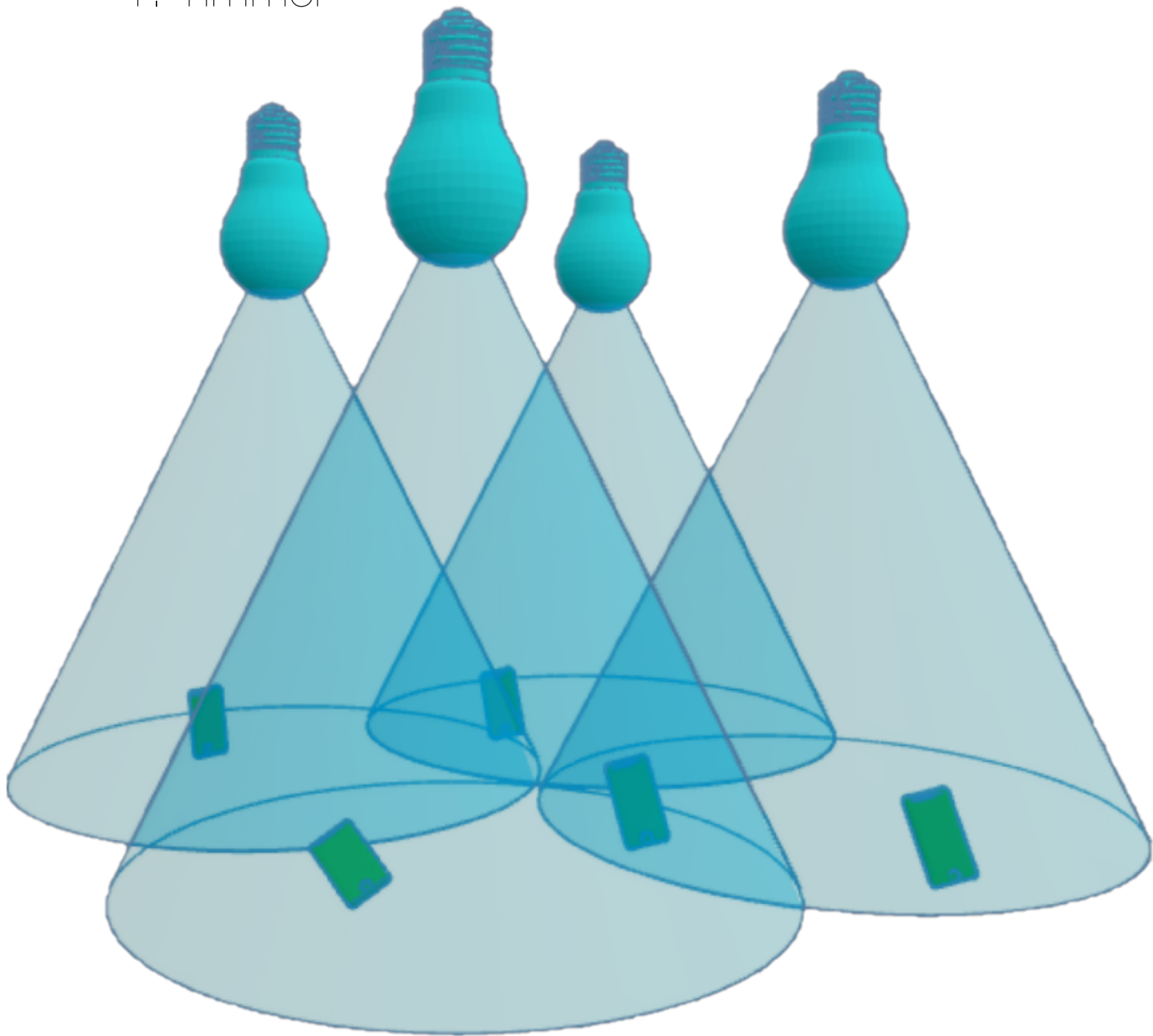


# Validating a LiFi communication system

An OMNEST emulation approach

T. Timmer



This page is intentionally left blank.

# Validating a LiFi communication system

An OMNEST emulation approach

by

T. Timmer

to obtain the degree of

**Master of Science**

in the field of Embedded Systems,  
specialization Software and Networking,

at the Delft University of Technology,  
to be defended on Tuesday 13 July 2021 at 14:00.

Student number: 4453158  
Project duration: November 2, 2020 – June 30, 2021  
Thesis committee: Dr. M. Zúñiga TU Delft, Chair  
Dr. C. Dandelski Signify, Supervisor  
Dr. F. Fioranelli TU Delft, additional member

An electronic version of this thesis is available at  
<http://repository.tudelft.nl/>.

This page is intentionally left blank.

# Abstract

Most wireless devices use the radio frequency spectrum which is reaching its limits, as almost all frequencies are in use. Devices which communicate on the same radio frequencies are interfering with each other and this creates problems in the communication. Currently, there are around 10 billion of these devices. This number is rapidly growing and this increases the issue of interference. One of the possible solutions for future wireless communication is a communication technology called **Light Fidelity (LiFi)**. This technology uses light waves to communicate rather than radio waves. Since light operates outside the common radio frequencies, **LiFi** does not introduce additional interference to today's commonly used radio communication technologies.

Signify is the first company with a commercially available **LiFi** system and has branded their system Trulifi. The Trulifi network can consist of multiple **Access Point (AP)**s and **End Point (EP)**s. These devices have a cone-like coverage area in which devices can communicate with each other. Outside this coverage area, communication is not possible. As with other communication technologies, devices within a **LiFi** system are also affected by interference. In the overlapping coverage areas between two APs two types of interference can occur. Downlink interference occurs whenever an **EP** receives messages from multiple **AP**s at the same time, while uplink interference occurs whenever an **AP** is able to receive messages from an **EP** which is connected to a neighbouring **AP**. To overcome this issue Signify uses a so called **LiFi controller (LC)**. The main task of the **LC** is to create an interference free communication schedule for all **AP**s in the **LiFi** network. This schedule is based on the interference reports received from the **AP**s and **EP**s.

Validating large scale **LiFi** systems can become complicated, in terms of high costs involved and in the very large spaces needed to install such a large scale system. To ensure the **LC** operates as specified, a stress-test simulation model was designed by Signify. The stress-test simulation's sole purpose is to ensure the **LC** keeps operating as specified in the worst case situation. E.g. all the possible **AP**s, all the possible **EP**s and all the interference that can occur. This test is, however, not realistic as these worst case situations are not common in real-world deployments of the system.

In this thesis, the problem of creating a realistic Trulifi simulation model is addressed. A literature study has been performed to investigate if existing models can be used. The conclusion of this literature study shows that a radio model needed to be designed which addressed the needs of the Trulifi system. This model has been designed with a high level of abstraction, while keeping the simulation as realistic as possible. To validate the new radio model, the Trulifi simulation has been compared to a real-world Trulifi system. Once this radio model was validated, the scheduling was compared based on different configuration sizes, as the main goal of the thesis is to create a realistic simulation model with which the Trulifi system can be validated. For this validation, the Trulifi simulation was compared to both the real-world Trulifi system and the stress-test simulation.

This page is intentionally left blank.

# Preface

Before you lies the thesis “Validating a LiFi communication system. An OMNEST emulation approach”, the basis of which is Signify’s Trulifi system, a system which brings wireless communication through the use of light. It has been written to obtain the degree of Master of Science, Embedded Systems at the Delft University of Technology. I was engaged in researching and writing this thesis from November 2020 to June 2021.

The project is performed to fulfil the request of Signify, where I undertook an internship. It has been their need for a simulation to test their Trulifi System at a larger scale that created this opportunity. Due to the global pandemic, the work was performed mostly at home. The remote working did not influence the quality of the research performed due to the daily stand-up sessions with my supervisor, Conrad Dandelski.

I would like to thank my supervisor for the excellent guidance and support throughout the duration of the research. I also wish to thank Signify for providing me with the opportunity to perform my research at their company. Furthermore, I would like to thank the university supervisor, Marco Zúñiga, for the monthly meetings and remarks which improved my research. I also benefited from debating issues with the rest of Signify’s LiFi software team, my friends, and especially my family. If I ever got stuck while designing or implementing the simulation, they were the ones who got me back on track. My family also deserves a special note of thanks: your support is what made me want to perform as best as possible and what has brought me to where I am now. Finally, I would like to thank my girlfriend, Iris Deurloo. She helped me improve my writing and motivated me to try to perform my research for a company in the first place.

I hope you enjoy reading the thesis as much as I enjoyed performing the research.

*Thijs Timmer*  
*Zoetermeer, June 2021*

# Contents

<b>Abstract</b>	<b>i</b>
<b>Preface</b>	<b>iii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 LiFi	2
1.2 Problem description	3
1.2.1 Prototyping products	3
1.2.2 Stress-test model	4
1.3 Thesis Contribution	5
1.4 Thesis outline	6
<b>2 Background</b>	<b>7</b>
2.1 Electromagnetic spectrum	7
2.2 Radio Frequency communication	8
2.3 Visible light communication	9
2.4 Wireless network simulation	10
2.4.1 Radio Frequency simulation	10
2.4.2 LiFi Simulation models	11
2.5 Mobility models	11
2.5.1 Employee behaviour in terms of movement	11
2.5.2 Random Way point	12
2.5.3 Linear Mobility	12
2.5.4 Human behaviour mobility models	12
2.6 Conclusion	13
<b>3 Methods and Tools</b>	<b>14</b>
3.1 Simulation Environment	14
3.1.1 Simulation engines	15
3.1.2 Simulation Realism	16
3.2 Mathematical Analysis	17
<b>4 Design</b>	<b>18</b>
4.1 Architecture	18
4.1.1 Devices	19
4.1.2 LiFiMedium	19
4.2 Message exchange	21
4.3 State machines	23
4.4 Validation	25
<b>5 Evaluation and validation</b>	<b>26</b>
5.1 Analysis of the LiFi transmitter	26
5.2 Light propagation validation	28
5.3 Trulifi simulation model analysis	32
5.3.1 Real-world Trulifi system implementation	32



---

5.3.2	Stress-test simulation implementation . . . . .	34
5.3.3	Trulifi simulation implementation . . . . .	34
5.4	Trulifi simulation model validation . . . . .	36
5.5	Conclusion . . . . .	45
<b>6</b>	<b>Conclusion and Future work</b>	<b>46</b>
6.1	Conclusion . . . . .	46
6.2	Future work . . . . .	47
	<b>List of Acronyms and Abbreviations</b>	<b>48</b>
	<b>Bibliography</b>	<b>49</b>

This page is intentionally left blank.

# 1

## Introduction

In the history of humankind, there was never an urge as large as now to connect with each other. This desire has been growing more and quicker since the introduction of the internet. At the time of writing this master thesis, more than 10 billion [Internet of Things \(IOT\)](#) devices are connected to the internet [1]. This number is rapidly growing and experts estimate that this number exceeds 25.4 billion in 2030. Most of these devices use wireless communication in the [radio frequency \(RF\)](#) spectrum, which ranges from 30 Hz to 300 GHz [2]. One of the issues with using [RF](#) is that the radio spectrum is already nearing its limits [3, 4], as almost all frequencies are in use. [5] shows a full overview of the current spectrum usage in the Netherlands. It can be seen that only small bands are licence free to communicate on. The spectrum from 400.05 MHz to 10 GHz is most intensely used by smart devices, navigation and medical applications. Another issue with [RF](#) is that the increasing demand and connectivity introduces more interference than ever measured before. Interference between radio frequencies and the limits are not the only issues with using the radio spectrum. Some medical equipment or equipment in planes is susceptible to these radio frequency waves, and might experience erratic behaviour when interacting with these signals, due to the electromagnetic interference that radio waves introduce [6–8].

The limits of the radio spectrum are the main drive behind finding an additional wireless communication method [9]. One of these additional wireless data communication methods is known as [Light Fidelity \(LiFi\)](#). This method builds on the discovery of transmitting data through light waves. This discovery, called the photo phone, was made by Alexander Graham Bell around the 1880s before the telephone was invented [10]. He found a way to transmit speech on a beam of light, by speaking into a cone with a reflective membrane. Before Alexander Graham Bell discovered this, an early version of a Heliograph was invented by Carl Friedrich Gauss in 1821 [11]. This device was capable of signalling by flashing sunlight through a reflective material. In 1989, a company in Australia came up with a device which was able to use fluorescent light to change store price tags [12]. Nine years later, in 1998, a US company came up with the idea of illuminating and communicating simultaneously [13]. Only in 2002, when members of the Nakagawa Laboratory saw the full potential of the idea of illuminating and communicating data simultaneously, the development of [Visual Light Communication \(VLC\)](#) continued [14, 15]. Nine years later, enough research was done in order for IEEE to release a proper standard, IEEE802.15.7-2011, on [VLC](#) [14]. In that same year, Harald Haas introduced the term [LiFi](#) as a subset of [VLC](#) during a talk on TEDGlobal [16].

## 1.1. LiFi

LiFi is a wireless communication method which uses the visible light or infrared spectrum, for transmitting data [17]. The visible light spectrum ranges from 400 THz to 800 THz, while the infrared spectrum ranges from 300 GHz to 400 THz. This wireless communication method shows increases in the data density by a magnitude of three, while avoiding interfering with existing RF networks [18, 19]. LiFi can be deployed on the existing lighting infrastructure as it uses light-emitting diode (LED)s [18].

A LiFi network can consist of multiple LED bulbs, which act as an Access Point (AP), the gateway to connecting to the network. Each AP is also equipped with its own photodetector to receive data. Users can access the network through the use of a dongle, called an End Point (EP). These user devices are equipped with a receiver and transmitter to enable two-way data communication. Figure 1.1 shows a typical LiFi setup, with multiple APs and EPs. LiFi communication requires Line-of-sight (LoS), a straight uninterrupted line from transmitter to receiver. This LoS is needed because signals which travel directly, without reflection, to the receiver carry most power. Photodetectors have a certain Field-of-view (FoV), the angle through which a receiver can pick up light. When designing VLC communication, the LoS and FoV are two important properties for allowing connection. Communication is theoretically possible with speeds up to 225 Gbps, due to the capability of these LED lights to change intensity very quickly [20]. Modulation techniques are needed to transmit data over the light.

While LiFi can be considered the next generation communication technique, which might solve the RF communication problems, there are limitations which need to be taken into account. One of these limitations is the hidden node problem, EPs which are in the coverage area of the same AP cannot see each other due to their FoV. This introduces difficulties in the medium access control (MAC), as the EPs cannot sense signals from other EPs. A sensing algorithm, like carrier sense multiple access (CSMA), before accessing the channel cannot be used. Another limitation involves downlink interference, which occurs in the overlapping areas of AP coverage areas, such an overlap is illustrated in Figure 1.1 with the darker shaded area. When an EP is located within this overlapping area, signals from the APs involved interfere with each other [21–23]. Since EPs use the same kind of transmission as APs, uplink interference can also occur. This happens when an AP receives data from an EP which is connected to another AP while also receiving data from an EP to which it is connected. This downlink and uplink interference can result in data loss and even communication loss if not handled properly.

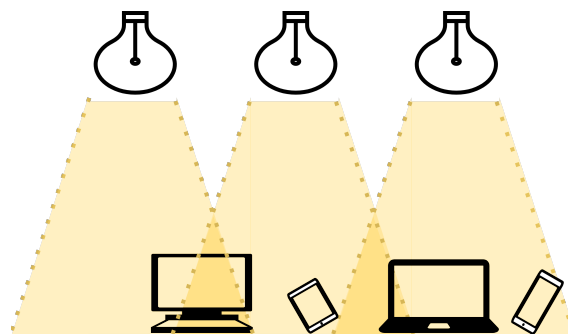


Figure 1.1: A typical LiFi Setup. Note that for simplicity, every light bulb represent an access point and the computers and phones represent end points.

### Signify Trulifi

Signify is the first company with a commercially available LiFi system. Trulifi, as Signify has branded their system, can support up to 1024 users per network at the time of writing this thesis [24, 25]. A typical Trulifi system setup can be seen in Figure 1.2a. Such a system consists of one LiFi controller (LC) per network. This controller can manage up to 64 modems, which in turn can each support a maximum of 6 transceivers. A modem in combination with 1 to 6 of these transceivers is called an AP. These modems are in charge of modulating the data over the light and of processing the incoming light to data. Trulifi allows for 16 EPs per AP. The EP acts as a receiver dongle, with a photodetector to receive the light and a processing chip to translate this light into useful data. To enable two-way communication, the EP is also equipped with its own LED transmitter. There are currently three versions of the AP available (6002.1, 6002.2, 6013). The 6002.1 and 6002.2 can be used to provide seamless LiFi connections over large areas, while the 6013 version is designed as a fixed point-to-point system, acting like a "wireless cable". In this thesis, version 6002.2 is used. In Figure 1.2b, the coverage area of an AP can be seen. From this diagram, one can see the cone-like coverage area together with the minimum and maximum distance at which the connection can be established. From [25], one can see that a Trulifi setup can also be created without the use of a controller. This can however only be done when there is no overlap between different APs.

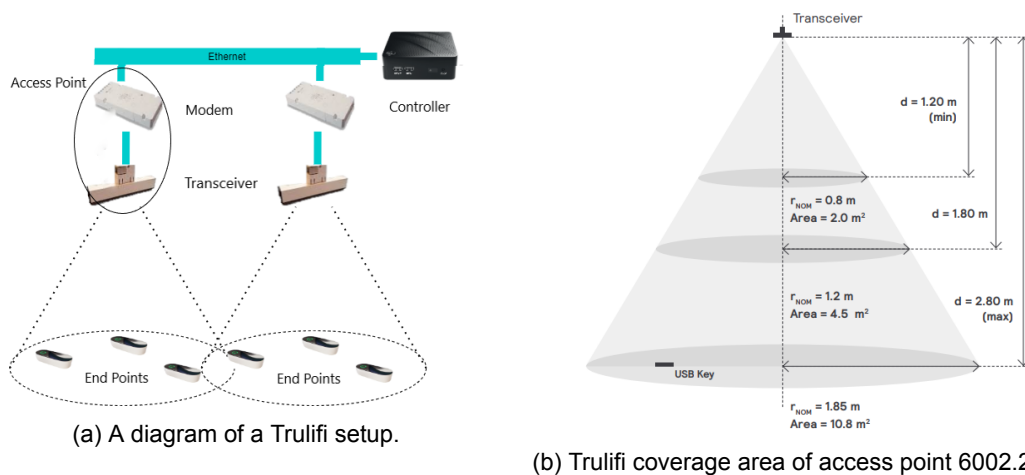


Figure 1.2: Trulifi specifications and setup.

## 1.2. Problem description

This section will provide the problem description in detail. First, prototyping will be explained. Once this has been done, the current simulation model will be explained in detail.

### 1.2.1. Prototyping products

During the prototyping phase of a product, some obstacles might rise. Testing a new product can sometimes require a vast quantity of the same product, which can be very expensive in terms of space, hardware costs and even time. In the case of Trulifi, testing the functionality of the system can quickly include numerous APs and an even larger amount of EPs. A setting with the maximum number of APs and EPs would be expensive, and a huge area is needed in order to deploy such a large scale test. Simulation models are often used to overcome these hindrances, and this is also the case for Trulifi. Signify can also benefit from a simulation from a selling point of view. Signify's customer

focusing team often faces new use cases and new situations, where a simulation could be used to easily validate the wanted system. It is challenging to design a simulation model which is mirroring a real setup. One can quickly introduce either too many or too few variables, making the model overly complicated or too simple. In the coming section, Signify's currently used simulation model is described thoroughly.

### 1.2.2. Stress-test model

The model which is currently used was designed with the purpose of testing the limits of the scheduler deployed on the LC. Due to this particular purpose of the emulator, it is designed in a static way, which ensures that the limits of this controller are reached. The model is implemented through the network simulation tool OMNEST, the commercial version of OMNeT++, and is thus not running on the system itself.

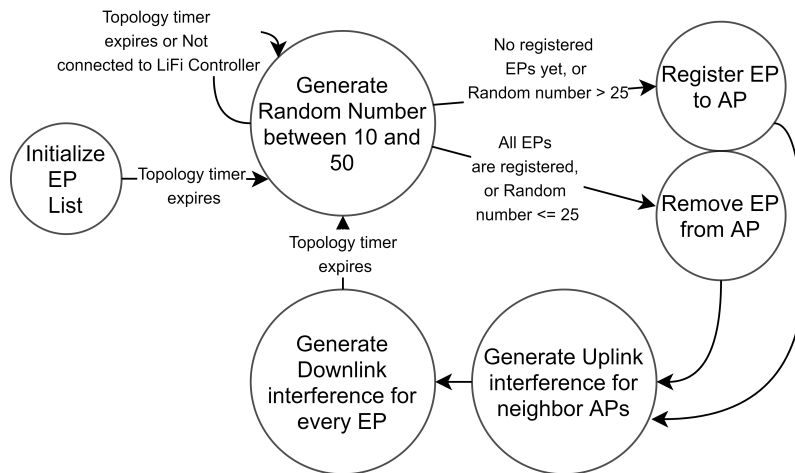


Figure 1.3: Topology functionality state machine of an access point within the stress-test model.

Figure 1.3 shows the topology functionality of an AP within the model as a state machine. The AP has more functionality. Since this does not influence its topology, it is not included in this state machine. One can see that an EP list is initialized. A topology timer is started in the background, which is in charge of starting the loop of changing the topology of the configuration. Once this timer expires, a random number is generated and the timer is reset. This random number will determine if an EP will be connected to or removed from the AP. If there are no connected EPs yet, a connection is made. Whenever all EPs from the list are connected, one is always removed. If there is at least one connection, and not all EPs are connected, connecting occurs with a probability of 62.5%, and removal occurs with 37.5%. As the current model is designed to test the limits of the LC, the experiment could also be started with all EPs connected. This would however overload the system, hence some randomness is introduced to create delay.

After the connection decision is made, the interference handling begins. In the stress-test model, interference is not added based on the positions of the EPs with respect to the coverage areas of the APs, instead the interference is determined by a random generator. How this random interference works is best explained by looking at the two different kinds of interference, uplink and downlink. First, the uplink interference is created. A decision is made on how many EPs (from a neighbouring AP) interfere with the current AP. This is done by performing a modulo operation on a random number, as seen in Equation 1.1

$$\text{interfering EPs} = \text{RandomNumber} \bmod (\text{neighbour EPs} + 1) \quad (1.1)$$

For the downlink interference, the decision is made differently. Instead of looking at the EPs of a neighbouring AP, the number of neighbouring APs is used. Equation 1.2 describes the same modulo operation, but with a different modulus.

$$\text{interfering APs} = \text{RandomNumber} \bmod (\text{neighbour APs} + 1) \quad (1.2)$$

The created interference is sent to the LC as a topology update. The loop then continues whenever the topology timer expires.

### Drawbacks

As described previously, the stress-test model relies on probabilities rather than on coordinates or realistic user behaviour. The model updates its topology whenever a timer runs out, rather than updating when a change is detected. The latter is currently not implemented in the model, while this is a functionality of the actual hardware used in the LiFi networks [26]. Every time this model is run for a certain amount of time, it ends up in a state where every AP has the maximum number of EPs connected. While this probabilistic approach is useful to test if the LC can handle the changes in the topology with a maximum number of connected EPs, it can not be applied to testing a more mobile configuration. With respect to the design challenge laid out in subsection 1.2.1, the current emulator model introduces too few variables. This makes it a simple but abstract model. Fit for what it was designed for, but not for thorough interference handling verification. Since the costs of evaluating the interference handling with a normal implementation will spiral if done thoroughly, a more realistic model should be implemented.

## 1.3. Thesis Contribution

In order to create a more realistic Trulifi simulation model, a number of requirements have to be defined to evaluate the model once it is created. This section discusses what the model should include and what the requirements are to reach this goal.

The simulation model should include a full Trulifi setup without the controller. The controller should be excluded from the simulation itself, since a hardware implementation of the controller has to be deployed in order to examine the controller's scheduling capabilities. Excluding the controller from the simulation will also make updating the controller software more convenient. The full simulation setup has to consist of the APs and EPs.

### Requirement list

To facilitate the model validation, certain requirements are defined:

- R1** The correct management messages (from the ITU G.9961 standard [27]) have to be used for communication.
- R2** The simulation should enable a large scale Trulifi system validation.
- R3** Mobility of the EPs has to be supported.
- R4** The system should not converge to the same state each run.
- R5** The model should be as realistic as possible in terms of the following aspects:
  - (a) Light propagation
  - (b) Coverage area
  - (c) Connectivity behaviour
  - (d) Scheduling
- R6** The model should be as computationally inexpensive as possible

With these requirements, the following question will be answered in this thesis.

Can Signify's Trulifi system be modelled in a realistic simulation?

This question is split into the following three sub questions:

1. Can the simulation model be created with existing wireless communication models?
2. How can a realistic LiFi simulation model be achieved?
3. How does the Trulifi simulation model compare to a real Trulifi system?

#### **1.4. Thesis outline**

This thesis will contain a total of 6 chapters. [Chapter 2](#) provides background information on different aspects of wireless communication. [Chapter 3](#) will describe the tools and methods used for creating network simulation models. The design and choices made for the newly developed Trulifi system simulation will be discussed in [chapter 4](#). The simulation is validated through experiments in [chapter 5](#). In [chapter 6](#), the main research question will be answered based on the three sub-questions, and future improvements will be proposed.



# 2

## Background

This chapter addresses the first research question **Can the simulation model be created with existing wireless communication models?** and will provide all needed background information. [Chapter 2.1](#) provides background information on the [electromagnetic \(EM\) spectrum](#). After an explanation of these waves, [section 2.2](#) describes RF communication methods. Visible light communication methods are described in [section 2.3](#). After these methods are explained, wireless communication simulation is discussed in [section 2.4](#). Mobility models are explained in [section 2.5](#). Finally, a discussion of the performed background research is given to answer the research questions in [section 2.6](#).

### 2.1. Electromagnetic spectrum

EM wave behaviour is substantially studied and implementations of these waves can be found all around the world. The EM spectrum is divided into 7 categories as seen in [Figure 2.1](#); radio, microwave, infrared, visible light, ultraviolet, X-ray and gamma ray.

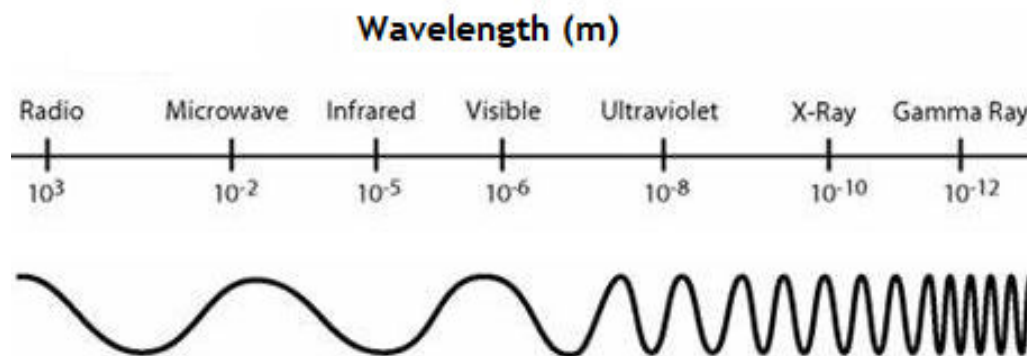


Figure 2.1: Electromagnetic spectrum [28].

The spectrum starts with radio waves, which have a large wavelength and thus a low frequency (30 Hz to 300 GHz). At the other end of the spectrum, gamma rays are found with a small wavelength and a high frequency ( $> 10$  EHz). Visible light can be seen in the middle of the spectrum, with a frequency range of 400 THz to 800 THz. When two waves with the same frequency collide, interference can be observed. The interference observed can either be constructive or destructive, depending on the phase difference between the two signals. The two kinds of interference are shown in [Figure 2.2](#). When the waves are completely out of phase ( $180^\circ$  difference), the amplitude of the resulting

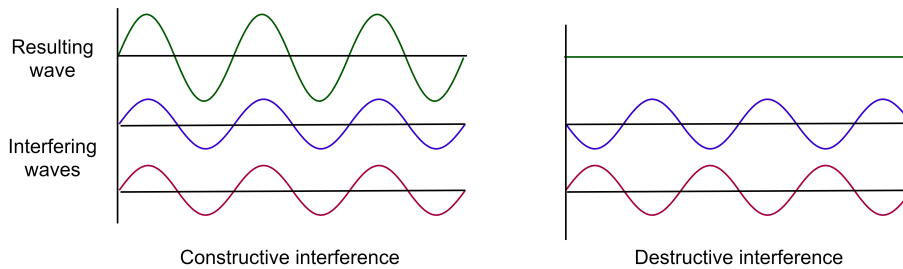


Figure 2.2: Interference of two electromagnetic signals. Phase difference  $0^\circ$  (left) creates constructive interference. Phase difference of  $180^\circ$  (right) creates destructive interference.

wave is completely zero. If the waves have  $0^\circ$  phase difference, the amplitude of the resulting wave is doubled.

The interference as seen in Figure 2.2 can also be observed if a wave passes through a material and collides with a wave at the other side of the material. Since the frequency of visible light is much higher than the frequency of the RF waves, the two do not interfere with each other. The way in which EM waves propagate through space depends on its frequency. The electrical characteristics of materials, together with the EM wave frequency, determine if the wave can pass through the material [29]. Since RF waves have a lower frequency than visible light waves, it is possible for these waves to propagate through non-conducting materials. Even though visible light waves cannot pass through non-conducting materials, they can propagate through good insulators like glass. Gamma rays can even pass between the atoms in detectors and cannot be reflected by mirrors or aluminium. While visible light waves do not experience interference from RF waves, signals do get affected by ambient (sun) light and other light fixtures in the room [30]. Another way for interference to occur is the phenomenon called multi-path fading. This occurs when a wave arrives at a receiver through multiple paths. These different paths not only influence the strength of the signals, but might also change the phase. Within visible light communication, multi-path fading can be discarded as the photodetector area is much bigger than the wavelength of the received waves. This can not be done in RF communication since the wavelength is much larger. Multi-path distortion however does need to be taken into account, since too much distortion can result in a too low Signal-to-noise ratio [14]. In visible light communication, LoS is often necessary since the signal strength deteriorates significantly in absence of LoS. This is also why visible light communication is confined to the illumination area. For RF waves, this LoS is not always needed due to the longer wavelength [30].

## 2.2. Radio Frequency communication

RF communication originates from 1888, when Heinrich Hertz verified the famous Maxwell's equations [31, 32]. Guglielmo Marconi succeeded in transmitting wirelessly over a distance of 2 kilometres in 1895. 7 years later, 1902, Marconi succeeded in the transatlantic transmission. Over the years, more knowledge was gathered about the propagation of radio waves around the globe. While the first wide-scale wireless communication system was installed in 1970, it took until 1990 until wireless communication systems became mainstream [33].

While there are many kinds of RF communication methods, ranging from maritime radio to satellite communication, the key component of all these systems is the antenna. These antennas are used to send and receive the signals needed for communication. By changing the design of these antennas, different propagation patterns can be cre-

ated. These different patterns can be categorized into three main groups; isotropic, omnidirectional and directional antennas [34]. An isotropic antenna radiates its energy equally in 3 dimensions. This kind of antenna cannot be physically created, but do act as a reference for other antenna directivity. Omnidirectional antennas radiate equal energy in a given plane. An example is the half-wave dipole antenna, which has a doughnut shape [34–36]. Figure 2.3 illustrates these isotropic and omnidirectional radiation patterns.

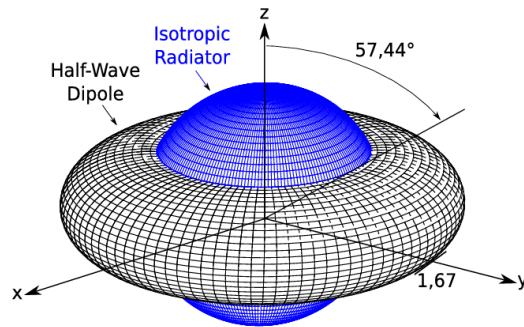


Figure 2.3: 3D propagation visualization of an isotropic and half-wave dipole antenna [37].

The third category of antennas describes directional antennas. This kind of antenna focuses its energy primarily into a specific direction. Figure 2.4 shows the general radiation pattern of a directional antenna. It can be seen that these kinds of transmitters consist of multiple so-called lobes and nulls. Energy is radiated in the direction of the lobes, but no energy is radiated in the nulls. The minor lobes are unwanted radiation. The antenna can be designed in such a way that these minor lobes shrink, but this will also negatively impact the main lobe.

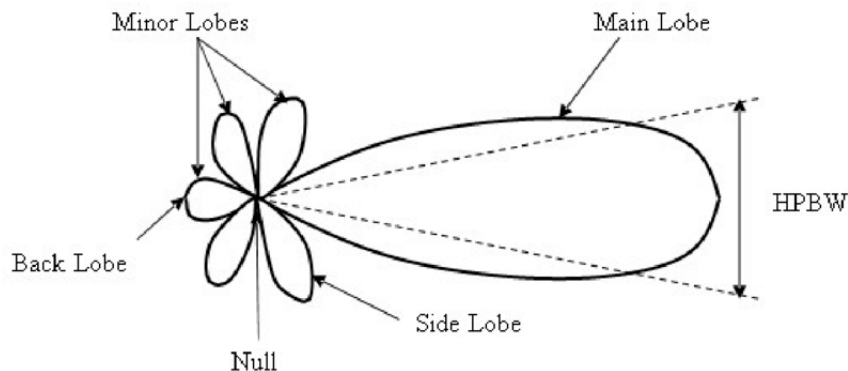


Figure 2.4: Generic directional antenna radiation pattern[38].

### 2.3. Visible light communication

This section aims to provide more insight into VLC techniques. As described in section 1.1, VLC uses the visible light spectrum for transmitting data. While laser diode (LD)s and LEDs can both be used to transmit the data, simultaneous illumination is not possible for LDs. Therefore, the most popular VLC popular transmitters are LEDs [39]. Another key component of a VLC system is the receiver. Photodetectors, like solar-panels or light-dependant resistors, are used to receive the communication signals. A LED is a semi-

conductor light source, which means that it emits light whenever current flows through. The light source can be surrounded by materials with different refractive indexes to create different radiation patterns. The radiation pattern of the most popular semiconductor light sources follow the Lambertian cosine law [14, 40, 41]. This law describes how the light-radiation intensity is proportional to the cosine of the angle between the surface normal and incident light direction. Off-the-shelf LEDs often have a coverage area of around 60°, but are coping with lots of power loss. Signify uses optics to narrow this beam and create a better light distribution to create the perfect directional transmitter. This perfect directional transmitter does not have the leakage radiation (minor lobes) as described in section 2.2.

## 2.4. Wireless network simulation

Simulations are often used in order to predict wireless network behaviour. These wireless network simulations can be deployed whether it is a new design for a network or an existing one and can assist in finding potential shortcomings. As RF is the most common EM wave used for wireless communication to this date, this section will provide insight in RF simulation. After this background knowledge has been described, existing LiFi simulations are looked into.

### 2.4.1. Radio Frequency simulation

When designing a RF communication simulation, one needs to take the signal's properties into account. Electromagnetic signals travelling through a medium will change based on the kind of medium and objects inside the medium. Path-loss, scattering, reflection, absorption and diffraction are only some of the effects that can change a signal's propagation path and power [42, p.47-61]. Even the atmospheric conditions and movement of transmitters and receivers can influence the signal. Modelling wireless communication calls for a mathematical representation of the communication channel. Two approaches can be taken for creating this mathematical representation, a statistical model based on experiments or a model based on geometry principles. Creating a statistical model is not often done, since experiments need to be carried out which cover all possible outcomes [42, p.47-61]. In wireless communication, one also needs to choose the antenna type for transmission, as described in section 2.2. According to [43] there are certain assumptions which should not be simplified when modelling mobile ad-hoc networks. They describe the following six principles, which are often used as a simplification:

1. The world is flat.
2. A radio's transmission area is circular.
3. All radios have equal range.
4. If I can hear you, you can hear me (symmetry).
5. If I can hear you at all, I can hear you perfectly.
6. Signal strength is a simple function of distance.

They convert these six principles into 5 testable concepts and conclude that using these concepts can substantially influence the difference between the simulation and the real system. These principles should only be used to simplify the simulation if the level of realism can be kept the same. In chapter 4, a validation of the new model will be given based on these six simplifications

### 2.4.2. LiFi Simulation models

While one can find many wireless network simulation models which make use of RF, there are not many LiFi network simulation models created yet. Two LiFi simulation models released to the public are analysed in this section [44, 45].

[44] aims to predict system performance with their simulation model in order to create a design space and system planning tool. The simulation model is written in c++ for the simulation framework OMNeT++. It is modelled after experiments carried out on a "Li-Fi Hotspot" system kit. Transmissions are created at the application layer by either TCP or UDP, after which it is passed through the rest of the open system interconnection (OSI) layer towards the physical layer. In this model, the wavelength, bandwidth, modulation scheme and radiant optical power is used to create an optical transmission. Only LoS signals are taken into account in the power calculation. After the power calculation, the background noise is calculated based on shot noise and thermal noise. Finally, the bit/packet error rates are calculated based on the SNR and modulation scheme. Their network consists of two optical access points(OAPs). One of these OAPs is placed on a ceiling, facing downwards, and one is placed on the ground, facing upwards. The latter is a mobile OAP, while the first is static. The simulation resembles a typical LiFi setup as depicted in Figure 1.1. The designed OMNeT++ simulation is verified to predict performance and behaviour of LiFi systems.

In [45], a physical layer LiFi model is created using OMNeT++. Their model is called simVLC. It is designed to simulate a star network topology with one MAC layer master, to which multiple nodes are connected. LoS is implemented as the channel path between devices and reflections are not taken into account. simVLC simulates the OSI layer stack. The MAC layer master has a different height than the connected nodes, but node mobility is only simulated in two dimensions.

## 2.5. Mobility models

The EPs described in section 1.1 are not of a static nature. They are able to join, leave and move around the network. These EPs are not able to move around on their own, but require some interaction by humans. Such an EP can for instance be connected to a mobile phone. Once the person wants to walk to the coffee machine, he takes his phone with him and in that way the EP is moving around the room. Since this is the case for a real-world system, the simulation should support this mobility, see requirement R3. In this section, the sitting and walking patterns of employees is discussed. After this discussion, two popular motion models are explained. Finally, two human movement models are discussed.

### 2.5.1. Employee behaviour in terms of movement

Studies into sedentary behaviour of employees at work are helping to better understand the working routine of employees. These studies can also help to create a motion model in which the average employee behaviour is mimicked. [46] has found that an employee spends on average 68.2% of the total work time sitting. The mean duration of uninterrupted sitting breaks down to 14.9 minutes, while walking during the day corresponds to uninterrupted periods of only 6.8 minutes. Another study [47], which looks into the changes in the behaviour when standing desks are introduced in the office, finds an average which is almost identical to the one in [46]. From these two papers, we find that human behaviour in an office involves many static periods.

### 2.5.2. Random Way point

In [48], the Random Waypoint model is discussed. A node its mobility is characterized by speed, direction and rate of change. In the Random Waypoint model, a random speed and direction is chosen. The node moves for a certain time, and then it waits for another time period before moving again. This model should have an average speed of  $\frac{V_{max}}{2}$ , but this is not the case as it does not reach a steady state. Nodes get *stuck* on long roads with a very low speed, thus slowing down the overall simulation. Due to the model not reaching this steady state, it should not be used to predict time averages of a system. This model can be improved by setting the minimum speed to a positive value instead of 0 m/s. The improvement of 1-20 m/s instead of 0-20 m/s is enough to make sure it can be used.

### 2.5.3. Linear Mobility

The linear mobility model allows movement based on speed and direction as well. The nodes will move at a constant speed in a certain direction, only changing direction if they would otherwise move outside the movement area, or if they collide with obstacles. If the speed is set to 0 m/s, the nodes will remain static.

### 2.5.4. Human behaviour mobility models

[49] describes a *working day movement* model. Their *Office Activity* sub-model consists of an office space and moving nodes (persons). In this model, a node enters and leaves the office space from a certain point, in this case the door. The nodes then walk to their dedicated desks and remain there for a certain time period, based on a Pareto distribution. After the time period, the node moves to another random spot in the room and waits again before returning to the desk. [50] lays out another mobility model, called *Selfsimilar Least-Action Walk (SLAW)*, and is based on four main points;

1. Human straight walks are power law distributed. There are many short walks and some longer walks.
2. People move only in their own movement area, and this differs hugely among people (size of area).
3. Inter contact times is truncated power law. This is the time elapsed between two contact points with the same person. Again, the short time between contacts often occur and the longer duration less often.
4. Human waypoints can be modelled by fractal points. Which means that people prefer certain *hotspots* to visit.

## 2.6. Conclusion

Chapter 2.1 has shown how the EM wave behaviour changes due to its wavelength. From this research, we can conclude that phenomenons like multi-path fading do not need to be modelled. It also became clear that different frequency EM waves cause limited interference to each other. While the perfect directional antenna cannot be created for RF, as explained in section 2.2, it became clear that a perfect directional transmitter can be created with LEDs in section 2.3. Chapter 2.4 described the current RF and LiFi network simulation models. Simplifications can significantly impact the realism of the simulation, which means that these should only be used if the realism can be kept the same. Current simulation models use modulation techniques, bit/packet error rates and reflections to transmit signals. Signify has a clear understanding on how their systems function, e.g. modulation, bit/packet error rates and other properties. Therefore, existing simulations will not be used as they would only add additional unwanted computations. Since requirement R2 states that the simulation model should enable a large-scale system setup, these existing models cannot be used. Mobility models were described in section 2.5. By using the Random Waypoint mobility model for the EPs, the human behaviour can be mimicked to some extent. If the simulation has to be deployed on a specific office space, a combination of the working day movement model and SLAW can be created for optimal mirroring of the real-world system. As the first research question has been answered with a *no*, the need for a new simulation model is validated.

# 3

## Methods and Tools

This chapter provides an overview of the methods and tools used during the design and evaluation of simulation models. When a new idea rises, there are three main methods which can be used to develop this idea into a product: Computer simulations, prototyping and mathematical analysis. If an idea is a mechanical one, mathematical analysis of the mechanics might be done before creating the first prototype. In do-it-yourself projects, prototyping is often the first step taken to see if the mechanical movement is even possible. When the first few prototypes are created, a simulation might be created to test other configurations or to evaluate how a larger version would behave. This is only one example of a design flow, and many other routes can be taken to reach the end goal. In this thesis, a computer simulation is described which replaces a large scale deployment of a hardware product. To reach this goal, a mathematical analysis performed by Signify is used within this simulation model. The model is validated in a later stage, see [chapter 5](#), by comparing experiments performed on a real-world Trulifi setup. In this way, all three designing methods are used to create a product which can be used to evaluate system performance. [Chapter 3.1](#) provides a detailed overview of the simulation environment and most common network simulation engines. The mathematical analysis performed by Signify will be described in [section 3.2](#).

### 3.1. Simulation Environment

Validation of a new idea through the use of hardware might get very expensive rather quickly, small changes might need a whole new hardware product to be build. A simulation is often used to prototype solutions and predict if the solution is able to solve the problem at hand, without the need of implementing any hardware [51]. Simulations can be used as design tools before building the real system, or as a way to evaluate the system under different circumstances. Within simulations, an optimal solution can often be discovered faster and with more ease than using multiple iterations of a hardware prototype. Once the system is thoroughly tested through these simulations, a hardware product might be created. To validate the full performance of a total system, a simulation can be created which connects to this hardware product. The total experimental setup is then called an emulation, as it imitates a real-world process or system over time [52]. According to [53–55], a network emulation is an experiment technique which contains both real and virtual network components. The previous statement describes the main difference between the two. Simulation contains only virtual components, while emulations contain both virtual and real (physical) components. The second main difference between the two is that



emulation models are designed to be operated in real-time, while simulation models are not [51]. Due to these differences, emulation models are more effective when testing for certain performances, or reactions of the real part of the system. It enables its user to reproduce extreme situations, like extreme numbers of connected users, without the use of the entirety of the hardware [53]. Before starting with the design of the simulation or emulation model, the existing simulation engines should be compared based on their possibilities and behaviour.

### 3.1.1. Simulation engines

A great variety of network simulation tools exist, and choosing one fit to the task at hand can be difficult. [56] provides 6 widely used network simulation tools: NS-2, GloMoSim, J-Sim, OMNeT++/OMNEST, OPNET and QualNet. [57] provides a comparison of four of these network simulation tools: OMNeT++, NS-2, NS-3 and OPNET. In this section, an overview of these widely used simulation tools will be given, and each tool will be compared to OMNeT++/OMNEST.

#### OMNeT++/OMNEST

OMNeT++ (Object Modular network Testbed in C++) is a discrete event simulation environment based on the C++ programming language. It offers an Eclipse based [integrated development environment \(IDE\)](#) and a graphical runtime environment [58]. OMNeT++ achieves modularity by building models from so-called modules. The most basic modules are called `simple modules`. Simple modules can be grouped together, resulting in `compound modules`, but they can not be split up. There is no limit in the hierarchy levels when it comes to combining these compound modules. Individual modules can be linked together either through communication channels or directly. Each module can be used as a component for a more complex module, which makes reusing of modules in multiple models possible, without the need to re-declare each module. Users can choose between running their simulation through the command line or through the [IDE](#). Users can even make an emulation model from their simulation models by using the real-time scheduler. This scheduler is able to capture packets from real network devices and injecting them into the simulation and sending packets to the network [59, 60].

#### NS-2 [61]

NS-2 (Network Simulator v.2) is a discrete event simulation tool as well. NS-2 is open-source and is written in C++. All [OSI](#) layers can be simulated with the use of NS-2. While it supports Wi-Fi and satellite communication models, no LiFi communication models are implemented up until now. There are two main reasons for not using NS-2, the first being that official support has been cut-off since 2011 due to the upgrade of NS-2 to NS-3. The second reason for not using NS-2 is that performance is limited for environments with numerous nodes, which is requirement [R2](#) for the simulation model.

#### NS-3

NS-3 was designed to overcome the abstractness in the NS-2 model environment and create more realistic models. NS-3 also enables the code used to create the simulation to be used on real implementations, which NS-2 did not enable. While NS-3 uses C++ as its programming language as well, it lacks a [graphical user interface](#) for evaluating the simulation. Since mobility of nodes is to be included in the simulation model, a [GUI](#) is preferred for easing the validation process.

### OPNeT [62]

OPNeT (Optimized Network Engineering Tools) modeller is designed to simulate network behaviour and performance. As is the case with OMNeT++/OMNEST, NS2 and NS3, the main programming language is C++ and it is based on discrete events. OPNET became part of a company *Riverbed* in 2012, and was renamed Riverbed Modeller. Riverbed Modeller does not support LiFi at the time of writing and is linked to license fees. Riverbed Modeller also has limited performance when validating large node environments, just as NS-2.

### GloMoSim [63]

GloMoSim (Global Mobile Information System Simulator) is designed to simulate network protocols to evaluate wired and wireless network systems. It is written in Parsec, a parallel programming language based on C. GloMoSim does not support connection to devices outside the simulation, as all events must be generated by a node inside the simulation. Since it cannot be connected to hardware and there are no new updates since 2000, GloMoSim is not a suitable option.

### J-Sim

J-Sim is written in Java and is component-based, which means that everything ( a link, node or protocol) is a component inside the large structure. It was initially designed for wired network simulation, but it has a wireless extension. It is open source, but its behaviour does not scale well for large amount of nodes. Due to scalability being one of the key requirements(requirement **R2** of the Trulifi system model, J-Sim is not a viable option.

### Qualnet

Qualnet is built on GloMoSim, and thus uses Parsec as its programming language. Qualnet its main advantage is that it can be used to evaluate a digital twin of the hardware system. Qualnet can also be used as an emulation tool, something that GloMoSim could not. However, Qualnet is a commercial product from which Signify does not have a licence.

#### 3.1.2. Simulation Realism

System simulation varies greatly in terms of complexity. To keep the computational power low, complex models can be simplified, but this results in a certain level of abstraction. Two main questions arise due to this abstraction; "What makes a good model? How to obtain a good model?"[52]. There are no universal answers to these questions, as models differ greatly. In [64], the adequacy of a simulation model is discussed. It is stated that a model should only be used in the way it was intended to be used, and that careful considerations should be taken in terms of abstraction and its representation. It also recommends to carefully assess the adequacy of the model. Even though these points are stated for simulation models instead of emulation models, they can still be taken to heart when designing the latter. Especially researching the abstraction level of the model should be carried out thoroughly. [51] states that the model should be designed with an accuracy that ensures that the emulation results are still statistically meaningful. For the Trulifi system simulation meaningfulness validation, multiple experiments are performed on the real-world system to compare to the simulation. The results of these experiments and the validation of the simulation are discussed in [chapter 5](#).

### 3.2. Mathematical Analysis

Signify has performed some real-world experiments to collect data from their Trulifi system propagation to create so-called *power equations* of the system. Developing these equations was not part of this thesis and is only explained for the sake of completeness. The gathered data consists of the background noise and the **power spectral density (PSD)** which the receiving device experiences at different distances from the sending device. The measured background noise power is used to create a noise floor of the system. While this noise floor is found to be uniform over the measured area, the **PSD** depends on both the vertical and horizontal distance from the transmitter. The gathered **PSD** data can be found in [Figure 3.1a](#). It has been normalized to avoid confidentiality issues.

From this received data, two equations are created through the use of MATLAB. The equations are used to calculate the **SNR** inside the simulation, but the exact equations will not be mentioned in this thesis as this is intellectual property of Signify. By using these equations, the propagation behaviour of a Trulifi system can be reconstructed and the received power at any point can be calculated. These MATLAB equations are extrapolated at a certain distance between the **AP** and **EP** and an offset needs to be used to reconstruct the curve at different distances. [Figure 3.1b](#) illustrates the usage of the formulas to reconstruct the curve, where  $f(r)$  is the with MATLAB derived equation. With the curve and the previously explained noise floor, the **SNR** of the signal can be calculated at any coordinate.

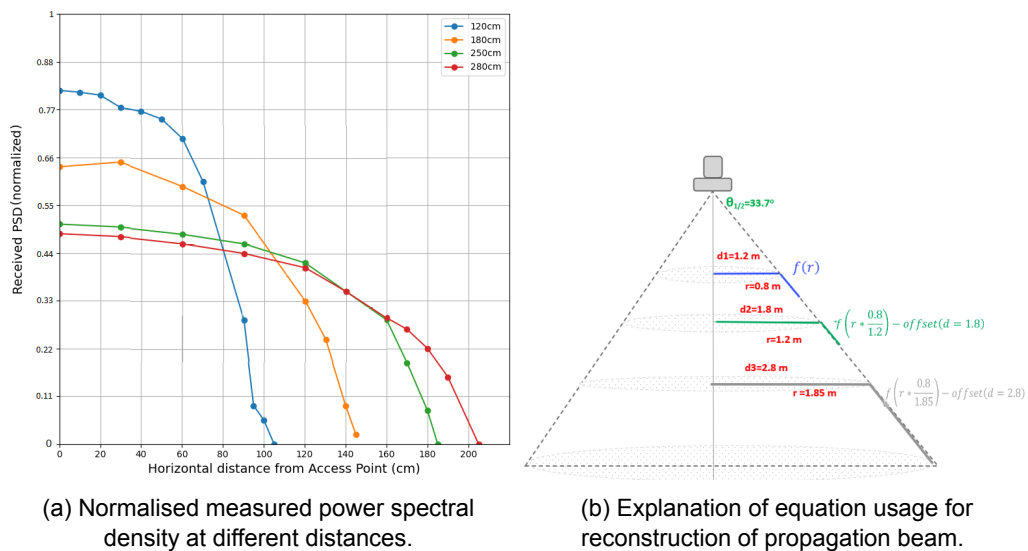


Figure 3.1: Mathematical analysis of Trulifi Coverage area.

# 4

## Design

This chapter will describe the design of the Trulifi system simulation model, based on the requirements discussed in [section 1.3](#) and the research provided in [chapter 2](#). As explained in [section 3.1](#), the model will be created in the OMNEST environment. The simulation architecture will be explained in [section 4.1](#). [Chapter 4.2](#) describes the management messages from the ITU G.9961 standard, which the model implements. The state machines of an AP and EP, as implemented in the simulation, will be described in [section 4.3](#). A validation of the simulation its realism, based on the design choices, will be discussed in [section 4.4](#).

### 4.1. Architecture

This section will describe the architecture behind the Trulifi simulation model. As OMNEST uses modules, the architecture will be described from the overall network to the simple modules. [Figure 4.1](#) shows the network setup of the model. The network consists of two kinds of nodes, APs and EPs, which represent the real-world devices. The visualizer module enables the IDE to show movement, backgrounds, text-bubbles and other visual effects. A neighbourTable module is implemented to allow the APs to report the interference to the LC and for the LiFiMedium module to keep track of where devices are within the simulation. The LiFiMedium module describes the LiFi communication channel. It keeps track of the transmissions, noise and transceivers itself and computes when, where and how transmissions arrive at other devices.

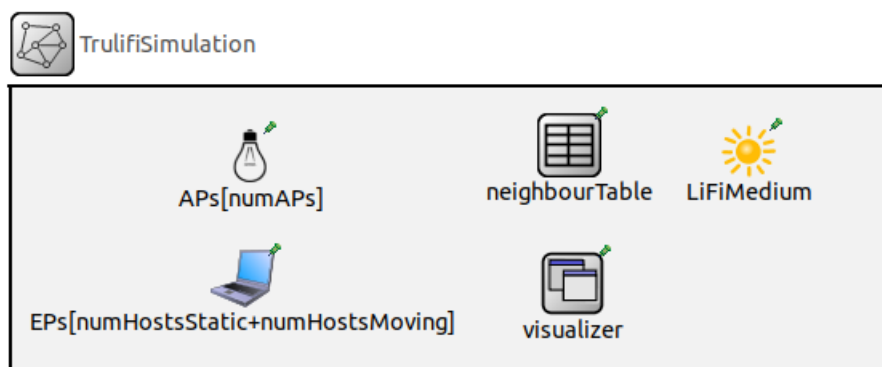


Figure 4.1: Highest layer in Trulifi simulation model.

### 4.1.1. Devices

The simulated AP and EP have a different architecture, as an AP has to communicate with the real-world LC, while the EP does not. Figure 4.2a and Figure 4.2b show the architecture of an AP and EP respectively. Both of the devices are equipped with a LiFi interface sub-module. The LiFi interface sub-modules can be seen in Figure 4.2c. This interface module contains the g.VLC MAC layer, which is in charge of creating and processing messages according to the ITU G.9961 Standard, the messages will be described in more detail in section 4.2. Finally, the LiFiRadio module is in charge of transmitting and receiving the messages. The APs are equipped with two additional sub-modules, called the socket and the App. The socket is the gateway to the real-world network, and the App module is in charge of handling the messages between the socket and the LiFi interface.

### Mobility module

Both of the devices are also equipped with a mobility module. The mobility module enables switching between the available OMNEST mobility types, like *StaticGridMobility*, which places devices in a grid, and *LinearMobility*, which allows linear movement of devices. While the APs will be placed statically in a grid with the *StaticGridMobility* module, EP mobility has to be implemented for a fair representation of the real-world Trulifi system as per requirement R3. As LoS is needed to create a connection, the APs and EPs are oriented facing down and up respectively. In OMNEST the *TurtleMobility* motion module allows users to define paths for the devices to follow in an *Extensible Markup Language (XML)* script. As described in subsection 2.5.1, employee moving patterns in offices tends to be rather static, as employees tend to spend 68.2% of the total working time sitting down. By making use of the *TurtleMobility* motion module, this behaviour can be mimicked in the simulation. Since in a real-world system some EPs will not move at all, this is also implemented by using the *StationaryMobility* model, which does not support mobile EPs. The EPs are placed at random coordinates each time the simulation is run, which should ensure that requirement R4 is satisfied. For the validation of the simulation, the *TurtleMobility* is used to replicate the real-world experiments.

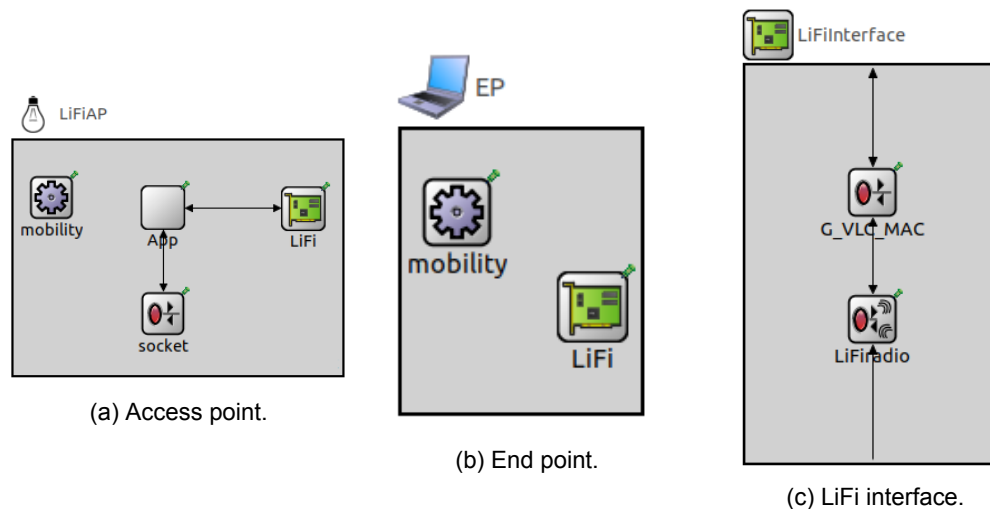


Figure 4.2: Simulation architecture for devices.

### 4.1.2. LiFiMedium

As explained in the previous section, the LiFiMedium is in charge of the messages within the simulation. Physical properties of the communication channel can be implemented

within sub-modules to change the channel's behaviour. Examples of these physical properties are path-loss, obstacle-loss and propagation speed. Another sub-module of the LiFiMedium is the analogModel, which describes how the transmissions are turned into receptions. There are multiple models which can be used off-the shelf, like a dimensional model, which allows the message power to change over time and/or frequency. The most simplistic model that is already implemented is the UnitDisk model, which determines reception power by evaluating the distance between sender and receiver, as can be seen in [Algorithm 1](#). When the distance between the two devices is smaller than or equal to the coverage area, the message can be received properly, and the message will be passed to the next module. If the distance between the two is larger than this coverage area, the message cannot be received and will therefore not be passed to the next module. Analog models like the dimensional model use more computational power than the simple UnitDisk model as the bit-rate, modulation technique and physical properties are included to calculate the reception possibilities.

Since the system behaviour of the Trulifi system is known, and captured through the power equation as explained in [section 3.2](#), there is no need to use a model like the dimensional one. Using the UnitDisk model would however be too simplistic, as the light propagation and coverage area would not be realistic (see requirement [R5a](#) and requirement [R5b](#)). To remain realistic, the reception evaluation of the model will be adjusted to use the power equation instead of the distance. By using the UnitDisk model as the basis, the computational power can be kept lower, which satisfies requirement [R6](#). The new algorithm is described by [Algorithm 2](#). First, the *reception threshold* and *interference threshold* are initialized according to the system specifications. Each time a message is received by a device, the received power is calculated. With this received power and the noise floor, the [SNR](#) can be calculated. If the calculated [SNR](#) is larger than the reception threshold, the message is deemed receivable. If this is not the case, but the [SNR](#) is still higher than the interference threshold, the message can not be demodulated, but it can be recognized as interference. Finally, if the [SNR](#) is smaller than this threshold, the reception fails.

---

#### Algorithm 1 UnitDisk reception calculation

---

```

1: CA = Coverage Area
2: Message arrives at receiver
3:  $D = |Coord_{sender} - Coord_{receiver}|$ 
4: if  $D \leq CA$  then
5:   Reception successful
6: else
7:   Reception failed
8: end if

```

---



---

#### Algorithm 2 LiFiMedium reception calculation

---

```

1: RT = Reception Threshold
2: IT = Interference Threshold
3: message arrives at receiver
4:  $P_{Received}$  = Power Equation( sender, receiver)
5:  $SNR = P_{Received} - Noise_{floor}$ 
6: if  $SNR > RT$  then
7:   Reception successful
8: else if  $SNR > IT$  then
9:   Message causes interference
10: else
11:   Reception failed
12: end if

```

---

## 4.2. Message exchange

According to requirement **R1**, the model has to use the data link layer messages from the ITU G.9961 standard [27]. The message exchange is divided in two layers. The first layer describes the communication between the **AP** and the **LC**, while the second layer describes the communication between the **AP** and the **EP**. The first layer communication was already implemented in the stress-test model, described in subsection 1.2.2, but the communication messages will still be discussed in this section. As the second layer was implemented from scratch, this will be discussed in more detail. Medium access is managed through a **medium access plan (MAP)** which the **LC** creates. Every **MAP** message contains the schedules for each **AP**. The **APs** should create schedules for the **EPs** within their allowed channel access time.

### Communication between access point and controller

Communication between the **AP** and **LC** can be divided into four main groups; Registration, Topology, Scheduling and Fast Handover. The registration group describes the messages used to connect the **AP** to the **LC**. Once the **AP** connects to the **LC**, the topology group is in charge of sending the topology information to the **LC**. This topology is used by the **LC** to compute the scheduling of the connected **APs**, which is sent to the **APs** through the messages in the scheduling group. The fast handover group also involves the **EP**, as only they can sense when to switch between **APs**. The **EP** then sends a request to the **LC** through the **AP** it is currently connected to, and the **LC** informs the other **AP**.

A typical communication exchange between an **AP** and **LC** can be seen in Figure 4.3. In this conversation, the **AP** boots up and first looks if there is a controller present in the network. If there is, it will ask to connect to said controller and once it is, it will send its connected **EPs** and the interference information. The **LC** will send a schedule based on this information. The **AP** will re-register itself periodically to the **LC** to keep up the connection. Finally, the **AP** reports changes in topology to the **LC** as soon as they are noticed.

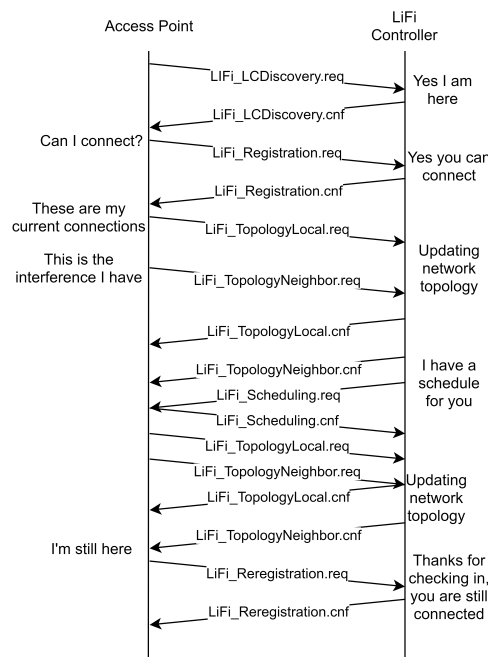


Figure 4.3: Typical conversation between an access point and the LiFi controller.

### Communication between access point and end point

Communication between the AP and EP can also be divided into the four groups; Registration, Topology, Scheduling and Fast Handover. From the moment that an EP is connected, the AP divides the received schedule into time slots for the EPs to access the channel in. Since the EPs cannot sense each other, this fine schedule is needed in order to manage the medium. The EP will only access the channel during their assigned time slot. If the EP is not connected, it can try to connect during a so-called *registration slot*. In a configuration with multiple EPs, this will cause some collisions. This occurs when multiple EPs try to connect at the same time. If the EP does not receive a confirmation of its connecting attempt, it will try again after 1 s for a total of four times. The EP will report changes in interference to the AP, which will update its topology and inform the LC.

A typical connection conversation between an AP and EP can be seen in Figure 4.4. From this conversation, it can be seen that the AP sends out a schedule periodically, so that any EP within the coverage area can pick this up. If the EP is not connected yet, it will attempt to connect to the AP from which it receives the strongest power. Once it is connected, it will set a re-registration timer, to make sure it will renew its registration. Once the EP notices some interference, it will inform the AP during the reserved time slot. If the EP wants to disconnect or try a fast handover, it will also send this during this reserved time slot. If a schedule is received, and the EP's re-registration timer runs out within the coming MAP-cycle, the EP will make sure to send the re-registration request during the coming available time-slot in order to keep up the connection.

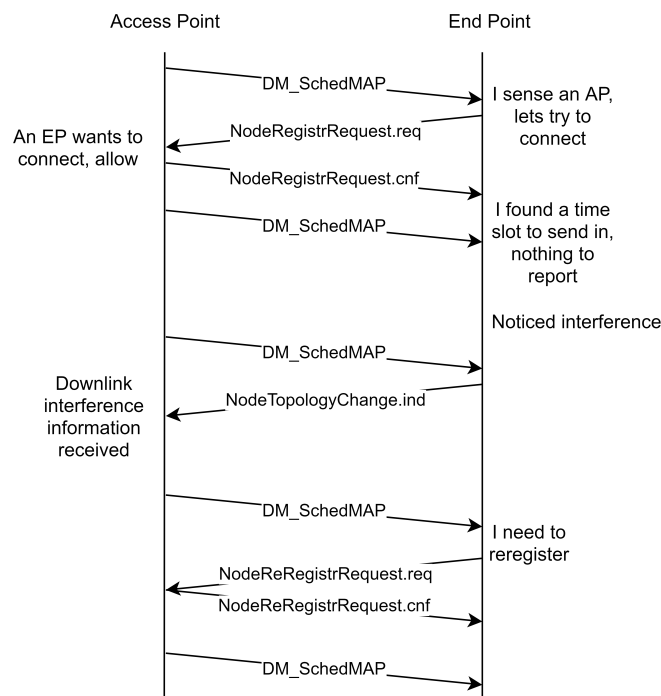


Figure 4.4: Typical conversation between an access point and an end point.



### 4.3. State machines

While some similarities are found in the message exchange between the AP - LC and AP - EP, the behaviour of the AP and EP is different. This section will describe the state machines of the two devices in more detail. The state machines are based on the ITU G.9961 messages [27] to allow requirement R1 to be satisfied.

The AP's state machine can be found in Figure 4.5. Once a schedule is received from the LC, the AP will create a Time Division Multiple Access (TDMA) schedule according to the number of connected EPs. Once this schedule is created, it will set a timer to send this schedule as a broadcast to all EPs within its coverage area. The LC sends out a heartbeat, to allow synchronization if the APs are out of sync. If a message arrives from an EP, the SNR will be calculated. If the SNR is higher than the reception threshold, the message can be demodulated and a response can be created. If the SNR is still higher than the interference threshold, the EP is added to the interference list. If the AP receives a registration request from an EP, it will first check if that particular EP is already registered. If the EP is not yet registered, the AP will check if it has a free spot left. If there is, a registration acknowledgement message will be sent, a re-registration timer will be set and a topology message will be sent to the LC. If the EP is already connected according to the AP however, a re-register now message will be sent. This will only occur if the registration message has not arrived at the EP. If this re-registration timer expires, the EP failed to re-register. This is only allowed once. If the EP fails to re-register twice, the AP will send a forced-resign message. If no resign request is received within 100 ms, the forced resignation message will be sent once more. After another 100 ms have passed, the EP will just be removed from the connected EPs and the topology will be updated to the LC. An EP is also allowed to resign itself from the AP. If such a request is received, a check is done to see if the EP is connected at all, after which it is either removed from the EP list (with confirmation to the EP) or a message is sent which states that the EP was not connected at all.

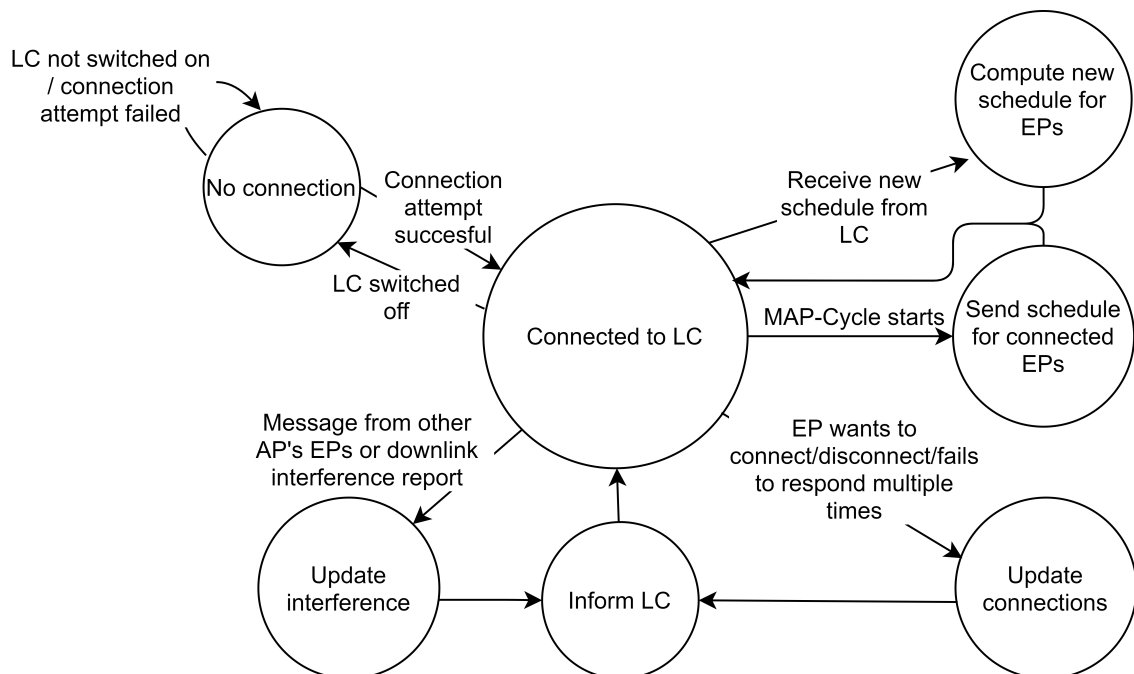


Figure 4.5: Simulated access point state machine.

The state machine for the EP can be found in Figure 4.6. At the time of starting the simulation, the EP is unconnected. Once it receives a message from an AP, the SNR will be calculated. If the SNR is higher than the interference threshold, the message its source address and corresponding SNR will be saved to a table, which is used later. Then, the EP will check if it is already connected. If it is still disconnected, it will check if the message was a directed or a broadcast message. If the message is directed, this means it is a response to a request message, which can mean that it is allowed to connect. If it is a broadcast message, the registration slot can be found from the schedule the message brings, and the EP can schedule its registration request. If the EP is connected when receiving a message, it will first check if the source address is the same as that of the AP it is connected to. If this is the case, and the message is meant for itself, the message can be processed, and a response can be created. If the source address is of another AP, a handover might be needed, the conditions are checked and a handover request can be created if needed. Once an EP receives a schedule from its respective AP, it will check if it needs to send something during its assigned time slot. The EP can either send a resignation request, re-registration request, topology change or fast handover request during this assigned time slot. If more than one message is queued to send, it will send one during this time slot and send another during the next available time slot. If enough time is available to send the queued message within this time slot, it will be sent as well.

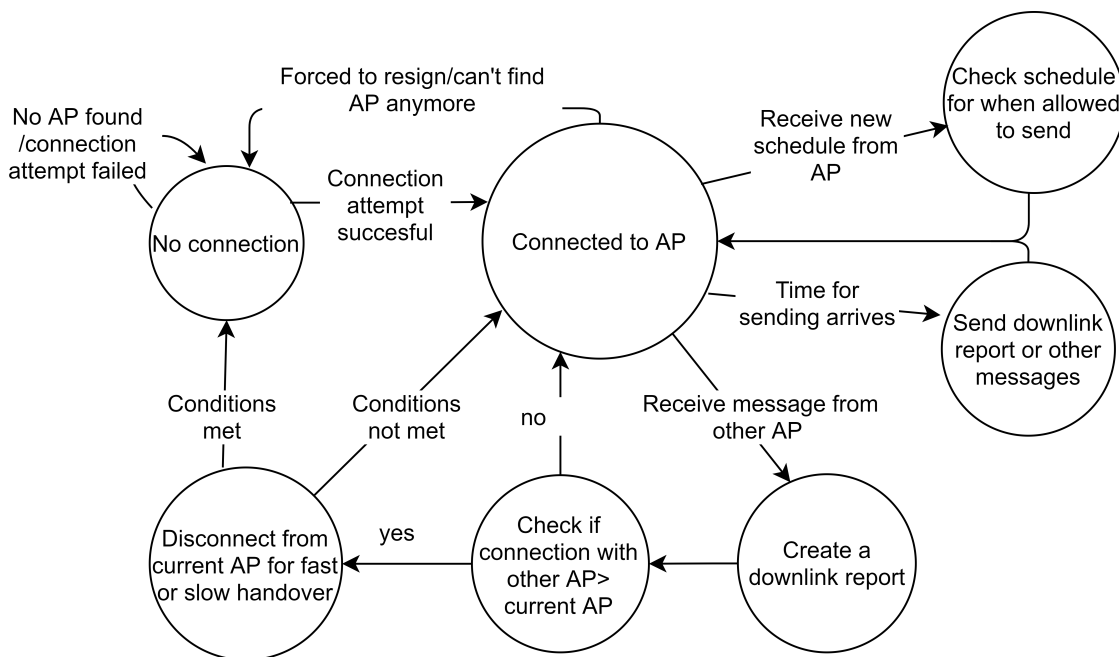


Figure 4.6: Simulated end point state machine.

## 4.4. Validation

This chapter provided the choices which are made to answer research question number two, **How can a realistic LiFi simulation model be achieved?**. This section is used to provide an answer to this question. If one of the simplifications discussed in [subsection 2.4.1](#) is used within the simulation, the level of realism should be kept the same in order to create a realistic model. This section will therefore provide an early validation of the realism based on these simplifications. The following list shows the simplifications again;

1. The world is flat.
2. A radio's transmission area is circular.
3. All radios have equal range.
4. If I can hear you, you can hear me (symmetry).
5. If I can hear you at all, I can hear you perfectly.
6. Message strength is a simple function of distance.

Orientation of the simulated [EPs](#) and [APs](#) is defined as upwards and downwards respectively. The distances between an [AP](#) and [EP](#) are significantly small compared to the radius of the earth. This size difference allows for the first assumption to be made without changing the realism and accuracy of the simulation. The transceivers of Signify's Trulifi system are designed in a way that they achieve a symmetrical light distribution. Hence, the [APs](#) and [EPs](#) modelled within OMNEST are designed with the same behaviour and all devices have equal range. The transmission area can therefore be simplified to a circle, but as a function of distance. If an [EP](#) is closer to the [AP](#) the circle has a smaller diameter than if the [EP](#) is further away. Since the system is symmetrical, if an [AP](#) can see an [EP](#), the [EP](#) can see the [AP](#). This means that simplifications two, three and four can be used because this also happens in the real-world system. message strength is calculated based on the power equation, which has been obtained from real-world measurements. So while this is breaks down to a simple function of distance, it is based on the real-world system behaviour. This allows the use of simplification six while keeping the realism level the same. messages are properly received whenever the calculated [SNR](#) is above the reception threshold. If a message is received between the interference threshold and the reception threshold, the message cannot be demodulated, thus simplification five is the only simplification which has not been used. By creating a LiFiMedium module and by implementing the ITU G.9961 standard data link layers, a simulation is designed which approaches the real-world system. [Chapter 5](#) will provide a thorough validation of the simulation.

# 5

## Evaluation and validation

This chapter analyses the performance of Trulifi simulation model based on LiFi-signal propagation. The experimental setup used to validate the LiFi transmitter is described in [section 5.1](#). The results of these experiments will be provided in [section 5.2](#). [Chapter 5.3](#) provides the experimental setups used to validate the simulation as a whole, and [section 5.4](#) will provide the results of these experiments. Finally, the third research question, **How does the new simulation model compare to a real Trulifi system?**, will be answered in [section 5.5](#).

### 5.1. Analysis of the LiFi transmitter

This section will provide an understanding of the setup used to validate the Trulifi simulation model its LiFi radiation pattern. In order to validate this Trulifi simulation model, experiments with a real-world implementation were executed. The results from these experiments are later compared with results obtained from simulation in [section 5.2](#).

#### Real-world setup

The real-world Trulifi system implementation consists of two devices, one AP and one EP. The goal of this experiment is to evaluate the SNR which can be calculated from EP measurements at different coordinates in a three-dimensional space. In order to achieve this goal, an AP is attached to the ceiling at a height of 2.9 m. The EP is attached to a so-called beam scanner. The beam scanner measures 2.6 mx2.6 m and consists of two movable axes. The EP can have a variable height by changing the length of the aluminium extrusion rail to which it is attached. By moving the EP around this three-dimensional space, the received power and noise levels can be retrieved at multiple coordinates. With these measurements, the SNR distribution heatmap can be created by subtracting the noise level from the received power. The real-world Trulifi system setup is shown in [Figure 5.1](#).

#### Trulifi simulation setup

For the simulated version, the topology is similar to the one used in the real-world Trulifi system implementation. The AP is placed at a user-defined ceiling height, and the EP is able to move in a three-dimensional space within the AP. [Figure 5.2](#) shows the simulated setup. The grey area in [Figure 5.2](#) describes the movement restrictions of the EP around the AP. To ensure that the EP visits roughly all places within this three-dimensional space, the node mobility is configured as the so-called *TurtleMobility*. This mobility-type is defined in OMNEST and enables the user to provide the simulation with a custom movement pattern, which is described in a XML script. While one could easily write this script by

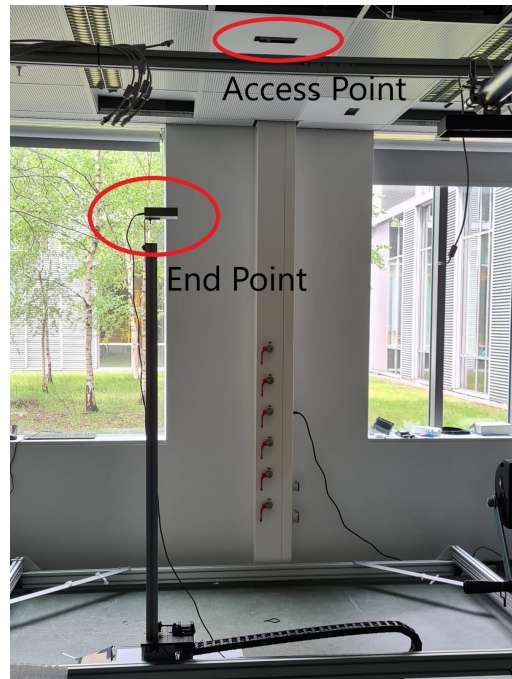


Figure 5.1: Beam scanner setup with End Point at 70 cm distance from Access Point.

hand for a small amount of coordinates, it gets tedious when declaring more coordinates to move to. To overcome this, a Python script was created which writes the XML script by using a loop through all three dimensions. This enables the user to quickly redefine the number of coordinates that the EP should visit. Once the simulation is started, the EP starts to move as defined and keeps repeating the defined path. Whenever a message is received from the AP, the current coordinates and the SNR are saved to a text file. As the simulated version its transmission power depends on the in section 3.2 defined power equation, one could argue that the results can also be obtained by running coordinates through the power equation. However, running the simulation was necessary in order to verify if any message can be received outside the cone-like coverage area of the AP. For both setups, a Python script is used to generate the SNR heatmaps.

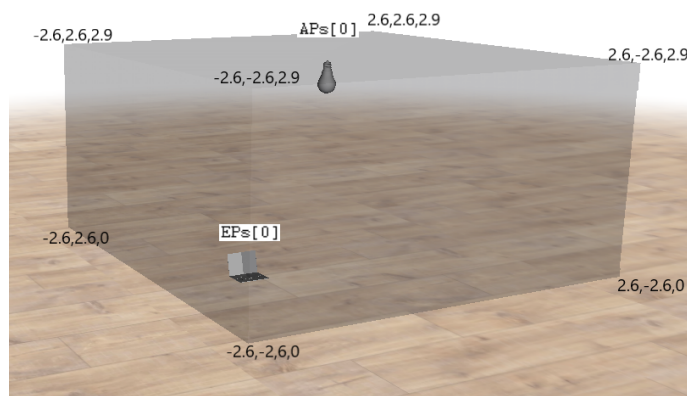


Figure 5.2: Simulation setup transmitter analysis, grey box indicates movement restrictions.

## 5.2. Light propagation validation

This section will provide the results of the experiments, performed on the Trulifi simulation and the real-world Trulifi system to validate the radio model, as described in [section 5.1](#). First, the real-world Trulifi system setup results will be shown. After this, the simulation results will be presented. This section will conclude with a discussion of these results.

### Real-world setup

To evaluate the real-world Trulifi system coverage area and rule out any outliers, in terms of the received power at a particular point, the experiment is run multiple times and the average is taken for comparison. The EP is used to measure the received power at five distances from the AP. Only five distances were chosen due to the available aluminium extrusion rails and the time it takes per measurement. The EP can be moved up and down on these extrusion rails, but then the extrusion rail interferes with the LoS at certain coordinates when moving in the xy-plane as can be seen in [Figure 5.3a](#). To prevent the loss of LoS within the coverage area, the EP is placed on top of the extrusion rail as shown in [Figure 5.3b](#).

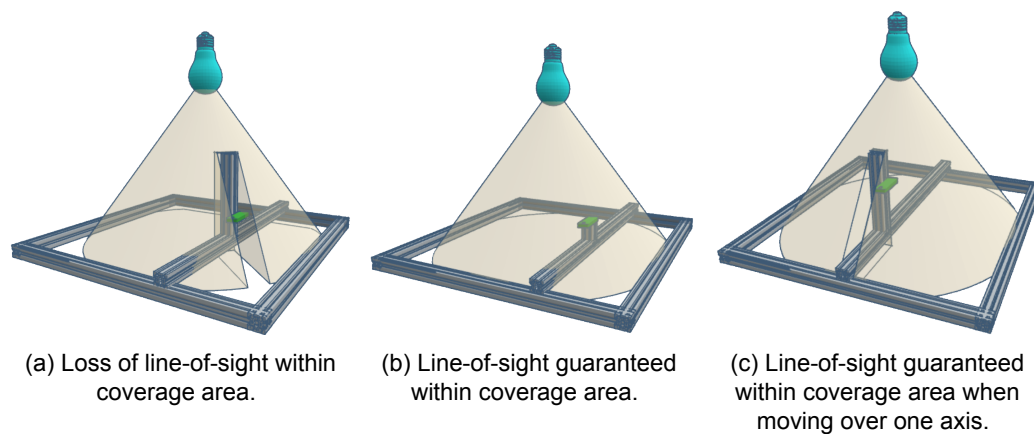


Figure 5.3: LoS illustrations of beam scanner.

Each axis is divided in five coordinates in order to cut down the time needed to perform the experiment. This resulted in a grid with 25 coordinates, which was used at each distance from the AP. After the first run, it was found that the received power decreases with an increasing distance from the center, which is immediately underneath the AP. Another observation that was made is that this power is the same on a circle around the center, [Figure 5.4a](#) illustrates these observations.

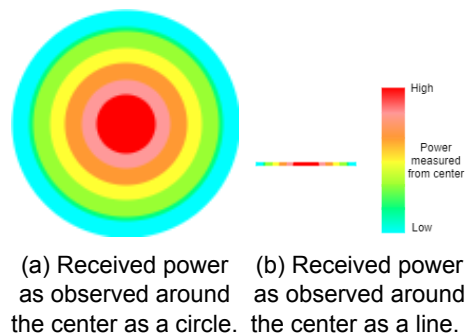


Figure 5.4: Observation of received power in first beam scanner experiment.

As performing 25 measurements over five distances takes a lot of time, and the power is symmetrical around the center, the choice was made to perform the measurements over one axis only. This resulted in 5 measurements per distance. The measurements then make up the line as shown in Figure 5.4b which can be converted to a 3D figure by rotating them around the z-axis. Changing the plane in which the optical beam was tested allowed for measuring at 10 distances instead of five. The issue with the LoS does not occur if the beam scanner is moved over one axis and the EP is attached correctly. This is illustrated by Figure 5.3c.

The increase in distances comes from the fact that the LoS can be guaranteed within the coverage area when moving the EP vertically and over one axis. The average results are converted into a three-dimensional heatmap, which can be found in Figure 5.5. Due to confidentiality reasons the actual SNR values can not be shown. Therefore the graphs contain min-max normalized values.

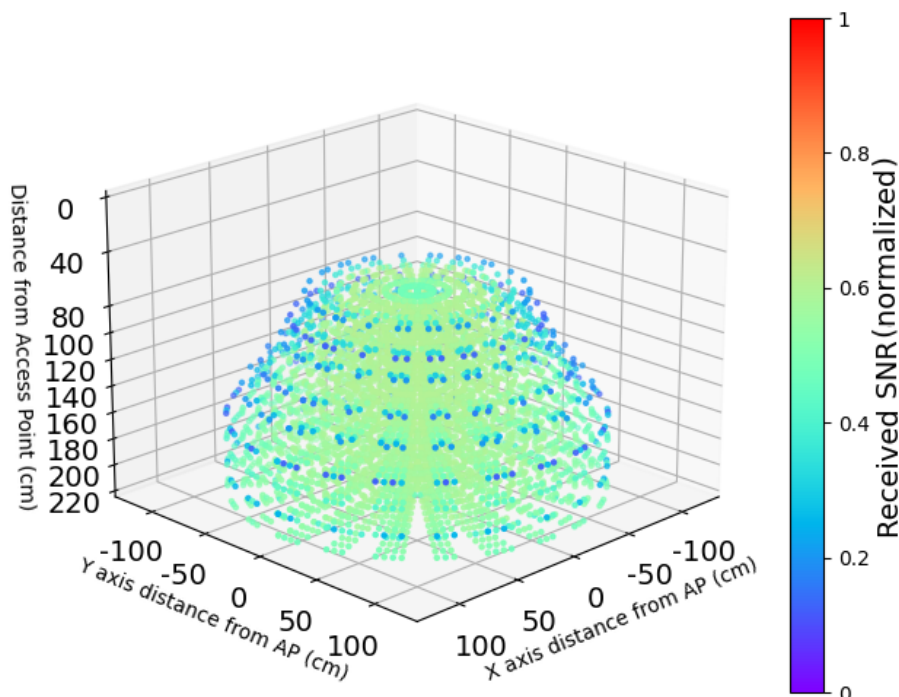


Figure 5.5: 3D heatmap of beam scanner measurements.

From Figure 5.5, the cone-like shape can be seen with the top cut off. As photodetectors are used to receive the data, the devices are limited by its physical characteristics. A photodetector can only collect a certain amount of light before it saturates. This means that if the EP moves to close to the AP then it over saturates due to too much light collected and the photodetector is not able to process this. This results in a connection loss, if the distance is smaller than 60 cm. Based on the heatmap, it can be seen that the power is not linear with respect to the distance from the z-axis. This is due to the used optics. The receiver's photodetector can only collect a certain amount of light. So, the receiver reports this maximum amount of light before over-saturating.

### Trulifi simulation setup

Due to the TurtleMobility script, the node will visit each point multiple times until the user stops the simulation. As the received power is not measured but calculated, running the simulation multiple times will give the exact same result. Therefore, the experiment

has to be performed only once to evaluate the Trulifi simulation beam. The EP should however be able to receive a message from the AP to calculate the power at every visited point, thus the simulation should be run for a sufficient time period. By using an automated script, the way-points for the mobility model can be generated. The simulated AP will send a message at the start of every MAP-cycle. The EP moves to the next coordinate within this time, and then waits for a duration of 40 ms before moving again. With a total of 1000 coordinates, this would mean the total runtime (EP moving and receiving a new message) would have to be at least 80 s. Within this time, the node should have received a message at every coordinate which was predefined. If no message is received, there is no power to be calculated. Figure 5.6 shows the results of this experiment.

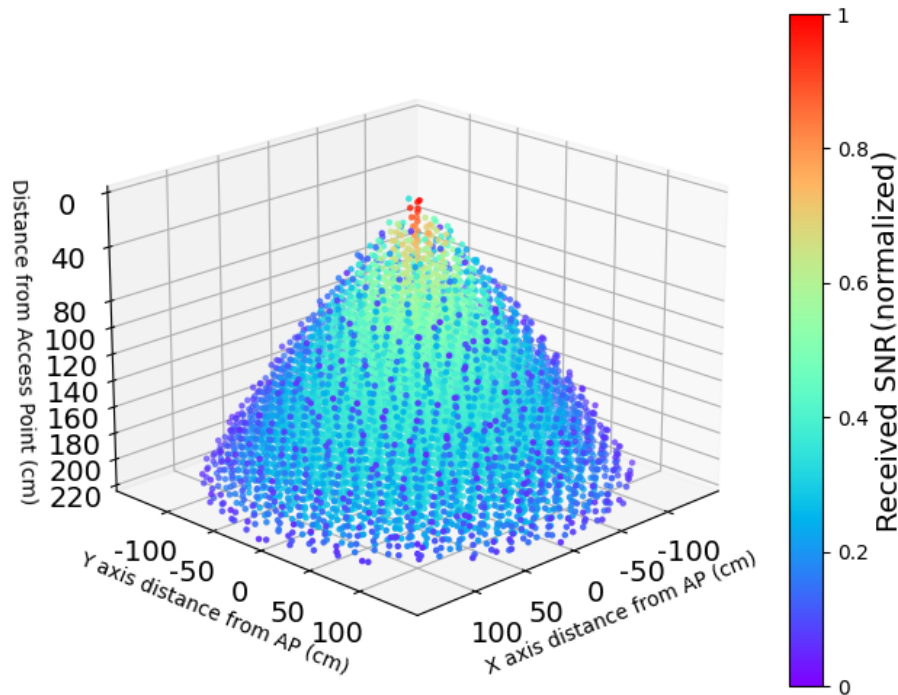


Figure 5.6: Received heatmap based on simulation experiment.

In Figure 5.6, the cone-like shape can be seen. The AP is located at the top of the cone and every point has been measured by the moving EP. It can be seen that the received SNR is lower when moving further away from the AP. Since the node also moved outside this cone area, some measurements might be expected there as well. Due to the nature of the LiFi transmitter, the message does not propagate outside the cone shape. The simulated transmission area resembles the theoretic radiation pattern of the LiFi message and can thus be compared to the real-world Trulifi system experiment results.

### Real-World vs. Simulated transmissions

The Trulifi simulation and real-world Trulifi system can be compared on two aspects. The first is the overall coverage area, while the second is the difference in received power at each measured point. When comparing the two implementations, there is one difference which can be seen from the individual 3D plots (Figure 5.5 and Figure 5.6) already. The simulated version does not take the saturation into account, so while the real-world system loses connection due to being too close, the simulated version does not. The 3D plots are not suited for easy coverage comparison. To enable a better comparison, a two-dimensional plot is created from the real-world Trulifi system measurements, see



**Figure 5.7.** The maximum coverage of the simulation is plotted as two green lines. From this plot, it can be seen that the coverage of the Trulifi simulation is almost identical to the real-world Trulifi system.

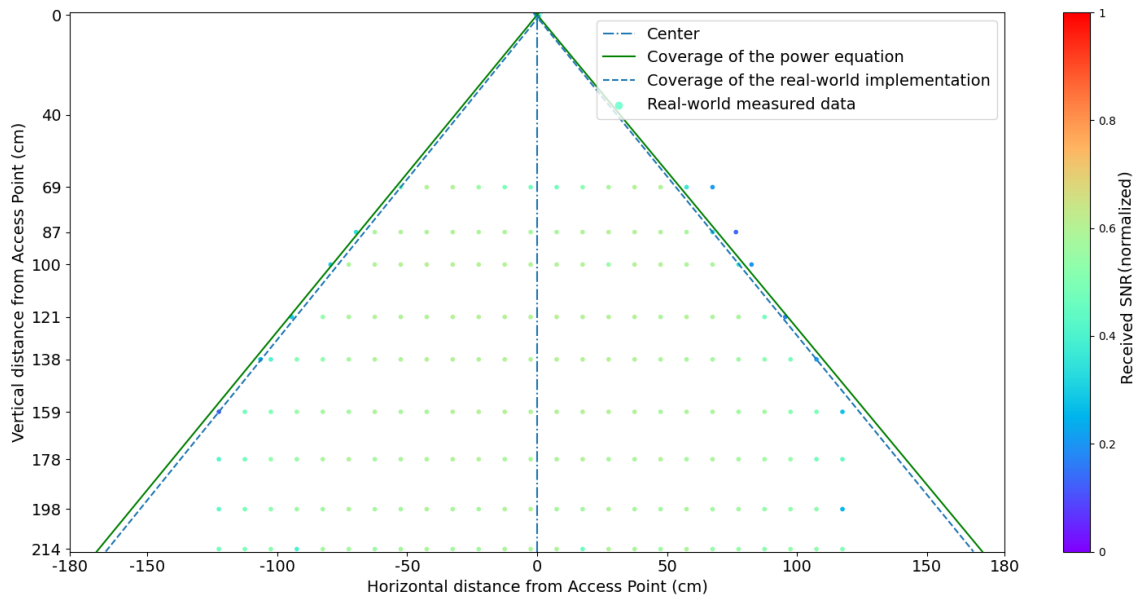


Figure 5.7: Two-dimensional heatmap of beam scanner with coverage area of the simulation.

To compare the difference in received power of the two implementations, two additional heatmaps are created. This time, the coordinates at which measurements were taken in the real-world system are used as an input for the power equation, this results in [Figure 5.8b](#). [Figure 5.8c](#) shows the difference between the calculated and measured values. From this plot, it can be seen that the power equation is distinctive from the measured values. The difference between the two is larger the closer one moves towards the center and top of the cone. From [Figure 5.8b](#), one might notice a correlation between the power equation and the distance from the middle of the cone, while this is not seen in [Figure 5.8a](#). This property comes from the fact that the power equation does not include a threshold for the maximum received power, while the real-world Trulifi system does have this limitation due to the photodetector saturation.

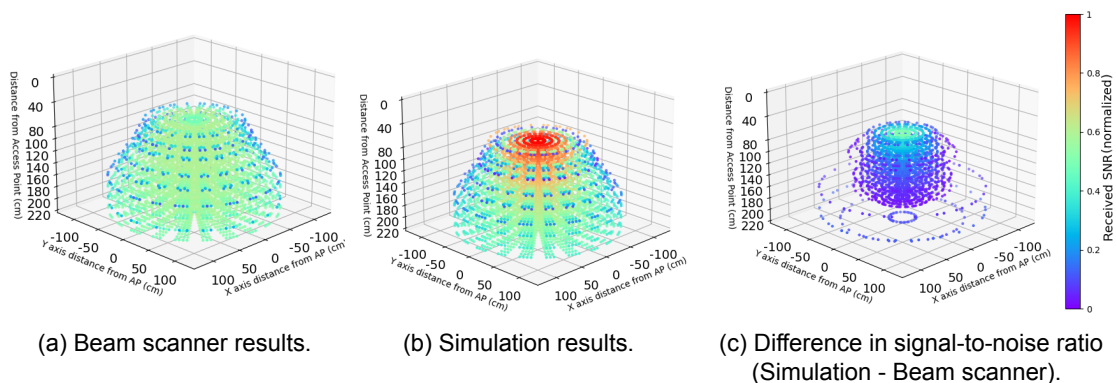


Figure 5.8: 3D comparison of received signal-to-noise ratio of real-world and simulated setup.

Based on the results presented in [Figure 5.7](#) and [Figure 5.8](#), it can be seen that the implemented radio model resembles the real-world setup. One improvement which can

be made to improve the radio model is the introduction of a minimum distance between the AP and EPs. This would allow the top of the cone to be cut off in the simulation as well and would result in Figure 5.8b. Due to the communication loss in the real-world Trulifi system implementation when the distance is smaller than 60 cm, the first improvement has to be implemented. Not implementing this feature would allow the simulation to create a connection between devices which would not be possible in a real-world setup.

Based on the above results and their discussion, the behaviour of the radio model can be considered almost identical to the real-world setup. To make the simulated version even more representative, a small improvement, a minimum distance between devices, is added. As the behaviour is almost identical in terms of light propagation and coverage area, requirement R5a and requirement R5b are met.

### 5.3. Trulifi simulation model analysis

This section will provide insights in the experimental setups used to validate the behaviour of the Trulifi simulation. For this validation, three experiments are conducted. The first experiment is performed on a real-world Trulifi system, while the other two are performed on simulated systems. The first simulated experiment is performed on the stress-test model, while the second simulation experiment is performed on the designed Trulifi simulation. The number of connections and schedule over time will be monitored during the experiments to allow for comparable measurements. The reason why these properties are monitored is explained in the coming paragraphs.

#### Connections over time

The Trulifi simulation should represent the real-world Trulifi system behaviour. In order to have a realistic behaviour, the connections between the EPs and APs should behave similarly. When switching on the real-world Trulifi system, there are no EPs connected. Once an EP can receive messages from an AP it will try to connect to that AP. The amount of time between starting and connecting will be compared to see if the start-up behaviour is similar. The number of connections over time will also be monitored, to see how the EPs are handed over between the APs during the experiment. Once the results from all three setups is known, they can be compared to each other.

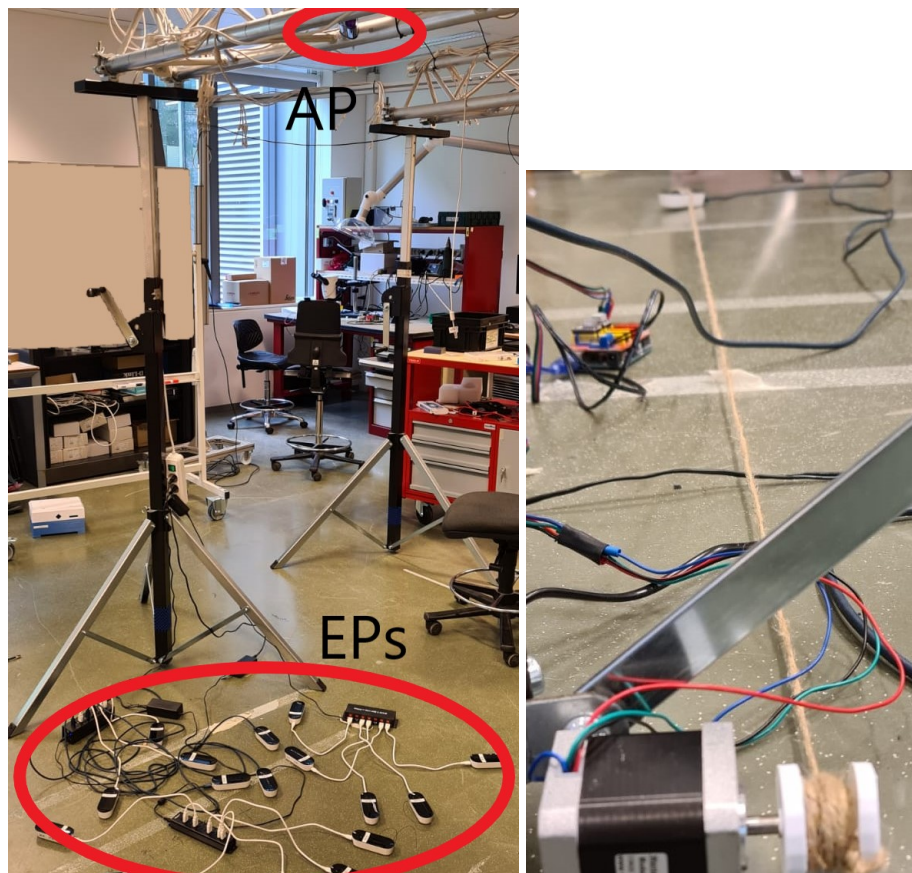
#### Schedule

While the system is running, the APs report downlink and pass on uplink interference reports, from the EPs, to the LC. The LC will create a schedule for the system in which this interference is removed. The Trulifi simulation model is designed to test the scheduler with interference. The LC should therefore create the same schedules in the real system implementation as in the Trulifi simulation model. As the stress-test model is based on probabilities, it is not possible to recreate a similar system behaviour in real world. Since the stress-test model was designed to validate the limits of the scheduling algorithm and not to represent real-world behaviour. Therefore, both the stress-test model and real-world Trulifi setup will only be compared to the Trulifi simulation model and not with each other. Because EPs will be moving in the Trulifi simulation model, it is expected that the schedule of the Trulifi simulation model will be different over the runs, while the outcome of the stress-test model will always be the same.

#### 5.3.1. Real-world Trulifi system implementation

The real-world Trulifi system consists of a variable number of APs and EPs. The maximum number of available APs is four and the maximum number of available EPs is 16. The APs are attached to a beam in a grid at a distance of 1.5 m from each other. The 16

EPs can be scattered around in an area of 3 m by 3 m underneath the grid. The height difference between the EPs and APs has been set to 2.1 m. Since the number of devices is limited, a test with four APs and the maximum number of connecting EPs, which is 16 per AP, cannot be conducted. The real-world EPs cannot move on their own, which is a limitation if an experiment needs to be performed with moving EPs. To overcome this issue, a simple device has been created to move the EP with a speed of 0.1 m/s and can be seen in Figure 5.9b. This device consists of an Arduino UNO and a motor controller which can handle up to four stepper motors. Each stepper motor is able to move one EP. Due to the limits in the number of devices and the movement, three sub-experiments are designed. The first will use one AP and 16 EPs. The second sub-experiment will use a small configuration with two APs and one EP. The final sub-experiment consists of 16 EPs and four APs, to represent a realistic system. An impression of the overall setup can be seen in Figure 5.9a.



(a) One access point and 16 end points.

(b) Device created to enable simple mobility of end points.

Figure 5.9: Real-world setup for evaluating Trulifi simulation model.

### One access point and 16 end points

The first sub-experiment is used to gather data on how fast the maximum number of EPs connect to one AP. Before beginning the experiment, the LC logging is cleared, and the EPs are placed within the coverage area of the AP. After every EP is placed, the LC is switched on to monitor the number of EPs in the system. Finally, the APs are switched on, allowing the EPs to start registering. The exact timing of the EPs connecting can be extracted by using the logging functionality of the LC. As there is only one AP, the

schedule does not need to be monitored, since the schedule will only change if there are multiple APs and interference is reported by either the APs or the EPs. This setup can be seen in Figure 5.9a.

#### Two access points and one end point

The second sub-experiment which will be conducted consists of two APs and one moving EP. The EP will be placed inside the coverage area of one of the two APs and the device, as seen in Figure 5.9b, will move the EP towards the coverage area of the other AP through their overlapping area. By moving the EP, the APs should report downlink interference and forward the uplink interference from the EP to the LC. The schedule should change according to this interference information. The detailed schedule information can be extracted from the logging functionality of the LC.

#### Four access points and 16 end points

The final sub-experiment will consist of the maximum number of devices available. This experiment is used to enable larger scale comparison with the schedule of the Trulifi simulation. To enable this, the 16 EPs will be distributed randomly within the coverage area of the four APs. As been stated before in subsection 5.3.1, the simple device enables movement of four EPs. This experiment will show how the APs can access the medium based on four moving EPs while 12 other EPs remain stationary. Once again, the scheduling information can be extracted from the logging functionality of the LC.

### 5.3.2. Stress-test simulation implementation

The stress-test simulation model can be altered by one parameter; the number of APs. As described in subsection 1.2.2, the EPs are connecting and disconnecting based on a certain probability. The uplink and downlink interference are based on probabilities as well. Since the number of EPs cannot be changed and the EPs have a static nature, there are two sub-experiments which will be conducted for this simulation implementation. The first sub-experiment makes use of one AP with 16 EPs. The second sub-experiment is used as a comparison for larger scale and makes use of four APs and 64 EPs.

#### One access point and 16 end points

The first sub-experiment follows the same setup as described in subsection 5.3.1, but the AP and EPs will be simulated. There should be no devices connected to the LC at the beginning of the simulation. Once the simulation is started and the AP is connected, the EPs will start to register based on the probabilities defined in subsection 1.2.2. The LC logging functionality is used to extract the exact timing of the EPs connecting.

#### Four access point and 64 end points

The second sub-experiment will aim to validate the scheduling. In the real-world experiments, the maximum number of EPs available was only 16 for the whole setup. Since the devices are simulated in the stress-test model, the maximum number of EPs can be connected to each AP. The interference should change based on the probabilities and the schedule should be adjusted accordingly. The connected number of EPs over time is also monitored. After conducting this experiment, the allowed access time per MAP-cycle and the number of connected EPs can be used for comparison.

### 5.3.3. Trulifi simulation implementation

The Trulifi simulation can be altered on more parameters than the stress-test simulation model. The number of EPs and APs can be altered. The mobility model, speed, height

and location of both of these devices can also be changed depending on the particular needs. This enables similar sub-experiment designs as described in [subsection 5.3.1](#) and [subsection 5.3.2](#). A total of five sub-experiments are designed for the Trulifi simulation. The first uses one AP and 16 EPs. The second uses two APs and one moving EP. Four APs and 16 EPs are used in the third experiment. The fourth is used to compare to the stress-test simulation and uses four APs and 64 EPs. The final sub-experiment is designed to validate an even larger-scale configuration of the system and uses 16 APs and 64 EPs.

### One access point and 16 end points

This sub-experiment is similar to the ones described for the Real-world Trulifi system and stress-test implementation. 16 EPs are distributed over the coverage area of one AP and the simulation will be started. The LC logging functionality is used to gather the exact timings of the EPs connecting. A 3D visualization of this setup is given in [Figure 5.10](#).

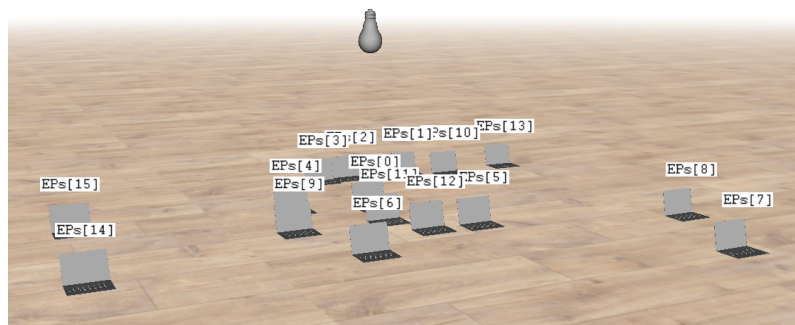


Figure 5.10: Simulation setup with one access point and 16 end points.

### Two access points and one end point

This sub-experiment will also be run similarly to the one described in [subsection 5.3.1](#). One EP will be placed in the coverage area of one AP and will be configured to travel towards the coverage area of the second AP. The uplink and downlink interference should be reported to the LC through the AP. Once again, the scheduling information can be extracted through the logging functionality of the LC.

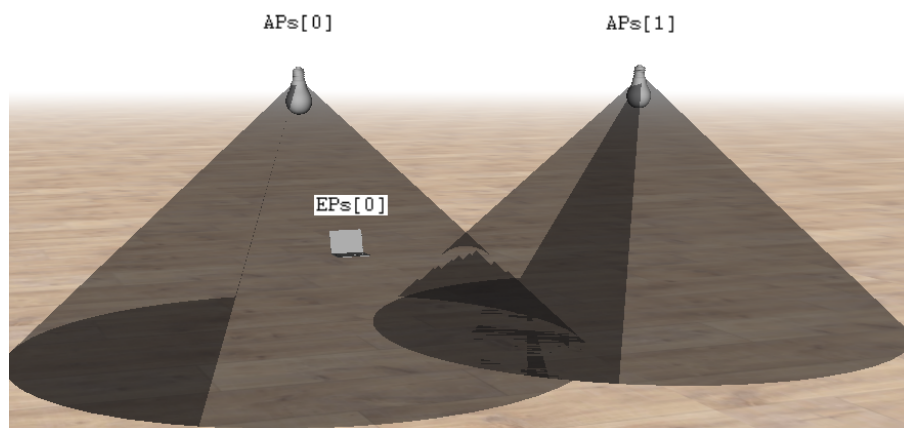


Figure 5.11: Simulation setup with two access points and one end point, grey cones indicate coverage area.

### Four access points and 16 end points

For this experiment, the real-world setup will be replicated. As four EPs were moved around in the real-world experiment, this will be done in the simulation as well. Each moving EP will move according to a different way point script. With these way points the moving behaviour of the simulated EPs are identical to the moving behaviour of the EPs in the real-world setup. Once again, the scheduling information can be extracted through the logging functionality of the LC.

### Four access points and 64 end points

The fourth sub-experiment will consist of four APs and 64 EPs. For this experiment, 16 EPs are distributed underneath each AP. The interference is expected not to change in the Trulifi simulation model once the EPs have been properly connected, while the stress-test model does change its interference due to the probabilities.

### 16 access points and 64 end points

The final sub-experiment will be used to validate if the Trulifi simulation can be used for large scale validation of the real-world system. 64 EPs will be distributed over 16 APs by placing them randomly in their coverage areas. This experiment is only used to see if the all simulated devices can connect to the LC and to see how fast this happens. Unfortunately, simulation of a larger setup was not possible since Signify's simulation infrastructure was not available at the time. An overview of the simulation setup can be found in Figure 5.12.

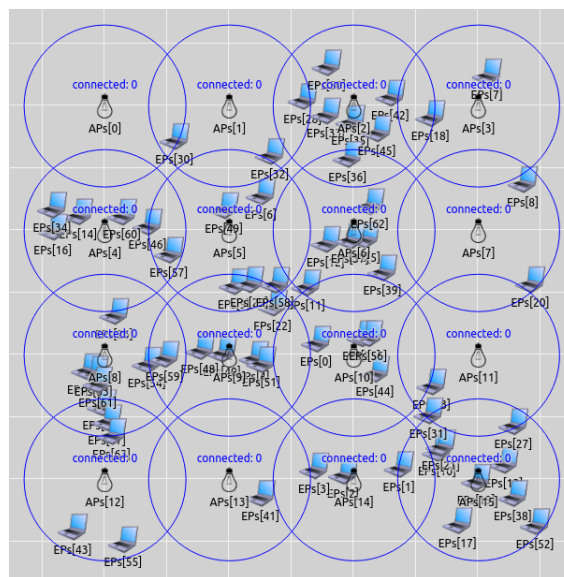


Figure 5.12: Simulation setup with 16 access points and 64 end points.

## 5.4. Trulifi simulation model validation

This section will provide the results of the experiments described in section 5.3. First, the time it takes for the maximum number of EPs to connect to an AP will be discussed. After this, the scheduling of the experimental setups will be compared.

### Connections over time

For the simulation to resemble the real-world Trulifi system behaviour, the speed at which the EPs connect to an AP in the simulation should be similar to the real-world connection

speed. To validate this, 16 EPs were connected to one AP and the LC logging output was monitored. From these logs, the connected number of EPs over time were extracted. The results can be found in Figure 5.13.

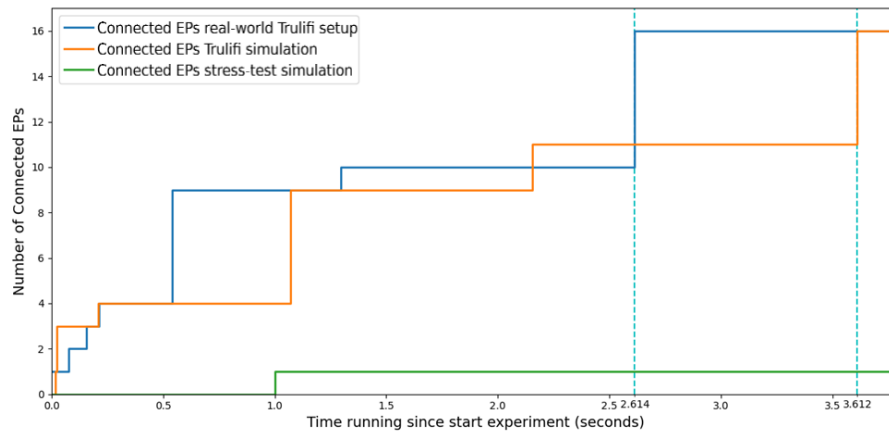


Figure 5.13: Connected number of end points of one access point at startup.

From Figure 5.13, it can be seen that the connection time of the Trulifi simulation is comparable to the real-world Trulifi setup. When comparing the stress-test model to the Trulifi simulation, it can be seen that the stress-test model EPs connect much slower, not even reaching two connections within the time that the Trulifi simulation is all connected. The second row of Table 5.1 shows the exact time it took for 16 EPs to connect to one AP for all three experimental setups. From this, it can be seen that there is a difference of 1.02 s between the Trulifi simulation model and the real-world Trulifi system, while the stress-test model takes 23.52 minutes longer to connect all EPs. When examining the difference between the Trulifi simulation and the real-world setup, the difference of around one second can be explained by looking at the back-off policy as described in section 4.2. This policy states that a connection attempt should be re-initiated one second after a failed attempt. This duration explains the longer periods in which no new EP connects to the AP which can be seen in both the Trulifi simulation model and the real-world implementation. The difference between the Trulifi simulation and the stress-test simulation can be explained by the probabilities which the stress-test simulation uses. Since the connections are not based on actual messages, but on these probabilities, the EPs are not connected as soon as the AP is "on". This results in the EPs connecting over a larger time frame than in the Trulifi simulation model.

Table 5.1: Time it takes for end points to connect to access points.

EPs	APs	Real-world Trulifi setup	Stress-test simulation	Trulifi simulation
16	1	2.614 s	23.56 min	3.612 s
16	4	1.033 s	-	4.096 s
64	4	-	23.56 min	33.31 s
64	16	-	-	6.73 s

Since the real-world Trulifi system setup had a limiting number of EP to connect, a four AP and 16 EP configuration is used to validate the Trulifi simulation against the largest possible real-world Trulifi setup configuration. Figure 5.14 shows the connected EPs over time, distributed over four APs in the real-world setup. The time during which four EPs were moved is shown in the figure with two vertical dashed lines. From the black arrows in

Table 5.2: Time it takes for the Trulifi simulation connections to complete.

	AP 1	AP 2	AP 3	AP 4
Connected to LC	1 s	1 s	1 s	0 s
EPs connected	2.106 s	3.096 s	3.096 s	1.153 s

Figure 5.14, it can be seen that the EPs can lose connection but can quickly reconnect. If an EP fails to receive confirmation messages or fails to re-register itself to the AP, the EP can lose this connection. If it is disconnected and receives a message from the AP again, it will try to reconnect. In Figure 5.14, the movement of the EPs between the APs can also be seen. In the real-world implementation, all 16 EPs are connected within 1.033 s which is even faster than connecting 16 EPs to one AP. When there are multiple EPs trying to connect to one AP, some registration messages might arrive at the AP at the same time. This will result in a collision of messages and the AP answering only one of these messages or none at all, which leads to a longer connection time. When the 16 EPs are distributed over four APs, this collision of registration messages occurs less.

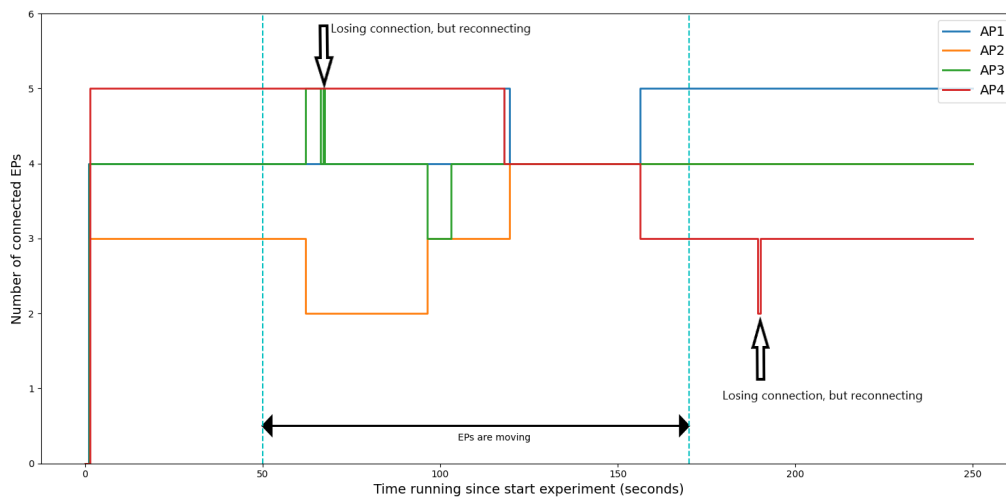


Figure 5.14: 16 end points connected to four access points in real-world Trulifi setup.

Figure 5.15 shows the results gathered from the Trulifi simulation model. In the Trulifi simulation, the time it takes for an AP to connect to the LC varies, see the second row of Table 5.2. Once an AP connects to the LC, simulated EPs receive messages from the AP and will try to connect. This results in topology updates to the LC, which results in delayed connection of the other APs due to the LC processing the messages.

Once an AP is connected, the EPs within its coverage area can start to connect to the AP. From the third row of Table 5.2, it can be seen that the time it takes for the EPs to connect to the AP varies between 1.153 s and 3.096 s. When adding this time to the connecting time of the AP, the total time it takes for all EPs to connect is 4.096 s. This is around four times longer than the connection in the real-world setup. The Trulifi simulation implements a random registration time, which causes collisions multiple EPs try to register at the same time. These collisions can happen when the EPs try to re-send the message after the back-off period of 1 s as well, which results in longer connection periods. Furthermore, the distribution of the EPs might be different causing the EPs to connect quicker in real-world than in simulation. Since the EPs can only access the channel during their own reserved time period, not the registration period, there will not be any



collisions in the simulation. As the simulation environment is *perfect*, the EPs do not lose connection while standing still, see Figure 5.15.

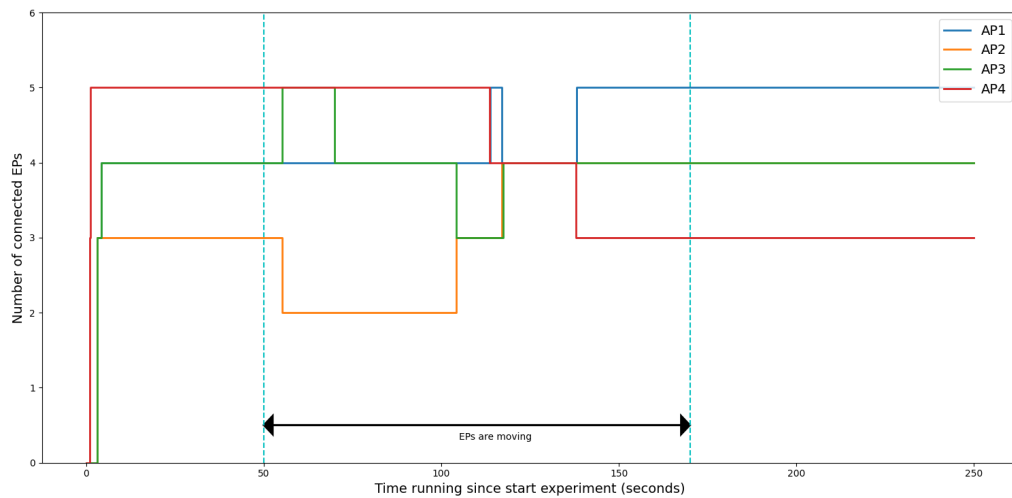


Figure 5.15: 16 end points connected to four access points in the Trulifi simulation.

When comparing Figure 5.14 and Figure 5.15 there are two main differences which can be recognized. The first difference is that the EPs in the simulation do not lose connection in the same way as the EPs in the real-world setup. As the simulation is *perfect*, EPs do not fail to re-register or miss any messages. This results in them not disconnecting as in the real-world setup. Furthermore, the handover between APs is done immediately in the Trulifi simulation, while the handover in the real-world setup can take more time. The second difference which can be seen is that the timings in the simulation are slightly different from the real-world experiment. While the device (Figure 5.9b) was created to limit the differences between the two, the rope used to pull the EPs sometimes got stuck, resulting in a slower speed overall. The experiment was run multiple times, but it always happened. Another observation was that due to the weight of the EP and its rubber bottom, the stepper motor was not always able to pull the EP in a straight line.

The connection speed of the Trulifi simulation on an even larger scale can only be compared with the stress-test model. Figure 5.16 shows the stress-test model connection behaviour. Since the probabilities for connecting/disconnecting depend on the same random generation seed, the first thing that can be seen is that the different APs behave exactly the same. As this behaviour is the same, it is no surprise that the connection speed with 64 EPs over four APs is exactly the same as with 16 EPs and one AP. Once the number of connected EPs is above 12 per AP, it will not drop below this value due to these probabilities either.

Figure 5.17 shows the results gathered from the large scale Trulifi simulation. OMNEST allows for simulations to be run from either the command line or the IDE. If this large-scale test is performed through the IDE, the visualization takes lots of computational power. The experiment is run through the IDE and through the command line to see the difference in computational power. It was found that using the IDE uses 100% of the available CPU power, while using the command line only used 60%. Therefore, the larger scale Trulifi simulations will be run through the command line. During such a large scale test, the EPs can sometimes not connect to an AP at all. This occurs when the EP is within the overlap area and tries to connect to an AP which is already full. The AP will inform the EP that it is full and the EP will try to connect to the other AP. If it then receives another message from

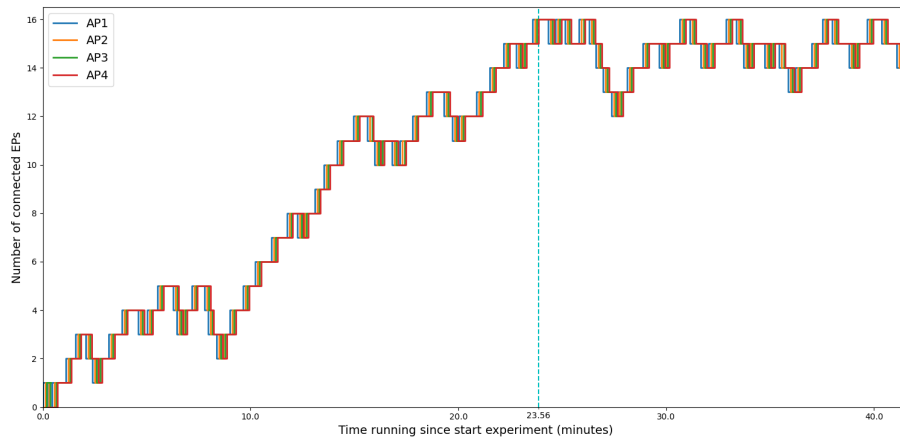


Figure 5.16: 64 end points connected to four access points in the stress-test simulation.

the full AP with a higher power than the other AP, it will try to connect to the full AP again. This results in an EP which will not connect and is something to solve as future work. For the current Trulifi simulation, this can be solved in two ways; not placing the EPs uniformly or making non-connecting EPs move if a connection seems impossible. Since this would also be a real-world behaviour of a user facing a connection issue, e.g. if the connection can not be established the user usually places the dongle differently. After multiple experiments were performed, it was found that it takes on average 33.31 s for all EPs to connect. If this time has passed, and not all EPs are connected, the non-connecting EPs will not connect any more. From Figure 5.17, it becomes clear that if simulated EPs lose connection, they are able to re-connect. As the EPs are placed uniformly and are able to move around, the distribution of EPs over the APs is different for many runs. This satisfies requirement R4.

The largest scale setup which is tested on the Trulifi simulation consists of 16 APs and 64 EPs. The experiments showed a CPU usage of 72% and all 64 EPs are connected in 6.73 s. The time it takes for all 64 EPs to connect is five times as short as with the four APs and 64 EPs configuration due to the distribution over more APs. This large scale experiment shows that a larger scale validation of the Trulifi system is possible with the Trulifi simulation.

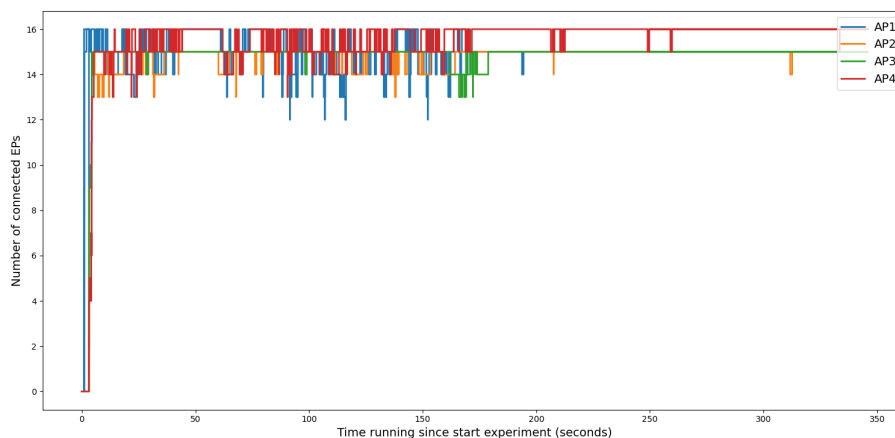


Figure 5.17: 64 end points connected to four access points in the Trulifi simulation.

### Evaluation of the scheduling

One of the main reasons for designing the new Trulifi simulation model is that it can be used to evaluate the system behaviour in larger deployments, including interference management and handovers. To validate if the simulation can be used for this, experiments are performed and the scheduling of the LC is analysed. The first experiment is used to compare the basic scheduling of the LC, with one EP. Figure 5.18 shows the schedule that the LC computes during movement of one EP between two APs in the real-world setup. It can be seen that the APs are allowed to access the channel simultaneously until interference is reported (0-6 s). Once interference is reported, the controller creates a schedule where the APs are prohibited to access the channel at the same time.

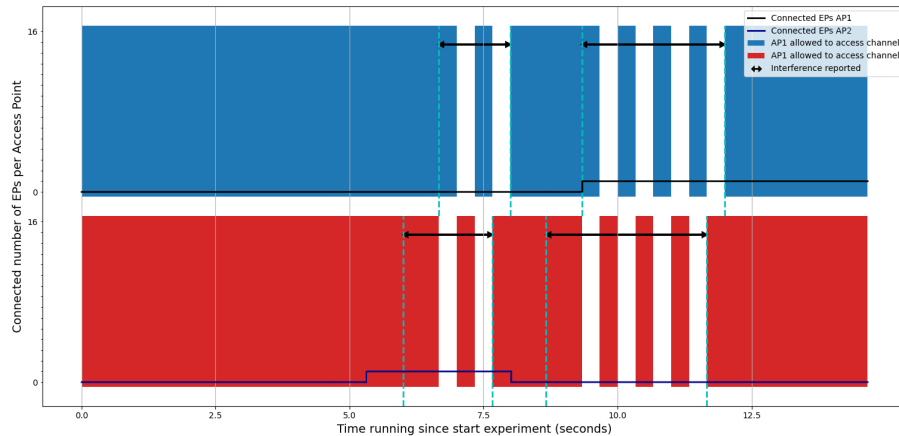


Figure 5.18: One end point moving between two access points in a real-world Trulifi setup.

Figure 5.19 shows the results of the same experiment, but conducted with the Trulifi simulation model. It can be seen that the handover of the EP between the two APs is instant in the simulation, while this takes some time in the real-world setup. By comparing Figure 5.17 with Figure 5.18, the similarity between the two schedules can be seen. In both experiments the interference was reported correctly which resulted in the same schedule created by the LC. The timing might be a bit different, however, this can be neglected. This difference could be simply due to the effects from the EPs being pulled by the stepper motor.

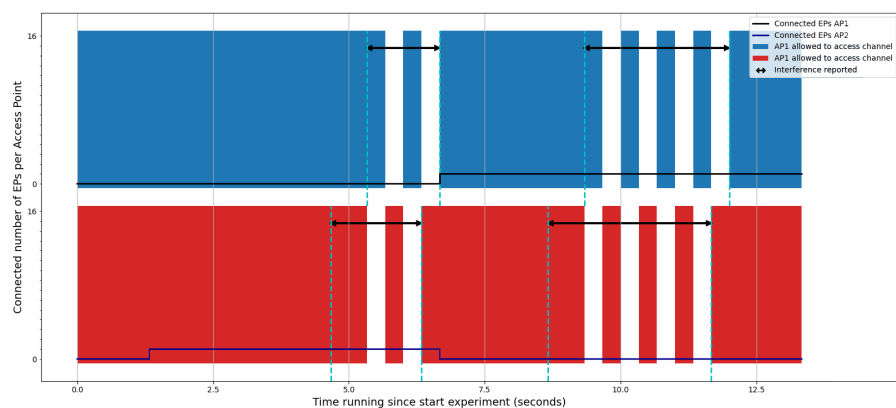


Figure 5.19: One end point moving between two access points in the Trulifi simulation.

As the smallest configuration is validated, an experiment of a larger scale can be performed. For the configuration with 16 EPs and four APs, plotting the scheduling as

shown in [Figure 5.18](#) gets very cluttered. It is more convenient to show the schedule and connected EPs in the same figure for only two AP, but in order to make sure the plots are still readable with a larger amount of APs, the allowed medium access time per MAP-cycle is shown in [Figure 5.20](#). Due to confidentiality reasons the actual timing values can not be shown. Therefore the graphs contain min-max normalised values.

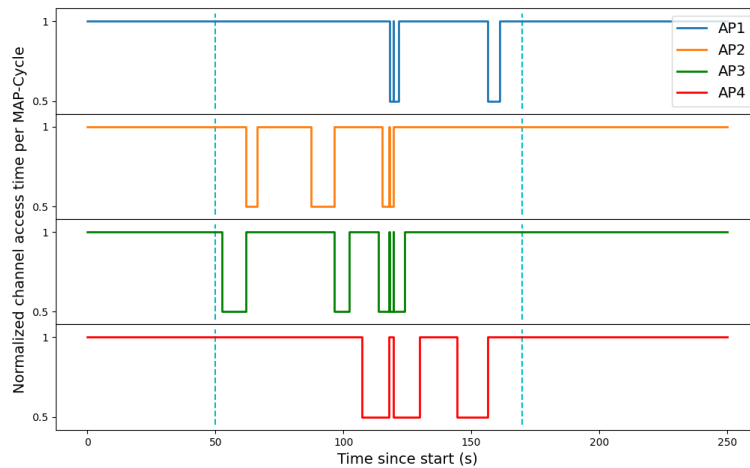


Figure 5.20: Scheduling of four access points and 16 end points of a real-world Trulifi setup.

To understand the data presented in [Figure 5.20](#), the data is divided into four sub-figures of equal time in [Figure 5.21](#). When the EPs are not moving (in [Figure 5.21a](#) and in [Figure 5.21d](#) from 170s), no interference is reported, which means that all APs are allowed to access the medium during the whole MAP-cycle. At the start of [Figure 5.21b](#)(from 50s), EPs start moving around, which results in interference being reported. This results in less time each AP can access the channel. In [Figure 5.21c](#), three EPs are moving around and one of these is crossing the coverage area of three APs during the movement, resulting into a schedule where these three APs have to limit their channel access. The AP for which no interference is reported can access the channel freely.

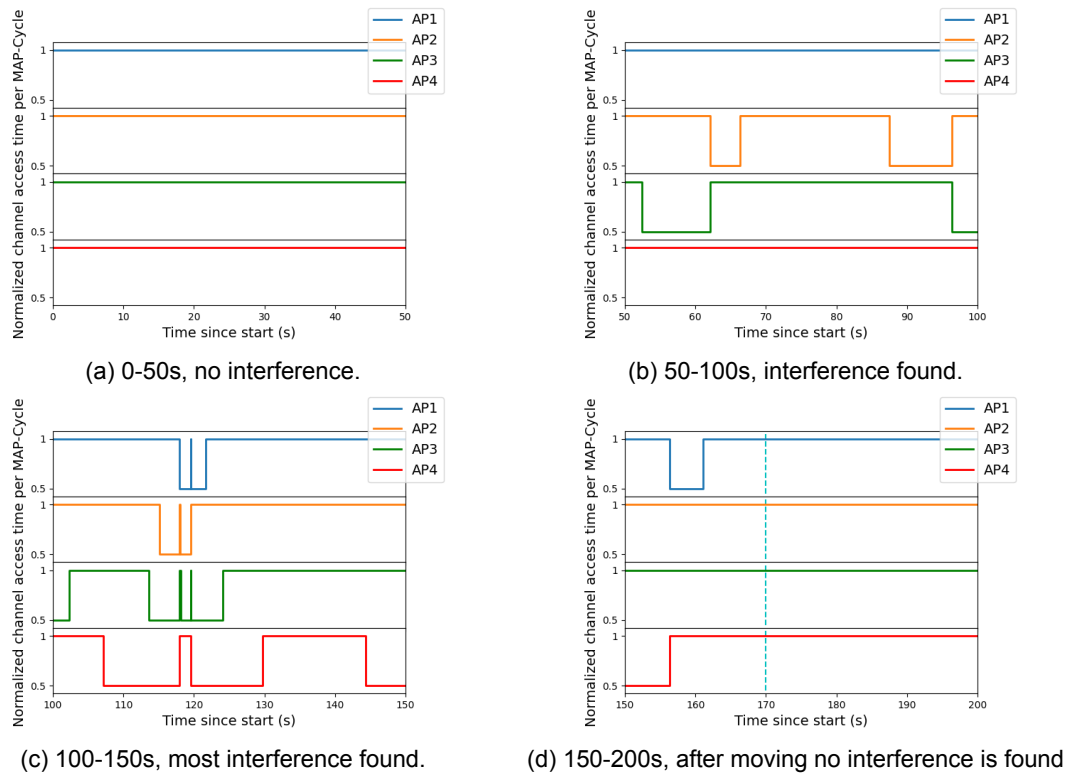


Figure 5.21: Scheduling of four access points and 16 end points of a real-world setup divided over four figures.

Figure 5.22 shows the results from the same experiment performed on the Trulifi simulation model. The results from the simulation show that the APs are limited in their channel access for shorter time periods than the APs in the real-world setup. This difference can be explained by the difference in the EP movement speed. The simulation has been run with lower speeds to try to represent the real-world setup even more, but due to the inconsistency in the movement of the real-world devices, the timings could not be identical. When looking past this difference in timings, the similarities between the simulation and real-world setup can be seen. The LC is able to configure a schedule for the APs in the simulation in a similar way as for the real-world setup. Due to the observed similarities, it can be concluded that the simulation can be used for validating larger scale Trulifi systems

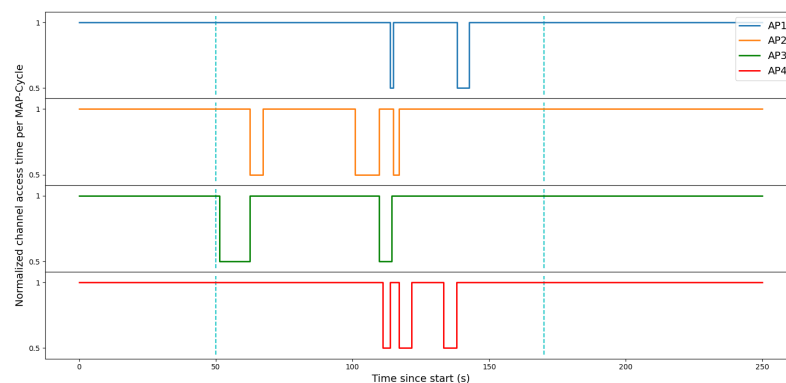


Figure 5.22: Scheduling of four access points and 16 end points of the Trulifi simulation.

As the small and medium scale configurations have proven to be similar, the maximum configuration was tested on the stress-test and Trulifi simulation. Figure 5.23 shows the results gathered from the stress-test simulation model. In the stress-test simulation, APs are only allowed to access the channel all the time during the first 20 s. From the moment more EPs are connected, more interference is reported and the APs are only allowed to access the channel for 20 ms or even only 10 ms. It can be seen that sometimes one of the APs is allowed to access the channel either the whole or not even half of MAP-cycle. As the stress-test simulation always ends up in the same state, the received scheduling always looks like the one shown in Figure 5.23.

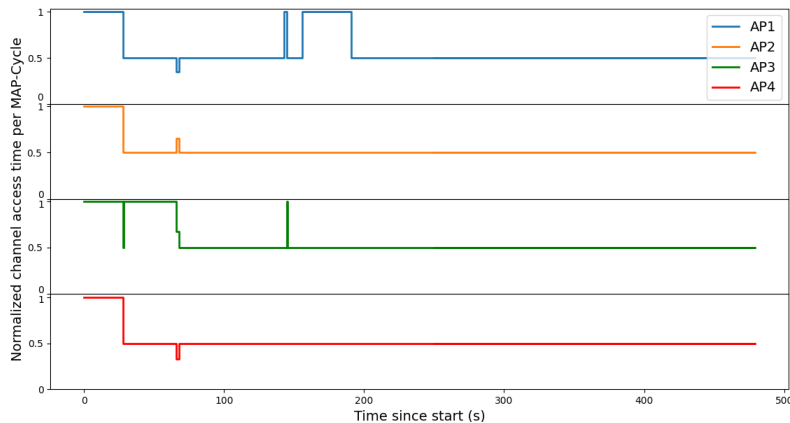


Figure 5.23: Schedule changes of four access points and 64 end points of the stress-test simulation.

With 16 EPs distributed uniformly over a circle underneath with a radius of 1.3 m underneath each AP, there is bound to be a lot of interference. EPs are handed over between APs, which can be seen in the received schedule in Figure 5.24. Once the EPs move out of this interference zone, the APs are allowed to access the channel during the whole MAP-cycle again. The random distribution of the EPs results in a different schedule over time each simulation run. The LC is able to create a schedule for the Trulifi simulation with the maximum number of connected EPs per AP. As a larger scale configuration, with 64 EPs and 16 APs, has also been validated based on the number of connections to the LC, it can be concluded that the Trulifi simulation satisfies requirement R2.

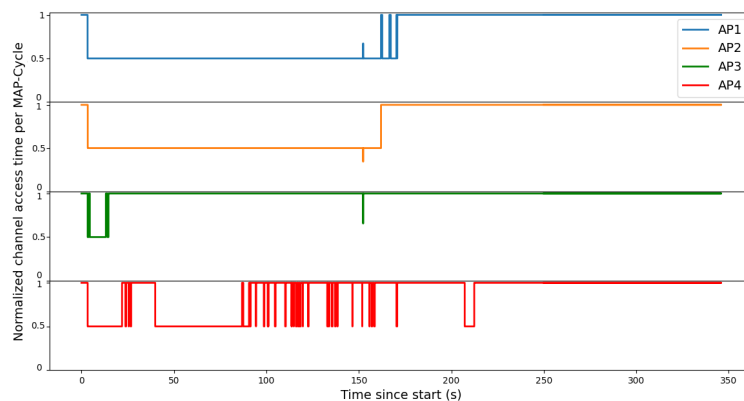


Figure 5.24: Schedule changes of four access points and 64 end points of the Trulifi simulation.

## 5.5. Conclusion

The experiments discussed in this chapter have been designed and carried out to validate if the simulation could satisfy the requirements established in [section 1.3](#). The following list shows the requirements once more:

- R1** The correct management messages (from the ITU G.9961 standard [27]) have to be used for communication.
- R2** The simulation should enable a large scale Trulifi system validation.
- R3** Mobility of the EPs has to be supported.
- R4** The system should not converge to the same state each run.
- R5** The model should be as realistic as possible in terms of the following aspects:
  - (a) Light propagation
  - (b) Coverage area
  - (c) Connectivity behaviour
  - (d) Scheduling
- R6** The model should be as computationally inexpensive as possible.

Requirement **R1** has been satisfied by implementing the ITU G.9961 messages for communication between the AP, EP and LC. The results of the experiment with 16 APs and 64 EPs, provided in [section 5.4](#), showed that the Trulifi simulation can successfully be used to replicate a large-scale real-world Trulifi setup. Therefore, requirement **R2** has been satisfied. EP mobility, requirement **R3**, has been implemented in the design and has been used to replicate a real-world setup in multiple experiments as described in [section 5.4](#). Since the EPs are distributed randomly over the APs, the simulation ends up with a different EP distribution each run. Even though a setup with the maximum number of EPs per AP always ends up in the same state eventually, the connections are not made similarly and thus requirement **R4** is also satisfied. Requirement **R5** states that the simulation model should be as realistic as possible on four different aspects; light propagation, coverage area, connectivity behaviour and scheduling. To validate the light propagation and coverage area, experiments were performed on a real-world Trulifi system with the beam scanner. By measuring the received power at multiple distances and replicating this in the Trulifi simulation it was found that coverage area and light propagation of the simulation is almost identical to the real-world system. The connectivity and scheduling behaviour of the Trulifi simulation has been validated by comparing it to the behaviour of a real-world system and the stress-test model. It was found that there are some differences based on connection timing between the Trulifi simulation and real-world system, but the LC manages to create similar schedules for both implementations. As the behaviour of both implementations was similar, requirement **R5** has been satisfied as well. Since all requirements regarding the realism have been satisfied, the Trulifi simulation has proven to be a realistic representation of a Trulifi system. Finally, requirement **R6** states that the computational power should be kept as low as possible. It was found that the Trulifi simulation itself uses less power when it is run through the command line instead of through the IDE. As the Trulifi simulation is a simplified version of existing simulations, it is speculated that the Trulifi simulation is computationally less expensive than other existing simulations.

# 6

## Conclusion and Future work

### 6.1. Conclusion

Throughout this thesis, three sub-questions have been answered in order to properly formulate an answer to the main research question, **Can Signify's Trulifi system be modelled in a realistic simulation?**. As Signify's current simulation model is designed to stress-test the LC it can not be used to properly validate a real-world setup. E.g. if a customer provides an application and Signify wants to validate if this application is possible with the system.

#### Can the simulation model be created with existing wireless communication models?

The first sub-question has been answered based on the literature study. As existing simulation models use modulation techniques and physical properties to predict system behaviour, the computational power will increase when experimenting with larger scale configurations. Since the computational power has to be kept as low as possible and the behaviour of the system is already known, these existing models will not be used.

#### How can a realistic LiFi simulation model be achieved?

The second sub-question has been answered during the design process of the interference simulation. Multiple simplifications were used to create a computational inexpensive simulation, but these simplifications have been validated to not influence the level of realism. By designing the LiFiMedium module and by implementing the ITU G.9961 standard data link layers, the simulation approaches the real-world system communication behaviour.

#### How does the Trulifi simulation model compare to a real Trulifi system?

The final sub-question has been used to validate the Trulifi simulation in practice. Based on the results from the experiments, the conclusion was made that the requirements describing the level of realism were all satisfied and thus the simulation has proven to have a similar behaviour as the real-world Trulifi system.

### Signify's Trulifi system **has been** captured in a realistic simulation

The Trulifi simulation has proven to have a similar behaviour to the real-world system on all validated fronts. Enriching the simulation model by replacing the simplifications with



a power equation based on measurements is the main novelty of the Trulifi simulation model. Signify can use the Trulifi simulation as a tool to validate large-scale Trulifi system configurations. Power equations can be created based on new design ideas. By replacing the current power equation with another, a completely different behaviour can be observed in the terms of signal communication and overall system behaviour. With respect to the stress-test model from which the new model is created, the new model provides a way to test scheduling scenarios which can not be deployed simply in a real-world Trulifi setup. By introducing movement and connection on control messages rather than probabilities, a more realistic model is created. This makes the simulation model a great enhancement for future research.

## 6.2. Future work

There are always improvements to be made to a product, and this section will describe the improvements which can be made to further enhance the simulation. The first improvement involves the EP mobility. The current mobility models used in the simulation are the RandomWaypoint, Turtle or Linear mobility models. However, during the research conducted in [section 2.5](#), multiple ways to model employee mobility behaviour have been found. If the EPs were to be equipped with these mobility models, more realistic movement scenarios can be tested and potential flaws in the system can be found before production. The second improvement involves realistic LoS. Currently, the signals can just travel within the coverage area without finding any blocking objects. In a real-world situation these blocking objects can still occur, a person might block the LoS or desks might be separated by large cabinets that block part of the coverage area as well. Scenarios with this kind of blockage can change the behaviour of the system, by having some EPs never connect or having no interference between the APs while they are close together. Enriching the simulation through these two improvements will benefit Signify even more.

# List of Acronyms and Abbreviations

<b>Notation</b>	<b>Description</b>	<b>Page List</b>
AP	Access Point	i, 2–5, 17–19, 21–30, 32–45, 47
EM	electromagnetic	7, 8, 10, 13
EP	End Point	i, 2–5, 11, 13, 17–19, 21–30, 32–45, 47
FoV	Field-of-view	2
GUI	graphical user interface	15
IDE	integrated development environment	15, 18, 39, 45
IOT	Internet of Things	1
LC	LiFi controller	i, 3–5, 18, 19, 21–23, 32–38, 41, 43–46
LD	laser diode	9
LED	light-emitting diode	2, 3, 9, 10, 13
LiFi	Light Fidelity	i, iii, 1–3, 5, 10, 11, 13, 18, 19, 26, 30
LoS	Line-of-sight	2, 8, 11, 19, 28, 29, 47
MAC	medium access control	2
MAP	medium access plan	21, 22, 30, 34, 42, 44
OSI	open system interconnection	11, 15
PSD	power spectral density	17
RF	radio frequency	1, 2, 7, 8, 10, 11, 13
SLAW	Selfsimilar Least-Action Walk	12, 13
SNR	Signal-to-noise ratio	8, 11, 17, 20, 23–27, 29, 30
TDMA	Time Division Multiple Access	23
VLC	Visual Light Communication	1, 2, 9
XML	Extensible Markup Language	19, 26, 27

# Bibliography

- [1] B. Jovanovic. Internet of things statistics for 2021 – taking things apart. [Online]. Available: <https://dataprot.net/statistics/iot-statistics/>
- [2] J. Scarpati. What is radio frequency (rf, rf)? [Online]. Available: <https://searchnetworking.techtarget.com/definition/radio-frequency>
- [3] E. Huibers. Lifi by signify. Signify. [Online]. Available: <https://www.youtube.com/watch?v=ZdTMr8k-Lyk&t=47s>
- [4] J. Celentano. Will we run out of spectrum? [Online]. Available: <https://insidetowers.com/cell-tower-news-will-we-run-out-of-spectrum/>
- [5] M. v. E. Z. e. K. Agentschap Telecom. Het nederlandse frequentiespectrum. [Online]. Available: <https://www.agentschaptelecom.nl/documenten/brochures/2017/juli/21/overzicht-van-het-nederlandse-frequentiespectrum>
- [6] A. Gladman and S. Lapinsky, “Wireless technology in the icu: Boon or ban?” *Critical care (London, England)*, vol. 11, p. 165, 02 2007.
- [7] S. M. Mana, P. Hellwig, J. Hilt, K. L. Bober, V. J. Hirmanova, P. Chvojka, R. Janca, and S. Zvanovec, “Lifi experiments in a hospital,” in *Optical Fiber Communication Conference*. Optical Society of America, 2020, pp. M31–2.
- [8] W. Ding, F. Yang, H. Yang, J. Wang, X. Wang, X. Zhang, and J. Song, “A hybrid power line and visible light communication system for indoor hospital applications,” *Computers in Industry*, vol. 68, 02 2015.
- [9] H. Haas, L. Yin, Y. Wang, and C. Chen, “What is lifi?” *Journal of Lightwave Technology*, vol. 34, no. 6, pp. 1533–1544, 2016.
- [10] T. Greenslade, “The photophone,” *The Physics Teacher*, vol. 17, pp. 382–383, 09 1979.
- [11] D. L. Woods, *A history of tactical communication techniques*. Orlando Division, Martin Company, Martin-Marietta Corporation, 1965.
- [12] T. T. Gray, “Transmission system,” U.S. Patent 5,635,915, May 21, 2006.
- [13] L. et al., “communication with non-flickering illumination,” U.S. Patent 7,016,115, June 3, 1997.
- [14] T. Cevik and S. YILMAZ, “An overview of visible light communication systems,” *International Journal of Computer Networks and Communications*, vol. 7, pp. 139–150, 11 2015.
- [15] N. et al., “Illuminative light communication device,” U.S. Patent 7,583,901, Sept 1, 2009.

- [16] H. Haas. Forget wifi. meet the new lifi internet. Ted Talk. [Online]. Available: [https://www.ted.com/talks/harald\\_haas\\_forget\\_wi-fi\\_meet\\_the\\_new\\_li-fi\\_internet/transcript](https://www.ted.com/talks/harald_haas_forget_wi-fi_meet_the_new_li-fi_internet/transcript)
- [17] K. Khandelwal and S. K. Jain, "A review paper on li-fi technology," in *IJIRST, National Conference on Innovations in Micro-electronics, Signal Processing and Communication Technologies (V-IMPACT-2016) February*, 2016.
- [18] H. Haas, L. Yin, C. Chen, S. Videv, D. Parol, E. Poves, H. Alshaer, and M. S. Islam, "Introduction to indoor networking concepts and challenges in lifi," *Journal of Optical Communications and Networking*, vol. 12, p. A190, 02 2020.
- [19] E. Ramadhani and G. Mahardika, "The technology of lifi: A brief introduction," *IOP Conference Series: Materials Science and Engineering*, vol. 325, p. 012013, 03 2018.
- [20] Y. S. Hussein and A. C. Annan, "Li-fi technology: High data transmission securely," *Journal of Physics: Conference Series*, vol. 1228, p. 012069, may 2019. [Online]. Available: <https://doi.org/10.1088/1742-6596/1228/1/012069>
- [21] F. Aftab, "Potentials and challenges of light fidelity based indoor communication system," *International Journal of New Computer Architectures and their Applications*, vol. 6, 10 2016.
- [22] J. Beysens, J. P. M. G. Linnartz, D. Van Wageningen, and S. Pollin, "Tdma scheduling in spatially extended lifi networks," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 1524–1538, 2020.
- [23] M. M. D. Van Wageningen, X. Chatel and A. SALAS MORENO, "Interference handling for lifi," *Optical Wireless Communication Conference*, 07 2020.
- [24] Signify. Controller unit 6800.01 and 6800.02 controller application 6800.20. [Online]. Available: <https://www.assets.signify.com/is/content/Signify/Assets/signify/global/20210118-4422-952-15619-590-trulifi-6800-controller-technical-specification.pdf>
- [25] ——. Specification sheet trulifi 6002 system with 6002.2 access points. [Online]. Available: <https://www.assets.signify.com/is/content/Signify/Assets/signify/global/20210112-4422-952-01201-590-trulifi-6002-2-system-technical-specification.pdf>
- [26] MaxLinear, "Spirit lifi technical description 2.0."
- [27] ITU, "G.9961 : Unified high-speed wireline-based home networking transceivers - data link layer specification," <https://www.itu.int/rec/T-REC-G.9961>, accessed: 2021-6-7.
- [28] The em spectrum. [Online]. Available: <http://labman.phys.utk.edu/phys222core/modules/m6/The%20EM%20spectrum.html>
- [29] C. Haslett, *Essentials of radio wave propagation*. Cambridge University Press Cambridge, 2008, vol. 91.
- [30] M. Ismail, M. Z. Shakir, K. Qaraqe, and E. Serpedin, *Radio Frequency and Visible Light Communication Internetworking*. Wiley-IEEE Press, 08 2016, pp. 119–138.

- [31] J. C. Maxwell, "Viii. a dynamical theory of the electromagnetic field," *Philosophical transactions of the Royal Society of London*, no. 155, pp. 459–512, 1865.
- [32] J. B. Andersen, "History of communications/radio wave propagation from marconi to mimo," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 6–10, 2017.
- [33] T. Rappaport, "The wireless revolution," *IEEE Communications Magazine*, vol. 29, no. 11, pp. 52–71, 1991.
- [34] C. A. Balanis and P. I. Ioannides, "Introduction to smart antennas," *Synthesis Lectures on Antennas*, vol. 2, no. 1, pp. 1–175, 2007.
- [35] D. K. Misra, *Radio-frequency and microwave communication circuits: analysis and design*. Wiley Online Library, 2004.
- [36] J. Seybold, *Introduction to RF propagation*. Hoboken, New Jersey: John Wiley & Sons, Inc, 2005.
- [37] S. Moser, F. Kargl, and A. Keller, "Interactive realistic simulation of wireless networks," in *2007 IEEE Symposium on Interactive Ray Tracing*. IEEE, 2007, pp. 161–166.
- [38] W. S. ABD ALAZIZ, "Design and development of multi-band microstrip rectangular fractal antenna for wireless applications," Ph.D. dissertation, Maharishi Markandeshwar University Mullana, 2011.
- [39] A. A. Al-Hameed, A. T. Hussein, M. T. Alresheedi, S. H. Younus, and J. M. Elmirghani, "Transmitters mapping of visible light communication system," in *2017 19th International Conference on Transparent Optical Networks (ICTON)*. IEEE, 2017, pp. 1–6.
- [40] T. Yoshizawa, *Handbook of optical metrology: principles and applications*. CRC press, 2017.
- [41] I. Moreno and C.-C. Sun, "Modeling the radiation pattern of leds," *Optics express*, vol. 16, pp. 1808–19, 03 2008.
- [42] K. Hari, "Channel models for wireless communication systems," in *Wireless Network Design*. Springer, 2011, pp. 47–64.
- [43] D. Kotz, C. Newport, R. Gray, J. Liu, Y. Yuan, and C. Elliott, "Experimental evaluation of wireless simulation assumptions," *SIMULATION*, vol. 83, 07 2004.
- [44] D. Pfefferkorn, K. Helmholdt, and H. Blume, "Performance estimation of indoor optical wireless communication systems using omnet++," in *2017 IEEE 22nd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 2017, pp. 1–5.
- [45] C. Ley-Bosch, R. Medina-Sosa, I. Alonso-González, and D. Sánchez-Rodríguez, "Implementing an ieee802.15.7 physical layer simulation model with omnet++," in *Distributed Computing and Artificial Intelligence, 12th International Conference*, S. Omatu, Q. M. Malluhi, S. R. Gonzalez, G. Bocewicz, E. Bucciarelli, G. Giulioni, and F. Iqba, Eds. Cham: Springer International Publishing, 2015, pp. 251–258.

- [46] M. A. Huysmans, D. Srinivasan, and S. E. Mathiassen, "Consistency of Sedentary Behavior Patterns among Office Workers with Long-Term Access to Sit-Stand Workstations," *Annals of Work Exposures and Health*, vol. 63, no. 5, pp. 583–591, 04 2019. [Online]. Available: <https://doi.org/10.1093/annweh/wxz022>
- [47] D. M. Hallman, S. E. Mathiassen, and H. Jahncke, "Sitting patterns after relocation to activity-based offices: A controlled study of a natural intervention," *Preventive Medicine*, vol. 111, pp. 384 – 390, 2018. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0091743517304723>
- [48] J. Yoon, M. Liu, and B. Noble, "Random waypoint considered harmful," *Proceedings - IEEE INFOCOM*, vol. 2, 03 2003.
- [49] F. Ekman, A. Keränen, J. Karvo, and J. Ott, "Working day movement model," in *Proceedings of the 1st ACM SIGMOBILE workshop on Mobility models*, 2008, pp. 33–40.
- [50] K. Lee, S. Hong, S. J. Kim, I. Rhee, and S. Chong, "Slaw: A new mobility model for human walks," in *IEEE INFOCOM 2009*, 2009, pp. 855–863.
- [51] I. McGregor, "The relationship between simulation and emulation," in *Proceedings of the Winter Simulation Conference*, vol. 2, 2002, pp. 1683–1688 vol.2.
- [52] K. Wehrle, M. Gnes, and J. Gross, *Modeling and Tools for Network Simulation*, 1st ed. Springer Publishing Company, Incorporated, 2010.
- [53] R. Beuran, *Introduction to Network Emulation*. Pan Stanford, 2013.
- [54] E. Lochin, T. Pérennou, and L. Dairaine, "When should i use network emulation?" *Annales Des Télécommunications - AdT*, vol. 67, 02 2010.
- [55] R. Beuran, "Introduction to network emulation," *Introduction to Network Emulation*, 08 2012.
- [56] S. Mehta, K. Kwak, and N. Sulatan, *Network and system simulation tools for next generation networks: a case study*. INTECH Open Access Publisher, 2010.
- [57] C. Dandelski, "Efficient control message dissemination in dense wireless lighting networks," *PhDs*, 2018. [Online]. Available: <https://sword.cit.ie/engdiss/4>
- [58] "OMNeT++ home page," <https://omnetpp.org/intro>, accessed: 2021-4-26.
- [59] A. Varga and R. Hornig, "An overview of the omnet++ simulation environment," in *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, 2008, pp. 1–10.
- [60] "OMNeT++ simulation manual," <https://doc.omnetpp.org/omnetpp/manual>, accessed: 2021-4-26.
- [61] T. Issariyakul and E. Hossain, "Introduction to network simulator 2 (ns2)," in *Introduction to network simulator NS2*. Springer, 2009, pp. 1–18.
- [62] Riverbed modeler. [Online]. Available: <https://www.riverbed.com/nl/products/npm/riverbed-modeler.html>

- 
- [63] X. Zeng, R. Bagrodia, and M. Gerla, "Glomosim: a library for parallel simulation of large-scale wireless networks," in *Proceedings. Twelfth Workshop on Parallel and Distributed Simulation PADS '98 (Cat. No.98TB100233)*, 1998, pp. 154–161.
- [64] S. Swarup, "Adequacy: What makes a simulation good enough?" in *2019 Spring Simulation Conference (SpringSim)*, 2019, pp. 1–12.