UNIVERSITY OF LIVERPOOL

Communication Models and Protocols for Diffusion Based Networks

Thesis submitted in accordance with the requirements of
the University of Liverpool for the degree of Doctor of Philosophy by

Athraa Juhi Jani

January 2018

Contents

Illustrations

List of Figures

List of Tables

This thesis is primarily my own work. The sources of other materials are identifed.

Dedication

To the memory of my parents.

Abstract

A nanomachine is the basic functional unit in nanotechnology that can perform simple tasks, like sensing and actuation. A set of nanomachines can perform more complex tasks through communicating and sharing information, and by that, they form a nanonetwork. Different communication techniques are proposed for information exchange among nanomachines. Molecular communication is one of these techniques, which is a bio-inspired communication mechanism. The characteristics and rules that govern molecular communication are motivated by communication in biological systems. The main research motivation for writing this dissertation is to emphasise that nanonetworks have different functionalities, environmental rules and objectives, and thus different approaches to system design and problem specifications are needed. The main goal of this dissertation is to explore and analyse molecular communication, through proposing models and defining different scenarios. These models study channel transmission reliability, the distance that transmitted molecules can reach. Beside that, many other aspects have been explored such as the effects of the medium noise, the effects of interference between the current transmitted molecules and previous transmitted molecules, nanomachines energy, consensus problems, and security issues in communication among nanomachines.

Ensuring that the transmitted molecules are received by the receiver nanomachine is an important challenge in molecular communication channel. Thus, PRISM model checker is employed to verify the acknowledgment of receiving molecules in one dimensional channel. Then, PRISM is utilized to check the probability of success/ failure of both transmission/ receiving molecules in a bi-direction and multi-access molecular communication channel.

The propagation of molecules in the communication medium is a significant topic to explore, in order to study the effects of noise, the residual molecules from previous communications and properties of the medium itself, on the sensed molecules, by the receiver nanomachine. Thus, the parameters that can affect the maximum distance that a diffused molecule can reach, and the parameters that have an impact on the pattern of diffusion recognition, are explored through experimentation.

The consensus problem is essential in any distributed system to fulfil an overall agreement or commitment to perform tasks. Thus, reaching consensus among nanomachines in molecular communication is an important topic. A consensus protocol among nanomachines is proposed. The steps of the protocol are implemented twice, after electing a leader nanomachine in the network, then, after proposing an energy harvesting model. Furthermore, reaching a consensus among $n \times n$ nanomachines has been verified using PRISM model checker. Moreover, the consensus problem in diffusion based network with the existence of an adversary nanomachine is presented, as an attempt to explore security issues in molecular communication.

Finally, in a security relevant issue, a protocol is proposed to apply a nanonetwork to jam the communication among bacteria, in order to prevent them from launching an attack.

Acknowledgements

I would like to express my gratitude first and foremost to my supervisors, Professor Dariusz Kowalski and Dr Alexei Lisitsa, for their support, guidance, research ideas and always making time for discussions have made the completion of my PhD possible. I am indeed privileged to have worked with them.

I would also like to express my gratitude to the members of my viva committee, Prof. Leszek Gasieniec and Prof. Tomasz Radzik, for reading my dissertation and providing valuable feedback.

I would also like to extend my profound gratitude to the Iraqi Ministry of Higher Education and Scientific Research for granting me a scholarship to study at University of Liverpool and the Iraqi Culture Attach in London for their kind and constant help.

I would like to thank my academic advisers, Prof. Paul Spirakis and Dr Igor Potapov for their time and constructive feedback and suggestions. The Department of Computer Science at the University of Liverpool has been an excellent place to conduct research; all staff members and colleges have been helpful whenever necessary. I would like to thank my office mates in room 3.08 since October 2013.

The warmest gratitude goes also to my family: sisters; brothers; sisters in law; nephews; and nieces for all their support, encouragement and prayers. A special thanks goes to my sister Bushra who is also studying in the UK for her support. I would also like to thank my extended family and friends in Iraq.

Chapter 1

Introduction

1.1 Introduction

Nanotechnology is a promising research field, which deals with structures, devices, systems and materials creation, by manipulating matter at an atomic and molecular scale, enabling devices development in a scale ranging from one to hundreds of nanometers. Nanonetwork is considered a new research branch, which derived from applying nanotechnology in the digital communication field [8].

The communication between nanoscale devices expands the possible applications, and moreover, it increases the complexity and range of operation of the system [2]. Many options for communications in nanoscale have been revealed and studied, several of which use natural mechanisms and processes as a model, by directly applying different elements from nature to serve their purposes [135]. Molecular communication is a bio-inspired communication mechanism, where information is exchanged through transmitting, propagating and receiving molecules between two nanometer-scale devices [155].

The characteristics and rules that govern molecular communication are motivated by the communication in biological systems. The main goal of this dissertation is to explore and analyse molecular communication. Through proposing models and defining different scenarios to study the channel transmission reliability, the distance that transmitted molecules can reach, the effects of medium noise, the effects of interference between the current transmitted molecules and previous transmitted molecules, nanomachines energy, consensus problem, and security issues in communication among nanomachines.

This chapter discusses the motivation and outline of the dissertation, and is organized in the following: Section 1.2 gives a brief background to nanotechnology and the concept of nanonetworks, including various approaches to develop nanomachines (the functional units in nanonetworks) in subsection 1.2.1; and exploring the potential applications of nanonetworks in subsection 1.2.2. Section 1.3 presents the communication techniques employed in nanonetworks; in subsection 1.3.1 the types of molecular communication techniques are described; and the types of electromagnetic communication techniques are presented in subsection 1.3.2. The main processes of molecular communication based systems are introduced in 1.4. Section 1.5 discusses the state of the problem and research motivations, while in Section 1.6 the research objectives are presented. In Section 1.7 the contribution of the thesis are stated. The outline of the dissertation is demonstrated in Section 1.8. Section 1.9 lists the presented and accepted abstracts.

1.2 Nanonetworks: Background

Nanotechnology can be defined as the science of engineering functional systems at an atomic and molecular scale, the prefix 'nano -' denotes a factor of 10^{-9} and means a billionth [163]. Nanotechnology was first envisioned in December 1959 by the Nobel laureate physicist Richard Feynman at Caltech, where in his lecture entitled "There's plenty of room at the bottom" he talked about the miniaturization of devices down to an atomic level [68]. The research field of nanotechnology is becoming a key area in science based on multidisciplinary collaborations involving medicine, engineering, physics, biology, computer science, and other disciplines, promising new solutions for different applications. A nanomachine is considered to be the basic functional unit in nanotechnology. The rapid evolution in nanotechnology has provided appropriate development in miniaturization and fabrication of nanomachines with simple sensing, computation, data storing, communication and action capability [8]. Further capabilities and applications can be enabled if multiple nanomachines communicate to perform collaborative and synchronous functions in a distributed manner to form a nanonetwork [161].

The Internet of Things concept carries the promise to create a global network, it is a paradigm which can facilitate the communication between everyday objects with one another and with other devices and services over the Internet to achieve different objectives, where these objects are equipped with identifiers, and sensing, processing and networking capabilities [26, 176]. This paradigm has a future vision of being extended by combining more devices, through the Internet of Nano-Things, where nano-scale devices are interconnected with conventional networks and the Internet. However, there are many communication challenges in implementing this idea [11, 88, 176].

1.2.1 Nanomachines Development Approaches

It is obvious that nanomachines are very small devices, with dimensions at or below micrometer range [40]. A manufacturing technique has been presented in [150] of a sensor nanomachine with the ability to detect and count a certain type of molecules. However, the main approaches for nanomachines development are as follows:

1. **Top-Down Approach:** This method is based on Richard Feynman's lecture in 1959. In this approach nanomachines are developed by downscaling current micro-scale devices (microelectronics and micro-electro-mechanical devices). Thus, as an example, a macro-scale machine is first fabricated into an exact copy of itself but with four times smaller scale. Then, after ensuring that it properly works, the

reduced-size machine is used to build a copy of itself with 16 time smaller than the original machine. Gradually, the process of fabricating smaller machines continue until finally producing a nano-scale machine with the ability to carry out simple tasks [161].

2. **Bottom-Up Approach:** In this method nanomachines are developed through investing science and technology at nano-meter scale using molecular compositions. As examples of techniques that use this approach to assemble nanomachines are molecular manufacturing and DNA scaffolding [163].

3. **Bio-hybrid Approach:** The development of nanomachines according to this method is inspired by the natural biological structures of living organisms. Nanomachines can be developed through mimicking existing biological components; as an example, a battery fabricated from adenosine triphosphate which simulates mitochondria conduction [163], or through reusing biological entities like DNA or proteins [161].

1.2.2 Applications of Nanonetworks

The communication between nanomachines enables them to achieve more complex tasks and to extend the dimensions of its potential application in different domains. Among others, here are some of the main areas:

Biomedical Applications: Due to this type of application, many hybrid fields have emerged, such as biomedical nanotechnology, bio-nanotechnology, and nanomedicine. Many examples of nanonetworks' applications in biology that could fundamentally help in disease detection [38]; and there is research [122, 172] that discusses the use of nano-particles in diagnosing brain cancer and oral cancer by utilizing carbon nanotubes and gold nano-particles. Current research is focused on the use of nano-particles to detect cancer cells. The future research aim is to utilize nanomachines rather than nano-particles, due to their computational and operational capabilities, and their ability to communicate with each other [163]. Nanomachines can also be employed to deliver drugs [35, 72] to the desired location inside the human body, and thus help to reduce the side-effects.

Environmental Surveillance: Nanonetworks applications can include environmental areas, such as the control of water pollution and air quality surveillance [163]. For example, nanomachines can be employed to monitor the environment and distinguish whether certain molecules (including radioactive molecules) that can cause environmental problems are present [18].

Food Science Applications: Nanotechnology in general can have an impact on different aspects of the food industry, starting with how food is produced and processed and ending with how it is packaged [163]. As an example, a nano-sensor network can be used in monitoring the quality of food, through sensing and detecting any toxic component which the product might contain [8]. Besides this, nanonetworks have the potential to develop the agricultural sector [14, 133].

Alongside the applications discussed above, nanonetworks can be employed in other fields, in [8] the authors present a promising research direction for extending the domains and applications of nanonetworks.

1.3 Communication Techniques in Nanonetworks

The following section outlines of the main techniques proposed for communication between nanomachines. The functionality range of nanomachines can be limited to their close nano-environment, as a result of their extremely small size [160]. Therefore, a very large number of nanomachines are needed to perform any meaningful task, alongside the requirement for control and coordination of nanomachines functions, involving a challenge for the research of nanomachines communications [112]. As explained earlier, nanonetworks are formed by the interconnection between nanomachines. The means of cooperation and information sharing among nanomachines can be provided by nanonetwork; thus, more complex tasks can be achieved through cooperation and information sharing [8]. A nanonetwork cannot be considered simply as a downscaled version of a conventional network, as communication techniques used in traditional networks, such as acoustic and electromagnetic communication, are unsuitable to operate at nano-scale; because there are difficulties in minimizing the size of the current transceivers alongside the energy constraints; thus, a comprehensive adjustment should be applied to the classical communication methods before employing it to interconnect nanomachines [110, 143]. There are several communication mechanisms proposed for the interconnection between nanomachines [8, 10, 12], which has led to two novel nanonetwork paradigms: molecular communications [20, 21, 23, 25, 56–58, 66, 69, 71, 92, 108, 153], and nano-electromagnetic communication [12, 24, 86, 87, 113, 124, 160].

Nanonetworks
≠
Downscale version of conventional network

FIGURE 1.1: Nanonetworks Require Different Communication Techniques

1.3.1 Molecular Communications

There are different types of molecular communications, inspired by communication among living cells [135]. The fact that a typical cell size can be 10 μm and typical cell mass can be 1 nanogram has led to the investigation and study of communication mechanisms among living cells [8, 71, 74, 80, 165], with the intention of applying these mechanisms to implement nanonetworks in biological scenarios [34, 123]. The information in molecular communication techniques is encoded as molecules concentrations, where the sender nanomachine uses molecules to encode and transmit information, and these molecules propagate through the communication medium to the receiver nanomachine(s) [66, 135, 165].

Biological studies revealed that distance has a strong impact on the molecular communications paradigm [71, 123]; thus, molecular communication techniques can be classified according to the distance between the transmitter nanomachine(s) and the receiver nanomachine(s) as follows [8, 135]:

- The first type is *short range* molecular communication, where the communication range is (from nm to μm). Examples of short range communication in biology are intra-cell and inter-cell communications. Much research has extensively explored this type of molecular communications. The following are the two known techniques:

  - **Calcium signalling** [136, 141]: in this molecular communication technique calcium ions are used to encode information that would be sent from the transmitter nanomachine to the receiver nanomachine(s). Calcium signalling in biology can be used as a communication between adjacent cells (direct access), where the calcium ions propagate from the transmitter cell through its membrane (in cell tissue, the membrane between cells contain a kind of gates called gab junctions, which molecules and ions can pass through) to be forwarded to the adjacent cells [8, 165]. In biological systems, the distance in this communication technique can vary and depends on the size of the cells tissue. Calcium signalling can also be utilized in communication among non-adjacent cells (indirect access), as the transmitted calcium ions propagate through the communication medium from regions of higher concentration to those of lower concentration (following the diffusion process) to reach the receiver cell(s) [163].

    Figure 1.2 shows a simple example of calcium signalling, where there are three cells. In case, one of these cells is triggered by stimulus from the environment (this stimulus can be a chemical signal, the pH value, or an increment in the temperature, etc.). That can cause the release of a chemical called IP_3 inside the cell. This IP_3 chemical enables the cell to emit calcium ions which are stored in an organelle (that represents a depot of the calcium ions).

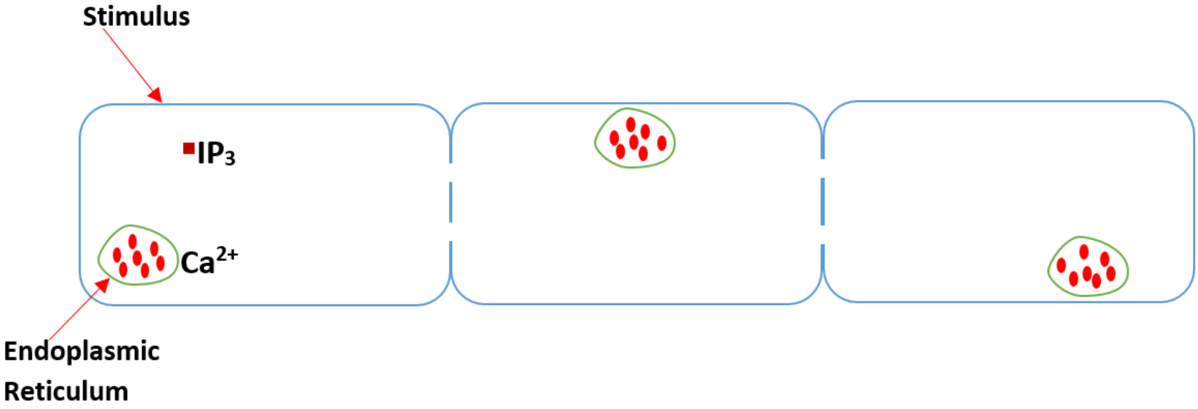


FIGURE 1.2: Simple example of calcium signalling

Then, the concentration of calcium ions increase inside the cell, due to the existence of IP_3 molecule. Thus, the IP_3 molecule moves through the gap junctions between the cells (Figure 1.3). Where, that would trigger the same reaction of releasing calcium ions inside the next cell. Thus, the same process continues to the rest cells of the tissue.

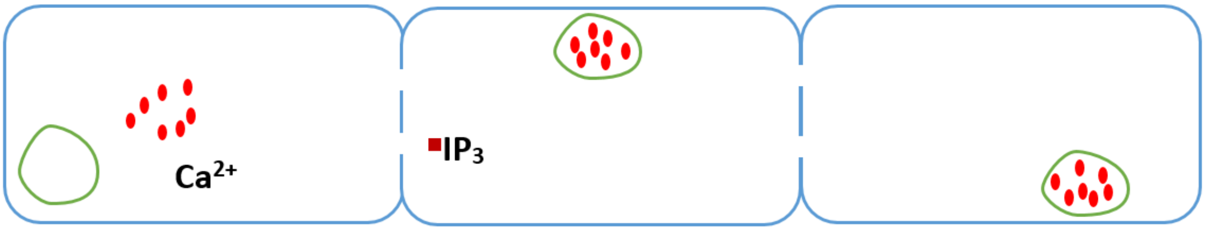


FIGURE 1.3: IP_3 moves to the next cell when Ca^{2+} concentration increases

- **Molecular motors**: in biology there are different types of proteins which can act as molecular motors, such as kinesin, myosin or dynein. Their task is to transport information (molecules) from the transmitter cell to the receiver cell. Molecular motors can move by transforming chemical energy into mechanical energy. They move on predefined rails molecules called microtubules, deployed in a way which lays down a complete railway network for intra-cell transportation, as Figure 1.4 shows. Molecular motors are explored in [79, 129, 132], and molecular communication in general is extensively surveyed in [8].

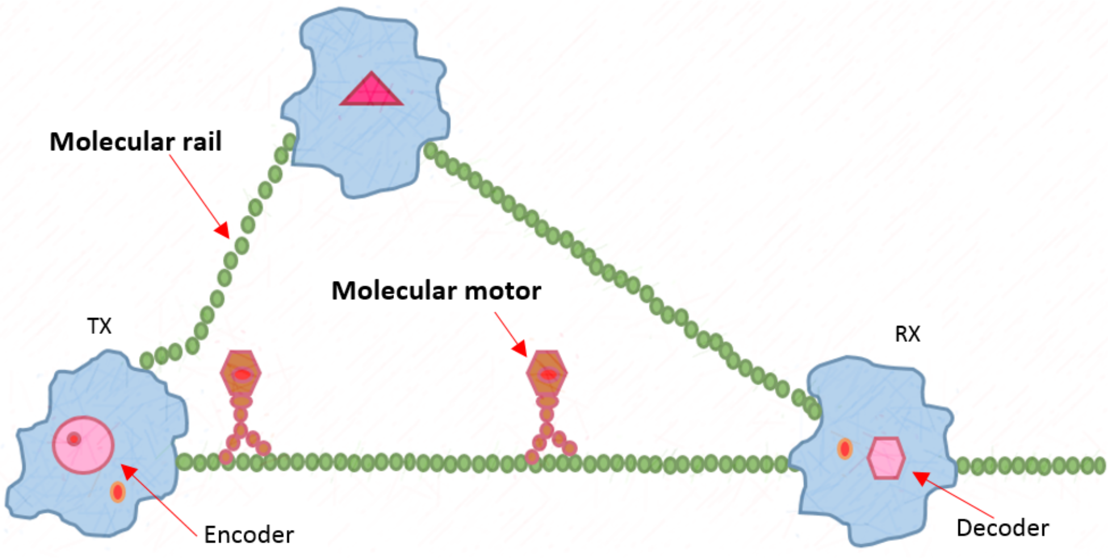


Figure 1.4: General representation of molecular motor communication systems

- The other type is *medium range* molecular communication, where the communication ranges from μm to mm. Medium range molecular communication has been investigated in much research [8, 74–76, 80]. Examples of medium range molecular communications are flagellated bacteria [76] and catalytic nanomotors [75], where information is encoded in DNA sequences and conveyed form the transmitter nanomachine to the receiver nanomachine by utilizing bacteria or nanomotors. This process is called DNA hybridization, where information molecules can be loaded or unloaded from the conveyed molecules. The information molecules at the transmitter are loaded onto the DNAs (which are complementary to the conveyed information). At the receiver, the information molecules are unloaded from the conveyed molecules [163].

- The third type is *long range* molecular communication, where the communication range is from mm to m. Various techniques have been proposed for long range molecular communications, which are inspired by communication techniques of various species, such as pheromone (which insects use for communication), axons, pollen and light transduction [71].

1.3.2 Nano-Electromagnetic Communication

Nanomachines can utilize electromagnetic radiation for communication; however, it is not feasible to scale down the current metallic antennas in order to correspond to the size of nanomachines, due to the expected extremely high resonant frequency [86]. The high frequency can lead to a very high bandwidth, but the transmission range of nanomachines can be almost zero because of the huge channel attenuation [10]. Thus, new electronic nano combinations have been explored and developed to overcome this constraint. As an example is graphene (which is also used to manufacture carbon nanotubes and graphene nanoribbons) [12]. In [87] to perform nano-electromagnetic communications between nanomachines a graphene-based nano antennas have been proposed. As the frequency radiated by an antenna can be obtained from the ratio of the wave propagation speed inside the antenna to the antenna length; thus, radiated electromagnetic waves from a graphene-based nanoantenna size μm, can be in the terahertz band (0.1, 10 THz).

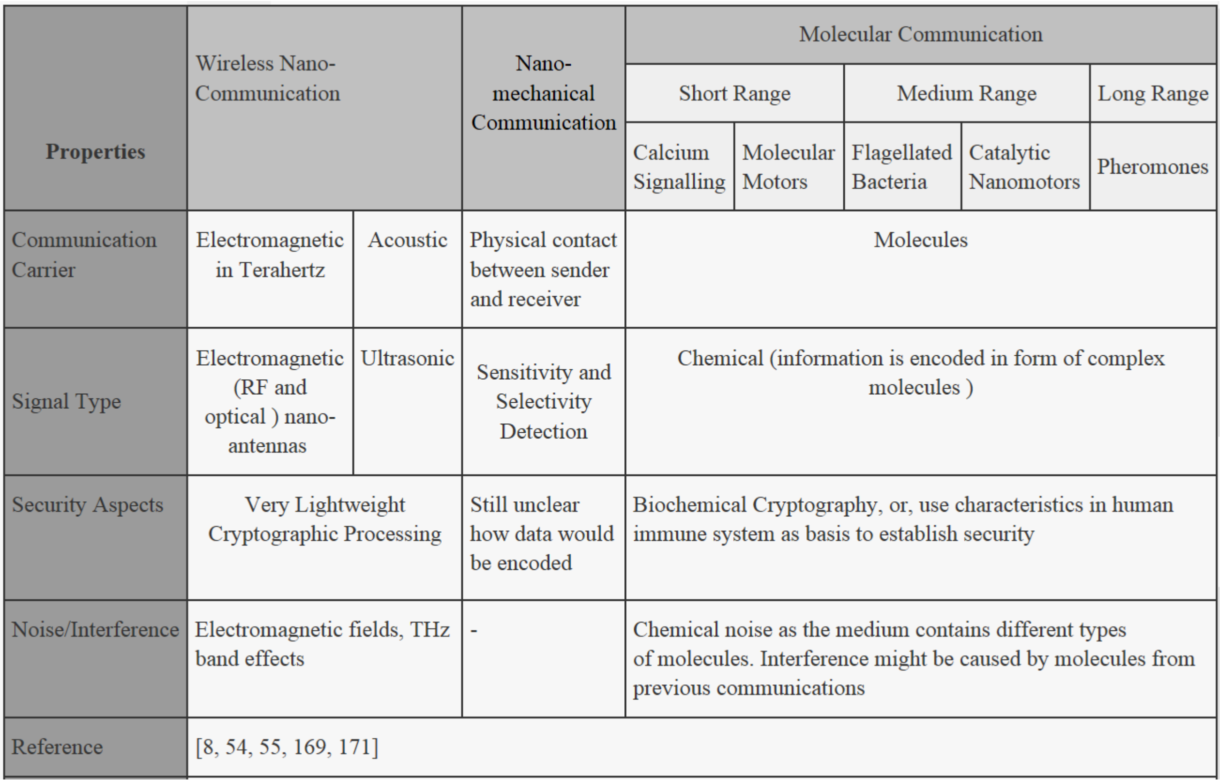
Nanoelectromagnetic communications in wireless nanosensor networks has been explored in [10] and a study of the challenges in modelling a terahertz channel has been presented. Carbon nanotube sensor network has been defined in [22] and the main challenges of such networks have been addressed. The properties of nano-dipole antenna (which is made from carbon nanotube) has been investigated in [87] comparing its resonant frequency and input impedance to those of a nano-patch antenna (which is made of graphene nanoribbons).

1.3.3 Communication Techniques Considered in This Thesis

The communication mechanisms of this thesis proposed models are inspired by molecular communication.

In Chapter 3 the communication technique between nonomachines in the proposed model is based on *calcium signalling* communication [43]. However, the proposed rules of communication are inspired by the *sandpile model* [85, 146].

In Chapters 4, 5, 6, 7 the communication mechanism among nanomachines in the proposed models is based on *diffusion based molecular communication* [163].



Properties	Wireless Nano-Communication		**Nano-mechanical Communication**	Molecular Communication				
				Short Range		Medium Range		Long Range
				Calcium Signalling	Molecular Motors	Flagellated Bacteria	Catalytic Nanomotors	Pheromones
Communication Carrier	Electromagnetic in Terahertz	Acoustic	Physical contact between sender and receiver	Molecules				
Signal Type	Electromagnetic (RF and optical) nano-antennas	Ultrasonic	Sensitivity and Selectivity Detection	Chemical (information is encoded in form of complex molecules)				
Security Aspects	Very Lightweight Cryptographic Processing		Still unclear how data would be encoded	Biochemical Cryptography, or, use characteristics in human immune system as basis to establish security				
Noise/Interference	Electromagnetic fields, THz band effects		-	Chemical noise as the medium contains different types of molecules. Interference might be caused by molecules from previous communications				
Reference	[8, 54, 55, 169, 171]							

Figure 1.5: General overview of nanonetworks communication techniques

The table in figure 1.5 represents an overview of the communication techniques in nanonetworks collected from [8, 54, 55, 169, 171].

1.4 Molecular Communication System Model

Molecular communication based systems consist of three main processes: transmission, propagation and reception (molecules sensing), as shown in figure 1.6.

- **Transmission process**: as explained earlier in molecular communication techniques, that when a nanomachine needs to transmit information, it utilizes molecules to encode information. In order to transmit information molecules, each nanomachine can have a specific buffer for messenger molecules, which can be used for communication [46]. The transmitter nanomachine releases a particular amount of messenger molecules into the communication environment, which represents an information symbol [163].

  In molecular communication, different types of modulation techniques can be used to encode information by the transmitter nanomachine. Three of the main techniques are briefly introduced here:

  - *Concentration Shift Kyeing (CSK)*: In using this modulation, the information is encoded according to the amounts of emitted molecules; here, only one type of molecules is used in communication [17, 18, 23, 25, 98].
  - *Pulse Position Modulation (PPM)*: The information in this modulation is encoded according to the temporal position of the pulse, i.e., emitting molecules in different time rounds; and only one type of molecule is used in this modulation [57, 69, 92].
  - *Molecule Shift Keying (MoSK)*: In this modulation technique, information is encoded by emitting different types of molecules. This modulation is similar to orthogonal modulation in the classical communications [17, 98, 164].

  Both (CSK) and (MoSK) can be affected by the noise and interference resulting from previous transmissions; however, the simulation results in [98] show that the (MoSK) modulation scheme is less affected than (CSK).

- **Propagation process**: the information molecules transmitted in the molecular communication channel are propagated through following random diffusion process (in which molecules proceed from regions of high concentration to regions with low concentration), and the environmental conditions such as temperature can affect the process of propagation [8].

- **Reception process**: in this process, the receiver nanomachine senses the information molecules concentration in the receiver sensing space, and the reception being recognized by the chemical receptors facility. These receptors uniformly deployed within the reception space, each receptor binding to its matching molecules [20, 21, 25, 153]. The amplitude of the received molecules concentration (the signal) can be represented through the number of molecules sensed by the receptors [163].

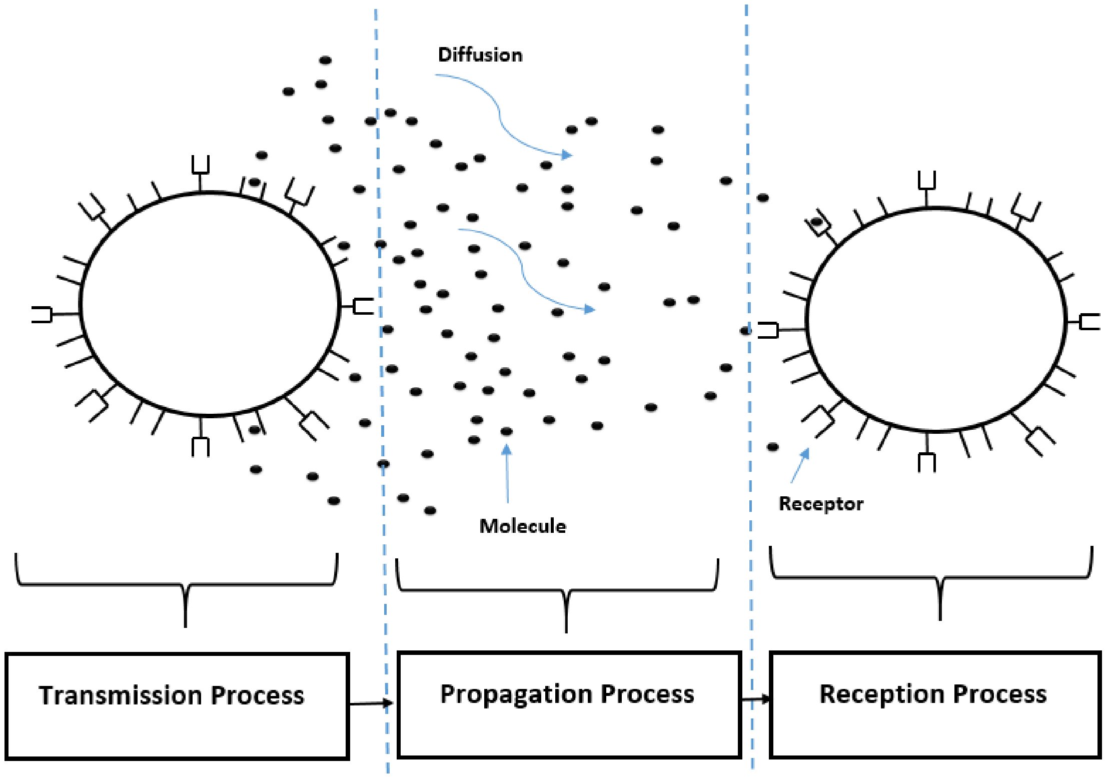


FIGURE 1.6: Outline representing a molecular communication system

1.5 State of the Problem and Research Motivations

Nanonetwork applications as given above, can vary from biomedical (e.g., drug delivery) to industrial (e.g., food and water control) and environmental (e.g., air pollution control) services. It can be noticed from the application domains that nanomachines can be implanted into the environment, food or even the human body. Therefore, any manipulation within the functionality of these nanomachines can have disastrous consequences [55]. Figure 1.7 summarize the first motivation point.

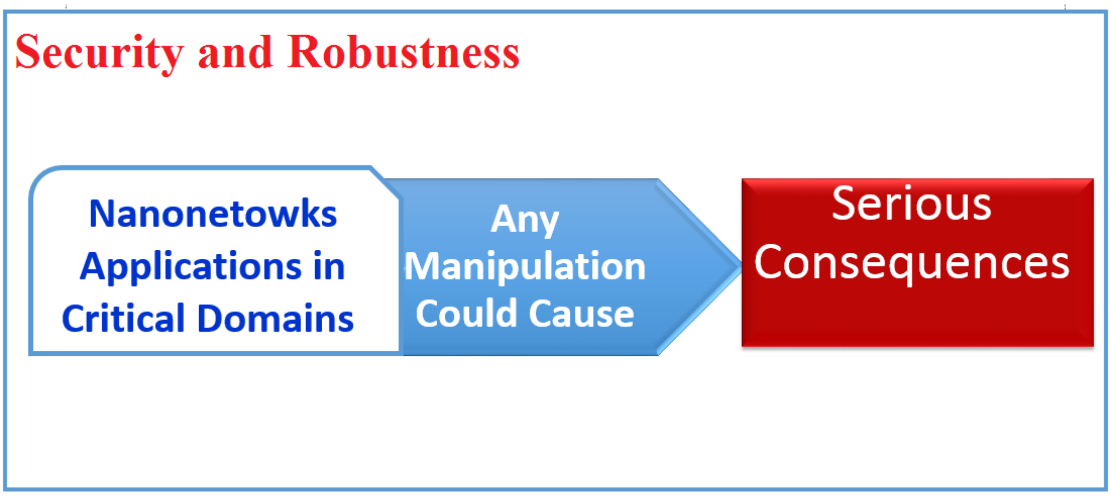

Security and Robustness
Nanonetowks Applications in Critical Domains
Any Manipulation Could Cause
Serious Consequences

Figure 1.7: The importance of nanonetworks robustness

The performance of nanonetwork systems with low resources, security and robustness requirements is one of the main challenges in designing nanonetworks. Specifically, molecular communication can be highly unreliable and suffer from a long delay, which is why communication protocols taking these properties into consideration are required. Most research has adopted the Internet communication protocol and applied it to nanonetworks, but comparing nano-systems with macro-systems it can be found that they have different functionalities, objectives and environmental rules; therefore, a different approach to system design and problem specification, resulting in different algorithms is needed. Figure 1.8 summarize the second motivation point.

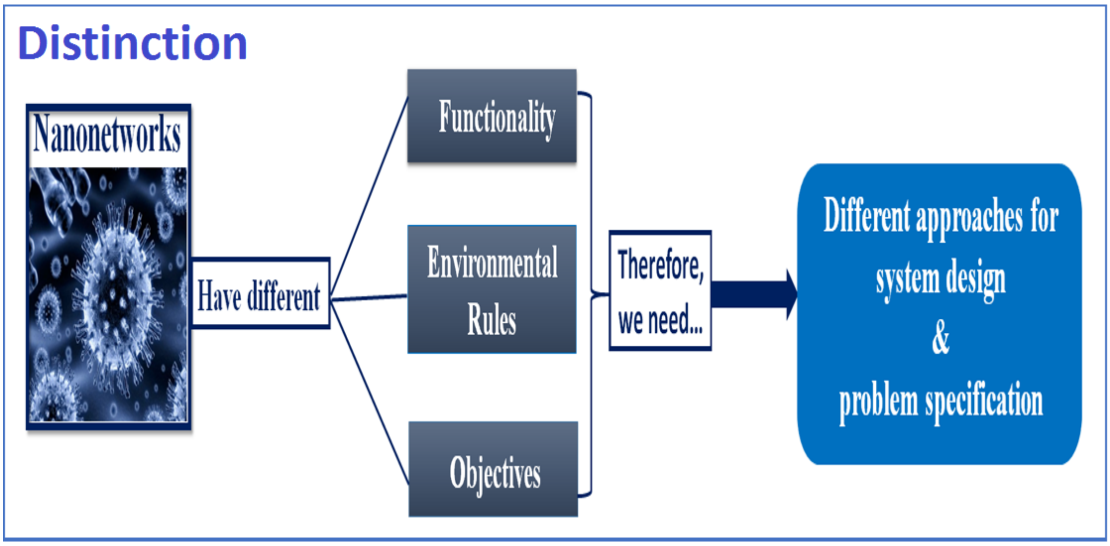


Figure 1.8: Distinction of Nanonetworks

1.6 Research Objectives

Among different types of nanonetwork communication techniques, this dissertation concentrates on studying diffusion-based molecular communication, in which information molecules propagate from a transmitter nanomachine to the receiver nanomachine following a random diffusion process in the environment (mostly in fluid environment). In related literature, the diffusion-based molecular communication is considered as one of the essential communication techniques among other molecular communication types. The key properties of diffusion-based molecular communication are as follows: firstly, it is a biologically inspired communication technique; secondly, the propagation of molecules is due to random walk of the molecules (thus, the energy is not needed for molecules movement) modelled by Brownian motion with random direction and there is a high delay in transmission [57, 92].

The research objectives addressed in this dissertation are given below

- To Model and analyse a bio-inspired molecular communication channel, through properties verification. Taking into account, the medium noise, and the channel reliability.
- To design a diffusion-based system to study the propagation medium and the parameters which have an impact on the communication among nanomachines. Then, to define and study different scenarios of consensus problem, taking into account, the medium noise, the nanomachines energy, properties verification and security issues.

The process by which these objectives are realised is broken down into the following steps:

- **Defining and verifying a simple communication system**: In this system, the transmitter and the receiver nanomachines are connected through a channel of nodes located in the distance between the transmitter and receiver. Thus, transmitted molecules pass through these nodes to reach the receiver nanomachine.
- **Problem specification**: For example, the energy of nanomachines as one of the possible problems in system configuration.
- **Algorithmic solutions step**: Including verification and simulation.

1.7 Contributions of the Thesis

1.7.1 Modelling and Analysing Bio-inspired Channels

Analysis of molecular communication channels through properties verification using PRISM model checker, taking in consideration:

- Medium noise,
- Channel reliability.

PRISM Model Checker:

- A tool for formal modelling and analysis systems;
- Model checking process consists of two main phases: model constructing, model checking;
- Model is represented in PRISM state-based language;
- Properties of the model to be verified, it should be expressed in a temporal logic language first.

Results:
The results of the PRISM properties verification show:

□ Acknowledgement verification in 1D molecular channel,

□ Acknowledgement verification in 1D channel with noise,

□ Probability of **transmission** *success/failure* in **bi-directional** and **multi-access** channel,

□ Probability of **receiving** *success/failure* in **bi-directional** and **multi-access** channel.

The experiments have been carried out in different sized networks. The results demonstrate that receiving acknowledgement can be affected by the size of the channel between the transmitter and receiver. The effects of noise and the amount of transmitted information molecules on the channel reliability are also demonstrated. The verification results show that PRISM model checker can be utilized in studying molecular communication models. However, building and verifying more complex models with larger sized networks requires longer time. In literature most research focus on studying the channel capacity in molecular communication. However, channel reliability and the insurance that the transmitted molecules are sensed (received) by the receiver nanomachine are an important challenges in molecular communication. Thus, it has been explored in Chapter 3, with different channel settings.

1.7.2 Designing diffusion-based systems

1. **Analysis of diffusion propagation medium**
   - Study the parameters that impact communication in diffusion based systems.
   - Different experiments carried out to explore:
     - **Maximum distance** that diffused molecules can reach (i.e., transmission range), taking into consideration the effects of following parameters:
       - noise;

    - ∗ residual molecules;
    - ∗ medium properties.
  - – **Pattern of diffusion** (how a receiver nanomachine can recognize the received information molecules), taking into consideration the effects of the following parameters:
    - ∗ distance;
    - ∗ time;
    - ∗ sensed molecular concentration;
    - ∗ interference.

Results:

(a) The experiment results to compute the transmission range show that:

- □ The transmission range decreases as the value of both noise and residual molecular concentration increase,
- □ The transmission range decreases as the value of the medium diffusion coefficient increases.

(b) The second case is that the transmitter nanomachine has a message consisting of six bits (as an example). The experiment results to compute the transmission range demonstrate:

- □ The higher transmission range is achieved with the message that consists of higher number of '1'. Alongside the effects of noise on the transmission range.

(c) The experiment results of diffusion pattern recognition show that:

- □ If the duration of symbol was relatively short, then the diffused symbols were not recognized correctly due to the short symbol duration. (Symbol duration is the time duration between two consequent transmissions). In this experiment the symbol duration was 0.02 *ms*.
- □ The receiver nanomachine was able to recognize the symbols correctly after increasing the symbol duration. The symbol duration increased to 3.2 *ms*.

(d) Experiments with two different transmitter nanomachines at different distances from the receiver nanomachines diffusing information molecules. The experiment results to check how the receiver nanomachine can recognize the pattern of diffusion show that:

- □ The receiver nanomachine can recognize the diffused symbols correctly. However, it cannot distinguish whether symbol came from one nanomachine or another. Due to the overlapping in the values of the sensed molecular concentration which come from each nanomachine in this experiment. Where the reason of the overlapping in these values came from the close distance between the two transmitter nanomachines.

- □ Changing the distance of the two transmitter nanomachines from the receiver nanomachine can make the overlapping quite low.

(e) In the case when one transmitter nanomachine is in a close distance from the receiver nanomachine, and the other nanomachine is in a far distance from the receiver nanomachine, the results show:

- □ The receiver nanomachine cannot recognize the diffused symbols correctly, because of the higher data rate of the sensed molecular concentration from the close transmitter nanomachine. Thus, the sensed molecular concentration from the far nanomachine is affected by the interference of molecules from the previous symbol duration,
- □ By increasing the symbol duration, the receiver nanomachine can recognize the diffused symbols correctly, distinguishing also if the symbol came from one nanomachine or another.

In literature there are research that aim to compute the distance between the transmitter and receiver nanomachine. However, we study the maximum distance that diffused information molecules can reach (transmission range). There are research which study the effects of Inter Symbol Interference on the channel capacity and the channel performance. However, we study the effects of the interference on recognizing the pattern of diffusion. Besides that, through experiments we study the effects of symbol duration, data rate, and distance on reducing the effects of the interference.

2. **Consensus problem in diffusion based system** The consensus problem is essential in any distributed system to fulfil an overall agreement or commitment to perform tasks. Thus, reaching consensus among nanomachines in molecular communication is an important topic. Different scenarios of consensus problem have been specified, taking in consideration:

- nanomachines energy;
- medium noise;
- properties verification;
- security issues.

(a) **Consensus protocol in diffusion system:**

Design and analyse consensus in an extended model from [59] by proposing a protocol. The proposed protocol consists of different processes during number of time slots. These processes have been implemented for two different scenarios:

- electing a leader nanomachine,
- proposing an energy harvesting model.

Results:

The experiments results show:

- □ Computation of the number of nodes in the nanonetwork,
- □ Computation of the sensed molecular concentration by each nanomachine during each time slot,
- □ Computation of the total number of time slots that are needed to perform the protocol steps,
- □ The results show the effects of energy constraint on the needed time slots number.

(b) **Consensus among $n \times n$ nanomachines:**

Proposal of a protocol based on two phase commitment protocol to study consensus problem in multi-dimensional model:

- implementation of the protocol steps in 3×3, 5×5 and 6×6 grids.

Results:

- □ The acquiring of consensus in the model is verified using PRISM model checker,
- □ The consensus problem in [59] is verified using PRISM model checker, through assuming a general case of $n \times n$ nanomachines, deployed arbitrarily in the environment.

(c) **Consensus in nanonetwork with an adversary nonomachine:**

Explore security issues in molecular communication, through:

- studying consensus problem in a nanonetwork of n nanomachines with the existence an adversary nanomachine,
- proposing a protocol in which nanomachines attempt to jam the communication among bacteria.

The consensus problem in a diffusion based network is explored with the existence of an adversary nanomachine, which aim is to jam the communication between the network nanomachines. In terms of security relevant issue, a protocol is proposed in order to apply a nanonetwork to jam the communication among bacteria, to prevent them from launching an attack. In litereture [54, 55] the issue of security in nanonetwork is addressed through highlighting the open questions and challenges. Also through outlining the possible threats and the directions of potential solutions by studying the immune system in human body. In our work we proposed a model with an adversary nanomachine which represents a threat for achieving consensus between nanomachines. The adversary nanomachine follows Poisson random distribution in diffusing its jamming molecules. The network nanomachines attempt to estimate the concentration of the jamming molecules. Thus, through k time slots, each nanomachine senses the molecular concentration and stores it in a vector. Then, each nanomachine attempts to estimate the average of the jamming molecular concentration, based on the stored molecular concentration during k time slots. Then, after estimating the jamming molecular concentration, the processes to reach consensus begin. Then we study another issue related to security, through exploring the biological process of quorum sensing in bacteria. This process is a form of consensus among bacteria population. In order to activate bacteria to perform its task (whether it is a useful or harmful), bacteria need to reach consensus first. Thus, the nanomachines in the defined model are employed to prevent a harmful bacteria from launching their attack. Through proposing a protocol that nanomachines follow. Thus, these nanomachines attempt to jam the communication among bacteria, through diffusing a molecule which has been tested in biological experiments to lock the bacteria receptors.

1.8 Structure of Thesis

This Ph.D. thesis is organized as follows:

Chapter 2 Literature Review : Presents a literature review of the related work and background issues. Starting by discussing the potentials of bio-inspired systems, and the main issues which have been investigated throughout litereture about molecular communication and molecular communication channel models. The last section of this chapter lists the topics explored in each chapter of the thesis and the research literature that are relevant to theses topics.

Chapter 3 Bio-Inspired Molecular Communication System : Presents a time-slotted one dimensional communication system between two nanomachines, that is based on bio-inspired rules. Then, using PRISM model checker tool, properties of this system have been verified. After that, the effects of the environment noise on the proposed system have been studied and analysed using PRISM, and fallowed by proposing a bi-directional system which has been extended to represent a multi access channel system with analysis of the success and failure of sending and receiving in the system.

Chapter 4 Performance Analysis of Molecular Communication Model : Introduces the proposed algorithm to measure the maximum distance in a diffusion based molecular communication. The chapter includes experiment results of the algorithm. Then, it explores the pattern of diffusion by defining the factors which can affect the sensed molecular concentration by the receiver, such as distance, interference, symbol duration and data rate. The experiment results related to the the pattern of diffusion are presented.

Chapter 5 Consensus Problem in Molecular Communication with Leader Election and Energy Harvesting Algorithms: Introduces the proposed protocol to reach consensus among nanomachines communicating via diffusion. The protocol's steps are applied first on a nanonetwork which required a leader election algorithm initially. The effects of nanomachine's energy constraint on the consensus protocol implementation are studied. In both cases, the time needed to reach consensus, taking into account how long is required to elect a leader nanomachine, and the duration required to harvest enough energy, is computed.

Chapter 6 Verification of Consensus Protocol for Diffusion based Molecular Communication: Presents the proposed protocol for consensus problem in time slotted model of $(n \times n)$ nanomachines grid. Then, the proposed protocol is verified using PRISM model checker, in grids of different sizes. The verification process carried out in three different experiments, the first one is deterministic experiment. While, the other two experiments take into consideration the effects of noise in environment and the effects of changing the value of threshold on the sensed molecular concentration.

Chapter 7 Consensus Problem with the Existence of an Adversary Nanomachine: Introduces the proposed protocol for consensus problem in diffusion based network with the existence of an adversary nanomachine. As an attempt to explore security issues in molecular communication. The adversary nanomachine aim is to jam the communication between the network nanomachines. The network nanomachines attempt to estimate the concentration of the jamming molecules. In a security relevant issue, this chapter presents the proposed protocol to apply a nanonetwork to jam the communication among bacteria, in order to prevent them from launching their attack.

Chapter 8 Conclusions and Future Work: This chapter concludes the thesis by reviewing the contributions with summery of the work presented. The chapter also presents discussion on possible future research directions.

1.9 Accepted Abstracts

The following section gives a brief description to the accepted abstracts.

1. **Athraa Juhi, Dariusz R Kowalski, Alexei Lisitsa,** *Probabilistic Model Checking of One-Dimensional Nano Communication System*, **The abstract of this paper accepted and presented in BCTCS 2016**

   The main objective of this paper is to model a simple time-slotted communication system between nanoscale machines in a one-dimensional environment. This communication system employs some bio-inspired rules that can be checked at each interval. The system model has been verified using the probabilistic model checking tool PRISM on different sized networks. We were able to verify that acknowledgement has been obtained, and thus, communication between these nano nodes has been ascertained. The results were promising for further study of more complex scenarios, such as, multi-access channels.

2. **Athraa Juhi, Dariusz R Kowalski, Alexei Lisitsa,** *Estimate the Number of Nodes in a Nanonetwork*, **Abstract accepted and presented in SCOR2016.**

   We consider a model of a nanonetwork consisting of N nano-machines, which are located in a space, and communicating between each other via diffusion. The goal of the model is to estimate N by certain devices in the nanonetwork, which can be done by adopting the mechanism of quorum sensing. Quorum sensing is a biological process that enables the synchronization of a population of bacteria. In order to synchronize with the group, each bacterium releases a particular type of molecules at a constant rate. The concentration of that type of molecules in the environment increases proportionally with the bacterial population. By this way, bacteria are able to sense their population density by detecting the level of that certain type of molecules. Thus, we are inspired by this biological mechanism to obtain the main objective of this model.

3. **Athraa Juhi, Dariusz R Kowalski, Alexei Lisitsa,** *Probabilistic Model Checking of Nano Communication System*, **Accepted abstract as a poster in UCNC 2016.**

   Our main objective is to model time slotted communication system between nanoscale machines in a grid of 3x3 nanonetwork. This communication system is adopting some bio-inspired rules that can be done at each time slot. All molecular concentration above threshold h will be distributed to the neighbours equally. We assume that the system is time slotted with slot duration of t. Each node at every time slot would check its concentration, if it is above threshold h or not. Every node can accept molecules up to buffer B limitation which is larger than h. We assume that sender and receiver nodes operate according to a number of mechanisms that can make them receive, send, and compare the amount of molecules they would receive each time slot and react according to the increment in the average concentration of the received molecules. In addition, sender node has the ability to wait for a while before sending new concentration.

4. **Athraa Juhi, Dariusz R Kowalski, Alexei Lisitsa,** *Performance Analysis of Molecular Communication Model*, **Accepted as an extended abstract and presented in IEEE NANO 2016.**

We intended to compute the maximum distance which the diffusion of the source could reach. Then we proposed an energy model to study its affect on the nanomachine performance. Besides that, we wanted to make sure that each node could recognize the pattern of diffusion of the source node.

5. **Athraa Juhi** *Consensus Problem in Molecular Communication with Leader Election and Energy Harvesting Algorithms*, **Accepted as a poster in ACM Wom-Encourage 2017.**

   We consider a communication network that consists of n nodes, one of these nodes $node_c$ has some special responsibilities to direct and control processes. The nodes, also called nanomachines, communicate through a shared unguided medium by stipulating and controlling diffusion processes. The messages in diffusion based molecular communication are encoded as molecules and conveyed by the changes in the concentration of molecules in the environment. The medium of communication might contain residual molecules from previous diffusion, and also contains other types of molecules which can be considered as a noise. We study consensus problem in molecular communication, inspired by model in[59], where the authors consider an iterative method for communication among nanomachines which enables information spreading and averaging in their nanonetwork. In this paper, we propose a consensus protocol among nanomachines in diffusion based molecular communication. The proposed protocol, includes two phases that take place throughout different time rounds. The first phase, is to estimate the number of nanomachines via nod_c, the second phase includes number of steps, where each of the nanomachine diffuse their initial value to nod_c. Then, $node_c$ computes the average of all initial value, which is considered as an important value to reach consensus. We consider two scenarios to implement the consensus protocol. In the first one, a leader election algorithm is utilized to elect a central node. In the second scenario, we assume that nanomachines have energy constraint, so we define an energy harvesting model as a nanomachine might not be able to communicate due to the lack of energy. In each scenario, we compute the consensus protocol's rounds number, taking into account the required time to elect a central node and the needed time to harvest enough energy.

Chapter 2

Literature Review

2.1 Introduction

To reach a comprehensive understanding of the research area, a general overview starting from bio-inspired systems, to nano-communications systems specifications and properties is needed. Besides that, an exploration on the communication mechanisms is used for the interconnection between nanomachines, the propagation channel properties, nanomachines energy, consensus problem in nano-systems, properties verification tools and a study of various existing distributed relevant communication models are all required for the development of this research.

This chapter includes a review of the literature relevant to the research on nanonetwork communication models. This survey is organized as the following: Section 2.2 discusses the potentials for bio-inspired systems and how it has been explored in some research in litereture. More details about molecular communications with its main three types, and how it has been discussed in the current research, and what are the main issues which have been investigated throughout relevant litereture, all that are presented in Section 2.3. Section 2.4 explores molecular communication channel models in details, and different techniques to encode information in molecules proposed in the literature, and other techniques to sense the information molecules by the receiver. Section 2.5 lists the topics explored in each chapter of the thesis and the research literature that are relevant to theses topics. Section 2.6 defines PRISM verification tool and gives the motivations of using it in analysing two models of the thesis. Section 2.7 presents the discussed issues in this chapter.

2.2 Bio-inspired Systems

There are different challenges facing the future network applications, such as; the increased complexity of large scale networks; the dynamic nature of such networks' heterogeneous architecture, resource constraints; and other challenges. However, in nature these challenges have been successfully handled, due to millions of years of evolution, which gave the urge to be inspired and to apply the biological mechanisms into the design and implementations of networks [52]. Biological systems exhibit many fundamentally appealing characteristics, such as robustness and being resilient to failures, adaptivity to environmental changes and the ability to perform complex behaviours using a restricted set of basic rules [51, 52].

The standard bio-networking architectures with a complete modelling approach and particular control frameworks have been explored in [173], where the authors attempt to apply biological rules and techniques to the design and implementations of network applications.

The authors in [47, 95] were inspired by the natural immune system to model network security and intrusion detection. In [126] one of the motivations is based on the interaction between an organ's cells and mutant cells (which could correspond to viral cells or cancer), in exploring alternative models for evolution on graphs networks. In [28] the authors exhibit the benefits of using a self-synchronization mechanism as a tool in wireless sensor networks to achieve global optimal decisions, and their inspiration is biological systems. The authors in [64] were motivated by the biological systems to support in the laying out of a scalable sensor network.

The authors in [52] provide a comprehensive demonstration of the potentials for bio-inspired networking through their survey, which is currently not fully recognized; and they also hope to boost the motivation for the research community to further explore this topic. The state of the art in bio-inspired networking based on examples of various networking paradigms is introduced. The authors highlight the necessity of modelling biological phenomena and their applications in networking by giving a list of some of the approaches in bio-inspired networking showing the advantages of using it in communication networks. This list includes: the field of swarm intelligence and social insects [36, 65], ant colony optimization [50], firefly synchronization [127, 159], artificial immune system [81], and cellular signalling networks [13, 53, 97, 151, 175].

Cellular signalling, in particular, is pertinent to the research on molecular communication in nanonetwork; thus, it is important to elaborate about its characteristics. Cellular communication process can be summarized in two steps: an extracellular molecule binds to a specific receptor on the target cell, which results in the activation of that receptor, then the receptor stimulates an intracellular biochemical pathway which produces a cellular response [52, 175]. Broadly speaking, cellular signalling can be divided into two techniques, thus [52, 97]:

In intracellular signalling, the signal is transmitted from an extracellular source to be conveyed through the cell membrane. At this point, inside the targeted cell, the process of signal transduction occurs, which includes complex signal cascades to transfer information, resulting in gene expression or an alteration in enzyme activity, representing the cell response.

In intercellular signalling, communication among cells can be carried out through cell surface molecules. In this signalling technique a surface molecule of one cell (or a soluble molecule), is released from one cell to bind directly to a specific receptor molecule on another cell.

By employing the bio-inspired approach, the nodes are modelled as a population of agents interacting with the environment, and each one of these agents has limited capabilities. Thus, achieving complex tasks can be accomplished through the collaborative behaviour of the population [48], which represents the basic principle of nanonetworks. Bio-inspired nano communication is a promising technique that can be used for information exchange among nano-scale devices [96]. A brief survey on informatics of cell signalling (communication) in terms of computer science and signal processing is presented in [106], exploring the framework of network informatics from the properties of the signal, information, coding and control to demonstrating the prospect of using it in designing nano communication systems.

2.3 Molecular Communications

In Chapter 1, a general description of nanonetworks different communication techniques was presented. In this chapter, molecular communications in particular is explored, as the main focus of the research in this dissertation is diffusion based molecular communication. Molecular communication is considered to be a bio-inspired promising technique which can be appropriate for communication between nanomachines [66, 135, 137]. However, there are a number of challenges [137, 138, 168]: for instance, the communication among nanomachines can be very unreliable and subject to long delay. Thus, designing communication protocols for nanomachines, should take these properties into consideration. Besides this, the development of such protocols needs to be suitable and proper to biological processes, and must consider domain factors such as particle decay and nanomachines computational constraints, alongside increased delay [170].

The following subsections give an overview of the literature on short range molecular communication techniques (i.e., molecular motors, calcium signalling, diffusion based molecular communication):

2.3.1 Molecular Motors

This technique is also known as **walkway-based molecular communication**, as the information molecules propagates through transporting molecules on predefined pathways, which connect the transmitter nanomachine to the receiver nanomachine. In [129] initial designs of molecular motors based molecular communication was presented, describing the environmental assumptions and the architecture of systems that use molecular motors to perform communication. The authors in [63] studied the pathways (microtubules tracks) which connect the transmitter nanomachines and receiver nanomachine, where molecular motors move over it, by proposing an approach for arranging microtubules for composing microtubules network. The authors consider two approaches to design this architecture of self-organizing microtubule tracks: the first approach is based on microtubule polymerization and de-polymerization; the second one is based on using molecular motors to reorganize the pathways.

The information molecules are enclosed in a *lipid bilayer* called the '*vesicle*' before being uploaded to the molecular motors. Vesicles can be defined as biological capsules which have a high affinity with molecular motors [8]. Vesicles are explored in [132], suggesting that vesicle can protect information molecules from environmental noise, and considering it as an interface between the transmitter nanomachine and the propagation system and also between the propagation system and the receiver nanomachine.

2.3.2 Calcium Signalling

This communication approach is also known as **advection-based molecular communication**, where the information molecules propagates through diffusion in a fluid medium, molecules being transport through *gap junctions* which can represent a channel between adjacent cells. The design of gap junction based molecular channels has been proposed and described in detail in [140, 141], indicating that the adjustment in the permeability and selectivity of gap junctions, can control different communication issues, such as; signal switching, filtering and aggregation functionalities. The channel capacity of calcium signalling based molecular communications has been investigated in [78], where the authors studied the signalling capacity of an astrocyte cell, by building a calcium signalling scheme on two cells and exploring the channel capacity based on noise level and symbol duration (where, *symbol duration*, can be defined as the time duration between two consequent transmissions).

The authors in [31] have proposed and analyses a calcium signalling molecular communication of different cellular tissues. A comparison study of calcium signalling in tissues consisted of three specific cell types as presented in [30]. The analysis of this study focuses on the dynamics of calcium concentration, and how it is influenced by intracellular signalling interference, and the channel capacity. A review of mechanisms used to enable calcium signalling based molecular communication system design has been presented in [29]. Besides this, information molecules encoding , modulation, propagation and; decoding have been explored, alongside future research directions. Calcium signalling, in which a low concentration of calcium ions is propagated in a fluid medium, can be considered as a special case of diffusion [108].

2.3.3 Diffusion based Molecular Communication

Information molecules are propagated through the fluid environment as a consequence of their spontaneous diffusion [18]. Diffusion based molecular communication is considered as one of the fundamental communication mechanisms, which can be employed for communication nanomachines; where, the information molecules are encoded onto the quantity of molecules; hence, the receiver nanomachine decodes the information molecules based on the number of molecules it receives during a predefined time interval (which is also known as *symbol duration*) [5].

In diffusion based molecular communication, the interactions (such as, collisions and electrostatic forces) among the transmitted molecules can be neglected because, the concentration of the diffused (transmitted) molecules is much lower than the concentration of the environmental fluid molecules [111]. For example, in [43] the author explored calcium signalling and stated that the range of the extracellular concentration of calcium ions was in the millimolar; however, the water concentration (which represents the main component of the extracellular fluid) was about 55.5 molar, which is higher with more than 4 orders of magnitude [111]. Thus, the molecules emitted by the transmitter nanomachine can be modelled by Brownian motion [93, 111]. Molecular diffusion can be characterized by Fick's laws of diffusion [152], with a diffusion coefficient that is a homogeneous in space and time, as the movement of each molecule is independent [108].

Diffusion based molecular communication has been investigated in [19], where the emitted information molecules are following the Brownian motion to cover the distance between by the transmitter nanomachine and the receiver nanomachine. Thus, the propagation of this pulse can be analytically modelled by solving Ficks laws of diffusion. If the transmitter releases Q molecules at the instant $t = 0$, the molecular concentration at any point in space is given by:

$$c(d,t) = \frac{Q}{4\pi Dt^{3/2}} e^{\frac{-d^2}{4Dt}} \tag{2.1}$$

where D is diffusion coefficient of the medium, t is time and d is the distance of a specific point from the transmitter nanomachine. Where Equation 2.1 is assumed for $3D$ topology network.

In [56] an estimation of the achievable information rates is presented for a diffusion based molecular communication, and information is encoded as a set of distinct molecules. Through extending the framework and results in [57], the outcomes in [56] show large gains in the information rate, compared to the case where the emitted molecules are of the same type.

In the literature, various studies have aimed to model the physical channel of the diffusion based molecular communication [108], governed by Fick's laws, in particular some research have explored the channel transfer function [153], while other research focused on channel capacity from information theoretical aspects [16, 21, 107, 139, 155]. The noise effects on channel capacity have been investigated in [131, 154, 157], concluding that diffusing a larger number of molecules increases the signal to noise ratio and could reduce the noise impact. The authors in [109] presented the design challenges and principles in diffusion based molecular communication, considering the propagation delay and channel distortion to be the main challenges.

Synchronization between the transmitter nanomachine and receiver nanomachine is considered one of the challenges in molecular communication systems. Synchronization is important, as it can affect the error rate performance of the receiver nanomachine [164]. Mostly in literature related to molecular communication, authors assume that the system is synchronized; however, studying biological mechanisms brings opportunity to find different tools that can be used to overcome challenges. In biology, there is a mechanism known as *quorum sensing* [3], in which bacteria can utilize to synchronize their behaviour, through the emission and sensing of a certain type of molecules called autoinducer. The authors in [1, 3] proposed *quorum sensing* as a tool to achieve synchronization among nanomachines in diffusion based molecular communication system. In [3, 4] the mechanism of *quorum sensing* was utilized as a tool to acquire signal (transmitted information molecules concentration) amplification, after achieving synchronization between a number of transmitters nanomachines, so that they emitted the same signal.

2.4 Molecular Communication Channel Models

In Chapter 1 a brief description of the main process of a molecular communication based system was presented. Here a more detailed description, based on the literature, is given. The basic components of molecular communication systems are: the transmitter, a propagation medium (channel), and the receiver.

The transmitter is the nanomachine which releases information molecules into the environment. *The transmitter* nanomachine encodes information in the diffusing molecules (*the input signal* of the diffusion channel). There are two main methods of information encoding [156] considered in the literature: in the first method [91, 92, 134] information is encoded at the time of the emission of each molecule in the diffusion medium (the channel), in the second method [9, 19, 153, 155] the information is encoded according to the fluctuation of molecules concentration in the medium. The method of information encoding chosen may depend on the application that the diffusion system is employed for [157].

Through the next step, these encoded molecules propagate through the environment (*channel*) by following diffusion dynamics, i.e the movement of molecules from areas of higher concentration to areas of lower concentration.

The receiver nanomachine is able to *detect* and *receive* the encoded information that are coming from the medium (*channel*), and to extract the information message from the received molecules [9, 18, 155]. The *detection* of information molecules can depend on how *the transmitter* nanomachine originally encoded the information. *The receiver* nanomachine computes the time of the molecules arrival at its location, in case the information is encoded according to the *time of molecule release* [91, 154, 156].

In case the information molecules have been encoded according to the variation of molecules in the environment, there are different types of *detection* techniques proposed in different models in the literature, which is discussed later in this section, though the authors in [116, 118, 120] studied concentration encoded molecular communication and suggested the main schemes that receiver can use to detect concentration encoded information molecules. These detection schemes are: sampling-based detection technique, energy based detection and observation in FSK modulation [120].

The receiver nanomachine captures (*receives*) molecules through its surface. The receiver's surface structure (*receptors*) type varies, it may be permeable to particular molecules (receiving through *absorbing*) [18, 84, 178], or the receiver's surface receptors may bind to the information molecule (receiving through *binding*) [18, 61, 156]. After capturing the information molecules *the receiver* nanomachine decodes the information carried by it, either through different chemical reactions or by computing the received concentration in order to recognize its type [18].

2.4.1 Receiver's Measurement Techniques of Molecules

2.4.1.1 Detection of Time Encoded Molecules

In [91, 92] the authors studied molecular communication, where the information was encoded based on the time of molecules release. After the molecules are released by the transmitter at position x and time t_0, these molecules are propagated in the medium, assuming that it is a fluid medium. Considering $X(t)$ as the position of the molecule at time t, if the fluid medium is static then the molecules scatters in any direction with equal probability. The probability density function of the location of a molecule diffused from the transmitter at position x at time t can be computed using the following diffusion equation $P(x, t)$:

$$P(x,t) = \frac{1}{(4\pi Dt)} * \exp(-\frac{x^2}{4Dt}) \tag{2.2}$$

where: D is the diffusion coefficient of the medium. The authors in [91] assume that the propagation of molecules in Equation (2.2) to be in one dimensional.

The models in [91, 92, 134] are durationally time slotted, T_s. The transmitter nanomachine diffuses the molecules at beginning of one of N time slots. The molecules then propagated through the medium and are captured by the receiver nanomachine in a later time slot. The receiver nanomachine then estimate the time slot in which the molecules were diffused. The receiver nanomachine would waits for M time slots, the authors in [91, 92] set M to be long enough for most of the diffused molecules to be captured by the receiver nanomachine. The receiver nanomachine decodes the transmitted information according to the number and time of absorption of molecules [91].

2.4.1.2 Detection of Concentration Encoded Molecules

In [116, 118, 120] the concentration based encoded information was studied intensively. Here, the focus on presenting two of the detection techniques from these papers: *Sampling-based Detection* and *Energy-based Detection.* In [120] the authors assumed that, when the transmitter nanomachine diffuses a unit u of molecules during each second of the time slot T_b (bit duration), this can represent '1'; and when the transmitter nanomachine does not diffuse any units of molecules during T_b , ths represent '0'. The authors assume that if the transmitter nanomachine diffuses a unit u of molecules at time t, and the distance between the transmitter and receiver nanomachines is d, then The probability density function of the location of the particle can be obtained by:

$$F(d,t) = \int_0^t \frac{u}{(4\pi Dt)^{\frac{3}{2}}} * \exp(-\frac{d^2}{4Dt}) \tag{2.3}$$

where: D is the diffusion coefficient of the medium. $F(d,t)$ is also known as the throughput of the molecular propagation channel. Where Equation 2.3 is assumed for $3D$ topology network.

- In ***Sampling-based Detection***, the transmitter nanomachine diffuses unit u of molecules during the time slot, so there would be a random bit sequence of '1' of N bits diffused by the transmitter nanomachine which will propagate through the channel [116, 118, 120]. The throughput $F(d,t)$ can be sampled during the bit duration T_b at any appropriate time instant [120]. As the authors in [116, 118, 120] assumed that the transmitter nanomachine is synchronized with the receiver nanomachine. They assumed that the proper time instant was in the middle of any bit duration. Thus it can be represented as follows:

  $$t_{samples} = \frac{T_b}{2}, \frac{3T_b}{2}, \frac{5T_b}{2}, ..., \frac{(2n-1)T_b}{2}, ... \frac{(2N-1)T_b}{2} \tag{2.4}$$

  where N is the entire number of bits in a bit sequence, T_b is the duration of each bit, and n=1,2,3,...,N is the index of bits [120]. To compute the detected concentration of molecules by the receiver nanomachine the authors in [120] represented it as detection variable Z_{SD}

  $$Z_{SD} = F\left[d, \frac{(2n-1)T_b}{2}\right] \tag{2.5}$$

  where Z_{SD} in 2.5 is for detecting the n^{th} bit. The bit duration T_b for a specific data rate U can be fixed at $T_b = 1/U$. Z_{SD} is affected by the distance d between the transmitter and receiver nanomachines [120]. The authors in [116, 118, 120] assumed that, in *Sampling-based Detection* the receiver nanomachine has knowledge of the diffusion rate of molecules U, and that the threshold can be given as:

  $$h_{SD} = \frac{u_{average}}{2} \tag{2.6}$$

  where: h_{SD} is the threshold, and $u_{average}$ represents the average of bit '1' during the bit duration T_b.

- In ***Energy-based Detection*** the waiting time is quite longer than the waiting in Sampling-based Detection. In this approach the throughput $F(d,t)$ of diffused molecules is integrated over any whole bit duration T_b, so that the detection variable Z_{ED} is the accumulated number of molecules during that T_b

[111, 116, 118, 120]:

$$Z_{ED} = \int_{(n-1)T_b}^{nT_b} F(d,t)dt \tag{2.7}$$

where: n=1,2,3,...,N is the index of bit '1', and N is the total number of bits in the random bit sequence. During the bit duration T_b, the receiver nanomachine detects the transmitted bit when the accumulated molecules concentration is greater than or equal to the threshold h_{ED} which is represented as in [120]:

$$h_{ED} = \frac{\int\limits_{0}^{2T_b} u(t)dt}{2} = \frac{u_{avarage}T_b}{2} \tag{2.8}$$

This means that h_{ED} is half the entire transmitted molecules concentration during the time 0 to $2T_b$.

- The previous two detection techniques are the general known techniques in *concentration based information encoding.* Thus, more detection techniques can be observed in some molecular communication literature. For example, in [83] the authors aimed to estimate the distance between the transmitter and receiver nanomachines, based on the peak of concentration of molecules detected by the receiver nanomachine (***peak detection***). The authors assume the network topology to be 1D and consider that the value of the impulse response in the diffusion channel after the transmitter nanomachine has diffused molecules could be obtained by:

$$c^*(d,t) = \begin{cases} \frac{1}{\sqrt{4\pi Dt}} e^{\frac{-d^2}{4Dt}}, & t \in (0,\infty) \\ 0, & t = 0 \end{cases} \tag{2.9}$$

where: D is the diffusion coefficient of the medium, t represents time (initially $t = 0$) and d is the distance between the transmitter and receiver nanomachines. When the transmitter nanomachine diffuses Q molecules in an impulse spike, the concentration would be Q times the impulse response, as in the following:

$$c_1(d,t) = Qc^*(d,t) \tag{2.10}$$

This means that the maximum concentration of molecules detected at the receiver nanomachine is $c(d,t)$. As in the molecules concentration expression $c_1(d,t)$ time t starts at 0 and it goes to infinity, a global maximum value of $c_1(d,t)$ should exist in $[0,\infty)$, through solving the following expression:

$$\frac{\partial c_1(d,t)}{\partial t} = \left(\frac{-1}{2t} + \frac{d^2}{4Dt^2}\right)\frac{Q}{\sqrt{4\pi Dt}} e^{\frac{-d^2}{4Dt}} = 0 \tag{2.11}$$

The solution is $t = \frac{d^2}{2D}$; thus, it equals to the time at which the molecules concentration reaches its peak at the receiver nanomachine. In [83] referred to it as t_p to denote the time to reach the peak, where $t_p = \frac{d^2}{2D}$. In this case the receiver would wait for t_p time to detect the information molecules. As the authors in [83] aimed to estimate the distance d, they demonstrates that it is possible to estimate the distance d if the receiver nanomachine is able to measure the peak of concentration correctly, where the value of peak concentration in time $t \in [0, \infty)$ can be obtained through:

$$C_p = c_1(d, t_p) = \frac{Q}{d\sqrt{2\pi}} \tag{2.12}$$

This gives us the idea that here the receiver nanomachine depends on *exact* values of the molecular concentration peak rather than a threshold.

In [125] the authors study the characteristics of (MIMO) Multi-Input Multi-Output transmissions in diffusion based molecular communications. The function to represent the molecular concentration at the receiver nanomachine after the transmitter nanomachine diffuses Q molecules is given in Equation (2.10). Although in [83] the value of t_p is considered to be $t_p = \frac{d^2}{2D}$. However, the time t_p in which the concentration reaches its peak at the receiver, in [125] is assumed to be computed as $t_p = \frac{d^2}{6D}$, and it is assumed that the value of peak concentration can be computed as the following:

$$C_p(d) = Q(\frac{3}{2\pi})^{3/2}\frac{1}{d^3} \tag{2.13}$$

Thus, Equation (2.12) computes the molecules concentration peak according to the assumption in [83]. While Equation 2.13 computes the value of peak concentration according to the assumptions in [125], where the authors in [125] assume that their nanomachines are communicating in a 3D environment.

As the authors in [125] took in account the effects of interference in the environment, the molecules concentration detected by the receiver nanomachine can be obtained through:

$$S = XQC_p(d) + I \tag{2.14}$$

where X can be either 0 or 1, if the transmitter nanomachine diffused molecules that represent 1, while, if no molecule emitted by the transmitter that signifies 0; I represents the interference in the environment (i.e., the interference of molecules diffused in previous time, on the current diffused molecules).

2.4.2 Receiver's Molecules Capturing Techniques

2.4.2.1 Capturing Molecules Through Absorbers

After the information molecules are emitted by the transmitter nanomachine, it propagates through the environment to reach the receiver nanomachine. A molecule can be received only when it binds to one of the receptors on the surface of the receiver nanomachine; after that, for different types of receptors, information molecules are absorbed by the receptors. Thus, each molecule contributes to the signal only once, as it being absorbed by the receptor [7]. In [179] the authors analysed a 3-D channel characterization in a molecular communication with an absorbing receiver. In [178] an end-to-end molecular communication simulator was utilized to verify the proposed channel model with an absorbing receiver. The authors in [7] investigated the effect of receptor size and density on the signal reception in a channel with an absorbing receptors receiver nanomachine.

2.4.2.2 Capturing Molecules Through Binding

In biology, the binding between the receptor and molecules is considered to be an equilibrium process, after the emitting and propagation of the molecules, it would be detected by the receptors through binding, which causes an activation of the receptor, leading to chemical interactions inside the cell and eventually a cellular response, when the molecule is removed from the receptor the cellular response is terminated [114]. Molecules (also called ligands) bind to the receptors and dissociate from them following the law of mass action, according to which, binding affinity can be defined as the measurement of how well a molecule fits a receptor, which is inversely related to the releasing (dissociation) constant (a parameter for removing molecules from receptors). A molecule that fit well with a receptor corresponds to a high affinity and low dissociation constant [15]. The receiver has a large number of receptors that molecules can bind to; from it, the receiver can estimate the concentration by averaging the total number of all binding receptors [61].

In [156] the authors explore the noise at the reception of the molecular information in diffusion based molecular communication with binding receptors through a mathematically analysed model of the reception noise. In [61] the authors develop a model to investigate the limitations of the receiver concentration sensing, in order to conclude the maximum rate of ligand-receptors that can be received in a diffusion based molecular system.

2.5 Challenges in Diffusion-Based Molecular Communication System Design

There are different properties in diffusion based molecular communication, which require that most of the techniques and protocols developed for wireless networks be reconsidered, in order to utilize them (after enhancing them or combining various existing relevant communication models) in molecular communication [109]. This research study motivated by the Beeping model [6, 82] and the SINR model [89, 90], as well as the bio-inspired models which integrate artificial nanosystems with natural bio environments.

Molecular communication channel capacity has been explored in a range of studies, as mentioned earlier. Chapter 3 of this dissertation studies the communication model, in which the channel between nanomachines consists of a number of locations (nodes) placed at the distance between the transmitter and receiver nanomachines (it is possible to imagine it as a tissue of cells, composed of a number of adjacent connected cells, the first cell being considered the transmitter and the last cell the receiver, while the cells between them are known as the channel). Calcium signalling [43] can be counted as one of the utilized communications techniques between adjacent cells; thus, it has been considered as a communication mechanism between the nanomachines in this model. However, the channel nodes of the model have a different mechanism, inspired by the sandpile model [85]. This model has been analysed and the process of acknowledgement has been verified. The effects of noise on the communication between the nanomachines have been studied. The same model has been extended to represent a bi-directional channel and multi-access channel; and the probability of success/failure of both transmission/ receiving information molecules has been verified in the extended models.

Estimating the distance between nanomachines in molecular communication has been studied in few research. For examples: the authors in [130] adopted techniques from electronic radio networks and applied it to molecular communication to measure the distance between nanomachines in a nanonetwork, in this setting, a nanomachine1 requests other nanmachine2 to transmit information molecules as a feedback signal (signal spike of molecules) throughout a short time slot, in order to measure the distance to nanomachine2. When a nanomachine1 receives the feedback signal, it estimates the distance through measuring the round trip time and the signal attenuation on the received signal. In [144]the authors estimated the distance through measuring the strength of the channel impulse response in diffusion based molecular communication (as it decreases when the distance expands). In [83] the authors estimated the distance in a one-dimensional diffusion based molecular communication through examining the peak concentration and double spikes. The results show that the distance estimation can be affected by the noise of the diffusion channel.

Chapter 4 studies molecular communication through a number of issues related to the performance of nanomachines. Starting by proposing an algorithm to measure the maximum distance in diffusion based molecular communication, assuming that a nanomachine would keep diffusing information molecules over a certain time interval, computing the distance that these information molecules can reach, and considering the noise effects on the computed distance. Then the pattern of diffusion has been explored, if one nanomachine in a distance *d* diffuses information molecules, how can the other nanomachine in the network distinguish that information? Then, the pattern of diffusion has been studied in case two nanomachines diffusing information molecules, in a manner that if one nanomachine diffuses the other one waits.

Energy of nanomachines in molecular communication has been explored in related literature motivated by the energy harvesting process in living cells from biological prospective. In [45, 46] the authors presented nanonetwork model for energy harvesting, in which molecules are considered as discrete entities, representing resources in a confined environment. The authors stated that the model can be used to understand the general properties of molecular communication, indicating that applying suitable resource harvesting mechanisms could make the model achieve an infinite network lifespan. In[100] an energy model presented for a diffusion based nanonetwork. Taking into account that in diffusion based communication system the energy is spent for molecule synthesis, production of the secretory vesicle, and carrying vesicle close to the cell membrane. Explaining that the energy budget is necessary to take into consideration for analysing capabilities and achievable performance in molecular communication system.

Quorum sensing process [77, 142] is an example of signalling between bacteria, where bacteria can use it to estimate the density of their population in the environment through estimating the concentration of a certain type of molecules. Consensus problem in diffusion based molecular communication has been studied in related research [59, 60, 62]. Mainly, [59] trying to map the Quorum Sensing to consensus problem under diffusion based molecular communication. The goal is to study consensus problem by spreading information about an event or any variation through a diffusion based network. Through communication all nanomachines try to obtain the best estimate of this random variable.

Chapter 5 studies consensus problem in molecular communication by proposing a protocol for the nanomachines, inspired by the consensus model in [59]. Two scenarios to implement the consensus protocol have been considered: (1) a nanonetwork without special node that control the protocol steps; thus, leader election algorithm has been adopted to elect that special node, (2) a nanonetwork with energy constraint nanomachines; thus, an energy harvesting algorithm has been proposed. In both scenarios, the number of rounds required to implement the protocol steps have been computed.

In Chapter 6, the consensus problem in time slotted model of a grid of $(n \times n)$ nanomachines communicating through diffusion has been explored. Inspired by the consensus model in [59], the aim is to study consensus problem in molecular communication but using multi-dimensional model. The acquiring of consensus in the model has been verified using PRISM model checker. The the authors in [59] stated that the general case of networks is more difficult to be analysed. Thus, PRISM model checker is also used to verify reaching consensus in their model. However, the nodes are assumed to be deployed arbitrarily, and not necessarily that each node observes the same distances to the other nodes in the network, as it was considered in [59].

Security in nanonetworks has been explored in few researches, as it is considered a serious challenge. The authors in [54, 55] discussed security in nano communication exploring the forms of possible threats and attacks, through studying sensor networks, in order to derive security requirements and to check the possibility of applying the available security solutions to the nano communication. Suggesting that new security solutions needed for the bio-inspired nano communication, as it have different functionalities, different environmental rules, therefore the existing security protocols won't be applicable. This led to form a new security field known as *biochemical cryptography*, which can be considered as new research direction. It can be used to provide a secure bio-inspired nano communication, as the protection mechanism is based on biological and molecular processes.

In [169] the idea of using the characteristics in human immune system as basis to establish security in nanonetwork is presented.

The authors in [115] gave an overview of the conventional ways to tackle attacks in wireless networks. Then, an inclusive general characterization of security and privacy in molecular communication is presented. The authors state that a collaboration between many diverse disciplines is required in order to efficiently understand the security in molecular communication based systems.

In Chapter 7, an attempt to explore security issues in molecular communication, consensus problem in diffusion based network with the existence of an adversary nanomachine is presented.The adversary nanomachine aim is to jam the communication between the network nanomachines. The adversary nanomachine is following Poisson random distribution in diffusing its jamming molecules. The network nanomachines attempt to estimate the concentration of the jamming molecules. Thus, through k time slots, each nanomachine senses the molecular concentration and stores it in a vector. Then, each nanomachine attempts to estimate the average of the jamming molecular concentration, based on the stored molecular concentration during k time slots. After estimating the jamming molecular concentration, the processes to reach consensus start, where each nanomachine in the network has an initial value. Each nanomachine diffuses its initial values to a special node in the network. This special node computes the average of all initial values and diffuses it to the network nanomachines. The special node is assumed to estimate the jamming molecular concentration in the same way and during the same interval that the network nanomachine attempted to estimate it. Thus, the special node takes in consideration the jamming molecular concentration when it computes the average of all initial values. Furthermore, this chapter explores the biological process of quorum sensing in bacteria. This process is a form of consensus among bacteria population. In order to activate bacteria to perform its task (whether it is a useful or harmful), bacteria need to reach consensus first. Thus, the nanomachines in the defined model are employed to prevent a harmful bacteria from launching their attack. Through proposing a protocol that nanomachines follow. Thus, these nanomachines are attempting to jam the communication among bacteria, through diffusing a molecule which has been tested in biological experiments to lock the bacteria receptors.

2.6 Verification Tool

PRISM is a verification tool that can be utilized for formal modelling and analysing of systems. It has been used to analyse different systems, which have applications in various domains, such as randomised distributed algorithms, biological systems, communication and multimedia protocols, security protocols and many other systems [148]. An overview of probabilistic model checking and of PRISM software tool has been presented in [103], exploring the limitations of model checking techniques, characterizing the way to overcome them, and highlighting the main challenges in this research domain. [105] describes the applications of PRISM model verification tool in studying biological systems and stochastic biological models, such as those resulting from biochemical reactions (as an example, cell signalling pathways, which result from molecules binding to a receptor). The author in [44] examined the utilization of probabilistic model checking to guarantee nanoscale devices behaviour.

As mentioned above, PRISM has been used in analysing and verifying biological systems which gives a motivation to use it in verifying two models described in this dissertation. Besides that, in diffusion based molecular communication, the molecules propagation in the environment is a stochastic process, as it is a consequent of spontaneous diffusion [109].

To give PRISM is a tool for formal modelling and analysis systems. It can be used to build and analyse several types of models, such as discrete-time Markov chains (DTMCs), continuous-time Markov chains (CTMCs), and Markov decision processes (MDPs). Basically the system behaviour is represented by constructing a mathematical model. Then the model is analysed using formally-specified quantitative properties, these properties are described using temporal logic language [103]. It is possible to utilize PRSIM to reason, not only about the probability that a system's behaves in a specific manner but also about a vast domain of quantitative measures which are related to the system behaviour [33, 149].

The process of model checking consisting of two main phases, namely model constructing, model checking, as figure 2.1 shows [149].

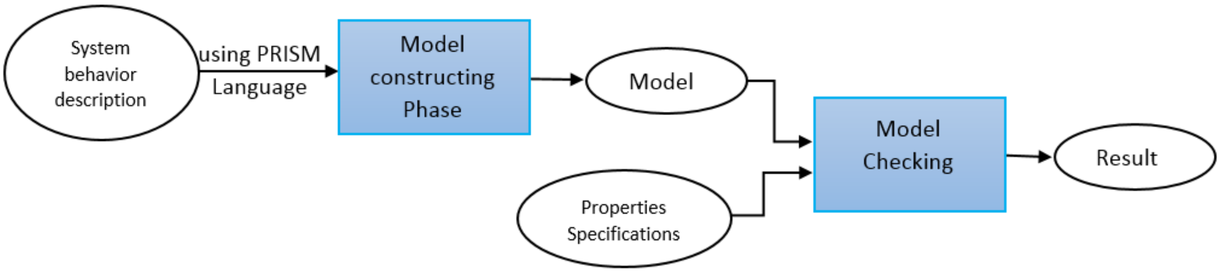
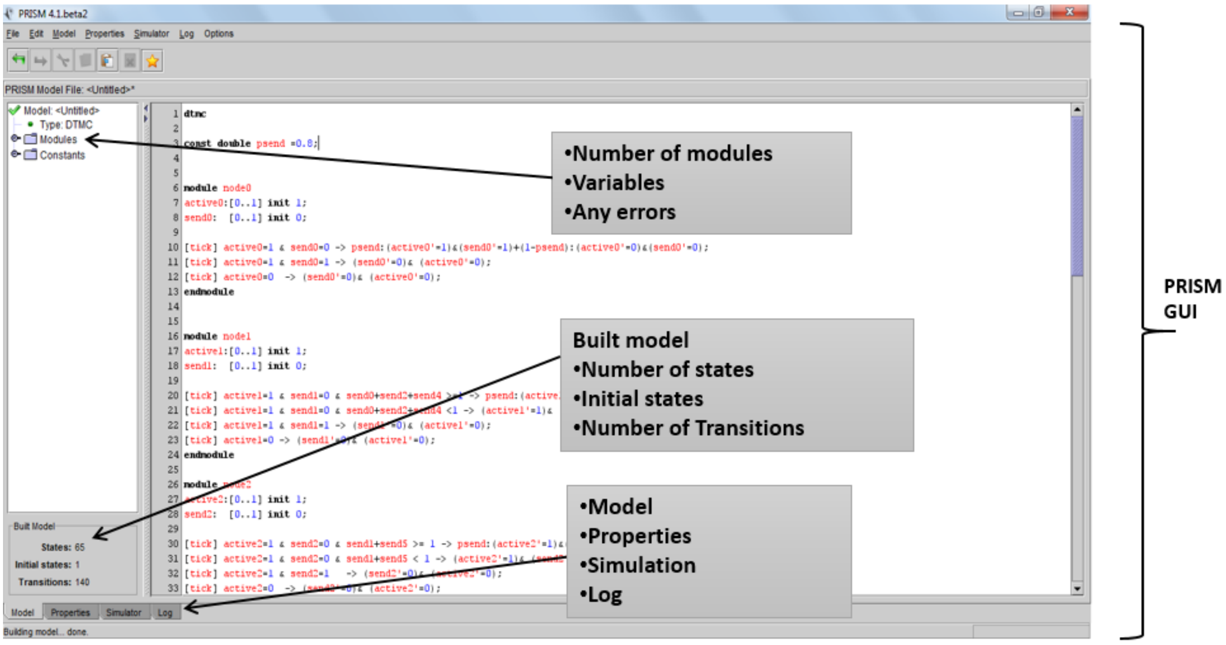


Figure 2.1: Model Checking Process [149]

The model construction phase is the process of converting the corresponding model into a PRISM language description, while model checking is processing/analysis of a constructed model by verifying a property specification and determining the result of that property [102].

Figure 2.2 represents a snapshot of the GUI of PRISM model checker. More details about representing models in PRISM are presented in Subsection 3.2.2.



Model: <Untitled>
Type: DTMC
Modules
Constants

Built Model
States: 65
Initial states: 1
Transitions: 140

```
dtmc

const double psend =0.8;


module node0
active0:[0..1] init 1;
send0:  [0..1] init 0;

[tick] active0=1 & send0=0 -> psend:(active0'=1)&(send0'=1)+(1-psend):(active0'=0)&(send0'=0);
[tick] active0=1 & send0=1 -> (send0'=0)& (active0'=0);
[tick] active0=0  -> (send0'=0)& (active0'=0);
endmodule


module node1
active1:[0..1] init 1;
send1:  [0..1] init 0;

[tick] active1=1 & send1=0 & send0+send2+send4 >=1 -> psend:(active
[tick] active1=1 & send1=0 & send0+send2+send4 <1 -> (active1'=1)&
[tick] active1=1 & send1=1 -> (send1'=0)& (active1'=0);
[tick] active1=0 -> (send1'=0)& (active1'=0);
endmodule

module node2
active2:[0..1] init 1;
send2:  [0..1] init 0;

[tick] active2=1 & send2=0 & send1+send5 >= 1 -> psend:(active2'=1)&
[tick] active2=1 & send2=0 & send1+send5 < 1 -> (active2'=1)& (send2
[tick] active2=1 & send2=1   -> (send2'=0)& (active2'=0);
[tick] active2=0  -> (send2'=0)& (active2'=0);
```

Building model... done.

FIGURE 2.2: The GUI of PRISM model checker

2.7 Summary

This chapter presents a literature review of the related work and background issues. Starting by discussing the potentials of bio-inspired systems, and the litereture which explore it. Then, the main issues which have been investigated throughout litereture about molecular communication types are discussed. Moreover, molecular communication channel models and information encoding techniques in literature are presented. The last part of this chapter lists the topics explored in each chapter of the thesis and the research literature that are relevant to theses topics, including PRISM verification tool.

Chapter 3

Bio-Inspired Molecular Communication System

3.1 Introduction

Molecular communication is considered a bio-inspired paradigm, in which molecules are transmitted, propagated and received between nanoscale machines. Establishing controlled molecular transmissions between theses nanomachines represents a major challenge. The main objective of this chapter is to model a simple time-slotted communication system between nanomachines by employing some bio-inspired rules that can be checked at each interval. This chapter is organized as following. In Section 3.2, abiologically inspired one-dimensional nano system has been defined; thus, in subsection 3.2.1 the model described; subsection 3.2.2 introduces the PRISM model checker, which has been used through out this chapter to verify the proposed system, and in subsection 3.2.3 the results of verifying properties related to the proposed system are given. Section 3.3 examines the effects of noise on the proposed model in the first section, along side properties analysis, using PRISM. In Section 3.4, a bidirectional nano system is proposed, with analysis of the system using PRISM, based on the proposed rules in the bi-directional system; subsection 3.4.1 discuses multi-access channel based nano system with its PRISM verification results. In Section 3.5 a summery of the discussed issues is presented.

3.2 One-Dimensional Bio-Inspired Nano System

The system consists of one transmitter nanomachine and one receiver nanomachine, and the channel between these two nanomachines is represented as a line of locations, Figure 3.1 represents a simple depiction of the system. The aim of the proposed communication model is to make sure that the receiver nanomachine senses the information molecules emitted by the transmitter nanomachine, by utilizing a verification tool to confirm the acknowledgement from the receiver nanomachine. The system can be imagined as a tissue shaped from adjacent cells, and the emitted information molecules from the transmitter nanomachine pass through these adjacent cells to reach the receiver nanomachine, in the same way as with *calcium signalling* communication [43]. However, the proposed rules of the communication channel are inspired by the *sandpile model* [85, 146], though; sand is substituted by molecules in implementing the proposed communication rules. The *sandpile model's* basic idea is that, assuming different locations with associated values that represent the grains of a sand pile slope, and through time, grains of sand can be added on top of that pile, and over time the pile grows until the slope surpasses a specified value (representing a threshold), resulting in the collapse of the pile at that location and sand grains being transferred to adjacent locations; consequently, the slope of the adjacent locations increase [27].

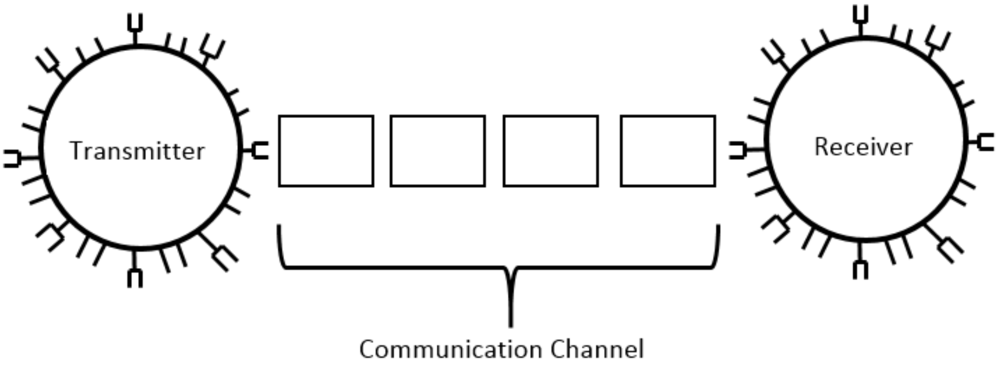


FIGURE 3.1: Basic Depiction of the System

3.2.1 Model

Network Environment: The system consists of a line of n locations Q_i (where Q_i, for $i \in \{1, 2, ..n\}$, are nodes that represent a channel) in a distance d between a transmitter Q_T and a receiver Q_R nanomachines in a fluid medium. Initially, molecular concentration is distributed randomly among each of the channel nodes Q_i, the transmitter Q_T and the receiver Q_R nanomachines. In the *sandpile model*, grains of sand could be added on top of the pile through time. In the proposed model here, it is assumed that the receiver nanomachine Q_R can sense and gather molecules from the environment, and that the sensed molecules can be at least τ, where $\tau \in \{0, 1, 2, .., h\}$. The environment may contain molecules considered to noise. In the first part of this work, the noise is neglected, while in the second part it is taken into account.

Thresholds: The vertical arrows in Figure are referring to the thresholds in each of the transmitter, receiver and the channel nodes. In the channel nodes Q_i, all molecular concentration above threshold h, will be distributed equally among its neighbours on both sides; the bold arrows in Figure 3.2 refer to the direction of the transmission of the molecules. For example, the exceeded molecule in node Q_2 would be transmitted to Q_1 and Q_3, the exceeded molecules in Q_4 would go to Q_3 and Q_R. However, for the receiver nanomachine Q_R the molecular concentration above threshold h would go to Q_4. The transmitter nanomachine Q_T has a different threshold X, which is higher than h; thus, Q_T would collect the surplus molecules from its neighbour, until it collects X molecules concentration. By then, it emitts this X concentration to its neighbour.

Every node Q_i in the network including Q_T and Q_R, can accept molecules up to the buffer B, where $B > X > h$.

Time Slot: The model is assumed to be a time slotted, with a slot duration t, where t is a system parameter and its length depends on the network geometric properties; thus, it can be $t = k\frac{d^2}{h}$, where k is a constant that can equal 1 and d is the distance between Q_T and Q_R. During each time slot, all nodes of Q_i and the receiver nanomachine Q_R inspect their molecules concentration compared to threshold h, while the transmitter nanomachine considers threshold X in comparing its molecules concentration.

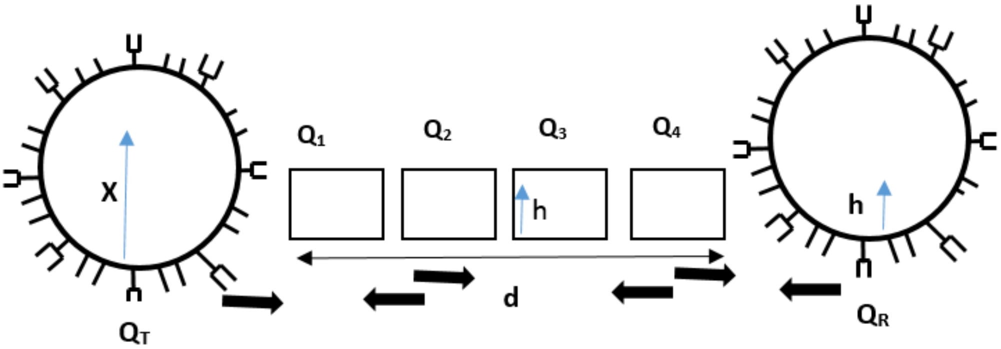


FIGURE 3.2: Graphical Depiction of the Proposed Model

Channel Nodes: As explained earlier, the channel nodes' Q_i mechanism is inspired by the *sandpile model*; when its molecular concentration exceeds the threshold h, it distributes the excess to its neighbour equally. This can be expressed in the following steps described in Algorithm 1; however, it is good to start by assuming that:

Q_i(t) is the level molecular concentration at time slot t, in node i.

e_i is the excess molecules from a node, where

$$e_i = Q_i(t) - h \tag{3.1}$$

Algorithm 1: Channel nodes

```
if $Q_i(t-1) > h$ then
    $Q_{i-1}(t) \leftarrow Q_{i-1}(t-1) + max\{e, 0\}/2$
    $Q_{i+1}(t) \leftarrow Q_{i+1}(t-1) + max\{e, 0\}/2$
    $Q_i(t) = Q_i(t-1) - e_i$
end
```

Floor and ceiling functions were utilized in the model verification experiment, in case the exceeded molecular concentration e_i was an odd number. The following chart in Figure 3.3 represents a part of the model experiment example, starting with the initial molecular concentration in three of the channel nodes Q_1, Q_2, and Q_3, during the first time slot t_1, with assumptions that threshold $h = 6$, and $n = 7$, (n represents the total number of channel nodes). The chart shows the changes in molecules concentration of these three nodes according to the above rules. In which node Q_2 transmits its exceeded molecules to the neighbours. The changes in molecular concentration can continue through the remaining time slots, with the existence of more channel nodes in between Q_T and Q_R.

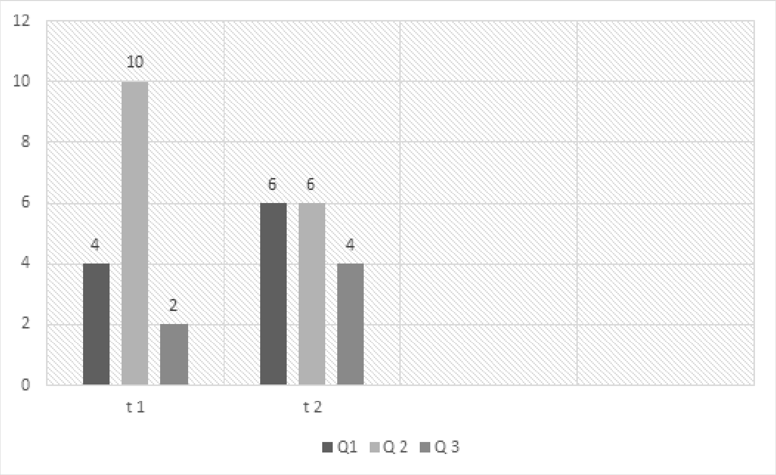


FIGURE 3.3: Molecular Concentration in Channel Nodes, h=6

Another example is shown in the following figures. Figures 3.4, 3.5 show one step of the process in the channel nodes:

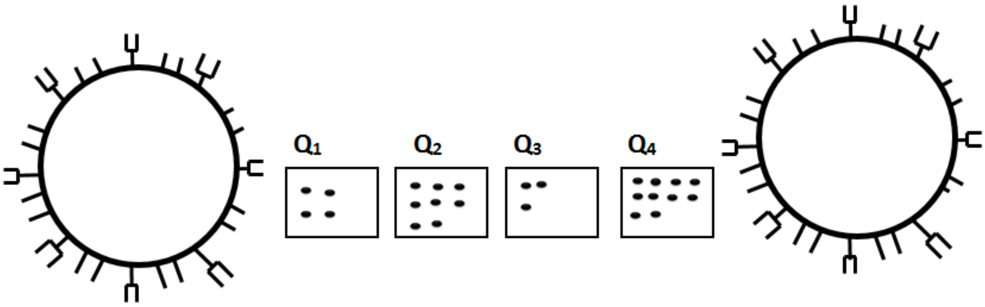


FIGURE 3.4: Initial molecular concentration in channel nodes, h=6

The example doesn't show the molecular concentration in Q_T and Q_R. As it mainly focus on showing how the channel nodes exchange molecular concentration, if their concentration is higher than threshold h. The molecular concentration of nodes Q_2 and Q_4 in Figure 3.4 is higher than 6 (which represents the value h). Thus, the exceeded molecular concentration in these nodes goes to their neighbours Q_1, Q_3, and Q_R as Figure 3.5 shows.

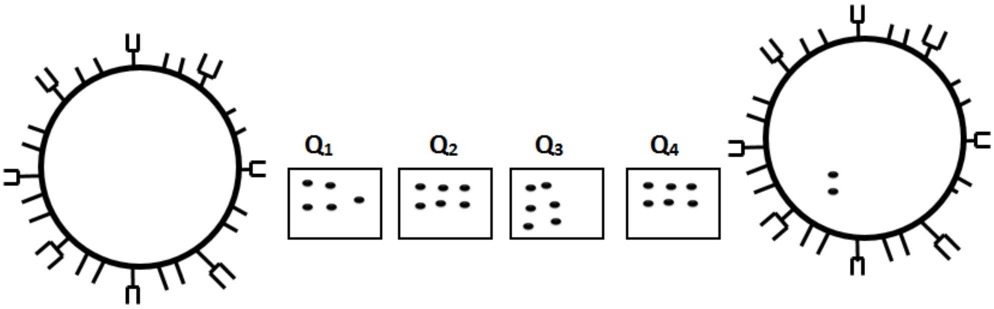


FIGURE 3.5: Molecules concentration in channel nodes after checking if it is above threshold h

Transmitter Nanomachine: The transmitter nanomachine can keep accepting excess molecules concentration from its neighbour (i.e., Q_1, the first node in the communication channel). This process is biologically justified, since cells usually do not pass through molecules into the environment one-by-one, but molecules can be grouped and enclosed in a vesicle (bi-lipid layer, spherical shaped container which comprises molecules) [100]. In diffusion based molecular communication, the arrival time of molecules from the propagation medium to the receiver nanomachine can be varied [131]; however, utilizing vesicles help in keeping all the transmitted molecules to reach the destination at the same time (vesicles are also assumed to be used with channel nodes and receiver nanomachine communications).

Assuming that $Q_T(t)$: is the amount of the transmitter molecules concentration at time slot t, and $e_T = Q_1(t-1) - h$, the receiver nanomachine mechanism can be represented as follows:

$$Q_T(t) = Q_T(t-1) + max\left\{e_T, 0\right\}/2 \tag{3.2}$$

The transmitter nanomachine Q_T attempts to collect X molecules concentration from its neighbour Q_1 surplus molecules. Then the transmitter nanomachine Q_T emits this X concentration to Q_1, as Algorithm 2 shows:

Algorithm 2: Transmitter nanomachine

```
if $Q_T(t-1) > X$ then
    $Q_T(t) \leftarrow Q_T(t-1) - X$
    $Q_1(t) \leftarrow Q_1(t-1) + X$
end
```

The transmitter nanomachine Q_T has its own mechanisms that are different from the channel nodes. After emitting X molecules concentration to Q_1, the transmitter nanomachine Q_T will stop receiving any surplus molecules from its neighbour Q_1. The purpose of Q_T pausing activity and not receiving molecules from a neighbour is to await the emitted X molecular concentration reaching to the receiver nanomachine Q_R, as these X molecular concentration is required to traverse through the channel nodes Q_i at the distance d between Q_T and Q_R, plus the time needed to get acknowledgement from the receiver nanomachine Q_R. Since the acknowledgement molecules should also traverse through the channel nodes Q_i in distance d between receiver Q_R and Transmitter Q_T nanomachines; thus, the total waiting time of Q_T is:

$$\text{Total waiting time of } Q_T \ = 2d \tag{3.3}$$

The process of waiting is also biologically motivated, as symbol duration defined in [99], in which a nanomachine can transmit molecules then wait for these molecules to be captured by the receiver nanomachine, before sending the next molecules. The distance d between the transmitter Q_T and receiver Q_R nanomachines is an unknown parameter; therefore, the transmitter nanomachine needs to estimate d. The molecules emitted by Q_T equal X; however, the receiver nanomachine Q_R might not receiver the entire amount of molecules transmitted by Q_T, since X is passing through the channel nodes Q_i, but it is likely to detect an increase in the average number of the received molecules from its neighbour Q_n (the last node of the channel). For simplicity, this increase is assumed to be at least h (the receiver detection process is described in the following section). Thus, the transmitter nanomachine Q_T can consider the higher amount of molecules concentration possibly received by Q_R from Q_n as h. The distance between the transmitter Q_T and receiver Q_R nanomachines can be estimated according to the following:

$$d \leq \log\left(\frac{X}{h}\right) \tag{3.4}$$

The transmitter nanomachine Q_T would wait for at least a $2d$ time duration, and after this duration has finished, Q_T returns to its activity of accepting excess molecules concentration from Q_1. However, the difference this time is that the transmitter nanomachine Q_T attempts to collect $(X+1)$ molecular concentration, then it would emit it to Q_1, and again, Q_T waits for a $2d$, (but here, the value of d shall be: $d \leq \log(\frac{X+1}{h})$). The transmitter nanomachine Q_T at each phase would attempt to increase the level of the collected molecules concentration until it reach to the point of collecting B molecular concentration. Collecting B molecular concentration which is the maximum capacity of each node reflects that the level of molecules concentration is high in the network, as a consequence of the molecules received from the environment by Q_R through out previous durations. Thus, the transmitter nanomachine Q_T emits this collected amount of B molecular concentration into the environment, so the process of communication is initiated again.

Receiver Nanomachine: The receiver nanomachine Q_R, as mentioned earlier, can sense and collect molecules from the environment, which helps to prevent the molecular concentration drain inside the system. The molecules propagating in the environment are following their spontaneous diffusion [18], which means, the time of their arrival at the receiver nanomachine Q_R can vary [131]. Thus, in this model it is assumed that Q_R can sense at least τ molecules concentration during time slot t, where $\tau <= h$, i.e., $\tau \in \{0, 1, 2, .., h\}$.

The receiver naonomachine Q_R has the same mechanisms as the channel nodes Q_i in dealing with the surplus molecules, but it emits all the excess concentration to its neighbour Q_n (the last node in the channel) in case it notifies (through a number of processes) that the transmitter nanomachine Q_T had sent its concentration (i.e., Q_T transmitted X molecules concentration). By assuming that $Q_R(t)$ is the amount of the receiver molecular concentration at time slot t, $e_R = Q_R(t-1) - h$, this process can be represented as shown in Algorithm 3 :

```
Algorithm 3: Receiver nanomachine
if $Q_R(t-1) > h$ then
    $Q_n(t) \leftarrow Q_n(t-1) + max\{e_R, 0\}/2$
end
```

However, the other difference is that Q_R monitors the amount of molecules it receives from its neighbour Q_n, by registering the received molecular concentration and comparing it to the next received concentration. If current received molecular concentration is higher than the previous one by at least h, then Q_R shall emit all its molecules concentration to Q_n. Here e_n is assumed to be: $e_n = Q_n(t-1) - h$, these mechanisms of the receiver nanomachine can be represented as follows:

$$Q_R(t) = Q_R(t-1) + max\{e_n, 0\}/2 \tag{3.5}$$

The receiver nanomachine Q_R considers the first received excess molecular concentration as max_R. Then, by the time Q_R receives a new excess concentration from Q_n, the new concentration is compared to max_R: if it is larger than max_R, it is considered as a new max_R. Then Q_R checks the incremental increase in the level of received concentration from Q_n, by finding the difference between the current max_R and the previous max_R. If the difference is high enough (i.e., greater or equal to h), then the receiver nanomachine Q_R assumes that the transmitter nanomachine Q_T had emitted its collected concentration to the channel nodes Q_i. Thus, it should send an *acknowledgement* to Q_T and that would be through transmitting all its molecular concentration $Q_R(t)$ to its neighbour Q_n. In order to allow the emitted concentration of Q_R to spread between the channel and potentially reach the transmitter nanomachine Q_T, the receiver nanomachine Q_R stops receiving any excess concentration from its neighbour for at least $\log \frac{Q_R(t)}{h}$. However, the receiver nanomachine continues to receive molecules from the environment. This process can be explained as follows:

The receiver nanomachine Q_R considers the first received amount of molecular concentration from its neighbour as max_R, as Equation (3.6) shows.

$$max_R = (Q_n(t) - h)/2 \tag{3.6}$$

Then, the process of comparing the received molecular concentration to checks the incremental increase in the level of received concentration from its neighbour Q_n. As Algorithm 4 shows:

Algorithm 4: Q_R checks if Q_T sent X molecular concentration

```
if $(Q_n(t+1) - h)/2 > max_R$ then
    $max_R \leftarrow (Q_n(t+1) - h)/2)$
    if $((Q_n(t+1) - h)/2) - ((Q_n(t) - h)/2) \geq h$ then
        $Q_n(t+1) = Q_R(t)$
    end
end
```

where $Q_n(t+1) = Q_R(t)$ means that molecular concentration Q_R at time t, will be pass to its neighbour Q_n in the next time slot $t+1$.

The transmitter nanomachine Q_T follows similar steps comparing the received molecules concentration from the neighbour node (obviously, after the waiting time is over) in order to recognize the acknowledgement from receiver nanomachine Q_R. Figure 3.6 shows the changes in molecules concentration of Q_R throughout the time slots. This is a part of experiment's results: the initial molecules concentration is 4, and it changes over time. These changes could be, due to receiving an excess concentration from the neighbour Q_n, or of molecules from the environment. The thresholds are h=6, X=12, and the maximum capacity of all nodes is B=20. The receiver nanomachine Q_R observed an increment in the received excess concentration during time slot 10; thus, it has emitted its entire molecular concentration to its neighbour as an *acknowledgement*, assuming that the transmitter nanomachine had sent X. In this work, PRISM verification tool is employed to make sure that the assumption of the receiver nanomachine Q_R was accurate.

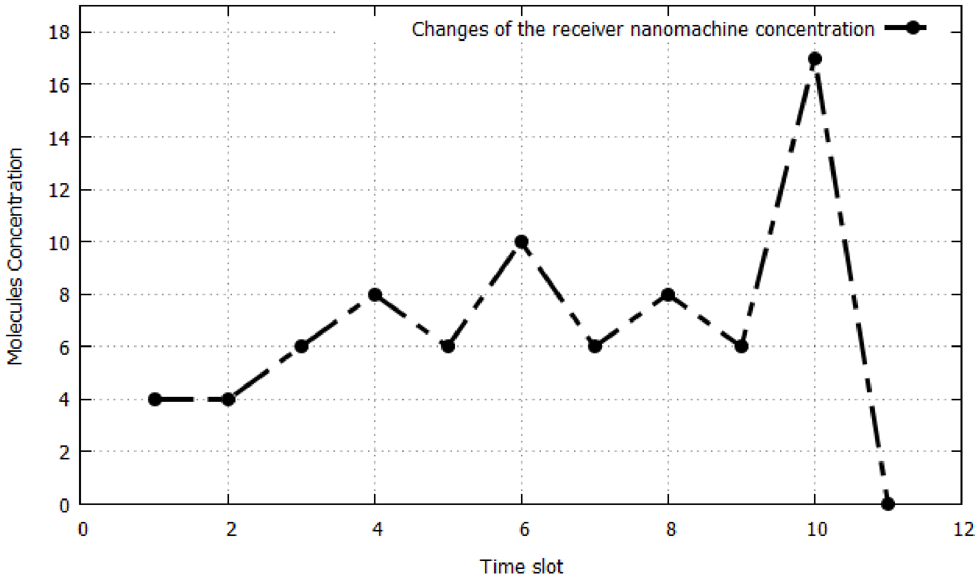


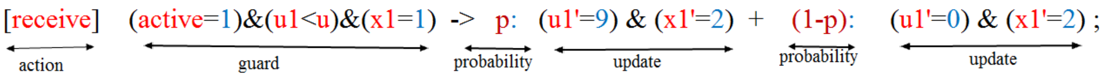
FIGURE 3.6: Changes of Q_R Molecular Concentration through Time Slots, h=6, B=20, X=12

3.2.2 PRISM Verification Tool

As defined earlier in Chapter 2, PRISM is a tool for formal modelling and analysis systems. The system behaviour is represented by constructing a mathematical model, then the model is analysed using formally-specified quantitative properties. These properties are described using temporal logic language [103]. It is possible to utilize PRSIM to reason, not only about the probability that a system's behaves in a specific manner but also about a vast domain of quantitative measures which are related to the system behaviour [33, 149].

The process of model checking consisting of two main phases, namely model constructing, model checking. The model construction phase is the process of converting the corresponding model into a PRISM language description. While model checking is processing/analysis of a constructed model by verifying a property specification and determining the result of that property [102]. Thus, to construct and analyse a model using PRISM, it should be specified in a state-based PRISM language first [103]. The PRISM language's basic components are *modules* and *variables.* PRISM model is composed of a number of *modules*, which can interact with each other. To implement the processes and actions of the module, a number of local *variables* are defined and utilized. The state of the module at any given time is determined by the values of these variables. Thus, the local state of all modules constitute *the global state* of the whole model [149].

A set of *commands* are used to describe each module behaviour, by specifying the changes which can occur in a particular state, with probabilities that are assigned to the corresponding change; PRISM commands that represent some actions can optionally be labelled [33]. The following is an example of a command written in PRSIM language:



```
[receive]  (active=1)&(u1<u)&(x1=1)  ->  p:  (u1'=9) & (x1'=2)  +  (1-p):  (u1'=0) & (x1'=2) ;
```

Each guard represents a predicate over all variables in the model, including the variables of each module. An update of the command describes a transition that the model can make if the guard is true. A transition represents the change in a particular state, and it is identified through giving the new values of the variables in the module [149].

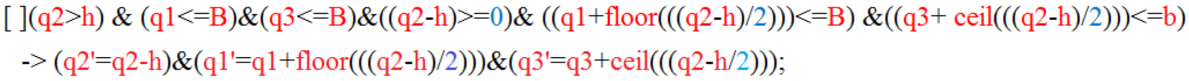
In PRISM model, each command can be executed independently, though in every state of the model there is a set of commands (in any of the modules) indicating that their guards are true can be elicited. Thus, choosing which command is performed can depend on the type of the model. However, there is another feature in PRISM language which allow synchronizations in executing transitions (commands) on more than one module of PRISM model. That can be done by using *labels*, i.e. all commands with matching action-labels can be executed simultaneously [33, 149].

3.2.3 Verification Results using PRISM

The model explained in Section 3.2.1 has been described in PRISM language to construct a PRISM model. This model consists of three modules: the first represents the rules related to the transmitter nanomachine Q_T. The performance of channel nodes Q_i is described in the second module; and in the third module the mechanisms of the receiver nanomachine Q_R are interpreted in PRISM language. The values of the related thresholds B, h, and X are defined in the PRISM model and considered as constants. Besides this, the initial concentrations of all nodes (including the transmitter and receiver nanomachines) are defined as integer variables. The system is verified using the PRISM model checking tool, PRISM 4.2.1 running on MS Windows 7, Intel(R) Core(TM)i3-2370M CPU 2.40GHz, 4.00 GB of RAM.

The model in Section 3.2.1 is represented as a Discrete-Time Markov Chains (DTMC) model. DTMC is a state-transition system consisting of a discrete set of states which represent the possible configurations of the system being modelled, and the transitions between states occur in discrete time-steps [101]. DTMC characterize each transition with a probability, in a way that means the sum of every outgoing transitions for each state equals one, it is possible to set a probability space through infinite paths over the model and quantitatively analyse the eventuality of the occurrence of a certain event [149].

However, the probabilities of the used transitions in representing this model are one, i.e. the transitions are deterministic, if the conditions are true, then a certain event will occur, as the following channel related command indicates:



```
[ ](q2>h) & (q1<=B)&(q3<=B)&((q2-h)>=0)& ((q1+floor(((q2-h)/2)))<=B) &((q3+ ceil(((q2-h)/2)))<=b)
 -> (q2'=q2-h)&(q1'=q1+floor(((q2-h)/2)))&(q3'=q3+ceil(((q2-h/2)));
```

This command in the channel module is about the second node Q_2 of the channel. The end conditions are: if the molecules concentration of Q_2 is higher than threshold h, and the molecular concentration in nodes Q_1 and; Q_3 are less than the capacity B, and if the molecular concentration of Q_2 remains larger or equal to zero after emitting the excess concentration, and if the molecules concentration in nodes Q_1, and Q_3 remain less than B after adding the excess concentration from Q_2¿ Then this command is enabled, and the following events can occur. These events are as follows: the excess concentration is deducted from the Q_2 molecular concentration; and each of the nodes Q_1 and; Q_3 get half of this, which is an example of the commands used to represent the model in PRISM language. The commands in transmitter module and receiver module are regulated in similar way (demonstrated in the above example) and according to the conditions explained in Section 3.2.1. However, flags and labels are utilized in the modules, to point out to the receiver nanomachine that the transmitter had already emitted X molecular concentration. Thus, when it senses an increment in the level of received molecular concentration from the neighbour, it transmits all its molecular concentration. This rise could be a consequence of the emitted X molecular concentration by the transmitter. These flags and labels can also indicate to the transmitter nanomachine that the receiver has already sent an acknowledgement and it can start receiving molecular concentration from its neighbour. Furthermore, the flags have been employed in the model analysing process, i.e., property verification.

Thus, to analyse the model that has been described and constructed in PRISM it is required to identify one or more properties related to the model which can be evaluated and analysed by the tool [149]. In PRISM the *property specification language* is based on probabilistic temporal logic [101]. In assigning properties to a model, one of the main tasks is to identify a specific set of the model's states; as an example, in order to verify if 'an algorithm eventually terminated successfully with probability 1', it is required to identify the states of the model which can identify the status in which 'the algorithm has terminated successfully' [149]. This can be achieved through writing an expression in the PRISM language containing references to variables (and constants) from the model that it relates. The expression evaluates a Boolean value, when the set of states corresponding to the expression evaluates as true, this means the expression is 'satisfied' in these states [102].

In PRISM property specification language, one of the most important operators is the **P** operator, which can be used to reason about the probability of an event's occurrence [149]. In PRISM properties verification, a wide range of paths properties are used with the **P** operator and, where path property is a formula that evaluates either true or false for a specific path in a model [103]. There are different types of temporal operator which can be used inside the **P** operator, one of which is 'Eventual path property': **F** *prop*. The property **F** *prop* can be true for a specific path if the prop eventually becomes true at some point along the path [149].

The construction of the model according to PRISM language and verification analysis processes have been carried out, first on a model consisting of two channel nodes with transmitter and receiver nanomachines. The values of thresholds are: $B = 15$, $h = 5$, $X = 9$, while the initial values of the molecules concentration are: $\{8\}$ in the transmitter nanomachine, $\{6\}$ in the receiver nanomachine and $\{4, 3\}$ in the channel nodes. Then, the verification process has been repeated on a model of five channel nodes, with the same values of the thresholds and same initial molecular concentration of the transmitter and receiver nanomachines, but the initial molecules concentration of the channel nodes are: $\{4, 3, 8, 7, 4\}$. Then another experiment is repeated on a model of ten channel nodes, where the initial molecules concentration in these channels nodes are $\{4, 3, 8, 7, 4, 9, 4, 7, 3, 10\}$. Finally, an experiment on a model of twenty channel nodes is implemented, the initial molecules concentration of the channel nodes are: $\{4, 3, 8, 7, 4, 9, 4, 7, 3, 10, 5, 11, 4, 9, 13, 6, 8, 10, 5, 7\}$. The threshold values and the initial molecules concentration in the transmitter and receiver nanomachines of the last two experiments are the same values in the first and second experiments. Table 3.1 represents the results of building these four PRISM models:

Number of Channel Nodes	2	5	10	20
Reachability Iterations	19	48	68	95
Time of Model Construction	0.122	11.93	425.412	6373.23
No. of States	1136	125399	10296733	207448902
No. of Transitions	2530	222490	40140786	516044219

Table 3.1: Outputs after Building PRISM Models

The results in Table 3.1 are computed within PRISM model checker. These results are shown if the model built correctly. Each module in PRISM consists of two parts: variables and commands. The variables describe the possible states that the module can be in; the commands describe the way in which the state changes over time. Transition is the change in particular state, and that happens through giving new values of the variables in the module. Thus, according to the number of modules in PRISM model, the number of states and transitions are computed. Linear increase in size of system can result in an exponential increase in the size of the model.

The properties which have been verified are related to receiving molecular concentration by Q_R when Q_T emits its X molecular concentration, and the receiving of an acknowledgement by Q_T after Q_R emits all its molecules concentration. Thus, some flags are used to refer to these events, in order to set the properties' expressions. The first flag is $sent_X$ which is related to Q_T, initially this flag is *false*, and when Q_T emits X molecular concentration $sent_X$ becomes *true*. Two parameters $maxR_{prv}$ and $maxR_{new}$ have been defined, which represent the previous and current received molecules concentration by Q_R. **If** $(maxR_{new} - maxR_{prv} \geq h)$ & $(sent_X = \text{true})$ **then** $Q_R = 0$, i.e. the receiver nanomachine Q_R emits all its molecular concentration when these conditions are enabled, and through this it is possible to verify this property thus:

$$\mathbf{P} =?[\mathbf{F}(Q_R = 0)] \tag{3.7}$$

Formula (3.7), when evaluated over the model, yields the probability that the molecular concentration of the receiver nanomachine Q_R equals zero, *eventually*. This property has been verified in the four models described above, as Figure 3.7 shows:

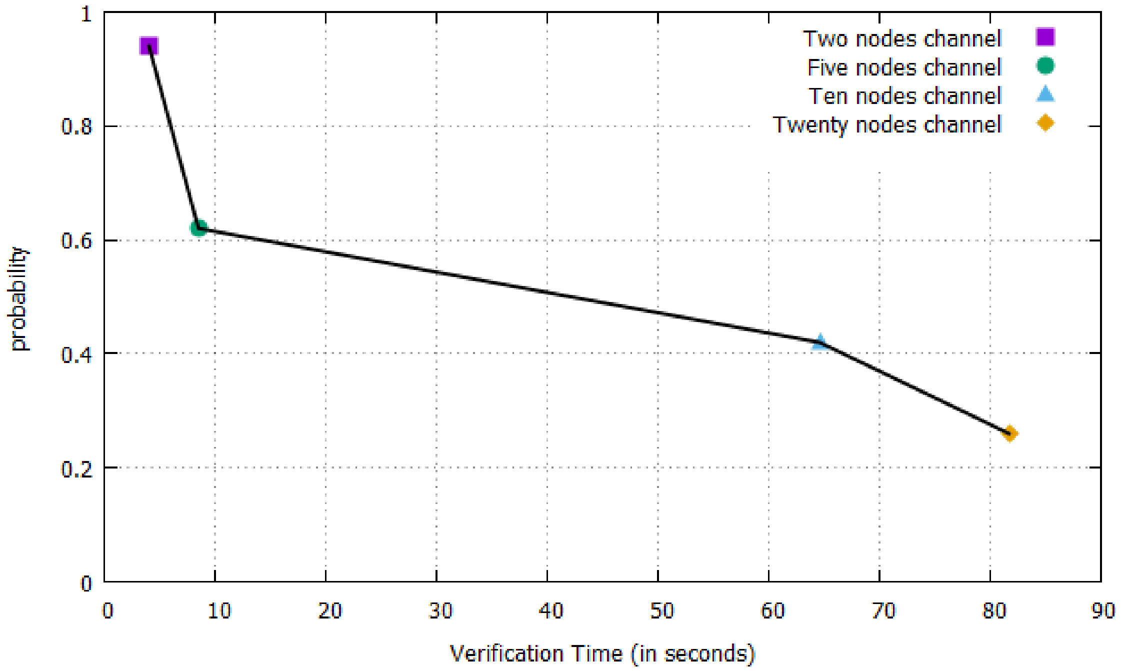


FIGURE 3.7: Verification Results of Q_R Sensing that Q_T Emitted X Molecules Concentration

The x-axis in Figure 3.7, (Verification Time) represents the time required for model checking, i.e., the time that PRISM took to verify property 3.7, while the y-axis signifies the probability that this property is (eventually) *true* in the model. The points in the figure denote the size of the modelled system (the number of the channel nodes between Q_T and Q_R). Figure 3.7 shows that the probability of Q_R detecting that Q_T has emitted molecular concentration is inversely proportional to the size of the channel between Q_T and Q_R. This is expected since the probability of molecules reaching Q_R from Q_T decreases significantly as the distance between transmitter and receiver increases. However, the figure shows that the time of model checking increasing as the size of the channel nodes increases. As the receiver nanomachine Q_R can sense and collect molecules from the environment. These molecules are supposed to follow spontaneous diffusion, and thus, the time of their arrival at the receiver nanomachine Q_R can vary. These molecules are defined as a random variable in the PRISM model as a random variable. This variable is affecting the deterministic process described in Section 3.2. To explain the precise meaning of the data points in Figure 3.7, here is a clarification example: In a network of five channel nodes the probability of verifying property (3.7) is 0.62. The time needed required to verify property (3.7) in a network of five channel nodes is 8.7 second. In Figure 3.7 the small square represents the result of the experiment with two nodes channel. The circle however represents the result of the experiment with five nodes channel. While the triangle represents the result of the experiment with ten channel nodes. Finally, the diamond shape represents the result of the experiment with twenty nodes channel.

When $Q_R = 0$, this enable the flag $sent_{all}$ to become *true* (which initially was *false*). This flag along with th two parameters $maxT_{prv}$, $maxT_{new}$ that represent the previous and current received molecules concentration by Q_T, can be used to find out if Q_T recognizes that Q_R has sent an acknowledgement, i.e. if flag *ack* becomes *true*: **If** $(maxT_{new} - maxT_{prv} \geq h)$ & $(sent_{all} = \text{true})$ **then** $ack = true$. Through these conditions, it is possible to verify the following property:

$$\mathbf{P} = ?[\mathbf{F}(ack = true)] \tag{3.8}$$

Formula (3.8), when checked over the model, gives the probability that the transmitter nanomachine Q_T receives an acknowledgement *eventually*, as flag *ack* becomes *true*. This property has been verified in four experiments with different sized systems, having two, five, ten and twenty channel nodes, respectively, as Figure 3.8 shows:

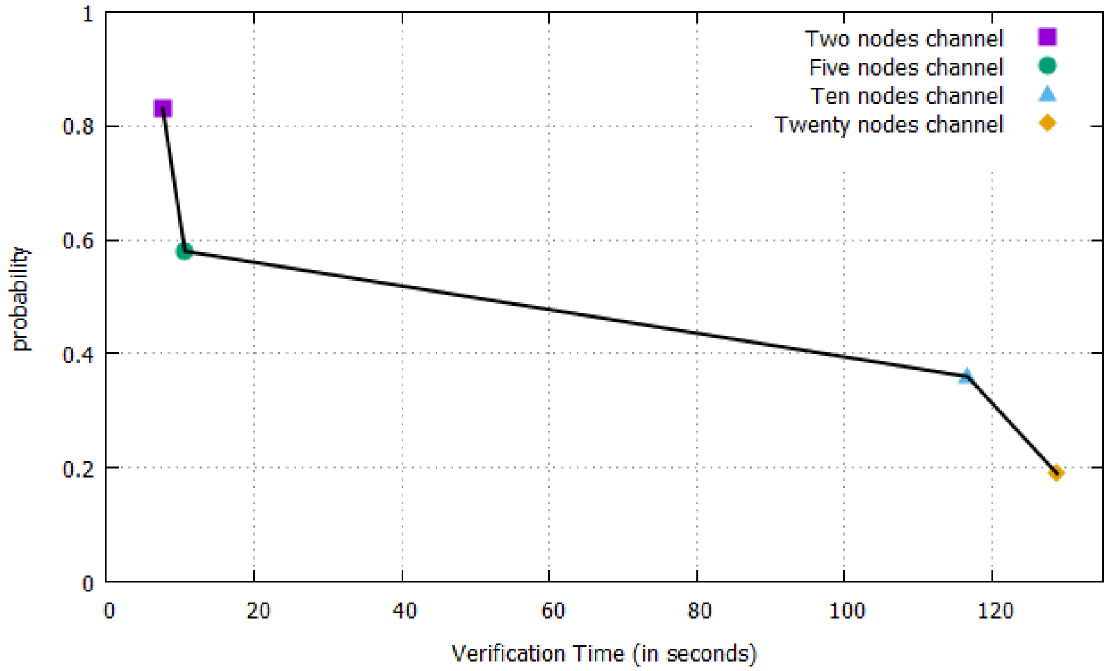


FIGURE 3.8: Verification Results of Q_T Receiving Acknowledgement from Q_R

Figure 3.8 shows that the probability of Q_T receiving an acknowledgement from Q_R is similar to the previous figure, the x-axis representing the required time to verify property 3.8, and the y-axis representing the probability that the flag *ack eventually* equals *true*, and the points denote the number of the channel nodes of the verified models. Figure 3.8 shows that the probability of Q_T receiving an acknowledgement is also inversely proportional to the size of the channel, as the distance between Q_T and Q_R increases. However, in term of the probabilities of each channel size in Figure 3.8 compared to the corresponding size in Figure 3.7, it is noticeable that the probability of receiving acknowledgement is always less than the probability of Q_R detecting that Q_T has sent the molecular concentration. This could occur because the number of molecular concentration emitted by Q_T is often higher than those which Q_R transmits as an acknowledgement.

The following sections explore the effect of noise on the proposed system, then assuming that the system is a bi-directional model in a way that Q_T and Q_R can both transmit and receive molecular concentration; and after this representing multiple access channel model.

3.3 One-Dimensional Bio-Inspired Nano System with Noise

The same model described in subsection 3.2.1 is assumed here, but noise of the environment is considered. It is thought that the receiver nanomachine Q_R can sense at least τ molecules during time slot t, where $\tau < h$, i.e., $\tau \in \{0, 1, 2, .., h\}$, considering that Q_R is able to receive molecules from other nodes in the environment. Here, it is assumed that the channel nodes Q_i can also sense molecules from the environment, but what Q_i senses is considered as noise, since Q_i nodes are supposed to receive molecular concentration only from Q_T and Q_R. Each one of the channel nodes Q_i can sense μ molecular concentration during time slot t, where $\mu \in \{0, 1, 2, .., h\}$. This affects the mechanism of channel nodes Q_i in dealing with its excess molecular concentration e_i. Instead of deterministically dividing it into two equal parts and emitting it to its neighbours, the channel nodes distribute its excess randomly, it may emit all its excess molecular concentration to one neighbour rather than another, as explained below in Algorithm 5:

Let Q_i(t)be the amount of node i molecular concentration at time slot t and e_i be the excess molecules from a node and is obtained from Equation 3.1.

Algorithm 5: Effect of noise on channel nodes mechanism

```
if $Q_i(t-1) > h$ then
    p: $[Q_{i-1}(t) \leftarrow Q_{i-1}(t-1) + e]$ + (1-p):$[Q_{i+1}(t) \leftarrow Q_{i+1}(t-1) + e]$
end
```

In order to verify the system with the assumptions of noise effects on the channel nodes, the PRISM representation discussed in Subsection 3.2.3 was repeated, with changes on the channel module, and by making the transitions based on the probabilities, in this representation, it has been assumed that the excess concentration of a certain node in Q_i can be emitted to Q_{i+1} with 0.5 probability and 0.5 to be emitted to Q_{i-1}. Four experiments have been carried out, based on different sized channel nodes: (two, five, ten and twenty), with the same initial values of molecular concentration that set in Subsection 3.2.3, and the same threshold values of (B, h, X). Table 3.2 represents the results of building these four PRISM models taking in consideration noise sensed by the channel nodes Q_i:

Number of Channel Nodes	2	5	10	20
Reachability Iterations	75	110	150	185
Time of Model Construction	1.39	35.97	471.32	47122.15
No. of States	109003	2051359	24173211	531722815
No. of Transitions	320837	4875150	42151022	725631920

Table 3.2: Outputs after Building PRISM Models with Noise

Besides verifying properties (3.7) and (3.8), in this experiment new properties have been verified, which are related to errors in receiving molecules from Q_T or in receiving acknowledgement form Q_R. Errors are a consequence of receiving higher molecules concentration either by Q_R or Q_T, even though flags $Sent_X$ and $Sent_{all}$ respectively are *false*. This increment in molecular concentration is due to the received molecular concentration from the environment by the channel nodes during each time slot. Thus, the error at the receiver nanomachine is the result of: **If** $(maxR_{new} - maxR_{prv} \geq h)$ & $(sent_X = \text{false})$ **then** $error_R = true$, where $error_R$ is a flag indicating that the increment of the molecular concentration level received by Q_R wasn't due to the transmitting of molecules concentration by Q_T, and initially $error_R$ is *false*. The property to be verified is represented as the following:

$$\mathbf{P} = ?[\mathbf{F}(error_R = true)] \tag{3.9}$$

Evaluating Property (3.9) over the model yields the probability that the Q_R senses an increment in the regular received molecular concentration, even if Q_T has not emitted X molecules. Both Properties (3.9) and (3.7) were verified in the four models, as described in Subsection 3.2.3 and showen in Figure 3.9:

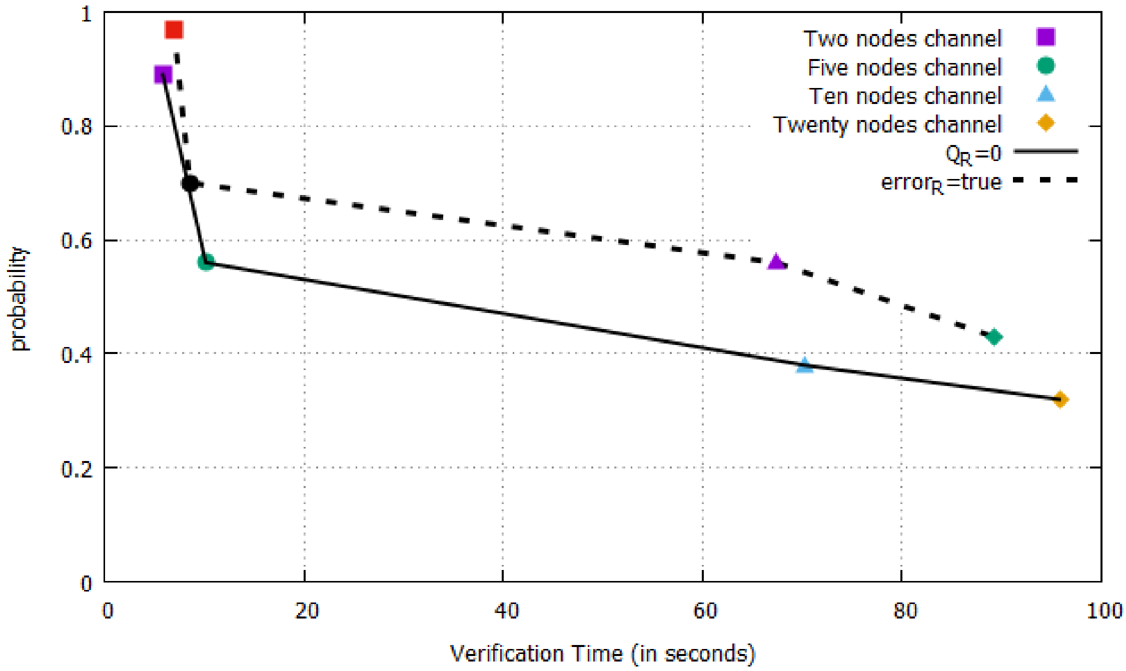


FIGURE 3.9: Verification Results of Properties (3.9) and (3.7)

The x-axis in Figure 3.9 signifies the time required to verify properties (3.9) and (3.7); the y-axis represents the probability that these properties are (eventually) *true* in the model. The points on the line denote the probability of occurring property (3.7); the points on the dashed line represent the probability of occurring property 3.9, the points represent the number of nodes between Q_T and Q_R. It can be seen that the probability of both properties are inversely proportional to the size of the channel between Q_T and Q_R, as the distance between the Q_T and Q_R expands. However, the probability of verifying Property (3.9) is higher. That could be a consequence of Q_R receiving an increased level of molecular concentration even before Q_T transmits X molecules concentration. As the channel nodes Q_i receive at least μ molecular concentration from the environment during each time slot and their concentration exceeds threshold h many times.

The second error $error_T$ is related to receiving an acknowledgement even though $sent_{all} \neq true$. **If** $(maxT_{new} - maxT_{prv} \geq h)$ & $(sent_{all} = \text{false})$ **then** $error_T = true$, initially $error_T$ is *false*. The property to be verified is represented as follows:

$$\mathbf{P} =?[\mathbf{F}(error_T = true)] \tag{3.10}$$

Figure 3.10 shows the verification probability of two properties, 3.8 and 3.10, which are related to receiving acknowledgement by Q_T from Q_R, and the error in receiving this acknowledgement. These properties have been verified in four experiments on the models described in subsection 3.2.3.

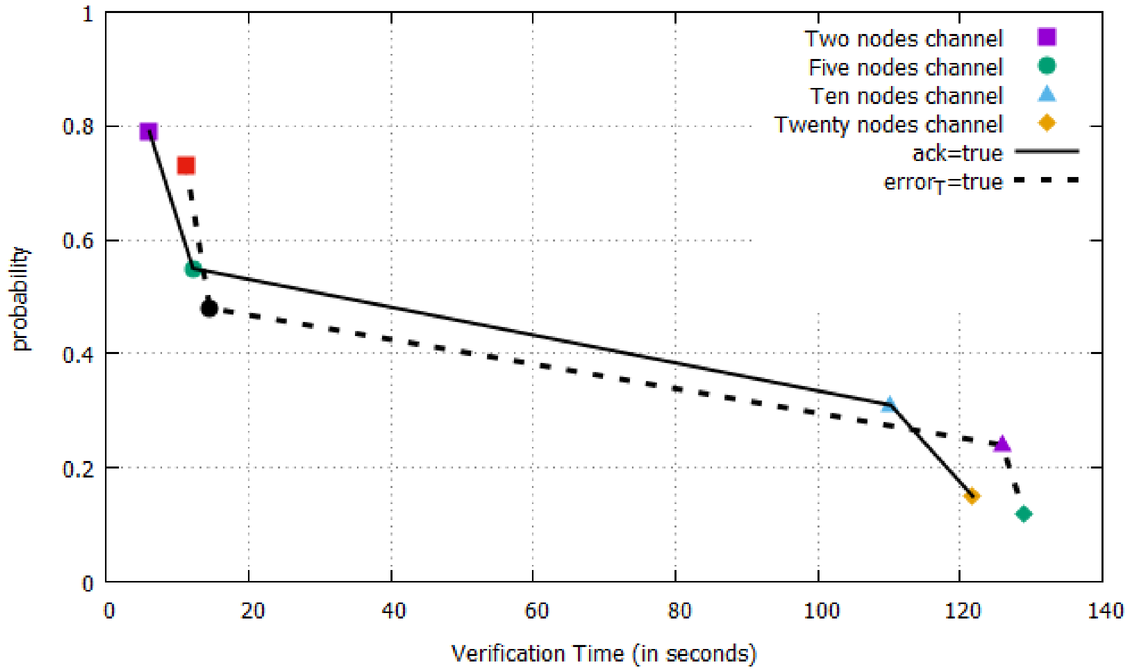


FIGURE 3.10: Verification Results of properties (3.10) and (3.8)

The points on the line in Figure 3.10 denote the probability of the occurring of Property (3.8), and the points on the dashed line represent the probability of the occurring of Property (3.10), each point represents a specific channel size. The probability of verifying property 3.10 is not high as of Property (3.9) specified in previous figure, that could be a consequence of Q_T waiting phase and not receiving excess concentration from neighbour after transmitting X; however, Q_T might sense an increment in the level of received molecules concentration from neighbour even before transmitting X, and obviously $Sent_{all}$ is not true yet, and that what can trigger Property (3.10).

3.4 Bidirectional Bio-Inspired Nano System

The model described in Section 3.2 is considered, but with a few changes and new rules, in order to assume that the model is bidirectional, inspired by rules and conditions characterizing the model in [177], gives the following:

- **Q_T and Q_R**

  In this section, both Q_T and Q_R are assumed to have two modes: a transmission mode and a receiving mode. However, a nanomachine can not be in a transmission and receiving mode simultaneously. If, for example, Q_R is in transmission mode, it cannot detect any transmitted molecular concentration. During each time slot t, both Q_T and Q_R can receive at least τ molecular concentration from the environment, where $\tau < h$ and t has been defines in Subsection 3.2.1. Each nanomachine in the model has its own timer or a local clock t_{local} and according to it a nanomachine switches between transmission and receiving mode. During each time t_{local} a nanomachine can transmit molecules concentration u, where $u < h$ with probability $p = \frac{1}{m}$, where m represents the number of nanomachines in the model. Here m is 2, i.e. Q_T and Q_R; thus, $t_{local} = \log(\frac{u}{h})$. However, for each local clock a nanomachine either listens till the end of this t_{local}, or transmits u molecular concentration at the beginning, and then it waits until the end of this t_{local}. The start of the duration of t_{local} in Q_T is not synchronized with the start of t_{local} in Q_R. When a nanomachine transmits u molecular concentration, there is a probability ε of failure, where $\varepsilon > 0$. Thus, when Q_T is in transmission mode and sends u molecular concentration through the channel nodes Q_i, then Q_R receives u with a probability of at least $1-\varepsilon$. In this model, it is assumed that nanomachines can transmit and receive molecules concentration, but sending acknowledgement is not considered here; however, verification of successful transmission and receiving has been carried out by employing PRISM model checker.

- **Channel nodes** Q_i

  Channel nodes Q_i have the similar mechanisms to those described in Section 3.2.1; however, there are few changes. Firstly, it is assumed that the initial value of molecular concentration in all Q_i is h. So that all the excess molecules concentration would be sent to the next neighbour, **if** $Q_i(t) > h$ **Then** $Q_{i+1}(t) = Q_{i+1}(t-1)+e$, where e is obtained through Equation (3.1), (e here represents u). Thus, Q_{i+1} transmits e to Q_{i+2}, and that continues until u reaches Q_R, in case Q_T had transmitted u at the beginning of its t_{local}. If both Q_T and Q_R were in transmission mode and did not detect molecular concentration from Q_i nodes. Then, either Q_1 (the first node of the channel near Q_T) or Q_n (the last node of the channel near Q_R) transmits e into the environment, so that all channel nodes return to their stable condition with h molecules concentration. Some nodes Q_i might receive e from their neighbours on both sides during short duration, which can happen when Q_T and Q_R are in transmission mode within a close period.

- **Model Verification**

  A PRISM model has been constructed according to the rules described in this section; thus, with probability 0.5, a nanomachine can be in transmission mode, otherwise it is in receiving mode, with 0.5 probability. Transmission mode means that a nanomachine would transmit a certain value, i.e. u, to its neighbour from the channel nodes Q_i (either Q_1 or Q_n), then the channel nodes would deliver it to the destination nanomachine. In receiving mode, a nanomachine waits to receive something from its neighbour. Few flags have been defined to facilitate the verification process. Two experiments have been implemented on a model with a channel of two nodes, and model with a channel of five nodes. The initial values of molecular concentration in all the channel nodes are h. The values of the thresholds in both experiments are $B = 15$, $h = 5$, the initial values of the molecules concentration are: 8 in Q_T and 6 in Q_R, and the value of $u = 3$. Table 3.3 represents the results of building these two PRISM models:

Number of Channel Nodes	2	5
Reachability Iterations	64	98
Time of Model Construction	5.72	71.23
No. of States	171541	2516220
No. of Transitions	311231	4984325

Table 3.3: Outputs after Building PRISM Models of Bi-directional System

Four properties have been verified that are related to the *success* of transition/receiving and the *failure* of transition/receiving. When either nanomachine in the mode is in a transmission mode, then a flag is enabled, i.e. $Q_T send = true$, $Q_R send = true$. When either nanomachine is in receiving mode, then other flags are enabled, i.e. $Q_T receive = true$, $Q_R receive = true$. The property **Send-Success** is true:

If $(Q_T send = true)$ & $(Q_R receive = true)$ & $(Q_1 = (Q_T - u) + Q_1)$ $\|$ $(Q_R send = true)$ & $(Q_T receive = true)$ & $(Q_n = (Q_R - u) + Q_n)$

This means that successful transmission requires that one of the nanomachines is in transmission mode, and the other in receiving mode, and that the nanomachine transmits u molecular concentration to its neighbour. If one of these conditions has not been satisfied, then the **Send-Fail** property is not enabled.

The property **Receive-Success** would be true:

If $(Q_T receive = true)$ & $(Q_R send = true)$ & $(Q_T = (Q_1 - h) + Q_1)$ $\|$ $(Q_R receive = true)$ & $(Q_T send = true)$ & $(Q_R = (Q_n - h) + Q_R)$

If one of these conditions is not satisfied, then the **Receive-Fail** property is not enabled.

Thus, the properties to be verified are:

$$\mathbf{P} =?[\mathbf{F}(Send - Success = true)] \tag{3.11}$$

$$\mathbf{P} =?[\mathbf{F}(Send - Fail = true)] \tag{3.12}$$

$$\mathbf{P} =?[\mathbf{F}(Receive - Success = true)] \tag{3.13}$$

$$\mathbf{P} =?[\mathbf{F}(Receive - Fail = true)] \tag{3.14}$$

The following figures show the results of verifying these properties:

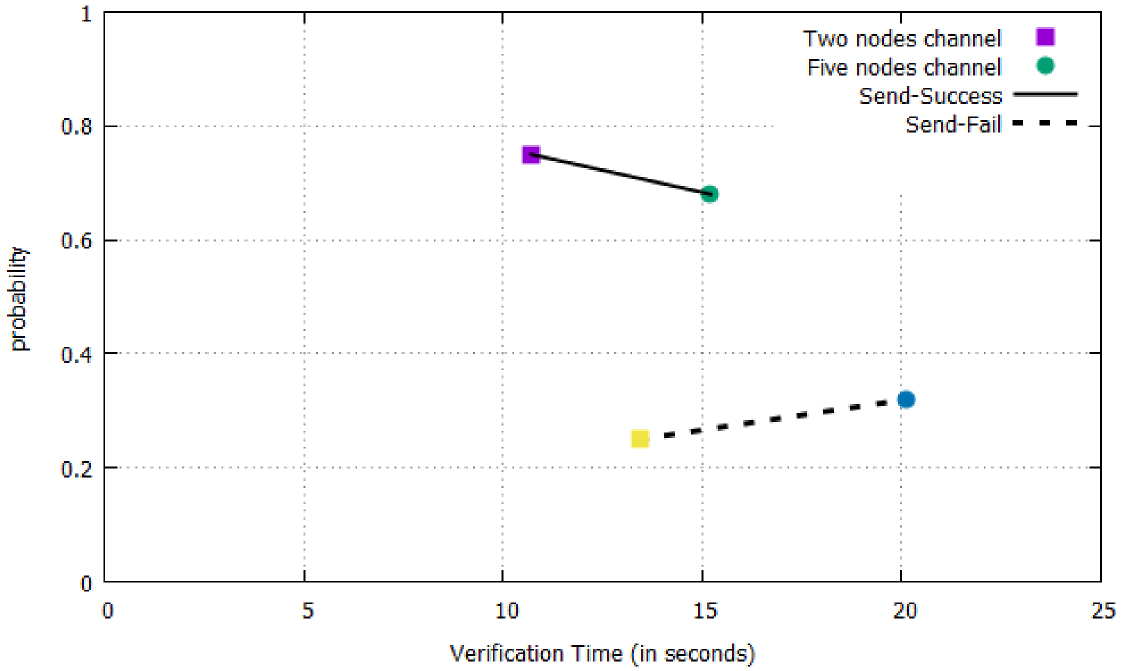


FIGURE 3.11: Verification Results of Properties (3.11) and (3.12)

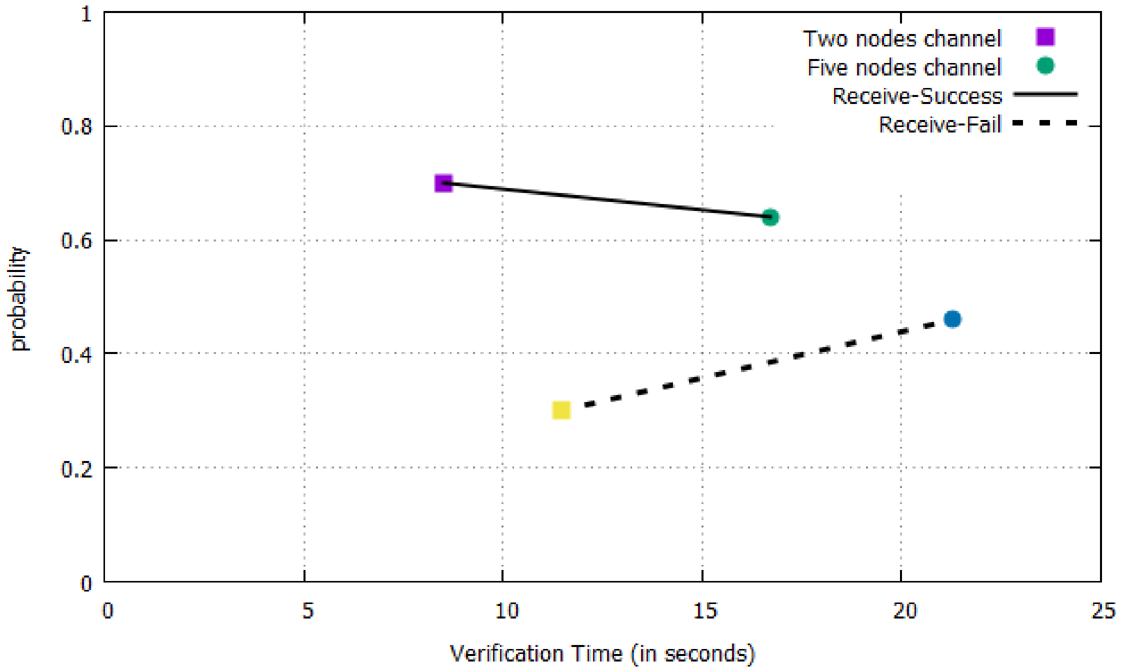


FIGURE 3.12: Verification Results of Properties (3.13) and (3.14)

Figures 3.11 and 3.12 show that the probability of success in sending and receiving is higher than the probability of failure, which is likely to be related to the simple model that has been verified, which consists of only two nanomachines; and, as the figures show that the effects of the channel size are not very high. This might be a result of the channel nodes rules in this model, in which it delivers the total amount of the molecules concentration (i.e. u) to the destination without many effects on this amount. However, if the size of the channel is bigger, one of the effects on transmission/receiving failure or success is that a destination nanomachine is in a receiving mode, but until u approaches from the transmitter through the channel nodes, its t_{local} ends and it might switches to a transmission mode, and thus be unable to detect u.

3.4.1 Multiple-Access Channel System

Applying the same rules discussed in relation to the bi-directional model in Section 3.4, with few changes, in a system of four nanomachines, each one can transmit and receive molecules concentration, as Figure 3.13 shows. Thus, the difference here is that, during each time t_{local}, a nanomachine can transmit molecular concentration u, with probability $p = \frac{1}{4}$. A nanomachine in transmission mode can send molecular concentration to a neighbour from one side, or to neighbours on both sides (i.e., sends u twice in the same t_{local}). However, a nanomachine in receiving mode can not receive molecular concentration sent from neighbours on both sides (i.e. two nanomachines transmit u to the same destination nanomachine in quite close duration) due to jamming in communication, which can be considered as another reason for transmitting/receiving failure.

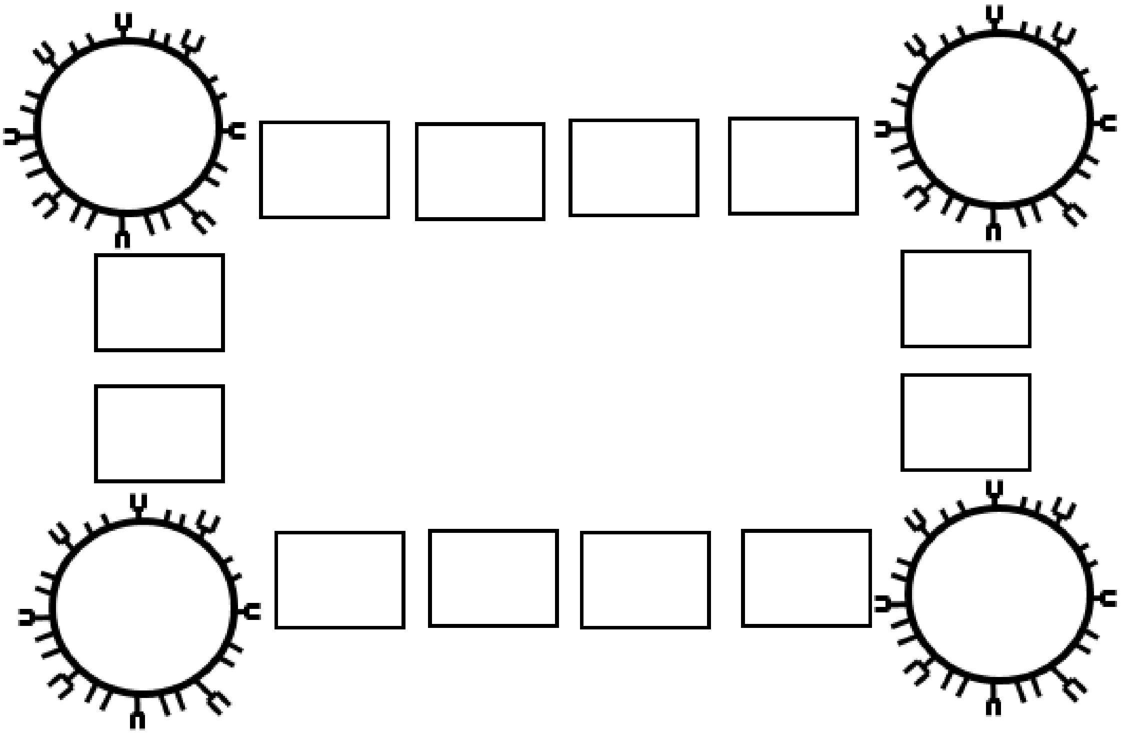


FIGURE 3.13: Multiple-Access Channel System

The constructed PRISM model in previous section has been extended and modified according to the rules of multi-access channel; thus, with a probability of 0.25 a nanomachine can be in transmission mode, otherwise it is in receiving mode with 0.25

probability. Transmission mode means that a nanomachine transmits a certain value, i.e., u, to one of its neighbours from the channel nodes, or to its two neighbours (on both sides, if it needs to communicate with nanomachines on its sides), then the channel nodes deliver it to the destination(s) nanomachine. Receiving mode, a nanomachine waits to receive something from either of its neighbours, and it simply cannot receive molecular concentration from both neighbours. Two experiments have been implemented on a model with a channel of two nodes between each two nanomachines, and a model with a channel of five nodes between each two nanomachines. The initial values of molecules concentration in all the channel nodes are h. The value of thresholds in both experiments were: $B = 15$, $h = 5$; the initial value of the molecular concentration in the four nanomachines were: $8, 6, 7, 9$; and the value of $u = 3$. Table 3.4 represents the results of building these two PRISM models:

Number of Channel Nodes	2	5
Reachability Iterations	82	121
Time of Model Construction	9.54	83.25
No. of States	2354211	31435212
No. of Transitions	443112	6571213

Table 3.4: Outputs after Building PRISM Models of Multi-Access Channel System

Four properties have been verified through these two experiments; these properties have already described in Section 3.4. The following figures show the results of verifying (3.11), (3.12), (3.13) and (3.14) properties:

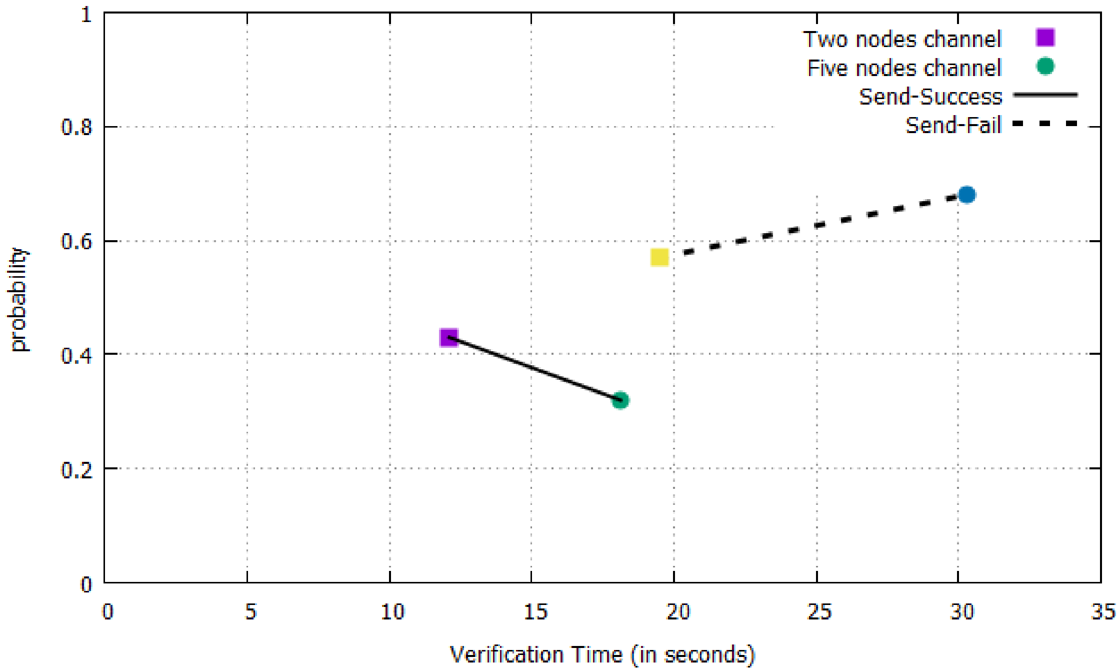


Figure 3.14: Verification Results of Properties (3.11) and (3.12) in Multi-Access Channel

Figure 3.14 represents the results of verifying these two properties (3.11) and (3.12). Where the dotted line represents the probability of fail in sending (i.e., result of verifying Property (3.12)) in two experiments with different sized network. Where the square represents the experiment with two nodes channel network and the circle represents the experiment with five nodes channel. While the line in the Figure 3.14 represents the probability of success in sending (i.e., result of verifying Property (3.11)) in two experiments with different sized network.

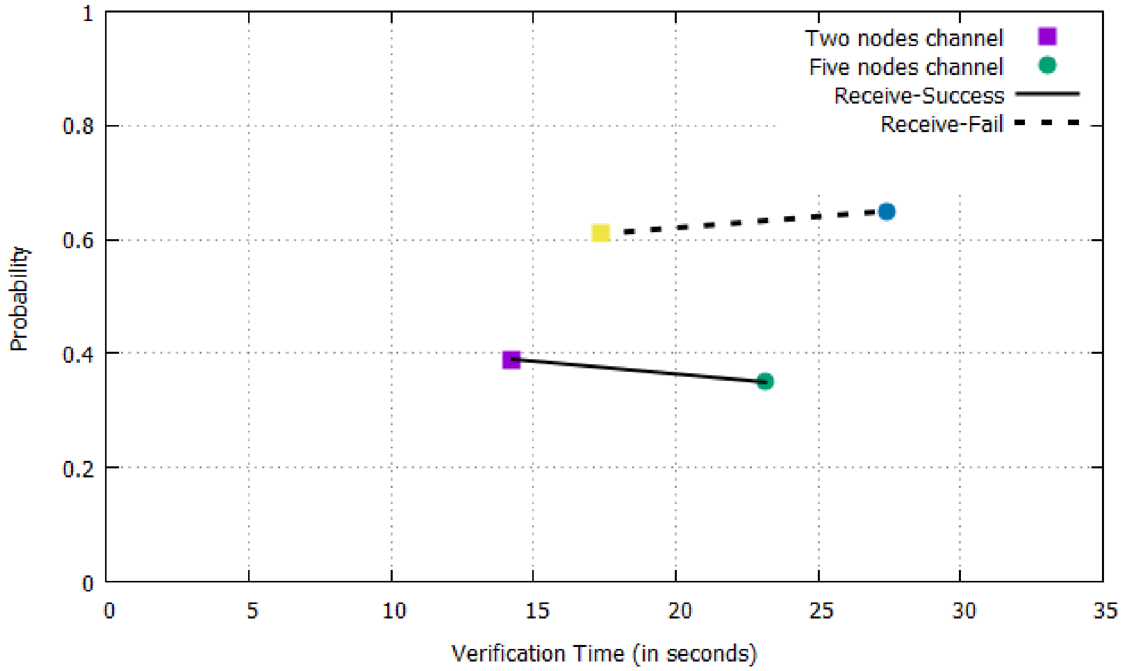


FIGURE 3.15: Verification Results of Properties (3.13) and (3.14) in Multi-Access Channel

Figure 3.15 represents the results of verifying the properties of success and fail in receiving. Where the dotted line represents the probabilty of fail in receiving (i.e., result of verifying Property (3.14)) in two experiments with different sized network. Where the square represents the experiment with two nodes channel network and the circle represents the experiment with five nodes channel. While the line in Figure 3.15 represents the probability of success in receiving (i.e., result of verifying Property (3.13)) in two experiments with different sized network.

Figures 3.14 and 3.15 show that the probability of failure in sending and receiving is higher than the probability of success, compared to the results in the previous section, which could be related to the number of nanomachines in this model, and that each nanomachine in a transmission mode can send u twice to each of its neighbours during one t_{local}. However, if one the destination nanomachines could not detect u (either because it has already received u from another nanomachines. Or because it is not in receiving mode), this can 0ause transmission failure (at the same time, it can cause receiving failure as the destination nanomachine which is in receiving mode has already received u from another nanomachine).

3.5 Summary

In this chapter, a time-slotted one dimensional communication system based on bio-inspired rules, between two nanomachines has been illustrated. Starting by defining the model, then introducing PRISM model checker which has been used to verify properties defined in the model. Later, the results of PRISM verification are presented. Then the defined model is modified to study the effects of the environment noise. The properties of the modified model are then verified using PRISM model checker. The verification results are presented later. That followed by, proposing a bi-directional system, the properties of this system are also verified through PRISM. The results of the bi-directional system then presented. The bi-directional system has been extended to represent a multi access channel system. Then, the analysis of the success and failure of sending and receiving in this system with its PRISM verification results are presented.

Chapter 4

Performance Analysis of Molecular Communication Model

4.1 Introduction

Molecular communication is considered a bio-inspired paradigm, in which molecules are transmitted, propagated and received between nanomachines [111]. Time slotted model consist of n nanomachines in a bounded environment is considered. These nanomachines are communicate according to diffusion based molecular communication. Information molecules are encoded based on the variation in the concentration of molecules in the communication medium. Thus, the receiver nanomachine decode the sensed information molecules during a certain time slot as '1', if its concentration exceeds a certain threshold τ, and '0' otherwise.

The main objective is to study the performance of diffusion based molecular communication model, an algorithm to find the maximum distance (transmission range) that information molecules can reach if a nanomachine kept diffusing molecules during certain time slots. The information molecules are represented as a 1-bit symbol with one threshold. Experiment results show the effects of noise and the medium diffusion coefficient on the transmission range of the diffused molecular concentration.

Next, the pattern of diffusion was explored, by inspecting how a receiver nanomachine could distinguish a message from one transmitter nanomachine at a distance d or two transmitter nanomachines at different distances. The information molecules were represented as 2-bits by using 2^2 different values with $2^2 - 1$ thresholds. In the literature, the performance of the molecular propagation channel has been explored in [117, 119, 121, 145, 167]. Through the evaluation of their results, the factors that can affect the sensed molecular concentration at the receiver nanomachine are concluded. These factors have facilitate the model assumptions in this chapter. Including the assumption that the estimation of the interference from previous diffused information molecules on the current diffused information molecules, depends on distance, time and sensed molecular concentration. Thus, in the implemented experiments the effects of distance, time, sensed molecular concentration and interference are considered.

The chapter is organized as follows. In Section 4.2 the model is described. The range of transmission algorithm and the experiment's results are presented in Section 4.3. In Section 4.4 the proposed pattern of diffusion algorithm, the factors that affect distinguishing the sensed molecular concentration, and the experiment results are demonstrated. A summary of the discussed issues is described in Section 4.5.

4.2 Model

Network Environment: a system of n nanomachines is considered, these nanomachines communicate according to diffusion based molecular communication. The environment of the communication might contain residual molecules from previous diffusion, and also contain molecules from other nanomachines (that are not among n) and these molecules can be considered as noise. In [100] the diffusion based communication system has showed that only the molecules last previous diffusion can affect the current diffusion. Nanomachines are assumed to have simple computational capability, and storage space for the needed computations.

Time Slots: The model is time slotted with length t; thus, a transmitter nanomachine can keep diffusing molecules during this t time slot, where $t = \frac{d^2}{D}$.

Information Molecule Encoding: the concentration of received molecules is considered to be the information molecules (though it is also called the transmitted symbol). Thus, a receiver nanomachine decode the received symbol as '1' in case the number of molecules sensde by the receiver nanomachine during t time slot is higher than a given threshold τ; if not, then the symbol is decoded as '0'. Generally nanomachines are assumed to sense at least $(\varepsilon + \mu)$ molecular concentration, where ε represents the residual molecular concentration and μ can represent the environmental noise. If the nanomachine sense molecular concentration that is higher than $(\varepsilon + \mu)$; i.e., τ, it can recognize that at least one nanomachine is diffusing molecules. In [98] modulation techniques have been explored, one of which is *Concentration Shift Keying*, where symbols (information molecules) can be represented as b bits through 2^b different values, and the levels of threshold can be $2^b - 1$.

Network Communication: The nanomachines n are assumed to communicate through diffusion; thus, once the molecules are released by a transmitter nanomachine into the propagation medium, these molecules shall be diffused freely according to the Brownian motion dynamics. The function of the molecular concentration at the receiver nodes in response to an impulse of information molecules (symbol) emission from the transmitter with Q molecules is of the form [125]:

$$Qh(t) = \int_0^T (Q \times \frac{1}{(4\pi Dt)} \times \exp(\frac{-d^2}{4Dt})) \tag{4.1}$$

where, t represents time and it is the integral variable and range from 0 to T, d denotes the distance between the receiver and the source, D is the diffusion coefficient, Q number of diffused molecules. Thus, the receiver nanomachine sense the accumulated molecular concentration diffused through the time slot t.

Rang of transmission: One of the model's assumptions is that the communication environment is bounded. It is also assumed that when a transmitter nanomachine diffuses information molecules, it can somehow reach all nodes at distance d. One of the objectives of this model is to compute the maximum distance which a diffusing transmitter nanomachine could reach if it kept on diffusing for a certain time. Besides that, to study through experiments, what could affect the value of the maximum distance.

Pattern of Diffusion: Assuming that the information molecules (symbol) are represented as 2 bits with 2^2 different values and $2^2 - 1$ thresholds, the aim is to check how a receiver nanomachine can distinguish a message transmitted from a transmitter nanomachine at distance d. A receiver nanomachine can sense the transmitted information molecules through the following expression, where [125] stated that the molecular concentration peak at a receiver nanomachine is obtained through:

$$Q(p) = Q\left(\frac{3}{2\pi}\right)^{3/2} \frac{1}{d^3} \tag{4.2}$$

where $Q(p)$ means the function to compute the peak of the diffused Q molecules.

From Equation (4.2) the peak of molecular concentration at a receiver nanomachine is inversely proportional to the cube of the distance d, and it is not affected by the diffusion coefficient D of the medium.

However, the time that it takes diffused molecules to reach their peak can be affected by D; thus a receiver nanomachine at distance d from a transmitter nanomachine can sense that then molecular concentration at d peaks at $t' = \frac{d^2}{6D}$, where t' here is computed from the derivative of Equation (4.1) [121].

4.3 Range of Transmission

There is a specific range in which the diffused molecules could be sensed or received. Thus, in order to compute the maximum distance (transmission range) d in which molecules from a transmitter nanomachine could be sensed by the receiver nanomachine, the assumption is that a transmitter nanomachine will keep diffusing as long as t less or equal to T which represents the maximum time the source can continue on diffusing. The receiver nanomachine will keep on sensing as long as the molecular concentration is higher or equal to $(\varepsilon + \mu)$ and t is less or equal to T. The molecular concentration sensed by the receiver nanomachine at each single time t is computed through Equation (4.1).

Algorithm 6 computes range of transmission, assuming that the source keeps diffusing as long as t is less or equal to T.

The initial values of d and t are incremented by 0.01 at each iteration, assuming that these molecules can be sensed at each $0.01nm$, to utilize each $0.01millisecond$ of the time slot, where the initial values of $d = 0$ and $t = 0.01$. The diffusion coefficient $D = 10^{-16}cm^2/s$, the value of $\mu = 2 \times 10^{16}$ molecules.cm^{-3}, and number of molecules diffused per information molecule (symbol) is $Q \approx 2 \times 10^{10}$ molecules, $T = 1000ms$. During simulation experiments, ε value initiated by 0.05, then at each run incremented by 0.05 until it reaches 0.95. Algorithm 6 is considered as an external computation of the transmission range. Thus, Algorithm 6 is not executed by the nanomachines in the network.

Algorithm 6: Finding the Range of Transmission

```
Input: Initial values of parameters in Equation (4.1)
Output: Range of transmission that diffused molecules can reach
$t \leftarrow 0$;
$d \leftarrow 0$;
$f \leftarrow 0$;
while $t \leq T_0$ do
    $d \leftarrow d + 0.01$;
    $t \leftarrow 0.01$;
    while $f \geq (\varepsilon + \mu)$ & $(t \leq T_0)$ do
        $f = f + [Q \times \frac{1}{(4\pi Dt)} \times \exp(\frac{-d^2}{4Dt})]$;
        $t = t + 0.01$;
    $f \leftarrow 0$;
return $d$;
```

The transmission range of diffusion has been calculated for each single value of ε, to check its effect on diffusion. The results of the experiment are 19 figures, where each figure represents the transmission range of the diffused molecules with different ε values. Another experiment was carried out to compute the transmission range with a different value of the medium diffusion coefficient, where $D = 10^{-2}cm^2/s$. The last experiment, assumed that a transmitter nanomachine has a message consisting of six bits, and at each time slot t the transmitter nanomachine diffuses one bit as an information molecule and depending on the message the range of transmission is computed.

□ The results of the first experiment are 19 figures. However, in this part a selective figures are shown to demonstrate the different values of transmission range.

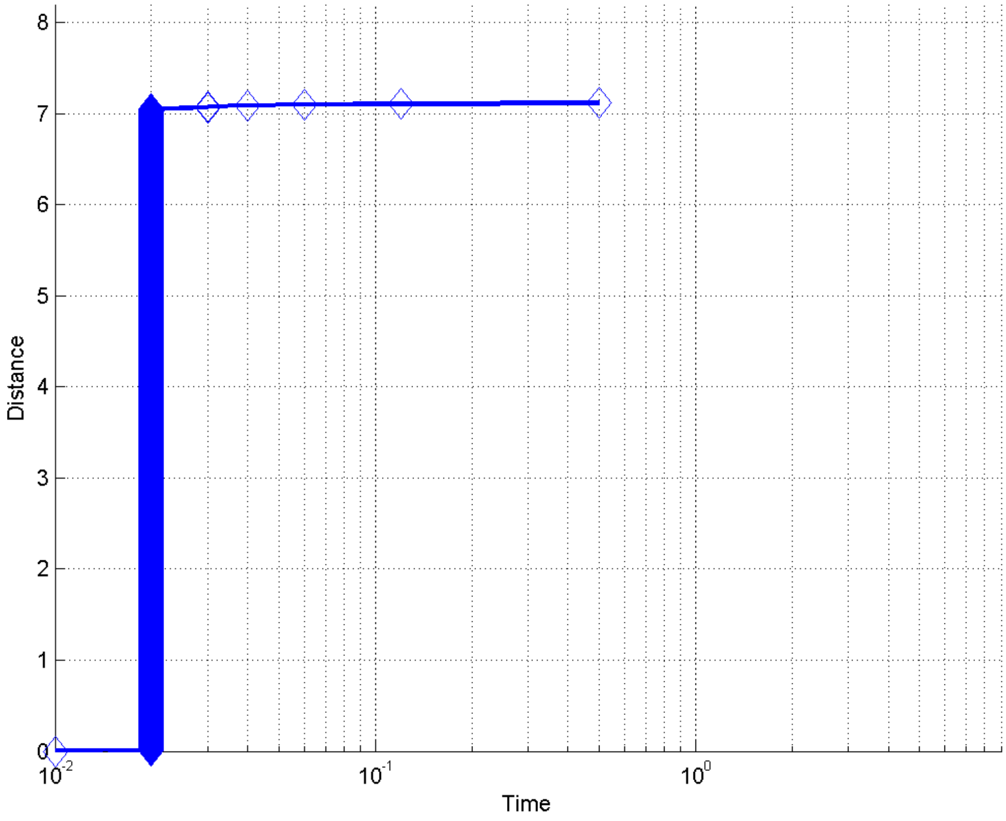


FIGURE 4.1: $max_d = 7.130$, $\varepsilon = 0.05$

The x-axis in Figure 4.1 represents the value of T and the required time to reach the maximum distance (highest range of transmission) when $\varepsilon = 0.05$. The y-axis represents the value of d, where the point $max_d = 7.130$ represents the highest range of transmission when $\varepsilon = 0.05$. The line in Figure 4.1, consists of several points which represents the value of d at each certain t. As the value of d was increased by 0.01 in the experiment at each 0.01ms of t.

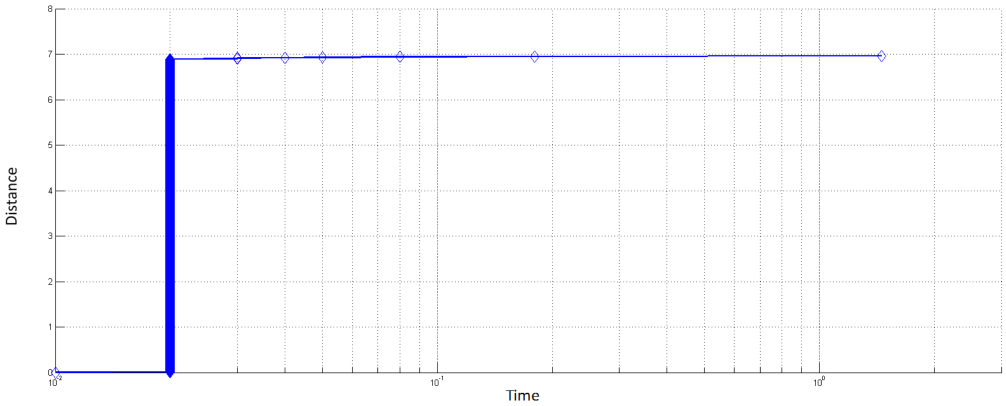


FIGURE 4.2: max_d=6.970, ε=0.50

Similar to 4.1, the x-axis in Figure 4.2 represents the value of T and the required time to reach the maximum distance, but when $\varepsilon = 0.50$. The y-axis represents the value of d, where the point $max_d = 6.970$ represents the highest range of transmission when $\varepsilon = 0.50$. Figure 4.2 shows that the value of the transmission range decreases as the value of ε increases;

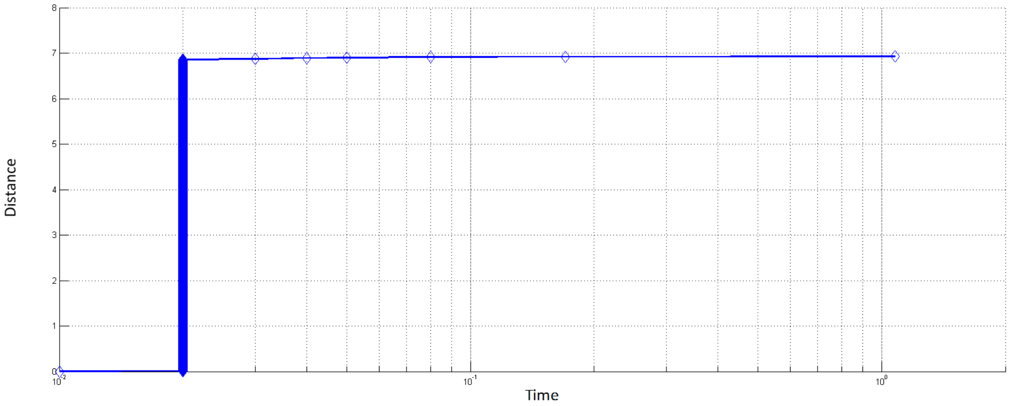


FIGURE 4.3: max_d=6.940, ε=0.75

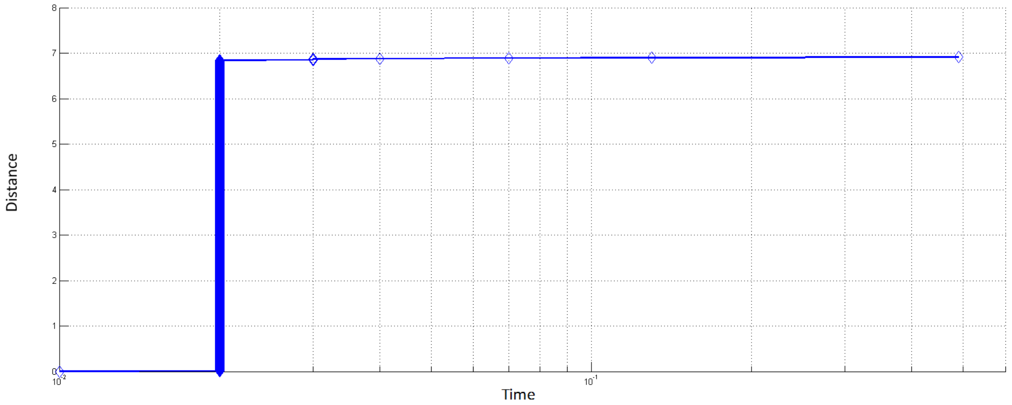


FIGURE 4.4: max_d=6.920, ε=0.95

Figures 4.3 and 4.4 show the value of the transmission range when ε=0.75 and 0.95 receptively. Where the values of max_d=6.940 and 6.920 represent the highest value of transmission range when ε=0.75 and 0.95 receptively. Figures 4.1 to 4.4 show that the value of the transmission range decreases as the value of ε increases.

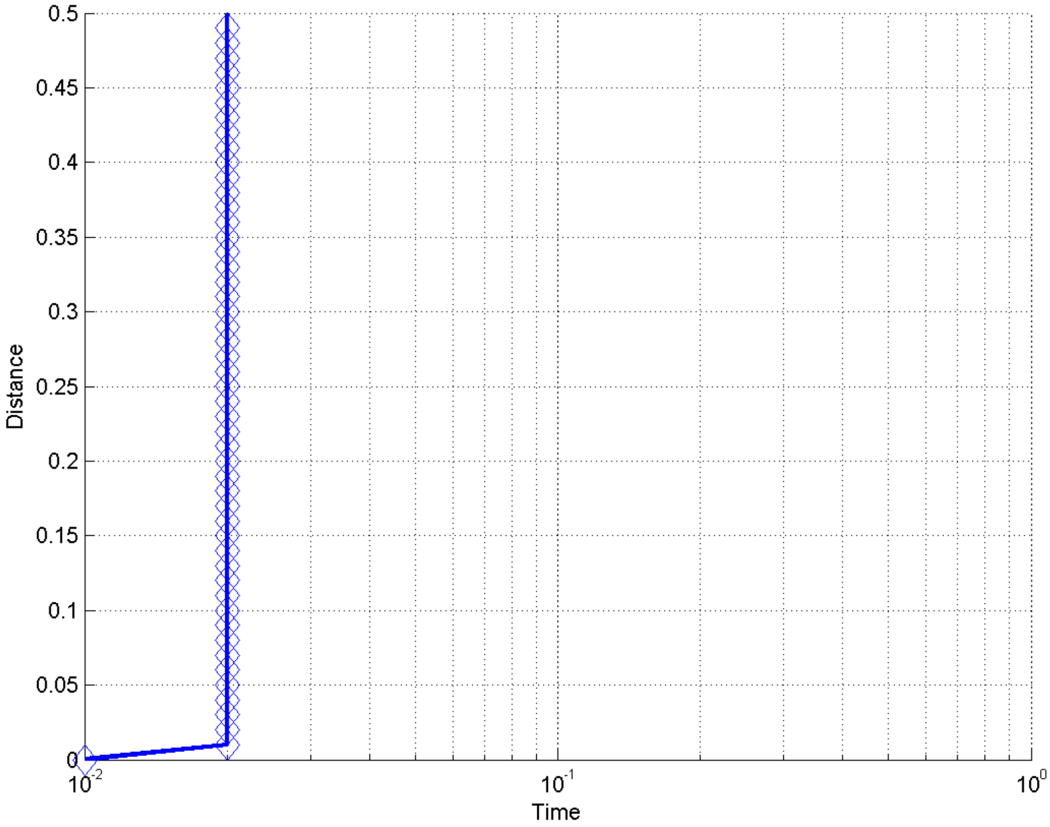


FIGURE 4.5: 'a zoom in' of each line

Figure 4.5 gives a 'zoom in' view of each line in the previous figures which represents the maximum transmission range. As the value of d was increased by 0.01 in the experiment; thus, at each 0.01 there is a new value of d, but this wasn't clear in Figures 4.1 to 4.4.

□ The following are the results of computing transmission range with a different value of the medium diffusion coefficient $D = 0.001 cm^2/s$. The values of the other parameters are the same as the values described in the first experiment. The figuers shows the results of transmission range with selective values of ε:

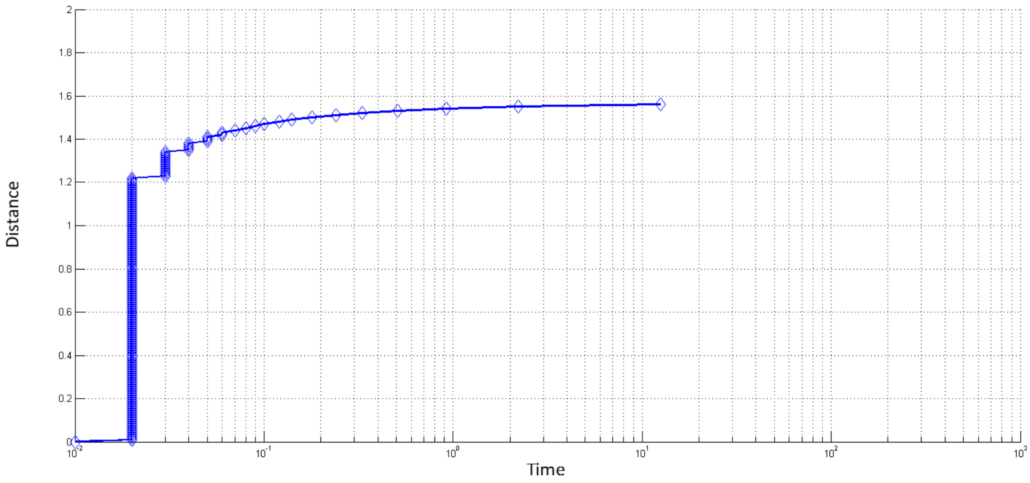


FIGURE 4.6: max_d=1.570, ε=0.05

The x-axis in Figure 4.6 represents the value of T and the required time to reach the maximum distance (highest range of transmission) when $\varepsilon = 0.05$. The y-axis represents the value of d, where the point $max_d = 1.570$ represents the highest range of transmission when $\varepsilon = 0.05$.

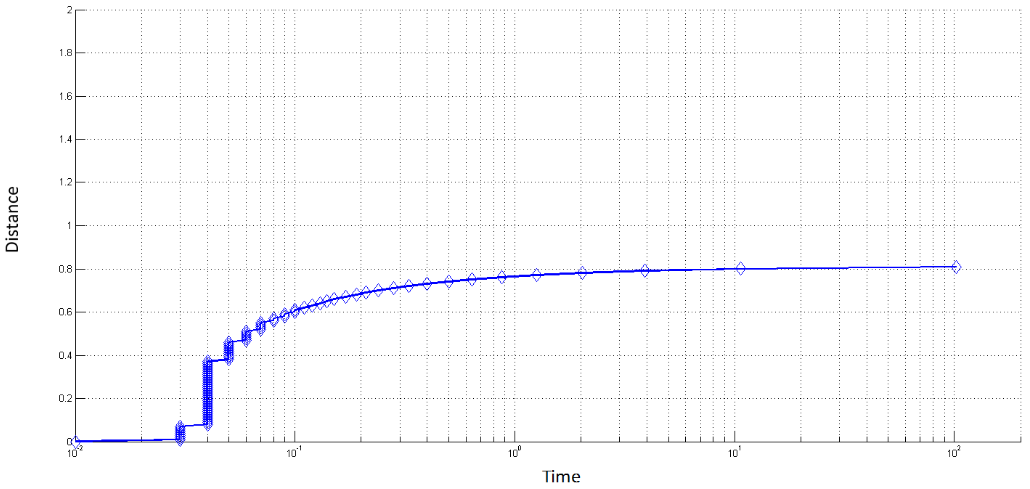


FIGURE 4.7: max_d=0.820, ε=0.30

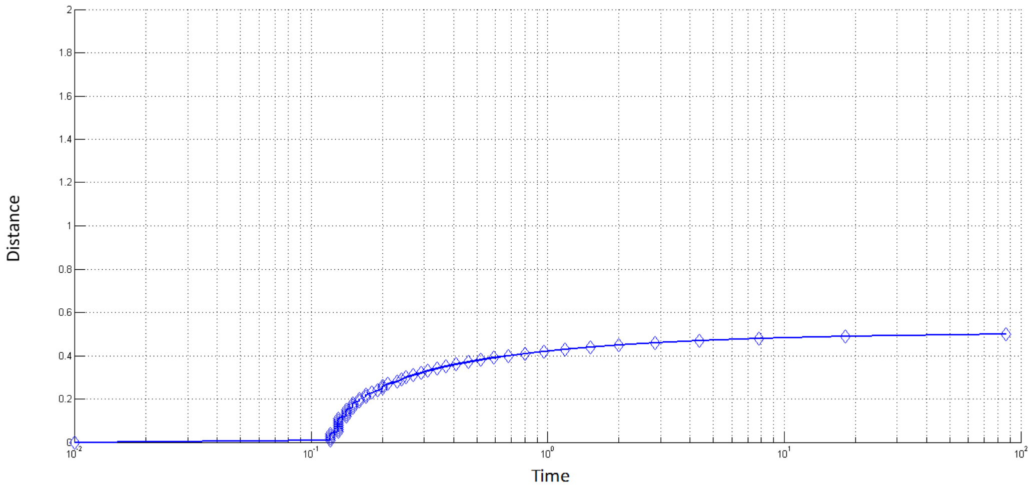


FIGURE 4.8: max_d=0.510, ε=0.45

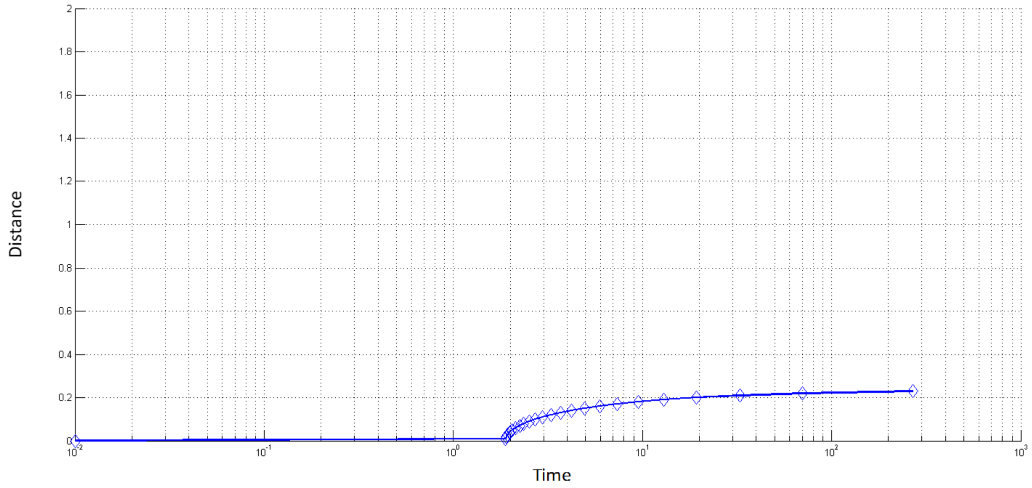


FIGURE 4.9: max_d=0.240, ε=0.55

The x-axis in Figures 4.7, 4.8 and 4.9 represents the value of T, i.e., the required time to reach the highest range of transmission when ε=0.30, 0.45, 0.55 receptively. The y-axis represents the value of d, where the point max_d=0.820, 0.510, 0.240 represents the highest range of transmission when ε=0.30, 0.45, 0.55 receptively.

The values of transmission range when $D = 0.001cm^2/s$ are less than the results when $D = 10^{-16}cm^2/s$. In this experiment, the result of the transmission range is always 0.01 when the value of ε is between [0.60, 0.95].

□ The following results are the transmission range value when a transmitter nanomachine has a message consisting of six bits. At each time slot t the transmitter nanomachine diffuses one bit (i.e., if the bit is 1, then, the transmitter nanomachine diffuses Q information molecules. Otherwise it does not diffuse any information molecules). The range of transmission has been computed for three different messages, with the same initial values described in the first experiment with different ε values. Thus Algorithm 6 is changed to represents Algorithm 7 as follows:

Algorithm 7: Transmission range of 6 bits message

```
$d \leftarrow 0$;
$t \leftarrow 0$;
$f \leftarrow 0$;
$s \leftarrow [011101]$; // a message example
$x \leftarrow 0$;
while $t \leq T_0$ do
    $d \leftarrow d + 0.01$;
    $t \leftarrow t + 0.01$;
    while $f \leq (\varepsilon + \mu)$ & $(t \leq T_0)$ do
        while $x \leq$ length of $s$ do
            if $s(x) == 1$ then
                $f = f + [Q \times \frac{1}{(4\pi Dt)} \times \exp(\frac{-d^2}{4Dt})]$;
            else
                f=f+0;
            t=t+0.01
    if $f == 0$ then
        break ;                                  // if the message is all zeros
    else
        $f \leftarrow 0$;
return $d$;
```

Three experiments have been carried out to check the range of transmission with three different messages:

1. Assuming that a transmitter nanomachine has [100000] message, range of transmission has been computed with four different ε values:

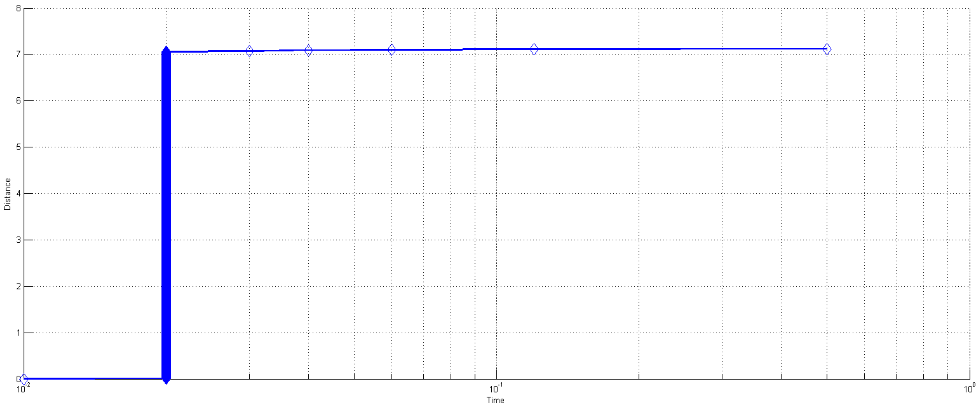


FIGURE 4.10: max_d=7.13, ε=0.05

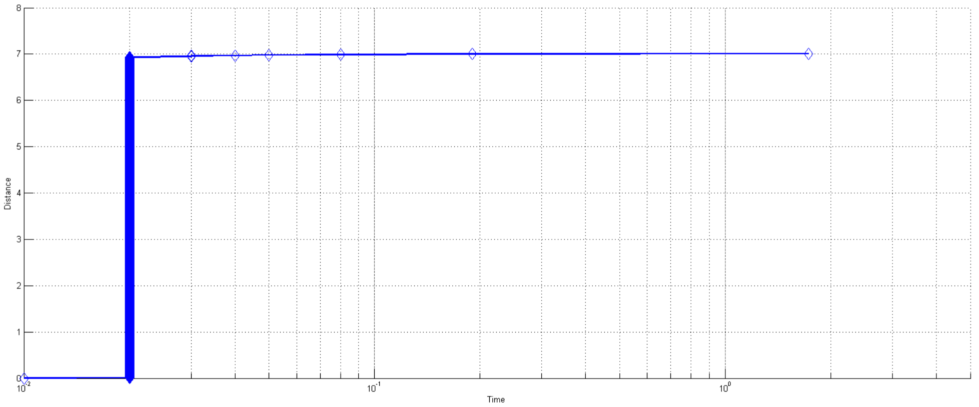


FIGURE 4.11: max_d=7.02, ε=0.25

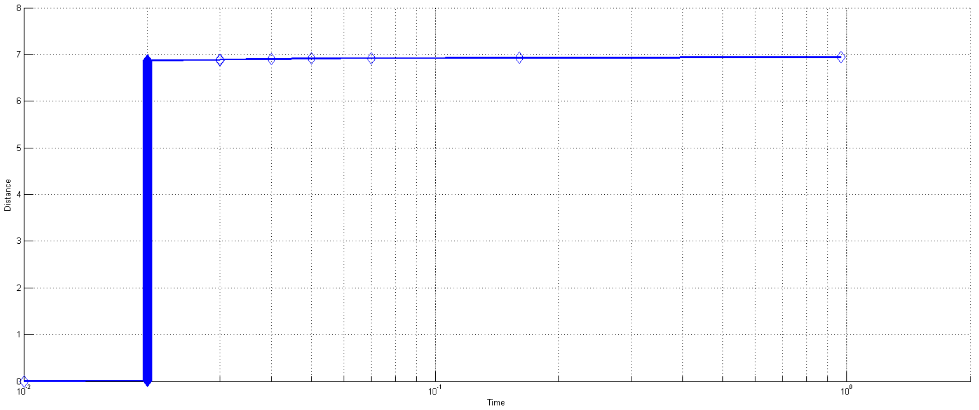


FIGURE 4.12: max_d=6.95, ε=0.65

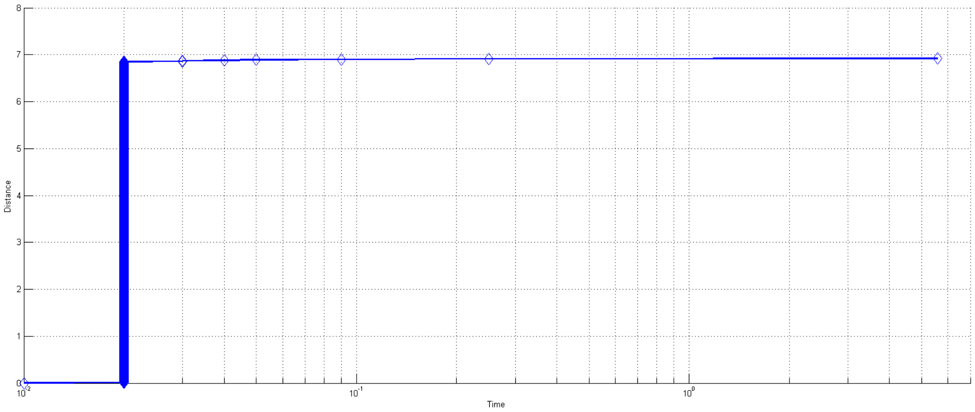


FIGURE 4.13: max_d=6.93, ε=0.90

2. Assuming that a transmitter nanomachine has a message [101010], the values of transmission range of different ε values are as follows:

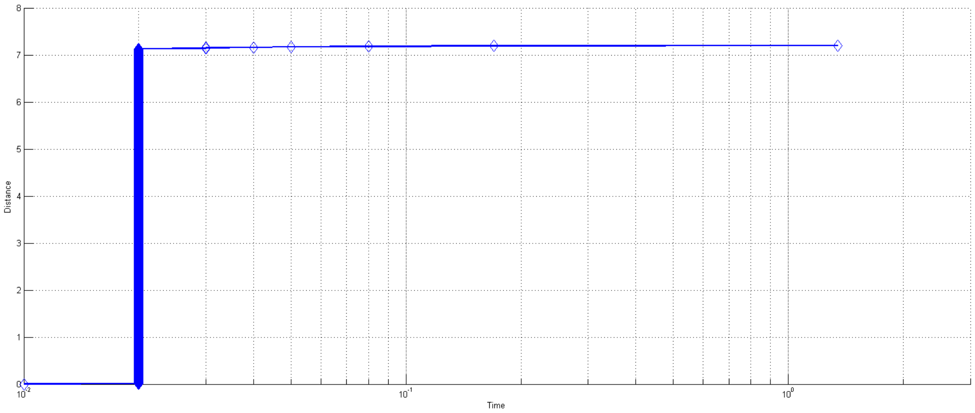


FIGURE 4.14: max_d=7.21, ε=0.05

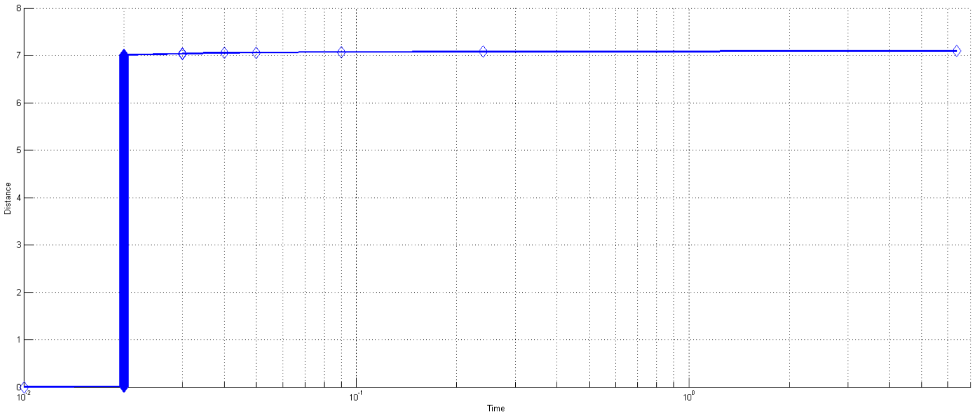


FIGURE 4.15: max_d=7.10, ε=0.25

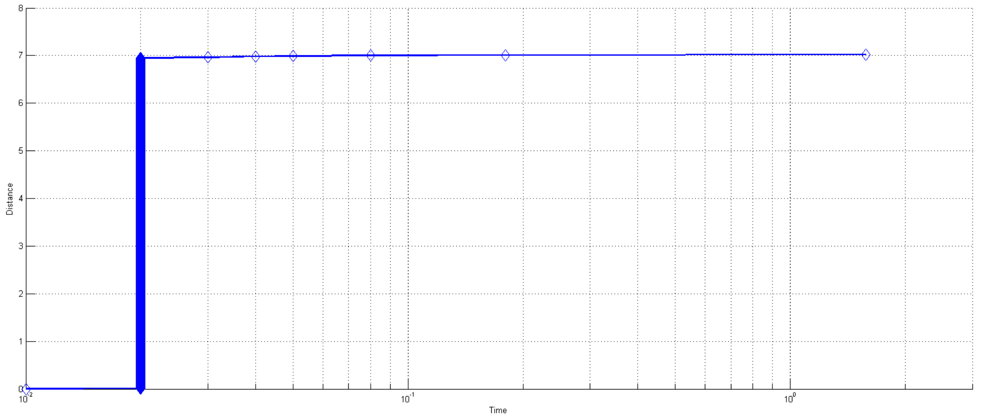


FIGURE 4.16: max_d=7.03, ε=0.65

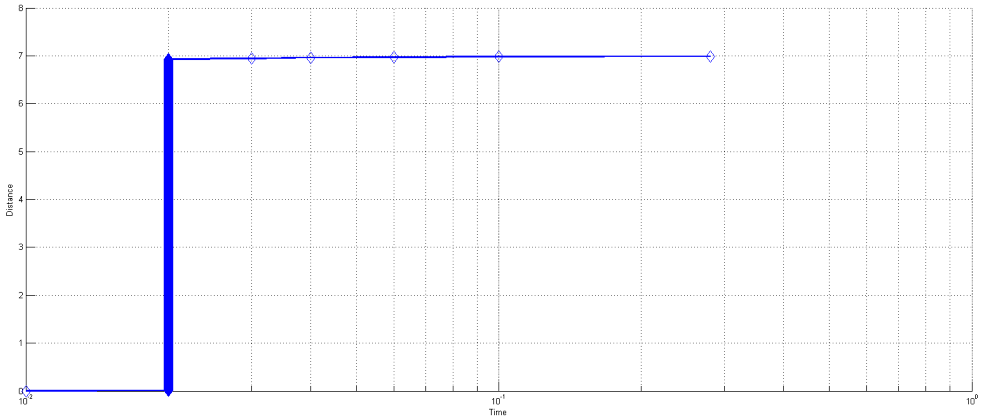


FIGURE 4.17: max_d=7.00, ε=0.90

3. Assuming a transmitter nanomachine has a message [111110], the transmission range values are as follows:

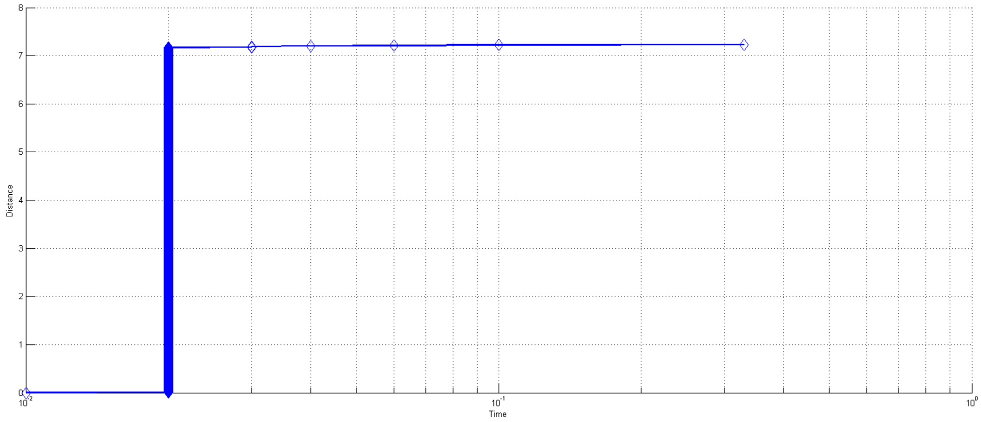


FIGURE 4.18: max_d=7.24, ε=0.05

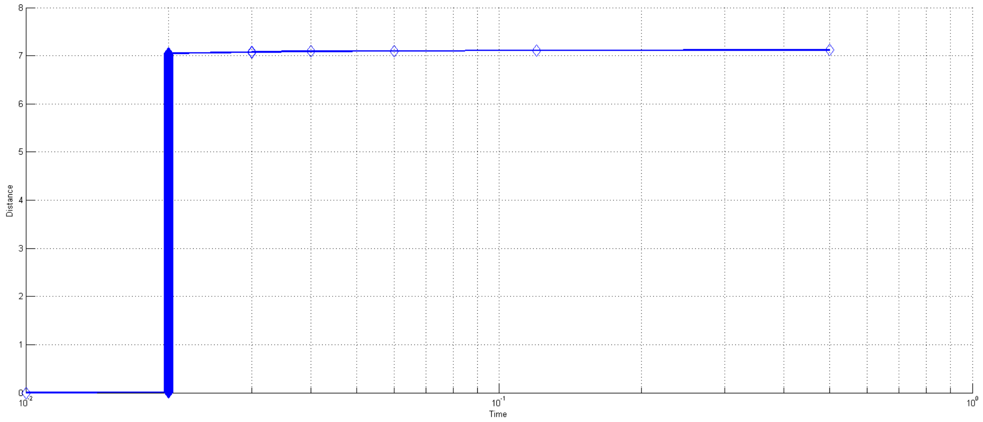


FIGURE 4.19: max_d=7.13, ε=0.25

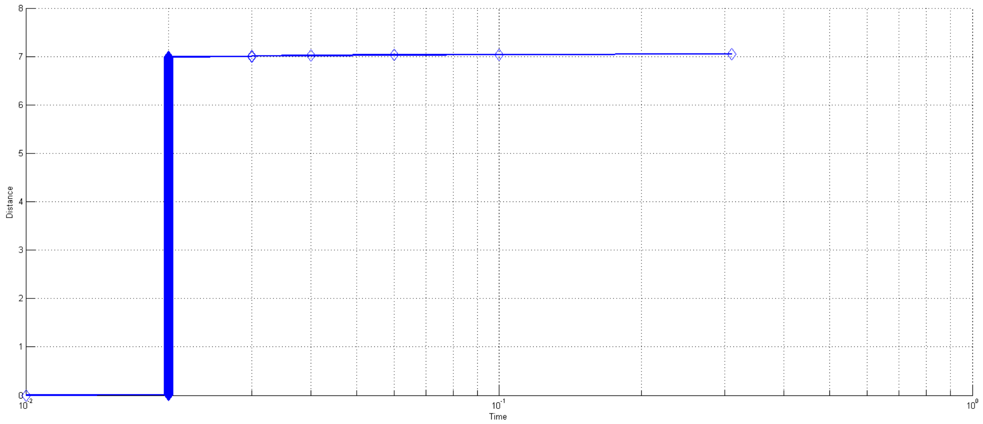


FIGURE 4.20: max_d=7.06, ε=0.65

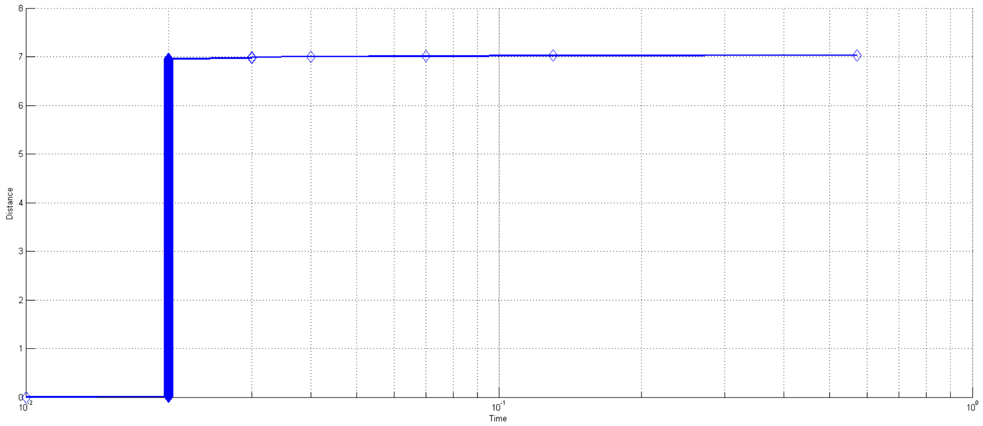


FIGURE 4.21: max_d=7.04, ε=0.90

The value of the transmission range decreases as the value of ε increases; however, the higher value of max_d is achieved with the message that consists of highest number of '1'.

4.4 Pattern of Diffusion

This section examines how a nanomachine at distance d can recognize the information molecule (symbol) that a transmitter nanomachine has been diffusing. Where the information molecules (symbol) is represented as 2 bits with 2^2 different values and $2^2 - 1$ thresholds, i.e., the symbol is encoded according to the Quadruple Concentration Shift Keying(QCSK) [98, 167], as Figure 4.22 shows.

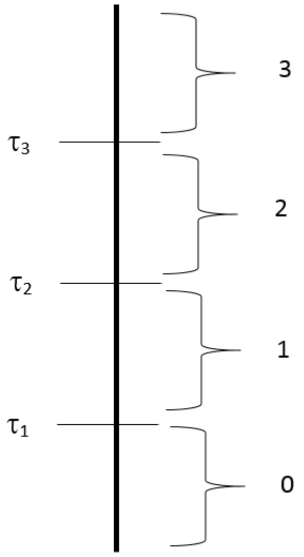


FIGURE 4.22: QCSK technique for 2 bits per symbol [98]

Thus, the diffused symbol is represented as two bits in different forms: 00, 01, 10, and 11, and there are three different thresholds for the receiver nanomachine to distinguish the symbol. In order to give values to these thresholds, the transmitter nanomachine in this section is assumed to diffuse units of molecules throughout 8 time slots, each slot of length t; and the accumulated value of the diffused units throughout 8 t would encode a symbol either $s_0 = (00)$, $s_1 = (01)$, $s_2 = (10)$, and $s_3 = (11)$. To encode a symbol, a transmitter nanomachine diffuses either Q molecules to represent 1 or 0 molecules to represent 0, during each t of the 8 t. Here, it is assumed that symbols are represented as the following:

Symbol	Number of Molecules	Molecules diffused through each t of 8 time slots	thresholds
s_0= 00	n_0	1 0 0 0 0 0 0 0	τ_0
s_1= 01	n_1	1 1 1 0 0 0 0 0	τ_1
s_2= 10	n_2	1 1 1 1 1 0 0 0	τ_2
s_3= 11	n_3	1 1 1 1 1 1 1 1	τ_3

Table 4.1: Symbols representation through 8 time slot

Although s_0 means that no information molecules would be diffused, but to distinguish between no diffusing and diffusing 0, it is assumed that n_0 is diffused to represent s_0. The number of molecules, for example n_1 equals to $(1 \times Q) + (1 \times Q) + (1 \times Q) + (0 \times Q) + (0 \times Q) + (0 \times Q) + (0 \times Q) + (0 \times Q)$. The expression in Equation (4.2) represents the peak of diffused information molecules, the values of thresholds can be computed through [125]:

$$\tau = X \times Q \left(\frac{3}{2\pi} \right)^{3/2} \frac{1}{d^3} + I \tag{4.3}$$

where X represents the diffused units in each time slot t', i.e., 0 or 1. The Inter Symbol Interference (ISI), which means the residue molecules from the previous symbol that can affect the current symbol is denoted by I. The ISI in [125] is assumed to come from a sufficient number of interfering sources (nanomachines which are diffusing), in a way that I follows a normal distribution. However, the experiments which have been carried out in this chapter, study the pattern of diffusion of one transmitter nanomachine and a receiver nanomachine, then assume that there are two transmitter nanomachines and one receiver. Thus, the diffused information molecules (symbol) will be affected by ISI from at least one nanomachine.

ISI can affect the successful detection of the (signal) diffused molecular concentration [121]. The effects of ISI can vary with the temporal spreading properties of molecules in the diffusion channel [117]. Thus, the increased distance between the transmitter and receiver nanomachines, and/or the higher data rate (sensed molecular concentration) can cause increased effects of ISI [121]. The authors in [145] proposed an enzyme-based scheme to reduce the effects of ISI, through diffusing enzymes which chemically interact with the ISI information molecules (from previous symbol) and form intermediate products; thus, in this way the information molecules from the previous symbol would not cause ISI in the current symbol. Symbol duration t_s which represents the required time to transmit a symbol (information molecule), can also affect ISI, and a longer symbol duration can help to reduce the ISI caused by the previous symbol [98].

In the experiments to check the pattern of diffusion in this section, the value of ISI was assumed to vary depending on the distance, data rate and symbol duration, in a way that ISI $\leq \log(\frac{d}{t_s} \times \text{data rate})$.

The rest of this section includes an algorithm to distinguish the pattern of diffusion, followed by the results of different experiments based on the algorithm. In order for a receiver nanomachine to distinguish the pattern of information molecules diffused by a transmitter at distance d, it should compute τ_1, τ_2, τ_3 and τ_4, through 4.3 and follow the steps in Algorithm 8.

Algorithm 8: Diffusion Pattern Recognition

Input: Initial values of Equation 4.3 parameters

Output: Recognize the Diffused Symbol

```
$t1 \leftarrow 0$;
$\tau \leftarrow 0$;
$i \leftarrow 1$;
while $i \leq 8$ do
    $x \leftarrow X(i)$;
    while $t1 \leq t$ do
        $\tau = \tau + (x \times Q\left(\frac{3}{2\pi}\right)^{3/2}\frac{1}{d^3} + I)$;
        $t1 = t1 + 0.01$;
if $(\tau > (\varepsilon + \mu))$ & $(\tau \leq \tau_1)$ then
    Symbol is 00 ;
else if $(\tau > \tau_0)$ & $(\tau \leq \tau_1)$ then
    Symbol is 01 ;
else if $(\tau > \tau_1)$ & $(\tau \leq \tau_2)$ then
    Symbol is 10 ;
else if $(\tau > \tau_2)$ & $(\tau \leq \tau_3)$ then
    Symbol is 11 ;
```

In Algorithm 8, it is assumed that each bit of the transmitter message is diffused during t time slot. For simplicity, the time needed for molecular concentration to reach its peak near the receiver nanomachine t' and symbol duration t_s are assumed to equal $(8 \times t)$. Recall that X represents the diffused units in each time slot. As there are 8 time slots, $X(i)$ represents the diffused unit at a specific t from the 8 time slots.

Different experiments have been carried out to distinguish the pattern diffusion at a receiver nanomachine following the steps in algorithm 8:

1. **Recognizing the Diffusion Pattern of One nanomachine:**

   The first experiment is to check how a receiver nanomachine can recognize the received information molecules. Taking in consideration the effects the time slot length and the inter symbol interference on the recognizing the received molecules correctly.

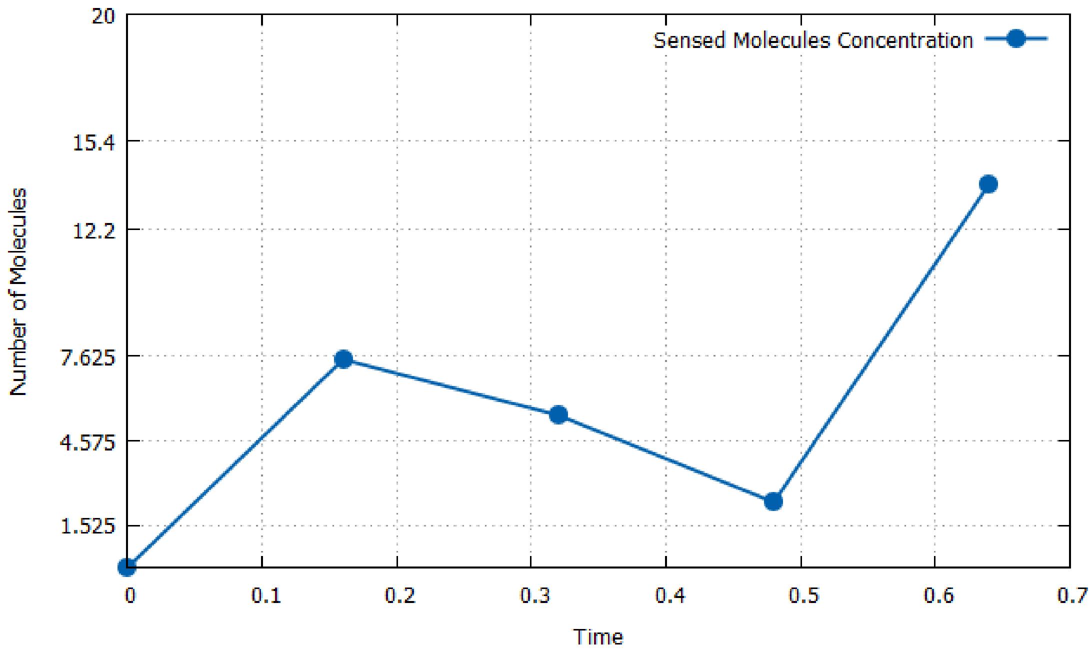


FIGURE 4.23: Sensed Symbols by Receiver from Transmitter at d=5 and t_s=0.16

Figure 4.23 represents an experiment where four symbols were diffused by a transmitter nanomachine at distance 5 from a receiver nanomachine. In the experiments, Q is assumed to equal 1000, and $(\varepsilon + \mu)$ is assumed to equal 0.25. The assumed time slot t to diffuse one bit of the 8bits symbol is 0.02 ms, and t', t_s=0.16 ms. The ticked points at the y-axis in Figure 4.23 represents the values of thresholds, where, τ_0= 1.525, τ_1=4.575, τ_2=7.625 and τ_3= 12.2. The x-axis represents the time required before the diffused symbol reaches to its peak at the receiver nanomachine. The y-axis represents the sensed molecular concentration (peak of the diffused symbol) at certain t'. Even though the receiver nanomachine can sense at least $(\varepsilon + \mu)$ in each time slot t, the figures which represent the experiments results have an initial value of the sensed molecules by the receiver that equals 0.

The four diffused symbols in Figure 4.23 are: $10, 01, 00, 11$. The effects of ISI during the first symbol duration is assumed to be quite low. Thus, it can be seen that the first diffused symbol sensed and distinguished correctly. However, the effects of ISI start from the second symbol; therefore, in Figure 4.23, the sensed molecular concentration during the second, third, and fourth t_s is higher than the actual diffused molecular concentration, and there is a chance of error in the process of distinguishing a symbol. Thus, symbols $01, 00, 11$ have not distinguished correctly in Figure 4.23. This can be due to the short symbol duration t_s.

In Figure 4.24 the assumed time slot t to diffuse one bit of the 8bit symbol is 0.4 ms, and t', t_s=3.2 ms. The diffused symbols in this figure are: $01, 11, 10, 00, 10$, and the results show that the receiver nanomachine sensed and distinguished the correct symbols. Thus, Figure 4.24 shows that the diffused symbols were not recognized correctly due to the short symbol duration.

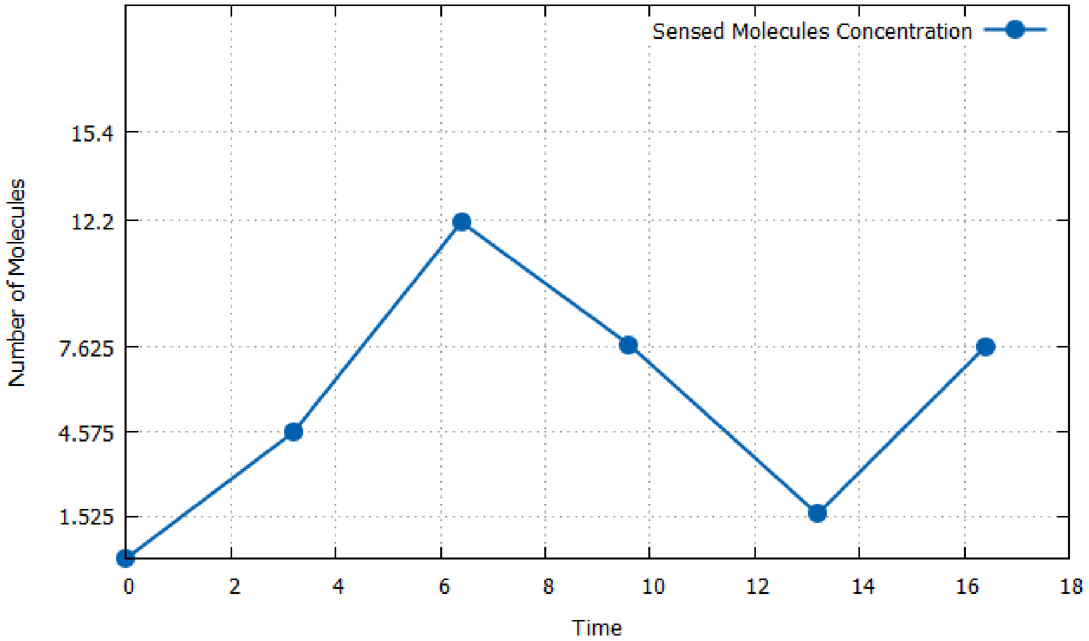


FIGURE 4.24: Sensed Symbols by Receiver from Transmitter at d=5 and t_s=3.2

Thus, the diffused symbols were not recognized correctly due to the short symbol duration in Figure 4.23. However, The receiver nanomachine was able to recognize the symbols correctly after increasing the symbol duration in Figure 4.24.

2. **Recognizing the Diffusion Pattern of One nanomachine**

   In case there are two different transmitter nanomachines at different d from a receiver nanomachine. These two transmitter nanomachines are assumed to be synchronized. In a way that, one nanomachine starts diffusing at a certain symbol duration, and the second one waits, then at the next symbol duration the second nanomachine diffuses and so on. The receiver nanomachine is assumed to have 8 thresholds to recognize the diffused symbol and to distinguish from which transmitter nanomachine it has come.

   Figure 4.25 shows the results of the sensed molecular concentration by a receiver nanomachine that was diffused from two transmitter nanomachines in d=3, and 5.

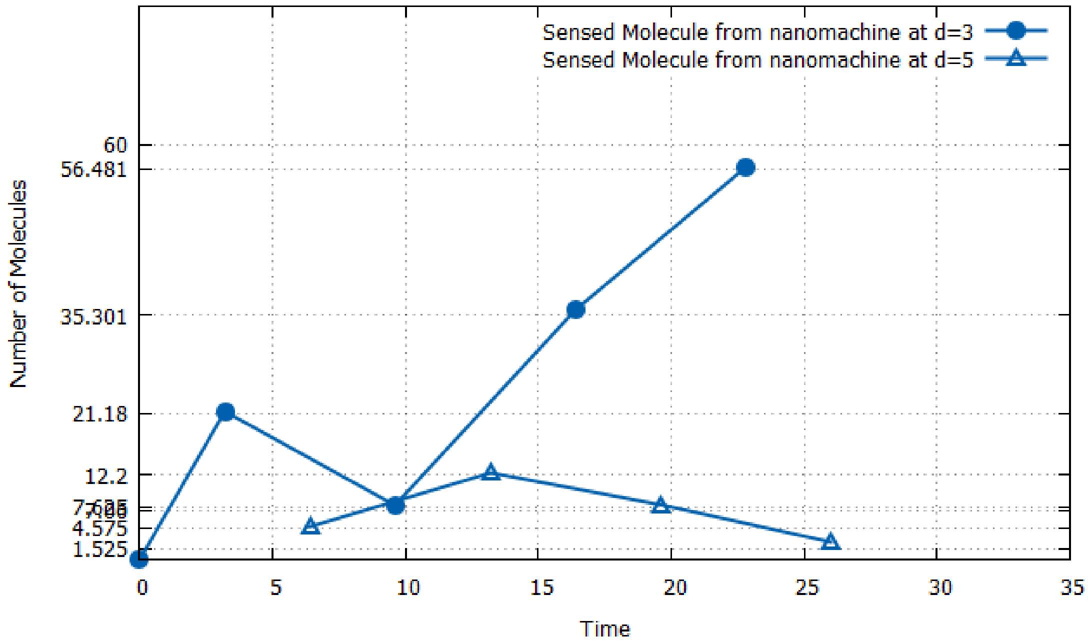


FIGURE 4.25: Sensed Symbols by Receiver during t_s=3.2 from Transmitters at d=3 and d=5

The number of molecules diffused by each transmitter nanomachine is assumed to equal Q =1000. The assumed time slot t to diffuse one bit of the 8bit symbol is 0.4 ms, and t', t_s=3.2 ms. The threshold values of the transmitter nanomachine at d=3 are: $(\tau_0 = 7.060, \tau_1 = 21.180, \tau_2 = 35.301, \tau_3 = 56.481)$. The remaining thresholds are as described in Figure 4.23. Thus, the ticks on the y-axis represent all 8 thresholds. The symbols diffused by the nanomachine at distance $d = 3$ are: $01, 00, 10, 11$, and the symbols from the nanomachine at d=5 are: $01, 11, 10, 00$. The results in Figure 4.25, seem to show that the receiver nanomachine can distinguish the symbols correctly, but, as some threshold values overlap, it is difficult for the receiver nanomachine to distinguish whether a symbol comes from one nanomachine or another. However, the figure differentiates between the symbols of each nanomachine (as the sensed molecular concentration of each nanomachine saved in a different array), but the receiver mainly just compares the sensed molecular concentration with the thresholds.

The experiment is repeated, but with different values of d, as Figure 4.26 shows:

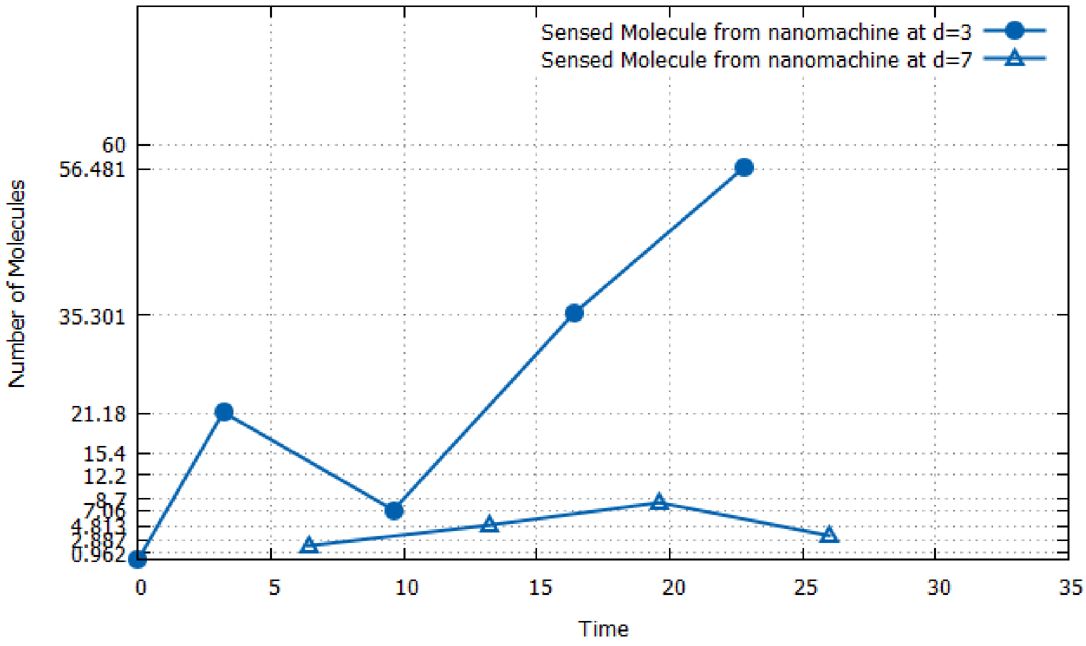


FIGURE 4.26: Sensed Symbols by Receiver during t_s=3.2 from Transmitters at d=3 and d=7

The overlapping of the threshold values in Figure 4.26 might look quite low, but most values of one nanomachine are quite close to one threshold of the other nanomachine. The two transmitters are assumed to be at distances d=3, and d=7. The threshold values of d =7 are: $(\tau_0 = 0.962, \tau_1 = 2.887, \tau_2 = 4.813, \tau_3 = 7.700)$. The diffused symbols of the transmitter nanomachine at d=3 are: $01, 00, 10, 11$, and symbols of transmitter nanomachine at d=7 are: $00, 10, 11, 01$.

Thus, in Figure 4.25 the receiver nanomachine can recognize the diffused symbols correctly. However, it cannot distinguish whether symbol came from one nanomachine or another. It is due to the overlapping in the values of the sensed molecular concentration which come from each nanomachine in this experiment. Where the reason of the overlapping in these values came from the close distance between the two transmitter nanomachines. However, in Figure 4.26 changing the distance of the two transmitter nanomachines from the receiver nanomachine can make the overlapping quite low.

In case one transmitter nanomachine is in a close distance from the receiver nanomachine. And the other nanomachine is in a far distance from the receiver nanomachine. This case is presented to avoid overlapping between thresholds values. Thus, different distances were selected in the next experiment.

Figure 4.27 shows the sensed molecular concentration by a receiver nanomachine, when two transmitter nanomachines at d=0.3 and d=3 diffuse information molecules.

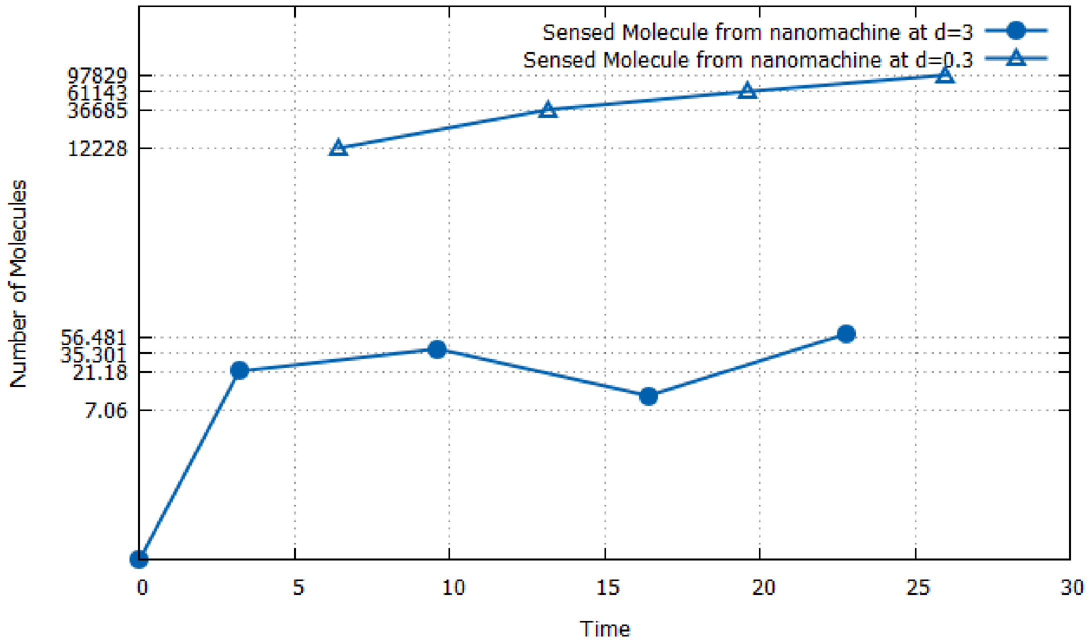


FIGURE 4.27: Sensed Symbols by Receiver during t_s=3.2 from Transmitters at d=0.3 and d=3

In Figure 4.27, the thresholds values of the sensed molecular concentration which come from the transmitter nanomachine at d=0.3 are: $(\tau_0 = 12228, \tau_1 = 36685, \tau_2 = 61143, \tau_3 = 97829)$, which shows a higher data rate at this distance. The symbols diffused by the nanomachine at d=0.3are: $00, 01, 10, 11$. The symbols diffused by the nanomachine at d=3 are: $01, 10, 00, 11$. The diffused symbols from the nanomachine at d=3 are affected by ISI. However, the diffused symbols from the nanomachine at d= 0.3 are not affected that much by ISI, as the range between its thresholds is high (for example, the difference between τ_0 and τ_1 is almost 24457). Besides this, the data rate of the diffused symbols from the nanomachine at d=3 is quite low.

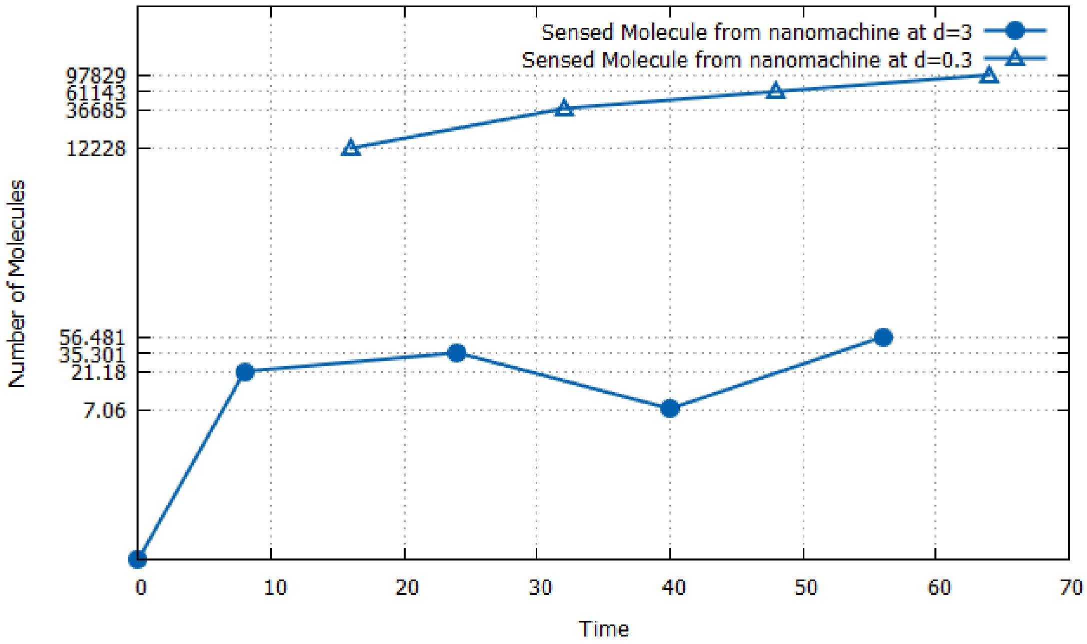


FIGURE 4.28: Sensed Symbols by Receiver during t_s=8 from Transmitters at d=0.3 and d=3

In Figure 4.28, the assumed time slot t to diffuse one bit of the 8bit symbol is 1 ms, and t', t_s=8ms. The diffused symbols from both nanomachine at d=0.3 and 3, are the same diffused symbols in Figure 4.27, as symbol duration increased the effects of ISI on the received symbols from the nanomachine at d=3 is decreased in Figure 4.28.

Thus, in Figure 4.27 the receiver nanomachine cannot recognize the diffused symbols correctly. Due to the higher data rate of the sensed molecular concentration from the close transmitter nanomachine. Thus, the sensed molecular concentration from the far nanomachine is affected by the interference of molecules from the previous symbol duration.

However, in Figure 4.28 by increasing the symbol duration, the receiver nanomachine can recognize the diffused symbols correctly. Beside distinguishing if the symbol came from one nanomachine or another.

4.5 Summary

In this chapter an algorithm to measure the range of transmission in a diffusion based molecular communication has been proposed, taking into consideration the environment noise and how it could affect the diffused molecules and the distance it can reach. The experiment results to compute the transmission range show that the transmission range decreases as the value of both noise and residual molecular concentration increase. Then, the diffusion coefficient of the medium was changed, in order to inspect its impact on the distance of the diffused molecules. The experiment results show that the transmission range decreases as the value of the medium diffusion coefficient increases. Later, a message which consisted of six bits is assumed to be diffused by a transmitter nanomachine to find the range of transmission. The experiment results to compute the transmission range demonstrate that the higher transmission range is achieved with the message that consists of higher number of '1'.

The pattern of diffusion has been explored, by defining the factors which can affect the sensed molecular concentration by the receiver, such as distance, interference, symbol duration and data rate. Experiment results showed the effects of increasing the symbol duration of diffused symbols on the sensed molecular concentration at the receiver nanomachine.

Two cases were considered in the experiment. The first case is how a receiver nanomachine can distinguish the pattern of diffusion of one transmitter nanomachine at distance d. The experiment results of diffusion pattern recognition show that the diffused symbols were not recognized correctly due to the short symbol duration. (Symbol duration is the time duration between two consequent transmissions). In this experiment the symbol duration was 0.02 *ms*. Then, the receiver nanomachine was able to recognize the symbols correctly after increasing the symbol duration. The symbol duration increased to be 3.2 *ms*.

The second case is related to recognizing the pattern of diffusion by the receiver nanomachine, when there are two transmitter nanomachines at different distances, diffuse molecular concentration. The experiment results to check how the receiver nanomachine can recognize the pattern of diffusion show that, the receiver nanomachine can recognize the diffused symbols correctly. However, it cannot distinguish whether symbol came from one nanomachine or another. It is due to the overlapping in the values of the sensed molecular concentration which come from each nanomachine in this experiment. Where the reason of the overlapping in these values came from the close distance between the two transmitter nanomachines. However, changing the distance of the two transmitter nanomachines from the receiver nanomachine can make the overlapping quite low. In case one transmitter nanomachine is in a close distance from the receiver nanomachine. And the other nanomachine is in a far distance from the receiver nanomachine. The results show that the receiver nanomachine cannot recognize the diffused symbols correctly. Due to the higher data rate of the sensed molecular concentration from the close transmitter nanomachine. Thus, the sensed molecular concentration from the far nanomachine is affected by the interference of molecules from the previous symbol duration. By increasing the symbol duration, the receiver nanomachine can recognize the diffused symbols correctly. Beside distinguishing if the symbol came from one nanomachine or another.

Chapter 5

Consensus Problem in Molecular Communication with Leader Election and Energy Harvesting Algorithms

5.1 Introduction

This chapter considers a communication nanonetwork that consists of n nanomachines, where one of these nanomachines has some special responsibilities for directing and controlling processes inside this nanonetwork, and its notation is $node_c$. The nanomachines communicate through a shared unguided medium by stipulating and controlling diffusion processes. The messages in diffusion based molecular communication are encoded as molecules and conveyed by the changes in the concentration of molecules in the environment. The medium of communication might contain residual molecules from previous diffusion, and also contains other types of molecules which can be considered as a noise. Consensus problem in molecular communication has been studied in this chapter, inspired by a model in [59], where the authors consider an iterative method for communication among nanomachines which enables information spreading and averaging in their nanonetwork.

A consensus protocol among nanomachines in diffusion based molecular communication has been proposed here. The proposed protocol, includes two phases that take place during different time rounds. The first phase, is to estimate the number of nanomachines via $node_c$. Each nanomachine has an initial value of molecules concentration (which can represents a certain parameter in the environment). The second phase includes number of steps, starting by each nanomachine diffuses its initial value to $node_c$; then, $node_c$ computes the average of all initial value (which is considered as an important value for reaching consensus); finally, $node_c$ diffuse the average value to the other nanomachines.

The number of nanomachines n is not known; however, an upper bound M of n is known. Thus, the exact value of n is calculated by the proposed protocol.

Two scenarios have been considered as ways of implementing the consensus protocol. In the first one, a leader election algorithm is utilized to elect $node_c$. In the second scenario, nanomachines are assumed to have an energy constraint; thus, an energy harvesting model has been defined, as a nanomachine might not be able to communicate due to the lack of energy. In each scenario, the consensus protocol's number of rounds has been counted, taking into account the required time to elect $node_c$ and the time needed to harvest enough energy.

The chapter is organized as follows. In Section 5.2 the network model and the consensus problem are described. The consensus protocol steps after electing $node_c$ are discussed in Section 5.3, with the results of an experiment to implement the protocol. In Section 5.4 the proposed protocol steps are applied into an energy constraint nanonetwork, the number of rounds needed to implement the protocol steps is estimated, and an experiment results are demonstrated. A summary of the issues discussed in the chapter is given in Section 5.5.

5.2 Model

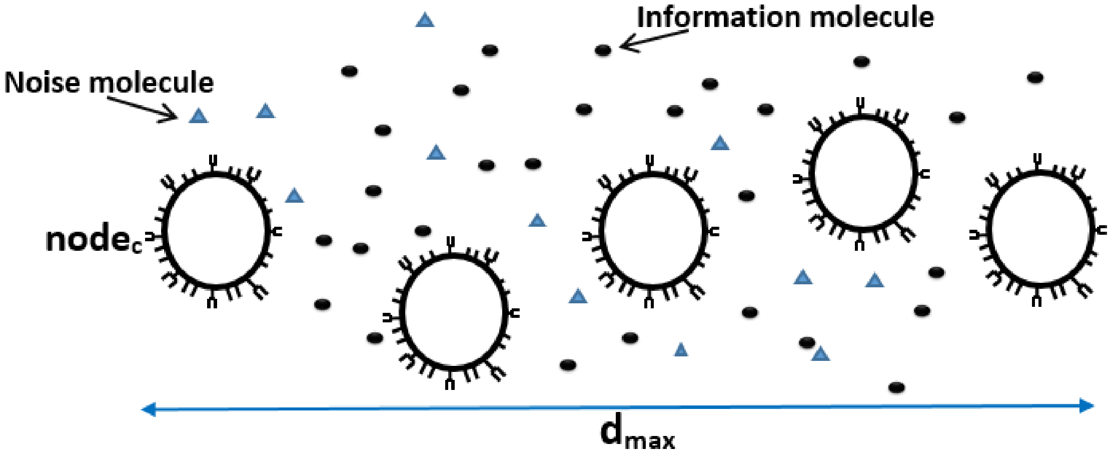


FIGURE 5.1: Model Representation

Network Environment and Communication: In this chapter a system of n nanomachines is considered, communicating according to diffusion based molecular communication 5.1. Each nanomachine $n(i)$ (where, i $\in \{1, 2, ..N\}$) has the ability to sense the concentration of molecules from the environment and to emit molecules at a particular rate into the environment, and one of these n nanomachines is considered to be a special node (leader node) $node_c$ (which has some responsibilities for directing and controlling processes of the consensus protocol), and other nanomachines in different positions from $node_c$ within its transmission range distance d_{max}. Information molecules are encoded based on the variation in the concentration of molecules in the communication medium. The communication medium might contain residual molecules from previous diffusion (as it is not necessary that all molecules to be received by the other nanomachines), and also contains molecules from other nanomachines (that are not among n) and these molecules can be considered as noise. The nanomachines are assumed to be located within a close range, so that if one nanomachine diffuses then all other nanomachines can receive some molecules.

Communication between nanomachines is based on diffusing and sensing molecules. Each nanomachine i from n can decide to diffuse, for example, a unit u of molecules at time t, and any other nanomachine j at distance d from the nanomachine i can sense the impulse of the released unit u of molecules within the interval $[t, t+T]$, through the following [125]

$$c(u, d, T) = \int_0^T u \cdot \frac{1}{(4\pi Dt)^{\frac{2}{3}}} \cdot \exp(\frac{-d^2}{4Dt}) \tag{5.1}$$

where D is the diffusion coefficient of the communication medium. If more than one nanomachine diffuses a unit u molecules, a receiver nanomachine j accumulates the sensed molecules through the summation of the values of $c(u, d, T)$ over diffusing nanomachines i, i.e., nanomachine j senses molecules in total during the interval $[t, t+T]$

$$\sum_i c(u_i, d_{(i,j)}, T_i) - c(u_i, d_{(i,j)}, [T_i - T]_+) \tag{5.2}$$

where, $d_{(i,j)}$ is the distance between nanomachines i and j, T_i is the time that passed from the diffusion of nanomachine i up to time $t+T$, i.e., nanomachine i diffused at time $(t+T-T_i)$, u_i is the unit of molecules by nanomachine i at that time, and $[T_i-T]_+$ equals to $\max\{T_i - T, 0\}$. In other words, the receiver nanomachine senses the total amount of molecules that have been in its nearest proximity in the time interval $[t, t+T]$ without being able to distinguish which molecules come from which transmitter.

If the amount of sensed molecules is greater than or equal to threshold τ, it will be considered as 1, otherwise it is 0. The curve of the function is shown in Figure 5.2.

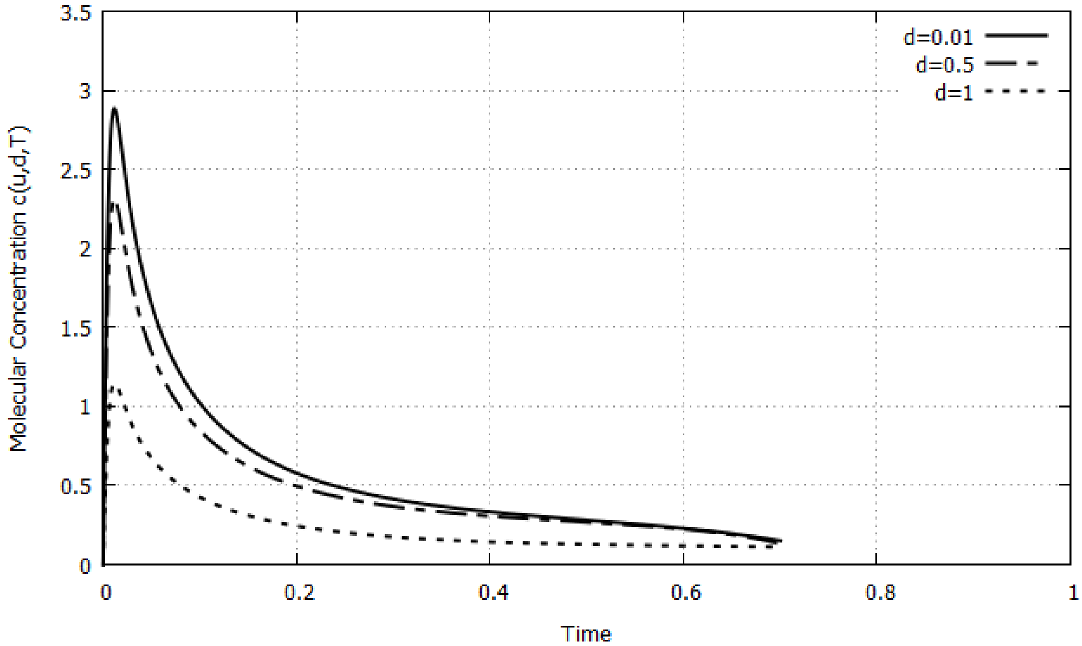


FIGURE 5.2: $c(u, d, T)$ for different distance d values

In Figure 5.2, $c(u, d, T)$ (the detected molecules) is computed, which represents the peak of molecular concentration sensed by receiver nanomachines at d= 0.01, 0.5, and 1 from a transmitter nanomachine. The transmitter nanomachine is assumed to diffuse u=1 molecule, in a medium with diffusion coefficient D=10 cm^2/sec, initial value of time t=0 (the parameters values are based on an experiment in [130]), the value of threshold is assumed to equal τ=0.5, through the experiment t is incremented by 0.01. The peak of the sensed molecular concentration is inversely proportional to the distance between transmitter and receiver nanomachines.

Consensus Problem: In the model the input of the consensus problem is the initial values of each nanomachine. Then, these nanomachines follow different processes which include the communication among them. The out put of the model problem is that each nanomachine should receive the avarage of all initial values. Thus, each nanomachine has an initial value (represents an estimation of a certain parameter in the environment). The initial values are diffused into the environment to be sensed by $node_c$. By then, the task of $node_c$ is to compute the average of all initial values, where this measurement is important and can be considered as the goal in the network. To compute the average of all initial values, $node_c$ needs to estimate the number of nanomachines; thus, this should be achieved first. As mentioned earlier, the number of nanomachines n is not known, but it assumed that an upper bound M of n is known. The exact value of n though is calculated by the proposed protocol.

The consensus problem in this chapter is inspired by the work in [59], where the nanonetwork's nodes attempt to estimate a certain parameter in the environment, the nodes communicate between each other to obtain the best estimation of that parameter. The authors in [59] study a discrete model of time epochs with length equal to T_0. The length of the epoch depends on the diffusion coefficient of the medium D and the network topology (specifically the effective radius of each node), $T_0 = k\frac{R^2}{D}$, where k is constant and R is effective radius (the authors assume a circle of radius R around each node which includes the nodes that it can effectively communicate with). In each epoch, node i estimates the environment parameter and convert this estimation into molecules concentration P_i. The authors assume that these estimated concentrations are drawn from Gaussian distribution $N(\mu, \sigma_0^2)$ with excepted value μ and variance σ_0^2. Every node in the network needs to obtain the average of initial estimates of nodes $P_{av} = \frac{1}{N}\sum_{i=1}^{N} P_i$. The differences between the proposed model here and the consensus problem discussed in [59] are :

1. In the explored problem here, the initial values are arbitrary. The estimation of the initial values is based on computing the average of all the initial values by $node_c$. Thus, the estimation of the initial values does not follow any particular distribution.

2. $node_c$ controls the process of consensus problem.

3. $node_c$ estimates the number nanomachines before the consensus protocol begin, and nanomachines estimates the effect of its distance from nod_c on the sensed molecules from $node_c$.

The proposed consensus protocol has been applied to two cases:

- In the first case, it is assumed that the nanonetwork does not include $node_c$; thus, a leader election algorithm is employed to elect $node_c$, then the consensus protocol steps proceed.

- In the second case, it is assumed that nanomachines have energy constraint. Therefore, an energy harvesting algorithm is proposed before implementing the consensus protocol steps.

Rounds: Nanomachines are assumed to be synchronized, and can communicate in predefined time round T_0. where T_0 is assumed to be a system parameter and its length depends on the network's geometric properties, in a way that $T_0 = k\frac{d_{max}^2}{D}$, where k is constant that can be equal 1, d_{max} is the transmission range distance of $node_c$, and D is the diffusion coefficient. The consensus protocol is consisting of number of different steps in different time rounds. After each step of the protocol nanomachines (n) wait at least for $k\frac{d_{max}^2}{D}$, the waiting time is needed to allow molecules to diffuse away from the communication environment.

Leader Election: In the first part of this chapter, the consensus protocol is applied on a nanonetwork without a distinguished nod_c. Thus, a leader election algorithm based on a beeping model in [70] is adopted to elect nod_c, where the authors in [70] studied the efficiency of electing a leader in a single-hop beeping network.

Energy Model: In the second part of this chapter, the nanomachine's energy is considered, and how it can affect the time at which consensus is reached, since nanomachines need energy to produce molecules for communication. Thus, an algorithm for energy harvesting is proposed, and the maximum time needed to harvest enough energy for producing a certain amount of molecules concentration is estimated.

5.3 Consensus after Electing $node_c$

In this section, consensus problem is studied in the case that the nanonetwork has no $node_c$; thus, it is reasonable to think about electing a nanomachine to be considered as the leader node. A randomized leader election algorithm proposed in [70] in a single hop network is adopted, where nodes communicate using beeps. The beeping model consists of a number of computational nodes, which interact by beeping in synchronous rounds. If a node decides to beep in a certain round, there will not be feedback from the channel to that node. Moreover, if a node chooses to listen, it can distinguish if no neighbour node in the network beeped during this round, and if one or more neighbour nodes beeped. This is similar to the cases of nanomachines in diffusion based molecular model, where it can differentiate if there are no nanomachines diffusing or one or more nanomachines diffusing at a certain time, which gives a motivation to select an algorithm from beeping model.

The leader election algorithm in [70] was proven for a given error bound $\epsilon \in \{0, 1/2\}$ and probabilistic precision g. The parameters of the algorithms were ϵ, g and $g' = min\{g, (1/\epsilon)\}$. To apply the algorithm idea in [70] into the model described in this chapter, it is fair to assume that every nanomachine is initially active and able to diffuse molecules into the environment. Through the rounds of the algorithm's loop, each nanomachine diffuses molecules with a probability $\frac{1}{g'}$, otherwise, it senses (listen to channel) molecules from the surrounding environment. If a nanomachine ever receives molecules (sense molecules above threshold τ), it is no longer active and would not diffuse during the remaining of the algorithm's loop. When the rounds of the algorithm are terminated, any nanomachine that remains active becomes the leader node i.e. $node_c$.

Algorithm 9: Leader Election-Leader Node $node_c$

```
$active \leftarrow 1$ ;        // initially all nanomachines can diffuse molecules
$h \leftarrow 1$;
$g' \leftarrow min\{g, (1/\epsilon)\}$;
$elect \leftarrow$ [Termination.Subroutine]$(active, h)$;
$h \leftarrow 0$;
while (not elect) do
    $participate \leftarrow random_{bit}(1/g')$ ;        // with probability $1/g'$
                                          // nanomachines can diffuse molecules

    $SenseEnvironment \leftarrow 1$ ;   // loop continues as environment contains
                                          // diffused molecules

                    // The following is- Knocking Out from the loop part
    if $active \wedge participate$ then
        $diffuse$ u ;             // nanomachine diffuses unit of molecules
    else
        $SenseEnvironment \leftarrow recv()$ ;    // nanomachine receives molecules
    if $active \wedge$ not $participate$ then
        if $SenseEnvironment = 1$ then
            $active \leftarrow 0$;
            $h \leftarrow 1$;
    if $SenseEnvironment = 0$ then          // Termination Detection part
        $elect \leftarrow$ [Termination.Subroutine]$(active, h)$;
        $h \leftarrow 0$;
if $active$ then                       // Active nanomachine would be Leader
    $leader \leftarrow 1$;
else
    $leader \leftarrow 0$;
return($leader$)
```

Algorithm Analysis The algorithm includes a *termination subroutine*, which returns *true* if there is $node_c$ (a leader), and *false* otherwise. This subroutine is called by all nanomachines simultaneously. The *termination subroutine* involves passing two parameters, the first is the value of *active*, which indicates whether or not the calling nanomachine is still competing to become $node_c$ (a leader). The second parameter is h, which indicates whether or not the nanomachine has been knocked out from the main loop in the algorithm.

The authors in [70] fixed a parameter $R = 4log_{g'}(max(n, 1/\epsilon))$, which represents a bound on the number of calls needed in the subroutine before the algorithm is possibly terminated, where n represents the number of nanomachines in the nanonetwork, and ϵ is an error bound $\in [0, 1/2]$. However, one of the assumptions in this model states that the number of nanomachines is not known. Thus, the upper bound M of n can be used in representing the value of R.

This leader election algorithm can be correct if combined with a termination subroutine which fulfils a number of properties that can be defined with respect to ϵ and R: It is possible to have *safety*, so that through the first R calls, the probability that in call it returns true with more than one active nanomachine is at most $\epsilon/2$. The other property is *eventual termination* if the subroutine has been called infinitely with only one active nanomachine, eventually it returns true, with probability 1. *Fast termination* is possible if the subroutine is called with only one active nanomachine and with at least one nanomachine, where $h = 1$ then the subroutine would return true.The knockout loop in the algorithm is considered to be silent if no nanomachine diffuses during it, where the termination routine is executed only in silent iterations of the knockout loop. The important thing now is to determine how long is needed to reduce the number of active nanomachines.

The proof of Lemma 3 in [70] shows that at least one active nanomachine exists, as any nanomachine becomes inactive only if it detects diffused molecules above the threshold τ from another active nanomachine. It has been shown that while there are at least 2 active nanomachines, the probability of R silent iterations of the knockout loop is bounded by n^2/g'^R. The calculated time cost for each iteration of the knockout loop is $t + O(1)$ time.

If $n > 1$, also from Lemma 3, it is known that with at least $1 - \epsilon$ probability, there is only one active nanomachine within $O(tR)$ time. Taking into account the last round which begins with at least two active nanomachines, during this round, at least one nanomachine is knocked out. Consequently when there is only one active nanomachine and at least one nanomachine with h equal to true, then the termination routine is called. Thus, all nanomachines terminate after this call with consideration to the fast termination property. Theorem 4 in [70] can be applied to prove that the algorithm is correct leader election algorithm.

5.3.1 Proposed Consensus Protocol

5.3.1.1 Estimate the Number of Nanomachines

After electing $node_c$, it is required to estimate the number of nanomachine in the network before the consensus protocol starts. Thus, $node_c$ diffuses a unit u of molecular concentration which equals a global parameter known for all the nanomachines in the network. The other nanomachines $n(i)$ should respond to this by diffusing the same amount of molecular concentration. Then, $node_c$ accumulates the total diffused molecules and divides it by the value of u, to estimate the number of nanomachines. Thus, during the first round, $node_c$ diffuses a general parameter u, and the other nanomachine $n(i)$ sense the diffused parameter u through equation 5.1. The steps during the first round are described in the following algorithm (which are considered based on Algorithm 6):

Algorithm 10: Sensed molecular concentration after $node_c$ diffuses u

```
Input: Initial values of 5.1 parameters
Output: Molecular concentration sensed by each of $n(i)$
$t \leftarrow 0.03$ ;        // time consumed from the round in diffusing by $node_c$
while $(c(u, d_i, T_0) \leq \tau)$ & $(t \leq T_0)$ do
    $c(u, d_i, T_0) = \int_{t=0.3}^{T} u \cdot \frac{1}{(4\pi Dt)^{\frac{2}{3}}} \cdot \exp(\frac{-d_i^2}{4Dt})$;
    $t = t + \delta$
```

During the first round, it is assumed that $node_c$ continues to diffuse unit u of molecules during the first 0.3 millisecond of the length of the first round. Thus, in the Algorithm 10 the initial value of t is assumed to equal 0.3. During the rest of the first round, each nanomachine $n(i)$ at distance d_i attempts to sense the diffused unit u, and continues sensing as long as the amount of molecules is larger or equal to the threshold τ and the value of t is less than the length of T_0. Thus, d_i represents the distance of each specific nanomachine $n(i)$ from $node_c$.

The sensed molecular concentration $c(u, d_i, T_0)$ of each nanomachine $n(i)$ is considered to be an estimation of the diffused unite u molecules, i.e., $u_{estimate}$. Each nanomachine from $n(i)$ should diffuse its estimation $u_{estimate}$ to $node_c$, during the next round; however, $node_c$ senses the accumulative molecular concentration of all the diffused $u_{estimate}$. During the first round $node_c$ diffused u and the other nanomachines $n(i)$ compute $u_{estimate}$. The second round is assumed to be a waiting round, so that the molecules diffuse away from the communication environment. Thus, during the third round the other nanomachines $n(i)$ diffuse their estimations $u_{estimate}$ to $node_c$ based on the following steps:

Algorithm 11: Sensed $u_{estimate}$ by $node_c$

```
$t \leftarrow 0.03$ ;        // time consumed from the round in diffusing by $n(i)$
while $(t \leq T_0)$ do
    $u_{estimate} = u_{estimate} + (c(u_{estimate}, d_i, t) - c(u_{estimate}, d_i, [t - T_0]_+))$
    $t = t + \delta$;
```

Each nanomachine of $n(i)$ is assumed to diffuse its $u_{estimate}$ during the first 0.3 millisecond of the length of the third round. During the rest of the third round length nod_c attempts to senses the accumulative molecular concentration of all the diffused $u_{estimate}$. The value of d_i in the algorithm represents the distance of each specific $n(i)$ from $node_c$.

During the first round, depending on the distance from $node_c$, the other nanomachines $n(i)$ might sense molecular concentration less than u. Thus, when the other nanomachines $n(i)$ diffuse their estimations $u_{estimate}$ during the third round, $node_c$ might not receive the same amount of molecular concentration it diffused, i.e., u. Eventually, $node_c$ can't estimate the number of nanomachines.

Figure 5.3 shows the results of diffusing u by $node_c$, and the sensed molecular concentration by the other nodes $n(i)$. Even though $node_c$ senses the accumulative molecular concentration when the other nanomachine $n(i)$ diffuse their estimation $u_{estimate}$. In this experiment it is assumed that $node_c$ senses molecular concentration from each nanomachine $n(i)$ separately. In order to check the sensed molecular concentration by $node_c$ when each nanomachines $n(i)$ diffuses its $u_{estimate}$.

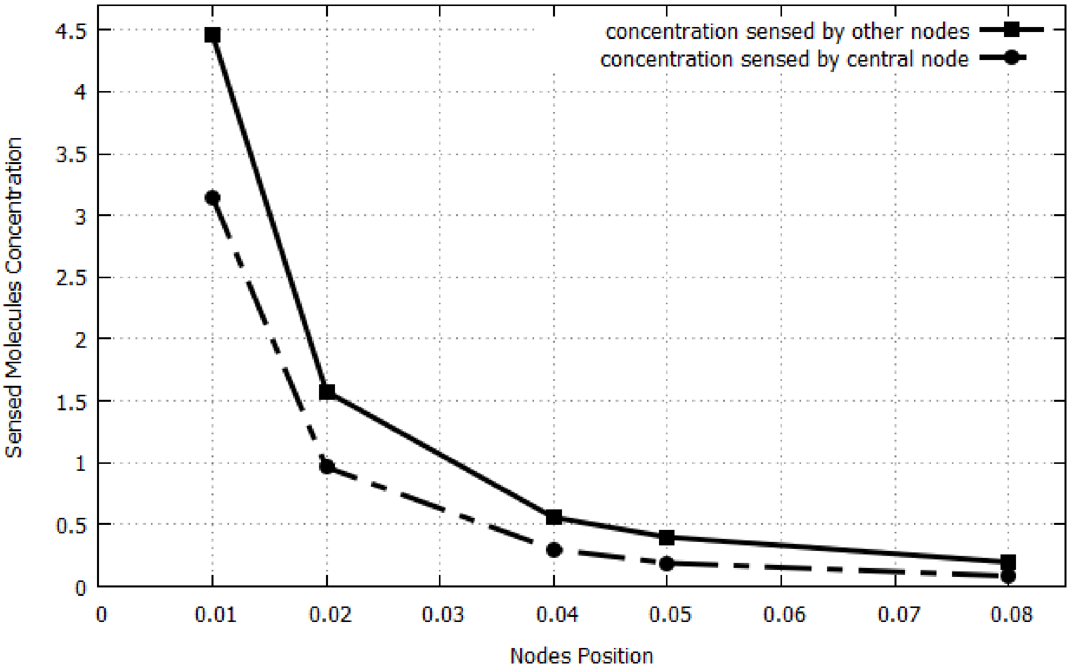


Figure 5.3: The sensed molecular concentration of u by $n(i)$ and the sensed molecular concentration of $u_{estimate}$ by nod_c

Figure 5.3 represents an experiment to implement diffusing/sensing of u and $u_{estimate}$ respectively, on a nanonetwork consisting of five nanomachines in different distances from $node_c$. where the transmission range of $node_c$ is d_{max}= 0.1, time slot length T_0= 0.1 millisecond, and the initial value of time t is 0.01. The nanomachines are assumed to be at distances (0.01, 0.02, 0.04, 0.05, and 0.08) from $node_c$. The number of diffused units by $node_c$ is u= 2×10^9 molecules, the value of the threshold τ= $2 \times 10^{16} molecules.cm^{-3}$, and the diffusion coefficient $D = 10^{-16} cm^2/sec$ (those parameters are based on experiments in [125]. In Figure 5.3 the x-axis represents the position of nanomachines $n(i)$ at certain distances from $node_c$, and the length of the x-axis equals the maximum transmission range d_{max}. The y-axis represents the amount of sensed molecules. When $node_c$ diffuses unit u of molecules, the first line in Figure 5.3 represents the amount of sensed molecules $u_{estimate}$ by the other nanomachines $n(i)$, where each point in the line represents the sensed molecules of nanomachine $n(i)$ at distance d_i. The dashed line in the figure represents the amount of sensed molecules by $node_c$ when the other nanomachines $n(i)$

diffuse their estimation $u_{estimate}$. It is possible to observe that the two lines in Figure 5.3, seem to be quite symmetrical. This could be due to the identical values of some of the used parameters. However, the difference is mainly in the amount of diffused molecules, where $node_c$ diffuses u of molecules, and the other nanomachines $n(i)$ diffuse their estimations $u_{estimate}$ which is less than u and could vary based on the distance of the nanomachine $n(i)$ from $node_c$.

Thus, the other nonomachines $n(i)$ expect to sense u molecular concentration, but it sense $u_{estimate}$ depending on its distance from $node_c$, as figure 5.3 shows. Each nanomachine from $n(i)$ attempts to compute the effects of its distance from $node_c$, to ensure that, when it diffuses u molecular concentration, $node_c$ senses approximately u molecules concentration. Thus, before diffusing back to $node_c$, each nanomachine $n(i)$ aims to increase the amount of its diffused molecules with a specific value $\rho(i)$.

Where:

$$\frac{1}{\rho(i)} = \frac{u_{estimate}(i)}{u} \tag{5.3}$$

Thus, the amount of molecules that the other nanomachines would diffuse depends on its ρ. The value of ρ can vary from one nanomachine to another depending on its distance from $node_c$. This means, each nanomachine from $n(i)$ diffuses:

$$u_{n(i)} = \rho(i) \times u \tag{5.4}$$

i.e. $u_{n(i)}$ is approximately u when it reaches $node_c$, $\rho \times u \approx u$. Thus, during the third round, each nanomachine $n(i)$ diffuses it $u_{n(i)}$ to $node_c$, through the similar steps described in Algorithm 10.

Meanwhile, $node_c$ accumulates the sensed molecules from the diffused $u_{n(i)}$, in order to get u_{total} by the end of the round. Then, $node_c$ estimates N which equals the number of nanomachines $n(i)$, by dividing the value of the total units it has received u_{total} by the value of unit u, where: $N = \frac{u_{total}}{u}$.

Thus, by now $node_c$ has estimated N, and each nanomachine from $n(i)$ has distinguished the effects on receiving the right amount of molecular concentration diffused by $node_c$ which equals $\rho(i)$.

5.3.1.2 Consensus Protocol Steps

After $node_c$ has estimated N, the consensus protocol steps can be initiated. The consensus protocol includes a number of steps in different time rounds. Each nanomachine $n(i)$ has an initial value, which represents an estimation of a certain parameter in the environment. In order to avoid the Inter Symbol Interference, it is assumed that there are waiting rounds. These waiting rounds are utilized so that molecules diffused a way from the communication environment. Thus, the 4th and 6th rounds are waiting rounds. The consensus protocol steps start from round five, can be explained as follows:

1. Each nanomachine from $n(i)$ diffuses its initial value to $node_c$ during the fifth round, in such a way, that $node_c$ senses almost the same diffused molecules concentration amount (i.e. the initial value of each $n(i)$). Thus, if $\zeta(i)$ is the initial value of a nanomachine from $n(i)$, this means, it should diffuse:

$$\zeta(i) \times \rho(i) \tag{5.5}$$

   to ensure that at least $\zeta(i)$ molecular concentration reaches $node_c$. Thus, $node_c$ shall follow the steps described in Algorithm 11 in order to sense the initial values of $n(i)$.

2. By the end of the fifth round, $node_c$ computes the average of all initial values:

$$\omega_{av} = \frac{\sum\limits_{i=1}^{N} \zeta(i)}{N}. \tag{5.6}$$

3. Throughout the seventh round, $node_c$ diffuses the average of all initial values ω_{av}, while the other nodes attempt to sense the diffused molecules in order to recognize the average of all initial values. Each nanomachine from $n(i)$, identified the effects of distance on the received concentration from $node_c$, before the consensus protocol steps started. Therefore, each nanomachine $n(i)$ would count on its $\rho(i)$ to distinguish the right value of ω_{av}:

$$\frac{1}{\rho(i)} = \frac{\omega_{av}(i)_{received}}{\omega_{av}} \tag{5.7}$$

   Thus:

$$\omega_{av} = \rho(i) \times \omega_{av}(i)_{received} \tag{5.8}$$

   Thus, in the steps of this consensus protocol, it is assumed that when the nanomachines are communicating, all the computed values of molecular concentration are exact values. Since the nanomachines utilize its $\rho(i)$ value to recognize the exact molecular concentration.

Time to Reach Consensus To compute the time needed to achieve the average of all initial values, it requires that the time needed for terminating Algorithm 9 and electing $node_c$ to be added.

$$t_{\omega_{av}} = \text{length of round} \cdot R(t + O(1)) + \text{length of round} \cdot \text{number of rounds} \tag{5.9}$$

Recall that $R(t + O(1))$ is the time needed to elect a leader node, i.e., $node_c$. To elect $node_c$ using the adopted leader election algorithm, the bound number of the subroutine calls before it possibly terminated is R, and time cost needed of each iteration of the knockout loop is $t + O(1)$, so total time is $R(t + O(1))$. The length of each round as mentioned earlier equals $T_0 = k\frac{d^2_{max}}{D}$. The number of rounds in the proposed consensus protocol are 7. Thus, $t_{\omega_{av}}$ would be:

$$t_{\omega_{av}} = T_0 \cdot R(t + O(1)) + 7T_0$$

Equation 5.3.1.2 represents the total needed time to complete the proposed consensus protocol. It shows that the time needed to elect a leader node dominates the whole computation.

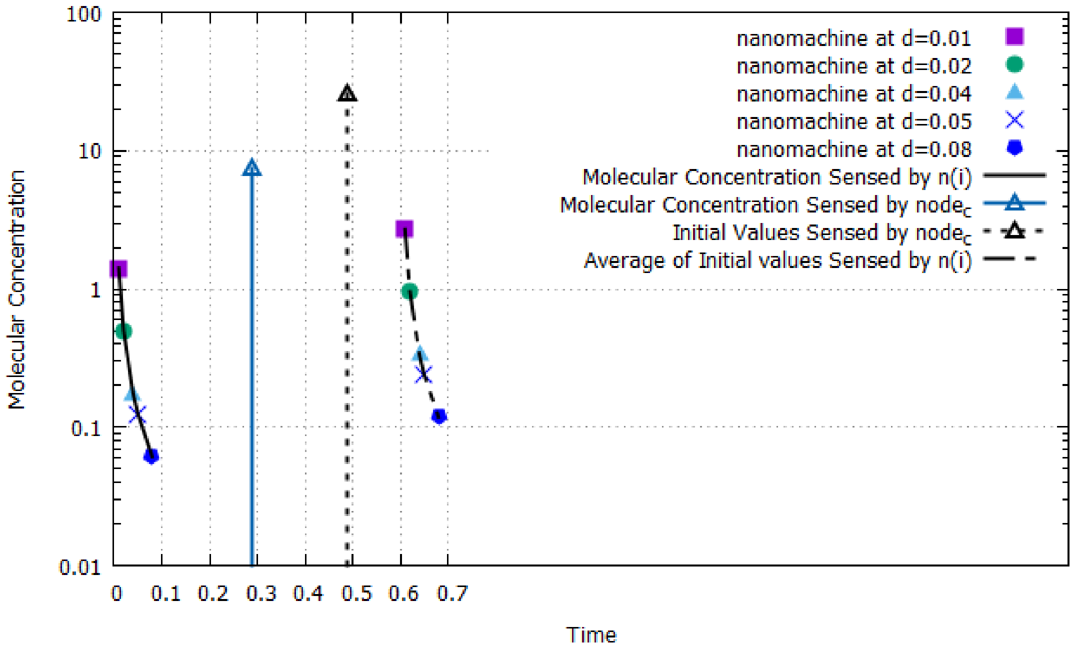


FIGURE 5.4: Consensus Protocol Steps

Figure 5.4 represents an experiment of the protocol steps on a nanonetwork consists of $node_c$ and five other nanomachines at distances (0.01,0.02,0.04,0.05,0.08) from $node_c$. The values of: $\varepsilon \times \mu = 2 \times 10^{14}$, D=1×10^{-16}, u=2×10^{12}, max_d=0.1, T_0=0.1. During the first round each nanomachine senses the diffused u molecular concentration according to their distances from $node_c$. The results are in form of $1.0e + 6$, but in Figure 5.4 they have been rounded. The second round is a waiting round. During the third round $node_c$ sense the accumulative molecular concentration diffused by the other nanomachines, the results are in form $1.0e + 14$ (as the other nanomachines raised their diffused molecular concentration), in the figure the results have been rounded. In the fifth round $node_c$ sense the accumulative molecular concentration of the diffused initial values related to each of the other nanomachines. The initial values of each nanomachine are: (1×10^8, 1×10^7, 1×10^9, 1×10^7, 789044344) respectively. Finally, during the seventh round, each nanomachine senses the average of all initial values depending its distance from $node_c$. The x-axis represents the sensed molecular concentration, and y-axis denotes the time duration of each round. The length of each round is 0.1, initially t=0.01, and through the process of diffusing or senseing it is incremented by 0.01, until it equals T_0, i.e., 0.1. The ticks on y-axis represents the utilized rounds to implement the protocol steps.

5.4 Consensus in Energy Constraint Nanonetwork

Considering the same model of n nanomachines, including $node_c$, these nanomachines have an energy constraint, which means that a nanomachine might not be able to produce its estimated rate of molecules due to a lack of energy, and it has to wait in order to harvest energy from resources (by absorbing molecules in the surrounding environment). Consequently there might be a delay in reaching consensus in the nanonetwork.

In this section, it is assumed that each nanomachine has a buffer of energy, part of it is used for routine activities and the rest for communication purposes (where a nanomachine would be able to generate molecules and diffuse them). Molecules can be counted as resources for nanomachines; thus, generating and diffusing new information molecules by nanomachines require resources harvesting. The idea is inspired by the harvesting mechanisms in [45, 46] which is accomplished by absorbing discrete particles into the nanomachine's buffer. A resource can be found either inside a nanomachine's reservoir (in a pure resource form) or at large in the operating environment. Resources (molecules) in the operating environment can have two different forms, either as information molecules (represent the communication molecules), or as other types of molecules which are considered to be noise; thus, nanomachines can harvest energy by absorbing molecules of either forms.

Each nanomachine is assumed to have an energy buffer b, which is divided into two parts: one part of the energy is used for routine background activities (living operations-from a biological perspective) b_l, and the rest of the energy b_c is utilized for communication purposes, where

$$b_c = b - b_l \tag{5.10}$$

For simplicity, in energy harvesting the focus is on the communication part b_c of the energy buffer, and assume that the part for routine background activities b_l is constant and not affected by the energy constraint. If b_c of a nanomachine is less than β, then it will not be able to produce molecules, where: β is the minimum required energy for communication. In this case the nanomachine needs to wait for sometime to harvest energy. As [45] suggested that in order for a nanomachine to diffuse Q molecules, it should have already harvested at least Q resources from the environment, assuming that $b_c = 0$, where: Q can be considered as the number of molecules a nanomachine intend to diffuse.

The number of absorbed molecules per time unit can be assumed to be m. Thus, the time required to harvest enough energy for communication can be computed as follows:

Algorithm 12: Node's Energy Harvesting

```
$t_{harvest} \leftarrow 0$;
while ($b_c \leq \beta$) do
    $b_c \leftarrow b_c + m$;
    $t_{harvest} \leftarrow t_{harvest} + 0.1$;
```

Algorithm 12 assumes that when the nanomachine's communication energy b_c is less than the minimum required energy β to generate a message (information molecules) and that $b_c < (b - b_l)$, the nanomachine will attempt to harvest energy by absorbing molecules, and as assumed, the amount of energy harvesting per single time unit of $t_{harvest}$ is m, and time unit is assumed to equal to $0.1ms$.

One of the main concerns is the length of time round T_0 and whether a nanomachine needs longer time to harvest energy, which could exceed the length of one round. The value of $t_{harvest}$ represents the required time to harvest enough energy, i.e., for a nanomachine with b_c=0, $t_{harvest}$ is the time to reach a point where $b_c = b - b_l$. As a nanomachine can absorb m resources per time unit, it is possible to assume that $t_{harvest} \leq (b - b_l)/m$. In general, it is possible to assume that in algorithm 12, a nanomachine is able to harvest energy within $O(b - b_l)$.

5.4.1 Consensus Protocol

5.4.1.1 Estimating the Number of Nanomachines

The consensus protocol starts by estimating the number of nanomachines by $node_c$, as explained in the previous section. Thus, during the first round, $node_c$ diffuses a unit u of molecular concentration, where u equals a global parameter that is known for all the nanomachines. Based on Equation (5.1), as with the steps described in Algorithm 10, $node_c$ diffuses u and the other nanomachines $n(i)$ sense molecular concentration according to their distance from $node_c$.

The other nanomachines $n(i)$ estimate u according to the sensed molecular concentration to compute $u_{estimate}$. As explained in the previous section, the other nanomachines $n(i)$ might sense molecules less than u, depending on their distance from $node_c$, as shown in Figure 5.3. However, most importantly the difference here is that nanomachines $n(i)$ might not have enough energy to diffuse what it received ($u_{estimate}$) in one round, and possibly need to harvest energy through Algorithm 12.

Starting from the third round, each nanomachine from $n(i)$ should diffuse $u_{n(i)}$ molecular concentration to $node_c$. In the meantime $node_c$ accumulates the sensed molecules from the diffused $u_{n(i)}$ to compute u_{total}. Due to energy constraint, nanomachines $n(i)$ might need more than one round to be able to diffuse $u_{n(i)}$; subsequently, $node_c$ needs to wait longer to estimate N which represents the number of nanomachines $n(i)$.

Algorithm 13 shows the steps that each $n(i)$ follows to diffuse $u_{n(i)}$ to $node_c$. It is assumed that the maximum amount of molecular concentration $n(i)$ can diffuse when $b_c = (b - b_l)$ is $sub_{u_{n(i)}}$. The initial amount of molecular concentration $n(i)$ can diffuse is sub.

Algorithm 13: Steps to diffuse $u_{n(i)}$ by the other nanomachines $n(i)$ to $node_c$

```
$less \leftarrow 0$;
$no._{round} \leftarrow 0$ ;  // to count rounds, in case energy harvesting is needed
                    // $sub_{u_{n(i)}}$ molecular concentration generated per full $b_c$
                                        // $sub$ initial amount that $n(i)$ can diffuse
while $(b_c < (b - b_l))$ & $(sub < u_{n(i)})$ do     // if $n(i)$ has no enough energy
    Call Node's Energy Harvesting Algorithm;
                              // nanomachine at $d_i$ from $node_c$ diffuses $sub_{u_{n(i)}}$
    if $t_{harvest} \geq T_0$ then      // if harvesting time longer than one round
        $no._{round} = no._{round} + 1$;
    else if $t_{harvest} + t \geq T_0$ then
                                    // t is time consumed in diffusing $sub_{u_{n(i)}}$
        $no._{round} = no._{round} + 1$;
    else
        $less \leftarrow 1$ ;                                   // in case $t_{harvest}$ < $T_0$

                              // nanomachine at $d_i$ from $node_c$ diffuses $sub_{u_{n(i)}}$
                                                // after harvesting enough energy
    $t \leftarrow 0.03$ ;     // time consumed from the round length in diffusing
                                                    // $node_c$ starts sensing $u_{n(i)}$
    $c(u_{n(i)}, d_i, T_0) = \int_{t=0.03}^{T_0} u_{n(i)} \cdot \frac{1}{(4\pi Dt)^{\frac{2}{3}}} \cdot \exp(\frac{-d_i^2}{4Dt})$
    $t = t + \delta$;
    $sub = sub + sub_{u_{n(i)}}$;
```

In the situation that a nanomachine from $n(i)$ does not have enough energy to produce molecules in order to diffuse $u_{n(i)}$, it follows the steps in Algorithm 12 to harvest energy.

With energy constraint, diffusing $u_{n(i)}$ might require more than one round; thus, a parameter $no._{round}$ is set to count the likely overall number of rounds, by checking when $t_{harvest}$ is larger than T_0.

Diffusing $u_{n(i)}$ starts from the third round, but depending on the energy of each nanomachine of $n(i)$, it could require longer than one round. As it is assumed in Algorithm 12, that the number of absorbed molecules per time unit is m, in other words a nanomachine would harvest energy to produce m molecules per time unit. Thus, the time needed to harvest energy to diffuse $u_{n(i)}$ is:

$$t_{harvest1} \leq \left\lfloor \frac{u_{n(i)}}{m} \right\rfloor \tag{5.11}$$

Throughout the processes explained in Algorithm 13 considering a general example in which a nanomachine from $n(i)$ has an energy buffer $b_c = 0$. If the number of times which this nanomachine from $n(i)$ needs to harvest energy in order to diffuse $u_{n(i)}$ is assumed to be on average $w1$. Where $w1$ is assumed to be an external parameter. The total time needed to diffuse $u_{n(i)}$ is likely to be:

$$t_{total1} \leq t_{harvest1} \cdot w1 \tag{5.12}$$

Thus, to compute the number of rounds needed to diffuse $u_{n(i)}$ by an average nanomachine with energy constraint, it is possible to assume that:

$$roundTotal1No. \leq \left\lfloor \frac{t_{total1}}{T_0} \right\rfloor \tag{5.13}$$

Hence, to compute u_{total}, in section 5.3 $node_c$ waits till the end of the third round. However, here with energy constraint as explained earlier, diffusing $u_{n(i)}$ in average can last at least for $roundTotal1No.$. Thus, $node_c$ waits at least for $roundTotal1No.$ to accumulate the sensed molecules from the diffused $u_{n(i)}$, and compute u_{total}. Then, $node_c$ estimates N which represents the number of nanomachines $n(i)$, through dividing the value of the total units it had receive u_{total} by the value of unit u, where: $N = \frac{u_{total}}{u}$. Energy harvesting is also needed for $node_c$, though it invests the waiting time $roundTotal1No.$ and the sensed molecules $u_{n(i)}$ to harvest enough energy.

5.4.1.2 Consensus Protocol Steps

Although the steps of the consensus protocol are similar to those explained in section 5.3; However, the energy constraints could affect the time and number of rounds needed to reach consensus.

Each nanomachine from $n(i)$ has an initial value, which represents an estimation of a certain parameter in the environment. The initial value of each nanomachine should be diffused to $node_c$. All nanomachines $n(i)$, before diffusing their initial value, it would attempt to increase the value of the diffused molecular concentration, so that $node_c$ can sense a concentration level that amounts to the initial values of each $n(i)$, as in equation 5.5. Due to energy constraint, some nanomachines might not have enough energy to diffuse this amount $(\zeta(i) \times \rho(i))$; thus, each nanomachine from $n(i)$ follows the steps described in Algorithm 13, in order to harvest energy while diffusing its raised initial value $(\zeta(i) \times \rho(i))$.

In Algorithm 12, the number of absorbed molecules per time unit is assumed to equal m, which means a nanomachine would harvest energy to produce m molecules per time unit. Thus, in general; the time needed to harvest energy to diffuse $(\zeta \times \rho)$ is as follows:

$$t_{harvest2} \leq \left\lfloor \frac{\zeta \times \rho}{m} \right\rfloor \tag{5.14}$$

According to Algorithm 13, if a general example of a nanomachine from $n(i)$ with energy buffer $b_c = 0$ is considered. Assuming that the number of times which this nanomachine from $n(i)$ needs to harvest energy in order to diffuse $(\zeta(i) \times \rho(i))$ on average can be $w2$ (which is assumed to represent an external parameter). So that the total time needed to diffuse this amount is likely to be:

$$t_{total2} \leq t_{harvest2} \cdot w2 \tag{5.15}$$

In order to compute the number of rounds needed to diffuse $\zeta \times \rho$ for an average nanomachine with energy constraints, it is assumed that:

$$roundTotal2No. \leq \left\lfloor \frac{t_{total2}}{T_0} \right\rfloor \tag{5.16}$$

Meanwhile, $node_c$ senses and accumulates the molecules by following the steps described in Algorithm 11 to compute the average of all the initial values through equation 5.6. However, the waiting time of nod_c is at least $roundTotal2No.$ according to equation 5.16.

Next, during the final step of the protocol, $node_c$ diffuses the average of all initial values ω_{av}, while the other nodes attempt to sense the diffused molecules, in order to recognize the average of all initial values. Before the consensus protocol steps are begun, each nanomachine from $n(i)$ has identified the effects of distance on the received concentration from $node_c$. Thus, each nanomachine $n(i)$ counts on its $\rho(i)$ to distinguish the right value of ω_{av}, as stated in equation 5.8.

Time to Reach Consensus To compute the total time to reach consensus in a nanonetwork with energy constraint, the required time for $node_c$ to estimate N, plus the needed time to compute ω_{av} should be taking in to account, where the nanomachines $n(i)$ has to invest some time for energy harvesting, during both processes. Adding to these, two rounds; one represents the first round in which $node_c$ diffused u, and one represents the last round in which $node_c$ diffused ω_{av}.

$$t_{\omega_{av}} \leq T_0 \cdot roundTotal1No. + T_0 \cdot roundTotal2No. + 2T_0 \tag{5.17}$$

Thus, with energy constraint, the number of rounds in the proposed consensus protocol will increase, due to the waiting time for energy harvesting, as explained above.

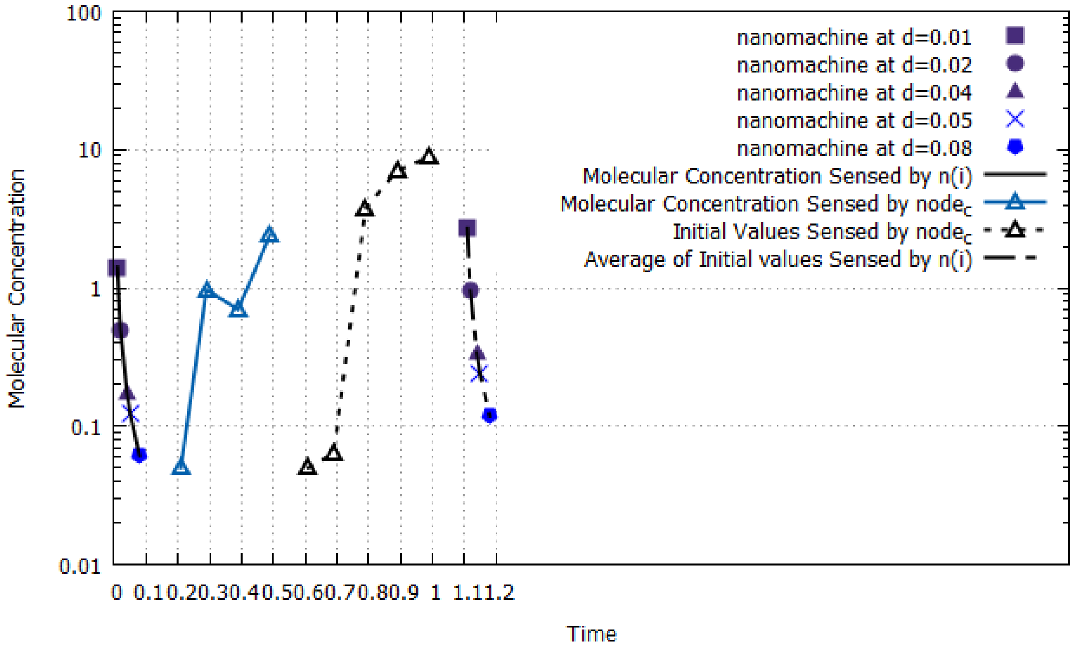


Figure 5.5: Consensus Protocol Steps in Energy Constraint Nanonetwork

Figure 5.5 represents an experiment of the protocol steps in a nanonetwork with energy constraint nanomachines. The parameters values in this experiment and the results are the same as in previous experiment in Figure 5.4. However, the difference is in the required time for nanomachines to diffuse their estimation and initial values; thus, more than one round has been utilized to diffuse these values. Based on the nanomachines energy and their initial values, nod_c senses the accumulative molecular concentration by the end of different rounds. The summation of the accumulated molecular concentration on each round would almost equal the molecular concentration that nod_c sensed in Figure 5.4 during round 3 and 5 respectively.

5.5 Summary

A protocol to reach consensus among nanomachines communicating via diffusion has been proposed in this chapter. In order to pursue the consensus protocol, it has been applied first on a nanonetwork which required a leader election algorithm initially. Then effects of nanomachine's energy constraint on the consensus protocol implementation have been studied. In both cases, the time needed to reach consensus, taking into account how long is required to elect a leader nanomachine, and the duration required to harvest enough energy, has been computed.

Chapter 6

Verification of Consensus Protocol for Diffusion based Molecular Communication

6.1 Introduction

This chapter considers a model of a nanonetwork with n nanomachines, that are arranged in a 2D network topology. These nanomachines are communicating through diffusing and sensing information molecules. The diffused molecules are propagated across the medium according to a stochastic process of Brownian motion. A protocol is proposed to study the consensus problem in such a model. Then, the proposed protocol is verified using PRISM model checker, on a grid of $(n \times n)$ nodes, taking into account deterministic and probabilistic cases. Consensus problem proposed in [59] is verified using PRISM model checker by considering a general case of $(n \times n)$. This chapter is organized as follows: In Section 6.2, a detailed description of the model is given. Then, the proposed consensus protocol steps are introduced in Section 6.3. This is followed by the model verification results using PRISM for different experiments on different grid sizes in Section 6.4. Section 6.5 includes PRISM verification results of the consensus model described in [59]. A summary of the issues discussed in the Chapter is provided in Section 6.6.

6.2 Model

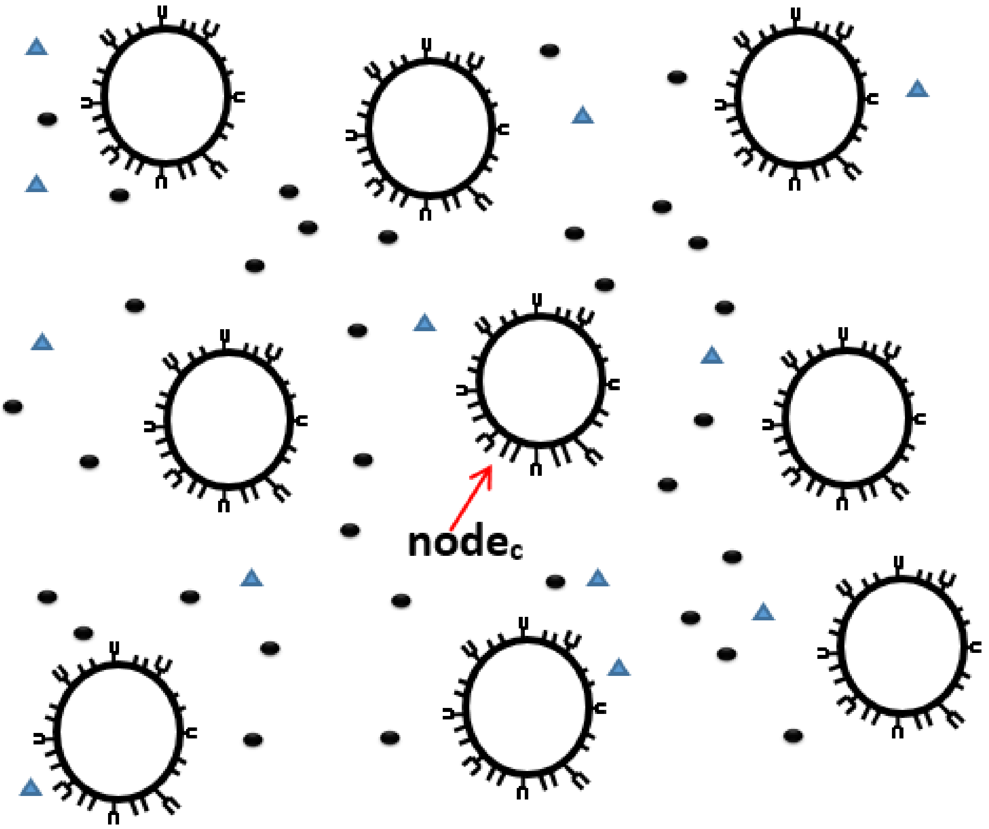


FIGURE 6.1: Model Representation

Network Environment: A system of $(n \times n)$ nanomachines is considered, where these nanomachines communicating through a shared, unguided medium by stipulating and controlling diffusion process Figure 6.1. Each nanomachine $n(i, j)$, (where $i, j \in \{1, 2, 3, \cdots, n\}$) has the ability to sense the concentration of molecules from the environment and to emit molecules at a particular rate into the environment. One of these $(n \times n)$ nanomachines is considered as a central node $node_c$, and the other nodes $n(i, j)$ are distributed arbitrarily within the central node's transmission range. The medium of communication might contain residual molecules from previous diffusion, and also molecules from other nanomachinse (that are not among $(n \times n)$) which can be considered as noise. These nanomachines are assumed to be placed within a close range so that if one diffuses then all other nanomachines can receive some molecules.

Communication Model: As it been described in Chapter5 in Section 5.1, the communication between nanomachines is based on diffusing and sensing molecules. When a nanomachine diffuses a unit u of molecular concentration into the propagation medium, these molecules are assumed to spread freely, and their dynamics can be described by the Brownian motion. The molecular concentration at the receiving points in response to the impulse of the released unit of molecules u from the transmitter can be found by

Equation (5.1). If more than one nanomachine diffuses a unit u of molecules, a receiver nanomachine j accumulates the sensed molecules through the summation of the values of $c(u, d, T)$, as in Equation (5.2).

Time slot: Nanomachines are assumed to be synchronized, and can communicate in a predefined time epoch T_0. Where T_0 is a system parameter and its length depends on the network topology, $T_0 = k\frac{d_{max}^2}{D}$, where k is constant, and can be equal 1, d_{max} is the transmission range distance of the central node $node_c$, and D is the diffusion coefficient.

Consensus Protocol: The proposed protocol includes a number of processes throughout different time epochs, starting in diffusing a unit u of molecules by the central node $node_c$ during the first epoch. Where, u represents the initial value of $node_c$. The other nanomchines will attempt to estimate u, and diffuse their estimations to the central node $node_c$ during the next epochs. Subsequently the central node checks whether the system reaches consensus via a few computations. It is assumed that consensus is reached if the average estimation value approaches to the value of u with a deviation of ε. The consensus problem in Chapter 5 is different from the consensus problem discussed in Chapter 6 mainly in the model and the assumed approach to reach consensus. In Chapter 5 it is assumed that $node_c$ computes the average of the other nodes initial values. The number of nanomachines is assumed to be computed by $node_c$ in Chapter 5. Nanomachines utilize its $\rho(i)$ value to recognize the exact (diffused/sensed) molecular concentration in Chapter 5. In Chapter 6 the idea of the proposed consensus protocol is inspired by a two-phase commit protocol, and the topology of the assumed model is different. The other nanomachine $n(i)$ are assumed to compute the average of $node_c$ initial value in Chapter 6. Reaching consensus in Chapter 6 have been verified using PRISM model checker.

Verification using PRISM: In diffusion based molecular communication, the molecules propagation in the environment is a stochastic process, as it is a consequence of spontaneous diffusion [109]. Besides this; noise in the medium can affect sensing the diffused molecules, and Equation (5.1) indicates that the sensed molecular concentration is inversely proportional to the cube of the distance. This is a motivation to use the probabilistic model checker, PRISM, to verify the proposed model. PRISM is used twice, (1) to verify the proposed consensus protocol. For simplicity, the model is represented as a gird of nodes where $node_c$ is in the centre, and the other nodes are distributed within its transmission range at different distances. Then, (2) as the proposed consensus protocol is inspired by the consensus problem in [59], where an iterative method is considered for communication among nodes which enables information spreading and averaging in the network. The authors in [59] stated that the general case of networks is more difficult to analyse. Thus, PRISM model checker is used to verify the acquisition of consensus in this model. In [59] it is assumed that each node observes the same distances to the other nodes in the network. However, in the verification process, it is assumed that nodes are deployed at different distances.

6.3 Proposed Consensus Protocol

Based on the model described above, where it consists of a nanonetwork of $(n \times n)$ nanomachines, and one of these nanomachines is distinct as central node $node_c$ and other nanomachines $n(i,j)$ are in different positions around $node_c$ within its transmission range distance d_{max}, it is assumed that the central node $node_c$ has an initial value u of molecular concentration, and this initial value can represent a certain parameter in the environment.

The first step is that the central node $node_c$ diffuses its initial value u through the environment where the other nanomachines $n(i,j)$ are spread within the central node's $node_c$ transmission range distance d_{max}. The potential aim of the other nanomachines $n(i,j)$ is to estimate the concentration of the diffused initial value u. Thus, the other nanomachines $n(i,j)$ attempt to estimate u and then diffuse their estimations $u_{estimate}$ to the central node $node_c$. Meanwhile, the central node, $node_c$, attempts to compute the average of all estimations u_{av} diffused by $n(i,j)$, and then diffuse the average of estimations to the other nodes $n(i,j)$. The verification process is based on checking if the deviation value of the average of all estimations u_{av} from the original value u is less or equal ε.

6.3.1 Consensus Protocol Steps

The consensus protocol consists of a number of different steps in different time epochs. The length of the time epoch is assumed to be quite enough, so that the molecules diffuse away from the communication environment:

1. During the first epoch, the central node $node_c$ diffuses its initial value which is represented as unit u of molecules. Algorithm 10 shows the steps that follow diffusion of u by $node_c$, and the sensed molecular concentration by the other nanomachines $n(i,j)$ through Equation (5.1). The other nanomachines $n(i,j)$ attempt to compute $u_{estimate}$ from the sensed molecular concentration.

2. Over the second epoch, each nanomachine $n(i,j)$ diffuses its estimated unit of molecules $u_{estimate}$ to the central node, $node_c$. Through this epoch, the central node $node_c$ accumulate the sensed molecular concentration following the steps in Algorithm 11.

3. The accumulative sensed molecular concentration of the diffused $u_{estimate}$ during the second epoch is considered as u_{total}. Through the next epoch, $node_c$ will attempt to compute the average of the total sensed estimations u_{total} which have been diffused by $n(i,j)$. For simplicity, it is assumed that $node_c$ is able to recognize the number of nanomachines N within its transmission range.

The central node $node_c$ computes the average of all estimations u_{av} by dividing the total received molecules units u_{total} by the number of nanomachines N.

$$u_{av} = \frac{u_{total}}{N} \tag{6.1}$$

When the central node $node_c$ obtains the average of estimations u_{av}; it diffuses u_{av} into the medium. The other nanomachines $n(i,j)$ will sense u_{av} according to their distance from $node_c$ following the steps in Algorithm 10.

Reaching Consensus: The model assumption suggests that consensus is reached when the average estimation value u_{av} approaches to the value of u with ε deviation, i.e.

$$\text{deviation of average estimations from original value is} \quad |(u - u_{av})| \leqslant \varepsilon \tag{6.2}$$

where, $\varepsilon <= 10$. The second assumed condition to reach consensus is that the variance between the sensed molecular concentration by $node_c$ (when each nanomachine from $n(i,j)$ diffuses its $u_{estimate}$) and the average of estimations u_{av} is less or equal $\varepsilon 1$, where $\varepsilon 1 <= 1$. This, however is not applicable on Algorithm 11 as $node_c$ senses accumulative concentration and does not recognize that this $u_{estimate}$ is from this $n(i,j)$. However, in the verification process using PRISM, each nanomachine has a specific module, and that enables to identify each $u_{estimate}$ sensed by $node_c$, as it is explained in Section 6.4.

Time to reach consensus: The time to get the average of estimations $t_{u_{av}}$ can be found by:

$$t_{u_{av}} = \text{length of epoch} \times \text{number of epochs} \tag{6.3}$$

Which can be equal to:

$$t_{u_{av}} = T_0 \times 3$$

6.4 Model Verification

6.4.1 PRISM Model Checker

PRISM is a flexible tool for working with probabilistic real-life models, as it allows for the specification of probabilities inside the model and in the properties. Additionally, the software will calculate the probability of failure of a given property after verification [104]. Many systems from different application domains, such as communication protocols, randomised distributed algorithms and biological systems have utilize PRISM for analysis purposes [148], giving a motivation for us to employ PRISM in verifying our proposed protocol, besides that, the diffusion based molecular communication and molecules propagation through the environment are based on a stochastic process.

The proposed protocol is verified using PRISM, through representing the steps of the protocol as an MDP (Markov Decision Process) model. MDP is considered as an extension of DTMCs, which allows nondeterministic choice. Thus, it is similar to DTMC, in that it consists of discrete set of states representing possible configurations of the system being modelled, and in that the transitions between states occur in discrete time-steps [149]. This model has been applied to many randomized algorithms, including the consensus algorithm [42]. Consensus algorithms are often hard to model check, because the state space grows exponentially with the number of participating processes. In [104], PRISM is applied only to a shared-coin subroutine, while full correctness relies on verification using Cadence SMV, as well as higher level manual proofs.

To analyse and verify the proposed protocol using PRISM, the model's specifications described in Section 6.2 are constructed in the PRISM state-based language. In order to simplify the PRISM representation of the proposed protocol, the nanonetwork is depicted as a grid of nanomachines, where $node_c$ is in the centre and the other nanomachines $n(i,j)$ are distributed at different distances, within the transmission range of $node_c$. Each nanomachine is defined in a separate module, and within each module the processes that could emerge at specific time epochs are assigned. Allocating the processes of each nanomachine during each epoch of the proposed protocol. 'Labels' are used to facilitate the synchronization between processes that could be initiated during a specific epoch. This is done alongside a distinctive module for $node_c$ processes. Hence, different experiments are carried out to verify the proposed protocol, a deterministic experiment (without taking in consideration environment noise) and probabilistic experiments. The process of system verification was undertaken using PRISM 4.2.1 running on MS Windows 7, Intel(R) Core(TM)i3-2370M CPU 2.40GHz, 4.00 GB of RAM.

6.4.2 3x3 Grid

6.4.2.1 Representation of 3x3 Grid in PRISM Language

The model of (3x3) grid network is represented in PRISM language, through defining a module for each node and representing the process of the proposed protocol.

```
mdp
const int N=8;               //number of nanomachines in the network
const double dist1=0.02;   //distance1
const double dist2=0.3;    //distance2

//central node module
module node_c

active_c:[0..1] init 1; //flag to show that node_c is active
```

```
                         // and can diffuse molecules
diffuse_c:[0..1] init 0; //flag to indicate that node_c
                         // hasn't diffuse u yet

[] active_c=1& diffuse_c=0->(diffuse_c'=1)&(active_c'=1); //indicates node_c
                                                           // diffused u
endmodule
```

Thus, the PRISM model starts by diffusing the initial unit of molecules u by the central node. Using labels the processes at each time epoch is represented. Two values are identified for the distance between the central node and the other nanomachines depending on their position from the central node in the same gird: 0.02 which is close to the central node and 0.3.

```
//module of nanomachine 1
module node1
active1:[0..1] init 1;   //flag indicates node1 is active and can diffuse
diffuse1:[0..1] init 0;  //flag indicates if node1 diffused its
                          // estimation or not
u1:[0..u] init 0;    //sensed molecular concentration by node1 from node_c
x1:[1..2] init 1;   //flag to differentiate between process
                     //in epoch 1 and 2
u_c1:[0..u]init 0; //diffused estimation by node1 to node_c

[tick1]active1=1&diffuse_c=1&u1<=u&x1=1->(u1'= sensed_1)&(x1'=2);
[tick2]u1>0&active1=1&diffuse1=0&u_c1<=u&x1=2->(u_c1'=estimate_1)
                                                  &(diffuse1'=1);
endmodule
```

The modules for the rest of the nanomachines have the same structure, except for the the values of $sense_1$ and $estimate_1$ which depend on their distance from $node_c$, their sensed molecular concentration, and their estimation of u. Labels [$tick1$], [$tick2$] represent the processes whereby a nanomachine can perform at each epoch. Label assists in synchronizing all modules; hence, [$tick1$] is implemented simultaneously in all modules of the nanomachines (i.e., all nanomachines from $n(i, j)$ compute the sensed molecules form the central node). The same is the case for [$tick2$], (i.e., all nanomachines from $n(i, j)$ diffuse their estimations to the central node). Then, the central node diffuses the average of all the received concentrations, which represent the final stage of the protocol.

The calculations of Algorithm 11 and 10 are quite complicated to represent in PRISM as with these parameters' high values the calculations for building the PRISM model take much time. Therefore, all the related computations of Algorithm 11 and 10 were implemented in MATLAB, and the output results employed inside PRISM command *updates* in values of $sense_i$ and $estimate_i$, (where, $i \in \{1, 2, \cdots, N\}$). Thus, the sensed molecular concentration (i.e., $u_{estimate}$) of each nanomachine when $node_c$ diffuses u, and the sensed molecular concentration of $node_c$ when the other nanomachines diffuse their estimations $u_{estimate}$ are computed outside PRISM.

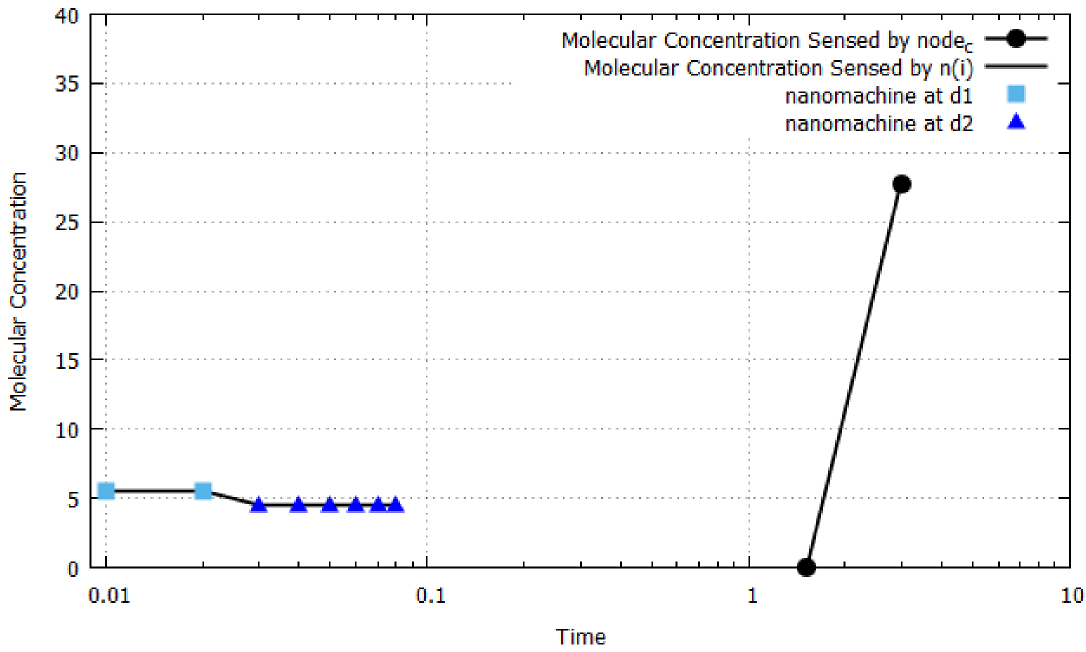


FIGURE 6.2: Sensed concentration by $n(i, j)$ and $node_c$ in $3x3$ grid

Figure 6.2 represents an experiment to implement the protocol first two steps (i.e., diffusing u by $node_c$ and sensing the diffused molecular concentration by the other nanomachines $n(i, j)$, then the diffusing of $u_{estimate}$ by the other nanomachines $n(i, j)$ and the accumulative molecular concentration sensed by $node_c$). The implementation is on a nanonetwork consisting of eight nanomachines at different distances around the central node, where $d1$=0.02nm, and $d2$=0.3nm. The transmission range of the central node, where d_{max}= 1nm, and time slot length T_0 is equal to 1.5ms. The initial value of time t is 0.01, the number of diffused units u by $node_c$ is $2 \times 10^{15} molecules$, the value of the threshold τ is $1 \times 10^{16} molecules.cm^{-3}$, and the diffusion coefficient D is 1×10^{-6} cm^2/s.

In Figure 6.2 the x-axis represents time, where the length of one epoch (i.e., T_0) is 1.5. The y-axis represents the amount of sensed molecular concentration. The first part (epoch 1) of the figure shows the sensed molecular concentration of each nanomachine from $n(i, j)$ after $node_c$ diffuses u. The second part (epoch 2) shows the accumulative sensed molecular concentration by $node_c$, after each nanomachine from $n(i, j)$ diffuses its $u_{estimate}$.

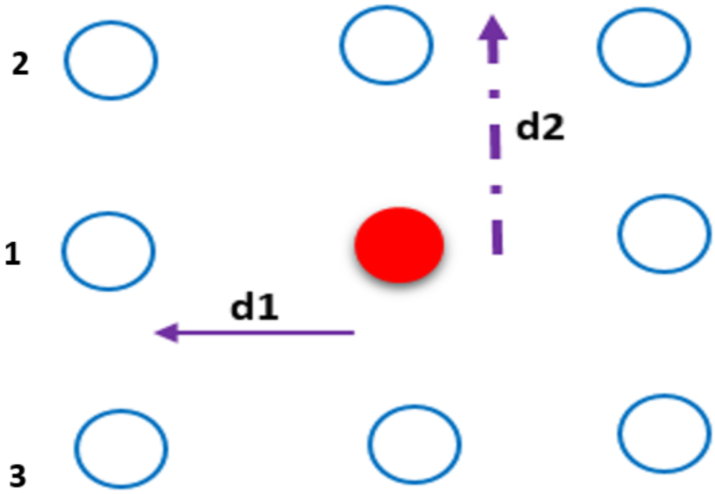


FIGURE 6.3: Assumed 3x3 grid representation

Figure 6.3 depicts the assumed 3x3 grid (like) model, where the filled colour circle represents the central node $node_c$, an the other outlined circles signify the other nanomachines $n(i, j)$. In Figure 6.3 the distance between the central node $node_c$ and all the other nodes on row 1 is $d1$. While the distance between $node_c$ and all the other nodes on row 2 and 3 is $d2$. However, the model topology is assumed to be *grid-like* topology.

In PRISM model each nanomachine of $n(i, j)$ has its specific *module*. Thus, the sensed molecular concentration by $node_c$ from each nanomachine needs to be computed, rather than the accumulative value of the sensed concentration as Figure 6.2 shows. Thus, the experiment in Figure 6.2 is repeated with the same parameter values for each individual nanomachine separately, to compute what $node_c$ can sense when a certain node at a certain distance diffuses its $u_{estimate}$. However, in Figure 6.2 the sensed molecular concentration by each of $n(i, j)$ when $node_c$ diffuses u is already computed.

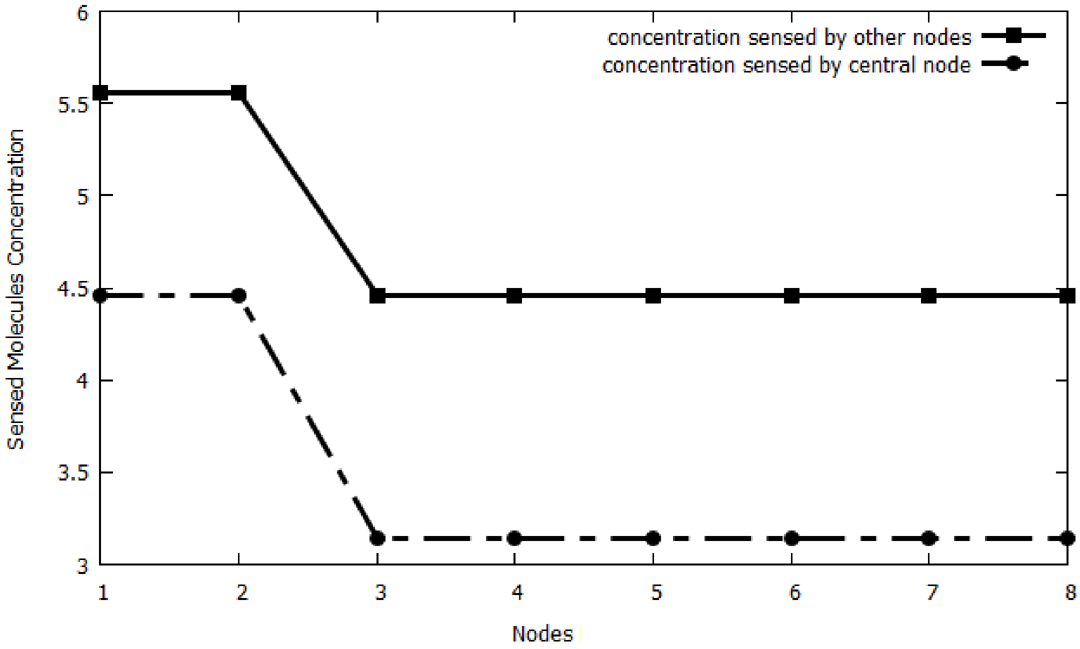


FIGURE 6.4: Sensed concentration by $node_c$ from each $n(i,j)$ and sensed concentration by $n(i,j)$ from $node_c$ in 3×3

Figure 6.4 represents an experiment to compute the sensed molecular concentration by each nanomachine $n(i,j)$ from the diffused u by $node_c$, and the sensed molecular concentration by $node_c$ from each nanomachine from $n(i,j)$. The x-axis represents the number of nanomachines (nodes) in the experiment, where the first two points are nanomachines at distance 0.02 from the central node $node_c$ and the rest points are nanomachines at a distance of 0.3 from $node_c$. The y-axis represents the amount of sensed molecules.

When the central node, $node_c$, diffuses unit u of molecules, the first line in Figure 6.4 represents the amount of sensed molecules $u_{estimate}$ by the other nanomachines $n(i,j)$, where each point in the line represents the sensed molecules of nanomachine $n(i,j)$ at distance d_i. The dashed line in the figure represents the amount of sensed molecules by the central node, $node_c$, when the other nodes diffuse their estimation $u_{estimate}$. It is observed that the two lines in Figure 6.4, seem to be quite symmetric, due to the identical values of some parameters used; however, the difference is mainly in the amount of diffused molecules, where the central node $node_c$ diffuses u of molecules, and the other nanomachines $n(i,j)$ diffuse their estimations $u_{estimate}$ which is less than u and could vary, depending on the distance of the nanomachine from $node_c$.

The result values of the MATLAB experiments were used in PRISM to represent the values of the sensed molecular concentration by each $n(i,j)$ during the first epoch, and the sensed molecular concentration by $node_c$ from each $n(i,j)$ during the second epoch. The following code segment represents the module of nanomachine number 8.

```
//node8
module node8
```

```
active8: [0..1] init 1;
diffuse8:[0..1] init 0;
u8:[0..u];
x8:[1..2] init 1;
u_c8:[0..u];

[tick1]active8=1&diffuse_c=1&u8<=u&x8=1->(u8'=sensed_8)&(x8'=2);
[tick2]active8=1&diffuse8=0&u_c8<=u&x8=2->(u_c8'=estimate_8)&(diffuse8'=1);
endmodule
```

This part of PRISM code ($u8' = sensed_8$) (it represents a command's *update*), means that the new value of $u8$ equals $sensed_8$ value. Where $u8$ represents $u_{estimate}$ of nanomachine 8 at distance $d2$ from $node_c$, and the value of $sensed_8$ is the computed $u_{estimate}$ of nanomachine 8 in the MATLAB experiment, while this part ($u_c8' = estimate_8$) means the new value of the u_c8 equals $estimate_8$. Where, u_c8 represents the sensed molecular concentration by $node_c$ from nanomachine 8 in particular when it diffuses its $u_{estimate}$, and the value of $estimate_8$ is the computed value of the sensed molecular concentration by $node_c$ from nanomachine 8 when it diffuses its $u_{estimate}$ in the MATLAB experiment. This process is repeated in the same way in the modules of nanomachines $\{1, 2, \cdots, 7\}$.

Three experiments were performed to represent the $3x3$ model in PRISM language:

- **Deterministic experiment**: where the noise effects on the received concentration are not considered, such that, the PRISM command's *updates* include one value of $sensed_i$ and one value of $estimate_i$, (where, depending the nanomachine module $i \in \{1, 2, \cdots, 8\}$). As the example of $node8$ module code segment.

- **Probabilistic experiment-1**: the noise effects on the received concentration is considered, such that, the PRISM command's *updates* include the summation of two probability values. Firstly; the value of sensed molecules (calculated using MATLAB), secondly; zero value, that when the noise is too high and nodes cannot sensed the diffused molecules. As an example:

```
[tick1]active1=1&diffuse_c=1&u1<=u&(x1=1)->0.6:(u1'=sensed_1)&(x1'=2)+
                                           0.4:(u1'=0)&(x1'=2);
```

- **Probabilistic experiment-2**: where the effects of threshold value on the sensed molecules are considered, besides the noise effects, so that the PRISM command's *updates* include three values. Beside the two values explained above, the third value would be the sensed molecular concentrations, in case that the assumed threshold is less than the actual value of τ. Thus, this can affect the values of $sensed_i$ and $estimate_i$ that are computed in the MATLAB experiment. As an example:

```
[tick2] active3=1&diffuse3=0&u_c3<=u& x3=2->
0.6:(u_c3'=estimat_3_1)&(diffuse3'=1)+
0.2:(u_c3'=estimat_3_2)&(diffuse3'=1)+
0.2:(u_c3'=0)& (diffuse3 '=1);
```

While performing all these experiments, a number of properties needed to be verifies. These properties and the results of verification are discussed in the following Subsection 6.4.2.2.

6.4.2.2 Verification of 3x3 Grid

The properties in Chapter 3 were based on the **P** operator. Using **quantitative properties** we can computes the actual probability that some behaviour of a model is observed, rather than just verifying whether or not the probability is above or below a given bound [149] in a form of $P = ?[path - prop]$. In this chapter, **quantitative properties** are also used, but in the form of: $Pmin = ?[path - prop]$ and $Pmax = ?[path - prop]$ to verify the specifications of the MDP PRISM models.

In this chapter, properties that relate to rewards are also used. The specification and analysis of properties based on costs and rewards are possible using PRISM. Thus, PRISM can be used to reason, not only about the probability that a model behaves in a certain manner but also about a wider range of quantitative measures relating to model behaviour [149]. For example:

```
rewards
    true : 1;
endrewards
```

This example assigns a reward of 1 to every state of the model, the left part of which (true) is a guard and the right part of which (1) is a reward. The reward is assigned to the states of the model which satisfy the predicate in the guard [149].

In a PRISM model, in order to verify that consensus between nanomachines is reached, the following computations and labels are used:

```
//compute u_total and u_av
formula u_total= u_c1+ u_c2 + u_c3+ u_c4 + u_c5 + u_c6 + u_c7 + u_c8;
formula u_av = floor (u_total/N);

//compute variance and deviation
formula variance= ((pow ((u_c1-u_av), 2) + pow ((u_c2-u_av), 2)
                  + pow ((u_c3-u_av), 2)+
pow ((u_c4-u_av), 2)+ pow ((u_c5-u_av), 2)+ pow ((u_c6-u_av), 2) +
pow ((u_c7-u_av), 2)+pow ((u_c8-u_av), 2))/(N-1));

formula deviation =|u-u_av|;

//labels to be used in the properties formation
label "agree"= variance>0 & variance<=1;
label "agree2" = deviation>0 & deviation<=10;


label "finish"= diffuse1=1 & diffuse2=1 & diffuse3 =1 & diffuse4 =1 &
                diffuse5 =1 & diffuse6 =1 & diffuse7 =1 & diffuse8 =1 ;

rewards "steps_no"
          true:1;
endrewards
```

where: $u_c1, u_c2, \cdots, u_c8$ in the variance formula, represents the sensed molecular concentration by $node_c$ where each nanomachine from $n(i, j)$ diffuses its $u_{estimate}$.

While performing the three different experiments listed in Subsection 6.4.2.1, a number of properties were considered to be verified, such as:

$$Pmin =?[F"finish"] \tag{6.4}$$

which computes the probability that all actions of the protocol have been calculated. Other properties are:

$$R"steps - no"min =?[F"finish"] \tag{6.5}$$

$$R"steps - no"max =?[F"finish"] \tag{6.6}$$

which represent the minimum and maximum expected number of steps required for the completion of the model, through defining a reward part in PRISM model.

However, the focus is on the main properties:

$$Pmin =?[F"agree"] \tag{6.7}$$

$$Pmin =?[F"agree2"] \tag{6.8}$$

which represent the probability of eventually reaching agreement between the nanomachines according to the described protocol, based on the defined formulas of *variance* and *deviation* in PRISM model. The following Figure 6.5 shows the result of verifying property (6.7)

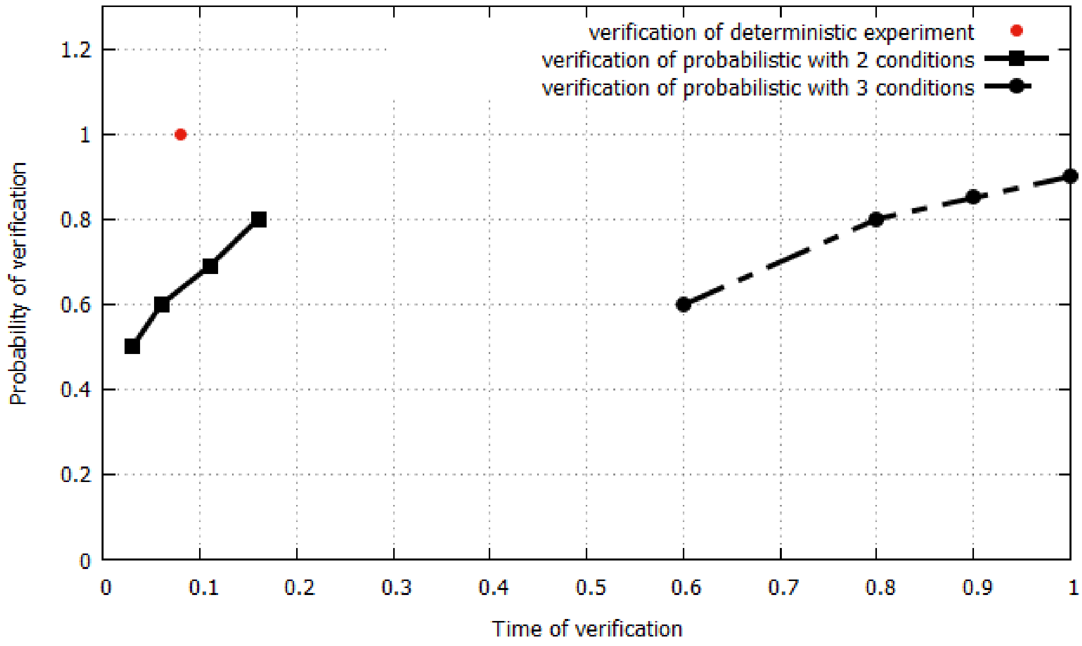


FIGURE 6.5: Verification of property 6.7 in 3 × 3 Grid

The x-axis in Figure 6.5 represents the time needed (in seconds) to verify the consensus property, and y-axis denotes the probability of reaching consensus, though the scale of y-axis is edited only to clarify the probability of the deterministic experiment. Figure 6.5 represents the results of different experiments on 3×3 grid to verify property (6.7), where, the first experiment is a deterministic one, and the probability of satisfying property (6.7) is 1. The other two experiments are probabilistic. In each, different probabilities are assumed for *update*. For example, in the two updates probabilistic experiment:

```
[tick1]active4=1&diffuse_c=1&u4<=u&x4=1->0.5:(u4'=sensed_4)&(x8'=2)+
                                            0.5:(u8'=0)&(x8'=2);
```

Starting with (0.5+0.5) where the probability that the sensed molecular concentration=$sensed_4$ is 0.5 and the probability that the molecular concentration= zero is 0.5, then, the probabilities are changed into (0.6+0.4), (07+0.3), and (0.8+0.2). In these experiments, it is assumed that sensing molecular concentration $sensed_i$ has the higher probability. Each point on the line in Figure 6.5, represents an experiment with different probabilities, starting with (0.5+0.5) and ending with the last point where the probability is (0.8+0.2). It is noticeable that verifying property (6.7) at the last point on the line has a higher probability than the assumed probability of sensing zero molecular concentration which equals 0.2. Similarly, the dotted line in Figure 6.5, represents the experiment with three updates, where the probabilities of each point start by (0.5+0.25+0.25), as in this example:

```
[tick2]  active3=1&diffuse3=0&u_c3<=u&x3=2->
0.5:(u_c3'=estimate_3_1)&(diffuse3'=1)+
0.25:(u_c3'=estimate_3_2)&(diffuse3'=1)+
0.25:(u_c3'=0)&(diffuse3'=1);
```

The next points probabilities are (0.6+0.2+0.2), (0.7+0.15+0.15)and (0.8+0.2+0.2) respectively. In the same way, the probability of reaching consensus is higher at the last point, as it is assumed that the probability of sensing molecular concentration (without the elevated effects of noise and the effects of the value of threshold) equal to 0.8.

In a similar way the deviation property (6.8) is verified in three experiments and with different probabilities, as explained in verifying property 6.7 above. The following Figure 6.6 shows the verification of property 6.8 :

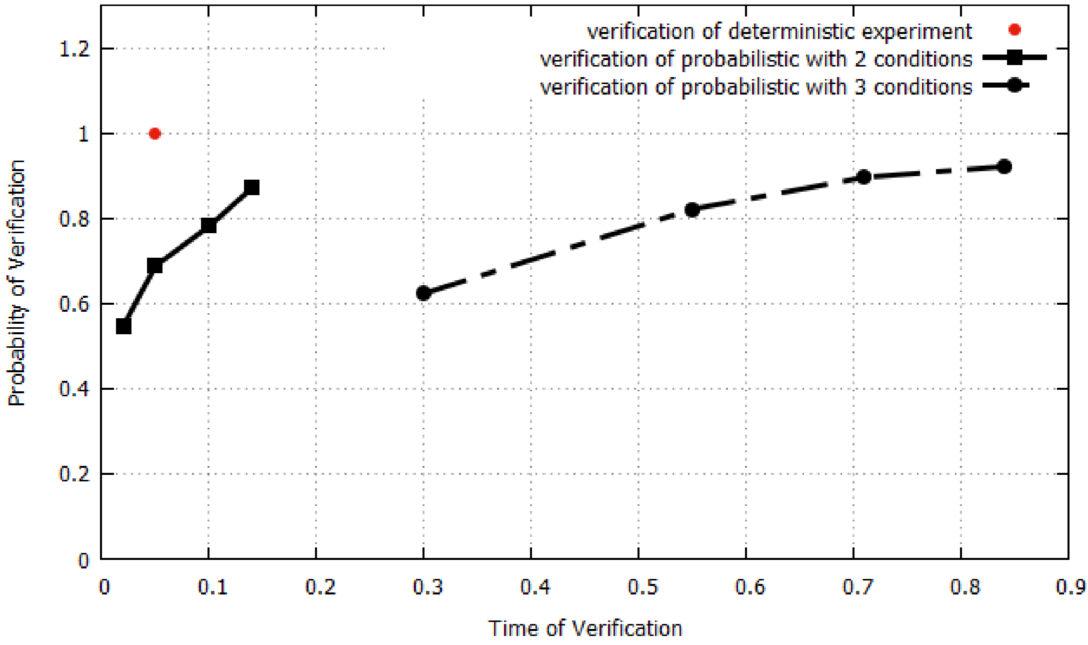


Figure 6.6: Verification of property 6.8 in 3 × 3 Grid

The probability of verifying property 6.8 in general is slightly higher than the probability of property 6.7.

Building PRSIM model Results: The results of building a model of 3×3 nanomachines using PRISM are:

- Deterministic Experiment Reachability (BFS): 4 iterations in 0.01 seconds (average 0.002500, setup 0.00)

Time of Model Building(in seconds)	0.06
No. of states	4
No. of transitions	4

Table 6.1: Model Construction (3 × 3) grid- Deterministic experiment

Reachability in MDP PRISM model, represents the minimum and maximum probability of *reaching target set*. *Target set* is all states labelled with proposition that the protocol reach to its main goal. Where the goal of the protocol is defined in the properties that have been verified.

- Probabilistic Experiment with two *updates* Reachability (BFS): 4 iterations in 0.00 seconds

Time of Model Building(in seconds)	0.02
No. of states	33026
No. of transitions	65921

TABLE 6.2: Model Construction (3 × 3) grid- probablistic experiment

- Probabilistic Experiment with three *updates* Reachability (BFS): 4 iterations in 0.00 seconds

Time of Model Building(in seconds)	0.13
No. of states	28704377
No. of transitions	57404377

TABLE 6.3: Model Construction (3 × 3) grid- probabilistic experiment

6.4.3 Verification of 5x5 Grid

The same experiments applied to 3 × 3 gird, are repeated with grid of 5 × 5 nanomachines.

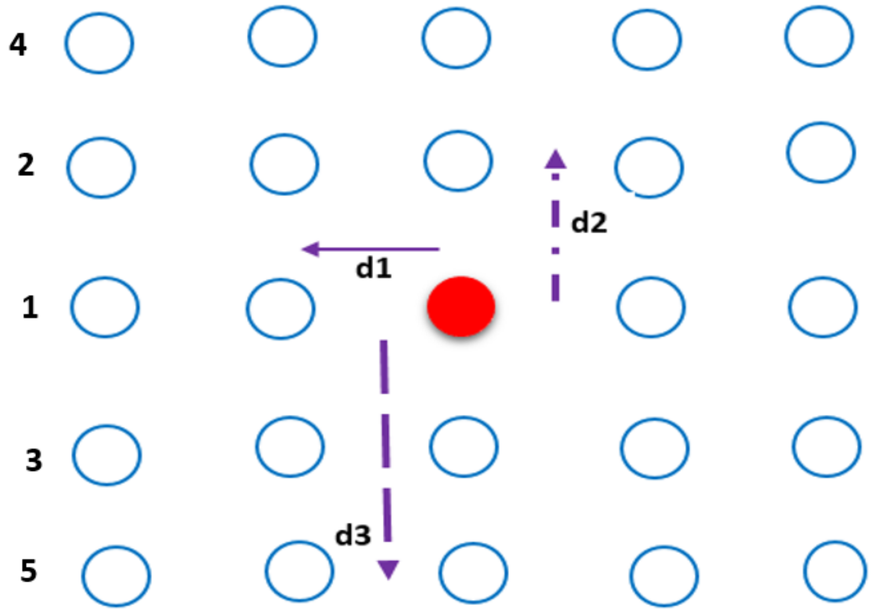


FIGURE 6.7: Assumed 5x5 grid representation

Figure 6.7 shows the assumed 5 × 5 grid model, where the filled colour circle represents $node_c$, and the other outlined circles signify the other nanomachines $n(i, j)$, where nanomachines were distributed in three different distances from $node_c$. In Figure 6.7 the distance between $node_c$ and all the other nodes on row 1 is $d1$. While the distance between $node_c$ and all the other nodes on row 2 and 3 is $d2$. However, $d3$ is the distance between $node_c$ and all the other nodes on row 4 and 5.

Thus, in order to represent the 5 × 5 model in PRISM, the sensed molecular concentration by $n(i, j)$ and $node_c$ were computed in MATLAB experiments.

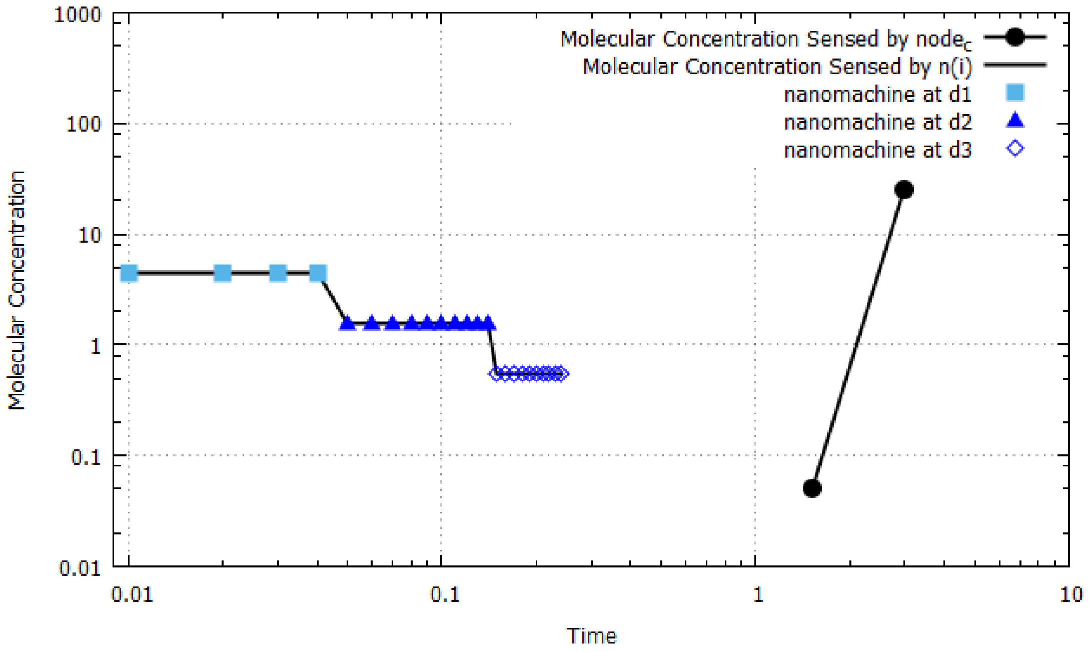


FIGURE 6.8: Sensed concentration by $n(i, j)$ and $node_c$ in 5 × 5 grid

Figure 6.8 represents an experiment to implement the first two steps of the protocol: The diffusing of u by $node_c$ and sensing the diffused molecular concentration by the other nanomachines $n(i, j)$. Then the diffusing of $u_{estimate}$ by the other nanomachines $n(i, j)$, the accumulative molecular concentration sensed by $node_c$). With the same parameter values presented in Figure 6.2.

Then, in order to compute the sensed molecular concentration by $node_c$ coming from each individual nanomachine in $n(i, j)$ and not as accumulated molecular concentration, the experiment of Figure 6.8 is repeated for each nanomachine separately as Figure 6.9 shows.

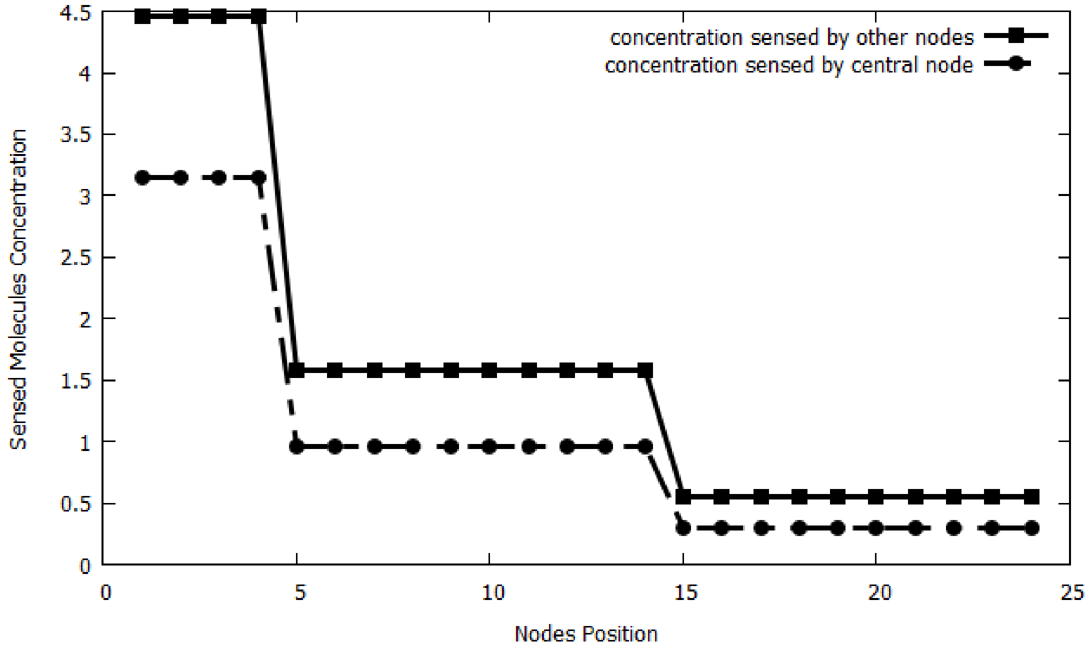


FIGURE 6.9: Sensed concentration by $node_c$ from each $n(i,j)$ and sensed concentration by $n(i,j)$ from $node_c$ in 5×5

Figure 6.9 displays the sensed molecular concentration by $n(i,j)$, when $node_c$ diffuses u, which take place during the first epoch, and the sensed molecular concentration by $node_c$ from each nanomachine $n(i,j)$ when it diffuses its $u_{estimate}$.

The result values of the MATLAB experiments used in PRISM to represent the values of the sensed molecular concentration by each $n(i,j)$ during the first epoch, and the sensed molecular concentration by $node_c$ from each $n(i,j)$ during the second epoch. As explained in subsection 6.4.2.1, however, the difference here is that PRSIM model is consisting of 24 modules for each nanomachine from $n(i,j)$ and a module for $node_c$. PRISM allows one to define modules based on predefined modules, instead of rewriting the code of the module, as an example:

```
//node15
module node15
active15:[0..1] init 1;
diffuse15:[0..1] init 0;
u15:[0..u];
x15:[1..2] init 1;
u_c15:[0..u];

[tick1]active15=1&diffuse_c=1& u15<=u &x15=1->0.6:(u15'=sensed_15)&(x15'=2)
                                              +0.4:(u15'=0)&(x15'=2);
[tick2] active15=1& diffuse15=0& u_c15<=u&x15=2->
0.6:(u_c15'=estimate_15)&(diffuse15'=1)+
0.4:(u_c15'=0)&(diffuse15'=1);
endmodule
```

```
//The processes of module node15 which represents nanomachine 15 at distance
//d3 from node_c , are the same processes of modules of nodes16-node20

module node16=node15[active15=active16 , send15=send16 , diffuse15=diffuse16 ,
                        u15=u16 , x15=x16 , u_c15=u_c16] endmodule
module node17=node15[active15=active17 , send15=send17 , diffuse15=diffuse17 ,
                  u15=u17 , x15=x17 , u_c15=u_c17] endmodule
module node18=node15[active15=active18 , send15=send18 , diffuse15=diffuse18 ,
                  u15=u18 , x15=x18 , u_c15=u_c18] endmodule
module node19=node15[active15=active19 , send15=send19 , diffuse15=diffuse19 ,
                  u15=u19 , x15=x19 , u_c15=u_c19] endmodule
module node20=node15[active15=active20 , send15=send20 , diffuse15=diffuse20 ,
                  u15=u20 , x15=x20 , u_c15=u_c20] endmodule
```

Verification Process: In 5 × 5 the verification of property (6.7) and property (6.8) is carried out through three different experiments (deterministic, probabilistic with two *updates*, probabilistic with three *updates*). In each experiment the probabilities are changed as explained in Subsection 6.4.2.2.

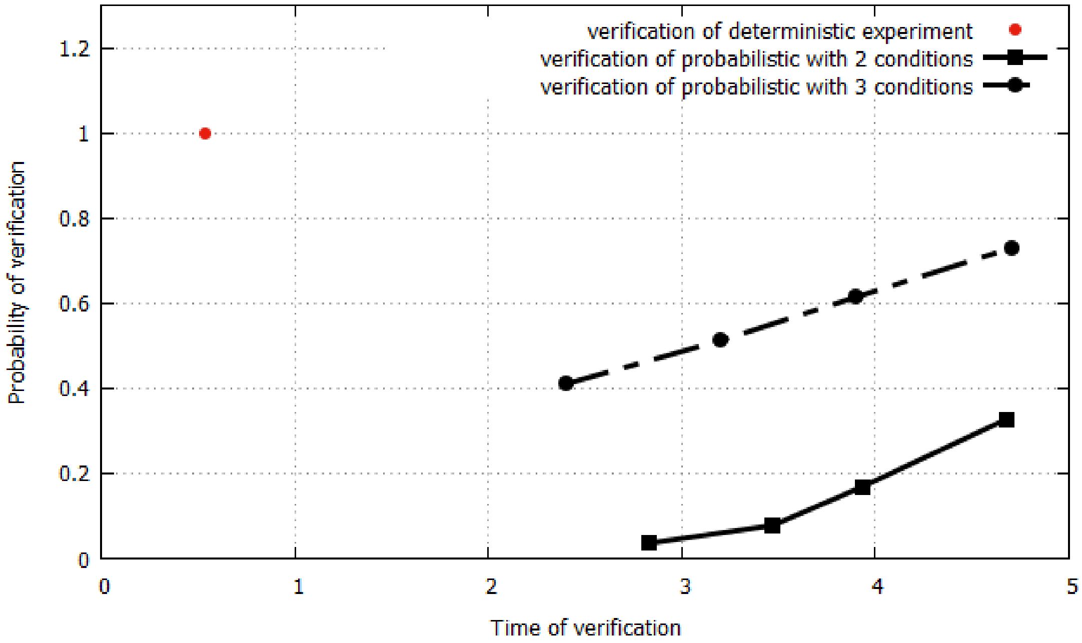


FIGURE 6.10: Verification of property (6.7) in 5 × 5 Grid

Figure 6.10 shows the results of verifying property (6.7) in 5 × 5 grid. Time of verification (in general, time of building PRISM model) increased in 5 × 5 compared to the 3 × 3 grid model (Figure 6.5), due to its increased number of defined modules. However, the probability values of verifying property (6.7) in 5 × 5 are quite significantly less than the verification probability in 3 × 3 grid.

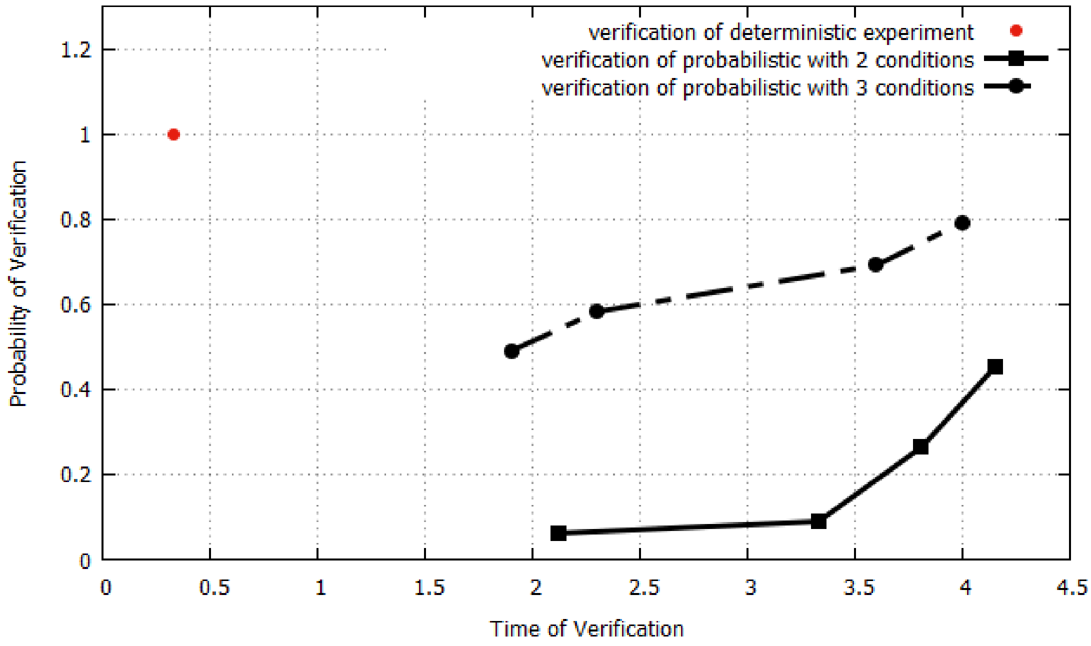


FIGURE 6.11: Verification of property (6.8) in 5 × 5 Grid

Figure 6.11 shows the results of verifying property (6.8) in 5 × 5. The verification probability in Figure 6.11 is slightly higher than the probability in Figure 6.10.

Building PRSIM model Results: The results of building a model of 5×5 nanomachines, using PRISM are as follows:

- Deterministic Experiment

  Reachability (BFS): 4 iterations in 0.00 seconds

Time of Model construction	0.25
No. of states	4
No. of transitions	4

TABLE 6.4: Model Construction 5 × 5 grid - deterministic experiment

Recall that **reachability** in MDP PRISM model, represents the minimum and maximum probability of *reaching target set.*

- Probabilistic Experiment with two *updates*

  Reachability (BFS): 4 iterations in 0.02 seconds (average 0.005000, setup 0.00)

Time of Model construction	0.32
No. of states	3707929545504
No. of transitions	7415858829849

TABLE 6.5: Model Construction 5 × 5 grid - probabilistic experiment

- Probabilistic Experiment with three *updates*

  Reachability (BFS): 4 iterations in 0.00 seconds

Time of Model construction	0.13
No. of states	6504921322212
No. of transitions	9202301157631

TABLE 6.6: Model Construction 5×5 grid - probabilistic experiment

6.4.4 Verification of 7x7 Grid

The processes which are applied to 3×3 and 5×5 grid were repeated in the 7×7 grid of nanomachines. Thus, the difference is that the number of nanomachines is 48 which are distributed at four different distances from nod_c, as shown in Figure 6.12.

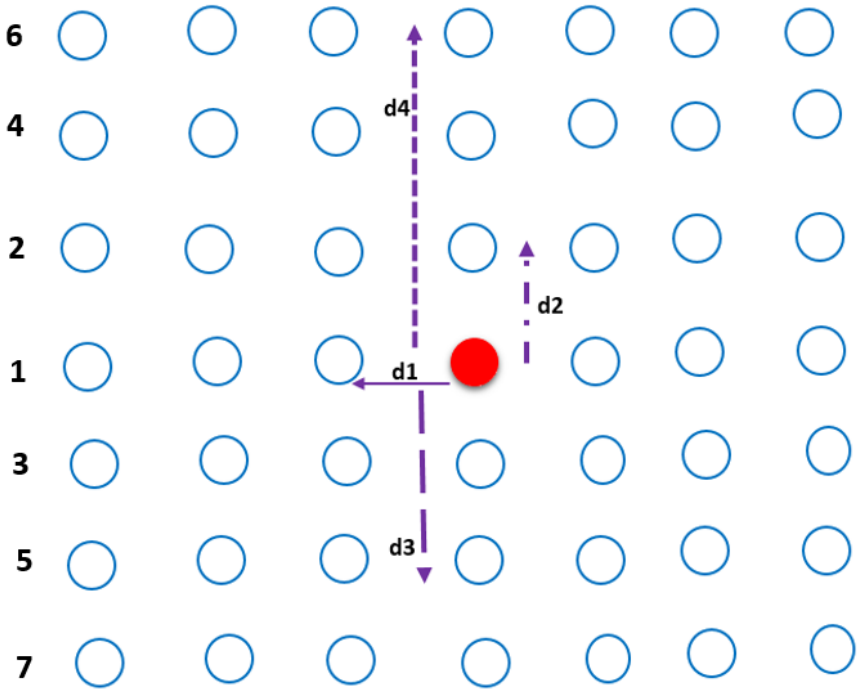


FIGURE 6.12: Assumed 7x7 grid representation

In Figure 6.12 the distance between $node_c$ and all the other nodes on row 1 is $d1$. $d2$ is the distance between $node_c$ all the other nodes on row 2 and 3. While the distance between $node_c$ and all the other nodes on row 4 and 5 is $d3$. $d4$ is the distance between $node_c$ all the other nodes on row 6 and 7.

As the processes of MATLAB and PRISM experiments have already been explained in previous subsections, only the output results for the 7×7 grid will be shown in this subsection.

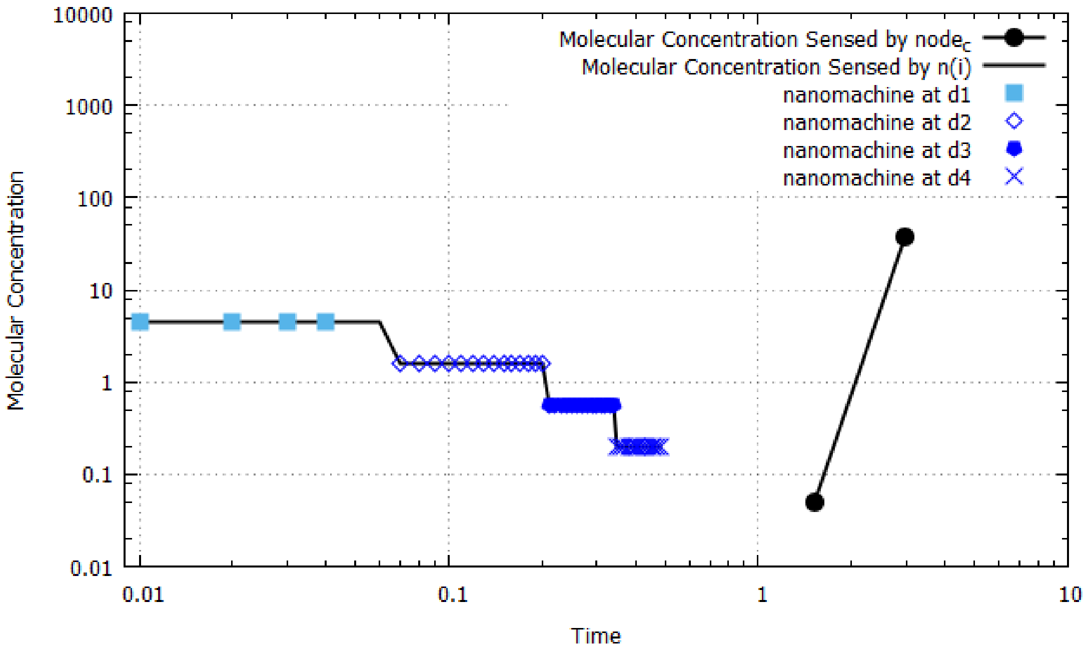


FIGURE 6.13: Sensed concentration by $n(i,j)$ and $node_c$ in 7×7 grid

Figure 6.13 represents a MATLAB experiment to implement the protocol's first two steps of diffusing u and sensing it by $n(i,j)$, then diffusing $u_{estimate}$ and sensing the accumulated concentration by $node_c$.

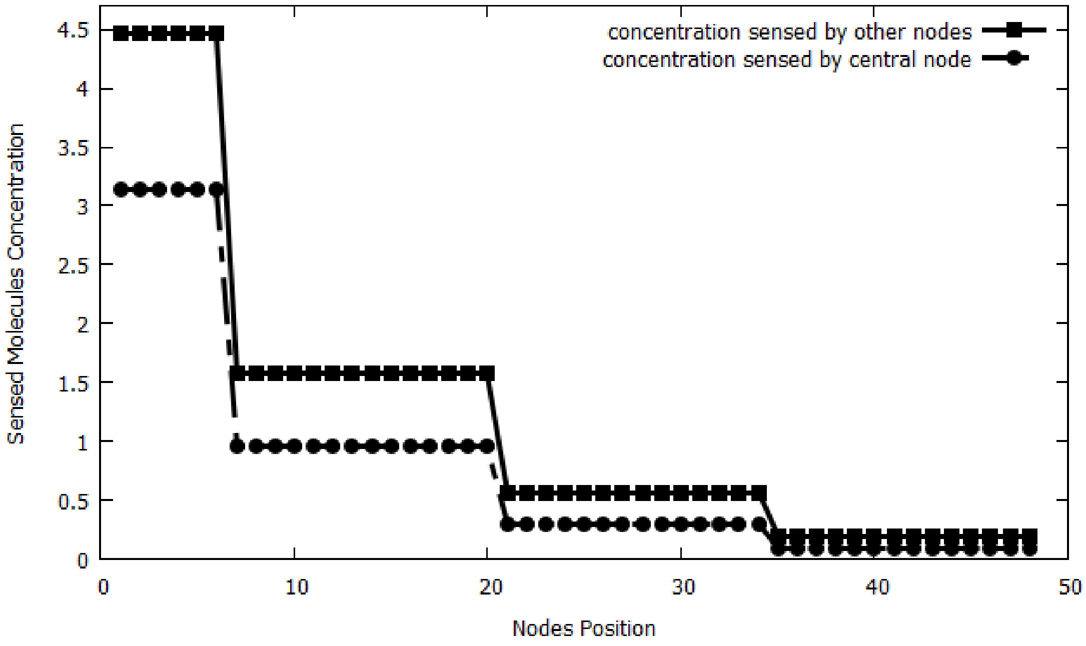


FIGURE 6.14: Sensed concentration by $node_c$ from each $n(i,j)$ and sensed concentration by $n(i,j)$ from $node_c$ in 7×7 grid

The main aim of Figure 6.14 is to show the sensed molecular concentration by $node_c$ from each nanomachine $n(i,j)$ when it diffuses its $u_{estimate}$.

Verification Process: In 7 × 7 the verification of properties (6.7) and (6.8) was implemented through three different experiments, and in each experiment the probabilities were changed as explained in previous subsections.

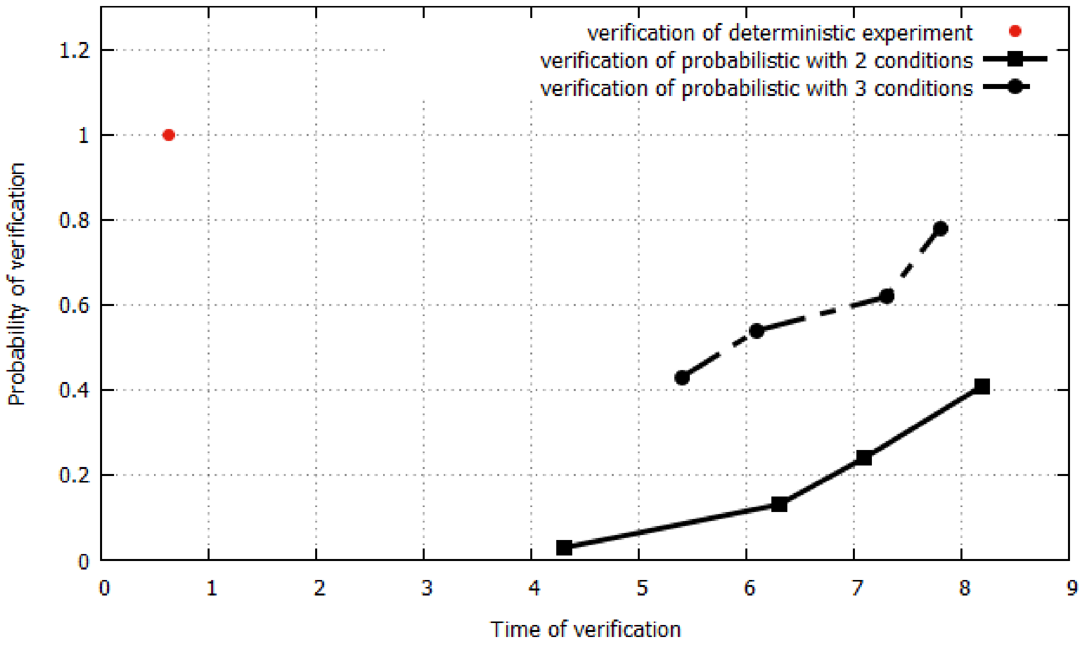


Figure 6.15: Verification of property (6.7) in 7 × 7 Grid

Figure 6.15 shows the results of verifying 6.7 in the 7 × 7 grid. The verification time increased in 7 × 7 compared to 5 × 5 grid model (Figure 6.10). The time taken to build PRISM model in 7 × 7 grid also increased. However, the probability values of verifying property (6.7) in 7 × 7 were slightly less than the verification probability in 5 × 5.

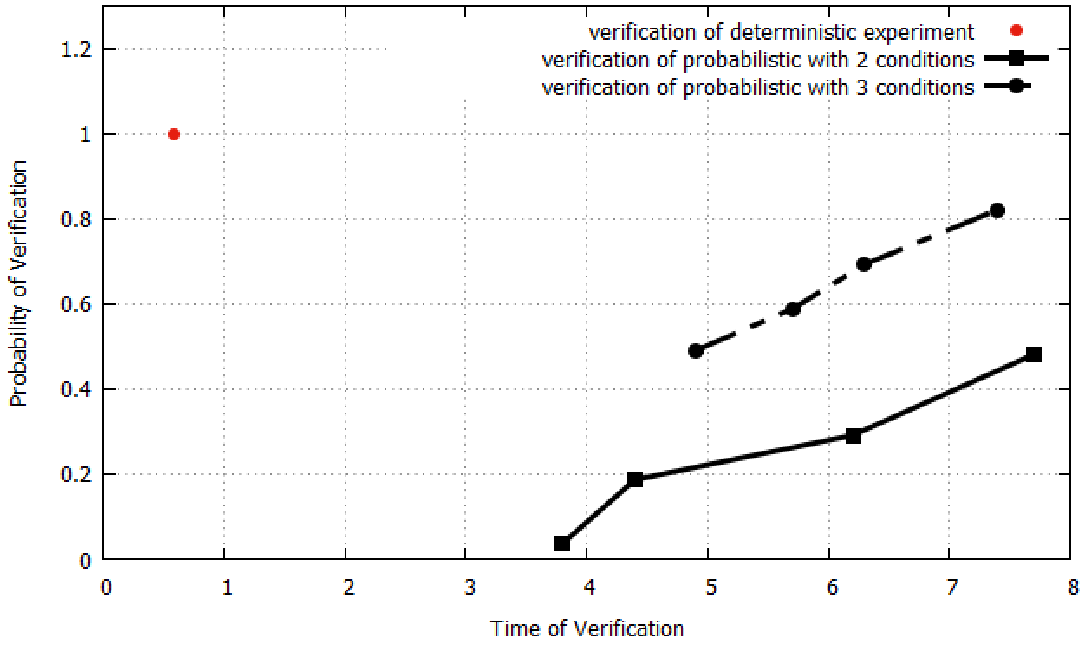


FIGURE 6.16: Verification of property (6.8) in 7 × 7 Grid

Figure 6.16 shows the results of verifying property (6.8) in 7 × 7. The verification probability in Figure 6.16 can still be seen to be slightly higher than the probability in Figure 6.15.

Building PRSIM model Results: The results of building a model of 7×7 nanomachines, using PRISM are as follows:

- Deterministic Experiment

  (BFS): 4 iterations in 0.01 seconds (average 0.002500, setup 0.00)

Time of Model construction	0.7
No. of states	4
No. of transitions	Infinity

TABLE 6.7: Model Construction 7 × 7 grid

- Probabilistic Experiment with two *updates*

  Reachability (BFS): 4 iterations in 0.37 seconds (average 0.092250, setup 0.00)

Time of Model construction	3.251
No. of states	6.235007847244561E21
No. of transitions	Infinity

TABLE 6.8: Model Construction 7 × 7 grid - probabilistic experiment

- Probabilistic Experiment with three *updates*

  Reachability (BFS): 4 iterations in 0.29 seconds (average 0.072500, setup 0.00)

Time of Model construction	3.852
No. of states	1.8878432524958716E24
No. of transitions	Infinity

Table 6.9: Model Construction 7 × 7 grid - probabilistic experiment

6.5 Verification of Consensus Protocol from [59]

The consensus protocol discussed in this chapter and Chapter 5 are inspired by the work in [59], where the nanonetwork's nodes attempted to estimate a certain parameter in the environment. The authors in [59] studied a discrete model of time epochs of length T_0. The length of epoch depends on the diffusion coefficient of the medium D and the network topology-specifically depends the effective radius of each node, $T_0 = k\frac{R^2}{D}$, where k is constant and R is the effective radius between each node and the central node. In each time epoch, node i estimates the environment parameter and convert this estimation into molecules concentration p_i. The authors assumed that these estimated concentrations are drawn from Gaussian distribution $N(\mu, \sigma_0^2)$ with excepted value μ and variance σ_0^2. Every node in the network needs to obtain the average of initial estimates of nodes $p_{av} = \frac{1}{N}\sum_{i=1}^{N} p_i$. Consensus is reached when the estimation value of each node approaches the average of all estimations and the variance of the nodes estimations approaches $\frac{\sigma_0^2}{N}$.

The authors in [59] stated that the general case of networks is more difficult to be analysed. Thus, in this section PRISM model checker was used to verify the acquiring of consensus in this model. However, nanomachines are assumed to be deployed arbitrarily, and it is therfore not necessarily the case that each node observes the same distance to the other nodes in the network, as considered in [59].

The model is represented as a 5 × 5 grid of nanomachines. Each nanomachine has an initial value of a certain parameter in the environment. The nanomachines diffuse their estimations during the first epoch. Then, during the next epoch each nanomachine starts to sense molecular concentration to change its initial value. The process of diffusing and sensing molecular concentration continues for each nanomachines until the sensed molecular concentration approaches the average of all initial values.

In PRISM model representation, it is assumed that the process of diffusing and sensing continues until the sensed molecular concentration is less or equal to average of all estimations. Each nanomachine is represented in a separate module. The following example shows the PRISM code of one nanomachine, and how to compute the average of all estimations and its variance.

```
formula av= floor((nit1+init2+init3+ init4+init5+init6+init7+init8+
init9+init10+init11+init12+init13+init14+init15+ init16+init17+init18+
init19+init20+init21+init22+init23+init24+init25)/N);

//node1
module node1

active1:[0..1] init 1; //flag indicates node1 is active and can diffuse
diffuse1:[0..1] init 0; //flag indicates if node1 diffused its estimation or not
sense_again1:[0..1]init 0; //flag indicates that nod1 can sense molecules again
sense1:[0..u]; //sensed molecular concentration by node1 from environment
send1:[0..u];// diffused molecular concentration by node1

//diffuse the estimation of the environment parameter by nod1
[tick1] active1=1& diffuse1=0&sense1>av&send1<=u&x1=2&sense_again1=0->
(send1'=init1)&(sense_again1'=1)&(diffuse1'=1)

//sense environment parameter
[tick2] active1=1& sense_again1=1&sense1<=u &sense1<=u&diffuse1=1->
(sense1'=init1)&(sense_again1'=0)&(diffuse1'=0);

endmodule

formula variance= ((pow((init1-av),2)+pow((init2-av),2)+
pow((init3-av),2)+pow((init4-av), 2)+ pow((init5-av), 2)+
pow ((init6- av),2)+pow((init7-av), 2)+pow((init8-av), 2))+
pow ((init9-av),2))+ pow ((init10-av), 2))+pow((init11-av), 2))+
pow ((init1-av),2))+pow ((init12-av), 2))+pow ((init13-av), 2))+
pow((init14-av),2))+pow((init15-av), 2))+pow((init16-av), 2))+
pow((init17-av),2))+pow ((init18-av), 2))+pow ((init19-av), 2))+
pow ((init20-av),2))+pow ((init21-av), 2))+pow ((init22-av), 2))+
pow ((init23-av),2))+pow ((init24-av), 2))+pow ((init25-av), 2))/(N-1));

//label to be used in the properties formation

label "agree_var"= variance>0 & variance<=1;
```

The initial values *init* are given for each nanomachines at the beginning of the implementation, thus, *av* is computed first. Then, nanomachines diffuse their initial values during the first epoch. The value of *init* is changed during the next epoch when the nanaomachines start sensing molecular concentration from the environment. The transitions [*tick*1], [*tick*2] continue as long as the conditions in the *guard* are true. The maximum molecular concentration that nanomachines can diffuse/sense is assumed to equal u (for the purpose of adding extra constraints into the guard, as each of the used values need to have a specific range).

Two experiments were implemented to represent the model in [59]. The first one is deterministic, and the second is probabilistic, with two *updates*, taking into consideration the noise effects, as explained in subsection 6.4.2.1. The probabilistic experiment was repeated four times with different values of probabilities, starting from 0.5+0.5 and ending with 0.8 + 0.2, as the following example shows:

```
//diffuse the estimation of the environment parameter
[tick1] active15=1&diffuse15=0&sense15>av&send15<=u&x15=2&sense_again15=0->
0.6:(send15'=init15)&(sense_again15'=1)&(diffuse15'=1)+
0.4:(send15'=0)&(sense_again15'=1)&(diffuse15'=1);
```

Property Verification: In each experiment of PRISM model that constructs the described model in [59], property (6.9) is verified, which represents the probability of eventually reaching agreement between the nanomachines according to the described model, based on the defined formula of *variance* defined in PRISM model.

$$Pmin =?[F''agree_{var}''] \qquad (6.9)$$

The following Figure 6.17 shows the verification results of property (6.9) in the 5 × 5 grid of nanomachines, during deterministic and probabilistic experiments.

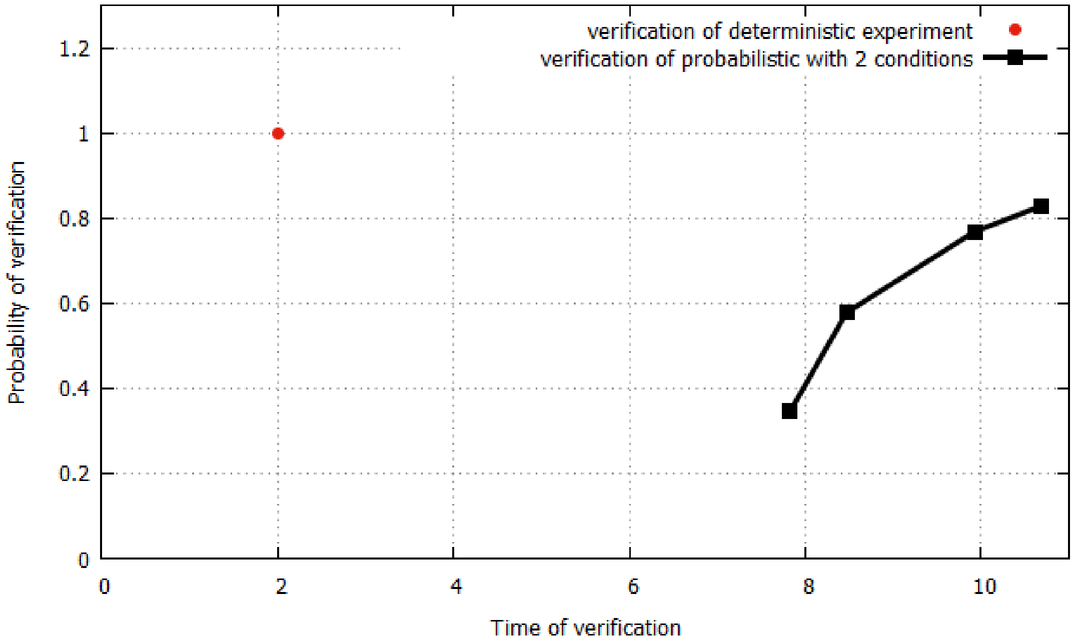


Figure 6.17: Verification of property (6.9) in 5 × 5 Grid of the model in [59]

In the deterministic experiment, the result of verifying 6.9 equals one. In the probabilistic experiment, the verification probability is varied, and each point on the line represented in Figure 6.17 shows the result of a certain experiment. The point with the highest probability represents the experiment with 0.8 + 02 *updates*, as the assumed probability of noise effect is 0.2. The time for verifying property (6.9) is higher compared to the verification time of property (6.7) in 5 × 5 grid in Figure 6.10.

Building PRSIM model Results: The results of building the model in [59] represented as a 5 × 5 nanomachines grid, using PRISM are as follows:

- Deterministic Experiment

  Reachability (BFS): 38 iterations in 0.54 seconds (average 0.00764, setup 0.00)

Time of Model construction	7.568
No. of states	38
No. of transitions	38

Table 6.10: Model Construction 5 × 5 grid - deterministic experiment of [59]

- Probabilistic Experiment with two *updates*

  Reachability (BFS): 38 iterations in 1.35 seconds (average 0.015100, setup 0.00)

Time of Model construction	15.87
No. of states	5150416231214
No. of transitions	8123132215216

Table 6.11: Model Construction 5 × 5 grid - probabilistic experiment of [59]

6.6 Summary

The consensus problem has been explored in time slotted model of $(n \times n)$ nanomachines grid. These nanomachines communicate through diffusion. A protocol which includes number of processes throughout different time epochs was proposed. Then, using PRISM model checker, the acquisition of consensus in the model was verified, in different girds of different sizes. The verification process carried out in three different experiments, the first one is a deterministic experiment, while, the other two experiments take into consideration the effects of noise in the environment and the effects of changing the value of the threshold on the sensed molecular concentration. Thus, the transitions in the represented PRISM model consist of *updates* with probabilities.

The idea of the proposed consensus protocol was inspired by a two-phase commit protocol. In a network of n nodes, in order to apply the dynamic of a two-phase commit protocol, one of the nodes should be considered as master (coordinator), and the rest of the nodes as participants (cohorts). The protocol can be broken down into: first, Commit-Request Phase, in which the coordinator node attempts to prepare (or wakeup) all the participants (the other nodes) to commit. Secondly, Commit Phase, where the coordinator node completes the protocol. In other words, there is a need to broadcast at most three messages in the two-phase commit protocol. The first one is to: requesting to commit, the second one is to: collecting the answers of the participant to commit, and the final one is to: distribute the results of the commit decision [158]. In distributed system, commit protocols are considered as a special type of agreement protocols, one example of these protocols is two-phase commit protocol [94]. However, in the proposed consensus protocol the assumption is that all nanomachines shall 'commit', (i.e.,diffuse their estimations). In a two-phase commit protocol not necessary all the nodes commit to the request of the master node.

In this chapter, besides verifying the proposed protocol, the defined consensus model in [59] has also been verified. As the authors stated, the general case of networks is more difficult to analyse. Thus, PRISM model checker was used to verify the acquisition of consensus in this model.

Chapter 7

Consensus Problem with the Existence of an Adversary Nanomachine

7.1 Introduction

In this chapter the consensus problem in diffusion based molecular communication is considered in a network of n nanomachines. A nanomachine $node_c$ that can control and direct processes in the network is one of the n nanomachines. The considered model is time slotted. An adversary nanomachine $node_A$ is located within the transmission range of the network, where $node_A$ aims to jam the communication among these n nanomachines. The adversary nanomachine $node_A$ is assumed to follow Poisson probabilistic distribution in diffusing its jamming molecules. The n nanomachines need to estimate the concentration of the jamming molecules, in order to improve the possibility of reaching consensus, taking into account the additional jamming molecules in the environment. Thus, during the first k time slots, each nanomachine from n senses (listens to) the jamming molecules diffused by $node_A$, and stores the sensed molecular concentration during each time slot in a vector of length k. Based on the stored molecular concentration in its vector, each nanomachine from n attempts to estimate the average of diffused jamming units. After estimating this value, the processes to reach consensus start. Each nanomachine n has an initial value, the initial values of all nanomachines are diffused to $node_c$. Then $node_c$ computes the average of all initial values. However, $node_c$ also needs to take into account the jamming molecules when it computes the average of initial values. Thus, same as nanomachines n, nod_c is assumed to estimate the jamming molecular concentration during the first k time slot. Then, $node_c$ diffuses the average of the initial values to the other nanomachines n.

Furthermore, this chapter explores the biological process of quorum sensing in bacteria. This process is a form of consensus among bacteria population. In order to activate bacteria to perform its task (whether it is useful or harmful), bacteria need to reach consensus first. Thus, the nanomachines n are employed to prevent a harmful bacteria from launching an attack, through proposing a protocol that nanomachines n follow, and assuming that these nanomachines n are jamming the communication among bacteria, through diffusing a molecule which has been tested in biological experiments to lock the bacteria receptors.

This chapter is organized as follows. In Section 7.2 the proposed model is described. Section 7.3 discusses the process of estimating parameter λ (which is related to the jamming molecules distribution) by all n nanomachines in the network. The steps of the proposed consensus processes are presented in Section 7.4. Section 7.5 explores the biological perspective of bacteria communication, and then presents the proposed protocol to prevent bacteria from launching an attack, through employing the network's n nanomachines. Section 7.6 gives a summary of the discussed issues in the chapter.

7.2 Model

The model described in this chapter is based on the model discussed in Chapter 5:

Network Environment: The considered system in this chapter is a diffusion based network of n nanomachines, one of of which is distinguished as $node_c$ that has some responsibilities for directing and controlling processes in the network. Within the transmission range d_{max} of $node_c$ the other nanomachines are distributed in different positions. The network also includes an adversary nanomachine $node_A$ which diffuses units of molecules with the aim of jamming the communication between the network's n nanomachines. Thus, the network environment contains jamming molecules, besides the noise molecules (i.e., the residual molecules from previous diffusions, and the molecules from other nanomachines, which are not among n).

Communication among Nanomachines: The communication between the n nanomachines is based on diffusing and sensing molecules. In such a way, that each nanomachine $n(i)$ (where, i $\in \{1, 2, ..n\}$) in the network has the ability to sense the concentration of molecules from the environment and to diffuse molecules at a particular rate into the communication medium. Information molecules are encoded based on the variation in the concentration of molecules in the communication environment. Thus, when a nanomachine from $n(i)$ decides to diffuse a unit u of molecules at time t, the other nanomachines at distance d from it can sense the concentration of the diffused molecules through Equation (5.1). In case more than one nanomachine diffuses a unit u of molecules, the receiver nanomachine accumulates the sensed molecular concentration during a specific interval as in Equation (5.2).

Adversary Nanomachine: The adversary nanomachine $node_A$ continues to diffusing jamming molecules. The distribution of the jamming molecules is stochastic. All the other n nanomachines are assumed to be placed in such a way that it can sense the same distribution range of the jamming molecules; thus, the distance between $node_A$ and the other nanomachines n is d_A. The adversary nanomachine is assumed to follow Poisson probabilistic distribution in diffusing its jamming molecules as shown in 7.1.

$$P(j) = \frac{\lambda^j e^{-\lambda}}{j!} \tag{7.1}$$

The number of molecules diffused by adversary nanomachine $node_A$ can vary in each time it diffuses; thus, j represents the observed units of molecular concentration from $node_A$, i.e., $j \in \{0, 1, 2, \cdots, N\}$, while λ represents the average of diffused molecular concentration (units of molecules) by $node_A$. $P(j)$ represents the diffusion rate of $node_A$ and the sensed molecular concentration by the other nanomachines n.

The other nanomachines n in the network need to estimate the value of λ in order to improve the possibility of reaching consensus, taking into account the additional jamming molecules in the environment. Thus, during the first part of the model, the nanomachines n including $node_c$ are assumed to sense the diffused molecular concentration for k time slots (each one with the length T_0), and store these molecular concentration in a vector. The statistics to estimate the value of λ show that it is the arithmetic mean of the samples [41].

$$\lambda = \frac{\text{summation of samples}}{\text{number of samples}} \tag{7.2}$$

Consensus Problem Each nanomachine $n(i)$ has an initial value. Through k time slots, each nanomachine $n(i)$ diffuses its initial value to nod_c. The accumulative molecular concentration of the initial values and the jamming molecules from $node_A$ are sensed by $node_c$. Thus, $node_c$ stores the sensed molecular concentration in a vector of length k. Then, $node_c$ computes the average of the sensed molecular concentration during the k time slots, while attempting to exclude the jamming molecules from the sensed initial values. Then, $node_c$ computes the average of the estimated initial values and diffuses it to the other nanomachines $n(i)$. The initial values of each nanomachine from $n(i)$ are assumed to be relatively close to the value of a known global parameter u in the network. Thus, consensus is reached, if the average of the initial values sensed by each nanomachine from $n(i)$ has a deviation of ϵ compared to the global parameter u. Although only the sum of the sensed jamming molecules is needed; however, the sensed jamming molecules in each time slot are stored in a vector. This can enable nanomachines to perform more specific computations that require a comparison between the received molecular concentration for example.

Time Slots Length: All the nanomachines n in the network are assumed to be synchronized, and can communicate in a predefined time slot T_0. Where T_0 is a system parameter and its length depends on the network's geometric properties, such that $T_0 = v\frac{d_{max}^2}{D}$, where v is a constant that can be equal 1; d_{max} is the transmission range distance of $node_c$, and D is the diffusion coefficient. During the first k time slots, all nanomachines sense molecular concentration from $node_A$ to estimate λ. Then, after estimating λ, the process of consensus and the diffusion of the initial values to $node_c$ begins, each node $n(i)$ diffuses its initial value to nod_c through k time slots. However, in order to avoid Inter Symbol Interference, each nanomachine $n(i)$ waits for at least $v\frac{d^2_{max}}{D}$ before diffusing its initial value again.

7.3 Estimating λ by Nanomachines n

Each nanomachine from n is assumed to recognize that the distribution of the jamming molecules from $node_A$ is based on Poisson distribution. Thus, each nanomachine from n aims to sense the jamming molecules through k time slots and stores the sensed molecular concentration in a vector, in order to estimate λ. Through the observation of the sensed molecular concentration, each nanomachine from n attempts to estimate λ. The following steps show the sensed molecular concentration by a nanomachine from n when an adversary nanomachine $node_A$ diffuses units of molecules u_A, during each time slot ts from the observed k time slots. The molecular concentration is stored in a *vector* of the nanomachine $n(i)$.

```
$ts \leftarrow 0$;
$t \leftarrow 0.1$;
while $ts \leq k$ do
    while $(c(u_A, d_A, T_0) \geq \tau)$ & $(t \leq T_0)$ do
        $c(u_A, d_A, T_0) = \int_t^{T_0} u_A \cdot \frac{1}{(4\pi Dt)^{\frac{2}{3}}} \cdot \exp(\frac{-d_A^2}{4Dt})$
    $vector[ts] = c(u_A, d_A, T_0)$;
    $ts = ts + 1$;
```

As equation 7.2 shows that λ is the mean of samples. The maximum likelihood of λ estimation can be as follows [73] :

$$P(u_A) = \frac{e^{-\lambda}\lambda^{u_A}}{u_A!}$$

Where u_A= $\{0, 1, 2, \cdots, N\}$, j=$\{1, 2, \cdots, k\}$. Thus, using the log likelihood, the function would be:

$$\begin{aligned} l(\lambda) &= \log \prod_{j=0}^{k} P(u_A(j), \lambda) \\ &= \sum_{j=1}^{k} \log \frac{e^{-\lambda}\lambda^{u_A(j)}}{u_A(j)!} \\ &= \sum_{j=1}^{k} \log e^{-\lambda} + \sum_{j=1}^{k} u_A(j) \log \lambda - \sum_{j=1}^{k} \log u_A(j)! \end{aligned}$$

Then, the first derivative is used to find the maximum likelihood of λ estimation:

$$\frac{\partial l(\lambda)}{\partial \lambda} = 0$$
$$\frac{\partial \sum_{j=1}^{k} \log e^{-\lambda} + \sum_{j=1}^{k} u_A(j) \log \lambda - \sum_{j=1}^{k} \log u_A(j)!}{\partial \lambda} = 0$$
$$-k + \frac{1}{\lambda} \sum_{j=1}^{k} u_A(j) = 0$$
$$\lambda = \frac{1}{k} \sum_{j=1}^{k} u_A(j)$$

i.e.

$$\lambda = \frac{\sum_{j=1}^{k} u_A(j)}{k} \tag{7.3}$$

Thus, each nanomachine from n computes the average of the sensed molecular concentration during k time slots, and considers it to be an estimation of λ. To compute the sensed molecular concentration from other n nanomachines diffusions, each nanomachine $n(i)$ deducts the estimated value of λ from the total amount of the sensed molecular concentration during a certain time slot T_0. Thus, in the case of communication between nanomachines n, each nanomachine $n(i)$ can sense units of molecular concentration u_n diffused by the other nanomachines n, plus the jamming molecules from the adversary nanomachine u_A. If the total amount of the sensed molecular concentration by a nanomachine $n(i)$ during a certain time slot T_0 is R_i, then nanomachine $n(i)$ can assume that:

$$u_{n(i)} \approx \begin{cases} R_i - (\lambda + \varepsilon), & \text{if } \varepsilon \leq \frac{1}{k} \\ R_i - (\lambda - \varepsilon), & \text{if } \varepsilon \geq 1 - \frac{1}{k^2} \end{cases} \tag{7.4}$$

where ε is also estimated by each nanomachine based on the number of nanomachines n in the network, during estimating λ, in such a way that:

$$\varepsilon \begin{cases} \leq \frac{1}{k}, & \text{if } k \leq n \\ \geq 1 - \frac{1}{k^2}, & \text{otherwise} \end{cases} \tag{7.5}$$

In [128], Chernoff bound is used for Poisson random variable estimation in **Theorem 5.4**, which shows that:

1. If $u_A > \lambda$ then
$$Pr(X \geq u_A) \leq \frac{e^{\lambda}(e\lambda)^{u_A}}{u_A^{u_A}};$$
2. If $u_A < \lambda$, then
$$Pr(X \leq u_A) \leq \frac{e^{\lambda}(e\lambda)^{u_A}}{u_A^{u_A}}.$$

7.4 Consensus Processes

After waiting for k time slots to estimate λ (so that the jamming molecules from $node_A$ are taken into account in the communication among nanomachines n), the processes to reach consensus begins. Each nanomachine $n(i)$ has an initial value $\zeta(i)$. Starting from time slot $k+1$ each nanomachine $n(i)$ diffuses its initial value to $node_c$. Thus, each nanomachine $n(i)$ raise its initial value according to Equation (5.5) as explained in Chapter 5. In the meantime, $node_c$ follows the steps described in Algorithm 11 in order to sense the accumulative molecular concentration of initial values from each $n(i)$ plus the jamming molecular concentration from the adversary nanomachine $node_A$. Each nanomachine $n(i)$ keeps diffusing its initial value to nod_c through k time slots, and $node_c$ keeps storing the sensed molecular concentration in its vector. However, these k time slots are not consecutive, as a nanomachine $n(i)$ diffuses its initial value during a time slot, then waits for at least T_0, before diffusing its initial value again, where, $node_c$ does not store the sensed molecular concentration during the waiting time slot as it is mainly jamming units. Thus, the sensed molecular concentration by $node_c$ during k time slots represents R_c:

$$R_c = (\sum_{i=0}^{n} \zeta(i) + u_A 1) + (\sum_{i=0}^{n} \zeta(i) + u_A 2) + \cdots + (\sum_{i=0}^{n} \zeta(i) + u_A k) \tag{7.6}$$

$$R_c = \sum_{j=1}^{k} (\sum_{i=0}^{n} (\zeta(i) + u_A j)) \tag{7.7}$$

where $(\zeta(i))$ is the initial value of nanomachine $n(i)$, and u_A is the amount of the jamming molecules from $node_A$. After waiting for $2k$ time slots, and sensing molecular concentration during one of these k time slots, nod_c estimates the initial values from the total stored molecular concentration in its vector that has been computed in R_c.

$$InitR_c = \frac{R_c - \sum_{i=1}^{k} u_A i}{k} \tag{7.8}$$

i.e.

$$InitR_c \approx \begin{cases} \frac{R_c - (k\lambda + \varepsilon)}{k}, & \text{if } \varepsilon \leq \frac{1}{k} \\ \frac{R_c - (k\lambda - \varepsilon)}{k}, & \text{if } \varepsilon \geq 1 - \frac{1}{k^2} \end{cases} \tag{7.9}$$

Then, during $2k+1$ slot, $node_c$ computes the average of the estimated initial values $InitR_c$, assuming that $node_c$ knows n the number of nanomachines in the network. Otherwise estimating the number of nanomachines as described in Subsection 5.3.1.1 of Chapter 5. However, here $node_c$ should take into consideration the jamming molecules while estimating the number of nanomachines according to Subsection 5.3.1.1. Thus, the average of the estimated initial values can be:

$$Init_{av}R_c = \frac{InitR_c}{n-1} \tag{7.10}$$

The estimated initial values $InitR_c$ is divided by $(n-1)$, as nod_c is assumed to be one of the n (non-jamming) nanomachines in the network. During the next time slot, $node_c$ diffuses $Init_{av}R_c$ to the other nanomachines, through the same steps described in Algorithm 10.

In the meantime, the other nanomachines attempt to sense the molecules diffused by $node_c$ (through the steps described in Algorithm 11), in order to recognize the average of all initial values. As explained in Chapter 5, each nanomachine from $n(i)$, identified the effects of distance on the received concentration from $node_c$; thus, each nanomachine $n(i)$ counts on its $\rho(i)$ to distinguish the right value of $Init_{av}R_c$, as in Equation (5.8). However, besides $\rho(i)$, here each nanomachine from $n(i)$ takes into consideration the jamming molecular concentration and the estimated value of λ, as shown in the following:

$$Init_{av}R_c \approx \begin{cases} \rho(i) \times Init_{av}R_c(i)_{received} - (\lambda + \varepsilon), & \text{if } \varepsilon \leq \frac{1}{k} \\ \rho(i) \times Init_{av}R_c(i)_{received} - (\lambda - \varepsilon), & \text{if } \varepsilon \geq 1 - \frac{1}{k^2} \end{cases} \tag{7.11}$$

where, $Init_{av}R_c(i)_{Received}$ is the sensed (received) molecular concentration by a nanomachine $n(i)$ when $node_c$ diffuses the estimated average of initial values.

Reaching Consensus: The model assumes that consensus is reached when the average of the initial values $Init_{av}R_c(i)$ approaches to the value of u with ϵ deviation, i.e., the deviation of average of the initial values from the assumed global parameter u is:

$$|u - Init_{av}R_c(i)| \leqslant \epsilon \tag{7.12}$$

where, $\epsilon > 0$. As the initial values of each nanomachine from $n(i)$ are assumed to be relatively close to the value of u, in order to increase the possibility to reach an agreement among the networks' nanomachines.

Time to Reach Consensus In order to compute the time needed to compute the average of all initial values $Init_{av}R_c$, the time needed to estimate λ and the time required for the consensus processes, should be taken into consideration. Thus, the first k time slots needed to estimate λ are multiplied by the length of the time slot, i.e., T_0. Besides this, the next $2k$ time slots needed for diffusing the initial values of nanomachines $n(i)$ with the two time slots required to compute the average of the initial values and then diffusing it by $node_c$, are also multiplied by T_0.

$$t_{Init_{av}R_c} = (T_0 \times k) + (T_0 \times (2k+2)) \tag{7.13}$$

7.5 Preventing Consensus among Bacteria

Quorum sensing process [77, 142] is an example of signalling between bacteria, where bacteria can use it to estimate the density of their population in the environment through estimating the concentration of a certain type of molecules. In [2, 4] quorum sensing was proposed as a valid tool for signal amplification and synchronization between nodes in a nanonetwork, respectively.

7.5.1 Communication among Bacteria: Biological Background

Bacteria are one cell organisms, their life is based on nutrition consumption, then growing and dividing into two cells, and that process continues [49]. However, bacteria in its different types interact with human body. Bacteria have very important roles in human health, but there are types of bacteria that can cause terrible harm to the human body [142]. However, one bacterium cannot do any harmful or useful role inside the human body, it needs to communicate with other cells of its type first by releasing a certain type of molecules (known as autoinducer) [77]. As the number of bacteria increases the concentration of that molecule increases in the environment, when the concentration of this molecule reaches a certain threshold, that can allow the bacteria to recognize how many neighbours it has. This process is known as quorum sensing [147]. The process of quorum sensing can be considered as a consensus between bacteria, i.e., each bacteria vote by emitting this molecule, then the vote is counted, and after this all bacteria respond to the vote [174]. Thus, the harmful bacteria when they get inside the human body, wait until their number increases, and through this specific molecule they count its population, which is when bacteria recognize that they have the right population size, make all the bacteria release their toxin in a synchronized way[174].

From the chemistry point of view, the autoinducer molecule in different types of bacteria consists of two parts, the left part of this molecule in every single type of bacteria is identical, but the right part is slightly different in every single type of bacteria [32, 67, 77, 142, 162, 166]. This shows, that each species of bacteria has its own language of communication and that it uses a specific molecule which can fit into a specific receptor of bacteria from the same type [142, 174]. However, bacteria can communicate with all other types of bacteria, by having a generic molecule which can be sensed by specific receptors on all types of bacteria [32, 67, 162, 166, 174].

Molecular biologist Bonnie Bassler and her team [32, 67, 77, 142, 162, 166, 174] were able to make molecules which look like the molecules that bacteria release for communication but with a little difference. These molecules lock bacteria receptors, and prevent bacteria recognizing the real number of their population, thus, preventing the bacteria from launching their attack. Thus, these molecules can be considered as anti-quorum sensing molecules, eventually can think about these molecules as antibiotics.

7.5.2 Anti-Quorum Sensing Nanomachines

Suppose that the model discussed in Section 7.2 without the adversary nanomachine, in which $n(i)$ nanomachines will attempt to jam the communication between bacteria through diffusing the modified autoinducer molecules (i.e. the anti-quorum sensing molecules). Further more, $node_c$ is responsible for detecting the release of the autoinducer molecule from the bacteria, and then triggering the other nanomachines $n(i)$ to start diffusing the modified autoinducer to jam quorum sensing and eventually prevent bacteria from launching their attack.

Consider that Q molecules are diffused by bacteria at time t into the environment with D diffusion coefficient. The molecular concentration $c$sensed by a bacterium (or a nanomachine) at distance d at certain time t is [37]:

$$c(Q,d,t) = \frac{2Q}{(4\pi Dt)^{\frac{3}{2}}} \exp \frac{-d^2}{4Dt} \tag{7.14}$$

When (for example) a nanomachine releases molecules, generally the molecular concentration at any given distance gradually decreases over time. However, in the case that a nanomachine continues to emit molecules (which is what happen in quorum sensing), then the molecular concentration will keep increasing. If a nanomachine continues on diffusing molecules for a long time, the molecular concentration could approach[4]:

$$c(Q,d) = \frac{Q}{4D\pi d} \tag{7.15}$$

Thus, bacteria constantly release autoinducer, until reaching a certain threshold τ_{qs}. This threshold is an important parameter for activating bacteria (i.e., launching their attack, in case of considering a harmful bacteria type) in quorum sensing phase; thus, threshold τqs value could approach the maximum achievable molecular concentration $c(Q,d)$ [4].

$$\tau_{qs} \approx c(Q,d) = \frac{Q}{4D\pi d} \tag{7.16}$$

Molecular concentration of autoinducer at a certain point d in the medium is the accumulated molecular concentration from the diffusion of all the bacteria N in the environment [4]:

$$c(Q,d,N) = \sum_{i \in N} c(Q,d_i) \tag{7.17}$$

Where d_i is the distance between the point d and the other bacteria in the medium:

$$\sum_{i \in N} (d_i - d)^2 \tag{7.18}$$

Thus, bacteria will be activated when molecular concentration approaches threshold τ_{qs}, and the time needed for molecular concentration at distance d approaches τ_{qs} is:

$$t_{\tau_{qs}}(d) \approx c(Q,d,N) = \sum_{i \in N} c(Q,d_i) \tag{7.19}$$

Steps to Prevent Quorum Sensing: Molecular biologist Bonnie Bassler and her team [32, 67, 162, 166] injected the harmful bacteria into a mouse and at the same time injected the modified autoinducer molecule. At this stage, the nanomachines $n(i)$ of the defined model are deployed to diffuse the modified autoinducer.

1. $node_c$ will attempt to monitor the environment within its transmission range. If it senses molecular concentration of autoinducer greater than or equal to threshold τ, it attempts to activate the other nanomachine $n(i)$ to diffuse the modified autoinducer. The receptors of $node_c$ are assumed to be able to distinguish between the real autoinducer molecule and the modified one.

2. Thus, $node_c$ diffuses unit u of the modified autoinducer to nanomachines $n(i)$, through time slot T_0, (T_0 is less than $t_{\tau_{qs}}(d)$). $node_c$ follows the steps in Algorithm 10 to diffuse unit u.

3. The other nanomachines $n(i)$ are activated when they sense the modified autoinducer. Then, nanomachines $n(i)$ attempt to diffuse their modified autoinducer into the environment following Poisson random distribution for k time slots (i.e. kT_0).

4. Meanwhile, $node_c$ attempts to estimate $t_{\tau_{qs}}(d)$. If kT_0 is less than the estimated $t_{\tau_{qs}}(d)$, then $node_c$ diffuses the modified autoinducer molecule again after kT_0, in order to activate nanomachines $n(i)$ to activate them again.

5. Nanomachines $n(i)$ follow Poisson distribution to diffuse their modified autoinducer again for kT_0.

6. After the next kT_0 finish, $node_c$ waits for $2T_0$ and attempt to sense the molecular concentration of the real autoinducer. If it is greater than or equal to the threshold τ, then it repeats the above steps; if it is not, nod_c keeps monitoring the environment for each T_0.

These steps are assumed to be in a specific transmission range of $node_c$, it is possible to assume that different networks are deployed in different positions. However, to present a more general way of preventing bacteria from launching their attacks, the important topic of nanomachines motility needs to be explored; in order to assume that theses nanomachines are moving in different positions and attempt to monitor the environment. Thus, this can be a motivation for future research.

7.6 Summary

In this chapter, consensus problem in diffusion based network is explored with the existence of an adversary nanomachine. The adversary nanomachine follows Poisson random distribution in diffusing molecules to jam the communication among the network's n nanomachines. The network's nanomachines attempt to estimate the jamming molecular concentration, through sensing molecular concentration for k time slots and storing this value in a vector. Then, the network nanomachines compute the average of the sensed jamming molecular concentration during k time slots, and take in consideration estimation error ε. The consensus processes start after estimating the jamming molecular concentration. Moreover, this chapter discusses a biological process that can be considered as consensus between bacteria. This process is the key to activate bacteria to launch an attack, or to do something useful (depending on the bacteria type). Then, the network's n nanomachines of the model are employed to perform a protocol to prevent bacteria from reaching consensus and eventually from performing harmful activities.

Chapter 8

Conclusions and Future Work

Different communication techniques are proposed for information exchange among nanomachines. Molecular communication is one of these techniques, it is a bio-inspired communication mechanism. The characteristics and rules that govern molecular communication are motivated by communication in biological systems. The main goal of this dissertation is to explore and analyse molecular communication. Through proposing models and defining different scenarios to study channel transmission reliability, the distance that transmitted molecules can reach, the effects of medium noise, the effects of interference between the current transmitted molecules and previous transmitted molecules, nanomachines energy, consensus problem, and security issues in communication among nanomachines.

In this chapter, Section 8.1 summarizes the contributions of the dissertations. Section 8.2 discuses the main the findings of the desertification and gives a summary of the work. Section 8.3 includes suggestions for future directions and explorations.

8.1 Contributions

The main contributions of the work in this dissertation are as follows:

- Receiving acknowledgement in molecular communication channel is verified using PRISM model checker.

- The parameters that can affect the maximum distance that a diffused molecule can reach, and the parameters that have an impact on the pattern of diffusion recognition, are explored through experimentation.

- A consensus protocol among nanomachines is proposed. The steps of the protocol are implemented twice: after electing a leader nanomachine in the network, then after proposing an energy harvesting model.

- Reaching a consensus among $n \times n$ nanomachines is verified using PRISM model checker.

- The consensus problem in [59] is verified using PRISM model checker, through assuming a general case of $n \times n$ nanomachines, deployed arbitrarily in the environment.
- The consensus problem in diffusion based network with the existence of an adversary nanomachine is presented, in an attempt to explore security issues in molecular communication.
- In terms of security relevant issue, a protocol is proposed in order to apply a nanonetwork to jam the communication among bacteria, to prevent them from launching an attack.

8.2 Main Findings and Contributions

Channel reliability and the insurance that the transmitted molecules are sensed (received) by the receiver nanomachine are an important challenge in molecular communication. In Chapter 3 the channel between the transmitter and receiver nanomachines is explored. The channel between nanomachines is assumed to consist of a number of locations (nodes) placed at a distance between the transmitter and receiver nanomachimes. The mechanism of the channel node is based on the sandpile model. To analyse the proposed model and the communication channel, a PRISM model checker was utilized to verify the reception of acknowledgement in one dimensional channel. The verification experiments were carried out on different sized channels. Then, the process of verification was repeated on the same model, taking into account the effects of the medium noise. Furthermore, the model has extended to represent a bi-directional and multi-access channel. Then, PRISM model checker was employed to check the probability of success/ failure of both transmission/ receiving information molecules in the proposed model. The results demonstrate that receiving acknowledgement can be affected by the size of the channel between the transmitter and receiver. The effects of noise and the amount of transmitted information molecules on the channel reliability are also demonstrated. The verification results show that PRISM model checker can be utilized in studying molecular communication models. However, building and verifying more complex models with larger sized networks requires longer time.

The propagation of molecules in the communication medium is a significant topic to be explored. In order to study the effects of noise, the residual molecules from previous communications and properties of the medium itself, on the sensed molecules by the receiver nanomachine(s). In Chapter 4 the issues related to the performance of nanomachines were studied. In a diffusion based molecular communication network an algorithm to measure the maximum distance that transmitted molecules can reach, has been proposed, taking into consideration medium noise and assuming that the transmitter nanomachine continues diffusing during a defined time interval. The experiment results of the proposed algorithm show the effects of noise and the medium diffusion coefficient on the range of the maximum distance of the diffused molecules. Moreover, the pattern of diffusion is explored, by inspecting how a receiver nanomachine could distinguish a message from one transmitter nanomachine at a distance d or two transmitter nanomachines at different distances. Through the implemented experiments, the effects of distance, time, sensed molecular concentration and interference are considered.

The experimental results to compute the transmission range show that the transmission range decreases as the value of both noise and residual molecular concentration increase. The transmission range decreases as the value of the medium diffusion coefficient increases. The higher transmission range is achieved with the message that consists of higher number of '1'. The experimental results of diffusion pattern recognition were as follows, the diffused symbols were not recognized correctly due to the short symbol duration. Then the receiver nanomachine was able to recognize the symbols correctly after increasing the symbol duration. The other case is two different transmitter nanomachines at different distances from the receiver nanomachines diffuse information molecules. The experiment results to check how the receiver nanomachine can recognize the pattern of diffusion show that the receiver nanomachine can recognize the diffused symbols correctly. However, it cannot distinguish whether symbol came from one nanomachine or another. It is due to the overlapping in the values of the sensed molecular concentration which come from each nanomachine in this experiment. The reason of the overlapping in these values came from the close distance between the two transmitter nanomachines. Changing the distance of the two transmitter nanomachines from the receiver nanomachine can make the overlapping quite low. In case one transmitter nanomachine is in a close distance from the receiver nanomachine. And the other nanomachine is in a far distance from the receiver nanomachine. The results are as follow, the receiver nanomachine cannot recognize the diffused symbols correctly. Due to the higher data rate of the sensed molecular concentration from the close transmitter nanomachine. Thus, the sensed molecular concentration from the far nanomachine is affected by the interference of molecules from the previous symbol duration. By increasing the symbol duration, the receiver nanomachine can recognize the diffused symbols correctly. Beside distinguishing if the symbol came from one nanomachine or another.

The consensus problem is essential in any distributed system to fulfil an overall agreement or commitment to perform tasks. Thus, reaching consensus among nanomachines in molecular communication is an important topic. Chapter 5 studies consensus problem in diffusion based molecular communication have been presented. Inspired by the consensus model in [59] a consensus protocol has been proposed. Each step of the protocol is performed in a specific round, and there is a waiting round after each step to reduce the effects of interference that comes from molecules of previous diffusion. The consensus protocol steps are initiated after estimating the number of nanomachines in the network through a special node in the network. To implement the consensus protocol, two scenarios are considered, (1) a network without special node that control the protocol steps; thus, a leader election algorithm is adopted to elect that special node;

and (2) a network with energy constraint nanomachines; thus, an energy harvesting algorithm is proposed. In both scenarios, the number of rounds required to implement the protocol steps were computed.

The conducted experiments include: Compute the number of nodes in the nanonetwork. Compute the sensed molecular concentration by each nanomachine during each time slot. Compute the total number of time slots that are needed to perform the protocol steps. The results show the effects of energy constraint on the needed time slots number.

In Chapter 6 the consensus problem in time slotted model of a grid of $(n \times n)$ nanomachines has been explored. The communication among these nanomachines is through diffusing and sensing molecules. This model is also inspired by the consensus problem described in [59], but the aim here is to study this problem in multi-dimensional model. The acquiring of consensus in the model is verified using PRISM model checker. The authors in [59] stated that the general case of networks is more difficult to analyse. Thus, PRISM model checker was also used to verify reaching consensus in their model. However, the nodes are assumed to be deployed arbitrarily, and not necessarily that each node observes the same distances to the other nodes in the network, as considered in [59]. The experiments results are: The acquiring of consensus in the model is verified using PRISM model checker. The consensus problem in [59] is verified using PRISM model checker, through assuming a general case of $n \times n$ nanomachines, deployed arbitrarily in the environment.

In Chapter 7 the consensus problem in a diffusion based network is explored with the existence of an adversary nanomachine, which aim is to jam the communication between the network nanomachines. The adversary nanomachine follows Poisson random distribution in diffusing its jamming molecules. The network nanomachines attempt to estimate the concentration of the jamming molecules. Thus, through k time slots, each nanomachine senses the molecular concentration and stores it in a vector. Then, each nanomachine attempts to estimate the average of the jamming molecular concentration, based on the stored molecular concentration during k time slots. After estimating the jamming molecular concentration, the processes to reach consensus begin, where, each nanomachine in the network has an initial value. Each nanomachine diffuses its initial values to a special node in the network. This special node computes the average of all initial values and diffuses it to the network nanomachines. The special node is assumed to estimate the jamming molecular concentration in the same way and during the same interval that the network nanomachine attempted to estimate it. Thus, the special node takes in consideration the jamming molecular concentration when it computes the average of all initial values. Furthermore, this chapter explores the biological process of quorum sensing in bacteria. This process is a form of consensus among bacteria population. In order to activate bacteria to perform its task (whether it is a useful or harmful), bacteria need to reach consensus first. Thus, the nanomachines in the defined model are employed to prevent a harmful bacteria from launching their attack. Through proposing a protocol that nanomachines follow. Thus, these nanomachines attempt to jam the communication among bacteria, through diffusing a molecule which has been tested in biological experiments to lock the bacteria receptors.

8.3 Future Work

There are still many unaddressed issues and challenges in this area. Future work can be inspired from the following potential research issues:

- **Diffusion with Drift**: to explore models of nanonetworks taking into consideration the drift of molecules in the medium. Research studies that address flow-based molecular communication are currently quite limited. The propagation medium in some of the nanonetworks applications can be in motion, as an example, nanomachines in biomedical applications are placed in human blood. Thus, it is not feasible to assume that the propagation medium is always stable. In some cases, a drift velocity may be applied on purpose to increase a molecular communication systems low throughput. Thus, it would be good to study model of molecular communication with drift as well as channel capacity, noise effects and other related issues.

- **Nanomachines motility**: As mentioned in Chapter 7, nanomachine movement is a very significant issue, especially when thinking about the medical applications of nanonetworks. Thus, the subject of cell motility needs to be explored and a way needs to be found to formulate the rules and mechanisms that cells use to move, which can then be employed it into nanomachines.

- **Security Protocol for Molecular Communication**: As explained in Chapter 1, the application domains of nanonetworks are quite critical and thus, any changes in the functionality of nanomachines can cause huge consequences [55]. Therefore, more explorations of security issues in molecular communication are needed, with the objective of proposing a security protocol. This can be acheived either by combining and enhancing existing relevant security protocols, or through studying the immune system in the human body, and then attempting to formulate the immune system mechanism into a security protocol for nanomachines.

- **Mutual Exclusion**: A proposal for future work is to consider mutual exclusion property on multiple access molecular communication channel. The aim is to design an algorithm to control the channel access by the nanomachines in the network, inspired by Peterson's algorithm (for example).

- **Theoretical Work**: There is a large scope for theoretical work to develop a theoretical methodology for analysing nanonetwork protocols. In all the above topics there is very little of theoretical analysis of algorithms and processes, because they are complex. However, it would be very helpful to have theoretically-proven and guaranteed properties of algorithms and processes in nanonetworks, especially for large-scale systems for which it is difficult to run simulations.

- **Verification Tools**: In order to analyse more complex models and larger networks sizes, a verification tool that can handle the amount of computation is needed. Such as dReach and dReal, where, dReach is a tool for safety verification of hybrid systems, and dReal is an automated reasoning tool [39].

Bibliography

[1] Sergi Abadal and Ian F Akyildiz, *Automata modeling of quorum sensing for nanocommunication networks*, Nano Communication Networks **2** (2011), no. 1, 74–83.

[2] ———, *Bio-inspired synchronization for nanocommunication networks*, Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE, IEEE, 2011, pp. 1–5.

[3] Sergi Abadal, Ignacio Llatser, Eduard Alarcón, and Albert Cabellos-Aparicio, *Quorum sensing-enabled amplification for molecular nanonetworks*, Communications (ICC), 2012 IEEE International Conference on, IEEE, 2012, pp. 6162–6166.

[4] ———, *Cooperative signal amplification for molecular communication in nanonetworks*, Wireless networks **20** (2014), no. 6, 1611–1626.

[5] Bilal Acar, Ali Akkaya, Gaye Genc, H Birkan Yilmaz, M Şükrü Kuran, and Tuna Tugcu, *Understanding communication via diffusion: Simulation design and intricacies*, Modeling, Methodologies and Tools for Molecular and Nano-scale Communications, Springer, 2017, pp. 139–163.

[6] Yehuda Afek, Noga Alon, Ziv Bar-Joseph, Alejandro Cornejo, Bernhard Haeupler, and Fabian Kuhn, *Beeping a maximal independent set*, Distributed computing **26** (2013), no. 4, 195–208.

[7] Ali Akkaya, H Birkan Yilmaz, Chan-Byoung Chae, and Tuna Tugcu, *Effect of receptor density and size on signal reception in molecular communication via diffusion with an absorbing receiver*, IEEE Communications Letters **19** (2015), no. 2, 155–158.

[8] Ian F Akyildiz, Fernando Brunetti, and Cristina Blázquez, *Nanonetworks: A new communication paradigm*, Computer Networks **52** (2008), no. 12, 2260–2279.

[9] Ian F Akyildiz, Faramarz Fekri, Raghupathy Sivakumar, Craig R Forest, and Brian K Hammer, *Monaco: fundamentals of molecular nano-communication networks*, IEEE Wireless Communications **19** (2012), no. 5, 12–18.

[10] Ian F Akyildiz and Josep Miquel Jornet, *Electromagnetic wireless nanosensor networks*, Nano Communication Networks **1** (2010), no. 1, 3–19.

[11] IAN F AKYILDIZ and JOSEP MIQUEL JORNET, *The internet of nano-things*, IEEE Wireless Communications (2010), 58–63.

[12] Ian F Akyildiz, Josep Miquel Jornet, and Massimiliano Pierobon, *Propagation models for nanocommunication networks*, Antennas and Propagation (EuCAP), 2010 Proceedings of the Fourth European Conference on, IEEE, 2010, pp. 1–5.

[13] Bruce Alberts, Dennis Bray, Julian Lewis, Martin Raff, Keith Roberts, and James D Watson, *Molecular biology of the cell. 3rd*, New York: Garland Pub **43** (1994), no. 1294, 67.

[14] Muhammad Azam Ali, Iqra Rehman, Adnan Iqbal, S Din, Abdul Qayyum Rao, Ayesha Latif, Tahir Rehman Samiullah, Saira Azam, and Tayyab Husnain, *Nanotechnology, a new frontier in agriculture*, Adv. life sci **1** (2014), no. 3, 129–138.

[15] EJ Ariens et al., *Affinity and intrinsic activity in the theory of competitive inhibition. 1. problems and theory.*, Archives internationales de pharmacodynamie et de thérapie **99** (1954), 32–49.

[16] Dogu Arifler, *Capacity analysis of a diffusion-based short-range molecular nano-communication channel*, Computer Networks **55** (2011), no. 6, 1426–1434.

[17] Hamidreza Arjmandi, Amin Gohari, Masoumeh Nasiri Kenari, and Farshid Bateni, *Diffusion-based nanonetworking: A new modulation technique and performance analysis*, IEEE Communications Letters **17** (2013), no. 4, 645–648.

[18] Barış Atakan, *Molecular communications and nanonetworks: from nature to practical systems*, Springer Science & Business Media, 2014.

[19] Baris Atakan and Ozgur B Akan, *An information theoretical approach for molecular communication*, Bio-Inspired Models of Network, Information and Computing Systems, 2007. Bionetics 2007. 2nd, IEEE, 2007, pp. 33–40.

[20] ———, *On channel capacity and error compensation in molecular communication*, Transactions on computational systems biology X, Springer, 2008, pp. 59–80.

[21] ———, *On molecular multiple-access, broadcast, and relay channels in nanonetworks*, Proceedings of the 3rd International Conference on Bio-Inspired Models of Network, Information and Computing Sytems, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, p. 16.

[22] ———, *Carbon nanotube sensor networks*, Proc. of IEEE Nanocom (2009), 1–6.

[23] ———, *Single and multiple-access channel capacity in molecular nanonetworks*, International Conference on Nano-Networks, Springer, 2009, pp. 14–23.

[24] ———, *Carbon nanotube-based nanoscale ad hoc networks*, IEEE Communications Magazine **48** (2010), no. 6, 129–135.

[25] ———, *Deterministic capacity of information flow in molecular nanonetworks*, Nano Communication Networks **1** (2010), no. 1, 31–42.

[26] Luigi Atzori, Antonio Iera, and Giacomo Morabito, *The internet of things: A survey*, Computer networks **54** (2010), no. 15, 2787–2805.

[27] Per Bak, Chao Tang, and Kurt Wiesenfeld, *Self-organized criticality: An explanation of the 1/f noise*, Physical review letters **59** (1987), no. 4, 381.

[28] Sergio Barbarossa and Gesualdo Scutari, *Bio-inspired sensor network design*, IEEE Signal Processing Magazine **24** (2007), no. 3, 26–35.

[29] Michael Taynnan Barros, *Ca2+-signaling-based molecular communication systems: Design and future research directions*, Nano Communication Networks (2017), 103–113.

[30] Michael Taynnan Barros, Sasitharan Balasubramaniam, and Brendan Jennings, *Comparative end-to-end analysis of ca 2+-signaling-based molecular communication in biological tissues*, IEEE Transactions on Communications **63** (2015), no. 12, 5128–5142.

[31] Michael Taynnan Barros, Sasitharan Balasubramaniam, Brendan Jennings, and Yevgeni Koucheryavy, *Transmission protocols for calcium-signaling-based molecular communications in deformable cellular tissue*, IEEE Transactions on Nanotechnology **13** (2014), no. 4, 779–788.

[32] Bonnie L Bassler, *Small talk: cell-to-cell communication in bacteria*, Cell **109** (2002), no. 4, 421–424.

[33] Marco Bernardo, Pierpaolo Degano, and Gianluigi Zavattaro, *Formal methods for computational systems biology: 8th international school on formal methods for the design of computer, communication, and software systems, sfm 2008 bertinoro, italy, june 2-7, 2008*, vol. 5016, Springer, 2008.

[34] Michael J Berridge, *The am and fm of calcium signalling*, Nature **386** (1997), no. 6627, 759.

[35] Elvin Blanco, Angela Hsiao, Guillermo U Ruiz-Esparza, Matthew G Landry, Funda Meric-Bernstam, and Mauro Ferrari, *Molecular-targeted nanotherapies in cancer: Enabling treatment specificity*, Molecular oncology **5** (2011), no. 6, 492–503.

[36] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz, *Swarm intelligence: from natural to artificial systems*, Oxford university press, 1999.

[37] William H Bossert and Edward O Wilson, *The analysis of olfactory communication among animals*, Journal of theoretical biology **5** (1963), no. 3, 443–469.

[38] Houria Boulaiz, Pablo J Alvarez, Alberto Ramirez, Juan A Marchal, Jose Prados, Fernando Rodríguez-Serrano, Macarena Perán, Consolación Melguizo, and Antonia Aranega, *Nanomedicine: application areas and development prospects*, International journal of molecular sciences **12** (2011), no. 5, 3303–3321.

[39] Daniel Bryce and Jichao Sun, *dreach and dreal web page*, `http://dreal.github.io/`, Accessed:20-09-2017.

[40] Stephen F Bush, *Nanoscale communication networks*, Artech House, 2010.

[41] Allan Chang, *Statistics toolkit : Poisson distribution:explained*, `https://www.statstodo.com/Poisson_Exp.php`, Accessed:01-09-2017.

[42] Ling Cheung, *Randomized wait-free consensus using an atomicity assumption*, International Conference On Principles Of Distributed Systems, Springer, 2005, pp. 47–60.

[43] David E Clapham, *Calcium signaling*, Cell **80** (1995), no. 2, 259–268.

[44] Frits GW Dannenberg, *Modelling and verification for dna nanotechnology*, Ph.D. thesis, University of Oxford, 2015.

[45] Deniz Demiray, Albert Cabellos-Aparicio, Eduard Alarcón, D Turgay Altilar, Ignacio Llatser, Luca Felicetti, Gianluca Reali, and Mauro Femminella, *Direct: A model for molecular communication nanonetworks based on discrete entities*, Nano Communication Networks **4** (2013), no. 4, 181–188.

[46] Deniz Demiray, Albert Cabellos-Aparicio, D Turgay Altilar, Ignacio Llatser, and Eduard Alarcón, *Distance effect of molecular harvesting over signal reception and harvesting performance in direct*, submitted for publication to Nano Communication Networks (2014), 1–4.

[47] Patrik D'haeseleer, Stephanie Forrest, and Paul Helman, *An immunological approach to change detection: Algorithms, analysis and implications*, Security and Privacy, 1996. Proceedings., 1996 IEEE Symposium on, IEEE, 1996, pp. 110–119.

[48] Paolo Di Lorenzo, Sergio Barbarossa, and Ali H Sayed, *Bio-inspired decentralized radio access based on swarming mechanisms over adaptive networks*, IEEE Transactions on Signal Processing **61** (2013), no. 12, 3183–3197.

[49] Raymond Nicholas Doetsch and Thomas Melbourne Cook, *Introduction to bacteria and their ecobiology*, Springer Science & Business Media, 2012.

[50] Marco Dorigo, Vittorio Maniezzo, and Alberto Colorni, *Ant system: optimization by a colony of cooperating agents*, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) **26** (1996), no. 1, 29–41.

[51] Falko Dressler and Ozgur B Akan, *Bio-inspired networking: from theory to practice*, IEEE Communications Magazine **48** (2010), no. 11, 2–10.

[52] ———, *A survey on bio-inspired networking*, Computer Networks **54** (2010), no. 6, 881–900.

[53] Falko Dressler, Isabel Dietrich, Reinhard German, and Bettina Krüger, *A rule-based system for programming self-organized sensor and actor networks*, Computer Networks **53** (2009), no. 10, 1737–1750.

[54] Falko Dressler and Frank Kargl, *Security in nano communication: Challenges and open research issues*, Communications (ICC), 2012 IEEE International Conference on, IEEE, 2012, pp. 6183–6187.

[55] ———, *Towards security in nano-communication: Challenges and opportunities*, Nano communication networks **3** (2012), no. 3, 151–160.

[56] Andrew W Eckford, *Achievable information rates for molecular communication with distinct molecules*, Bio-Inspired Models of Network, Information and Computing Systems, 2007. Bionetics 2007. 2nd, IEEE, 2007, pp. 313–315.

[57] ———, *Nanoscale communication with brownian motion*, Information Sciences and Systems, 2007. CISS'07. 41st Annual Conference on, IEEE, 2007, pp. 160–165.

[58] ———, *Timing information rates for active transport molecular communication*, International Conference on Nano-Networks, Springer, 2009, pp. 24–28.

[59] Arash Einolghozati, Mohsen Sardari, Ahmad Beirami, and Faramarz Fekri, *Consensus problem under diffusion-based molecular communication*, Information Sciences and Systems (CISS), 2011 45th Annual Conference on, IEEE, 2011, pp. 1–6.

[60] ———, *Data gathering in networks of bacteria colonies: Collective sensing and relaying using molecular communication*, Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on, IEEE, 2012, pp. 256–261.

[61] Arash Einolghozati, Mohsen Sardari, and Faramarz Fekri, *Capacity of diffusion-based molecular communication with ligand receptors*, Information Theory Workshop (ITW), 2011 IEEE, IEEE, 2011, pp. 85–89.

[62] ———, *Networks of bacteria colonies: A new framework for reliable molecular communication networking*, Nano Communication Networks **7** (2016), 17–26.

[63] Akihiro Enomoto, Michael J Moore, Tatsuya Suda, and Kazuhiro Oiwa, *Design of self-organizing microtubule networks for molecular communication*, Nano Communication Networks **2** (2011), no. 1, 16–24.

[64] Deborah Estrin, Ramesh Govindan, John Heidemann, and Satish Kumar, *Next century challenges: Scalable coordination in sensor networks*, Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking, ACM, 1999, pp. 263–270.

[65] Muddassar Farooq, *Bee-inspired protocol engineering: from nature to networks*, Springer Science & Business Media, 2008.

[66] Nariman Farsad, H Birkan Yilmaz, Andrew Eckford, Chan-Byoung Chae, and Weisi Guo, *A comprehensive survey of recent advancements in molecular communication*, IEEE Communications Surveys & Tutorials **18** (2016), no. 3, 1887–1919.

[67] Michael J Federle and Bonnie L Bassler, *Interspecies communication in bacteria*, Journal of clinical investigation **112** (2003), no. 9, 1291.

[68] Richard P Feynman, *There's plenty of room at the bottom*, Engineering and science **23** (1960), no. 5, 22–36.

[69] Nora Garralda, Ignacio Llatser, Albert Cabellos-Aparicio, and Massimiliano Pierobon, *Simulation-based evaluation of the diffusion-based physical channel in molecular nanonetworks*, Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on, IEEE, 2011, pp. 443–448.

[70] Seth Gilbert and Calvin Newport, *The computational power of beeps*, International Symposium on Distributed Computing, Springer, 2015, pp. 31–46.

[71] Lluís Parcerisa Giné and Ian F Akyildiz, *Molecular communication options for long range nanonetworks*, Computer Networks **53** (2009), no. 16, 2753–2766.

[72] Navid Goodarzi, Mohammad Hossein Ghahremani, and Rassoul Dinarvand, *Targeting cd44 by hyaluronic acid-based nano drug delivery systems may eradicate cancer stem cells in human breast cancer*, Journal of Medical Hypotheses and Ideas **5** (2011), 26.

[73] Christian Gourieroux, Alain Monfort, and Alain Trognon, *Pseudo maximum likelihood methods: Applications to poisson models*, Econometrica: Journal of the Econometric Society (1984), 701–720.

[74] Maria Gregori and Ian F Akyildiz, *A new nanonetwork architecture using flagellated bacteria and catalytic nanomotors*, IEEE Journal on selected areas in communications **28** (2010), no. 4, 612–619.

[75] Maria Gregori, Ignacio Llatser, Albert Cabellos-Aparicio, and Eduard Alarcón, *Physical channel characterization for medium-range nanonetworks using catalytic nanomotors*, Nano Communication Networks **1** (2010), no. 2, 102–107.

[76] ———, *Physical channel characterization for medium-range nanonetworks using flagellated bacteria*, Computer Networks **55** (2011), no. 3, 779–791.

[77] Brian K Hammer and Bonnie L Bassler, *Quorum sensing controls biofilm formation in vibrio cholerae*, Molecular microbiology **50** (2003), no. 1, 101–104.

[78] Akif Cem Heren, M Sukru Kuran, H Birkan Yilmaz, and Tuna Tugcu, *Channel capacity of calcium signalling based on inter-cellular calcium waves in astrocytes*, Communications Workshops (ICC), 2013 IEEE International Conference on, IEEE, 2013, pp. 792–797.

[79] Yuichi Hiratsuka, Tetsuya Tada, Kazuhiro Oiwa, Toshihiko Kanayama, and Taro QP Uyeda, *Controlling the direction of kinesin-driven microtubule movements along microlithographic tracks*, Biophysical Journal **81** (2001), no. 3, 1555–1561.

[80] Satoshi Hiyama, Yasushi Isogawa, Tatsuya Suda, Yuki Moritani, and Kazuo Sutoh, *A design of an autonomous molecule loading/transporting/unloading system using dna hybridization and biomolecular linear motors*, arXiv preprint arXiv:0708.1839 (2007), 1–6.

[81] Steven A Hofmeyr and Stephanie Forrest, *Architecture for an artificial immune system*, Evolutionary computation **8** (2000), no. 4, 443–473.

[82] Bojun Huang and Thomas Moscibroda, *Conflict resolution and membership problem in beeping channels*, International Symposium on Distributed Computing, Springer, 2013, pp. 314–328.

[83] Jiun-Ting Huang, Hsin-Yu Lai, Yen-Chi Lee, Chia-Han Lee, and Ping-Cheng Yeh, *Distance estimation in concentration-based molecular communications*, 2013 IEEE Global Communications Conference (GLOBECOM), IEEE, 2013, pp. 2587–2591.

[84] Vahid Jamali, Arman Ahmadzadeh, Christophe Jardin, Heinrich Sticht, and Robert Schober, *Channel estimation techniques for diffusion-based molecular communications*, Communications (ICC), 2016 IEEE International Conference on, IEEE, 2016, pp. 1–7.

[85] Antal A Járai, *Sandpile models*, arXiv preprint arXiv:1401.0354 (2014), 1–66.

[86] Josep Miquel Jornet and Ian F Akyildiz, *Channel capacity of electromagnetic nanonetworks in the terahertz band*, Communications (ICC), 2010 IEEE International Conference on, IEEE, 2010, pp. 1–6.

[87] ———, *Graphene-based nano-antennas for electromagnetic nanocommunications in the terahertz band*, Antennas and Propagation (EuCAP), 2010 Proceedings of the Fourth European Conference on, IEEE, 2010, pp. 1–5.

[88] ———, *The internet of multimedia nano-things*, Nano Communication Networks **3** (2012), no. 4, 242–251.

[89] Tomasz Jurdzinski and Dariusz Kowalski, *Distributed backbone structure for algorithms in the sinr model of wireless networks*, Distributed Computing (2012), 106–120.

[90] Tomasz Jurdzinski, Dariusz R Kowalski, Michal Rozanski, and Grzegorz Stachowiak, *Distributed randomized broadcasting in wireless networks under the sinr model*, International Symposium on Distributed Computing, Springer, 2013, pp. 373–387.

[91] Sachin Kadloor and Raviraj Adve, *A framework to study the molecular communication system*, Computer Communications and Networks, 2009. ICCCN 2009. Proceedings of 18th Internatonal Conference on, IEEE, 2009, pp. 1–6.

[92] Sachin Kadloor, Raviraj S Adve, and Andrew W Eckford, *Molecular communication using brownian motion with drift*, IEEE Transactions on NanoBioscience **11** (2012), no. 2, 89–99.

[93] Ioannis Karatzas and Steven Shreve, *Brownian motion and stochastic calculus*, vol. 113, Springer Science & Business Media, 2012.

[94] Bettina Kemme, *Traditional concurrency control for replicated databases*, Encyclopedia of Database Systems, Springer, 2009, pp. 3144–3149.

[95] Jeffrey O Kephart et al., *A biologically inspired immune system for computers*, Artificial Life IV: proceedings of the fourth international workshop on the synthesis and simulation of living systems, 1994, pp. 130–139.

[96] Sergey Knyazev, Sergey Tarakanov, Vladimir Kuznetsov, Yu Porozov, Yevgeni Koucheryavy, and E Stepanov, *Coarse-grained model of protein interaction for bio-inspired nano-communication*, Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2014 6th International Congress on, IEEE, 2014, pp. 260–262.

[97] Bettina Krüger and Falko Dressler, *Molecular processes as a basis for autonomous networking*, IPSI Transactions on Advances Research: Issues in Computer Science and Engineering **1** (2005), no. 1, 43–50.

[98] Mehmet S Kuran, Huseyin Birkan Yilmaz, Tuna Tugcu, and Ian F Akyildiz, *Modulation techniques for communication via diffusion in nanonetworks*, Communications (ICC), 2011 IEEE International Conference on, IEEE, 2011, pp. 1–5.

[99] Mehmet Şükrü Kuran, H Birkan Yilmaz, Tuna Tugcu, and Ian F Akyildiz, *Interference effects on modulation techniques in diffusion based nanonetworks*, Nano Communication Networks **3** (2012), no. 1, 65–73.

[100] Mehmet Şükrü Kuran, H Birkan Yilmaz, Tuna Tugcu, and Bilge Özerman, *Energy model for communication via diffusion in nanonetworks*, Nano Communication Networks **1** (2010), no. 2, 86–95.

[101] Marta KWIATKOWSKA, *Probabilistic model checking*, Modeling and verification of parallel processes. Summer school, 2001, pp. 189–204.

[102] Marta Kwiatkowska, Gethin Norman, and David Parker, *Prism: Probabilistic symbolic model checker*, International Conference on Modelling Techniques and Tools for Computer Performance Evaluation, Springer, 2002, pp. 200–204.

[103] ______, *Advances and challenges of probabilistic model checking*, Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on, IEEE, 2010, pp. 1691–1698.

[104] Marta Kwiatkowska, Gethin Norman, and Roberto Segala, *Automated verification of a randomized distributed consensus protocol using cadence smv and prism?*, International Conference on Computer Aided Verification, Springer, 2001, pp. 194–206.

[105] Marta Z Kwiatkowska and Chris Thachuk, *Probabilistic model checking for biology.*, Software Systems Safety **36** (2014), 165.

[106] Jian-Qin Liu, *Molecular informatics of nano-communication based on cells: A brief survey*, Nano Communication Networks **1** (2010), no. 2, 118–125.

[107] Qiang Liu and Kun Yang, *Channel capacity analysis of a diffusion-based molecular communication system with ligand receptors*, International Journal of Communication Systems **28** (2015), no. 8, 1508–1520.

[108] Ignacio Llatser, Eduard Alarcón, and Massimiliano Pierobony, *Diffusion-based channel characterization in molecular nanonetworks*, Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on, IEEE, 2011, pp. 467–472.

[109] Ignacio Llatser, Albert Cabellos-Aparicio, and Eduard Alarcon, *Networking challenges and principles in diffusion-based molecular communication*, IEEE Wireless Communications **19** (2012), no. 5, 36–41.

[110] Ignacio Llatser, Albert Cabellos-Aparicio, Eduard Alarcón, Josep Miquel Jornet, Albert Mestres, Heekwan Lee, and Josep Solé-Pareta, *Scalability of the channel capacity in graphene-enabled wireless communications to the nanoscale*, IEEE Transactions on Communications **63** (2015), no. 1, 324–333.

[111] Ignacio Llatser, Albert Cabellos-Aparicio, Massimiliano Pierobon, and Eduard Alarcón, *Detection techniques for diffusion-based molecular communication*, IEEE Journal on Selected Areas in Communications **31** (2013), no. 12, 726–734.

[112] Ignacio Llatser, Deniz Demiray, Albert Cabellos-Aparicio, D Turgay Altilar, and Eduard Alarcón, *N3sim: Simulation framework for diffusion-based molecular communication nanonetworks*, Simulation Modelling Practice and Theory **42** (2014), 210–222.

[113] Ignacio Llatser, Christian Kremers, Albert Cabellos-Aparicio, Josep Miquel Jornet, Eduard Alarcón, and Dmitry N Chigrin, *Graphene-based nano-patch antenna for terahertz radiation*, Photonics and Nanostructures-Fundamentals and Applications **10** (2012), no. 4, 353–358.

[114] Harvey Lodish, Arnold Berk, S Lawrence Zipursky, Paul Matsudaira, David Baltimore, James Darnell, et al., *Molecular cell biology*, vol. 3, Scientific American Books New York, 1995.

[115] Valeria Loscri, César Marchal, Nathalie Mitton, Giancarlo Fortino, and Athanasios V Vasilakos, *Security and privacy in molecular communication and networking: Opportunities and challenges*, IEEE transactions on nanobioscience **13** (2014), no. 3, 198–207.

[116] MH Mahfuz, Dimitrios Makrakis, and Hussein T Mouftah, *Concentration encoded molecular communication: Prospects and challenges towards nanoscale networks*, Proceedings of 2013 International Conference on Engineering Research, Innovation and Education (ICERIE 2013), Sylhet, Bangladesh, vol. 1113, 2013, p. 508513.

[117] Mohammad Upal Mahfuz, Dimitrios Makrakis, and Hussein T Mouftah, *Characterization of molecular communication channel for nanoscale networks.*, BIOSIGNALS, 2010, pp. 327–332.

[118] ———, *On the characterization of binary concentration-encoded molecular communication in nanonetworks*, Nano Communication Networks **1** (2010), no. 4, 289–300.

[119] ———, *Characterization of intersymbol interference in concentration-encoded unicast molecular communication*, Electrical and Computer Engineering (CCECE), 2011 24th Canadian Conference on, IEEE, 2011, pp. 000164–000168.

[120] ———, *On the detection of binary concentration-encoded unicast molecular communication in nanonetworks.*, BIOSIGNALS, 2011, pp. 446–449.

[121] ———, *Concentration-encoded molecular communication in nanonetworks. part 1: Fundamentals, issues, and challenges*, Modeling, Methodologies and Tools for Molecular and Nano-scale Communications, Springer, 2017, pp. 3–34.

[122] Ruchika Malhotra, Vyomesh Patel, Jose Pedro Vaqué, J Silvio Gutkind, and James F Rusling, *Ultrasensitive electrochemical immunosensor for oral cancer biomarker il-6 using carbon nanotube forest electrodes and multilabel amplification*, Analytical chemistry **82** (2010), no. 8, 3118–3123.

[123] Ignacio Llatser Martı, *Exploring the scalability limits of communication networks at the nanoscale*, Master's thesis, Universitat Politècnica de Catalunya, 2011.

[124] Ignacio Llatser Martí, Christian Kremers, Albert Cabellos-Aparicio, Josep Miquel Jornet, Eduard Alarcón, Dmitry N Chigrin, and Dmitry N Chigrin, *Scattering of terahertz radiation on a graphene-based nano-antenna*, AIP Conference Proceedings, vol. 1398, AIP, 2011, pp. 144–146.

[125] Ling-San Meng, Ping-Cheng Yeh, Kwang-Cheng Chen, and Ian F Akyildiz, *Mimo communications based on molecular diffusion*, Global Communications Conference (GLOBECOM), 2012 IEEE, IEEE, 2012, pp. 5380–5385.

[126] George B Mertzios, Sotiris E Nikoletseas, Christoforos Raptopoulos, and Paul G Spirakis, *Natural models for evolution on networks.*, WINE, Springer, 2011, pp. 290–301.

[127] Renato E Mirollo and Steven H Strogatz, *Synchronization of pulse-coupled biological oscillators*, SIAM Journal on Applied Mathematics **50** (1990), no. 6, 1645–1662.

[128] Michael Mitzenmacher and Eli Upfal, *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*, Cambridge university press, 2017.

[129] Michael Moore, Akihiro Enomoto, Tadashi Nakano, Ryota Egashira, Tatsuya Suda, Atsushi Kayasuga, Hiroaki Kojima, Hitoshi Sakakibara, and Kazuhiro Oiwa, *A design of a molecular communication system for nanomachines using molecular motors*, Pervasive Computing and Communications Workshops, 2006. PerCom Workshops 2006. Fourth Annual IEEE International Conference on, IEEE, 2006, pp. 6–pp.

[130] Michael J Moore, Tadashi Nakano, Akihiro Enomoto, and Tatsuya Suda, *Measuring distance from single spike feedback signals in molecular communication*, IEEE Transactions on Signal Processing **60** (2012), no. 7, 3576–3587.

[131] Michael John Moore, Tatsuya Suda, and Kazuhiro Oiwa, *Molecular communication: modeling noise effects on information rate*, IEEE transactions on nanobioscience **8** (2009), no. 2, 169–180.

[132] Y Moritani, S Hiyama, and T Suda, *Molecular communication among nanomachines using vesicles*, Proceedings of NSTI nanotechnology conference, vol. 2, 2006, pp. 705–708.

[133] Siddhartha S Mukhopadhyay, *Nanotechnology in agriculture: prospects and constraints*, Nanotechnology, science and applications **7** (2014), 63.

[134] Yonathan Murin, Nariman Farsad, Mainak Chowdhury, and Andrea J Goldsmith, *On time-slotted communication over molecular timing channels.*, NANOCOM, 2016, pp. 9–1.

[135] Tadashi Nakano, Andrew W Eckford, and Tokuko Haraguchi, *Molecular communication*, Cambridge University Press, 2013.

[136] Tadashi Nakano, Yu-Hsiang Hsu, William C Tang, Tatsuya Suda, Diane Lin, Takako Koujin, Tokuko Haraguchi, and Yasushi Hiraoka, *Microplatform for intercellular communication*, Nano/Micro Engineered and Molecular Systems, 2008. NEMS 2008. 3rd IEEE International Conference on, IEEE, 2008, pp. 476–479.

[137] Tadashi Nakano, Michael Moore, Akihiro Enomoto, and Tatsuya Suda, *Molecular communication technology as a biological ict*, Biological functions for information and communication technologies (2011), 49–86.

[138] Tadashi Nakano, Michael J Moore, Fang Wei, Athanasios V Vasilakos, and Jianwei Shuai, *Molecular communication and networking: Opportunities and challenges*, IEEE transactions on nanobioscience **11** (2012), no. 2, 135–148.

[139] Tadashi Nakano, Yutaka Okaie, and Jian-Qin Liu, *Channel model and capacity analysis of molecular communication with brownian motion*, IEEE communications letters **16** (2012), no. 6, 797–800.

[140] Tadashi Nakano, Tatsuya Suda, Takako Koujin, Tokuko Haraguchi, and Yasushi Hiraoka, *Molecular communication through gap junction channels: System design, experiments and modeling*, Bio-Inspired Models of Network, Information and Computing Systems, 2007. Bionetics 2007. 2nd, IEEE, 2007, pp. 139–146.

[141] Tadashi Nakano, Tatsuya Suda, Michael Moore, Ryota Egashira, Akihiro Enomoto, and Kayo Arima, *Molecular communication for nanomachines using intercellular calcium signaling*, Nanotechnology, 2005. 5th IEEE Conference on, IEEE, 2005, pp. 478–481.

[142] Wai-Leung Ng and Bonnie L Bassler, *Bacterial quorum-sensing network architectures*, Annual review of genetics **43** (2009), 197–222.

[143] Y Nishi, *Scaling limits of silicon cmos and non-silicon opportunities*, Future Trends in Microelectronics, S. Luryi, J. Xu, and A. Zaslavsky, Eds.: John Wiley & Sons, Inc (2007), Chapter 18.

[144] Adam Noel, Karen C Cheung, and Robert Schober, *Bounds on distance estimation via diffusive molecular communication*, Global Communications Conference (GLOBECOM), 2014 IEEE, IEEE, 2014, pp. 2813–2819.

[145] ———, *Improving receiver performance of diffusive molecular communication with enzymes*, IEEE Transactions on NanoBioscience **13** (2014), no. 1, 31–43.

[146] Guglielmo Paoletti, *The abelian sandpile model*, Deterministic Abelian Sandpile Models and Patterns, Springer, 2014, pp. 9–35.

[147] Kai Papenfort and Bonnie L Bassler, *Quorum sensing signal-response systems in gram-negative bacteria*, Nature Reviews Microbiology **14** (2016), no. 9, 576–588.

[148] David Anthony Parker, *Implementation of symbolic model checking for probabilistic systems*, Ph.D. thesis, University of Birmingham, 2003.

[149] ______, *Probabilistic model checking*, University Lecture, 2011.

[150] R. Patel, *Nanorobotics ideas in nanomedicine*, Asian Journal of Pharmaceutical Sciences and Research **3** (2013), no. 3, 15–22.

[151] Tony Pawson, *Protein modules and signalling networks*, Nature **373** (1995), no. 6515, 573.

[152] Jean Philibert, *One and a half century of diffusion: Fick, einstein, before and beyond*, Diffusion Fundamentals **4** (2006), no. 6, 1–19.

[153] Massimiliano Pierobon and Ian F Akyildiz, *A physical end-to-end model for molecular communication in nanonetworks*, IEEE Journal on Selected Areas in Communications **28** (2010), no. 4, 602–610.

[154] ______, *Diffusion-based noise analysis for molecular communication in nanonetworks*, IEEE Transactions on Signal Processing **59** (2011), no. 6, 2532–2547.

[155] ______, *Information capacity of diffusion-based molecular communication in nanonetworks*, INFOCOM, 2011 Proceedings IEEE, IEEE, 2011, pp. 506–510.

[156] ______, *Noise analysis in ligand-binding reception for molecular communication in nanonetworks*, IEEE Transactions on Signal Processing **59** (2011), no. 9, 4168–4182.

[157] ______, *Capacity of a diffusion-based molecular communication system with channel memory and molecular noise*, IEEE Transactions on Information Theory **59** (2013), no. 2, 942–954.

[158] Yoav Raz, *The dynamic two phase commitment (d2pc) protocol*, International Conference on Database Theory, Springer, 1995, pp. 162–176.

[159] Carl A Richmond, *Fireflies flashing in unison*, Science **71** (1930), no. 1847, 537–538.

[160] Negar Rikhtegar and Manijeh Keshtgary, *A brief survey on molecular and electromagnetic communications in nano-networks*, International Journal of Computer Applications **79** (2013), no. 3, 16–28.

[161] Neus Roca Lacasa, *Modeling the molecular communication nanonetworks*, Master's thesis, Universitat Politècnica de Catalunya, 2009.

[162] Stephan Schauder, Kevan Shokat, Michael G Surette, and Bonnie L Bassler, *The luxs family of bacterial autoinducers: biosynthesis of a novel quorum-sensing signal molecule*, Molecular microbiology **41** (2001), no. 2, 463–476.

[163] Hoda ShahMohammadian, *System design for nano-network communications*, Ph.D. thesis, University of Calgary, 2013.

[164] Hoda ShahMohammadian, Geoffrey G Messier, and Sebastian Magierowski, *Optimum receiver for molecule shift keying modulation in diffusion-based molecular communication channels*, Nano Communication Networks **3** (2012), no. 3, 183–195.

[165] Tatsuya Suda, Michael Moore, Tadashi Nakano, Ryota Egashira, Akihiro Enomoto, Satoshi Hiyama, and Yuki Moritani, *Exploratory research on molecular communication between nanomachines*, Genetic and Evolutionary Computation Conference (GECCO), Late Breaking Papers, vol. 25, 2005, pp. 29–24.

[166] Michael G Surette, Melissa B Miller, and Bonnie L Bassler, *Quorum sensing in escherichia coli, salmonella typhimurium, and vibrio harveyi: a new family of genes responsible for autoinducer production*, Proceedings of the National Academy of Sciences **96** (1999), no. 4, 1639–1644.

[167] Junichi Suzuki, Tadashi Nakano, and Michael John Moore, *Modeling, methodologies and tools for molecular and nano-scale communications: Modeling, methodologies and tools*, 2017.

[168] Christof Teuscher, Cristian Grecu, Ting Lu, and Ron Weiss, *Challenges and promises of nano and bio communication networks*, Proceedings of the Fifth ACM/IEEE International Symposium on Networks-on-Chip, ACM, 2011, pp. 247–254.

[169] Vasos Vassiliou, *Security issues in nanoscale communication networks*, 3rd NaNoNetworking Summit, Network Research Laboratory, 2011, pp. 1–53.

[170] Frank Walsh and Sasitharan Balasubramaniam, *Reliability and delay analysis of multihop virus-based nanonetworks*, IEEE Transactions on Nanotechnology **12** (2013), no. 5, 674–684.

[171] Frank Walsh, Sasitharan Balasubramaniam, Dmitri Botvich, Tatsuya Suda, Tadashi Nakano, Stephen F Bush, and Mícheál Ó Foghlú, *Hybrid dna and enzyme based computing for address encoding, link switching and error correction in molecular communication*, International Conference on Nano-Networks, Springer, 2008, pp. 28–38.

[172] Jianxin Wang, Yongzhuo Huang, Allan E David, Beata Chertok, Lei Zhang, Faquan Yu, and Victor C Yang, *Magnetic nanoparticles for mri of brain tumors*, Current pharmaceutical biotechnology **13** (2012), no. 12, 2403–2416.

[173] Michael Wang and Tatsuya Suda, *The bio-networking architecture: A biologically inspired approach to the design of scalable, adaptive, and survivable/available network applications*, Applications and the Internet, 2001. Proceedings. 2001 Symposium on, IEEE, 2001, pp. 43–53.

[174] Christopher M Waters and Bonnie L Bassler, *Quorum sensing: cell-to-cell communication in bacteria*, Annu. Rev. Cell Dev. Biol. **21** (2005), 319–346.

[175] Gezhi Weng, Upinder S Bhalla, and Ravi Iyengar, *Complexity in biological signaling systems*, Science **284** (1999), no. 5411, 92–96.

[176] Andrew Whitmore, Anurag Agarwal, and Li Da Xu, *The internet of things–a survey of topics and trends*, Information Systems Frontiers **17** (2015), no. 2, 261.

[177] Jiri Wiedermann and Lukas Petru, *Communicating mobile nano-machines and their computational power*, International Conference on Nano-Networks, Springer, 2008, pp. 123–130.

[178] H Birkan Yilmaz and Chan-Byoung Chae, *Simulation study of molecular communication systems with an absorbing receiver: Modulation and isi mitigation techniques*, Simulation Modelling Practice and Theory **49** (2014), 136–150.

[179] H Birkan Yilmaz, Akif Cem Heren, Tuna Tugcu, and Chan-Byoung Chae, *Three-dimensional channel characteristics for molecular communications with an absorbing receiver*, IEEE Communications Letters **18** (2014), no. 6, 929–932.