POLITECNICO DI MILANO

Corso di Laurea Magistrale in Ingegneria Informatica

Dipartimento di Elettronica, Informazione e Bioingegneria

ThermoSense: a complaint-based approach for thermal comfort control in indoor environments

Relatore: Prof. Marco D. Santambrogio

Correlatore: Francesco Trovò, Ph.D.

Tesi di Laurea di:

Piercarlo Serena

Matricola: 800419

Anno Accademico 2015-2016

ThermoSense

"Some people are thermometers, some are thermostats. You're a thermostat. You don't register the temp in the room, you change it."

André Agassi

Abstract

In the last decades, smart environments received a lot of attention both from academic and industrial research due to the increasing demand of systems able to automatically manage environments such as homes and buildings. In the design of smart environments different goals are taken into account, such as energy savings and usability. While the former is a topic widely explored both in literature and industry, usability still has to be improved. In fact, the standard control-based approach for environmental systems (like thermostats) does not allow occupants to express their real desires: in most cases they do not know their ideal environmental settings, and the multiplicity of occupants (an issue present in many open spaces such as libraries, laboratories and gyms) results into an arise of conflicts. All these issues cause an over-heating or over-cooling of the ambient, which from an economic point of view it means a monetary waste. On the other hand, often the occupants do not have even any control on the environment, thus they remain forced to the will of building administrators and their energy savings policies, which provides no thermal comfort at all to the occupants, ensuring them to work into an unpleasant environment. Studies prove that this condition can drastically decrease the occupants' productivity.

In contrast, the proposed solution is based on thermal comfort feedback: using a participatory sensing approach based on mobile technologies, occupants provide their actual thermal complaints, which are collected along with sensed environmental measurements like temperature. Using machine learning techniques, the system provides a thermal comfort model tailored for both individuals and groups of occupants, which is then updated every time an user submits a new complaint. Starting from this model, the system provides the best actuation in terms of temperature that maximizes the energy savings while satisfying the thermal comfort needs of the users.

Evaluation conducted through simulation gives an improvement in power consumption efficiency within the range of 15-20% compared with the state-of-the-art, while maintaining similar performance on control action and user comfort.

Sommario

NEgli ultimi decenni, l'ambito degli *Ambienti Smart* ha ricevuto una grande attenzione sia dalla ricerca accademico che industriale, a causa della crescente domanda per sistemi in grado di gestire autonomamente ambienti come case ed edifici. Durante la progettazione di Ambienti Smart entrano in gioco diversi obbiettivi, tra cui il risparmio energetico e l'usabilità da parte degli utenti. Mentre il primo obbiettivo è un argomento largamente esplorato sia in letteratura che in ambito industriale, la ricerca legata all'usabilità deve ancora essere perfezionata.

Infatti, l'approccio per il controllo usato dai sistemi ambientali standard (come i termostati) non permettono alle persone al proprio interno di esprimere i loro reali desideri: nella maggior parte dei casi ignorano quale sia il loro setting ambientale ideale, and il fatto di avere molte persone all'interno dello stesso ambiente (problematica riscontrabile non sono negli uffici, ma anche in molti *open space* come biblioteche, laboratori e palestre) può essere fonte di conflitti. Tutte queste considerazioni causano generalmente un utilizzo sbagliato del sistema di controllo, generalmente traducibile in un sovra-riscaldamento o sovra-raffreddamento dell'ambiente, il quale da un punto di vista economico rappresenta uno spreco di soldi. Dall'altra parte, spesso le persone non hanno la possibilità di controllare

l'ambiente circostante, spesso rimanendo costretti alla volontà degli amministratori e delle loro politiche di risparmio che non considerano affatto il comfort termico degli utenti, obbligandoli a lavorare in ambienti non gradevoli. E' stato dimostrato che questa condizione di discomfort può ridurre drasticamente la produttività dei lavoratori.

In contrasto con questa tendenza, la soluzione proposta è basata sul feedback relativo al comfort termico degli utenti: usando un approccio *participatory sensing*, basato su tecnologie mobile, le persone presenti in uno spazio al chiuso possono inviare lamentele sul loro stato termico. Queste lamentele vengono registrate assieme a misurazioni ambientali come la temperatura. Utilizzando poi tecniche di machine learning, il sistema fornisce un modello di comfort termico su misura sia individualmente a ogni singola persona sia per gruppi di persone, che viene aggiornato ogniqualvolta un utente invia una nuova lamentela. Da questo modello, il sistema fornisce il miglior valore di attuazione possibile in termini di temperatura che massimizza il risparmio energetico soddisfacendo le necessità di comfort termico degli utenti.

Valutazioni sperimentali condotte tramite simulazione indicano un miglioramento nell'efficienza del consumpo di potenza compreso tra il 15 e il 20% rispetto allo stato dell'arte, il tutto conservando performance simili relativamente al controllo ambientale e comfort degli utenti.

Contents

List of Figures

List of Tables

CHAPTER 1

Context Definition

In this chapter, we are going to provide an introduction to the Smart Buildings and Building Automation context, focusing on the main goals of a Building Management Systems. We are going to explain the reasons why it is important to maintain an environment as comfortable as possible while minimizing the energy. We are going to show also the issues and the conflicts that arise in assessing these goals.

1.1 Introduction

Nowadays, all buildings present electrical and mechanical devices for the control of environment such as Heating, Ventilation and Air-Conditioning (HVAC)[28]. These types of systems can vary from a simple switch for powering on or off the heating and cooling, to a classic thermostat, which is able to power itself on when the temperature decreases over a certain threshold, to the most recent automation systems which can learn the occupant's preferences and control autonomously the environment [47]. These types of computer-based control systems are generally referred as Home and Building Automation Systems (HBAS) [45]. These controllers often offer a wide selection of functionalities apart from the environmental control, such as lighting and electrical power control, fire alarm, security automation and occupants detection.

1.1.1 Smart Building: definition, features and goals

In the context of Building Automation, the term *Smart Buildings* refers to those building where there is a significant number of heterogeneous objects, such as sensor, actuators, control devices, protocols, all of them cooperating in order to monitor the indoor environment and to autonomously take decision actions over it [63], with the main goal of increasing and maintaining the comfort of its occupants as high and constant as possible [11]. These systems are usually controlled by a Building Management System (BMS) able to automatically manage and self-organize under some user-specified rules, e.g., energy saving policies [46].

Narrowing the concept of user comfort, usually, in the field of Smart Buildings, the aspect that influent mostly the user wellness is the *thermal comfort*, which is the user satisfaction with the surrounding environment with respect to the thermal factors affecting it, such as temperature, humidity, ventilation, air quality.

Thus, the main goal that a Smart Building must assess is to maintain the environment in a thermal comfortable status for the occupants, or at least in a Thermal Neutrality Zone, which is the temperature range where the users are feeling neither uncomfortable nor comfortable (see Section 1.2.1 for a formal definition of Thermal Comfort).

Building management systems

From an architectural point of view, a smart building can be seen as organized into an Hardware level, a Management level and a Human Interface level [45], as shown in Figure 1.1.

The Hardware level is composed by a network of sensors, used for collecting environmental data such as temperature, humidity, luminosity, user location, etc..., and a network of actuators, able to modify the environment, such as HVAC systems, lights, blinds, windows. These devices can vary with respect to the amount of "intelligence" they carry: for instance, in the market it is possible to find from simply devices attachable to the Internet to intelligent thermostat able to learn user occupancy patterns for automating actuation, such as Nest Learning Thermostat [47], Ecobee Wifi Thermostat [19] and Honeywell Lyric Thermostat [30]

The Management level represents an abstraction layer over the Hardware level. It is composed by a BMS which works as a gateway, gathering all the data from the sensor network, as a decision making engine, computing the action to be executed, and as a deliver to these action to the actuator network. Some examples are represented by Building Depot [2], BOSS [14] and Building Rules [44].

Finally, the most recent smart buildings provide also a human interface that can simplify the interaction with the system. They are often developed as web-based or smartphone app-based services, and provide the occupants with features such as monitoring, system programming and feedback provider, such as Zonepac [4] and Energy@Home [20].

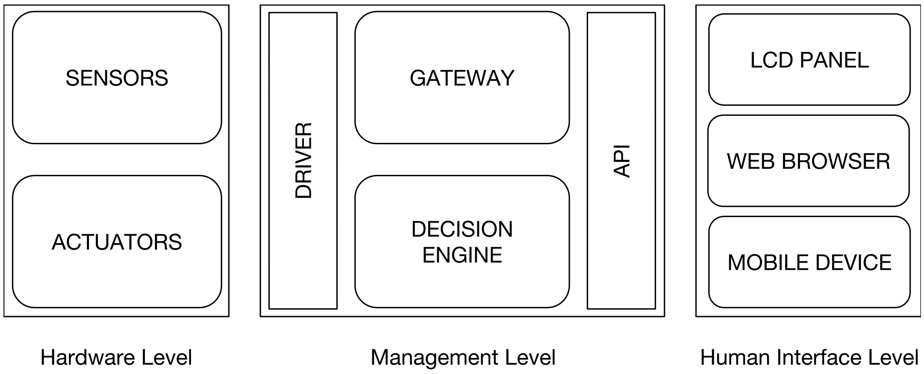


Figure 1.1: Standard architecture of a smart building

Environmental control

An HVAC is a system whose technology aims in providing environmental comfort (specifically, thermal comfort) and indoor air quality inside a building. Referring to the architecture scheme shown in Figure 1.1, they can be seen as the main actuator for the thermal environment. In normal building, an HVAC can usually be programmed and interacted to with a thermostat. The complexity of these systems vary with respect to the complexity and the nature of the building: residential buildings such as homes, apartment and hotels have simpler HVAC systems, with a central unit, simpler and fewer policies easily editable, and an unique thermostat. On the other hand, in complex and commercial buildings, these implants are often distributed, with multiple central units, very strict policies editable only by the administration and with several thermostat - or not at all.

A common point between residential and commercial HVAC systems are the type of control loop on which they work and perform the actuation over the environment: in fact, it is a traditional control loop based on set-point provided by the users, as shown in Figure 1.2. Modern thermostats have an underlining control loop of this type. The main problems are related to the way they interfaces with the users. First of all, not always the users know the exact set-point for having a sufficient satisfactory comfort and not an excessive energy consumption. This translates often in an over-heating or over-cooling of the environment, or a continuing interaction with the system, bringing to a waste of energy and/or not bringing any immediate comfortable environment. Secondly, these systems do not consider all the different preferences of the occupants: the inserted set-point is supposed to be good for all the people inside that room or building. Thirdly, this kind of control loop does not provide any information over the actuation, for example the amount of energy spent.

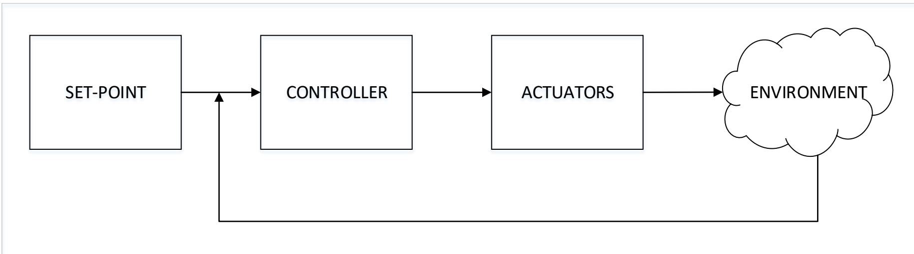


Figure 1.2: Traditional set-point-based control loop.

1.2 Problem Definition

In the context of thermal environment control, as already shown in the previous section, two main goal can be identified: *providing thermal comfort* to all the building occupants and *minimizing the energy consumption*. In the following section we are going to briefly present a theoretical introduction to the concepts of thermal comfort and energy consumption, for then presenting how these are the two main goals when considering the smart environmental control.

1.2.1 The need for thermal comfort

By definition, Thermal Comfort *is a subjective response, or state of mind, where a person expresses satisfaction with the thermal environment* [56]. Its assessment must be taken as a primary goal when designing a building, a Building Management Systems or simply a Heating, Ventilation and Air Conditioning Systems. Thermal Comfort is a subject that has been studied since the 1970s, and it has a strong theoretical model based on Heat transfer and Building engineering theory. Different models have been proposed throughout the decades. In what follows, two of them will be shown: The predicted mean vote (PMV) model and the Adaptive Comfort model.

ASHRAE Standard

The American Society of Heating, Refrigerating, and Air-Conditioning Engineers (ASHRAE) is the publisher of several standards and guidelines in the HVAC and Refrigeration systems. In the Standard 55 [56], they proposed a 7-point scale for comfort evaluation, with linguistic labels associated with each numerical value. The scale is shown in Table 1.1. A value equals to 0 corresponds to a neutral comfort status, or in other words a sensation of thermal comfort, whereas extreme values (+3 and -3) correspond to a total discomfort. Hence, people will vote with positive values when in hot environments and with negative values when in cold environments.

According to ASHRAE Standard 55, building designers must take into account the comfort of the occupants, providing a thermal environment able to maintain the comfort value as close as possible to 0. More specifically, a comfortable range is accepted between -0.5 and 0.5.

Table 1.1: ASHRAE 7-point comfort scale.

predicted mean vote	Linguistic Labels
+3	Hot
+2	Warm
+1	Slightly warm
0	Neutral
-1	Slightly cool
-2	Cool
-3	Cold

Static Model: PMV/PPD model

The PMV model was proposed by Fanger in 1970 [22], and later standardized by ASHRAE in the ISO 7730 Standard [32]. It was the first predicting model that formally related environmental and personal information to the individual thermal sensation.

The human comfort level is predicted by a set of four environmental and two personal variables, presented in Table 1.2.

Table 1.2: Input parameters for the PMV model.

Parameters	Sign	Unit measure	Type
Air Temperature	T_a	$°C$	Environmental
Mean Radiant Temperature	T_{mrt}	$°C$	Environmental
Relative Humidity	RH	%	Environmental
Air Velocity	V_a	m/s	Environmental
Activity Level	M	met	Personal
Clothing Insulation	I_{cl}	clo	Personal

Air Temperature It is the average temperature surrounding the occupant, depending on the time and its location inside the building. It is often called *dry-bulb temperature*, since it is measured with a dry-bulb thermometer, a thermometer exposed to air but protected from heat-source radiation and moisture (in contrast with the wet-bulb temperature). It is measured in degree Celsius ($°C$)

Mean Radiant Temperature It is related to the amount of radiant heat transferred from a surface to the human body. It depends on the material emissivity, that is the ability to absorb or emit heat, and temperature difference between the object and the human body. It can be calculated from the measured temperatures of the surrounding surfaces and their position with respect to the person. It is measured in degree Celsius ($°C$). Sometimes Air Temperature and Mean Radiant Temperature are combined into one unique metric called Operative Temperature.

Relative Humidity It is the ratio of the amount of water vapor in the air to the amount of water vapor in the air at a specific temperature and pressure. It is expressed as a percentage.

Air Velocity It it is the average speed of the air to which the body is exposed, with respect to location and time. It is usually considered uniform on the body exposure. Being a velocity, it is measured in m/s.

Activity Level According to the ASHRAE Standard 55 [56], metabolic rate is the level of transformation of chemical energy into heat and mechanical work by metabolic activities within an organism, usually expressed in terms of unit area of the total body surface. It depends on various factors, such as age, body shape, health, food and beverage intake. It is measured in met, where $1\ met = 58.2W/m^2$, a standard value corresponding to the energy produced per unit surface area by a person at rest. ASHRAE Standard 55 (and Fanger before it) provides tables with estimations of metabolic rates for several types of activities.

Clothing Insulation It refers to the thermal insulation provided by clothing. Thermal insulation is the reduction of heat transfer, usually under form of heat loss. As Activity Level, ASHRAE Standard 55 (and Fanger befor it) provides various tables with thermal insulation values for each kind of clothing and garments. It is measured in clo, where $1\ clo = 0.155m^2 \cdot K/W$, a standard value corresponding to the thermal insulation provided by trousers, a long-sleveed shirt and a jacket.

PMV model consists of a set of equations, developed on heat balance theory.

Given the above variables, PMV model predicts the mean comfort value of a group of occupants in the ASHRAE 7-point scale presented above 1.1.
Fanger equations are shown in the following:

$$
\begin{aligned}
PMV = \quad & (0.028 + 0.3033e^{-0.036M} \cdot \{(M - W) - 3.05[5.733 \\
& - 0.000699(M - W) - Pa] - 0.42[(M0 - W) - 58.15] \\
& - 0.0173M(5.867 - Pa) - 0.0014M(34 - T_a) \\
& - 3.96 \cdot 10^{-8} fcl[(T_{cl} + 273)^4 - (T_{mrt} + 273)^4] \\
& - fcl \cdot h_c(T_{cl} - T_a)\}
\end{aligned} \tag{1.1}
$$

where

$$
\begin{aligned}
T_{cl} = \quad & 35.7 - 0.028(M - W) - 0.155I_{cl}[3.9610^{-3} fcl[(T_{cl} + 273)^4 \\
& - (T_{mrt} + 273)^4] - fcl \cdot h_c(T_{cl} - T_a)]
\end{aligned} \tag{1.2}
$$

$$
h_c = \begin{cases} 2.38(T_{cl} - T_a)^{0.25} & for \quad 2.38(T_{cl} + T_a)^{0.25} \geq 12.1\sqrt{V_{air}} \\ 12.1\sqrt{V_{air}} & for \quad 2.38(T_{cl} - T_a)^{0.25} \leq 12.1\sqrt{V_{air}} \end{cases} \tag{1.3}
$$

Each parameter is defined as:

PMV: predicted mean vote;

M: metabolism (met);

W: rate of mechanical work accomplished (W/m^2);

I_{cl}: thermal insulation of clothing (clo);

fcl: ration of the surface area of the body when fully clothed to the surface are of the body when nude (clothing area), dimensionless;

T_a: air temperature ($^\circ C$);

T_{mrt}: mean radiant temperature ($^\circ C$);

V_{air}: relative air velocity (m/s);

Pa: partial water vapour pressure (Pa);

h_c: convective heat transfer coefficient ($W/(m^2 \cdot^\circ C)$);

T_{cl}: surface temperature of clothing ($^\circ C$);

Beside the PMV, Fanger developed also a complementary index, the *Predicted Percentage of Dissatisfied* (PPD), predicting the percentage of occupants that will be dissatisfied with the thermal comfort. It is a function of the PMV, defined as

$$PPD = \quad 100 - 95e^{[-(0.3353PMV^4+0.2179PMV^2)]} \tag{1.4}$$

Figure 1.3 shows how the PPD value changes with respect to PMV. We can notice one important key issue: it is impossible to satisfy all the occupants, as proved also by analyzing Equation (1.4). Mathematically, it is not possible to get a PPD value equals to 0%. This key point will get importance when considering the necessity of a trade-off between energy consumption and user satisfaction. However, ASHRAE standard recommends to maintain the PPD value less than 10%.

Fanger originally provided also various psychrometric charts that permits to compute the operative temperatures and humidity ranges within which the thermal comfort is achieved. Figure 1.4 shows an example of psychrometric chart, where combinations between Air Temperature and Relative Humidity are shown in order to highlight the comfortable region. The blue zone represents the comfort zone with a PPD<10% and PMV between -0.5 and 0.5.

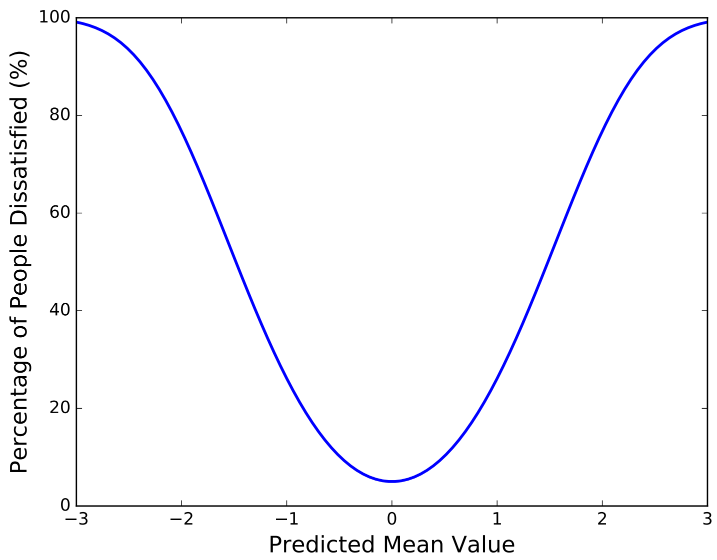


Figure 1.3: Predicted Percentage of Dissatisfied (people).

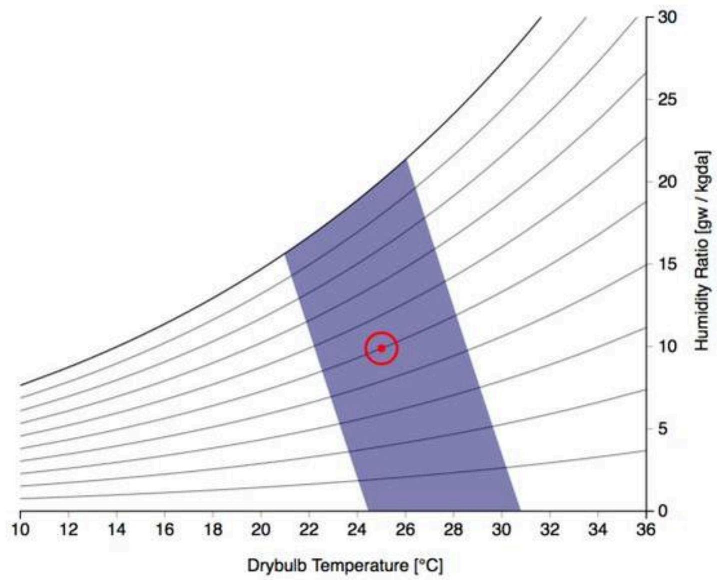


Figure 1.4: The Psychrometric chart representing the acceptable thermal region within which a comfortable status is provided (image taken from: `http://comfort.cbe.berkeley.edu/`).

The PMV/PPD model strength lies on the fact that it gives a direct tool for predicting human thermal comfort inside a building, without bothering in asking every time the occupants their thermal comfort feedback. Several HVAC systems relying on the PMV/PPD model has been proposed in literature, as Chapter 2 will show.

However, PMV/PPD model carries various weaknesses. First of all, PMV equations are complex to solve, due to their non-linearities. This has always been an issue for those HVAC systems based on this model. In addition, while the environmental factors can be easily measured through sensors, the personal ones must be calculated analytically or with complex and expensive equipment [31]. To remedy to this issue, Fanger proposed a set of tables for computing easily PMV values starting from several standard input values. Besides, several approaches have been proposed to speed up the PMV computation [[27], [55], [23]].
Secondly, PMV/PPD model has been designed on invariant environmental conditions, without considering outdoor conditions or seasonality, and considering all the occupants with the same characteristics. The original approach takes as fundamental that humans cannot adapt naturally to the surrounding environment, thus considering constant environment and constant occupants [58].

Adaptive Comfort model

The Adaptive Comfort model, proposed by de Dear and Brager [15], is based instead on the idea that humans can adapt themselves to the surrounding environment, thus playing an active role in creating their own comfortable environment.

This adaption can be distinguished in three main categories:

Behavioral Adjustment all the changes a person can perform on himself (e.g., adding or removing a clothing layer, drinking a beverage, resting) or over the environment (e.g., turning on/off the cooling system).

Physiological the natural adaption resulting from an exposure to a specific thermal environment, which from a starting situation of discomfort can become comfortable. It can be genetic (changes from one generation to the others) or within an individual lifetime span (acclimatization).

Psychological it refers to an altered perception (often born as a reaction to) of sensory information due to past experience (previous interaction with the control system) and expectations.

Adaptive comfort model assess the problem from an opposite point of view than the PMV/PPD, which proposes a static approach where the environmental conditions do not change over time and, moreover, where the outdoor climate is not taken into consideration. In fact, PMV/PPD model predicts comfort temperature with a reasonable accuracy in most air-conditioning buildings whereas it fails when applying it to naturally-ventilated buildings.
Starting from ASHRAE-55 2010 Standard [57], an update of the 2004 corresponding, adopted the Adaptive model by introducing the mean outdoor temperature as input variable. Compared to the PMV/PPD model, in order to apply the adaptive model it is necessary that no mechanical cooling systems are working in the controlled environment, which must be totally natural conditioned and controllable by the occupants. De Dear and Brager showed that in this kind of environ-

ments, occupants are more tolerant to a wider range of temperatures.

Building occupants, whether the building is an home or an office, want to feel comfortable with the surrounding environment. They do not want to feel an excessively hot nor cold, excessively humid nor dry environment, too much ventilated nor too much still air. They prefer a status where their body does not feel any particular discomfort. This is obviously true for residential building, but it is also true for commercial ones. In fact, studies show that not taking care of thermal comfort occupants, especially in buildings, may lead to Sick Building Syndrome symptoms, such as eye and throat irritation, headaches, fatigue, lack of concentration, dryness, cough and weeze [33]. It has been shown also that not providing a comfortable working place such as offices may reduce productivity [37]. Therefore, it is in the managers interest to provide a comfortable environment to their employees.

1.2.2 Energy savings in houses and offices

When designing a building, it is important to take into consideration its energy efficiency, which is defined as:

$$\frac{Energy\ consumed}{Built\ area} \tag{1.5}$$

In reality, more factors are influent over the energy efficiency: for instance, the material that can be reuse when the building is dismantled and, moreover, the activity level of the building - that is, whether it is used or rarely occupied [34]. Thus, the Equation (1.5) becomes:

$$\frac{Embodied\ energy + Energy\ consumed - Energy\ recovered}{Built\ area * Utilization\ rate * Quality\ factor} \tag{1.6}$$

From Equation (1.6) it is evident how, for increasing energy efficiency, it is possible to operate on different factors. In this thesis it has been focused on the energy consumed factor.

For reducing the consumption of energy, several approaches are possible. First of all, the most straight-forward is to design the building in a green way, adopting strategies that take advantage of the surrounding environment, like orientated-facing windows against the sun for increasing the passive solar heating and natural lighting, or using trees and landscaping for providing shade and protecting from wind. Also, choosing insulating materials can be another effective strategy: it has been shown that designing energy efficient materials-made windows, doors and walls can help in reducing the heat loss.

Other approaches are identified by the adoption pf occupancy detection strategies, where using technologies like Bluetooth can help in identifying whether a building is occupied or not, and setting different energy policies as a consequence [10].

Since the most energy consumption derives from HVAC systems, the design choices when installing systems such those are fundamental. Adopting newer and more green heat pump (like ground source) can further decrease energy consumption. In addiction, the adoption of sustainable energy sources such as solar panels are always effective.

Another source of energy consumption is represented by the way occupants interact with the HVAC (and in second phase also to the electronic devices): often they tend to overheat or over-cool the environment, resulting in a waste of energy. It has been prove that making them aware about their energy consumption is an effective first weapon in energy savings [13]. This thesis aims to provide users an easy tool for avoiding over-heating and over-cooling. Other solutions are analyzed in Chapter 2.

Considering the energy consumed along the energy grid, it has been shown that the 70% is consumed by commercial buildings [1], of which an important contribution is given by HVAC and lighting systems [29]. Nowadays, energy efficiency is a concert that has been addressed also by institutions: European Commission targeted a reduction of 20% in energy efficiency in building within 2020 and 30% within 2030 [9].

The goal of reducing energy consumption not only has an sustainable reason, but also an economic one: the interests of managers and company holders (as well as families) are of course to reducing outcomes and expenses as much as possible.

1.2.3 Issues related to the thermal comfort control problem

Analyzing the traditional control loop, based on the exact set-point, we can identify three main issues:

- In residential building, the main problem is that the occupants do not know which optimal set-point to provide to the thermostat, turning into an inadequate interaction with the control system. As an example, consider a single occupant not satisfied with the environment cause he is feeling hot. He will then set a set-point with a smaller temperature. Two cases can occurs: the temperature is not small enough, meaning that the user will still feel uncomfortable and will have to set a new set-point; or the temperature is too small, meaning that this time the user may start feeling cold instead of hot, but on the other side there will be an excessive and useless energy consumption.

- In commercial and complex building, usually occupants do not have any kind of control over the environment, which is subjected to strict energy savings policies from the administration. Hence, if people can not control the environment, they can not control their own thermal comfort. This translates into a situation that if a person is feeling uncomfortable, he remains uncomfortable (unless to perform some adaptive actions) [7].

- People have different preferences about their thermal comfortable environment. When the number of people inside a room is high (for instance an office, a library, a gym, an airport), the probability that those preferences are conflicting drastically increase, possibly arising technical issues on the actuation system.[5].

Generalizing from the examples presented above, we can then state that the main problem is that the two main objectives for a thermal environment controllers, the occupants thermal comfort and the energy consumption minimization, are conflicting. Thus, we can see that the two goals are in a inversely proportional relationship, that is, the energy consumption minimization decreases when the occupant thermal comfort maximization increases. A practical example of this relationship is the borderline case: for maximizing the thermal comfort we can simply turn off the HVAC, but we will not provide any kind of thermal comfort to the occupants. This problem in the literature has been addressed extensively, and several solutions has been proposed. Chapter 2 presents the most influent works.

1.3 Motivating example

We present now a case study to experimentally validate the problem presented in the previous section. We conducted a survey in a study room at Politecnico di Milano, called "Acquario". The room has 200 seats and 12 computer desks. On the long sides the room is provided with stained glass and two double door entrances. The light is provided both by natural light coming from the outside and artificial light from the lamps attached to the walls and to the desks.
The room has a centralized HVAC systems, which can be controlled only by the administrators from a different room. Thus, the occupants have no kind of control over the environment. In addition, the heating and cooling are subjected to strict policies, common for all the environments in Politecnico (usually depending on specific season). This means that often the HVAC does not respond to the actual needing of the occupants, more over in the period of the year where the room is most occupied (e.g., during exam sessions).
We conducted the survey on May 18th, 2016, in an anonymous way. We interviewed 94 people, of which 64 male and 30 female. Regarding the age, only 2 were over 30 while all the others were between an age range of 21-30 (a pretty obvious result, since the environment is in a public university). The demographic results are synthesized in Table 1.3

Figure 1.5 and Figure 1.6 shows the frequency with which the occupants use

Table 1.3: General information of the survey.

Gender	Age		**Total**
	Under 30	31-50	
Male	62	2	**64**
Female	30	-	**30**
Total	62	2	94

the study room and the numbers of hours which they stay in there. It is interesting to see that the students that stay more for a prolonged period of time are also the students that use most frequently the study room, whereas students not used to this study room do not stay for a long time.

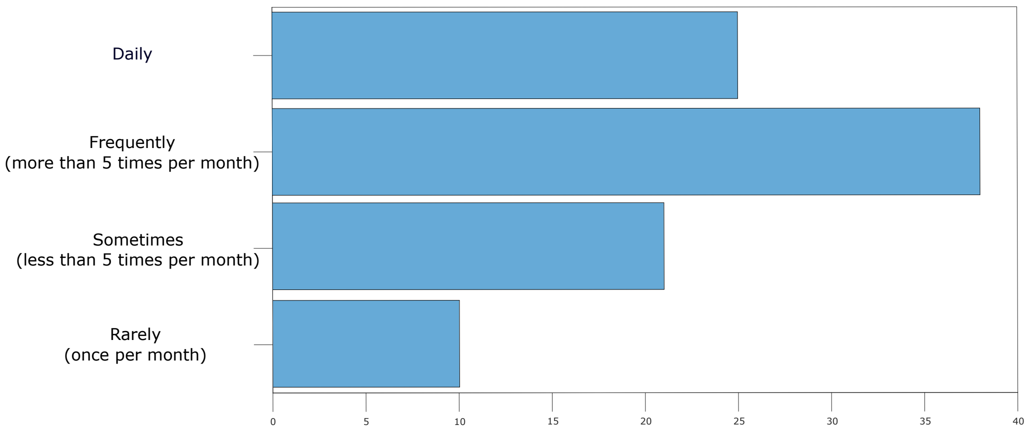


Figure 1.5: Day frequency with which the occupants use the study room.

Through the survey we asked question regarding the overall environmental satisfaction, the source of dissatisfaction and how having control over the envi-

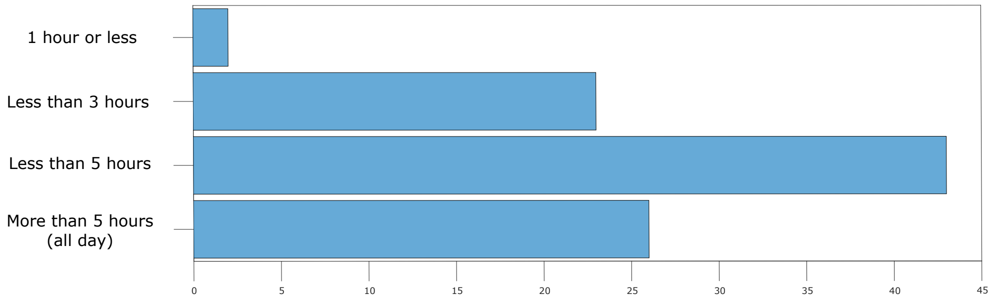


Figure 1.6: Avreage numbers of hours with which the occupants use the study room.

ronment through HVAC accessibility could help in achieving comfort according to the study room occupants. Regarding the overall satisfaction, the results are shown in Figure 1.7, where appears that students in that moment were satisfied with the environment. The cause can be identified by the fact that the room was not too crowded, and the outside spring weather was good.

Figure 1.8 shows the percentage of whether the occupants think their comfort would increase if they would have access to the control system, or not. It is interesting to see that the majority of the occupants are positively convinced about that. Figure 1.9 shows instead if they would know how to satisfy all the occupants by choosing a standard value. Also in this case, they are convinced that a standard value (e.g., 22°C) could achieve a common comfortable environment, without considering the personal difference between each individuals. Finally, Figure 1.10 shows which factors the occupants consider most important. Unsurprisingly, the majority of the people thinks more of their personal comfort than energy sustainability, which is more an administration issue.

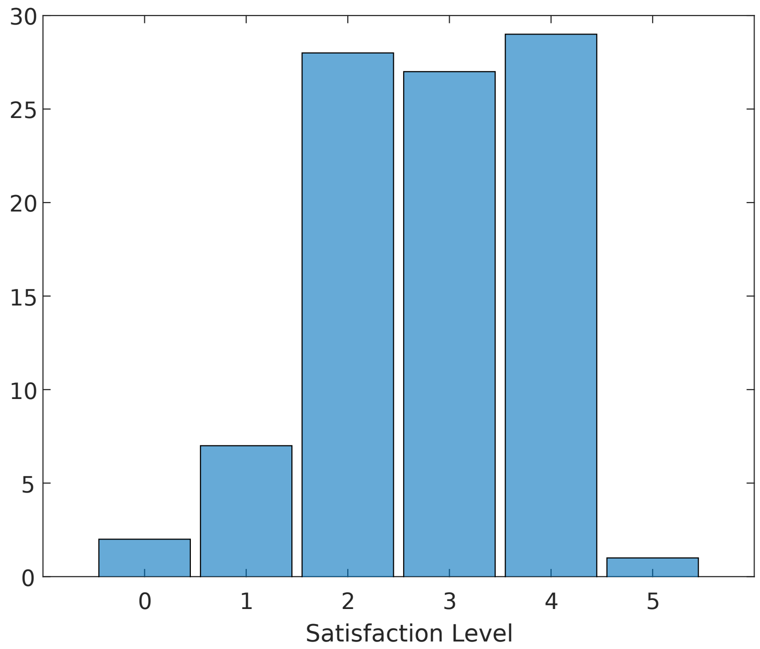


Figure 1.7: Overall satisfaction of the study room occupants.

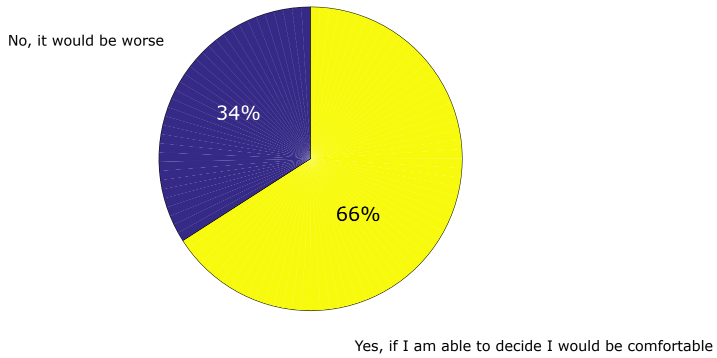


Figure 1.8: Results of the question: Do you think that having control over the environment could help you in achieving thermal comfort?

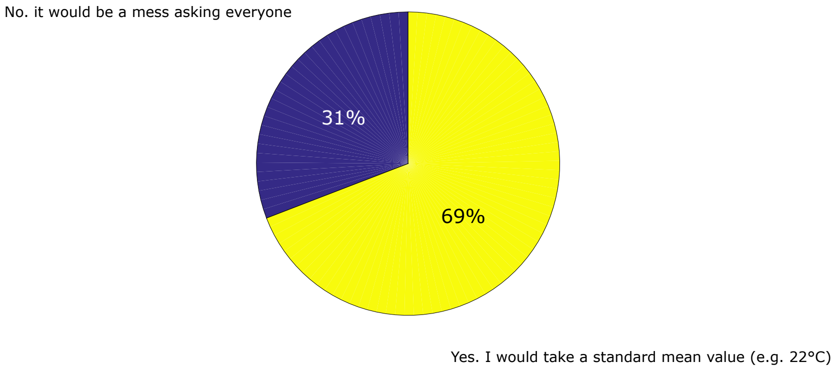


Figure 1.9: Results of the question: If you were able to control at least the thermostat, you would know which temperature to set in order to achieve a thermal comfort good for all?

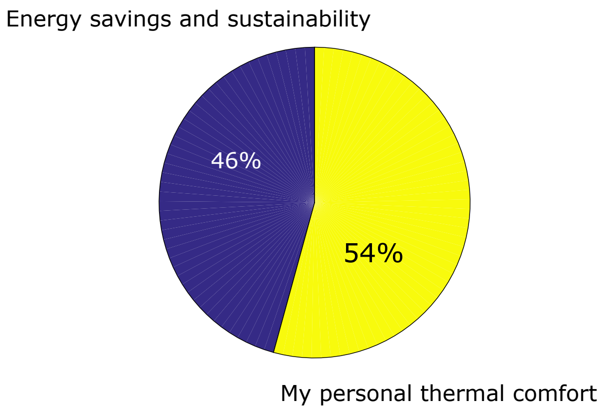


Figure 1.10: Results of the question: According to your value sets, which factors do you consider most important?

This results confirm our supposition regarding providing control to the occupants: they are convinced about the usefulness of having control over the environment, suggesting that providing an user-friendly and easy accessible HVAC could satisfy them all; they tend to rely on standard value, suggesting that an HVAC should provide a personalized comfort model; they tend to think first on themselves, suggesting that the HVAC should provide an underlining energy savings approach.

1.4 Contribution and outline

The aim of this thesis is to provide a control engine that is able to assess the conflicting problem of providing thermal comfort to occupants while minimize energy consumption. As the next Chapter will present, the current approach of let the people express their preferences through exact set-point (as most of the thermostats allow) may arise the issues presented above. In particular, the overheating and over-cooling of the ambient is caused by the ignorance of the occupants about they ideal temperature set-point, and the conflicts between different preferences in shared indoor environments.

In this thesis we present ThermoSense, a complaint-based approach that changes the paradigm of thermal comfort control, far stepping from the set-point based approach. ThermoSense is intended to be a full middle-layer system between the occupants and actuation system, interacting with both, and deciding autonomously which is the best actuation action in terms of thermal comfort and energy efficiency, relying only on the feedbacks of the occupants.

The thesis is structured as follows: Chapter 2 presents the current state of the art about environmental and thermal comfort control, with a focus on how they try to assess the problem presented in the previous sections. Chapter 3 shows the methodology behind the ThermoSense approach, while Chapter 4 focuses on its implementation. Chapter 5 gives an insight of the tests and the simulations conducted for the evaluation and shows the results obtained compared to the state-of-the-art. Finally, Chapter 6 will present the conclusions of this thesis.

CHAPTER 2

State of the Art

In this chapter we are going to give an analysis of the current state-of-the-art regarding environmental control in smart buildings, from an historical perspective. We start from the traditional set-point controllers towards comfort-based control, whose solutions are the comparison with our approach.

2.1 Introduction

The environmental control problem for assessing thermal comfort in indoor environment is an old problem, since the first HVAC appeared. The initial, and main goal, for an HVAC is *to provide and maintain a comfortable environment to the occupants*. Throughout the decades, this goal has been joined to the energy efficiency problem, that is, trying *to save as most energy as possible*. As shown in Chapter 1, these two objectives are conflicting, and a trade-off is often necessary. In the context of smart building, an HVAC should be able to assess this two goals in an autonomous way. ThermoSense assesses this multi-goal problem by providing the occupants a control system which adopts a *participatory sensing approach*, that it means that the users are involved through the collection of their feedback, and *relying only on their thermal complaints* when deciding which control action to take, instead of the exact numeric set-point. In addition, ThermoSense is developed using *mobile technologies* and in a *distributed architecture*, thus permitting the occupants to express their feedbacks even if they are usually not allowed to change the environmental setting, and without using a centralized controller device such as a thermostat.

In the following section, we are going to analyze the state-of-the-art with respect to these characteristics.

2.2 Set-point based control systems

As already shown in Chapter 1, the most widespread type of environmental control is the set-point based. This type of control system relies on the exact actuation value (e.g., Temperature) to be set on the control device. When the value to be controlled changes over a specific threshold, the control device actuates again, closing the loop. Figure 2.1 shows graphically the loop.

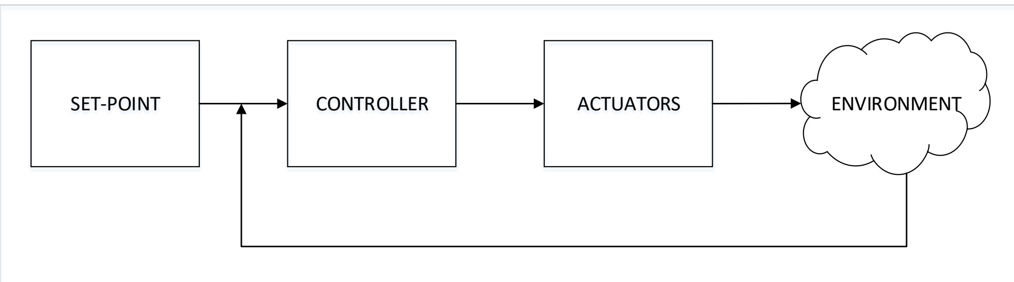


Figure 2.1: Traditional set-point-based control loop.

According to Mirinejad et al. [42], it is possible to classify an indoor environment control system based on the mode of its controller. Three types of controllers can be identified: *Traditional Controllers*, *Advanced (or Adaptive) Controllers* and *Intelligent Controllers*.

2.2.1 Traditional Control

Traditional control is represented by those systems which the only aim to minimize the power consumption through the control of temperature, leaving the assessment of thermal comfort to the users. Standard On/Off switch systems, thermostats and the Proportional, Integral and Differential (PID) control are the most used controllers, all of them allowing a simple structure and a low initial cost, even though they do not provide enough accuracy and quality. On/Off control is the simplest one, providing only two kind of inputs and outputs: maximum or zero. The control is completely left to the user, who switches the HVAC system when he/she wants. They do not autonomously provide neither thermal comfort nor energy saving.

Thermostats are able to behave autonomously on a prior user programming, and to control the environment just on the temperature changes. The PID controller, despite its long time, still stands thanks to the simplicity of its implementation. Several types of PID have been developed for a HVAC systems [24, 38]. However, these kind of controllers do not take in consideration the users thermal comfort, which still have to provide the controller the exact personal ideal value.

2.2.2 Advanced Control

Advanced control is able to lead to better thermal comfort and less energy consumption than the traditional one. Several research has been carried out since 1980s, leading to the development of predictive, adaptive and optimal controllers. Predictive control is able to create a model for future disturbance [18], while adaptive control includes Auto-tuning PID, where the parameters of the controller are tuned automatically based on a transfer-function model or on a set of heuristic rules (Wang et al. [62], Dexter et al. [16]), and Non-linear control, which in turn includes different solutions and methods (Arguello-Serrano and Vélez-Reyes [3], Semsar-Kazerooni et al. [52]). However, the above solutions turned out to be infeasible in industrial application, due to the requirement of a model of the building, and some other limitations, such as the cost of control elements, the lack in addressing the user thermal comfort and in the usability [53]. In addition, even if the control is more accurate and precise, most of the controllers proposed in this category do not still have any kind of autonomy in the control action.

2.2.3 Intelligent Control

Intelligent control is the most recent, started being researched in the 1990s. It includes several techniques in various fields of Artificial Intelligence, such as Model-based Predictive control techniques and Learning-based methods, that tries to model the thermal comfort of the users in an optimized way.

Model-based Predictive Control

Model-based Predictive Control (MPC) is a powerful simulation-based framework for designing a common and ad-hoc energy plant controller, and it has been successfully applied in building system controls for different plant controllers, such as cooling [41], ventilation[64] and floor heating [35], which simulations showed an effective energy savings. On the other hand, it often does not consider the thermal comfort in the controller design, relying on the thermal insulation efficiency of the building instead of the feeling of the users. In addition, the required steps (modeling, data collection, deployment) involve excessive cost if applied to medium size buildings.

Learning-based Control

Instead of being a paradigm for designing control systems, learning-based control represents more a feature in a several industrial thermostats. It consists in the application of machine learning techniques for allowing the control system to learn occupants preferences and activity patterns in order to automatize the control process. Learning approaches can be applied to both Traditional, Advanced and MPC control. Some examples of machine learning techniques are Neural Networks [39], [43] and Support Vector Machine [48], while an iconic example of industrial application of learning thermostat is Nest [47]. This approach can give benefits in the overall user experience, but only in terms of autonomy of the control action, which is still based on set-point.

2.3 Feedback-based control systems

The feedback-based control is a paradigm which aims to bring the human in the control loop, as Figure 2.2 shows. It falls within the concept of *participatory sensing*, where the people contribute in creating knowledge with their own sensory information. In this case, the people contribution is represented by their actual thermal comfort feedbacks, often represented with the Fanger notation, which is used for building a thermal model and for computing the actuation set-point. The reasons about how this approach can be an advantage for environmental control area have already been shown in Chapter 1. In this section we describe the most relevant works assessing the environmental control problem with this approach.

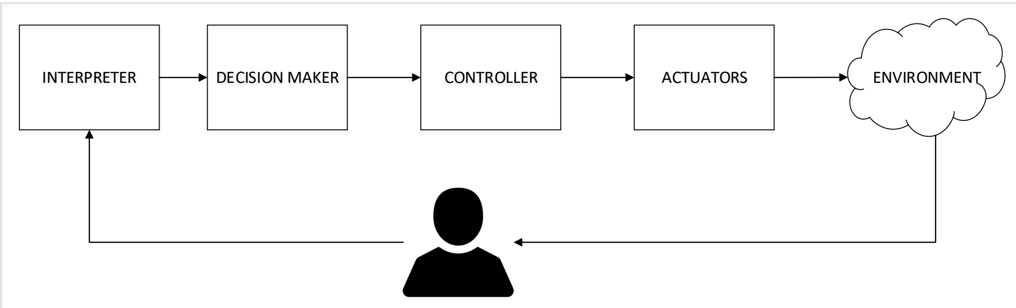


Figure 2.2: Feedback-based control loop.

The first participatory sensing solutions firstly appeared in the 90s. The two most relevant works are represented by [17] and [27], who both proposed to use actual thermal feedback from people for a comparison with computed PMV index. The actuation is then performed through a series of fuzzy rules, which helps in computing the PMV index.

Most recently, different feedback-based approach have been proposed. [21] proposed Thermovote, a system that adjusts estimated PMV index by comparing with the actual mean vote collected through smartphone application, by finding a temperature offset that equalizes the PMV with the occupants feedback. Despite Thermovote does not implement any energy savings strategy, authors show how improving thermal comfort can also potentially increase energy savings. Some limitations to this solution is the PMV computation, which is by definition computationally expensive, and the 7-scale PMV index, which can increase the possibility of bias in the overall vote.

[36] proposed an temperature-comfort correlation model, which establishes a correlation between the comfort feedback of the occupants, collected through a smartphone application, and the indoor and outdoor temperature. Along with the model, it also proposed a set-point optimization algorithm which compute the optimal temperature set-point for a group of occupants by iteratively computing a temperature set-point candidate satisfying as many occupants as possible. The main problem with this solution is that this algorithm at each iteration eliminates those occupants with the farthest temperature from the candidate one, reducing at each step the number of satisfied people. [49] proposed a model-free based only on the occupants feedbacks, which are simplified to be only *hot/cold* complaints. All the feedbacks from the occupants are collected and merged into a single vote, and the temperature is changed with a given step value according to the value of the overall vote. One big limitation to this solution is the estimation of the optimal step value in real-time, which depends both on the system characteristics (the rate of cooling and heating) and on the information provided by the users.

[25] proposed a knowledge-based approach that learns thermal comfort preferences (when the user wants the environment warmer or cooler) and builds a

zone level personalized profiles. The set-point is then calculated by solving an optimization problem that takes into consideration also the energy consumption. However, this solution only calculates a daily set-point, without considering preferences fluctuations over the same day. A similar solution is also proposed in [26], where occupants feedbacks, preferences and environmental information are processed by a learning algorithm that computes the optimal set-point. Also this solution does not implement any energy efficiency strategies inside the optimization set-point algorithm but only adopts an energy efficiency approach when the environment is unoccupied.

In [61] and previous works ([66], [65] and [67]), the authors proposed a solution where the *hot/cold* complaints from the people are aggregated with the temperature and humidity values received from the sensors in the room. The complaints are then used to train a Multi-linear One-Class classifier, which computes the borders of an hot and cold complaint region, as shown in Figure 2.3. From these two regions, it is computed a comfort region where lies all the $< Temperature, RelativeHumidity >$ set-point samples that are potentially comfortable for all the occupants. Then, from this model, the actuation algorithm extracts an optimal comfort set-point through minimal change principle. The main problem with this solution is represented by the choice of designing a classifier that computes linear inequalities for representing borders that are non-linear. This approximation brings an excessive over-fitting of the regions boundaries: the borders appear too much tight and depending on lonely value placed in borderline part of the region. This issue can be solved through other classification techniques such as Support Vector Machines, which permits to get a non-linear representation of these boundaries, and thus a representation of the regions that are more precise and closer to reality.

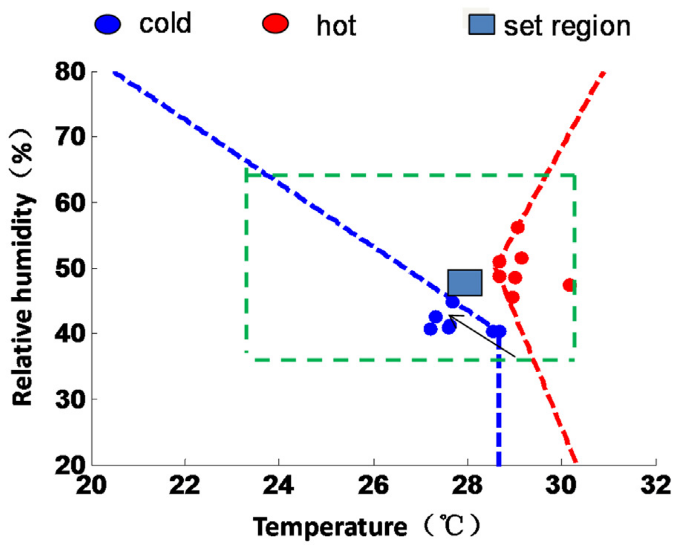


Figure 2.3: The thermal model proposed in [66], [65] and [67], composed by the computed complaint and comfort regions.

2.4 Other approaches

Throughout the years, there have been proposed more approaches that do not fall into the categories already presented. An interesting approach is represented by *personalized comfort*, which intends to provide to the individuals a micro-environment with personal parameters and preferences. This approach finds its application in offices and rooms with personal working desks, where it is possible to create a dedicated space for each occupant with ad-hoc devices. Representative works are [50], who proposed a system that monitors the users occupancy and the environment surrounding it and provides heating with a dedicated device close to it, while [40] proposed a wearable device that heats specific parts of the body that influent massively the overall thermal comfort.

2.5 Conclusion

As shown in this Chapter, many control systems have been proposed to assess the conflicting problem of providing thermal comfort to building occupants while minimizing energy consumption. However, all of them lacks of different features, from the absence of any energy consumption strategy to inefficient management of the occupants feedbacks. Table 2.1 summarizes all the main paradigms and approaches comparing them between each other with respect to the following key points, already presented at the beginning of the chapter:

- Explicit and autonomous thermal comfort assessment
- Explicit and autonomous energy savings policies
- Participatory sensing approach
- Complaint-based thermal model and control action
- Mobile technology and distributed architecture

Our contribution with this thesis is to develop a similar approach to the one presented by [61], but trying to overcome the limitations of this solution, specifically the excessive over-fitting of the classification. Our solution overcomes this issue by proposing a One-Class Support Vector Machine for classifying complaints from all the users, which permits to model in a more precise way the boundaries of the complaint regions, compared to the linear model. In addition, instead of interacting directly with the environment (thus interfacing itself

Table 2.1: Comparison between related works with respect to the key points.

Solution	Thermal comfort	Energy savings	Participatory sensing	Complaint-based	Mobile technology
Traditional Control	-	-	-	-	-
Advanced Control	-	x	-	-	-
Intelligent Control	x	x	-	-	x
Feedback-based control					
[17], [27]	x	-	x	-	-
[21]	x	-	x	-	x
[36]	x	x	x	-	x
[49]	x	x	x	-	x
[25], [26]	x	-	x	-	-
[61]	x	x	x	x	-
[50], [40]	x	-	-	-	-
ThermoSense	x	x	x	x	x

with the actuation subsystem), the control system is intended to interact with an already existing trigger-action based BMS [44].

CHAPTER 3

Methdology of the proposed approach

In this chapter, we are going to explain the methodology used for the ThermoSense approach, providing a brief insight of the theoretical foundations on which the methodology is built upon, and describing all the steps of the workflow for obtaining the solution of the problem described in the previous chapters.

3.1 Proposed Approach

In Chapter 1, we showed how the traditional set-point control is a source of technical issues related to thermal comfort and energy savings. First of all, occupants do not know their ideal set-point, and this brings often to an useless over-cooling or over-heating of the environment. Secondly, having several occupants with different preferences may arise conflicts between desired set-points. Lastly, it is not so rare that these environments do not permit to change their set-points, leaving people unsatisfied with their thermal comfort. For all of these reasons, a new paradigm of control is necessary. Instead of relying on precise set-point provided by occupants and set on thermostat, key point of the ThermoSense approach is to rely only on the user complaints regarding their actual thermal feedback, that is, if they are feeling *hot* or *cold*. As an output, we provide a set-point that makes the user feeling comfortable while trying to spend as less energy as possible. The overall algorithm can be structured into the following steps:

Collects the complaints for each user.

Periodically:

    Build thermal model with the collected complaints.

    Extract the energy optimal set-point.

    Provide the set-point to the actuation

Then the algorithm starts again from step 1. The above workflow is intended to be a loop, starting when an occupant sends a complaint after an actuation is performed, as Figure 3.1 shows.

The approach is very similar to the one presented by Wang et al. in [61] and their previous works. As already said in Chapter 2, they proposed a similar algorithmic approach, but using Multi-Linear One-Class classifier for building the thermal model, which suffers of over-fitting problems caused by the excessive of approximation of the linearization. In our approach, we use instead Support Vector Machines, which seems to be better fitted for the modeling of the thermal regions, as we are going to shown in the rest of this chapter.

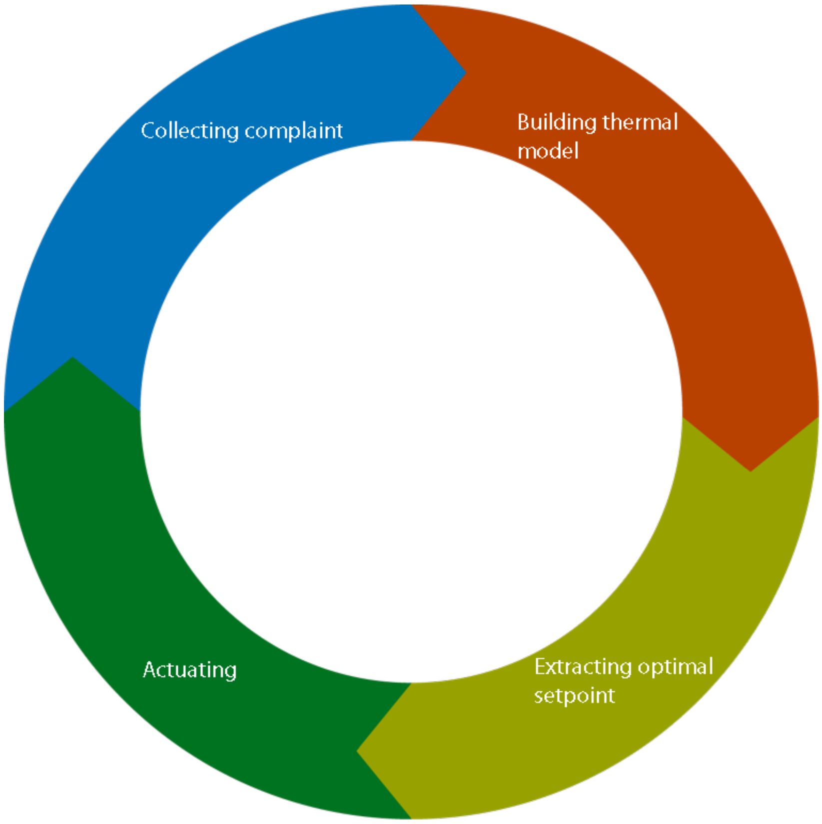


Figure 3.1: The ThermoSense approach workflow.

3.2 Data collection and aggregation

The first step is to collect user thermal feedbacks. We ask occupants to provide their thermal feedback through thermal complaints when feeling a discomfort, such as feeling hot or cold. We provide a smartphone application for doing so (more details are provided in Chapter 4). Every complaint is aggregated to environmental measurements obtained through sensors placed in the surrounding environment, in order to relate the user discomfort to the actual environmental setting. Using a mathematical notation, we can model a single complaint as a sample $(\mathbf{x}, y)$ where $\mathbf{x}$ is a vector $(x_1, x_2, ..., x_n)$ of environmental measurement and $y = c$, where $c \in [-1, 1]$ (or, using a more natural notation, $c \in [Cold, Hot]$). In our case, the environmental measurements are the Air Indoor Temperature and the Relative Humidity. This choice has been made since these variables are easy to measure with simple and not expensive sensors, and they are generally used also as output variable for control action. Hence, a complaint sample in ThermoSense is composed as $(< T, RH >, y)$ where y is given by the complaining occupant. Modeling the complaints through the aggregation with the Temperature and Relative Humidity permits to visualize them in a Cartesian <T,RH> space, as Figure 3.2 shows.

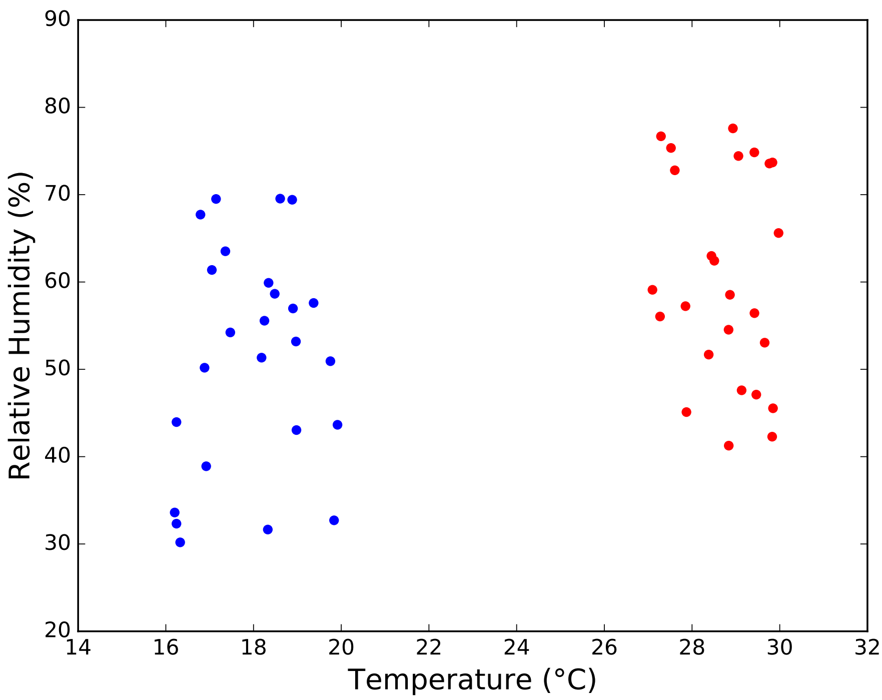


Figure 3.2: Complaint feedbacks visualized in the Temperature-Humidity space.

It is easy to notice how the complaints tend to concentrate in two distinct regions. In the left-bottom area, where are concentrated the blue-dotted points, it is more probable to find samples corresponding to cold complaints, whereas in the right-top it is more probable to find samples corresponding to hot complaints, red-dotted. This is an important key point: if we express in a mathematical notation these two regions, then we will be able to find a complementary region containing all the potential comfort samples. This task is covered by the process of building the thermal model.

3.3 Building thermal model

In this step the focus is to obtain a division of the input space such that it is possible to separate the complaint samples from the comfort samples. In order to do so, the goal is to find a boundary that divides the two regions, for then using this boundary for obtaining the region of all the possible comfort samples from which extracting the one with optimal energy consumption.
One simple solution is to model the boundary as a linear equation, as proposed in [61], but since the boundaries of the regions are not linear, this would be an approximation (which is already shown that brings over-fitting in the model).
Since we have the data from which to build the regions (the complaint samples), we can apply some classification techniques from Machine Learning. In this case, the classes represent the regions. After built the model by training the classifier with an initial dataset of complaint samples, we can use it for both processing new samples, and decide if a model update is required, and for identifying the comfort region.

3.3.1 One-Class Support Vector Machines: preliminary knowledge

For solving this classification problem, we decided to use Support Vector Machines (SVM). SVM has been proposed firstly by Vapnik [59], and the idea behind it is to solve an optimization problem for finding a hyperplane that separates into two different classes the set of training samples, with the maximum margin. This hyperplane is then used for classifying new input samples. This machine learning technique helps in reducing the risk of over-fitting when training the model, unlike other similar works using different classification techniques such as Least-Square Estimation [66].

Since the ThermoSense approach relies only on a kind of feedback from the users (their comfort complaints), the stated problem can be seen as One-Class Classification (OCC) problem. OCC are problems when the goal is to construct models from samples of one class. Several machine learning techniques have been proposed for solving OCC problems, including One-Class Support Vector Machines (OCSVM), proposed by Schölkopf [51]. Following the general idea of the SVM, the goal is to classify the samples into an unique class, dividing them from the rest of the samples in the input space (which are seen as a unique negative class). More rigorously, assume a set of n training samples with all positive labels in an input space X: $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n \in X$. If Φ is the function mapping input samples to the feature space F, then we can define the hyperplane separating the positive samples from the origin as $f(\mathbf{x}) = (\mathbf{w} \cdot \Phi(\mathbf{x})) - \rho = 0$ where $\mathbf{w}$ and ρ are the normal vector and the offset of the hyperplane. In a mathematical form:

$$\begin{aligned} &\min_{\mathbf{w}\in F,\xi\in\mathbb{R}^2,\rho\in\mathbb{R}} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{1}{\nu n}\sum_i \xi_i - \rho \\ &s.t.\, (\mathbf{w}\cdot\Phi(\mathbf{x}_i)) \geqslant \rho - \xi_i \\ &\xi_i \geqslant 0,\, i = 1, ..., n \end{aligned} \tag{3.1}$$

where ξ_i are the slack variables and ν is a parameter such that $\nu \in (0, 1]$.

Note that the decision function only relies on the dot-product function of the vector in the feature space F. If the feature space is high-dimensional, performing the dot-product becomes computationally heavy. In order to overcome this problem, a mathematical trick has been designed, by applying a transformation on the input data, called *kernel function*, that is used instead of the dot-product. A kernel function is defined as $K(\mathbf{x}_i, \mathbf{x}_j) = \Phi\mathbf{x}_i \cdot \Phi\mathbf{x}_j$, and allows to represent the input samples in the feature space F [54]. Different kind of kernel function has been designed, such that linear, polynomial, sigmoidal and moreover the Gaussian Radial Basis Function:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\gamma}\right) \tag{3.2}$$

where γ is the Gaussian width parameter.

For this project, we are going to use an OCSVM with a Gaussian kernel.

3.3.2 Complaint and comfort regions

In our case, finding the boundary separating the complaint and the comfort samples corresponds to finding the hyperplane separating the samples of the positive class (the complaint samples) from all the other samples (the potential comfort samples). The two kind of samples are thus contained in two different regions: a *Complaint Region* and a *Comfort Region*, or using a mathematical notation: $C_{complaint}$ and $C_{comfort}$, such that $C_{complaint}, C_{comfort} \subseteq < T, RH >$, and $C_{complaint} \cap C_{comfort} = \varnothing$.

Figure 3.3 shows a graphical representation of how the OCSVM divide the input T-RH space in the two regions. $C_{complaint}$ is the yellow-colored space, while $C_{comfort}$ is the blue one. The dots are the complaint samples (where, for clearness, we colored the "hot" complaints as red dots and the "cold" complaints are blue dots). The hyperplane is represented by the border of the two regions. The shape and the size of these regions depend on the SVM parameters, in particular ν and γ.

3.3.3 Training the model

The model is trained at the beginning of the computation. It uses a set of pre-collected complaint samples in order to build the initial model. Another solution would be to train the model online step by step whenever a complaint arrives. This approach involves inevitably a training period where the thermostat would not work correctly, providing uncomfortable environment. For avoiding this situation, the first approach of relying on historical data is more appropriate.

Algorithm 1 shows the training procedure. The functionalities of Support Vector Machines are provided by a specific framework (more info in Chapter 4). The model works with normalized data, so in order to function, it is necessary to perform a normalization of the training data and all the complaints that arrive afterwards. The mean and the standard deviation of the training data are stored in order to use them in the normalization of the new complaints. The training process

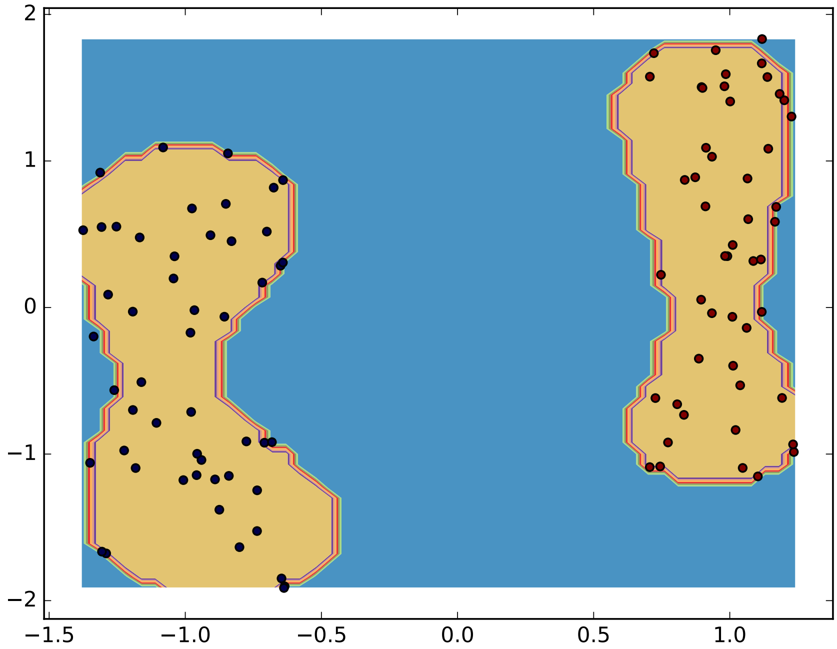


Figure 3.3: The complaint regions computed by the OCSVM, in the normalized T-RH space.

is automatically performed by the function `fit()`, while the model is instantiated with the function `createOneClassSVM`, which takes as input the parameters of the SVM, that are ν and γ and the kernel type.

Algorithm 1 Training of the model

Input: ν, γ, training data

normalized_data $\leftarrow$ normalize(training data)

kernel $\leftarrow$ $gaussian$

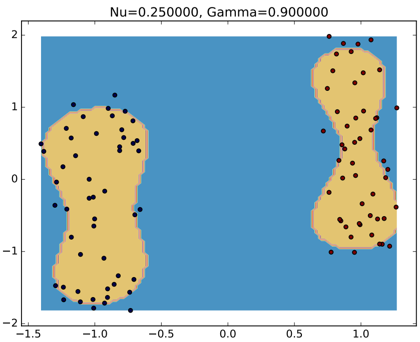
model $\leftarrow$ create OCSVM with ν, γ, kernel

fit OCSVM with normalize_data

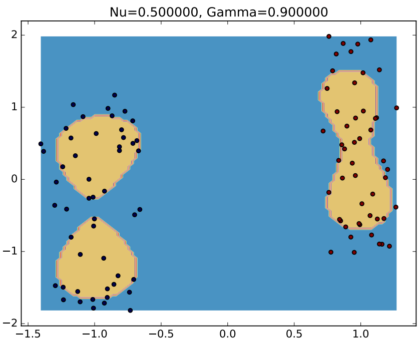
Since in OCSVM it is not possible to perform cross-validation in order to estimate the best parameters, he choice of the ν and γ parameters has been done empirically performing some preliminary tests of the model training and seeing which set of parameters give the best regions in terms of shape, dimensions and coverage of the sample. Figure 3.5 and Figure 3.4 show how ν and γ respectively affect the shape and dimension of the SVM. We can say that ν influences the dimensions of the region, narrowing them while its value increase, whereas γ influences the fitting of the values, generalizing the classification while its value remains low. For our purpose, we took $\nu = 0.12$ and $\gamma = 1$, which represents values that give broad and well-fitted regions.

3.3.4 Updating the model

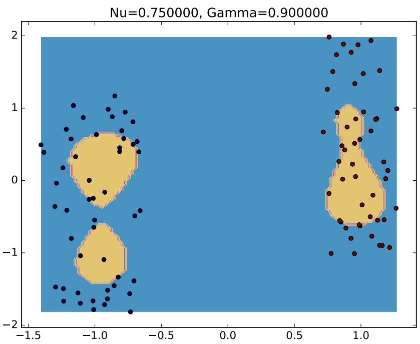
As already mentioned, the model is trained at the beginning with a set of pre-collected samples. When a new complaint arrived, the model is used for classifying the new complaint. Note that we are not interested in analyzing the precision of our model in predicting the target of each complaint (as a standard classification problem), but instead we classify the sample in order to check if it is significant, that it means if it changes the shape of the complaint region. This situation can occur in two cases: whether the sample is a support vector, that is, a point lying near the boundary (with a distance from the margin between 0 and 1), which contributes in its mathematical definition, or the sample is misclassified by the model (if the distance from the margin is less than 0). In both the situations, the model will be re-trained including the new sample in the training dataset. If this situation does not happen, then the model is used for the actuation phase. The



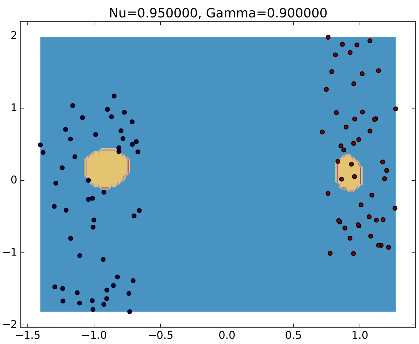
(a) $\nu = 0.25$



(b) $\nu = 0.5$



(c) $\nu = 0.75$

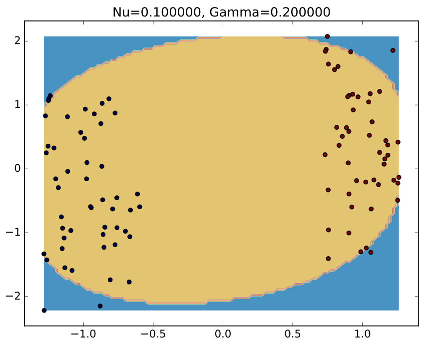


(d) $\nu = 0.95$

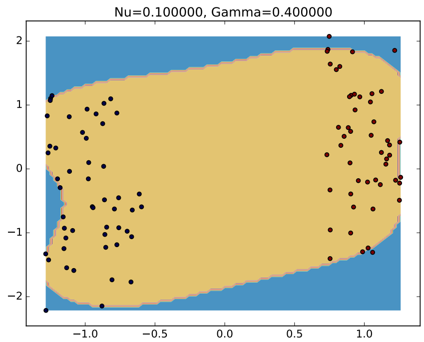
Figure 3.4: Effect of the ν parameter on the SVM shape.

pseudo-code of the updating algorithm is shown in Algorithm 2.

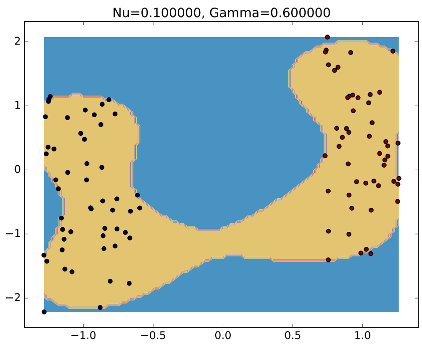
Theoretically, the model should be updated every time a sample arrives and is classified as significant. But since in real environments there would be several people sending more complaints at the same time, it is reasonable to think in a 'batch' way, launching the update algorithm every 15 minutes. Since the typical working day is of 8 hours (from 9AM to 5PM), there will be around 30 actuation



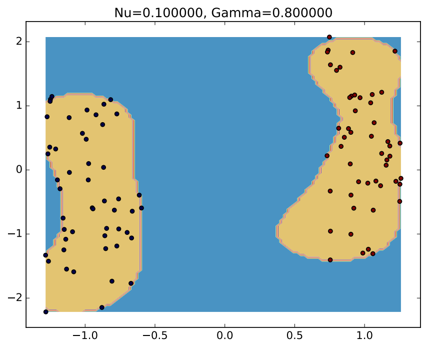
(a) $\gamma = 0.2$



(b) $\gamma = 0.4$



(c) $\gamma = 0.6$



(d) $\gamma = 0.8$

Figure 3.5: Effect of the γ parameter on the SVM shape.

each day. 15 minutes is a reasonable amount of time between each actuation and the next, since it considers the time for the occupants to adapt to the new environment and provides small variations between the old set-point and the new one. Every time the update algorithm is launched, it collects all the complaints arrived in the last 15 minutes since the previous actuation and checks each of them in order to classify them as significant or not.

Algorithm 2 Update the model with the new complaints

Input: arrived complaints

for all complaint **in** arrived complaints **do**
  if complaint **is** classified as complaint **then**
    if complaint **is** significant **then**
      retrain_flag $\leftarrow$ true
      add comp to the model
    end if
  end if
end for
if retrain_flag = true **then**
  retrain the model
end if

3.4 Extracting the optimal actuation set-point

Once we obtained the thermal model and we used it for classifying a new complaint sample (and eventually updating the model), the actuation algorithm is performed. The underlying idea is the following: referring to Figure 3.3, if the yellow region holds all the $< T, RH >$ values that are certainly complaints, then the blue region contain all the possible $< T, RH >$ values that potentially may be comfortable for the users. The mechanism for obtaining all the comfort samples from the model is the following: we create a grid of all the possible $< T, RH >$ values that the HVAC can span with its control action (usually a range of [10, 30]°C for the temperature and a [0,100]% for the relative humidity), and we provide it to the model. Each sample of the grid is classified by the SVM, which returns the classification output: 1 if the $< T, RH >$ input is classified as a complaint sample, -1 if a comfort sample. In this way, the complaint and comfort regions are provided as a simple set of $< T, RH >$ values with different classification labels. Figure 3.6 shows the output of the SVM when provided of the grid, where the red-dotted points are the samples of the complaint region while the green-dotted are the samples of the comfort region. The comfortable values are then extracted, stored in a data structure and provided to the actuation algorithm. Note that this is a brute force method, but it is affordable since the number of samples in the input grid is a low number (around 100 samples), mainly because the thermostats have a low sensibility.

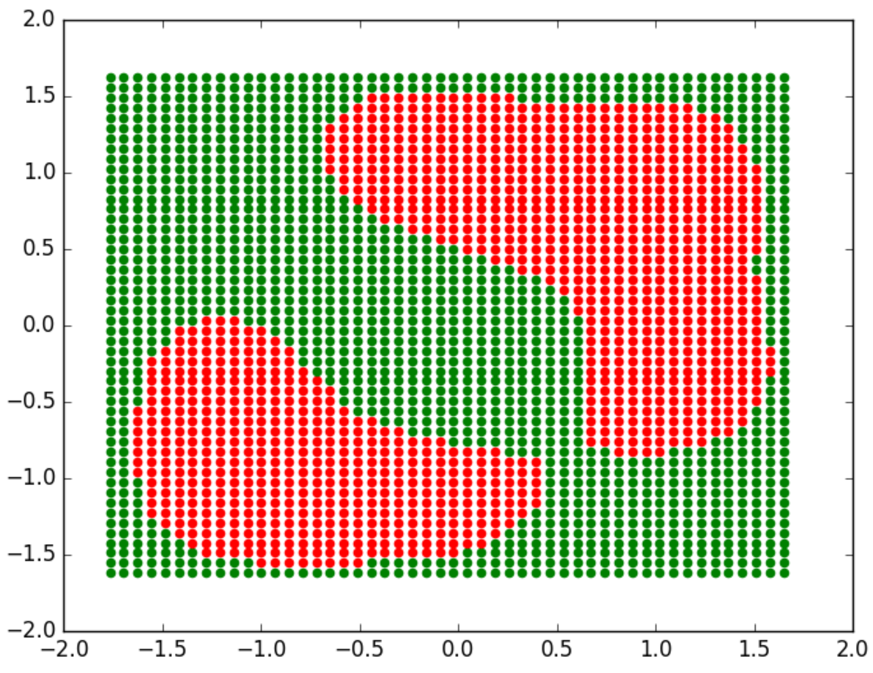


Figure 3.6: Results in 2d

The actuation algorithm has the goal to compute a set-point value to provide to the actuation system. The simplest way to do it is to extract a sample belonging to $C_{comfort}$. Unfortunately, in this way only the goal of providing thermal comfort would be assessed. In order to assess also the minimization of the energy consumption, it is necessary to design a heuristic that is able to include the energy consumption in the set-point choice. For our case, we decided to consider the *principle of least action*, also called *minimum change principle*. In the context of physics, it consists in finding a status, reachable by performing some action on a mechanical or thermodynamical system, that has the least change from the actual status. In terms of environmental actuation, the least action value represents a specific combination of $< T, RH >$ which is the closest to the actual values of the environmental $< T, RH >$. Since the input $< T, RH >$ space is a Cartesian space, the terms "closest" refers to the euclidean distance between the two samples. Therefore, we want to solve the following minimization problem:

$$(T, RH) = \min_{<T,RH>} \sqrt{(T - T_{actual})^2 + (R - RH_{actual})^2} \; |(T, RH) \in C_{comfort} \tag{3.3}$$

Algorithm 3 shows the pseudo-code for the actuation algorithm. The decision of whether to take a colder or a warmer set-point depends on the majority of the votes received at the moment the algorithm is launched: if the majority of the complaints are cold, then the actuation will choose a warmer set-point, instead if the majority of the complaints are hot, then the algorithm will compute a colder set-point. In case of parity, the choice of the set-point will be made considering the smallest energy consumption. This control is performed at the beginning of the actuation algorithm, before the eventual update of the model, setting a `hotFlag` to `True` or `False` whether a cooling or a heating is required, respectively.

Also in this case, we use a brute force method in order to find the closest optimal set-point, by computing the distance between the actual set-point and each comfortable value. Until the number of samples of the grid are under a grade of magnitude of hundreds of samples, the brute force method is reasonable since the computation is not affected. In case the dimension should increase, other better techniques must be used, such as for example performing an optimized local search starting from the actual point until a comfortable point is reached.
Note that despite we are going to extract a tuple $< T, RH >$, the real actuation will depend on the type of HVAC systems, in particular on which kind of actuators are implemented on the HVAC: usually the temperature is the only action variable that can be controlled, whereas the humidity can be controlled explicitly if the HVAC has also a humidistat. If the control system provides to the occupants only the thermostat, then from the actuation extraction will be considered only the temperature set-point. This consideration opens to some possible future features, for which the reader is referred to Chapter 6.

Algorithm 3 Compute the optimal actuation set-point

Input: comfort region, actual set-point

Output: closest comfortable set-point

```
min_dist ← 100
if hot_flag = true then
    for all sample in comfort region do
        if T_sample < T_actual and RH_sample < RH_actual then
            d ← euclidean distance between sample and actual
            if d < min_dist then
                closest ← sample
                min_dist ← d
            end if
        end if
    end for
else
    for all sample in comfort region do
        if T_sample > T_actual and RH_sample > RH_actual then
            d ← euclidean distance between sample and actual
            if d < min_dist then
                closest ← sample
                min_dist ← d
            end if
        end if
    end for
end if
return closest
```

CHAPTER 4

Implementation

The previous chapter described the methodology of the proposed approach. In this chapter the actual implementation of ThermoSense will be described, showing each single component and how it interacts with all the other ones.

4.1 ThermoSense system architecture

ThermoSense is a distributed system composed by four fundamentals elements: a **mobile front-end**, used for collecting data from the occupants, a **sensor wireless network**, used for obtaining data from the environment, a **back-end server** for managing the data from the nodes and for launching the ThermoSense Core actuation engine, and for last a **Building Management System** (BMS) interfacing directly with the actuation system. Figure 4.1 shows the structure of the system: the data from the front-end and the sensors are aggregated and send to the server through a REST API interface. The back-end pre-processes and stores the received data, while every 15 minutes launches the actuation algorithm (presented in the previous chapter). The optimal set-point is then sent to the BMS which manages automatically the actuation over the environment.

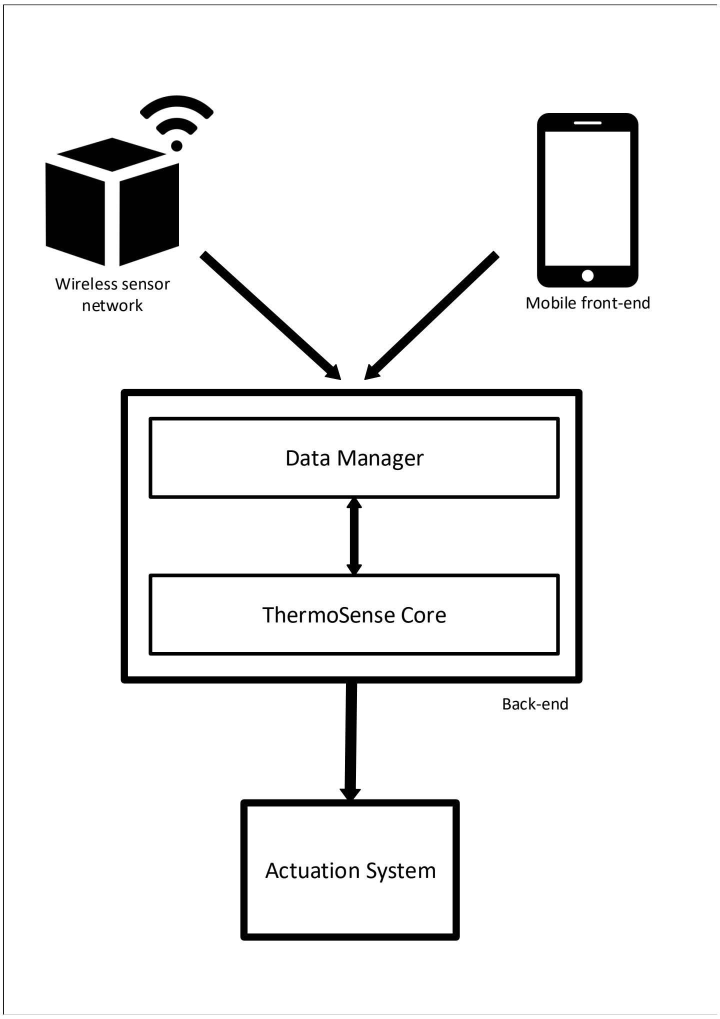


Figure 4.1: Overall architecture of the ThermoSense system.

The rest of this chapter is dedicated to the detailed presentation of each component.

4.2 Front-end: a smartphone-based user interface

The front-end represents the interface with which the users (the occupants of the building) interact with ThermoSense. It is composed by a mobile application from where users can submit their thermal complaints every time they are feel uncomfortable. The mobile application has been developed for Android™[1] and iOS™[2] platforms. Figure 4.2 shows some screenshots as an example. It provides the users of a gradient panel where they can tap on dependently on how they are feel uncomfortable. The red parts stands for 'I am feeling hot' while the blue part stands for 'I am feeling cold'. This kind of interface is based on [60], a project which relates thermal comfort sensation to colors, which in the literature is called *hue-heat hypothesis* [6], and on this choice there will be based future investigations on including lighting as a variable of the model (see Chapter 6 for more informations).

The application limits the frequency of the complaints to 1 every 15 minutes, in order to avoid the compulsive submission of complaints from the users, from one side for giving enough time to the system to actuate over the environment and let the people adapt themselves to the new settings, and from the other side to prevent system security malfunctioning (e.g., DoS attacks). The check on the time period is done server-side, returning an exception to the application, which shows a small alert to the users, in case the time period is <15.

The application also features the possibility of specifying the indoor position of

[1] Download URL: `https://play.google.com/store/apps/details?id=polimi.necst.box`

[2] Download URL: `https://itunes.apple.com/us/app/necstbox/id1125348130?mt=8`

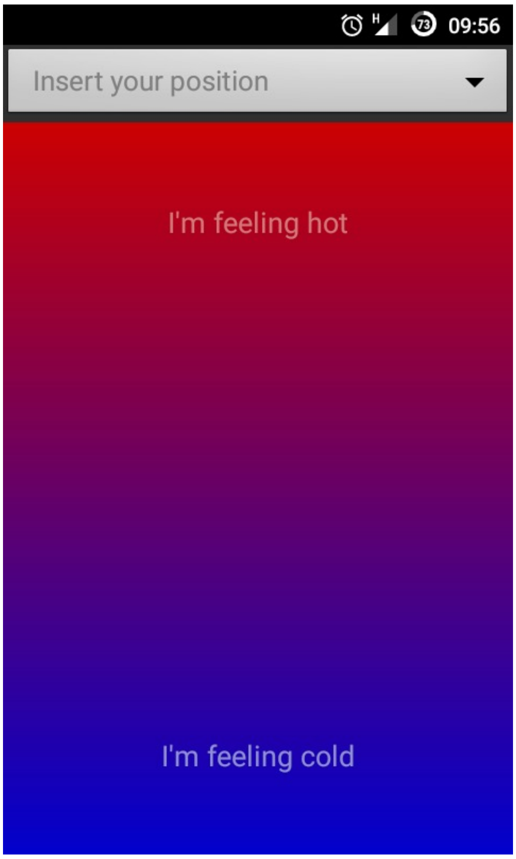


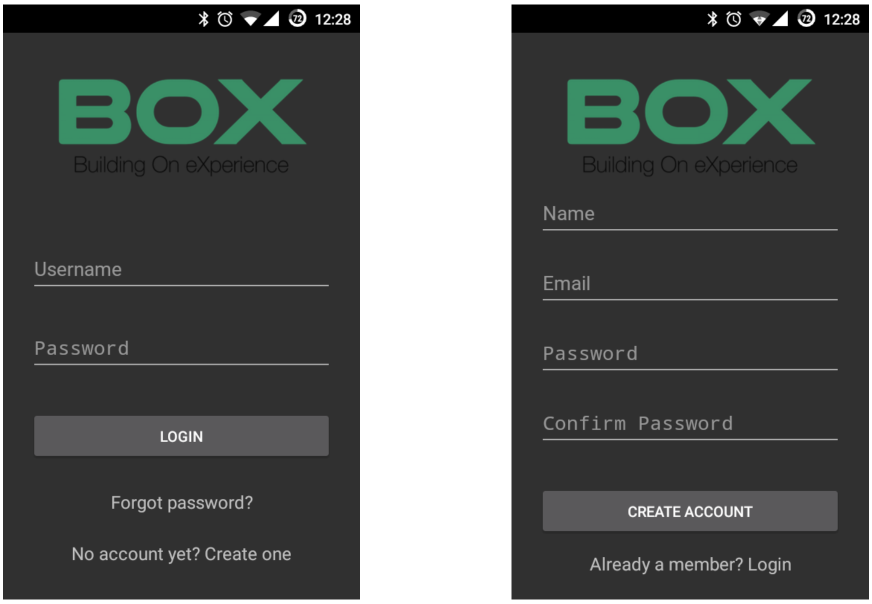
Figure 4.2: Screenshot from mobile app showing the gradient panel for submitting complaints.

the user in the building. This feature is strongly dependent on the deployment on the building, and in particular to the room and thermal zones it is composed of. This feature consists in a drop-down menu where are listed all the rooms and zones where the user is allowed to stay, and from which the user can pick up his actual position. This position is used to perform the actuation to that specific room or zone. This information is mandatory, if the user does not select any of the position from the menu, the application will show an alert asking the user to do it. The indoor position is an important factor not only regarding the actuation, but it is also a very important parameter on which the indoor temperature depends on: for example, the indoor temperature varies considerably with the distance to the windows. A future feature will consist in including the indoor position in the thermal model presented in the previous chapter, with the goal to adjust the intensity of the actuation with respect to this variable. Another future feature will be the implementation of an automatic indoor localization mechanism using the Bluetooth Low Energy mechanism [12], which will enable a more precise position in real-time.

Finally, the application also provides a sign up and login mechanism, which are required in order to use the application, as shown in Figure 4.3. The sign up data from the users can be then used for restricting their usage to limited rooms or zones of the building through the design of authorization policies, every time the system will be deployed to a real building. The reason of that is to avoid that a person not authorized to stay in a room can give feedbacks related to that room.

4.3 Sensor Network for ambient data measurements

In order to gather environmental information, in particular Temperature and Relative Humidity values, there has been designed an ad-hoc wireless sensor network (WSN) composed by nodes placed around the environment. Each node is composed by a Raspberry Pi 3 with a Grove TH02 Temperature and Relative Humidity sensor. Figure 4.4a shows an image of the Grove sensor. The sensor mounted on the chip is a TH02, which is able to register temperature and humidity values from a range of $0 \sim 70° \ C$ and $0 \sim 80\%$ respectively, with an accuracy of of $\pm 0.5° \ C$ and $\pm 4.5\%$ respectively. The sensor is provided of an I2C port

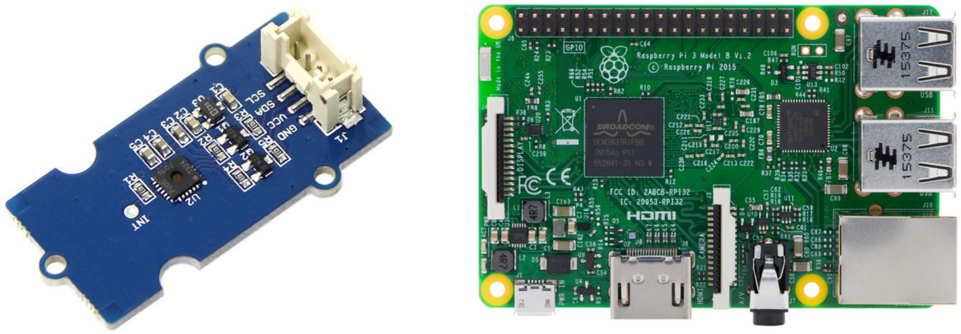


(a) Login screenshot (b) Sign up screenshot

Figure 4.3: Screenshot from the mobile app of login and sign up features.

which is used for communication.

Figure 4.4b shows an image of the Raspberry Pi 3. This device has been choice mainly for the easiness in prototyping the node. The sensor can be mounted through the serial port (in the top part of the board in the figure) connecting the corresponding pins, or through an apposite I2C shield. Raspberry mounts a Debian-based linux image, which runs a Python script that each 1 minute queries the sensor for retrieving temperature and humidity values. These values are then packed and sent to a back-end server, where they are pre-processed and stored. Another features of this device is the Bluetooth Low Energy bundle already installed on the chip, which can be used for exploiting indoor localization analysis. Raspberry Pi 3 is also a good choice in terms of future features: more sensors can be attached to the chip in order to increase the input variables.



(a) The Grove TH02 sensor

(b) Raspberry Pi 3 board

Figure 4.4: Components of the sensor node for measuring environmental data.

In complex areas, where the rooms are big or divided into zones or sub-rooms, the problem of placing the nodes in a way that all the considered area is covered is crucial. For solving this problem we rely on a floor plan automatic mechanism specifically designed for indoor location [8].

4.4 ThermoSense back-end: data storage and core logic

The back-end server has two main goals: receiving the data from the mobile app and the wireless sensor network, performing some pre-processing in order to aggregate and formatting them, and launching the algorithm for building the thermal model and computing the actuation value. The back-end is divided into two main entities with the two different features: the server for the data storage and manager, and the server for the ThermoSense Core. The architecture is built using Backend-as-a-Service (BaaS) technologies for simplifying the deployment process.

Figure 4.5 shows the Sequence Diagram of the interactions between the front-end and the back-end architecture. The aggregation of the data from the occupants and from the environment is done by the Data Manager server, just before the storage. Every 15 minutes it triggers the ThermoSense Core to perform the actuation algorithm, which gets the data from the Data Manager server for updating the model and computing the optimal set-point. The set-point is finally sent to the actuation system.

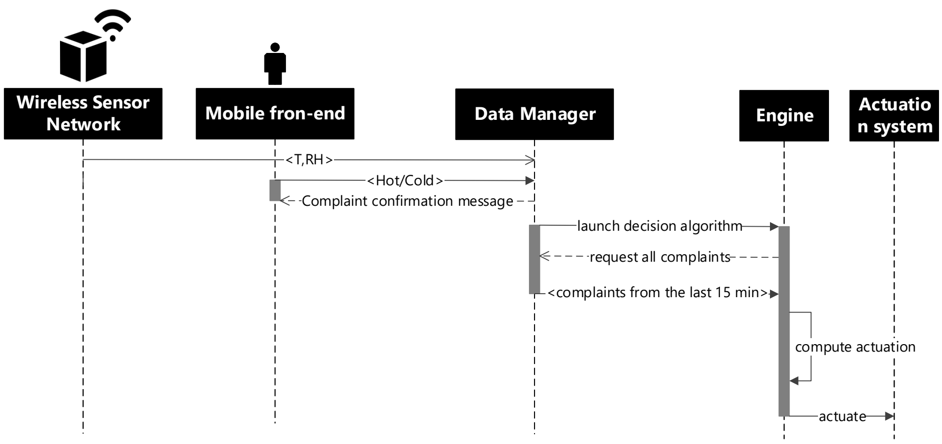


Figure 4.5: Interaction between the front-end and the back-end.

4.4.1 Cloud-based data storage and manager

As already said, the Data manager collects the data from the front-end. The server has been deployed locally using Parse [3], a Mobile Backend-as-a-Service technology which permits to easily build and deploy a backend infrastructure for mobile applications. It uses MongoDB as database, and JavaScript for running server code. It also provides API and SDKs for several mobile-oriented technologies and programming languages, such as Android, iOS and JavaScript (the first two used for developing the corresponding mobile applications presented in the previous section).

The database can be managed by an online dashboard, which shows each class as a spreadsheet. Figure 4.6 shows a screenshot of the dashboard. It is similar to other database manager, thus much easier to use.

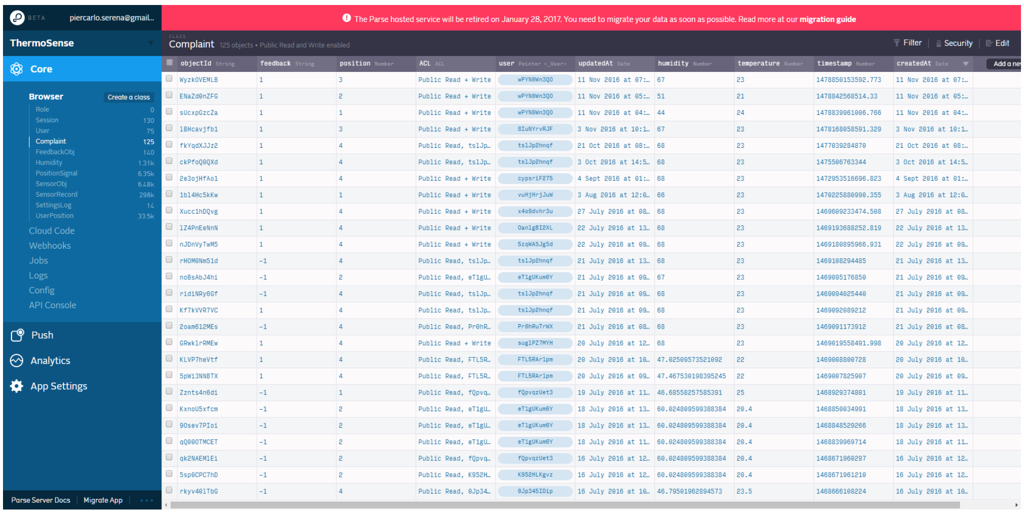


Figure 4.6: Screenshot of the Parse database dashboard.

[3] https://www.parse.com/

Database classes

The classes in the database are created using Parse API, as `objects`. Each object is created in the classic object-oriented programming paradigm. Parse automatically add some default fields during the creation process:

- `objectId`, which is an unique and alphanumeric string used as a primary key for the object;
- `createdAt`, a Date field set once at the creation of the object;
- `updatedAt`, a Date field which is set every time the object is modified;
- `ACL`, which specifies the permissions on accessing the objects, they are automatically managed by Parse.

User This class specifies the account information for each occupant. Using the mobile application a person can register to the system, and afterwards perform the login. Sign up and Login are performed automatically using the API provided by Parse. When the user is registering, he is asked to choose an `username`, a `password` and to insert his `email`. The email is needed in the case of a password recovery is requested. The password are encrypted and cannot be read by anyone, neither the database administrator, who can only update manually the field from the database dashboard.

SensorRecord This class is created by the nodes composing the Wireless Sensor Network described in the previous section. It consists in a log including all the records read from the sensors. Each object represents a record. Every 1 minute the sensor reads the temperature and humidity value and it creates an SensorRecord object and sends it to Parse. SensorRecord is composed by `temperature` and `humidity` fields set with the values from the sensors. It also has a `sensorID` field containing the MAC address of the node where the object is sent from (the MAC is unique and unambiguous, so it can be a good ID for the node). The preprocessing code on the server just before storing the data on the database adds a new field `position` which maps the MAC address to an more readable integer number. This mapping is helpful when a change of the node is necessary: instead of addressing every time the position of an occupant by the MAC address of the node, we abstract it by using the position value.

Complaint This class is created when an user submits a feedback with the mobile application. It consists in the numeric `feedback`, `position` fields and the `objectID` of the `User` who sent the complaint. The `feedback` field can only assume values of 1 and -1, while position field is a integer number corresponding to a zone inside the room or building. At the moment is directly asked to the user, but in the future it will be automatically set using automatic indoor localization technologies. At the moment the room and the building are not modeled as entities in the database, but in the future, when deployed in complex building, the room modeling will be important since it will permits to specify some important information such as which occupants are allowed to stay in and which kind of actuation can be performed in each single room. The preprocessing code running on the server adds to this object three more fields: `temperature` and `humidity`, retrieved from the `SensorRecord` from the node placed in the zone specified by the `position` field, and the `objectID` of the `SensorRecord` itself.

Figure 4.7 shows the diagram of the main classes composing the database. We outline the relationships between the main classes: an User can both submit more than one complaint or submitting none of them, while a Complaint is always from one User, and comes from a single zone, thus linked with an unique SensorRecord. Parse API also provides a standard class, called `Role`, which permits to specify groups of `User` (by their `objectID`) for setting group permissions. This class has not been used yet, but it would be in case ThermoSense would be deployed in complex organizations building with several occupants and several roles with different policies.
Another default class is `Session`, which permits to an User to stay logged in the application, avoiding the login phase each time they use the mobile app. It is linked to a single User and it characterized by an expiration date after which the session object is erased. The session is also erased when the user logs out from the application. If the user accesses the app from multiple devices, a new session is created for each device.

Server cloud code for storage pre-processing

Since the front-end has been developed for different mobile platforms, the data coming from the users can be different depending on the differences between the platforms. In addition, the sensor network sends even different data. In order to properly aggregate homogeneous data, a pre-process is needed. Parse provide a server Cloud Code, a feature enabling the possibility to run JavaScript from server, in order to provide logic not running on mobile devices. Cloud code permits to define functions that can be triggered in different moments, such as before and after saving a specific object. It permits also to define Jobs that run in back-

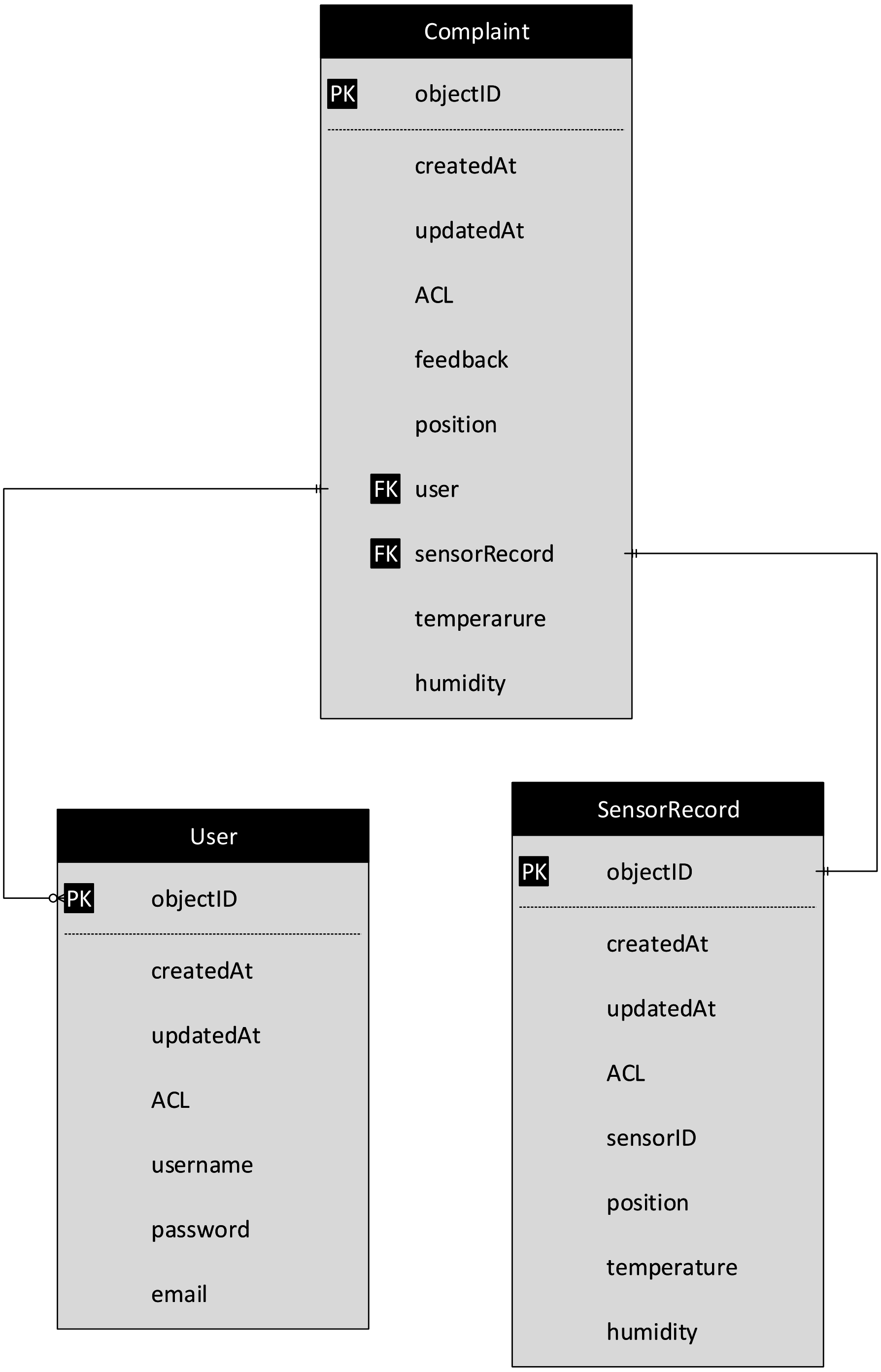


Figure 4.7: Diagram of the database with main classes composing ThermoSense.

ground without stopping the remaining of the execution waiting for a response. In ThermoSense, this feature is used for enabling pre- and post- processing to the data arriving from the front-end, in order to aggregate heterogeneous data and format them properly.

The trigger used in ThermoSense cloud code is `beforeSave` for the Complaint and SensorRecord object. In the first case, the goals are mainly to check if the user has already sent a complaint in the last 15 minutes, by performing a query on the database searching for the last complaint from the same user, and to encapsulate the temperature and humidity values into the Complaint object, by performing a query on the SensorRecord class searching for the last log from the same sensor node of the complaining user. In the second case instead, the trigger is activated just for mapping the MAC address of the sensor node to an integer number.

4.4.2 ThermoSense Core

The ThermoSense Core is responsible for running the algorithm related to the building and the updating of the thermal model and to the algorithm related to optimal set-point extraction. The ThermoSense Core has been entirely developed using Python 2.7 language, with several external libraries. This choice is mainly due to the powerfulness of the Python language, its easiness in fast developing prototypes and the high number of libraries and external modules that can extend considerably its functionalities.

Since Parse, the BaaS used for the Data manager deployment, only supports JavaScript for the Cloud code functionalities, ThermoSense Core has been deployed on a different server, with a different technology. The BaaS chosen is Tsuru [4]

[4] `http://tsuru.io/`

which provides a full-stack architecture for deploying and running web applications. Tsuru supports several languages environments, frameworks and database, resulting to be versatile, flexible and very powerful in scaling and high availability.
In the following paragraph we are going to show the stack of Python libraries and external modules used in the ThermoSense Core, with their functionalities. Later we are going to provide a detailed explanation of each inner module.

Python developing stack The functionalities provided by ThermoSense are enabled by the following Python library stack:

- Matplotlib: a fundamental library for plotting graphs and diagrams in 2D and 3D;
- NumPy: a basic package that enables advanced mathematical and scientific computation, in particular provides N-dimensional array object and linear algebra capabilities;
- SciPy: the main package for the scientific computation, built upon NumPy. It provides efficient numerical and optimization routines;
- Scikit-learn: a tool framework for data mining and data analysis, built on the previous 3 packages. It provides the functionalities of Support Vector Machine which are used for building the model;
- Pandas: a library that enables advanced and high-performance data structures.

The ThermoSense Core is basically structured into two parts: a **model** module and a **logic** module. The model defines the homonym class, and implements the routines for creating, training and updating the SVM thermal model. Instead, the logic implements all the algorithms for managing the actuation algorithm. In the following paragraph we are going to explain in details the functions designed in each of the two module.

Model module As already said, the model is implemented as a Python class. Basically, this class provides all the functionalities for *creating* and *training* a Support Vector Machine. When initializing a new instance of the model class, it is necessary to provide to the constructor the input data, which are an initial set of complaints characterized by two features, *temperature* and *humidity*, and an output label, which can be 1 (hot) or -1 (cold). Alternatively, the SVM can be trained on-line complaint after complaint, but this would need an apposite training period. Anyway, this approach is beyond the scope of this thesis, but of course it would be interesting analyzing the differences between the off-line and the online training. In addition, the model provides two function that, when provided a new complaint, they return the classification output and the distance between the decision boundary computed by the SVM. These two functions are used in the Model updating algorithm (Algorithm **??**), when the complaint is checked whether it is significant for the model (and thus the model needs to be updated including that complaint in the dataset) or not. All the low-level functions for the SVM functionalities are provided by the Scikit-learn library.

Logic module Logic is a standalone module implementing all the algorithms for the ThermoSense work-flow. First of all one of the main functions is the decision algorithm, launched every 15 minutes, which decides if warming up or cooling down through majority voting, and then checks if each complaint arrived is significant (see Chapter 3 for more information), and in case it launches the update of the model (which is a new model training with the dataset, added of the significant new complaints). Finally, it launches the actuation algorithm, which has been already explained in Algorithm **??**. Briefly, the actuation firstly computes all the comfortable points, by creating a 50x50 grid of Temperature and Humidity values, in a range defined by the minimum and maximum values registered by the already collected inputs. Each point of the grid is then provided to the model, which returns its classification output inside a matrix. We then extract from this matrix the values equals to -1, which corresponds to the comfortable ones, and save them in another matrix.

The second phase is to return the optimal one, by choosing the closest to the actual $< Temperature, RelativeHumidity >$ value. We firstly set as temporary minimum distance a symbolic value of 100. For each value of the matrix provided by the previous step, we compute the euclidean distance between that candidate value and the actual environmental one. If the actual distance is less than the temporary one, than the temporary one is updated with the candidate one, and so on until all the comfortable points are checked, for finally returning the point with the minimum distance. Since the computed points are not so many, performing this kind of brute force method is sufficient. We proposed two version of the actuation algorithm: one considering both the temperature and relative humidity as control variables (thus the actuation value will return the optimal set-point formed by the two environmental values), and a second one that considers only the temperature as control variable, meaning that euclidean distance is computed only for the temperature of the actual set-point and the candidate one. This choice has been made for flexibility: most thermostats permit only to control temperature in a direct way, while few of them can directly control the humidity (which is controlled indirectly with the ventilation, that in turn is difficult to measure). Of course, if the control system allows also a humidistat, then the first version can be used.

Simulator module for experimental evaluation

Finally, we briefly explain the simulator module, which will be extensively covered by the next chapter. In order to evaluate ThermoSense from an energy consumption point of view, we are going to test it with an energetic simulator. Since it was impossible to obtain a test room, we decided to create synthetically the complaints and the relative actuation. In order to do so, we simulate the behavior of 10 occupants inside a working room, for 30 working days. We conduct two simulations, one in winter condition (low initial temperatures) and summer (high initial temperatures). Each occupant is implemented as a *thread* (we use the `threading` Python module). Each thread is created and ran from the main thread. Each thread sends a complaint after a random time interval between 30 and 90 minutes. The nature of the complaint (hot/cold) is determined by the PMV model, which returns a thermal discomfort index as a function of the actual temperature (see Chapter 1 for more information). So the generation of the complaints is related to the indoor temperature trending. Each complaint is added to a dedicated collection specified in the model.

In addition, we create another thread from where we launch the decision algorithm every 15 minutes. We consider a working day as starting at 9:00 AM and ending at 17:00 PM, so there will potentially be 35 actuation during a day. Just after the decision algorithm returns its actuation value, the list of the added complaints in the model (which is updated by the occupants every time they send a complaint) is initialized again, so at the next actuation only the complaints arrived in the last 15 minutes are considered. Every actuation value is saved into a log file, along with a timestamp.

The main thread runs the 10 occupant threads and the decision algorithm thread every day. The decision algorithm thread stops when it reaches 35 actuations, while the occupant threads keep submitting complaints until the decision algorithm is running. When all the occupant threads are stopped (and thus the decision algorithm thread too), the current day ends and after a brief time sleep a new day begins.

In total there will 12 threads running: 1 main thread, 10 occupant threads and 1 decision algorithm thread. An important point is that the simulation time is decreased by 90 times with respect to real time, in order to shorten the time of execution. This means that instead of 15 minutes (900 seconds) the decision algorithm is launched every 10 seconds, and instead of waiting for a random period of time between 30 and 90 minutes, each occupant thread will wait from 20 to 50 seconds before submitting a new complaint.

4.5 Conclusion

In this section we have presented the ThermoSense actual implementation. We show the overall system architecture in which ThermoSense is organized, for then focusing of the single module presenting the technology used and the design choice made. Since ThermoSense is a middle-layer between the occupants and the actuation system, it should be presented also how ThermoSense interact with actuators and plants placed in the indoor environments, but, as the next chapter will explain in details, we were not able to have a test-room where to deploy the system in order to evaluate it. However, the deployment in real environment with an integration with actuation systems is in our future work plan and it will be addressed soon.

CHAPTER 5

Experimental Results

In this chapter, we are going to present the evaluation work-flow, composed by a simulation of the ThermoSense functioning compared with the actual state-of-the-art of complaint-based approach for comfort control [61]. The simulation is run for both the baseline model and ThermoSense, and evaluation on control performance and power consumption has been carried out.

5.1 Introduction

As every environmental control system, the evaluation should be done on a real environment, involving people as occupants who provides actual feedbacks. The most important characteristic for an environment to be suited for this kind of evaluation is to be featured with a proper actuation system, which means that anyone should be able to access a thermostat or similar devices that influence that specific environment, thus avoiding HVAC managed by a central system. In addition, it should be possible to access these control devices from a computer, and in case providing programming features in order to automatize and schedule the control actions. If it is possible to have this environment, then it is easy to install some sensor to gather environmental measurements and perform experiments with real data that evaluate the system functioning in likely situations.

Unfortunately, for the evaluation of this thesis it has not been possible to obtain such an environment. For overcoming this limitation, we performed the evaluation through ThermoSense functioning with a simulation which mimics the situation of a shared office with 10 occupants, in two different seasons (thus with different outdoor conditions): July (summer), January (winter).

5.2 Evaluation and Metrics

The evaluation of the simulation relied on the comparison between the model considered in ThermoSense and the current state-of-the-art model, the multilinear one-class approach presented in [61] and already presented in Chapter 2. For the evaluation of their solution, they performed two real-case tests in two testbeds located in different places with very different outdoor environments. The two tests consists in placing a set of occupants inside the testbed room (equipped with sensor and actuator devices) and to give them the possibility to control the environment through the baseline set-point approach and their complaint-based approach. They evaluated the comparison of the two tests on the following metrics: *control performance*, *user acceptance*, *user work performance* and *energy consumption*. Since user acceptance and work performance are collected through questionnaires and ad-hoc tests directly with the users, and since our constraint to perform only simulation-based tests, we considered as metrics only the **control performance** and the **energy consumption**.

Control Performance it is based on the trending of the Indoor Air Temperature, the Relative Humidity, and the People discomfort with respect to the time. The goal is to obtain a environment control which does not fluctuate and that remains under specific threshold of control. The discomfort trending is represented by the number of complaints recorded throughout the day: the goal is to design a model which learns the complaint environmental values in order to decrement the number of complaints after a learning phase, and thus to provide the best comfortable environment as fast as possible.

Energy Consumption it is based on the electricity consumed by the thermal environment control devices. The plant is composed by an Air-Conditioner (AC), a Fresh Air Handling Unit (FAHU) and a humidifier. The authors proposed an equation in order to estimate the consumption of the AC, while the FAHU and the humidifier consumptions rely on direct measurements on the field. For these reasons, we decided to compare our solution to the baseline only on the AC power consumption. The equation represents a fitted model from the AC specification data, and returns the electric power P_{AC} given the indoor and outdoor temperatures. The AC model is shown in Equation (5.1), where the RE is the ratio between the actual electricity power and the rated power of the AC $P_{AC,rated}$ (whose values are shown in Table 5.1, that gives the AC specification data with respect to the indoor and outdoor temperature), T_{in} and T_{out} are the indoor and outdoor temperature, respectively.

Table 5.1: Air Conditioner power consumption specification data (kW).

Outdoor temperature (°C) / Indoor temperature (°C)	20	25	30	32	35
20	0.76	0.83	0.91	0.94	0.98
22	0.76	0.84	0.91	0.94	0.98
25	0.77	0.84	0.91	0.94	0.99
27	0.77	0.84	0.92	0.95	0.99
30	0.78	0.85	0.92	0.95	1.00
32	0.78	0.85	0.93	0.96	1.00

$$RE = (0.0012 * T_{in} + 0.4870) * (0.0289 * T_{out} + 0.9216) \tag{5.1a}$$

$$P_{AC} = P_{AC,rated} * RE \tag{5.1b}$$

Since the given specification data do not cover all the possible outdoor-indoor values, we performed a curve fitting to the given data in order to find an fitting equation, which turned out to be a straight line, shown in Equation (5.2).

$$P_{AC,rated}(T_{in}, T_{out}) = -0.0001313 * T_{in} + 0.01472 * T_{out} + 0.4687 \tag{5.2}$$

5.3 Test-bed

We already gave a brief explanation of the simulation in Chapter 4, focusing on the functioning of the simulator module. We give now more details regarding the simulation design choice.

We tried to refer all the simulations in a test-bed as close as possible as the ones performed in [61]. This means that we relied on their environmental condition. They performed the simulation in two different environments, represented by a shared office located in two different locations in China: Guangzhou and Lanzhou, in South and Northwest China respectively. The first location is characterized by a hot humid summer and warm winter, while the second location is characterized by hot and dry summer and cold winter. Thus, these two locations represent two interesting testbeds for summer simulation (Guangzhou) and winter simulation (Lanzhou).

Unfortunately, we could not replicate the exact conditions of the baseline, since we do not have access to the dataset used in [61]. In order to overcome this limitation, we made some assumptions. First of all, regarding the outdoor temperature and relative humidity, instead of taking precise hourly-based values we considered it constant and set to the monthly average values, which are 6°C and 45% for winter condition, and 32°C and 70% for summer condition.

A second consideration is represented by the indoor environment: instead of maintaining indefinitely the last actuation set-point, we assume that if there are not complaints within a hour, the plant enters in a 'saving energy' mode and power-off itself. The indoor temperature and relative humidity start then to evolve towards the outdoor conditions, with a decay effect of the %0.01 of the difference between the indoor and outdoor temperatures every hour. This fluctuation inserts some variability into the simulation. Note that we did not performed a detailed analysis of the relationship between the indoor and outdoor conditions, nor we considered the physics of the building in order to choose the exact parameter for the decay function: we are only interested in simulating a dynamic indoor environment in order to see how the model reacts against the variation of the indoor environment.

Another simplification we made is related to the Mean Radiant Temperature, which depends highly on the outdoor trending and the physics of the building and, along with the indoor temperature, is one the most influent factors affecting thermal comfort. Since this variable cannot be controlled and its measurements is complex, we decided to perform the simulation considering it constant, with two different values for summer and winter, reflecting the impact of the wall insulation against the outdoor condition (especially against the heat from sunlight). The last assumption we made is related to how we provide complaints for the simulation. Since we do not rely on real complaints from people, we generate them synthetically. The generation is based on the PMV/PPD model [22], which has been shown in Chapter 1. At first, we check if a specific user is complaining at a specific time, by computing the PPD value with the temperature and the relative humidity of the current set-point. All others parameters are considered constant (see next section for their values). In other words, we allow only the percentage of users expressed by the PPD to complain. If that specific user is then complaining, we calculate its PMV with the same parameters. Since the PMV is an index that have a range of [-3, +3], and we rely on 'Hot/Cold' complaints, we collapse the PMV into +1 for positive values of PMV and -1 for the negative values.

Table 5.2: Parameters of the PMV/PPD model for complaint generation.

Air Velocity	m/s	0.1	Standard value
Mean Radiant Temperature	°C	22/25	Winter/Summer values
Clothing insulation	clo	0.5/1	Typical summer/winter clothing
Activity level	met	1	Typing

Table 5.3: List of the performed tests.

Test	Season	Month	Days	Occupants	Outdoor T (°C)	MRT (°C)
Test 1	Winter	January	30	10	6	22
Test 2	Summer	July	30	10	32	25

5.4 Tests and results

The tests consists in the simulation of the functioning of ThermoSense in two different conditions: summer and winter. We performed the simulation both for the linear baseline and the ThermoSense model. We considered a low and high value of the MRT, that are 22°C and 25°C respectively. The other parameters are shown in Table 5.2, and referred to the typical office conditions: clothing according to the season, low air ventilation and office activity.

Table 5.3 shows a list of the performed test with the respective parameters. The Outdoor T refers to the monthly average of outdoor dry-bulb temperature, while the MRT refers to the Mean Radiant Temperature and, as already said, as a simplification it is considered constant throughout the month and the single day.

In what follows the results for each test with respect to the given metric are shown. Firstly we show the results regarding the control performance and later we summarize the power consumption ones.

5.4.1 Control performance results

In the following paragraph the results of the two tests are shown, focusing on the trending of indoor temperature, relative humidity, number of complaints and power consumption, and for each test are given the diagrams of each simulation (Linear baseline and ThermoSense).

Test 1: Winter season, 10 occupants The trending of the temperature and the relative humidity are shown in Figure 5.1 and 5.2 respectively. It is interesting to notice that in the case of temperature, the trending is almost the same (with a slightly increase of the preferred temperature for the ThermoSense model), whereas the relative humidity is more controlled in the Linear model while in the ThermoSense model we can clearly identify how the humidity at the beginning tends to vary a lot for stabilizing later, due to the training process. Table **??** shows the statistics regarding the number of complaints recorded.

Table 5.4: Statistics of the arrived complaints for both the models in Test 1.

Model	Mean	Standard Deviation
Linear	21.6	4.92
ThermoSense	26.7	9.621
Gain	-23.42%	

Comparing this table with Figure 5.3, we can say that also the trend of the number of complaints throughout the month are quite similar: the model requires a starting period of time where performing the training process. Still, ThermoSense seems to need a bigger training dataset in order to decrease the number of complaints to the ones of the baseline (the big negative gain is due to the big number of complaints arrived at the beginning of the simulation for ThermoSense model), but with time the trend are practically the same. From this point of view, we can say that the ThermoSense has similar performance to the Baseline.

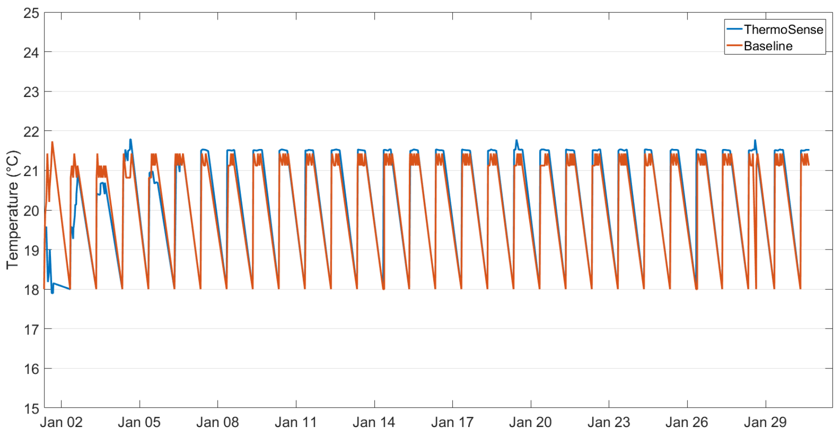


Figure 5.1: Test 1 - Air temperature trending.

Test 2: Summer season, 10 occupants The air temperature and the relative humidity trending are shown in Figure 5.4 and 5.5 respectively. We can see that the ThermoSense trending (blue line) gives a better performance in terms of control and values: in Figure 5.4 we can see how temperature values reach lower values (around 25.2°C), giving thus a better comfortable environment compared to the

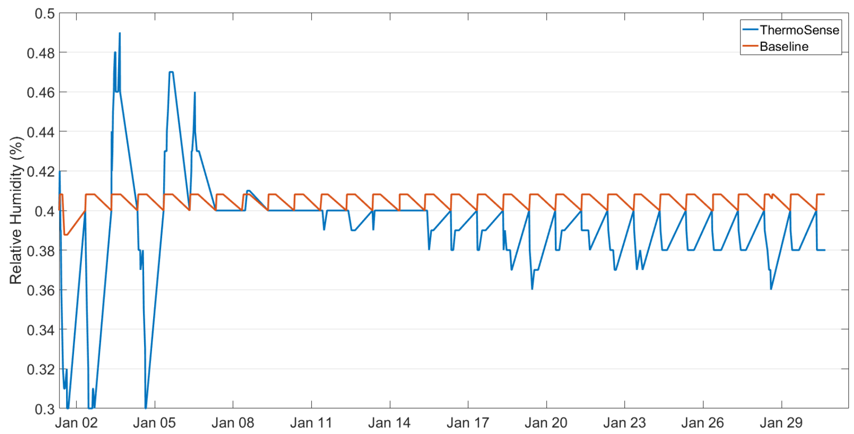


Figure 5.2: Test 1 - Relative humidity trending.

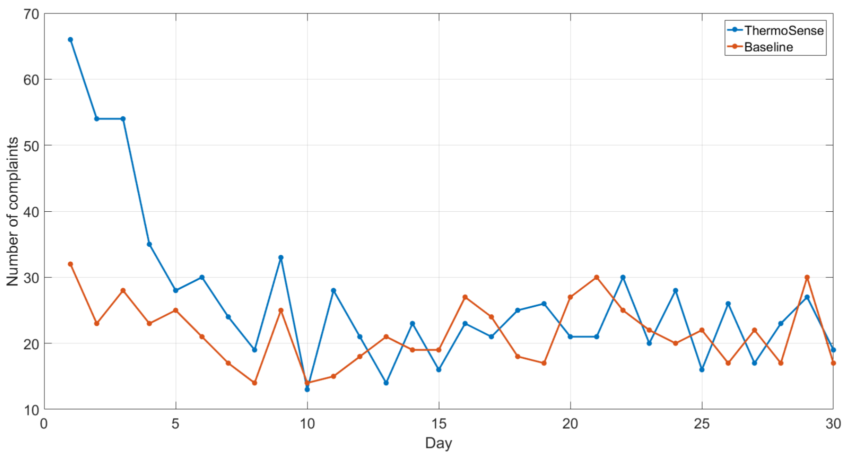


Figure 5.3: Test 1 - Complaint trending.

Linear (orange), which in turn provides more variable set-point values. Regarding the relative humidity, we can see from Figure 5.5 that it gives better controlled values within the range of 65-70%, while the Linear provides an higher humidity range (around 68-72%). The trends of the complaints are shown in Figure 5.6. From the figure we can see how the ThermoSense model (orange) tends to decrease and stabilize around the 10-12 complaints per day (that it means, around 1 complaint per day for each occupant), in contrast with the baseline which slowly decrease as well, but without a real stabilization. The statistics for the two models are shown in Figure **??**

Table 5.5: Statistics of the arrived complaints for both the models in Test 2.

Model	Mean	Standard Deviation
Linear	11.5	4.58
ThermoSense	12	4.28
Gain	-4.3%	

In this case, the Linear model has a better performance in terms of the overall number of complaints per day throughout the whole month compared to ThermoSense model, which on its side compensates with a better convergence.

5.4.2 Power consumption results

For evaluating the power efficiency of the ThermoSense model, we rely on the daily average values of the electric power consumption obtained by the two sim-

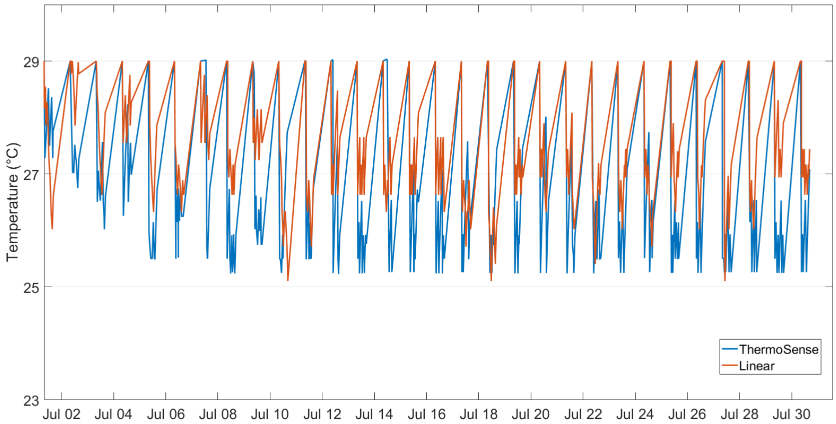


Figure 5.4: Test 2 - Air temperature trending.

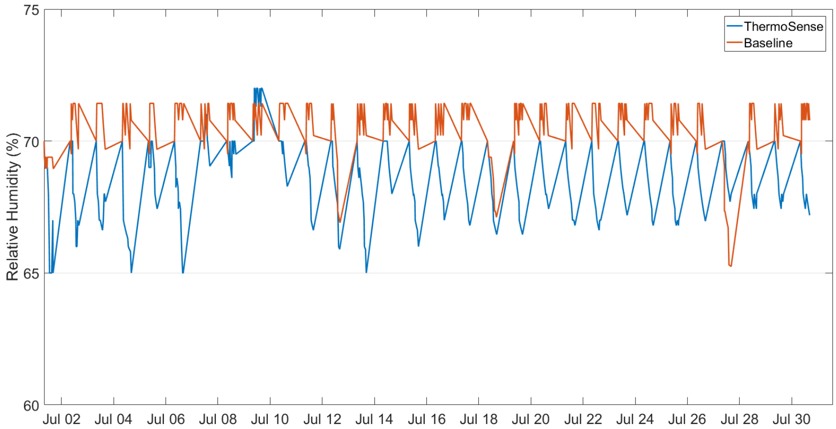


Figure 5.5: Test 2 - Relative humidity trending.

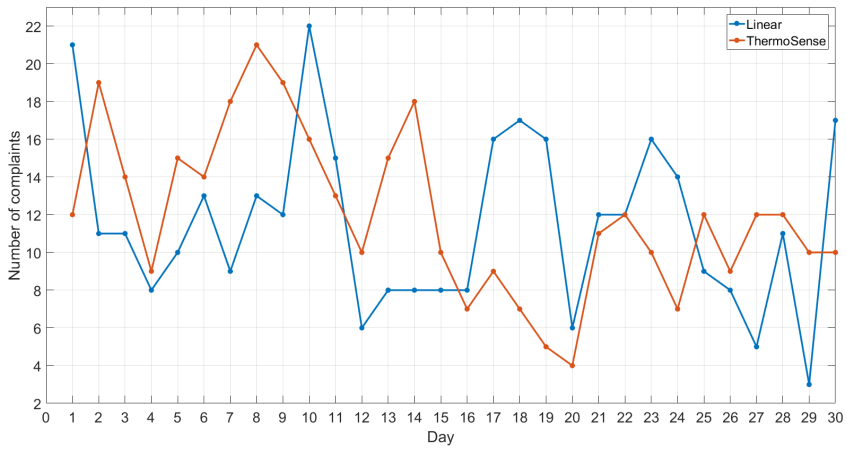


Figure 5.6: Test 2 - Complaint trending.

ulations ran with ThermoSense and the Linear models, for each season. The comparison is done them by obtaining the gain of ThermoSense model with respect to the Linear baseline, with the following formula;

$$G = \frac{E_{baseline} - E_{ThermoSense}}{E_{baseline}} * 100 \tag{5.3}$$

The electric power consumption is measured in kWh. We performed the daily average of the electric power consumption throughout the whole month. The results for the power consumption are shown in Table 5.6. As already said, we estimated the electric power of the AC system using Equation (5.1). For obtaining the daily average, we compute the power consumption for each actuation (thus, not considering the temperature alterations caused by the decay effect), and then we obtain the daily overall power consumption by summarizing all the single actuation power consumption for that day. We then compute the monthly mean of the power consumption. From the results, we can observe how during the summer season, ThermoSense model gives an improvement in power consumption

Table 5.6: Results of the power consumption estimation for both the models.

Model	Daily average power consumption (kWH)	
	Test 1 (winter condition)	Test 2 (summer condition)
$E_{baseline}$	1.98	5.46
$E_{ThermoSense}$	1.66	4.22
G (%)	16%	22%

of 20%. Under winter conditions we have similar results, gaining a 16% of reduced power consumption. This value, compared with the control performance presented above, let us state that even though the Baseline and the ThermoSense model are quite similar in terms of control action, ThermoSense can provide a reducing in the electric power consumption, and thus a saving of energy.

5.5 Conclusion

As a conclusion, we can state that ThermoSense turned out to be a good approach in comfort-based control environment, providing at least the same performance of the Baseline linear model and at the same time reducing the power consumption. On the other hand, the performed tests suffer of some big simplifications that should be addressed with a wider experimental campaign. In addition, despite the promising results obtained, the only way to really evaluate this kind of systems is to test it in a real environment, with real humans and real devices.

CHAPTER 6

Conclusions

We have presented ThermoSense, a complaint-based approach for the thermal comfort control in indoor environments. The problem of comfort control is a recognized issue while considering indoor environment characterized by many occupants, such as offices, co-working spaces, libraries. In these locations the multi-objective problem of providing a thermally comfortable environment while not consuming an excessive amount of energy arises, especially if the environmental controller is the typical set-point based. The causes are mainly the differences between the thermal preferences of the people and the lack of knowledge of the ideal set-point balancing comfortable environment and least energy consumption.

We try to overcome this issue by changing completely the control paradigm: instead of asking people which kind of environment they want (e.g., which temperature to set), we let them express complaints about their thermal comfort status, for then providing an actuation set-point which represents the best possible trade-off between comfort and energy consumption. In the last two decades many feedback-based solutions have been proposed, but they use the feedback from the users more as a supporting knowledge for improving estimation and computation of thermal comfort. Conversely, we do not want to estimate the user thermal comfort, but we directly asked it to them, since we think that no estimation can be as good as considering the real perceived thermal comfort of the users. In addition, our model relies only on complaints, with no other information but the environmental measurements. The proposed model is basically a One-Class Support Vector Machine (OCSVM) which is built upon complaint samples. A sample is composed by environmental features, e.g, the Air Temperature, the Relative Humidity and corresponding complaint. The goal for the OCSVM is to classify each complaint in order to group them into a spatial region of the feature space called *complaint region*. From this region we then compute the *comfort region* which is the portion of the T-RH space that contains all the possible comfortable points. We then extract the actual set-point from this region using minimal change principle, by considering as energy consumption the distance between the actual set-point and the new one. Among all the related works, the only solution which really proposed a similar approach used an ad-hoc Multi-linear one-class classifier, which however is not a flexible solution since it approximates the non-linear boundaries of the complaint regions with linear ones. Instead, the choice of using SVM can help in their modeling, especially with a particular Kernel function that allows to have non-linear region boundaries quite easily.

For the deployment of the system, we also proposed a smartphone application which can be used by the occupants in order to submit their feedbacks to a cloud server, which collects all the complaints and periodically (every 15 minutes) launches a procedure that updates the model with the new complaints and provide the new actuation set-point.

We evaluated our approach by simulation: unfortunately we have not been able to obtain a real test environment where to deploy our system. The simulation consists in the replication of a typical office scenario, simulating the behavior of a set of occupants during their working hours for an entire month in two different seasons (summer and winter) with different environmental and personal parameters. The synthesis of the complaints is based on the PMV/PPD model proposed by Fanger. We tested our solution compared to the Multi-linear baseline classifier, running the simulation with the same parameters. We considered as metrics the control performance (temperature, humidity and complaint trending throughout the month) and the Air-conditioner power consumption estimation. We obtained that the ThermoSense model can provide similar performance regarding the control, but with an increase in the power efficiency of 20% for summer condition and 16% for winter condition.

6.1 Limitations and future ideas

Despite the encouraging results obtained through simulation, ThermoSense model suffers of some limitations that should be assessed in the future in order to have a system that can be deployable in real ambients.

- First of all, as already said in the previous chapter, even though the simulation can be performed accurately to be very likely, nothing can be compared to the application on real environments, especially regarding the involvements of real people (we generate the complaints with a very accurate estimation model, that remains a mathematical model despite its accuracy).

- The simulation is performed with some simplification: for example, we considered a constant outdoor environment, the mean radiant temperature, the personal parameters such as clothing and activity level. In addition, we did not consider the transient stage in which the people enter the room and neither the physics of the building. All the variables should be considered through appropriate simulation, possibly before a test of ThermoSense in real environments.

- Regarding the model, another simplification has been made: the complaints are considered by the SVM as a unique class, without differentiating between hot and cold complaints. This consideration can arise potential malfunctioning regarding the decision action (whether to cool or or to warm) in unusual conditions (e.g., when an occupant feels cold at 26 ° C). A possible solution to this issue can be to use two different OCSVM for each complaint region, autonomous from each other, that can build up themselves specifically on hot and complaints.

- The model suffers sometimes of malfunctioning when computing the actuation set-point: sometimes it takes too much iteration for reaching a comfortable set-point, sometimes it extracts values that are considered good for the model but not enough good for the users. A better training and tuning must be performed on the model.

- The parameters of the SVM have been chosen empirically, in a way that the complaint region can actually cover almost all, if not, the samples, but maintaining separated the hot and cold complaint regions, nor to be too fitted with data. This approach has been chosen mainly because the cross-validation cannot be applied to OCSVM. It would be interesting to implement some other mechanisms for choosing the best parameters for the OCSVM.

In addition to the presented limitations, we now propose some ideas for future features, that can be implemented in order to make ThermoSense more accurate and more general-purpose.

- We plan to integrate ThermoSense with already existing BMS and actuation systems, in particular BuildinRules [44], for the deployment in real environments.
- Considering only Temperature and Humidity as input variables is quite limiting: comfort depends on more factors. A future features can be to expand the model by taking as features variables also other environmental values (e.g., air quality, ventilation) and personal ones (such as clothing and activity).
- Actuating only on Temperature is a choice that we made since most of the environmental controllers allow only to control this value, but it is a limiting feature: a future extension is represented by the increase of the set of control action, for instance with ventilation, lighting and air quality.
- A final long-term idea is to expand ThermoSense in order to not consider only *thermal* comfort, but to generalize the concept of comfort introducing other kinds, such as acoustic, visual, mental, physical. The idea behind this point is to move from thermal comfort towards the general wellness that

occupants may experience inside an indoor environment, still having the same approach and the same model behind.

Bibliography

[1] Buildings energy data book. Technical report, US Department of Energy, August 2012. URL `http://buildingsdatabook.eren.doe.gov/`.

[2] Yuvraj Agarwal, Rajesh Gupta, Daisuke Komaki, and Thomas Weng. Buildingdepot: an extensible and distributed architecture for building data storage, access and sharing. In *Proceedings of the Fourth ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, pages 64–71. ACM, 2012.

[3] Betzaida Arguello-Serrano and Miguel Vélez-Reyes. Nonlinear control of a heating, ventilating, and air conditioning system with thermal load estimation. *IEEE Transactions on Control Systems Technology*, 7(1):56–63, 1999.

[4] Bharathan Balaji, Hidetoshi Teraoka, Rajesh Gupta, and Yuvraj Agarwal. Zonepac: Zonal power estimation and control via hvac metering and occupant feedback. In *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, pages 1–8. ACM, 2013.

[5] Bharathan Balaji, Jason Koh, Nadir Weibel, and Yuvraj Agarwal. Genie: A

longitudinal study comparing physical and software-augmented thermostats in office buildings. *arXiv preprint arXiv:1601.07229*, 2016.

[6] Corwin A Bennett and Paule Rey. What's so hot about red? *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 14(2): 149–154, 1972.

[7] Gail Brager, Gwelen Paliaga, and Richard De Dear. Operable windows, personal control and occupant comfort. *Center for the Built Environment*, 2004.

[8] Andrea Cirigliano, Alessandro Antonio Nacci, Roberto Cordone, and Marco Domenico Santambrogio. Floor plan design and automatic nodes deployment for indoor location and monitoring systems. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, pages 45–48. ACM, 2016.

[9] European Commission. Communication from the commission to the european parliament and the council: Energy efficiency and its contribution to energy security and the 2030 framework for climate and energy policy. 2014.

[10] Giorgio Conte, Massimo De Marchi, Alessandro Antonio Nacci, Vincenzo Rana, and Donatella Sciuto. Bluesentinel: a first approach using ibeacon for an energy efficient occupancy detection system. In *BuildSys@ SenSys*, pages 11–19, 2014.

[11] Diane Cook and Sajal Das. *Smart environments: Technology, protocols and applications*, volume 43. John Wiley & Sons, 2004.

[12] A Corna, L Fontana, AA Nacci, and Donatella Sciuto. Occupancy detection via ibeacon on android devices for smart building management. In *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition*, pages 629–632. EDA Consortium, 2015.

[13] Sarah Darby et al. The effectiveness of feedback on energy consumption. *A Review for DEFRA of the Literature on Metering, Billing and direct Displays*, 486:2006, 2006.

[14] Stephen Dawson-Haggerty, Andrew Krioukov, Jay Taneja, Sagar Karandikar, Gabe Fierro, Nikita Kitaev, and David Culler. Boss: building operating system services. In *Presented as part of the 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 13)*, pages 443–457, 2013.

[15] Richard J De Dear, Gail Schiller Brager, James Reardon, Fergus Nicol, et al. Developing an adaptive model of thermal comfort and preference/discussion. *ASHRAE transactions*, 104:145, 1998.

[16] AL Dexter, G Geng, and P Haves. The application of self-tuning pid control to hvac systems. In *Control in Building Energy Management Systems, IEE Colloquium on*, pages 4–1. IET, 1990.

[17] AI Dounis, MJ Santamouris, and CC Lefas. Implementation of artificial intelligence techniques in thermal comfort control for passive solar buildings. *Energy conversion and management*, 33(3):175–182, 1992.

[18] Anastasios I Dounis and Christos Caraiscos. Advanced control systems engineering for energy and comfort management in a building

environment—a review. *Renewable and Sustainable Energy Reviews*, 13(6): 1246–1261, 2009.

[19] Ecobee. Ecobee Thermostat. https://www.ecobee.com, 2015. [Online; accessed 30/06/2015].

[20] Energy@Home JEMMA. Java energy management application (jemma). http://www.energy-home.it/SitePages/Activities/JEMMA.aspx.

[21] Varick L Erickson and Alberto E Cerpa. Thermovote: participatory sensing for efficient building hvac conditioning. In *Proceedings of the Fourth ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, pages 9–16. ACM, 2012.

[22] Povl Ole Fanger. Thermal comfort. analysis and applications in environmental engineering. *Thermal comfort. Analysis and applications in environmental engineering.*, 1970.

[23] Clifford C Federspiel. *User-adaptable and minimum-power thermal comfort control.* PhD thesis, Massachusetts Institute of Technology, 1992.

[24] Guang Geng and GM Geary. On performance and tuning of pid controllers in hvac systems. In *Control Applications, 1993., Second IEEE Conference on*, pages 819–824. IEEE, 1993.

[25] Ali Ghahramani, Farrokh Jazizadeh, and Burcin Becerik-Gerber. A knowledge based approach for selecting energy-aware and comfort-driven hvac temperature set points. *Energy and Buildings*, 85:536–548, 2014.

[26] Santosh K Gupta, Sam Atkinson, Ian O'Boyle, John Drogo, Koushik Kar, Sandipan Mishra, and John T Wen. Bees: Real-time occupant feedback and environmental learning framework for collaborative thermal management in multi-zone, multi-occupant buildings. *Energy and Buildings*, 125:142–152, 2016.

[27] Maher Hamdi, Gerard Lachiver, and François Michaud. A new predictive thermal sensation index of human response. *Energy and Buildings*, 29(2): 167–178, 1999.

[28] ASHRAE Handbook et al. Fundamentals. *American Society of Heating, Refrigerating and Air Conditioning Engineers, Atlanta*, 111, 2001.

[29] Robert K Harle and Andy Hopper. The potential for location-aware power management. In *Proceedings of the 10th international conference on Ubiquitous computing*, pages 302–311. ACM, 2008.

[30] Honeywell. Lyric Thermostat. https://http://lyric.honeywell.com, 2015. [Online; accessed 30/06/2015].

[31] D Int-Hout. Thermal comfort calculations/a computer model. *ASHRAE Transactions*, 96(1):840–844, 1990.

[32] En Iso. 7730: 2005. *Ergonomics of the thermal environment. Analytical determination and interpretation of thermal comfort using calculation of the PMV and PPD indices and local thermal comfort criteria*, 2006.

[33] J.J.K. Jaakkola, O.P. Heinonen, and O. Seppänen. Sick building syndrome, sensation of dryness and thermal comfort in relation to room temperature

in an office building: need for individual control of temperature. *Environment international*, 15(1):163–168, 1989.

[34] Pekka Tuominen & Irmeli Wahlgren Juha Forsström, Pekka Lahti, Esa Pursiheimo, Miika Rämä, Jari Shemeikka, Kari Sipilä. Measuring energy efficiency. Technical report, VTT Technical Research Centre of Finland, 2011.

[35] Henrik Karlsson and Carl-Eric Hagentoft. Application of model based predictive control for water-based floor heating in low energy residential buildings. *Building and environment*, 46(3):556–569, 2011.

[36] Abraham Hang-yat Lam, Yi Yuan, and Dan Wang. An occupant-participatory approach for thermal comfort enhancement and energy conservation in buildings. In *Proceedings of the 5th international conference on Future energy systems*, pages 133–143. ACM, 2014.

[37] Li Lan, Pawel Wargocki, and Zhiwei Lian. Quantitative measurement of productivity loss due to thermal discomfort. *Energy and Buildings*, 43(5): 1057–1062, 2011.

[38] Geoff J Levermore. *Building energy management systems: applications to low-energy HVAC and natural ventilation control*. Taylor & Francis, 2000.

[39] Jian Liang and Ruxu Du. Thermal comfort control based on neural network for hvac application. In *Proceedings of 2005 IEEE Conference on Control Applications, 2005. CCA 2005.*, pages 819–824. IEEE, 2005.

[40] Guillaume Lopez, Takahiro Tokuda, Naoya Isoyama, Hiroshi Hosaka, and Kiyoshi Itao. Development of a wrist-band type device for low-energy consumption and personalized thermal comfort. In *Mechatronics (MECATRONICS)/17th International Conference on Research and Education in Mechatronics (REM), 2016 11th France-Japan & 9th Europe-Asia Congress on*, pages 209–212. IEEE, 2016.

[41] Yudong Ma, Francesco Borrelli, Brandon Hencey, Brian Coffey, Sorin Bengea, and Philip Haves. Model predictive control for the operation of building cooling systems. *IEEE Transactions on Control Systems Technology*, 20(3):796–803, 2012.

[42] Hossein Mirinejad, Seyed Hossein Sadati, Maryam Ghasemian, and Hamid Torab. Control techniques in heating, ventilating and air conditioning (hvac) systems 1. 2008.

[43] Jin Woo Moon and Sung Kwon Jung. Development of a thermal control algorithm using artificial neural network models for improved thermal comfort and energy efficiency in accommodation buildings. *Applied Thermal Engineering*, 103:1135–1144, 2016.

[44] Alessandro A. Nacci, Bharathan Balaji, Paola Spoletini, Rajesh Gupta, Donatella Sciuto, and Yuvraj Agarwal. Buildingrules: A trigger-action based system to manage complex commercial buildings. In *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers*, UbiComp/ISWC'15 Adjunct, pages

381–384, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3575-1. doi: 10.1145/2800835.2800916. URL http://doi.acm.org/10.1145/2800835.2800916.

[45] Alessandro Antonio Nacci. *Methods and tools for effective smart buildings deployment*. PhD thesis, Italy, 2016.

[46] Alessandro Antonio Nacci, Vincenzo Rana, and Donatella Sciuto. A perspective vision on complex residential building management systems. In *Embedded and Ubiquitous Computing (EUC), 2014 12th IEEE International Conference on*, pages 209–214. IEEE, 2014.

[47] Nest. Nest Learning Thermostat. https://nest.com/thermostat/meet-nest-thermostat/, 2015. [Online; accessed 30/06/2015].

[48] Jose Luis Gomez Ortega, Liangxiu Han, and Nicholas Bowring. Modelling and detection of user activity patterns for energy saving in buildings. In *Emerging Trends and Advanced Technologies for Computational Intelligence*, pages 165–185. Springer, 2016.

[49] Sean Purdon, Branislav Kusy, Raja Jurdak, and Geoffrey Challen. Model-free hvac control using occupant feedback. In *Local Computer Networks Workshops (LCN Workshops), 2013 IEEE 38th Conference on*, pages 84–92. IEEE, 2013.

[50] 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*, pages 75–84. ACM, 2016.

[51] Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

[52] Elham Semsar-Kazerooni, Mohammad Javad Yazdanpanah, and Caro Lucas. Nonlinear control and disturbance decoupling of hvac systems using feedback linearization and backstepping with load estimation. *IEEE Transactions on control systems technology*, 16(5):918–929, 2008.

[53] Pervez Hameed Shaikh, Nursyarizal Bin Mohd Nor, Perumal Nallagownden, Irraivan Elamvazuthi, and Taib Ibrahim. A review on optimized control systems for building energy and comfort management of smart sustainable buildings. *Renewable and Sustainable Energy Reviews*, 34:409–429, 2014.

[54] John Shawe-Taylor and Nello Cristianini. *Kernel methods for pattern analysis*. Cambridge university press, 2004.

[55] Max Sherman. A simplified model of thermal comfort. *Energy and buildings*, 8(1):37–50, 1985.

[56] ASHRAE Standard. Standard 55-2004—thermal environmental conditions for human occupancy. *ASHRAE Inc., Atlanta, GA*, 2004.

[57] ASHRAE Standard. Standard 55-2010. *Thermal environmental conditions for human occupancy*, 2010.

[58] Joost Van Hoof. Forty years of fanger's model of thermal comfort: comfort for all? *Indoor air*, 18(3):182–201, 2008.

[59] Vladimir Vapnik. *The nature of statistical learning theory*. Springer Science & Business Media, 2013.

[60] Annamaria Andrea Vitali, Marco Spadafora, Alessandro A Nacci, Donatella Sciuto, and Margherita Pillan. Spell: affecting thermal comfort through perceptive techniques. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, pages 183–186. ACM, 2014.

[61] Fulin Wang, Zheliang Chen, Qingqing Feng, Qianchuan Zhao, Zhijin Cheng, Zhi Guo, and Zhipengn Zhong. Experimental comparison between set-point based and satisfaction based indoor thermal environment control. *Energy and Buildings*, 128:686–696, 2016.

[62] Qing-Guo Wang, Chang-Chieh Hang, Yong Zhang, and Qiang Bi. Multivariable controller auto-tuning with its application in hvac systems. In *American Control Conference, 1999. Proceedings of the 1999*, volume 6, pages 4353–4357. IEEE, 1999.

[63] Mark Weiser, Rich Gold, and John Seely Brown. The origins of ubiquitous computing research at parc in the late 1980s. *IBM systems journal*, 38(4): 693, 1999.

[64] Shui Yuan and Ronald Perez. Multiple-zone ventilation and temperature control of a single-duct vav system using model predictive strategy. *Energy and Buildings*, 38(10):1248–1261, 2006.

[65] Qianchuan Zhao, Zhijin Cheng, Fulin Wang, Yi Jiang, and Jinlei Ding. Experimental study of group thermal comfort model. In *2014 IEEE*

international conference on automation science and engineering (CASE), pages 1075–1078. IEEE, 2014.

[66] Qianchuan Zhao, Yin Zhao, Fulin Wang, Yi Jiang, and Fan Zhang. Preliminary study of learning individual thermal complaint behavior using one-class classifier for indoor environment control. *Building and Environment*, 72:201–211, 2014.

[67] Qianchuan Zhao, Zhijin Cheng, Fulin Wang, Zheliang Chen, Yi Jiang, and Zhipeng Zhong. Experimental assessment of a satisfaction based thermal comfort control for a group of occupants. In *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, pages 15–20. IEEE, 2015.