A Power-Efficient Smart Lighting System: Modeling, Implementation and Evaluation

by

Yerbol Aussat

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2019

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Lighting load accounts for approximately one third of overall energy consumption in modern office buildings. To reduce this load, we have designed a smart lighting control system that attempts to minimize power consumption, while simultaneously increasing occupant comfort, by dynamically accommodating heterogeneous illuminance requirements as well as changes in occupancy. Most current daylight-harvesting lighting systems measure lighting levels at the luminaires or at the walls to deduce illuminance on work surfaces. However, this computation is prone to error, which can potentially result in compromised user comfort. Instead, our system measures illuminance and occupancy directly from sensors located at each work station. It uses sensor readings to dynamically estimate the relationship between the dimming level of each luminaire and the illuminance at each work station using an unobtrusive calibration process. Subsequently, a linear-programming-based adaptive control algorithm determines power-efficient and comfort-preserving dimming levels for each luminaire. Plug-and-play design lets us seamlessly connect and disconnect system components, such as additional luminaires and sensing modules, even while the system is in use. Based on the deployment of our system in a real office environment, we demonstrate that it maintains the desired illuminance at work surfaces despite environmental fluctuations. We also show, through extensive simulations using 7 months of collected daylight and occupancy data, that our system substantially reduces energy consumption compared to even an occupancy-aware LED lighting system.

Acknowledgements

First, I would like to thank my advisor, Prof. S. Keshav, for his patient guidance, support and advice throughout my graduate research. Many thanks to Costin Ograda-Bratu for helping me to build and test the system prototype. I would also like to acknowledge Dr. Ansis Rosmanis as a contributor to this work, who came up with the unobtrusive calibration technique. I would like to thank Prof. Tim Brecht and Prof. Omid Abari for providing valuable insights on improving this thesis. Finally, I thank Aliya T. for the continued support through this journey.

Dedication

To my parents, grandparents and my younger brother, who have supported me unconditionally throughout my life.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction and Background

1.1 Introduction

Artificial lighting load accounts for approximately a third of overall electrical energy consumption in commercial office buildings [58, 52, 15, 38]. It is important to reduce this load and associated carbon footprint, ideally without compromising the comfort of building occupants. The illuminance level that must be provided by a lighting system depends on the occupancy of the room and individual preferences of occupants. Standard comfort levels require providing a specific illuminance level on work surfaces with typical values ranging between 300 and 500 lux [2, 38].

To reduce lighting energy consumption, beside the use of more energy-efficient luminaires such as LEDs [15], numerous lighting control methods have been designed and implemented. The luminous output of LED luminaires can be controlled easily and accurately. This, in combination with sensing technology, makes it possible to reduce lighting load by adapting the level of illumination to occupancy changes and availability of daylight. Many lighting systems capable of occupancy detection and daylight harvesting have been studied over the past two decades [39, 7, 5, 56, 22, 45]. Although these systems have shown that a reduction in energy of up to 61 % is feasible [45], many of them suffer from a potentially severe drawback. A study by Newsham et al. [30] suggests that the illuminance on a work surface is the main determinant of occupant comfort. Thus, lighting systems should place illuminance sensors close to these surfaces [52]. However, many existing systems measure the illuminance not on a work surface, but at a lighting fixture [39, 45, 5, 1] or at the walls [50] instead, and then deduce the illuminance on the desktop. This computation is necessarily flawed, thus potentially compromises user comfort. For example, if a sensor

mounted on a luminaire can sense daylight, but this daylight does not reach a work surface, then the luminaire would be dimmed inadvertently [49, 48, 17].

Several market adoption barriers prevent daylight-harvesting and occupancy-aware lighting systems from being widely used, despite a clear energy-saving potential. Among these barriers are difficulty in installation, deployment, and calibration, low return-on-investment, and integration with other building systems [3, 36].

In this thesis we present a power-efficient smart lighting control system that meets the following goals:

- Energy saving
- Personalized lighting comfort
- Robustness to modeling errors
- Responsiveness to changes in the environment
- Plug-and-play deployment

Our design relies on the key insight that rapidly declining sensor costs make it economically feasible to incorporate occupancy and illuminance sensors at each work station. With "per-desk" sensing, it is possible to realize energy reductions while ensuring user comfort.

To provide personal lighting comfort, the lighting level at each location in an office can be chosen by a user (or the set of users sharing a common space). To reduce power consumption despite occupancy and daylight changes, we periodically compute a *nearly optimal* dimming level of each luminaire using a linear program, whose objective is to minimize power consumption subject to user comfort requirements. Furthermore, to account for modeling and calibration errors, we use an iterative control algorithm that converges dimming levels so that desired illuminances are achieved at all work surfaces. We use off-the-shelf luminaires and sensing systems with a carefully-chosen software architecture to allow plug-and-play deployment.

Our evaluation in a realistic setting demonstrates that the proposed system reacts to changes in illuminance in under 2 seconds, and changes in occupancy - in about 350 milliseconds. We also find that it utilizes 40% less power compared to a conventional lighting system[1] that requires all luminaires to be fully on whenever the office is occupied, which is the typical use case in current buildings.

[1]Throughout this thesis, "conventional lighting systems" refers to LED-based systems that do not have occupancy and illuminance information and cannot control dimming levels of individual luminaires.

Our work makes the following contributions:

- We have modeled and designed a power-efficient lighting control system that exploits daylight and occupancy information to minimize energy consumption while satisfying individual lighting preferences of all occupants.
- We have implemented the system with off-the-shelf components and deployed it in a realistic environment. A plug-and-play design makes deployment rapid and cost-effective, as well as allows seamlessly connecting and disconnecting system components, such as additional luminaires and sensing modules, even while the system is in use.
- We have performed extensive laboratory experiments to study the performance of our system in a typical office setting under changing daylighting, occupancy and occupants' lighting preferences.
- We have conducted an extensive evaluation using 7 months of collected daylight and occupancy data to demonstrate the system's ability to reduce energy consumption.

The rest of this thesis is structured as follows. Section 1.2 presents background on lighting comfort and an overview of prior work. In Chapter 2, we formulate a mathematical model and propose the design of our smart lighting system. Implementation details are presented in Chapter 3. In Chapter 4 we evaluate the system's robustness and responsiveness to changes in environmental lighting, occupancy and users' illuminance preferences, as well as estimate the reduction in energy consumption. Finally, in Chapter 5, we conclude the thesis.

1.2 Related Work

This section discusses related work.

1.2.1 Occupant Preferences for Lighting

What makes occupants of an office happy about their lighting? The answer to this question gives us some important clues that we use in our design. We begin by discussing research that addresses this issue.

Newsham et al. [30] investigated how well various metrics correlate to occupant satisfaction with office lighting. Their study featured two adjacent, identical rooms with similar illumination that had south-southeast facing windows. Only one of the two rooms was occupied. A person working in that room was given the chance to change lighting levels in this room regularly, which identically adjusted lighting levels in both rooms. Meanwhile, the second room was equipped with photometric equipment measuring illuminance and luminance at multiple locations. As the occupant was performing various office tasks, lighting data in the second room was collected.

The authors observed that illuminance *measured on the work surface* was the best predictor of whether participants were satisfied with their lighting level. Particularly, illuminance measured at the ceiling was a substantially worse predictor. They also observed that people are not sensitive to minor lighting changes, often choosing not to adjust to changed lighting conditions. Hence, automated lighting systems could employ a dead-band range (i.e., a range of lighting levels within which no control actions are taken) around the target illuminance without disturbing occupants. Another important finding of this work is that preferred illuminance levels differ both among different people and for the same person throughout the day.

Newsham et al. also studied occupants in an open-plan office laboratory [32, 31] and demonstrated that occupants whose preferred light levels were met had significant improvements in mood, productivity, and comfort. This result has been corroborated by Lashina et al. [24], who showed that giving light control to occupants in multi-occupancy offices results in increased occupant satisfaction, compared to the case with a fixed light level. Galasiu et al. [18] also concluded that fully automated systems have low acceptance among occupants. The work demonstrated that acceptance becomes higher when users are provided with at leat partial control of their lighting system. This result has guided our work in allowing building occupants to specify their desired lighting level at multiple points within their working area.

It has been found that occupants strongly prefer daylight to artificial lighting [18]. The presence of daylight improves lighting quality, positively affects psychological comfort and overall health. Indeed, some of the known benefits of daylight include stress reduction, increased productivity [19, 54] and human circadian rhythms regulation [14]. Thus, in our work, we design a system that co-exists with natural lighting, compensating only when natural lighting is insufficient to provide adequate lighting comfort.

To sum up, a well-designed lighting system measures illuminance on (or near) work surfaces, allows users to control their lighting conditions, and works synergistically with natural lighting. These results inform our design.

1.2.2 Recent Surveys on Lighting Control

Much research has been done on various lighting control approaches in office buildings in the past decade. In this section, we discuss prior work on smart lighting systems, relying on several recent surveys of this area. We note that, despite the existence of a considerable body of research, we are not aware of a system, other than our own, which meets our design criteria of power-optimality, personalized lighting comfort, robustness to estimation errors, fast response time, and plug-and-play deployment.

A survey by Haq et al. [52] classifies lighting control systems into three categories, namely, timers, occupancy-based controls, and daylight-linked controls. The authors discuss various trade-offs that one has to consider when designing an automated lighting system. For example, depending on circumstances, one has to choose between switching and dimming, between open-loop and closed-loop daylighting, and among various occupancy sensing technologies.

Pandharipande and Newsham [36] discuss the evolution of lighting control strategies, particularly focusing on open-plan office applications. The authors review and compare two state-of-the-art lighting system architectures: a wireless distributed lighting control system and a centralized power-over-Ethernet system with cloud connectivity. The difficulty of lighting control commissioning, lack of methods to evaluate the return on investment and challenges related to integration with other systems are indicated as the main barriers of smart lighting. In our work, we attempt to mitigate these issues.

In another survey, Williams et al. [58] did a meta-analysis of 240 energy saving estimates from 88 papers and case studies. They report that lighting controls can reduce, on average, 24% of energy use by using occupancy detection, 28% using daylighting, and 38% when combining these and additional methods. Their analysis excluded savings that are due to more efficient light and other aspects not directly related to control techniques. They also excluded savings reported in simulations, as they found that, in comparison to actual installations, simulations tend to highly over-estimate potential energy savings, especially for daylighting (i.e., the use of natural lighting). Thus, in our work, we do real-world evaluation, rather than simulations. The savings from our system are in line with these results.

A more recent comparative analysis by Chew et al. [14] also argues that once installed in real buildings, most daylight harvesting systems do not achieve the expected energy savings. According to the authors, daylight harvesting effectiveness depends on multiple factors, including the presence of shading devices, geographical location, building orientation, as well as geometry and reflectance of the office. Thus, it is generally suggested that the

performance of lighting systems should be validated through real field measurements, rather than just simulations alone [58, 14, 15]. We note that in our work, in the spirit of this conclusion, we use real field measurements.

A thorough analysis of why the use of daylight-linked control of lighting systems is limited, despite all potential benefits, was conducted by Bellia et al. [3]. The difficulty related to design, calibration, and installation of daylight-linked control systems is indicated as one of the main factors preventing their spread. Problems related to evaluating realistic energy savings and determining the payback period is another important reason, which is in agreement with Ref. [36]. Our design goals of plug-and-play deployment (which includes self-calibration) attempts to address this issue.

Finally, a study by Ding et al. [16] concludes that a general tendency in smart lighting research increasingly emphasizes occupants' visual comfort, in addition to energy efficiency. We do not address visual comfort in our work, leaving this to future work.

1.2.3 Automated Lighting Strategies in the Literature

We now focus on work that is closely related to ours, in that it uses a control system for lighting control. For each such system, we discuss where it fails to meet our design goals.

Perhaps the earliest work in this area was by Rubinstein et al. [47], who gave an early demonstration of a closed-loop daylight-harvesting system based on illuminance sensing. Many automated lighting systems since then have been proposed, using a variety of control methods and sensing approaches. Although this work is more than 25 years old as of the time of this writing, its insights are still relevant today.

A number of *open-loop* control systems are presented in the literature. For example, Wen and Agogino [56] present a linear optimization-based lighting system for open-plan offices that provides each occupant with preferred task lighting while improving energy efficiency. However, open-loop control systems cannot perform daylight harvesting because daylight levels can vary rapidly. Therefore, such systems are only suitable for offices with no windows. In our work, we used closed-loop control to allow daylight harvesting.

Closed-loop control

Closed-loop lighting strategies based on a mathematical formulation can be classified into three groups: logic-based controllers, regulation-based controllers, and optimization-based controllers [20]. Logic-based methods utilize decision-making techniques to determine actions of the lighting system based on the illuminance measurements [28]. Regulation-based strategies strive to achieve closed-loop stability while matching generated illuminance

to a reference value [48, 49, 53, 1, 23, 6]. In optimization-based methods, the control problem is formulated as an optimization problem [59, 57, 9, 38, 10, 7]. Our work lies in the third category.

Optimization-based approaches

Other researchers have also used an optimization framework to exploit non-occupancy and presence of daylight. One of the control strategies that is suitable for open-plan offices is to adjust individual luminaires' intensities to minimize power consumption while supplying some minimum required illumination to all occupied regions. Some researchers modeled this optimization as a linear programming problem [59, 11, 38]. These studies mainly focus on a theoretical framework for the optimization problem, assuming that illuminance contributions of both artificial luminaires and daylight on the workspace surface are known. Another study where the knowledge of illuminance levels is assumed was conducted by Boscarino et al [6], who developed a lighting strategy based on self-tuning multivariable controller. In contrast, our approach uses both theory and implementation; we also do not assume knowledge of daylight contribution to illuminance.

Luminaire-based light sensors

Many studies consider a lighting system where illuminance and occupancy sensors are embedded in luminaires, and each luminaire corresponds to a specified work zone [39, 37, 35, 1, 8, 53, 48, 10]. With this configuration, the system is initially calibrated during the night time, mapping illuminance on work surfaces to illuminance on corresponding ceiling-based luminaire sensors. Then, the ceiling-based sensors are used for control. The advantages of such design include simplified sensor commissioning through sharing a power supply between luminaire and sensors. For example, Pandharipande et al. [35] and van de Meugheuvel et al. [53] proposed regulation-based control strategies (though they focus on achieving desired illuminances on desks, and not on minimizing power consumption). However, their approach has not been validated using experiments.

In other work, Koroglu et al. [23] propose a regulation-based distributed control method for a lighting control system with only local communication between luminaires. Their approach claims to address problems of cross-illumination and daylight disturbances. However, it has been found that under high cross-illumination between luminaires, this approach fails to consistently achieve uniform illumination and track the desired illuminance level [20]. Similarly, a study by Afshari et al [1] presents a decentralized feedback control for a system that has spectrally tunable luminaires with co-located color sensors and with no communication between luminaires. This work assumes that measurements on each light sensor are mainly due to the fixture's own light source, precluding the use of daylight harvesting. Moreover, this requires special luminaires, so is not plug-and-play.

Studies by Caicedo et al. [8, 12] present optimization-based control strategies for systems with co-located luminaires and light sensors. They implicitly assume knowledge of the daylight contribution at light sensors by using specialized software to simulate daylight distribution. In practice, isolated daylight contribution is unknown, as light sensors measure illuminance from a combination of daylight and artificial lighting. Moreover, simulation of light paths requires detailed modeling of offices, which precludes plug-and-play deployment.

We note that *all* luminaire-based sensor configurations have a potentially severe drawback. While a calibration step allows a reasonable estimation of illuminance contribution on work surfaces due to artificial luminaires, estimation of daylight contribution is prone to error. For example, Pandharipande and Caicedo reviewed two control strategies for the lighting system with luminaire-based photosensors: proportional integral (PI) control and linear optimization [39]. Their benchmarking shows that, under high daylight penetration, both approaches, in comparison to an optimization system that had full knowledge about illuminances on the work surface, under-illuminate the office space. Similarly, the studies by Rubinstein et al. [49, 48] and Doulos et al. [17] demonstrate that ceiling-to-working plane illuminance ratio is not constant throughout the day due to its dependence on the variability of daylight distribution, with the most consistent correlation observed in the regions further away from windows. We conclude that, in general, a ceiling-based photosensor is not able to track illuminance at the working surface accurately. Besides, depending on the field of view of light sensors, accurately detecting cross-illumination on work-surfaces might be a challenge for the lighting system with luminaire-based photosensors.

This motivates the use of desk-based light sensors, discussed next.

Desk-based light sensors

Several researchers have studied using desk-based light sensors to prevent under-illumination due to luminaire-based sensors. In particular, a decrease in under-illumination was demonstrated in the methods presented in References [7] and [4, 5]. Unfortunately, both of these methods have certain limitations, discussed next.

Caicedo et al. [7] placed additional wireless light sensors at each work surface that provide low-rate feedback to luminaire-based sensors. However, since additional light sensors are subject to blockage and occlusion, this method does not provide a significant advantage over the design that places light sensors near the work surface. Also, as experiments conducted by the authors showed, the response of the proposed system to changes in environmental illuminance is quite slow with the adaptation to changes in environmental illuminance taking around 100 seconds. This is primarily due to the low transmission rate of work surface-based light sensors. Moreover, the system imposes an additional im-

plicit constraint that the number of work surface-based light sensors and the number of luminaires has to be the same.

In their work, Borile et al. [4, 5] proposed an interesting data-driven approach for determining the linear mapping between measurement points on the ceiling and points of interest at the work surface. This approach requires collecting sensor measurements from both work plane-based and luminaire-based sensors during the daytime when the office is not occupied. Then, the collected data is used to learn the daylight mapping. One practical limitation of the proposed method is that system re-calibration takes a lot of time, as it requires the collection of a new training data set. Also, the proposed method does not guarantee good performance under different weather conditions and varying amounts of daylight. For instance, the daylight mapping that was obtained based on the training data collected on cloudy days would potentially perform poorly on a sunny day. Finally, data-driven approaches like the proposed method should be validated with real experiments, in addition to a simulation, which was not done.

Closely-related work

We now discuss the work that is closest to ours in that it uses wireless illuminance sensors near work surfaces without associating luminaires one-to-one with work areas [57, 29, 50]. An important advantage of such a design choice is that due to the explicit knowledge of illuminance near the work surface, such systems are usually capable of finding optimum, or nearly-optimum dimming levels, achieving target illuminances at all work surfaces. However, as we discuss, prior work makes some assumptions that precludes practical deployment.

Miki et al. [29] present a distributed lighting control strategy that utilizes infra-red communication between neighboring luminaires. This requires special-purpose luminaires, so is not plug-and-play.

Similar to our work, Wen and Agogino [57] propose an energy-efficient linear optimization-based lighting control. However, to generate an illuminance model and determine artificial light distribution in the office, the authors use image rendering program RADIANCE [44], which is based on backward ray tracing. This illuminance model generation step requires explicit knowledge of several office parameters, such as office dimensions, internal surface reflectance, locations and geometries of furniture and other objects, as well as luminaire parameters and locations. This precludes practical deployment of their approach.

In their work, Tan et al. [50] employ a PI controller and measure the illuminance near work surfaces. In addition to savings from automatic control, they also study what savings can be achieved through the use of more efficient lamps and direct current (DC) microgrids. Some practical concerns related to their work are that illuminance sensors placed directly

on work surfaces can be occluded or shadowed. In our work, we place sensors on the edges of shelves or monitors, preventing occlusion.

Yeh et al. [60] and Pan et al. [34] present a system for daylight harvesting, assuming that locations of office occupants are known and that they carry wireless light sensors on their mobile phones. Lighting control strategies based on linear programming and sequential quadratic programming algorithms were proposed to satisfy individual illumination requirements of the users, based on their activities. Again, this approach suffers from occlusion of sensors. Moreover, not all office occupants may be willing to have sensing software installed on their personal mobile devices.

Matta and Mahmud [28] present an ad hoc daylight harvesting method. However, it relies on only one illuminance sensor inside the room (at the ceiling level), so it is only suitable for small, single-occupant offices. In addition to this sensor, a daylight harvesting sensor is placed next to the window, which automatically controls a Venetian blind. We do not consider blind control in our work.

AI-based approaches

We note, in conclusion, that in recent years, artificial intelligence-based algorithms have been proposed for smart lighting. For example, Madias et al. [27] apply an evolutionary multi-objective genetic algorithm to minimize energy consumption and maximize illumination uniformity subject to task illuminance requirements at the desks. Tran et al. [51] and Wang et al. [55] present a sensorless lighting system, where the relationship between luminaire intensities and illuminance levels on work surfaces in a multi-occupancy office setting was modeled using an artificial neural network (ANN). While being able to successfully minimize the power consumed by the luminaires, these approaches do not take into consideration environmental illuminance and daylighting. Moreover, this data-driven approach requires collection of a training data set with a varying number of active lights, dimming levels, and the daylight intensity, which precludes plug-and-play deployment. In addition, based on the provided experimental results, it takes the system around 200 seconds to respond to changes in environmental illuminance, which can potentially compromise the comfort of the occupants.

Chapter 2

System Design

2.1 Problem Formulation

Consider an office that has N workspaces, such as the one illustrated in Figure 2.1. The office has two sources of lighting: daylight coming from one or more windows, and a set of M dimmable LED luminaires. Light and occupancy sensors placed at each work station provide per-desk illuminance and occupancy sensing. The office can be multi-occupancy or single-occupancy with several workspaces belonging to the same person. In addition, occupants can adjust illumination levels at their work stations, when desired, allowing personalization.

The goal of a smart lighting system is to provide the desired level of illuminance at each of the work stations while minimizing energy consumption through the exploitation of daylight and occupancy information. Thus, the system strives to dynamically accommodate heterogeneous illuminance requirements, taking full advantage of daylight and resorting to artificial lighting only to compensate for insufficient daylight. The sensors installed at each work station communicate occupancy status and illuminance level of their respective work station to the central controller, which, in turn, determines nearly optimal dimming levels for all individual luminaires, and sends them control signals.

In response to the analysis by Bellia et al.,[1] an additional requirement imposed on the system is a flexible plug-and-play design that can be easily deployed in all types of spaces without requiring any manual calibration, regardless of the geometry and configuration of the room.

[1]The study by Bellia et al. [3] concludes that the main barrier that prevents daylight-linked control systems from becoming widely applied is a difficulty in calibration, installation and deployment.

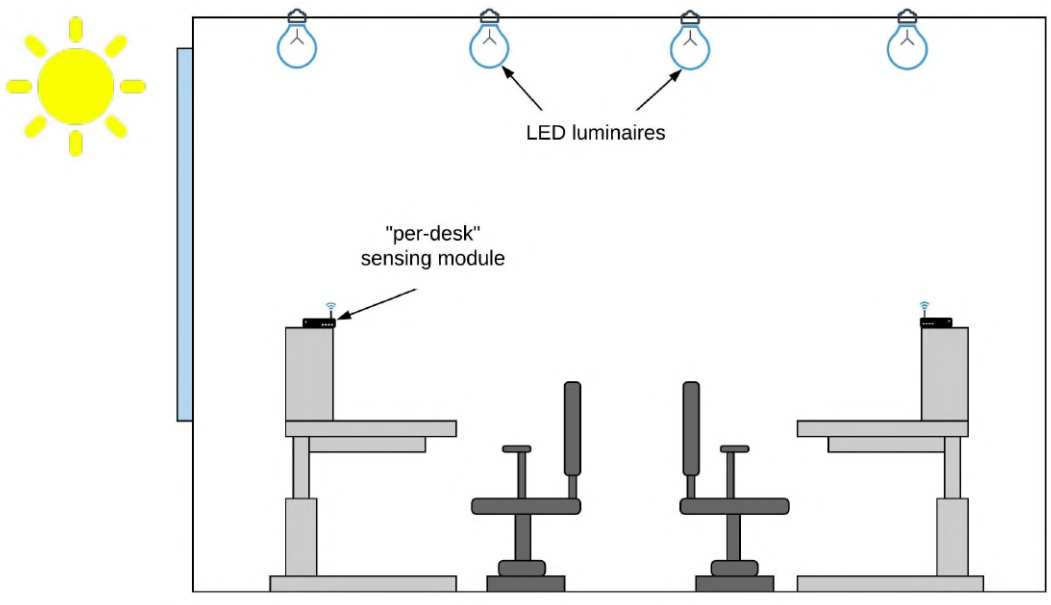


Figure 2.1: Illustration of the office.

2.2 Luminaire Model

We assume that an office is lit by a combination of environmental lighting, such as daylight, and dimmable LED luminaires, and we only have control over the latter.[2] This section develops a mathematical model of PAR38 Philips Hue LED bulbs [41], used in our study. A similar model can be developed for other commercially-available luminaires.

Let $A_{ij}(t)$ be the illuminance gain of sensor i from a fully-lit luminaire j at time t. We assume that $A_{ij}(t)$ is time-dependent because it may be affected by objects or people between sensor i and luminaire j, as well as by slight accidental movements of sensors and changes in the luminaires' temperatures [25]. Let $d_j \in [0.0; 1.0]$ be a *dimming level* of the jth luminaire. In this work we define dimming level as a measure of the bulb's relative luminous power output, linearly scaled with illuminance. In other words, if $L_{ij}(d_j, t)$ is the illuminance gain of sensor i from a luminaire j whose dimming level is d_j at time t, then

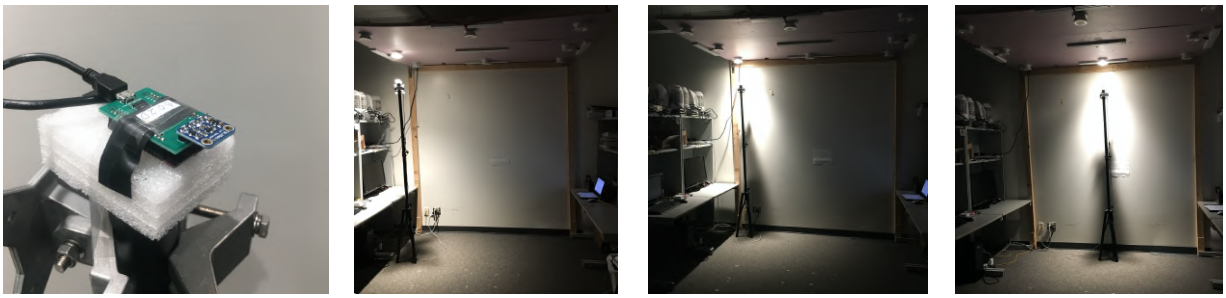
$$d_j = L_{ij}(d_j, t)/L_{ij}(1.0, t) = L_{ij}(d_j, t)/A_{ij}(t) \tag{2.1}$$

Therefore, $L_{ij}(d_j, t) = d_j A_{ij}(t)$. Thus, when a bulb is completely off, $d_j = 0$ and

[2] Some studies [28, 22, 45] also consider an automatic control of daylight using motorized blinds.

$L_{ij}(0,t) = 0.0$. On the other hand, when it is fully lit, $d_j = 1.0$ and $L_{ij}(1,t) = A_{ij}(t)$.

2.2.1 Dimming Level vs. Brightness Control Value for a PAR38 Philips Hue Bulb



(a) (b) (c) (d)

Figure 2.2: Experimental setup for measuring the relationship between the dimming level and the brightness control value for PAR38 Philips Hue bulbs. a) Light sensing module that measures illuminance. b-d) Experiments conducted for different PAR38 Philips Hue luminaires.

Unfortunately, the Philips Hue API [40] does not allow us to control dimming levels directly. Instead, it only allows turning a luminaire on and off, as well as setting its *brightness control value* to an integer between 0 and 255. The brightness control value on luminaire j is denoted as $b_j \in \{0, 1, ..., 255\}$.

The mapping between brightness control value and dimming level, $d_j(b_j)$, was determined empirically through experiments that were conducted for 8 different PAR38 Philips Hue bulbs. The experimental setup is illustrated in Figure 2.2. In each experiment, a light sensor was placed directly under one of the luminaires. Next, as the luminaire's brightness control value was gradually increased, illuminance readings on the light sensor were recorded. Throughout each experiment, lighting in the office was supplied only by a luminaire being evaluated, i.e., all other luminaires were completely off and there was no environmental illumination in the office. Finally, measured illuminance values were converted into dimming levels, according to Equation 2.1.

Experimental results, shown in Figure 2.3, were very consistent across all experiments. An analytic relationship between the dimming level and the brightness control value was

determined by fitting a curve to the experimental data points. Note that when the brightness control value is 0, a bulb's dimming level is 0.047, i.e. a bulb's luminous output power is at 4.7%. The dimming level becomes 0 only when a bulb is completely turned off. Thus, the $d_j(b_j)$ relationship can be written as follows:

$$d_j(b_j) = \begin{cases} 2.64 \cdot 10^{-5} \cdot b_j^{1.90} + 0.047 & \text{if bulb j is on and } 0 \leq b \leq 255 \\ 0 & \text{if bulb j is off} \end{cases} \tag{2.2}$$

This equation provides a 1:1 mapping between the dimming level and the brightness control value. Unlike the Philips Hue brightness control value, however, the dimming level has a clear physical meaning, which is useful for modeling the optimization problem, presented in Section 2.4.

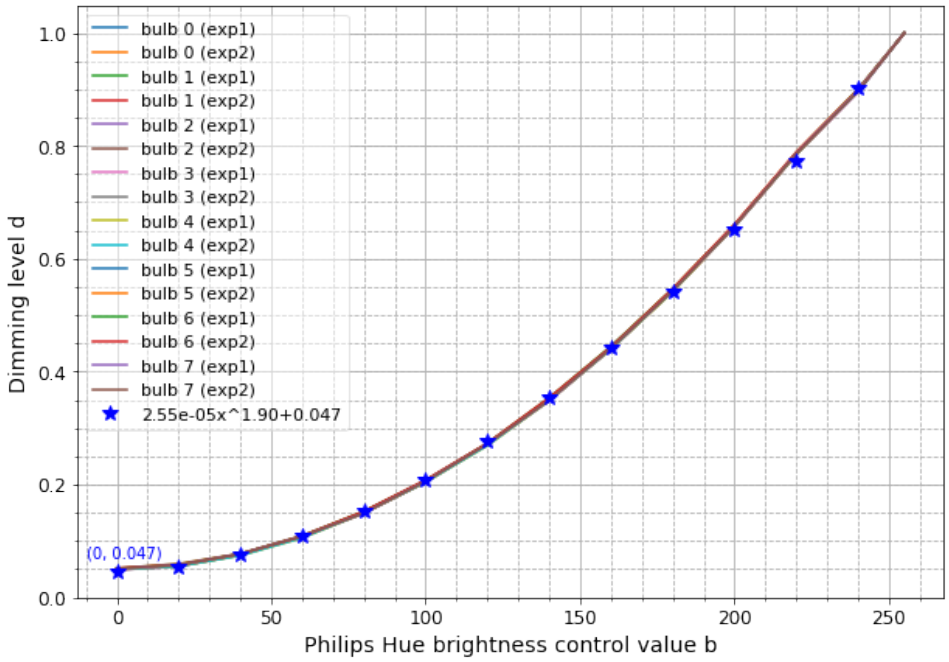


Figure 2.3: Empirically obtained relationship between dimming level d_j and brightness control value b_j of a PAR38 Philips Hue bulb, and the corresponding best-fit curve.

Note that, as the temperature of luminaire j increases, its luminous power output slightly decreases[3] [25] and consequently, illuminance contributions of a fully lit luminaire

[3]This has been empirically confirmed by an experiment presented in Figure A.1 in Appendix A.

j on all sensors, represented by the jth column of matrix A, also proportionally decrease. At the same time, dimming level d_j, being a measure of *relative* luminous output (see Equation 2.1), stays the same because both $L_{ij}(d_j, t)$ and $L_{ij}(1, t)$ decrease by the same fraction. Since both the brightness control value b_j and the dimming level d_j are independent of the luminaire's temperature, the relationship in Equation 2.2 is also temperature-independent.[4] Thus, a decrease in illuminance on light sensors caused by the increase in the bulbs' temperature is fully captured by the *illuminance gains matrix* $A(t) = (A_{ij}(t))$.

2.2.2 Power Consumption vs. Dimming Level for a PAR38 Philips Hue Bulb

Recall that the goal of the smart lighting system is to find a dimming level configuration that minimizes the energy consumption of the system. This task requires the knowledge of the relationship between luminaires' power consumption P_j and dimming level d_j. Conventionally, it is assumed in the literature [38, 56] that the electric power consumption of an LED bulb is linearly proportional to its dimming level. Our laboratory experiment conducted for PAR38 Philips Hue bulbs is mostly in agreement with this assumption, with a couple of remarks.

First, we have observed that similarly to the luminous power output, the electrical power consumption of PAR38 Philips Hue bulbs depends on their temperature. To investigate this dependence, after turning a bulb on at time $t_0 = 0$, at which point it is at room temperature, we measure how much the amount of drawn power changes over time ($\sim$70 minutes) as the bulb gets warmer. Our findings are shown in Figure 2.4, from which one can see that, over 50 minutes, the power consumption decreases by about 3%, and then it becomes steady.

Note that, since the dimming level does not depend on the luminaire's temperature, while the power consumption does, the relationship between power and dimming level is temperature-dependent. In our work, we use a steady-state estimate of the power vs. dimming level relationship to model the luminaires. Thus, the experiments to estimate this relationship are run after the bulbs are fully lit for 70 minutes, to ensure that power consumption stabilizes. This can introduce an error of up to about 3% in determining the optimal power level.

As Figure 2.5a indicates, the power vs. dimming level relationship is nearly linear for the experimental data points corresponding to the turned on bulb with brightness control values $b_j \in \{0, 1, ..., 255\}$. On the other hand, a slight non-linearity is observed when the

[4] This is demonstrated in an experiment shown in Figure A.2 in Appendix A.

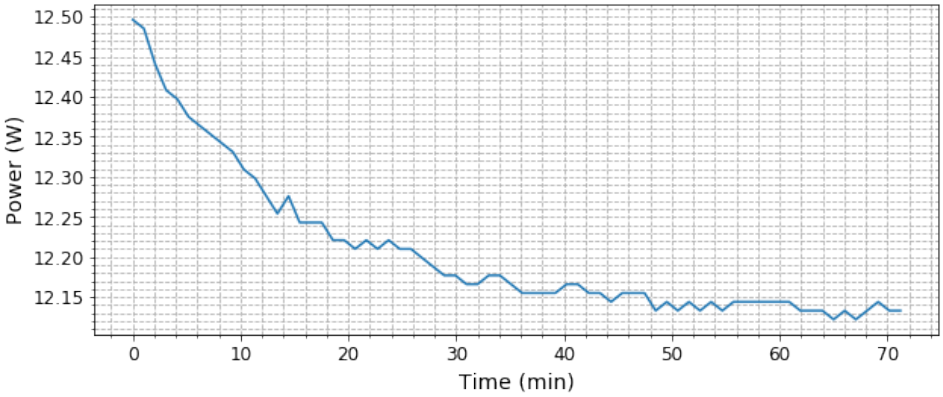


Figure 2.4: The reduction of power consumption of a fully lit PAR38 Philips Hue LED luminaire, as it heats up. At time $t_0 = 0$ the luminaire is at the room temperature. (Note that the vertical axis does not originate at 0.)

bulb is turned off and the dimming level is 0.[5] It should also be pointed out, that the bulb has a standby power draw of 1.18 W, which is consumed as long as it is connected to a power supply. This standby power consumption is due to the Zigbee protocol [61] that most wireless Philips Hue products use. Taking into consideration the aforementioned non-linearity, the power vs. dimming relationship can be written as:

$$P_j(d_j) = \begin{cases} 1.18 & \text{if } d_j = 0.0 \\ 9.97d_j + 2.47 & \text{if } 0.0 < d_j \leq 1.0 \end{cases} \tag{2.3}$$

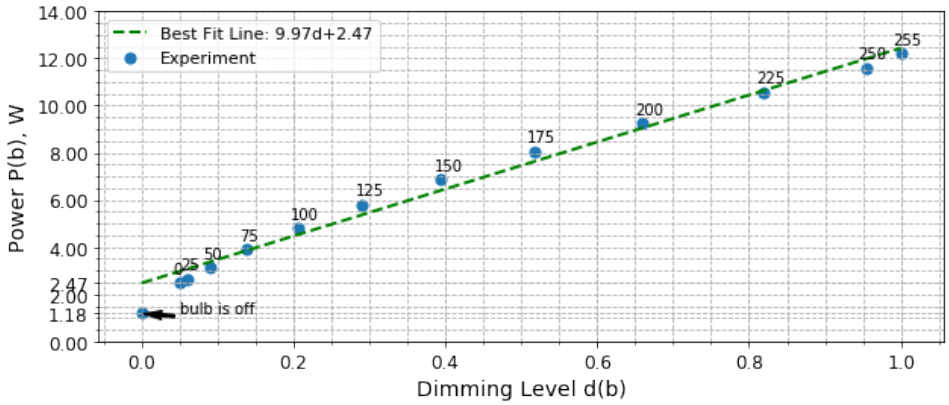
where $9.97d_j + 2.47$ is the best fit line to the data points for the turned on bulb.

For modeling convenience, we further relax this relationship by fully linearizing it, as illustrated in Figure 2.5b. This is achieved by making the line pass through the standby power value of 1.18 W at the dimming level of 0,[6] followed by the best-fit-line behavior for the rest of the dimming level range $(0.0; 1.0]$. If we denote the standby power as $P_{standby} = 1.18\ W$ and the effective power as $P_{effective} = 11.83\ W$, the relaxed model for Philips Hue power consumption can be written as:

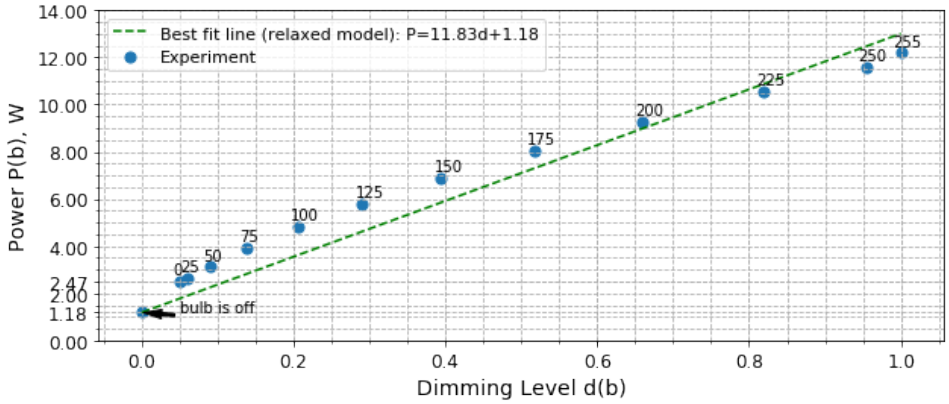
$$P_j(d_j) = P_{effective}d_j + P_{standby} = 11.83d_j + 1.18 \tag{2.4}$$

[5]The observed non-linearity indicates that the bulb requires some extra activation power to stay on, even if its dimming level is negligibly small.

[6]Through this requirement we model the power consumption of switched off bulbs accurately, which is important because switched off state is typically the most common state of a bulb throughout a day.



(a) A line is fitted to data points corresponding to a turned on bulb.



(b) A line is fitted to data points with an additional constraint that power equals to the standby power at the dimming level of 0.

Figure 2.5: Empirically obtained relationship between power $P_j(b_j)$ and dimming level $d_j(b_j)$ of a PAR38 Philips Hue bulb. Each experimental data point is labeled with a corresponding Philips Hue brightness control value b_j.

As one can see from Figure 2.5b, the effect of this linearization is slight over-estimation (up to $0.75W$) of power consumption for high dimming levels, and under-estimation (up to 1.25 W) for lower dimming levels.

Lastly, as mentioned above, when developing the luminaire model we used power measurements of the fully heated bulbs. However, in practice, the bulbs are not always fully heated. For example, the bulb cools down if it is off for a long time, or it might not get fully heated when operated at low dimming levels. Even though our model indeed further under-estimates power consumption when the bulb is not fully warmed, from Figure 2.6 one can see that this under-estimation is insignificant in magnitude, compared to the effect of linearization of the power vs. dimming level relationship.

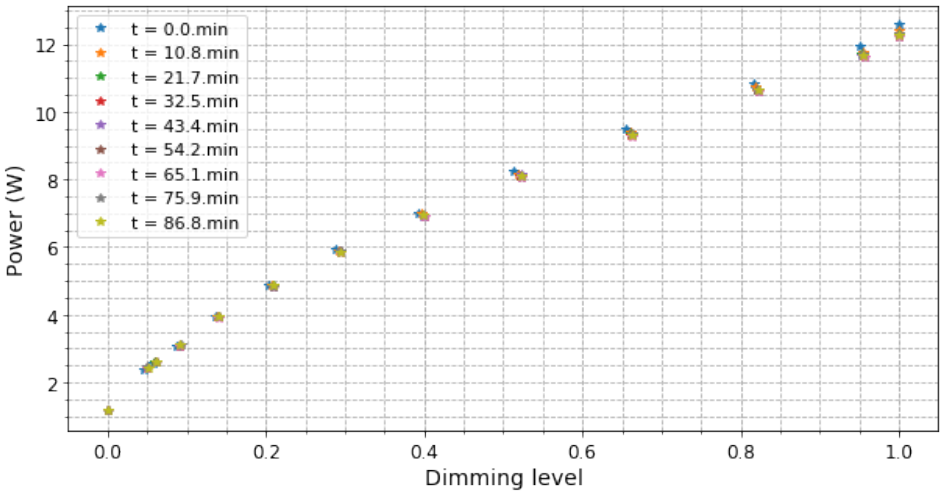


Figure 2.6: This experiment demonstrates how the power vs. dimming level relationship changes, as a PAR38 Philips Hue bulb heats up. At time $t_0 = 0$ the luminaire is at room temperature.

2.3 Calibration

We now discuss how to calibrate our system.

2.3.1 Illuminance as a Linear Combination

For sensor i, let $R_i(\boldsymbol{d}, t)$ be its illuminance at time t when the luminaire dimming levels are $\boldsymbol{d} = [d_1, ..., d_M]$, where M is the total number of luminaires. We also refer to $\boldsymbol{d}$ as the *dimming vector*. According to the additivity of light [48, 39], the total illuminance level at any point is a linear combination of illuminance contributions of all light sources. Thus, taking into consideration Equation 2.1, we can write

$$R_i(\boldsymbol{d}, t) = E_i(t) + \sum_{j=1}^{M} L_{ij}(d_j, t) = E_i(t) + \sum_{j=1}^{M} d_j A_{ij}(t) \tag{2.5}$$

where $E_i(t)$ is the illuminance gain of sensor i from the environment, which comprises of daylight and other environmental light sources out of our control. Similar to $A_{ij}(t)$, $E_i(t)$ is also time-dependent.

To evaluate daylight distribution at work stations, the literature on daylight harvesting commonly places daylight sensors near windows [28, 57] or infers $E_i(t)$ from illuminance levels at the ceiling [45, 39, 38]. We propose a novel method to estimate both environmental gains $\boldsymbol{E}(t) \in \mathbb{R}^{N\times 1}$ and the illuminance gains matrix $A(t) \in \mathbb{R}^{N\times M}$ without distracting occupants of the room. This unobtrusive calibration method is described next.

2.3.2 Calibration Procedure for Matrix A

We start by obtaining the illuminance gains matrix $A(t)$. Recall that $A_{ij}(t)$ is the illuminance gain of sensor i from a fully lit bulb j ($d_j = 1.0$). Therefore, one straightforward way to obtain $A(t)$ is to first measure environmental illuminance gains by getting light sensor readings while all luminaires are off. Next, we could sequentially turn on one bulb at a time, record new illuminance readings, and subtract respective environmental illuminance gains from these readings. Even though this calibration process allows us to estimate the matrix $A(t)$, it is obtrusive, and thus cannot be performed when the system is in use. To address this, we developed an unobtrusive calibration method, based on the observation that, while a human eye is insensitive to minor lighting changes [30], photosensors are capable of detecting them accurately. The method is described next.

Suppose that, just before calibration, luminaires are dimmed according to $\boldsymbol{d} \in [\mathbf{0}; \mathbf{1}]$. Let $\hat{\boldsymbol{d}}^{(j)}$ be the dimming level vector with all the entries but j-th equal to those of $\boldsymbol{d}$, and

$$\hat{d}_j^{(j)} = \begin{cases} d_j + S & \text{if } d_j < B \\ d_j - S & \text{if } d_j \geq B \end{cases} \tag{2.6}$$

where, in our experiment, we use the *dimming step* $S = 0.1$ and the pivot point $B = 0.65$.

We estimate $A(t)$ as follows. First, we record illuminance readings $R_i(\boldsymbol{d}, t)$. Immediately after, sequentially for every luminaire j we do the following. We change the dimming level of luminaire j to $\hat{d}_j^{(j)}$, record illuminance readings $R_i(\hat{\boldsymbol{d}}^{(j)}, t')$ for all sensors i, and then restore the brightness of the luminaire to its original value d_j, and proceed to the next luminaire. We ensure that t and t' are at most a few seconds apart, and we assume that neither daylight nor the illuminance gains matrix change much on such a timescale.

Hence, considering that $A_{ij}(t) \approx A_{ij}(t')$ and $E_i(t) \approx E_i(t')$ and using Equation 2.5 we get

$$R_i(\boldsymbol{d}, t) - \sum_{j=1}^{M} d_j A_{ij}(t) = R_i(\hat{\boldsymbol{d}}^{(j)}, t') - \sum_{j=1}^{M} d_j^{(j)} A_{ij}(t')$$

$$|\hat{d}_j^{(j)} - d_j| A_{ij}(t) = S A_{ij}(t) = |R_i(\hat{\boldsymbol{d}}^{(j)}, t') - R_i(\boldsymbol{d}, t)|$$

$$A_{ij}(t) = \frac{|R_i(\boldsymbol{d}^{(j)}, t') - R_i(\boldsymbol{d}, t)|}{S} \tag{2.7}$$

for all sensors i and luminaires j.

Values of S and B are open to further tuning. Sizing of S provides a trade-off between precision and unobtrusiveness of the calibration. Note that this calibration procedure allows estimating the illuminance gains matrix without requiring explicit knowledge of office geometry and locations of luminaires and photosensors, thereby contributing to the system's plug-and-play design.

2.3.3 Inference of Environmental Illuminance Gains E

Because we obtain columns of the illuminance gains matrix one after another in quick succession, we assume that its values, as well as environmental illuminance gains, do not change drastically during this procedure. Thus, knowing total illuminance values $R_i(\boldsymbol{d}, t)$

on all sensors i, dimming level settings d_j on all luminaires j, and illuminance gains $A_{ij}(t)$, we can estimate the environmental illuminance gains $E_i(t)$ from Equation 2.5 as

$$E_i(t) = R_i(\boldsymbol{d}, t) - \sum_{j=1}^{M} d_j A_{ij}(t) \tag{2.8}$$

or, in a vector form,

$$\boldsymbol{E}(t) = \boldsymbol{R}(\boldsymbol{d}, t) - A(t)\boldsymbol{d} \tag{2.9}$$

2.3.4 Estimation Error for A and E

The calibration Equation 2.7 demonstrates how one can obtain $A_{ij}(t)$ using two consecutive illuminance readings from sensor i, with one of the readings measured when dimming level of bulb j is shifted by S. The term $A_{ij}(t)$ is a time-varying ground-truth illuminance contribution value on sensor i from luminaire j, which assumes perfect sensor measurements and no environmental fluctuations. At the same time, the proposed calibration process gives us only an *empirical estimate of a snapshot of matrix A at the calibration time*, which we denote as $\widetilde{A} \in \mathbb{R}^{N \times M}$. $\widetilde{A}$ is not a function of time and is subject to estimation errors.

Let $\epsilon(t) \in \mathbb{R}^{N \times M}$ be an *estimation error*, defined as the difference between the estimated and the true illuminance gains matrices:

$$\epsilon(t) = \widetilde{A} - A(t) \tag{2.10}$$

The estimation error $\epsilon(t)$ varies with time, because, while $\widetilde{A}$ is constant, $A(t)$ is time-dependent. Thus, $\epsilon(t)$ captures both calibration errors and estimation errors caused by the time-varying nature of $A(t)$.

Likewise, Equation 2.9 uses the true matrix $A(t)$ to estimate the environmental illuminance contribution on sensors. However, in practice, only the estimated matrix $\widetilde{A}$ is available through the calibration process. By combining Equations 2.9 and 2.10, we get

$$\boldsymbol{R}(\boldsymbol{d}, t) = \boldsymbol{E}(t) + (\widetilde{A} - \epsilon(t))\boldsymbol{d} \tag{2.11}$$

which is the equation for additivity of illuminance, expressed in terms of $\widetilde{A}$.

Since the error term $\epsilon(t)$ in the equation above is unknown, environmental contribution $\boldsymbol{E}(t)$ cannot be calculated directly. We can only estimate it as:

$$\widetilde{\boldsymbol{E}}(t) = \boldsymbol{E}(t) - \epsilon(t)\boldsymbol{d} = \boldsymbol{R}(\boldsymbol{d}, t) - \widetilde{A}\boldsymbol{d} \tag{2.12}$$

where $\widetilde{\boldsymbol{E}}(t)$ is an estimate of the environmental illumination at time t, and $-\epsilon(t)\boldsymbol{d}$ is the associated estimation error in the environmental illumination.

Note that, as Equations 2.10 and 2.12 indicate, the smaller the estimation error $\epsilon(t)$ is, the more accurate $\widetilde{A}$ and $\widetilde{\boldsymbol{E}}(t)$ are. The effect of the estimation error on the system performance is further discussed in Sections 2.4.5 and 4.2.5.

2.4 Optimization and Adaptive Control

2.4.1 Installed Light Capacity

Installed light capacity is the maximum amount of illumination the system's luminaires can supply. Depending on the occupancy of a work station and a user's desired illuminance level, we want the illuminance on sensor i to be at least a certain target value h_i. We suppose that the installed light capacity is such that, for every sensor i, h_i is no more than

$$c_i = \sum_{j=1}^{M} 1 \cdot A_{ij} \tag{2.13}$$

that is, the illuminance on the sensor with all lights fully on.

Or, in a vector form,

$$\boldsymbol{c}(t) = A(t) \cdot \mathbf{1} \geq \boldsymbol{h} \tag{2.14}$$

2.4.2 Optimization Problem Formulation

The goal of the system is to minimize the total power consumed by M luminaires while providing (at least) the target illuminance levels at N work stations. Thus, we consider the following optimization problem:

$$\begin{aligned} \underset{\boldsymbol{d}}{\text{minimize}} \quad & \sum_{j=1}^{M} P(d_j) \\ \text{subject to} \quad & R_i(\boldsymbol{d}, t) \geq h_i \quad \text{for all i} \in 1 \ldots \text{M} \\ & 0 \leq d_j \leq 1 \qquad \text{for all j} \in 1 \ldots \text{N} \end{aligned} \tag{2.15}$$

Recall that, based on the luminaire model developed in Section 2.2, the relationship between power and dimming level is linear (Equation 2.4). Therefore, the optimization problem is reduced to minimizing the sum of dimming levels subject to illumination requirements. By combining the optimization problem formulation above with Equations 2.4 and 2.5, we get

$$\begin{aligned} \underset{\boldsymbol{d}}{\text{minimize}} \quad & sum\{\boldsymbol{d}\} \\ \text{subject to} \quad & \boldsymbol{E}(t) + A(t)\boldsymbol{d} \geq \boldsymbol{h} \\ & \mathbf{0} \leq \boldsymbol{d} \leq \mathbf{1} \end{aligned} \tag{2.16}$$

Based on this optimization problem formulation, one can tell that the system saves energy by avoiding unnecessary over-illumination of workspaces, whenever possible. Note that the optimization problem 2.16 is always feasible, because, when all the lights are fully on (that is, $d_j = 1$ for all bulbs j), we have

$$\boldsymbol{R}(\mathbf{1}, t) = \boldsymbol{E}(t) + A(t) \cdot \mathbf{1} \geq \boldsymbol{c} \geq \boldsymbol{h}$$

To solve the linear program 2.16, the knowledge of $A(t)$ and $\boldsymbol{E}(t)$ is required. If we knew the real $A(t)$ and $\boldsymbol{E}(t)$ at every moment t, our system would be amenable to a simple snapshot optimization.[7] However, in reality, we can only estimate $A(t)$ and $\boldsymbol{E}(t)$, and these estimations are prone to error. Next, we present an approximated optimization problem and an iterative control algorithm, which allows compensating for inaccuracies in the system model.

2.4.3 Approximated Optimization Problem

To design a practical system, rather than using a true unknown matrix $A(t)$, we express the optimization problem 2.16 in terms of $\widetilde{A}$, obtained in the process of calibration, and the

[7] If we knew $A(t)$ and $\boldsymbol{E}(t)$, the optimal dimming levels could be chosen at each optimization step without regard to the past or future lighting conditions.

associated estimation error $\epsilon(t)$, defined in Equation 2.10. Moving forward with developing the iterative algorithm, for the sake of convenience, let us use indices to indicate different iterations of the algorithm. An index k denotes a variable's value at the moment of iteration k, thus capturing its time-varying nature. The optimization program can then be written as

$$\begin{aligned} \underset{\boldsymbol{d}_k}{\text{minimize}} \quad & sum\{\boldsymbol{d}_k\} \\ \text{subject to} \quad & \boldsymbol{E}_k - \epsilon_k \boldsymbol{d}_k + \widetilde{A}\boldsymbol{d}_k \geq \boldsymbol{h} \\ & \mathbf{0} \leq \boldsymbol{d}_k \leq \mathbf{1} \end{aligned} \tag{2.17}$$

This linear program cannot be solved directly because neither ϵ_k nor $\boldsymbol{E}_k$ is known. Recall from Equation 2.12 that the aggregate sum of $\boldsymbol{E}_k$ and $-\epsilon_k \boldsymbol{d_k}$ is denoted by the environmental illumination estimate $\widetilde{\boldsymbol{E}}_k$ and calculated as

$$\widetilde{\boldsymbol{E}}_k = \boldsymbol{E}_k - \epsilon_k \boldsymbol{d}_k = \boldsymbol{R}_k - \widetilde{A}\boldsymbol{d}_k \tag{2.18}$$

However, both terms on the right-hand side of Equation 2.18 can be measured only after $\boldsymbol{d_k}$ has already been computed and set on luminaires, thus, they are unavailable to the optimizer. We propose to approximate the estimated environmental illumination $\widetilde{\boldsymbol{E}}_k$ by $\widetilde{\boldsymbol{E}}_{k-1}$, which can be readily calculated with Equation 2.18. Then we can write

$$\boldsymbol{E}_k - \epsilon_k \boldsymbol{d}_k = \widetilde{\boldsymbol{E}}_k \approx \widetilde{\boldsymbol{E}}_{k-1} = \boldsymbol{E}_{k-1} - \epsilon_{k-1}\boldsymbol{d}_{k-1} = \boldsymbol{R}_{k-1} - \widetilde{A}\boldsymbol{d}_{k-1} \tag{2.19}$$

To see why $\boldsymbol{E}_k - \epsilon_k \boldsymbol{d}_k \approx \boldsymbol{E}_{k-1} - \epsilon_{k-1}\boldsymbol{d}_{k-1}$ is a reasonable approximation, let us look at all of the respective variables on both sides individually. First, since we want the dimming levels on luminaires to converge, we force $\boldsymbol{d}_k \approx \boldsymbol{d}_{k-1}$. The condition for $\boldsymbol{d}_k$'s convergence is discussed in more detail in the next section. Second, considering that the time between iterations is at most a few seconds, $\boldsymbol{E}_k \approx \boldsymbol{E}_{k-1}$ when there are no abrupt changes in environmental illuminance. Similarly, $\epsilon_k \approx \epsilon_{k-1}$ when no big sensor movements, shadows or reflections occur between two consecutive iterations.[8]

After substituting $\boldsymbol{E}_k - \epsilon_k \boldsymbol{d}_k$ according to Equation 2.19 in the program 2.17, we get

[8]Note that, in general, due to the control loop, discussed in Section 2.4.4, the system is capable of quickly and effectively adapting even if $\boldsymbol{E}_k$ and ϵ_k change substantially between the two iterations. This is demonstrated by a set of experiments presented in Chapter 4.

$$\begin{aligned} \underset{\boldsymbol{d}_k}{\text{minimize}} \quad & sum\{\boldsymbol{d}_k\} \\ \text{subject to} \quad & \widetilde{A} \cdot (\boldsymbol{d}_k - \boldsymbol{d}_{k-1}) + \boldsymbol{R}_{k-1} \geq \boldsymbol{h} \\ & \mathbf{0} \leq \boldsymbol{d}_k \leq \mathbf{1} \end{aligned} \tag{2.20}$$

Now, all of the terms, except for the unknown dimming vector $\boldsymbol{d}_k$, are readily available. Specifically, to solve the optimization problem 2.20 and find optimal dimming levels $\boldsymbol{d}_k$, one needs the matrix $\widetilde{A}$, empirically obtained through the calibration process, a target illuminance vector $\boldsymbol{h}$,[9] a previous dimming vector $\boldsymbol{d}_{k-1}$, and illuminance levels $\boldsymbol{R}_{k-1}$ on the light sensors achieved when $\boldsymbol{d}_{k-1}$ is set on the luminaires.[10] Then, a solution can be obtained by standard linear programming algorithms, such as Simplex or Interior-Point.

2.4.4 Adaptive Control Algorithm

The linear program 2.20 is approximated, and consequently, dimming levels obtained by solving it are also approximate. The proximity of illumination achieved by setting these dimming levels on luminaires to target set points depends on the accuracy of our system model, namely on the magnitude of the estimation error ϵ_k, defined in the Equation 2.10. As ϵ_k approaches 0, $\widetilde{A}$ tends to A_k and $\widetilde{\boldsymbol{E}}_k$ tends to $\boldsymbol{E}_k$. However, since the calibration process is subject to various errors and due to time-varying nature of A_k, error ϵ_k can be significant, which can result in a significant deviation of the achieved illumination from the target.

To force the illuminance on light sensors to converge to target set points[11] and make the system robust to the aforementioned model imperfections, an iterative control algorithm is used. Figure 2.7 shows the corresponding control diagram. At each iteration k the controller executes the following steps:

1. Using target illumination $\boldsymbol{h}$, the previous iteration's dimming vector $\boldsymbol{d}_{k-1}$ and illuminance readings $\boldsymbol{R}_{k-1}$, solve the optimization problem 2.20, whose solution is a dimming vector $\boldsymbol{d}_k$.

[9] Target illuminance is a function of office occupancy and individual illuminance preference of users. This is discussed in greater detail in Section 3.4.2.

[10] If it is the first iteration of the algorithm ($k = 1$), we can use the currently set dimming levels as $\boldsymbol{d}_{k-1}$, and the resulting light sensor readings as $\boldsymbol{R}_{k-1}$.

[11] Whenever target illumination is physically feasible on the light sensors, the system will converge to it. On the other hand, if the exact target illumination is physically infeasible, the system will deliver the amount of light at least equal to the target set points to all work stations, while minimizing the sum of luminaires' dimming levels. This is due to the formulation of the optimization problem 2.16.

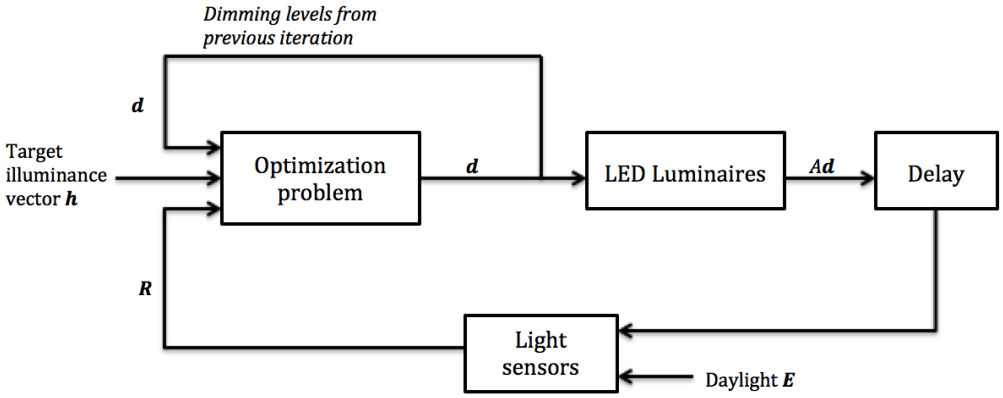


Figure 2.7: Control diagram for the smart lighting control.

2. Set the computed dimming levels $\boldsymbol{d}_k$ on luminaires, and wait until the luminaires fully adapt to the new dimming levels.[12]

3. Measure new illuminance $\boldsymbol{R}_k$ on the light sensors, which is composed of artificial lighting and environmental lighting.

4. Go back to step 1.

2.4.5 Convergence of Dimming Levels

We now discuss the convergence conditions of the proposed lighting control. To make our analysis easier, let us assume that the target illumination is physically achievable by all sensors, i.e., all illuminance constraints in the optimization problem 2.20 can be satisfied with equality. Another assumption used in the convergence analysis is that the system converges sufficiently fast so that environmental illuminance gains and the estimation error matrix do not change much on such timescale. In other words, we assume that $\epsilon_k = \epsilon_{k-1} = \epsilon$ and $\boldsymbol{E}_k = \boldsymbol{E}_{k-1} = \boldsymbol{E}$.[13]

[12] Our measurements indicate that the full adaption of the luminaires to the new dimming levels typically requires less than 700 milliseconds, but, for additional robustness, we allocate 1300 milliseconds for this adaption.

[13] Note that these assumptions are only used in the convergence analysis. In general, the smart lighting system can adapt to the time-varying environmental lighting, and is robust to the time-varying estimation errors.

First, let us go through several steps of the control algorithm. At iteration k the controller solves the optimization problem 2.20 and computes the dimming vector $\boldsymbol{d}_k$. The computed vector $\boldsymbol{d}_k$ satisfies the illuminance constraint with equality so that

$$\widetilde{A} \cdot (\boldsymbol{d}_k - \boldsymbol{d}_{k-1}) + \boldsymbol{R}_{k-1} = \boldsymbol{h}$$

or, considering Equation 2.19,

$$\widetilde{A}\boldsymbol{d}_k - \epsilon \boldsymbol{d}_{k-1} + \boldsymbol{E} = \boldsymbol{h} \tag{2.21}$$

Then, after the luminaires fully adapt to the new dimming levels $\boldsymbol{d}_k$, light sensors measure the illuminance levels $\boldsymbol{R}_k$, which comprise environmental illuminance $\boldsymbol{E}$ and artificial illuminance $A_k\boldsymbol{d}_k$:

$$A_k\boldsymbol{d}_k + \boldsymbol{E} = \boldsymbol{R}_k$$

where A_k is a true theoretical illuminance gains matrix. By expressing A_k in terms of the empirically obtained matrix $\widetilde{A}$, as specified in Equation 2.10, we can write it as

$$\widetilde{A}\boldsymbol{d}_k - \epsilon \boldsymbol{d}_k + \boldsymbol{E} = \boldsymbol{R}_k \tag{2.22}$$

Now, using the updated dimming vector $\boldsymbol{d}_k$ and the most recent illuminance readings $\boldsymbol{R}_k$, along with $\widetilde{A}$ and $\boldsymbol{h}$, the controller solves the optimization problem 2.20 again and finds the new solution $\boldsymbol{d}_{k+1}$, while satisfying the illuminance constraint with an equality:

$$\widetilde{A}\boldsymbol{d}_{k+1} - \epsilon \boldsymbol{d}_k + \boldsymbol{E} = \boldsymbol{h} \tag{2.23}$$

By subtracting Equation 2.22 from Equation 2.21 we get:

$$\epsilon(\boldsymbol{d}_k - \boldsymbol{d}_{k-1}) = \boldsymbol{h} - \boldsymbol{R}_k \tag{2.24}$$

On the other hand, subtracting Equation 2.22 from Equation 2.23 gives

$$\widetilde{A}(\boldsymbol{d}_{k+1} - \boldsymbol{d}_k) = \boldsymbol{h} - \boldsymbol{R}_k \tag{2.25}$$

Note that, since Equations 2.24 and 2.25 have the same right-hand side, they can be combined into:

$$\widetilde{A}(\boldsymbol{d}_{k+1} - \boldsymbol{d}_k) = \epsilon(\boldsymbol{d}_k - \boldsymbol{d}_{k-1}) \tag{2.26}$$

By introducing $\Delta\boldsymbol{d}_k = \boldsymbol{d}_k - \boldsymbol{d}_{k-1}$, we can rewrite Equation 2.26 as

$$\widetilde{A}\Delta\boldsymbol{d}_{k+1} = \epsilon\Delta\boldsymbol{d}_k \tag{2.27}$$

and, therefore

$$\Delta\boldsymbol{d}_{k+1} = \widetilde{A}^{-1}\epsilon\Delta\boldsymbol{d}_k \tag{2.28}$$

Equation 2.28 indicates the convergence condition of the control system. Thus, whether the system converges depends on the product of matrices $\widetilde{A}^{-1}$ and ϵ. Intuitively, if the elements of matrix ϵ are smaller than the elements of matrix $\widetilde{A}$, the convergence occurs. To demonstrate this, we consider the simplest possible example with one bulb and one light sensor, such that $\widetilde{A}, \epsilon \in \mathbb{R}^{1\times1}$. If, for example, $\widetilde{A} = [1000]$ and $\epsilon = [1]$, then the system converges very quickly:

$$\begin{aligned} 1000\Delta d_{k+1} &= 1\Delta d_k \\ \Delta d_{k+1} &= 0.001\Delta d_k \\ \Delta d_{k+2} &= 0.000001\Delta d_k \\ &\dots \end{aligned}$$

On the other hand, when $\widetilde{A} \approx \epsilon$, we get:

$$\begin{aligned} \widetilde{A}\Delta\boldsymbol{d}_{k+1} &= \epsilon\Delta\boldsymbol{d}_k \\ \Delta\boldsymbol{d}_{k+1} &\approx \Delta\boldsymbol{d}_k \\ \Delta\boldsymbol{d}_{k+2} &\approx \Delta\boldsymbol{d}_{k+1} \\ &\dots \end{aligned}$$

This result has been empirically confirmed by a set of testbed experiments discussed in Section 4.2.5. Even when some sensors are partially occluded during the calibration process, making $\widetilde{A}$ inaccurate, the system converges quickly as long as $\widetilde{A} \gg \epsilon$. On the other hand, if elements of ϵ are too big, convergence might not occur.[14] Thus, we have

[14]Note that, the calibration process proposed in Section 2.3 guarantees that $\widetilde{A} \gg \epsilon$, provided that the sensor blockage is not too severe.

demonstrated the convergence conditions of the lighting control algorithm in the small-error regime. When the dimming vector converges, target illuminance is achieved on all sensors, which is forced by the illuminance constraint of the linear program 2.20.

2.4.6 The System's Response to Changes in Environment and Occupancy

Changes in environmental illumination $\boldsymbol{E}_k$ (e.g., daylight) are handled directly by the optimizer in the control epoch following these changes. Specifically, after the environmental changes occur, the illuminance readings on the light sensors deviate from the target set points. Then, based on these readings, the optimizer calculates the new dimming vector (by solving the optimization problem 2.20) to recover the target illuminance on the sensors. The convergence of the dimming levels occurs as discussed in Section 2.4.

On the other hand, changes in target illuminance set points $\boldsymbol{h}$ (e.g., occupancy and users' illuminance preferences) are handled differently for time-efficiency reasons: when a change in target illuminance occurs, we terminate the ongoing control iteration, and immediately start the next one with the updated vector $\boldsymbol{h}$. More details on specifics of the implementation are presented in Chapter 3.

2.5 Integrating New Sensing Modules and Luminaires into the System

One of the goals of the smart lighting system is a flexible plug-and-play design that does not require any manual calibration. Thus, we have developed a seamless method for integrating new components, such as luminaires and sensing modules, into the lighting system, even while it is in use. This method utilizes the proposed calibration process that allows the system to unobtrusively obtain illuminance gains on desired sensors from desired bulbs. This section describes how integration of new components into the system is modeled, whereas implementation details and performance evaluation of this method are discussed in Sections 3.2.4 and 4.2.4, respectively.

Let the lighting system have N light sensors and M luminaires, and assume that it is in use. To integrate a new sensing module into the system, illuminance gains on its light sensor $(N+1)$ from each luminaire $j \in \{1, ..., M\}$ have to be estimated and added to the illuminance gains matrix $\widetilde{A}$ as an $(N+1)$'th row. These illuminance gains can be obtained

through the calibration routine described in Section 2.3, where we slightly dim one bulb at a time, and infer the respective illuminance contributions. Therefore, time complexity of integrating a new sensing module into the system is proportional to the number of luminaires, namely $\theta(M)$.

Similarly, in order to integrate a new luminaire into the smart lighting system, illuminance gains on each sensor $i \in \{1, ..., N\}$ from this luminaire $M + 1$ have to be obtained and added to the illuminance gains matrix $\widetilde{A}$ as an $(M + 1)$'th column. This column can be obtained using just one step of the calibration routine, i.e., we slightly dim only the new luminaire, and deduce illuminance gains on each of the sensors from it. Therefore, integrating a new luminaire always takes a constant time $\theta(1)$. Finally, to disconnect a sensing module or a luminaire from the system, it is only required to remove a corresponding row or column from the matrix $\widetilde{A}$.

Chapter 3

Implementation Details

As a proof-of-concept of the proposed smart lighting system, we have implemented a prototype testbed. It consists of three principal components: stand-alone sensing modules that measure occupancy and illuminance of each work station, dimmable LED luminaires, and a central controller that receives sensor inputs and sends control signals to each luminaire. A high-level diagram of the system is shown in Figure 3.1. Next, we describe the testbed and each of its components in greater detail.

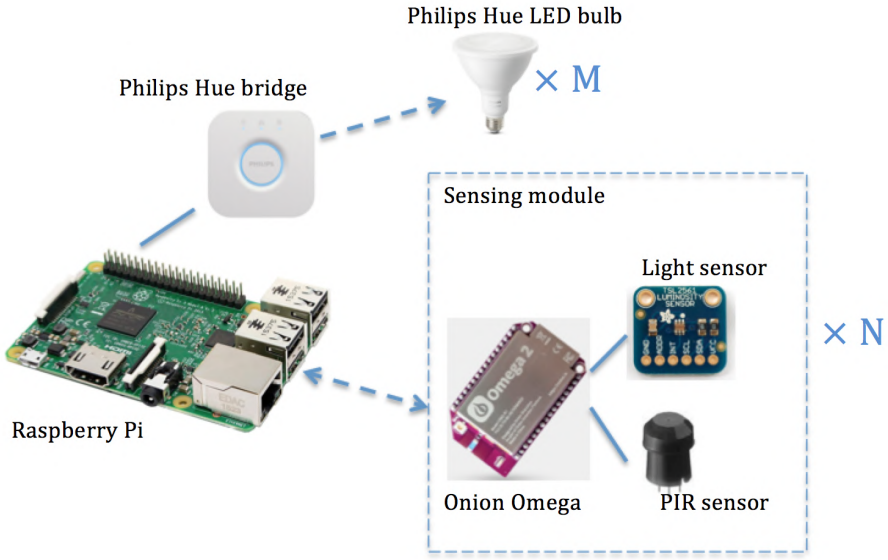


Figure 3.1: High-level diagram of the smart lighting system.

3.1 Experimental Testbed: High-level Overview

While the following sections discuss individual system components and communication between them in greater detail, this section provides a high-level overview of our testbed implementation of the smart lighting system, shown in Figure 3.2.

Figure 3.2: Testbed implementation of the smart lighting system.

To implement the system, we have installed a 2.45 m. high ceiling above four desks in a secluded 3.0 × 3.8 m. work area. The space is illuminated by eight dimmable LED luminaires (Section 3.3), evenly installed in the ceiling, that wirelessly communicate with a central control module (Section 3.4). Each desk has a wireless sensing module (Section 3.2) deployed on the edge of its shelf, which periodically sends occupancy status and illuminance level to the control module via a local wireless network. Also, each occupant can express individual illuminance preference at their desk through that desk's sensing module. Users'

illuminance preference, along with sensor readings, serve as inputs to the proposed control algorithm, discussed in Section 2.4.4.

Our testbed does not have a window, therefore, we imitate daylight by switching on/off the laboratory's central light. Note that from the smart lighting system's perspective, the external artificial light and daylight are equivalent, and both of them are modeled as environmental illuminance gains $\boldsymbol{E}(t)$. Thus, switching the artificial lighting on and off is roughly equivalent to quickly opening and closing blinds.

Several experiments have been run on this experimental testbed to comprehensively evaluate the performance of the proposed smart lighting system. The results of these experiments are discussed in Chapter 4.

3.2 Sensing Modules

The smart lighting system is capable of sensing occupancy and illuminance at every work station, which is accomplished by "per-desk" sensing modules. The main purpose of a sensing module is to communicate sensor readings to the main controller with a certain periodicity. In our implementation, a user can also use it to express an illuminance level preference at their work station.

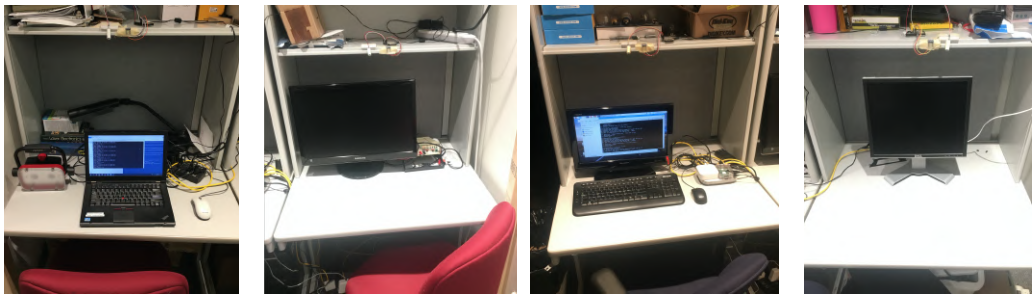
In general, communication between sensors and the controller can be implemented in either a wireless or a wired fashion, and both of these implementations have certain pros and cons, discussed in Section 3.2.5. Since one of the major goals of this work is to develop a system that can be easily installed and deployed, we implement wireless stand-alone sensing modules for our prototype testbed.

Each sensing module consists of a microcomputer, a light sensor and an occupancy sensor, as illustrated in Figure 3.1. In particular, our sensing module implementation uses the following hardware components:

- **Onion Omega2** [33]: an expandable single-board microcomputer with 580 MHz MIPS CPU processor, 64 MB of RAM, built-in Wi-Fi, a Linux Operating System, and flexible GPIOs.
- **Adafruit TSL-2561 light sensor** [26]: a light-to-digital converter that transforms light intensity to a digital signal output compatible with a direct Inter-Integrated Circuit ($I2C$) interface.

- **Panasonic AMN-32111 motion sensor** [42]: a digital passive infrared (PIR) motion sensor with a sensing range of 2 meters.

Figure 3.3 shows stand-alone sensing modules, deployed at every work station of the prototype testbed.



(a) (b) (c) (d)

Figure 3.3: Wireless stand-alone sensing modules that provide per-desk sensing. In our implementation, sensing modules are placed on the edge of each desk's shelf.

3.2.1 Illuminance Sensing

We want to develop a system that can sense illumination at each work station, ideally, at the work surface level. However, placing light sensors directly on the work surfaces is impractical because of the high possibility of sensor blockage, occlusion or shadowing. In order to avoid occlusion and alleviate shadowing, for every work station we propose mounting one upward-facing illuminance sensor on the top of a computer monitor, on the edge of a shelf, or on an adjacent wall.[1] In particular, our implementation places illuminance sensors on the edge of each desk's shelf, as depicted in Figure 3.3, and assumes that illuminance measured there is a good approximation for the work surface illuminance.

The TSL-2561 is not a calibrated instrument [26]. Therefore, before using sensors, we calibrate them, as shown in Figure 3.4. Each sensor stores its calibration value. Sensing modules use these calibration values to compute accurate lux values, before sending measurements to the controller.

[1] If neither monitors nor shelves are available, one could design a wireless peel-and-stick sensing module, for example, sitting on top of a miniature L-shaped shelf stack to a wall.

(a) Illuminance sensor calibration setup.

```
-------------
| 183 | 183 |
-------------
| 183 | 183 |
-------------
```

(b) Output of calibrated illuminance sensors.

Figure 3.4: Illuminance sensor calibration.

3.2.2 Occupancy Sensing

Digital PIR motion sensors are used for per-desk occupancy detection. These sensors produce a binary output, which indicates whether a motion is detected within their sensing range. To work with a stream of binary motion signal, we maintain a motion history queue containing the L most recent motion readings, which is asynchronously updated every 100 milliseconds. To deduce the occupancy status of a work station, first we calculate an occupancy score, defined as a discounted sum of elements of the motion history queue:

$$\text{occupancy\_score} = \sum_{p=0}^{L-1} \gamma^p \ \text{motion\_history[p]} \tag{3.1}$$

where γ is a position discount and `motion_history` is a binary array that implements a motion history queue. Note that `motion_history` is indexed in a way that `motion_history[0]` is the most recent motion reading. Thus, a higher weight is given to more recent motion values. Next, the occupancy status is determined as:

$$\text{occupancy\_status} = \begin{cases} 1 & \text{if occupancy\_score} \geq \text{T} \\ 0 & \text{if occupancy\_score} < \text{T} \end{cases} \tag{3.2}$$

where `T` is a threshold above which we consider the sensor's surroundings occupied. γ and `T` are tuning parameters of the proposed occupancy detector. Since it is crucial that occupied work stations are illuminated properly, these parameters should be chosen to reduce false-negative errors[2] in occupancy detection. In our implementation, we use

[2]In this case, a false negative error is when the occupancy detector thinks that a work station is not occupied, while in reality, it is.

$\texttt{T} = 0.8$, and $\gamma = 0.95$, which empirically demonstrate good performance. Note that since $\texttt{T} = 0.8$, whenever a slight motion is detected in a PIR sensor's field of view, the occupancy status immediately changes from 0 to 1. On the other hand, the status changes from 1 to 0 only if there has been no registered motion for some time.

3.2.3 Concurrent Tasks of a Sensing Module

A sensing module has to perform several asynchronous tasks. To accommodate these tasks, the following threads run concurrently on a sensing module's microcomputer:

1. The first thread continuously updates the motion history queue by adding a new motion sensor reading to the queue every 100 milliseconds, while maintaining its size at L.

2. The second thread listens for user input to disconnect from the smart lighting system or change preferred illuminance level on a work station. Whenever a user input is received, a sensing module updates a locally stored persistent variable `illuminance_preference` accordingly. `illuminance_preference` indicates how much illumination a user wants to have on their work station when they occupy it. When a command to disconnect a sensing module from the system is received, `illuminance_preference` is set to -1.

3. The third thread listens for incoming requests from a control module. Whenever a request is received, it reads the illuminance sensor reading, computes occupancy status as described in Section 3.2.2, reads the value of the `illuminance_preference` variable, and returns an `(illuminance, occupancy, illuminance_preference)` tuple to the controller. When the system is in use, requests from the controller are received every 100 milliseconds.

The code for sensing modules has been implemented in Python.

3.2.4 Additional Portable Sensing Modules

New sensing modules can be seamlessly integrated into the system, even while it is in use. Thus, a user can place a sensing module on a surface of interest where illumination needs to be controlled, and send a connection request from that sensing module to the

controller. Once the controller accepts the connection request, it initiates the re-calibration procedure to integrate the new module into the system, as discussed in Section 2.5. After re-calibration is finished, the system starts maintaining the desired illuminance level on this sensing module's light sensor.

In addition to the regular "per-desk" sensing modules, one can use portable sensing modules to maintain the desired illumination at any arbitrary surface. This can be useful for tasks requiring a temporary illumination of some areas of the office. We assume that portable sensing modules are used irregularly, and disconnected from the system, once a user finishes the task. Therefore, in terms of hardware requirements, the only difference between a "per-desk" and a portable sensing module is that the latter does not require an occupancy sensor since the occupancy is implied as long as it is connected to the system. Figure 3.5 shows an example implementation of a portable module, consisting of a microcomputer and a sensor. Note that, in fact, "per-desk" sensing modules are portable too, as they are not restricted to any specific location.

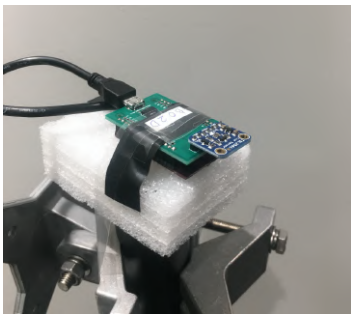


Figure 3.5: Example implementation of a portable sensing module.

In general, any device equipped with a light sensor and Wi-Fi connectivity, such as a smartphone or a tablet, can be used as a portable sensing module. Thus, an example use case is maintaining the desired illuminance by simply placing a smartphone with an upward-facing light sensor to a surface of interest, and connecting it to the system. Note that if a phone is lifted or moved significantly, its illuminance gains become inadequate. To address this, the smartphone's built-in accelerometer can be used to detect if the device is lifted, moved, or tilted significantly. Then, whenever it happens, the smartphone can immediately disconnect from the lighting system.

3.2.5 Alternative Implementation of Occupancy and Illuminance Sensing

Our testbed implementation uses wireless stand-alone sensing modules to measure per-desk occupancy and illuminance. An advantage of this implementation is the simplicity of installation and deployment, which makes the system flexible and plug-and-play. However, since the problems of sensing and optimal lighting control are decoupled, our lighting control strategy, discussed in Chapter 2, does not impose any restrictions on the used illuminance and occupancy sensing technology. At a high level, a sensing module has only one role - to communicate the illumination level and occupancy status of a work station to the controller with a certain periodicity, and there are many possible implementations that achieve this goal. Moreover, it should be straightforward to integrate any sensing implementation with the smart lighting system, as long as it is capable of providing per-desk occupancy and illuminance measurements.

In particular, sensing can be implemented so that the light and occupancy sensors are directly connected to the controller via wires. This implementation would reduce the hardware cost by eliminating the need for additional microcomputers for the stand-alone sensing module implementation. Moreover, considering that each microcomputer consumes around 1 W of power, the system's power consumption and associated cost would also go down. However, the wired sensing implementation would potentially require significant installation and deployment effort, especially in larger offices.

3.3 Luminaires and Actuation

Illumination of the office is done by screw-base controllable **PAR38 Philips Hue** LED bulbs [41], whose power rating is 13 W and luminous flux rating is 1300 lumen. The bulbs wirelessly communicate with the controller through a Philips Hue bridge, as depicted in Figure 3.1. Section 3.4 discusses communication between the luminaires and the controller in more detail.

To control Philips Hue bulbs we use its Python API [40]. This API supports switching each bulb on and off, as well as setting its brightness control values to an integer between 0 and 255. The model of the PAR38 Philips Hue bulbs is presented in Section 2.2.

3.3.1 Alternative Wired Implementation

Similarly to the implementation of sensing, we can implement luminaire actuation in a hardwired fashion. The main advantage of a wired implementation over the wireless one is power savings. As Section 2.2.2 demonstrates, each PAR38 Philips Hue luminaire consumes 1.18 W of power for wireless communication, even when it is off. Besides, hardwired systems tend to be more reliable and secure. On the other hand, the wired implementation of individual luminaire actuation would result in a significantly higher installation and deployment costs, which is especially true when retrofitting existing lighting systems. Also, once installed, wired systems are difficult to upgrade.

3.4 Control and Communication

We do not impose any restrictions on the controller technology being used, other than that it has to be able to quickly solve the optimization problem 2.20 and orchestrate luminaire switching and communication with sensing modules. For our controller implementation, we use the **Raspberry Pi 3 B+** [46] microcomputer with 1.4 GHz Broadcom BCM2837B0 quad-core A53 processor and 1 GB of RAM, running Raspbian Jessie operating system. The code for the controller logic is implemented in Python.

As illustrated in Figure 3.1, the controller communicates with individual luminaires and sensing modules wirelessly. Communication with the luminaires is achieved through a Philips Hue bridge. The controller is connected to the bridge via an Ethernet cable. The bridge, in turn, transmits signals to the luminaires via the wireless ZigBee protocol [61]. On the other hand, communication between the controller and sensing modules occurs via Wi-Fi. In particular, the Raspberry Pi (controller) serves as a wireless access point, via which sensing modules connect to the network. Then, TCP/IP peer-to-peer socket communication is established between the controller and each sensing module.

The controller is where the system's main logic resides, i.e., it is the "brain" of the system. It performs multiple tasks, some of which have to be asynchronous. To accomplish these tasks, the controller runs several processes presented next.

3.4.1 Calibration Process

A **calibration process**, as its name suggests, is responsible for conducting the system calibration, i.e., estimating an illuminance gains matrix. The **calibration process** is the

first process run when the system is started. In addition, it can be initiated by other processes, when necessary.

3.4.2 Sensing Process

A **sensing process** is the process triggered right after the initial calibration completes. This process always maintains an open connection with sensing modules and requests `(illuminance, occupancy, illuminance_preference)` readings from them every 100 milliseconds. These readings are used to update the respective illuminance vector $\boldsymbol{R}$, occupancy vector $\boldsymbol{o}$, and illuminance preference vector $\boldsymbol{p}$, which are persistently stored on the controller. In particular, the illuminance preference value of $p_i = -1$ received from a sensing module i is interpreted as a request to disconnect that module from the system.

Also, whenever new sensor readings are received, a target illuminance vector $\boldsymbol{h}$ is re-computed. Note that the target illuminance vector $\boldsymbol{h}$ is a function of occupancy and illuminance preferences:

$$h_i = \begin{cases} p_i & \text{if } o_i = 1 \\ H^U & \text{if } o_i = 0 \end{cases} \tag{3.3}$$

where p_i and o_i are an illuminance preference and an occupancy status of a sensing module i respectively, and H^U is a universal constant illuminance set point that is used for all unoccupied work stations to provide ambient lighting in unoccupied areas.

The **sensing process** also initiates a **control process**, as well as establishes and maintains a connection with it. Whenever there is a change in target illuminance $\boldsymbol{h}$, the **sensing process** immediately notifies the **control process** by sharing a new vector $\boldsymbol{h}$.

Finally, the **sensing process** initiates the re-**calibration process** with a certain periodicity set by users to make sure that matrix $\widetilde{A}$ remains reasonably accurate over time. Meanwhile, it also communicates with the **control process** to pause the control algorithm before running the re-calibration, and restart it after the re-calibration is finished.

3.4.3 Control Process

A **control process** is responsible for running the optimization control algorithm discussed in Section 2.4.4. It is implemented with two subprocesses. An *optimizer* subprocess is where the optimization control algorithm is run, while a *listener* subprocess listens for

commands from the **sensing process** to pause the iterative control algorithm or to restart it with a new target illuminance vector $\boldsymbol{h}$.

The optimizer has access to the illuminance vector $\boldsymbol{R}$,[3] which is maintained by the **sensing process**. At the same time, each optimizer subprocess is passed a target illuminance vector $\boldsymbol{h}$ only once - at the moment of its initiation. Then, the same constant $\boldsymbol{h}$ is used throughout the entire existence of the subprocess, until it is killed. Using $\boldsymbol{R}$, $\boldsymbol{h}$ and the current dimming settings,[4] in every iteration it computes a new dimming vector by solving the optimization problem 2.20 and uses it to adjust dimming levels on luminaires. Thus, each optimizer subprocess executes the logic for dynamically adapting to changes in environmental illumination and maintaining a constant target illumination $\boldsymbol{h}$ at the work stations.

When the listener receives a command from the **sensing process** to reset the illuminance set points $\boldsymbol{h}$, it terminates the optimizer subprocess, thereby interrupting an ongoing iteration of the control algorithm. Then, it immediately starts a new optimizer subprocess using the new $\boldsymbol{h}$. The reason we interrupt the ongoing iteration of the control algorithm, instead of letting it finish, is related to the desired reaction time to occupancy changes.[5] When a work station becomes occupied, it is important to illuminate it as quickly as possible. At the same time, waiting for the ongoing iteration of the control algorithm to finish might take up to $\sim$2 additional seconds. While the reaction time of a few seconds is acceptable for changes in illuminance, it is too slow for changes in occupancy. Besides, changes in target illuminance do not occur nearly as frequently as changes in daylight.

3.4.4 Incoming Connection Listener Process

An **incoming connection listener** process listens for incoming connection requests from new sensing modules. Whenever a connection request is received, the process integrates a new sensing module into the system by adding a new row to the illuminance gains matrix $\widetilde{A}$ through the re-**calibration process** described in Section 2.5. It also communicates with the **control process** to ensure that the lighting control is paused during the sensing module integration. Once the re-calibration is completed, the control is restarted with the updated matrix $\widetilde{A}$.

[3] Note that vector $\boldsymbol{R}$ refers to vector $\boldsymbol{R}_{k-1}$ from Section 2.4.

[4] Here "current dimming settings" refers to vector $\boldsymbol{d}_{k-1}$ from Section 2.4.

[5] Note that an alternative slower implementation of the response to changes in target illuminance would be to update vector $\boldsymbol{h}$ between iterations, to make sure that the next iteration uses the most up-to-date $\boldsymbol{h}$ in its optimization program. In fact, this is how changes in environmental illuminance are addressed - we supply a new most recent $\boldsymbol{R}$ to the optimizer before each iteration.

Chapter 4

System Evaluation

This chapter discusses an evaluation of the smart lighting system, based on the experimental testbed described in Section 3.1. Specifically, we answer the following questions:

- Does the proposed adaptive control achieve target illuminance levels at all work stations? (Section 4.2)
- How quickly does the system adapt to changes in environmental lighting? (Section 4.2.2)
- How quickly does the system respond to changes in occupancy and users' illuminance preferences? (Section 4.2.3)
- How seamless is the integration of new sensing modules into the smart lighting system, and how well does the system maintain target illuminance on them? (Section 4.2.4).
- What is the effect of an error in the estimated illuminance gains matrix on the performance of the system? (Section 4.2.5).
- What are the potential energy savings from daylight harvesting and occupancy detection are, and how much energy does our system consume in comparison to other lighting systems? (Section 4.3)

To investigate these questions, a series of experiments were conducted. These experiments are discussed next.

4.1 Obtaining the Illuminance Gains Matrix A

By using an unobtrusive calibration procedure described in Section 2.3, the following illuminance gains matrix $\widetilde{A} \in \mathbb{R}^{4\times 8}$ is obtained for our experimental testbed with 4 light sensors and 8 luminaires[1]:

```
176.19    5.46    13.66    329.15   2.73    6.83    25.95    13.66
5.40      195.84  12.16    1.35     430.85  21.61   4.05     9.45
198.54    2.70    6.75     14.86    4.05    5.40    301.19   17.56
5.87      199.69  5.87     2.94     13.21   431.68  5.87     19.09
```

From the matrix $\widetilde{A}$ the installed light capacity at sensor i can be computed as $c_i = \sum_{j=1}^{8} \widetilde{A}_{ij}$. Therefore, the installed light capacity on the four work stations is

```
573.63    680.72    551.06    684.23
```

Thus, our experimental testbed is able to provide the required standard comfort levels, which are typically between 300 lux and 500 lux [2, 38], on all desks. Note from the matrix $\widetilde{A}$ above that the illuminance gains on all sensors from bulbs 3 and 8 (which correspond to the third and the last columns in the matrix) are relatively small, due to these bulbs being located further away from the sensing modules. If bulbs 3 and 8 are removed from the system, the resulting illuminance gains matrix $\widetilde{A} \in \mathbb{R}^{4\times 6}$ becomes

```
176.19    5.46    329.15   2.73    6.83    25.95
5.40      195.84  1.35     430.85  21.61   4.05
198.54    2.70    14.86    4.05    5.40    301.19
5.87      199.69  2.94     13.21   431.68  5.87
```

Then, the corresponding installed light capacity is

```
546.31    659.11    526.75    659.27
```

As one can see, six 1300 lumen-rated bulbs are sufficient to illuminate the office space (delivering at least 500 lux to all work stations), even when no environmental lighting is available. Therefore, in the evaluation of the reduction in energy consumption, presented in Section 4.3, we consider systems with six bulbs and the estimated illuminance gains matrix $\widetilde{A} \in \mathbb{R}^{4\times 6}$ above.

[1] Recall that $\widetilde{A}_{ij}$ is the estimated illuminance gain of sensor i from a fully-lit luminaire j.

4.2 System Performance

Now we investigate the performance of the smart lighting system, and in particular, how quickly and accurately it converges to the target illuminance levels. The experiments presented in Sections 4.2.2, 4.2.3, and 4.2.4 are run after a careful calibration of the system. Thus, we assume that the estimated illuminance gains matrix $\widetilde{A}$ in these experiments is close to accurate. On the other hand, experiments in Section 4.2.5 are performed with matrices $\widetilde{A}$ of different degrees of inaccuracy to investigate the effect of these model inaccuracies on the system's performance.

4.2.1 Timescales

The control of luminaires' dimming levels is done via an iterative control algorithm, presented in Section 2.4.4. Therefore, the system's timescales can be expressed in terms of the number of iterations. Based on our empirical measurements, the maximum time of one iteration is $\sim$*1.65 seconds*. To understand where this number comes from, refer to the control diagram from Figure 2.7 and consider each step of the closed loop separately:

1. Obtaining an illuminance level vector $\boldsymbol{R}$ is the first step of the loop. The duration of this step is primarily governed by the I/O procedure of reading the illuminance vector R from a file persistently stored on the disk, as described in Section 3.4.3. This step takes up to *50 milliseconds*.

2. The next step is computing optimal dimming levels by solving the optimization problem (Equation 2.20). The Raspberry Pi 3 B+ solves this problem in up to *200 milliseconds*.

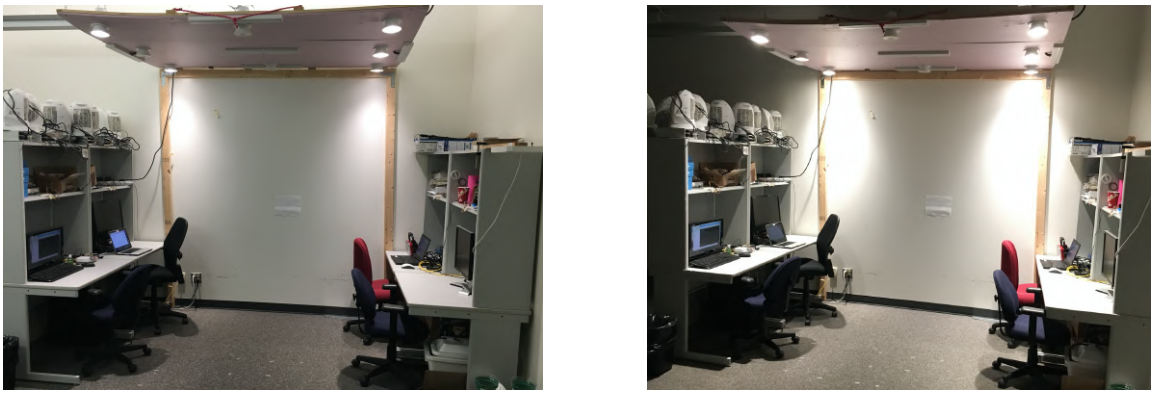
3. The third step is setting the dimming levels on the Philips Hue LED luminaires. The duration of this step is determined primarily by the ZigBee network performance, i.e., the latency with which the luminaires receive packets from the Philips Hue bridge. Based on our measurements, the maximum time of this step is around *100 milliseconds*.

4. Finally, after the dimming level is changed, the LED luminaires need time to fully adapt to the new dimming level. Our measurements indicate that this typically requires less than *700 milliseconds* in most cases, but, for additional robustness, we allocate *1300 milliseconds* for this adaption. This step is indicated as "Delay" on the control diagram in Figure 2.7.

Thus, the total time required for one iteration of the control algorithm adds up to 1.65 seconds.

4.2.2 Responsiveness to Changes in Environmental Illuminance

Daylight is the most common source of environmental illumination in offices. However, the laboratory where our experimental testbed is deployed has no windows, thus, there is no access to daylight. On the other hand, as mentioned in Section 3.1, the laboratory has binary (on-off) central lighting, which we use to provide environmental illuminance to the system. From the smart lighting system's perspective, switching the artificial lighting on and off, as shown in Figure 4.1, is roughly equivalent to quickly opening and closing blinds.



(a) Laboratory's central lighting is on.

(b) Laboratory's central lighting is off.

Figure 4.1: Experimental setup with external lighting on and off.

To evaluate the smart lighting system's responsiveness to changes in environmental illuminance, we run an experiment. First, we set the system to maintain heterogeneous illuminance levels on the four sensing modules,[2] namely, 300, 350, 450 and 500 lux. As the laboratory's central lighting is turned on and off, the following parameters with corresponding timestamps are recorded:

1. Dimming levels on luminaires at the moment when they are set;
2. Illuminance readings from the four sensing modules when they are received by the central controller, i.e., every ∼100 milliseconds;

[2]There is one sensing module deployed per work station, as discussed in Section 3.2.

3. Timestamps for the central light turn on/turn off events.

The results of this experiment are illustrated in Figure 4.2. White and yellow regions correspond to the laboratory's main light being off and on, respectively. The top time series show the illuminance signals on the four sensing modules. The bottom time series show dimming levels on the luminaires, with the circles corresponding to the moments when dimming levels are set.

Abrupt spikes on the illuminance time series correspond to abrupt changes in environmental illuminance, i.e., turning on or turning off the laboratory's central light. As one can see, it takes the system 2-4 seconds to fully adapt to the environmental changes and restore illuminance levels on all sensors. From the dimming level time series one can see that the system's reaction[3] occurs within $\sim$2 seconds of the change in external lighting. These time series also demonstrate that in most cases only 1 iteration is required for the system to fully converge, which is due to a carefully estimated matrix $\widetilde{A}$.

The exact reaction time depends on where in the control loop (Figure 2.7) the system is when the environmental changes happen. In theory, the best possible reaction time occurs if the change in environmental illuminance happens a moment before a new control iteration starts[4] and the updated illuminance values are accurately captured before the new iteration. Thus, the best time of the system's reaction to changed environmental illuminance is around *350 milliseconds*, which corresponds to steps 1-3 of the control algorithm discussed in Section 4.2.1. On the other hand, the worst possible reaction time occurs if the environmental illuminance change happens a moment after a new iteration starts. In that case, the reaction time is one full iteration plus 350 milliseconds, which is around *2 seconds*.

Provided that we have an accurately estimated matrix $\widetilde{A}$ and it takes 1 iteration for the system to fully converge, illuminance on the sensors should be restored within another 1.3 seconds[5] after the system's reaction. This makes the total time of adaptation to changes in environmental illuminance *1.65-3.3 seconds*. Note that if the matrix $\widetilde{A}$ is poorly estimated, it would take more iterations for the system to converge. System performance with inaccurate matrices $\widetilde{A}$ is considered in Section 4.2.5.

Finally, it should be pointed out that, while this experiment considers significant abrupt changes in environmental illuminance, natural changes in daylight are much smoother and

[3] A reaction time is defined as the time it takes the system to set new dimming levels on luminaires after the change in environmental illuminance happens.

[4] Here we refer to the control algorithm steps discussed in Section 4.2.1.

[5] We assume that 1.3 seconds is a maximum time that it takes luminaires to adapt to new dimming levels.

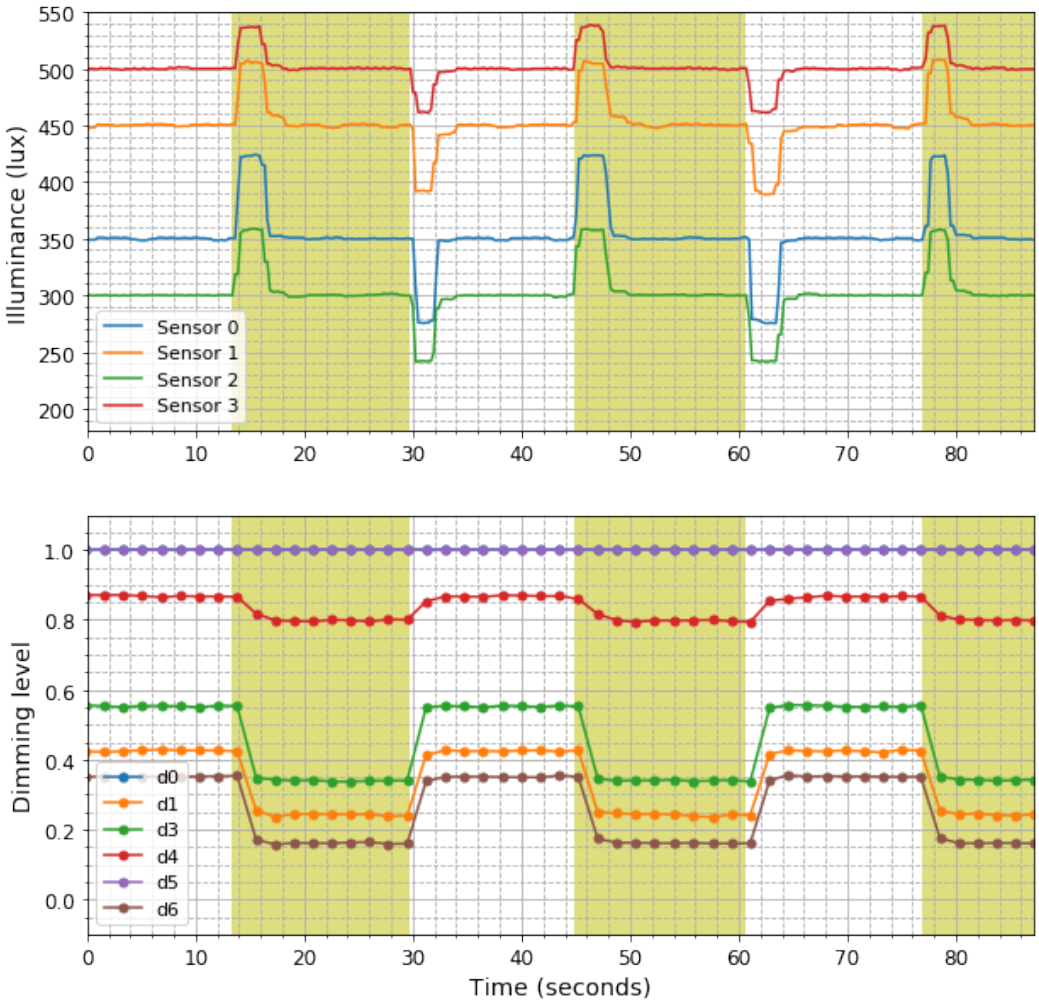


Figure 4.2: Smart lighting system's response to changes in environmental illuminance. Note that the bulbs that are off throughout the entire experiment, i.e., bulbs 3 and 8 (corresponding to the dimming level signals d2 and d7), are not shown in the bottom plot.

subtler. When the environmental lighting changes per iteration are lower than the eye sensitivity, the system adapts to these changes seamlessly.

4.2.3 Responsiveness to Changes in Users' Illuminance Preference and Occupancy

This section evaluates and analyzes the system's responsiveness to changes in users' illuminance preference and occupancy. Changes in illuminance preference and occupancy are considered together because, as discussed in Sections 3.4.2 and 3.4.3, they are handled by the controller in the same way. Specifically, illuminance preference and occupancy vectors together determine target illuminance $\boldsymbol{h}$, as specified by Equation 3.3.

To evaluate the system's responsiveness to changes in users' illuminance preferences, we run an experiment. This time the system is set to maintain constant illuminance levels of 300, 350 and 500 lux on three of the desks, while on the fourth desk a new illuminance preference is set every 10-15 seconds by its occupant. For this experiment, the following time series are recorded:

1. Dimming levels on luminaires at the moment when they are set;
2. Illuminance readings from the four sensing modules when they are received by the central controller, i.e., every ∼100 milliseconds;
3. The occupants' illuminance preferences, and timestamps corresponding to changes in the occupants' illuminance preferences.

The results of this experiment are shown in Figure 4.3. On the top plot, solid lines correspond to sensor readings, while the dashed line corresponds to the user's illuminance preference that has been changed several times throughout the experiment. The bottom plot shows dimming levels on the luminaires, with the circles corresponding to the moments when dimming levels are set.

By comparing the top time series to the bottom ones, one can see that the target illuminance changes are followed by the system's reaction[6] almost immediately, within a fraction of a second. For instance, based on the experimental data, the first drop in the

[6]The "system's reaction" refers to the system setting new dimming levels on luminaires to respond to the new target illuminance.

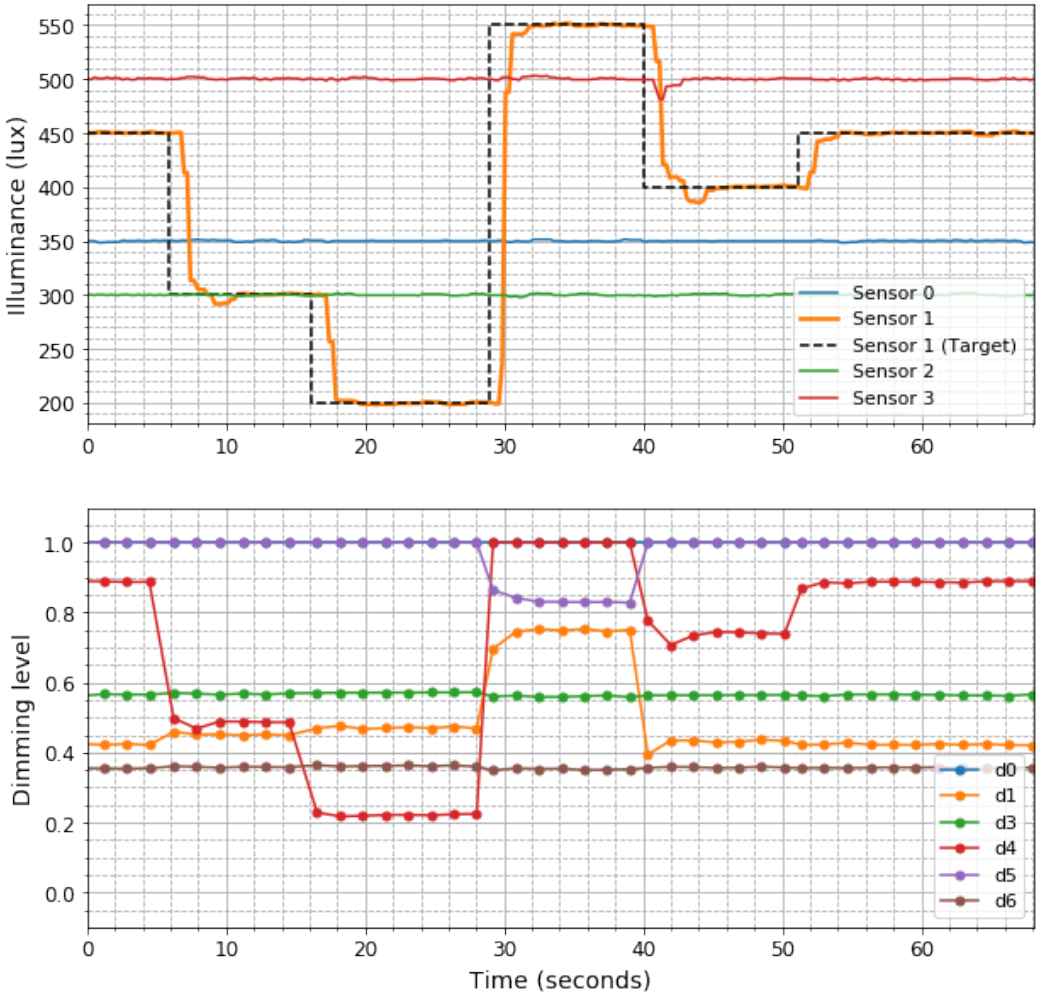


Figure 4.3: Smart lighting system's response to changes in target illuminance. Note that the bulbs that are off throughout the entire experiment, i.e., bulbs 3 and 8 (corresponding to the dimming level signals d2 and d7), are not shown in the bottom plot.

target illuminance occurs at time 5.92 seconds, which is followed by bulbs 1 and 4 taking new dimming level values at time 6.23 seconds.

Recall from Section 3.4.3 that when a change in target illuminance is registered, the system immediately terminates the ongoing iteration of the control loop, and restarts the optimizer with the new target illuminance settings. This strategy results in the system's reaction to target illuminance (and occupancy) changes being consistently fast. Our measurements indicate that the reaction time is up to *350 milliseconds*, which corresponds to getting illuminance values, solving an optimization program and sending commands to the luminaires.[7]

After dimming levels are set, it takes the luminaires at most 1.3 seconds to fully adapt to these new dimming settings. Thus, provided that we have an accurately estimated matrix $\widetilde{A}$ and it takes 1 iteration for the system to fully converge, the maximum total time to achieve the target illuminance on all sensors is *1.65 seconds*.[8] It should also be emphasized that illuminance levels on other sensors, whose target illuminance is constant throughout the entire experiment, remain unaffected despite the various changes in luminaires' dimming levels.

In the next experiment, we test the system's response to sequential changes in occupancy of all 4 work stations. We simulate the scenario where users come to their work station one by one, stay for 50-60 seconds, and then leave. The target illuminance on occupied desks is set according to user preferences, which are chosen to be 300, 350, 450 and 500 lux. On the other hand, the target illuminance of unoccupied desks is 0 lux, as we assume that an unoccupied desk does not have to be illuminated. For this experiment, the following time series are recorded:

1. Dimming levels on luminaires at the moment when they are set;
2. Illuminance readings from the four sensing modules when they are received by the central controller, i.e., every $\sim$100 milliseconds;
3. Timestamps corresponding to changes in the occupancy of each work station.

The results of this experiment are shown in Figure 4.4. The top four plots show real and target illuminances on the four light sensors, indicated by solid and dashed lines,

[7]Note that these are steps 1-3 discussed in Section 4.2.1.

[8]If the illuminance gains matrix estimate $\widetilde{A}$ is inaccurate, it would take more iterations for the system to converge, as discussed in Section 4.2.5.

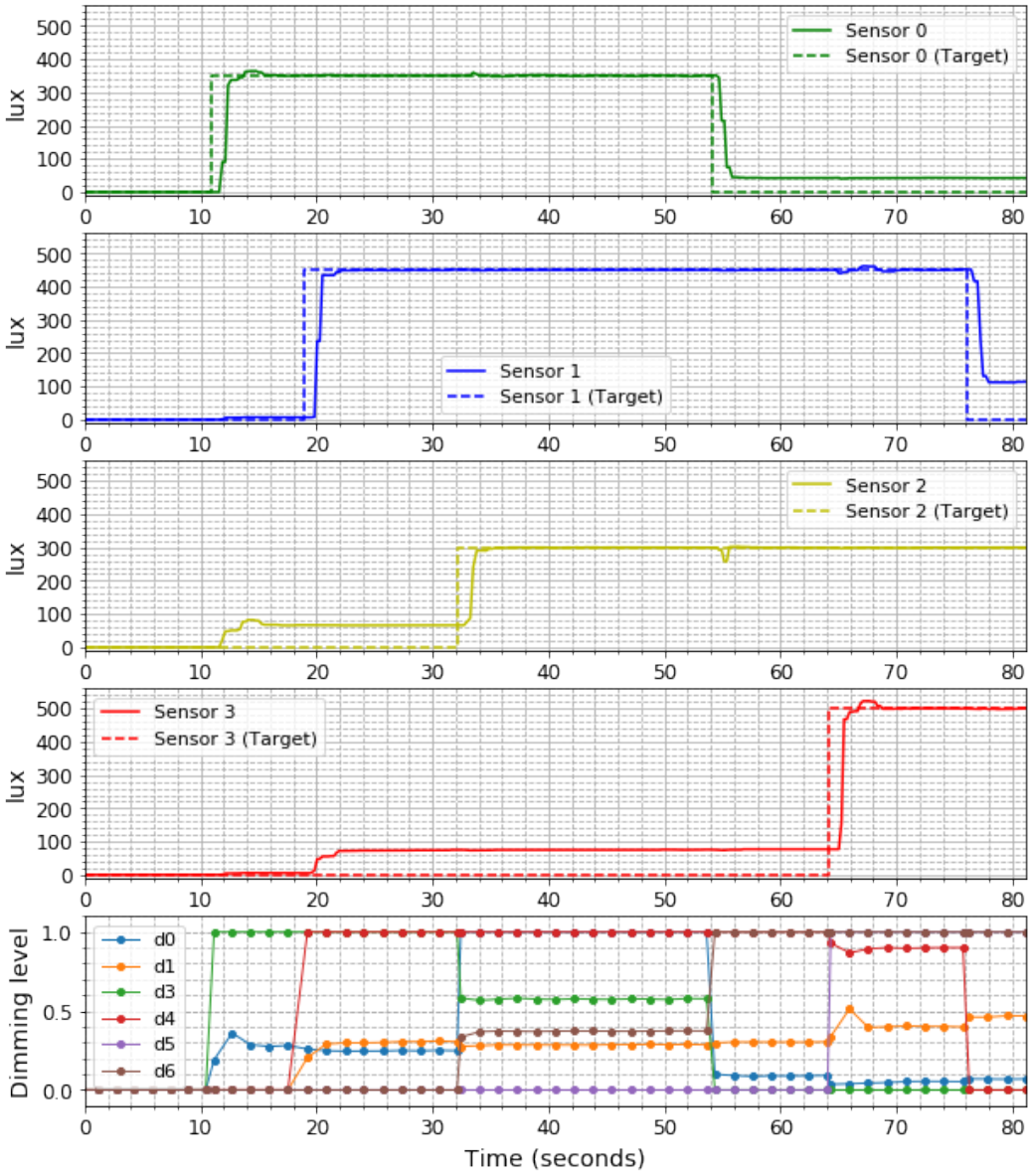


Figure 4.4: Smart lighting system's response to changes in occupancy. Note that the bulbs that are off throughout the entire experiment, i.e., bulbs 3 and 8 (corresponding to the dimming level signals d2 and d7), are not shown in the bottom plot.

respectively. The bottom time series show the dynamically changing dimming levels of the luminaires.

This experiment demonstrates that the smart lighting system succeeds at illuminating the occupied work stations according to the users' preferences, while not illuminating the unoccupied ones. By examining the experimental data, we also confirm that the system reacts to changes in target illuminance within 350 milliseconds, and fully adapts to them in ∼1.65 seconds.

Note that sometimes it is impossible to achieve the exact target illuminance on a sensor due to physical limitations. For instance, in Figure 4.4, the illuminance levels at unoccupied desks are sometimes higher than the target value of 0 lux. The unoccupied desk gets some unintended illumination, when a neighboring desk's target illuminance value is high. However, it should be pointed out that the control algorithm always tries to minimize any over-illumination, due to the formulation of the optimization problem's objective function.

4.2.4 Using Portable Sensing Modules

In this section, the integration of portable sensing modules into the smart lighting system is experimentally evaluated. Specifically, we investigate whether the process of integrating sensing modules into the system is unobtrusive. Also, we test how accurately the system maintains the desired target illuminance levels on all sensors throughout the experiment.

In this experiment, the system is set to maintain a constant illuminance of 400 lux on all four work stations. Two portable sensing modules are placed in the office at arbitrarily chosen locations and heights, as shown in Figure 4.5. First, we integrate one of the portable sensing modules into the system and set a new illuminance preference on it every ∼10 seconds. Then, we repeat the same for the second portable sensing module. Finally, both sensing modules are disconnected from the system one after the other. Throughout the experiment the following time series are recorded:

1. Dimming levels on luminaires at the moment when they are set;
2. Illuminance readings from both "per-desk" and portable sensing modules when they are received by the central controller, i.e., every ∼100 milliseconds;
3. Illuminance preferences on the portable sensing modules at the moment when they are set, as well as the time of their connection/disconnection.

Figure 4.6 illustrates the results of this experiment. The top plot shows the actual illuminance levels on all sensing modules and target illuminance settings on the portable modules, represented by solid and dashed lines, respectively. The bottom plot shows the luminaires' dimming levels with the circles corresponding to moments when the dimming levels are set.

Figure 4.5: Experimental setup for evaluating the integration of portable sensing modules into the smart lighting system. Two portable sensing modules are mounted on the top of tripods.

After accepting the connection request from a sensing module, the smart lighting system runs an automated calibration process to integrate this module, which is indicated by grey regions in Figure 4.6. For our system with eight luminaires, this process takes around 11 seconds. Note that on the bottom plot, as expected, we see sequential spikes in the luminaires' dimming levels during the calibration process. At the same time, the top plot demonstrates that the calibration causes only slight fluctuations in illuminance levels on the sensors. These slight fluctuations, while being sufficient for estimating the illuminance

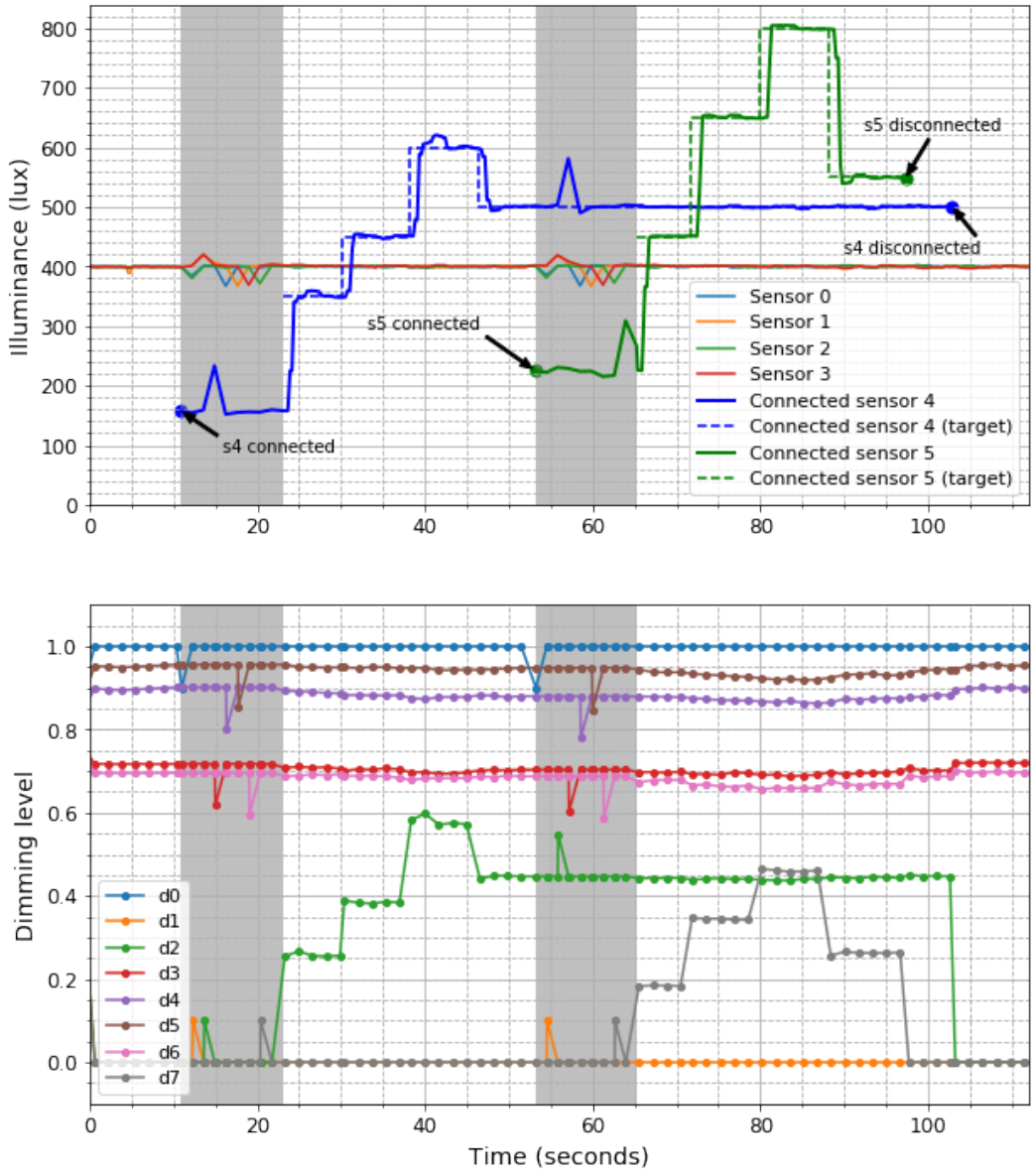


Figure 4.6: Experiment for connection, operation, and disconnection of the additional sensing modules. Grey regions on the plots correspond to an automated calibration process to integrate the additional sensing modules. (Note that the preferred target illuminance can be set on the sensing modules only after they have been integrated into the system.)

gains matrix, are nearly invisible to an eye.

This experiment also confirms that the smart lighting system successfully achieves and maintains the desired illuminance levels on both "per-desk" and portable sensing modules. Note that when the illuminance on one of the sensing modules changes due to either the varying user preferences or the module being disconnected from the system, illuminance levels on the rest of the modules remain mostly undisturbed.

4.2.5 The Effect of Error in the Estimated Illuminance Gains Matrix on the System's Performance

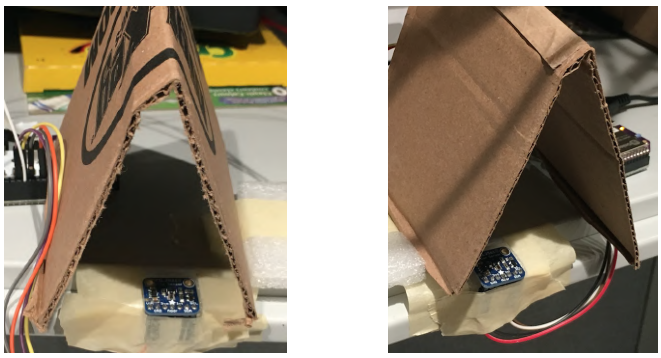
Previous sections of this chapter evaluate the performance of the smart lighting system with a carefully estimated illuminance gains matrix. In particular, Section 4.2.2 demonstrates that the system fully adapts to abrupt changes in environmental illuminance after one iteration of the control algorithm, provided that $\widetilde{A} \approx A(t)$. However, in practice, an estimated illuminance gains matrix is subject to error due to several factors including inaccurate calibration, accident sensor movements, and decrease in bulbs' luminous flux with temperature. While Section 2.4.5 discusses the effect of this error on the convergence of luminaires' dimming levels from a theoretical point of view, this section evaluates it empirically. Specifically, we run a set of experiments to investigate the performance of the system when the estimated illuminance gains matrix $\widetilde{A}$ has various degrees of inaccuracy.

This evaluation consists of a calibration phase (phase 1) and an operation phase (phase 2). In phase 1, we obtain matrices $\widetilde{A}$ of various degrees of error ϵ by using different levels of light sensor occlusion *during the calibration process*. To quantify the level of sensor occlusion, we first make the system achieve exactly 400 lux on all *unoccluded* sensors. Next, after partially covering the sensors (as shown in Figure 4.7), we measure their illuminance levels, note a decrease in illuminance due to the occlusion,[9] and run the calibration. Once the calibration is completed, we *uncover the light sensors*, which makes the estimated illuminance gains from luminaires on the sensors inaccurate. As shown in Figure 4.8, the level of sensor occlusion during the calibration increases from experiment D1 to D5, which results in increasingly inaccurate matrices $\widetilde{A}$.[10]

In phase 1, we obtained matrices $\widetilde{A}$ of different degrees of estimation error. In phase 2, we evaluate the lighting system's performance with these (inaccurately estimated) matrices,

[9] Larger deviations from 400 lux on the illuminance sensors correspond to more severe occlusions and, consequently, higher estimation errors in the matrix $\widetilde{A}$.

[10] Estimated illuminance gains matrices $\widetilde{A}$ obtained in experiments D1-D5 are presented in Appendix B.



(a) (b)

Figure 4.7: Illuminance sensors are partially occluded during the phase 1 (calibration phase) of the experiments to obtain the illuminance gains matrix estimates of various degrees of inaccuracy.

400	400
400	400

(a) Exp. D1 (phase 1)

309	298
303	311

(b) Exp. D2 (phase 1)

298	252
229	340

(c) Exp. D3 (phase 1)

209	190
203	196

(d) Exp. D4 (phase 1)

144	146
165	142

(e) Exp. D5 (phase 1)

Figure 4.8: Light sensor readings resulting from different levels of sensor occlusion, where illuminance on all unoccluded sensors is set to 400 lux.

by setting the system to maintain constant heterogeneous illuminance levels on the four light sensors, namely, 175 lux, 200 lux, 225 lux and 250 lux. As the laboratory's central lighting is turned on and off, the following parameters with corresponding timestamps are recorded[11]:

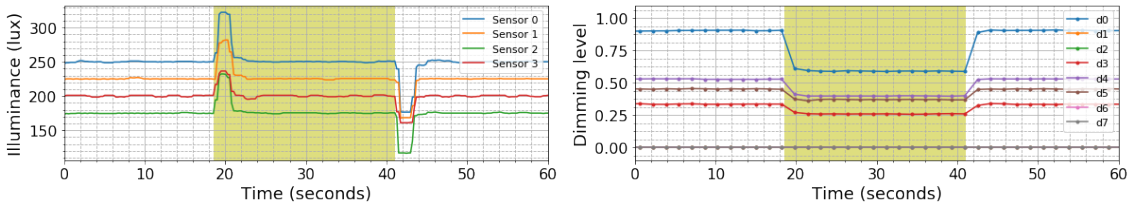
1. Dimming levels on luminaires at the moment when they are set;
2. Illuminance readings from the four sensing modules when they are received by the central controller, i.e., every ∼100 milliseconds;
3. Timestamps for the central light turn on/turn off events.

The results of these experiments are shown in Figure 4.9. White and yellow regions correspond to the laboratory's main light being off and on, respectively. The left-hand-side time series show the illuminance signals on the four sensing modules. The right-hand-side time series show dimming levels on the luminaires, with the circles corresponding to the moments when dimming levels are set.

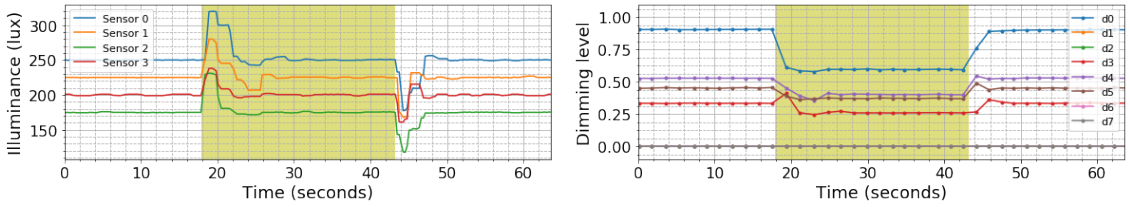
This evaluation demonstrates that as the error in the estimated illuminance gains matrix $\widetilde{A}$ increases, the system requires more iterations to converge. However, as long as matrix $\widetilde{A}$ is sufficiently accurate, as in the experiments D1-D3, the dimming levels converge quickly and target illuminance is achieved on all sensors within a few iterations of the control algorithm. Once the error in matrix $\widetilde{A}$ becomes significant, as in the experiment D4, dimming levels on some of the luminaires converge slowly. Finally, the system does not converge at all when the error in the illuminance gains matrix estimate is too large, as in the experiment D5. Overall, the experiments presented in this section empirically verify the conclusions in Section 2.4.5, namely:

1. The convergence occurs quickly when the estimated illuminance gains are greater than the estimation errors: $\widetilde{A} \gg \epsilon(t)$ (as demonstrated in the experiments D1-D3);
2. The smaller $\epsilon(t)$ is relatively to $\widetilde{A}$, the faster convergence occurs (as demonstrated in the experiments D1-D4);
3. The system does not converge if $\widetilde{A} \approx \epsilon(t)$, (as demonstrated in the experiment D5).

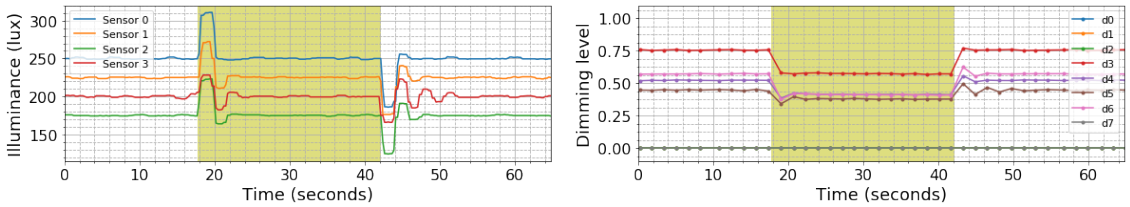
[11] Phase 2 of this evaluation is the same as the experiment discussed in Section 4.2.2, except that it uses matrices $\widetilde{A}$ of different degrees of inaccuracy.



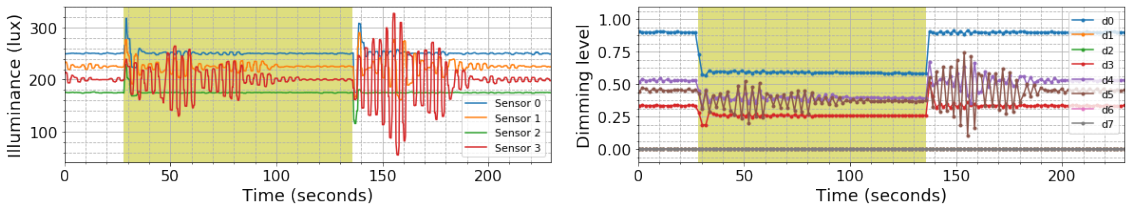
(a) System performance in Experiment D1 (phase 2).



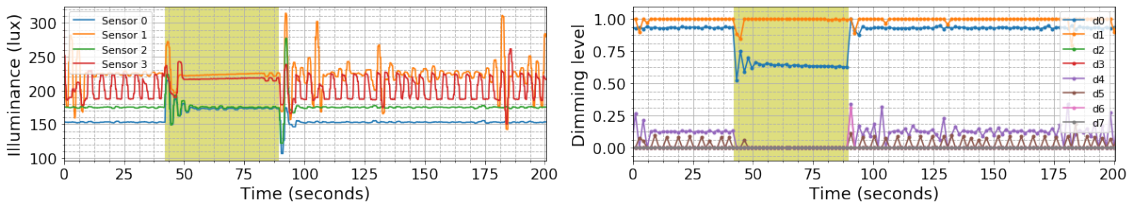
(b) System performance in Experiment D2 (phase 2).



(c) System performance in Experiment D3 (phase 2).



(d) System performance in Experiment D4 (phase 2).



(e) System performance in Experiment D5 (phase 2).

Figure 4.9: Performance of the smart lighting system using matrices $\widetilde{A}$ of various degrees of estimation error.

4.3 Reduction in Energy Consumption

The main motivation behind the smart lighting system is a reduction in energy consumption. To evaluate the proposed system's energy saving potential, we estimate its average daily energy consumption through extensive simulations using 7 months of collected illuminance and occupancy data. We also run the same simulations for several other common lighting systems and then compare different systems' energy efficiency. Section 4.3.1 describes the simulation procedure, and Section 4.3.2 discusses results.

4.3.1 Simulation Description

The goal of this part of system evaluation is to estimate how much energy the proposed system can save through daylight harvesting and occupancy sensing. To evaluate the system's energy efficiency, we simulate its power consumption under dynamically changing daylight illuminance and occupancy conditions.

The power consumption of each luminaire is a function of its dimming level, and for our luminaires, this relationship is modeled by Equation 2.4. Recall that, according to the system's theoretical model, at every time t the optimal dimming vector $\boldsymbol{d}(t)$ can be computed by solving the optimization problem 2.16, which is rewritten below for convenience:

$$\begin{aligned} \underset{\boldsymbol{d}}{\text{minimize}} \quad & sum\{\boldsymbol{d}\} \\ \text{subject to} \quad & \boldsymbol{E}(t) + A(t)\boldsymbol{d} \geq \boldsymbol{h} \\ & \mathbf{0} \leq \boldsymbol{d} \leq \mathbf{1} \end{aligned}$$

Thus, to obtain a vector $\boldsymbol{d}$, the simulation requires the knowledge of the daylight contribution $\boldsymbol{E}(t)$,[12] target illuminance vector $\boldsymbol{h}$ and illuminance gains matrix $A(t)$ at every moment in time t. Next, we discuss assumptions about each of these variables made in the simulation.

First of all, the same office as our experimental testbed described in Section 3.1 is considered. Section 4.1 has demonstrated that six 1300 lumen-rated luminaires provide sufficient installed light capacity to illuminate all four work stations of our experimental testbed. Therefore, a lighting system with six luminaires ($M = 6$) and four work stations ($N = 4$) is considered. We also approximate the time-varying illuminance gains matrix $A(t)$ by the constant estimated matrix $\widetilde{A} \in \mathbb{R}^{4\times 6}$ presented in Section 4.1.

[12] For this simulation, we assume that daylight is the only source of environmental illumination.

Two target illuminance requirements are imposed depending on whether a work station is occupied. For the occupied work stations we require at least 450 lux on light sensors, whereas unoccupied work stations do not have to be illuminated.[13] We can write this as

$$h_i = \begin{cases} 450 & \text{if } o_i = 1 \\ 0 & \text{if } o_i = 0 \end{cases} \tag{4.1}$$

where h_i and o_i are a target illuminance and an occupancy status of a sensing module i, respectively. Thus, the target illumination at time t is determined solely from the time-varying occupancy signal $\boldsymbol{o}(t)$.

With these assumptions, the only pieces of information required by the simulation to compute time-varying optimal dimming levels (and the corresponding power consumption) are occupancy and daylight signals from each sensing module. Since $N = 4$, for each simulation we need $\boldsymbol{o}(t) \in \mathbb{R}^{4\times 1}$ and $\boldsymbol{E}(t) \in \mathbb{R}^{4\times 1}$. To mimic these dynamically changing occupancy and daylight signals we used the 7 months of collected illuminance and occupancy data, discussed next.

Recall that our experimental testbed has no windows. Therefore, the daylight illuminance data was collected from three other offices, each with a large window, during the 7-month period from October 1, 2018, to May 1, 2019, in Waterloo, Canada. Each of the offices had two sensing modules, similar to the one illustrated in Figure 3.5, installed at two distinct work surfaces. These sensing modules logged illuminance measurements every minute. Occupancy data used in the simulation was collected by SPOT personal thermal comfort devices developed and deployed by Rabbani et al. [43]. The dataset contains 7-month long occupancy signals collected from 20 distinct work stations that belong to graduate students, faculty members or administrative staff also at the University of Waterloo.[14] Since the sampling frequency of the occupancy signals in the dataset is 30 seconds, we had to downsample them to 1 minute to match the illuminance signal's frequency. Thus, to sum up, our illuminance dataset consists of six 7-month long daylight illuminance signals collected from three different offices, and our occupancy dataset consists of twenty 7-month long occupancy signals collected from different work stations. Next, we have

[13] Note that since unoccupied work stations do not have to be illuminated, the illuminance constraints for them are trivial and always satisfied: $R_i \geq 0$.

[14] Note that the actual occupancy traces, collected by the SPOT devices, were 4-month long. Therefore, to match their length to the length of the illuminance traces, we extended them to 7 months as follows. First, with the assumption that occupancy patterns for each workday/weekend are independent, we concatenated each 4-month long occupancy signal with itself, while forcing consistency of time of a day and day of a week. Next, we trimmed these signals at 7 months.

to select occupancy and illuminance signals from these datasets and combine them in a meaningful way to get the desired vectors $\boldsymbol{o}(t) \in \mathbb{R}^{4\times 1}$ and $\boldsymbol{E}(t) \in \mathbb{R}^{4\times 1}$.

The occupancy status of different work stations is independent. Therefore, we can use any 4 occupancy signals in the simulation. On the other hand, daylight illuminance at different parts of the same office is strongly correlated. Thus, we can only use the daylight data collected from the same office in the same simulation. Since we only have two illuminance signals collected per office (while there are four work stations in the testbed), each daylight signal is used to mimic the daylight illuminance on two sensing modules from neighboring work stations, as shown in Figure 4.10. In addition, to avoid using identical traces in the same simulation and make the simulation more "realistic", we apply 5% of random noise to each daylight signal, when it is used.

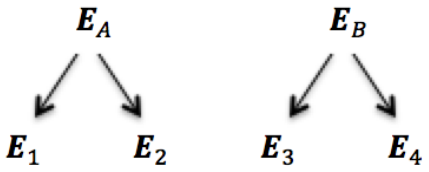
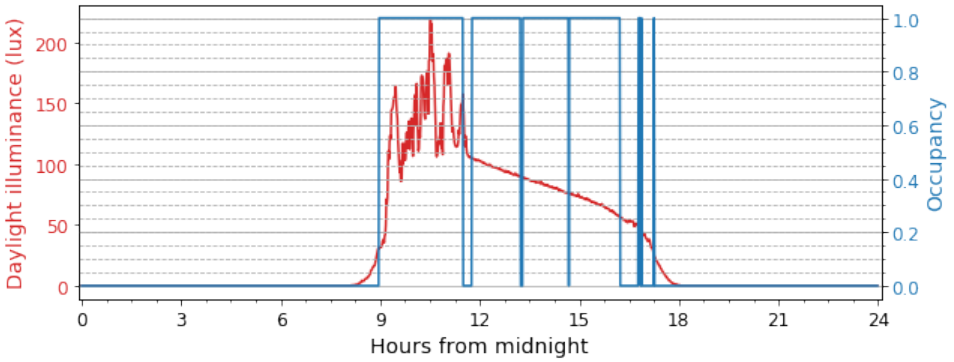


Figure 4.10: Two daylight traces $\boldsymbol{E}_A$ and $\boldsymbol{E}_A$, collected from the same office, are used to mimic environmental illuminance on the four work stations ($\boldsymbol{E}_1$, $\boldsymbol{E}_2$, $\boldsymbol{E}_3$, $\boldsymbol{E}_4$) in the simulation. Note that the signals $\boldsymbol{E}_1$ and $\boldsymbol{E}_2/\boldsymbol{E}_3$ and $\boldsymbol{E}_4$ are slightly different due to the random noise.

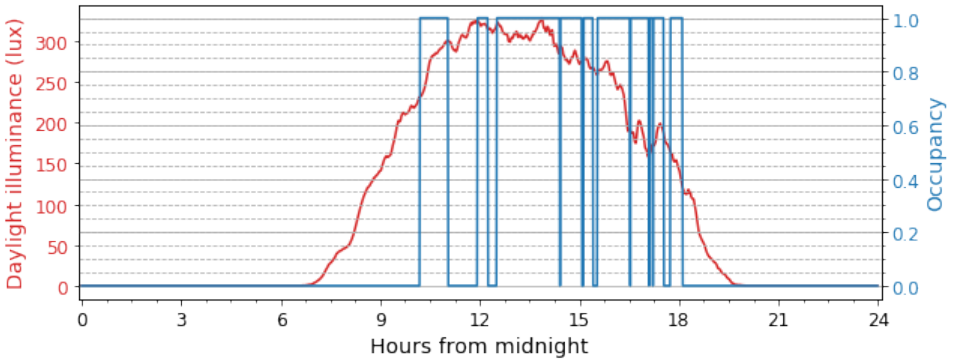
After choosing the illuminance and occupancy signals for the simulation, these signals are combined by matching their timestamps. As a result, we get a 7-month long combined $(t, \boldsymbol{E}(t), \boldsymbol{o}(t))$ signal with a frequency Δt of 1 minute, which can be readily consumed by the simulation. Examples of combined sensor signals for a single day are shown in Figure 4.11.

By solving the optimization problem 2.16 for a time snapshot t of the $(t, \boldsymbol{E}(t), \boldsymbol{o}(t))$ signal, we can estimate the amount of energy consumed by luminaires in each 1-minute interval.[15] If we repeat this calculation for the entire 7 months of data, we can estimate the average daily energy consumption of the simulated system. Thus, by supplying a $(t, \boldsymbol{E}(t), \boldsymbol{o}(t))$ signal to the simulation as an input, we get the average daily energy use as an output.

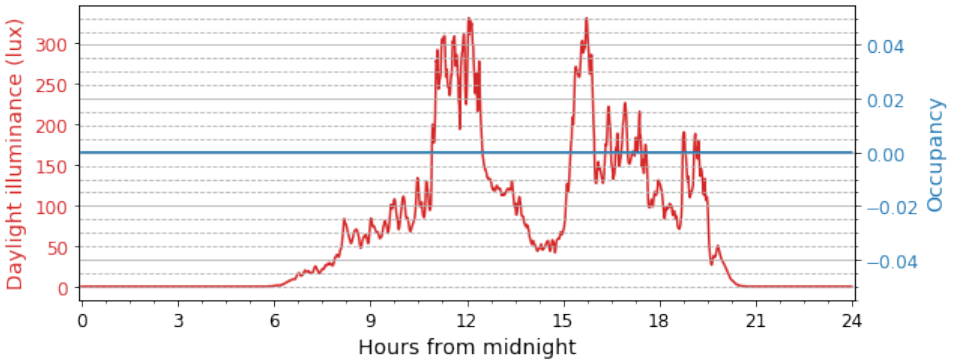
[15]Specifically, we first compute an optimal dimming level vector $\boldsymbol{d}(t)$ at time t by solving the optimization problem 2.16 that consumes $\boldsymbol{E}(t)$, $\boldsymbol{o}(t)$ and $\widetilde{A}$. Next, vector $\boldsymbol{d}(t)$ is used to find the luminaires' power consumption $P(t)$ using Equation 2.4. Finally, energy use during a time step Δt is estimated as $P(t)\Delta t$.



(a) Combined signals for December 5, 2018.



(b) Combined signals for March 12, 2019.



(c) Combined signals for April 13, 2019.

Figure 4.11: Examples of combined daylight illuminance and occupancy signals for two different workdays (a, b) and a weekend (c).

Luminaire type	Occupancy detection	Daylight harvesting
Incandescent	Yes/No	No
CFL	Yes/No	No
LED	Yes/No	Yes/No

Table 4.1: Lighting system configurations for which the average daily energy consumption has been estimated through simulations, based on 7 months of collected illuminance and occupancy data.

To further assess the energy efficiency of the smart lighting system, we compare it to several other lighting system configurations. Specifically, as presented in Table 4.1, we consider the lighting systems using light-emitting diode (LED), compact fluorescent light (CFL) and incandescent luminaires, with/without an occupancy detection capability, and with/without a daylight harvesting capability.[16] Assuming the same office layout as our experimental testbed, we run simulations to calculate how much energy these systems consume on average per day, based on the same $(t, \boldsymbol{E}(t), \boldsymbol{o}(t))$ signals.

To make the comparison fair, we consider the lighting systems with equal installed light capacities. Thus, we assume that all systems use six 1300 lumen-rated luminaires of a corresponding type which, as discussed in Section 4.1, is sufficient to achieve the illuminance of 450 lux on all work stations. Typical 1300 lumen-rated CFL and incandescent bulbs consume 27 W [13] and 85 W [21], respectively. For the LED-based systems, the luminaire model proposed in Section 2.2 and the constant illuminance gains matrix presented in Section 4.1 are assumed. In addition, note that the systems with an occupancy detection and/or daylight harvesting capability consume additional power for sensing, computing and communicating. In our system implementation, the main control module (Raspberry Pi 3 Model B+) consumes 2.2 W, and each sensing module (Onion Omega 2) consumes 1.0 W. Therefore, the total power consumption of the system's microcomputers is 6.2 W, which translates into the additional daily energy consumption of 0.1488 kWh. This number is added to the total daily energy consumption of these systems.

Lastly, in our evaluation, only the *occupancy-aware daylight harvesting system* is able exploit the desk-level occupancy information. The systems with only the occupancy de-

[16]Note that daylight harvesting is only considered for LED-based systems. Daylight harvesting for systems using CFL and incandescent bulbs would require the knowledge of respective luminaire models and appropriate control algorithms, which is out of the scope of this work. Thus, CFL and incandescent bulbs are assumed to be controlled in a binary (on/off) fashion in this evaluation.

tection (but not the illuminance sensing/daylight harvesting) capability, while being aware which desks are occupied, do not have sufficient information to determine how much light needs to be delivered to the individual occupied spaces (and which bulbs need to be on/off).[17] Therefore, we assume that these systems have only an office-level occupancy detection capability, and thus they have to turn all six bulbs fully on if at least one desk is occupied. Furthermore, for the occupancy-unaware systems, it is assumed that the first-to-arrive occupant turns on the light in the morning, the last-to-leave occupant turns it off in the evening, and the light stays on during the time in between.

4.3.2 Simulation Results

For each lighting system configuration from Table 4.1 we run 90 simulations using randomly combined daylight and occupancy traces. Specifically, to generate a combined $(t, \boldsymbol{E}(t), \boldsymbol{o}(t))$ signal for the simulation, we randomly pick four occupancy signals (out of the twenty available) and two daylight signals collected from the same office (out of the six available), and combine them, as described in Section 4.3.1.

Figure 4.12 shows the average daily energy consumption of different lighting systems, based on the simulations. The error bars show 95 % confidence intervals. As expected, the systems using incandescent bulbs consume roughly 3 times more energy than the systems using CFL bulbs that, in turn, consume 2-3 times more energy than LED-based lighting systems. Office-level occupancy detection decreases the average daily energy consumption of the incandescent bulb-based and CFL-based systems by 15% and 8.5%, respectively. At the same time, the LED lighting system with the office-level occupancy detection capability on average consumes slightly more energy than the basic[18] one. This is because our implementation of occupancy sensing requires an additional constant power of 6.2 W. Thus, the more efficient a system's luminaires are, the less rewarding the occupancy detection is.

Note that the average daily energy consumption of the LED lighting systems with either a daylight harvesting or an office-level occupancy detection capability alone is not significantly different from the average consumption of the basic LED lighting system (0.87 kWh vs. 0.78 kWh vs. 0.85 kWh, respectively). At the same time, the LED lighting system that utilizes the combination of daylight harvesting and per-desk occupancy detection on average consumes 0.51 kWh of energy per day, which is a 40% reduction compared to the basic LED lighting system.

[17] Although, in general, it is possible to divide the space into smaller lighting control zones, this approach is impractical for smaller offices, such as our experimental testbed.

[18] "Basic lighting systems" refers to systems with neither occupancy detection nor daylight harvesting capability.

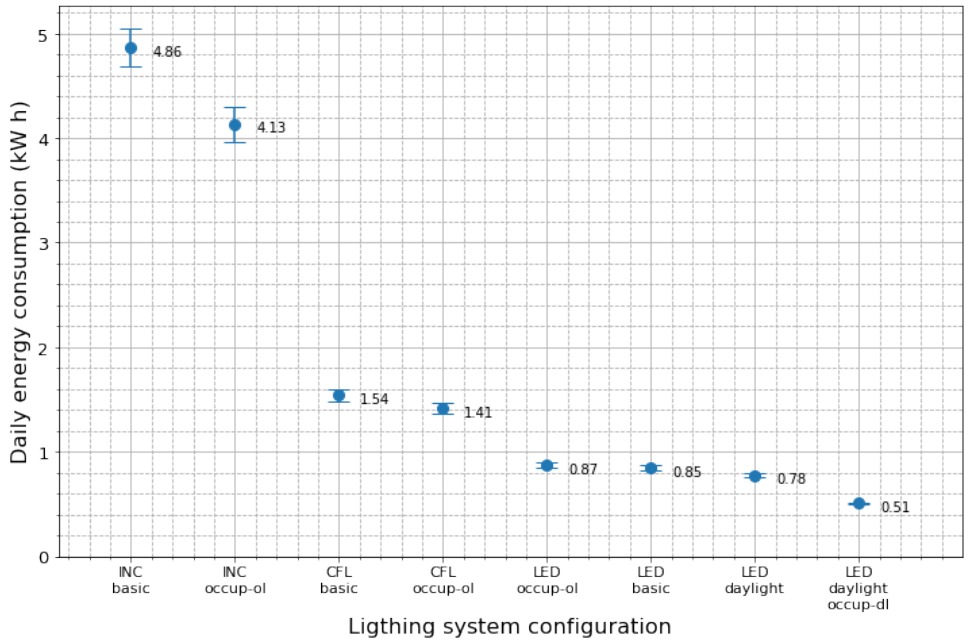


Figure 4.12: 95% confidence intervals for the average daily energy consumption of different lighting system configurations. **INC**: incandescent bulbs, **CFL**: CFL bulbs, **LED**: LED bulbs, **basic**: "basic" lighting system with neither occupancy detection nor daylight harvesting, **daylight**: daylight harvesting, **occup-ol**: office-level occupancy detection, **occup-dl**: desk-level occupancy detection.

.

Recall that there is an extra energy cost associated with lighting systems' sensing and actuation being wireless. Specifically, as mentioned above, 1.0 W of power is consumed by each wireless sensing module. Also, as discussed in Section 2.2, each PAR38 Philips Hue bulb has a standby power consumption of 1.18 W, which does not directly contribute to the bulb's luminous output. Thus, if one is willing to give up the system's flexibility and the ease of installation and deployment, more energy can be saved by using a wired implementation of sensing and actuation, discussed in Sections 3.2.5 and 3.3.1, respectively.

Figure 4.13 demonstrates a total energy consumption comparison between different LED lighting systems with the wired and wireless implementation of sensing and actuation. It should be pointed out that the average daily energy consumption of a fully wired implementation of the smart lighting system is only 0.24 kWh. This is less than a half of what a fully wireless implementation of the smart lighting system consumes and less than a third of what the basic LED lighting system consumes. The smart lighting system with wired actuation (and wireless sensing) consumes 0.34 kWh and the same system with wired sensing (and wireless actuation) consumes 0.41 kWh, which is respectively a 33% and a 20% reduction in the average daily energy consumption, compared to the fully wireless implementation. Note that even though the wired implementation of sensing and actuation leads to further reduction in energy consumption, it can potentially impose significant installation and deployment efforts, along with the associated costs.

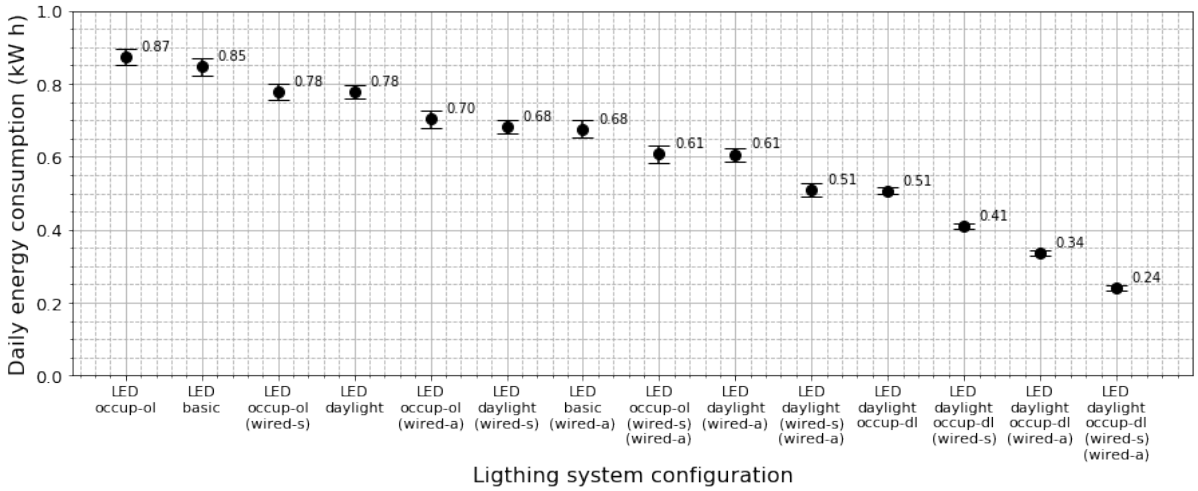


Figure 4.13: 95% confidence intervals for the average daily energy consumption of different LED lighting system configurations. **LED**: LED bulbs, **basic**: "basic" lighting system with neither occupancy detection nor daylight harvesting, **daylight**: daylight harvesting, **occup-ol**: office-level occupancy detection, **occup-dl**: desk-level occupancy detection, **wired-s**: wired sensing, **wired-a**: wired actuation. (Sensing and actuation are wireless, unless stated otherwise.)

Chapter 5

Conclusions

5.1 Meeting Design Goals

In this work, we have presented a power-efficient smart lighting control system designed and implemented with the following goals in mind:

- **G1** Saving energy
- **G2** Personalized lighting comfort
- **G3** System robustness to modeling errors
- **G4** Fast system responsiveness
- **G5** Plug-and-play deployment

In this section, we discuss how each of these goals is met.

To achieve energy savings (goal **G1**), the smart lighting system attempts to minimize unnecessary office over-illumination by taking full advantage of the available daylight and non-occupancy of work stations. Thus, the system uses artificial lighting only to compensate for insufficient daylight. Our evaluation has demonstrated that the smart lighting system reduces energy consumption by about 40%, when compared to a conventional system requiring all luminaires to be fully lit whenever the office is occupied.

Personalized lighting comfort (goal **G2**) is achieved by allowing each user to express individual lighting preference at their work station to the system via a "per-desk" sensing

module. The system, in turn, is able to maintain heterogeneous illumination in the office, while quickly responding to dynamically changing illuminance preferences of the occupants, as demonstrated by experiments.

The smart lighting system is robust (goal **G3**) to inaccurate calibration, accidental sensor movements, decreases in a bulb's luminous flux and other inaccuracies of the model, thanks to the proposed iterative control algorithm. It has been demonstrated both mathematically and empirically that the system is able to satisfy occupants' lighting preferences (goal **G2**), unless the estimated model substantially deviates from the "real" unknown model (i.e., $\widetilde{A} \approx \epsilon(t)$).

A concurrent implementation of the controller and sensing modules significantly contributes to the system's responsiveness (goal **G4**). As demonstrated by extensive evaluations, the system reacts to changes in environmental illuminance, such as daylight, within at most 2 seconds. At the same time, it reacts to changes in occupancy in 350 milliseconds, which is achieved by early termination of an ongoing iteration of the control loop.

Wireless communication between all components of the system makes its installation and deployment easy, rapid and cost-effective (goal **G5**). All that needs to be done to deploy the system is replacing currently installed bulbs by off-the-shelf wireless Philips Hue bulbs, putting a wireless sensing module on each work station, plugging the control module, and clicking a "start" button.

In addition, the proposed automated calibration method, which allows us to unobtrusively estimate an illuminance contribution on every light sensor from every luminaire, further contributes to the system's plug-and-play design. With this method, new system components, such as additional luminaires and sensing modules, can be seamlessly connected and disconnected, even while the system is in use.

5.2 Limitations and Future Work

Our work suffers from a few limitations. During the system modeling phase, we made several simplifying approximations. First, we did not model the effect of temperature on the luminaires' power consumption, assuming the power vs. dimming level relationship of the fully heated bulb. We also relaxed the luminaire model by linearizing it. These two approximations can potentially cause an error in a bulb's power consumption estimation. Another limitation is related to the estimation of the illuminance gains matrix, which is prone to an inherent time-varying error. Note that, in most cases, all these modeling errors

result in a slightly sub-optimal (in terms of power consumption) dimming level setting on luminaires, while not affecting occupants' comfort, due to the adaptive control algorithm.

The question of light sensor placement was not considered in detail in this study. Our work proposed placing light sensors on the top surface of a computer monitor, on an adjacent wall, or on the edge of a shelf to avoid sensor occlusion and alleviate shadowing. However, the question of sensor placement when none of these are available (for example, in a conference room), remains open. Also, our system measures illuminance near work surfaces and not directly on them. However, in practice, this is likely not an issue because users can readjust the desired illuminance levels to compensate for the sensor placement errors.

Another inherent limitation is a considerable standby power consumption of PAR38 Philips Hue bulbs. Even though the system is implemented with the PAR38 Philips Hue bulbs, other luminaires can be used too, provided that their dimming levels can be controlled, and the relationship between their power and dimming level is close to linear (which is the case for LED bulbs).

In future work, in addition to controlling luminaires, we could also consider controlling the daylight that enters the room. Such ability to curb the daylight would allow us to use an optimization that imposes both the lower and the upper bound on the illuminance of the sensors, not only the former as in the present study. Furthermore, to improve the system's ease of use, we could implement a graphical user interface (GUI) that would allow users to easily and intuitively set their illuminance preferences, as well as keep track of the levels of illumination at their work stations. Finally, we could also investigate power-efficient control strategies for lighting systems using luminaires with non-linear power vs. dimming level relationship.

References

[1] S. Afshari and S. Mishra. A plug-and-play realization of decentralized feedback control for smart lighting systems. *IEEE Transactions on Control Systems Technology*, 24(4):1317–1327, July 2016.

[2] Recommended lighting levels in buildings. https://www.archtoolbox.com/materials-systems/electrical/recommended-lighting-levels-in-buildings.html. Accessed: 2019-06-19.

[3] Laura Bellia, Francesca Fragliasso, and Emanuela Stefanizzi. Why are daylight-linked controls (DLCs) not so spread? A literature review. *Building and Environment*, 106:301 – 312, 2016.

[4] S. Borile, A. Pandharipande, D. Caicedo, A. Cenedese, and L. Schenato. An identification approach to lighting control. In *2016 European Control Conference (ECC)*, pages 637–642, June 2016.

[5] S. Borile, A. Pandharipande, D. Caicedo, L. Schenato, and A. Cenedese. A data-driven daylight estimation approach to lighting control. *IEEE Access*, 5:21461–21471, 2017.

[6] G. Boscarino and M. Moallem. Daylighting control and simulation for LED-based energy-efficient lighting systems. *IEEE Transactions on Industrial Informatics*, 12(1):301–309, Feb 2016.

[7] D Caicedo, S Li, and A Pandharipande. Smart lighting control with workspace and ceiling sensors. *Lighting Research & Technology*, 49(4):446–460, 2017.

[8] D. Caicedo and A. Pandharipande. Distributed illumination control with local sensing and actuation in networked lighting systems. *IEEE Sensors Journal*, 13(3):1092–1104, March 2013.

[9] D. Caicedo and A. Pandharipande. Sensor-driven lighting control with illumination and dimming constraints. *IEEE Sensors Journal*, 15(9):5169–5176, Sep. 2015.

[10] D Caicedo and A Pandharipande. Daylight and occupancy adaptive lighting control system: An iterative optimization approach. *Lighting Research & Technology*, 48(6):661–675, 2016.

[11] D. Caicedo, A. Pandharipande, and G. Leus. Occupancy-based illumination control of LED lighting systems. *Lighting Research & Technology*, 43(2):217–234, 2011.

[12] David Caicedo, Ashish Pandharipande, and Frans M.J. Willems. Daylight-adaptive lighting control using light sensor calibration prior-information. *Energy and Buildings*, 73:105 – 114, 2014.

[13] 27 W full spectrum CFL light bulb 5500 K, 1300 lumens (Amazon page). https://www.amazon.com/ALZO-Spectrum-Light-Lumens-Daylight/dp/B0018OS06S. Accessed: 2019-08-01.

[14] Ivan Chew, Dilukshan Karunatilaka, Chee Pin Tan, and Vineetha Kalavally. Smart lighting: The way forward? Reviewing the past to shape the future. *Energy and Buildings*, 149:180 – 191, 2017.

[15] Christel de Bakker, Myriam Aries, Helianthe Kort, and Alexander Rosemann. Occupancy-based lighting control in open-plan office spaces: A state-of-the-art review. *Building and Environment*, 112:308 – 321, 2017.

[16] Xisheng Ding, Junqi Yu, and Yifang Si. Office light control moving toward automation and humanization: a literature review. *Intelligent Buildings International*, 0(0):1–32, 2018.

[17] L. Doulos, A. Tsangrassoulis, and F.V. Topalis. Multi-criteria decision analysis to select the optimum position and proper field of view of a photosensor. *Energy Conversion and Management*, 86:1069 – 1077, 2014.

[18] Anca D. Galasiu and Jennifer A. Veitch. Occupant preferences and satisfaction with the luminous environment and control systems in daylit offices: a literature review. *Energy and Buildings*, 38(7):728 – 742, 2006. Special Issue on Daylighting Buildings.

[19] Lisa Heschong. Daylighting and human performance. *ASHRAE Journal*, 44:65–67, 06 2002.

[20] M.H. Toufiq Imam, Sina Afshari, and Sandipan Mishra. An experimental survey of feedback control methodologies for advanced lighting systems. *Energy and Buildings*, 130:600 – 612, 2016.

[21] Energy guide: How do I read the energy label of a light bulb? https://www.energuide.be/en/questions-answers/how-do-i-read-the-energy-label-of-a-light-bulb/50/. Accessed: 2019-08-01.

[22] Nandha Kumar Kandasamy, Giridharan Karunagaran, Costas Spanos, King Jet Tseng, and Boon-Hee Soong. Smart lighting system using ANN-IMC for personalized lighting control and daylight harvesting. *Building and Environment*, 139:170 – 180, 2018.

[23] M. T. Koroglu and K. M. Passino. Illumination balancing algorithm for smart lights. *IEEE Transactions on Control Systems Technology*, 22(2):557–567, March 2014.

[24] Tatiana Lashina, Sanae van der Vleuten-Chraibi, Marija Despenic, Paul Shrubsole, Alexander Rosemann, and Evert van Loenen. A comparison of lighting control strategies for open offices. *Building and Environment*, 149:68 – 78, 2019.

[25] How are LEDs affected by heat? https://www.lrc.rpi.edu/programs/nlpip/lightinganswers/led/heat.asp. Accessed: 2019-06-19.

[26] Documentation for TSL-2561 light-to-digital converter. https://cdn-shop.adafruit.com/datasheets/TSL2561.pdf. Accessed: 2019-06-18.

[27] Evangelos-Nikolaos D. Madias, Panagiotis A. Kontaxis, and Frangiskos V. Topalis. Application of multi-objective genetic algorithms to interior lighting optimization. *Energy and Buildings*, 125:66 – 74, 2016.

[28] S. Matta and S. M. Mahmud. An intelligent light control system for power saving. In *IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society*, pages 3316–3321, Nov 2010.

[29] M. Miki, A. Amamiya, and T. Hiroyasu. Distributed optimal control of lighting based on stochastic hill climbing method with variable neighborhood. In *2007 IEEE International Conference on Systems, Man and Cybernetics*, pages 1676–1680, Oct 2007.

[30] G. Newsham, M. Aries, S. Mancini, and G. Faye. Individual control of electric lighting in a daylit space. *Lighting Research and Technology*, 40:514 – 523, 2008.

[31] G. Newsham and J. Veitch. Lighting quality recommendations for VDT offices: a new method of derivation. *Transactions of the Illuminating Engineering Society*, 33(2):97–113, 2001.

[32] G. Newsham, J. Veitch, C. Arsenault, and C. Duval. Effect of dimming control on office worker satisfaction and performance. *Proceedings of the IESNA Annual Conference*, pages 19–41, 01 2004.

[33] Onion website: Omega2. https://onion.io/omega2/. Accessed: 2019-06-18.

[34] M. Pan, L. Yeh, Y. Chen, Y. Lin, and Y. Tseng. A WSN-based intelligent light control system considering user activities and profiles. *IEEE Sensors Journal*, 8(10):1710–1721, Oct 2008.

[35] A. Pandharipande, D. Caicedo, and X. Wang. Sensor-driven wireless lighting control: System solutions and services for intelligent buildings. *IEEE Sensors Journal*, 14(12):4207–4215, Dec 2014.

[36] A Pandharipande and GR Newsham. Lighting controls: Evolution and revolution. *Lighting Research & Technology*, 50(1):115–128, 2018.

[37] A. Pandharipande, M. Rossi, D. Caicedo, L. Schenato, and A. Cenedese. Centralized lighting control with luminaire-based occupancy and light sensing. In *2015 IEEE 13th International Conference on Industrial Informatics (INDIN)*, pages 31–36, July 2015.

[38] Ashish Pandharipande and David Caicedo. Daylight integrated illumination control of LED systems based on enhanced presence sensing. *Energy and Buildings*, 43(4):944 – 950, 2011.

[39] Ashish Pandharipande and David Caicedo. Smart indoor lighting systems with luminaire-based sensing: A review of lighting control approaches. *Energy and Buildings*, 104:369 – 377, 2015.

[40] Python API for Philips Hue luminaires (github page). https://github.com/studioimaginaire/phue. Accessed: 2019-07-24.

[41] Philips hue website. https://www2.meethue.com/en-us/p/hue-white-single-par38-outdoor/046677476816. Accessed: 2019-06-19.

[42] Documentation for AMN-32111 passive infrared sensor. https://eu.mouser.com/ProductDetail/Panasonic-Industrial-Devices/AMN32111?qs=sEN%2FkO1EG6bJ5oBrD44pvw==. Accessed: 2019-06-18.

[43] Alimohammad Rabbani and S. Keshav. The SPOT* personal thermal comfort system. In *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*, BuildSys '16, pages 75–84, New York, NY, USA, 2016. ACM.

[44] Radiance synthetic imaging system. https://floyd.lbl.gov/radiance/index.html. Accessed: 2019-07-19.

[45] B. Roisin, M. Bodart, A. Deneyer, and P. DHerdt. Lighting energy savings in offices using different control systems and their real consumption. *Energy and Buildings*, 40(4):514 – 523, 2008.

[46] Raspberry Pi website. https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/. Accessed: 2019-06-18.

[47] F. Rubinstein, M. Siminovitch, and R. Verderber. Fifty percent energy savings with automatic lighting controls. *IEEE Transactions on Industry Applications*, 29(4):768–773, July 1993.

[48] Francis Rubinstein. Photoelectric control of equi-illumination lighting systems. *Energy and Buildings*, 6(2):141 – 150, 1984.

[49] Francis Rubinstein, Gregory Ward, and Rudy Verderber. Improving the performance of photo-electrically controlled lighting systems. *Journal of the Illuminating Engineering Society*, 18(1):70–94, 1989.

[50] Y. K. Tan, T. P. Huynh, and Z. Wang. Smart personal sensor network control for energy saving in DC grid powered LED lighting system. *IEEE Transactions on Smart Grid*, 4(2):669–676, June 2013.

[51] D. Tran and Y. K. Tan. Sensorless illumination control of a networked LED-lighting system using feedforward neural network. *IEEE Transactions on Industrial Electronics*, 61(4):2113–2121, April 2014.

[52] Mohammad Asif ul Haq, Mohammad Yusri Hassan, Hayati Abdullah, Hasimah Abdul Rahman, Md Pauzi Abdullah, Faridah Hussin, and Dalila Mat Said. A review on lighting control technologies in commercial buildings, their performance and affecting factors. *Renewable and Sustainable Energy Reviews*, 33:268 – 279, 2014.

[53] Niels van de Meugheuvel, Ashish Pandharipande, David Caicedo, and P.P.J. van den Hof. Distributed lighting control with daylight and occupancy adaptation. *Energy and Buildings*, 75:321 – 329, 2014.

[54] JA Veitch, GR Newsham, PR Boyce, and CC Jones. Lighting appraisal, well-being and performance in open-plan offices: A linked mechanisms approach. *Lighting Research & Technology*, 40(2):133–151, 2008.

[55] Zizhen Wang and Yen Kheng Tan. Illumination control of LED systems based on neural network model and energy optimization algorithm. *Energy and Buildings*, 62:514 – 521, 2013.

[56] Y.-J. Wen and A. Agogino. Control of wireless-networked lighting in open-plan offices. *Lighting Research and Technologyy*, 43:235 – 248, 2011.

[57] Yao-Jung Wen and Alice M. Agogino. Personalized dynamic design of networked lighting for energy-efficiency in open-plan offices. *Energy and Buildings*, 43(8):1919 – 1924, 2011.

[58] Alison Williams, Barbara Atkinson, Karina Garbesi, Erik Page, and Francis Rubinstein. Lighting controls in commercial buildings. *LEUKOS*, 8(3):161–180, 2012.

[59] Yao-Jung Wen and A. M. Agogino. Wireless networked lighting systems for optimizing energy savings and user satisfaction. In *2008 IEEE Wireless Hive Networks Conference*, pages 1–7, Aug 2008.

[60] L. Yeh, C. Lu, C. Kou, Y. Tseng, and C. Yi. Autonomous light control by wireless sensor and actuator networks. *IEEE Sensors Journal*, 10(6):1029–1041, June 2010.

[61] Zigbee wireless protocol (wikipedia page). https://en.wikipedia.org/wiki/Zigbee. Accessed: 2019-07-19.

APPENDICES

Appendix A

Effect of Temperature on the PAR38 Philips Hue Bulb's Performance

The luminous power output of LED bulbs depends on their temperature [25]. To investigate this, we conducted the following experiment for the PAR38 Philips Hue bulbs. By turning the bulb fully on at time $t_0 = 0$, at which point it is at the room temperature, we measured how illuminance on an illuminance sensor, which is located 1 meter away from the bulb, changes over time as the bulb get warmer. All other sources of light were turned off during this experiment, and there was no daylight. Our findings are shown in Figure A.1, from which one can see that, over 35 minutes, the luminaire lost about 5.8% of its luminous output.

On the other hand, the bulb's dimming level d_j, being a measure of *relative* luminous output, is time-independent. To confirm this, we ran an experiment similar to the previous one, but this time every $\sim$10 minutes we measured illuminance on the sensor, as the bulb's brightness control value b_j was varied from "off" to 255, and calculated corresponding dimming levels using Equation 2.1. The results of this experiment, shown on Figure A.2, demonstrate that the dimming level vs. brightness control value relationship does not change, as the luminaire warms up.

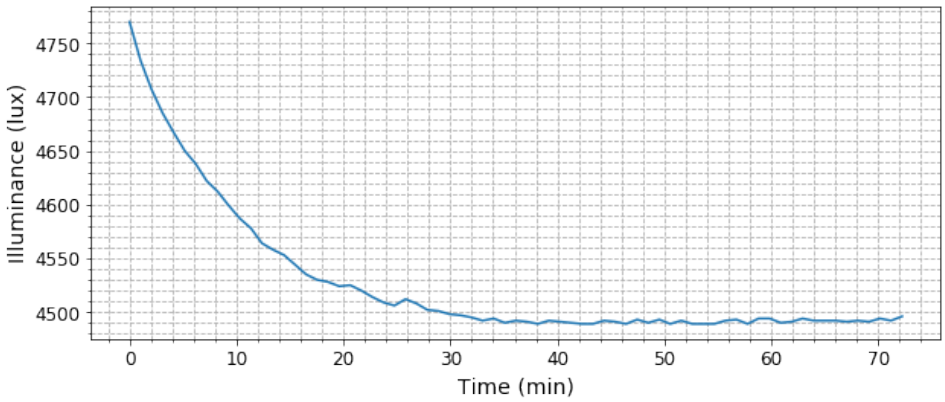


Figure A.1: The reduction in luminous power output of a fully lit PAR38 Philips Hue bulb, as a it heats up. At time $t_0 = 0$ the luminaire is at the room temperature. (Note that the vertical axis does not originate at 0.)

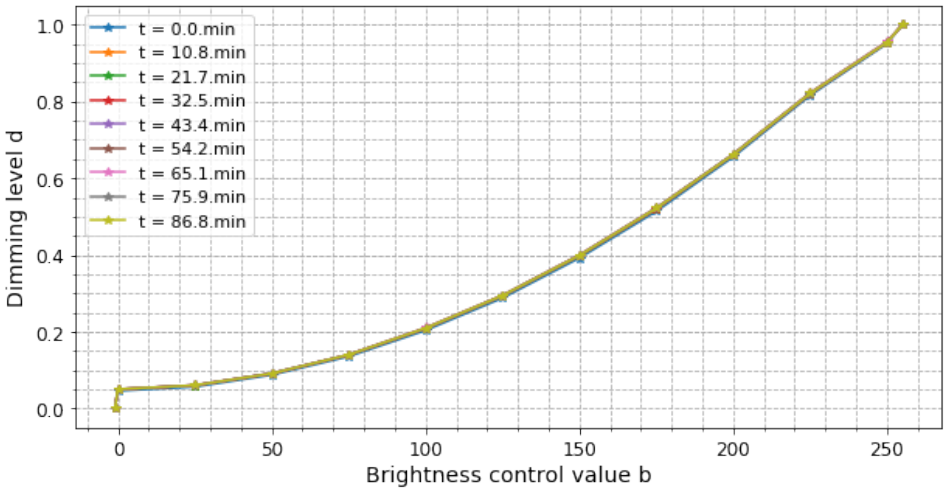


Figure A.2: This experiment demonstrates that the dimming level vs. brightness control value relationship does not change, as a PAR38 Philips Hue luminaire heats up. At time $t_0 = 0$ the luminaire is at the room temperature.

Appendix B

Estimates of Matrix A

Presented in this Appendix are the estimated illuminance gains matrices $\widetilde{A}$ from experiments D1-D5, discussed in Section 4.2.5.

	0	1	2	3	4	5	6	7
0	163.89	4.10	13.66	322.32	2.73	5.46	21.85	10.93
1	4.05	185.04	12.16	4.05	401.14	21.61	5.40	10.81
2	190.44	6.75	9.45	17.56	5.40	9.45	297.14	21.61
3	7.34	199.69	8.81	5.87	16.15	414.06	7.34	20.56

(a) Estimated matrix for Experiment D1.

	0	1	2	3	4	5	6	7
0	81.95	2.73	12.29	315.50	2.73	6.83	6.83	9.56
1	5.40	185.04	13.51	6.75	321.45	8.10	8.10	13.51
2	182.34	4.05	6.75	2.70	2.70	8.10	186.39	17.56
3	2.94	7.34	7.34	2.94	0.00	314.22	5.87	17.62

(b) Estimated matrix for Experiment D2.

	0	1	2	3	4	5	6	7
0	5.46	4.10	12.29	311.40	2.73	4.10	4.10	9.56
1	5.40	8.10	13.51	5.40	314.70	4.05	6.75	9.45
2	183.69	5.40	8.10	13.51	2.70	5.40	216.10	17.56
3	2.94	16.15	7.34	2.94	1.47	270.17	5.87	19.09

(c) Estimated matrix for Experiment D3.

	0	1	2	3	4	5	6	7
0	6.83	6.83	15.02	274.52	6.83	8.19	8.19	16.39
1	1.35	31.06	6.75	1.35	208.00	1.35	4.05	10.81
2	172.88	5.40	5.40	2.70	4.05	6.75	31.06	17.56
3	2.94	7.34	5.87	1.47	1.47	199.69	7.34	19.09

(d) Estimated matrix for Experiment D4.

	0	1	2	3	4	5	6	7
0	154.33	4.10	12.29	5.46	4.10	5.46	9.56	10.93
1	5.40	185.04	13.51	5.40	140.47	6.75	9.45	12.16
2	116.15	4.05	5.40	2.70	1.35	6.75	28.36	17.56
3	2.94	7.34	5.87	2.94	1.47	149.77	5.87	19.09

(e) Estimated matrix for Experiment D5.

Figure B.1: Estimated illuminance gains matrices for experiments D1-D5.