Msc. Thesis

Applied Visible Light Communication for Robotics

Martino Secchi
msec@itu.dk

Supervisor:
Kasper Støy

June 2017

Todo list

Contents

Abstract

[This is my abstract, what I did and my results, in about 100 words]

1 Introduction

Visible Light Communication, often abbreviated to VLC, is a type of wireless communication achieved with the transmission of signals in the spectrum of visible light. For most applications, this communication form is implemented through the use of normal light emitting diodes (LEDs) or fluorescent lamps switched on and off at high rates to produce light signals. It is gaining increasing popularity as an application of pervasive computing, since many light emitting devices are commonly present in everyday life and used everywhere. This form of communication has been proven by research to achieve comparable performances to more classic ones, competing in speed and distance (section 2). Although it has some clear limitations, it also provides appealing features in certain contexts. By definition, a VLC system will not work outside of a line of sight with the source, excluding cases where light reflection might be sufficient. This can be seen as a limitation for obvious reasons, but also an advantage in security sensitive applications. Communication can in fact be restricted in specific areas delimited by physical boundaries. However, it will result less effective in presence of direct light from other sources, like direct sunlight, and is generally designed for indoor usage or nocturnal.

A very distinctive trait of VLC systems is that light is a form of **situated** communication, where the message is not necessarily separated from the physical environment in which it has meaning. This allows to transmit a message that includes additional information embedded in the medium of transmission rather than its content. In particular, information relative to localisation and direction can just be derived from the light source's position, instead of it being encoded in the message. This is in general impossible to achieve with many other technologies, like WiFi or Bluetooth, that can only give a sense of directionless proximity to the source, at least without employing techniques like triangulation or fingerprinting or sharing a common reference system. With situated communication, the properties of the transmission can enrich the information to be received, that will be more than just the content of the message. Very much like human hearing, messages like "I'm here" or "come here" can be fully understood and acted upon.

This characteristic has been proven very useful in the field of mobile robotics, and in particular in **swarm robotics**. Swarm robotics is a control paradigm used in multi-robot systems in which usually large amounts of robots interact with each other to form a collective behaviour, with the ultimate goal of performing a task. The strong point of swarm robotics is that relatively simple agents can produce fairly complex swarm behaviours, exploiting constant feedback and communication between the agents. This research field originates from the observation of emergent behaviour exhibited by social organisms, examples of which can be insects like ants and termites, birds, fish, quadrupeds and bacteria. This kind of systems rely intrinsically on local communication and mutual localisation between the agents [14]. A communication method that allows both would therefore allow greater simplicity in the design of such systems. Visible light communication has been applied to swarm robotics in numerous occasions, most notably in the Swarm-bots and Swarmanoid projects funded by the European Commission in 2002 and 2006. In these projects, robotic agents communicate states with RGB LEDs in a colour based system. Each agent can communicate a state coded into a specific colour, and perceive other agents' states. This functionality can be used to achieve dynamic shape formation of self assembly robotic agents [9], or distributed coordination of self organising agents for navigation [5] [17], navigation and path formation [15].

These examples, like many others, use a rather reactive approach to obtain and communicate information. In a colour based system like the before mentioned one, agents simply react to certain colours in certain ways, like they would do for any other sensory information when the behaviour is programmed. It is the same with other kinds of information like the ones coming from proximity sensors or thermometers, accelerometers, gyroscopes and so on. There are circumstances however where this is not enough, and more complex information need to be passed from one agent to another. For instance this could be the case of embodied evolutionary robotics, where agents of a system need to communicate information necessary for reconfiguration of hardware or behaviour, in order to adapt to previously unknown or dynamically changing

conditions autonomously [12]. This kind of complex information could be passed through a generic purpose communication system that allows any kind of message to be sent and received, while still maintaining the advantages of a situated medium of transmission. Such a technology could be used in both very simple or more advanced forms, the simplest of which would be mono directional, environment to robot communication. This could be used in the context of mobile robotics to provide an aid for environment navigation, for example providing useful information about indoor positioning, or a form of remote control of agents, instructing robots on what actions to perform in predetermined locations, to assist in the execution of specific tasks. The technology could also be used in a slightly more complex scenario of robot-to-robot 1- or even 2-way communication between multiple agents to enable cooperation, behavioural reconfiguration, hardware/shape reconfiguration, and in general to exchange information.
The design of a prototype system will help analyse the properties of a generic VLC system, its performances and results in the given applicative scenario of mobile robotics. The design process will be focused on simplicity and inexpensiveness where possible, by using mostly off-the-shelf and easily obtainable hardware components to achieve sufficient performance within reasonable cost and computational power.

more introduction

talk about situated communication

2 Visible Light Communication

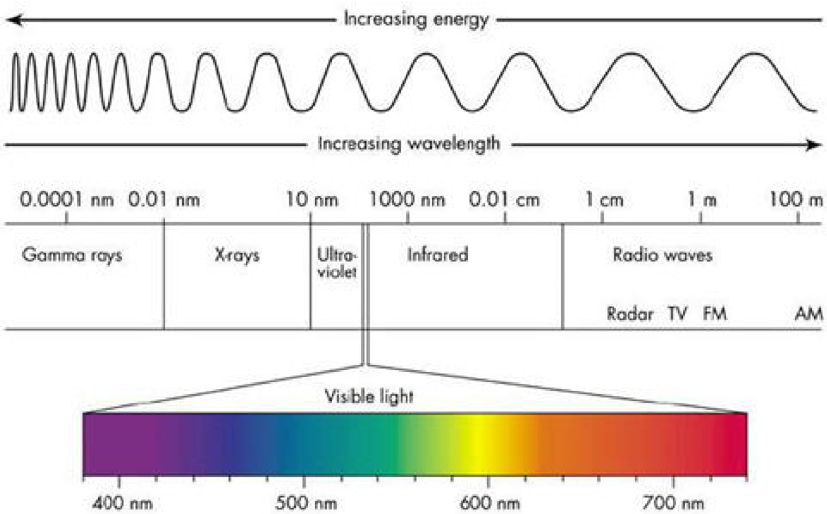

Figure 1: Wavelengths, visible light band

Visible Light Communication is a form of optical communication that uses signals in the visible light band to transmit information. Contrary to the more common fiber-optic communication, VLC is wireless, and transmitted over free space. Typically, normal fluorescent lamps or LEDs, rather than complex communication devices, are used to transmit in VLC. In general, receivers are electronic devices that include one or more photoresistors, in order to measure light signals from a source. In some cases, digital cameras can also be used to form a receiver.

cite use of digital camera

In recent years optical communication has seen a great spread mostly thanks to optical fiber. As opposed to optical fibre communication, VLC is meant to be propagated wirelessly over free space. By modulating a light source, like could be a common LED bulb, through a controller it is possible to encode messages from digital form into light signals. The simplest example of this encoding is a binary representation of data through light, where the presence of light represents a binary 1 and its absence a 0. This form of communication is a simplified variation of the technology known as Li-Fi, a term that stands for Light-Fidelity.

mention fiber vs open space?

2.1 Indoor positioning

philips and supermarkets, indoor location, also other potential usages for indoor positioning, hospitals and so on

2.2 Optical Communication

Optical communication is allegedly one of the oldest types of communication from a distance in the history of human kind. Examples of early communication techniques that use light to carry signals can be traced back millennia, with the first lighthouses, navigation lights, beacon fires that would assist in navigation or communicate danger. It is only with the spread of electricity though, that optical communication technologies could really develop. Many trace the start of Visible Light Technology to 1880, when Alexander Graham Bell and Charles Sumner Tainter invented the photophone, a device able to transmit wireless voice messages over several hundred meters modulating sunlight. Since then, optical communication has been developed to include many different variants, the most common of which is **fiber optic communication**. Other common optical communication technologies include **infrared** light (IR) and **ultraviolet** (UV).

reference to photophone

talk fiber

talk IR and UV

2.3 RONJA

http://ronja.twibright.com/

talk RONJA

2.4 Li-Fi

The term Li-Fi was first introduced by a german physicist, Harald Haas, co-founder of PureLiFi, in occasion of a TED Global talk on Visible Light Communication in 2011. [10] Since then, the term has been reported in many articles and has gained popularity as a common synonymous for Visible Light Communication. After the name was adopted by the Optical Wireless Communication (OWC) community with the launch of the LiFi Consortium in 2011, an industry group that promotes OWC technologies, it can be sometimes extended to describe general wireless data access points that use light, visible or otherwise. This includes also the infra-red and ultraviolet band.

The main reason why this technology is quickly gaining popularity in the research is because it potentially allows to unlock a vast amount of electromagnetic spectrum in the visible light region, unused for transmission.[11] This has been seen as a promising reaction to the saturation of the Radio Frequency (RF) spectrum, a very likely outcome of the huge success of wireless technology, also predicted by the US Federal Communication Commission[6].

A second reason is that transmission through light can achieve surprising high speeds. Starting from 2010, research had been able to improve the transmission rate further and further. In his TED talk in 2011 Haas demonstrated real time video streaming from a white LED at data rates up to 130 Mbps[10], while another group achieved over 513 Mbps[13]. In the following years, there have been continuous reports of improved data rates in transmission. A single white LED has been proven to transmit from about 1 Gbps[1], up to 3.5 Gbps[7], while 3.4 Gbps have been demonstrated with a single RGB LED[8].

This rates can be further improved by the use of arrays of light sources and more complex systems. The Mexican software company Sisoft together with scientists from the Autonomous Technological Institute of Mexico in 2014 reached the surprising data rate of 10 Gbps, setting the record.

Other appealing aspects to this technology are the low cost of implementation for the use of off-the-shelf LED bulbs and its security, for the reason that communication can be eavesdropped only in direct line of sight within short distance.

There are a few groups that are leading the research in the field of Visible Light Communication. One of these is the Li-Fi R&D Centre at the University of Edinburgh, of which prof. Harald Haas is the director. Another very important research group operating in the same area is Disney Research, often in collaboration with ETH Zurich. Another group worth mentioning is the Li-Fi Consortium, an international organisation formed by companies in optical wireless communication technology and research institutes. A good part of the research is also carried out by multiple other research groups worldwide, especially in Asia and India in particular.

2.5 Standard and specifications

Visible light communication is regulated by a standard similar to the one of wireless networks, in the same IEEE 802 family.[4] The IEEE 802.15.7 standard defines a draft of the physical layer (PHY) and the media access control (MAC) layer for VLC. According to Gordon Povey, former CEO at PureLiFi[16], the MAC layer as of April 2011 supports three multiple access topologies: peer-to-peer, star configuration and broadcast mode.
The Physical layer is divided into three types, that use different modulation schemes. The three modulation schemes are:

- On-Off Keying
- Variable pulse position modulation (VPPM)
- Colour shift keying (CSK)

On-Off Keying
On-Off Keying is the simplest modulation scheme. In this scheme, a digital 1 is represented by the light state being on, and 0 otherwise. In the 802.15.7 standard, Manchester coding is used to ensure the period of each pulse is the same. This type of encoding, instead of having a digital 0 represented by a low signal and a digital 1 by a high signal, encodes each data bit as either low then high (1), or high then low (0), in equal time.

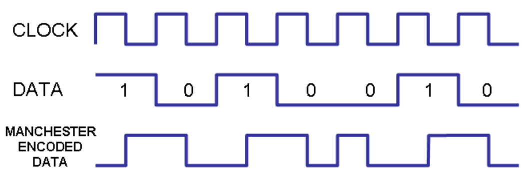

Figure 2: OOK modulation using Manchester coding.

Variable pulse position modulation (VPPM)
VPPM is similar to pulse position modulation (PPM), in which the data is encoded using the position of the pulse within a set time period. In this modulation scheme, light dimming is allowed as long as the period containing the pulse is long enough to allow different positions to be identified. As in the Manchester coding a positive pulse at the beginning of the period followed by a negative pulse at the end can represent a digital 0, and a 1 is represented by a negative pulse at the beginning followed by a positive one at the end.

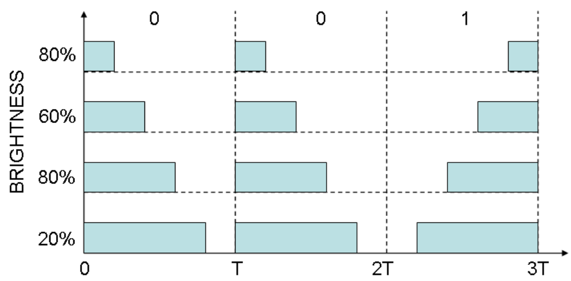

Figure 3: Variable pulse position modulation to support light dimming.

Colour shift keying (CSK)
This scheme allows the light intensity to be constant by encoding the information in the colour of the light. For implementing this kind of transmission the system must use RGB type LEDs.

2.6 Comparison with other technologies

compare ranges and speeds, maybe also bands

3 System Overview

[In the next chapters I describe the overall characteristics and limitations that a VLC system should present according to my findings, identify the variables involved and the set of parameters necessary for it to work This chapter should provide a useful starting point for whoever plans to implement one in practice]
What does one need to make his own VLC system?

Every visible light communication system shares the same underlying structure, as also general other communication systems. Communication needs to be established between two ends, one end that acts as a transmitter and the other that acts as a receiver. There are at least three levels of abstraction needed to transform a digital information into a signal that can be sent and received, and these can be defined as:

- **logical layer**, where the information is handled at a software level
- **control layer**, responsible of managing the variations of the physical property used for communication in a organised and structured manner
- **physical layer**, where the properties of the communication are limited by the physical properties of the signal and the characteristics of the hardware

Each of these levels present different characteristics that combined form the overall performance of the system.

In VLC, the transmitter end of the communication operates on a light emitter to vary a property of the light depending on the modulation scheme used (see 2.5), like brightness or colour. The receiver on the other end needs to be able to detect such variations in a measurable manner. This level can be described as the **physical layer** of the communication. At this level, the system's performance can be influenced by physical properties of the hardware, like maximum brightness of the light emitter, warmup time or time that the emitter takes to turn on, but also by other factors, like distance between the two ends, the medium of transmission, the angle of incidence of the light between transmitter and receiver and so on.

This variations in the properties of light need to be controlled and organised to carry specific information. A **control layer** is necessary in order to make the link between logical information and variation of

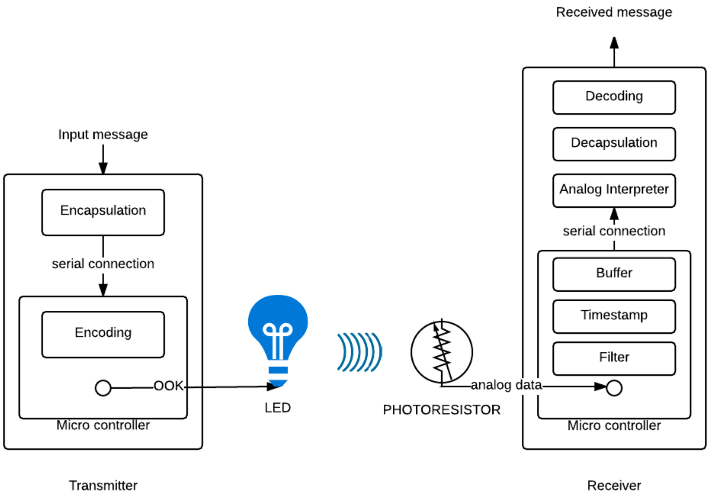

Figure 4: Prototype system.

physical property. This layer is of critical importance for the performance of any system, since it influences the frequency of the physical variations which results in the rates for transmission and reception. Factors that are to be considered in this layer are directly linked to the hardware components, namely any characteristic that influences the overall speed of transmission/reception, like clock rates of the micro controllers, speed of the transistors used and so on.

The outer most level of abstraction is the **logical layer**, where information is handled and manipulated at a software level. For a transmitter, this layer produces the instructions to pass on to the control layer in order to generate a signal given a specific information. This process can be seen as the logical encoding of the information, ready to be transferred and become physical encoding. At the opposite end of the transmission, the logical layer of a receiver is given data about physical variations measured by the sensor(s) used in the system, and needs to reconstruct and interpret such data back to the original information. Contrary to the previous layers, the performance of the logical layer doesn't rely much on the hardware and the physical characteristics of the components, but rather on the software techniques and algorithms that implement it. A well structured logical layer can even add more reliability to an otherwise uncertain medium of communication, as will be seen in later chapters.

do a graph of logical control and physical.

3.1 Experimental setup

In order to verify feasibility, investigate characteristics and test performance of general VLC systems, a prototype system has been developed. The system is composed of two main modules: a transmitter module, and a receiver. The transmitter module uses On Off Keying with Manchester Coding (see 2.5) to convey signals with light. In order to allow testing, this module takes some arbitrary input from a user, encapsulates the information and encodes it to produce variations in light intensity, through the control of a light emitter. The receiver side measures the light variations through the use of a photoresistor, or light sensor, reconstructing and interpreting the sensor data back into the original messages. Each module includes different components, listed in section 3.2. Fig. 4 shows an overview of the prototype's architecture.

in the experimental setup, it's considered almost exclusively low power DC LED, discuss role of big bulb

3.1.1 Transmitter

Transmission starts from a terminal, where a client can input messages as strings. These are then encapsulated into Protocol Data Units (PDUs) and sent to a micro controller through a serial connection. In this case, the micro controller that was used for transmission is an Arduino board. The board encodes the received bytes into Manchester Code, and then controls the light signal by switching on and off the emitter accordingly. A light emitting diode is the furthest end of the transmitter module. For this part, multiple setups have been tried. The fastest emitter that has been tested is single low power LED connected to the board and powered directly by it, which allows very fast switching. A second setup has also been tested with a commercial LED bulb powered by an external power source (mains electricity), and controlled by the board through a switcher.

mention that we use LEDs because they are faster and do not wear out.

3.1.2 Receiver

The signal is received by a photodiode read as analog input by a micro controller. On the board, values are software-filtered to reduce noise, and sent to the receiving terminal with a timestamp. The board and the terminal are connected through a serial connection. Some controllers could read directly analog input and perform the remaining processes to interpret the signal, with enough computational power.
For this application, the final terminal is a Raspberry Pi, since it has a good power/size ratio for the designed purpose and allows a simple prototyping process. The micro controller in between the sensor and the terminal is necessary to read and forward analog data, which is not possible directly from the Raspberry Pi. On the final terminal, the variations of sensory data are interpreted as sequences of digital 0s and 1s, finally decapsulated and decoded back into a message.

3.2 List of components

Each module of the prototype system is composed of different components, listed in the following.
Transmitter:

1. personal computer as main terminal for user input and information processing
2. Genuino Uno micro controller board, based on ATmega328P [2]
3. Light emitting diodes:
   (a) low power LED, Blue 10 mm, forward voltage 3.0-3.4VDC, luminosity of 8 000-10 000 MCD, directional, 30° viewing angle
   (b) commercial LED bulb, 230 V AC, 3 W, 240 lumen, directional, 120° viewing angle
4. Solid State Relay for Arduino, 5V activation, 240V load

Receiver:

1. Photoconductive Cell VT900
2. Genuino Yun micro controller board, based on ATmega32u4 [3]
3. Terminals:
   (a) personal computer for test of communication
   (b) Raspberry Pi 3 Model B to be applied on a mobile robot

3.3 Wiring

In figure **??** is shown the wiring of all the components of the prototype system.

add wiring? diagram for small setup and also setup with ssr and big bulb

4 Physical layer

This layer is about physical properties of the hardware and the signal itself.
In this section these properties will be analysed through the aid of various experiments performed on the prototype system. This is a first step in understanding the potential and the limitations of a system, because whatever limitation derives from the physical layer, it will be a hard limitation that will affect the entire system. Therefore, starting to look at physical constraints of the system will give a good overview of what can be achieved and what cannot early on, before the rest of the designing process. All the following experiments will be performed on the prototype system as described in section 3.1, with the use of the low power 10mm LED unless otherwise stated. A remark on the experiments that will follow is that brightness will be always reported in percentage. This serves the purpose of making the data applicable in general cases, even with different light sources. The LED used for testing produces a luminous intensity of about 8 000 to 10 000 MCD at 25° . reference datasheet for led

4.1 Warmup time

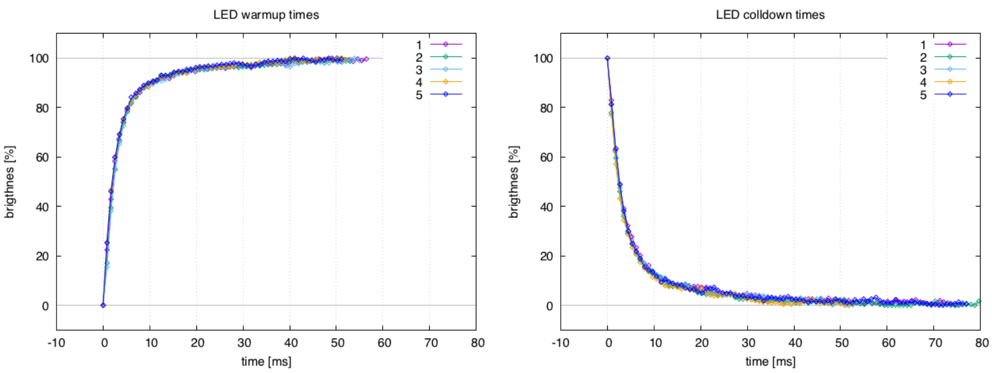

Figure 5: LED warmup and cooldown times.

The speed of light transmission depends in primis on the speed at which the light itself can be turned on and off. Figure 5 shows the warmup times for the low power LED over multiple instances, meaning the time that it takes for turning the light completely on from completely off, and vice versa. These measurements depend on the reception rate of the system, which will be discussed in section 5.2. Table 1 shows the times for the LED to switch between specific brightness levels. Each row represents the time to reach the level on each column, for example the first row represents the time to reach any brightness level starting from 0% brightness. The table works both ways, meaning it shows the time for the warmup as well as the time for the cool down of the LED. As another example, the last row shows how long it takes to reach any level from a completely ON state, meaning starting from 100% of brightness.

From the table as well as from fig. 5, it can be seen that the LED is slightly faster at being turned on rather than off. Also pretty indicative is the fact that about 90% of the time used for turning a LED completely on or completely off, is spent to make a variation in the last 20% of brightness levels, which means that a LED can reach a reasonably high brightness in a time relatively small compared to its full potential. Interesting for later sections is that 50% of the brightness can be reached in about 2 ms. stats for big bulb?

4.2 Interference from ambient light

Previous data about times to reach maximum brightness were measured in a condition of full darkness of the environment surrounding the light source, at minimal distance between transmitter and receiver. But

	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
0%	-	0.3	0.66	1.08	1.52	2.12	2.82	3.78	5.38	10.4	50.66
10%	55.88	-	0.36	0.78	1.22	1.82	2.52	3.48	5.08	10.1	50.36
20%	61.86	5.98	-	0.42	0.86	1.46	2.16	3.12	4.72	9.74	50.0
30%	63.76	7.88	1.9	-	0.44	1.04	1.74	2.7	4.3	9.32	49.58
40%	64.9	9.02	3.04	1.14	-	0.6	1.3	2.26	3.86	8.88	49.14
50%	65.76	9.88	3.9	2.0	0.86	-	0.7	1.66	3.26	8.28	48.54
60%	66.4	10.52	4.54	2.64	1.5	0.64	-	0.96	2.56	7.58	47.84
70%	66.98	11.1	5.12	3.22	2.08	1.22	0.58	-	1.6	6.62	46.88
80%	67.48	11.6	5.62	3.72	2.58	1.72	1.08	0.5	-	5.02	45.28
90%	67.92	12.04	6.06	4.16	3.02	2.16	1.52	0.94	0.44	-	40.26
100%	68.28	12.4	6.42	4.52	3.38	2.52	1.88	1.3	0.8	0.36	-

Table 1: Warmup times in [ms] of the LED, for specific levels of brightness. Row: from brightness, Column: to brightness.

visible light communication is not necessarily used with this restriction, it is in fact meant to be used over an open space, wirelessly, meaning that interference from natural light is very likely. Experiments have been performed to quantify the influence of ambient light. This is done by comparing the difference between the brightness measured by the sensor when the transmitter is turned ON and the one when the transmitter is turned OFF. This difference will be measured without ambient light interference and used as a reference for the measurements that occur with ambient light interference. Since communication will later rely on the quantification of light variations, it's important to establish if this variations will be substantially different with or without interference present. For these experiments, for ambient light it is meant light from any source that is not directed to the receiver but naturally permeates the environment surrounding the system. If the variation ON/OFF in a fully dark environment is represented as 100%, the experiment shows that this difference is between 100% and 120% when ambient light is present, averaging at around 110%. Measurements were taken at the same conditions of distance and angle. This increase could be explained with a higher sensibility of the sensor when exposed to higher levels of base brightness. Overall it is fair to deduce the ambient light interference doesn't pay a big role when in reduced amount and without distance limitations.

mention interference increase over distances, otherwise it seems like more light is better

4.3 Distance

Another factor that might affect the communication over light is the distance between emitter and sensor, closely bound with the brightness reachable from the light emitter and the presence and weight of light interference. Intuitively, brighter lights will be visible from farther. A low power LED cannot reach high levels of brightness, therefore it won't be visible at long distances. Table 2 shows the results of different measurements taken at increasing distances sensor to light source. The reference for maximum brightness is achieved with a distance of 0 cm between receiver and light source, with the two nearly touching. Figure 6 shows the measurements of the same experiments in a graphical manner. In the figure, each colour represents a different distance, and each peak a different experiment. The experiment was performed in a dark environment, minimising the risk of interference from other light sources.

From the table and figure it can be seen that the maximum brightness registered from the sensor drops clearly with distance. This obviously makes reliable communication over long distances harder. An important implication with this results is the role of noise in the reception. Noise doesn't scale with the communication, meaning that if the maximum brightness perceived at a certain distance is 50% of the brightness at 0 distance, the noise in this first case is not 50% of the one used for reference, but stays constant. This means that the lower the variation perceived, the more noise will play a role in the communication. A remark on the experiments performed is that the LED used for testing is directional, with an optimal viewing angle of

distance	max. brightness
0 cm	100%
10 cm	42.81%
20 cm	14.28%
30 cm	8.57%
40 cm	5.00%
50 cm	3.57%
100 cm	2.14%

Table 2: Maximum brightness over different distances.

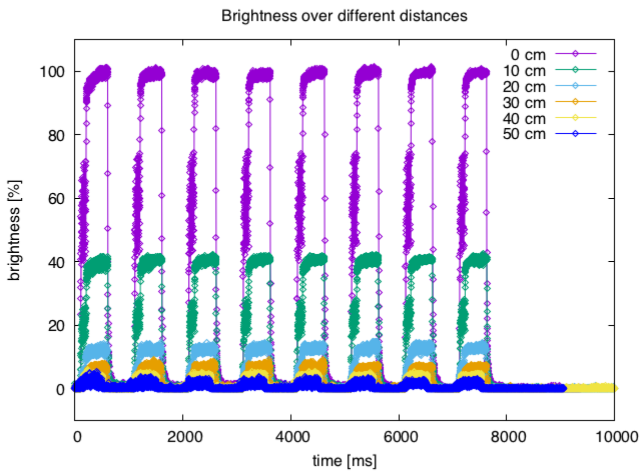

Figure 6: Brightness at different distances.

about 30° from the centre of the emitter. The receiver, at the different distances, was always placed in the optimal viewing point in relation to the light emitter direction.

4.4 Angle

In addition to the previous factors, also the angle of incidence between the light emitter and the light sensor might be of relevance when establishing communication between the two ends, under certain circumstances. So far, in all of the communication tests performed on the system, the receiver was always in direct line of sight with the light emitter and centred to its focus. However, if the light source is directional or semi-directional, the angle of incidence would affect the ability of the sensor to register light variations. This could also be the case for omni-directional light sources, when reflection of the light would be critical for it to reach the sensor. For example, the light source could not be bright enough to reach reflecting surfaces or these could be absent. In the case of a regular room as the location of the communication system, the light source must be powerful enough to either reach the sensor directly with sufficient intensity, or to reach it indirectly by reflecting on the walls, ceilings or other surfaces. Experiments have been performed with different angles to show how this factor affects overall reception with a directional light source. The measurements have been taken at a distance of 10 cm between light source and sensor, in a otherwise dark environment. These results suggest that the larger the angle of exposure, the less powerful the reception.

Angle	max.brightness
0°	100%
40°	60%
60°	50%

Table 3: Maximum brightness measured at different angles.

4.5 Powerful lights and power source

As can be implied from the previous results, brighter and faster light sources are preferred to achieve better communication, at preferably low angle. The brighter a light, the farther it can be perceived, and the faster it is the more it allows fast switching, which in turn allows faster communication. Such lights have a higher power consumption than less performant ones. If the vision is to have a smart environment with lights that serve the double purpose of lighting the environment for humans and transmit information to devices, the most natural thing would be to connect these lights to mains electricity like any other lamp. This power source potentially allows the usage of much more powerful light bulbs then the low power LED used in the prototype. In Europe, electric power supply for mains is of 230 V at a frequency of 50 Hz . A distinctive trait of this power source is that it uses Alternating Current (AC). Alternating current periodically reverses direction whereas direct current (DC) always flows in the same direction. AC voltage can be expressed by a sinusoidal function, which means the current will vary its intensity in time. Current will be higher near the peak, and lower near the time of the periodic switch in direction. This behaviour is clearly represented in figure 7, when the light is ON after the warmup in the figure on the left, or before the cool down in the figure on the right. The ON/OFF difference when measuring the light intensity with the receiver is over 500% bigger than the LED of the prototype system, with similar rise speeds. However, the fluctuations of AC during the ON state make it harder to use it for communication using On Off Keying, since that modulation scheme relies on variations in intensity. A way around this would be to convert AC to DC with a process called rectification . This involves the use of a capacitor connected between AC V^+ and Ground, which lowers the amplitude of the AC significantly thus producing something very similar to DC. Another problem to face however, as shown in the picture, is that these LED bulbs have a much slower fall time when switched off, which also would affect the switching frequency during transmission.

talk about AC vs DC, discussion about power vs brightness vs distance vs speed

reference mains information.

reference recification

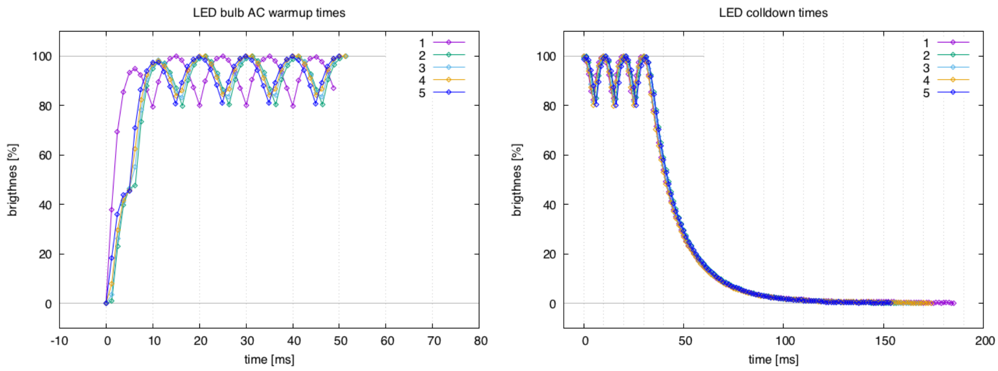

Figure 7: AC bulb warmup time, and phases of alternating current.

remove outlier in the picture of ac?

5 Control layer

This layer is responsible of connecting the physical and the logical layers of the system, connecting the physical variations that produce the signal with the logical information they carry. For a transmitter this translates to the process of producing variations of the physical properties in a structured way according to what dictated by the logical layer, in a way that these variations reflect the information given. For the receiver, the control layer is responsible of reading the values from the sensory layer and forward them to the logical layer, perform filtering and other minor operations. In the prototyped system, this layer is physically represented by micro controller boards connected with a terminal through a serial connection on one side, and with the components of the physical layer on the other. This layer is crucial for establishing performance of the system in terms of **rates**, as in how many values can be read from a sensor per time unit for reception, and how fast can a signal be sent to control the light emitter for transmission.

5.1 Connection to the logical layer

Being in the middle of the system, this layer has two connections, one outgoing and one incoming. The transmitter will have an incoming connection with the logical layer and one outgoing to the physical, while the receiver will have an outgoing connection to the logical layer and one incoming from the physical layer. Since this layer is the one that produces transmission and reception rates, it's worthwhile to spend some time in analysing both types of connections, because the sum of both will result in the overall performance of this layer. Transmission rate, as will be discussed, depends on the speed of the outgoing connection from control layer to the physical layer for obvious reasons. However, the speed of the incoming connection from the logical layer would also play a role. The same discussion applies for the reception rates, in reverse. In the prototype, the connections between control and logical layer are implemented as serial connections between the controller board and the terminal where the logical layer is running. Both boards are Arduinos, and handle the serial connection in the same way. There are a few aspects to consider when implementing the link between control layer and logical layer, from the logical layer point of view. One aspect to consider is that the serial buffer has a limited size of 64 bytes in most Arduino boards, so it's important to keep the serial communication as light as possible to avoid overflow and loss of information. This means that if the message that needs to pass through is encoded as a string, it would need to be sent one character at a time and avoid unnecessary additional information. In most cases, it's possible to encode a single character into one single 8-bits byte, but not all the programming languages implement this automatically. The system prototyped in this case was implemented using Python 2.7, which uses dynamic types and therefore doesn't have a specific type for *chars*. In Python, strings are objects with an overhead of 37 bytes, plus one byte for every character in the string. This would result in a very heavy serial transmission for just a few characters. Characters need to be converted in single bytes before sending them.

information about clock rates of the boards

5.2 Reception rates

Reception happens as fast as the micro controller allows, in fact the control process doesn't force a specific speed on the analog input coming from the sensor, but rather the rate of reception is bound to the clock rate of the processor in the control layer and the speed of the physical sensor. The average reception rate is of about one value every 0.833 ms, about 1200 Hz (or bits per second). This value naturally becomes the upper bound for the transmission rate, since if signals were sent faster, they wouldn't be received in time. The control layer also performs the two operations of appending a timestamp to each value once received before sending it to the next process, and filtering each value as an average of its direct predecessor and itself, to reduce noise. With the use of timestamps it's fairly simple to measure the reception rate before the serial connection. This rate is not constant, but has a standard deviation of about 0,0184 ms.

5.3 Transmission rates

The transmission process takes as input a message that needs to be converted into Manchester code (see 2.5). Just like pure binary, Manchester encoding only has two symbols, 1s and 0s. A symbol, in telecommunication, represents the smallest amount of data that can be sent in form of an analog signal. The symbol rate (symbols per time unit) is measured in baud. Each pair of symbols in Manchester represent a single binary bit. Given the limitations for the reception side, it's in practice very difficult to measure transmission rates above 1000 Hz in the prototype system. The reception rate imposes a first limitation on the transmission rate, since transmission faster than reception would be extremely faulty. A second limitation is introduced by the physical speed of the LED used. From the measurements of warmup times of the LED presented in section 4, it's possible to see what amount of variations are achievable in 1 ms, which would result in a rate of 1000 Hz. In the best case, it's at most a 30% increase starting from 0 and a 25% decrease starting from 100, although in average it would be less. Along the middle of potential brightness, where transmission is supposed to happen, 1 ms could cause a variation of about 15% in brightness. The control layer is directly responsible for the transmission rate, therefore it needs to guarantee a rate that allows a reliable reception. The previous limitations in reception rate and LED speed would suggest that a maximum acceptable rate is of 1000 Hz, with a potential difference between an average 1 and an average 0 of at most 15% of brightness, and a reception rate that is slightly higher. Would this rate produce acceptable readings on the receiver side?

Experimentally it was found that sending signals at 1ms intervals produces a degree of uncertainty of about 30%, while a 2 ms interval performs much better at 0% uncertainty. More on the experiments and results in section 5.4. According to these results, each single symbol is forced to last for 2 ms before the transmitter board can send the next. The transmission rate would therefore be at best of 500 bauds in theory, in the case where the activity of the micro controller pays no role. This value was experimentally measured to be slightly smaller, at around 450 bauds.

5.4 Experiments and Results for Transmission Rate

To find the perfect transmission rate experimentally, the prototype has been set up to transmit sequences of alternating 1s and 0s of known length multiple times. For this experiment, the distance and angle from the sensor to the light source have been minimised to obtain the best reception possible. The experiment aims to measure and count the number of local maximums and local minimums in the transmission, that would define the number of 0 and 1 signals received. Also, the brightness levels have been measured for such local peaks.

The fist transmission rate to be tested is a 1000 bps rate, with an interval of 1 ms between subsequent signals. Over 30% of the signals haven't been received.

A second rate of about 500 bps has also been tested to compare with the first one. This time the interval between two subsequent signals is of 2 ms. Intervals of 3 ms and 4 ms have also been tested. Table 4 shows the results of various measurements with the various rates, while fig. 8 shows overlapped samples of the transmissions with the different setups.

As can be seen in the figure, during transmission the brightness achieved from the LED used for testing is not at 100%, but transmission is still clearly recognised at a level of about 60%. In the figure, the starting and ending points of the transmission are at the minimum and maximum brightness level of the LED, to the far left and far right respectively.

The transmission is always the same over all the instances and in all setups. Reception of the same message in different rates happens at different speeds, consistently with the rate difference. The x-axis in the figure shows the progression of time receiving the message, and it appears clear that lower rates produce slower receptions. Another difference that is clearly noticeable is that the difference between 1s and 0s also varies depending on the time that the signal is forced to stay on. These results suggest that the lower the rate, the more reliable the communication, since broader variations would be more robust to interference and noise. However, the rate of 2 ms per signal has been chosen to be final transmission rate in the prototype system with the low power LED, being the fastest reliable rate. With a system that uses a different light emitter, or designed to be used at longer distances, different experiments would be advised.

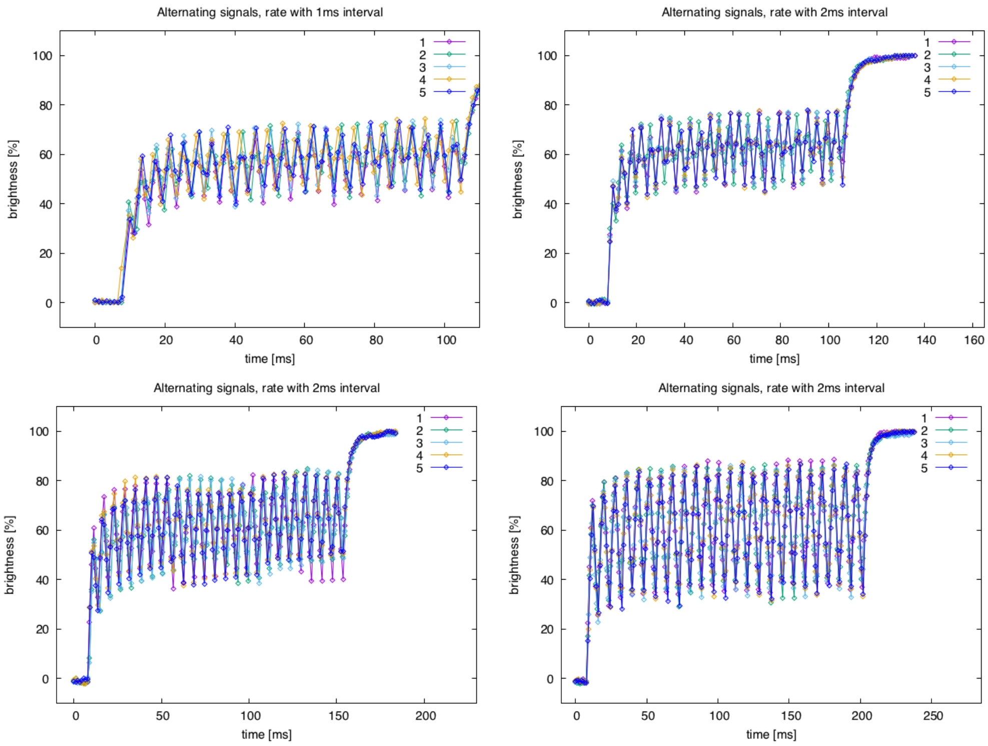

Figure 8: Test of rate with 1 ms, 2 ms, 3 ms and 4 ms intervals (left to right, top to bottom).

5.5 Distance

As we can see from table 4, 2 ms is the fastest reliable rate, but how does it scale over different distances? In section 4.3 it was discussed how distance influences the reception in terms of maximum brightness registered from the sensor. However, a new relevant question is how distance would influence reception when there is transmission. Like in the previous paragraph, an answer to this question can be found experimentally by repeatedly switching the light on and off, and measuring the characteristics of the signal that is received, at different distances.

Table 5 shows the results of these experiments. Contrarily to section 4.3, the percentages of brightness are relative to each separate experiment, in the sense that each separate experiment with a different distance has a different level for 100% of brightness received. This means that for each experiment, a 100% represents the maximum brightness measured in that experiment. This is to shift the focus on the variations and the relation between high signals and low signals, to ultimately verify if even at different overall brightness levels transmission is still comparable. Maximum brightness levels are still reported on the table however.

rate interval	missing 1s	missing 0s	average 1 brightness	average 0 brightness
1 ms	34.52%	30.62%	66.75% +- 6.10	55.78% +- 6.72
2 ms	0.00%	0.00%	70.22% +- 7.06	50.18% +- 5.79
3 ms	0.00%	0.00%	76.72% +- 6.18	43.03% +- 5.20
4 ms	0.00%	0.00%	82.77% +- 4.00	35.54% +- 3.60

Table 4: Rates results compared.

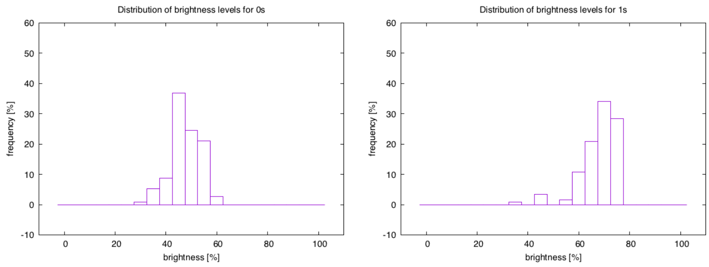

Figure 9: Distributions of brightness levels during transmission (2 ms interval), brightness of binary 0s to the left, 1s to the right.

distance	max. brightness	missing 1s	missing 0s	average 1 brightness	average 0 brigthness
0cm	100%	0.00%	0.00%	70.22% +- 7.06	50.18% +- 5.79
10cm	42.81%	0.00%	0.00%	60.30% +- 10.64	43.61% +- 10.67
20cm	14.28%	4.68%	4.68%	49.99% +- 12.22	36.41% +- 12.01
30cm	8.57%	2.38%	1.78%	54.49% +- 18.83	39.69% +- 20.37

Table 5: Test of 2ms rate over different distances.

Two important aspects emerge from these results: one, that reliability remains reasonably high, and two that uncertainty in the average brightness for 1s and 0s grows a lot, probably due to the increasing weight of noise.

5.6 Controlling AC powered lights

ssr talk

As was mentioned, the control layer is supposed to control the switching of the light bulb by directly control its power supply. This is easily achieved in the prototype since the LED is powered directly from the controller board. However, in the optic of using light bulbs powered from AC mains, the controller layer would need to act differently. Different switching mechanisms have been tried, and by far the most performant resulted to be a Solid State Relay switcher, or SSR for short. A SSR has the power to close or open the circuit connected to the AC source. With a small activation signal it can allow the current to flow through or stop it. The activation signal can easily be sent from the controller board. However, the use of a SSR brings additional limitations to the system, especially on possible speed achievable in transmission. Although faster than other types of relays, SSRs still have some limitations, like in the activation speed. Fast switching at a rate faster than the activation time of the relay will have no effect, and switching at a rate that is only slightly slower will allow only a limited amount of current through before switching off, as the experiments suggest.

experiments ssr

graph of ssr speed (real/bigLED/test1)

wiring with ssr diagram

6 Logical layer

This layer represents the pure software component of the system.
This layer handles the input of the messages to be transmitted, the communication and encoding to the control layer, and most importantly the decoding, interpretation and forwarding of the message to its final consumer. Logical encoding may be able to add robustness to the unreliable aspects of the communication. A good logical layer could handle bit recovery in cases where symbols are missing or mistaken, and overall reduction of noise importance. Also a basic communication protocol could help in making the communication more stable and structured. The logical layer for the transmitter is not very complex, it takes some message as input, manipulates it to meet the encoding and protocol requirements and redirects it to the

control layer in a certain way, as previously discussed. The focus of this section will therefore be shifted to the receiving side of the communication.

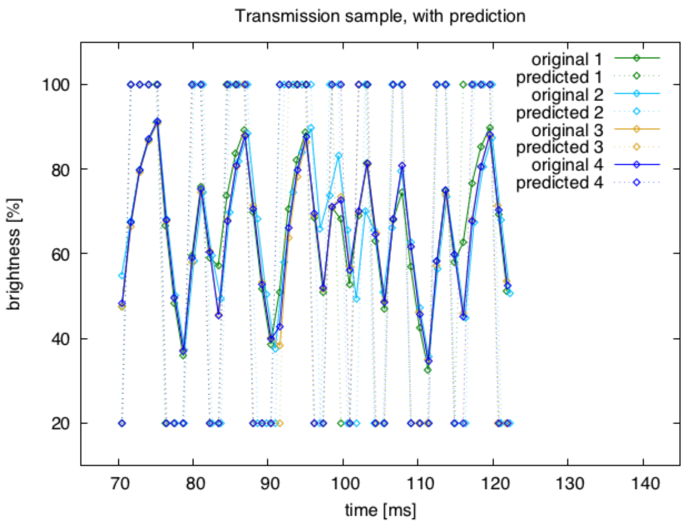

Figure 10: Example of the results of interpreting digital information from analog input.

6.1 Reading the signal

A key part of this layer is to convert analog sensor data representing light variations into binary code. Fig. 10 shows a sample of transmission. The green line represents the original values from the light sensor, while the purple line is a reconstruction of what the digital transmission would look like. Manchester encoding allows transmission to only have two kinds of peaks, either representing single bits, like a 1 or a 0, or two bits of the same kind, like a 11 or a 00. This is because each binary 1 is represented as the sequence 01, and each 0 as 10. Whenever two subsequent bits are different, there will be a sequence of two symbols of the same kind in Manchester encoding. As an example, the sequence 0 0 0 0 0 0 0 1, representing the byte 1, would be encoded as 10 10 10 10 10 10 10 01. When there is the switch between the binary 0 and the final bit 1, a sequence of two 0 symbols appear in the Manchester encoded byte. During transmission this would translate to a longer and deeper 0 symbol, and it would be the same in reverse for a double 1. Luckily, longer same symbol sequences are not possible with this encoding.

The design of an algorithm that could go through this sequences of sensory values and interpret them into binary data has undergone through different stages and different attempts.

6.1.1 Machine Learning approach

The first version of it made use of Machine Learning techniques to train a model with training data, "teaching" a classifier what kinds of data and results one would expect in this application. The data was taken in different circumstances of ambient light interference, with different messages sent and at different times. Multiple classifiers were tested, with a maximum success rate of about 75% of correct predictions. The classifiers included NaiveBayes, RandomForest, AdaBoost and ArtificialNeuralNetworks, all with comparable results.
There were major difficulties with this approach. For starters, there were problems identifying the labels to train these models, namely the classes or categories where the data would belong. Initially, the four

categories were "1", "0", "11", and "00". The basic idea is that given a sequence of values it should fall in one of these four classes. The problem was though that single bits and double bits have different sizes in amount of values that are received from the sensor. This size is also called the number of features.
Most standard modules for machine learning require the same number of features for all the labels.
In order to get around this rule, the labels became "10", "01", "11", and "00". Eventually, also these labels were produced by different numbers of features. Other combinations were tried, and eventually all leading to the same result. One way to get around this problem was to have a fixed number of features, but big enough to include all the possibilities. Samples smaller than this maximum amount (virtually all of the samples) would be filled up with blank values until they reached the intended size.
Another way would be to train a classifier for each specific number of features that may occur. In this case, the amount of classifiers needed would be around 4, for number of features ranging between 7 and 10. The downside of this approach is that each classifier becomes very specific to a restricted set of the training data, potentially loosing track completely of certain labels.
The final setup for machine learning prediction used the labels "101", "100", "011", "110", "001", "11", "00", "10", "01", scaling of the features and larger sample size to be filled up with blanks. This produced overall the best results for the machine learning approach, but never above 80% success rate.
A second downside to this approach would be that when receiving values from the sensor in real time, one would need to keep a moving window of the size of the number of features and try to obtain a prediction from the classifier each time. This might likely slow down the reading process.

6.1.2 Custom algorithm approach

These poor results ultimately led to a change of approach for reading the data. Just looking at fig. **??** it's pretty straightforward to see that the high peaks represent 1s and the low peaks represent 0s. By setting some rules, a custom classifier could predict a result without the need to be trained with sample data. In particular, the classifier analyses sequences of values, trying to find sequences that are either monotonically increasing or monotonically decreasing. Also a noise factor has to be taken into account, in this way small enough variations are not considered. As mentioned before, there are only two kinds of peaks in the sequences, either single or double peaks. Therefore, it's of critical importance to be able to distinguish the two cases. One criteria that was originally deployed for this was to count the number of values that progress in the same direction, either up or down. This was later changed to the more precise criteria of duration of such sequences, and for this reason for each sensor value a timestamp is also included in the control layer, representing the time of reception for that specific value.
With this in mind, the algorithm has been developed to very simply count the duration of either monotonically increasing or decreasing sequences of values, and produce a prediction when the direction changes, based on such duration. This technique performs particularly well in this application, producing up to 100% success rate in certain cases. Runtime wise, the algorithm takes constant time for each value that is received, and may or may not produce a resulting prediction. A simplified version of the algorithm can be found in fig. 12, written in Python 2.7. The algorithm has been simplified to be more readable.

6.2 Protocol

As can be seen the rates of transmission and reception in this prototype system are not very high, if compared to average wireless transmission speeds, which are measured in Mbps. Transmission in visible light is also exposed to a high degree of uncertainty, depending on parameters like distance, maximum brightness, interference, noise, and so on. In this prototype, communication is only established in one direction, which additionally increases the threat of getting wrong results. In order to compensate all this aspects, an effort to strengthen the success rate has been made by structuring the communication into a very basic protocol. Each PDU is included between two single bytes: STX, that indicates the start of a message, and a byte ETX which indicates its end. These two are bytes 0x02 and 0x03 respectively. To avoid that the reader assumes an ETX byte wrongfully while reading the message, a length byte LEN is also included in

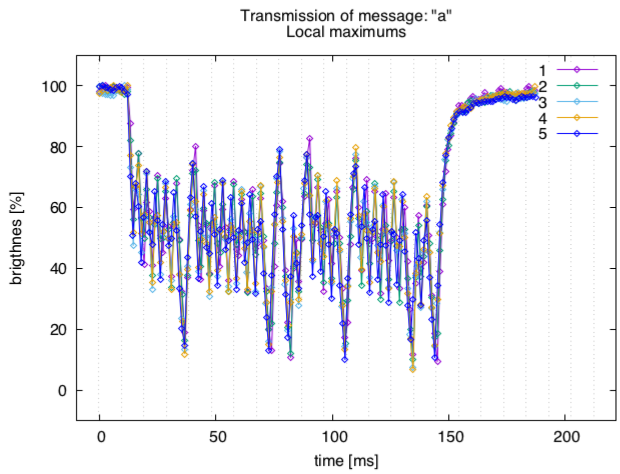

Figure 11: Example of transmission of message ”a”, the transmission include STX, ETX, and LEN bytes.

the header of the PDU, to specify the length of the message in bytes. Additionally, each PDU is restrained to have a maximum size, and longer messages need to be split into more PDUs. This is mostly to preserve the serial connection between logical layer and control layer to overflow the serial buffer. This three fields combined should make it fairly easy for the receiver to know if some parts of the message were lost during transmission. Since the communication is established mono directionally, the transmitter wouldn't know if the message was delivered, but the receiver most likely would. Mistaken symbols are very unlikely in this kind of communication, it is much more likely to entirely miss some. Therefore the receiver could easily know if the message was delivered or not. To increase reliability, the transmitter could be transmitting the same message over and over.

6.2.1 Additional fields

Increasing the functionalities or complexity of the system, additional parameters in the PDU might result useful if not necessary. Imagining a scenario with multiple receivers of messages broadcasted through light in a unilateral way, a receiver field could be added to the PDU. Also, it would be wise to perform encryption on the PDUs, in a way that only the intended receivers would be able to read the message directed to them. Establishing 2-way communication would also require more steps. 2-way communication would be a very good way to ensure correctness of the transmission. For this, at least 2 more parameters seem necessary: a sequence number to identify the packet, and a checksum to check correctness of the received message. A header for visible light communication doesn't require many parameters since it's likely to be the very last end of a communication process. Since it's mainly short distanced, it doesn't need any routing parameters as it would in big networks.

6.3 Results

protocol vs no protocol? bits correct per bits sent? compare other technologies?

Figure 11 shows an example of the transmission of the message ”a”, including STX, ETX and LEN bytes. In ASCII, letter ”a” is represented by the byte 97, or 01100001 in binary. As previously mentioned, the STX is byte 2, ETX byte 3, and LEN in this case would be 1. In binary, the entire message would then be: 00000010 - 00000001 - 01100001 - 00000011. In Manchester encoding: 1010101010100110 - 1010101010101001 - 1001011010101001 - 1010101010100101. The traits of the signal in the figure can be visibly associated with the Manchester encoding of the message, clearly visible at least in the occurrence of double symbols. A 00 and a 11 are clearly visible at the end of the first byte, a 00 almost conclude the second byte, follows a 11 connecting the second to the third byte, then a 00, 11, and so on.

6.3.1 Experimental setup

To test performance of the communication at its final stage, multiple tests have been performed on the prototype. For each test, the system takes user input for initialising the message to send, encodes it and sends it through light. The receiver reconstructs the message depending on values retrieved from the sensor. Tests have been performed at very close proximity of the sensor to the light source, without other lights directly facing the sensor but with presence of natural light. A test is considered successful if every bit is correctly received and interpreted. Messages of different sizes have been tested. Table 6 illustrates the results of these tests.

distance	ambient light	tests	successful
0 cm	yes	50	100%
0 cm	yes	50	100%
0 cm	yes	50	100%
0 cm	yes	50	100%
10 cm	yes	50	92%
10 cm	yes	50	96%
10 cm	yes	50	88%
10 cm	yes	50	60%

Table 6: Tests of full transmission, success criteria is 100% correctness.

7 Evaluation and Results

[How did it go as an application to robotics? does it make sense? performance? tests? comparisons to other communication forms?]

7.1 Experimental setup

7.2 Experiments

7.3 Performance

7.4 Discussion

8 Discussion

[This is some discussion. Here I can link my results to the vision that was presented in the introduction, are the results good enough for some of those applications? Are those scenarios plausible according to my findings? what could be done with my system, more than what has been done? How could one take my work and use it for other purposes, or expand it? how to make it faster?]

9 Conclusion

[This is the conclusion. Here I sum up all the crucial points of my work, from the vision, to my results, to some of the major points in the discussion.]

```python
def feed(self, time, value):
        pred = None

        # staying
        if abs(value - self.prev) <= self.epsilon:
                pass

        #going down
        elif value <= self.prev:
                if self.direction: # up
                        pred = self._predict(time)

        # going up
        elif value > self.prev:
                if not self.direction: # down
                        pred = self._predict(time)

        self.prev = value
        return pred

def _predict(self, time):
        m = self.direction
        delta = time - self.seqstart
        pred = '-'

        if delta >= self._doubletime:
                pred = '11' if m else '00'
        else:
                pred = '1' if m else '0'

        self.direction = not self.direction
        self.seqstart = time
        return pred
```

Figure 12: Simplified algorithm for signal interpretation (Python 2.7).

References

[1] R. Corsini P. Choudhury E. Ciaramella A.M. Khalid, G. Cossu. 1-gb/s transmission over a phosphorescent white led by using rate-adaptive discrete multitone modulation. *IEEE Photonics Journal 4, 14651473*, 2012.

[2] Arduino. Arduino uno overview. `https://www.arduino.cc/en/main/arduinoBoardUno`.

[3] Arduino. Arduino yun overview. `https://www.arduino.cc/en/Main/ArduinoBoardYun`.

[4] IEEE Standards Association. Ieee 802.15 wpan task group 7 (tg7) visible light communication. `http://www.ieee802.org/15/pub/TG7.html`.

[5] Nolfi S. Baldassarre G., Parisi D. Distributed coordination of simulated robots based on self-organisation. *Artificial Life*, 12(3):289–311, Summer 2006.

[6] US Federal Communication Commission. `https://www.fcc.gov/general/spectrum-crunch`.

[7] S. Rajbhandari J.J.D. McKendry S. Videv E. Gu M. Haji S. Watson A. Kelly G. Faulkner M.D. Dawson H. Haas D. OBrien D. Tsonev, H. Chun. A 3-gb/s single-led ofdm-based wireless vlc link using a gallium nitride led. *Photonics Technology Letter 99(99)*, 2014.

[8] P. Choudhury R. Corsini E. Ciaramella G. Cossu, A.M. Khalid. 3.4 gbit/s visible optical wireless transmission based on rgb led. *Optics Express 20, B501B506*, 2012.

[9] R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Autonomous self-assembly in swarm-bots. *IEEE Transactions on Robotics*, 22(6):1115–1130, 2006.

[10] Harald Haas. Ted talk online by harald haas on wireless data from every light bulb. `http://www.ted.com/talks/harald_haas_wireless_data_from_every_light_bulb`.

[11] Harald Haas. High-speed wireless networking using visible light. 2013.

[12] Evert Haasdijk, Nicolas Bredeche, and A. E. Eiben. Combining environment-driven adaptation and task-driven optimisation in evolutionary robotics. *PLOS ONE*, 9(6):1–14, 06 2014.

[13] S. Nerreter K. D. Langer J. W. Walewski J. Vucic, C. Kottke. 513 mbit/s visible light communications link based on dmt modulation of a white led. *J. Lightwave Technol. 28 (24), pp. 35123518*, 2010.

[14] D. Rus. N. Correll. Architectures and control of networked robotic systems. *Handbook of Collective Robotics, pp. 81-104*, 2013.

[15] S. Nouyan and M. Dorigo. Chain based path formation in swarms of robots. volume 4150, pages 120–131. 2006.

[16] Gordon Povey. `http://visiblelightcomm.com/an-ieee-standard-for-visible-light-communications/`.

[17] V. Trianni, S. Nolfi, and M. Dorigo. Cooperative hole avoidance in a *swarm-bot*. *Robotics and Autonomous Systems*, 54(2):97–103, 2006.