulli random variable whose value is 1 in case of non-empty subset. The coverage intensity of a given point in the monitored area, $C_p$, as originally specified in [52], is then

$$C_p = \frac{E[X] \times T_i}{k \times T_i} \tag{3.5}$$

where $E[X]$ denotes the expectation of $X$ (Equation 3.4), and $T_i$ designates the length of a scheduling cycle. In the equation above, the variable $T_i$ is kept intentionally unsimplified, so that the mathematical definition correctly refelects the textual one regarding the time aspect.

Similar to the specification of an empty subset, presented in Definition 3.6, we can describe a non-empty sub-network by a Bernoulli random variable with the complement probability of $\left(1 - \frac{1}{k}\right)^n$.

**Definition 3.7.** *(The non-empty subset random variable)*

```
⊢ ∀X p k c.
    sbst_non_empty_rv X p k c = bernoulli_distr_rv X p (1 − (1 − 1/&k)^c).
```

In higher-order logic, we model the coverage behavior of a specific point (Equation (3.5)) by the following predicate `cvrge_intsty_pt`.

**Definition 3.8.** *(The coverage intensity of a specific point)*

```
⊢ ∀X p k s pr.  cvrge_intsty_pt X p k s pr =
    expectation p (λx.  SIGMA (λi.  ((X pr) i) x) s) / (&k).
```

where $X$: a random variable that returns an extended real number, $p$: the probability space, $k$: the number of sub-networks, $s$: the summation set whose cardinality is $k$, and $pr$: the probability of a non-empty subset.

In Theorem 3.3, we have been able to formally verify the following mathematical expression for the coverage intensity of a point of the monitored area.

**Theorem 3.3.** *(The coverage intensity of a specific point)*
*In a WSN of $n$ nodes, randomly-scheduled into $k$ partitions, consider a list of $k$ random variables modelling the non-empty subsets $\{X_0, X_1, ..., X_{(k-1)}\}$, each with the probability $pr = \left(1 - (1 - \frac{1}{(\&k)})^c\right)$, the coverage intensity of a specific point satisfies:*

$$C_p = pr$$

```
⊢ ∀X p k s c.  (prob_space p) ∧ (FINITE s) ∧ (CARD s = k) ∧
  (1 < k) ∧ (pr = 1 − (1 − 1/(&k))^c) ∧
  (∀i.  i ∈ s ⇒ sbst_non_empty_rv ((X pr) i) p pr)
  ⇒ (cvrge_intsty_pt X p k s pr = Normal pr).
```

where

- The assumption `(∀i.  i ∈ s ⇒ sbst_non_empty_rv ((X pr) i) p pr)` indicates that every element of the set $s$ is modelled as a random variable of type `sbst_non_empty_rv` (Definition 3.7).

- The HOL function `Normal` is used to convert a real value to its corresponding value in an extended real.

*Proof.* The proof of the above theorem is mainly based on Theorem 3.4 about the linearity of the expectation property. It is also a prerequisite to show the measurability of the used events, along with some analysis on extended reals.

**Theorem 3.4.** *(The expectation property)*
*Given a list of random variables $\{X_0, X_1, ..., X_s\}$ over the sample space $\Omega$, each with a finite expectation, the expectation property satisfies:*

$$E[\sum_{i \in s} X_i] = \sum_{i \in s} E[X_i]$$

```
⊢ ∀p X s.  (prob_space p) ∧ (FINITE s) ∧
  (∀i.  i ∈ s ⇒ real_random_variable (X i) p ∧
  (expectation p (X i) ≠ PosInf) ∧ (expectation p (X i) ≠ NegInf))
⇒ (expectation p (λx.  SIGMA (λi.  (X i) x) s) =
  SIGMA (λi.  expectation p (X i)) s).
```

*Proof.* We proved Theorem 3.4 based on the proof of a more general result of the expectation property which states that $E[aX + bY] = aE[X] + bE[Y]$, where $X$, $Y$ are random variables and $a, b$ are real numbers. Since the expectation is basically specified using an integral (Definition 2.3), the latter proof required operations from the Lebesgue theory coupled with some reasoning on the function integrability, as well as some analysis on extended reals.

## 3.4 Formalization of the Network Coverage Intensity

We show that the coverage of every point of the monitored area is described by a coverage intensity $C_p$ (Definition 3.8). The average value of the coverage intensity over all points of the given area represent a single performance metric, qualified as the network coverage intensity. Mathematically, the network coverage intensity, denoted by $C_n$, is specified, in Equation (3.6), as the expectation of the coverage intensity of a specific point $C_p$ [13, 52].

$$C_n = E[C_p] \tag{3.6}$$

Based on the expression of $C_p$, shown in Theorem 3.3, we proved that the coverage intensity $C_p$ is equal to $\left(1 - \left(1 - \frac{1}{k}\right)^c\right)$. Accordingly, we can rewrite Equation (3.6) as

$$C_n = E[1 - \left(1 - \frac{1}{k}\right)^c] \tag{3.7}$$

From the above equation, we can notice how the value of $C_n$ depends mainly on $c$ which is the number of nodes covering a given point of the field. Intuitively, a sensor node covers or not a given point with the probability $q = \frac{r}{a}$. We can thus assimilate this fact to a Bernoulli trial with success probability $q$. Consider now the variable $c$ among the $n$ nodes of the network, it becomes a Binomial random variable ($C$) with the probability given in Equation (4.3). Thereby, the network coverage intensity $C_n$, shown in Equation (3.7), is not a simple expectation, but rather an expectation of a function of a random variable.

$$Pr(C = j) = C_n^j \times \left(\frac{r}{a}\right)^j \times \left(1 - \left(\frac{r}{a}\right)\right)^{n-j} \tag{3.8}$$

where $C_n^j$ is the binomial coefficient, $r$ is the size of the sensing area of each sensor, $a$ is the size of the monitored area, and $\left(\frac{r}{a}\right)$ is the probability that each sensor covers a given point. In HOL, we formalize the Binomial random variable with $n$ trials and success probability $q = \left(\frac{r}{a}\right)$ as follows.

**Definition 3.9.**

```
⊢ ∀X p q n.  binomial_distr_rv X p q n =
        (real_random_variable X p) ∧
        (IMAGE X (p_space p) = IMAGE (λx.&x) (count (SUC n))) ∧
        (∀m.  &m ∈ (IMAGE X (p_space p)) ⇒
        (distribution p X {&m} = &(binomial n m) × q^m × (1 − q)^(n−m))).
```

where $X$ is a real random variable on the probability space $p$, and `IMAGE (λx.&x) (count (SUC n))` gives the support of the Binomial, while the operator `&` allows the conversion of the natural number $m$ into its extended number counterpart. The function `binomial`, used in the above definition, is the higher-order-logic formalization of the binomial coefficient for reals, which we defined in HOL as follows.

**Definition 3.10.**

```
⊢ ∀n k.  binomial n k = (binomial n 0 = (1:num)) ∧
            (binomial 0 (SUC k) = (0:num)) ∧
            (binomial (SUC n) (SUC k) = binomial n (SUC k) + binomial n k).
```

The coverage intensity of the whole WSN with $n$ nodes has been formally specified by the function `cvrge_intsty_network`, shown in Definition 3.11. This function takes as parameters: $X$: a random variable that returns an extended real number,

$p$: the probability space, $s$: the summation set used in Definition 3.8, $k$: the number of sub-networks, $C$: the random variable describing the number of covering nodes, $n$: the total number of nodes, and $q$: the probability that each sensor covers a given point.

**Definition 3.11.** *(The network coverage intensity)*

```
⊢ ∀X p k s C n q.  cvrge_intsty_network X p k s C n q =
  expectation p (λx.  cvrge_intsty_pt X p k s (Pr_cov k C n q x)).
```

where the function `expectation` designates the higher-order-logic formalization of the expectation of a random variable that returns an extended real, and the function `Pr_cov` is defined by the following equation.

$$\texttt{Pr\_cov k C n q x} = \left(1 - (1 - \frac{1}{\texttt{\&k}})^{\texttt{num(C n q x)}}\right). \tag{3.9}$$

The values `(num(C n q x))`, in the above definition, are the output values of the random variable `(C n q)`. The function `num`, used here, converts an extended real; `(&m)`, to its corresponding natural value `m`, using the real function `floor`. This conversion is mandatory since the power function in HOL takes as a coefficient a natural number, whereas the random variable function $(C\ n\ q)$ returns an extended real.

Based on the higher-order-logic formalizations developed so far, we have been able to formally verify the final network coverage intensity in the following theorem.

**Theorem 3.5.** *(The network coverage intensity)*

*Given a list of $k$ random variables $\{X_0, X_1, ..., X_{(k-1)}\}$ modelling the non-empty subsets generated by the randomized scheduling, each with the probability (Pr_cov), and a Binomial random variable describing the number of nodes covering a given point with a finite expectation, the network coverage intensity is:*

$$C_n = 1 - \left(1 - \frac{q}{k}\right)^n$$

```
⊢ ∀X p k s C n q.  (prob_space p) ∧ (events p = POW (p_space p)) ∧
  (0 < q < 1) ∧ (1 ≤ n) ∧ (1 < k) ∧
  FINITE s ∧ (CARD s = k) ∧ (sn_covers_p (C n q) p q n) ∧
  (expectation p (C n q) ≠ PosInf) ∧
  (expectation p (C n q) ≠ NegInf) ∧
  (∀i x.  (i ∈ s) ∧ (x ∈ p_space p) ⇒
  sbst_non_empty_rv (X (Pr_cov k C n q x) i) p (Pr_cov k C n q x))
  ⇒ (cvrge_intsty_network X p k s C n q = Normal (1 − (1 − q/&k)^n)).
```

where

- The assumption `(events p = POW (p_space p))` describes the set of events to be the power set of the sample space $\Omega$.
- The assumptions `(1 ≤ n)` ensures that the WSN include at least one node, while `(0 < q < 1)` checks that the probability $q$ lies in [0..1].
- `sn_covers_p` is the Binomial random variable (Definition 3.9) with a finite expectation, i.e., `(expectation p (C n q) ≠ PosInf) ∧ (expectation p (C n q) ≠ NegInf)`. The variables `(PosInf)` and `(NegInf)` are the higher-order-logic formalizations of positive infinity and negative infinity, respectively.
- The function `(sbst_non_empty_rv (X (Pr_cov k C n q x) i) p (Pr_cov k C n q x))` is the function specified in Definition 3.7 where the input probability function `(Pr_cov k C n q x)` is specified in Equation (3.9).

*Proof.* The proof of Theorem 3.5 is firstly based on Theorem 3.3 together with the linearity of the expectation property, which has been already verified for the proof of Theorem 3.4. We then performed the verification of both produced expectations for a constant random variable, and the function $f\_fct$ of the Binomial random variable $C$ (Theorem 3.6). It has been also necessary to show that the expectation of the function of random variable is finite which further involved operations on integral by backchaining. Finally, a considerable amount of real analysis associated to the Binomial theorem for reals (Theorem 3.7), and to the summation function was required to complete the main proof.

**Theorem 3.6.**

$\vdash$ ∀C p q n k.

(prob_space p) ∧ (events p = POW (p_space p)) ∧ (0 < q < 1) ∧

(1 ≤ n) ∧ (1 < k) ∧ (sn_covers_p C p q n)

⇒ (expectation p (λx. f_fct (num (C x)) k) = Normal $(1 - \frac{q}{(\&k)})^n$).

where the function f_fct is defined as follows

$$\texttt{f\_fct}\ \texttt{x}\ \texttt{k} = \texttt{Normal}\left(1 - \frac{1}{\texttt{k}}\right)^{\texttt{x}}. \tag{3.10}$$

*Proof.* The proof of Theorem 3.6 has been possible using intermediate results on the injectivity of some of the functions, as well as, some properties related to the random variables functions. A lot of reasoning associated with the use of extended real and the floor function, has also been required.

**Theorem 3.7.**

$\vdash$ ∀a b n. $(a + b)^n = \sum_{i=0}^{n}(\lambda i.\ \&(\text{binomial } n\ i) \times a^{(n-i)} \times b^i)$.

In this section, we presented our higher-order-logic formalizations of the $k$-set randomized scheduling for wireless sensor networks, using the recently developed probability theory available in the HOL theorem prover [60]. These formalizations have been then very useful to formally reason about the coverage performance properties. The corresponding HOL code of the current formalizations is available at [19]. In the next section, we will illustrate how the developed generic theorems extremely facilitate the formal analysis of real-world WSN applications.

## 3.5 Application: Forest Fire Detection

Forest fires are considered to be one of the worst terrific disasters causing a lot of environmental degradations. According to recent statistics [82], more than $100,000$ wild fires are annually reported throughout the world. For example, in Tunisia, 103 fires destroyed 287 hectares of forests just between May 1, 2012 and July 25, 2012 [66]. For early detection of wild fires and thus their prevention, robust surveillance systems satisfying critical real-time constraints are required. More particularly, these systems should be able to ensure a quick and accurate detection of any fire breakthrough. In this respect, wireless sensor network technology meets all these requirements and has been hence extensively explored for the detection of forest fires [18, 35, 5, 92, 95, 45].

Thereafter, we are interested in formally analyzing the coverage performances of a forest fire detection system using wireless sensor network. Because of the harsh nature of the target field, a random deployment by air-dropping sensors is obviously much more practical in this context. The main goal of the dispersed nodes is to sense and communicate values of temperature, humidity and barometric pressure to a base station. A processing step is then performed in order to alarm the final user in case of abnormal values. Hence, using a WSN to detect forest fires has the merits

to guarantee a large monitoring area with an efficient real-time surveillance through automatic alarms.

Due to the safety-critical feature of the target application, the deployed WSN has to remain alive for a long period while ensuring a good coverage of any fire breakthrough. Nevertheless, most of the existing systems for forest fire detection using WSN suffer from serious lifetime limitations. For example, the system, presented in [33], reported that a sensor deployed in a wild environment without a sleeping cycle, cannot be kept alive for more than 5 days. In order to extend the whole network lifetime, the $k$-set randomized scheduling algorithm has been proposed for use in the given forest fire detection application [74, 92]. In the specified application, the nodes have a sensing area $r = 30$, and are deployed into forest region of size $a = 100m \times 100m$, whereas the success probability $q$ of a sensor covering a point, is $q = \frac{r}{a} = 0.003$.

Based on our theoretical development done in the previous section, we now conduct a formal asymptotic analysis of the probabilistic coverage based on the key design parameters: $n$; the total number of sensor nodes and $k$; the number of scheduled sub-networks. This important analysis is made possible thanks to Theorem 3.5 which gives a clear relationship between the network coverage intensity $C_n$ and the two parameters $n$ and $k$. For that purpose, we are going to first prove the generic case and then instantiate it for the given forest fire application. Hence, the generic network coverage intensity `(cvrge_intsty_network X p s k C n q)` is simply denoted by `(Cn_wsn X p s k C n q)`. Besides, the coverage of our forest fire detection application can be specified by specializing Definition 3.11 since it describes the generic coverage intensity of a WSN using a $k$-set randomized scheduling algorithm.

**Definition 3.12.**

```
⊢ ∀X p s k C n q.
  Cn_frst X p s k C n = cvrge_intsty_network X p s k C n (0.003).
```

Then, we can easily check in HOL that `(Cn_frst X p s k C n)` equals

$$\texttt{Normal}\left(1-\left(1-\frac{(0.003)}{\texttt{k}}\right)^{\texttt{n}}\right) \tag{3.11}$$

It is important to note that, for space constraints and in all the next asymptotic analysis, we will restrict the presented assumptions to the main mathematical ones related to the used variables. Whereas, the complete HOL code for these asymptotic analysis can be found in [19].

### 3.5.1 Formal Analysis based on the Number of Nodes

In a randomly-scheduled WSN, the number of deployed nodes $n$ is known to be a common critical attribute which has a significant impact on both energy and coverage. Intuitively, deploying too many nodes will surely lead to a waste of energy since some of the regions would be simultaneously covered by many sensors at once. On the other hand, deploying too few nodes may not guarantee a good coverage if, for example, a given point of the area does not have any of the deployed sensors in its surrounding area. In the next analysis, we formally confirm this intuition through verifying the coverage behavior of the whole network based on the number of nodes $n$.

Targeting a network coverage intensity $Cn\_wsn$ of at least $t$, we verify, in Lemma 3.1, the minimum number of nodes; $n_{min}$, that are required to deploy for a given number of subsets $k$.

**Lemma 3.1.** *(The lower bound on the number of nodes $n$ given $Cn\_wsn = t$)*

$\vdash \forall$X p s k C n q t. (1 $\leq$ n) $\wedge$ (1 < k) $\wedge$ (0 < q < 1) $\wedge$

(0 < t < 1) $\wedge$ (Normal t $\leq$ Cn_wsn X p s k C n q)

$\Rightarrow \left\lceil \frac{\ln(1-\mathtt{t})}{\ln\left(1-\frac{\mathtt{q}}{\mathtt{k}}\right)} \right\rceil \leq$ &n.

*Proof.* The higher-order-logic proof of the above lemma is based on some properties of transcendental functions along with some arithmetic reasoning.

Next, we focus on studying the network coverage performance according to the variation on the number of nodes $n$. Hence, we have been able to formally verify, in Lemma 3.2, that the network coverage intensity $Cn\_wsn$ is an increasing function of $n$, i.e., a larger $n$ value leads to a better coverage intensity. In this case, more points of the monitored area are expected to be covered, since it is more likely that many more sensor nodes are deployed in its surrounding area.

**Lemma 3.2.** *($Cn\_wsn$ is an increasing sequence versus $n$)*

$\vdash \forall$X p s k C q. (1 < k) $\wedge$ (0 < q < 1)

$\Rightarrow$ (mono_incr ($\lambda$n. real(Cn_wsn p X k s C n q))).

where the function `real` is used to convert the network coverage intensity of type extended real to its corresponding real value, and `mono_incr` is the HOL definition of an increasing sequence, which we present in Definition 3.13.

*Proof.* The proof is based on Theorem 3.5 and some real analysis.

**Definition 3.13.** *(increasing sequence)*

$\vdash \forall$f. mono_incr f $\Leftrightarrow \forall$n. f n $\leq$ f (SUC n).

We can deduce hence that under the randomized scheduling, which divides the network into a given number $k$ of sub-networks, any network coverage intensity $Cn\_wsn$ can be achieved by increasing the number of deployed nodes $n$.

In Lemma 3.3, we formally check the asymptotic property regarding the number of nodes $n$, that is when $n$ is very large. Hence, as $n$ becomes infinite, $Cn\_wsn$ approaches its ideal value 1.

**Lemma 3.3.** *(Limit of $Cn\_wsn$ when $n$ is very large)*

$\vdash$ ∀X p s k C q. (1 < k) ∧ (0 < q < 1)
$\Rightarrow$ ( $\lim_{\mathtt{n}\to+\infty}$ (λn. real(Cn_wsn X p s k C n q)) = 1).

*Proof.* We proved Lemma 3.3 using basic properties from the sequence theory.

Lemma 3.1 can be used to deduce useful results for the given forest fire detection application using WSN. Hence, suppose that a network coverage intensity of at least 70% is targeted [89], then the lower bound on the number of required nodes $n$ is verified in Lemma 3.4.

**Lemma 3.4.** *(The lower bound on the number of nodes $n$ given $Cn\_frst = 0.7$)*

$\vdash$ ∀X p s k C n. (1 ≤ n) ∧ (1 < k) ∧ (Normal (0.7) ≤ Cn_frst X p s k C n)
$\Rightarrow \left\lceil \frac{\ln(1-0.7)}{\ln\left(1-\frac{0.003}{\mathtt{k}}\right)} \right\rceil \leq$ &n.

More concretely, if the randomized scheduling splits the set of nodes into ($k = 4$) sub-networks, at least 1606 nodes are required to be deployed over the forest area in order to achieve a network coverage intensity of 70%.

In addition, we established, in Lemmas 3.2 and 3.3, that any network coverage intensity $Cn\_wsn$ can be achieved by increasing the number of deployed nodes $n$, regardless of the input values $k$ and $q$. These results can be easily verified for the network coverage intensity, $Cn\_frst$, in the context of the given forest fire application (Lemmas 3.5 and 3.6).

**Lemma 3.5.** *(Cn_frst is an increasing sequence versus n)*

```
⊢ ∀X p s k C. (1 < k)
  ⇒ (mono_incr (λn.  real(Cn_frst X p s k C n))).
```

**Lemma 3.6.** *(Limit of Cn_frst when n is very large)*

```
⊢ ∀X p s k C. (1 < k)
  ⇒ ( lim_{n→+∞} (λn.  real(Cn_frst X p s k C n)) = 1).
```

### 3.5.2 Formal Analysis based on the Number of Subsets

According to Lemmas 3.2 and 3.3, enhancing the coverage capacities of the deployed WSN, is usually possible through the deployment of more nodes. Nevertheless, after the first deployment, the number of sensor nodes becomes known and fixed. Besides, a second deployment would be very costly in the context of wild fields such as forests, since nodes are generally deployed by throwing them from an airplane. Considering a fixed number of nodes $n$, we formally study now the effect of the number of sub-networks $k$ on the coverage performance of the whole network. In particular, we explore the asymptotic network coverage as well as many other useful properties according to the number of subsets $k$.

Investigating the impact of the $k$-values on coverage, the general intuition about the randomized scheduling approach is as follows: with the increase on the number of subsets $k$, the individual sensor energy decreases since there will be probably few sensors in each subset. On the other hand, too many scheduled sub-networks means also a shorter schedule round, which in turn normally translates to a worse network coverage intensity $Cn\_wsn$. Based on these remarks, we next make a formal derivation of the limiting coverage according to the parameter $k$.

Hence, we have been able to first formally verify, in Lemma 3.7, that a smaller $k$ value induces a larger network coverage *Cn_wsn*, i.e., *Cn_wsn* decreases while increasing $k$.

**Lemma 3.7.** *(Cn_wsn is a decreasing sequence versus k)*

```
⊢ ∀X p s C n q.  (1 ≤ n) ∧ (0 < q < 1)
  ⇒ (mono_decr (λk.  real (Cn_wsn X p s k C n q))).
```

where the HOL function `mono_decr` is given in Definition 3.14.

*Proof.* Similar to Lemma 3.2, the above proof can be easily deduced using Theorem 3.5 and some real analysis.

**Definition 3.14.** *(Decreasing sequence)*

```
⊢ ∀ f.  mono_decr f ⇔ ∀n.  f (SUC n) ≤ f n.
```

As expected, we also formally confirm, in Lemma 3.8, that given a fixed number of nodes $n$, the network coverage intensity *Cn_wsn* goes to 0 when $k$ becomes very large. In other words, the network coverage intensity *Cn_wsn* definitely decreases when the WSN is partitioned into a quite large number of sub-networks $k$.

**Lemma 3.8.** *(Limit of Cn_wsn when k is very large)*

```
⊢ ∀X p s C n q.  (1 ≤ n) ∧ (0 < q < 1)
  ⇒ (lim_{k→+∞} (λk.  real (Cn_wsn p X p s k C n q)) = 0).
```

*Proof.* The proof of the above lemma is deduced using intermediate results associated to real and sequential limits. The above three lemmas, showing the relationship between the $k$-values and the probabilistic coverage of the network, are very consistent with our intuition about the randomized scheduling. They can be hence useful to deduce interesting results in the context of the given forest fire detection application.

Consequently, for our forest fire detection application, increasing $k$ surely saves more energy, but leads to a very low network coverage intensity $Cn\_frst$ (Lemma 3.9), which is not good at all.

**Lemma 3.9.** *($Cn\_frst$ is a decreasing sequence versus $k$)*

```
⊢ ∀X p s C n.  (1 ≤ n)
  ⇒ (mono_decr (λk.  real (Cn_frst X p s k C n))).
```

In addition, we reconfirm the result of Lemma 3.8 using Lemma 3.10, i.e., increasing the number of deployed nodes $n$ gives smaller network coverage and hence a poor performance of the deployed application.

**Lemma 3.10.** *(Limit of $Cn\_frst$ when $k$ is very large)*

```
⊢ ∀X p s C n.  (1 ≤ n)
  ⇒ (lim_{k→+∞} (λk.  real (Cn_frst X p s k C n)) = 0).
```

The randomized scheduling is thus a dynamic approach which provides performance adjustments of the deployed WSN application according to the value of $k$.

### 3.5.3 Formal Analysis based on Uniform Partitions

The randomness in the node scheduling approach leads to sub-networks of different sizes with respect to the number of nodes. Obviously, the ideal case arises when the algorithm makes a fair organization of the network into subsets of the same size. In this case, the parameters $k$ and $n$ are proportional so that the number of nodes $n$ can be written as $k \times m$, where $m$ is the number of nodes per subset. In what follows, we closely investigate the asymptotic performance behavior of the k-set randomized algorithm regarding coverage in the case of a *uniform* split of the nodes.

In particular, as the number of sub-networks $k$ goes infinite, the upper limit of the network coverage *Cn_wsn* has been formally verified in Lemma 3.11.

**Lemma 3.11.** *(Limit of Cn_wsn if n and k are proportional)*

$$\vdash \forall \mathtt{X\ p\ s\ C\ m\ q.}\quad (0 < \mathtt{q} < 1)$$
$$\Rightarrow \lim_{\mathtt{k}\to+\infty} \quad (\lambda \mathtt{k.}\quad \mathtt{real(Cn\_wsn\ X\ p\ s\ k\ C\ (m \times k)\ q))} = 1 - \mathtt{e}^{-\mathtt{q}\times(\&\mathtt{m})}.$$

*Proof.* In the HOL theorem prover, the proof of the above lemma has been quite challenging requiring the important mathematical result stated in Lemma 3.12, which has not been available in HOL and we had to prove it part of our development.

**Lemma 3.12.** *(Exponential limit)*

$$\vdash \forall \mathtt{x.}\quad \lim_{\mathtt{k}\to+\infty} (1 + \tfrac{\mathtt{x}}{\mathtt{k}})^{\mathtt{k}} = \mathtt{e}^{\mathtt{x}}.$$

*Proof.* The main prerequisite for the proof of the above result consists in Lemma 3.13. For that purpose, we first proceed by considering the 2 sequences $S_n = \sum_0^n \frac{x^k}{k!}$ and $U_n = (1+\frac{x}{n})^n$, get their difference $|S_n - U_n|$, show that $|S_n - U_n| \leq \frac{x^2}{n} \times e^{|x|}$ and then apply Lemma 3.13 such that $H = |S_n - U_n|$ and $V = \frac{x^2}{n} \times e^{|x|}$. The proof steps involve thus long complex real analysis including summation, some factorial properties, real product and arithemetic series, as well as, many properties related to the sequence convergence.

**Lemma 3.13.** *(Convergence property for 2 sequences)*

$$\vdash \forall \mathtt{H\ V.}\ (\forall \mathtt{n.}\quad 0 \leq \mathtt{H\ n}) \wedge (\forall \mathtt{n.}\quad \mathtt{H\ n} \leq \mathtt{V\ n}) \wedge (\mathtt{V} \to 0)$$
$$\Rightarrow (\mathtt{H} \to 0).$$

*Proof.* To prove Lemma 3.13, we start by rewriting with the limit definition and then apply some real properties.

Based on Lemma 3.11, we can hence verify that when $m$ becomes very very large, the uniform network coverage will surely approach 1. Such result is considered as a second verification of Lemma 3.3 in the specific case where $n$ and $k$ are proportional.

**Lemma 3.14.** *(Limit of uniform coverage Cn_wsn)*

$\vdash \forall$X p s C q. (0 < q < 1)

$\Rightarrow \lim_{m \to +\infty} (\lambda m. \lim_{k \to +\infty} (\lambda k.$ real(Cn_wsn X p s k C (m $\times$ k) q)) = 1.

Finally, we show that the two results, obtained above, are also valuable for the given forest fire detection application through a simple instantiation of the input parameter $q$ by its value. The corresponding HOL analysis is given in the following 2 lemmas.

**Lemma 3.15.** *(Limit of Cn_frst if n and k are proportional)*

$\vdash \forall$X p s C m.

$\Rightarrow \lim_{k \to +\infty} (\lambda k.$ real(Cn_frst X p s k C (m $\times$ k))) = 1 - $e^{-(0.003) \times (\&m)}$.

**Lemma 3.16.** *(Limit of uniform coverage Cn_frst)*

$\vdash \forall$X p s C.

$\lim_{m \to +\infty} (\lambda m. \lim_{k \to +\infty} (\lambda k.$ real(Cn_frst X p s k C (m $\times$ k))) = 1.

The formal analysis of the behavior of the presented forest fire application using WSN, done in this section, is a very interesting illustration of the useflness of our coverage developments. Table 3.2 summarizes the set of properties verified for the corresponding application. Unlike traditional analysis techniques for the validation of a WSN for forest fire detection, using the $k$-set randomized scheduling algorithm, our approach is much more efficient. While paper-and-pencil based analysis [92] or simulation [95] cannot guarantee the correctness of the scheduling performance results, the reported theorems in this chapter are accurate given the inherent soundness

of theorem proving and its generic nature, e.g., the coverage intensity for any given randomly-scheduled WSN application can be computed by instantiating Theorem 3.5 with appropriate values of $n$ and $k$. Contrarily, simulation is usually restricted to specific network configurations, while probabilistic model checking is frequently using parameter abstraction in order to cope with the state-space explosion problem. Moreover, for each of the formally verified theorems, the set of required assumptions is clearly stated so that there is no doubt about missing a critical assumption. Such aspect can never be ensured in simulation and model checking where many assumptions can be taken into account without explicitly mentioning them.

| **Verified Property** | **Formulation** |
|---|---|
| The lower bound on $n$ given $(Cn\_frst = t)$ | $n \geq \frac{\ln(1-t)}{\ln\left(1-\frac{q}{k}\right)}$ |
| $Cn\_frst$ is an increasing sequence versus $n$ | mono_incr $(Cn\_frst)$ |
| $Cn\_frst$ approaches 100% when $n$ is very large | $\lim_{n \to +\infty} Cn\_frst = 1$ |
| $Cn\_frst$ is a decreasing sequence versus $k$ | mono_decr $(Cn\_frst)$ |
| $Cn\_frst$ definitely decreases when $k$ is very large | $\lim_{k \to +\infty} Cn\_frst = 0$ |
| Limit of $Cn\_frst$ if uniform partitions $(n = k \times m)$ | $1 - e^{-q \times m}$ |

Table 3.2: Coverage Analysis of the Forest Fire Application

## 3.6 Summary and Discussions

The work, presented throughout this chapter, constitutes the first step towards our higher-order-logic theorem prover based approach for the formalization of the $k$-set randomized scheduling within the sound core of the HOL theorem prover (see Figure 1.2). For that purpose, we provided the fundamental formalizations of the randomized scheduling first and then based on them we developed our formalizations of the two key coverage performance measures, i.e., the coverage intensity of a specific point and the network coverage intensity. We have been also able to show the practical

effectiveness of our formalizations on a WSN application for forest fire detection.

Compared to probabilistic model checkers where statistical properties are not so accurately specified, we have been able to achieve formal and precise analysis of the network coverage as a statistical measure of the coverage intensity for a specific point. In addition, the formal performance analysis of the coverage behavior of the forest fire application clearly show the effectiveness of our theoretical developments. Thanks to the proposed approach, this is the first time, to the best of our knowledge, that the performance analysis of this kind of a WSN application is analyzed in a complete formal manner. It has been thus possible to formally provide a generic asymptotic analysis for all possible values of the design parameters, and in the specific case of the considered forest fire application. Furthermore, such verification enables reliable asymptotic reasoning of the deployed WSN. It is important to note here that the presented application is a simple case study illustrating the practical interest of our work, but the claimed generic results can be obviously valuable for any other WSN application as well. Besides, the coverage behavior of a randomly-scheduled WSN for volcanic earthquakes detection has been formally analyzed in [22].

The HOL development consumed about 1500 lines of code for the formal analysis of the randomized scheduling, the coverage performance properties and the WSN application for forest fire detection. Many challenges have been encountered in the current work. Firstly, although the higher-order-logic modelling seem to depend on simple discrete random variables, the major difficulty was to understand the initial probabilistic model of the algorithm and translate it into higher-order logic. This includes the efforts involved to establish, based on some abstract mathematical models [13, 52, 89], the right formalizations using the appropriate random variables and higher-order-logic functions. Moreover, the existing probabilistic models are generally

not so reliable either regarding the complete set of assumptions or the correctness of the mathematical analysis done by hand, which may include human errors. Neither the assumptions, nor the list of the design parameters were exhaustive in the existing textbooks [13, 52, 89]. Moreover, it is very common that some mathematical steps, taken as granted for specialists, require great investigation from a reader's perspective. Indeed, the theoretical flow of the analysis, usually based on a lot of intuition and restricted to some mathematical steps, found to be confusing for higher-order-logic formalization. However, to sucessfully achieve our main formalization task, every step, in the original analysis, has to be deeply investigated at the mathematic level in order to correctly map it into HOL. Such difficulties have been, for example, noticed when formally specifying the network coverage (Definition 3.11). There was no real explication about the network coverage as the expectation of a function of a Binomial random variable. It has been directly used within the analysis and a lot of mathematical efforts have been thus involved to find out the main mathematical relations. Besides, the higher-order-logic definition of expectation, available in the HOL theorem prover, has been found to be general enough to handle the expectation of a function.

Secondly, the HOL library of theorems cannot be regarded as exhaustive and thus it may happen that a foundational result to verify a desired theorem is missing. At the outset, the formal verification time and effort becomes quite high. Even a very good knowledge of the prover abilities does not permit to completely avoid such problem. Therein, the proof of the Bionimal theorem for reals (Theorem 3.7), required to complete the main proof of coverage (Theorem 3.5), is a very good illustration. Also, the missing theorem $\lim_{k \to +\infty} (1 + \frac{x}{k})^k = e^x$, has made the proofs of Lemmas 3.11 and 3.14, quite tedious consuming on their own 500 lines of HOL code.

It is worthy to remind that early formalizations of the $k$-set randomized scheduling algorithm and the coverage attributes, have been subject to migration into the new HOL probability theory developed in the HOL theorem prover [60]. At the beginning of the thesis, we built upon another probabilistic framework developed in the HOL theorem prover [34] to formally analyze the $k$-set randomized scheduling algorithm. In [20], we presented the HOL formalization of the corresponding coverage properties, whereas the efficiency of our higher-order-logic developments have been shown on a real-world WSN application for forest fire detection [21]. Recently, a more generic formalization of probability theory has been made available in the HOL theorem prover [60]. Since our HOL formalizations constitute the first part of the whole methodology, described in Figure 1.2, we decided thus to migrate our previous higher-order-logic formalizations into the new HOL probability theory. Such decision, even difficult and time consuming, has been primarily motivated by the fact that we are targeting more evolutive probabilistic analysis of the $k$-set randomized scheduling with the formalization of further performance aspects that will be shown in the rest of this thesis. These aspects should require some probabilistic features which are not available in [34].

Due to fundamental differences in the foundations of the two probability theories in [34] and [60], the current resulting formalizations is completely different from the previous one [20]. The new probability theory allows indeed to cater for arbitrary probability spaces and is thus more generic and complete compared to the previous formalization in which the probability space has to be the universe of a set. Moreover, the specification of the randomized algorithm has been found to be much more straightforward with [60]. Unlike the work in [20], the developed proofs also required

much less reasoning about sets and lists producing thus less lengthy proofs. However, these proofs have been more laboured involving usually results from the three HOL theories: Lebesgue, measure and extended reals. An extensive understanding of the inherent theoretical foundations of [60] was thus required to successfully achieve the target formalizations in the HOL theorem prover. Hopefully, the existing results from the formalized probability theory helped us to keep the amount of proof efforts reasonable.

Finally, it will be very interesting to formally check the relationship between the coverage and detection performances showing that coverage can reflect detection [52]. These interesting characteristics can be analyzed based on the formalization of the detection properties, which will be elaborated in the next chapter.

# Chapter 4

# Detection Analysis

In this chapter, we describe the fundamental formalizations of the key detection metrics of randomly-deployed wireless sensor networks using the randomized scheduling of nodes. For that, based on the probability theory available in the HOL theorem prover, we first formally reason about the intrusion period of any occurring event. Then, we build upon this characteristic to formally verify the detection probability and the detection delay. For illustration purposes, we formally analyze the detection performance of a WSN deployed for border security monitoring.

## 4.1 Formalization of the Intrusion Period

Based on the description of the $k$-set randomized algorithm, given in Chapter 2, the $k$ formed subsets of nodes $\{S_i, 0 \leq i \leq (k-1)\}$ are disjoint and work alternatively within their scheduling time cycles/slots $\{T_i, 0 \leq i \leq (k-1)\}$. In a wireless sensor network, an event, e.g., the outbreak of a fire in a forest, randomly occurs at any time. The duration of this event, denoted $L$, will obviously overlap with a number of scheduling cycles $T$ (see Fig. 4.1). We are interested in formally verifying the average

number of overlapping cycles with an intrusion period $L$.

Consider $s$; the remainder of the intrusion period $L$ in terms of the number of slots $T$. Mathematically, by expressing $L$ in terms of $T$, the variable $s$ can be specified by the following equation [89].

$$s = \frac{L}{T} + 1 - \left\lceil \frac{L}{T} \right\rceil \tag{4.1}$$

Let $t_0$ be any reference time and $t_z$ the beginning of the intrusion event. Fig. 4.1 shows how the interval $[t_0, t_0 + T]$ is split into two regions according to $s$. Hence, if $t_z$ belongs to the interval

- $[t_0, t_0 + (1 - s) \times T]$, then $L$ overlaps $\lceil \frac{L}{T} \rceil$ with the probability $(1 - s)$.
- $]t_0 + (1 - s) \times T, T[$, then $L$ overlaps $\left(\lceil \frac{L}{T} \rceil + 1\right)$ with the probability $s$.

As an example, let us take an intrusion event which lasts for a duration $L = 2.8T$, as illustrated in Fig. 4.1. Hence, $L$ overlaps either $\lceil \frac{L}{T} \rceil = \lceil \frac{2.8T}{T} \rceil = 3$ cycles with the probability $(1 - s = 0.2)$, or 4 cycles with the probability $(s = 0.8)$.

We can now formalize in higher-order logic the average number of overlapping cycles with an intrusion period $L$. For this purpose, we proceed by first formally specifying the corresponding random variable which describes the number of overlapping cycles within an intrusion period $L$. Based on the above description, we model this behavior by a random variable denoted by $IT$. This random variable can be characterized in higher-order logic by the following predicate `intr_distr_rv` on the probability space $p$ such that the image of $IT$ on `(p_space p)` is in $\{\lceil \frac{\mathtt{L}}{\mathtt{Ts}} \rceil; \lceil \frac{\mathtt{L}}{\mathtt{Ts}} \rceil + \mathtt{1}\}$, and its probability distribution over $\{\lceil \frac{\mathtt{L}}{\mathtt{Ts}} \rceil\}$ is $(1 - s)$.

**Definition 4.1.**

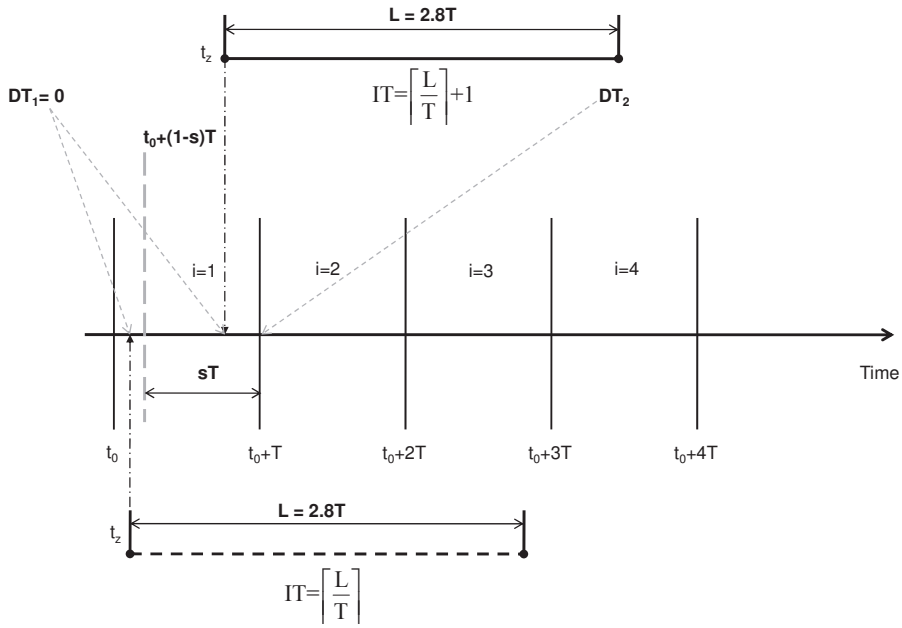


Figure 4.1: Detection Analysis [89].

```
⊢ ∀IT p s (L:real) (Ts:real).  intr_distr_rv IT p s L Ts =
                                 (real_random_variable IT p) ∧
                                 (IMAGE IT (p_space p) = {⌈L/Ts⌉; ⌈L/Ts⌉ + 1}) ∧
                                 (distribution p IT {⌈L/Ts⌉} = 1 - s).
```

The definition above accepts five parameters: $IT$: a random variable that returns an extended real number, $p$: the probability space, $s$: the variable specified in Equation (4.1), $L$: the length of the intrusion period, and $Ts$: the length of a time slot. Please note that for the sake of simplicity, we take $s$ as a separate variable, although it depends only on $L$ and $Ts$.

It is important to note that the original specification [89] does not give any indication about the random variable $IT$. Indeed, the reference textbook was just reasoning on a binary random variable, taken intuitively, with values in the set $\{0; 1\}$.

Clearly, the latter random variable is completely different from the random variable, $IT$, that we specify to describe the number of overlapping cycles with an intrusion period $L$, which is $\{\lceil \frac{L}{Ts} \rceil ; \lceil \frac{L}{Ts} \rceil + 1\}$.

We can now formally verify, in Theorem 4.1, the main property of interest, i.e., the average number of overlapping cycles with an intrusion period $L$ as the expectation of the random variable $IT$.

**Theorem 4.1.**

$\vdash$ ∀IT p s L Ts. (0 < Ts) ∧ (0 < L) ∧ (intr_distr_rv IT p s L Ts)

$\Rightarrow$ (expectation p IT = Normal($\frac{\mathtt{L}}{\mathtt{Ts}} + 1$))).

where the function `expectation`, used in the above theorem, designates the higher-order-logic formalization of the expectation of a random variable that returns an extended real, whereas, the HOL function `Normal` is used to convert a real value to its corresponding value in an extended real. The proof of Theorem 4.1 is based on the verification of the probability distribution on $\{\lceil \frac{\mathtt{L}}{\mathtt{Ts}} \rceil\}$ and $\{(\lceil \frac{\mathtt{L}}{\mathtt{Ts}} \rceil + 1)\}$, along with some analysis on extended real.

## 4.2 Formalization of the Detection Probability

In a randomly-scheduled WSN, the probability of detecting an intrusion event (D) is usually specified using the probability of the event "being unable to detect an intrusion (UD)" [93, 89]. Thus, using the probability rule of complement, we have:

$$Pr(D) = 1 - Pr(UD) \tag{4.2}$$

The detection performances of a wireless sensor network mainly depends on the number of nodes covering the occurring events. In Chapter 3, we demonstrated that

the number of nodes covering a point where the intrusion event happens is a Binomial random variable (C) with the following probability.

$$Pr(C = j) = C_n^j \times \left(\frac{r}{a}\right)^j \times \left(1 - \left(\frac{r}{a}\right)\right)^{n-j} \frac{n!}{j!\,(n-j)!} \tag{4.3}$$

where $C_n^j$ is the binomial coefficient indexed by the number $j$ of nodes covering an occurring event and the total number $n$ of deployed nodes. Please note that we refer to the above random variable (C), by $c$ throughout the next analysis.

Given that the events $\{c = j, 0 \leq j < n\}$ form a partition of the entire sample space ($\Omega$ `= p_space p`), we can establish from Equation (4.2), using the law of total probability (Equation (2.7)), that

$$Pr(D) = 1 - \sum_{j=0}^{n} Pr(UD \mid c = j) \times Pr(c = j) \tag{4.4}$$

where $Pr(UD \mid c = j)$ is the conditional probability of being unable to detect the intrusion event given that $(c = j)$.

Based on the analysis done in [89], we discuss the probability $Pr(UD \mid c = j)$ according to the values of $j$, i.e., the number of sensor nodes covering a point when the intrusion event happens, and $L$, i.e., the intrusion period.

- **Case 1.** $(j = 0)$ and for any duration $L$, $Pr(UD \mid c = 0) = 1$. Given that there is 0 covering nodes, it is sure that an intrusion event can never be detected.

- **Case 2.** $\{0 < j \leq n\} \cap \{L \geq (k-1) \times Ts\}$, $Pr(UD \mid c = j) = 0$.
  Since there are $k$ working rounds, each of length $T$, an event lasting more than $(k-1) \times T$, and having at least one covering active node $(0 < j)$ will be always detected.

- **Case 3.** $\{0 < j \leq n\} \cap \{L < (k-1) \times T)\}$, $Pr(UD \mid c = j) \neq 0$. An event

lasting less than $(k-1) \times T$ with at least one covering active node $(0 < j)$, will be usually detected with a given probability which is not null.

By extracting the first term (j=0) of the summation in Equation (4.4), we obtain

$$Pr(D) = 1 - (Pr(UD \mid c = 0) \times Pr(c = 0) + \sum_{j=1}^{n} Pr(UD \mid c = j) \times Pr(c = j)) \quad (4.5)$$

According to case 1, we have $Pr(UD \mid c = 0) = 1$, and we hence can rewrite Equation (4.5), using Equation (4.3), as

$$Pr(D) \;=\; 1 - ((1-q)^n + \sum_{j=1}^{n} Pr(UD \mid c = j) \times Pr(c = j)) \quad (4.6)$$

In the following, we are interested in formally verifying the detection probability $Pr(D)$ for occurring events of any length $L$. More particularly, we focus on the formalization of the summation term of Equation (4.6). For that purpose, we distinguish 2 cases, i.e., $\{L < (k-1) \times T)\}$ and $\{L \geq (k-1) \times T)\}$.

### 4.2.1 Detection Probability for Short Events

The mathematical model for the performance analysis of the detection probability has directly given the final result of Equation (4.6). Only few explanations related to pure mathematical steps can be found in [93]. However, in order to achieve accurately the higher-order-logic formalizations of Equation (4.6), we require to reason about all the implicit steps related to the probabilistic analysis.

According to the intrusion period analysis, done in Subsection 4.1, we know that the intrusion period $L$, for events lasting $\{L < (k-1) \times T)\}$, may overlap either

$\lceil \frac{L}{T} \rceil$ or $(\lceil \frac{L}{T} \rceil + 1)$ scheduling cycles $T$. Thus, an intrusion event which lasts $L$, cannot be detected either when $L$ overlaps $\lceil \frac{L}{T} \rceil$ cycles, or when $L$ overlaps $(\lceil \frac{L}{T} \rceil + 1)$ cycles. Using the following events

- $A_{12}$ = The intrusion period $L$ overlaps $\lceil \frac{L}{T} \rceil$ cycles.
- $A_{22}$ = The intrusion period $L$ overlaps $(\lceil \frac{L}{T} \rceil + 1)$ cycles.

It is possible to express the whole event of non-detection, denoted by $UD$, as follows

$$UD = UD \cap (A_{12} \cup A_{22}) \tag{4.7}$$

Now, applying Equations (2.4) and (2.6) to $Pr(UD \mid c = j)$ in Equation (4.6), along with the fact that the events $A_{12}$ and $A_{22}$ are disjoint, we get the following result.

$$\begin{aligned} Pr(UD \mid c = j) &= Pr(UD \mid A_{12} \cap (c = j)) \times Pr(A_{12} \mid c = j) + \\ &\quad Pr(UD \mid A_{22} \cap (c = j)) \times Pr(A_{22} \mid c = j) \end{aligned} \tag{4.8}$$

Intuitively, for a given intrusion event of length $L$, the occurrence of the event ($A_{12}$ = L overlaps $\lceil \frac{L}{Ts} \rceil$ cycles), and the event $(c = j)$ describing that there are $j$ covering nodes, are governed by distinct and noninteracting physical processes [27]. Hence, the two events turn out to be independent. According to Equation (2.3), we get hence $Pr(A_{12} \mid c = j) = Pr(A_{12}) = Pr(IT = \lceil \frac{L}{Ts} \rceil)$, where $IT$ is the intrusion random variable as specified in Definition 4.1. Similarly, we obtain $Pr(A_{22} \mid c = j) = Pr(IT = \lceil \frac{L}{Ts} \rceil + 1)$. This allows us to rewrite the RHS of Equation (4.8) as

$$Pr(UD \mid A_{12} \cap (c = j)) \times Pr(A_{12}) + Pr(UD \mid A_{22} \cap (c = j)) \times Pr(A_{22}) \tag{4.9}$$

On the other hand, the event "$UD \mid A_{12} \cap (c = j)$" indicates the event of "being unable to detect an intrusion event" given that "the intrusion period $L$ overlaps $\lceil \frac{L}{Ts} \rceil$ cycles" and "there are $j$ covering nodes". Indeed, if an event, covered with $j$ nodes and overlapping $\left(h = \lceil \frac{L}{T} \rceil\right)$ rounds, is not detected, then it means that all the $j$ covering nodes miss the $h$ consecutive subsets. In other words, the sequence of $h$ subsets do not contain covering nodes. Such event is expressed by the following equation.

$$B_{h,c} = H_{1,c} \cap H_{2,c} \cap .. \cap H_{i,c} \cap .. \cap H_{h,c} = \left( \bigcap_{i=1}^{h} H_{i,c} \right) \tag{4.10}$$

where $H_{i,c}$ is the event that none of the $c$ covering sensor nodes belongs to the working subset $i$, i.e., $H_{i,c}$ is empty, and the set of events $\{H_{1,c}, H_{2,c}, ..., H_{h,c}\}$ is mutually independent. We say that a finite set of events is mutually independent if and only if every event is independent of any intersection of the other events [27]. The probability of the above event (Equation (4.10)) has been proved in [20], to be equal to $\left(\frac{k-h}{k}\right)^c$, where $k$ is the number of disjoint subsets.

Accordingly, Equation (4.6) becomes

$$Pr(D) = 1 - ((1-q)^n + \sum_{j=1}^{n} \left[ Pr(A_{12}) \times Pr(B_{\lceil \frac{L}{T} \rceil, j}) + Pr(A_{22}) \times Pr(B_{\lceil \frac{L}{T} \rceil + 1, j}) \right]) \tag{4.11}$$

Based on the above reasoning, we successfully verify, in Theorem 4.2, the final expression of the detection probability $Pr(D)$ for events lasting $\{L < (k-1) \times T\}$.

**Theorem 4.2.**

$$
\begin{aligned}
&\vdash \forall \mathtt{p\ X\ IT\ UD\_rv\ k\ q\ n\ s\ L\ Ts.}\ \ (\mathtt{prob\_space\ p}) \wedge (1 < \mathtt{k}) \wedge \\
&(1 \leq \mathtt{n}) \wedge (0 < \mathtt{q} < 1) \wedge (\mathtt{sn\_covers\_p\ X\ p\ q\ n}) \wedge (0 < \mathtt{Ts}) \wedge \\
&(0 < \mathtt{L}) \wedge (\mathtt{L} < \mathtt{\&(k\text{-}1)}\times\mathtt{Ts}) \wedge (0 < \mathtt{s} < 1) \wedge \\
&((\mathtt{udset\ n\ k\ s\ L\ Ts\ q}) \in \mathtt{events\ p}) \wedge (\mathtt{intr\_distr\_rv\ IT\ p\ s\ L\ Ts}) \wedge \\
&(\mathtt{sbst\_empty\_sch\_rv\ (UD\_rv\ (SUC\ i))\ p\ k\ c\ (SUC\ i)}) \wedge \\
&(\mathtt{indep\_rv\ p\ IT\ X\ Borel\ Borel}) \wedge (\mathtt{cond\_prob\ p\ (udset\ n\ k\ s\ L\ Ts\ q)} \\
&\qquad (\mathtt{PREIMAGE\ X}\ \{0\} \cap \mathtt{p\_space\ p}) = 1) \wedge \\
&(\mathtt{A12} = \mathtt{PREIMAGE\ IT}\ \{\lceil \tfrac{\mathtt{L}}{\mathtt{Ts}} \rceil\} \cap \mathtt{p\_space\ p}) \wedge \\
&(\mathtt{A22} = \mathtt{PREIMAGE\ IT}\ \{\lceil \tfrac{\mathtt{L}}{\mathtt{Ts}} \rceil + 1\} \cap \mathtt{p\_space\ p}) \wedge \\
&(\mathtt{Hic} = \mathtt{IMAGE}\ (\lambda \mathtt{i.}\ \ \mathtt{PREIMAGE\ (UD\_rv\ (SUC\ i))}\ \{1\} \cap \mathtt{p\_space\ p})) \wedge \\
&(\forall \mathtt{x.}\ \ \mathtt{x} \in \mathtt{count\ (SUC\ n)} \Rightarrow (\mathtt{cond\_prob\ p\ (udset\ n\ k\ s\ L\ Ts\ q)} \\
&(\mathtt{A12} \cap (\mathtt{PREIMAGE\ X}\ \{\mathtt{\&x}\} \cap \mathtt{p\_space\ p})) = \mathtt{prob\ p}\ \left(\bigcap_{(\mathtt{i} < \lceil \frac{\mathtt{L}}{\mathtt{Ts}} \rceil)} \mathtt{Hic}\right) \wedge \\
&(\mathtt{cond\_prob\ p\ (udset\ n\ k\ s\ L\ Ts\ q)} \\
&(\mathtt{A22} \cap (\mathtt{PREIMAGE\ X}\ \{\mathtt{\&x}\} \cap \mathtt{p\_space\ p})) = \mathtt{prob\ p}\ \left(\bigcap_{(\mathtt{i} < \lceil \frac{\mathtt{L}}{\mathtt{Ts}} \rceil + 1)} \mathtt{Hic}\right))) \\
&\Rightarrow (\mathtt{prob\ p\ (p\_space\ p\ DIFF\ (udset\ n\ k\ s\ L\ Ts\ q))} = \\
&1 - (1 - \mathtt{s}) \times \left(1 - \frac{(\lceil \frac{\mathtt{L}}{\mathtt{Ts}} \rceil)}{\mathtt{k}} \times \mathtt{q}\right)^{\mathtt{n}} - \mathtt{s} \times \left(1 - \frac{(\lceil \frac{\mathtt{L}}{\mathtt{Ts}} \rceil + 1)}{\mathtt{k}} \times \mathtt{q}\right)^{\mathtt{n}}).
\end{aligned}
$$

where

- `sn_covers_p` is the Binomial random variable (Definition 3.9).
- `intr_distr_rv` is the intrusion random variable (Definition 4.1).
- `sbst_empty_sch_rv` is the higher-order-logic formalization of an empty sub-network in HOL. We modelled such behavior by a Bernoulli random variable with success probability $\left(1 - \frac{1}{k}\right)^c$.

- The assumption `(indep_rv p IT X Borel Borel)` ensures the independence between the two random variables $X$ and $IT$.

- The HOL function `(udset n k s L Ts q)` models the main event of non-detection $UD$, as specified in Equation (4.2). This function depends on various design parameters, i.e., $n$: the number of sensor nodes, $k$: the number of sub-networks, $L$: the intrusion period, $Ts$: the scheduling time slot, and $s$: the remainder of $L$ in terms of $Ts$.

- The assumption `(cond_prob p (udset n k s L Ts q) (PREIMAGE X {0} ∩ p_space p) = 1)` reflects the first case, discussed at the beginning of this subsection.

- The events `A12`, `A22`, and `Hic` are the HOL formalizations of the same events used throughout our mathematical reasoning.

- The last assumption is the probability equality discussed just after Equation (4.9).

- The event `((p_space p) DIFF (udset n k s L Ts q))` formalizes the complement event of $UD$.

The proof of the above theorem is primarily based on the application of the total probability law (Equation (2.7)) which further requires the verification of the corresponding assumptions regarding the partition of the events (Theorem 2.2). Moreover, various conditional probability rules (Equations (2.3), (2.4), (2.5), (2.6) and (2.7)), have been used as well. For that purpose, the proof utilizes the measurability of the different events and the verification of the probability distributions of the events $A_{21}$ and $A_{22}$. A lot of real analysis related to Theorem 3.7, verified in Chapter 3, and

formalizing the Binomial theorem for reals, and to the summation function, has been also required to achieve the proof.

### 4.2.2 Detection Probability for Long Events

According to the second case, discussed at the beginning of this Subsection, we simply verify that the detection probability $Pr(D)$, for events which length is $L \geq (k-1) \times T$, equal to

$$Pr(D) = 1 - (1 - q)^n \tag{4.12}$$

using Theorem 4.2. Such result is very significant since it illustrates the linking between our coverage formalizations, done in [20], and the new results on the detection probability $Pr(D)$. In general, a point in the area is covered if any occurring event at this point can be detected. Such feature is measured through the network coverage intensity $C_n$, which determines how well the monitored area is covered [52]. When an event lasts for a duration ($L \geq$ than $(k-1) \times T$), it means that a full working cycle, lasting $k \times T$, is spent at least one time, and all the sub-networks $\{S_i, 0 \leq i \leq n\}$ have been hence working at least once. The intuition is that such event is surely detected within one of the working subsets, and its detection probability is equal to the coverage measurement of the network, when the whole network is assimilated to one sub-network, i.e, $C_n$ for $(k = 1)$. The above equation formally confirms this intuition, and shows how the behavior of the detection probability $Pr(D)$ for events lasting $(L \geq (k-1) \times T)$ matches the one for network coverage intensity $C_n$ for $(k = 1)$.

## 4.3 Formalization of the Average Detection Delay

Within a wireless sensor network, the average detection delay is generally defined as the expectation of the time elapsed from the occurrence of an intrusion event to the time when this event is detected by some sensor nodes [89, 52]. In this part, we target the formal verification of this average detection delay, denoted by $E(D)$. Mathematically, $E(D)$ is specified as the expectation of the random variable $D$ describing the detection delay. We suppose that $E(D)$ is finite.

Let $DT_i$ the average time that the intrusion is detected in the $i^{th}$ round. For the first round ($i = 1$), the delay is obviously zero ($DT_1 = 0$). Since the subsets of nodes are working by rounds (cf. Fig. 4.1), it is thus intuitive that the delay for detecting an intrusion depends on the detection round $i$. In addition, the $DT_i$ values depend also on the starting time, $t_z$, of the intrusion, i.e., $A_{12}$ and $A_{22}$. Hence, for the second round ($i = 2$), based on Fig. 4.1, we can find that

- If $t_z \in [t_0, t_0 + (1 - s) \times T]$, then ($DT_2 = T - \frac{(1-s)\times T}{2}$).
- If $t_z \in ]t_0 + (1 - s) \times T, T[$, then ($DT_2 = \frac{s\times T}{2}$).

More generally, according to the original specification [89, 52], if $t_z \in [t_0, t_0 + (1 - s) \times T]$, i.e., given $A_{12}$, then:

$$DT_i \mid A_{12} = \begin{cases} 0 & \text{if } i = 1 \\ \left((i-1) - \frac{(1-s)}{2}\right) \times T & \text{if } 1 < i \leq \left\lceil \frac{L}{T} \right\rceil \end{cases} \tag{4.13}$$

However, when $t_z \in ]t_0 + (1 - s) \times T, T[$, we have

$$DT_i \mid A_{22} = \begin{cases} 0 & \text{if } i = 1 \\ \left((i-2) + \frac{s}{2}\right) \times T & \text{if } 1 < i \leq \left\lceil \frac{L}{T} \right\rceil + 1 \end{cases} \tag{4.14}$$

Note that the notations $(DT_i \mid A_{12})$ and $(DT_i \mid A_{22})$ refer to the values taken by the random variable $D$ given $A_{12}$ and $A_{22}$, respectively.

Based on Equations (4.13) and (4.14), we notice how the detection delay values depend on the detection round $i$. Consider the random variable $DR_i$ that describes the detection round. Conditioning on the events $A_{12}$ and $A_{22}$, the values of $DR_i$ are

$$DR_i \mid A_{12} = \{i+1 | 0 \leq i \leq ph1 - 1\} \text{ where } ph1 = min(k, \left\lceil \frac{L}{T} \right\rceil) \tag{4.15}$$

$$DR_i \mid A_{22} = \{i+1, 0 \leq i \leq ph2 - 1\} \text{ where } ph2 = min(k, \left\lceil \frac{L}{T} \right\rceil + 1) \tag{4.16}$$

The minimum values for the variables $ph1$ and $ph2$ are considered since we have at most $k$ detection rounds (cf. Fig. 4.1). As an example, consider a WSN which is randomly scheduled into $(k = 3)$ sub-networks, and two intrusion events $E1$ and $E2$ whose starting time $t_z$ is in $[t_0, t_0 + (1 - s) \times T]$, and lasting $(L1 = 1.8 \times T)$ and $(L2 = 3.2 \times T)$, respectively. In the case of event $E1$, $\lceil \frac{L1}{T} \rceil = 2$, and the possible rounds of detection would be $i = \{1, 2\}$. For event $E2$, $\lceil \frac{L2}{T} \rceil = 4$, but the potential detection rounds are $i = \{1, 2, 3\}$, i.e., at most 3 which is equal to $k$.

According to the two above equations, we formally define a general HOL function that describes the detection round random variable in Definition 4.2.

**Definition 4.2.**

```
⊢ ∀DR p ph.  delay_rnd_rv DR p ph =
              (real_random_variable DR p) ∧
              (IMAGE DR (p_space p) = IMAGE (λj.  &SUC j) (count ph)).
```

The main expected detection delay $E(D)$ has been formalized in HOL using the function `delay_wsn`, which is specified as follows

**Definition 4.3.**

```
⊢ ∀p D n k q.  delay_wsn p D n k q = expectation p D.
```

where $p$ is the probability space, $D$ is a random variable, $n$ is the number of deployed nodes, $k$ is the number of disjoint subsets, and $q$ is the probability that each sensor covers a given point. The expected detection delay $E(D)$ can be mathematically written, using the total expectation law (Equation (2.13)) and Equation (4.3), as

$$\begin{aligned} E(D) &= \sum_{j=1}^{n} E(D \mid c = j) \times Pr(c = j) \\ &= \sum_{j=1}^{n} E(D \mid c = j) \times C_n^j \times \left(\frac{r}{a}\right)^j \times \left(1 - \left(\frac{r}{a}\right)\right)^{n-j} \end{aligned} \tag{4.17}$$

where $E(D \mid c = j)$ is the conditional expectation of the real random variable $D$ with respect to the event $(c = j)$. Notice that the case $(c = 0)$ is not considered in Equation (4.17). Indeed, if there is no covering node, then an intrusion can never be detected, and the delay $E(D)$ will be infinite which is not desirable.

In higher-order logic, we model the detection delay behavior, in Definition 4.4, as a real random variable with a finite image on the space $\Omega$.

**Definition 4.4.**

```
⊢ ∀D p.  delay_rv D p = (real_random_variable D p) ∧
                        FINITE (IMAGE D (p_space p)).
```

In the following, we focus on the formal verification of the term $E(D \mid c = j)$ in Equation (4.17) for occurring events of any length $L$. Based on the definition

of conditional expectation (Equation (2.10)), $E(D \mid c = j)$ can be mathematically expressed as

$$E(D \mid c = j) = \sum_d (D = d) \times Pr(D = d \mid c = j) \tag{4.18}$$

Applying the total probability law (Equation (2.7)) on the partition $\{A_{12}, A_{22}\}$, and given the independence of the random variable $IT$ and $c$ (Equation (2.3)), we can establish, using Equation (2.6), that

$$\begin{aligned} E(D \mid c = j) &= (1 - s) \times \sum_d (D = d) \times Pr(D = d \mid A_{12} \cap (c = j)) + \\ & \quad s \times \sum_d (D = d) \times Pr(D = d \mid A_{22} \cap (c = j)) \end{aligned} \tag{4.19}$$

The RHS of Equation (4.19) can be now rewritten, using the reverse definition of conditional expectation for two events (Equation (2.11)), as

$$(1 - s) \times E(D \mid A_{12}, (c = j)) + s \times E(D \mid A_{22}, (c = j)) \tag{4.20}$$

Based on the above equation, we can clearly distinguish two distinct conditional expectations given the events $A_{12}$ and $A_{22}$. According to the analysis done at the beginning of this subsection, these conditional expectations can be established as

$$E(D \mid A_{12}, (c = j)) = E(DC1 \mid c = j) \tag{4.21}$$

$$E(D \mid A_{22}, (c = j)) = E(DC2 \mid c = j) \tag{4.22}$$

where $DC1$ and $DC2$ are the random variables describing the detection delay when ($A_{12} = L$ overlaps $\lceil \frac{L}{T} \rceil$ cycles) and ($A_{22} = L$ overlaps ($\lceil \frac{L}{T} \rceil + 1$) cycles), respectively. More specifically, based on Equations (4.13) and (4.14), $DC1$ and $DC2$ can be written as

$$DC1 = (\lambda x. \left( x - \frac{3}{2} + \frac{s}{2} \right) \times T) \circ DR1 \tag{4.23}$$

$$DC2 = (\lambda x. \left( x - 2 + \frac{s}{2} \right) \times T) \circ DR2 \tag{4.24}$$

where the $\circ$ operator denotes the function composition, and $DR1$ and $DR2$ are the delay round random variables given $A_{12}$ and $A_{22}$, respectively, as described in Equations (4.15) and (4.16).

Plugging the above two equations, into Equations (4.21) and (4.22), and applying the conditional expectation of a function of a random variable (Equation (2.12)), we derive, from Equation (4.20), that the conditional expectation of $D$ given $(c = j)$, $E(D \mid c = j)$, equals

$$\begin{aligned} &(1 - s) \times \sum_{i=2}^{ph1} (i - \frac{3}{2} + \frac{s}{2}) \times T \times Pr(DR1 = i \mid A_{12} \cap (c = j)) + \\ &s \times \sum_{i=2}^{ph2} (i - 2 + \frac{s}{2}) \times T \times Pr(DR2 = i \mid A_{22} \cap (c = j)) \end{aligned} \tag{4.25}$$

Now, analyzing the relationship between the random variables, we can establish that $DR1$ and $IT$ are conditionally independent given the random variable $c$. Indeed, in terms of events, the information $A_{12}$ does not add anything about $(DR1 = i)$ if we already know that $(c = j)$. Similarly for $(DR2 = i)$ and $A_{22}$ given $(c = j)$. Using

Equation (2.8), we can simplify Equation (4.25) into

$$\begin{aligned} E(D \mid c = j) &= (1-s) \times \sum_{i=2}^{ph1} (i - \frac{3}{2} + \frac{s}{2}) \times T \times Pr(DR1 = i \mid c = j) + \\ & \quad s \times \sum_{i=2}^{ph2} (i - 2 + \frac{s}{2}) \times T \times Pr(DR2 = i \mid c = j) \end{aligned} \tag{4.26}$$

Developing the terms $Pr(DR1 = i \mid c = j)$ and $Pr(DR2 = i \mid c = j)$, in the above equation, according to the definition of conditional probability (Equation (2.2)) along with Equation (2.7), we get the following result.

$$\begin{aligned} E(D \mid c = j) &= (1-s) \times \sum_{i=2}^{ph1} \frac{(i - \frac{3}{2} + \frac{s}{2}) \times T \times Pr((DR1 = i) \cap (c = j))}{\sum_{i=1}^{ph1} Pr((DR1 = i) \cap (c = j))} \\ &+ \; s \times \sum_{i=2}^{ph2} \frac{(i - \frac{3}{2} + \frac{s}{2}) \times T \times Pr((DR2 = i) \cap (c = j))}{\sum_{i=1}^{ph2} Pr((DR2 = i) \cap (c = j))} \end{aligned} \tag{4.27}$$

We formally verify, in Theorem 4.3, the HOL theorem formalizing Equation (4.27).

**Theorem 4.3.**

```
⊢ ∀p X D n q IT s L Ts DC1 DC2 DR1 DR2 ph1 ph2.
  (prob_space p) ∧ (events p = POW (p_space p)) ∧
  (delay_rv D p) ∧ (intr_distr_rv IT p s L Ts) ∧
  (1 < k) ∧ (0 < q < 1) ∧ (0 < L) ∧ (0 < Ts) ∧
  (1 < ph1) ∧ (1 < ph2) ∧ (0 < s < 1) ∧
  (sn_covers_p X p q n) ∧ (indep_rv p IT X Borel Borel) ∧
  (delay_rnd_rv DR1 p ph1) ∧ (delay_DC_rv DC1 DR1 p 3/2 s Ts) ∧
```

$$
\begin{aligned}
&\texttt{(delay\_rnd\_rv DR2 p ph2)} \wedge \texttt{(delay\_DC\_rv DC2 DR2 p 2 s Ts)} \wedge \\
&\texttt{(cond\_indep\_rv p DR1 IT X Borel Borel Borel)} \wedge \\
&\texttt{(cond\_indep\_rv p DR2 IT X Borel Borel Borel)} \wedge \\
&(\forall \texttt{i.}\ \ (1 \leq \texttt{i}) \wedge (\texttt{i} < \texttt{SUC n}) \Rightarrow \\
&((\texttt{cond\_expec\_2 D IT X} \left\lceil \tfrac{\texttt{L}}{\texttt{T}} \right\rceil \texttt{(\&i) p Dsx} = \\
&\texttt{cond\_expec\_2 DC1 IT X} \left\lceil \tfrac{\texttt{L}}{\texttt{T}} \right\rceil \texttt{(\&i) p DC1sx)} \wedge \\
&(\texttt{cond\_expec\_2 D IT X} \left(\left\lceil \tfrac{\texttt{L}}{\texttt{T}} \right\rceil + 1\right) \texttt{(\&i) p Dsx} = \\
&\texttt{cond\_expec\_2 DC2 IT X} \left(\left\lceil \tfrac{\texttt{L}}{\texttt{T}} \right\rceil + 1\right) \texttt{(\&i) p DC2sx)))} \\
&\Rightarrow (\forall \texttt{i.}\ \ (1 \leq \texttt{j}) \wedge (\texttt{j} < \texttt{SUC n}) \Rightarrow \\
&\texttt{E}(\texttt{D} \mid \texttt{c} = \texttt{j}) = (1 - \texttt{s}) \times \textstyle\sum_{\texttt{i}=2}^{\texttt{ph1}} \frac{(\texttt{i} - \frac{3}{2} + \frac{\texttt{s}}{2}) \times \texttt{Ts} \times \texttt{Pr}((\texttt{DR1}=\texttt{i}) \cap (\texttt{c}=\texttt{j}))}{\sum_{\texttt{i}=1}^{\texttt{ph1}} \texttt{Pr}((\texttt{DR1}=\texttt{i}) \cap (\texttt{c}=\texttt{j}))} + \\
&\qquad \texttt{s} \times \textstyle\sum_{\texttt{i}=2}^{\texttt{ph2}} \frac{(\texttt{i} - 2 + \frac{\texttt{s}}{2}) \times \texttt{Ts} \times \texttt{Pr}((\texttt{DR2}=\texttt{i}) \cap (\texttt{c}=\texttt{j}))}{\sum_{\texttt{i}=1}^{\texttt{ph2}} \texttt{Pr}((\texttt{DR2}=\texttt{i}) \cap (\texttt{c}=\texttt{j}))}.)
\end{aligned}
$$

where

- The assumptions `(cond_indep_rv p DR1 IT X Borel Borel Borel)` and `(cond_indep_rv p DR2 IT X Borel Borel Borel)` ensure the conditional independence between the different random variables.

- The variables `DC1` and `DC2`, as described in Equations (4.23) and (4.24), are characterized through the HOL function `(delay_DC_rv DC DR p a s Ts)` which is defined as follows

  $$
  \begin{aligned}
  \vdash\ &\forall \texttt{DC DR p a s Ts.}\ \ \texttt{delay\_DC\_rv DC DR p a s Ts} = \\
  &\qquad (\forall \texttt{x.}\ \ \texttt{x} \in \texttt{(p\_space p)} \Rightarrow (0 \leq \texttt{DC x})) \wedge \\
  &\qquad (\texttt{DC} = ((\lambda \texttt{x.}\ \ (\texttt{x - a} + \tfrac{(\texttt{Normal s})}{2}) \times \texttt{(Normal Ts)})) \circ \texttt{DR}).
  \end{aligned}
  $$

- The variable `Dsx = (IMAGE D (p_space p), POW (IMAGE D (p_space p)))`,

and the same equality applies to `DC1sx` and `DC2sx` for the corresponding variables `DC1` and `DC2`, respectively.

The proof of Theorem 4.3 is quite similar to the proof of Equation (4.27) from Equation (4.18). In particular, the reasoning was primarily based on the specification of the above function `(delay_DC_rv DC DR p a s Ts)` by considering only positive values, given that it describes the detection delay behavior which can never be negative. In this case, the terms $(i - \frac{3}{2} + \frac{s}{2})$ and $(i - 2 + \frac{s}{2})$ can be shown to be equal 0 for $(i = 1)$, and the correct summation index of the numerator can be hence proved. Moreover, a lot of reasoning associated with the use of summation including the proof of injectivity for some functions, and real analysis, was also required.

In Equation (4.27), the event "$(DR1 = i) \cap (c = j)$" indicates that "the intrusion event is detected in the $i^{th}$ round" and "there are $j$ covering nodes". Indeed, if an event, covered with $j$ nodes, is detected in the $i^{th}$ round, then it means that all the $j$ covering nodes miss the $(i - 1)$ consecutive subsets, and the first covering nodes belong to the subset $i$. Such event is exactly the same as the following event.

$$\begin{aligned} A_{i,j} &= \left( \bigcap_{m=1}^{(i-1)} H_{m,j} \cap \overline{H}_{i,j} \right) \\ &= \left( B_{i-1,j} \cap \overline{H}_{i,j} \right) \end{aligned} \tag{4.28}$$

where

- $H_{m,j}$ and $B_{i-1,j}$ are the same events used in Equation (4.10).
- the set of events $\{B_{i-1,j}, \overline{H}_{i,j}\}$ is mutually independent.

The probability of the above event (Equation (4.28)) has been already formally verified

in [20], and is equal to $\left[\left(\frac{k-i+1}{k}\right)^j - \left(\frac{k-i}{k}\right)^j\right]$.

At the end, we establish that the final average detection delay $E(D)$ (Equation (4.17)) is

$$E(D) = \sum_{j=1}^{n} E(D \mid c = j) \times C_n^j \times \left(\frac{r}{a}\right)^j \times \left(1 - \left(\frac{r}{a}\right)\right)^{n-j} \tag{4.29}$$

where

$$\begin{aligned} E(D \mid c = j) &= (1-s) \times \sum_{i=2}^{ph1} \frac{(i - \frac{3}{2} + \frac{s}{2}) \times T \times \left[\left(\frac{k-i+1}{k}\right)^j - \left(\frac{k-i}{k}\right)^j\right]}{\sum_{i=1}^{ph1} \left(\frac{k-i+1}{k}\right)^j - \left(\frac{k-i}{k}\right)^j} \\ &+ s \times \sum_{i=2}^{ph2} \frac{(i - 2 + \frac{s}{2}) \times T \times \left[\left(\frac{k-i+1}{k}\right)^j - \left(\frac{k-i}{k}\right)^j\right]}{\sum_{i=1}^{ph2} \left(\frac{k-i+1}{k}\right)^j - \left(\frac{k-i}{k}\right)^j)} \end{aligned} \tag{4.30}$$

It is important to note that the final HOL theorem for the verification of the main function of the average detection delay `delay_wsn` (Definition 4.3) has not been presented here but an interested reader can access it from [19].

In this section, we detailed the higher-order-logic formalizations of the detection performances of wireless sensor networks using the $k$-set randomized scheduling. The corresponding HOL code is available at [19]. In the next section, we will demonstrate how the resulting universally quantified theorems greatly facilitate the formal analysis of real-world WSN applications.

## 4.4 Application: Formal Analysis of WSN for Border Surveillance

Wireless sensor networks have been widely explored for border monitoring applications [3]. The main goal of a WSN deployed for border monitoring is to continuously detect intruding elements with a high probability and a small delay. These systems are useful for the detection of forces or vehicles in a military context [37], or the prevention of illegal intrusions of migrants or terrorists along a country border. In this context, the potential harsh nature of the field of interest makes a random deployment by air-dropping sensors much more practical. In this section, we are interested in formally analyzing the detection performances of a wireless sensor network deployed for a border monitoring application [92, 76].

Due to the safety-critical feature of the target application, the deployed WSN has to remain alive as long as possible while ensuring an efficient detection. Nevertheless, as stated in [3], most of the existing WSN for border monitoring suffer from lifetime limitations, e.g., a REMBASS sensor node, once deployed, can be functional for 30 days only [37]. In case of using the WSN to monitor terrorist intrusions along a mountainous border, it is obviously not required to monitor the whole area at all times. Thus, we can use the $k$-set randomized scheduling algorithm to preserve energy in a given border monitoring application [92]. In the specified application, the nodes have a sensing area $r = 30$, and are deployed into an area of size $a = 10000m^2$, whereas, the success probability $q$ of a sensor covering a point, is $q = \frac{r}{a} = 0.28$.

In the previous section, we analyzed the detection probability $Pr(D)$ according to the intrusion length $L$ by distinguishing 2 cases: $\{L < (k-1) \times Ts)\}$ and $\{L \geq (k-1) \times Ts)\}$. It is important to note that, in the current application analysis, we

focus on the first case; $\{L < (k-1) \times Ts)\}$, which reflects transient events, that may not be detected, and is thus the most pertinent part of this analysis. For the other case, i.e., $\{L \geq (k-1) \times Ts)\}$, we have already discussed that the detection probability $Pr(D)$ equals the network coverage, and its asymptotic behavior has been investigated in [21].

Based on our theoretical development done in the previous section, we now conduct a formal asymptotic analysis of the probabilistic detection and delay based on the parameters $n$ and $k$. For that, we are going to tackle the generic case and then instantiate it for the given border monitoring application. Hence, we simply denote `(prob p (p_space p DIFF (udset n k s L Ts q)))` by `(Pd_wsn p n k s L Ts q)` and `(delay_wsn p D n k q)` as `(D_wsn p D n k q)`. In the context of our application, we basically verify two main properties of interest related to the detection probability of the events of interest and the detection delay. Thus, we easily check in HOL that `(prob p (p_space p DIFF (udset n k s L Ts (0.28))))` equals

$$1-(1-s)\times\left(1-\frac{\left(\lceil\frac{L}{Ts}\rceil\right)}{k}\times(0.28)\right)^{n}-s\times\left(1-\frac{\left(\lceil\frac{L}{Ts}\rceil+1\right)}{k}\times(0.28)\right)^{n} \quad (4.31)$$

and, the expected detection delay, `(delay_wsn p D n k (0.28))`, is

$$\sum_{j=1}^{n} E(D \mid c=j)\times C_n^j\times(0.28)^j\times(1-(0.28))^{n-j} \quad (4.32)$$

where $E(D \mid c=j)$ represents the expression specified in Equation (4.30). Next, we simply denote Equation (4.31) and Equation (4.32), by `(Pd_surv p n k s L Ts (0.28))` and `(D_surv p D n k (0.28))`, respectively. It is important to note that, for space constraints, and in all the asymptotic analysis below, we only mention

the main mathematical assumptions related to the used variables in the detection probability and delay. Whereas, the complete HOL code for these asymptotic analysis can be found in [19].

### 4.4.1 Formal Analysis based on the Number of Nodes

We formally verify that the detection probability is an increasing function of $n$, i.e., a larger $n$ value leads to a better detection probability.

**Lemma 4.1.**

```
⊢ ∀p k q s L Ts.  (1 < k) ∧ (0 < s < 1) ∧ (0 < L) ∧ (0 < Ts) ∧
  ∧ (L < &(k-1)×Ts) (0 < q < 1)
  ⇒ (mono_incr (λn.  Pd_wsn p n k s L Ts q))).
```

where `mono_incr` is the HOL definition given in Definition 3.13.

Besides, we formally verify, in Lemma 4.2, that the probability of detecting an intrusion event approaches 1 as the number of deployed nodes becomes very very large.

**Lemma 4.2.**

```
⊢ ∀p k q s L Ts.  (1 < k) ∧ (0 < s < 1) ∧ (0 < L) ∧ (0 < Ts) ∧
  (L < &(k-1)×Ts) ∧ (0 < q < 1)
  ⇒ lim_{n→+∞} (λn.  Pd_wsn p n k s L Ts q) = 1.
```

where `lim` is the HOL formalization of limit for real sequences.

Similarly, it is also very useful to investigate the delay behavior of the randomized scheduling. Thus, we formally verify, in Lemma 4.3, that the detection delay $D\_wsn$ starts to be decreasing versus the number of nodes $n$ from a given range,

denoted $n_0$. Consequently, $D\_wsn$ becomes smaller when a large number of nodes is deployed. In this case, an intrusion is expected to be detected more quickly, since it is likely that many more covering nodes are deployed in the surrounding area.

**Lemma 4.3.**

```
⊢ ∀p k q s L Ts.  (1 < k) ∧ (0 < s < 1) ∧ (0 < L) ∧ (0 < Ts)
  ∧ (0 < q < 1)
  ⇒ (mono_decr_range (λn.  (real (D_wsn p D n k q))))).
```

where the function `real` is used to convert the detection delay of type extended real to its corresponding real value, and the HOL function `mono_decr_range` is specified in Definition 4.5. In addition, looking for the range from which the detection delay starts to be decreasing versus $n$, was somewhere tricky. Given the complexity of the mathematical expressions of the detection delay, the HOL analysis of the lemma 4.3 has required a lot of real reasoning on the convergence of series and the properties of infinite sums.

**Definition 4.5.**

```
⊢ ∀ f.  mono_decr_range f ⇔ (∃n0.  ∀n.  n ≥ n0 ⇒ f (SUC n) ≤ f n).
```

Based on Lemmas 4.1 and 4.2, we establish that any target detection probability $Pd\_wsn$ can be achieved by increasing the number of deployed nodes $n$, for any values of the input variables $k$, $q$, $s$, $L$, and $Ts$. More specifically, these results can be easily verified for the detection probability, $Pd\_surv$, in the context of the given border monitoring application (Lemmas 4.4 and 4.5).

**Lemma 4.4.**

```
⊢ ∀p k s L Ts.  (1 < k) ∧ (0 < s < 1) ∧ (0 < L) ∧ (0 < Ts)
```

```
∧ (L < &(k-1)×Ts)
⇒ (mono_incr (λn.  Pd_surv p n k s L Ts (0.28)))).
```

**Lemma 4.5.**

```
⊢ ∀p k s L Ts.  (1 < k) ∧ (0 < s < 1) ∧ (0 < L) ∧ (0 < Ts)
  ∧ (L < &(k-1)×Ts) ⇒ lim_{n→+∞} (λn.  Pd_surv p n k s L Ts (0.28)) = 1.
```

In addition, we reconfirm the result of Lemma 4.3 using Lemma 4.6, i.e., increasing the number of deployed nodes $n$ gives smaller detection delays and thus a better performance of the deployed application.

**Lemma 4.6.**

```
⊢ ∀p k s L Ts.  (1 < k) ∧ (0 < s < 1) ∧ (0 < L) ∧ (0 < Ts)
  ⇒ (mono_decr_range (λn.  (real (D_surv p D n k (0.28))))).
```

According to Lemmas 4.1 and 4.3, enhancing the detection capacities of the deployed WSN, is possible through the deployment of more nodes. However, random deployment is known to be very costly for most WSN applications. In the context of a WSN using the $k$-set randomized scheduling, it is usually possible to improve the whole detection capacity of the network by simply updating the number of disjoint subsets $k$ by a suitable value.

### 4.4.2 Formal Analysis based on Uniform Partitions

Next, we formally study the asymptotic performance behavior of the $k$-set randomized algorithm when the nodes are *uniformly* partionned.

In particular, we successfully verify, in Lemma 4.7, the upper limit of the detection probability $Pd\_wsn$ when $n = k \times m$ and $k$ goes to infinity.

**Lemma 4.7.**

$$\vdash \forall \mathtt{p\ m\ q\ s\ L\ Ts.}\ \ (0 < \mathtt{s} < 1) \wedge (0 < \mathtt{L}) \wedge (0 < \mathtt{Ts}) \wedge$$
$$(0 < \mathtt{q} < 1) \wedge (\forall \mathtt{k.}\ \ \mathtt{L} < \mathtt{\&k} \times \mathtt{Ts}) \Rightarrow$$
$$\lim_{\mathtt{k} \to +\infty} (\lambda \mathtt{k.}\ \ \mathtt{Pd\_wsn\ p\ (k \times m)\ k\ s\ L\ Ts\ q}) =$$
$$1 - (1 - \mathtt{s}) \times \mathtt{e}^{-\lceil \frac{\mathtt{L}}{\mathtt{Ts}} \rceil} \times \mathtt{q} \times \mathtt{m} - \mathtt{s} \times \mathtt{e}^{-\left(\lceil \frac{\mathtt{L}}{\mathtt{Ts}} \rceil + 1\right)} \times \mathtt{q} \times \mathtt{m}.$$

Similar to the proof of Lemmas 3.11 and 3.14 in Chapter 3, the proof of the above lemma is based on the mathematical result consisting in $\lim_{\mathtt{k} \to +\infty} (1 + \frac{\mathtt{x}}{\mathtt{k}})^{\mathtt{k}} = \mathtt{e}^{\mathtt{x}}$, which we had to prove first in order to correctly achieve this proof.

Based on Lemma 4.7, the analysis of the above limit versus various parameters such as the intrusion period $L$, and the number of nodes per subset $m$, is now feasible. We hence verify that when $m$ is very large, the detection probability will surely approach 1. Such result is considered as a second verification of Lemma 4.2 in the specific case where $n = k \times m$.

**Lemma 4.8.**

$$\vdash \forall \mathtt{p\ q\ s\ L\ Ts.}\ \ (0 < \mathtt{s} < 1) \wedge (0 < \mathtt{L}) \wedge (0 < \mathtt{Ts}) \wedge (0 < \mathtt{q} < 1) \wedge$$
$$(\forall \mathtt{k.}\ \ \mathtt{L} < \mathtt{\&k} \times \mathtt{Ts})$$
$$\Rightarrow \lim_{\mathtt{m} \to +\infty} (\lambda \mathtt{m.}\ \ \lim_{\mathtt{k} \to +\infty} (\lambda \mathtt{k.}\ \ \mathtt{Pd\_wsn\ p\ (k \times m)\ k\ s\ L\ Ts\ q})) = 1.$$

Finally, we show that the above mentioned two results are also valuable for the given application for border surveillance through a simple instantiation of the input parameter $q$ by its value. The corresponding HOL analysis is given in the following 2 lemmas.

**Lemma 4.9.**

$$\vdash \forall \mathtt{p\ m\ s\ L\ Ts.}\ \ (0 < \mathtt{s} < 1) \wedge (0 < \mathtt{L}) \wedge (0 < \mathtt{Ts}) \wedge$$
$$(\forall \mathtt{k.}\ \ \mathtt{L} < \mathtt{\&k} \times \mathtt{Ts})$$

$$\Rightarrow \lim_{\mathtt{k}\to+\infty} (\lambda \mathtt{k}.\ \ \mathtt{Pd\_surv\ p\ (k\times m)\ k\ s\ L\ Ts}\ (0.28)) =$$
$$1-(1-\mathtt{s})\times \mathtt{e}^{-\lceil \frac{\mathtt{L}}{\mathtt{Ts}}\rceil}\times(0.28)\times\mathtt{m}-\mathtt{s}\times\mathtt{e}^{-\left(\lceil \frac{\mathtt{L}}{\mathtt{Ts}}\rceil+1\right)}\times(0.28)\times\mathtt{m}.$$

**Lemma 4.10.**

$$\vdash \forall \mathtt{p\ s\ L\ Ts}.\ \ (0<\mathtt{s}<1)\wedge(0<\mathtt{L})\wedge(0<\mathtt{Ts})\wedge(\forall \mathtt{k}.\ \ \mathtt{L}<\&\mathtt{k}\times\mathtt{Ts})$$
$$\Rightarrow \lim_{\mathtt{m}\to+\infty}(\lambda \mathtt{m}.\ \ \lim_{\mathtt{k}\to+\infty}(\lambda \mathtt{k}.\ \ \mathtt{Pd\_surv\ p\ (k\times m)\ k\ s\ L\ Ts}\ (0.28))) = 1.$$

Thanks to the sound support of the detection attributes developed within the HOL theorem prover, we have been able to provide an accurate analysis of the WSN application for border surveillance using the $k$-set randomized scheduling. Table 4.1 gives a summary of the verified properties. Based on the discussion, presented in Chapter 4 of this thesis, it is clear that other analysis techniques can never have this efficiency. Indeed, previous simulation works are mainly based on pseudo-random modelling. Similarly, compared to probabilistic model checkers, a major novelty provided in this chapter is the ability to perform formal and accurate reasoning about statistical properties of the problem. Hence, it was possible to verify the detection delay as a statistical measure using conditional expectation. Moreover, the generic nature of theorem proving and the high expressibility of higher-order logic, allows us to set up theorems for any values for the number of nodes $n$, the number of disjoint subsets $k$, the success probability $q$, the intrusion period $L$, and the scheduling time slot $T$. Obviously, such generality can never be achieved by simulation and model checking. Finally, because missing a critical assumption can lead to verification failure within the theorem prover, the current approach is distinguishable by its completeness regarding the minimum set of assumptions.

| **Verified Property** | **Formulation** |
|---|---|
| $Pd\_surv$ is an increasing function of $n$ | mono_incr ($Pd\_surv$) |
| $Pd\_surv$ approaches 1 when $n$ is very large | $\lim_{n \to +\infty} Pd\_surv = 1$ |
| Limit of $Pd\_surv$ if uniform partitions ($n = k \times m$) | $1 - (1 - s) \times e^{-\lceil \frac{L}{T} \rceil} \times q \times m$ $-s \times e^{-\left(\lceil \frac{L}{T} \rceil + 1\right)} \times q \times m$ |
| $D\_surv$ a decreasing function of $n$ from a certain range | mono_decr_range ($D\_surv$) |

Table 4.1: Detection Analysis of the Border Surveillance Application

## 4.5 Summary and Discussions

In this chapter, we developed the formalizations of the detection properties of wireless sensor networks using the $k$-set randomized scheduling within the HOL theorem prover. In Section 4.1, we have been able to achieve accurate formalizations of the intrusion period of any occurring event, upon which we have built our formal developments of the detection probability and delay. Besides, the formal performance analysis of the detection behavior of the border surveillance application, presented in Section 4.4, definitely show the usefulness of the theoretical higher-order-logic developments. Furthermore, such verification enables reliable asymptotic reasoning of the deployed WSN. These formalizations allow us to formally verify the detection related characteristics of most WSN using the $k$-set randomized scheduling, and many other general WSN applications since the formalized detection metrics are widely used in this context.

The theoretical development of the detection properties consumed approximately about 260 man hours and 2400 lines of code within the HOL theorem prover. Whereas, the formal analysis of our application took, in total, 1900 lines of HOL code including 1500 lines for the proof of Lemma 4.3. We believe that many challenges are incurred in the current work. Similar to the coverage formalization, done in Chapter3, the first major challenge was to map a probabilistic model of a real WSN

algorithm [89, 52], which is far from a pure mathematical problem, into higher-order logic. The mathematical modelling of real-world systems is commonly very intuitive. The support textbooks [89, 52] hence included many hidden steps with very few attached explanations either when considering the random variables or when applying the probability rules. Nevertheless, to achieve the higher-order-logic formalizations of the detection attributes, we have to reason correctly about all missing probabilistic steps so that we can understand the flow of the theoretical analysis. For example, apart some indications about the random variables, given in Equations (4.13) and (4.14), the probabilistic model implicitly considers all the other random variables without any attached textual explanations. With the number of random variables involved, their formal specification was thus very complex. Furthermore, the mathematical analysis was very abstract regarding many aspects like the use of conditional expectation or the correspondence between the different probability events. Even a mathematical specialist cannot efficiently address these critical issues, since such specialist should have comprehensive notions from the WSN side as well. At this stage, a good background on probability theory and a solid knowledge of the WSN context are both required to effectively understand the probabilistic reasoning.

The second main difficulty is the underlying limitations of the libraries, available in the HOL theorem prover, which were missing many mathematical concepts mandatory for the current formalization. We have thus expanded the HOL probability theory by various aspects related to conditional reasoning, such as the conditional independence (Equation 2.9), the conditional expectation (Definition 2.4), and some of the associated properties like the law of total expectation (Equation 2.13). In additional to that, to show the behavior of the detection delay versus the parameter $n$ in Lemma 4.3, we have to construct a formal-reasoning friendly proof, which is quite

different than the paper-and-pencil model [89] involving thus a considerable amount of additional reasoning support in HOL.

On the other hand, our previous development on coverage, presented in Chapter 3, helped significantly to keep the amount of proof efforts sufficiently acceptable. Fortunately, we have been able to take advantage of some reutilizations. We hence checked how the foundational formalizations of the randomized scheduling, in particular the formalization of an empty sub-network (Definition 3.6), has been commonly useful in reasoning about the detection metrics. In addition, the Binomial theorem for reals, shown in Theorem 3.7, has been readily used to verify Theorem 4.2. Finally, the proofs of Lemmas 4.7 and 4.8 have been based on the mathematical result ($\lim_{\mathtt{k}\to+\infty}(\mathtt{1}+\frac{\mathtt{x}}{\mathtt{k}})^{\mathtt{k}} = \mathtt{e}^{\mathtt{x}}$), already proved for Lemmas 3.11 and 3.14 in the coverage work.

The formalizations achieved in Chapters 3 and 4 lay interesting foundations for our future work on the higher-order-logic formalization of the lifetime properties of WSN using the $k$-set randomized scheduling, which will be described in the next chapter. Similarly, once the formal reasoning support of the lifetime aspect is developed in the HOL theorem prover, the performance of other interesting WSN applications, such as underwater monitoring, can also be formally analyzed.

# Chapter 5

# Lifetime Analysis

In WSN using the $k$-set randomized scheduling, the network lifetime is considered as the most critical performance attribute reflecting energy efficiency. In this chapter, we make use of the formalizations of coverage and detection in HOL to provide the higher-order-logic formalization of the optimal network lifetime. More particularly, we formally analyze the optimal lifetime problem under Quality of Service (QoS) constraints associated to the maximization of the network coverage and the detection probability and the minimization of the detection delay.

## 5.1 Problem Formulation

In the coverage analysis, presented in Chapter 3, we have formally verified the minimum number of nodes that are required to deploy in order to ensure a network coverage intensity $C_n$ of at least $t$, denoted here as $C_{nreq}$, for a given number of subnetworks $k$. Hence, if we suppose that a network coverage intensity of at least $C_{nreq}$ is targeted, then the lower bound on the number of required nodes, i.e., $n_{min}$, has been verified as follows.

$$n \geq \left\lceil \frac{\ln(1 - C_{nreq})}{\ln\left(1 - \frac{q}{k}\right)} \right\rceil. \tag{5.1}$$

where $q$ is the probability that a sensor covers a given point inside the field.

Moreover, we can formally deduce that for a given $n$ and a network coverage intensity of at least $C_{nreq}$, the upper bound on the number of disjoint subsets $k$ is given as follows:

$$k \leq \frac{q}{1 - e^{\frac{\ln(1-C_{nreq})}{n}}}. \tag{5.2}$$

which is equivalent to

$$k \leq \frac{q}{(1 - (1 - C_{nreq})^{\frac{1}{n}})}. \tag{5.3}$$

In both scenarios, a network coverage intensity of at least $C_{nreq}$ is achieved. However the other detection metrics, are not guaranteed. For example, in the first case, deploying $n_{min}$ nodes to ensure a coverage quality of at least $C_{nreq}$, may lead to the worst values for the detection metrics, which is not desired at all.

Since the main goal of the $k$-set randomized scheduling is extending the network lifetime, all the performance metrics should be set so that this lifetime is maximized. In other words, it would be good if we can achieve appropriate values of the network coverage intensity; $C_n$, the detection probability; $P_d$, and the detection delay; $D$, while maximizing the network lifetime; $T_{Nlife}$. These appropriate values constitute Quality of Service (QoS) constraints which are pre-defined values, given by the user. These QoS constraints mainly depend on specific design requirements according to the target WSN application.

In the context of a wireless sensor network using the randomized scheduling to

preserve energy, the network lifetime is "the elapsed time during which the network functions well" [88, 89]. In other words, the network lifetime is the time spent from the point when the network starts to be functional until the network is no longer able to detect intrusions. Mathematically, the network lifetime, denoted by $T_{Nlife}$, has been mathematically specified as follows [88, 89].

$$T_{Nlife} = k \times T_{Slife} \tag{5.4}$$

where $k$ is the number of subsets and $T_{Slife}$ is the average lifetime of a typical sensor, which is a constant value. It is important to note that the network is assumed to be composed of identical sensors.

Consequently, given a number of nodes $n$, we are looking to maximize the network lifetime $T_{Nlife}$ under the conditions of minimizing the delay $D$, maximizing the detection probability $P_d$ and the network coverage intensity $C_n$, respectively. The lifetime maximization problem can be formulated as follows [88, 89].

$$\begin{cases} 1.\ D \leq QoS_{DD} \\ 2.\ P_d \geq QoS_{DP} \\ 3.\ C_n \geq QoS_{C_n} \\ 4.\ n = c. \end{cases} \tag{5.5}$$

where $QoS_{DD}$, $QoS_{DP}$, and $QoS_{C_n}$ are predefined QoS constraints associated to the delay $D$, the detection probability $P_d$, and the network coverage intensity $C_n$, respectively, and $c$ is a constant value.

According to Equation (5.4), maximizing the network lifetime $T_{Nlife}$ is to maximize the number of sub-networks $k$. Nevertheless, based on the detection analysis, we have already verified that the detection delay $D$ is an increasing function of $k$.

A very large $k$ will thus imply a large detection delay $D$, which is not suitable in this context. There is thus an upper bound on the $k$-values so that a good coverage $C_n$ can be ensured with an acceptable delay $D$. This bound on the $k$-values is presented in Equation (5.3). The third assumption of the lifetime problem (Equation (5.5)) can be thus substituted by Equation (5.3). The main issue here is to optimize the lifetime under the given QoS constraints rather than maximizing it. The lifetime maximization problem becomes an optimization problem.

$$\begin{cases} 1.\ D \leq QoS_{DD} \\ 2.\ P_d \geq QoS_{DP} \\ 3.\ 1 \leq k \leq \frac{q}{(1-(1-QoS_{C_n})^{\frac{1}{n}})} \\ 4.\ n = c. \end{cases} \tag{5.6}$$

where $q$ is the probability that a sensor covers a given point inside the field.

## 5.2 Mathematical Analysis of the Optimal Lifetime

Theorem 5.1 presents the conditions under which the optimal lifetime problem, given in Equation (5.6), has an optimal solution [88, 89].

**Theorem 5.1.**

The optimal lifetime problem has an optimal solution if:

1. $D \leq QoS_{DD} < \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)}\,[1-(1-q)^n]$,

2. $P_d \geq 1-(1-q)^c \geq QoS_{DP} > 0$,

3. $1 \leq k \leq \frac{q}{(1-(1-QoS_{C_n})^{\frac{1}{n}})}$,

4. $0 < QoS_{C_n} < 1$,

5. $n = c$.

with $Q = \lceil \frac{L}{T} \rceil$ and $s = \frac{L}{T} + 1 - \lceil \frac{L}{T} \rceil$, where $L$ is the duration of an occurring event and $T$ is the length of a scheduling cycle. The above theorem is equivalent to

$$\begin{aligned} S_a &= \{k \mid D \leq QoS_{DD} < \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)}\left[1-(1-q)^n\right], \\ & P_d \geq 1-(1-q)^c \geq QoS_{DP} > 0, \\ & 1 \leq k \leq \frac{q}{(1-(1-QoS_{C_n})^{\frac{1}{n}})}, 0 < QoS_{C_n} < 1, n = c\} \end{aligned} \tag{5.7}$$

is non-empty and is bounded.

***Proof.*** The proof consists in a mathematical analysis of the optimization problem based on various properties associated to the different performance metrics. This mathematical analysis states that an optimal solution exists, if there exist values of $k$ satisfying the conditions of the problem. Indeed, each condition of the optimization problem (Equation (5.6)) generates a set of $k$-values, which has to be proved to be non-empty and bounded. The term bounded, used here, basically means "bounded above" since the integer set of $k$-values is naturally bounded below. Unfortunately, the reference textbooks [88, 89] provide a very abstract proof deducing directly the main conclusion, i.e., the big set $S_a$ is non-empty and bounded. A larger investigation from the mathematical view as well as the WSN one, has been necessary to be able to understand the whole reasoning. In what follows, the proof steps regarding the main properties "bounded" and "non-empty", are detailed for each of the performance aspect.

- **Detection delay**

According to the optimization problem (Equation 5.6), the detection delay metric generates the following set of $k$-values.

$$S_D = \{k \mid D \leq QoS_{DD} < \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)}\left[1-(1-q)^n\right], n = c\} \tag{5.8}$$

The proof that $S_D$ is non-empty and is bounded requires the following 2 results, i.e.,

- The detection delay $D$ is an increasing function of $k$, that we denote here by $D(k)$.
- The detection delay $D$ tends to a function independent of $k$ when $k$ is large enough and $\lim_{k\to\infty} D = \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)}\left[1-(1-q)^n\right]$.

Based on that, we can deduce that the maximum possible values of $D$ is $\lim_{k\to\infty} D$, i.e., $QoS_{DD} < lim_{k\to\infty} D$.

Given the complexity of the delay expression, it is clear that trying to get the concrete bounds of $k$ to prove that the set $S_D$ is bounded, will not be straightforward. Thus, we try to verify that there should be some mathematical results that have been directly applied to prove that $S_D$ is bounded. Through a deeper mathematical study, we succeed to find out these results, which are explained below.

**Theorem 5.2.**

*If a given sequence $a_n \to a$, then $\forall \varepsilon > 0$, there are only finitely many $n$ for which $\mid a_n - a \mid \geq \varepsilon$.*

***Proof.*** Consider $\varepsilon > 0$, and the set $A_\varepsilon = \{n \in \mathbb{N} : \mid a_n - a \mid \geq \varepsilon\}$. Using the limit definition for the real sequence $a_n$, we have: $\forall \varepsilon > 0$ there exists $N$ such that

$\mid a_n - a \mid < \varepsilon$ whenever $\forall n.\ n \geq N$. The set of $n$ for which $\mid a_n - a \mid \geq \varepsilon$ will be contained in the set $\{1, 2, ..., N\}$, and hence finite.

The second important result states that:

**Theorem 5.3.**

*Every finite set of integer s has an upper bound.*

***Proof.*** The proof is based on induction on the finite set $s$ and some results from the arithmetic theory.

Consider Theorem 5.2 for the sequence $D(k)$, that describes the detection delay, and by taking $\varepsilon = (\lim_{k\to\infty} D) - QoS_{DD} = \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)}\left[1-(1-q)^n\right] - QoS_{DD}$, we can then deduce that the set $S_D$ is finite. Consequently, based on Theorem 5.3, we can get that $S_D$ is bounded.

Now, using the monotonicity of the detection delay sequence on the number of partitions $k$ together with some reasoning on the quality of service constraints, we can conclude that the set $S_D$ is non-empty. Indeed, we know that the detection delay is an increasing function of $k$. According to the lifetime optimization problem (Equation 5.5), we look for minimizing the delay $D(k)$ so that the values of $D(k)$ are below $QoS_{DD}$, but cannot go below $D(1)$. Hence, we have $D(k) \geq D(1)$, which gives $D(1) \leq QoS_{DD}$. This ensures that $1 \in S_D$, and hence non-empty.

- **Detection probability**

The second set $S_{Pd}$ (Equation 5.9) has also to be proved as non-empty and bounded.

$$S_{Pd} = \{k \mid P_d \geq P_{d|k=1} = (1-(1-q)^c) \geq QoS_{DP} > 0, n = c\} \tag{5.9}$$

For that, we require to analyze the behavior of the detection probability $P_d$ regarding the parameter $k$, requiring thus the following 2 results.

- The detection probability $P_d$ is a decreasing function of $k$.

- The detection probability $P_d$ tends to 0 when $k$ is large, i.e., $\lim_{k\to\infty} P_d = 0$.

According to the detection probability expression, shown in Chapter 4, it is clearly quite difficult to get the bounds on the parameter $k$ for the set $S_{Pd}$, using pure mathematical operations. Similar to the detection delay, analyzed above, we can prove that $S_{Pd}$ is finite using Theorem 5.2 with $\varepsilon = QoS_{DP}$, which is $> 0$. Applying Theorem 5.3 together with the latter result, we can establish that the set $S_{Pd}$ is bounded.

Using the same reasoning on $S_D$, we can make sure that the set $S_{Pd}$ is non-empty. Hence, while the detection probability is decreasing with $k$, the lifetime optimization problem (Equation 5.5) tries to find the optimal $k$-values that maximize the detection probability. Consequently, $Pd(k) \geq QoS_{DP}$, but $QoS_{DP}$ cannot go above $Pd(1)$, i.e, $QoS_{DP} \leq Pd(1)$. Finally, we can deduce that $(k = 1) \in S_{Pd}$, which guarantees that the set $S_{Pd}$ is non-empty.

- **Network coverage**

Unlike the detection metrics, the $k$ upper bound for the coverage set, $S_{Cn}$, can be obtained through some mathematical operations. Hence, we can clearly show that:

$$S_{Cn} = \{k \mid 1 \leq k \leq \frac{q}{(1-(1-QoS_{C_n})^{\frac{1}{n}})}, n = c\} \tag{5.10}$$

is bounded and is non-empty.

Finally, we can easily show that the big set

$$
\begin{aligned}
S_a &= \{k \mid D \leq QoS_{DD} < \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)}\left[1-(1-q)^n\right], n = c\} \quad (5.11) \\
&\cap \{k \mid P_d \geq 1-(1-q)^c \geq QoS_{DP} > 0, n = c\} \\
&\cap \{k \mid 1 \leq k \leq \frac{q}{(1-(1-QoS_{C_n})^{\frac{1}{n}})}, 0 < QoS_{C_n} < 1, n = c\}
\end{aligned}
$$

is bounded, using the above reasoning on the three different sets $S_D$, $S_{Pd}$ and $S_{Cn}$. Since the event $(k = 1)$ has been already shown to be in each of the three sets $S_{Cn}$, $S_{Pd}$ and $S_D$, we can easily deduce that this event $(k = 1)$ is also in the intersection of these sets, i.e., $\in S_a$. Hence, the big set $S_a$ is also non-empty.

## 5.3 Formalization of the Optimal Lifetime

In HOL, we have to formally verify Theorem 5.1 which gives the conditions under which the optimal lifetime solution exists. According to the above analysis, we have to demonstrate that the set $S_a$ is non-empty and bounded. Using the higher-order-logic formalizations developed in the previous two chapters, each one of the generated sets is proved to be non-empty and bounded. The main intermediate lemmas underlying these proofs are as follows.

- The detection delay $D$ is an increasing function of $k$.
- $\lim_{k\to\infty} D = \frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)}\left[1-(1-q)^n\right]$
- The detection probability $P_d$ is a decreasing function of $k$.
- $\lim_{k\to\infty} P_d = 0$.
- $1 \leq k \leq \frac{q}{(1-(1-C_{nreq})^{\frac{1}{n}})}$.

Consider that $Cn_{wsn}$, $Pd_{wsn}$ and $D_{wsn}$ represent the network coverage, the detection probability and the detection delay, respectively. Next, we describe our formal verification of each of the required lemmas for the lifetime problem, where the considered assumptions are the commonly used assumptions for the variables: $n$, $q$, $L$, $Ts$, and $s$, which designates the number of deployed nodes, the probability that each sensor covers a specific point, the length of the intrusion period, the scheduling round and the remainder of the intrusion period $L$ in terms of the number of slots $T$ (Equation 4.1), respectively.

- **Detection delay**

First, we show, in Lemma 5.1, that the detection delay of a randomly-scheduled WSN, $D\_wsn$, increases as the value of $k$ increases. In other words, the detection delay $D\_wsn$ becomes very large when the WSN is divided into a quite large number of sub-networks $k$. In this case, the allocated time slot for each subset would be small, so that the active nodes do not have enough time to detect the occurring intrusion.

**Lemma 5.1.**

```
⊢ ∀p q n s L Ts.  (1 ≤ n) ∧ (0 < s < 1) ∧ (0 < L) ∧ (0 < Ts)
  ∧ (0 < q < 1) ⇒ (mono_incr (λk.  real (D_wsn p D n k q))).
```

where the HOL function `mono_incr` is given in Definition 3.13. The proof of the above lemma has been based on computing the derivative of the corresponding real functions and applying the Mean Value Theorem (MVT). Thus, this reasoning involves a large amount of real analysis with very complicated mathematical expressions including summations and using various properties of sequences and series of real numbers. It is important to note that the original proof of the above lemma in [93, 89] was missing

a whole fraction term, which is fortunately positive and thus does not finally affect the validity of the function monotonicity.

Next, the limit of the detection delay $D\_wsn$ regarding the parameter $k$ is shown in the following lemma.

**Lemma 5.2.**

$$\vdash \forall \mathtt{p\ q\ n\ s\ L\ Ts.}\ \ (1 \leq \mathtt{n}) \wedge (0 < \mathtt{s} < 1) \wedge (0 < \mathtt{L}) \wedge (0 < \mathtt{Ts})$$
$$\wedge\ (0 < \mathtt{q} < 1) \Rightarrow (\ \lim_{\mathtt{k}\to\infty} \mathtt{D\_wsn} = \tfrac{(\mathtt{Q}-1+\mathtt{s})(\mathtt{Q}^2-1+\mathtt{s})}{2\mathtt{Q}(\mathtt{Q}+1)}\,[1-(1-\mathtt{q})^{\mathtt{n}}]).$$

We verified Lemma 5.2 based on the Mean Value Theorem (MVT).

- **Detection probability**

Based on the parameter $k$, we perform now an interesting study of the limiting behavior of the detection probability. We formally verify, in Lemma 5.3, that a smaller $k$ value induces a larger detection probability $Pd\_wsn$, i.e., $Pd\_wsn$ decreases while increasing the value of $k$. Increasing $k$ surely saves more energy, but a significant increase in $k$ may induce several sub-networks, which in turns translates to a poor detection probability.

**Lemma 5.3.**

$$\vdash \forall \mathtt{p\ k\ q\ n\ s\ L\ Ts.}\ \ (1 \leq \mathtt{n}) \wedge (0 < \mathtt{s} < 1) \wedge (0 < \mathtt{L}) \wedge (0 < \mathtt{Ts}) \wedge$$
$$(0 < \mathtt{q} < 1) \wedge (\forall \mathtt{k.}\ \ \mathtt{L} < \mathtt{\&k} \times \mathtt{Ts})$$
$$\Rightarrow (\mathtt{mono\_decr}\ (\lambda \mathtt{k.}\ \ \mathtt{Pd\_wsn\ p\ n\ k\ s\ L\ Ts\ q})).$$

where the HOL function `mono_decr` is specified in Definition 3.14. The proof of the above lemma has been based on some real theoretic reasoning. According to the two results, shown in Lemmas 5.1 and 5.3, increasing the number of subsets $k$ leads to poor performances in terms of both detection probability and delay.

Now, we formally confirm, in Lemma 5.4, that given a number of nodes $n$, the detection probability $Pd\_wsn$ goes to 0 when $k$ becomes very large. This result thus gives a lower bound on allowable detection probability.

**Lemma 5.4.**

$$\vdash \forall \mathtt{p\ k\ q\ n\ s\ L\ Ts.}\ \ (1 \leq \mathtt{n}) \wedge (0 < \mathtt{s} < 1) \wedge (0 < \mathtt{L}) \wedge (0 < \mathtt{Ts}) \wedge (0 < \mathtt{q} < 1) \wedge (\forall \mathtt{k.}\ \ \mathtt{L} < \mathtt{\&k} \times \mathtt{Ts})$$
$$\Rightarrow \lim_{\mathtt{k} \to +\infty} (\lambda \mathtt{k.}\ \ \mathtt{Pd\_wsn\ p\ n\ k\ s\ L\ Ts\ q}) = 0.$$

Consequently, the randomized scheduling algorithm appears to perform good detection for networks with acceptable $k$ values, but the above result shows that performance may be definitely degraded if the number of partitions $k$ increases. We reconfirm then that the randomized scheduling has a dynamic property enabling performance adjustments of the deployed WSN application according to the value of $k$.

- **Network coverage**

We formally deduce that for a given $n$ and a network coverage intensity of at least $t$, the upper bound on the number of disjoint subsets $k$ is given as follows.

**Lemma 5.5.**

$$\vdash \forall \mathtt{p\ X\ k\ s\ C\ n\ q.}\ \ (1 \leq \mathtt{n}) \wedge (1 < \mathtt{k}) \wedge (0 < \mathtt{q} < 1) \wedge (0 < \mathtt{t} < 1) \wedge (\mathtt{Normal\ t} \leq (\mathtt{Cn\_wsn\ p\ X\ k\ s\ C\ n\ q}))$$
$$\Rightarrow \mathtt{k} \leq \frac{\mathtt{q}}{1 - \mathtt{e}^{\frac{\ln(1-\mathtt{t})}{(\&\mathtt{n})}}}.$$

This result formally confirms the general intuition about the randomized scheduling approach. Increasing $k$ saves energy, but leads to several sub-networks, which in turns translates to a worse network coverage intensity $Cn\_wsn$. This can decrease

the performance of the whole network, which makes the accuracy in the probabilistic analysis of the value of $k$ very important after the deployment.

The higher-order-logic formalizations of Theorem 5.2 and Theorem 5.3 are as follows.

**Theorem 5.4.**

$\vdash \forall$`U (e:real) (A:real).` $(0 \leq \mathtt{e}) \wedge (\mathtt{U} \rightarrow \mathtt{A})$

$\Rightarrow$ `FINITE` $\{(\mathtt{k} : \mathtt{num}) : \mathtt{e} \leq |\, \mathtt{U}(\mathtt{k}) - \mathtt{A} \,|\}$.

**Theorem 5.5.**

$\vdash \forall$`(s:num->bool).` `FINITE s` $\Rightarrow$ $(\exists \mathtt{m}.\, \mathtt{n}.\, (\mathtt{n} \in \mathtt{s}) \Rightarrow \mathtt{n} < \mathtt{m})$.

Table 5.1 outlines the required properties for the lifetime analysis versus the coverage and detection performances.

| **Verified Property** | **Formulation** |
|---|---|
| $D\_wsn$ is an increasing function of $k$ | mono_incr $(D\_wsn)$ |
| Limit of $D\_wsn$ when $k$ is very large | $\frac{(Q-1+s)(Q^2-1+s)}{2Q(Q+1)}\left[1-\left(1-\frac{r}{a}\right)^n\right]$ |
| $Pd\_wsn$ is an decreasing function of $k$ | mono_decr $(Pd\_wsn)$ |
| $Pd\_wsn$ definitely decreases when $k$ is very large | $\lim_{k\to+\infty} Pd\_wsn = 0$ |
| The upper bound of $k$ when $(Cn\_wsn = t)$ | $\frac{q}{1-e^{\frac{\ln(1-t)}{n}}}$ |

Table 5.1: Verified Properties for the Lifetime Analysis

## 5.4 Summary and Discussions

In this chapter, we have been able to formally analyze, within the HOL theorem prover, the optimal lifetime problem (Equation 5.5) under Quality of Service (QoS) constraints, for wireless sensor networks using the $k$-set randomized scheduling. These

QoS constraints are associated to the key performance metrics, i.e., the network coverage, the detection probability and the detection delay. More particularly, there are two main conditions on the $k$-values, under which the optimal lifetime solution exists for such problem. These conditions require that the big set $S_a$ of $k$-values, shown in Equation (5.7), is non-empty and bounded. For that, we built upon the higher-order-logic foundations developed in the previous two chapters to verify this minimal set of conditions.

The current lifetime analysis, presented in this chapter, primarily illustrates the great interest of the higher-order-logic developments achieved for the other performance metrics. Indeed, the lifetime verification has been possible thanks to the sound and complete formalizations of the network coverage, done in Chapter 3, together with the detection probability and delay, presented in Chapter 4. The successful verification of the lifetime optimization problem thus clearly highlights the main advantages of our theoretical developments of the coverage and detection attributes in terms of precision and coherence. Hence, it would not have been possible to effectively achieve the main lifetime proof if, for example, there was a missing assumption on one of the design parameters in the detection part.

While the main goal of the previous formalizations on coverage and detection was to formally verify the mathematical expressions associated with the probabilistic attributes of interest, the lifetime problem is considered in a completely different way. Indeed, the lifetime definition of a randomly-partitioned wireless network, as specified in the paper-and-pencil probabilistic models [88, 89], is very simple (Definition 5.4) and does not require any investigation from the formalization side. However , it was found to be quite interesting to tackle the formal analysis of the lifetime optimization

problem (Equation 5.7) under quality of service constraints. Clearly, the higher-order-logic formalization process for the network lifetime is quite different from the three other performance metrics where the main idea was to formally analyze the conditions under which the optimal network lifetime exists, rather than verify the lifetime in itself.

Comparably to the other performance aspects, many difficulties have been implied in the lifetime verification. Although the lifetime proof seems simple, there were many hidden steps making the understanding of the main proof quite challenging. Hence, except for the coverage set where the concrete bounds on $k$ were simple to get, the other sets on the delay $D$ and the detection probability $P_d$ have been directly deduced to be non-empty and bounded. These deductions, based on some missing steps, have required significant mathematical investigations. No indication was given about which mathematical result is applied. Nevertheless, it is very common that some details which seem obvious for mathematicians turn out to be very hard to follow from the reader's side.

Secondly, the high degree of interactivity required within a theorem prover in general and in HOL, in particular, was also a huge obstacle for a quick formalization. Hence, tedious mathematical efforts may be needed to prove a basic result or just to correctly handle complicated summations. For instance, the proof of Lemma 5.1 of a half-page in the original textbook [89], has been switched into 12 pages of HOL code. For the same lemma, we discovered that a whole fraction term was missing in the original mathematical analysis [89]. This discrepancy would have a crucial impact on the final result if the term was of opposite sign. On the other hand, it is clear that it would not have been possible to catch this error based on a manual inspection unless the proof is redone step by step. Such interesting finding clearly highlights the main

strength of formal methods guaranteeing accurate and complete results.

Finally, it is very worth to note that the formal developments of lifetime can be quite valuable to analyze any randomly-scheduled WSN like the WSN applications, already done in the previous two chapters, or even a general surveillance framework for WSN [91].

# Chapter 6

# Conclusions and Future Work

## 6.1 Conclusions

This thesis presents a whole methodology (shown in Figure 1.2) for the formal probabilistic analysis of the performances of wireless sensor networks using the $k$-set randomized scheduling, which is a widely used algorithm to preserve energy in this context. Hence, using the measure theoretic formalization of probability theory in the HOL theorem prover, we provided the foundational formalizations of the randomized node scheduling algorithm and verified its key performance attributes which are the network coverage, the detection probability and the detection delay. These are the most important performance metric associated to energy efficiency. We also described the formal analysis of the lifetime maximization problem under Quality of Service (QoS) constraints of coverage and detection. The theoretical formalizations offer us the possibilities to formally handle the performance characteristics of most WSNs using the $k$-set randomized scheduling. In order to illustrate the practical effectiveness of our foundational results, we utilize them to perform the formal probabilistic analysis of various WSN applications for forest fire detection and border security surveillance.

It is true that formal methods have been considered, for a while, from a pure theoretical side so that their application on real world problems turns out to be quite difficult and limited. The top raised challenge of the current work is on how to fit a formal method technique, which is theorem proving, into a practical algorithm of the wireless sensor network context. Compared with the existing approaches such as traditional paper-and-pencil probabilistic modelling, simulation and probabilistic model checking, our theorem-proving based approach allows a generic formal verification of randomly-scheduled wireless sensor networks regardless of the values of the design parameters. Besides, due to the sound support of probability theory available in the HOL theorem prover, our approach enables much more reliable validation of the probabilistic performance attributes of interest including statistical quantities. Finally, unlike most of the previous work that focuses on the validation of the functional aspects of WSNs, our work is distinguishable by addressing the performance aspects.

## 6.2 Future Work

The proposed approach, described in this thesis, can be generalized to tackle the formal analysis of the $k$-set randomized scheduling under other assumptions, and even other variant of the algorithm [48, 6, 42]. Actually, the presented formalizations can be valuable to formally verify the same algorithm with, for example, a modified shape of the intrusion object [90], a Poisson deployment [47], and in a three-dimensional monitoring space [92]. A very interesting extension of the coverage formalization, done in Chapter 3, is the formal analysis of the resilience of the $k$-set randomized algorithm to clock asynchrony. Once the higher-order-logic formalization of the Gaussian random variable is made available, we can formally re-verify the relationship between the coverage result and the clock synchronization aspect. We can also think about the formal

analysis of the complexity of the binary search procedure, proposed in [89], to find the best $k$-values for the optimal lifetime problem, presented in Chapter 5. Based on the paper-and-pencil analysis done in [63], the formalization of the optimal detection probability can be also investigated in the same way of the network lifetime (Chapter 5). As a complement verification to the performance analysis done, an interesting research challenge would be to perform a functional verification of the extra-on rule [52] which ensures the connectivity property within randomly-scheduled WSN.

A fundamental open question in the WSN context consists in establishing the formal analysis of probabilistic problems, requiring Markov chains modelling, within a higher-order-logic theorem prover. Examples of such problems include MAC protocols [24] for WSN. That way, the higher-order-logic formalizations of some common random variables such as Bernoulli or Binomial, readily developed, can be very useful.

# Bibliography


[1] Z. Abrams, A. Goel, and S. Plotkin. Set K-cover Algorithms for Energy Efficient Monitoring in Wireless Sensor Networks. In *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, pages 424–432. ACM, 2004.

[2] G. Agha, J. Meseguer, and K. Sen. PMaude: Rewrite-based Specification Language for Probabilistic Object Systems. *Electronic Notes in Theoretical Computer Science*, 153(2):213–239, 2006.

[3] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora, and M. Miyashita. A Line in the Sand: a Wireless Sensor Network for Target Detection, Classification, and Tracking. *Computer Networks*, 46(5):605–634, 2004.

[4] P. Audebaud and C. Paulin-Mohring. Proofs of Randomized Algorithms in Coq. *Science of Computer Programming*, 74(8):568–589, 2009.

[5] M. Bahrepour, N. Meratnia, and P. J. M. Havinga. Automatic fire detection: A survey from wireless sensor network perspective. *The Atmospheric Sciences*, 01, 2008.

[6] R. Bakhshi, F. Bonnet, W. Fokkink, and B. Haverkort. Formal Analysis Techniques for Gossiping Protocols. *SIGOPS Oper. Syst. Rev.*, 41(5):28–36, 2007.

[7] P. Ballarini and A. Miller. Model Checking Medium Access Control for Sensor Networks. In *Proceedings of the Second International Symposium on Leveraging Applications of Formal Methods, Verification and Validation*, pages 255–262, Washington, DC, USA, 2006. IEEE Computer Society.

[8] G. Behrmann, A. David, K.G. Larsen, J. Håkansson, P. Pettersson, W. Yi, and M. Hendriks. UPPAAL 4.0. In *Proceedings of the 3rd International Conference on the Quantitative Evaluation of SysTems*, pages 125–126. IEEE Computer Society, 2006.

[9] C. Bernardeschi, P. Masci, and H. Pfeifer. Early Prototyping of Wireless Sensor Network Algorithms in PVS. In *Computer Safety, Reliability, and Security, LNCS 5219*, pages 346–359. Springer-Verlag, 2008.

[10] C. Bernardeschi, P. Masci, and H. Pfeifer. Analysis of Wireless Sensor Network Protocols in Dynamic Scenarios. In *Stabilization, Safety, and Security of Distributed Systems, LNCS 5873*, pages 105–119. Springer-Verlag, 2009.

[11] V. I. Bogachev. *Measure Theory.* Springer, 2006.

[12] M. Bozga, S. Graf, and L. Mounier. If-2.0: A Validation Environment for Component-based Real-time Systems. In *Proceedings of the International Conference on Computer Aided Verification*, volume 2404, pages 343–348. Springer-Verlag, 2002.

[13] Chong C. Liu, K. Wu, and V. King. Randomized Coverage-preserving Scheduling Schemes for Wireless Sensor Networks. In *Proceedings of the 4th IFIP-TC6*

*International Conference on Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communication Systems*, pages 956–967. Springer-Verlag, 2005.

[14] A. Cerpa and D. Estrin. ASCENT: Adaptive Self-Configuring sEnsor Networks Topologies. *IEEE Transactions on Mobile Computing*, 3(3):272–285, 2004.

[15] 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*, 8(5):481–494, 2002.

[16] E.M. Clarke, O. Grumberg, and D.A. Peled. *Model Checking*. The MIT Press, Cambridge, USA, 2000.

[17] *Coq*. http://coq.inria.fr/.

[18] D. M. Doolin and N. Sitar. Wireless sensors for wildfire monitoring. In *Proceedings of SPIE symposium on smart structures and materials*, pages 477–484, 2005.

[19] M. Elleuch. *Formalization of the Detection Properties of WSNs in HOL*, 2013. HOL Code Available at: http://hvg.ece.concordia.ca/projects/prob-it/wsn.php.

[20] M. Elleuch, O. Hasan, S. Tahar, and M. Abid. Formal Analysis of a Scheduling Algorithm for Wireless Sensor Networks. In *International Conference on Formal Engineering Methods*, volume 6991 of *Lecture Notes in Computer Science*, pages 388–403. Springer Berlin/Heidelberg, 2011.

[21] M. Elleuch, O. Hasan, S. Tahar, and M. Abid. Formal Probabilistic Analysis of a Wireless Sensor Network for Forest Fire Detection. In *Symbolic Computation in Software Science, EPTCS 122*, pages 1–9. Open Publishing Association, 2013.

[22] M. Elleuch, O. Hasan, S. Tahar, and M. Abid. Towards the Formal Performance Analysis of Wireless Sensor Networks. In *Proceedings of the 22nd International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pages 365–370. IEEE Computer Society, 2013.

[23] M. Elleuch, O. Hasan, S. Tahar, and M. Abid. Formal Probabilistic Analysis of Detection Properties in Wireless Sensor Networks. *Formal Aspects of Computing*, 27(1):79–102, 2015.

[24] M. Elleuch, O. Hasan, S. Tahar, and M. Abid. Formal Analysis of MAC Protocols for WSNs: a Review. Technical report, CES-ENIS, Sfax University, May 2011. [10 Pages].

[25] A. Fehnker, M. Fruth, and A. McIver. Graphical Modelling for Simulation and Formal Analysis of Wireless Network Protocols. In *Methods, Models and Tools for Fault Tolerance, LNCS 5454*, pages 1–24. Springer-Verlag, 2009.

[26] A. Fehnker, L. Van Hoesel, and A. Mader. Modelling and Verification of the LMAC Protocol for Wireless Sensor Networks. In *Proceedings of the 6th international conference on Integrated Formal Methods*, pages 253–272. Springer-Verlag, 2007.

[27] W. Feller. *An Introduction to Probability Theory and Its Applications*, volume 1. John Wiley & Sons, 1968.

[28] M. Fruth. Probabilistic Model Checking of Contention Resolution in the IEEE 802.15.4 Low-Rate Wireless Personal Area Network Protocol. In *Proceedings of the Second International Symposium on Leveraging Applications of Formal*

*Methods, Verification and Validation*, pages 290–297. IEEE Computer Society, 2006.

[29] P. B. Godfrey and D. Ratajczak. Naps: Scalable, Robust Topology Management in Wireless Ad Hoc Networks. In *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, pages 443–451. ACM, 2004.

[30] M.J.C. Gordon and T.F. Melham. *Introduction to HOL: A Theorem Proving Environment for Higher-Order Logic*. Cambridge Univ. Press, Cambridge, UK, 1993.

[31] A. Gupta. Formal Hardware Verification Methods: a Survey. *Form. Methods Syst. Des.*, 1:151–238, 1992.

[32] Y. Hanna, H. Rajan, and W. Zhang. Slede: a Domain-Specific Verification Framework for Sensor Network Security Protocol Implementations. In *Proceedings of Conference on Wireless Network Security*, pages 109–118. ACM, 2008.

[33] C. Hartung, R. Han, C. Seielstad, and S. Holbrook. Firewxnet: A multi-tiered portable wireless system for monitoring weather conditions in wildland fire environments. In *Proceedings of the fourth international conference on mobile systems, applications and services*, pages 28–41. ACM, 2006.

[34] O. Hasan. *Formal Probabilistic Analysis using Theorem Proving*. PhD thesis, Concordia Univ., Montreal, QC, Canada, 2008.

[35] M. Hefeeda and M. Bagheri. Wireless Sensor Networks for Early Detection of Forest Fires. In *Proceedings of IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems*, pages 1–6. IEEE, 2007.

[36] F. Heidarian, J. Schmaltz, and F. Vaandrager. Analysis of a Clock Synchronization Protocol for Wireless Sensor Networks. *Theoretical Computer Sciences*, 413(1):87–105, 2012.

[37] M. Hewish. Reformatting Fighter Tactics. Jane's International Defense Review, 2001.

[38] *HOL-Light*. http://www.cl.cam.ac.uk/˜jrh13/hol-light/.

[39] *The HOL Theorem Prover*. http://hol.sourceforge.net/.

[40] J. Hölzl and A. Heller. Three Chapters of Measure Theory in Isabelle/HOL. In *Interactive Theorem Proving, LNCS 6898*, pages 135–151. Springer-Verlag, 2011.

[41] GJ. Holzmann. The model checker spin. *IEEE Trans. Softw. Eng.*, 23(5):279–295, 1997.

[42] C. Hsin and M. Liu. Network Coverage Using Low Duty-cycled Sensors: Random & Coordinated Sleep Algorithms. In *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, pages 433–442. ACM, 2004.

[43] J. Hurd. *Formal Verification of Probabilistic Algorithms*. PhD thesis, Univ. of Cambridge, Cambridge, UK, 2002.

[44] *Isabelle*. http://isabelle.in.tum.de/.

[45] S. Isik, M. Donmez, C. Tunca, and C. Ersoy. Performance Evaluation of Wireless Sensor Networks in Realistic Wildfire Simulation Scenarios. In *Proceedings of the 16th ACM International Conference on Modeling, Analysis & Simulation of Wireless and Mobile Systems*, pages 109–118. ACM, 2013.

[46] S. Jain and S. Srivastava. A Survey and Classification of Distributed Scheduling Algorithms for Sensor Networks. In *Proceedings of International Conference on Sensor Technologies and Applications*, pages 88–93. IEEE Computer Society, 2007.

[47] J. Jiang, C. Liu, G. Wu, and W. Dou. On Location-free Node Scheduling Scheme for Random Wireless Sensor Networks. In *Proceedings of the Second International Conference on Embedded Software and Systems*, pages 484–493. Springer-Verlag, 2005.

[48] S. Kumar, T. H. Lai, and J. Balogh. On K-coverage in a Mostly Sleeping Sensor Network. In *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking*, pages 144–158. ACM, 2004.

[49] D.R. Lester. Topology in PVS: Continuous Mathematics with Applications. In *Proceedings of the Second Workshop on Automated Formal Methods*, pages 11–20. ACM, 2007.

[50] *Overview of Systems Implementing Mathematics in the Computer.* http://www.cs.ru.nl/˜freek/digimath/index.html.

[51] C. Liu. Randomized Scheduling Algorithm for Wireless Sensor Neworks. in Project Report of Randomized Algorithm, University of Victoria, 2004.

[52] C. Liu, K. Wu, Y. Xiao, and B. Sun. Random Coverage with Guaranteed Connectivity: JointScheduling for Wireless Sensor Networks. *IEEE Transactions on Parallel and Distributed Systems*, 17(6):562–575, 2006.

[53] L. Liu. *Formalization of Discrete-time Markov Chains in HOL.* PhD thesis, Concordia Univ., Montreal, QC, Canada, May 2013.

[54] Y. Luo and J.J.P. Tsai. A Graphical Simulation System for Modeling and Analysis of Sensor Networks. In *Proceedings of the 7th IEEE International Symposium on Multimedia*. IEEE, 2005.

[55] D.J.C. MacKay. Introduction to Monte Carlo Methods. In *Proceedings of NATO Advanced Study Institute on Learning in graphical models*, pages 175–204. Kluwer Academic Publishers, 1998.

[56] Q. Mamun. A Coverage-Based Scheduling Algorithm for WSNs. *International Journal of Wireless Information Networks*, 21(1):48–57, 2014.

[57] K.L. Man, T. Krilaviius, T. Vallee, and HL Leung. TEPAWSN-A Formal Analysis Tool for Wireless Sensor Networks. *International Journal of Research and Reviews in Computer Science (IJRRCS)*, 1:24–26, 2010.

[58] A. K. McIver and A. Fehnker. Formal Techniques for the Analysis of Wireless Networks. In *Proceedings of the Second International Symposium on Leveraging Applications of Formal Methods, Verification and Validation*, pages 263–270. IEEE Computer Society, Washington, DC, USA, 2006.

[59] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M.B. Srivastava. Coverage Problems in Wireless Ad-hoc Sensor Networks. In *Proceedings of the IEEE International Conference on Computer Communications*, pages 1380–1387. IEEE Computer Society, 2001.

[60] T. Mhamdi. *Information-Theoretic Analysis using Theorem Proving*. PhD thesis, Concordia Univ., Montreal, QC, Canada, December 2012.

[61] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge Univ. Press, 1995.

[62] T. Nipkow, M. Wenzel, and L.C. Paulson. *Isabelle/HOL: A Proof Assistant for Higher-order Logic*. Springer-Verlag, 2002.

[63] A. Olteanu, Y. Xiao, K. Wu, and X. Du. Weaving a Proper net to Catch Large Objects in Wireless Sensor Networks. *IEEE Transactions on Wireless Communications*, 9(4):1360–1369, 2010.

[64] P. Ölveczky and S. Thorvaldsen. Formal Modeling and Analysis of the OGDC Wireless Sensor Network Algorithm in Real-time Maude. In *Formal Methods for Open Object-based Distributed Systems, LNCS 4468*, pages 122–140. Springer-Verlag, 2007.

[65] S. Owre, J. Rushby, N. Shankar, and D. Stringer-Calvert. PVS: an experience report. In *Applied Formal Methods, LNCS 1641*, pages 338–345. Springer-Verlag, 1998.

[66] *Republic of Tunisia. Ministry of Agriculture, Hydraulic Resources and Fisheries.* http://www.onagri.nat.tn/.

[67] C. Ratel, N. Halbwachs, and P. Raymond. Programming and Verifying Critical Systems by Means of the Synchronous Data-flow Language LUSTRE. In *Proceedings of the conference on Software for citical systems*, pages 112–119. ACM, 1991.

[68] S. Ren, Q. Li, H. Wang, X. Chen, and X. Zhang. Design and Analysis of Sensing Scheduling Algorithms Under Partial Coverage for Object Detection in Sensor Networks. *IEEE Transactions on Parallel Distributed Systems*, 18(3):334–350, 2007.

[69] *The Real-Time Tool*. http://heim.ifi.uio.no/peterol/RealTimeMaude/.

[70] J. Rutten, M. Kwaiatkowska, G. Normal, and D. Parker. Mathematical Techniques for Analyzing Concurrent and Probabilisitc Systems. *CRM Monograph*, 23, 2004.

[71] L. Samper, F. Maraninchi, L. Mounier, and L. Mandel. GLONEMO: Global and Accurate Formal Models for the Analysis of Ad hoc Sensor Networks. In *Proceedings of the First ACM International Conference on Integrated Internet Ad hoc and Sensor Networks (InterSense'06)*, New York, USA, 2006. ACM.

[72] S. Shakkottai, R. Srikant, and N.B. Shroff. Unreliable Sensor Grids: Coverage, Connectivity and Diameter. *Ad Hoc Networks*, 3(6):702–716, 2005.

[73] S. Slijepcevic and M. Potkonjak. Power Efficient Organization of Wireless Sensor Networks. In *Proceedings of the International Conference on Communications*, pages 472–476. IEEE, 2001.

[74] B. Son, Y. Her, and J. Kim. A design and implementation of forest-fires surveillance system based on wireless sensor networks for south korea mountains. *International Journal of Computer Science and Network Security*, 6(9):124–130, 2006.

[75] J. Sun, Y. Liu, J.S. Dong, and J. Pang. PAT: Towards Flexible Verification under Fairness. In *Proceedings of the 21st International Conference on Computer Aided Verification*, pages 709–714. Springer-Verlag, 2009.

[76] Z. Sun, P. Wang, M.C. Vuran, A.M. Al-Rodhaan, A.M. Al-Dhelaan, and I.F. Akyildiz. BorderSense: Border Patrol through Advanced Wireless Sensor Networks. *Ad Hoc Networks*, 9(3):468–477, 2011.

[77] D. Tian and N.D. Georganas. A Coverage-preserving Node Scheduling Scheme for Large Wireless Sensor Networks. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pages 32–41. ACM, 2002.

[78] D. Tian and N.D. Georganas. Connectivity Maintenance and Coverage Preservation in Wireless Sensor Networks. *Ad Hoc Networks*, 3(6):744–761, 2005.

[79] S. Tilak, N. Abu-Ghazaleh, and W. Heinzelman. A Taxonomy of Wireless Microsensor Network Models. *SIGMOBILE Mob. Comput. Commun. Rev.*, 6:28–36, 2002.

[80] The Matlab tool. http://www.mathworks.com/products/.

[81] S. Tschirner, L. Xuedong, and W. Yi. Model-based Validation of QoS Properties of Biomedical Sensor Networks. In *Proceedings of the 8th ACM International Conference on Embedded Software*, pages 69–78. ACM, 2008.

[82] *Republic of Turkey. Ministry of Environment and Forestry, General Directorate of Forestry.* http://www.ogm.gov.tr/.

[83] L. Wang and Y. Xiao. A Survey of Energy-efficient Scheduling Mechanisms in Sensor Networks. *Mobile Networks and Applications*, 11(5):723–740, 2006.

[84] The PRISM Model Checker Website. http://www.prismmodelchecker.org/.

[85] The PVS Theorem Prover Website. `http://pvs.csl.sri.com/`.

[86] J. Woodcock, P.G. Larsen, J. Bicarregui, and J. Fitzgerald. Formal Methods: Practice and Experience. *ACM Computing Surveys*, 41(4):19:1–19:36, 2009.

[87] K. Wu, Y. Gao, F. Li, and Y. Xiao. Lightweight Deployment-aware Scheduling for Wireless Sensor Networks. *Mobile Networks and Applications*, 10(6):837–852, 2005.

[88] Y. Xiao, H. Chen, K. Wu, C. Liu, and B. Sun. Maximizing Network Lifetime under QoS Constraints in Wireless Sensor Networks. In *Proceeding of the Global Telecommunications Conference (GLOBECOM)*, pages 1–5. IEEE Computer Society, 2006.

[89] Y. Xiao, H. Chen, K. Wu, B. Sun, Y. Zhang, X. Sun, and C. Liu. Coverage and Detection of a Randomized Scheduling Algorithm in Wireless Sensor Networks. *IEEE Transactions on Computers*, 59(4):507–521, 2010.

[90] Y. Xiao, H. Chen, Y. Zhang, X. Du, B. Sun, and K. Wu. Intrusion Objects with Shapes under Randomized Scheduling Algorithm in Sensor Networks. In *Proceedings of the 28th International Distributed Computing Systems Workshops*, pages 315–320. IEEE, 2008.

[91] Y. Xiao and Y. Zhang. Divide-and conquer-based Surveillance Framework using Robots, Sensor Nodes, and RFID tags. *Wireless Communications and Mobile Computing*, 11(7):964–979, 2011.

[92] Y. Xiao, Y. Zhang, M. Peng, H. Chen, X. Du, B. Sun, and K. Wu. Two and Three-dimensional Intrusion Object Detection under Randomized Scheduling Algorithms in Sensor Networks. *Computer Networks*, 53(14):2458–2475, 2009.

[93] Y. Xiao, Y. Zhang, X. Sun, and H. Chen. Asymptotic Coverage and Detection in Randomized Scheduling Algorithm in Wireless Sensor Networks. In *Proceedings of International Conference on Communications*, pages 3541–3545. IEEE, 2007.

[94] J. Yick, B. Mukherjee, and D. Ghosal. Wireless Sensor Network Survey. *Computer Networks*, 52(12):2292–2330, 2008.

[95] L. Yongsheng, G. Yu, C. Guolong, J. Yusheng, and L. Jie. A Novel Accurate Forest Fire Detection System using Wireless Sensor Networks. In *Proceedings of the International Conference on Mobile Ad-hoc and Sensor Networks*, pages 52–59. IEEE Computer Society, 2011.

[96] H. Zayani, K. Barkaoui, and R.Ben Ayed. Probabilistic Verification and Evaluation of Backoff Procedure of the WSN ECo-MAC Protocol. *International Journal of Wireless & Mobile Networks*, 2(2):156–170, 2010.

[97] H. Zhang and J.C. Hou. Maintaining Sensing Coverage and Connectivity in Large Sensor Networks. *Ad Hoc & Sensor Wireless Networks*, 1(1-2), 2005.

[98] M. Zheng, J. Sun, Y. Liu, J.S. Dong, and Y. Gu. Towards a Model Checker for NesC and Wireless Sensor Networks. In *Formal Methods and Software Engineering, LNCS 6991*, pages 372–387. Springer-Verlag, 2011.

# Biography

## Education

- **Ecole Nationale d'Ingénieurs de Sfax (ENIS)**: Sfax, Tunisia

  Ph.D candidate, Lab. "Computer & Embedded Systems", (Sep. 09 - present)

- **Ecole Nationale d'Ingénieurs de Sfax (ENIS)**: Sfax, Tunisia

  M.A.Sc candidate, Lab. "Computer & Embedded Systems", (Sep. 07 - Jul. 08)

- **Ecole Nationale des Sciences de l'Informatique (ENSI)**: Tunis, Tunisia

  B.Sc, Computer Engineering, (Sep. 03 - Jun. 06)

- **Institut Préparatoire IPEIS**: Sfax, Tunisia

  Diploma in Math and Physics Studies, (Sep. 02 - Sep. 03)

- **Institut Préparatoire IPEST**: La Marsa, Tunis, Tunisia

  First year in Math and Physics Studies, (Sep. 01 - Sep. 02)

## Work History

- **École Supérieure des Sciences et de la Technologie de Hamam Sousse (ESSTHS)**: Sousse, Tunisia

Teaching Assistant, Department of Electrical & Computer Engineering (2014-Present)

- **École Nationale d'Ingénieurs de Sfax (ENIS)**: Sfax, Tunisia
  Teaching Assistant (Contractual), Department of Computer Engineering (2008-2011)

# Publications

- **Journal Papers**
  - **Bio-Jr1** M. Elleuch, O. Hasan, S. Tahar, and M. Abid. "Formal Probabilistic Analysis of Detection Properties in Wireless Sensor Networks". Journal Formal Aspects of Computing, 27(1):79–102, January 2015, Springer (2013 IF: 0.609).
- **Refereed Conference Papers**
  - **Bio-Cf1** M. Elleuch, O. Hasan, S. Tahar, and M. Abid. "Towards the Formal Performance Analysis of Wireless Sensor Networks". In *Proceedings of the 22nd International conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'13)*, pages 365–370. IEEE Computer Society, 2013. (Rank B).
  - **Bio-Cf2** M. Elleuch, O. Hasan, S. Tahar, and M. Abid. "Formal Probabilistic Analysis of a Wireless Sensor Network for Forest Fire Detection". In *Proceedings of the 4th International Symposium on Symbolic Computation in Software Science (SCSS'12)*, Electronic Proceedings in Theoretical

Computer Science (EPTCS 122), pages 1–9. Open Publishing Association, 2012. (Best Paper Award).

- **Bio-Cf3** M. Elleuch, O. Hasan, S. Tahar, and M. Abid. "Formal Analysis of a Scheduling Algorithm for Wireless Sensor Networks". In *Proceedings of the 13th International conference Formal Methods and Software Engineering (ICFEM'11)*, Lecture Notes in Computer Sciences (LNCS 6991), pages 388–403. Springer-Verlag, 2011. (Rank B).
- **Bio-Cf4** Y. Aydi, R. Tligue, M. Elleuch, M. Abid, and J-L. Dekeyser. "A Multi Level Functional Verification of Multistage Interconnection Network for MPSOC". In *Proceedings of the 16th IEEE International Conference on Electronics, Circuits, and Systems (ICECS'09)*, pages 439–442. IEEE, 2009.
- **Bio-Cf5** M. Elleuch, Y. Aydi, and M. Abid. "Formal Specification of Delta MINs for MPSOC in the ACL2 Logic". In *Proceedings of the Forum on specification & Design Languages (FDL'08)*, pages 253–254 (Poster). IEEE, 2008. (Rank C).
- **Bio-Cf6** Y. Aydi, M. Elleuch, and M. Abid. "Formal Specification and Verification of a Delta-MIN Based Interconnection Architecture for MPSoC". In *Proceedings of the International workshop Reconfigurable Communication centric Systems-on-Chip (ReCoSoC'08)*, pages 10–17 (CDRom). 2008.

## • Technical Report

- **Bio-Tr1** M. Elleuch, O. Hasan, S. Tahar, and M. Abid. "Formal Analysis of a Scheduling Algorithm for Wireless Sensor Networks". Technical

Report, Department of Electrical and Computer Engineering, Concordia University, February 2011. [27 Pages].

- **Bio-Tr2** M. Elleuch, O. Hasan, S. Tahar, and M. Abid. "Formal Generic Frameworks for WSNs: a Review". Technical Report, CES-ENIS, Sfax University, May 2011. [13 Pages].
- **Bio-Tr3** M. Elleuch, O. Hasan, S. Tahar, and M. Abid. "Formal Analysis of MAC Protocols for WSNs: a Review". Technical Report, CES-ENIS, Sfax University, May 2011. [10 Pages].

République Tunisienne
Ministère de l'Enseignement Supérieur,
de la Recherche Scientifique
et de la Technologie

Université de Sfax
École Nationale d'Ingénieurs de Sfax

Ecole Doctorale
Sciences et Technologies

**Thèse de *DOCTORAT***
***Nom du Doctorat***

**N°d'ordre: 568/2014**

# FORMAL PROBABILISTIC VERIFICATION OF WIRELESS SENSOR NETWORKS

**Maissa ELLEUCH SAHNOUN**

**الخلاصة**: يتنزل هذا العمل ضمن الإطار العام لتطبيق الطرق الرسمية عبر فنية المبرهن النظري لتحليل الخصائص المختلفة الأحتمالية لشبكات الاستشعار اللاسلكية المستخدمة لجدولة العقد العشوائية. وقد تم تطوير هذا العمل ضمن نظرية المبرهن HOL4.

**Résumé** : Ce travail préconise l'utilisation de la logique d'ordre supérieur, à travers la technique de démonstration de théorèmes, pour analyser formellement diverses propriétés probabilistes de performance de Réseaux de Capteurs Sans Fil (RCSF) utilisant l'ordonnancement aléatoire de noeuds. Ce travail a été développé au sein du prouveur de théorèmes HOL4.

**Abstract:** This work advocates the use of higher-order-logic, through theorem proving, to formally analyze various probabilistic performance properties of Wireless Sensor Networks (WSN) using the randomized node scheduling to save energy. This work has been developed within the HOL4 theorem prover.

**المفاتيح:** مظاهرة من النظريات ، شبكات الاستشعار اللاسلكي ، التقسيم العشوائي للعقد ، تقييم الأداء ، مبرهن النظريات HOL4.

**Mots clés**: Démonstration de théorèmes, Réseaux de capteurs sans fil, Partitionnement aléatoire des nœuds, Analyse de performance, Prouveur HOL4.

**Key-words**: Theorem proving, Wireless sensor networks, Randomized node scheduling, Performance analysis, HOL4 theorem prover.