DEMAND RESPONSE THROUGH TRANSACTIVE CONTROL OVER PERSONAL ENVIRONMENTAL CONTROL SYSTEMS

Thesis submitted in partial fulfilment

of the requirements for the degree of

Doctor of Philosophy in IT in Building Science

by

Sam Babu Godithi

201415669

sambabu.g@research.iiit.ac.in

(Adviser: Prof Vishal Garg)

International Institute of Information Technology

Hyderabad – 500 032, INDIA.

May 2024

International Institute of Information Technology

Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "*Demand Response Through Transactive Control Over Personal Environmental Control Systems*" by Sam Babu Godithi (201415669), has been carried out under my supervision and is not submitted elsewhere for a degree.

Date: Adviser: Prof. Vishal Garg

This thesis is dedicated to the three most precious women in my life:

My mother, Late Smt. Godithi ***Raj Kumari*** *(Dondapati Yalana Swarajya Lakshmi), for inspiring me,*

My wife, ***Madhavi,*** *and daughter,* ***Swaramika*** *for their endless love, support, and encouragement.*

Acknowledgments

I embarked on the PhD journey with a dream – a dream to delve into the realm of research. I am married and have a daughter. Leaving my job to pursue a PhD was not merely a career choice, but the pursuit of my lifelong aspiration. I am indebted to those who have knowingly and unknowingly contributed to this journey.

Foremost, I wish to acknowledge the vision of Dr. B R Ambedkar, whose enduring legacy has enabled individuals like me, hailing from the Dalit community (so-called untouchable caste) in India, for whom education was once an unattainable distant star, to envision and realize such dreams.

Furthermore, I must extend my heartfelt gratitude to my supervisor, Prof. Vishal Garg, Professor and Head of the Centre for IT in Building Science, IIIT Hyderabad. Words alone cannot capture the depth of my appreciation. Without his mentorship, support, and encouragement, this journey would have been significantly more challenging. Prof. Garg has been my confidant, providing unwavering flexibility and opportunities for me to shine. Under his leadership, I have surmounted countless obstacles and gained profound insights. He has consistently challenged me and nudged me toward the right path. Despite his busy schedule, he has reviewed my progress, offered invaluable suggestions, and bolstered my spirits when they faltered. I owe him an immeasurable debt of gratitude.

During the course of PhD, I have faced numerous challenges. These difficulties have been both physical and mental, and medical and financial. The COVID-19 pandemic, with its confinements, hindered my progress. I grappled with dengue fever, a battle that led me to the intensive care unit, where I faced a cocktail of antibiotics. I confronted clinical depression (RDD - Severity 20/27). My financial woes escalated, causing substantial stress and pressure. Furthermore, the loss of my father, who was my hero and pillar of strength, was a profound blow. However, despite the immense challenges, my dedication to my PhD remained unwavering. This commitment was made possible through the unwavering support and assistance I have received from the family, the institute, and friends.

Notably, the financial assistance extended by Mrs. Meenakshi Viswanathan (and Mr. Jairam), Prof. Vishal Garg, Ramesh Chandranath Vallabhaneni, Anil Choudary, Ranjan George, and Niranjan, as well as the support from PG-Cell - Prof Jawahar and Prof Ponnurangam Kumaraguru (PK), have been invaluable. For which I am sincerely grateful and deeply obliged.

Besides, I would also like to acknowledge Christian Kohler and Richard Brown (Lawrence Berkeley National Laboratory), Prof. Sekhar Narayana Kondepudi (National University of Singapore), Dr Srinivas Katipamalu (Pacific Northwest National Laboratory), and Prof. K S Rajan and Prof. Kishore Kothapalli (IIITH), who dedicated their precious time to assessing my work and providing invaluable insights and guidance, and directions on how best to conduct the research.

Additionally, I extend my gratitude to Aviruch Bathia, Prabhakar Rao, Niranjan Reddy, and Giri for their technical assistance in designing, building, testing, troubleshooting, and maintaining the necessary hardware and test labs. And I greatly appreciate Ramana Garu, Srinivas Garu, Appaji Garu, Kishore Garu, Rambabu Garu, Bal Santosh Garu, Nalini Kumar Garu, Sailaja Garu, Prathima Mandapati Garu, and Pushpalatha Garu for their support. Special thanks to Debbie Redmond (Nymi) for her diligent review of the thesis and invaluable feedback.

Finally, I would like to acknowledge the Department of Science and Technology (DST), Government of India, for their funding support for this thesis through the bilateral Indo-US Center for Building Energy Research and Development (CBERD) and a joint India-UK call on the Energy demand reduction in the Built Environment (RESIDE) projects. Furthermore, I want to express my gratitude to the CBERD team, particularly Christian Kohler, Richard Brown, Reshma Singh, and Prof Rajan Rawal, for their guidance and support. It is a privilege to be associated with them for the review paper and the US patent.

Thank You All!

Abstract

The building sector consumes a significant amount of energy. The three major building performance areas are comfort, energy efficiency, and demand response capabilities.

In the recent past, widespread penetration of Distributed Energy Resources (DER) have increased interest in using end-user devices and equipment in buildings as flexible devices to balance grid supply and demand, i.e., "*Grid-Responsive*" buildings. DER refer small-scale power generation or storage technologies that can be deployed close to the point of energy consumption. These resources are often decentralized and include renewable energy sources such as solar panels, wind turbines, and energy storage systems like batteries. Integrating DER introduces many operational challenges and uncertainty into the grid.

Demand Response (DR) strategies are applied to address these challenges. DR refers to the practice of actively adjusting electricity consumption in response to signals such as price or incentives from grid operators, utilities, or aggregators. It involves reducing or shifting electricity usage during peak periods or in response to grid constraints to balance supply and demand, enhance grid reliability, and avoid or mitigate grid emergencies.

The approach that is used to manage the DR using price as the key operational parameter is called Transactive Controls (TC). TC is a market-based control paradigm that uses "*price*" as the key operational parameter. GridWise Architecture Council defined it as "*a set of economic and control mechanisms that allow the dynamic balance of supply and demand across the entire electrical infrastructure using value as a key operational parameter*". Economists extensively dealt with TC in microeconomics. However, TC is a domain-free approach that integrates market-based coordination and value-based control for a group of

resources to achieve global objectives.

Furthermore, the Building Energy Management Systems (BEMS) manage TC at the building or zone level in a commercial building. For example, in case of load shedding, it switches off low priority zone air conditioning or raises the set-point for cooling irrespective of usage in the whole building; dims all the lights regardless of the occupants and their needs. The occupant does not get an opportunity at the time of the DR event to decide on which comfort parameters he is willing to forgo to meet the energy demand.

Besides, building energy managers usually operate the buildings to maintain homogeneous indoor ambient conditions (like zone temperatures and lighting). However, occupants have individual thermal and visual comfort preferences. Maintaining a homogeneous indoor environment throughout the building/zones leads to unnecessary energy consumption as well as the inability to meet the comfort needs of the occupants. This has led the building science research community to pursue Personal Environment Control Systems (PECS), such as local thermal conditioning systems like heated computer keyboard, personal heaters, desk fans, and radiant cooling cubicles and task lighting systems such as desk lamps. PECS create favourable micro-ambient conditions around each occupant.

Nevertheless, the literature study shows that there is a gap and a need for a system/framework to integrate PECS within the task environment and between task & ambient controls to address the challenges. Hence, as part of this thesis, a system, iSPACE - **i**ntelligent **S**ystem for **P**ersonal **A**mbient **C**ontrol and energy **E**fficiency, has been developed to address the challenges. iSPACE integrates transactive control methods to facilitate demand response strategies within personal environmental control systems.

This thesis introduces an innovative approach to integrating Personal Environmental Control Systems (PECS) into building energy management, emphasising demand response through transactive controls. It addresses the limitations of traditional demand response mechanisms by empowering individual occupants to manage energy usage at a granular level. The proposed framework establishes a hierarchical distributed multi-agent system architecture, seamlessly integrating PECS devices like SmartHub, RadiantCubicle, and SmartStrip into building energy management systems.

The focal point of this thesis is the implementation of transactive control within the building environment, facilitating interactions between energy-consuming devices and the building systems at a localized level. While traditional supply-demand balancing occurs at the grid level by utilities or balancing authorities, our research explores the efficacy of implementing transactive control within buildings to optimize energy usage and enables end-users to participate in managing DR events at the task level based on their priorities.

Personalised demand response strategies are facilitated by enabling real-time interaction between occupants and energy-consuming devices. Key contributions include developing and implementing transactive controls at the task level, allowing users to adjust energy consumption based on dynamic pricing signals. Detailed features, system conceptualisation, mathematical modelling, and pricing formulation are provided, along with comparisons of transactive control platforms. The framework's efficacy is demonstrated through case studies and simulations. The metrics indicate a 26% energy demand flexibility compared to the benchmark, highlighting the substantial flexibility it offers for demand response management. Also, demonstrated that 100% convergence is possible by simulation study using the ground truth data derived from the experiment, which consisted of 1M test runs each at various levels (Building, Zone, and Task levels) on random system states and various parameter changes. Recommendations address system performance, scalability, and limitations, supported by an online codebase and hardware design. Establishing a functional testbed with the iSPACE prototype at IIITH's FDD lab fosters further research and collaboration. Overall, this thesis offers insights and methodologies to advance energy management practices in smart buildings, contributing to more sustainable and responsive built environments.

Keywords: Building Energy Conservation, Transactive Control, Demand Response, Personal Comfort Systems.

Table of Contents

List of Tables

List of Figures

List of Pseudo Code

Chapter 1

Introduction

"Space and light and order. Those are the things that men need just as much as they need bread or a place to sleep." – Le Corbusier, The New York Times [obituary] (28-08-1965).

This thesis addresses the concept of using transactive control methods to implement demand response strategies within personal environmental control systems.

Also, in the recent past, widespread penetration of Distributed Energy Resources (DER) have increased interest in using end-user devices and equipment in buildings as flexible devices to balance grid supply and demand, i.e., "Grid-Responsive" buildings [1], [2], [3], [4]. DER refer to various small-scale power generation or storage technologies that can be deployed close to the point of energy consumption [5]. These resources are often decentralized and can include renewable energy sources such as solar panels, wind turbines, and energy storage systems like batteries.

Integrating DER into smart grids and buildings introduces concerns around bidirectional power distribution, grid stability, complex modelling, low inertia systems, and unpredictable variables [6]. These complexities have driven the shift towards Demand Response (DR) strategies [7]. There is an increasing consensus within the research domain on the significance of integrating end-consumers in DR strategies, employing market-driven mechanisms like Transactive Control (TC) to bolster the efficiency and reliability of building operations [8], [9], [10], [11].

However, existing research on DR mechanisms in buildings targets a macro perspective, focusing on entire building or zone levels; it often overlooks the granular possibilities of load reduction at an individual task level, tailoring to user priorities. More profound benefits are possible if the optimal quantum of services at the task level can be measured (i.e., energy demand footprint with an optimal combination of thermal/lighting/connected loads with the ambient controls) and prioritised as per end-user choices.

Furthermore, conventionally, building operations aim for a uniform indoor ambience. Such a one-size-fits-all approach often clashes with the diverse comfort needs of the occupants. The innate human desire for personalised thermal and visual settings challenges the status quo of homogenised indoor environments, leading to both unnecessary energy wastage and unmet comfort needs. This divergence motivates the building science community to pursue Personal Environment Control Systems (PECS). PECS create favourable micro-ambient conditions around the occupant, equipped with typical devices like localised thermal regulators (like heaters, fans, radiant cooling), task lighting systems, and more. However, many PECS studies focused on energy conservation and enhanced personal comfort, and the challenges of integrating PECS with building systems and a unified interface still need to be explored.

Hence, for an effective grid-responsive building, a system that allows all the systems to exchange data in a unified way, capture end-user preference, deliver services, and measure energy footprint at the task level is needed [12]. By empowering the end-user to adapt to fluctuating in-building power rates — indicative of both grid and local power accessibility and concurrent demand — such a system will promote active user involvement in demand response management and ensure optimal energy usage that is aligned with user priorities. Further, instead of just keeping transactive control at the building level, it can be brought to the user space, thus enabling the users to perform cooperative trading to achieve comfort by optimal resource allocation.

1.1 Motivation And Background

The building industry is pivotal in both industrial and economic sectors, profoundly impacting the quality of life and the environment. A closer look at energy consumption data, such as those presented by the Energy Information Administration (EIA) [3], [13] and the International Energy Agency (IEA) [14], [15], showcases the building sector's immense energy consumption, approximately 30% globally. When juxtaposed with the transportation and manufacturing sectors, consuming 27% and 31%, respectively (Figure 1-1). Per World Energy Outlook 2023 by IEA, over the last decade, total energy consumption in the buildings sector has increased by an average of 1% per year, with electricity now accounting for over one-third of energy demand within this sector [15]. Rapid population growth and increasing income level, particularly in regions like India, Africa, and Other Asia-Pacific, are driving continued growth in energy consumption within building[15].

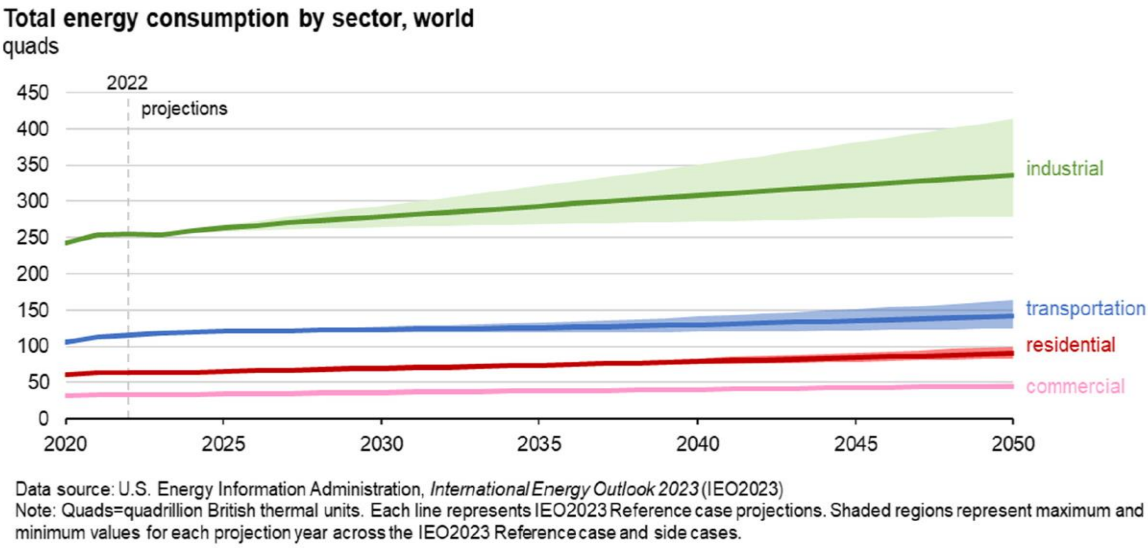


Figure 1-1: World total energy consumption by sector

Furthermore, market dynamics and regulatory frameworks have accelerated the integration of DERs like fuel cells, photovoltaics, and wind turbines. Notably, renewable energy sources now cater to 40% of the surge in primary energy demand [14]. Policy and cost drivers have notably propelled the growth of renewable energy, with solar and wind energy consumption outpacing other sources, contributing to a substantial increase in the non-fossil fuel share of primary energy from 21% in 2022 to a projected range of 29% to 34% by 2050 [3] and it is reported that the anticipated surge in renewable energy consumption is primarily driven by its increased use for electric power generation (Figure 1-2).

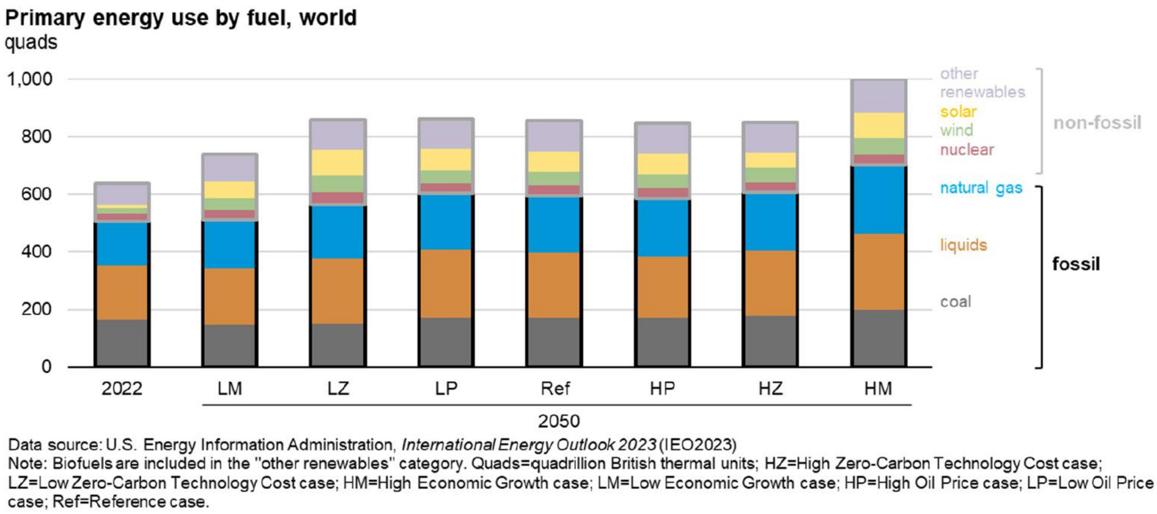


Figure 1-2: World primary energy use by the fuel

The building sector in India currently accounts for over 30% of the total electricity consumption, with residential and commercial sectors comprising approximately 75% and 25%, respectively [16]. According to the EIA's International Energy Outlook 2023, energy consumption in commercial and residential buildings in India is forecasted to triple by 2050 compared to 2022 [3]. This growth is particularly pronounced in the commercial (office, hospitality, retail, hospitals) and residential sectors. In 2010, as part of the ECO-III project, it was estimated that 70% of the building stock expected by 2030 was yet to be constructed in India [17]. In addition, in 2017, Kumar et al. developed the Commercial Building Stock Energy Modelling (CBSEM) to provide national-level estimates of floor area and energy consumption for various commercial building types. According to the results of the CBSEM, as of 2017, the total floor area of commercial building stock is 1.1 billion m^2, with an energy intensity of 68 kWh/m^2, and over the next decade (by 2027), the commercial floor area is projected to increase to 1.78 billion m^2, representing an approximate 62% increase with an energy intensity of 81 kWh/m^2, an increase of approximately 19% [18]. Besides, a 2021 report, India Energy Outlook 2021 by the IEA revealed that energy demand in buildings has surged by 40% since 2000, and it is projected the electricity and renewable electricity consumption to triple by 2040 compared to 2019 level [19]. Consequently, India faces the challenge of enhancing efficiency amidst the explosive growth in floor space and the commercial sector's escalating energy demands requirements.

Nevertheless, despite this extensive energy consumption, satisfaction with the indoor environment is often not achieved. Hence, demand response strategies using transactive control to manage supply and demand using economic or market constructs and personal environment control systems that create favourable micro-ambient conditions around each occupant have gained significant interest among the research community.

The following three sections give a brief introduction/background of demand response, and transactive control, personal environment control systems. Later in the literature survey chapter, gaps identified in these fields that need further studies are presented.

1.1.1 Demand Response (DR)

The electric grid can be made “smarter” and more resilient by using innovative technologies, computer processing, and controls systems that communicate and work together to deliver electricity more reliably and efficiently. Factors like energy market liberalization, decentralization of energy generation, and climate protection imperatives have galvanized this transformation [20]. Growing consumer interest in clean energy, combined with governmental regulations, is accelerating the integration of DER, such as PVs, fuel cells, and wind power, into the contemporary electric grid. Integration of DER units introduces many operational challenges like bidirectional power flow, grid stability issues, modelling, low inertia, and uncertainty. This has driven the industry towards DR strategies.

According to the Federal Energy Regulatory Commission, Demand Response is defined as "*Changes in electric usage by end-use customers from their normal consumption patterns in response to changes in the price of electricity over time, or to incentive payments designed to induce lower electricity use at times of high wholesale market prices or when system reliability is jeopardized*" [21].

In the context of a smart energy system, as outlined by Lund et al., DR mirrors the "*Dynamic Demand*" mechanisms that modulate electricity consumption based on supply dynamics ([9], [22]). DR refers to the practice of actively adjusting electricity consumption in response to signals such as price or incentives from grid operators, utilities, or aggregators. It involves reducing or shifting electricity usage during peak periods or in response to grid constraints to balance supply and demand, enhance grid reliability, and avoid or mitigate grid emergencies. As Lasseter et al. described, these mechanisms can involve building managers curtailing consumption during peak periods or reacting to market prices, [23]. To facilitate the orchestration of DR in smart grids, advanced DR initiatives and cutting-edge technologies, such as Advanced Metering Infrastructure (AMI), energy controllers, Energy Management Systems (EMS), and both wired and wireless communication platforms, are essential, as highlighted by Siano [7].

Consumer devices, including smart meters, in-home displays, load control instruments, and thermostats, embed DR functionalities. These functionalities allow homeowners to engage in residential demand response initiatives. In contrast, demand response management in commercial buildings primarily occurs at the building level. This strategy often means the individual occupant lacks the autonomy during a DR event to choose which comfort features they are willing to sacrifice for energy conservation. Typically, the decision is in the hands of the building's energy manager. For instance, during load shedding, strategies may include turning off air conditioning in deemed low-priority zones without considering actual occupancy, uniformly increasing the cooling set-point throughout the building, or globally dimming lights, regardless of the occupants' preferences and needs.

1.1.2 Transactive Control (TC)

Transactive Control is a domain-agnostic approach that integrates "*market-based coordination*" and "*value-based control*" for a group of resources to achieve global objectives [5]. The TC approach is well studied in microeconomics and similar approaches are successfully applied in other areas.

The term "*transactive control*" is used to refer to techniques that manage the supply and demand of energy in a system by using economic or market-based constructs while considering grid reliability constraints and building energy efficiency. Building energy efficiency is closely related to TC but not inherently a part of it. While TC focuses on the dynamic pricing and control mechanisms to balance supply and demand in real-time, building energy efficiency primarily concerns measures and technologies to reduce overall energy consumption within buildings over time. While TC may influence energy usage patterns within buildings, addressing energy efficiency typically involves longer-term strategies such as improving insulation, upgrading HVAC systems, or implementing energy-efficient appliances. Therefore, while related, building energy efficiency operates on a different time scale and focuses on different aspects compared to TC. The term "transactive" stems from the notion that decisions are based on value. These decisions may be analogous to economic transactions. Moreover, as defined by [24], "*transactive energy*" means "*a set of economic and control mechanisms that allow the dynamic balance of supply* *and demand across the entire electrical infrastructure using value as a key operational parameter.*" In an equilibrium market, a price is established such that supply meets demand.

A typical TC system is shown in Figure 1-3. Kelly et al. adopted this TC for rate control algorithm using shadow prices for communication networks[25]; Samadi et al. adopted for optimal real-time pricing algorithm for smart grids[26]; and Akkermans et al. adopted for PID controllers in buildings [27].

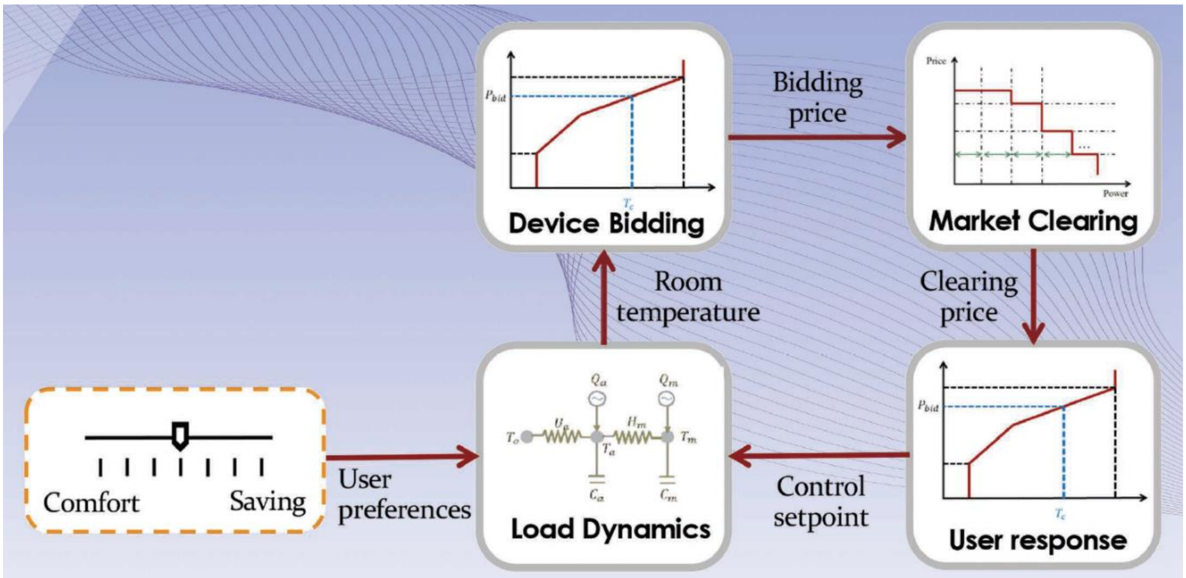


Figure 1-3: Transactive Control Systems: The Market-Based Coordination of Distributed Energy Resources © 2020 IEEE. Reprinted, with permission, from Li et. al. (Aug 2020), IEEE Control Systems

As explained by Clearwater, TC is "*Market-based control is the paradigm for controlling complex systems that would otherwise be difficult to control, maintain, or expand. A very abstract definition of a market is a system with locally interacting components that achieve coherent global behaviour. The fascinating aspects of a market are that through the simple interactions of trading, i.e., buying and selling, among individual agents, a desirable global effect can be achieved, such as stable prices or fair allocation of resources. People have used markets for thousands of years to get things done.*" [28].

In short, Economics is defined as the study of how people make decisions in resource-constrained conditions [29]. The economics study is divided into two subfields, Macroeconomics and Microeconomics. The scope of Macroeconomics deals with theories about decision made at a national or global scale. In contrast, Microeconomics deals with theories about decisions made on a personal scale and includes studying individual and group choice within market and nonmarket processes.

"*Microeconomics is often called price theory to emphasise the important role of prices. Microeconomics explains how the actions of all buyers and sellers determine prices and how prices influence the decisions and actions of individual buyers and sellers*" [30].

The supply-demand relationship is assumed to depend on the utility functions. As defined by Perloff, a utility function describes the degree of well-being that a product provides for consumers; i.e., it defines different responses to various prices [30]. If a consumer's preferences have the properties of completeness[1] and transitivity[2], then we say that the user's preferences are rational. People are rational, and rationality requires maximisation. Actors in the market (that is sellers, buyers, and agents) do not intend for equilibrium to result. Instead, they try to maximise whatever is of interest to them.

A systems-level theory of large-scale intelligent and distributed control was formulated in studies by Akkermans et al. [27] and Kok [31] (Figure 1-4). The study presents the control strategies for an interactive society of actors represented by agents, each with an individual control task. Many software agents are competitively negotiating and trading on an electronic market to optimally achieve their local control action goals in a market-based control.

[1] The completeness property holds that, when facing a choice between any two bundles of goods, a consumer can rank them using preference relation. This property rules out the possibility that the consumer cannot decide which bundle is preferable.

[2] According to this property, a consumer's preferences over bundles is consistent in the sense that, if the consumer weakly prefers a to b, and weakly prefers b to c, then the consumer also weakly prefers a to c.

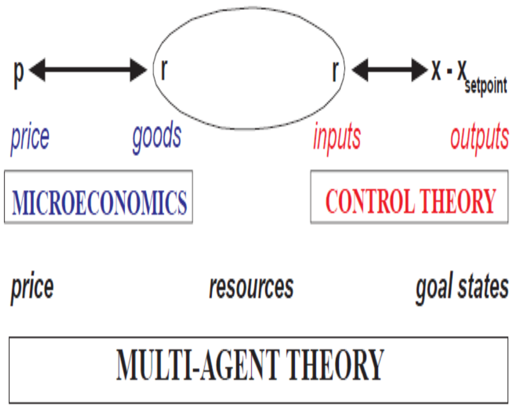


Figure 1-4: Microeconomics and control engineering unified in multi-agent theory
Source: from [27]

These theories unify microeconomics and control theory into a multi-agent system. In [27], a general market theorem was derived that proves two important properties about agent-based microeconomic control:

1. computational economies with dynamic pricing mechanisms can handle scarce resources for control adaptively in ways that are optimal locally as well as globally ('*societally*'), and
2. in the absence of resource constraints the total system acts as a collection of local independent controllers that behave per conventional control engineering theory.

The mathematical underpinning for the system is adopted from the principle of rate control algorithm using shadow prices for communication networks [25], optimal real-time pricing algorithm based on utility maximization for smart grids [26], and general market theorem for agent-based microeconomic control [27] which demonstrates that computational economies with dynamic pricing mechanisms can handle constrained resources for control adaptively that are optimal locally as well as globally ('*societally*') and the interactions of maximizing agents usually result in equilibrium with fairness and stability. For these class of algorithms with appropriate formulation of the overall optimisation problem, the stability and fairness is assured.

With the increasing penetration level of DERs and renewable energy sources in power systems, TC is emerging as one of the most innovative and effective approaches towards the future smart grid [32]. TC approach applied to networks is called Transactive Network [25]; and approach that is applied to energy systems is called Transactive Energy [26]. Throughout this thesis, the term Transactive Control (TC) has been consistently employed. The Department of Energy's (DOE's) Building Technologies Office (BTO) supports the development of the concept of TC to enable energy, operational, and financial transactions between building systems (e.g., rooftop units -- RTUs), and between building systems and the electric power grid [33]. The research community has demonstrated the potential benefits of TC at grid-building [26], [34], [35] and building zones [36], [37], [38].

However, there has been no study on implementing transactive control for demand response through personal environmental control systems to involve the end-user in the decision-making process. There exists a necessity for a system that empowers end-users to participate in demand response management actively, striving to achieve a global optimum by harmonizing individual objectives. In this research, we broaden these concepts to the task level.

1.1.3 Personal Environmental Control Systems (PECS)

At its core, a PECS, also known as a task-ambient conditioning system (TAC), is defined as "*any space conditioning system that allows local conditioning (e.g., regularly occupied work locations) to be individually controlled by building occupants while still automatically maintaining acceptable environmental conditions in the ambient space of the building*" [39]. One alternative approach to achieving higher levels of occupant thermal comfort works by manipulating the occupant's perception of their environment without significant heat transfer [40]. Figure 1-5 shows a few example PECS.

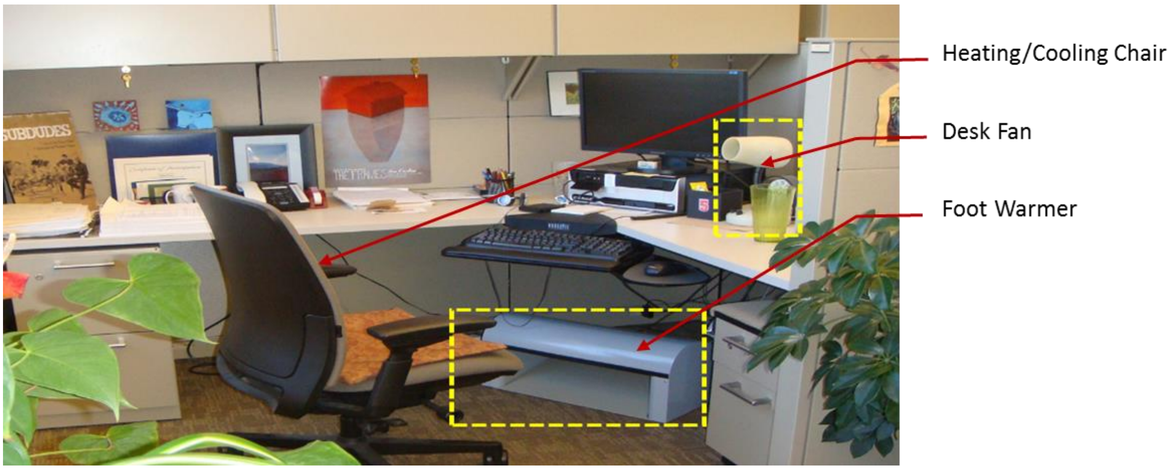


Figure 1-5: Task Ambient Conditioning (TAC) system
Source: http://www.cbe.berkeley.edu

Prof. Frederick H. Rohles Jr., a renowned personality in areas of HVAC, discussed several aspects of thermal environmental conditions for human occupancy and stated that at extreme temperatures, we all respond the same and at the so-called “comfortable” temperatures that we differ the most [41]. This observation indicates that caution should be used when predicting a response to a temperature value in the middle range. He further emphasised the importance of the individual's personal preference and that minor and apparent insignificant variation in the environment can alter a person’s condition of mind. Moreover, the facility manager measures energy use but few measure comfort [42].

However, most PECS studies have focused on improving energy efficiency and personal comfort, often sidelining integrating PECS within the task environment and between task and ambient controls. Further, the existing studies do not address interoperability and communication issues.

1.2 Problem Statement

There is a need to have a framework that connects personal environment systems with ambient environment systems, thereby integrating the user into the DR infrastructure through TC. There is no standardized framework which can seamlessly connect devices across the task, ambient, and building levels, ensuring user participation in DR scenarios while leveraging the concepts of TC.

1.3 Aim and Research Questions

This research aims to create an integrated PECS that uses TCs. This system will make buildings grid-responsive, allowing end-users to manage DR events at the task level, prioritize load reduction, and maintain their comfort preferences. A significant component of this work involves establishing a framework for a distributed, agent-based PECS. This system will react to local power prices within the building, which indicates power availability from both the grid and on-site generation.

To realize this aim, we seek to answer the following research questions:

1. How can we effectively integrate PECS into the task environment?
2. What methods will allow the task environment system to connect seamlessly with the ambient system?
3. How can we deploy TC to ensure DR management at the task level?

1.4 Methodology

For this research work, a system-building research methodology was adopted. This approach involves developing a system or its components that offer significant enhancements in performance or functionality previously unavailable.

As part of this thesis, initially, a generalised architecture and framework of an iSPACE system has been developed. This system facilitates end-user participation in real-time DR, managing the energy usage with much more granularity (i.e., at a task level). And it allows the occupant to choose the comfort parameter that they are ready to forgo in the case of a demand reduction scenario. Furthermore, a functional prototype system was developed and tested in a lab environment and evaluated for functional and non-functional performance. Using the ground truth data derived from these evaluations, simulation studies, which consisted of 1M test runs on random system states, were conducted to determine the convergence rate.

1.4.1 System Description

A comprehensive system description was formulated, that details the critical features and requirements of the iSPACE system at multiple abstraction levels. This description encompasses the system's 'what,' 'how,' 'where,' 'who,' 'when,' and 'why'. Additionally, artifacts like use cases, class diagrams, and UML models were presented. The interactions and roles of several system actors, including Building Management Systems (BMS), PECS, sensors, DR events, users, trading entities, and others, were also developed.

1.4.2 End-to-End Functional System Development

A proof-of-concept (POC), lab-scale prototype functional system has been developed to evaluate specific use-cases. In its developement,

- The necessary hardware and software components of the systems were created and implemented,
- Available existing platforms/devices (modified accordingly) to suit the needs of the system were used, and

- Relevant cost functions, utility functions and other simulated data streams available in the literature were used.

1.4.3 System Evaluation

The developed system was critically analysed in a laboratory setup through a combination of quantitative and qualitative analyses. Simulation studies, rooted in the ground truth data from the experiments, were conducted to understand the convergence rate. Insights regarding system performance were subsequently discussed and reported.

1.5 Organisation of The Thesis

The thesis is organised into six chapters as follows:

1. *Chapter 1 Introduction*: The current chapter gives an overview and background for the research. Also, the issues and research questions relevant to the research are identified in this chapter.
2. *Chapter 2 Literature Survey*: This chapter provides a literature survey to identify the current methods and gaps that this research addresses.
3. *Chapter 3 iSPACE – intelligent System for Personal-Ambient Control and Energy efficiency*: This chapter describes the system, the framework, and the structure underlying the system as mentioned in the methodology section 1.4.1 are described in this chapter.
4. *Chapter 4 Development of PECS (SmartHub, SmartStrip and RadiantCubicle)*: The hardware and software components that were developed for complete functional testing as mentioned in the methodology section 1.4.2 are detailed in this chapter.
5. *Chapter 5 System Evaluation*: A pilot setup deployed in a controlled laboratory environment and tested to evaluate the system, as explained in the methodology

section 1.4.3, is detailed in this chapter. A critical discussion concludes this chapter, which draws pertinent insights into the comprehensive understanding of the system's strengths, potential areas for improvement, challenges, including scalability, and limitations of the systems and recommendations.

6. *Chapter 6 Conclusion*: Finally, the concluding chapter encapsulates the summary and conclusion of the research undertaken, its scope, the implications of the current work, and the significant contributions.

Chapter 2

Literature Survey

"What the eye doesn't see, and the mind doesn't know, doesn't exist." – D. H. Lawrence

2.1 Introduction

A comprehensive understanding of the field forms the backbone of any research endeavour. In the context of this study, the literature survey delves into categories crucial for establishing a contextual understanding. The comprehensive literature review had been published as a review article in a journal [12]. The gaps identified through the literature survey are briefed in section 2.6. The research gaps identified in this paper are the ones that are addressed in this thesis.

The following sections present a distilled version of this comprehensive review. The significant areas covered are:

1. **Demand Response (DR)**: This focuses on understanding the mechanisms wherein end-user consumption patterns can be influenced in response to external signals, particularly from energy providers. Such systems allow for more dynamic and adaptive energy management.
2. **Transactive Controls (TC)**: A pivotal area, TC revolve around the economic and control techniques that enable automated transactions and negotiations between various entities, including users and utilities, in energy systems.
3. **Personal Environment Control Systems (PECS)**: PECS are imperative for crafting tailored environmental experiences. These systems allow users to modify

and maintain their immediate surroundings to their preferences, enhancing comfort and productivity.

4. **Integration of PECS, DR, and TC**: This dimension explores how the synergy between PECS, DR, and TC can lead to holistic systems that maximize user comfort and energy efficiency.

2.2 Demand Response (DR)

According to [43], the evolution of building energy supply systems has grown increasingly intricate. The traditional expectation that electricity would arrive at their respective meters within a building no longer holds. Instead, energy conversion, heat recovery, and renewable energy capture now happen concurrently at multiple points within the building's energy infrastructure. Modern buildings, often described as '*prosumers*', act both as energy consumers and producers[10]. Such buildings can contribute to grid stability by managing their overall electrical demand in response to current smart grid conditions. Lawrence et al. suggest that a promising approach to enhance buildings' interaction with the smart grid is to break down consumption data at the equipment and zone levels within the building, enabling more precise demand reduction targeting [10].

Further, Lawrence et al. suggested that control capability and data exchange are fundamental keys to integrating a smart grid and smart buildings [10]. Though consumer devices like smart meters, in-home displays, load control devices, thermostats, etc., incorporate some Demand Response (DR) event handling for the home users to participate in the residential demand response programs. However, demand response management operates at a broader building level in commercial buildings. During a DR event, the individual occupant (end-user) does not get the choice to determine which comfort features they are willing to compromise for energy demand adjustments. Typically, these decisions fall under the purview of the building's energy manager. Such decisions entail turning off air conditioning in less crucial zones without considering actual use, universally increasing cooling set-points, or dimming lights without assessing the specific needs and preferences of the occupants.

The pricing methods and range of optimization algorithms available in the literature in the context of demand response programs for the smart grids are presented in a study by Vardakas at al. [44]. A few of the demand response methods based on offered motivation to the participating customer are as follows:

1. Time-Of-Use (TOU): customers are charged different rates for different periods.
2. Critical Peak Pricing (CPP) is like TOU, but at least the price of one period can change. The customer receives a notification of the price change, usually a day ahead.
3. Peak Load Pricing (PLP): The day is divided into several periods, and different prices are determined for each period. These prices are announced to the customers ahead of each day. The price value for each period is calculated based on the average power consumption of the customers in each period to maximize the payoff to the energy provider. In addition, it is used for peak load shifting, expecting a reaction from the customers to the higher prices.
4. Peak Day Rebates (PDR) or Peak Time Rebates (PTR): customers are under their standard tariff, but they have an opportunity to receive a rebate payment for any load reduction.
5. Real-Time Pricing (RTP): the energy provider regularly announces new electricity prices on a rolling basis. The new prices are based upon changing needs (random events impacting the supply-demand) and the customer's responses to the previous prices.

A survey of the potential benefits of DR in smart grids is presented in a study by Siano [7]. Also, as one of the examples of industrial case studies, a case study of Amy's kitchen facility in California is presented. The facility participated in a DR program using OpenADR[3]. The facility includes several large cool rooms, freezers, blast freezers, a spiral freezer, and multiple support loads, such as HVAC and lighting. The utility notifies the facility of a day-ahead using OpenADR about the DR event period. The signals are received by the facility EMS that is associated with an OpenADR client. When the DR event starts, the EMS triggers preprogrammed DR strategies such as shutting off some freezers and the battery chargers and raising the set point on other freezers and cold rooms. The study highlighted critical research areas that need further investigation, including measurement and settlement processes, developments in integrated electronic circuits, optimisation and control systems, and information and communications technologies. A DR control system is presented for a commercial building in [45]. The study presents a method to determine the DR potential of the building considering occupant comfort. However, these studies are limited to either building level or zone level.

In summary, the successful promotion of DR programs necessitates the active involvement of end-consumers in the energy supply chain. This hurdle can be addressed by channelling DR event notifications, received by Building Energy Management Systems (BEMS), directly to end-users via the PECS, allowing the end-user to handle the DR event at the task level. Further, the issues raised in the above studies need to be studied from this perspective.

[3] Open Automated Demand Response (OpenADR) is an open and standardized way for electricity providers and system operators to communicate DR signals with each other and with their customers using a common language over any existing IP-based communications network, such as the Internet. http://www.openadr.org/

2.3 Transactive Control (TC)

There are many approaches to energy management. As discussed in [11], the approaches can be classified into four main categories vis-a-vis top-down switching, price reaction, centralised optimisation, and transactive control and coordination (Figure 2-1).

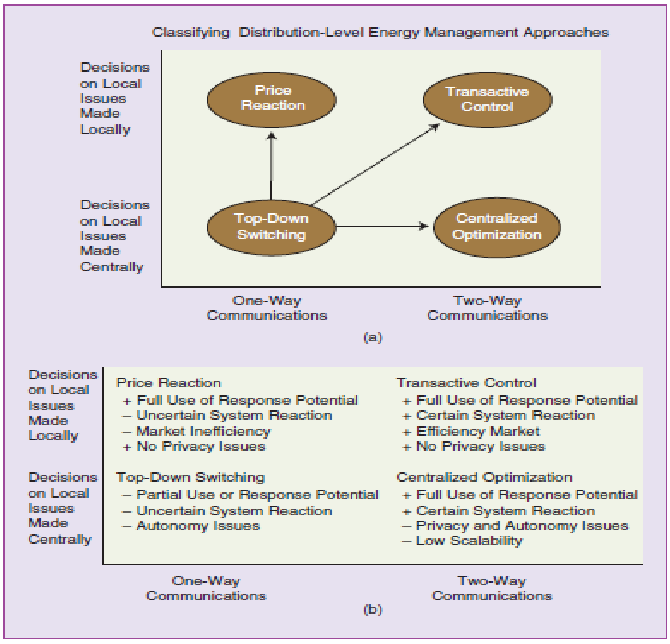


Figure 2-1: The energy management matrix
© 2016 IEEE. Reprinted with permission from [11]

Furthermore, as discussed in [17] and [36], TC implementation raises the following research challenges:

- Multi-objective optimisations issues because of the incorporation of DER user's priorities/needs/utilities/costs into the operation of the power systems to meet all the objectives and constraints,
- The price-response behaviour of DERs and the need to design the optimal pricing strategy,
- The need to how to create and operate a market where efficiency and transparency are guaranteed,
- On the method front, how to devise strategies that guarantee the convergence of transactive control applications and expedite the convergence rate,
- ICT infrastructure for communication among various stakeholders, and
- The standardisation of an interface of transactive control is essential for successful implementation.

As part of DOE’s transactional network initiative, PNNL has developed an open-source, open-architecture platform Eclipse VOLTTRON™ to deploy energy efficiency and grid services [33]. PNNL studied two applications, 1) Energy efficiency and 2) Grid service for networked rooftop units (RTU). The energy efficiency service includes automated fault detection and diagnostic for packaged RTUs and air handling units (AHU). Moreover, as a grid service application, it minimises electricity consumption during peak periods on a critical peak pricing day. The system synergizes demand response with transactive control via a decentralized, agent-driven control mechanism. It reacts to localized power pricing within the building, representing the power supply from the grid and on-site generation. This mechanism facilitates a more agile demand-response management in buildings, countering the sporadic energy production from renewable sources.

The TC implementations focus on residential devices like water heaters, refrigerators, washing machines, electric vehicles' charging, utility devices or Rooftop HVAC Packaged Units (RTU) and Air Handling Units (AHU) for commercial buildings. However, it needs to be noted that the issues raised need to be addressed for transactive control implementation in personal environment control systems. Further, more studies are needed to calculate the baseline and use the ‘shadow price’ in practice.

Besides, only large producers and consumers participate in equilibrium markets, and small consumers and produces are excluded as it is difficult to handle large participants. However, this limitation can be overcome with an appropriate market mechanism [47].

2.4 Personal Environmental Control Systems (PECS)

The quest for personal comfort in built environments led to the emergence of PECS. These are bespoke solutions, tailored to provide occupants with the ability to modulate their immediate environment to achieve optimum comfort [12], [48], [49], [50], [51], [52], [53], [54], [55], [56]. The roots of PECS trace back to early research by Bauman and his colleagues in the early '90s [57]. Since then, the area has been extensively studied, resulting in more than 200 scholarly articles underscoring the efficacy of PECS. For an exhaustive overview, one can refer to our review paper titled "*A review of advances for thermal and visual comfort controls in personal environmental control (PEC) systems*" [12] which rigorously surveys the advances in thermal and visual comfort facets of PECS which has cited 124 references. Additionally, more recent review like "*Personal comfort systems: A review on comfort, energy, and economics*" [51] and "*Thermal comfort and energy performance of personal comfort systems (PCS): A systematic review and meta-analysis*" [58] further accentuate the continuous evolution and the effectiveness of these systems, citing 184 and 103 references respectively. Established methodologies for providing energy-efficient thermal and lighting solutions indicate that personalised comfort and lighting systems - incorporating intelligent control sensors and integrated with natural elements like outdoor air and daylight - can decrease energy usage while enhancing comfort, and the challenge remains on the integration and controls aspects.

PECS capitalize on the understanding that the parameters for ensuring human health are broader than the parameters for natural comfort. Manipulating conditions to make an individual feel comfortable beyond their typical comfort range but still within a healthy range poses no harmful effects [59], [60]. These systems mirror cybernetic principles[4]. Lichtenbelt et al. highlighted the beneficial impacts of varying environmental temperatures, the built environment, and health on the human energy equilibrium, advocating for occasional ventures outside the typical comfort zone [61]. Much of PECS research concentrates on thermal and visual comfort, areas ripe for innovation, especially given the diverse individual preferences and needs [57].

Reference models representing standard HVAC and building design practice were used to simulate the impact of thermostat setpoint ranges on annual HVAC energy consumption [60]. Raising the cooling setpoint from 22.2°C (72°F) to 25°C (77°F) results in an average cooling energy savings of 29% and an overall HVAC energy reduction of 27%, without compromising user satisfaction [60] and accepted by 80% to 90% of the occupants in the building[62]. Expanding temperature ranges, using methods like fans or individual controls, can lead to HVAC energy savings between 32% and 73%, contingent on regional climate conditions [60], as illustrated in Figure 2-2.

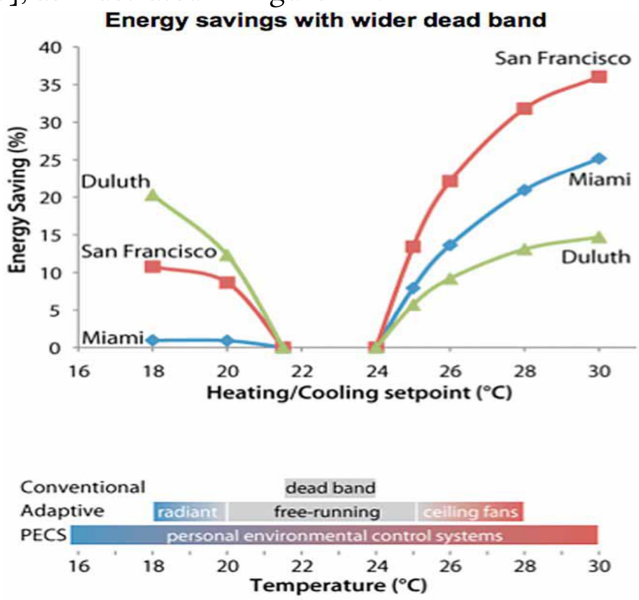


Figure 2-2: Percent energy savings for widened air temperature setpoints relative to conventional range © 2015 Elsevier, reprinted with permission from [80]

[4] "Norbert Wiener defined cybernetics in 1948 as 'the scientific study of control and communication in the animal and the machine.' In the 21st century, the term is often used to imply 'control of any system using technology'. In other words, it is the scientific study of how humans, animals, and machines control and communicate with each other." - Wikipedia

Subsequent research in the field has been extensive. For instance, [63] explored the effects of isothermal airflows on individuals seated in chairs equipped with two fans: one beneath the seat and another behind the backrest. [64] evaluated the reactions of 48 participants in room temperatures of 20°C, 22°C, and 26°C. Another study [65] assessed the efficacy of personalized ventilation systems with headrest-mounted air terminals. A further investigation by [66] gauged the reactions of 24 participants to localized convective cooling, radiant cooling, and a combination of the two at a condition of 28°C with 50% relative humidity. [67] also examined the advantages of modulating airflow interactions in micro-environments. The existing literature of IEQ and its impact on occupant comfort and productivity advocate study of forms of engagement taken by building inhabitants to handle the environment and to modify it to their comfort [68], [69], [70], [71]. Several studies have been published on the relationship of personal control, comfort and productivity of users [39], [57], [67], [69], [70], [72], [73], [74], [75], [76], [77], [78], [79], [80], [81], [82]. These studies show that PECS influence occupant satisfaction and productivity.

However, while the primary emphasis of most PECS research has centred on enhancing energy efficiency and individual comfort, the integration of PECS into broader building systems needs to be studied. Limited studies, such as [40], have delved into how individual occupants can actively engage in demand response management. Additionally, there is a gap in the literature regarding the interoperability and communication challenges within specific task environments and between task and ambient settings. The current studies further do not elaborate on comfort parameters the user would be willing to forgo, given an option, to align the energy demand restrictions.

2.5 Integration of PECS, DR, and TC

The interplay between Personal Environment Control Systems (PECS), Demand Response (DR), and Transactive Control (TC) presents a promising frontier for a more sustainable, efficient, and personalized energy management paradigm. As illustrated by the extant literature, this interplay, while promising, is still in the nascent stages of exploration.

Several studies [7], [10], [11], [33] have emphasized incorporating innovative technologies to integrate smart buildings with smart grids seamlessly. However, only [40] delved into integrating PECS with demand response management. In this study, a PECS chair was showcased (as seen in Figure 2-3), along with the introduction of a micro-zone-attuned building control system (represented in Figure 2-4).

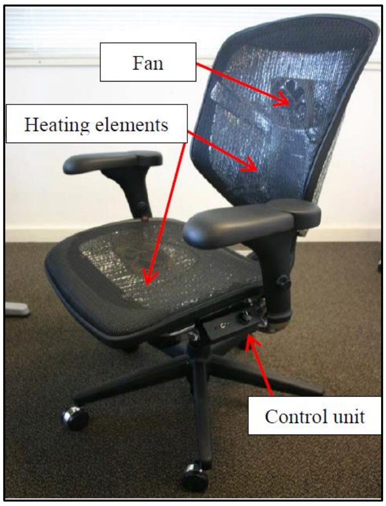


Figure 2-3: Mesh PCS chair
Source from [40]

The research also presented an array of control algorithms tailored for comfort-centric setpoint adjustments and lucid demand response mechanisms. A unique "*microzone-centric*" approach was proposed, facilitating the melding of

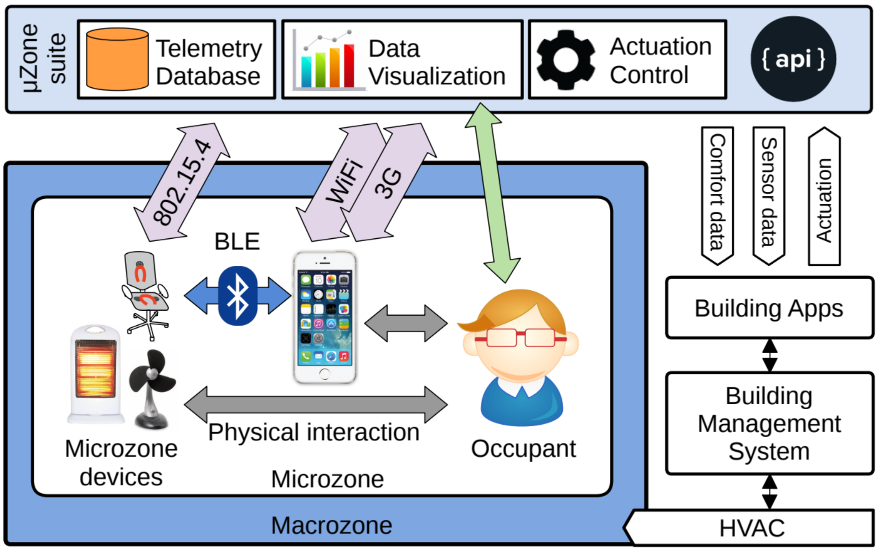


Figure 2-4: Architecture of a microzone-aware building control system
Source from [40]

PECS into a building's control framework. In this structure, users directly interface with their immediate environment devices, retaining full command. From these individualized settings, broader comfort metrics are deduced. Using chair telemetry, occupant comfort was evaluated in real-time, subsequently influencing the modulation of HVAC systems. The Centre for the Built Environment (CBE) at the University of California, Berkeley, envision creating a new occupant based paradigm for HVAC control, integrating low energy PECS into HVAC operations [83].

In summary, there is a noticeable absence of research centred on applying transactive control in demand response through personal environment control systems (PECS) that actively encompass the end-user. There is a need for a system that not only actively involves the user in demand response but also optimises personal objectives and a collective global optimisation aim.

2.6 Identified Gaps

Integrating Personal Environmental Control Systems (PECS), demand response management, and transactive control represents a transformative step for sustainable and efficient energy management in smart buildings. However, as identified, notable gaps in the literature hinder this fusion. Here is an in-depth exploration of these gaps:

1. **Lack of PECS Integration Within Task Environment & Between Task & Ambient:** PECS are inherently designed for individualized comfort. Their integration at the task level (immediate workspace) and connection with the ambient (broader environment) is not straightforward. Without this seamless integration, achieving holistic demand response and user comfort is challenging.

2. **Lack of monitoring and automatic control of devices connected at task level:** PECS and other connected devices create an intricate web of energy loads. Understanding these connected loads' demands, consumption patterns, and interdependencies is essential for efficient management. Insufficient understanding can lead to sub-optimal energy utilization, redundant energy expenditure, and inefficacies in demand response.
3. **Lack of unified User Interface with Actionable Information:** As systems become increasingly sophisticated, presenting users with comprehensible, actionable data is crucial. No matter how advanced, a system is of little use if its users cannot interact with it effectively. An ineffective user interface might lead to reduced user participation in demand response initiatives or misuse of PECS, negating potential energy demand reduction.
4. **Lack of integration of PECS with Transactive Control for Demand Response:** Transactive control, at its core, aims to make energy transactions (buying/selling) efficient by considering real-time prices and grid demands. Integrating this with PECS requires a bridge between individualized comfort preferences and broader grid dynamics. Without this integration, the potential of PECS to contribute to grid stability and efficiency remains untapped. It also means users miss potential cost savings from real-time energy transactions.

In summary, the identified gaps present both challenges and opportunities. Addressing them is essential for achieving energy efficiency and realizing the vision of truly smart buildings that prioritize user comfort and grid sustainability.

Chapter 3

iSPACE

intelligent System for Personal Ambient Control and energy Efficiency

"The greatest good for the greatest number" – Philosophy of Utilitarianism, Jeremy Bentham

3.1 Introduction

As discussed in section 1.4.1, this chapter elaborates on the system architecture and framework. It provides detailed insights into the system's features, conceptualization, mathematical modelling, and pricing formulation. The focal point of this thesis is the implementation of transactive control within the building environment, facilitating interactions between energy-consuming devices and the building systems at a localized level. While traditional supply-demand balancing occurs at the grid level by utilities or balancing authorities, our research explores the efficacy of implementing transactive control within buildings to optimize energy usage and enables end-users to participate in managing DR events at the task level based on their priorities.

The primary contribution of this chapter is the design and development of iSPACE - intelligent System for Personal-Ambient Control and Energy Efficiency, a generalised hierarchical distributed multi-agent system architecture and framework. It aims to address the identified gaps outlined in section 2.6. iSPACE empowers individual occupants to manage energy usage at a granular level, integrating Personal Environmental Control Systems (PECS) devices such as SmartHub, RadiantCubicle, and SmartStrip seamlessly into building energy management systems.

The system model and pricing formulation are adapted from the works of [27], and [26]. While [26] developed it for smart grids and [27] for PID controls in buildings, we extended it to integrate PECS and enhanced it for hierarchical distributed systems. Additionally, we introduced the innovative construction of Price Function (pf) and Energy Demand Function (edf) for the demand side, substituting utility functions for energy-consuming devices. These functions, a novel aspect of this thesis, are precomputed and adhere to comfort parameter constraints.

Furthermore, we identified Eclipse VOLTTRON™ as a suitable platform for implementing the framework. The integration of PECS within tasks, providing a unified interface, and connecting them with ambient control systems are discussed. Additionally, the hardware and software developed are detailed in the subsequent chapter, followed by a chapter on system evaluation encompassing advantages, challenges, scalability, limitations, and recommendations. Besides, the simulation study results about convergence rates, validating the effectiveness of our system.

3.2 iSPACE system

As shown in Figure 3-1, a typical building would consist of multiple zones. Each zone would serve different purposes. Some zones would be densely occupied (example open-plan offices) and some would be sparsely occupied (example reception offices) and some rarely occupied (example server rooms or storerooms). Moreover, the occupants would have varying needs. Similarly, each occupant would be using a banquet of different PECSs with different objectives to customise the task environment.

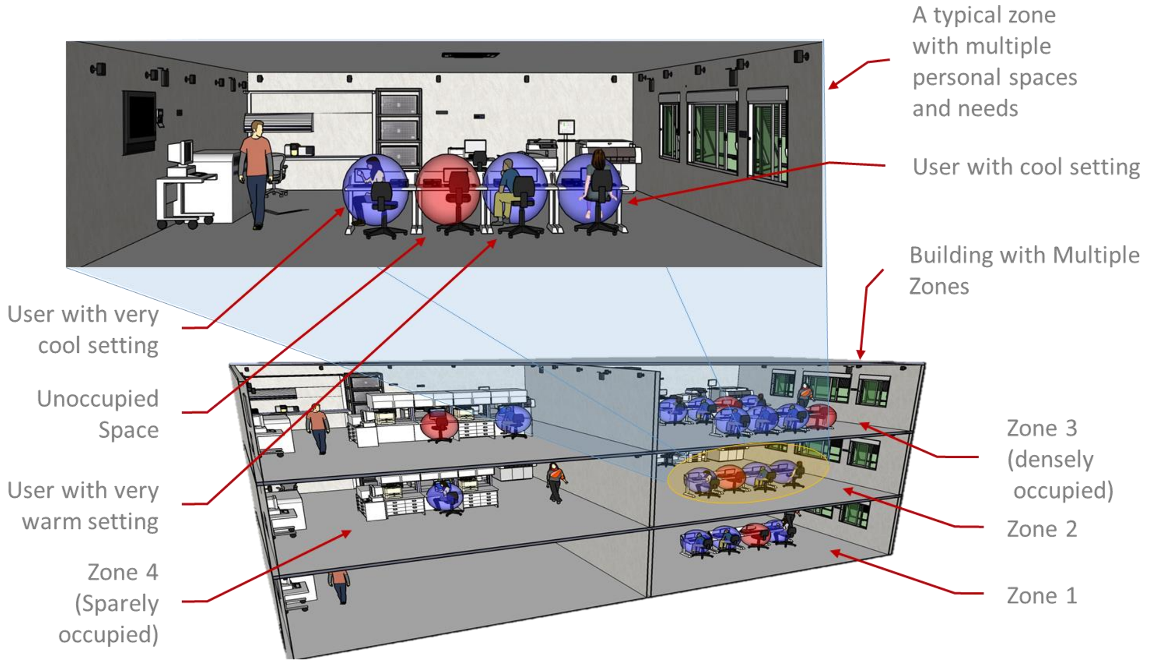


Figure 3-1: iSPACE - intelligent System for Personal-Ambient Control and Energy Efficiency

3.2.1 The Approach

We envision a system that can personalise the task environment to the needs of individual users and micromanage the energy requirements at the task level. It integrates a diverse range of Personal Environmental Control Systems (PECS) with ambient controls in a cohesive manner, enabling real-time consideration of user preferences during demand response events.

The system has two primary features:

1. Energy using devices automatically adjust their usage according to a local power price point that reflects the scarcity of power, and
2. A unified way to interact with the task environment allows occupants to adjust their local conditions to suit their personal needs.

To address the previously mentioned problems, as part of this research, concepts of transactive controls (refer to section 1.1.2) has been extended to the task level.

A market-based control paradigm using "*price*" as the key operational parameter is used to integrate the PECSs available within the task and with the ambient control system. A US patent was granted [84]. The system and apparatus for and methods of control in both the US patent and India patent are the ones that are addressed in this thesis. Claims are mentioned in the appendix (A-1 Patent claims).

Dual decomposition computing algorithms can mathematically explain the approach, and the "*prices*" are dual variables that reflect the equilibrium [5]. The system uses a Stackelberg game class of sequential game theory. A Stackelberg game is a two-stage problem where the leader (a co-ordinator agent) makes its decision in the first stage. And the followers observe the leader and act upon the leader's decision to optimise its objectives in the second stage [5].

The agents are discussed in detail in section 3.3 and its subsections. The system model and pricing formulation used for the case study implementation of the system and implemented as default optimisation algorithm in the framework is presented in section 3.4. In the default pricing formulation, a *non-cooperative game theoretical framework* (self-enforcing agreements) models the problem in a distributed environment. Based on the game-theoretical approach, the hierarchical distributed algorithm is determined. Minimization of the aggregate energy demand and hence the total energy cost is achieved while constrained by maximizing the aggregate utility of the energy-consuming devices. The framework facilitates support for more complex algorithms. These algorithms can be implemented as external agents or cloud services if required.

In the system, there is an energy provider (upstream Price Controller Agent, (PCA)), there are several energy-consuming devices (PECS, HAVC systems, Lighting Systems, and such others), and coordinator agents (local Price Controller Agents (PCA)). Each or group of energy-consuming devices are associated with Price Agent/s (PA). A PA for each energy-consuming device computes a new setpoint that operates the device optimally. Once the PA adjusts the setpoint, a Device controller Agent (DCA), i.e., a traditional controller, takes over. The PAs also coordinate with local PCA to receive prices and respond with energy demand bids. The local PCA receives a price from the upstream PCA at each level. Using an iterative process, the local PCA, in coordination with the PAs at that level, computes a price that minimizes its energy demand (operational cost) while constrained by comfort parameters.

Furthermore, each individual PAs computes an energy demand bid that maximizes its welfare for a given price. The local PCAs collates all the energy bids of the associated energy-consuming devices and respond to upstream PCA with its total energy demand bid. An agreement is reached between PCA, local PCA, and PAs after specific iterations based on exit criteria.

3.2.2 Systems Functional Hierarchy

The functional hierarchy of the system is as shown in Figure 3-2. The building level controller receives a budget or a price point from the grid energy provider during a demand response event. The building level controller then publishes new budgets or price points to all the subscribed zones to operate the associated processes in an economically optimal way. A similar iteration is done at the zone controller and the SmartHub (task level).

At the task level, a Personal Environment Network (PEN) is formed, in which the SmartHub

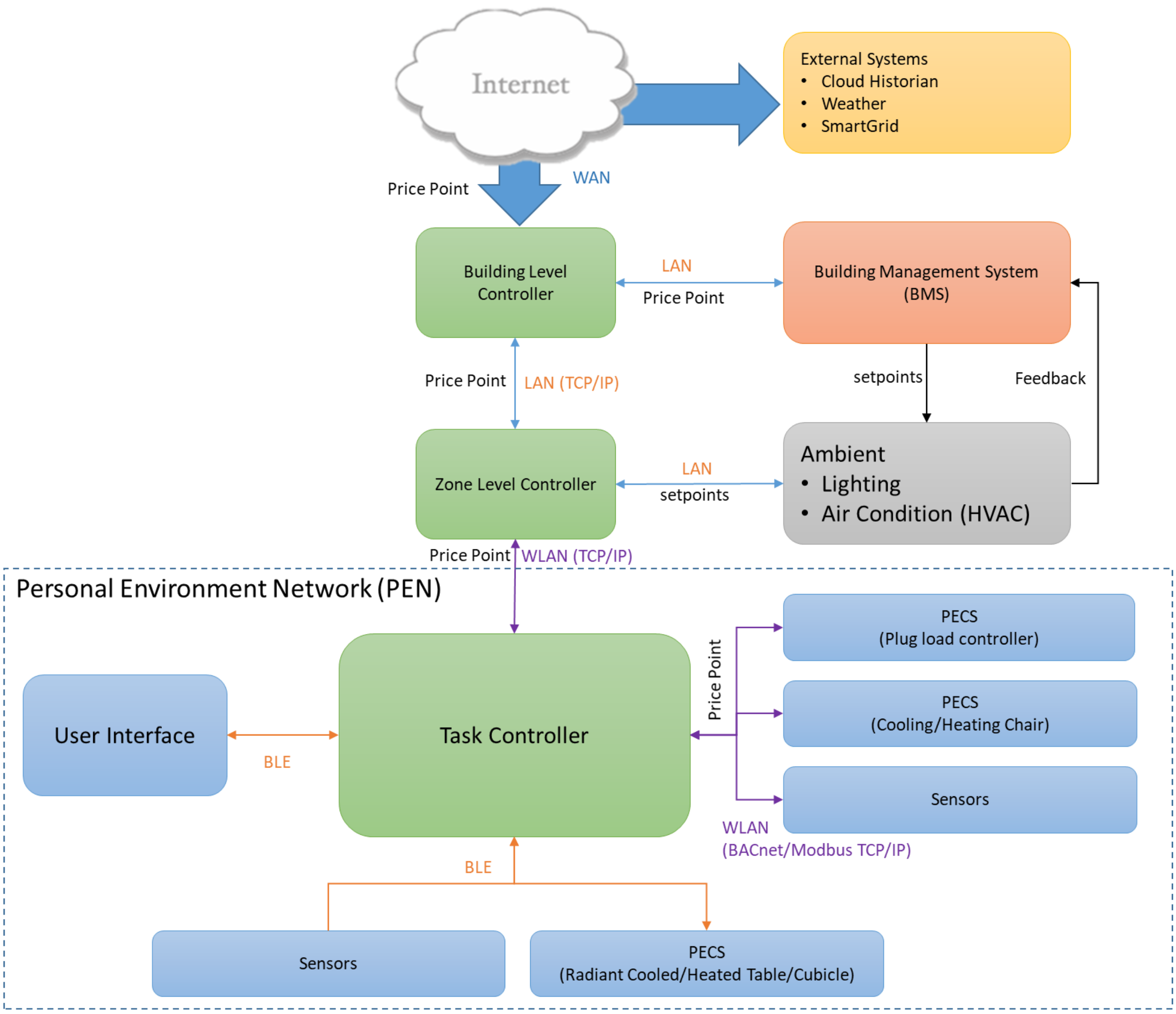


Figure 3-2: Functional hierarchy of the iSPACE with transactive energy for demand response

acts as the central coordinator for the with-in task devices like PECS, sensors, and so forth. The PECS and sensors in the figure are representational only, and the system needs to support many other PECS and sensors with diverse communication protocols. Central to the iSPACE system is a SmartHub (Figure 3-3), and typical task level components of the system are as shown in Figure 3-4. There are three key aspects to the SmartHub.

1. A PECS (Task Fan & Task Light),
2. Co-ordinator for the task level PECSs,
3. Interface with the Occupant – capture user inputs (like preferences and configurations parameter) and provide meaningful alerts and feedback to the occupant based on local sensors data and the trends based on historical data.

Figure 3-3: iSPACE with a SmartHub, a SmartStrip, and a RadiantCubicle

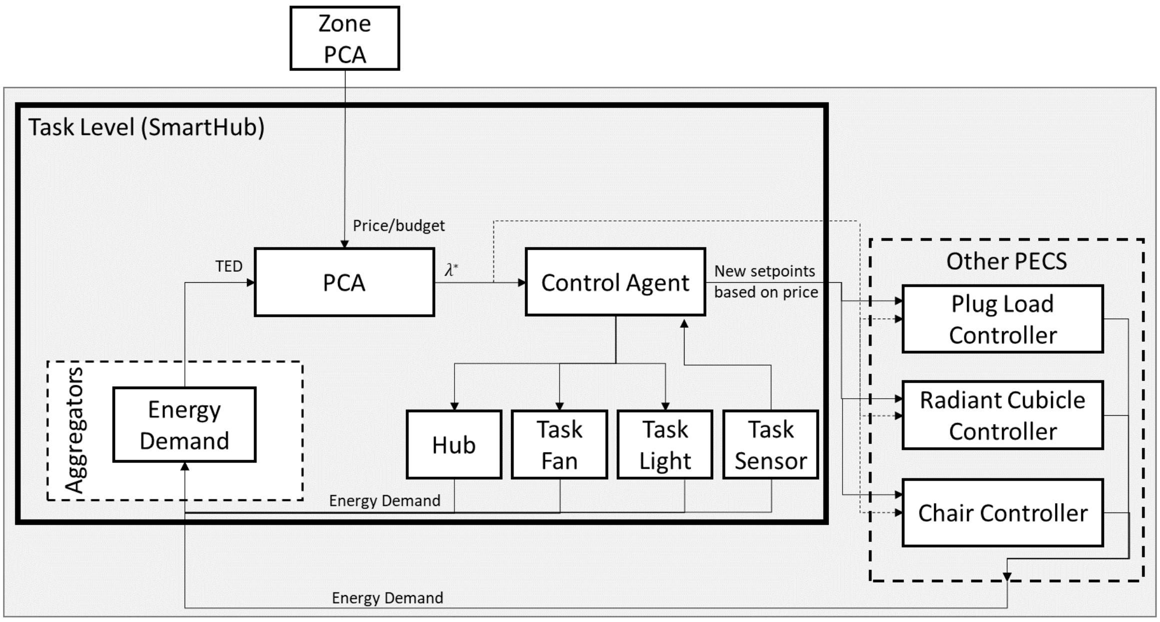


Figure 3-4: Task level system components

Legend

λ^* → Price/Budget
TED → Total Energy Demand
PCA → Price Control Agent

The SmartHub is a controller with a mobile user interface at the task level that allows occupants to participate in the demand response. The SmartHub also facilitates integrating various PECS and sensors available within the task environment with the ambient control systems. For this purpose, the necessary device discovery and registration mechanism has been developed. The SmartHub would discover various with-in task devices (PECS & sensors) and maintain a register with the device features and capabilities. Also, users can interact with the registered devices through the SmartHub interface and manage the conditioning at the task in a unified way.

The user interface to SmartHub is provided through a mobile application that communicates over Bluetooth Low Energy (BLE)[5] for real-time control, alerts, and data visualisation. Besides, if required, the mobile app can be extended to display historical data over a Wi-Fi data connection.

Each user is provided with a SmartHub to which all the PECSs that are available within the task associate,

1. In the SmartHub, a price control agent operates all the associated PECS economically optimal.
2. The information available to the SmartHub price control agent is the current price from the zone and the utility functions[6] of the associated PECSs.
3. SmartHub computes a new price point or budget that maximises user welfare based on this information.

---

[5] Bluetooth Low Energy (BLE) is a wireless communication technology designed for short-range communication between devices, typically within 10 meters or less. BLE is an energy-efficient version of the classic Bluetooth technology optimized for low power consumption. It is ideal for small, battery-powered devices such as wearable gadgets, sensors, and smart home devices. It enables these devices to transmit and receive small bursts of data while consuming minimal power, extending battery life significantly compared to traditional Bluetooth connections.

[6] A utility function describes the degree of well-being the product provides for consumers, that is, it defines different responses to various prices. Refer to section 1.1.2 for details.

4. This new price point or budget is published to all the associated PECS to support transactive control. If not, the control agent in the SmartHub can give the setpoints if PECS publishes prior energy demand curves to SmartHub.
5. Once the setpoint is communicated, the device controller in the respective PECS control to achieve the desired setpoint, and
6. Such a transactive control process continues.

3.2.3 System Architecture

Figure 3-5 shows the system architecture and the data flow between different layers. The right side of the centre dashed line is the existing infrastructure in a typical building. The left side of it consists of the system at task level.

A hierarchical distributed multi-agent system (shown in Figure 3-6) with various controls in the building divided into three levels vis-à-vis building level, zone level, and task level has been used. At the building level, the system consists of energy resources like utility providers, locally available distributed resources like solar PV systems, wind farms, diesel generators, and central control systems like building management systems (BMS). The systems consist of the HVAC system for ambient cooling/heating, ambient lighting, and ambient sensing sensors at the zone level. At the task level, the system consists of a SmartHub for task controller that integrates various other PECS (including a SmartStrip for plug load control) and a mobile device for the user interface. Various entities are represented by their respective agents in the system and communicate over a message bus coordinated through communication infrastructure. The system consists of energy provider agents (PCA), agents that act on the price (Price Agents) for each energy-consuming device, traditional device control agents (DCA). The system incorporates a mixture of object- and service-based agents.

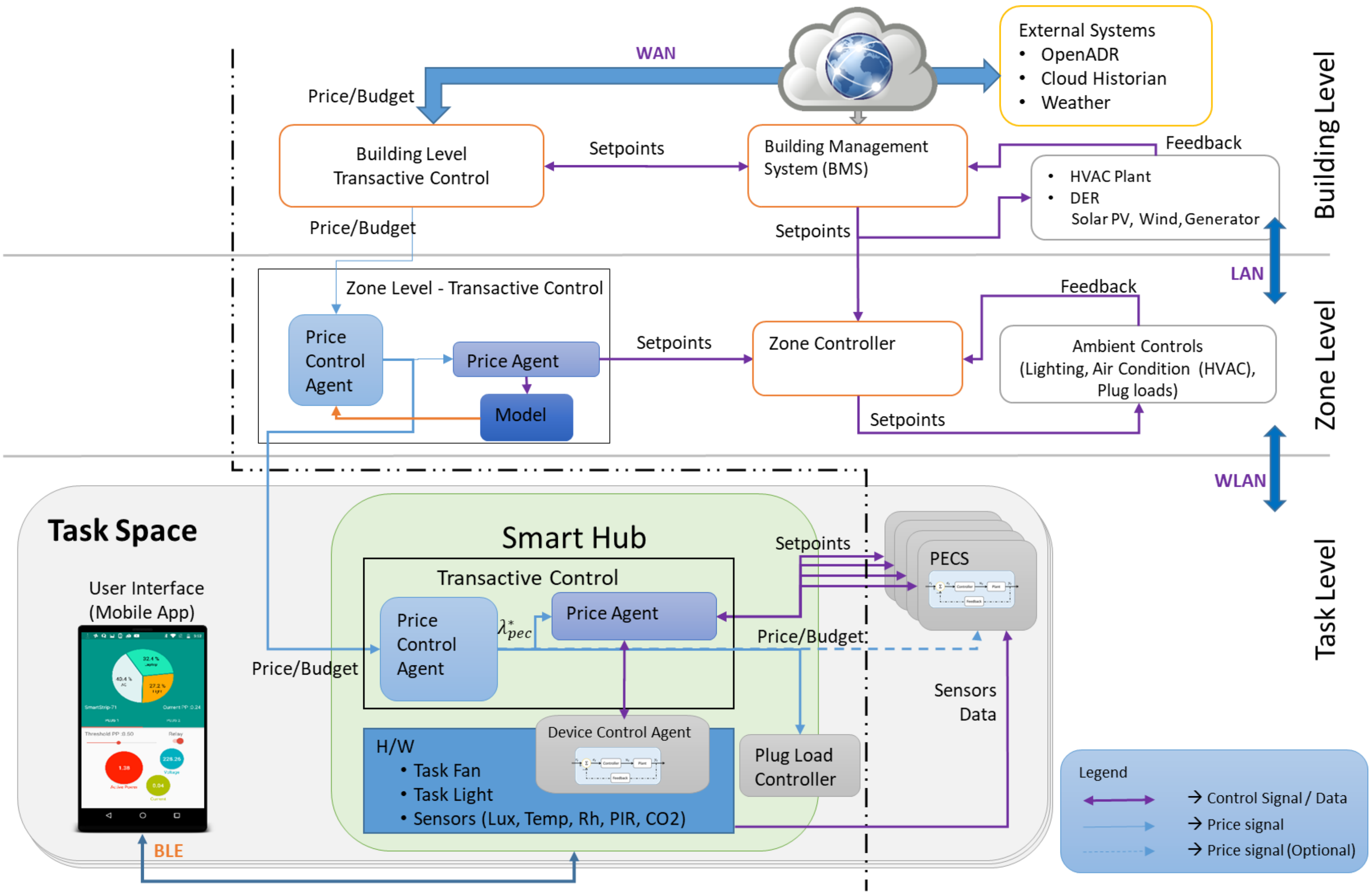


Figure 3-5: System architecture

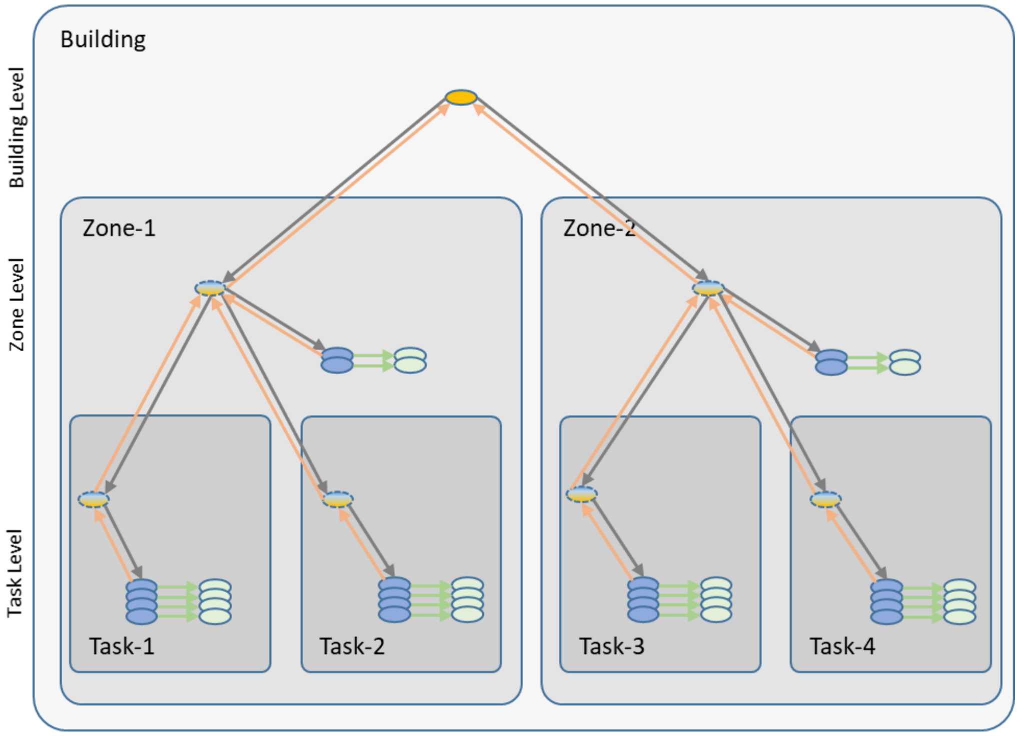
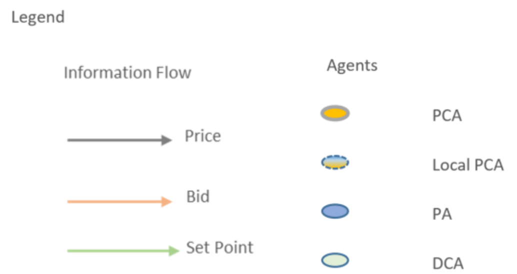


Figure 3-6: iSPACE Hierarchical Distributed Multi-Agents System architecture



The Agents are housed in a multi-agent system (MAS) platform that incorporates MAS management infrastructure (refer to section 3.5). The various agents can be classified as – energy provider agents (producers like smart grid, DER, upstream PCAs), consumer agents (HVAC, lighting, PECS, (PA + DCA)), auctioneer agents (PCAs, local PCAs). The local PCAs have a dual role. They function as energy-consumer agents for the upstream level and as a coordinator agent for the current and downstream energy-consuming device agents. The price information flows from the upstream PCA to the associated PAs and the downstream local PCAs. Similarly, the bid information flows from downstream local PCAs and the local PAs to the upstream PCA. For brevity, supporting agents like Bridge agents, Actuator agents, and other supporting agents are not represented in the figure.

For one or a group of energy-consuming devices, we assume that a price control agent tries to operate the process associated with that device in an economically optimal way and coordinated with the energy provider through a communication infrastructure. Moreover, a price agent for each device determines the consumption setpoint that meets the demand. The new budgets or price points can be computed based on numerous strategies. For example, a simple rule-based engine with a set of business rules would be used wherein the rules would be as simple as some linear or ramp function of the budget/price or can be as complex as random decision forests. Alternatively, the strategy would use complex optimisation techniques such as welfare maximisation. The system model and pricing formulation adapted from [26] and [27] for the case study implementation and implemented as the default option is presented in section 3.4. The system facilitates support for more complex algorithms. These algorithms can be implemented as external agents or cloud services if required.

More details about these agents are provided in sub section 3.3.2 and the details about the information exchanges described in sub section 3.3.1.

As mentioned previously, central to the iSPACE system is a SmartHub (Figure 3-3). The following subsection describes the architecture for the SmartHub and the SmartStrip (a plug load controller).

A. SmartHub Architecture

SmartHub is a vital component in the iSPACE system. It acts as a nexus point, coordinating and interacting with various entities in the system. The architectural framework for SmartHub is stratified into three layers (Figure 3-7):

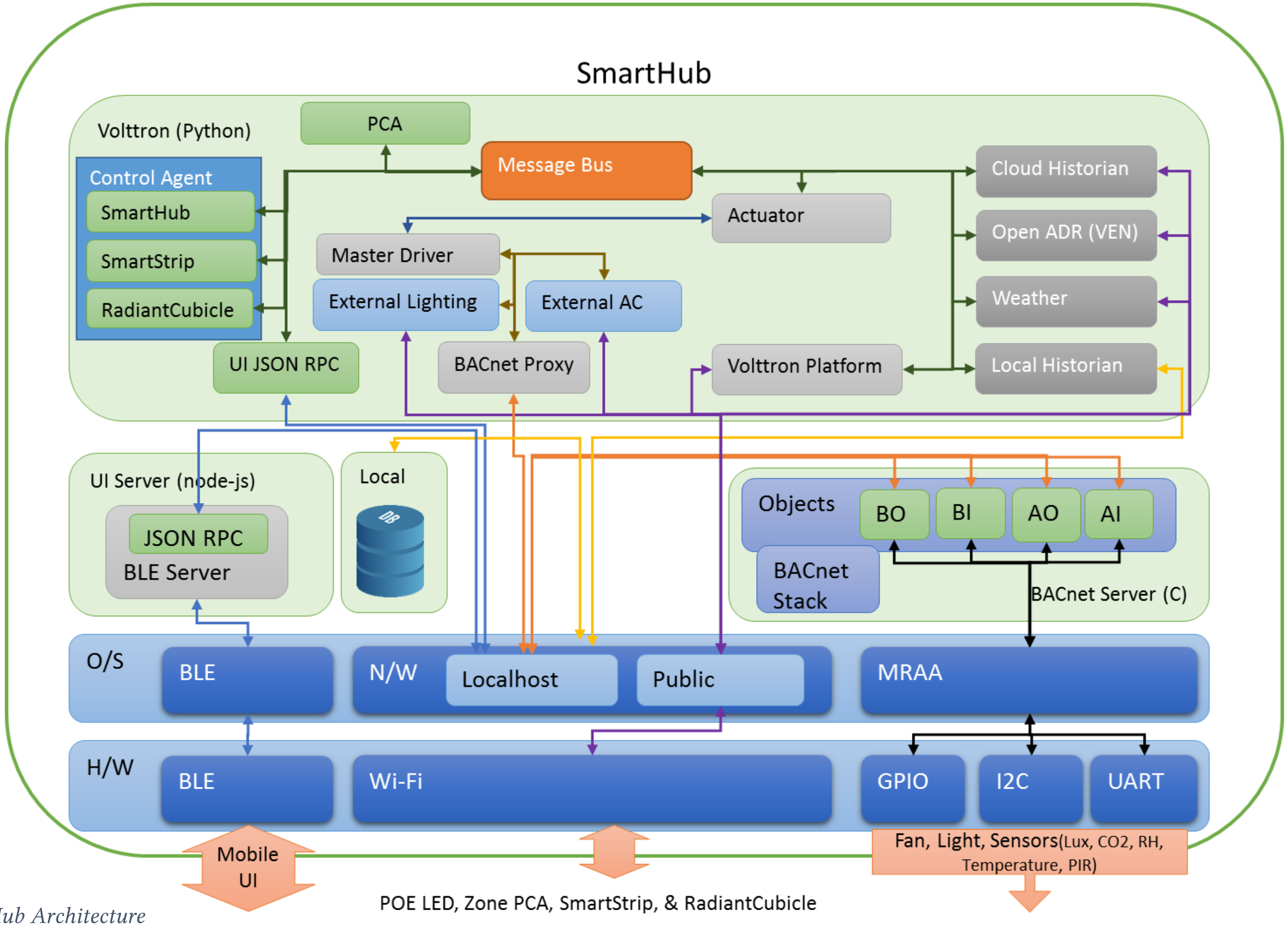


Figure 3-7: SmartHub Architecture

1. **Hardware and Interface Layer (Bottom Layer):**

   - **Main Components**:
     - **I/O Breakout Board:** This is essential for interfacing with the hardware components, aiding in input/output functions.
     - **BACnet Server:** It facilitates access to the hardware components by representing them as BACnet objects over the TCP/IP protocol, which is a standard communication protocol for building automation and control systems.
   - **Functionality:**
     - This foundational layer primarily deals with the raw hardware and acts as a bridge between the physical components and the software elements.
     - By operating in this manner, it abstracts away the dependency on specific computer modules or a specific Transactive platform, making the system more adaptable to varying hardware configurations.

2. **Transactive Platform (Middle Layer):**

   - **Main Components:**
     - **Agents:** They are the key players in this layer, particularly the price control agent and control agent.
   - **Functionality:**
     - Agents in this layer engage in data interchange using JSON, a lightweight data-interchange format.
     - They can provide directives to locally available PECS (like

SmartStrip, RadiantCubicle) based on prior energy demand patterns that are routed to the SmartHub.

3. **Presentation Layer (Top Layer):**

- **Main Components:**
    - **UI Gateway:** This component allows for communication with mobile applications over Bluetooth Low Energy (BLE).
- **Functionality:**
    - As the name suggests, the presentation layer is more user-facing. It's the layer users directly interact with, often without realizing the intricate processes occurring in the layers below.
    - It supports bidirectional message transfer, meaning it can both send (write) and receive (read) data.
    - Like the middle layer, the data interchange occurs using the JSON format, ensuring uniformity across the system, and facilitating easier data parsing and manipulation.

The systematic layering of the architecture ensures that each component of the SmartHub has a defined role and function. This modularity makes it easier for future expansions, troubleshooting, and adaptability to various use cases. The use of standard protocols and formats (like BACnet, TCP/IP, and JSON) ensures compatibility and scalability.

B. SmartStrip Architecture

SmartStrip is another pivotal component within the iSPACE system. While its architecture (Figure 3-8) is simpler than the SmartHub, it still plays a crucial role in controlling plug loads in an intelligent manner.

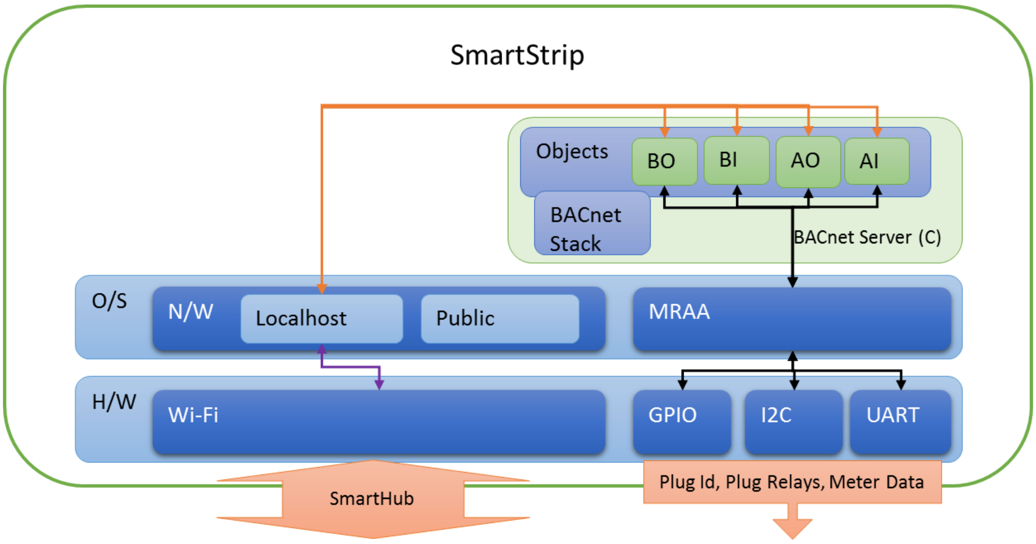


Figure 3-8: SmartStrip Architecture

1. **Hardware and Interface Layer (Bottom Layer):**

- **Main Components:**
  - **I/O Breakout Board:** This component aids in establishing a connection between the physical hardware of the SmartStrip and its software. It assists in both input and output functions, allowing for the seamless transmission and reception of data.
  - **BACnet Server:** Like its role in the SmartHub, the BACnet server in the SmartStrip allows access to the hardware elements by portraying them as BACnet objects over a TCP/IP protocol. BACnet's adoption here aligns the SmartStrip with standard building automation communication norms.

- **Functionality:**
    - As the foundational layer of the SmartStrip, this level is pivotal for the correct operation of the entire architecture. It acts as the primary interface between the raw hardware of the SmartStrip and its software components.
    - This layer offers flexibility by removing dependency on specific computer modules. By doing so, the system remains adaptable to varying hardware configurations and can smoothly integrate with multiple devices without necessitating extensive overhauls.

The architecture of the SmartStrip is streamlined and efficient, primarily focusing on plug load control. The emphasis on using standardized protocols and communication methods (such as BACnet and TCP/IP) ensures that the SmartStrip is both compatible and scalable. Its two-tier architecture ensures that the system remains simple enough for rapid deployment while retaining the sophistication required for intelligent energy management.

3.2.4 Communication

A real-time communication infrastructure that provides connectivity among systems, devices, agents, and applications is essential for the efficient and reliable operations of the system.

For holistic integration of PECSs with-in task and with the ambient controls systems, we need high-quality information (i.e., data that is dependable, complete, and consistent and is high-resolution data). Besides, the ability to effectively share this information among all the stakeholders for a cooperative advantage is a key tenet of any integration. However, to prevent information overload and keep the information as secure as possible, how much information is shared and where it is stored needs to be carefully addressed.

The communication framework consists of two components - Physical Layer Communication, and Application Layer Communication

A. <u>Physical Layer Communication</u>

As shown in Figure 3-5, the communication between the smart grid and building management system of the building is done over Wide Area Network (WAN) to facilitate sending and receiving DR signals from a utility using OpenADR (Open Automated Demand Response Communications Specification). For this research work, a suitable price stream from the literature has been used to evaluate the system. Similarly, the control and reporting messages, as defined in the framework between the price control agents (that is between two distinct levels), would be exchanged over the local area network (LAN) or wireless network (WLAN).

Further, the SmartHub could communicate with various other PECS over different communication channels like BACnet TCP/IP, BLE, and Modbus for interoperability. As part of this research, Bluetooth profiles exchanged between various entities (such as agents, PECS, mobile app and so forth) has been developed. The information exchange between the SmartHub and a mobile app would be through BLE. For this purpose, 3 GATT profile (Bluetooth Generic Attributes) has been developed (detailed in Appendices A-7), though the framework allows for further expansion based on future requirements.

B. Application Layer Communication

The agents in the system would communicate through a message bus using publish/subscribe design pattern. Various messages/data types used as part of the Framework has been identified and are discussed in next sub-section 3.3.1. Besides, the MAS platform, Eclipse VOLTTRON™, has built-in drivers to support communication with physical devices that implement the BACnet or Modbus protocol. These two protocols are widely used in building automation systems. Thus, using message bus and drivers support, interoperability with a broad set of heterogeneous devices is enabled, both at application and physical layers.

3.3 Framework

As discussed in section 3.2.1, the framework uses a Stackelberg game class of sequential game theory. A Stackelberg game is a two-stage problem where the leader, a co-ordinator agent (energy provider), makes its decision in the first stage, and the followers, the energy-consuming device agents, observe the leader and act upon the leader's decision to optimise its objectives in the second stage [5]. [5] studied available tools and results in the literature and presented a unifying framework for transactive controls systems to solve the market-based coordination problem. We extend it to task level as a hierarchical distributed multi-agent system with various controls divided into three levels vis-à-vis building level, zone level, and task level.

Distinct entities of the framework at a particular level (vis-a-vis Building Level or Zone Level or Task Level) are shown in Figure 3-9. Figure 3-10 shows various agents at a particular level. Furthermore, the overall flow of information structure at a particular level is shown in Figure 3-11.

The key components are following and discussed in subsequent subsections:

1. iSPACE Messages
    a. Control Message (CM),
    b. Probe Message (PM), and
    c. Reporting Message (RM).
2. iSPACE Agents
    a. Device Controller Agents (DCA),
    b. Price Agents (PA),
    c. Price Controller Agents (PCA), and
    d. Bridge Agents (BA).

Briefly,

1. In Figure 3-9, the boxed items with continuous lines represent various agents. The dashed line boxes represent various key topics on the message bus. Various message types are defined in the framework. The orange lines (dash type with a single dot) represent the flow of Control Messages and Probe Messages, and the blue line (continues line) represents the flow of Reporting Messages.
2. Various energy-consuming devices available at a particular level associate and coordinated through communication infrastructure and exchange information over a message bus using iSPACE messages.
3. As shown in Figure 3-10, one device acts as the coordinator (running the Price Controller Agent). A device with higher processing power or one that is most active can be assigned as the coordinator. Price Agents of the other energy-consuming devices registers with the coordinator and communicate using iSPACE messages.
4. With the help of the Bridge Agent, the PCA receives various iSPACE messages from the upstream level PCA and the downstream level PCAs.

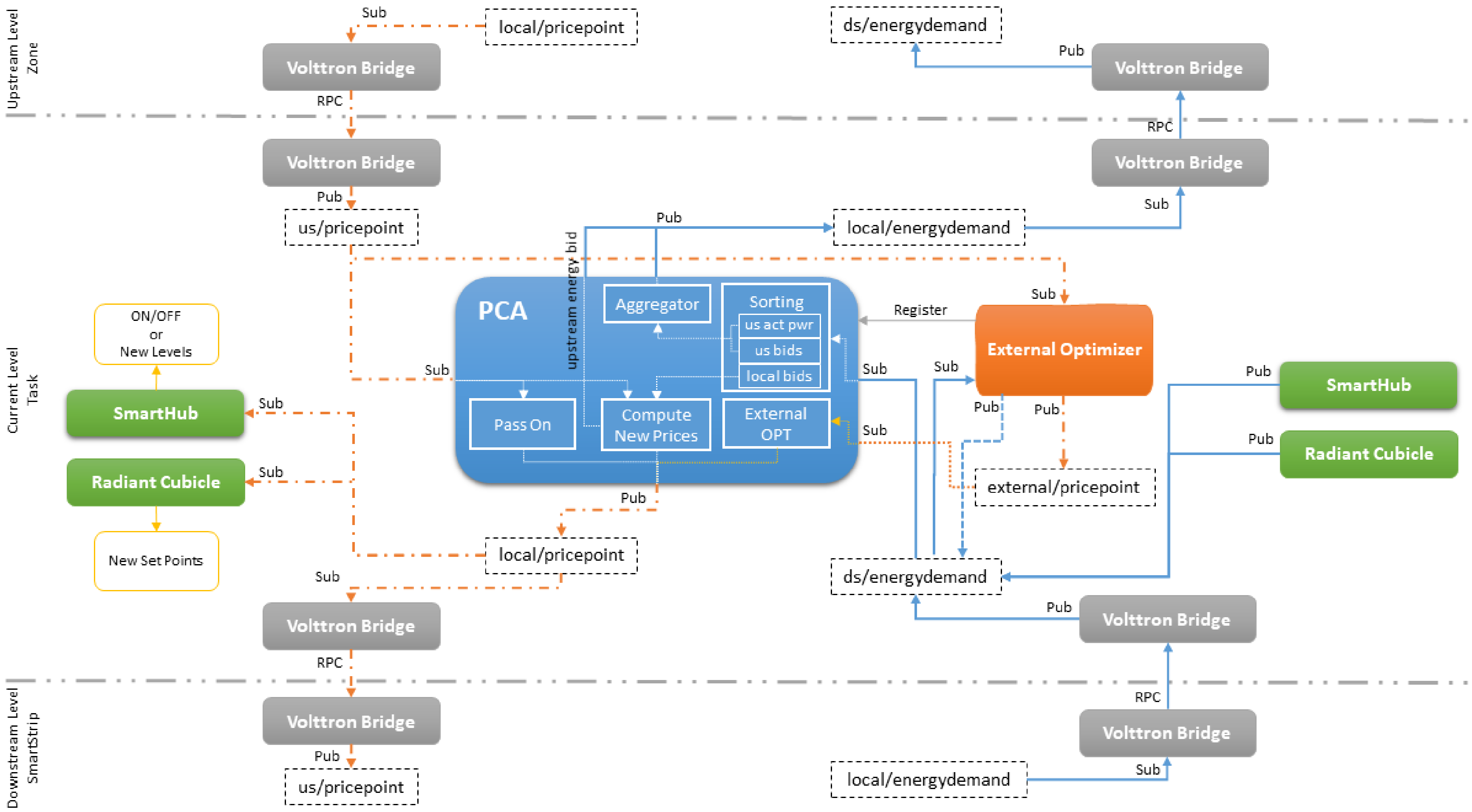


Figure 3-9: iSPACE framework's key components

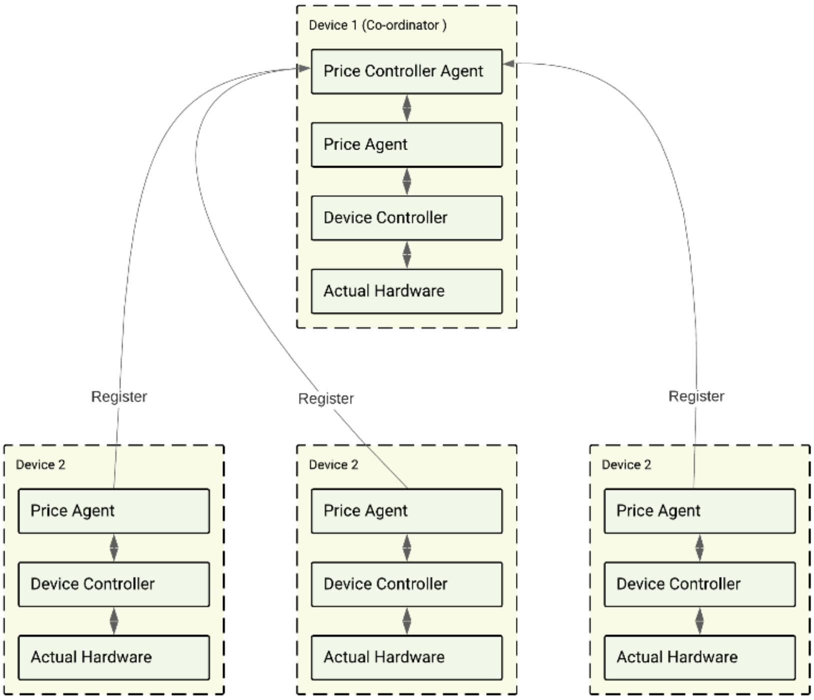


Figure 3-10: iSPACE Agents at a particular level

5. As shown in the flow chart in Figure 3-11, upon receiving a new bid price or a bid budget from the upstream PCA, the PCA initiates a local bidding process and publishes new prices. The information available with the PCA is the energy demand, the current states, and the sensors data of the associated devices.
6. The local PCA can use any energy/comfort optimization strategy in coordination with the PA. Some examples are cost minimization of energy, comfort maximization, a combination of energy and comfort (Multi-Objective Optimization).
7. Each device's Price Agent (PA) computes the new energy demand bids corresponding to the bid price and sends the energy demand bids to the PCA upon receiving the new bid prices.
8. The local bidding process continues until the prices no longer change (or the changes are within a pre-specified threshold). Upon local bidding process termination, the PCA sends the total energy demand to the upstream PCA.

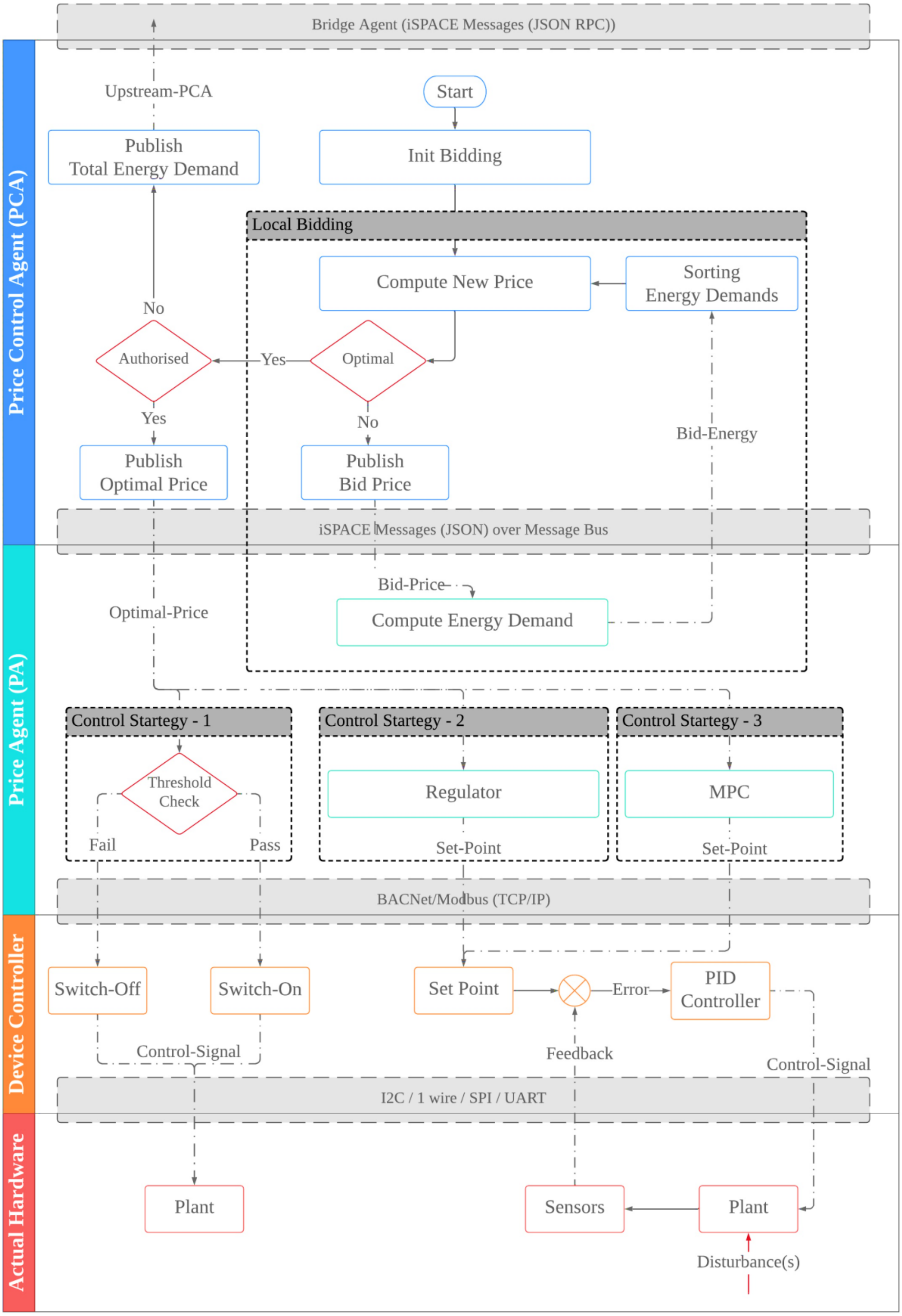


Figure 3-11: Overall flow chart at a particular level

9. The PCA that initiated the bidding in the hierarchy (one authorised to conclude the bidding, primarily the building level PCA) publishes the market clearing price (optimal price).
10. Their respective Device Controller Agents represent each energy-consuming device, and these agents operate the device for designed outputs (example: comfort) for given inputs (example: set-points).
11. Besides, each device has a Price Agent that registers itself with the coordinator (PCA). The price agent receives various iSPACE Control and Probe Messages and responds with Reporting Messages. The price agent can use any energy/comfort optimization strategy. Some examples of such a strategy are Switch on-off based on device-specific price thresholds; or a suitable power consumption regulator that is a function of the price point. Even autonomous agents that adjust the threshold price learned based on user behaviour or priorities can be employed. Alternatively, one can modify existing model predictive control (MPC) models to incorporate price as an additional parameter. Also, the price agents' facilities the energy-consuming device to participate in the bidding process.

Let us illustrate the process with a Zone Level example for better understanding. Consider a zone comprising one central AC, an Ambient Light Controller, and two SmartHubs for two cubicles. Let us assume a one-hour slot and a new price point computation at the slot's start. Suppose the previous market clearing price point was 0.74, with a corresponding total energy consumption of 926 Wh and an energy cost of 685.24 during the previous slot.

Now, assume a price point 0.297 is received from the upstream building level Price Controller Agent (PCA) by the Zone level PCA. To maintain a constant cost (i.e., 685.24), the Zone PCA computes a new target energy demand of 2307 Wh (926 x 0.74 / 0.297).

The Zone PCA then initiates a local bidding process to compute an optimal price point meeting this target for operating the associated energy-consuming devices. Let us assume a computed bid price of 0.67 based on the zone's cost function. This bid price is published to local AC and Light Price Agents (PAs) and the downstream SmartHub's PCAs.

Considering the AC's price and energy demand functions, the demand for the AC at the bid price of 0.67 is 1700 Wh. Similarly, the ambient light energy demand is 100 Wh. Each SmartHub conducts its local bidding process and converges at 259 Wh energy demand. Thus, the Zone PCA computes the total energy demand for the bid price point 0.67 as 2318 Wh (1700 Wh for AC + 100 Wh for Light + 259 Wh for SmartHub1 + 259 Wh for SmartHub2).

As the new bid total energy demand, 2318 Wh, is close to the target energy demand of 2307 Wh, the bid concludes. The bid price of 0.67 is published as the market clearing price (optimal price) to the local AC and Light PAs and the downstream SmartHub's PCA. Finally, the corresponding set points are computed from the respective price functions published to the DCAs.

3.3.1 iSPACE Messages

iSPACE messages have been defined for effective communication of data between various agents. The data format used for these messages is JSON. Various agents exchange data using these messages over the message bus. The agent post to a particular topic on the message bus, and agents interested in a particular message subscribe to that topic.

These messages have been categorised into 1) Control Message, 2) Probe Message, and 3) Reporting Message based on the intended functionality of the message. Similarly, the messages have been divided into four types based on the data contained in the message.

The four message types are:

1) Price Message,
2) Budget Message,
3) Energy Message, and
4) Active Power Message.

The Price Message and Budget Messages can further be sub-divided into two subtypes – 1) One with the value corresponding to optimal condition and 2) The other with the value corresponding with the bid condition. The categories and types of the messages are as shown in Figure 3-12 and summarised in Table 3-1, and the functionality is explained in subsequent sub-sections. The "Optimal" column in the table denotes whether the message's value field is optimal. If optimal, the value represents the market clearing price/budget derived through the system model and pricing formulation methodology detailed in section 3.4. If this parameter of the message is true, then the messages is a Control Message category, and it impacts in the state change in the device. Otherwise, the message would be a Probe Message or Reporting Message.

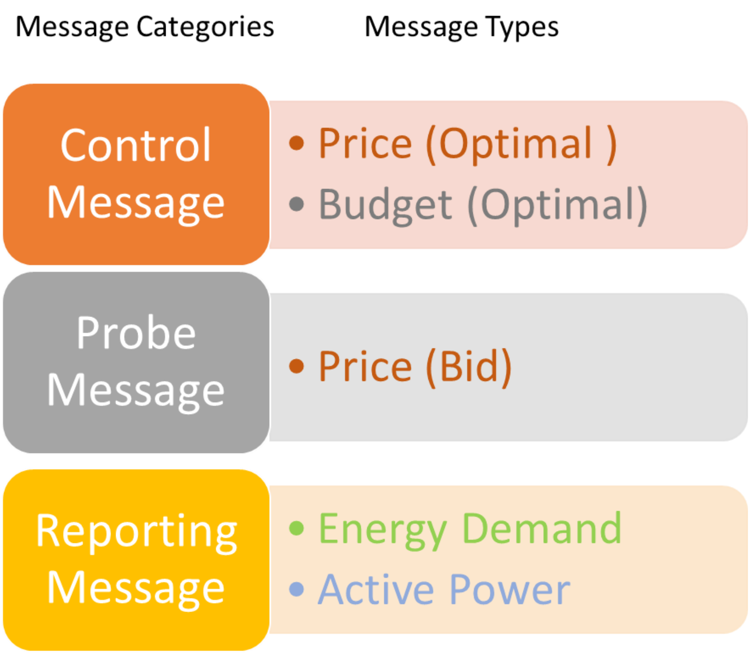


Figure 3-12: iSPACE Messages (Categories/Types)

Table 3-1: iSPACE Messages

Sl. No.	Category	Message Type	Optimal	Description
1	Control Message	Price	True	This message contains the optimal price point and impacts the state change in the devices.
		Budget	True	This message contains the optimal budget and impacts the state change in the devices
2	Probe Message	Price	False	This message contains the bid price point and does not change the device state.
		Budget	False	This message contains the bid budget intended for PCA and impacts in a state change in devices associated with the PCA
3	Reporting Message	Energy	False	This message contains the total energy demand corresponding to a bid price point.
		Active Power	True	This message contains the active power corresponding to the latest optimal price point

A. <u>Message Category</u>

Based on the intended functionality of the message in the iSPACE system, the messages have been categorised into three categories - 1) Control Message, 2) Probe Message, and 3) Reporting Message. The Control Message (CM) and Probe Message (PM) flow upstream to downstream. At the same time, the Reporting Message (RM) flow from downstream to upstream. The CMs are either optimal price points or the optimal budget, and the agents act upon adjusting their states. The end devices report their active power using RM corresponding to the latest CM at the regular interval. Unlike CMs, the PMs are either bid price point or bid budget. These messages trigger routines that compute expected/predicted energy demand for a corresponding PM, and the computed energy demand is published using an RM.

B. Message Types

All the messages inherit from root class iSPACE_Msg, and the inheritance class diagram of various message types is as shown in Figure 3-13. Whereas Appendices A-2 lists and details all the parameters of each message.

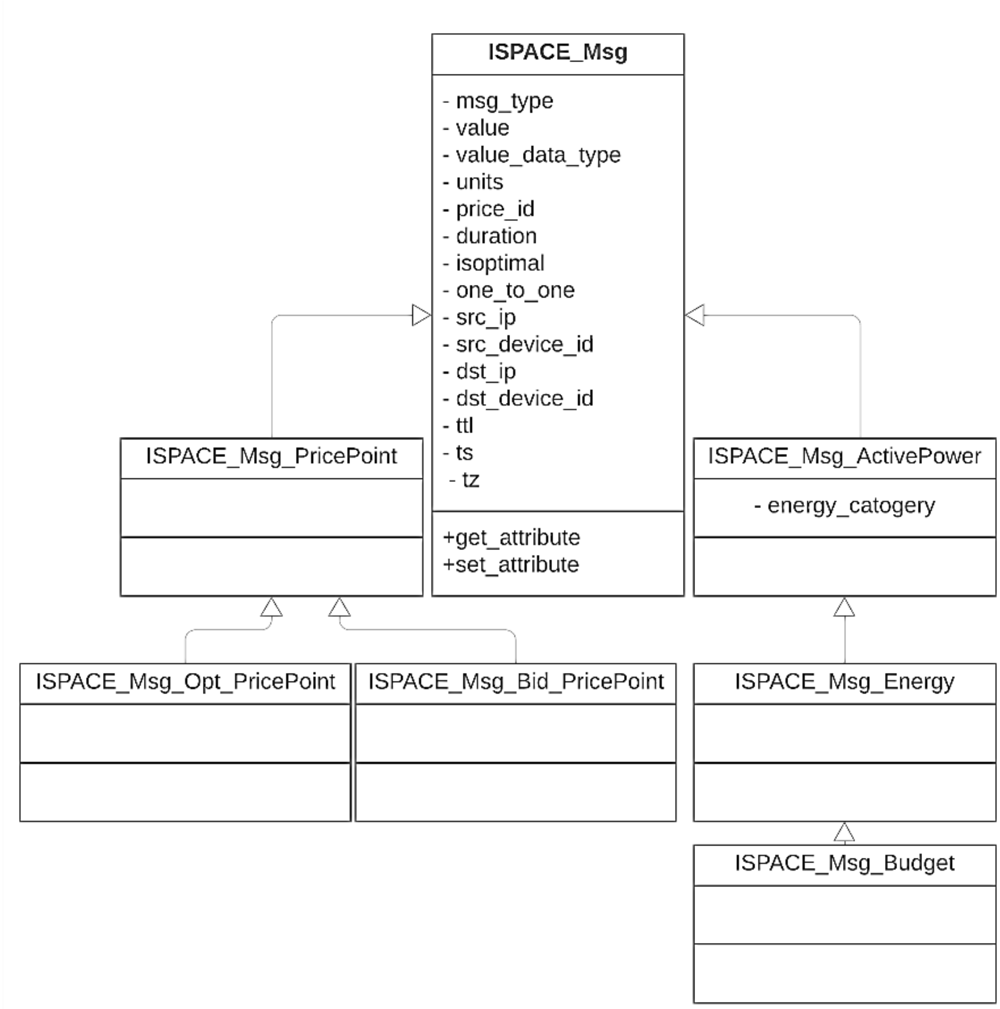


Figure 3-13: iSPACE Message inheritance class diagram

Various Message types are as follows:

a. Optimal Price Point Messages

Optimal Price Point Messages are Control Messages and contain the optimal price point. On receiving this message, respective agents act upon it by applying a pricing policy accordingly.

b. Optimal Budget Messages

Optimal Budget Messages are Control Messages and contain the optimal budget. On receiving this message, the Price Controller Agent redistributes the budget according to the corresponding device's weightage and publishes the new individual budgets to the devices associated with the PCA. The end devices compute an optimal price corresponding to this budget and apply a pricing policy.

c. Bid Price Point Messages

Bid Price Point Messages are Probe Messages and contain a bid price. The intent is that all the participating devices respond with their corresponding bid energy for the bid price. On receiving this message from the upstream controller, the Price Controller Agents co-ordinates with all the local and downstream devices and collates the energy demand. On receiving all the energy demands, it responds with total energy demand corresponding to this price point to the upstream controller. The end devices receiving this message respond with their total energy demand corresponding to this price point.

d. Bid Budget Messages

Bid Budget Messages are Control Messages intended for the Price Controller Agent and contain a bid budget. On receiving this message from the upstream controller, the Price Controller Agents coordinates with all the local and downstream devices and computes an optimal price that limits the total energy demand to the allocated budget. The PCA then publishes this optimal price for all the devices associated.

e. Energy Demand Messages

Energy Demand messages are Reporting Messages and contain bid energy associated with the bid price message received by the agents.

f. Active Power Messages

Active Power Message is a Reporting Message and contains active power associated with the agent's latest optimal price or budget message. All energy-consuming devices post their active power at regular intervals.

An example optimal price point message (Control Message) is shown in Figure 3-14. Here, the building level controller posted an optimal price point message of 0.2 cents. The price_id is '84663032'. This message is applicable for 1 hour (duration 3600 seconds) starting 2020-07-29 11:31 hours and is addressed to all the devices. Corresponding to this Control Message, a sample active power message reported by the SmartStrip is shown in Figure 3-15. The SmartStrip is power consumption is 46.46 W for the price corresponding to price_id '84663032' (i.e., 46.46W @ 0.2 cents).

```
{
  "msg_type": 0,
  "value": 0.2,
  "value_data_type": "float",
  "units": "cents",
  "price_id": 84663032,
  "duration": 3600,
  "isoptimal": true,
  "one_to_one": false,
  "src_ip": "192.168.1.11:8080",
  "src_device_id": "BuildingController-11",
  "dst_ip": null,
  "dst_device_id": null,
  "ttl": 30,
  "ts": "2020-07-29 11:31:27.329872Z",
  "tz": "UTC"
}
```

Figure 3-14: Example Optimal Price Point JSON Message (Control Message)

```
{
  "msg_type": 2,
  "value": 46.46,
  "value_data_type": "float",
  "units": "W",
  "price_id": 84663032,
  "duration": 3600,
  "isoptimal": true,
  "one_to_one": false,
  "src_ip": "192.168.1.72:8080",
  "src_device_id": "SmartStrip-72",
  "dst_ip": "192.168.1.11:8080",
  "dst_device_id": "BuildingController-11",
  "ttl": 30,
  "ts": "2020-07-29 11:32:41.792588Z",
  "tz": "UTC",
  "energy_category": 9
}
```

Figure 3-15: Example Active Power JSON Message (Reporting Message)

3.3.2 iSPACE Agents

Each energy-consuming device has two principal agents: A) Device Controller Agent (DCA), and B) Price Agent (PA). Depending on the device's available computing power and complexity, the two agents can either be clubbed into a single agent or run in the coordinator. For example, in the case study implementation of the framework (refer to Chapter IV), the DCA & PA have been clubbed into a single agent for the SmartHub. Whereas for the Radiant Cubicle, the PID controller run in the automation server of the building management system and the DCA & PA are clubbed into a single agent and run in the SmartHub (the co-ordinator for task level). Besides, one of the devices at each level acts as a coordinator. The coordinator has two additional agents: C) Price Controller Agent (PCA) and D) Bridge Agent.

A. Price Agent (PA)

Each device has a Price Agent that registers itself with the Price Control Agent (PCA). The price agent receives various Control and Probe Messages (either price or budget) and responds with Reporting Messages.

The key functions of the PA are as follows:

1. The PA registers with the co-ordinator, PCA, as an energy-consuming device.
2. Receive various Control and Probe Message (either price, budget) from the PCA and respond accordingly.
3. On receiving an optimal price or budget from the PCA, the PA computes new setpoints that optimally operate the device. Once the setpoint is adjusted, the DCA takes over.
4. On receiving a bid price message from the PCA, the PA computes the predicted energy demand based on energy demand functions and participate in the bidding process by publishing the bid energy.

On receiving an optimal price from PCA, the new setpoints are computed based on the respective price functions of each energy-consuming device. Figure 3-16 shows the process flow. Similarly, the PA receives an optimal budget (target energy) from PCA and computes an optimal price corresponding to target energy using price and energy demand functions. And then, like the case when the optimal price is received from PCA, the new setpoints are computed based on the respective price functions. Figure 3-17 illustrates the process flow.

A detailed explanation on price functions is provided in section 3.4.3. The price functions define the relation between various price points and corresponding set-points considering user preferences.

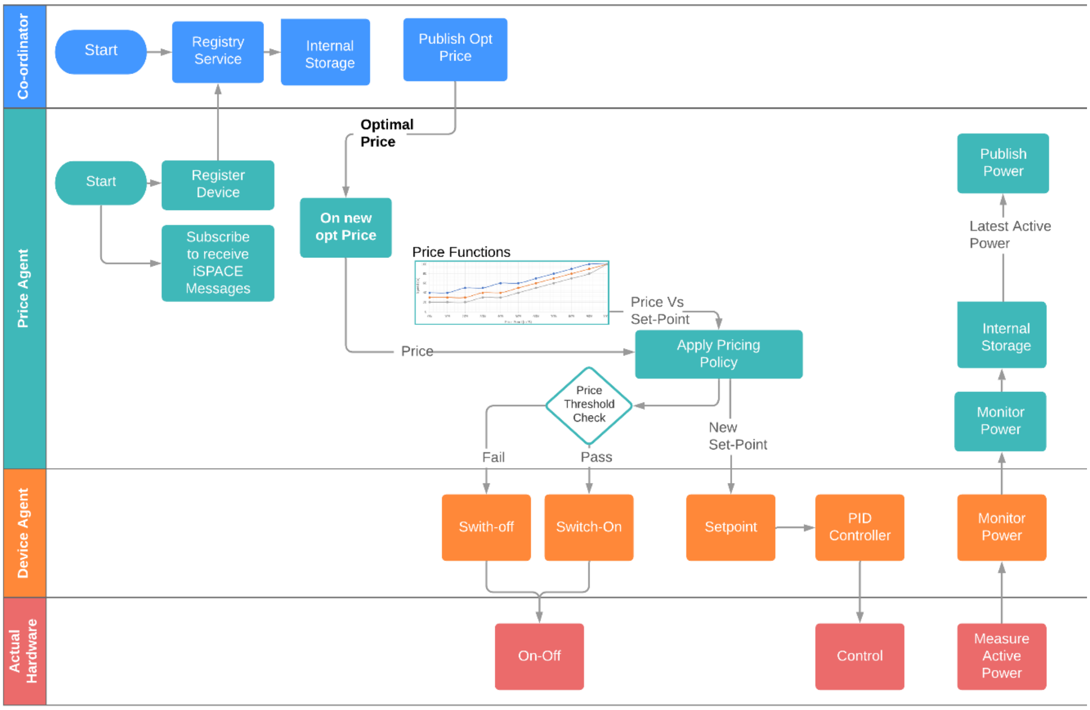


Figure 3-16: Price Agent process flow diagram for the optimal price received from PCA

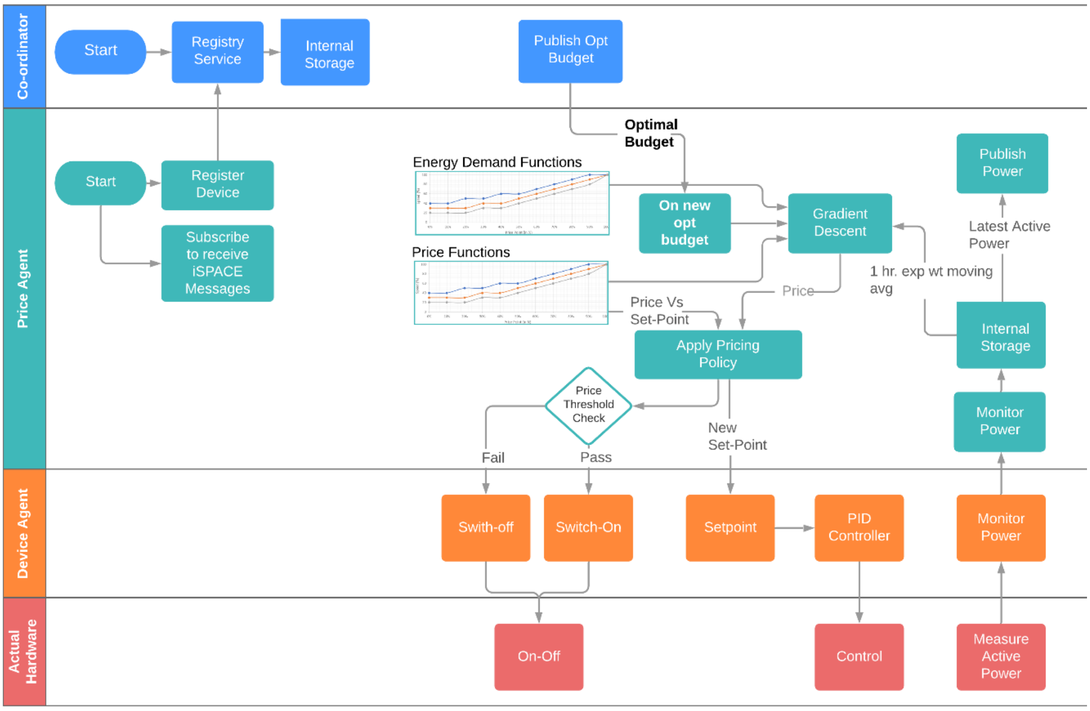


Figure 3-17: Price Agent process flow diagram for the optimal budget received from PCA

B. Price Controller Agent (PCA)

The main functions of a PCA are as follows:

1. Register with upstream PCA to receive new prices or budgets from upstream PCA.
2. Coordinate with all the local and downstream energy-consuming devices associated with the PCA.
3. Upon receiving a new bid price or bid budget from the upstream PCA, the PCA initiates a local bidding process and publishes new prices. The information available with the PCA is the energy demand, the current states, and the sensors data of the associated devices. The PCA can use any energy/comfort optimization strategy.
4. Collate and sort the bids from the associated device's Price Agents (PA) and downstream PCAs.
5. This process continues until the prices no longer change (or the changes are within a pre-specified threshold). Then, the process terminates, and PCA sends the total energy demand as its bid to the upstream PCA.
6. If the PCA is authorised to conclude the bidding, primarily the building level PCA, publishes the final state of the price as optimal price.

The functionality of the PCA depends on its state and mode of operations. Various states and modes of operations are defined in Table 3-2 and Table 3-3, respectively.

The key configuration inputs necessary to execute the PCA are listed in Appendices A-3. These configurations can be pre-configured through configuration files or changed at run time using JSON RPC methods.

Table 3-2: PCA States

Sl. No.	States	Description
1	ONLINE	In this state, the PCA register itself with the upstream PCA and actively receives CM and PM from upstream PCA and post RM accordingly. In coordination with all the devices associated with the PCA at that level, the PCA optimally operates the associated processes if the PCA mode is set either to DEFAULT_OPT or EXTERN_OPT mode (refer to Table 3-3 for various PCA modes).
2	STANDALONE	In this state, PCA acts as the primary price controller of all the associated devices at that level and the downstream price controller agents. PCA changes to this state when the duration of the latest Control Message from the upstream elapsed, and no further communication from upstream is possible. Besides, when changed to this state, the PCA publishes a new default optimal price to all the associated devices to bring the devices to a default state.

Table 3-3: PCA Modes

Sl. No.	Mode	Description
1	PASS_ON_PP	In this mode, PCA acts as a passive agent and pass on the Control/Probe Messages received from upstream to all the associated devices accordingly. In effect, the upstream PCA acts as a central auctioneer. 1. The optimal price or bid price received from upstream PCA is passed on to all the associated devices without any changes.

		2. Furthermore, if an optimal or bid budget is received, it is distributed according to the device's weightage, and respective budgets are published to the devices.
2	DEFAULT_OPT	In this mode, PCA acts as the local auctioneer. On receiving a bid price or budget from the upstream PCA, the PCA initiates a local bidding process to compute an optimal energy demand corresponding to the upstream bid price or budget.
3	EXTERN_OPT	The optimization functionality is off-loaded to an external optimizer agent in this mode. However, the coordination of all the devices is facilitated by the PCA. PCA publishes bid prices and collates all the bid energy demands from all the devices associated with the PCA.

In the case study implementation of the framework, two strategies have been implemented: 1) Mode PASS_ON (budget distribution based on device's weightage), and 2) Mode DEFAULT_OPT (cost minimization using gradient descent with momentum).

a. Mode PASS_ON (budget distribution based on device's weightage):

When the PCA is configured to PASS_ON modes, the optimal budget received from the upstream PCA is divided per each device's weightage among all the active devices. Assume $[w_1, w_2, w_3, \ldots\ldots w_n]$ are the respective device's weightage. Assume $[b_1, b_2, b_3, \ldots\ldots b_n]$ are the individual budgets, respectively. Assume B is the allocated budget by the upstream PCA. The individual budget of active devices is calculated as $b_i = \frac{w_i}{sum_{weightage}} * B$, where $sum_{weightage}$ is sum of all active device's weightage (Pseudo Code 1).

Pseudo Code 1: To compute individual budgets based on the device's weightage

Initialization

$$sum_{weightage} = 0$$

for each device

    if device active

$$sum_{weightage} = w_i + sum_{weightage}$$

end for

for each device

    if device active

$$b_i = \frac{w_i}{sum_{weightage}} * B$$

    Else

$$b_i = 0$$

end for

b. Mode DEFAULT_OPT (cost minimization using gradient descent with momentum):

When the PCA is configured to DEFAULT_OPT modes, the PCA initiates the local bidding process upon receiving bid price or bid budgets. Gradient descent with momentum has been used to compute the optimal energy demand corresponding to the received bid price or bid budget. Pseudo Code 2 gives typical steps involved. This pseudocode represents an iterative process where the optimal price is adjusted based on the deviation between the target and actual energy demand, aiming to converge towards an optimal solution that balances energy usage and cost-effectively. Besides, on receiving the optimal budget from the upstream PCA, the allocated budget is divided as per each device's weightage among all the active devices.

Pseudo Code 2: New optimal price using gradient descent with momentum

1: **Initialization**

$$pp_{old} = pp_{previous\_opt\_price}$$

$$ed_{old} = exp\_wt\_mv\_avg(power) * duration$$

$$ed_{target} = \left(\frac{device_{weightage}}{\sum device_{weightage}}\right) * budget$$

$$ed_{delta} = ed_{target} - ed_{old}$$

2: **Repeat until convergence**

3: $$pp_{new} = pp_{old} - \sum\left\{\delta * \left(\frac{device_{weightage}}{\sum device_{weightage}}\right) * \left(ed_{target} - ed_{old}\right)\right\} + \sum\{\alpha * ed_{delta}\}$$

4: $Publish\ pp_{new}\ to\ all\ asociated\ energy\ consuming\ device$

5: $Get\ ed_{new} from\ all\ asociated\ energy\ consuming\ device$

6: $$ted_{new} = \sum ed_{new}$$

7: if close (budget, ted_{new}, deadband) #check for other convergence criteria.
return pp_{new}

8: $ed_{delta} = ed_{new} - ed_{old}$
$ed_{old} = ed_{new}$
$pp_{old} = pp_{new}$

Step 1 initialises the variables such as the previous optimal price (ppold), the exponentially weighted moving average of energy consumption (edold), the target energy demand (edtarget), and others. Step 3 calculates a new price based on the previous price, the difference between the target and actual energy demand, and other factors, using gradient descent (step size (δ) and momentum (α)). The new price is published to all the energy consuming devices (including local Price Agents and downstream Price Controller Agents). In step 5, the energy demand corresponding to the new price point is collated from all the energy consuming devices. Sum up the individual energy demands to obtain the total energy demand. Finally, in step 7, verify if the total energy demand is close to the target within a defined deadband. If so, return the new price as market clearing price (optimal price); otherwise, the process repeats from Step 3.

The success of any one of the below criteria has been used for the exit strategy.

1. The total energy demand is within a specific dead band of the budget, or
2. There is no change in price for a certain number of iterations, or
3. The number of iterations reached pre-defined maximum iterations, or
4. The duration of the bidding process exceeds a pre-defined bidding timeout.

C. Device Controller Agent (DCA)

Each energy-consuming device are represented by their respective DCA (traditional controllers), and these agents run traditional control algorithm like PID, on/off, and such mechanism to achieve desired output (example: comfort) for given inputs (example: set-points).

The key functions of the agent are:

1. Access local sensors and report the sensors data at a regular interval.
2. Control the local actuators for various set-points/levels/speeds accordingly and provide APIs for other agents to interact with them (especially the Price Agent).

D. Bridge Agent (BA)

As discussed previously, the agents in a device communicate over a message bus using iSPACE messages. There is a need for communication between two different Price Controller Agents (PCA) running in two different devices in the framework.

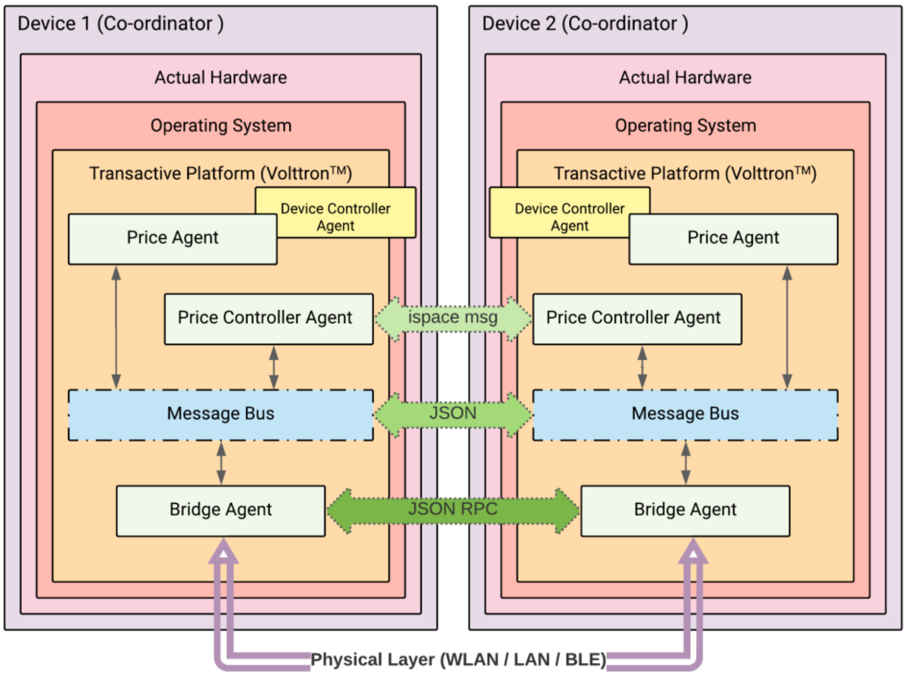


Figure 3-18: Role of Bridge Agent

The primary role this agent to maintain a registry of the all the local price agents (energy consuming devices) at the task level and any downstream price controller agent and can also transfer messages from one message bus of one instance to another message bus of a different machine/instance has been developed. For this purpose, a Bridge Agent (Figure 3-18) was developed. It is responsible for packet forwarding, including routing.

On booting, the downstream BA registers with the upstream BA. The BAs subscribe to the relevant topics on their respective message buses and marshal the iSPACE messages posted by the PCA to the related topics of the remote message bus using the Remote Procedure Call (RPC) mechanism.

The main functionality of the BA is to retrieve data from the message bus and publish it to either an upstream or downstream message bus. When sending data downstream, the information pertains to price points, while upstream data includes energy demand or active power. The assumption here is that the bridge communicates with a single instance for upstream (US) and multiple devices for downstream (DS):

- Price point data follows a one-to-many communication pattern.
- Energy demand and active power data follow a one-to-one communication pattern.

Upon startup, DS devices register with the upstream bridge. The bridge is aware of upstream devices and registers with them. Any changes in energy demand are promptly posted to the upstream bridges. However, the bridge is not initially aware of downstream devices. It starts posting messages (price points) to them as soon as the downstream bridges register with it.

The primary concern in a distributed system is the unavoidable communication failures that need to be addressed sufficiently. On failure to post for a maximum number of retries, the BA de-registers the downstream device. Subsequent messages would be posted to the device when the device becomes active and registers again with the BA. The PCA can get the list of actively associated energy-consuming devices and act accordingly.

3.4 System Model and Pricing Formulation

The system model and pricing formulation has been adapted from [26] and [27] for the case study implementation and implemented as the default option is described in this section. However, the framework facilitates support for more complex algorithms. And these algorithms can be implemented as external agents or cloud services if required.

In general, in the literature [26] and [27], the problem formulation is devised to achieve efficient energy allocation, maximising the utility of energy-consuming devices while optimising the energy provider's payoff functions, thus adopting a social welfare maximisation approach. The primal problem, with its inner optimisation problem as a constraint, presents a complex challenge. Assuming the utility function and payoff function are quasilinear, the primal problem, a concave maximisation problem, can be solved using convex programming techniques in a centralised manner. However, the utility functions of the energy-consuming devices and the pay-off functions of energy providers agents are private to their respective agents, which makes this approach infeasible. One solution method found in the literature [26] and [27] is to reformulate the primal problem using the primal-dual approach. This approach transforms the primal problem into an unconstrained objective function and aim to find the minimum efficiently. The solution to the dual problem is obtained by introducing a shadow price (Lagrange multiplier) representing the optimal solution. This approach allows each energy-consuming device and the energy provider to solve their local optimisation problems, leading to optimal energy consumption and energy generation. These algorithms have been extended to quadratic utility and pay-off functions, where the globally optimal solution can be also guaranteed [26] and [27].

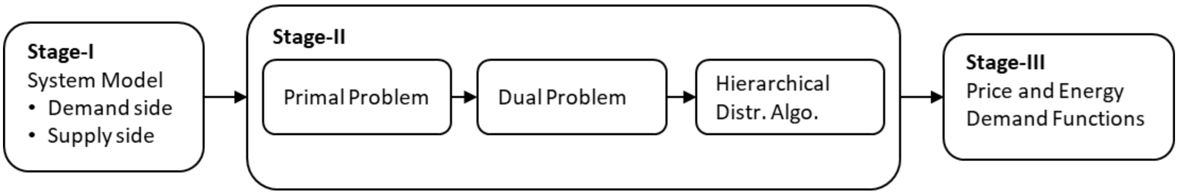


Figure 3-19: The optimization problem solution approach

Figure 3-19 presents the methodology adopted for the default system model and pricing formulation in this thesis. Section 3.4.1 initiates the discussion by outlining the demand and supply side optimisation problems. Subsequently, in Sub-section 3.4.2, these individual optimisation problems are combined to formulate a global primal problem as a social welfare optimisation problem. The primal-dual approach transforms the primal problem into an unconstrained convex optimisation problem, facilitating solving through a hierarchical distributed pricing algorithm.

Moreover, Section 3.4.3 introduces the construction of price and energy demand functions for the demand side, substituting utility functions for energy-consuming devices. These functions, a novel aspect of our study, are precomputed and adhere to comfort parameter constraints.

Furthermore, Section 5.7.2.E presents the crucial simulation study results about convergence rates, validating the effectiveness of our methodology.

3.4.1 System Model

In the system, there is an energy provider (upstream Price Controller Agent, (PCA)), there are several energy-consuming devices (PECS, HAVC systems, Lighting Systems, and such others), and there is a coordinator (local Price Controller Agent). A Price Agent (PA) for each energy-consuming device computes a new setpoint that operates the device optimally.

The PAs also coordinate with local PCA to receive a price and respond with energy demand bids. Once the PA adjusts the setpoint, a Device controller Agent (DCA), i.e., a traditional controller, takes over.

A. Demand Side (Agents Preference and Utility Function)

Figure 3-20 shows typical feedback based closed-loop PECS control mechanism, an energy-consuming device. The controller (DCA) attempts to minimize the error over time by adjusting the control variable, u_t.

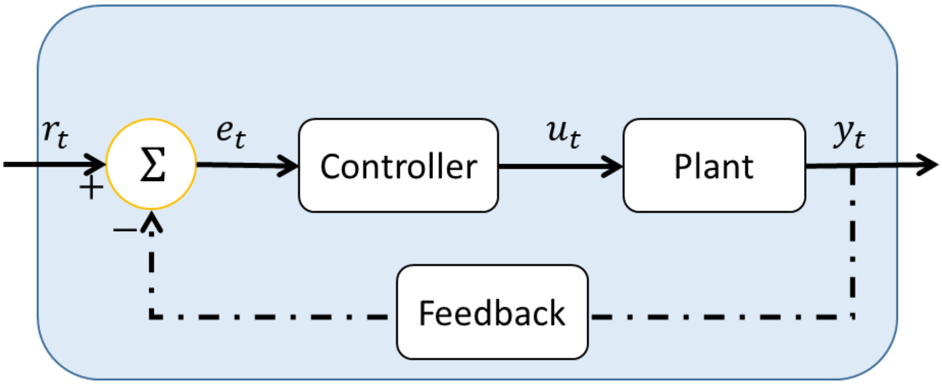


Figure 3-20: The block diagram of a closed-loop control system.

where,

r_t *is the desired process value or SetPoint* (SP)

y_t *is the measured process value* (PV)

e_t *is the error value,* $r_t - y_t$

u_t *is the control variable*

For a PID controller (generally used for HVAC systems, RadiantCubicle and such), it is represented in standard form as:

$$u_t = K_p e_t + K_i \int_0^t e_\tau \, d\tau + K_d \frac{de_t}{dt} \qquad \text{Eq. 3-1}$$

where,

K_p *is the proportional gain, a tuning parameter*

K_d *is the derivative gain, a tuning parameter*
u_t *is the control variable*
e_t *is the local error*
τ *is the variable of integration*
(generally takes on historical error values from time 0 *to the present* t*)*

Mathematically,

$$u_t = \mathbb{C}(e_t) \qquad \text{Eq. 3-2}$$

Where $\mathbb{C}$ is a piecewise linear function operating on the local error e_t (i.e., $r_t - y_t$) and t is the time or instantaneous time. Similar linear functions can be constructed for a simple regulator or on-off based energy-consuming devices.

Let $\mathcal{N}$ denote the set of all energy-consuming device, where $N \stackrel{\text{def}}{=} |\mathcal{N}|$. It is assumed that the devices are independent of each other. Let us assume, the intended time cycle for the operation of the energy-consuming devices is divided into k time slots, where $K \stackrel{\text{def}}{=} |\mathrm{K}|$ and K is set of all time slots. For each energy-consuming device $i \in \mathcal{N}$, let x_i^k denote the amount of power consumed by energy-consuming device i in time slot k and each energy-consuming device follows the control rule of Eq. 3-2 and specifically Eq. 3-1 if it is a PID controller. Mathematically,

$$x_i = \mathcal{O}_i(e_i) \qquad \text{Eq. 3-3}$$

That is, $\mathcal{O}_i$ is a piecewise linear operator operating on the local error e_i consuming x_i amount of energy for the energy-consuming device $i \in \mathcal{N}$ to achieve its goal state by eliminating error e_i (for brevity, the dependency on time t is explicitly not indicated).

Some devices, such as HVAC systems, may run at idling. Such systems consume minimum power and are not switched off. If m_i^k, M_i^k denote the minimum and maximum power consumed by the device $i \in \mathcal{N}$ in time slot $k \in K$, then consumed power x_i^k has to satisfy

$m_i^k \leq x_i^k \leq M_i^k$. And the power consumption interval I_i^k can be defined as:

$$I_i^k \stackrel{\text{def}}{=} [m_i^k,\ M_i^k] \qquad \text{Eq. 3-4}$$

Hence, for each time slot $k \in$ K, if R_k^{min} and R_k^{unc} denote the minimum load and unconstrained maximum load to cover the power requirements of all users, then it follows:

$$R_k^{min} \stackrel{\text{def}}{=} \sum_{i \in \mathcal{N}} m_i^k\ , \qquad \forall\ k\ \in\ K \qquad \text{Eq. 3-5}$$

$$R_k^{unc} \stackrel{\text{def}}{=} \sum_{i \in \mathcal{N}} M_i^k\ , \qquad \forall\ k\ \in\ K \qquad \text{Eq. 3-6}$$

For each time slot $k \in$ K, if L_k^* denote the available power from the upstream energy provider as defined sub-section 3.4.1.B, and let L_k^{min} and L_k^{max} denote the minimum and constrained maximum available power, respectively, we have:

$$0\ \leq L_k^{min}\ =\ R_k^{min} \stackrel{\text{def}}{=} \sum_{i \in \mathcal{N}} m_i^k\ \leq\ L_k^* \leq L_k^{max}\ \leq \sum_{i \in \mathcal{N}} M_i^k \stackrel{\text{def}}{=} R_k^{unc}\ \forall\ k\ \in\ K \qquad \text{Eq. 3-7}$$

The local PCA (coordinator) ensures that the upstream PCA (energy provider) has the minimum capacity to provide the minimum power requirements of all the energy-consuming devices R_k^{min} in each time slot.

In the Eq. 3-7, R_k^{unc} is the total 'free' demand of all agents taken independently, as implied by the sum of the energy-consuming devices control equations in a situation with an unconstrained supply of resources. But there may be a smaller cap L_k^* on the total available resource if it must be shared by the agent society. The total demand R_k^{unc} by the individual agents in the unconstrained case derives from local information (according to Eq. 3-3), whereas the resource limitation to L_k^* is the result of an external action or situation. That is, in the absence of resource constraints the total system acts as collection of local independent controllers that behave in accordance with conventional control engineering theory.

We propose utility functions as defined in microeconomics for all the energy-consuming devices. It is critical that these functions are differentiable. Hence, it is assumed that these functions are decomposable over time and are differentiable for a particular time slot and can be created for all the energy-consuming devices in the system (for example, task fan, task light, HVAC, lighting). That is, the utility functions assigned to each energy-consuming devices can be analysed or broken down into its constituent parts that vary over different time intervals in such a way that the original function can be reconstructed (i.e., recomposed) from those parts and the constituent parts are differentiable during their respective time intervals. We further assume that the energy-consuming devices behave independently, and the utility functions of various energy-consuming devices are independent.

The energy demand for each device may vary based on different parameters. For example, the user preferences, a user would prefer cool over the warm environment; the time of day, a device would need more power during day than night; and such other factors. The utility function represents the degree of well-being (ordinal measure, a relative ranking) the device provides for the user as a function of its energy demand. Popular utility functions include Cobb-Douglas, linear, and quadratic functions [85]. The different responses of different devices to various price scenarios can be modelled either analytically or statically.

Let $U(x_i^k, \omega_i^k)$ represent the utility function each energy-consuming device of the device $i \in \mathcal{N}$ in time slot $k \in K$ where x_i^k is the power consumption level and ω_i^k is the weight factor or device preference. For each energy-consuming device, it is assumed that the utility functions, $\mathrm{U}(\mathrm{x}_\mathrm{i}^\mathrm{k}, \omega_\mathrm{i}^\mathrm{k})$, are monotonically increasing, strictly concave, and continuously differentiable functions with respect to power consumption for the given resource constraints.

That is, mathematically, the assumptions are as follows:

1) **Utility functions are non-decreasing**: energy-consuming devices are interested in consuming more power if possible until they reach their maximum power consumption levels. This implies that we have:

$$\frac{\partial U\left(x_i^k, \omega_i^k\right)}{\partial x_i^k} \geq 0 \qquad \text{Eq. 3-8}$$

2) **The marginal utility is non-increasing:** For notational convenience, if we denote marginal utility as:

$$V\left(x_i^k, \omega_i^k\right) \stackrel{\text{def}}{=} \frac{\partial U\left(x_i^k, \omega_i^k\right)}{\partial x_i^k} \qquad \text{Eq. 3-9}$$

It follows that:

$$\frac{\partial \mathrm{V}\left(\mathrm{x}_\mathrm{i}^\mathrm{k}, \omega_\mathrm{i}^\mathrm{k}\right)}{\partial \mathrm{x}_\mathrm{i}^\mathrm{k}} \leq 0 \qquad \text{Eq. 3-10}$$

That is, the utility functions, $U\left(x_i^k, \omega_i^k\right)$ is strictly concave w.r.t x_i^k and the level of satisfaction can gradually get saturated.

3) **Able to rank the devices:** The energy-consuming devices can be ranked based on their utilities. That for a given power consumption x, a larger ω implies a larger $U(x, \omega)$, that can be expressed as:

$$\frac{\partial U\left(x_i^k, \omega_i^k\right)}{\partial \omega_i^k} > 0 \qquad \text{Eq. 3-11}$$

4) No power consumption brings no benefit: Intuitively, no power consumption by the device brings no benefit, so we have:

$$U\left(0, \omega_i^k\right) = 0, \quad \forall\, \omega > 0, i \in \mathcal{N}, and\ k \in K \qquad \text{Eq. 3-12}$$

Literature results allow using a quadratic utility function[26], [27], [31]. For mathematical simplicity, let us assume a quadratic utility function $U\left(x_i^k, \omega_i^k\right)$ as follows:

$$U\left(x_i^k, \omega_i^k\right) = (-)\frac{1}{2c_i}\left[x_i^k - \mathcal{O}_i\left(e_i^k\right)\right]^2$$
$$where\ \omega_i^k \stackrel{\text{def}}{=} {}^{1}\!/_{c_i} \text{ is a weight factor } > 0 \qquad \text{Eq. 3-13}$$

The weight factor may express individual preference differences and priorities and allows the agent to make a concession in its utility maximization. A higher ω makes the utility function sharper, and a lower ω makes the utility function broader. Moreover, ω can be used to express some social hierarchy. For example, when occupancy levels are low, PECS's device would be preferred more than ambient conditioning devices; a server room is more important than the cafeteria. Furthermore, c can be seen as a measure of the agent's willingness to make concessions.

A device i that consumes x_i^k kW electricity during time slot $k \in K$ at a rate λ^k per kWh is charged $\lambda^k x_i^k$ per hour. Hence the device's welfare, $W(x_i^k, \omega_i^k)$ is defined as:

$$W(x_i^k, \omega_i^k) = U(x_i^k, \omega_i^k) - \lambda^k x_i^k \qquad \text{Eq. 3-14}$$

For each announced price λ^k, each device tries to adjust its power consumption $x_i^{k*}(\lambda^k)$ to maximize its welfare and is therefore defined as:

$$\boldsymbol{x_i^{k*}(\lambda) = \underset{m_i^k \leq x_i^k \leq M_i^k}{argmax}\left[U(x_i^k, \omega_i^k) - \lambda^k x_i^k\right]} \qquad \text{Eq. 3-15}$$

Where the first term, $U(x_i^k, \omega_i^k)$, represents the utility of consuming x_i^k units of electricity and $\lambda^k x_i^k$ is the payment of electricity for each energy-consuming device $i \in \mathcal{N}$, in time slot $k \in K$.

The above model gives an intricate representation of how energy-consuming devices (agents) in a system can optimize their power usage based on price signals from an energy provider.

The literature suggests that if all agents possess equal weights and if individual preferences are proportional to free demand, each agent receives the same relative cut in resources [25], [26], [27], [86], [86], [87].

For example, let's consider that for each announced price λ^k, each energy-consuming device tries to adjust its power consumption x_i^k to maximize its welfare defined by Eq. 3-15. If price λ^{k*} represents the market clearing price at equilibrium, maximum welfare can be achieved by setting the derivative of Eq. 3-14 equal zero and the marginal benefit of the user would be equal to the announced price. Hence, we have,

$$\frac{\partial W\left(x_i^k, \omega_i^k\right)}{\partial x_i^k} = 0 \qquad \forall\, \omega > 0, i \in \mathcal{N}, and\ k \in K$$

$$\Rightarrow \frac{\partial}{\partial x_i^k}\left[U\left(x_i^k, \omega_i^k\right) - \lambda^{k*} x_i^k\right] = 0 \qquad \text{Eq. 3-16}$$

Substituting Eq. 3-13 in Eq. 3-16, we have

$$\frac{\partial}{\partial x_i^k}\left[(-)\frac{1}{2c_i}\,[x_i^k - \mathcal{O}_i\left(e_i^k\right)]^2 - \lambda^{k*} x_i^k\right] = 0 \qquad \text{Eq. 3-17}$$

$$where\ \mathcal{O}_i\left(e_i^k\right)\ represents\ the\ uncontrained\ case\ and$$

$$x_i^k\ constrained\ condtion$$

$$\Rightarrow \frac{-1}{2c_i}.\,2x_i^k + \frac{1}{2c_i}.\,2\mathcal{O}_i\left(e_i^k\right) - \lambda^{k*} = 0$$

$$\Rightarrow \frac{-x_i^k}{c_i} + \frac{\mathcal{O}_i\left(e_i^k\right)}{c_i} - \lambda^{k*} = 0$$

$$\Rightarrow -x_i^k + \mathcal{O}_i\left(e_i^k\right) - c_i\lambda^{k*} = 0$$

$$\Rightarrow x_i^k = \mathcal{O}_i\left(e_i^k\right) - c_i\lambda^{k*} \qquad \text{Eq. 3-18}$$

Hence, total power required $\forall i$:

$$\sum_{i \in \mathcal{N}} x_i^k = \sum_{i \in \mathcal{N}} [\, \mathcal{O}_i\left(e_i^k\right) - c_i\lambda^k\,]$$

$$= \sum_{i \in \mathcal{N}} \mathcal{O}_i\left(e_i^k\right) - \sum_{i \in \mathcal{N}} c_i\lambda^k \qquad \text{Eq. 3-19}$$

From Eq. 3-7, for constrained cases, we have:

$$\sum_{i \in \mathcal{N}} x_i^k = L_k^* \qquad \text{Eq. 3-20}$$

Furthermore, for unconstrained cases, we have:

$$\sum_{i \in \mathcal{N}} \mathcal{O}_i(e_i^k) = R_k^{unc} \qquad \text{Eq. 3-21}$$

At equilibrium, substituting Eq. 3-20 and Eq. 3-21 in Eq. 3-19, we have:

$$L_k^{max} = R_k^{unc} - \sum_{i \in \mathcal{N}} c_i \lambda^{k*}$$

$$\Rightarrow \lambda^{k*} = \frac{(R_k^{unc} - L_k^*)}{\sum_{i \in \mathcal{N}} c_i} \qquad \text{Eq. 3-22}$$

Substituting Eq. 3-22 in Eq. 3-18, we have:

$$\boldsymbol{x_i^{k*} = \mathcal{O}_i(e_i^k) - \frac{c_i}{\sum_{i \in \mathcal{N}} c_i}(R_k^{unc} - L_k^*)} \qquad \text{Eq. 3-23}$$

Some special cases of Eq. 3-23 are of interest to show explicitly.

Case-1: all agents are equal in the sense of having equal weights:

$$\forall i : c_i = 1 \Rightarrow x_i^{k*} = \mathcal{O}_i(e_i^k) - \tfrac{1}{N}(R_k^{unc} - L_k^*), \text{ and}$$

$$\lambda^{k*} = \frac{(R_k^{unc} - L_k^*)}{N} \qquad \text{Eq. 3-24}$$

This equation says that if all agents are equal, they all must take the same absolute cut in resources.

Case-2: Each agent gets the same relative cut in resources if the agent's preferences are proportional to their unconstrained demand:

$$\forall i : c_i = \mathcal{O}_i(e_i^k) \Rightarrow x_i^{k*} = \mathcal{O}_i(e_i^k)\left(\frac{L_k^*}{R_k^{unc}}\right) \text{ and}$$

$$\lambda^{k*} = \frac{(R_k^{unc} - L_k^*)}{R_k^{unc}} \qquad \text{Eq. 3-25}$$

This equation says that each agent gets the same relative resource cut if all preferences are proportional to free demand.

In essence, when all agents are equal in weight, they must all experience an equal absolute cut in resources. Furthermore, if each agent's preferences align proportionally with their unconstrained demand or are proportional to free demand, every agent will receive an equivalent relative resource cut.

B. Supply Side (Energy Cost Model)

The energy provider supplies electricity and receives payments. Assume a monotonically increasing, strictly concave, and continuously differentiable quadratic cost function $C_k(L_k)$ indicating the cost of providing L_k units of energy offered by the energy provider in each time slot $k \in K$. There are different functions in different ranges; however, a common representation is the quadratic cost function, as supported by literature findings [26], [34], [35]:

$$C_k(L_k) = a_k L_k^2 + b_k L_k + c_k, \qquad \forall k \in K \qquad \text{Eq. 3-26}$$

Where C_k is the operating cost of the energy provider, L_k is the electrical power output, and $\mathrm{a_k}, \mathrm{b_k}, \mathrm{c_k} \geq 0$ are pre-determined fuel cost coefficients of the energy provider.

This function could capture a case where demand-supply events take place. Then it has the following payoff function:

$$\sum_{i \in \mathcal{N}} \lambda^k x_i^k - C_k(L_k) \qquad \forall\, k \in K \qquad \text{Eq. 3-27}$$

Where the first term, $\sum_{i \in \mathcal{N}} \lambda^k x_i^k$ represents the total payment from the distributed energy, and the second term, $C_k(L_k)$ represents the production cost for the energy provider.

The supplier maximizes the payoff and is therefore defined as:

$$\boldsymbol{L_k^* = \underset{L_k^{min} \le L_k \le L_k^{max}}{maximize} \lambda^k L_k - C_k(L_k)} \qquad \text{Eq. 3-28}$$

3.4.2 Pricing Formulation

The demand side and supply side optimisation problems defined in the previous section 3.4.1 are combined to formulate a global primal problem as a social welfare optimisation problem. The primal-dual approach transforms the primal problem into an unconstrained convex optimisation problem, facilitating solving through a hierarchical distributed pricing algorithm.

A. Primal Optimization Formulation

It is desirable to utilize the available power allocated by upstream PCA at each level. The sum of the utility functions of all energy-consuming devices is maximized, and the cost imposed for the power consumed is minimized. However, each energy-consuming device will choose its consumption level to maximize its welfare function introduced in Eq. 3-14.

These individual consumptions levels may not be optimal at a societal level for a general price announced by the energy provider. To align these individual optimal consumption levels with the optimal societal case, we need to adopt the sum of all utility functions minus the cost imposed as the objective functions while the consumption levels are constrained by available capacity. Under this model, the problem that needs to be solved for an optimal power consumption is as follows:

$$\underset{\substack{x_i^k \in I_i^k, \\ L_k^{min} \le L_k \le L_k^{max}, \\ i \in \mathcal{N},\ k \in K}}{\boldsymbol{maximize}} \sum_{k \in K} \sum_{i \in \mathcal{N}} U\left(x_i^k, \omega_i^k\right) - C_k(L_k)$$

$$\boldsymbol{Subject\ to} \sum_{i \in \mathcal{N}} x_i^k \le L_k \qquad \forall\, k \in K \tag{Eq. 3-29}$$

Where $U\left(x_i^k, \omega_i^k\right)$ are the utility functions, as defined in Eq. 3-13, $C_k(L_k)$ is defined in Eq. 3-26, and ω_i^k is the ω parameter of energy-consuming device i in time slot k.

The problem formulated in Eq. 3-29 is a concave maximization problem. It can be solved using convex programming techniques such as the interior point method [26]. However, the utility functions of energy-consuming devices are private to the corresponding energy-consuming device. Hence, the PCA will not have sufficient information to solve the problem Eq. 3-29.

Since the utility functions are assumed to be decomposable over time, the Eq. 3-29 is decomposable in k and can be solved independently for each time slot $k \in K$ [26]. Hence, we have the following *primal optimization problem* for each time slot $k \in K$:

Primal Problem:

$$\underset{\substack{x_i^k \in I_i^k, \\ L_k^{min} \leq L_k \leq L_k^{max}, \\ i \in \mathcal{N}}}{\boldsymbol{maximize}} \sum_{i \in \mathcal{N}} U(x_i^k, \omega_i^k) - C_k(L_k) \qquad \text{Eq. 3-30}$$

$$\boldsymbol{Subject\ to} \sum_{i \in \mathcal{N}} x_i^k \leq L_k$$

B. Dual Decomposition Approach

Although the objective function in Eq. 3-30 is further separable in x_i^k and L_k, the variables x_i^k and L_k are coupled by the imposed constraint that the total power consumption cannot exceed the available capacity in Eq. 3-30.

For the *primal problem* Eq. 3-30, the Lagrangian is defined as [88]:

$$\begin{aligned} \mathcal{L}(x, L_k, \lambda^k) &= \sum_{i \in \mathcal{N}} U(x_i^k, \omega_i^k) - C_k(L_k) - \lambda^k \left(\sum_{i \in \mathcal{N}} x_i^k - L_k \right) \\ &= \sum_{i \in \mathcal{N}} \left(U(x_i^k, \omega_i^k) - \lambda^k x_i^k \right) + \lambda^k L_k - C_k(L_k) \end{aligned} \qquad \text{Eq. 3-31}$$

where λ^k is the Lagrange multiplier and $x = (x_i^k, i \in \mathcal{N})$ for a fixed $k \in K$. Due to the separability of the first term in the Lagrangian, we can write the objective function of the dual optimisation problem, $\mathcal{D}(\lambda^k)$, for a fixed time slot $k \in K$ as:

$$\mathcal{D}(\lambda^k) = \underset{\substack{x_i^k \in I_i^k, \\ L_k^{min} \leq L_k \leq L_k^{max}, \\ i \in \mathcal{N}}}{\boldsymbol{maximize}} \; [\mathcal{L}(x, L_k, \lambda^k)] \qquad \text{Eq. 3-32}$$

$$\mathcal{D}(\lambda^k) = \underset{\substack{x_i^k \in I_i^k, \\ L_k^{min} \leq L_k \leq L_k^{max}, \\ i \in \mathcal{N}}}{\boldsymbol{maximize}} \left[\sum_{i \in \mathcal{N}} U(x_i^k, \omega_i^k) - C_k(L_k) - \lambda^k \left(\sum_{i \in \mathcal{N}} x_i^k - L_k\right)\right] \qquad \text{Eq. 3-33}$$

$$= \underset{\substack{x_i^k \in I_i^k, \\ L_k^{min} \leq L_k \leq L_k^{max}, \\ i \in \mathcal{N}}}{\boldsymbol{maximize}} \left[\sum_{i \in \mathcal{N}} \left(U(x_i^k, \omega_i^k) - \lambda^k x_i^k\right) + \lambda^k L_k - C_k(L_k)\right] \qquad \text{Eq. 3-34}$$

And the dual problem is:

Dual Problem:

$$\underset{\lambda^k > 0}{\boldsymbol{minimize}} \, \mathcal{D}(\lambda^k) \qquad \text{Eq. 3-35}$$

Due to the separability of the first term in the Lagrangian, we can re-write objective function of the dual optimization problem as [26]:

$$\underset{\lambda^k > 0}{\boldsymbol{minimize}} \left[\sum_{i \in \mathcal{N}} \mathcal{B}_i^k(\lambda^k) + \mathcal{S}_k(\lambda^k)\right] \qquad \text{Eq. 3-36}$$

where, $\mathcal{B}_i^k(\lambda^k) = \underset{x_i^k \in I_i^k}{\boldsymbol{maximize}} \, U(x_i^k, \omega_i^k) - \lambda^k x_i^k$, and Eq. 3-37

$$\mathcal{S}_k(\lambda^k) = \underset{L_k^{min} \leq L_k \leq L_k^{max}}{\boldsymbol{maximize}} \; \lambda^k L_k - C_k(L_k) \qquad \text{Eq. 3-38}$$

It has been assumed that the energy-consuming devices behave independently, and the utility functions of various energy-consuming devices are independent. Hence, the first term in the Eq. 3-36 can be decomposed into $\mathcal{N}$ separable sub problems in form of Eq. 3-37, which can be solved by the energy-consuming device PAs, and another sub problem in the form of Eq. 3-38, which energy provider (PCA) can solve. It can be shown that strong duality holds, and the dual problem Eq. 3-35 can be solved instead of the primal problem Eq. 3-30

[5], [26], [27]. In this case, we can obtain the solution to the dual problem λ^{k*}, and each energy-consuming device and the energy provider can simply solve their local optimization problems determined by Eq. 3-37 and Eq. 3-38 to obtain x_i^{k*} and L_k^* respectively.

Further, the local problem Eq. 3-37 that has to be solved by each energy-consuming device is similar to Eq. 3-15, introducing each energy-consuming device's welfare. That is if the energy provider would be able to charge at a rate λ^{k*}, and each energy-consuming device tries to maximize its welfare function, and it will be guaranteed by the strong duality that the total energy consumption will not exceed the provided capacity.

In the case study implementation, it is assumed that L_k^*, the local optimizer of Eq. 3-38 for a given λ^k is proportional to price point and computed as:

$$L_k^* = \left[\frac{\lambda^{(k-1)*}}{\lambda^k}\right] * \sum_{i \in \mathcal{N}} x_i^{(k-1)*}\left(\lambda^{(k-1)*}\right) \qquad \text{Eq. 3-39}$$

However, the local optimizer Eq. 3-38 can be updated to include locally available distributed energy resources like rooftop solar PV systems and energy storage systems. Any suitable energy cost model can be used as required.

C. Hierarchical Distributed Algorithm

In the previous section on dual decomposition approach, for a group of resource agents (energy-consuming device agents and energy provider agents), it has been explained that by charging energy-consuming devices with the solution of the dual problem λ^{k*}, the solution to the primal problem can be achieved. Further, each energy-consuming device can solve its local optimization problems determined by Eq. 3-37. It is possible to solve the previous section dual problem iteratively.

The Walrasian Tatonnement[7] process can utilise standard price-oriented market protocols like Wellman's WALRUS algorithm [89]. The agents (consumers and producers) respond to a price signal. The coordinator agent coordinates the agent's interactions, who adjusts the general price level towards a general equilibrium, announcing interim prices to elicit responses from the producer and consumer agents. Furthermore, this kind of primal-dual iterative algorithm enables parallel implementation and scales well with the transactive system's size [5].

Samadi et al. used the gradient projection method and developed a distributed algorithm [26]. In our approach, we extend the distributed algorithm to a hierarchical distributed multi-agent system to integrate PECS within the task environment and between task & ambient controls in a building. The agents (energy provider and energy-consuming device agents) are segregated based on their location vis-a-vis building level, zone level and task level. Within each level, the agents are further sub-grouped, forming a hierarchy of groups (that is, the building contains multiple zones, zones contain multiple task cubicles, and task cubicles contain multiple PECS).

For each sub-group, there is a coordinator agent, local PCA. The local PCA is represented as an energy-consuming agent to the upstream PCA. It bids on behalf of the energy-consuming devices in its sub-group to the upstream PCA. Also, it coordinates among energy-consuming devices within the sub-group to arrive at an optimal energy bid for the sub-group (an additional round of iterations for each additional level).

[7] Tatonnement (economics) (French for trial and error): A form of hill climbing (local search, iterative, start with an arbitrary solution, then attempt to find a better solution by making an incremental change to the solution).

In such a case, a price update for the coordinator agent at a particular level is:

$$\lambda^k_{t_{level}+1} = \left[\lambda^k_{t_{level}} - \gamma \frac{\partial \mathcal{D}(\lambda^k_{t_{level}})}{\partial \lambda^k_{t_{level}}}\right]^\dagger$$

$$= \left[\lambda^k_{t_{level}} + \gamma \left(\sum_{i \in \mathcal{N}_{level}} x_i^{k*}(\lambda^k_{t_{level}}) - L_k^*(\lambda^k_{t_{level}}) \right)\right]^\dagger \qquad \text{Eq. 3-40}$$

where $t_{level} \in \mathcal{T}_{level}$, and $\mathcal{T}_{level}$ is the set of time instances at which the PCA updates $\lambda^k_{t_{level}}$. Here, $\mathcal{N}_{level}$ denote the set of all energy-consuming devices at that level (that is, downstream level PCAs and energy-consuming devices at that level), where $N \stackrel{\text{def}}{=} |\mathcal{N}_{level}|$. $x_i^{k*}(\lambda^k_{t_{level}})$ local optimizer of Eq. 3-37 for that level's energy-consuming devices (that is, optimal bids from downstream level PCAs and the current level's energy-consuming devices) and $L_k^*(\lambda^k_{t_{level}})$ is the local optimizer of Eq. 3-38 for a given $\lambda^k_{t_{level}}$, respectively. Also, $\lambda^k_{t_{level}}$is the value of $\lambda^k_{t_{level}}$ in instance $t_{level} \in \mathcal{T}_{level}$, and γ is the step size.

Since γ is a constant, Eq. 3-40 can be conveniently written as:

$$\lambda^k_{t_{level}+1} = \left[\lambda^k_{t_{level}} + \left(\sum_{i \in \mathcal{N}_{level}} \gamma_i \frac{c_i}{\sum c_i} x_i^{k*}(\lambda^k_{t_{level}}) - L_k^*(\lambda^k_{t_{level}}) \right)\right]^\dagger \qquad \text{Eq. 3-41}$$

Where c_i is the device weightage and γ_i can be computed as $\frac{(\lambda_{max}-\lambda_{min})}{(M_i^k-m_i^k)}$. m_i^k, M_i^k denote the minimum and maximum power consumed by the energy-consuming devices $i \in \mathcal{N}_{level}$ in time slot $k \in K$.

Figure 3-21 depicts the interactions between the Price Controller Agents (PCA) and the energy-consuming device's Price Agents (PA). The energy-consuming device's Price Agent (PA) at Building Level and Zone Level are not depicted for brevity. However, these PAs subscribe to the corresponding level PCA and are like Task level PAs in functionality.

The system starts with its initial condition, which is assumed to be random. A random optimal price is published. All the PAs compute their corresponding setpoints or device states accordingly. Furthermore, the device controller maintains these states. The system is allowed to stabilize. Once the system is ready (sufficient historical data is available to compute the current energy demand), the market mechanism is initiated.

As illustrated in Figure 3-21, the steps are as follows:

1. The building level PCA updates the capacity value $L_k^*\left(\lambda_{t_{bld}}^k\right)$ by solving corresponding Eq. 3-38 for building. It further computes the new value of $\lambda_{t_{bld}}^k$ using corresponding Eq. 3-41 for building in each instance $t_{bld} \in \mathcal{T}_{bld}$. And broadcast the new value of $\lambda_{t_{bld}}^k$ to all the associated building level PAs and downstream PCAs (that is, zone PCAs). Refer to sequence number 1 in the figure.
2. The zone level PCA updates the capacity value $L_k^*\left(\lambda_{t_{bld}}^k\right)$ by solving corresponding Eq. 3-38 for the zone. It further computes the new value of $\lambda_{t_{zone}}^k$ using corresponding Eq. 3-41 for the zone in each instance $t_{zone} \in \mathcal{T}_{zone}$. And broadcast the new value of $\lambda_{t_{zone}}^k$ to all the associated zone level PAs and downstream PCAs (that is, task PCAs). Refer to sequence number 2 in the figure.
3. The task-level PCA updates the capacity value $L_k^*\left(\lambda_{t_{zone}}^k\right)$ by solving corresponding Eq. 3-38 for the task. It further computes the new value of $\lambda_{t_{task}}^k$ using corresponding Eq. 3-41 for the task in each instance $t_{task} \in \mathcal{T}_{task}$.
4. And broadcast the new value of $\lambda_{t_{task}}^k$ to all the associated task level PAs. Refer to sequence number 3 in the figure.
5. The task PAs updates the consumption value $x_i^{k*}(\lambda_{t_{task}}^k)$ by solving corresponding Eq. 3-37 for the PA and the updated $x_i^{k*}(\lambda_{t_{task}}^k)$ is communicated to the corresponding zone PCA. Refer to sequence number 4 in the figure.

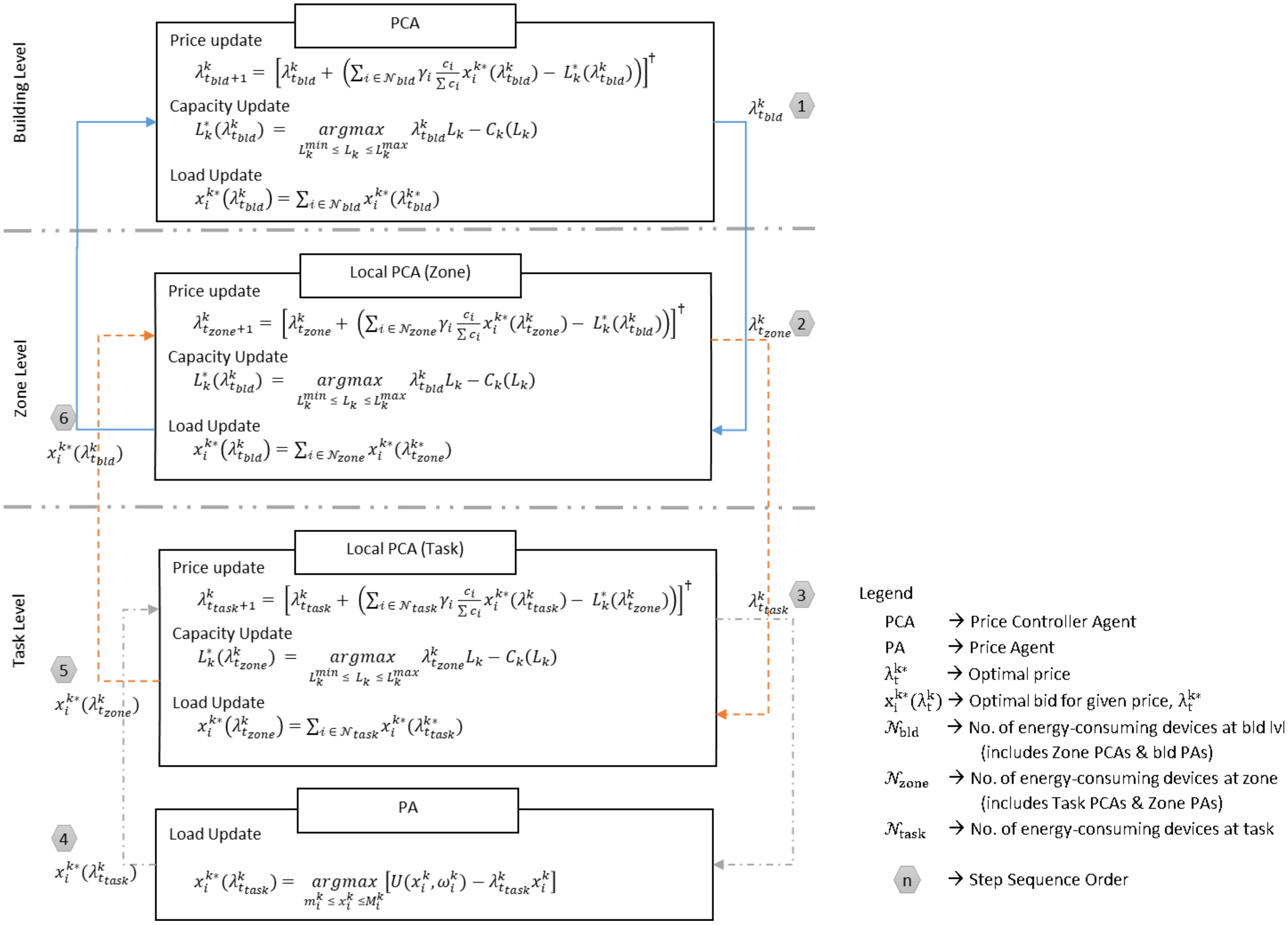


Figure 3-21: iSPACE – Hierarchical Distributed Multi-Agents Algorithm

6. The above steps 3 to 5 are repeated till an optimal bid, $x_i^{k*}(\lambda_{t_{zone}}^k)$ is computed by the task level PCA.
7. The task-level PCA communicates the optimal bid, $x_i^{k*}(\lambda_{t_{zone}}^k)$ to the zone PCA. Refer to sequence number 5 in the figure.
8. The above steps 2 to 6 are repeated till an optimal bid, $x_i^{k*}(\lambda_{t_{bld}}^k)$ is computed by the zone level PCA.
9. The zone level PCA communicates the optimal bid, $x_i^{k*}\left(\lambda_{t_{bld}}^k\right)$ to the building level PCA. Refer to sequence number 6 in the figure.
10. The above steps 1 to 8 are repeated till the market equilibrium is achieved, that is, an optimal price $\lambda_{t_{bld}}^{k*}$ is computed by the building level PCA
11. Finally, the building level PCA broadcasts the new optimal value of $\lambda_{t_{bld}}^{k*}$ to all the associated building level PAs and downstream PCAs (zone PCAs). The zone level PCAs, in turn, broadcast the new optimal value of $\lambda_{t_{bld}}^{k*}$ to all the associated zone level PAs and downstream PCAs (task PCAs). The task-level PCAs, in turn, broadcast the new optimal value of $\lambda_{t_{bld}}^{k*}$ to all the associated task level PAs.
12. All the PAs on receiving the new optimal value of $\lambda_{t_{bld}}^{k*}$, apply respective pricing policies.

This hierarchical structure ensures efficient utilization of available energy resources while maximizing the overall welfare of individual energy-consuming devices. By adopting such a distributed multi-agent system with a hierarchical structure, scalability and flexibility are achieved, allowing for easy integration and management of a large number of energy-consuming devices in complex environments like buildings.

3.4.3 Price and Energy Demand Functions

Each energy-consuming device solves local optimization problems. That is, compute the optimal energy consumption x^* for a given price λ based on each energy-consuming device utility function and operate the device accordingly to meet this energy consumption level (refer to section 3.4.1.A for details on demand side agent preference and utility function).

In this thesis, a slightly different approach is used. Building domain knowledge/information is employed to generate and implement simplified utility functions for the energy-consuming devices in the form of price and energy demand functions. We define ***Price Function (pf)*** and ***Energy Demand Function (edf)***. These functions are precomputed functions that are constrained by the comfort parameter ranges.

In buildings, primarily the energy-consuming devices (like HVAC, lighting, PECS) are controlled based on a process variable (setpoint) to provide personal comfort, occupant satisfaction, and productivity. The two important factors that affect these are Thermal comfort and Visual Comfort [12]. Building domain research focuses on models and control mechanisms, and exhaustive research and standards are available for the required operating setpoint ranges.

In general, the central HVAC system operating temperature setpoints are used for thermal comfort. The operating temperature is dependent on the air temperature, mean radiant temperature, and air speed. With a 50% humidity and 0.15 m/s mean velocity, the operating temperature setpoint range of 20.5 °C to 24.7 °C provides thermal comfort to 80% to 90% of the occupants [62]. This temperature setpoint range can be extended to 18 °C to 30 °C by providing personal comfort systems. While these temperatures may seem extreme, the personal comfort system aims to mitigate comfort locally, potentially making these extremes acceptable based on individual preferences and comfort requirements [60]. Similarly, dimming or illumination level setpoint ranges are defined for visual comfort. And surface temperature setpoint range of 24.5 °C to 27.0 °C for Radiant cubicles [90], [91]. Each energy-consuming device will choose its consumption level to maximize its welfare function introduced in (Eq. III-28). The welfare function is constrained by these comfort parameters operating setpoint ranges.

Static methods (analytical and empirical methods) are available to calculate the energy demand for various operating parameters like setpoint, external weather conditions, time of day, occupancy patterns, CO_2 levels, occupant's clothing [41], [59], [92]. Furthermore, dynamic methods like Artificial Neural Networks (ANN), Reinforced Learning (RL), and other machine learning techniques have been applied recently [45], [59], [93], [94].

Let us assume a monotonic function, ***Price Function, denoted by*** $\boldsymbol{pf}$, can be constructed that provides optimal operating setpoint for a given price for each energy-consuming device. And an ***Energy Demand Function, denoted by*** $\boldsymbol{edf}$**,** that computes required energy consumption level for various operating setpoints.

Let $\mathcal{N}$ denote the set of all energy-consuming device, where $N \stackrel{\text{def}}{=} |\mathcal{N}|$. It is assumed that the devices are independent of each other. We define *Price Function,* $pf_i(\lambda)$ for each energy-consuming device $i \in \mathcal{N}$ as:

$$pf_i(\lambda): \qquad r_i^* = a_i\lambda^2 + b_i\lambda + c_i \qquad \text{Eq. 3-42}$$

Where $a_i, b_i, and\ c_i$ are constants, λ is the price point, and r_i^* is the desired optimal process value or Setpoint (SP) with ranges defined by the above-discussed comfort constraints for energy-consuming device $i \in \mathcal{N}$.

And energy demand function, $edf_i(r_i)$ is defined as:

$$edf_i(r_i): \quad x_i = a_i {r_i}^2 + b_i r_i + c_i \qquad \text{Eq. 3-43}$$

Where a_i, b_i, c_i are constants, x_i^* is the optimal energy demand of the device $i \in \mathcal{N}$ for a given desired optimal process value or Setpoint (SP) ($r_i^{\square}$) for the slot time.

Substituting Eq. 3-42 in Eq. 3-43, we have:

$$x_i^* = edf_i(pf_i(\lambda)) \qquad \text{Eq. 3-44}$$

The Eq. 3-44 is analogous to the welfare optimisation introduced in Eq. 3-15 for an energy-consuming device but with added comfort constraints.

Application in the Case Study: For example, for the case study implementation, it is desired that the ambient air conditioning, which is central HVAC system, usage is maximized, and the Radiant Cubicle cooling (a PECS) usage is minimized at lower price points. The zone set points typically rise with a price increase. Consequently, the radiant cubicle offsets the comfort decline at the zone level by lowering the cooling set point when prices rise.

For such a scenario, the price functions used in the case study are shown in Figure 3-22 for zone AC and Radiant Cubicle. The line (blue-round marker) represents the price function for Zone AC, the zone operative temperature setpoint for a given price point as price increases. The dashed lines (blue-round, orange-square, and grey-diamond markers) represent the radiant cubicle surface temperature setpoints for different prices for various occupant comfort preferences (vis-e-vis, cool, normal, and warm); setpoint reduces as a price increase. The corresponding price function is used for the radiant cubicle surface temperature setpoint based on the occupants' comfort preference.

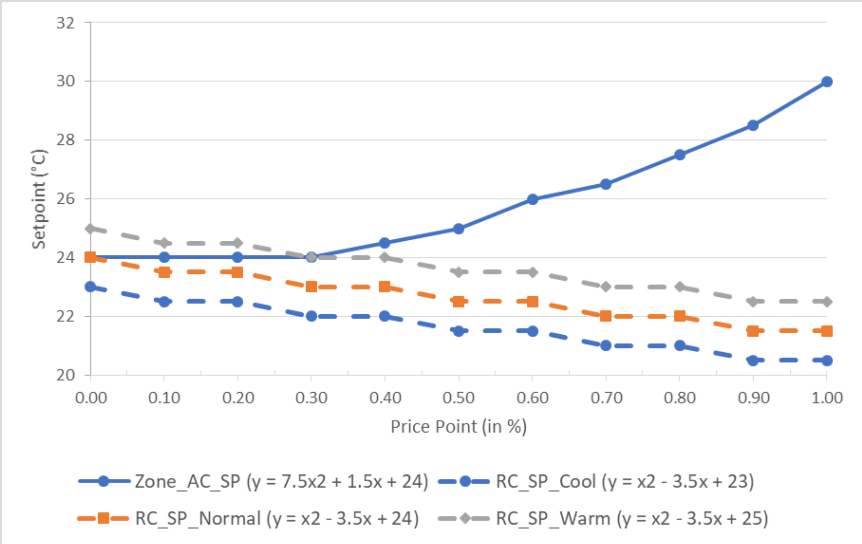


Figure 3-22: An example Price Functions for Zone AC and Radiant Cubicles

Note: As detailed in agents' configuration section 5.5, these configurations are a means to test the efficacy of the framework, and the efficacy of these parameters is not studied as part of the thesis. However, these functions were established by carefully analysing the measurement data obtained during FDD Lab benchmark trial runs. In practise, the price and energy demand functions must be precomputed and provided to the framework.

Besides, the energy demand functions were empirically constructed based on the experimental data. Zone-1 and Zone-2 were subjected to random price points for 51 hrs for the above price functions. The cooling energy consumption was measured with BTU meters, and the corresponding electrical energy equivalent was computed (with COP 2.29 for the primary circuit, that is, chiller. Refer to Figure 5-8 for details). The electrical energy consumption of Secondary Pumps and Air Handling Units fans (AHU) was measured using WATT Nodes. The total electrical energy is computed as the sum of the cooling-energy electrical equivalent of AHU and secondary pump. The randomness of price point was measured using *runstest*[8] and an h=0 and p=1.0 were observed. The Setpoint Vs Energy Demand were plotted for Zone AC (Figure 3-23) and RadiantCubicle (Figure 3-24). The

[8] The 'runstest' is a MATLAB function utilized to assess randomness. It employs a test statistic computed as the difference between the number of runs and its mean divided by its standard deviation. This test statistic follows a normal distribution when the null hypothesis is true. A returned value of 'h = 0' indicates that the 'runstest' does not reject the null hypothesis, suggesting that the values in 'x' is in random order at the default 5% significance level. The p-value of the test, denoted as 'p', is a scalar value within the range [0, 1]. A small p-value suggests doubt regarding the validity of the null hypothesis. For further details, refer to the MATLAB documentation at the following link: runstest - MATLAB Documentation.

energy demand functions were developed by curve fitting. The R-square value of Zone AC and RadiantCubicle equation was found 0.96 and 0.95, respectively.

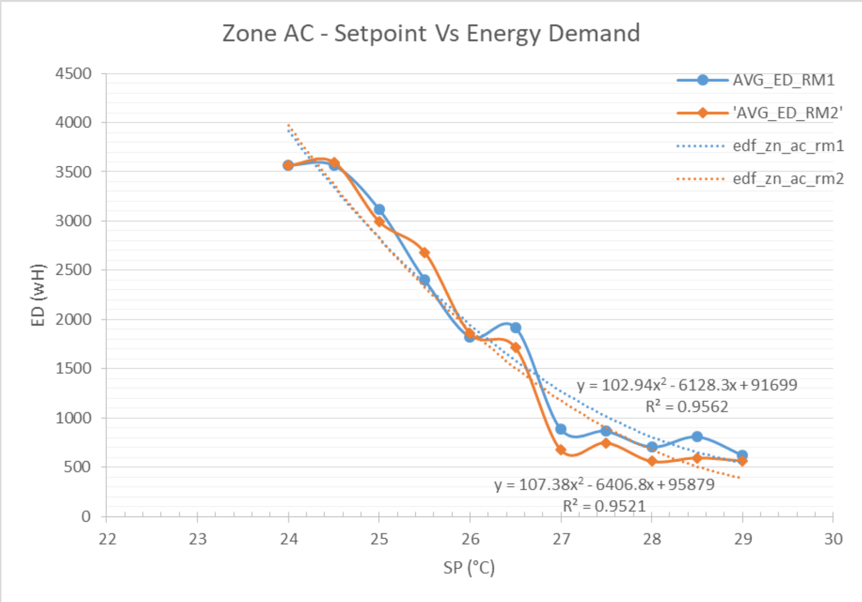


Figure 3-23: An example Energy Demand Function for Zone AC

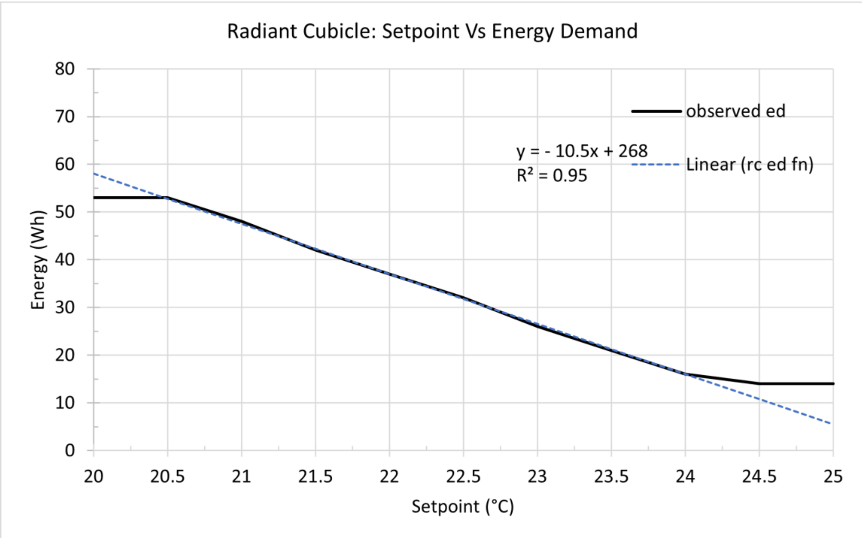


Figure 3-24: Radiant Cubicle Energy Demand Functions

The corresponding price and energy demand functions are used based on user preferences (vis-a-vis Cool, Normal, and Warm). These functions allow for a more streamlined and efficient way to control the energy consumption of devices in a building, ensuring optimal performance while adhering to comfort parameters. It provides a bridge between the energy price and the energy demand, ensuring the comfort of the occupants and optimizing energy consumption.

However, there are a few challenges to the methodology outlined in section 3.4:

- **Convergence Speed:** The iterative nature of the algorithm could lead to slow convergence in specific scenarios, especially when if the initial conditions are far from the equilibrium.
- **Complexity**: The hierarchical structure, while providing scalability, can also add complexity, especially when the number of levels (building, zone, task) is high.
- **Communication Overhead:** The constant exchange of information between various levels can lead to significant communication overheads, which might strain the network infrastructure, especially in large setups.

Despite these challenges, this approach offers a methodology for a robust framework to integrate various energy-consuming devices effectively and optimize their operations in real time. The construction of price and energy demand functions for the demand side, substituting utility functions for energy-consuming devices, a novel aspect of the thesis, streamline the process. And constructing price functions for different energy-consuming devices intuitively seems more practical. Similarly, energy demand functions can be derived using analytical or statistical methods. Notably, simulations, encompassing over a million test runs with random initial states at all levels, suggest that convergence within the desired maximum number of iterations is attainable.

3.5 Selection of MAS Platform

Choosing the right platform is critical for effectively implementing the iSPACE system and its framework. It necessitates a platform that facilitates reliable and secure two-way communication among agents, accommodating various communication protocols, hardware, and software. Additionally, efficient data management capabilities are imperative due to the system's extensive data points. The ideal platform should seamlessly integrate distributed processing with centralised control to support large-scale implementations. In this subsection, we explore existing transactive platforms in the literature. After careful consideration, we identify Eclipse VOLTTRON™ as the most suitable platform for our requirements.

Drawing from Bellifemine et al. [95], irrespective of the domain, certain MAS characteristics remain consistent. Core among these is the need for software infrastructure to underpin agent communication, MAS platforms or frameworks. These platforms serve as the bedrock, supporting agent-to-agent interactions, ensuring they transpire seamlessly. The Foundation for Intelligent Physical Agents (FIPA) develops and sets computer software standards for heterogeneous and interacting agents and agent-based systems. A primary result of FIPA's efforts has been specifications to facilitate agent operations between different agent middleware. The most widely adopted FIPA standards are the Agent Management and Agent Communication Language (FIPA-ACL). Some of the essential elements that were specified are shown in Figure 3-25.

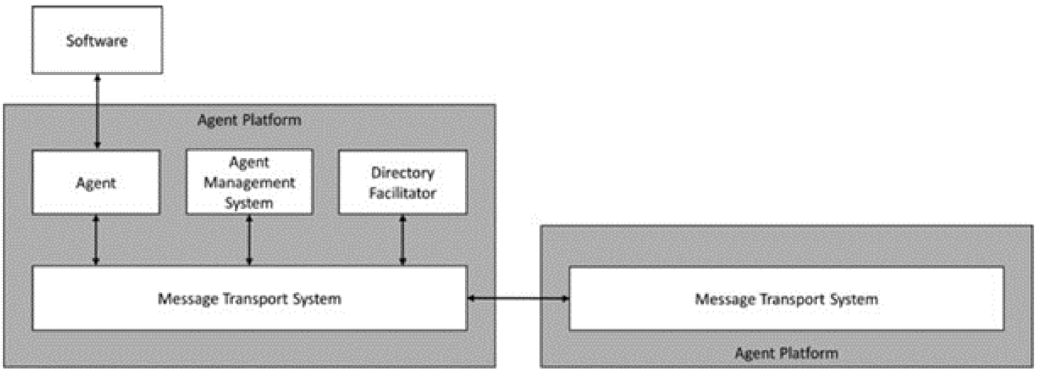


Figure 3-25: Basic intended structure for FIPA-compliant MAS frameworks from (Windham and Treado 2016)

The agent communication language (ACL) standardises a fundamental set of communicative tools that carry specific meaning and protocols between agents. The directory facilitator (DF) acts as “yellow pages” for agents to look up other agents and observe their capabilities. The agent management system (AMS) is responsible for sending and receiving ACL messages. The message transport service transports (MTS) messages between agents and platforms.

In the world of Multi-Agent Systems (MAS), several platforms have been developed to facilitate agent interaction and behaviour in various domains. Some of the most widely used open-source MAS platforms include: 1) JADE (Java Agent DEvelopment Framework), 2) SPADE (Smart Python Agent Development Environment), 3) Eclipse VOLTTRON™, 4) AgentFactory, and 5) SeSAm (Shell for Simulated Agent Systems). Refer to Table 3-4 for a brief comparison across a set of parameters.

The ideal MAS platform depends on the specific needs and constraints of the project. The decision should align with the application domain, required features, and available expertise. If FIPA compliance is a strict necessity, then JADE or SPADE would be preferable. For those looking to avoid programming, SeSAm offers a unique visual approach. However, for energy-related applications, especially with a focus on building energy efficiency and grid services , Eclipse VOLTTRON™ is the most specialized and suited for the current work.

Table 3-4: Comparison of MAS platforms

Sl. No.	Platform	Origin/ Developer	Open Source	FIPA Compliant	Pgm. Lang.	Intended Use/Application Domain	Special Features	Platform Requirements	Community Support
1	JADE (Java Agent DEvelopment Framework)	Telecom Italia Lab	Yes	Yes	Java	General-purpose, but widely used in telecoms and networking.	Provides GUI to monitor and manage agents. Supports mobile agents that can move across different hosts.	Java-based, so it is platform-independent.	Robust community with a plethora of available resources.
2	SPADE (Smart Python Agent Development Environment)	University of Murcia	Yes	Yes	Python	General-purpose, with emphasis on easy agent development.	Built-in support for several agent protocols. Integrates with XMPP (Jabber) for agent communication.	Python-based, making it highly portable.	Active development community.
3	Eclipse VOLTTRON ™	Pacific Northwest National Laboratory (PNNL)	Yes	Strictly Not but designed with similar concepts.	Python	Initially for the power grid, later extended to buildings and integration with the grid.	Distributed control and sensing platform with a focus on energy efficiency and grid	Linux environment.	Backed by PNNL and the U.S. Department of Energy

Sl. No.	Platform	Origin/ Developer	Open Source	FIPA Compliant	Pgm. Lang.	Intended Use/Application Domain	Special Features	Platform Requirements	Community Support
							services.		with good community support.
4	AgentFactory	University College Dublin	Yes	Yes	Java and C#	General-purpose with a focus on pervasive computing.	Emphasizes the creation of intelligent agents and supports several reasoning engines.	Java and .NET platforms.	Active academic community.
5	SeSAm (Shell for Simulated Agent Systems)	University of Würzburg	Yes (with restrictions for commercial use)	Partial	Proprietary visual language	For the simulation of agent-based models, particularly in the domain of environmental science.	Visual interface for agent modelling without requiring coding, suitable for non-programmers.	Windows OS.	Specific community in the environmental and simulation domain.

To Summarise: The Potential of Eclipse VOLTTRON™

Among the various platforms, Eclipse VOLTTRON™, emerges as a beacon of promise. While its adherence to FIPA is yet to be ratified, its unique attributes render it compelling:

1. **Pedigree**: Developed under the auspices of Pacific Northwest National Laboratory (PNNL) and financially backed by the U.S. Department of Energy (DOE), its lineage speaks volumes.
2. **Open Architecture**: As an open-source platform, it rides on the wave of collective wisdom, benefiting from a plethora of contributions.
3. **Versatility**: It was conceived initially for the power grid. Nevertheless, its adaptability has seen it find application in buildings and their integration with the grid.
4. **Rich Ecosystem**: The platform brims with components and agents that provide diverse services.

The Eclipse VOLTTRON™ platform comprises many components and agents that provide services to other agents. Figure 3-26 shows the various components of the platform. Several key components and service agents related to this work are described below, while more detailed descriptions of the platform can be found in its online documentation.

Key Features:

- **Information Exchange Bus:** A robust messaging system for agents to share and receive data.
- **Remote Procedure Calls (RPCs):** For direct agent-to-agent interactions, particularly beneficial for event-driven tasks.
- **Drivers:** Built-in support for widely used building automation system protocols

such as BACnet and Modbus.

- **Actuator Agent:** Facilitates device control through agent requests.
- **Message Bus:** Promotes inter-agent communication using a publish/subscribe mechanism.

The platform has low CPU, memory, and storage requirements on small-form-factor computers such as the Raspberry Pi and Intel Edison. These small-form-factor computers provide cost-effective processing platforms for the system. Supporting community and developers has produced different applications for this platform, including HVAC fault diagnosis agents, Smart home appliances control agents, Electric vehicle charging station control, Power grid communication, and many more. For this research work, Eclipse VOLTTRON™ 5.1.0 version has been used.

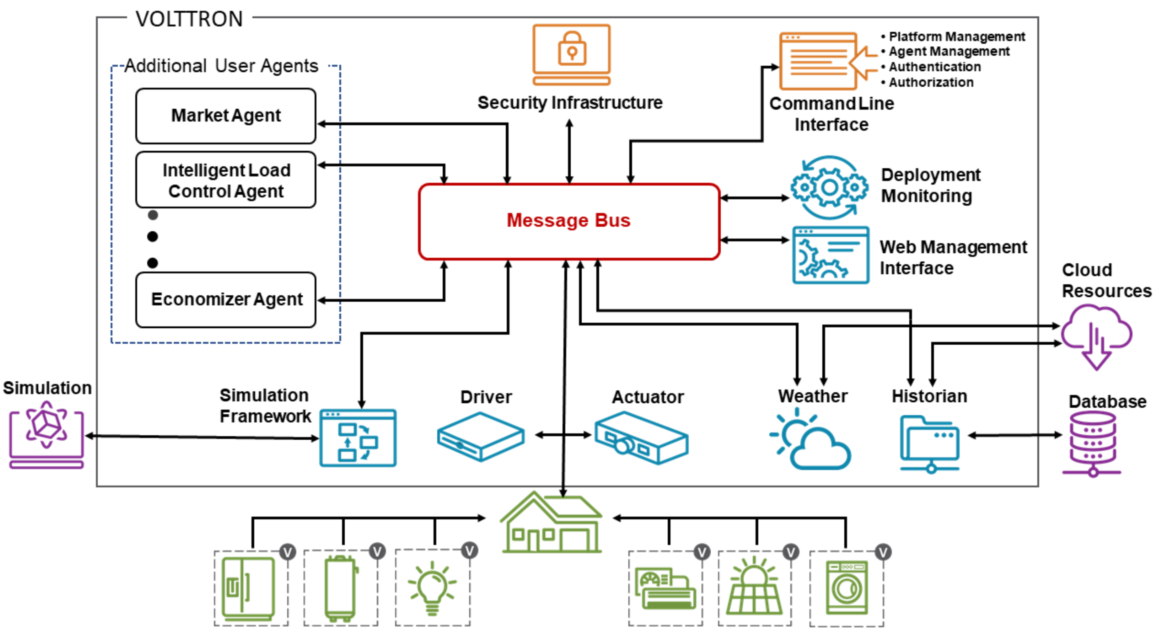


Figure 3-26: Various components of the Volttron platform

A summary of Volttron performance metrics based on the available literature is as follows:

1. **Scalability:** Volttron is designed to be scalable, supporting numerous devices and agents in building and grid environments [96], [97].

2. **Latency:** Given its design for real-time energy systems, latency is low. However, exact figures would depend on the deployment scenario, network conditions, and the number of concurrently active agents.

3. **Throughput:** The message bus in Volttron is built on the ZeroMQ Python library, which provides high throughput for message communications. The pub/sub pattern used by the platform ensures efficient data sharing among agents [98].

4. **Memory Usage:** Being Python-based, Volttron may have a moderate memory footprint. However, it's optimized to run on small-form-factor computers like the Raspberry Pi, indicating its lightweight design [33].

5. **Development Tools:** Volttron is open-source and has tools and documentation available for development. Its Python foundation allows developers to leverage the vast array of libraries available in the Python ecosystem.

6. **Community Support:** Volttron benefits from institutional backing by the Pacific Northwest National Laboratory (PNNL) and the U.S. Department of Energy (DOE), ensuring robust support. Additionally, its recent integration into the Eclipse Foundation has expanded its community reach, fostering specialised growth and a broad user base.

7. **Security:** Volttron has a strong focus on security, given its application in critical energy systems. The platform supports secure communications and has mechanisms to ensure the reliability and authenticity of agent communications.

3.6 Conclusion

Chapter 3 introduces the iSPACE system as a step towards energy efficiency and integrated personal ambient controls. The iSPACE system creates an energy-efficient and highly personalized environment for individual occupants. The robust framework, the system's communicative aspect (iSPACE Messages), and iSPACE Agents ensure the system remains dynamic, adaptive, and efficient. Introducing price and energy demand functions, along with the system model and pricing formulations presented in this thesis, allows iSPACE to adapt in real-time. This adaptability uses transactive controls to focus on demand response management and occupant comfort at the task level.

Chapter 4

Development of PECS (SmartHub, SmartStrip, and RadiantCubicle)

"Strive for perfection in everything you do. Take the best that exists and make it better. When it does not exist, design it." — Sir Henry Royce

4.1 Introduction

The three PECS that have been designed and developed to deploy a pilot setup, as mentioned in section 1.4.2, are detailed in this chapter.

4.2 SmartHub

The SmartHub is a controller with a mobile user interface at the task level, integrating personal environment control systems with the ambient control systems and allowing occupants to participate in the demand response. Suppose other personal environment control systems like foot warmers, personal radiant cooling/heating systems, heating/cooling chairs, and others can communicate over either BLE, BACnet or Modbus TCP/IP. In

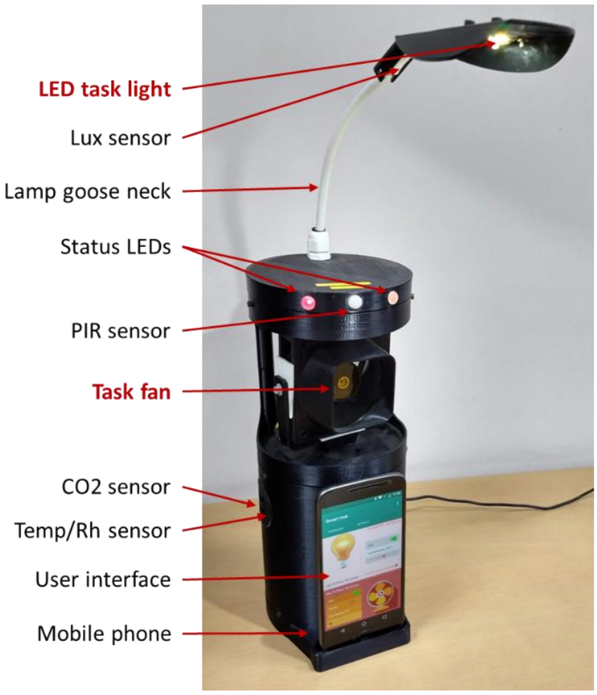


Figure 4-1: SmartHub with user interface

that case, the SmartHub can interact with them and manage the conditioning at the task level. The user interface to SmartHub has been provided through an app on a mobile phone that communicates over BLE for real-time control, alerts, and data visualisation. Besides, the app can provide historical data views over Wi-Fi data connection.

The SmartHub hardware has been developed and is ready for use. Besides, the core agents needed for the system has been developed.

4.2.1 SmartHub Hardware

The current SmartHub that has been developed has a built-in fan (40cfm, 1.8W, 19dB-A), a LED lamp (150 lumens, 10W), an array of sensors (Lux, Temperature, Rh, PIR, and CO_2). The SmartHub is powered by a computer-on-module (Intel Edison) with a 500 MHz CPU and 1GB RAM. The computer-on-board module also has one 100 MHz micro-controller, 1GB RAM, 4 GB eMMC flash, Wi-Fi, and Bluetooth Low Energy (BLE) on-board modules (Figure 4-2) and runs Yacto Linus OS.

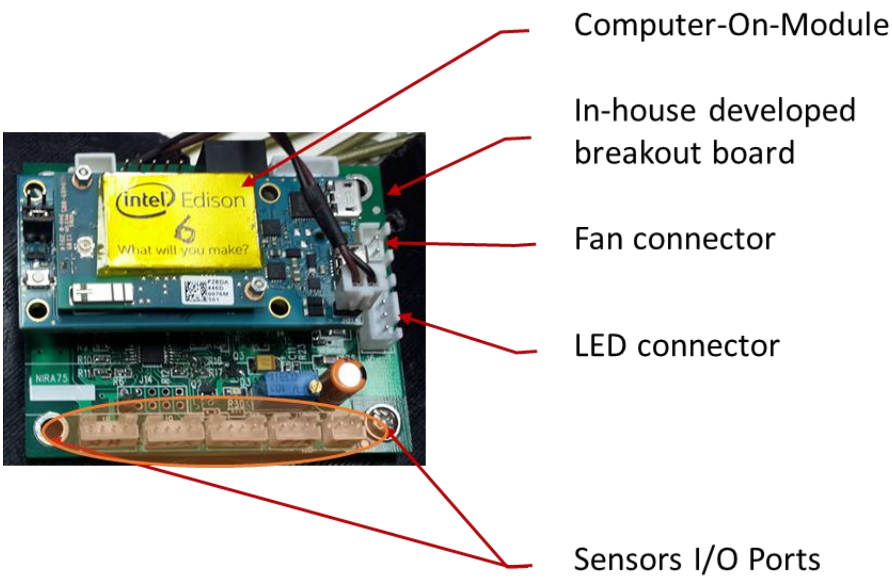


Figure 4-2: SmartHub internal, H/W Top View

Besides, the SmartHub also has a built-in battery charge controller with 6600 mAh power. Moreover, an in-house I/O breakout board has been developed (Figure 4-3) to decouple the

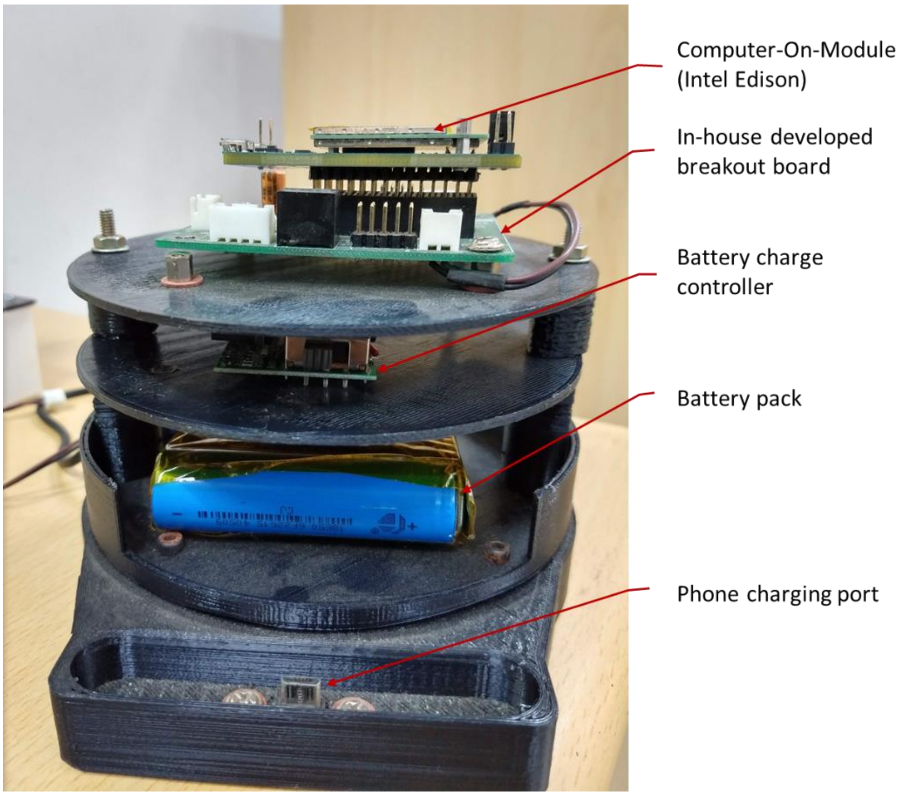


Figure 4-3: SmartHub internal, H/W Front View

dependency on any computer-on-module. PCB design is provided in Appendices A-4.

The Fan can be controlled for speed and the Light is dimmable. These can be controlled based on price or manually by user through mobile app interface. Also, they are switched on-off based on threshold price.

The board also has a 5V and 12V regulated supply to drive sensors like CO2; a 10W LED constant current driver IC to control the light intensity through a PWM control signal; a 5W PWM driver to control the fan speed. For details, may refer to the Appendices A-5 for the schematic.

4.2.2 SmartHub Software

The key software modules that have been developed are as follows:

A. **BACnet Server**

The devices available in the SmartHub (fan, light, & sensors) are interfaced through a BACnet server app running in the SmartHub. This app enables access to these devices as BACnet objects. The list of objects is as detailed in Table 4-1.

B. **Transactive Platform**

The SmartHub has a transactive platform, Eclipse VOLTTRON™ (detailed in the section 3.5), instance running on it. This platform provides tools for the development of necessary agents. The instance registers itself with the respective Zone's transactive platform instance to receive the price and update its energy demands to the zone.

A few of the key agents that have been developed are as follows:

a. Price Controller Agent,
b. SmartHub Price Agent, and
c. SmartHub UI client agent.

C. **UI Server (Gateway)**

A UI server in node.js has been developed that act as a gateway between the transactive platform (TCP/IP) and the mobile phone (BLE).

Table 4-1: SmartHub BACnet Objects.

Sl. No.	Reference Point Name	Volttron Point Name	Units	Unit Details	BACnet Object Type	Writable	Index	Notes
1	ANALOG INPUT 0	SensorLux	lux	-	analogInput	FALSE	0	Lux Sensor
2	ANALOG INPUT 1	SensorRh	percent	-	analogInput	FALSE	1	Rh Sensor
3	ANALOG INPUT 2	SensorTemp	celcius	-	analogInput	FALSE	2	Temperature Sensor
4	ANALOG INPUT 3	SensorCO2	ppm	-	analogInput	FALSE	3	CO2 Sensor
5	ANALOG OUTPUT 0	LEDPwmDuty	percent	-	analogOutput	TRUE	0	LED PWM Duty
6	ANALOG OUTPUT 1	FanPwmDuty	percent	-	analogOutput	TRUE	1	Fan PWM Duty
7	BINARY INPUT 0	SensorOccupancy	Enum	0-1	binaryInput	FALSE	0	Occupancy Sensor
8	BINARY OUTPUT 0	SmartHub	Enum	0-1	binaryOutput	TRUE	0	Smart Hub Init/Deinit
9	BINARY OUTPUT 1	LEDDebug	Enum	0-1	binaryOutput	TRUE	1	LED for Debug
10	BINARY OUTPUT 2	LED	Enum	0-1	binaryOutput	TRUE	2	LED On/Off
11	BINARY OUTPUT 3	Fan	Enum	0-1	binaryOutput	TRUE	3	Fan On/Off
12	BINARY OUTPUT 4	FanSwing	Enum	0-1	binaryOutput	TRUE	4	FanSwing On/Off

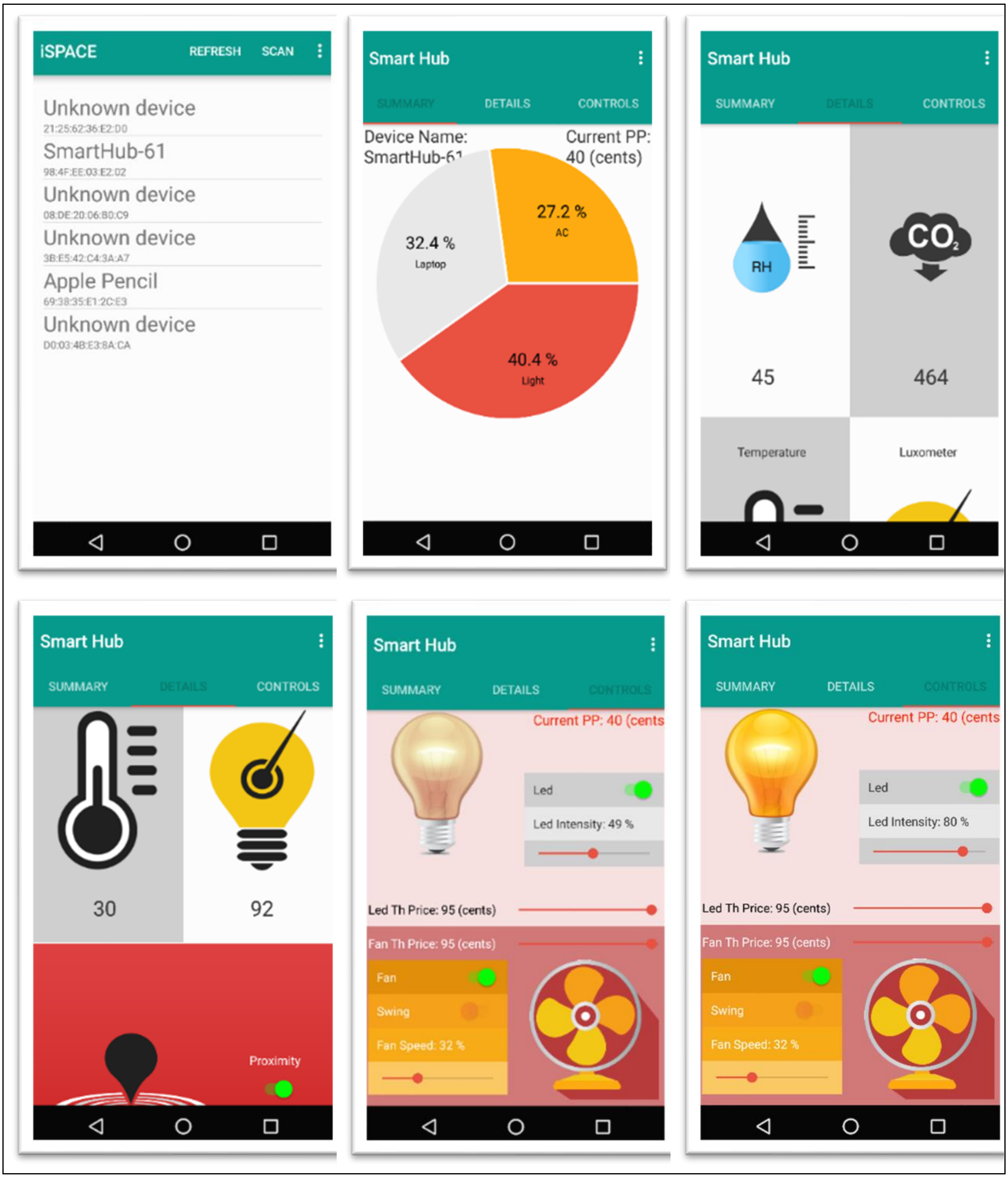


Figure 4-4: SmartHub Mobile App UI screenshots

D. Mobile Application

For providing a user interface for the user to interact with the system and participate in the demand response management at the user level, an Android-based mobile app has been developed. The mobile phone app communicates with SmartHub over BLE. Currently, it supports manual control of the task led and task fan (on/off & intensity) by the user. Besides, it can also view the current price point and change the threshold price point of the task led and task fan. Also, the app displays the instantaneous sensor's data (Lux, Temperature, Rh, PIR, and CO2). Some of the UI screenshots are shown in (Figure 4-4). A gateway agent has been developed to enable mobile apps to exchange data with Eclipse VOLTTRON™ Message Bus.

E. Bridge Agent

The details about the role of the Bridge agent are provided in iSPACE Agents section 3.3.2.C. The primary role this agent to maintain a registry of the all the local price agents (energy consuming devices) at the task level and any downstream price controller agent and can also transfer messages from one message bus of one instance to another message bus of a different machine/instance has been developed. Besides, a gateway agent has been developed to enable mobile apps to exchange data with message bus.

4.2.3 Design Challenges

A. LED dome

The design of the LED dome presented unique challenges. The desire to achieve a lightweight design necessitated the exploration of materials like copper, which boasts superior thermal conductivity. However, the laser cutting process, ideal for precision design, is incompatible with copper. Therefore, we pivoted to aluminium, balancing weight, heat dissipation, and manufacturing feasibility.

B. 3D printed shell

3D printing technologies provided the flexibility and precision needed to design the shell. However, this method posed challenges related to material selection and the positioning of sensors and components. Through numerous trials and design adjustments, we ensured the 3D-printed shell was durable for daily use and optimized for component placement.

C. Fan

The fan is a critical component for air circulation within the PECS. For purpose of Fan used in the PECS, there are specific specifications. Off-the-shelf fans did not meet the specific profile and specifications needed for our design. Thus, we had to be creative, selecting a CPU fan that met the performance and form factor requirements.

D. Lux sensor position

Determining the optimal position for the Lux sensor was pivotal to ensuring accurate light measurement. Given the sensor's sensitivity and the potential for interference from other components, its placement required careful consideration.

E. Mobile phone docking

The diversity of mobile phone connectors, from Micro USB to USB-C, posed a challenge in designing a universal docking mechanism for the SmartHub. Considering future shifts in phone connector standards, a solution had to be devised to ensure compatibility with most phones.

Developing the SmartHub involved navigating through a maze of design and technical challenges. Each hurdle necessitated innovative solutions, underlining the significance of adaptability in product development. The iterative process, marked by trials, errors, and triumphs, has culminated in a practical and useable product as a desktop personal environment control system.

4.3 SmartStrip

SmartStrip is designed to enhance energy efficiency and control the plug loads, particularly in offices and commercial buildings. This intelligent power strip integrates advanced sensing, control, and communication technologies to optimize energy usage and provide personalized control over connected plug loads. SmartStrip has multiple outlets, each independently controllable and monitored for energy consumption. Keeping in mind the various pin configurations, a modular approach was used. By changing the plug receptacles, the SmartStrip could be adopted to any pin configurations. These outlets can accommodate various electrical devices such as computers, monitors, and other PECS like footwarmers. By enabling individual control over each outlet, SmartStrip allows to selectively power on or off specific devices based on their usage patterns and preferences, thereby reducing unnecessary energy consumption. Some SmartStrips have metering ICs to monitor voltage, current, and power consumption parameters. Besides, some of these ICs provide enhanced capabilities to monitor the harmonics and others, which advanced AI algorithms can use for plug load identification.

Hence, a typical SmartStrip, including the necessary hardware and the core agents needed for the system, was developed and deployed in the pilot setup to demonstrate the efficacy of the iSPACE system.

This integration enables centralized control and monitoring of multiple plug loads within a task, facilitating coordinated energy-saving strategies and providing insights into energy usage patterns for further optimization.

The SmartStrip with the following specifications and features was developed:

- Electrical Specification:
  - Input Voltage Range: 90-240VAC.
  - Input Current (full load) at 115/230VAC: 85mA/40mA.
  - Rated Power(max): 3 Watts.
  - Input Frequency Range: 47-63 Hz.
- Component and Subsystem Details:
  - Intel® Atom™ processor Z34XX, 500MHz Operating Frequency.
  - 4 GB NAND Flash.
  - 1 GB LPDDR3.
  - Dual-Band Wi-Fi IEEE 802.11 a/b/g/n and Bluetooth 4.0.
- Operating System:
  - Yacto Linux.
- Features:
  - 90-240VAC/6A software-controllable dual sockets for loads.
  - Load identification using Unique 64bit ChipOnPlug technology.
  - Separate energy monitoring SoC (78M6610+PSU) to measure.

Voltage	Active Power
Current	Reactive Power
Frequency	Harmonics
Power Factor	

4.3.1 Hardware

The SmartStrip (Figure 4-5 and Figure 4-6) comes equipped with four 3A / 240V plugs. By altering the plug receptacles, it is compatible with a 120V supply, making it suitable for the US market. Internally, the SmartStrip utilizes Latching Relays to modulate the power supply to the plugs. Moreover, an innovative two-pin mechanism detects the GUID embedded inside an EPROM chip connected to each plug. PCB design is provided in Appendices A-5.

Figure 4-5: SmartStrip with plugs connected

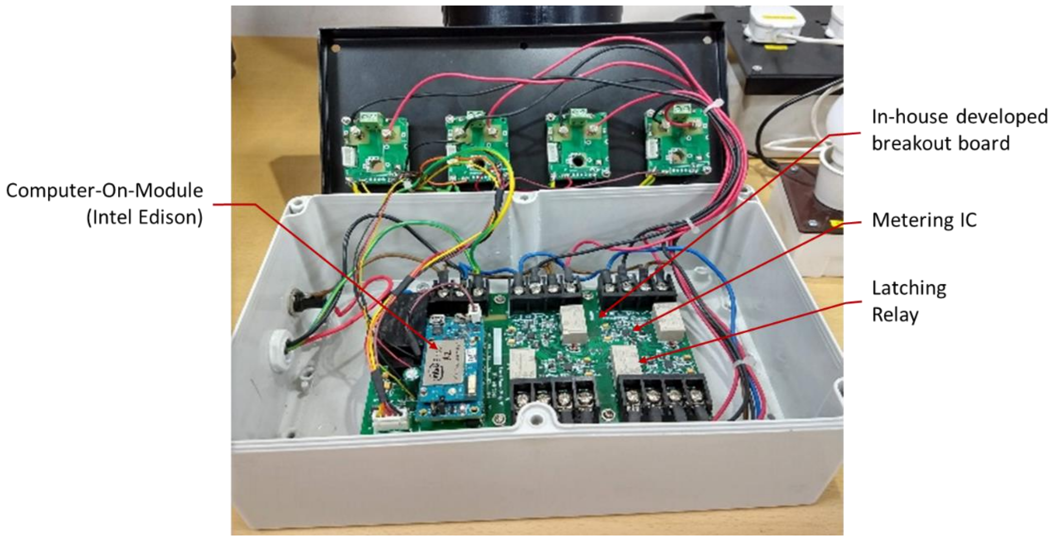


Figure 4-6: SmartStrip internal hardware

The device's core is powered by a computer-on-module (Intel Edison), boasting a 500 MHz

CPU and 1GB RAM. Additionally, this module incorporates a 100 MHz microcontroller, 1GB RAM, 4 GB eMMC flash, Wi-Fi, and Bluetooth Low Energy (BLE) onboard modules running on the Yacto Linus OS.

4.3.2 Software

A BACnet server app is implemented that exposes all the available functions of the SmartStrip as BACnet objects. The list of objects is as detailed in Table 4-2. A SmartStrip Price Agent running in the SmartHub controls the SmartStrip. The control logic of the SmartStrip Price Agent is as shown in Figure 4-7.

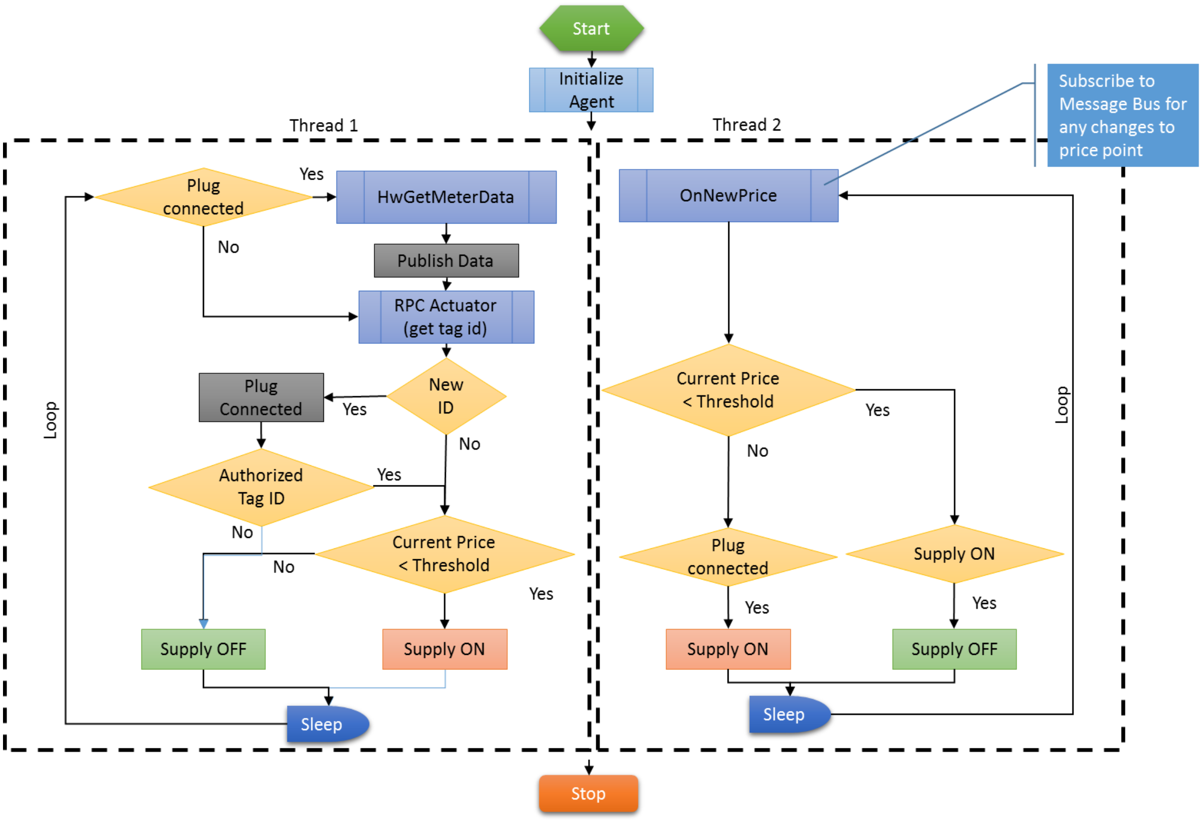


Figure 4-7: SmartStrip Control Logic

4.3.3 Design Challenges

A. <u>Chip-on plug</u>

The initial plan was to harness existing plug load identification algorithms from the literature. However, an evident scarcity of algorithms fit for power supply fluctuations, especially like those in India, prompted a switch to the Chip on Plug technique. Here, each connected load plug is paired with an EPROM-based chip-on-plug module. An additional two-pin mechanism in each plug of the SmartStrip reads the unique 64-bit GUID embedded in the EPROM chip.

A noteworthy challenge was the BACnet protocol's data point size restriction of 32 bits. This restriction necessitated dividing the plug ID across two data points, which would later be merged at the receiving end. Furthermore, connectivity glitches arose between the chip-on plug and the two-pin mechanism, especially with bulkier laptop adapters.

B. Electrical safety

The inclusion of the chip-on-plug technique demanded the design of custom 3D receptacles. A conspicuous absence of standard enclosures for these custom receptacles led to the creation of tailor-made ones. The initial top cover for these receptacles was fashioned from MS sheets using laser cutting. However, it faced rejection for testing by our US counterpart due to electrical safety regulations. Accompanying concerns like unlisted UL/CSA fuse holders and an ungrounded enclosure prompted a redesign of the SmartStrip, incorporating the provided feedback.

Figure 4-8: Few more photos of Smartstrip design iterations

Figure 4-9: Comparative images of the Indian and US plug respectale versions of the SmartStrip.

Table 4-2: A detailed tabulation of SmartStrip BACnet Objects

Sl. No.	Point Name	Volttron Point Name	Units	Unit Details	BACnet Object Type	Writable	Index	Notes
1	BINARY OUTPUT 1	SmartStrip	Enum	0-1	binaryOutput	TRUE	0	Smart Strip Init/Deinit
2	BINARY OUTPUT 2	Plug1Relay	On / Off	0-1	binaryOutput	TRUE	1	Plug 1 Relay On/Off
3	BINARY OUTPUT 3	Plug2Relay	On / Off	0-1	binaryOutput	TRUE	2	Plug 2 Relay On/Off
4	BINARY OUTPUT 4	LEDDebug	On / Off	0-1	binaryOutput	TRUE	3	LED for Debug
5	BINARY OUTPUT 5	Plug3Relay	On / Off	0-1	binaryOutput	TRUE	4	Plug 3 Relay On/Off
6	BINARY OUTPUT 6	Plug4Relay	On / Off	0-1	binaryOutput	TRUE	5	Plug 4 Relay On/Off
7	ANALOG INPUT 0	TagID1_1	noUnits	-	analogInput	FALSE	0	Plug 1 Tag ID 1_1
8	ANALOG INPUT 1	TagID1_2	noUnits	-	analogInput	FALSE	1	Plug 1 Tag ID 1_2
9	ANALOG INPUT 2	TagID2_1	noUnits	-	analogInput	FALSE	2	Plug 2 Tag ID 2_1
10	ANALOG INPUT 3	TagID2_2	noUnits	-	analogInput	FALSE	3	Plug 2 Tag ID 2_2
11	ANALOG INPUT 4	Plug1Voltage	volts	-	analogInput	FALSE	4	Plug 1 Voltage
12	ANALOG INPUT 5	Plug1Current	amperes	-	analogInput	FALSE	5	Plug 1 Current
13	ANALOG INPUT 6	Plug1ActivePower	watt	-	analogInput	FALSE	6	Plug 1 Active Power
14	ANALOG INPUT 7	Plug2Voltage	volts	-	analogInput	FALSE	7	Plug 2 Voltage

Sl. No.	Point Name	Volttron Point Name	Units	Unit Details	BACnet Object Type	Writable	Index	Notes
15	ANALOG INPUT 8	Plug2Current	amperes	-	analogInput	FALSE	8	Plug 2 Current
16	ANALOG INPUT 9	Plug2ActivePower	watt	-	analogInput	FALSE	9	Plug 2 Active Power
17	ANALOG INPUT 10	TagID3_1	noUnits	-	analogInput	FALSE	10	Plug 3 Tag ID 3_1
18	ANALOG INPUT 11	TagID3_2	noUnits	-	analogInput	FALSE	11	Plug 3 Tag ID 3_2
19	ANALOG INPUT 12	TagID4_1	noUnits	-	analogInput	FALSE	12	Plug 4 Tag ID 4_1
20	ANALOG INPUT 13	TagID4_2	noUnits	-	analogInput	FALSE	13	Plug 4 Tag ID 4_2
21	ANALOG INPUT 14	Plug3Voltage	volts	-	analogInput	FALSE	14	Plug 3 Voltage
22	ANALOG INPUT 15	Plug3Current	amperes	-	analogInput	FALSE	15	Plug 3 Current
23	ANALOG INPUT 16	Plug3ActivePower	watt	-	analogInput	FALSE	16	Plug 3 Active Power
24	ANALOG INPUT 17	Plug4Voltage	volts	-	analogInput	FALSE	17	Plug 4 Voltage
25	ANALOG INPUT 18	Plug4Current	amperes	-	analogInput	FALSE	18	Plug 4 Current
26	ANALOG INPUT 19	Plug4ActivePower	watt	-	analogInput	FALSE	19	Plug 4 Active Power
27	ANALOG OUTPUT 0	PricePoint	-	-	analogOutput	TRUE	0	Price Point

4.4 Radiant Cubicle

A radiantly cooled PECS system, "*RadiantCubicle*", introduced in the previous chapter (Figure 3-3) uses radiant cooling. RadiantCubicle is an innovative personal environment control system (PECS) designed to enhance occupant comfort and energy efficiency within indoor environments [99]. At its core, RadiantCubicle leverages radiant heating and cooling techniques to regulate the temperature of the occupant's immediate surroundings, providing targeted thermal comfort without significantly impacting the overall ambient conditions of the surrounding space and aims to optimize energy usage while ensuring personalized comfort levels by focusing on the occupant's microclimate [91], [100].

Hence, a typical RadiantCubicle, including necessary setup and the core agents needed was developed as shown in Figure 4-10 and deployed in the pilot setup to demonstrate the efficacy of the iSPACE system.

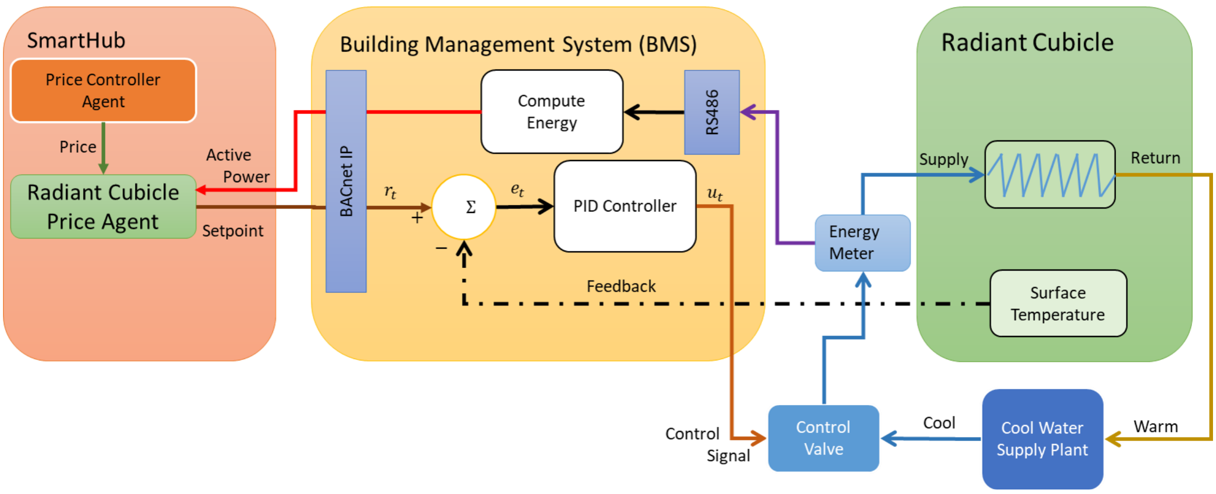


Figure 4-10: Radiant Cubicle Control System

The RadiantCubicle price agent (PA) operates within the SmartHub, while the RadiantCubicle device controller agent (DCA) is integrated into the Building Management System (BMS) to oversee the physical RadiantCubicle's control. The PA receives its price data (depicted by the green arrow) from the SmartHub PCA and computes the corresponding setpoint based on this price from the RadiantCubicle's price functions. Subsequently, the computed setpoint is communicated to the DCA's PID Controller module via the BACnet IP protocol (illustrated by the deep brown arrow). Upon receiving the setpoint, the DCA relays the necessary control signals to the actuator valve (indicated by the light brown arrow), which regulates the chilled water supply to the RadiantCubicle. An energy meter, in line with the chill water supply, monitors the flow rate and both supply and return temperature. This data is transmitted over BACnet MS/TP to the DCA's compute energy modules, which calculate the radiant cooling energy. Periodically, the PA retrieves the cooling energy information from the BMS using BACnet IP, enabling continuous monitoring of the cooling energy consumption. The DCA's PID Controller and Energy Compute modules are shown in Figure 4-11 and Figure 4-12.

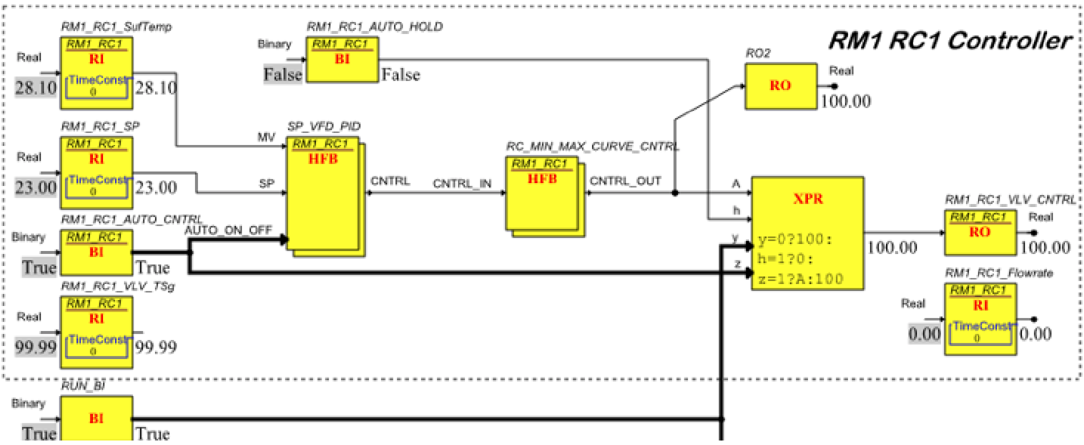


Figure 4-11: Radiant Control Logic in the BMS

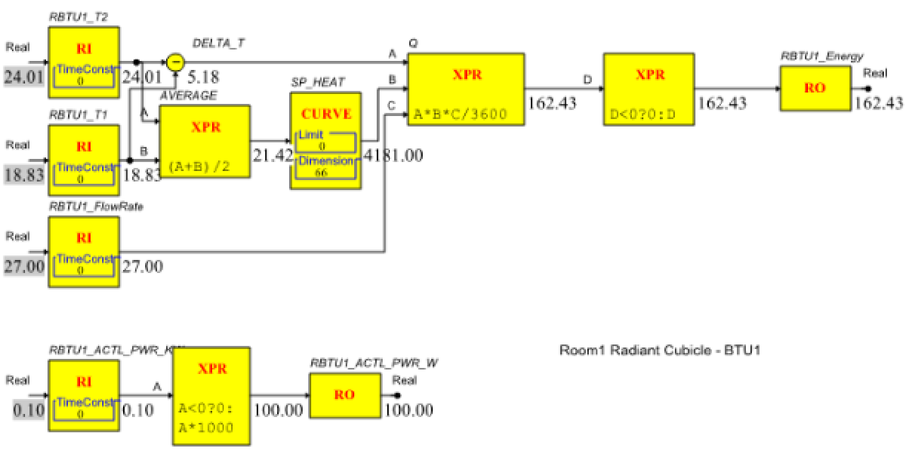


Figure 4-12: The radiant cubicle cooling energy calculations

4.4.1 Hardware

It features a mesh of copper pipes within the cubicle that circulates chilled water, cooling its surface. Research indicates that, for peak efficiency, such systems demand chilled water at 20 °C with a flow rate of 350 litres per hour [101], [102]. The flow rate is modulated using a PID controller housed in the BMS to achieve a designated surface temperature setpoint. This PID controller is housed in the BMS and manipulates actuator valves in response to the temperature setpoint, communicating over RS485.

4.4.2 Software

RadiantCubicle's Price Agent is within the SmartHub transactive platform (Figure 4-10). This agent computes the surface temperature setpoint according to a monotonically decreasing price function, depicted in Figure 5-25. Figure 5-25 shows the price function used in the case study setup. The setpoint is then relayed to the BMS via the BACnet protocol and RadiantCubicle device controller takes over to maintain this setpoint.

4.4.3 Design Challenges

A few of the challenges that were encountered during the development of the Radiant Cubicles used in the case study test set up are follows:

A. **Cubicle design**

The maiden version of Radiant Cubicle was crafted using Plaster of Paris for its radiant surface. Although it boasted commendable thermal inertia, its substantial weight hampered mobility and longevity, with issues like surface cracking emerging. Consequently, the design transitioned to utilize sandwiched corrugated aluminium panels, as illustrated in Figure 4-13.

Figure 4-13: Radiant Cubicles design

B. PID Controller & agents

Radiant Cubicle did not have its own standalone controller. Hence, a PID controller as shown in Figure 4-11 was implemented in Schneider's BMS using "editor app" by using built-in PIDA. The PIDA controller controls cool water flow rate through actuator valves based on the surface temperature setpoint. The actuator valves and temperature sensor of the Radiant Cubicle communicate with BMS over RS485. An inline BTU meter, which communicates with the BMS over BACnet IP, measures flow rates and inlet/outlet temperatures. The Radiant Cubicle Price Agent, operational within the SmartHub platform, interfaces with these endpoints.

C. Desired process value or set point (SP)

Generally, the Radiant Cubicle is controlled based on the Standard Effective Temperature (SET) [91][9]. The SET was computed factoring in variables such as Mean Radiant Temperature (MRT), air parameters, clothing, and activity levels. However, in the case study for evaluating the framework, without loss of generality, Radiant Cubicle surface temperature was chosen over the SET as the controlling parameter.

D. Energy calculations

The least count by the currently available BTU meters is 100W. However, energy changes at the task level are anticipated to be lower than 100W. A more nuanced energy computation method was thus necessitated to calculate the energy at a much granular level. Hence using temperature sensors and the flow rate data from the BTU meter, an energy compute module as shown in Figure 4-12 was implemented in the BMS to calculate the energy.

The bottom part shows the energy monitoring from the BTU meter. And the top part, current implementation, calculates energy through the following equation:

$$Q = K * (t_2 - t_1) * F$$

$where\ Q\ is\ cooling\ energy,$

$K\ is\ specific\ heat\ of\ water,$

t_1 is temperature at the inlet,

t_2 $is\ temperature\ at$ the $outlet, and$

$F\ is\ flowrate\ of\ the\ chilled\ water\ through\ the\ radiant\ cubicle$

The inputs to this module are the flow rate and both supply and return temperature

[9] Standard Effective Temperature (SET) create a normal basis for measuring the equivalence of any combination of environmental factors and personal factor. As per ASHRAE standard 55, SET is defined as the temperature of an unreal environment at 50% RH, less than 0.1 m/s airspeed, and MRT equal to DBT, in which the total heat loss from an imaginary occupant with 1.0 met activity level and 0.6 clo clothing level is the same as that from a person in the actual environment, with actual clothing and activity level.

and the output is the computed energy. By leveraging this module, a more granular and accurate assessment of the energy at the task level was achieved, ensuring a detailed understanding of the system's operation.

4.5 Challenges in integrating these PECS with TC

1. **Heterogeneity**: PECS devices exhibit diverse utility functions, ranging from monotonic increasing to decreasing functions. For instance, the SmartHub fan and light and the RadiantCubicle demonstrate monotonic increasing functions, where utility rises with increasing price. In contrast, the SmartStrip plugs and Zone AC and Light exhibit monotonic decreasing utility functions. Our system model and pricing formulation assume that all devices have increasing or decreasing monotonic functions to ensure convergence.

   However, suppose the devices can be grouped in such a way that the practical utility of the group of the devices would be an increasing or decreasing monotonic function. In that case, the convergence can be assured. Our framework supports such grouping through hierarchical distributed algorithms. Pilot deployment evaluations demonstrate the feasibility of this approach. For instance, although PECS at the task level may have mixed monotonic functions, their combined price vs. energy demand (analogous to utility functions) is monotonically decreasing and aligns with other energy-consuming devices like Zone AC and Light.

2. **Scalability:** As the number of PECS devices increases, managing the grouping complexity escalates. While pilot studies show convergence feasibility with a limited number of devices, further research or simulations with more extensive deployment are necessary to handle large-scale deployments effectively.

3. **Real-time Responsiveness:** TC systems require timely responses to dynamic energy price and demand changes. Our pilot deployment assumed one-hour slots, completing market clearing within 15 minutes before the designated slot time. Although sufficient for practical purposes, enhancing energy demand flexibility may necessitate further studies with different slot times.
4. **User Acceptance and Adoption:** Successful PECS integration with TC hinges on user acceptance and adoption. The iSPACE system offers user-friendly interfaces, personalized controls, and feedback mechanisms to enhance adoption. However, additional studies are required to gauge user acceptance effectively.

4.6 Conclusion

The development journey of PECS, encompassing SmartHub, SmartStrip, and RadiantCubicle, epitomizes the transition of the iSPACE system from a mere idea to a tangible fruition. The chapter highlights the complexity of the development process, highlighting the balance between innovative design and pragmatic challenges, underscoring the PECS system's potential for revolutionizing building energy management. These innovations empower users to tailor their surroundings for a more personalized experience and moderate energy efficiency on a granular scale.

Chapter 5

System Evaluation

"It doesn't matter how beautiful your theory is, it doesn't matter how smart you are. If it doesn't agree with experiment, it's wrong." - Richard P. Feynman

5.1 Introduction

This chapter details the evaluation of the iSPACE system as mentioned in section 1.4.3. It consists of a subsection on system evaluation methodology, the pilot setup, measurements, experiment and data analysis to access the system function and non-function properties (like MAS performance and convergence rate) through a combination of quantitative and qualitative analyses. A critical discussion concludes this chapter, drawing pertinent insights into the comprehensive understanding of the system's strengths, potential areas for improvement, challenges, including scalability, and limitations of the systems and recommendations.

5.2 System Evaluation Methodology

As highlighted in the section 1.3, the system's three objectives are:

- **Objective 1:** Seamless integration of PECS within the task environment.
- **Objective 2:** Bridging the task environment system with the ambient system.
- **Objective 3:** Utilizing TC to drive DR at the task-specific level.

To provide tangible evidence of these objectives' validity, the system has been evaluated as per the flow captured in *Figure 5-1.*

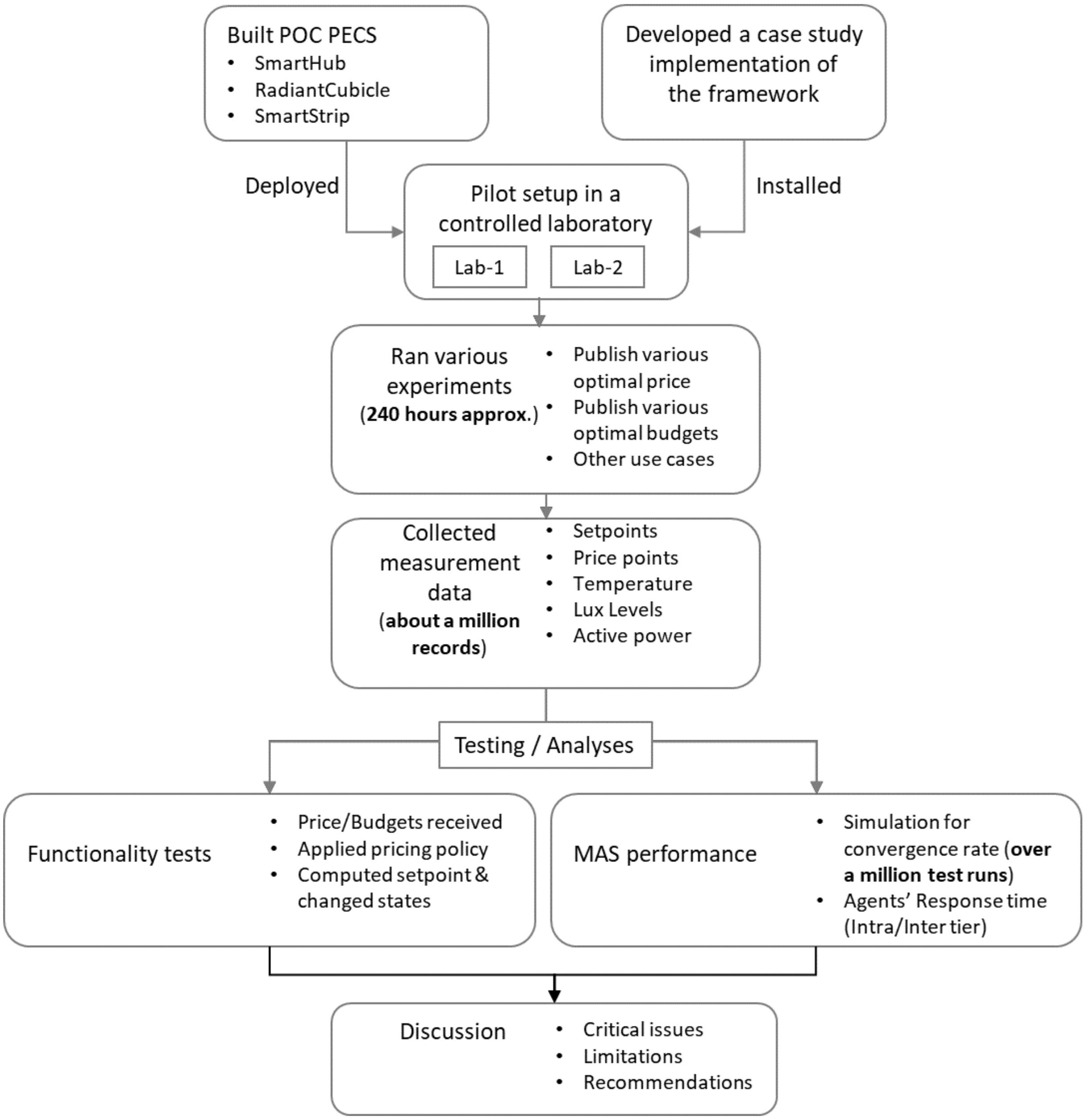


Figure 5-1: Evaluation methodology flow chart

In the first stage, a proof of concept (POC) Personal Environmental Control Systems (PECS) such as the SmartHub, Radiant Cubicle, and SmartStrip as detailed in *Chapter 4 have* been developed. This proof-of-concept laid the foundation for a comprehensive framework (detailed in section 3.3*)* case study, seamlessly integrating the PECS within and with the ambient controls. The framework is designed to gather real-time inputs from end-users or employ autonomous agents that function based on pre-configured preferences, such as warmth or coolness. This study implemented the latter, with the complete source code available in a public repository (See Appendices A-5). In the second stage, a pilot setup was introduced within a controlled laboratory, detailed in section 2. A SmartHub, SmartStrip, and Radiant Cubicle were installed at each designated workstation, totalling 32 individual energy-consuming device, each with unique price functions or threshold prices for control (See Appendices A-8). Among these, 12 energy-consuming devices utilized monotone price functions for setpoint control, while 20 energy-consuming devices employed a threshold price for device operations. Agent configurations are detailed in following section 5.5. In the third stage, various experiments were conducted, spanning 240 hours approx. and aggregating a million records. Finally, this data was used in evaluating system functionality and performance, including system simulation runs based on experimental data.

Important:

1. We utilised a dimensionless variable known as "price point" instead of the actual price for the case study. This variable ranges from zero to one and serves as a scaled price representation.
1. Every experiment commenced with a two-hour default setpoint run, ensuring environmental stability. Following this, the scheduled experiment took place from 08:00 hrs to 18:00 hrs daily. At the end of each day, the accumulated data was systematically archived for subsequent analysis.
2. Additionally, we assume that each experiment is divided into one-hour slots. On average, it takes 25 iterations at each level, so there are 781,250 iterations for each slot. Assuming each iteration can be completed on an average of 2ms, it takes approximately 13 minutes to achieve market equilibrium for each slot. Consequently, the market clearing process is initiated 15 minutes before the start of each slot, and a new price is published at the start of each slot. The experiment runs for the duration of slot to maintain the set point corresponding to the price point.

5.3 Pilot setup

The pilot's inception was twofold: First, to identify and benchmark a controlled environment conducive to the experiments, and second, to configure the iSPACE systems to run various experiment to evaluate the iSPACE system efficacy.

Figure 5-2: FDD Lab - Cubicles, AHU, Chillers, VAV boxes, and sensors

5.3.1 FDD Lab and Benchmarking

The Fault Detection and Diagnostics (FDD) lab, nested within the research centre and comprising two identical 10x10 feet rooms, was deemed fit for the experiments. Equipped with HVAC equipment (such as AHU, Chillers, and VAV boxes) and managed by Schneider's Building Management System (BMS). Some of instruments include Schneider EcoStruxure Building software, Schneider automation servers and multiple IO module, Grundfos CM3-3 pumps, Belimo VAVs, Danfoss VFD, Johnson Controls Actuator, Pettinaroli - pressure independent control valves, and Kamstrup Multical 403/603 BTU meters (Figure 5-2, Figure 5-3, Figure 5-4, and Figure 5-5). The lab's ambient environment can be tailored for precision. However, the BMS controllers were initially set to default configurations, primarily operating on simple on-off mechanisms. This rudimentary setup led to irregular energy fluctuations and inconsistent profiles between rooms.

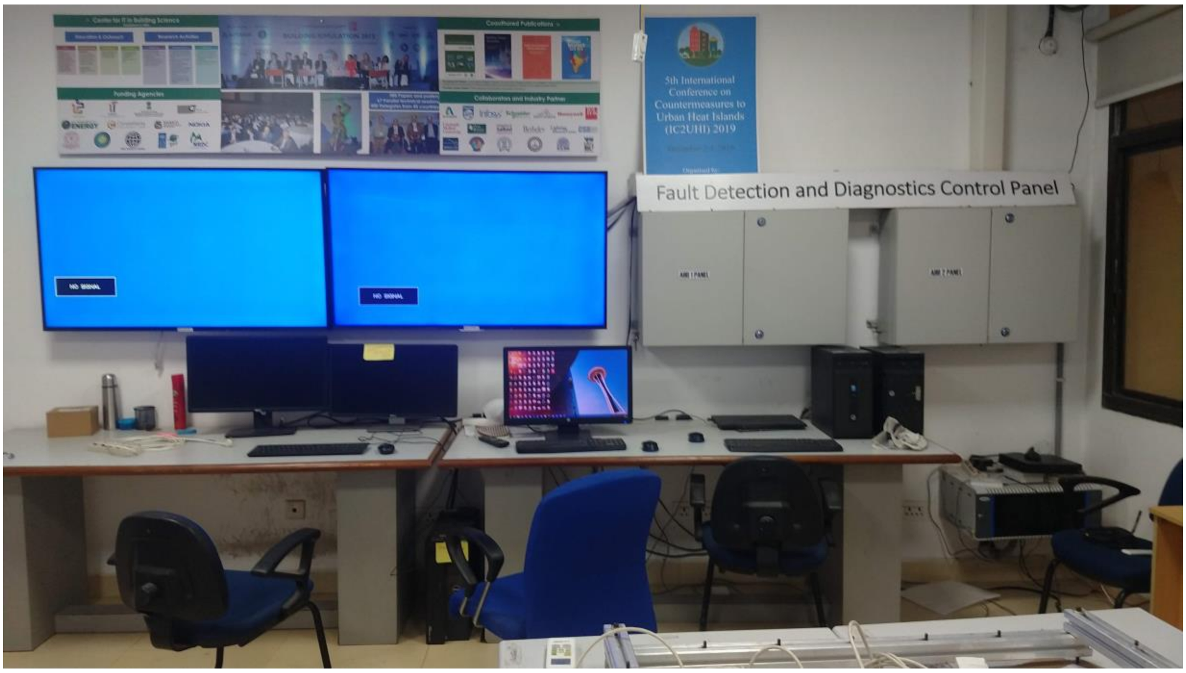


Figure 5-3: FDD Lab - control room

Figure 5-4: FDD Lab - control panels (Schneider's BMS)


Figure 5-5: Schneider's BMS - Automation contoller, & IO modules

Hence, a tailored control logic was meticulously developed and integrated, painstakingly fine-tuning and calibrating multiple controllers, including those for the AHU control valve for chill water supply, AHU fan VFD, secondary pumps VFD, room VAV, and the Radiant Cubicle PID. The intricacies of this complex control logic are depicted in Figure 5-6 and Figure 5-7.

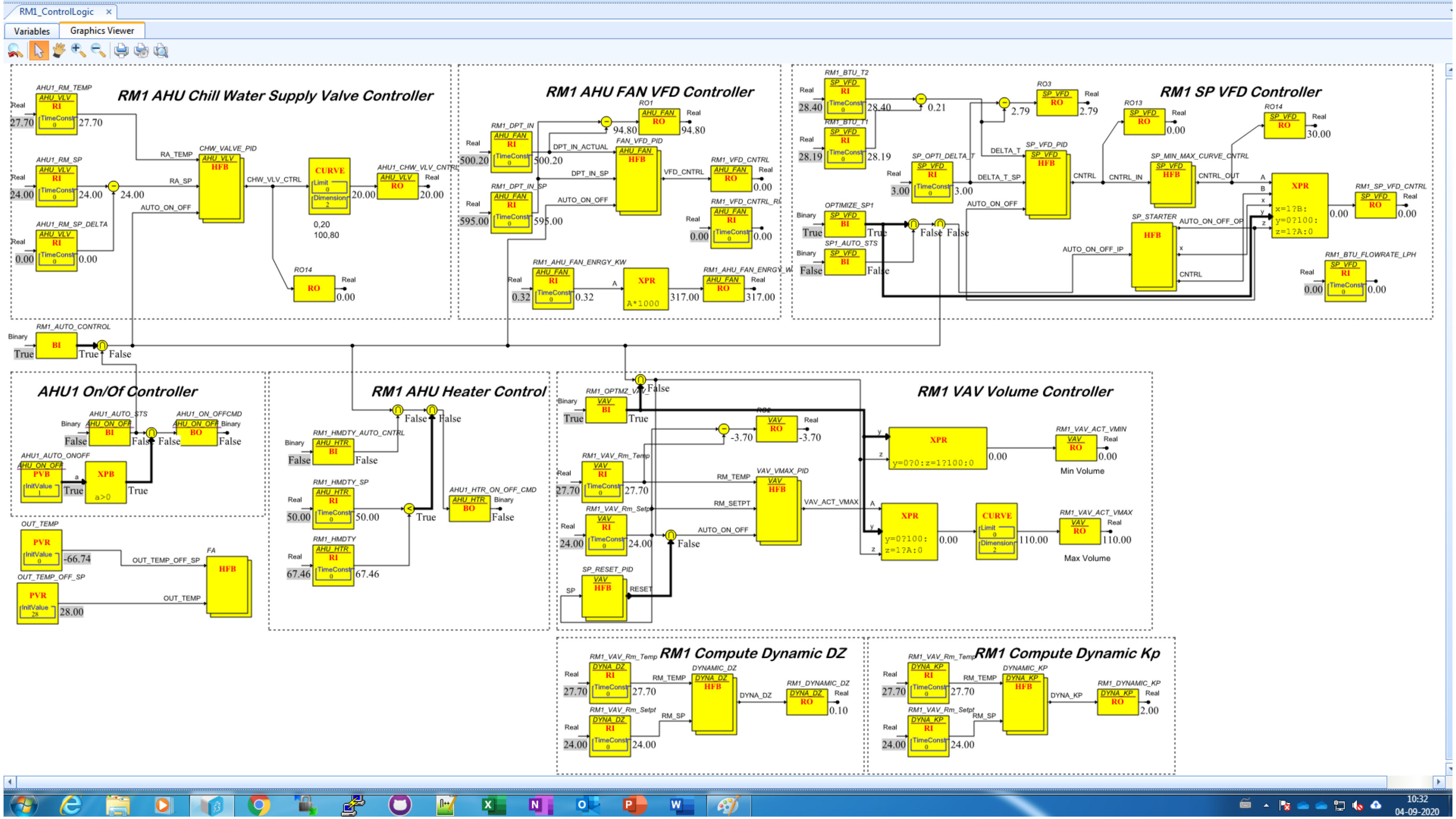


Figure 5-6: Room1 Ambient AC Control Logic

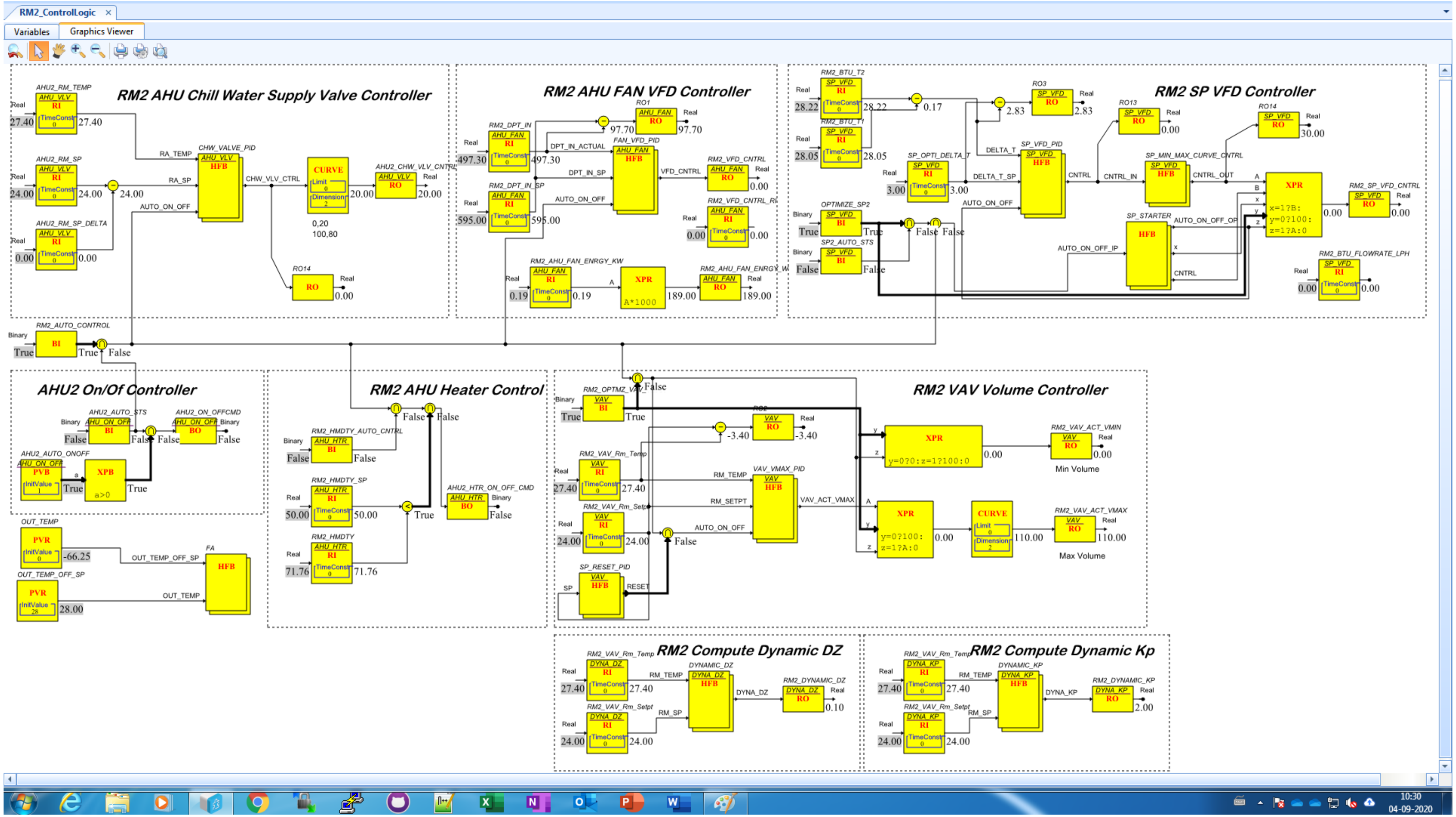
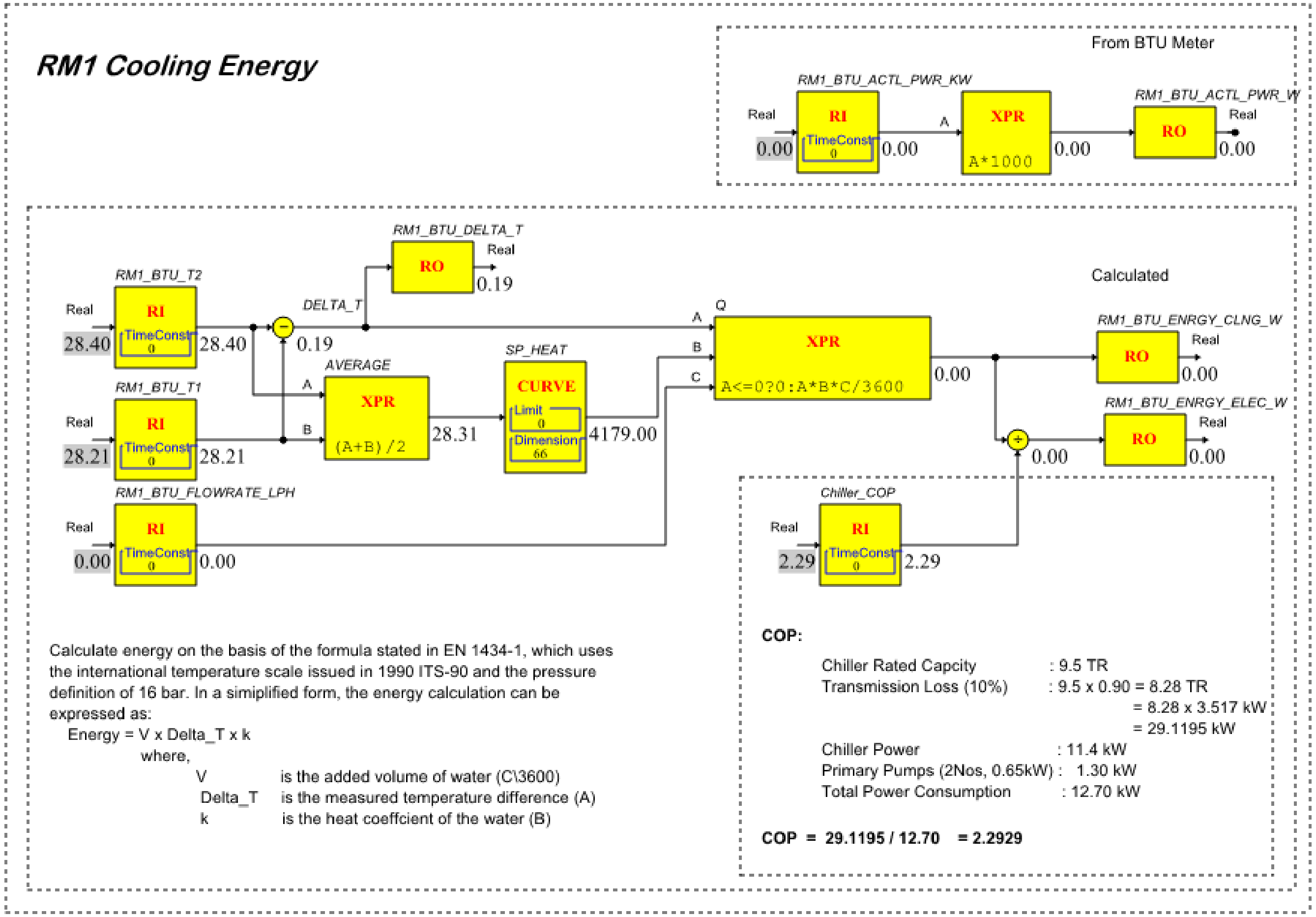


Figure 5-7: Room2 Ambient AC Control Logic



Heat Coeffcient of Water

Temp,	k
0,	4217
1,	4213
2,	4210
3,	4207
4,	4205
5,	4202
6,	4200
7,	4198
8,	4196
9,	4194
10,	4192
11,	4191
12,	4189
13,	4188
14,	4187
15,	4186
16,	4185
17,	4184
18,	4183
19,	4182
20,	4182
21,	4181
22,	4181
23,	4180
25,	4180
26,	4179
29,	4179
30,	4178
38,	4178
39,	4179
44,	4179
45,	4180
48,	4180
49,	4181
51,	4181
52,	4182
54,	4182
55,	4183
57,	4183
58,	4184
59,	4184
60,	4185
61,	4185

Figure 5-8: Room Cooling Energy (Electrical) Calculation

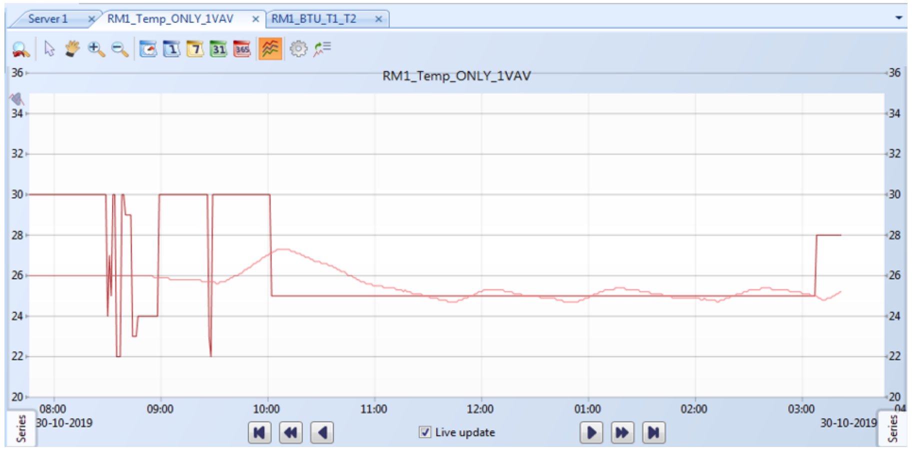


Figure 5-9: Room 1 VAV Vs Temp

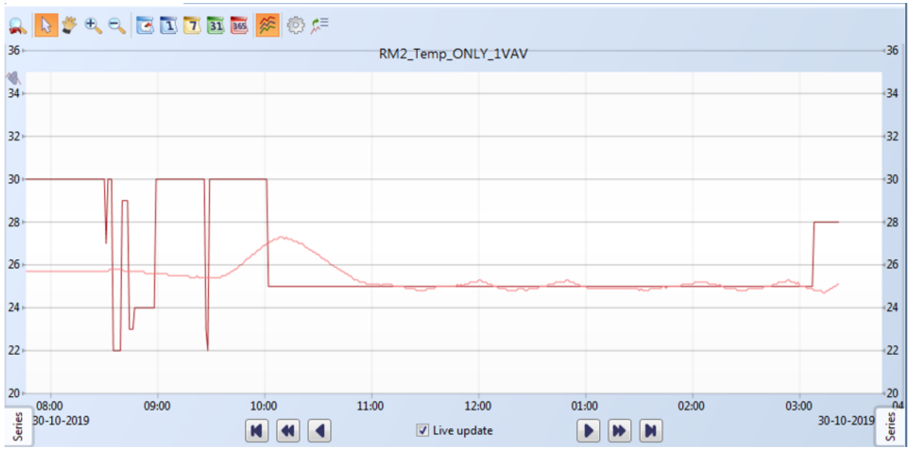


Figure 5-10: Room 2 VAV Vs Temp

Figure 5-11: Room 1 Energy Profile

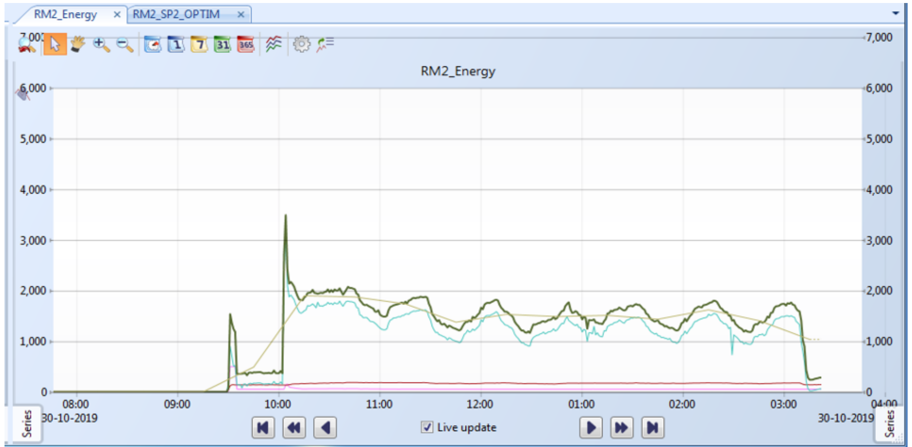


Figure 5-12: Room 2 Energy Profile

To facilitate comparative analysis, thermal cooling energy was translated to equivalent electrical energy using a COP of 2.29. COP vary due to multiple factors, introducing uncertainty in electric energy estimation. However, in the context of this thesis, our focus is primarily on understanding the energy demand flexibility facilitated by the iSPACE system rather than directly comparing energy savings. To address this, we used a constant COP assumption, particularly for benchmarking purposes, where similar operating conditions were assumed between the benchmark and experimental scenarios. While this approach may introduce some level of uncertainty, it allows us to isolate the impact of the iSPACE system on energy demand and flexibility, which aligns with the objectives of our study.

As depicted in Figure 5-8, the COP is calculated as ratio of rated chiller capacity (assuming a 10% transmission loss) and total power consumed in the primary circuit, which includes the rated chiller power and the primary pumps power. And the thermal cooling energy delivered to the rooms is computed for each room and converted to its electrical equivalent using the COP.

The total energy of a room was computed by combining the equivalent electrical energy of thermal cooling energy, the electrical energy from AHU fan and the secondary pump. These measured data were captured at 10-second intervals, averaged every 60 seconds, and recorded. The Trapezoidal Rule was employed to deduce the energy consumption over a set duration.

The outcome was a successful alignment of energy profiles for both rooms. When juxtaposing the energy profiles of both rooms, their similarities become evident (Figure 5-9 and Figure 5-10 provide a typical setpoint vs. room temperature and Figure 5-11 and Figure 5-12 provide a typical energy profiles for both rooms, respectively). Over two months, the control parameters were further refined, ensuring both rooms mirrored each other's thermal cooling and energy profiles, and the deviation is less than 2%. Three benchmarks were established. A comprehensive comparison of these metrics is encapsulated in Table 5-1 and further detailed in Figure 5-13, Figure 5-14, and Figure 5-15. The price point randomness during the benchmark was assessed using Matlab function "*runstest*" and a h=0 and p=0.9960 was observed[10].

With its standardized environment and an optimized control mechanism, the meticulously configured FDD Lab provides an ideal foundation for a precise and insightful evaluation of the proof-of-concept iSPACE system.

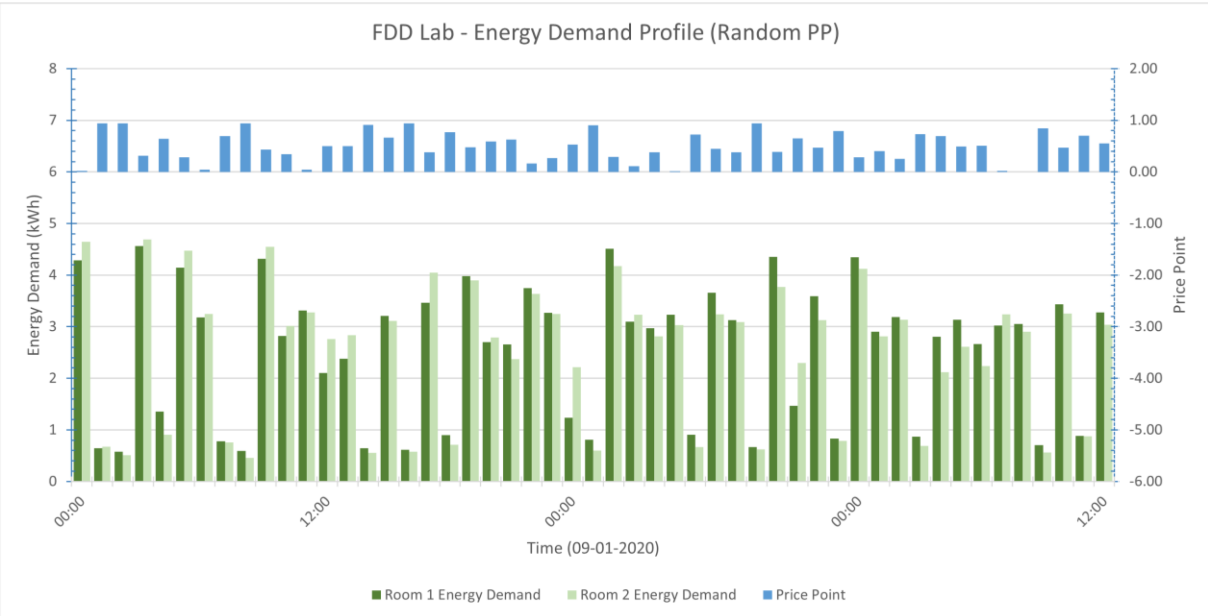


Figure 5-13: FDD Lab - rooms energy demand (kWh) comparision (Random pp).
Total Energy demand difference for the rooms is less than 2%.

[10] runstest uses a test statistic which is the difference between the number of runs and its mean, divided by its standard deviation. The test statistic is normally distributed when the null hypothesis is true. The returned value of h = 0 indicates that runstest does not reject the null hypothesis that the values in x are in random order at the default 5% significance level. p-value of the test, returned as a scalar value in the range [0,1]. p is the probability of observing a test statistic as extreme as, or more extreme than, the observed value under the null hypothesis. Small values of p cast doubt on the validity of the null hypothesis.

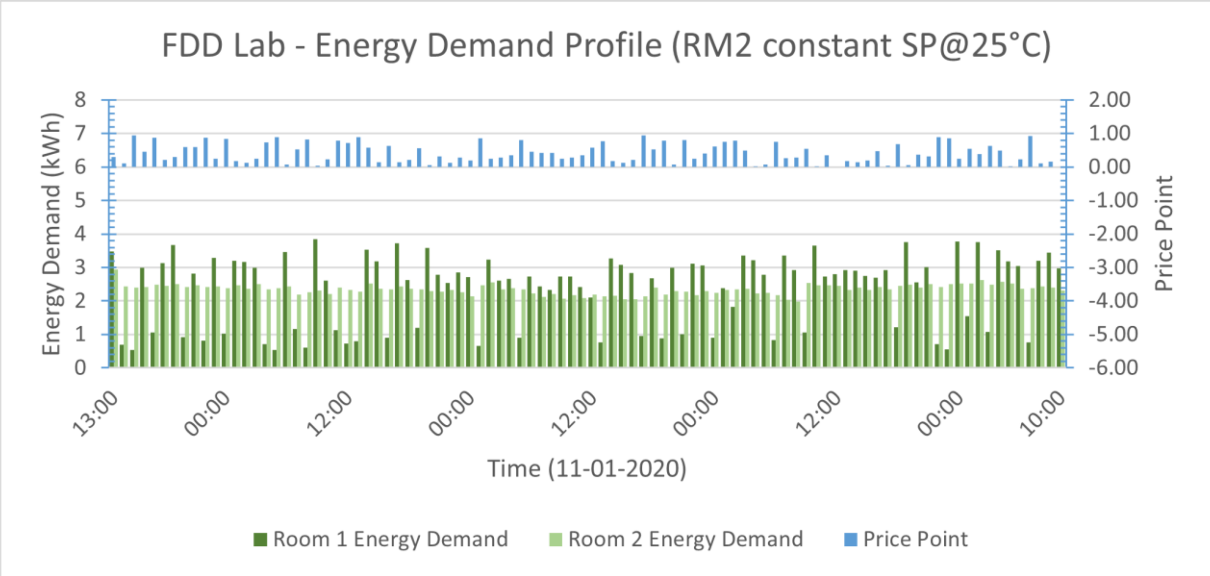


Figure 5-14: FDD Lab - rooms energy demand (kWh) comparision (RM2 contant SP@25°C).

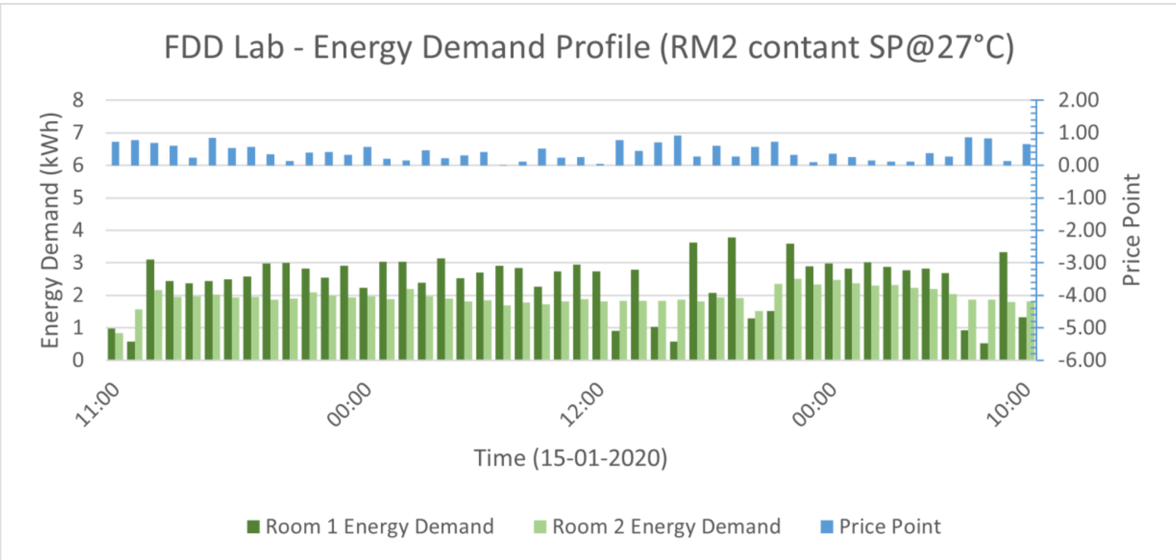


Figure 5-15: FDD Lab - rooms energy demand (kWh) comparision (RM2 contant SP@27°C).

Table 5-1: FDD Lab energy demand benchmarks & metrics

			Energy Demand (kWh)							
			Room1				***Room2***			
Sl. No.	**Test Description**	**Duration**	***Total***	***Min***	***Max***	***Avg***	***Total***	***Min***	***Max***	***Avg***
Benchmark-I (Figure 5-13)	Random price point is published to both the Rooms, 1&2	51 hrs	128.98	0.58	4.57	**2.53**	126.97	0.46	4.69	**2.49**
Benchmark-II (Figure 5-14)	Random price published to room 1, but room was maintained at constant setpoint of 25°C	94 hrs	219.32	0.53	3.84	**2.33**	220.91	1.98	2.94	**2.35**
Benchmark-III (Figure 5-15)	Random price published to room 1, but room was maintained at constant setpoint of 27°C	48 hrs	116.99	0.53	3.78	**2.53**	93.51	0.84	2.51	**2.49**

5.3.2 iSPACE Deployment

The experimental facility was divided into two zones, each containing two cubicles. Every cubicle was outfitted with key components of the iSPACE system: a SmartHub, a SmartStrip, and a Radiant Cubicle at each workstation as shown in Figure 5-16. Additionally, each zone was equipped with an ambient AC and lighting system to maintain the desired ambient conditions. A Raspberry Pi was employed to seamlessly integrate the Building Management System (BMS) with the transactive framework, running a dedicated transactive control instance (Figure 5-17). The necessary BACnet data points were created in the BMS, such as setpoints. The transactive framework regularly published updated values to these data points through the BACnet protocol.

Figure 5-16: iSPACE system pilot deployment in the FDD lab

The experimental setup consists of 32 distinct energy-consuming devices, each governed by specific price functions or predetermined threshold pricing. Of these, twelve energy-consuming devices, including zone ambient temperature controllers, ambient light

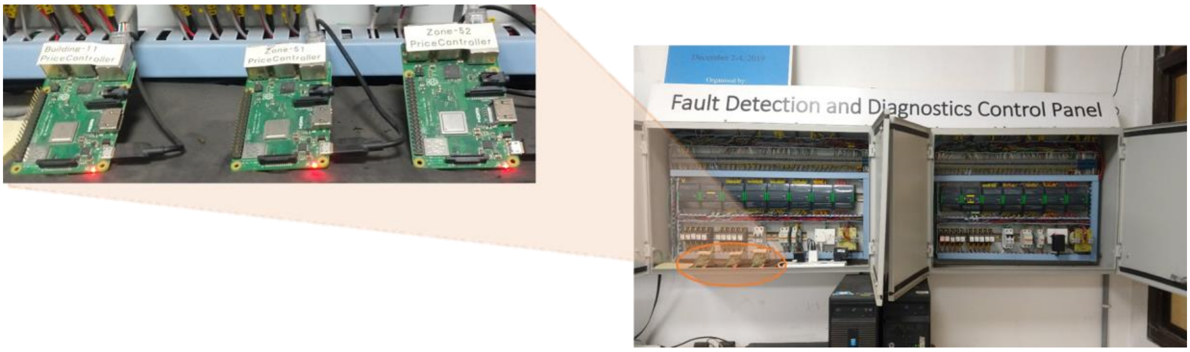


Figure 5-17: RPis running a dedicated transactive control instance to integrate with BMS

controllers, SmartHub fans, and Radiant Cubicle controllers, operate using monotone price functions for setpoint control. The remaining twenty energy-consuming devices function based on threshold pricing, which determines whether a device is activated or deactivated. The user can adjust this threshold pricing, offering flexibility and control. The experimental parameters utilized the threshold price points listed in Table 5-6 (refer section 5.5.3).

For the Radiant Cubicle, a PID controller in the BMS modulates the flow rate through actuator valves, contingent on the preset surface temperature. These valves communicate synchronously with the BMS via RS485 protocol. Internally, the Price Agent for the radiant cubicle operates within the SmartHub's transactive platform. This agent determines a new surface temperature setpoint based on the depicted monotone price function in Figure 5-25 (refer section 5.5.3 C). The corresponding price function depends on user preference, allowing users to switch between 'Cool', 'Normal', and 'Warm' settings. For the experiment, 'Normal' is used. This new temperature setpoint is relayed to the BMS through the BACnet protocol.

In the case of the SmartHub Fan, its Price Agent calculates the optimal fan speed relying on the monotone price function highlighted in Figure 5-23 (refer section 5.5.3 B). User comfort is paramount, allowing the system to adjust based on individual preferences, either 'Cool', 'Normal', or 'Warm'. As with the Radiant Cubicle, users can adjust settings in real-time. Once the ideal setpoint is deduced, it is transmitted to the message bus, from where the device controller assumes control to maintain this setpoint.

Lastly, the Raspberry Pi, which runs the zone controller's transactive control instance, houses the Price Agent. The Price Agent computes a new ambient zone temperature setpoint and ambient lighting setpoint based on the monotone price function shown in Figure 5-19 and Figure 5-21(refer section 5.5.2 B). Once determined, these new setpoints are directly communicated to the BMS via the BACnet protocol, The new setpoint is published to the BMS over BACnet protocol, and the BMS tries to maintain the new set point.

5.4 Measurement

Primary and secondary measurements were considered vital for a comprehensive understanding of the system's functionality and efficiency. Over 350 distinct data points were earmarked for logging, and the required Extract, Transform, Load (ETL) processes were set up within the Building Management System (BMS). While the BMS inherently provides measurements for various parameters concerning the test rooms, it was imperative to incorporate task-level data, like the sensors data, control signals and status, other events like price change, active energy.

For most of these parameters, the data was captured in 30-second intervals. However, for energy computation metrics, the data points were recorded every 10 seconds and averaged over 60 seconds. The Trapezoidal Rule provided an efficient method to calculate the energy consumption over specified intervals. In addition to this meticulous data capturing, field tests were conducted on the sensors to ensure consistency and accuracy (Appendices A-9). Some of the data points are as shown in Table 5-2.

Table 5-2: Measurement data points

Name	Description	Object ID	BACnet name	BACnet type
OUT DOOR RH	Outdoor Relative Humadity	<analog-input, 1>	OUT DOOR RH	analog-input
OUT DOOR TEMP	Outdoor temperature	<analog-input, 2>	OUT DOOR TEMP	analog-input
RC_WTR_CLR_AUTO_CNTRL	Radiant Cubicle Water Cooler Auto Co…	<binary-output, 7>	RC_WTR_CLR_AUTO_CNTRL	binary-output
RC_WTRCLR_TSP	Radiant Cubicle supply water cooler s…	<analog-output, 7>	RC_WTRCLR_TSP	analog-output
RM1_HMDTY_AUTO_CNTRL	Room 1 - Humidity Auto Control	<binary-output, 8>	RM1_HMDTY_AUTO_CNTRL	binary-output
RM1_HMDTY_SP	Room 1 - Humidity Setpoint	<analog-output, 5>	RM1_HMDTY_SP	analog-output
RM1_HTR_AUTO_CNTRL	Room 1 - Heater Auto Control	<binary-output, 1>	RM1_HTR_ACTIVATE	binary-output
RM1_RC1_AUTO_CNTRL	Room 1 - Radiant Cubicle 1 - Auto Con…	<binary-output, 3>	RM1_RC1_AUTO_CNTRL	binary-output
RM1_RC1_TSP	Room 1 - Radiant Cubicle 1 - Surface…	<analog-output, 1>	RM1_RC1_TSP	analog-output
RM1_RC2_AUTO_CNTRL	Room 1 - Radiant Cubicle 2 - Auto Con…	<binary-output, 4>	RM1_RC1_AUTO_CNTRL_copy	binary-output
RM1_RC2_TSP	Room 1 - Radiant Cubicle 2 - Surface…	<analog-output, 2>	RM1_RC1_TSP_copy	analog-output
RM1_SP	Room 1 - ambient SP	<analog-value, 1>	ROOM1_SETPOINT	analog-value
RM2_HMDTY_AUTO_CNTRL	Room 2 - Humidity Auto Control	<binary-output, 9>	RM1_HMDTY_AUTO_CNTRL_copy	binary-output
RM2_HMDTY_SP	Room 2 - Humidity Setpoint	<analog-output, 6>	RM1_HMDTY_SP_copy	analog-output
RM2_HTR_AUTO_CNTRL	Room 2 - Humidity Auto Control	<binary-output, 2>	BACnet_Over_IP_5_Application_RM1_…	binary-output
RM2_RC1_AUTO_CNTRL	Room 2 - Radiant Cubicle 1 - Auto Con…	<binary-output, 5>	BACnet_Over_IP_5_Application_RM1_…	binary-output
RM2_RC1_TSP	Room 2 - Radiant Cubicle 1 - Surface…	<analog-output, 4>	BACnet_Over_IP_5_Application_RM1_…	analog-output
RM2_RC2_AUTO_CNTRL	Room 2 - Radiant Cubicle 2 - Auto Con…	<binary-output, 6>	RM1_RC2_AUTO_CNTRL_copy	binary-output
RM2_RC2_TSP	Room 2 - Radiant Cubicle 2 - Surface…	<analog-output, 3>	RM1_RC2_TSP_copy	analog-output
RM2_SP	Room 2 - ambient SP	<analog-value, 5>	ROOM2_SETPOINT	analog-value

However, a unique challenge was posed by the BTU meters. The smallest measurement they could capture was 100W. A more detailed energy calculation approach became imperative since energy fluctuations at the task level often fell below this mark. Hence, the need to calculate energy more precisely is necessary.

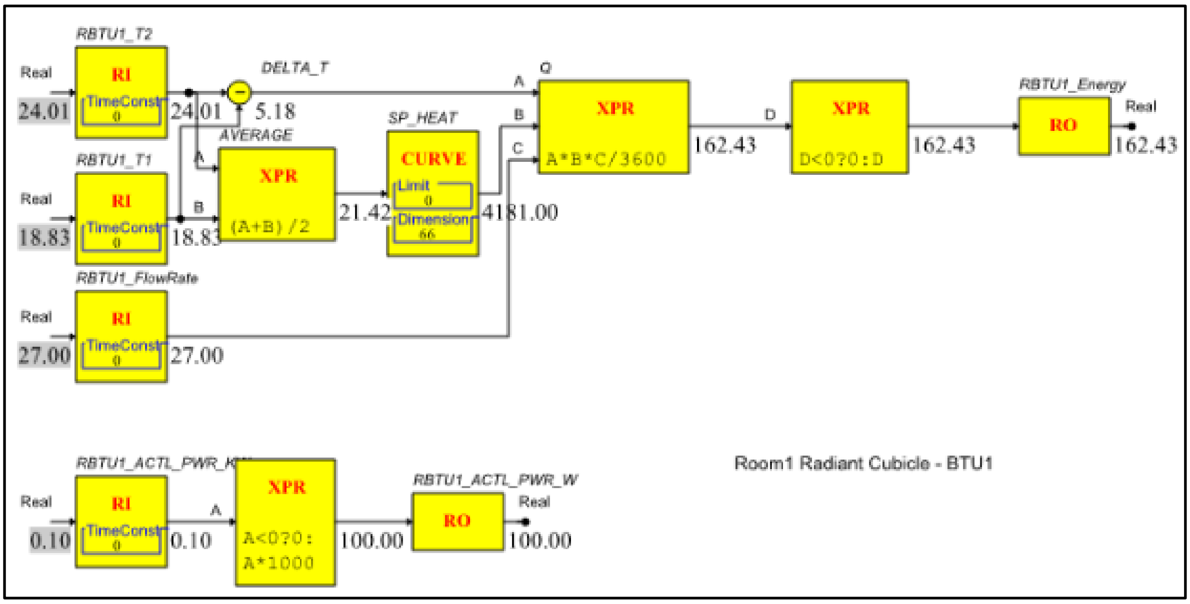


Figure 5-18: The cooling energy calculations based on EN 1434-1

As shown in Figure 5-18, by utilizing temperature sensors alongside flow rate data sourced from the BTU meter, energy computations were computed using formula stated in EN 1434-

1 and the formula is as follows:

$$Q = K * (t_2 - t_1) * F$$

$where\ Q\ is\ cooling\ energy$

$K\ is\ specific\ heat\ of$ the $water$,

$t_1\ is$ temperature at the inlet,

$t_2\ is\ temperature\ at$ the $outlet, and$

$F\ is\ flowrate\ of\ the\ chilled\ water\ through\ the\ radiant\ cubicle$

By leveraging this method, a more granular and accurate assessment of the energy was achieved, ensuring a detailed understanding of the system's operation.

5.5 Agent Configurations

From April through June 2019 and January through Feb 2020, the pilot setup was subjected to multiple tests involving manual adjustments of various control parameters. These adjustments encompassed parameters such as:

1. AC setpoints in the test room
2. Light setpoints
3. Radiant cubicle surface temperature setpoint
4. Chilled water flowrates
5. Various settings for the PECS and other similar parameters.

The modifications were made through the Building Management System or by sending BACnet commands directly to the devices. The main goal of these adjustments was to calibrate the parameters various controllers.

Several configuration sets were established by carefully analysing the measurement data obtained during these trial runs. These configurations encompassed:

- Different energy demand functions
- Suitable setpoints for devices

The configurations are a means to test the efficacy of the system, and the efficacy of these parameters is not studied as part of the thesis. And the following assumption were made for the system evaluation:

1. It's presupposed that there are no other energy providers at zone and task levels, and the upstream Price Control Agents (PCA) is the energy provider.
2. The current energy demand is determined based on historical active power data.
3. The local coordinator agent, that is, the local PCA ensures that it leverages the maximum potential of the energy supplied by the primary energy provider.
4. The Price Agents (PA) associated with energy-consuming devices compute setpoints as configured for the corresponding device.
5. The device control agents (DCA) rely on the data from various sensors (such as those measuring temperature, humidity, CO2 levels, occupancy, lux levels, etc.) to operate their controllers as directed by the setpoints provided by the PA.
6. It is assumed that the experiment is divided into one-hour slots.
7. There are four levels (Building Level PCA, Zone Level PCA, Task level PCA, and PA).
8. Further assumed that each level takes 25 iterations. Hence, there are a total of 781,250 iterations for each slot. Assuming each iteration can be completed in 2ms, it takes 13 min to achieve the market equilibrium for each slot.

Based on these assumptions, the bids timeout and iterations are computed and adjusted accordingly.

Lastly, a crucial parameter, a constant exponential weighted moving average factor, denoted as "rc_factor", configured with the same value across all the agents. The exponential-weighted moving average factor determines how the average adapts to the latest trend, especially when computing the exponential-weighted moving average active power. As sensor readings are taken every 30 seconds, the rc_factor is set at 120. This corresponds to a 1-hour exponential weighted moving average active power.

The agent's configurations used for the experimental validation are detailed in the following subsections.

5.5.1 Building Level

The agent of relevance to the current system evaluation at the building level is Price Controller Agent (PCA). The key configuration parameters used for the agent are as follows:

A. **Price Controller Agent (PCA)**

The key configuration parameter values for PCA at the building level are listed in Table 5-3.

Table 5-3: Building Level PCA configuration

Sl. No.	Configuration Parameter	Value	Remark
1	mode_pass_on_params		
1.1	bid_timeout	20 seconds	

Sl. No.	Configuration Parameter	Value	Remark
1.2	weight_factors	[0.0, 0.5, 0.5]	The budget is divided equally among zone 1 and Zone 2. There is no building-level energy demand.
2	mode_default_opt_params		
2.1	publish_optimal	TRUE	Building Level PCA is authorised to conclude the auctioning.
2.2	us_bid_timeout	900 seconds	The max period in which the bidding process should be completed is set to 15 min.
2.3	lc_bid_timeout	180 seconds	The max period for each bidding iteration is set to 3 min. This parameter is the max period in which the downstream PCA should conclude its local bidding and respond with its energy demand bid.
2.4	max_iterations	30 counts	Allowed number of maximum iterations.
2.5	max_repeats	10 counts	The allowed maximum number of consecutive iterations that result in no change in price is set to 10.

Sl. No.	Configuration Parameter	Value	Remark
2.5	deadbands	[100, 100, 100]	Energy demand deadbands.
2.7	gammas	[0.0, 0.0002, 0.0002]	Step size, $\delta = \frac{max_{price} - min_{price}}{max_{energy} - min_{energy}}$
2.8	alphas	[0.0, 0.0035, 0.0035]	Learning rates
2.9	weight_factors	[0.0, 0.5, 0.5]	Zone's weightage

5.5.2 Zone Level

The two agents relevant to the current system evaluation at zone level are A) Price Controller Agent and B) Zone Controller Agent (Price Agent and Device Controller Agent for building). The key configuration parameters used for these agents are as follows:

A. **<u>Price Controller Agent (PCA)</u>**

The key configuration parameter values for PCA at the zone level are listed in Table 5-4.

Table 5-4: Zone Level PCA configuration

Sl. No.	Configuration Parameter	Value	Remark
1	mode_pass_on_params		
1.1	bid_timeout	15 seconds	
1.2	weight_factors	[0.75, 0.15, 0.15]	72% budget allocated for zone and the rest is divided equally among SmartHub 1 and SmartHub 2.

Sl. No.	Configuration Parameter	Value	Remark
2	mode_default_opt_params		
2.1	publish_optimal	FALSE	Building Level PCA is authorised to conclude the auctioning.
2.2	us_bid_timeout	175 seconds	The max period in which the bidding process should be completed is set to ≈ 3 min approx.
2.3	lc_bid_timeout	35 seconds	The max period for each bidding iteration is set to ≈ 1/2 min. This parameter is the max period in which the downstream PCA should conclude its local bidding and respond with its energy demand bid.
2.4	max_iterations	30 counts	Allowed number of maximum iterations.
2.5	max_repeats	10 counts	This parameter represents the maximum allowed iterations that result in no change in the price in consecutive messages.
2.5	deadbands	[100, 100, 100]	Energy demand deadbands.
2.7	gammas	[0.0002, 0.0017, 0.0017]	Step size,

Sl. No.	Configuration Parameter	Value	Remark
			$\delta = \frac{max_{price} - min_{price}}{max_{energy} - min_{energy}}$
2.8	alphas	[0.0035, 0.0035, 0.0035]	Learning rates
2.9	weight_factors	[0.5, 0.5, 0.5]	Zone's weightage

B. Zone Controller (Price Agent)

There are two types of parameters that need to be configured, and they are:

1) **Price functions and energy demand functions**: The functions used for zone air conditioning and lighting are as defined in Figure 5-19, Figure 5-20, Figure 5-21, and Figure 5-22.

   Note: The lighting setpoint for the Philips LED we used is represented as a fraction, where 1 signifies 100% intensity.

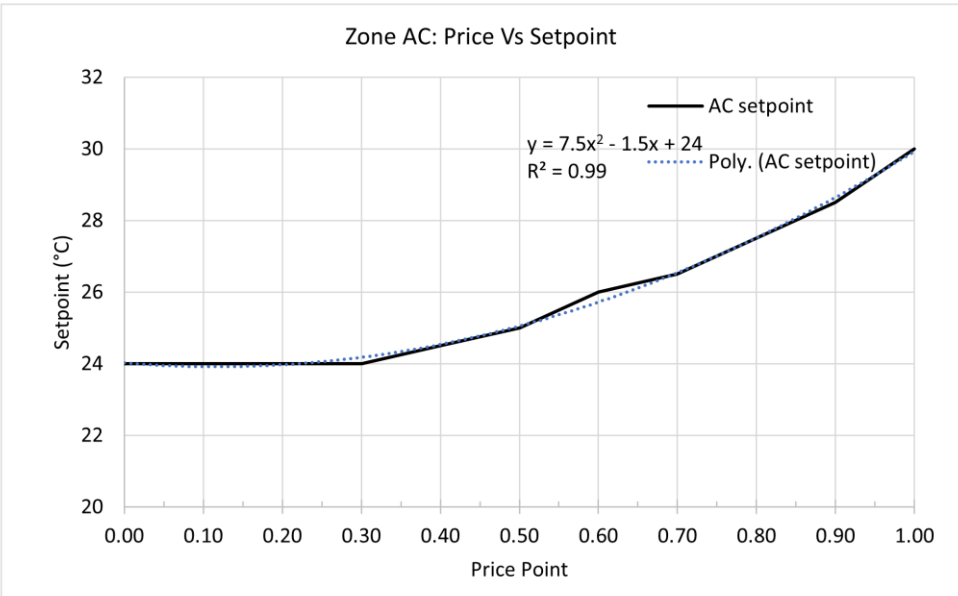


Figure 5-19: Zone AC price function

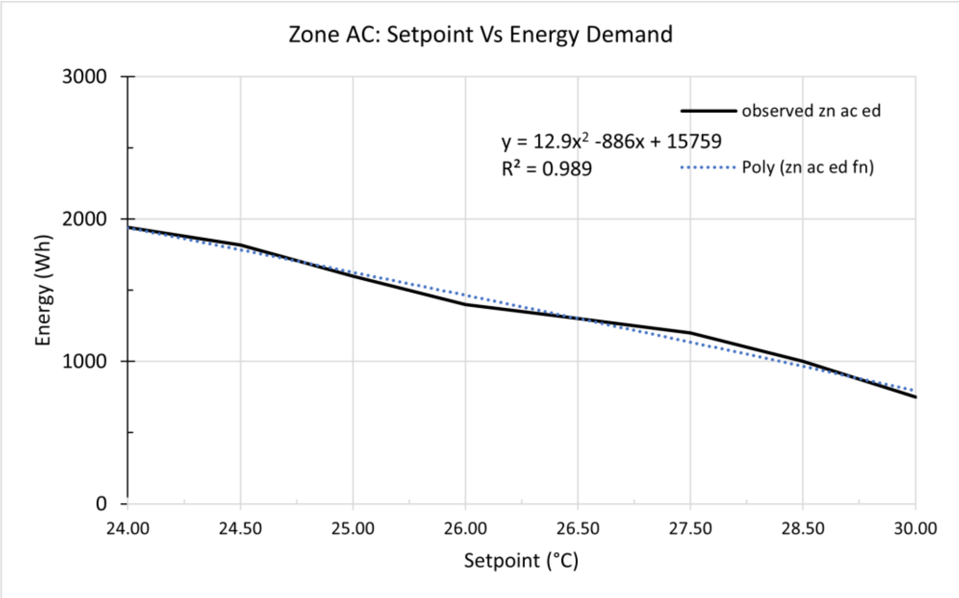


Figure 5-20: Zone AC energy demand function

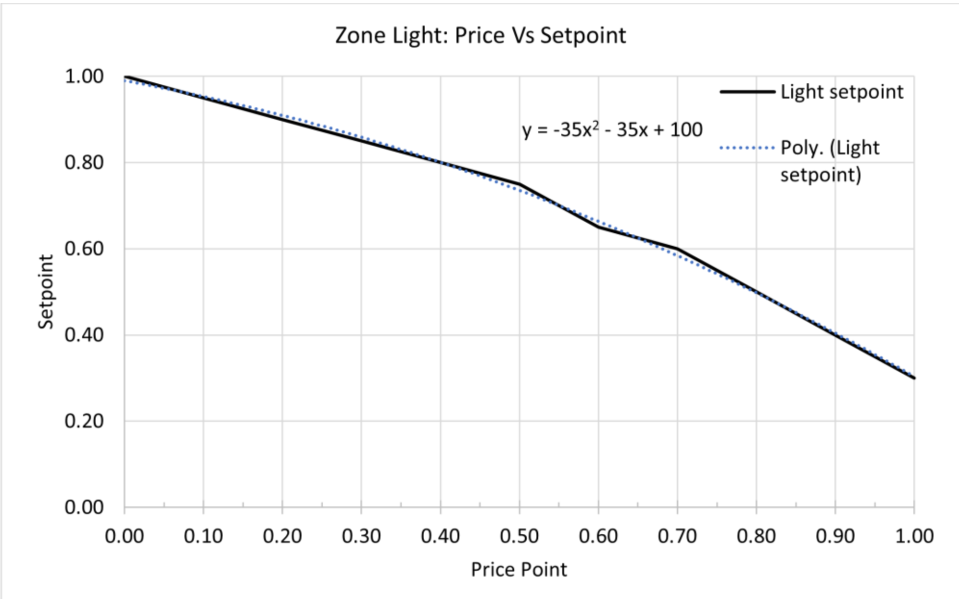


Figure 5-21: Zone Light price function

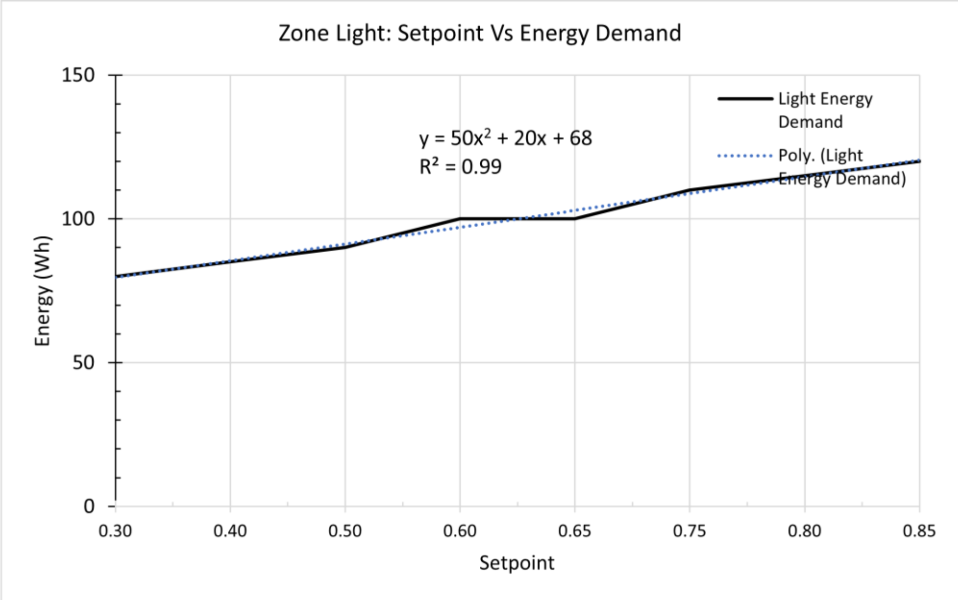


Figure 5-22: Zone Light energy demand function

2) **Gradient Descent parameters**: The Zone Controller gradient descent parameter configurations are as listed in Table 5-5.

Table 5-5: Zone Controller gradient descent configuration

Sl. No.	Configuration Parameter	Value	Remark
1	gd_params		
1.1	max_iterations	100 counts	Allowed number of maximum iterations.
1.2	max_repeats	10 counts	This parameter represents the maximum allowed iterations that result in no change in the price in consecutive messages.
1.3	deadbands	100	Energy demand deadbands.
1.4	gammas	{"ac": 0.0002, "light": 0.0125}	Step size, $\delta = \frac{max_{price} - min_{price}}{max_{energy} - min_{energy}}$
1.5	weight_factors	{"ac": 30, "light": 1}	Zone's local loads weightage

5.5.3 Task Level

The threshold price point, minimum power, and maximum power for various energy-consuming devices at the task level are listed in Table 5-6.

Table 5-6: Task Level devices default configurations for threshold price, min and max power

Sl. No.	Device	Threshold Price Point*	Minimum Power (W)	Maximum Power (W)	Remarks
1	SmartHub				
1.1	Fan	0.95	3	8	Controlled based on price function (refer subsection B)
1.2	Light	0.95	3	10	
2	Radiant Cubicle	NA	50	300	
3	SmartStrip				
3.1	Plug 1	0.35	0	30	Mobile phone charging
3.2	Plug 2	0.50	0	150	Secondary LED monitor
3.3	Plug 3	0.75	0	150	Laptop
3.4	Plug 4	0.95	49	50	Power for SmartHub

* The users can dynamically change the threshold price points to suit their needs.

The agents that are of relevance to the current experiment at the task level are A) Price Controller Agent, B) SmartHub Agent (Price Agent and Device Controller Agent for SmartHub), C) Radiant Cubicle, and D) SmartStrip. The key configuration parameters used for these agents are mentioned in the following subsections.

A. Price Controller Agent (PCA)

The key configuration parameter values for PCA at the task level are listed in Table 5-7.

Table 5-7: Task Level PCA configuration

Sl. No.	Configuration Parameter	Value	Remark
1	mode_pass_on_params		
1.1	bid_timeout	10 seconds	
1.2	weight_factors	[0.3, 0.5, 0.2]	The budget is divided as follows: 30% to the SmartHub, 50% to the Radiant Cubicle, and 20% to the SmartStrip.
2	mode_default_opt_params		
2.1	publish_optimal	FALSE	Building Level PCA is authorised to conclude the auctioning.
2.2	us_bid_timeout	30 seconds	The max period for the bidding process to be completed is set to 0.5 min.
2.3	lc_bid_timeout	6 seconds	The max period for each bidding iteration. This parameter is the max period in which the downstream PCA should conclude its local bidding and respond with its energy

Sl. No.	Configuration Parameter	Value	Remark
			demand bid.
2.4	max_iterations	30 counts	Allowed number of maximum iterations.
2.5	max_repeats	10 counts	This parameter represents the maximum allowed iterations that result in no change in the price in consecutive messages.
2.5	deadbands	[10, 10, 10]	Energy demand deadbands.
2.7	gammas	[0.0794, 0.0040, 0.0030]	Step size, $\delta = \frac{max_{price} - min_{price}}{max_{energy} - min_{energy}}$
2.8	alphas	[0.0035, 0.0035, 0.0035]	Learning rates
2.9	weight_factors	[0.3, 0.5, 0.2]	Same as weight_factors for mode_pass_on_params

B. SmartHub (Price Agent)

The fan in the SmartHub is controlled based on price function as defined in Figure 5-23. The appropriate price function is used to compute the setpoint based on the user preference (vis-a-vis Cool, Normal, and Warm). However, the default mode 'Normal' is used for the current experiments. The SmartHub Controller gradient descent parameter configurations are listed in Table 5-8.

Table 5-8: SmartHub Controller gradient descent configuration

Sl. No.	**Configuration Parameter**	**Value**	**Remark**
1	gd_params		
1.1	max_iterations	100 counts	Allowed number of maximum iterations.
1.2	max_repeats	10 counts	This parameter represents the maximum allowed iterations that result in no change in the price in consecutive messages.
1.3	deadbands	5	Energy demand deadbands.
1.4	gammas	{"fan": 0.1786, "light": 0.1429}	Step size, $\delta = \dfrac{max_{price} - min_{price}}{max_{energy} - min_{energy}}$
1.5	weight_factors	{"fan": 0.5, "light": 0.5}	SmartHub' s local loads weightage

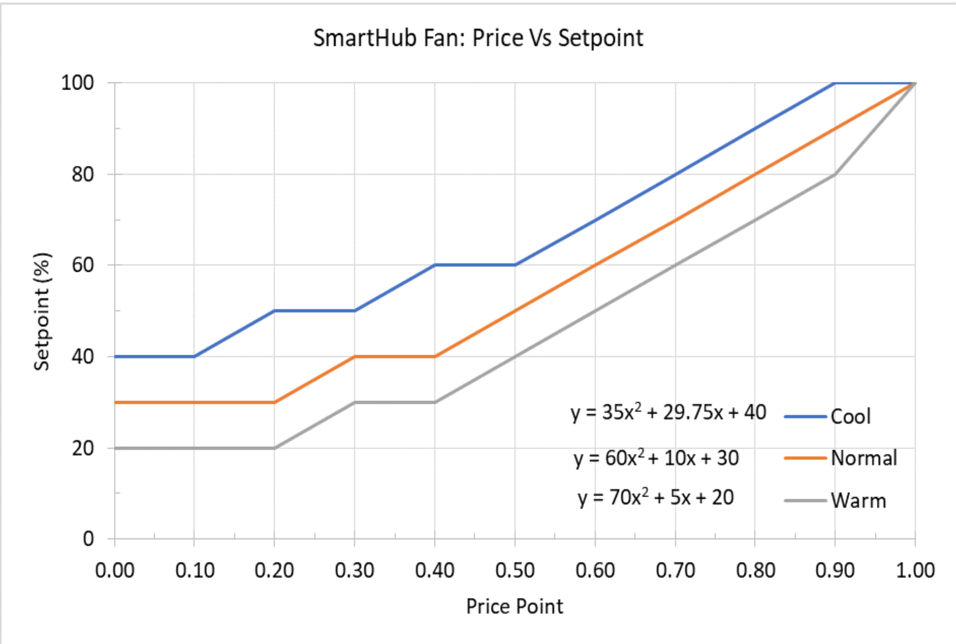


Figure 5-23: Smart Hub's fan price functions.

C. Radiant Cubicle (Price Agent)

The RadiantCubicle is controlled based on price, and energy demand functions and are defined in Figure 5-25 and Figure 5-24, respectively. For the current experiments, default mode 'Normal' is set. The Radiant Cubicle Controller gradient descent parameter configurations are listed in Table 5-9.

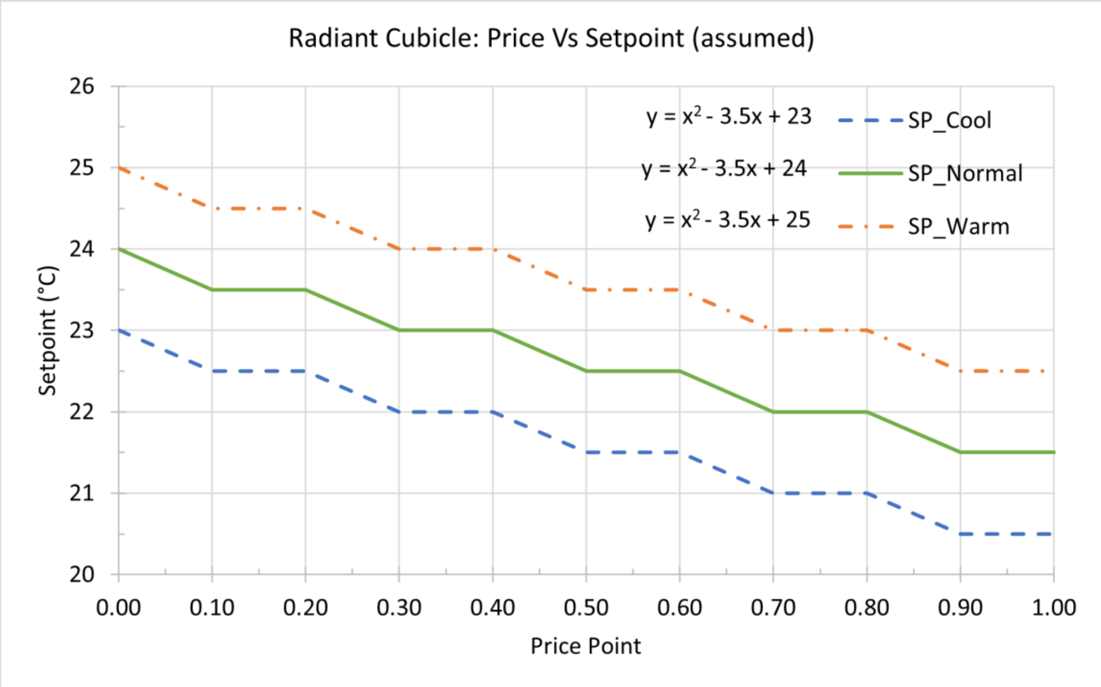


Figure 5-25: RadiantCubicle price functions

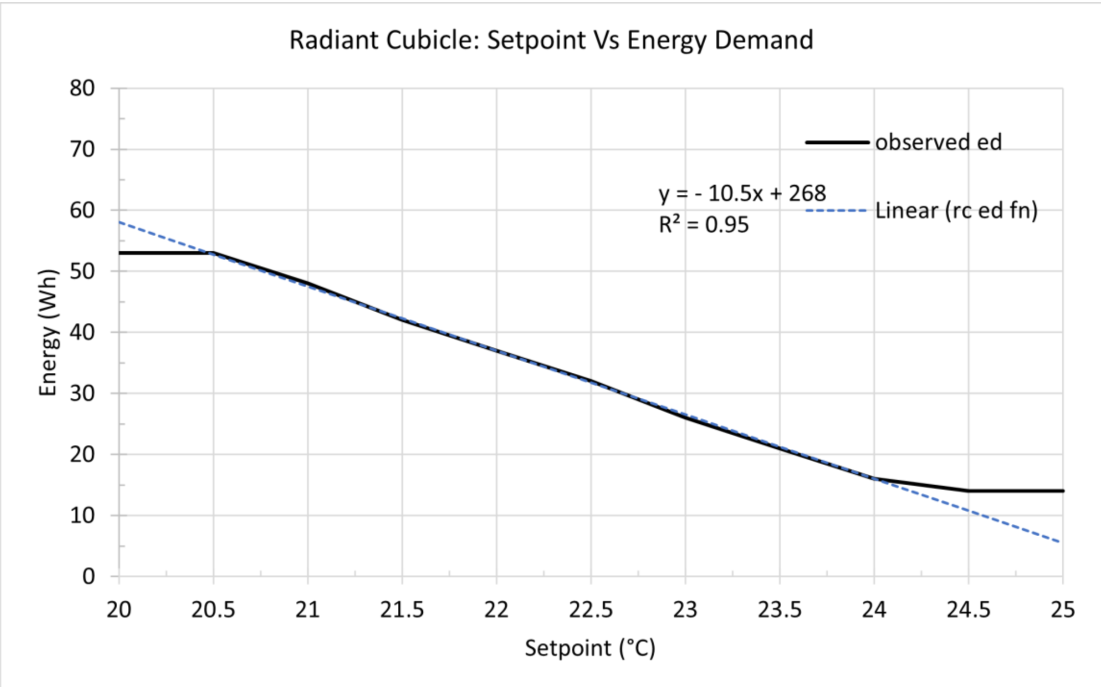


Figure 5-24: RadiantCubicle energy demand functions

Table 5-9: Radiant Cubicle Controller gradient descent configuration

Sl. No.	Configuration Parameter	Value	Remark
1	gd_params		
1.1	max_iterations	100 counts	Allowed number of maximum iterations.
1.2	max_repeats	10 counts	This parameter represents the maximum allowed iterations that result in no change in the price in consecutive messages.
1.3	deadbands	10	Energy demand deadbands.
1.4	gammas	{"rc": 0.0040}	Step size, $\delta = \frac{max_{price} - min_{price}}{max_{energy} - min_{energy}}$
1.5	weight_factors	{"rc": 1}	RadiantCubicle' s local loads weightage

D. SmartStrip

The SmartStrip has a transactive platform instance running on it. The two agents of relevance to the current system evaluation are A) Price Controller Agent, B) SmartStrip Agent (Price Agent and Device Controller Agent for SmartHub). The key configuration parameters used for these agents are as follows:

a. Price Controller Agent (PCA)

The key configuration parameter values for SmartStrip PCA are listed in Table 5-10. However, the PCA is configured for "mode_pass_on" for the experiments.

Table 5-10: SmartStrip PCA configuration

Sl. No.	Configuration Parameter	Value	Remark
1	mode_pass_on_params		
1.1	bid_timeout	5 seconds	
1.2	weight_factors	[0.25, 0.25, 0.25, 0.25]	The budget is divided equally among all the plugs.
2	mode_default_opt_params		
2.1	publish_optimal	FALSE	Building Level PCA is authorised to conclude the auctioning.
2.2	us_bid_timeout	5 seconds	The parameter is the max period in which the bidding process should be completed.
2.3	lc_bid_timeout	2 seconds	The max period for each bidding iteration.
2.4	max_iterations	30 counts	Allowed number of maximum iterations.
2.5	max_repeats	10 counts	This parameter represents the maximum allowed iterations that result in no change in the price in consecutive messages.
2.5	deadbands	[5, 5, 5, 5]	Energy demand deadbands.
2.7	Gammas	[0.0333, 0.0067, 0.0067, 1.0000]	Step size, $\delta = \frac{max_{price} - min_{price}}{max_{energy} - min_{energy}}$
2.8	Alphas	[0.0035, 0.0035,	Learning rates

Sl. No.	Configuration Parameter	Value	Remark
		0.0035, 0.0035]	
2.9	weight_factors	[0.25, 0.25, 0.25, 0.25]	Zone's weightage

b. SmartStrip (Price Agent)

The SmartStrip Controller gradient descent parameters configuration are listed in Table 5-11.

Table 5-11: SmartStrip Controller gradient descent configuration

Sl. No.	Configuration Parameter	Value	Remark
1	gd_params		
1.1	max_iterations	100 counts	Allowed number of maximum iterations.
1.2	max_repeats	10 counts	This parameter represents the maximum allowed iterations that result in no change in the price in consecutive messages.
1.3	deadbands	5	Energy demand deadbands.
1.4	gammas	{"plug1": 0.0333, "plug2": 0.0067, "plug3": 0.0067}	Step size, $\delta = \frac{max_{price} - min_{price}}{max_{energy} - min_{energy}}$
1.5	weight_factors	{"plug1": 0.25, "plug2": 0.25, "plug3": 0.25}	Zone's local loads weightage

The above configuration details show a systematic approach to setting up the iSPACE system. These configurations provided a clear structure for the agents at various levels – building, zone, and task, to communicate, bid, and allocate energy resources in real time. The agent configurations also enable real-time adjustment of energy consumption based on user preferences and operational requirements. The dynamic nature of the configuration allows the agents to be highly responsive and adaptive to changing conditions, optimizing energy utilization, and achieving a balance between demand and supply.

5.6 Experiment

The experiment was conducted over the period of 29th July 2020 to 5th August 2020. Every day before the start of the experiment, the test rooms was run with the default setpoint for 2 hours to attain a stable environment, after which the test experiments were started.

Experiment procedure:

1. **Default setpoint conditioning:** Each day, the test room's default setpoint ran for 2 hours to maintain a stable environment.
2. **Optimal Price/Budget publishing:** Throughout the experiment, a test script published the optimal price/budget for the building at hourly intervals between 08:00 and 18:00 hrs.
3. **Days 1 & 2 - Price-Based Control:**

- Optimal price were published at 1-hour interval as per the schedule show in Figure 5-26 and Figure 5-27, respectively.
- PCAs were set to PASS_MODE ONLINE and PASS_ON_PP mode, which means the optimal price from the primary PCA was directly forwarded to all associated devices.

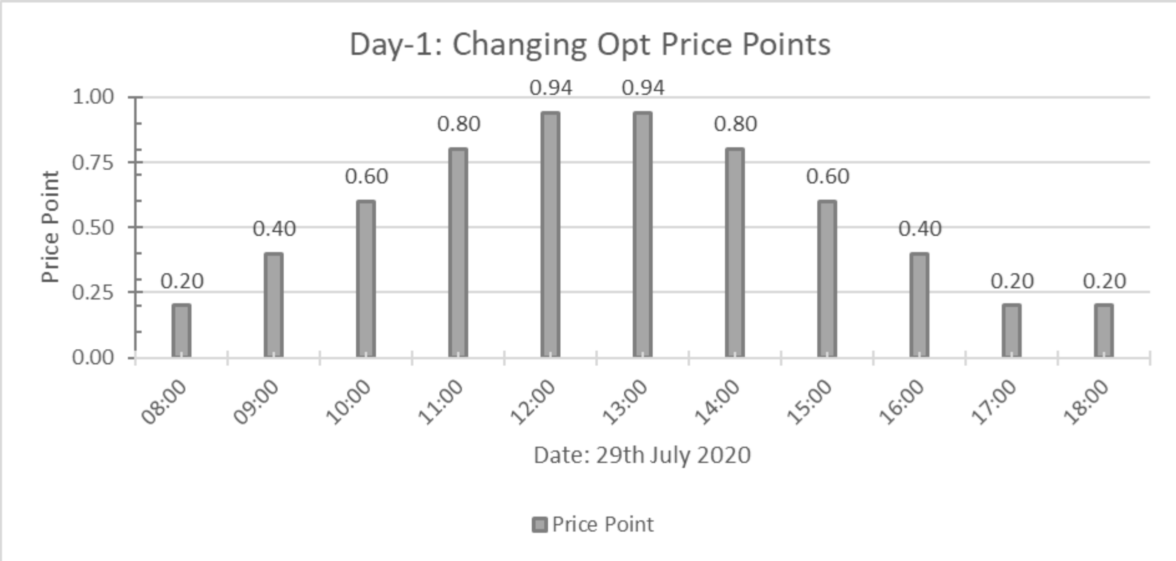


Figure 5-26: Day 1 - changing price published at 1hr Interval

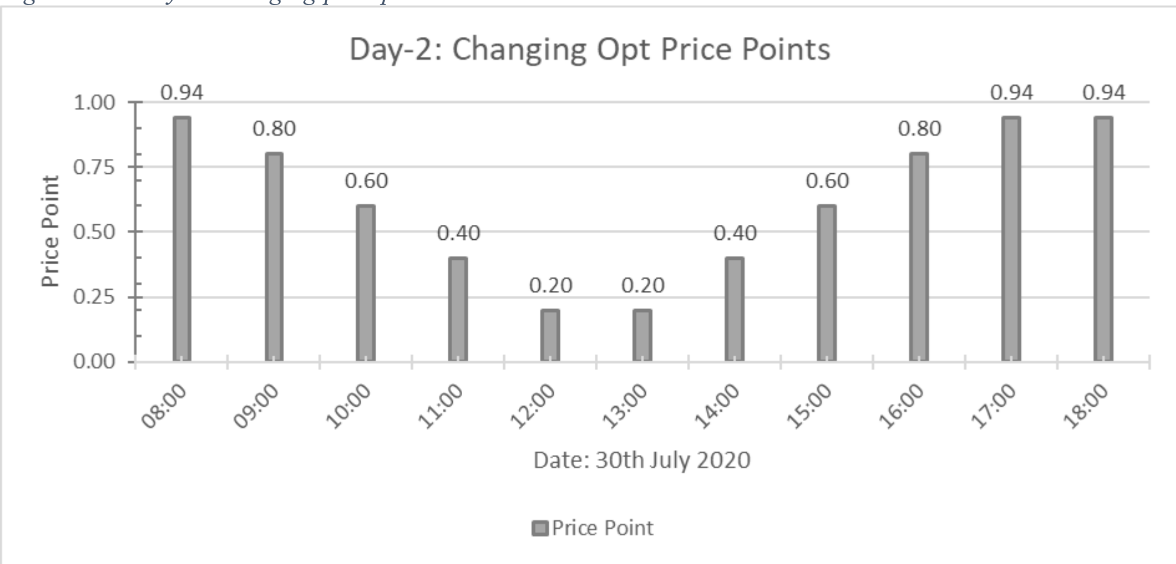


Figure 5-27: Day 2 - changing price published at 1hr Interval

- Anticipated Chain of Actions:
  - The Building PCA relayed the received optimal price to its associated zones.
  - Zone PCAs would then relay these prices to the registered zone-level PAs and the task-level PCAs. Task-level PCAs further distribute this price to task-level PAs.
  - Upon receiving an optimal price, PAs calculated the new setpoints or device states as per their respective price functions.
  - Once the setpoints or states are computed, the corresponding DCAs take over and maintain the setpoints or states.

4. **Day-3, Day-6, & Day-7:**

- On Days 3 and 6, optimal budgets were disseminated hourly and for Day 7, it was every 2 hours as per as per the schedule show in Figure 5-28, Figure 5-29, Figure 5-30 , respectively.
- The PCAs are configured to PASS_MODE ONLINE state and PASS_ON_PP mode (that is, the optimal budget received from upstream PCA is passed on to all the associated devices as per the configured weight_factors).

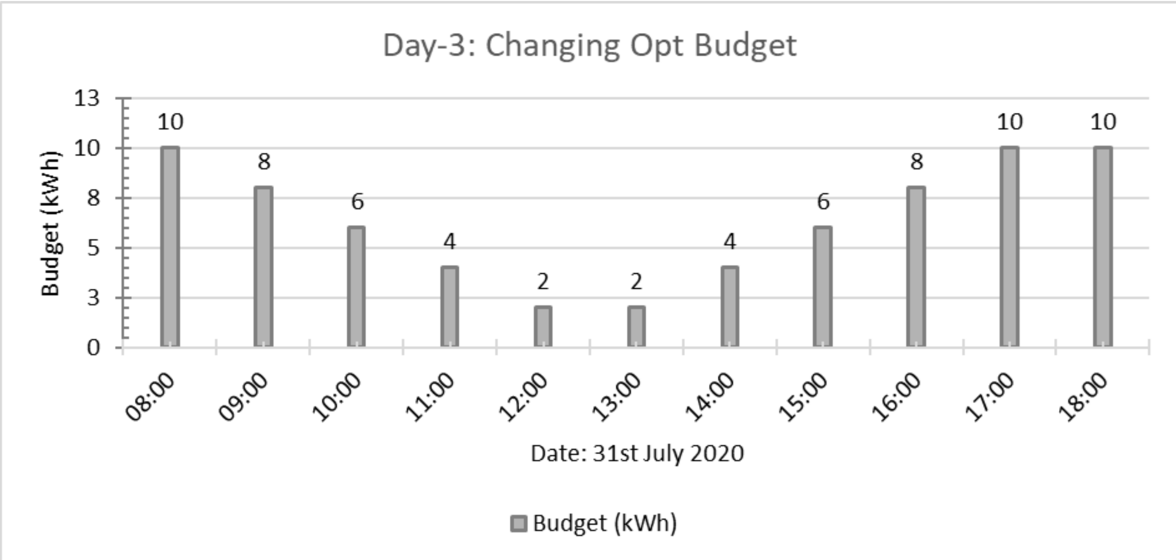


Figure 5-28: Day 3 - changing budget published at 1hr Interval

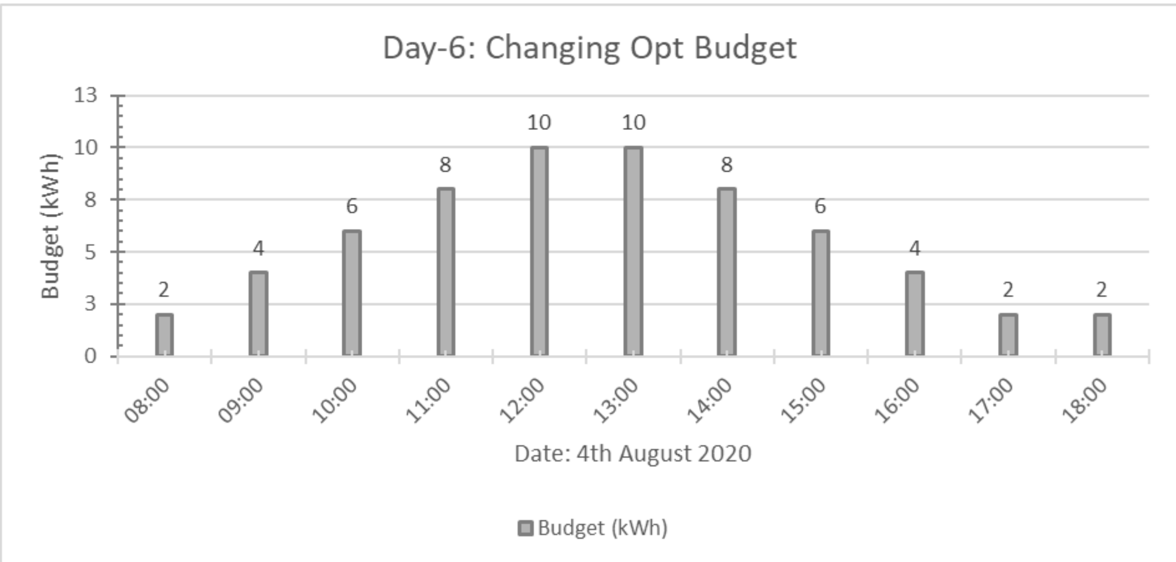


Figure 5-29: Day 6 - changing budget published at 1hr Interval

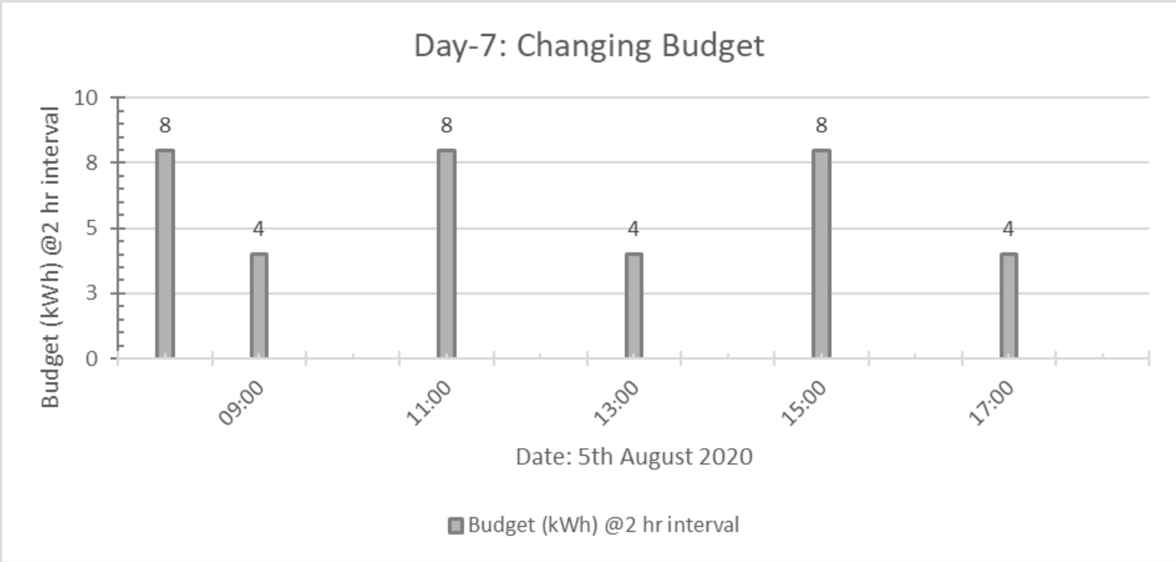


Figure 5-30: Day 7 - changing budget published at 2hr Interval

- Anticipated Chain of Actions:
    - The received budget at the building level was split equally between Zone 1 and Zone 2 PCAs.
    - Zones then distributed their budgets according to preset weight factors [0.75, 0.15, 0.15], that is 72% budget allocated for zone and the rest is divided equally among SmartHub 1 and SmartHub 2 PCAs.
    - Task Level PCAs further divided the budget among task-level PAs. according to preset weight factors [0.3, 0.5, 0.2], that is the budget is divided 30% to the SmartHub, 50% to the Radiant Cubicle, and 20% to the SmartStrip.
    - PAs calculated an optimal price point using the allocated budget, determining device setpoints or states. That is, for the given optimal budget,
        - The Zone PA initially computes a local optimal price points that can operates the zone local loads constrained to the allocated budget. And based on the so computed local optimal price, ambient zone temperature and light setpoints from Figure 5-19 and Figure 5-21, respectively computed.

- Similar iteration is done at the SmartHub PA, the Radiant Cubicle PA, and the SmartStrip PA. Based on their respective local optimal price point, the corresponding setpoint or states computed.
- Once the setpoints or states are computed, the corresponding DCAs take over and maintain the setpoints or states.

  - This chain of operations ensured that each energy-consuming device is operated based on its allocated budget.

5. **Days 4 & 5 - System Stability Check:**

- Random optimal prices were published hourly, while random bid prices were published every 2 minutes as illustrated in Figure 5-31.
- Being a weekend, a dry run is conducted primarily for assessing iSPACE system stability and MAS performance. Ambient systems were offline, which means that even though the agents computed setpoints, the BMS did not maintain them.

6. **Continuous Monitoring:** Throughout the experiment, all power-consuming devices relayed their active power data every 30 seconds based on the most recent

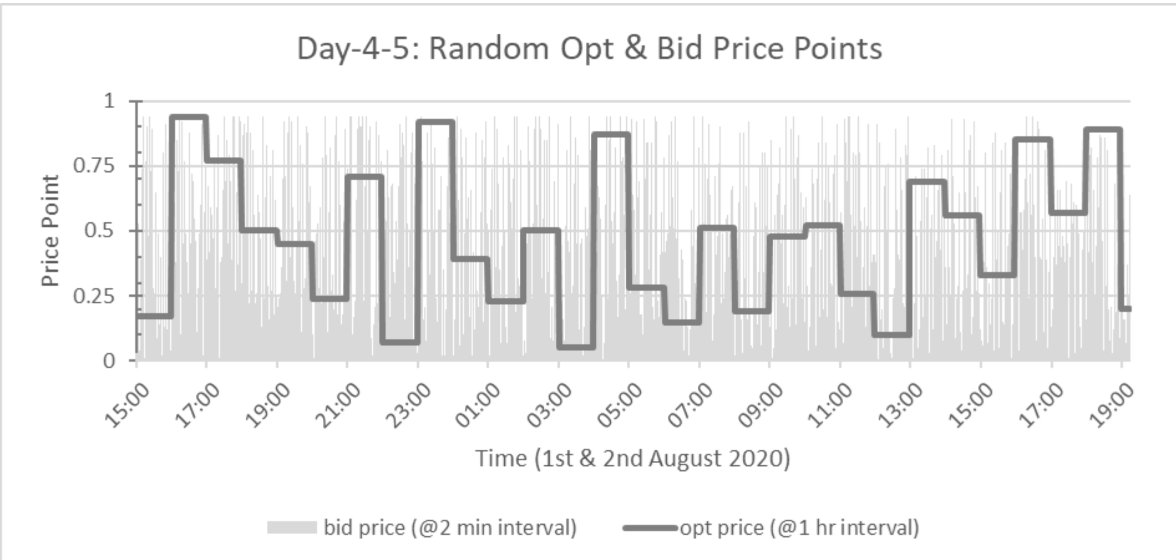


Figure 5-31: Random optimal price @1-hour interval and bid price @2-minutes interval

optimal price. Their corresponding upstream PCAs then computed and published total active power. For further analysis, all reporting messages were logged.

7. **Manual Interventions:** During the experiment, the system was under surveillance for necessary manual interventions. For instance, on Day 1, an issue with the RM2 setpoint update in the BMS at 09:45 for the 10:00 slot was noted, due to a Schneider workbench glitch. This was a one-time occurrence. Manual intervention was required to continue the experiment.
8. **Summary:** The pilot deployment of the iSPACE system highlighted the features and anticipated actions of both PCA and PA under varying configurations. These were detailed in Table 5-12 and Table 5-13.

In the following subsections, the data from these experiments and the overall performance of pilot deployment of the iSPACE system have been analysed in detail to ascertain its effectiveness.

Table 5-12: iSPACE pilot deployment – implemented features and expected behaviour for various Price Controller Agents (PCA)

Sl. No.	Parameters *State*	*Mode*	*Message Type*	*Is optimal Price/Budget Message*	Expected behaviour
1	online	pp_pass_on	price point	yes	PCA agent pass on this message to end devices. End devices apply a pricing policy for this price
2	online	pp_pass_on	price point	no	PCA agent pass on this message to end devices. End devices respond with a bid energy corresponding to this price point. PCA agent aggregates this bid energy demand
3	online	pp_pass_on	budget	yes	PCA computes new budgets for local and downstream devices according to the weights. Local and downstream devices compute a new price point that corresponds to this budget and applies to price policy for the new computed price point
4	online	pp_pass_on	budget	no	PCA computes new budgets for local and downstream devices according to the weights. End device behaviour is the same as previous, i.e., is_opt true case

5	online	default_opt	price point	yes	PCA agent pass on this message to end devices End devices apply a pricing policy for this price point
6	online	default_opt	budget	yes	PCA computes new budgets for local and downstream devices according to the weights. Local and downstream devices compute a new price point that corresponds to this budget and applies price policy for the new computed price point
7	online	default_opt	price point	no	new_act_pwr = old_act_pwr * (old_dur / new_dur) * (old_pp / new_pp) new_energy_demand = calc_energy_wh(new_act_pwr, new_dur_sec)
8	online	default_opt	budget	no	new_energy_demand = wt_factor * budget
9	online	extern_opt	price point	yes	Not Implemented
10	online	extern_opt	budget	yes	Not Implemented
11	online	extern_opt	price point	no	Not Implemented
12	online	extern_opt	budget	no	Not Implemented
13	standalone	-	-	-	will not participate in u/s bidding
14	standby	-	-	-	actively report, no action

Table 5-13: iSPACE pilot deployemt – implemented features and expected behaviour for various Price Agents (PA)

Sl. No.	***Parameters*** ***Message Type***	***Is optimal Price/Budget Message***	**Expected behaviour**
1	price point	yes	Apply pricing policy for this price point
2	price point	no	Respond with energy demand corresponding to this price point
3	budget	yes	Compute a new price point corresponding to this budget using Gradient Descent $new\_pp = old\_pp - \sum \delta * (new\_ed - old\_ed)$ Apply pricing policy for this price point
4	budget	no	This scenario is not applicable; handle spurious message of this type by ignoring the message

5.7 Data Analysis and Key Insights

Evaluating the efficacy and operability of the proposed system necessitates an intricate and multifaceted analysis. The evaluation criteria outlined in this thesis aim to encompass various essential facets, guaranteeing an in-depth and holistic comprehension of the system. The specific evaluation components include:

1. **System's Functional Properties Evaluation**: To determine if the system operates as intended and meets all functional requirements. This typically involves testing the system in a controlled environment, observing its behaviour, and checking its adherence to the design and requirement specifications. It is crucial to meet the system's core functionalities.
2. **Non-Functional Properties Evaluation**: To analyse the system's attributes that are not related to its functionality but pertain to reliability, scalability, and maintainability. The performance of the Multi-Agent System (MAS) entails measuring metrics like convergence rate (how quickly agents reach a consensus) and agents' response metrics (how quickly and accurately agents respond to various stimuli or changes). A high-performing MAS is characterized by quick convergence rates and efficient agent responses, indicating that the system can handle changes effectively and that agents can work together harmoniously.
3. **Energy Demand Evaluation**: To assess the system's efficiency in terms of energy demand. This would involve comparing energy demand before and after the system's implementation and using benchmarks or standards as reference points. A significant reduction in energy demand post-implementation would indicate the system's effectiveness in promoting efficient demand response strategies.

Through a combination of above quantitative and qualitative evaluations, a comprehensive understanding of the system's strengths, potential areas for improvement, and its impact on energy efficiency have been discussed in the subsequent section.

Note: During the data extraction for analysis of the log files, it had been observed that a few auditing messages are not logged which is identified as an OS filesystem I/O buffering issue especially in the low-end compute modules. Also, the BTU meters at times gave spurious readings (that is the readings are not within the BTU meters specs range). For example, on Day-1, only five auditing messages are missing, and 3 energy reporting messages were spurious against a total of 20K records. In such a scenario, the previous 30sec record was used.

5.7.1 System's Functional Properties Evaluation

In this section, we analyse the system's behaviour during experimentation. The analysis focuses on the reactions of Price Agents (PAs) and Price Control Agents (PCAs) as they interact with the control signals introduced at the building level. The published control signals at building level, optimal price and budget signals were detailed in the preceding section 5.6.

The investigation is categorised into distinct subsections, each emphasising different facets of the PA and PCA agents' behaviour when subjected to these control signals, which is pivotal in comprehending the broader functional properties of the system at hand. The analysis provides the systems' operational dynamics and intricacies and their current efficacy and hints at future enhancements, simulation studies, and optimisations.

A. Price Signal

Central to this thesis is a market-based control strategy, wherein "*price*" becomes a critical operational parameter. This section explores understanding the system's responsiveness to the published price control signals in this context. This analysis aims to understand how price fluctuations influence system behaviours, clarifying the relationship between price and system control.

a. PA Price Control and Response

Consistent Price Reception: All Price Agents (PAs) across the various levels received the optimal price control signal at regular intervals, as seen by the dark green lines in Figure 5-32 and Figure 5-33 (SmartHub PA), Figure 5-35 (RadiantCubicle PA), Figure 5-38 to V-42 (SmartStrip PA), and Figure 5-43 (Zone PA). Such consistent signalling is crucial for achieving a stable and predictable response from the agents.

Agent Control Behaviours: The Price Agents (PAs) at all levels exhibited a control response in line with the received price signal, as visualized in SmartHub (Figure 5-33), RadiantCubicle (Figure 5-36 and Figure 5-37), SmartStrip (Figure 5-38, Figure 5-39, Figure 5-40, and Figure 5-41), and Zone PA (Figure 5-44).

i. SmartHub

Figure 5-32 displays the price control signal received by the SmartHub PA from Task-level PCA for 08:00 to 18:00 hrs. Concurrently, the SmartHub managed the Fan coordinated with the price function detailed in Figure 5-23, a relationship further illustrated in Figure 5-33. At 10:00 hrs in Figure 5-32, a price shift from 0.40 to 0.60 is evident. Referring to Figure 5-23, this change correlates to a 50% fan set point. This adjustment is mirrored in Figure 5-33, where the set point transitions from 30% to 50%. In Figure 5-33, “LedThPP” and

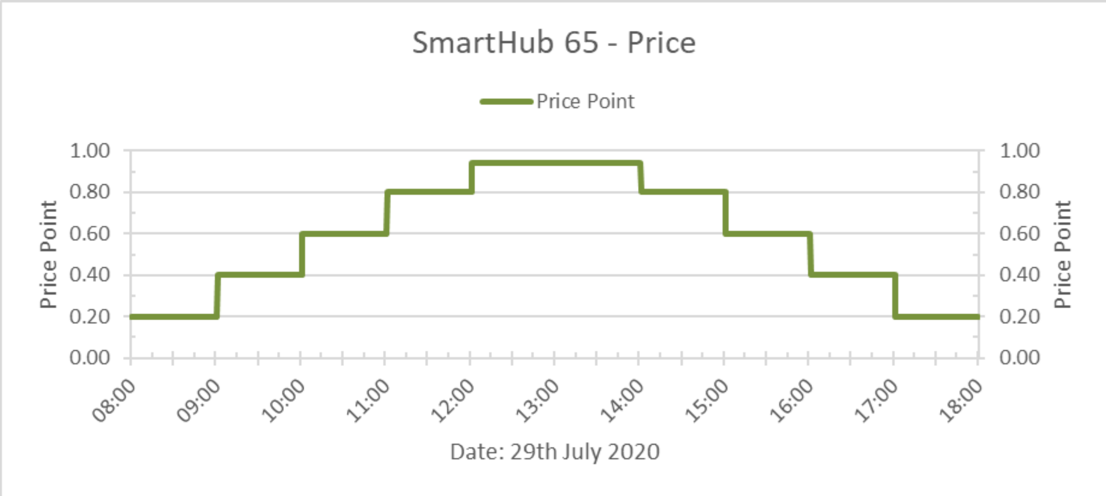


Figure 5-32: Price control signal received by SmartHub PA at Level-3

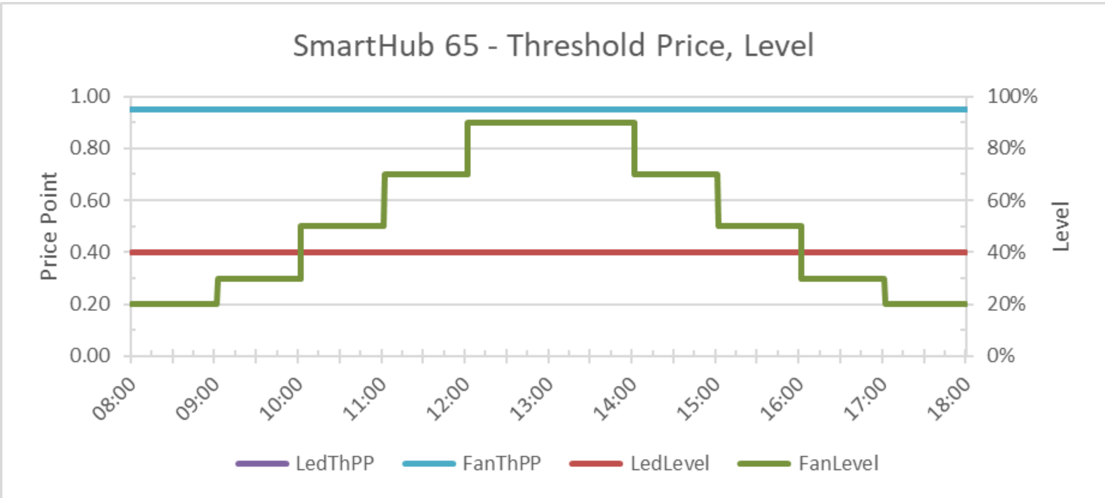


Figure 5-33: SmartHub PA at Level-3 controlling the fan level in response to the price control signal

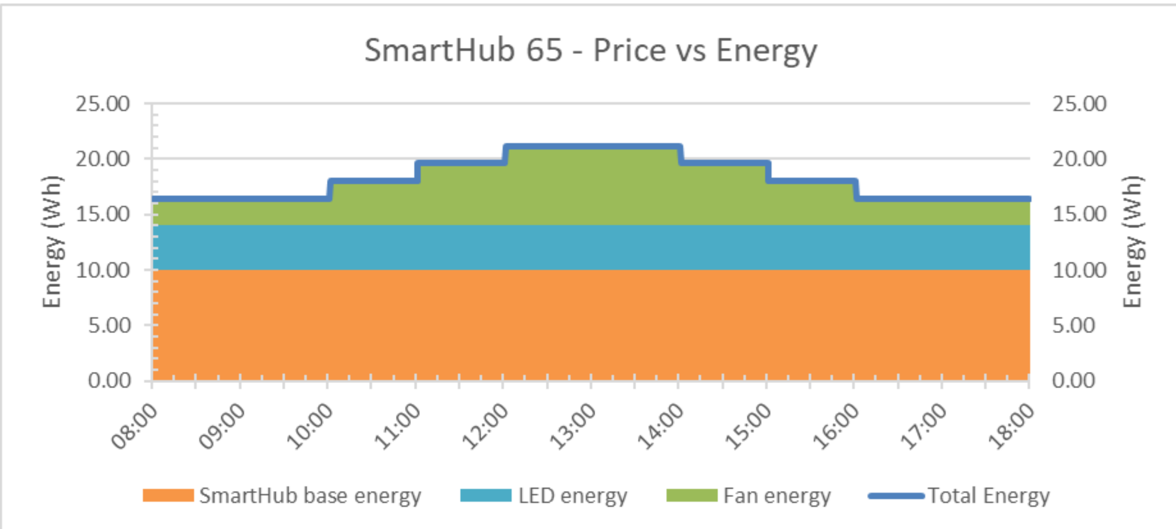


Figure 5-34: SmartHub PA at Level-3 total energy consumption in response to the price control signal

“FanThPP” represent the threshold price point set for the SmartHub’s Led and Fan, respectively. And “LedLevel” and “FanLevel” represent their set point. Figure 5-34 subsequently charts the energy usage by the SmartHub PA corresponding to the price changes presented in Figure 5-32.

It is noteworthy to mention that on Day-1 of the experiment, an operating system malfunction occurred with one of the SmartHubs. This SmartHub, having seen rigorous utilization during the FDD lab benchmark, had its OS reinstalled, resuming its role in the subsequent experiment slot. Excessive RAM use might have precipitated this issue. All the SmartHubs and SmartStrip were reprogrammed that evening, and no further incidents were reported during the remainder of the experiment.

ii. RadiantCubicle

Figure 5-35 displays the price control signal received by the RadiantCubicle PA from Task-level PCA for 08:00 to 18:00 hrs. Concurrently, the RadiantCubicle managed the surface temperature coordinated with the price function detailed in Figure 5-25, a relationship further illustrated in Figure 5-36. At 10:00 hrs in Figure 5-35, a price shift from 0.40 to 0.60 is evident. Referring to Figure 5-25, this change correlates to a 21.5°C surface temperature set point. This adjustment is mirrored in Figure 5-36, where the set point transitions from 22°C to 21.5°C simultaneously. Figure 5-37 subsequently charts the energy usage by the RadiantCubicle PA corresponding to the price changes presented in Figure 5-35.

In Figure 5-36, a dashed red line also depicts the room's ambient temperature. It was observed that room temperature significantly impacts the RadiantCubicle's controllability, leading to notable variability in its active power, as highlighted by the green dotted line in Figure 5-37.

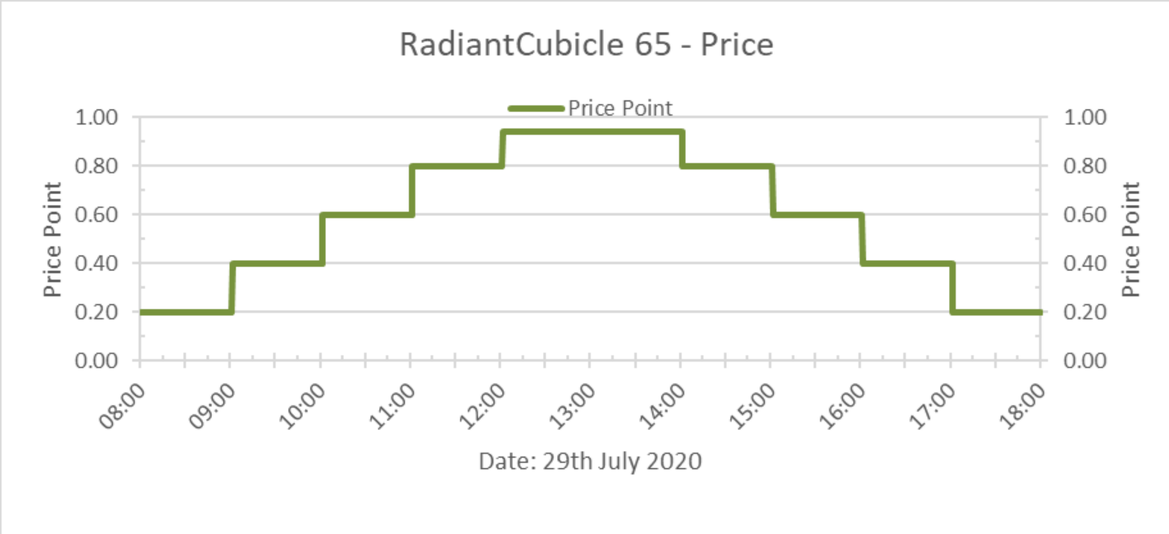


Figure 5-35: Price control signal received by RadiantCubicle PA at Level-3

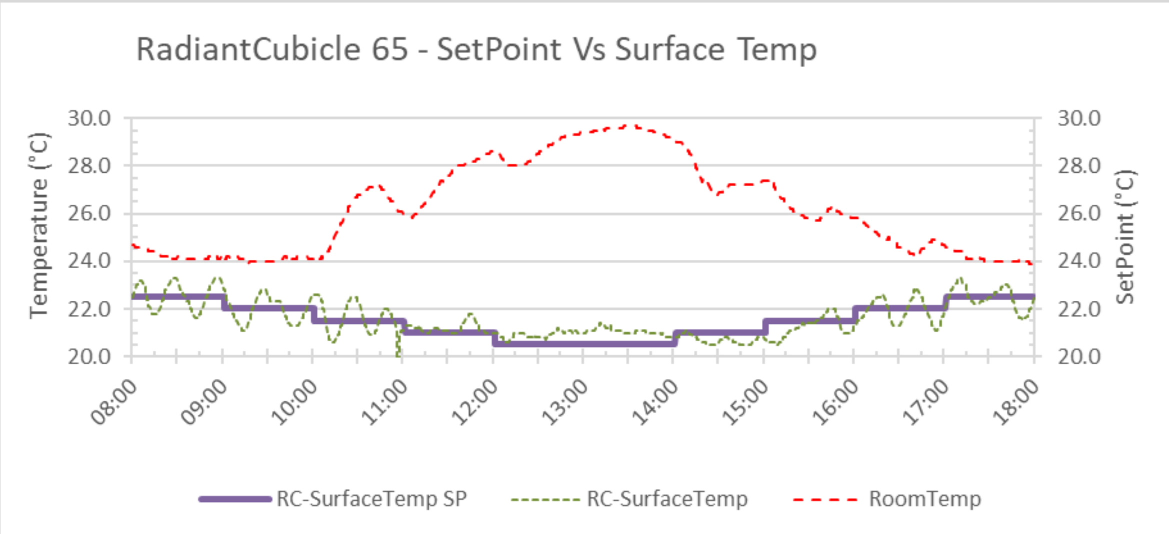


Figure 5-36: RadiantCubicle PA at Level-3 controlling the surface temperature SP in response to the price control signal

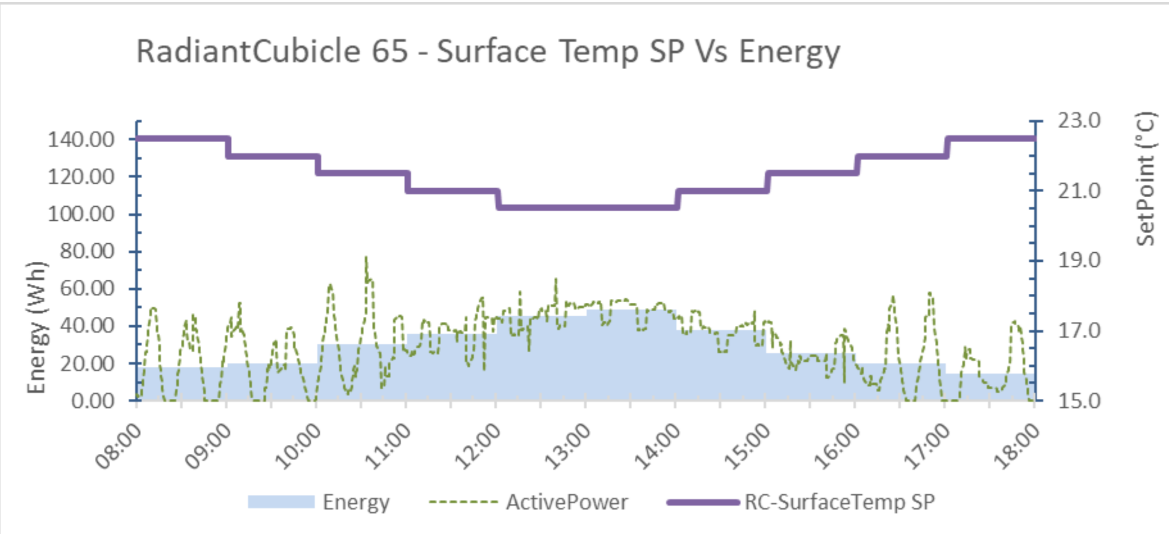


Figure 5-37: RadiantCubicle PA at Level-3 surface temperature on respose to changed SP and the energy consumption in response to the price control signal

This fluctuation becomes especially marked when lower room temperatures coincide with a higher surface temperature set point for the RadiantCubicle, as opposed to scenarios with higher room temperatures and a reduced surface temperature setpoint. Refining the PID controller parameters for the RadiantCubicle would be beneficial to stabilise this inconsistency.

iii. SmartStrip

The SmartStrip operates by toggling its plugs on or off according to the designated threshold price points listed in Table 5-6, Sl. No 3. Figure 5-38 through V 42 present the state transitions (On/Off) for Plug-1, Plug-2, Plug-3, and Plug-4 of the SmartStrip, respectively.

These transitions are subject to the fluctuating price control signals received from the Task Level PCA and align with the stipulated thresholds. Figure 5-42 charts the energy usage by the SmartStrip PA corresponding to the price changes. What's evident from these figures is the SmartStrips agility in adjusting its operations in real-time.

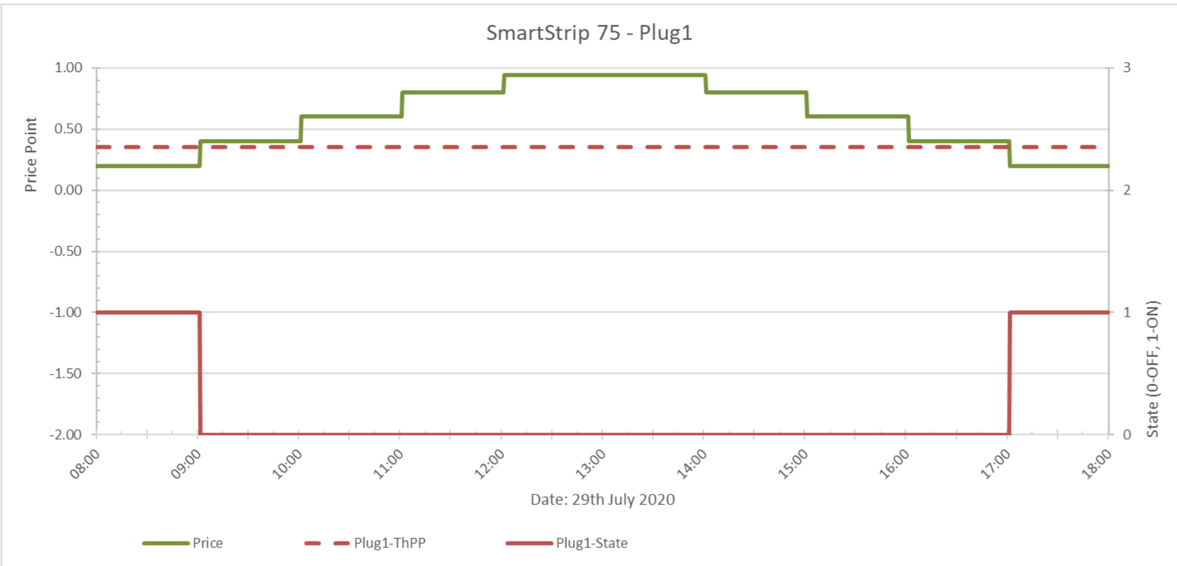


Figure 5-38: SmartStrip PA at Level-3 controlling the plug 1 in response to the price control signal

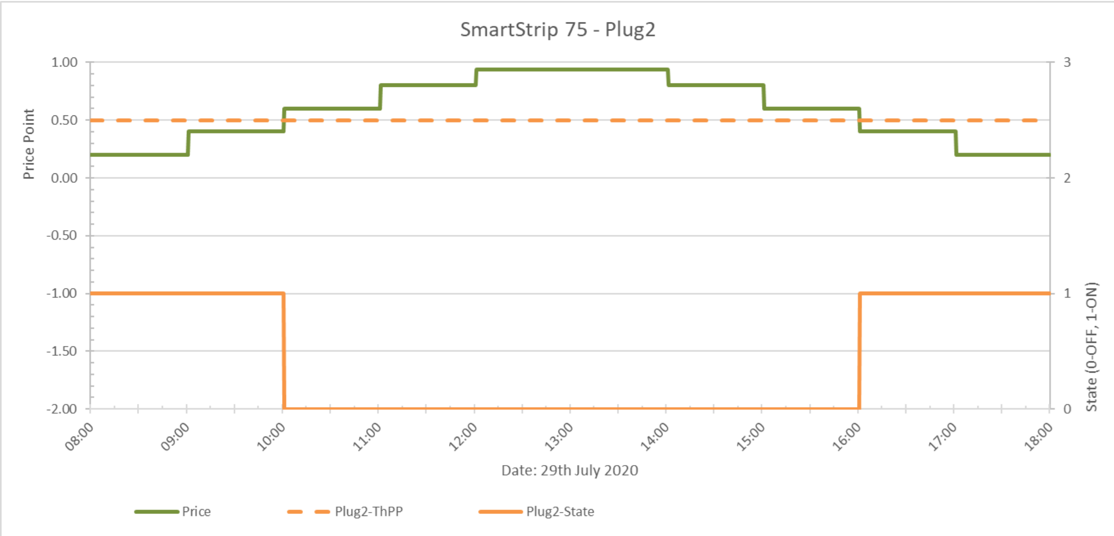


Figure 5-39: SmartStrip PA at Level-3 controlling the plug 2 in response to the price control signal

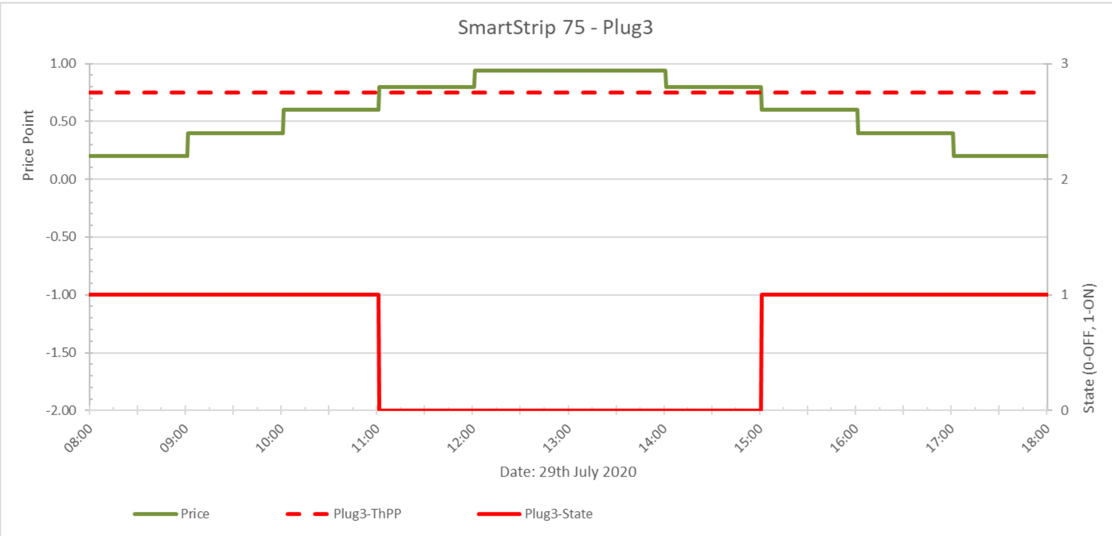


Figure 5-40: SmartStrip PA at Level-3 controlling the plug 3 in response to the price control signal

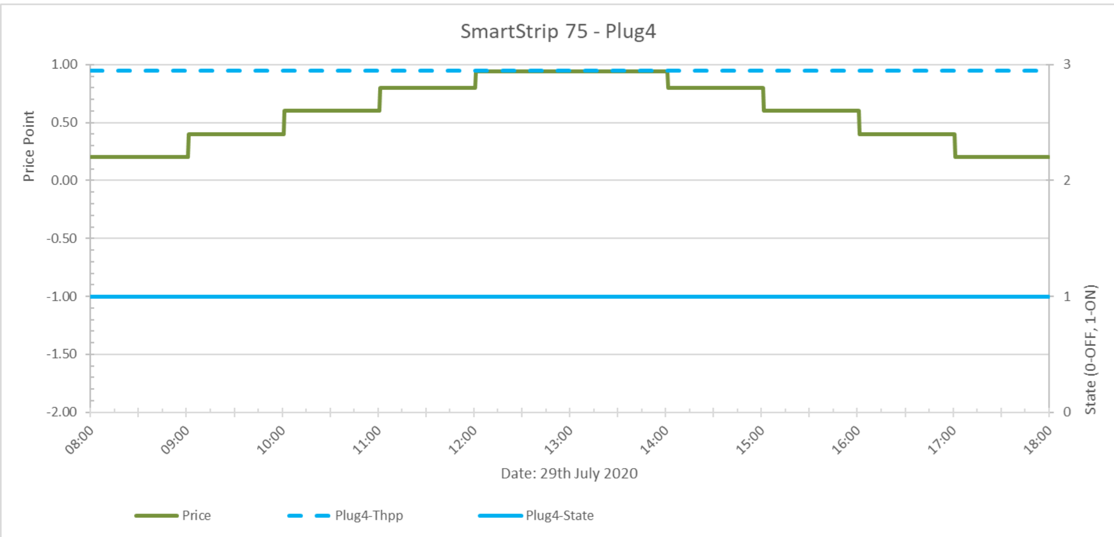


Figure 5-41: SmartStrip PA at Level-3 controlling the plug 4 in response to the price control signal

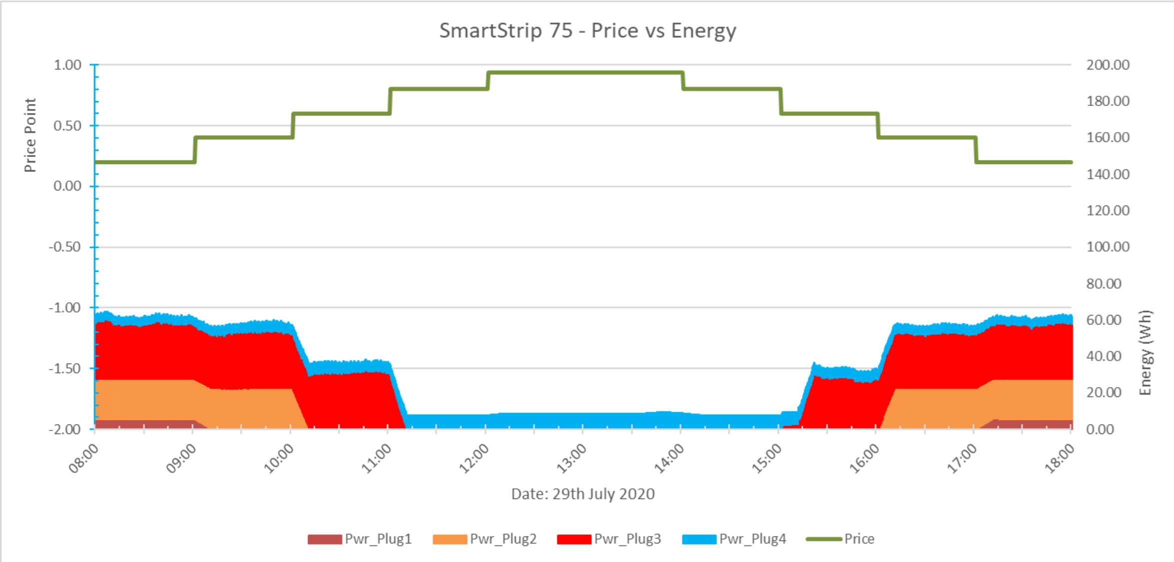


Figure 5-42: SmartStrip PA at Level-3 energy consumption in response to the price control signal

During the period from 11:00 hrs to 15:00 hrs, elevated price points at the Task level prompted the SmartStrip to cut power to three of its plugs. Consequently, the laptop batteries connected to these plugs were fully drained, shutting down the laptops. Once the price decreased at 15:00 hrs, a notable surge in power consumption was observed (Figure 5-42). Even though the laptops remained off, their batteries began to recharge. It is imperative to strategize against extended high-price scenarios to ensure more efficient control and reduce such sudden surges in demand.

iv. Ambient Controls (Zone AC, Light)

Figure 5-43 displays the price control signal received by the Zone PA from Zone-level PCA for 08:00 to 18:00 hrs. Concurrently, the Zone PA managed the ambient AC and light with the price function detailed in Figure 5-19 and Figure 5-21, a relationship further illustrated in Figure 5-44 and Figure 5-46, respectively. Figure 5-45 and Figure 5-46 charts the respective energy usages corresponding to the price changes.

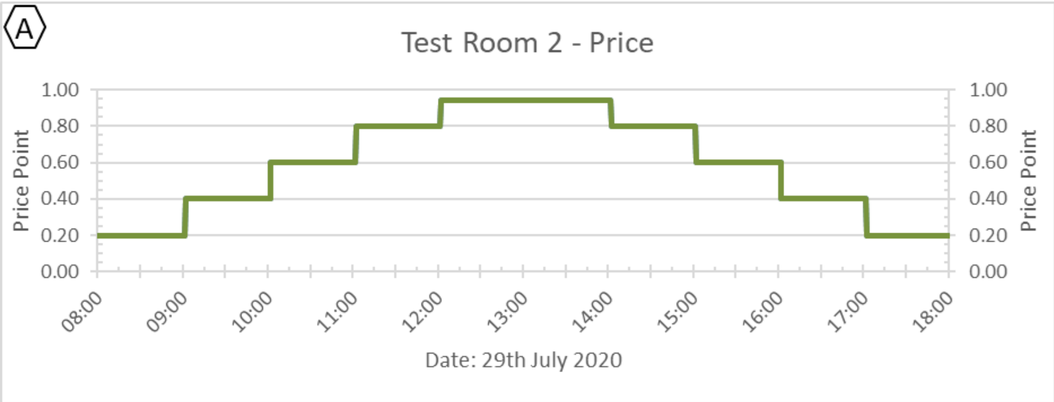


Figure 5-43: Price control signal received by Zone PA at Level-2

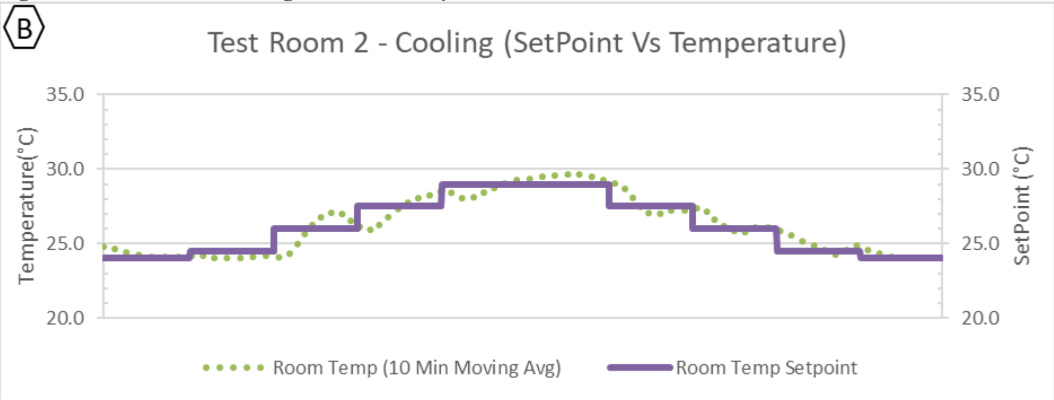


Figure 5-44: Zone PA at Level-2 controlling the room AC SP in response to the price control signal

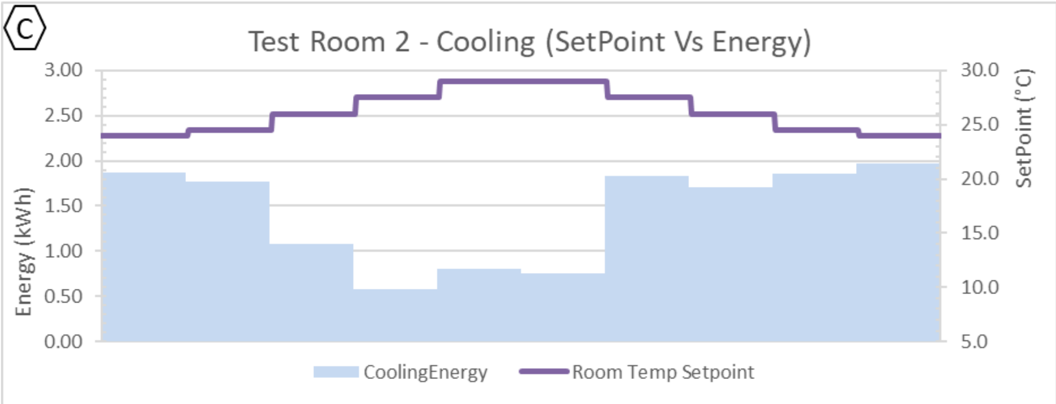


Figure 5-45: Zone PA at Level-2 AC energy consumption to the change in SP, which is in response to the price control signal

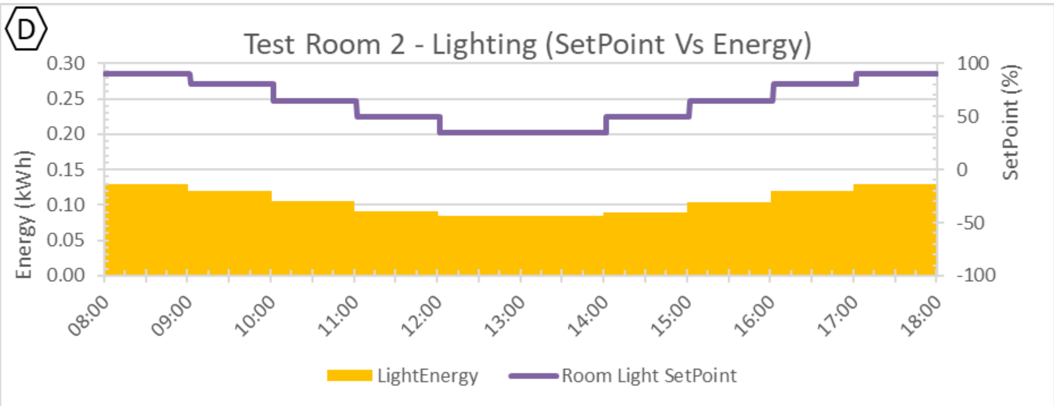


Figure 5-46: Zone PA at Level-2 Light energy consumption to the change in SP, which is in response to the price control signal

Ambient Temperature Regulation: Figure 5-44 highlight that the room temperature (represented by the dashed line) was regulated as per the computed setpoint (continuous line) for the given price point. This illustrates that the temperature control mechanism is working effectively.

In Figure 5-45, it can be observed a sudden surge in energy consumption at 15:00 hrs slot. Due to the high price point in the past two consecutive cycles, the AC setpoint was set remarkably high (29°C), the HVAC system was running idle, and the rooms became warmer. When the price came down, the cooling kicked in across all the zones resulting sudden demand for energy. It is imperative to strategize against extended high-price scenarios.

Lighting Control: As shown in Figure 5-43, the lighting levels were controlled in response to the received price signals. These adjustments adhered to the setpoint outlined by the violet line in Figure 5-46, calculated based on the designated price point from Figure 5-21. Figure 5-46's yellow line also illustrates the concurrent energy consumption resulting from these lighting adjustments.

b. PCA Price Control and Response

Price Controller Agents (PCAs) across different hierarchical tiers consistently received their designated optimal price control signals. As illustrated by the dark green lines in Figure 5-47 for Building Level PCA, Figure 5-48 for Zone Level PCA, and Figure 5-49 for Task Level PCA, these signals were not only rerouted to their associated Personal Agent (PA) (as previously outlined) but also propagated to succeeding PCA layers. Specifically, signals flowed from the Building Level PCA to Zone Level PCAs and then from Zone Level PCAs to Task Level PCAs.

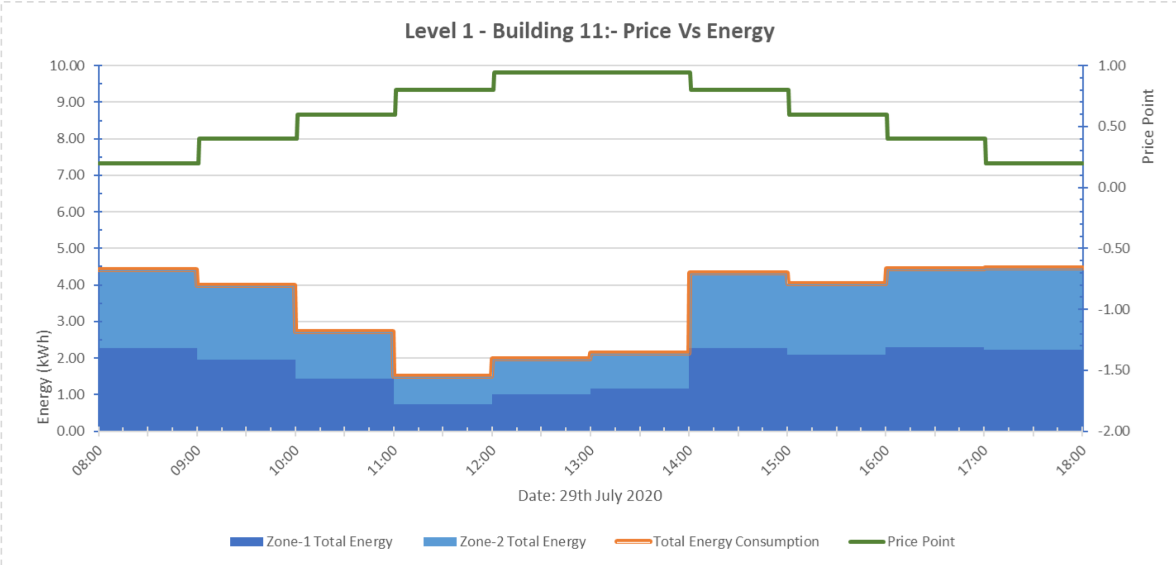


Figure 5-47: Building PCA at Level-1 total energy consumption in response to the price control signal

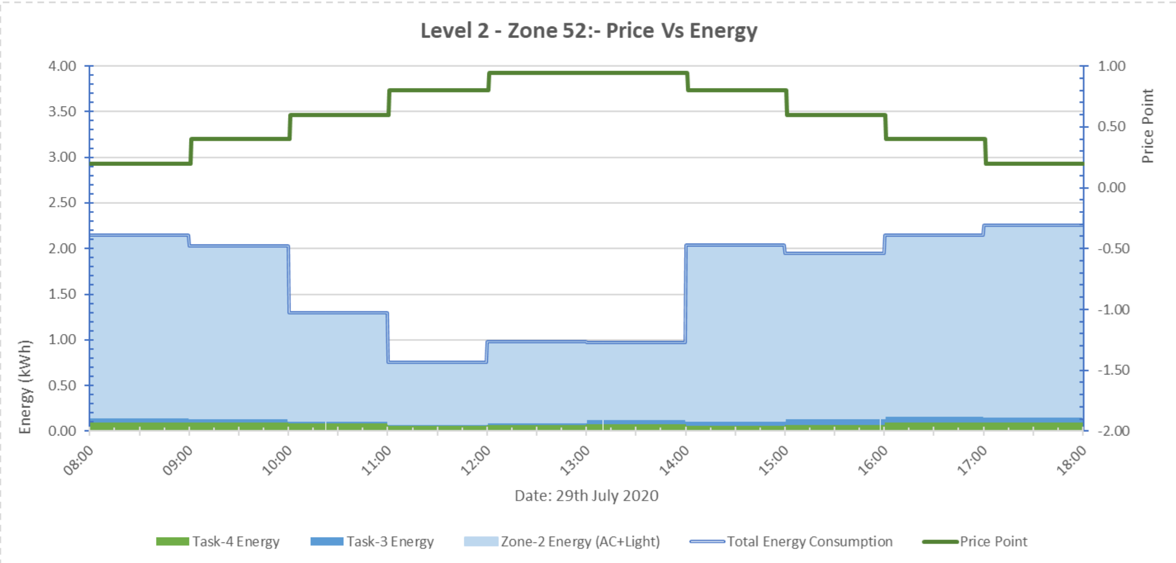


Figure 5-48: Zone PCA at Level-2 total energy consumption in response to the price control signal

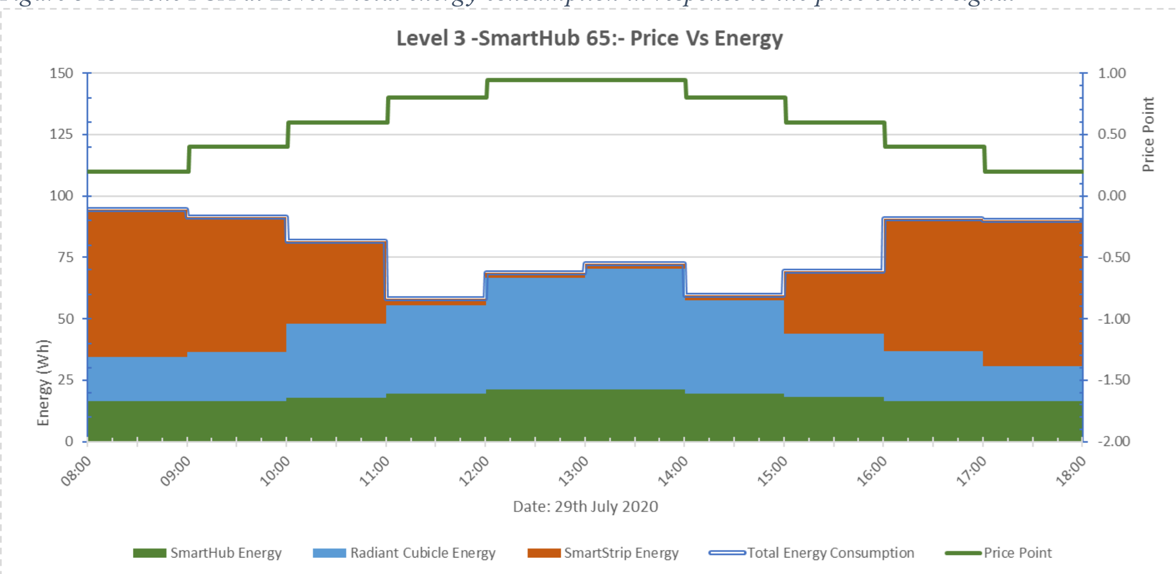


Figure 5-49: SmartHub PCA at Level-3 total energy consumption in response to the price control signal

In addition, PCAs collated energy consumption from their connected devices. This collated energy consumption facilitated the computation of an aggregate energy consumption profile for each of their operational tiers. Observations corroborated our initial suppositions: when prices ascend, there is a noticeable downtrend in the cumulative energy usage at the building and zone levels, and vice-versa. However, price increases corresponded with increased energy consumption at the task level.

B. Budget Signal

This section explores understanding the system's responsiveness to the published budget control signals. Beyond responding to price control signals, the system is designed for budget control signals. When a budget is published and the PCA operating in PASS_ON mode, allocations are made per the pre-established weights of the associated devices. Each PA agent, representing their energy-consuming device, then deduces an ideal price to ensure energy consumption remains within this allocated budget.

In contrast, when the PCA is set to Optimization Mode (encompassing both DEFAULT_OPT and EXTERN_OPT modes), upon receiving a budget from a higher-tier PCA controller, it collaborates with both local and downstream devices. This collaboration results in computing an optimal price that keeps the cumulative energy consumption within the allocated budget. This computed optimal price is then published to all interconnected devices.

Consistent Budget Reception: All Price Controller Agents (PCAs) across the various levels received the budget control signal at regular intervals per schedule illustrated in Figure 5-28, Figure 5-29, Figure 5-30. Such consistent signalling is crucial for achieving a stable and predictable response from the agents.

Agent Control Behaviours: Table 5-3, Table 5-4, and Table 5-7 illustrate the configuration Building, Zone, and Task Level PCA, the respectively. Corresponding these configurations, each PCA distributed the received budget among the associated PAs.

Consider a scenario on Day-7 (Figure 5-30). At 07:00 hours, a building-level budget of 8000 Wh is announced. This budget is equally shared between Zone 1 and Zone 2 PCA, amounting to 4000 Wh each. The Zone PCA then distributes its share: 72% (or 2857 Wh) is allocated for its own PA, while the remainder is evenly divided between two Task-level PCAs (571 Wh to each). These Task-level PCAs then distribute their budget at [0.30, 0.50, 0.20] weights among SmartHub PA, RadiantCubicle PA, and SmartStrip PA, resulting in allocations of [171, 285, 114] Wh, respectively.

The PA agent at the Zone then calculates an ideal local price for its allocation (2857 Wh), arriving at a figure of 0.48. This behaviour is mirrored at 09:00 for a building-level allocation of 4000 Wh, with the Zone PA receiving 1429 Wh and computing a price of 0.72. The corresponding responses in the form of room AC and lighting adjustments to this local optimal price control signal are showcased in Figure 5-50. Further Figure 5-51, Figure 5-52, and Figure 5-53 illustrate the Zone PA controlling the room AC and Light setpoint in response to this local optimal price control signal.

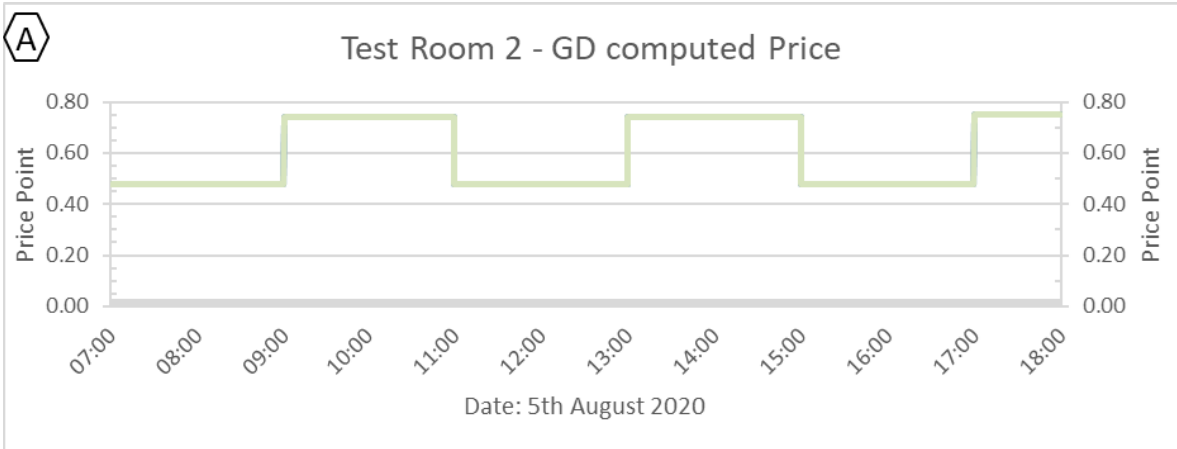


Figure 5-50: Zone PA at Level-2 computed local optimal price in reponse to the budget control signal

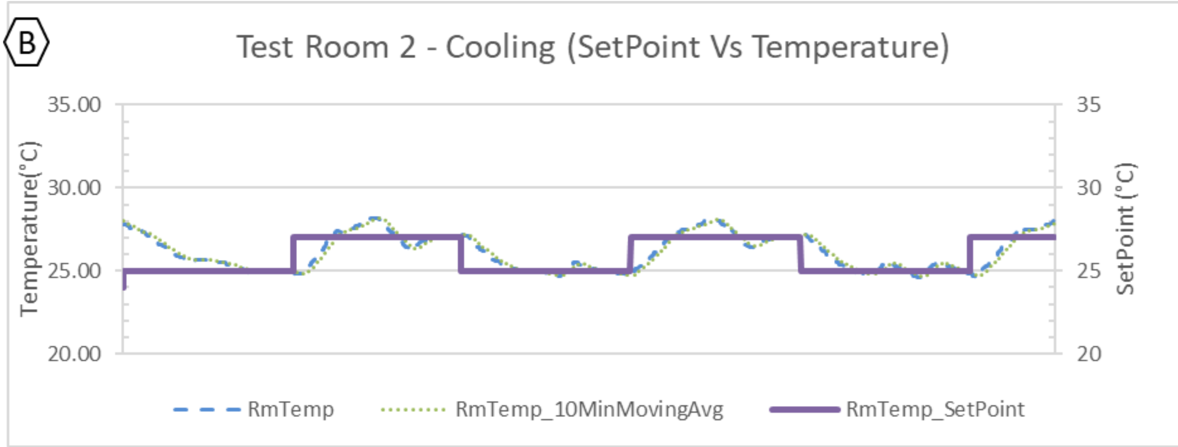


Figure 5-51: Zone PA at Level-2 controlling the room AC SP in response to the budget control signal

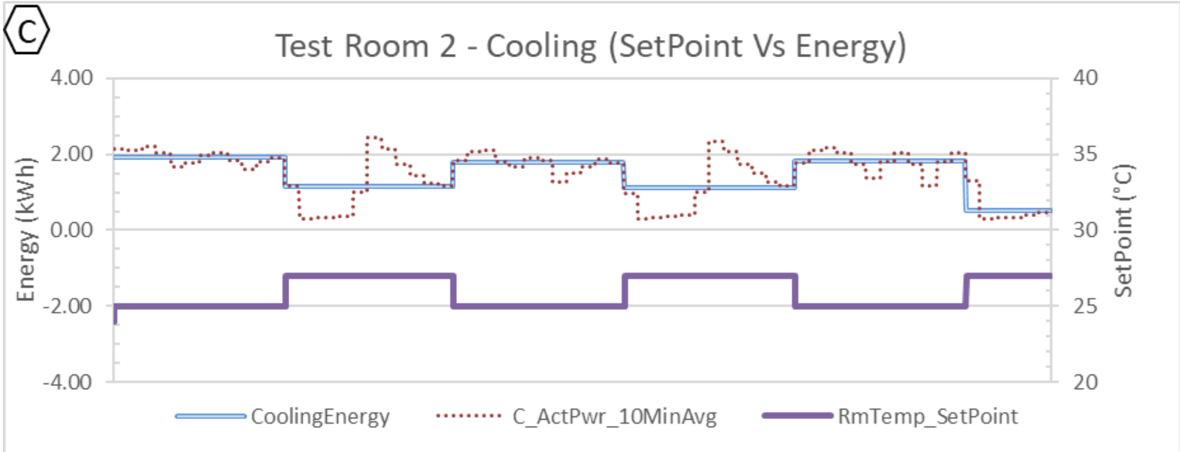


Figure 5-52: Zone PA at Level-2 AC energy consumption to the change in SP, which is in response to the budget control signal

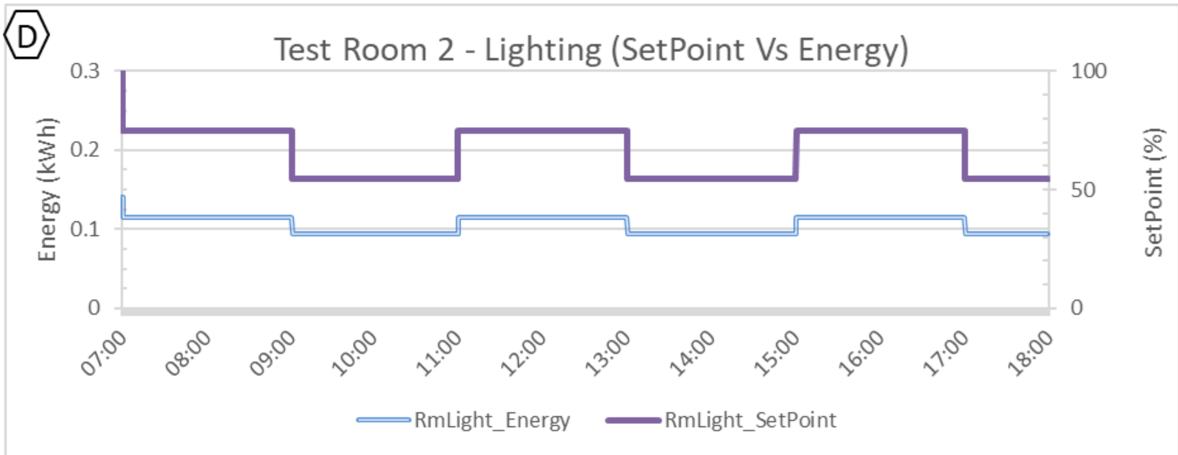


Figure 5-53: Zone PA at Level-2 Light energy consumption to the change in SP, which is in response to the budget control signal

Time	Zone Budget (b)		Expected Weightages			Price fn. SP		Ed fn. Energy (p) - Wh			Actual energy (a), Wh		
Slot	***PCA***	***PA***	***AC (30)***	***Light (1)***	**Price**	***AC***	***Light***	***AC***	***Light***	***Total***	***AC***	***Light***	***Total***
07:00	4000	2857	2765	92	0.48	25	75	1672	110	1782	2051	115	2166
09:00	2000	1429	1382	46	0.72	27	55	1241	95	1336	1210	95	1305
11:00	4000	2857	2765	92	0.48	25	75	1672	110	1782	1863	115	1978
13:00	2000	1429	1382	46	0.74	27	55	1241	95	1336	1270	95	1365
15:00	4000	2857	2765	92	0.48	25	75	1672	110	1782	1860	115	1975
17:00	2000	1429	1382	46	0.76	27	55	1241	95	1336	1210	95	1305

Table 5-14: Zone Level PCA and PA agents response to the budget control signal

Table 5-15: Zone Level PCA agents functional performce to the budget control signal (vis-e-viz Budget vs Predicted vs Actual)

Time Slot	Budget vs Predicted	Budget vs Actual	Predicted vs Actual
07:00	38%	24%	(-)22%
09:00	6%	9%	2%
11:00	38%	31%	(-)11%
13:00	6%	4%	(-)2%
15:00	38%	31%	(-)11%
17:00	6%	9%	2%

Table 5-14 and Table 5-15 provide a comprehensive look into the responses from Zone level PA to the budgetary signals. These tables highlight the discrepancies between predicted and observed energy consumption. The expected ratio between the energy consumption of the zone AC and light is 30:1. In the trails, this ratio is closer to 18:1. Furthermore, the actual consumption exceeds predicted values. This divergence suggests the need for further tuning of configuration parameters. While this research does not delve deep into the efficacy of these configurations, it emphasizes that the system operates within designed parameters. Nevertheless, it is evident that the system maintains stability and fairness.

This section underlines the responsiveness of the system to budget control signals. The presented tables and figures serve as a testament to its adaptability and the scope for improvement.

Key insights

1. **Consistent Control Signal Reception:**

- All Price Controller Agents (PCAs) and Price Agents (PAs) received optimal price/budget control signals without interruption, ensuring stable agent responses.

2. **Agent Control Response to control messages**

- Price Agents (PAs):
    - SmartHub: Adjusted its fan set point as the price varied. There was a one-time malfunction due to RAM usage, which was addressed.
    - RadiantCubicle: Altered the surface temperature set point with changing prices, influenced by the room's ambient temperature.
    - SmartStrip: Adapted in real-time, turning plugs on/off with fluctuating

prices. High prices caused battery drainage in laptops, suggesting a need for strategy adjustments.

    - Ambient Controls: Temperature and lighting controls were effectively managed. However, in scenarios like consecutive high-price cycles, there was a sudden energy demand surge when the price decreased.

- Price Controller Agents (PCAs):
    - In the context of price control messages:
        - Re-routed their received optimal price control signals to PAs and other PCAs.
    - In the context of budget signals:
        - Allocated budget according to pre-set device weights in PASS_ON mode.
        - Collaborated with local and downstream devices for optimal price computation in Optimization Mode, ensuring the allocated budget was maintained.
        - A breakdown of how the budget was distributed from the building to task level and agents' subsequent energy usage response was illustrated.

3. **Analysis of Control Response:**

- Tables V 14 and V 15 presented the Zone level PA's response to budget signals. These tables highlighted discrepancies between anticipated and actual energy consumption suggest a need for refining configuration parameters.
- As desired, aggregate energy consumption decreased with price hikes at the building and zone levels but increased at the task level.

4. System's Adaptability and Room for Improvement:

- The system demonstrates adaptability to both price and budget control signals.
- However, further optimization and improvement areas were identified, especially in managing high-price scenarios and refining configuration parameters for better alignment with expected consumption patterns.

The system's functional properties have been tested against price and budget signals, revealing its adaptability, effectiveness, and areas needing refinement.

5.7.2 Non-Functional Properties Evaluation (MAS Performance)

In multi-agent systems (MAS), understanding and evaluating non-functional properties is essential. Such properties often dictate the system's efficacy, reliability, and adaptability in varied scenarios. This section details these non-functional attributes of the MAS, assessing its performance beyond mere functionality. We aim to uncover the underlying behaviours and patterns influencing the system's robustness, scalability, and responsiveness. By evaluating these non-functional properties, we can ascertain the system's readiness for real-world deployments and highlight potential areas of improvement.

A. Agents' registration time

Agents' registration time is a agents performance metric that measure the amount of time required for an agent to register accordingly after it starts. This metric impacts minimum time that is required for a PECS to join and participate.

For an energy-consuming device to participate in the iSPACE, it must register with the local apparatus hosting the Price Controller Agent (PCA). For example, the RadiantCubicle's Price Agent (PA) at task level registers with the SmartHub's PCA.

Similarly, the device that hosts the PCA must register with an upstream device responsible for hosting the PCA designated for that level. For example, the SmartHub PCA registers with Zone controller PCA.

An exhaustive trawl was executed across all the experimental log data to derive the value for the agent's registration performance metric. This was specifically for the log messages that denoted agents registering with their appropriate PCAs. It means local PAs affirming their registration with their respective local PCA and, in turn, these PCAs confirming their registration with a superior or upstream PCA. The corresponding response time, in milliseconds (ms), has been catalogued in Table 5-16. A visual representation of the same data can also be observed in Figure 5-54. The high response time, designated in purple on the table and the chart, are anticipated occurrences. These arise when the upstream PCA or local PCA has not been initialized and, as a result, is not equipped to process device registrations. However, once these agents initialize and are primed to accept registrations, the registration process progresses and completes.

Table 5-16: Agents' registration response time (ms)

Comm. Medium	**Instance no. (→) Agents (↓)**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Wi-Fi	72 SS PCA - 62 SH PCA	7187	1328	678	24639	2669	87500	2404	1973	630	519	840	581
	73 SS PCA - 63 SH PCA	11783	551	2099	24045	2111	86956	2274	3062	773	620		
	74 SS PCA - 64 SH PCA	1521	618	7689	1323	36345	2622	96521	2828	2210	561	883	
	75 SS PCA - 65 SH PCA	1640	1654	86980	2299	1197	745	1741					
	62 SH PCA - 51 ZN PCA	481	402	584	681	1251751	56	715	1644				
	63 SH PCA - 51 ZN PCA	465	706	289	1257492	107	469	376					
	64 SH PCA - 52 ZN PCA	660	510	566	576	372	598						
	65 SH PCA - 52 ZN PCA	510	344	484	621	239	643						
Lan	51 ZN PCA - 11 BD PCA	119	98	140	174	103	102						
	52 ZN PCA - 11 BD PCA	115	105	94	141	89	118						
Local (internal)	72 SS PA - 72 SS PCA	1	1	1	2	2	7	3	1	1	1		
	73 SS PA - 73 SS PCA	3	1	2	2	1	2	2	2				
	74 SS PA - 74 SS PCA	1	1	1	2	1	3	3	9				
	75 SS PA - 75 SS PCA	1	3	2	1	1	3	12					

Comm. Medium	Instance no. (→) Agents (↓)	1	2	3	4	5	6	7	8	9	10	11	12
	62 SH PA - 62 SH PCA	3	4	2	3	2	19	3	4	3	4	2	1
	62 RC PA - 62 SH PCA	3	11	1	2	2	1	4	5	2	7	2	
	63 SH PA - 63 SH PCA	2	2	3	4	2	2	9	1	2			
	63 RC PA - 63 SH PCA	3	2	1	1	1	1	2	2	1			
	64 SH PA - 64 SH PCA	1	1	3	1	2	1	2					
	64 RC PA - 64 SH PCA	14	2	1	1	1	1	2					
	65 SH PA - 65 SH PCA	8	1	1	1	2	3	2	1	1	1	2	
	65 RC PA - 65 SH PCA	7	1	11	1	1	2	4	1	2	1	1	
	51 ZN PA - 51 ZN PCA	1	8	1	1	1	1						
	52 ZN PA - 52 ZN PCA	2	1	1	1	1	1						

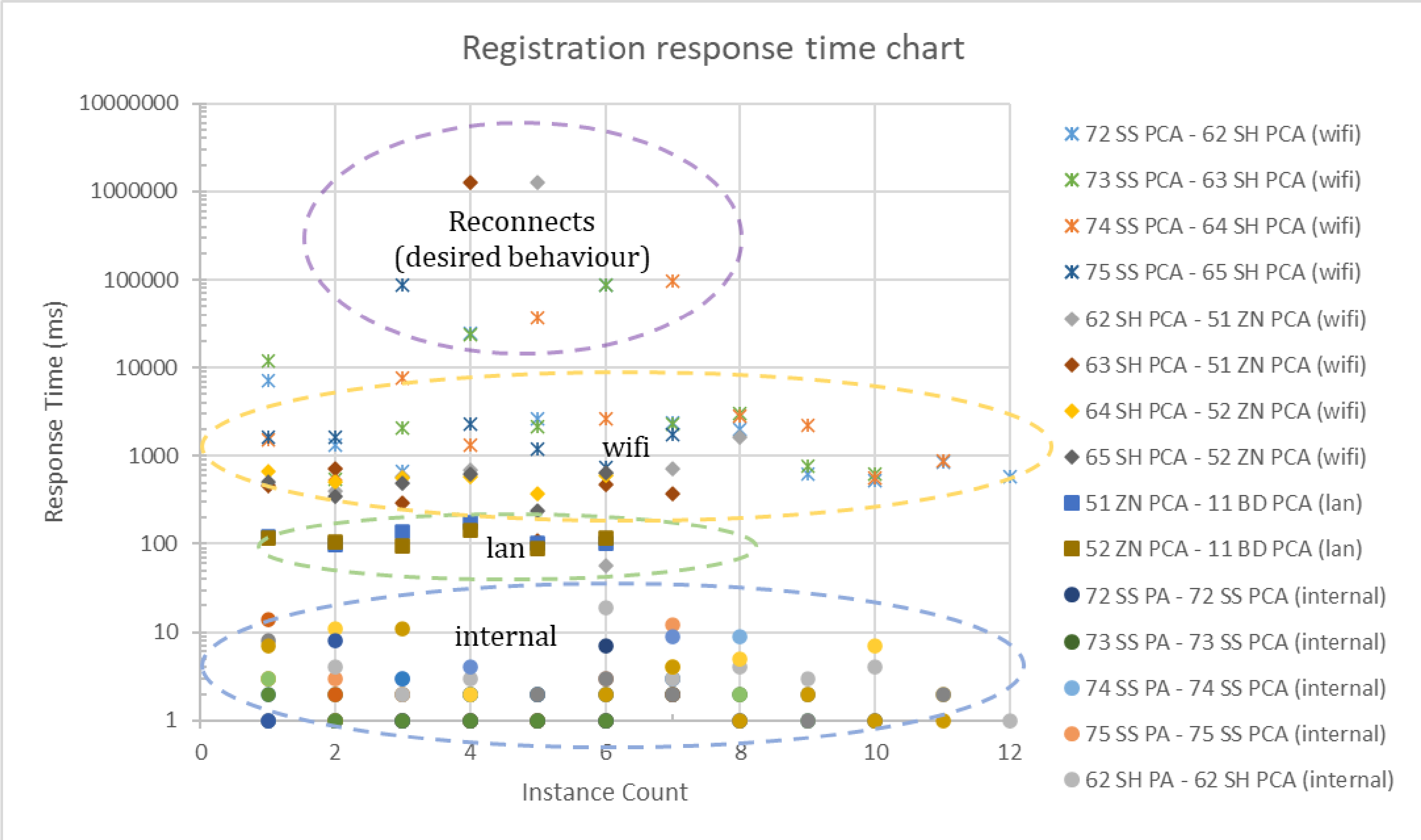


Figure 5-54: Agents' registration response time chart

Key insights:

Table 5-17 comprehensively summarises the agents' registration time metrics. A crucial observation from the data indicates that the registration time is notably shorter when the source (PA) and destination (PCA) agents operate on the same level or communicate via a wired connection. In contrast, registration times tend to elevate when reliant on a Wi-Fi connection.

Table 5-17: Agents' registrations time(ms) metrics

	No. of records	Min	Max	Mean	Std dev
Wi-Fi (PCA→PCA)	67	56	7689	1174	1175
Lan (PCA→PCA)	12	89	174	117	23
Local (PA→PCA)	122	1	19	3	3
All	**201**	**1**	**7689**	**355**	**829**

1. **Wi-Fi (PCA→PCA) Registration:** Out of 67 records, the registration time ranged from a swift 56 ms to a maximum of 7689 ms. The average time for Wi-Fi-based registration is 1174 ms with a standard deviation of 1175 ms. This suggests variability in Wi-Fi connection quality or potential network congestion at times.
2. **LAN (PCA→PCA) Registration:** Among the 12 records, the shortest registration time recorded was 89 ms, and the lengthiest was 174 ms. The average registration time was 117 ms, and the process showed consistency with a low standard deviation of 23 ms.
3. **Local (PA→PCA) Registration:** Out of the 122 records observed, the registration times were notably faster, ranging from 1 ms to 19 ms. The average registration time was just 3 ms, indicating a rapid local registration process. The standard deviation was also 3 ms, highlighting consistent performance.

4. **Overall Performance:** Analysing all the 201 data points, registration times varied significantly from 1 ms to 7689 ms. The overall average registration time was 355 ms, with a standard deviation of 829 ms, pointing to the influence of the Wi-Fi-based registrations on the average and variability.

The analysis shows that while local registrations (PA→PCA) are swift and consistent, Wi-Fi-based registrations (PCA→PCA) can experience delays, due to the inherent nature of wireless communication and potential network issues.

B. <u>Agents' response time (for given optimal price message)</u>

Agents' response time is a pivotal performance metric, assessing agents' duration to respond to a designated price. This metric encompasses the calculation phase, where the set point is computed from price functions, followed by publishing these computed setpoints onto the message bus, which the device controller agent then leverages. The architecture houses two specific agent types at each level: the Price Controller Agent (PCA) and the Price Agent (PA), spread across four hierarchical levels.

The corresponding response time, in milliseconds (ms), has been as shown in Table 5-18. A visual representation of the same data can also be observed in Figure 5-55 and Figure 5-56. Figure 5-57 and Figure 5-58 illustrate histograms coupled with the Bell Curve for SmartHub62 PCA and PA response times which offer a more granular statistical understanding.

Table 5-18: Agents' response time metrics (ms) for an optimal price received

Level	Agent	No. of records	Min	Max	Mean	Std dev
Level 1	PCA	27	32	134	67	25
Level 2	PCA	56	33	110	71	16
	PA	42	391	884	686	104
Level 3	PCA	111	102	218	123	22
	PA	197	1881	8618	4515	1232
Level 4	PCA	110	105	194	128	23
	PA	109	1149	6454	2035	793
Total	**PCA**	**304**	**32**	**218**	**110**	**33**
	PA	**348**	**391**	**8618**	**3276**	**1795**

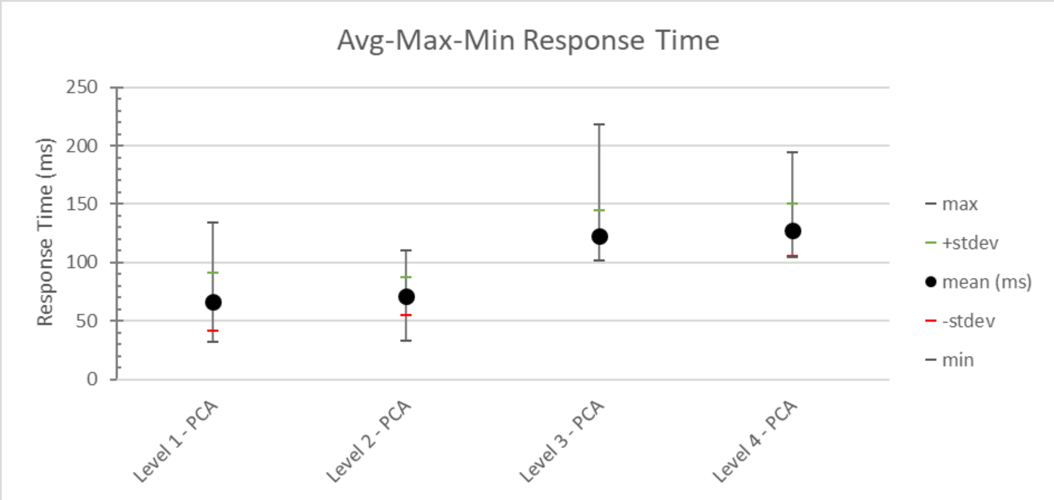


Figure 5-55: PCA Agents Computing Performance metrics

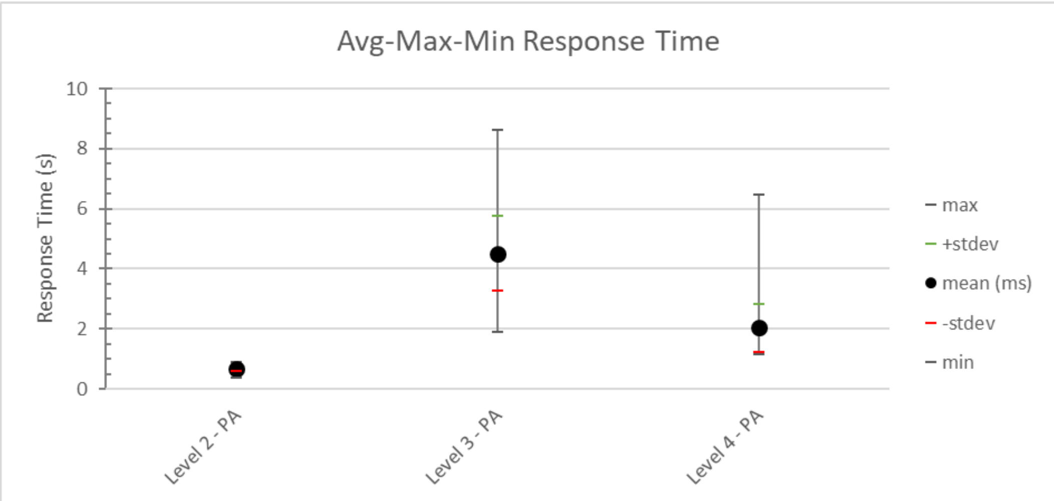


Figure 5-56: PA Agents Computing Performance metrics

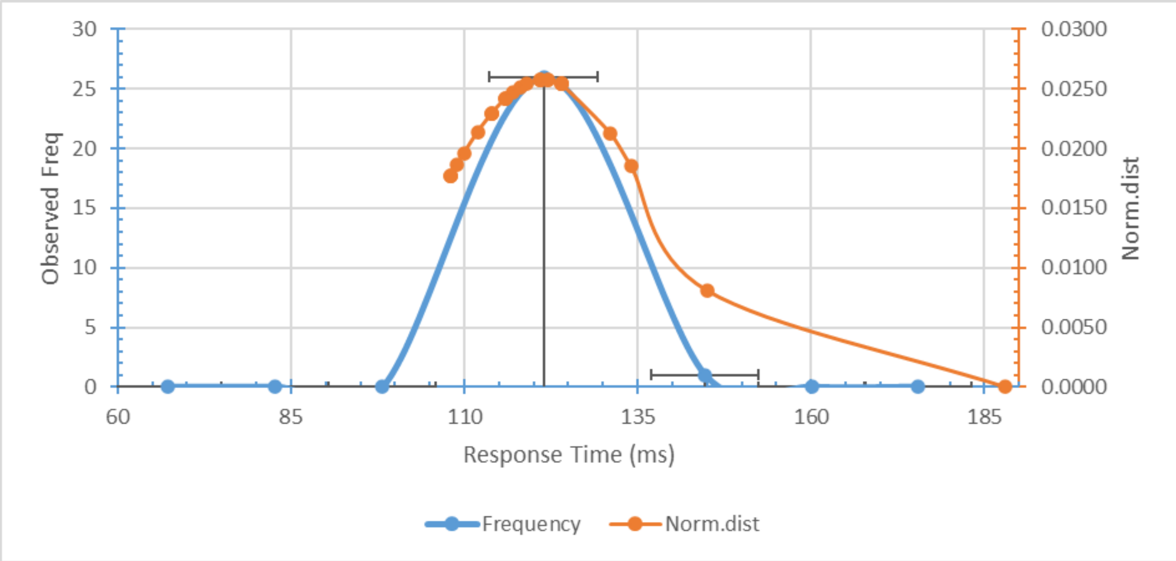


Figure 5-57: Histogram with bell curve for SmartHub62 PCA response time

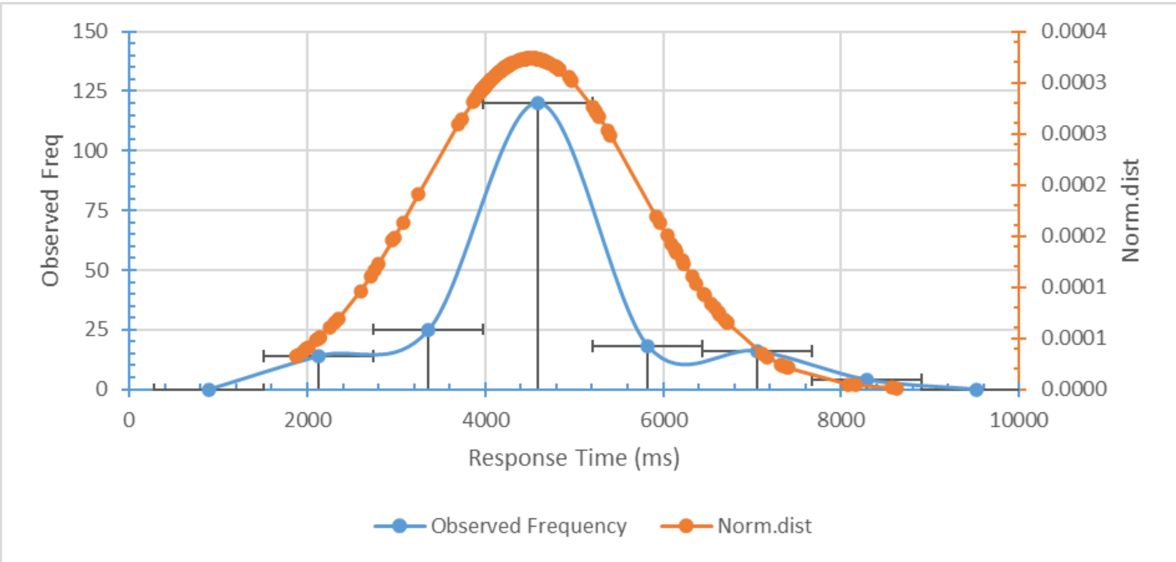


Figure 5-58: Histogram with bell curve for SmartHub62 PA response time

Key insights:

1. **Level Differences:** Agents in Levels 3 and 4, both PCA and PA, display noticeably prolonged response times compared to their Level 1 and Level 2 counterparts. This divergence is attributable to employing lower processing compute modules, like Intel-Edison, in the former, whereas the latter leverages more robust processing units, such as RPis.
2. **Standard Deviation Insights:** For PA, Among the 197 instances recorded, the first standard deviation envelops 74% of the instances, translating to 145. Meanwhile, the second standard deviation encompasses 92%, equivalent to 182 instances. For PCA, among the 28 instances recorded, the first standard deviation envelops 93% of the instances, translating to 26.
3. **Entropy Comparison:** The calculated entropies for a normal distribution of PCA and PA response times are 2.70 and 3.71, respectively. Nevertheless, the captured entropies for PCA and PA agents' response times are 0.27 and 1.81, respectively. These values are reduced compared to a typical normal distribution. Such a reduced entropy indicates that the response times of these agents, especially when viewed in conjunction, are more predictable.

To sum it up, while there are inherent differences in response times across agent types and levels, a considerable portion of this variability stems from the underlying hardware and compute capabilities. The overarching aim should be to optimize these response times, ensuring the system operates seamlessly and efficiently.

C. Agents' intra-level (within) response time (for given optimal price)

The intra-level response time of agents is defined as the duration from when a Price Controller Agent (PCA) broadcasts a price message on the message bus until it is received and acted upon by the corresponding Price Agent (PA) within the same hierarchical level.

The corresponding response time, in milliseconds (ms), has been as shown in Table 5-19. Figure 5-59 and Figure 5-60 visually interpret this data. Specifically, Figure 5-60 employs histograms complemented with the Bell Curve for the intra-level response times of Level-3 agents, serving a detailed statistical insight.

Table 5-19: Agents' intra-level (within) response time metrics (ms) for an optimal price

Level	Device Ids [src pca – dst pa]	Agents	No. of records	Min	Max	Mean	Std dev
Level 1	[11, 11]	pca_et-pca_st	27	32	134	67	25
Level 2	[51, 51] [52, 52]	pa_et-pca_st	47	84	964	699	231
Level 3	[62, 62] [63, 63] [64, 64] [65, 65]	pa_et-pca_st	200	199	8819	4652	1343
Level 4	[72, 72] [73, 73] [74, 74] [75, 75]	pa_et-pca_st	109	1308	6720	2213	810
Total			**383**	**32**	**8819**	**3150**	**1994**

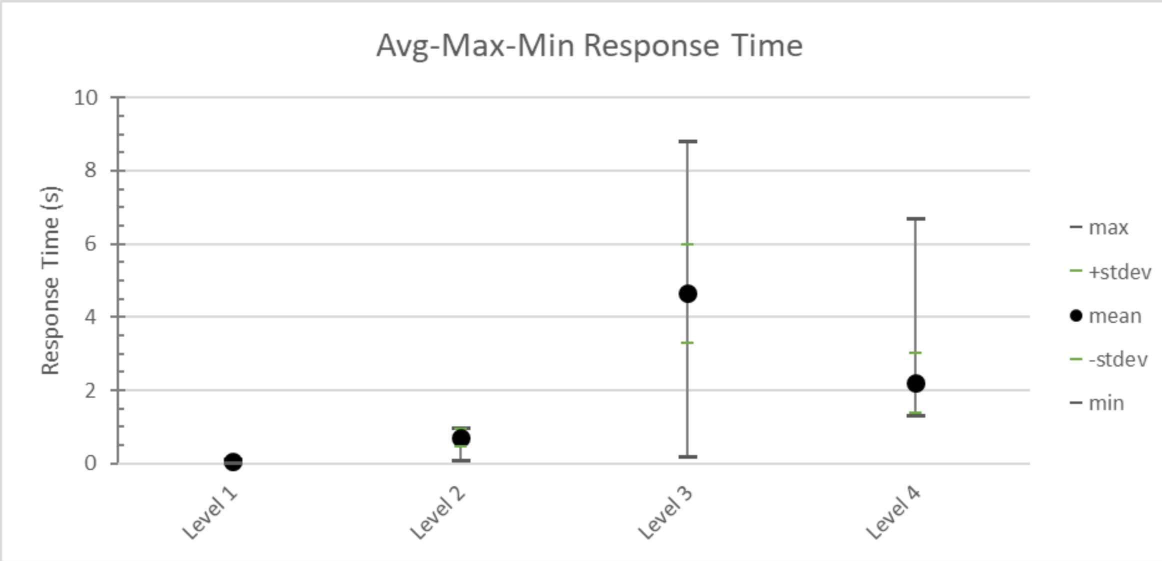


Figure 5-59: Agents' intra-level (within) response time metric (ms) for an optimal price chart

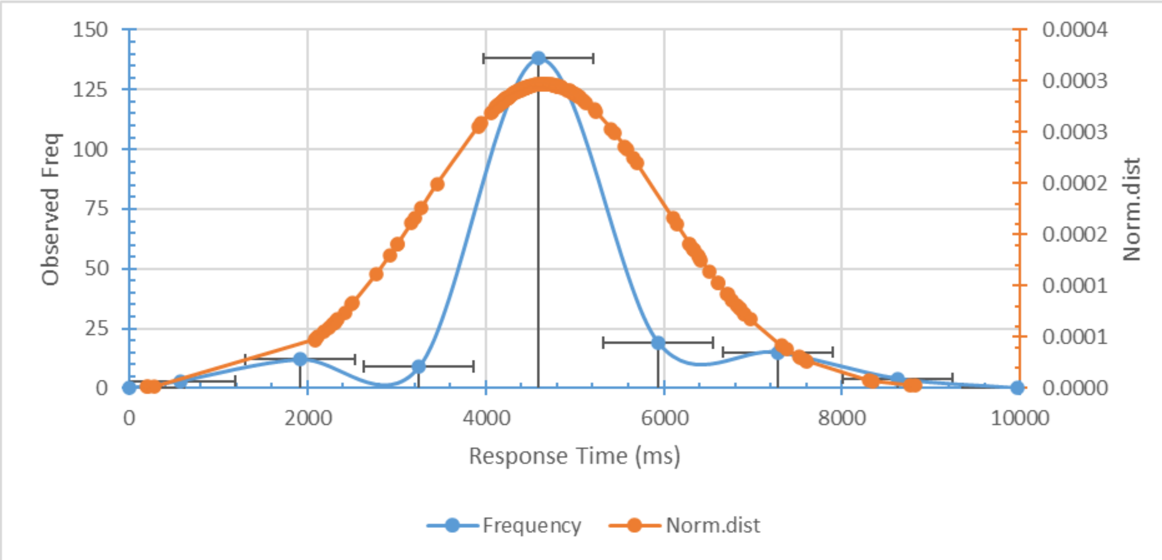


Figure 5-60: Histogram with bell curve for Level-3 response time

Key insights:

1. **Level 3 Insights:** Of the 200 instances documented for Level 3:

- The first standard deviation encompasses 70% of the cases, translating to 139 instances.
- The second standard deviation covers 91%, equivalent to 181 instances.

2. **Entropy Analysis:** Given the above mean and variance, a typical normal

distribution yields an entropy of 3.74. However, the entropy for the intra-level response time at Level 3 is only 1.53. This lower entropy signifies a predictability that is more pronounced than a typical normal distribution.

In conclusion, this data helps map out the agents' internal functioning within a specific level, showing that while there may be inherent latencies based on various levels and device types, the system, especially at Level 3, operates with a fair amount of predictability.

D. Agents' inter-level (between) response time (for given optimal price)

The inter-level response time for agents denotes the interval that elapses from when a source Price Controller Agent (PCA) broadcasts a price message on the message bus until it is retrieved and processed by a PCA in a subsequent downstream level.

Table 5-20: Agents' inter-level (between) response time (for given optimal price)

Level	Device Ids [src pca, dst pca]	No. of records	Min	Max	Mean	Std dev
Level 1-2	[11, 51] [11, 52]	56	226	2238	1282	591
Level 2-3	[51, 62] [51, 63] [52, 64] [52, 65]	111	406	2545	1367	519
Level 3-4	[62, 72] [63, 73] [64, 74] [65, 75]	109	491	3013	1667	665
Level 1-4	**[11, 72] [11, 73] [11, 74] [11, 75]**	**110**	**1912**	**6439**	**4139**	**1026**

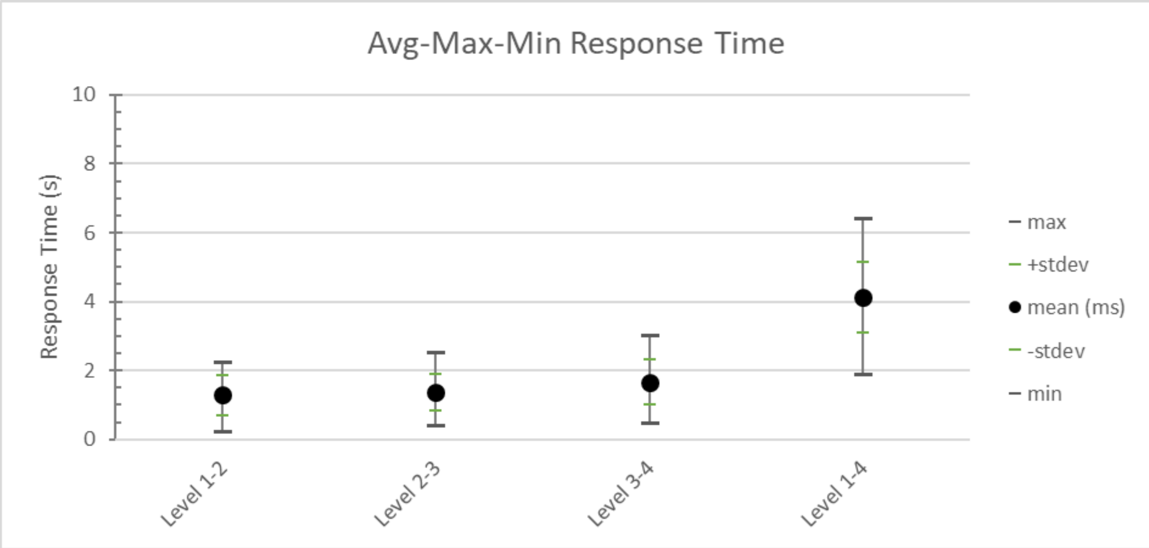


Figure 5-61: Agents' inter-level (between) response time metric (ms) for an optimal price chart

Key insights:

1. **Increasing Response Time with Level Gap:** As the distance between levels increases, so does the inter-level response time. For instance, the transition from Level 1 directly to Level 4 has a notably higher mean response time (4139 ms) than transitions between immediately consecutive levels, such as Level 1 to Level 2 (1282 ms). This metric suggests that intercommunication between non-adjacent levels may require more intermediate processing or routing, increasing the latency.
2. **Variability in Response Times:** The standard deviation provides an idea about the spread or variability in the data. A higher standard deviation, as seen in the transition from Level 1 to Level 4 (1026 ms), indicates more variability in the response times, due to inconsistencies in network traffic, system load, or other unforeseen factors.
3. **Potential Hardware or Infrastructure Bottlenecks:** The response times also indicate the computation capabilities of the devices involved or the efficiency of the communication channels between them. For instance, the higher response times seen at the Level 3-4 transition may reflect the underlying hardware or network capabilities.

4. **Criticality of Efficient Communication:** The data underscores the importance of efficient communication protocols and infrastructure, especially if timely decision-making is crucial. Delays in response times can potentially delay crucial decisions or actions, which can be critical in scenarios requiring real-time or near-real-time responses.

In conclusion, while the inter-level response times offer a clear understanding of communication latencies between different agent levels, they also highlight areas requiring further investigation and optimization. As systems scale and become more complex, ensuring efficient inter-level communication will be pivotal to their overall performance and reliability.

E. Convergence rate analysis

To determine the convergence rate, a Python program was developed to replicate the Price Controller Agents (PCA) behaviour using a gradient descent algorithm. Specifically, the program simulates the mechanism when the DEFAULT_OPT option is enabled. Over one million test runs were executed for each combination of DEFAULT_OPT activated at various levels, including building, zone, and task levels. The resulting convergence rates and descriptive statistics metrics for each scenario were presented.

Our observations indicated that achieving 100% convergence is feasible, given the appropriate selection of step sizes and deadbands.

For these simulations, energy demand functions were derived from data accrued during the prior experiment. The simulations operated under the presumption that all devices shared equal weights. Additionally, message transmission time between agents were disregarded.

In each test iteration, agents began in a randomized state. Each agent was assigned a random prior price and energy level. The energy parameter was restricted to a range determined by the minimum and maximum limits of the related energy-consuming device. The price and energy demand functions, detailed in section 5.5, were employed. These details are further briefed in Table 5-21.

Table 5-21: Detail of Energy-Consuming devices

Energy consuming device	**Energy Demand (Wh)**		**Price Function**	**Energy Demand Function**
	Min	***Max***		
Ambient AC	800	1950	Figure 5-19	Figure 5-20
Ambient Light	80	140	Figure 5-21	Figure 5-22
SmartHub	16	21	-	-
RadiantCubicle	16	42	Figure 5-25	Figure 5-24
SmartStrip	2	68	-	-

A random price was allocated to the PCA to initiate the bidding process. Its objective was to calculate an optimal price aligned with a target energy demand proportional to the initially assigned random price and energy demand, the principle that energy costs must remain constant. A simulation terminates when it nears the target energy within a set deadbands or after completing 30 iterations.

The various simulated scenarios are presented in Table 5-22. This analysis offers crucial insights into the performance and adaptability of the PCA gradient descent algorithm in various configurations.

Table 5-22: Outline of simulation scenarios

Scenario	**DEFAULT_OPT** *Level-1 (Building)*	*Level-2 (Zone)*	*Level-3 (SmartHub)*	**Test Run Level**
Scenario-1	-	-	True	Level-3
Scenario-2	-	True	False	Level-2
Scenario-3	-	True	True	Leve-2
Scenario-4	True	False	False	Level-1
Scenario-5	True	True	True	Level-1

Figure 5-62, Figure 5-63, Figure 5-64, Figure 5-65, and Figure 5-66 presented provide a visual representation, mapping the number of test runs against the number of iterations required for the specified dead band and gamma (step size).

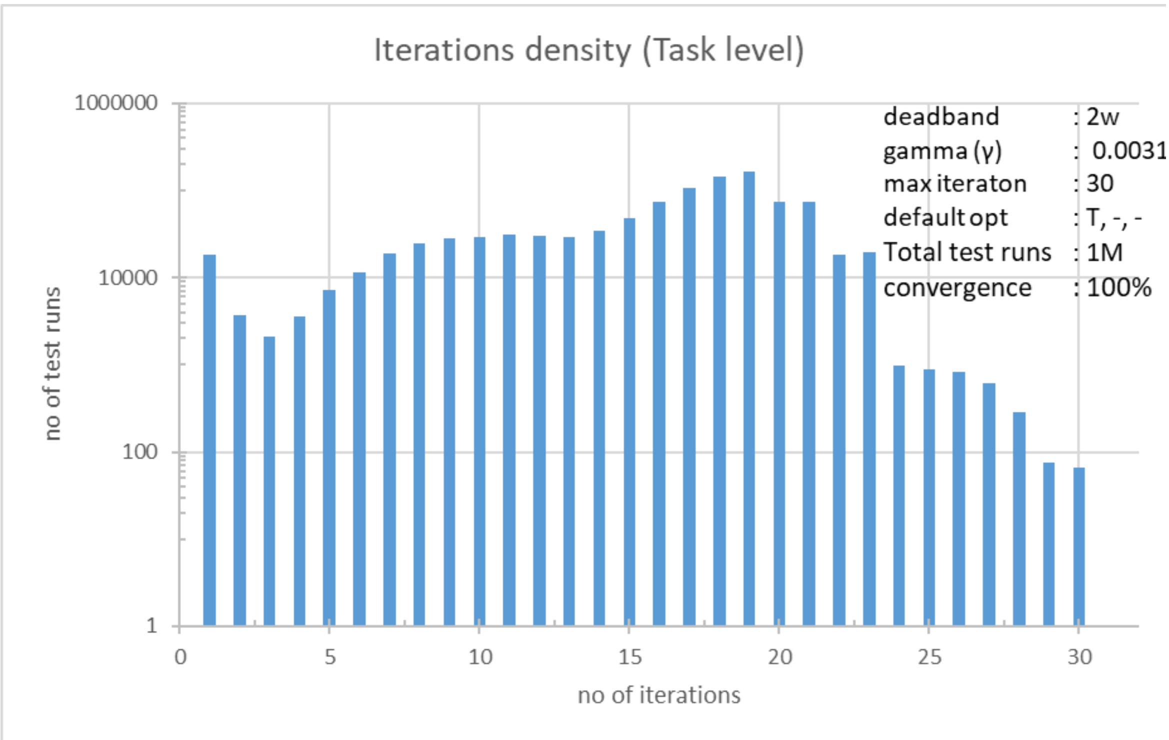


Figure 5-62: Scenario-1 (Task level, SmartHub) convergence rate

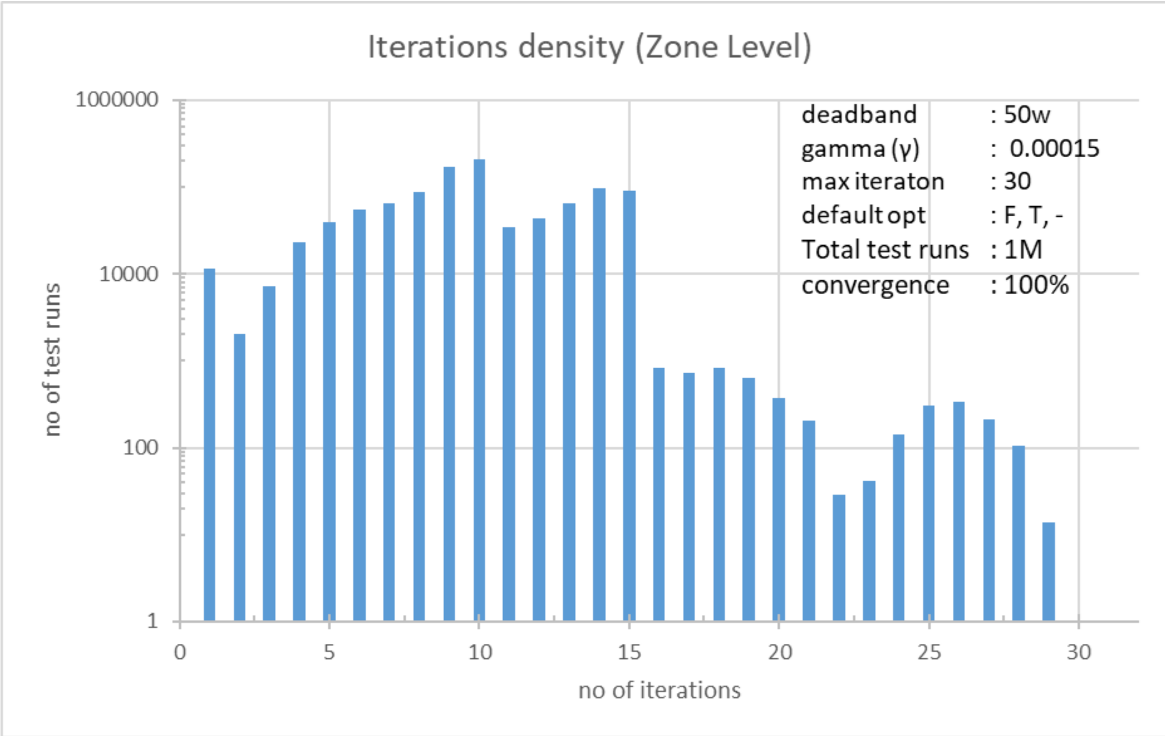


Figure 5-63: Scenario-2 (Zone level) convergence rate

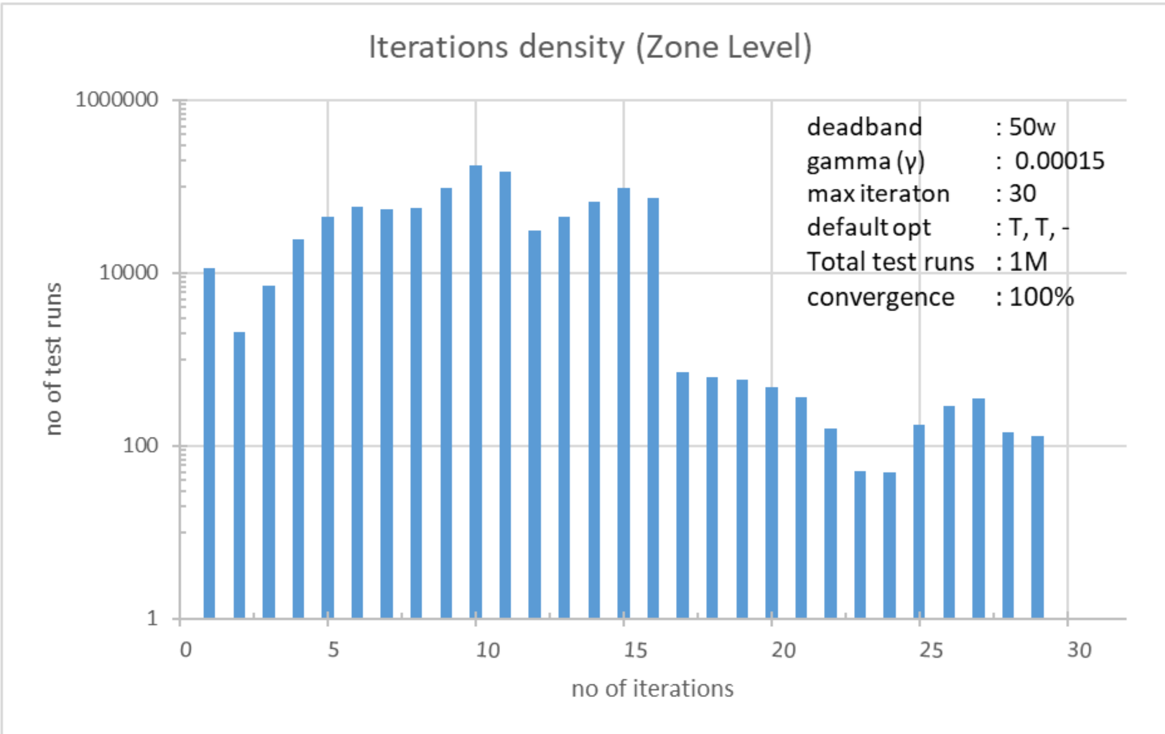


Figure 5-64: Scenario-3 (Zone level) convergence rate

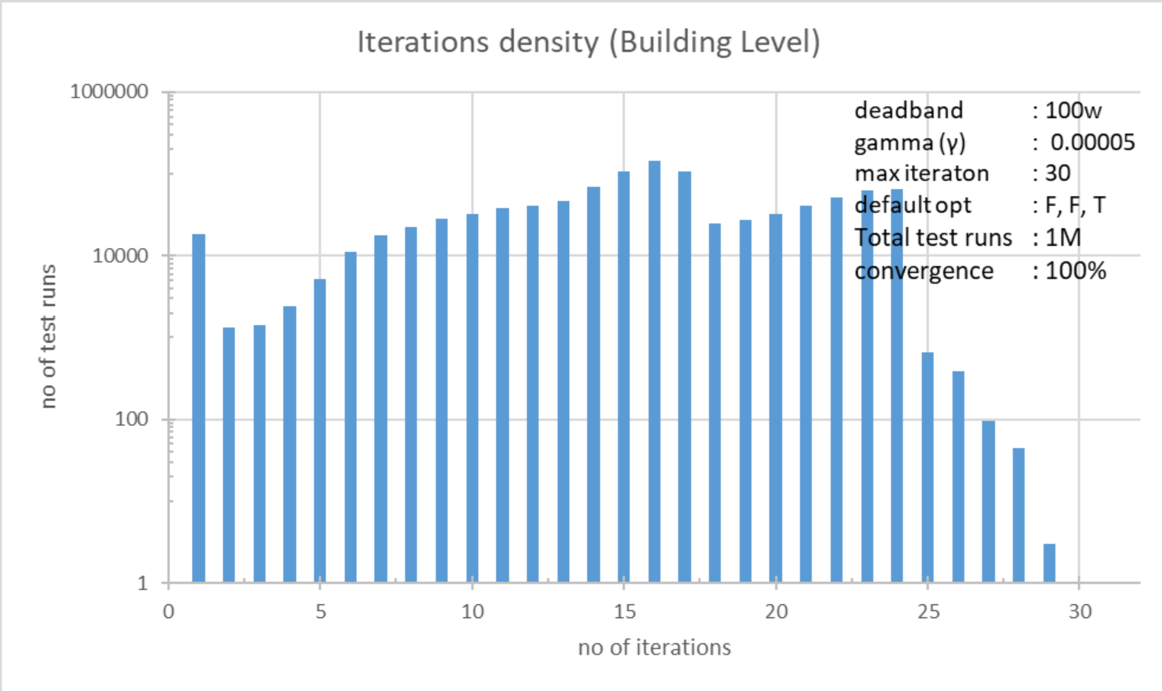


Figure 5-65: Scenario-4 (Building level) convergence rate

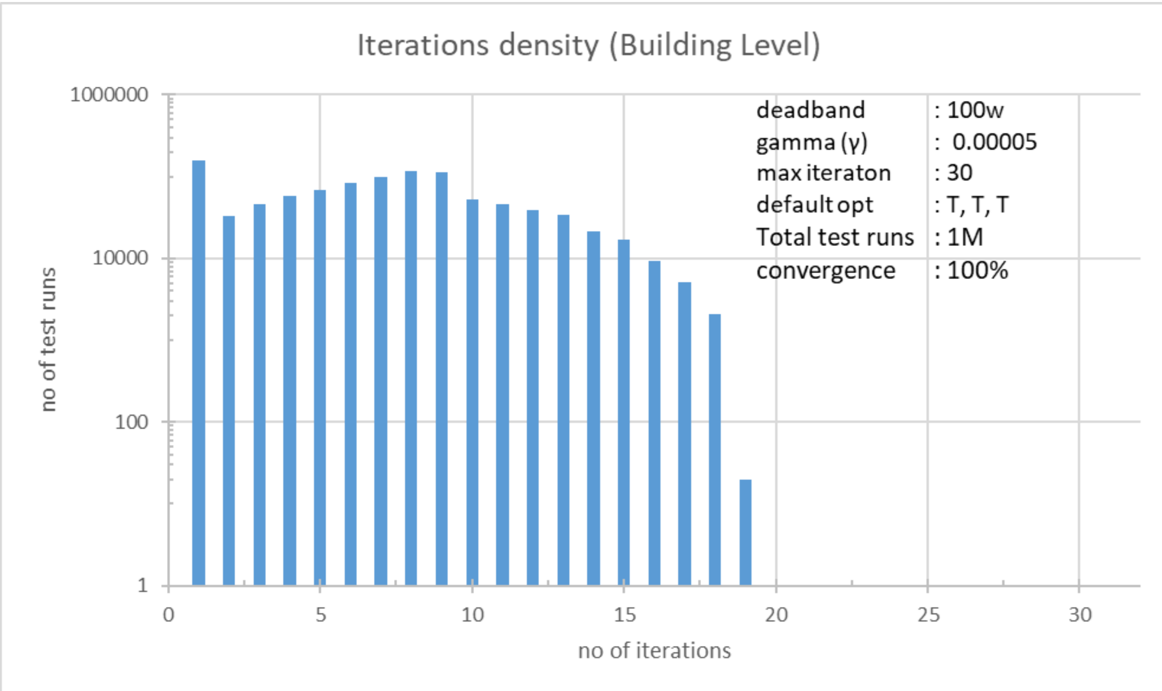


Figure 5-66: Scenario-5 (Building level) convergence rate

Table 5-23 offers a consolidated overview detailing the convergence behaviour associated with each scenario.

Table 5-23: Summary of Statistics for All Scenarios

Scenario	Mean	Median	Std Dev	99th Percentile
Scenario-1	16	17-19	16	~20
Scenario-2	10	~10	10	~15
Scenario-3	10	~10	11	~16
Scenario-4	16	~16	17	~19
Scenario-5	7	~8	8	~17

Note:

1. The Weighted Average (Mean) Iterations is computed as $\frac{\sum(iterations\ X\ count\ of\ test\ runs)}{total\ test\ runs}$
2. The Median is estimated by observing which iteration number the count of test runs begins to decrease in number significantly.
3. The 99th Percentile is an approximation. It's the iteration by which approximately 99% of the test runs have converged.

Key insights:

1. **Scenario-1 (Task level):**

- 100% convergence is achieved using a gamma of 0.0031 and a dead band of 2W.
- Most test runs exhibit convergence between the 16th and 19th iterations. The third quartile (Q3), representing the 75th percentile (the iteration number by which 75% of the test runs have converged), is at the 18th iteration. The 99th percentile of test runs reach convergence by the 20th iteration.

2. **Scenario-2 & Scenario-3 (Zone level):**

- Both scenarios achieve 100% convergence with a gamma (step-size) of 0.00015 and a 50W dead band. This behaviour is consistent whether the DEFAULT_OPT is enabled at the task level (Scenario-3) or not (Scenario-2).

- Convergence for most test runs occurs around the 10th iteration. The third quartile (Q3) is between the 12th and 13th iterations, and the 99th percentile of test runs converge by the 15th iteration.
- Although the convergence rates of the two scenarios are statistically similar, Scenario-3 shows a modestly more weighted average number of iterations compared to Scenario-2, indicating a minor delay in convergence for Scenario-3.
- Nevertheless, given the minimal difference, Scenario-3 is more desirable. It provides the ability to fine-tune at the task level and better manage how the allocated constraint resource is re-distributed, considering end-user preferences.

3. **Scenario-4 & Scenario-5 (Building level):**

- At the building level, 100% convergence is observed using a gamma (step-size) of 0.00005 and a dead band of 100W. It is consistent whether the DEFAULT_OPT is enabled at the zone and task level (Scenario-5) or not (Scenario-4).
- A notable distinction is present in the convergence rate between Scenario-4 and Scenario-5:

  i. Scenario-4: Most test runs converge around the 16th iteration. The third quartile (Q3) is at the 19th iteration, and the 99th percentile of test runs converge by the 24th iteration.
  ii. Scenario-5: Convergence typically occurs around the 7th iteration for most test runs. The third quartile (Q3) is also at the 7th iteration, while the 99th percentile of test runs converge by the 16th iteration.

- Scenario-4 utilizes a centralized optimization method, whereas Scenario-5 employs a hierarchical distributed optimization technique. Scenario-5's hierarchical approach converges more swiftly than the centralized method of Scenario-4, making it the preferred choice. Moreover, it offers a refined mechanism to efficiently allocate and manage constrained resources, keeping in line with the priorities and preferences of energy-consuming devices.

These observations offer valuable insights into the convergence characteristics and efficiency within the system across various levels, such as task, zone, and building. They also underscore the merits of the hierarchical distributed optimization approach.

5.7.3 Energy Demand & Comfort Evaluation

Measuring and evaluating system performance in terms of energy demand is paramount for understanding energy demand flexibility against the baseline and the impact of the price vs energy demand on convergence. The section focuses on this crucial aspect, providing insights into the system's energy demand patterns. While energy demand is at the forefront of our evaluation, it is also essential to understand comfort, ensuring that demand response mechanisms do not compromise basic user needs. The evaluation primarily assesses the system's energy demand flexibility while emphasising the incidental comfort aspect.

A. Energy Demand

In this section, the total energy demand with iSPACE was calculated and compared against the energy demand for the FDD Lab Benchmark scenarios. The total energy demand was computed by combining the equivalent electrical energy. The Trapezoidal Rule was employed to deduce the energy demand over duration for the experiment at Rooms 1 & 2 and the building level. The energy demand metrics are summarised in Table 5-24.

Table 5-24: Experiments energy demand metrics for day 1-4

		Day-1	Day-2	Day-3	Day-4	Total
Level-2 (Room 1)						
Duration (hrs)		10	10	10	10	40
Energy Demand (kWh)	Total	17.48	17.70	18.10	17.69	70.97
	Min	0.75	0.74	0.84	0.69	0.69
	Max	2.30	2.37	2.23	2.16	2.37
	Mean	1.75	1.77	1.81	1.77	1.77
Level-2 (Room 2)						
Duration (hrs)		10	10	10	10	40
Energy Demand (kWh)	Total	16.57	16.70	16.84	17.47	67.58
	Min	0.75	0.87	0.75	0.70	0.70
	Max	2.25	2.15	2.13	2.14	2.24
	Mean	1.66	1.67	1.68	1.75	1.69
Building level						
Duration (hrs)		10	10	10	10	40
Energy Demand (kWh)	Total	34.05	34.40	34.95	35.15	138.55
	Min	1.50	1.62	1.52	1.39	1.39
	Max	4.48	4.52	4.36	4.30	4.52
	Mean	**3.40**	**3.44**	**3.49**	**3.52**	**3.46**

It is observed that energy demand varies from 1% to 7% across the four days between Room 1 and Room2, respectively. However, the energy demand for Day-1 and Day-2 (optimal price mode) is 5% and 6%, and for Day-3 and Day-4, it is 7% and 1% between Room 1 and Room 2, respectively. When the optimal price is received, the price agents compute the desired set point based on the price functions. However, when the budget is received, the price agents use gradient descent to compute the optimal price based on the energy demand functions and subsequently compute the set point for this optimal price.

Nevertheless, the energy demand flexibility, when observed against the benchmark (Table 5-1), a reduced energy demand of 26%, can be observed for the pilot deployment of the iSPACE system, which aligns with other studies in the literature [103]. Figure 5-67 show the comparative energy demand metrics during the FDD Lab Benchmark and the pilot deployment of the iSPACE system. The left side of the chart (marked in light grey) represents the energy demand for the FDD Lab Benchmark scenarios. The right side (marked in light green) represents the energy demand for the pilot deployment of the iSPACE system during the experiment.

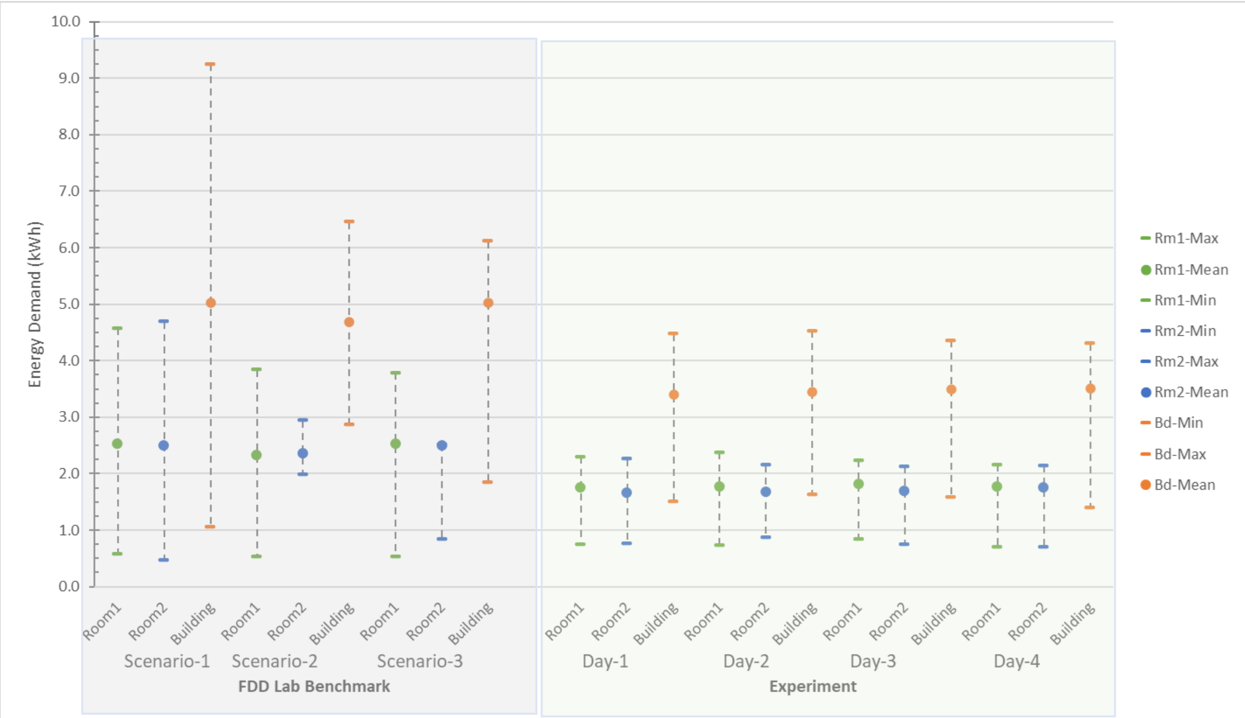


Figure 5-67: Energy Demand comparision - FDD Lab Benchmark Vs the pilot deployment of the iSPACE system

B. <u>Price vs Energy Demand</u>

The energy consumption of each device within an hour slot was calculated, and the corresponding setpoint vs. energy demand plots were generated for each energy-consuming device. The energy demand functions for each device were derived through curve fitting **(for detailed explanation on how to generated energy demand**

functions was explained in section on price and energy demand functions, section 3.4.3). Iterating over each energy-consuming device price function (price vs setpoint), the corresponding energy demand for the price was derived from the energy demand function (setpoint vs energy) and plots depicting the energy-consuming device's price vs. energy demand were created. The observed energy consumption for various prices at Level-3 (Task Level), Level-2 (Zone Level) and Level-3 (Building Level) during the experiment are plotted in Figure 5-68, Figure 5-69, and Figure 5-70, respectively.

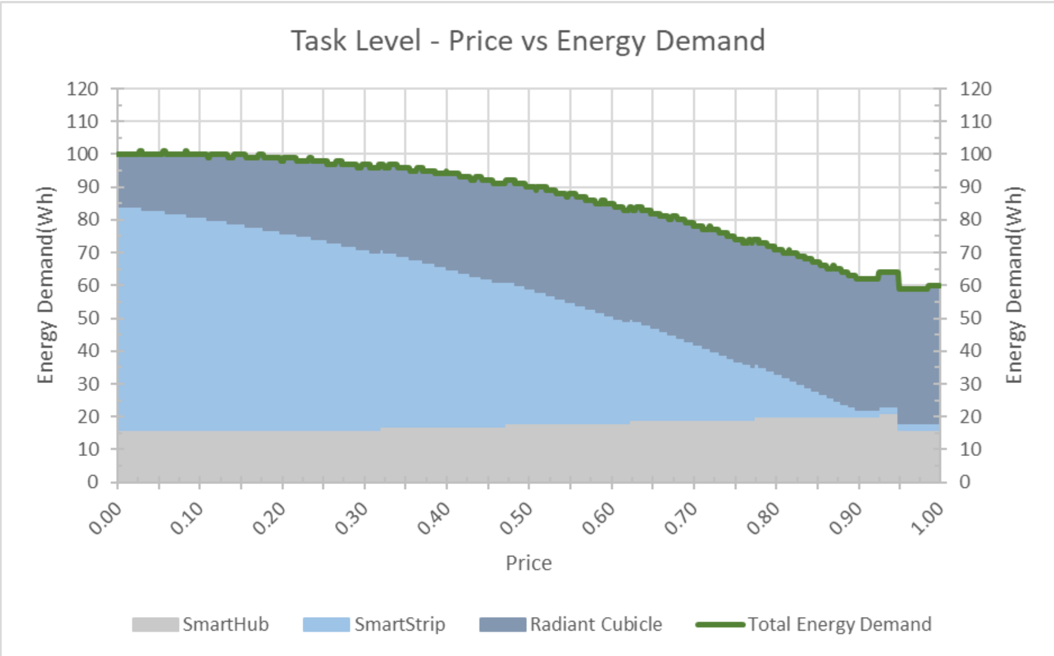


Figure 5-68: Level-3 (task level) price vs energy demand

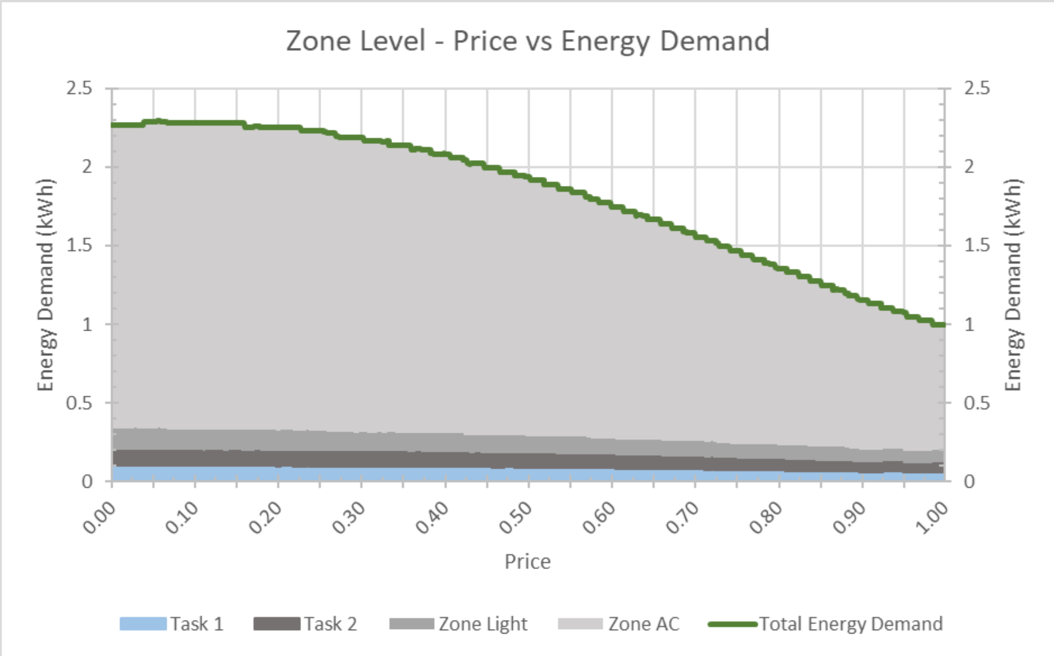


Figure 5-69: Level-2 (zone level) price vs energy demand

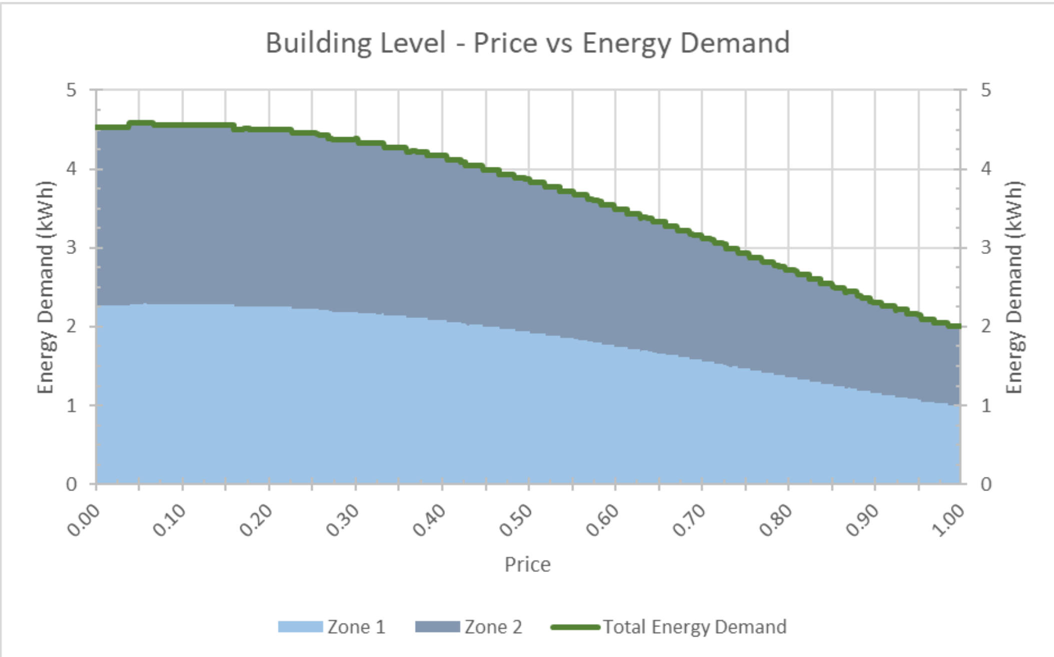


Figure 5-70: Level-1 (building level) price vs energy demand

In Figure 5-68, for the Task Level, it can be observed that the SmartHub (fan, lamp, and other sensors attached to the hub) energy consumption (in light grey) and RadiantCubicle energy consumption (in dark grey) are increasing as price increases. Further, the SmartStrip energy consumption (in blue) decreases with increasing price.

Overall energy consumption at task level, represented by the green line, decreased with raising prices. Similarly, overall energy consumption at zone and building level, represented by the green line (Figure 5-69, and Figure 5-70, respectively), decrease with raising prices.

At task level, a slight perturbation (increase to 64 Wh from 62 Wh and then decrease to 59 Wh) was observed during a price change from 0.95 to 0.96 resulting in a non- monotonic function. Similar non-monotonic behaviour was observed at the Zone Level and Building Level due to change in order resulting from a price change from 0 to 0.10. At zone level, the energy demand increased to 2.290 kWh from 2.267 kWh and then started decreasing. And at building level, the energy demand increased to 4.58 kWh from 4.53 kWh and then started decreasing. However, these minor spikes could be mitigated by appropriate deadband adjustments or modifying the step size.

These price vs energy demand functions correlate with the assumed quadratic utility functions which are doubly differentiable, ensuring convergence, as observed in convergence rate analysis (section 5.7.2.E).

C. Comfort

Initial observations indicate that room temperatures remained within comfort standards. This is crucial as energy-saving mechanisms should not compromise user comfort. While the temperatures were within prescribed limits, individual comfort can vary. Future studies should consider incorporating user feedback or preferences to fine-tune the system and achieve energy savings and optimal comfort. Further comprehensive studies should encompass all these aspects to understand the comfort performance completely.

5.8 Discussion

This section discusses the system overview, advantages, challenges including scalability, and limitations of the systems. In this thesis, the system, one approach, architecture, and framework have been suggested. As mentioned in the section 1.4, for this research work, a system building research methodology has been followed (which involves developing a system or parts thereof that provides a significant improvement in performance or functionality that was not available before).

5.8.1 System overview, advantages, and challenges

The system put forth offers a unique approach, architecture, and framework. Based on the system building research methodology, it proposes:

1. A framework for integrating Personal Environment Control Systems (PECS) devices with building controls.
2. A market-driven control paradigm that employs "*price*" as the key operational parameter and use the framework to integrate the PECSs available within the task and with the ambient control system.
3. Introduction of the SmartHub (a task level PECS which acts as a central coordinator for the with-in task devices like PECS, sensors, and so forth), which act as primary coordination devices within task-oriented environments.

Incorporating default system models and pricing formulations from Akkermans et al. [27] and Samadi et al. [26] provide a foundational layer for the framework, which allows advanced algorithms to be added if necessary. The evidence from the case study suggests the system's potential to yield significant energy demand flexibility, underpinned by stability and fairness.

The system provides greater flexibility in organizing agents, allowing for better control strategies. At the task level, the local PCA has the flexibility to choose how the allocated energy is utilised to maximise the user's welfare by considering the user preferences, vis-e-vis preference for thermal comfort or lighting comfort or plug loads. Moreover, the PECSs PA agents can operate as per the occupant's preferences (vis-e-vis, for example, warm, normal, or cool is preferred for thermal comfort), maximizing utilization of allocated energy. This means that energy allocation can be dynamically modified based on user comfort preferences, ensuring user welfare is maximized. This flexibility extends to energy trading, and billing at task level.

Similarly, at the zone level, the local zone level PCA can choose how the allocated energy is re-distributed among the task and the ambient controls by manipulating the weights (that is, whether to increase the usage of ambient controls or PECSs) according to zone level operating conditions (like total occupancy patterns, climate). For instance, when overall occupancy is minimal, operating the zone control systems at their lowest is advantageous while prioritizing energy allocation to the PECS. Conversely, when occupancy is substantial, there is a preference for amplifying ambient controls (by directing more energy to the zone controls), using the PECS to fine-tune boundary conditions. On a building scale, the PCA can reallocate energy in line with zone priorities. For instance, ensuring optimal conditions in a server room might precede climate control in a cafeteria.

Furthermore, the task heater to keep the user warm may increase energy consumption in an unregulated environment. At the same time, the ambient air conditioner system is cooling the space, which results in wasted energy. However, in our system, the energy allocation is regulated among the task device and ambient systems, potentially decreasing the energy wastages.

If the intermittent PCAs are set to state online and mode PASS_ON, the hierarchical distributed MAS becomes a single-level distributed MAS.

In the case study experiments, we have seen that this architecture is suitable because the PECS can work independently. Still be controlled, even if communication with the primary devices fail. Besides, communication is required majorly at the time of computing market equilibrium. Once the price is communicated among various agents, there is a period associated with every published price. If the device loses communications after receiving the price, it can operate for the published price associated period.

Furthermore, on elapse of the period, it can change to predefined device default operating mode and continue to operate as a non-participating device. In practice, this mechanism gives ample time to re-establish communication. Nevertheless, if required, demand flexibility measures can be considered to minimise the effect of such scenarios.

The approach presented is expected to increase the cost of providing comfort because of the increased cost of embedded hardware and software. However, each PECS does not need to have its computing resource for its price functions; it is possible to have distributed computing. Therefore, the SmartHub might have a different function for different devices instead of each PECS having price control in each device. One crucial point is that if this must be suitable and without any coordinator, each device should have a minimum capability to act as coordinator and control itself. That is, each device works on its own. However, when two or more devices are available, one can act as the coordinator. Otherwise, there can be slaves, which are less intelligent, and there can be a coordinator to control them. Usually, when such kind of more processing is given to devices, sometimes processing might take more power than the kind of power we want to minimize (i.e., the energy required to provide comfort). Hence, all these computing/metering devices must be extremely low power. Nevertheless, an overall energy saving potential is found for the case study implementation.

Also, it would be argued that will individual participants really "*worry*" about optimization and such and get distracted from his core activities and loose productivity. However, the framework facilitates either the agent takes the end-user inputs in real-time or autonomous agents that work based on the end-user preferences like (prefer warm/cool/normal and such) as parameters in the configuration files. The users give their preference once and then automatically things will take care and control in such a way that the user preferences are met. User intervention is not required every time once the user has given the preferences. And the system will keep working on its own unless the user wants to change them. Also, users do not worry too much about these things and don't like to be also worried about this. However, they are worried about somebody else taking control of their comfort parameters and deciding on their behalf. It is important for the user to have control and having control over your environment that itself makes user comfortable. And the studies show increased productivity [Bauman, Johnson control, placebo effect].

Also, considering these aspects, in the framework, a hierarchical structure has been used wherein it is feasible to cluster group of energy-consuming devices into logical or physical separate groups and energy demand can be aggregated and bid accordingly.

5.8.2 Scalability

The system's hierarchical design enables logical and physical demarcation of energy-consuming devices and their agents and supports scalability. Key features that contribute to this scalability include:

1. **Efficient Data Communication:** Utilizing JSON messages, the system ensures streamlined data transfer, addressing robustness against communication disruptions and security.
2. **Message Attributes:** Each message is enriched with various attributes such as time-to-live, message timestamp, source and destination device ID and IP addresses, and time zone. These attributes, coupled with data values, types, and specific message types, ensure the reliability and relevance of each communication.
3. **Quality of Service Measures:**
    a. **Message Limitation:** The system minimizes unnecessary communication overhead by constraining the number of messages exchanged between agents.
    b. **Retransmission Control:** If a message fails, it is re-sent only if the message's time-to-live has not expired, ensuring timely communications.
    c. **Parallel Publishing:** Enables simultaneous communication to multiple devices, enhancing efficiency.
    d. **Quality Checks:** Additional quality control measures are integrated at communications' sending and receiving ends.
4. **Optimized Data Flow:** The system practices grouping strategies like considering PECS within a task as a single entity under the umbrella of higher-level Price Controller Agents. This approach also curbs excessive data flow from these devices to the top tiers.
5. **PECS Independence:** PECS devices are designed to function autonomously, providing resilience against potential communication failures with primary devices. This independence is particularly crucial during market equilibrium computations when communication becomes vital. After price communication among agents, devices can operate without further communication for a set duration

corresponding to the communicated price. A device can still function using the provided price details if it loses connection after this phase. Furthermore, once this duration ends, it defaults to a predefined operational mode, behaving as a non-participating device. This design ensures that devices have chances to restore communication.

5.8.3 Security and privacy

Security and privacy stand out as pivotal concerns in this system. There exists potential for sensitive information, such as occupancy patterns and setpoint preferences, to be leveraged for user profiling. To mitigate this, the system adopts several protective measures:

1. **Limited Data Communication:** Agents in the system are designed to exchange only essential information over secure channels, ensuring the minimization of any unnecessary data exposure.
2. **Opaque Energy Redistribution:** The specifics of how allocated energy gets redistributed among downstream devices are kept concealed from upstream levels. Only aggregated data is relayed upward, maintaining data granularity and user privacy.
3. **Private Utility Functions:** The utility functions are exclusive to each end device. This design choice ensures that profiling a user at the task level becomes challenging for upstream agents.
4. **Inherent Privacy:** The system's architecture and design principles embed privacy at its core, making it resilient against potential profiling and privacy breaches.
5. **Robust Security Protocols:** The MAS platform, in this context Eclipse VOLTTRON™, fortifies security. It employs TLS/SSL for secure communication between agents. In addition, other security measures, such as authentication

protocols, are in place, ensuring the integrity and confidentiality of data exchanges.

5.8.4 Limitation

Transactive Control (TC) in PECS demands integration of individual preferences, which can be subjective. Translating these preferences into actionable data within a micro-environment is challenging. The introduction of smaller, personal systems into the traditional domain of larger devices presents inherent complexities. Aspects like the '*shadow price*' in PECS need to be considered more deeply, reflecting the nuanced nature of comfort valuation. Besides, quantifying comfort can be subjective and varies from one individual to another. How does one weigh a personal comfort need against broader energy-saving goals in real-time?

Additionally, the system has certain limitations:

1. **Energy Management in Non-cooperative Scenarios:** There are situations where agents might either underutilize or overutilize their energy allocations in non-cooperative game setups. While monitoring mechanisms can track and potentially shut off defaulting agents that surpass their contractual energy demand, this primarily addresses connected plug loads. There remains a crucial need to regulate other types of energy-consuming devices.
2. **Convergence Challenges:** The transactive control application has inherent challenges in ensuring convergence.
3. **Technological Delays:** Overheads associated with specific protocols can cause transmission time lags, posing challenges to real-time operations.
4. **Influence of Hierarchical Levels on Convergence Rate:** The rate at which the system reaches convergence can be significantly influenced by the number of

hierarchical levels present. A more considerable number of levels might mean longer convergence times, depending on the system's intricacy.

5. **Message Transmission Time Lag:** Experimental data revealed that the time lag for message transmission can vary widely, ranging from 300 milliseconds to 2 seconds. Such variability can significantly impact the number of iterations taken to reach convergence. Additionally, the number of hierarchical levels in the system directly affects the convergence rate. Nevertheless, comprehensive simulations (with over a million test runs at each level under random initial states) conducted for this thesis have demonstrated a promising 100% convergence rate.

5.8.5 Recommendations

1. **User Empowerment and Awareness:** For such a system to be successful, end-users should be informed and aware of the implications of their choices. This requires intuitive interfaces, leveraging smart AI assistants that guide decisions without overwhelming the user.
2. **Balancing Individual Comfort with Broader Goals:** Ensuring a delicate balance between individual comfort and broader energy-saving or grid-stability goals is crucial. An overemphasis on one might compromise the other.
3. **Security and Privacy:** With increased interconnectivity, ensuring the security of the system and the privacy of the users becomes paramount. This entails rigorous cybersecurity measures and clear data privacy guidelines.

5.9 Conclusion

We demonstrated the control of PECSs within a task and ambient control system based on price. Our experiments at the lab scale made the PECS devices connect and communicate with each other. Moreover, they have an integrated interface for the user to control them. The integration has been achieved using the open source transactive network platform Eclipse VOLTTRON™. The platform is lightweight enough to be deployed on IoT scale microcontrollers and h/w platforms. The platform provides the messaging interface and transactive function for different devices. The price signal coming from the grid is communicated from building level to zone level to various PECS, and the price functions are controlling the setpoints or on/off. Moreover, the PECS work independently and control their output.

Similarly, ambient HVAC and ambient lighting setpoints can also be controlled using the price functions at the space level, and the HVAC and lighting loads respond to the price control signals. This was achieved by connecting the Eclipse VOLTTRON™ over the industry-standard communication protocol (BACnet/Modbus) with the industry available BMS (Schneider).

The overall observations about the system evaluation.

1. The results show that the system can operate as designed for a given optimal price. The corresponding setpoints and comfort parameters are maintained well within the prescribed ranges. Also, from the data, the actual energy consumed by various devices and at various levels is within the computed energy demand from the energy demand curves. It is noticed that the actual energy demand for a given time slot is within the predicted energy demand computed from respective energy demand functions. However, at times, the active power spikes are observed, potentially impacting the overall energy demand goals.
2. The Price Vs Energy demonstrates that the load can be shed in response to the price. Further, it is observed that there is a energy demand flexibility of 26% compared to

the baseline condition, which is in line with the simulation studies available in the literature. Currently, quantitative evaluation of the energy shedding potential for demand response and energy savings is not done. However, we see that there is good potential and needs further study.

In the evolving landscape of smart buildings and grids, the transformative potential of PECS in energy management stands out. Merging the paradigms of DR, TC, and PECS promises a future where energy consumption is not just efficient but also personalized to the nth degree. Yet, turning this vision into reality requires coordinated research, development, and a strong emphasis on user-centric innovation.

Future steps include extensive field testing to ascertain real-world efficiency and garner feedback from occupants on comfort levels. After all, comfort remains a subjective domain, and true energy demand saving potential need to be thoroughly examined.

In summary, while the exploration of TC in PECS is promising, it necessitates a multidisciplinary approach, drawing on expertise from energy, behavioural science, systems design, and economics to ensure its holistic implementation. Such collaboration will be pivotal in bringing about a comprehensive and effective implementation.

Chapter 6

Conclusion

"We can only see a short distance ahead, but we can see plenty there that needs to be done" —Alan Turing

The presented chapter provides a concise summary and conclusion of the research undertaken, its scope, the application of the current work, and the significant contributions made through the thesis.

6.1 Overview

This thesis provides a comprehensive overview of the conducted research, highlighting the pertinent research questions that emerged. The research focused on the challenge of integrating personal environmental control systems (PECS) with broader building energy management to bring the occupant (that is, the end-user) into the demand response loop; thereby, end-users can manage Demand Response (DR) events at the individual task level and prioritise load reduction using Transactive Control.

Central to this thesis is the generalised architecture and framework of iSPACE - an intelligent System for Personal-Ambient Control and Energy Efficiency. The system integrates various PECS in a unified way with the ambient control system for demand response management using transactive control concepts to bring the end-user into the demand response loop. This system facilitates end-user participation in real-time demand response, managing energy usage with much more granularity (i.e., at a task level).

A pioneering aspect of this work is the market-based control strategy, wherein "*price*" becomes a key operational parameter; a US patent was granted [84]. The system offers a novel approach, architecture, and framework derived from systematic building research methodology. It introduces a framework for seamlessly integrating Personal Environment Control Systems (PECS) with the ambient control and building management systems. The underlying mathematical underpinning of the concept has been identified. The framework's system models, and pricing formulations permit the integration of advanced algorithms if needed. Case study evidence suggests the system's potential for substantial energy efficiency enhancements while ensuring stability and fairness during demand reduction.

The architecture offers enhanced agent organisation flexibility, promoting refined control strategies. At the task level, the local PCA can decide the optimal utilisation of the allotted energy, factoring in user preferences regarding thermal comfort, lighting, or plug loads. The PECS's PA agents operate based on the user's chosen settings, optimising the use of the allocated energy. This dynamism ensures energy allocation adjustments per user comfort preferences, thus maximising user welfare.

On the zone level, the local PCA determines the optimal energy redistribution between the task and ambient controls, adjusting the prioritisation based on factors like total occupancy or climate. For example, in scenarios with low overall occupancy, the focus shifts to directing energy towards PECSs. Conversely, high occupancy scenarios might demand more emphasis on ambient controls.

6.2 Scope

The scope for this research was limited to the following:

1. **Integration of PECS:** Integrating various PECS within task environments and ensuring seamless communication between the task and ambient.
2. **Application of Transactive Control:** Integration of PECS with transactive control mechanisms for efficient demand response in an office environment.

6.3 Research Contributions

The milestones achieved in this endeavour encompass theoretical and practical aspects of the study. The main contributions made through this research are:

1. A comprehensive literature review of research (about 130 research papers) in the area of the PECS, Demand Response, and Transactive Control was conducted, and research gaps were identified. A review paper was published based on this work - titled "*A review of advances for thermal and visual comfort controls in personal environmental control (PEC) systems*" [12]. The research gaps identified in this review are the ones that are addressed in this thesis.
2. Designed and developed a generalised hierarchical distributed multi-agent system architecture and framework, iSPACE - **i**ntelligent **S**ystem for **P**ersonal-**A**mbient **C**ontrol and Energy **E**fficiency, to address identified gaps (section 2.6*)*. A pioneering aspect of this work is the market-based control strategy, wherein "*price*" becomes a key operational parameter; a US patent was granted [84].
3. The critical features/requirements of the system and conceptual the system at several levels of abstractions (i.e., what, how, where, who, when, and why of the system) along with various artefacts (e.g. use cases, class diagrams, and UML

models), various actors of the systems (vis-a-vis PECS, sensors, events, and users), and interactions in the system were detailed.

4. Designed and developed the interfaces between hardware and software components, fostering communication between devices and system stakeholders, presenting a technology agonistic communication between various PECS and ambient controls.
5. Developed the underlying mathematical underpinning of the system model and pricing formulation. The framework permits the integration of advanced algorithms.
6. Developed an original approach of price and energy demand functions replacing utility functions. Intuitively, it is easier to construct the price functions for various energy-consuming devices for practical purposes. Moreover, analytical or statistical methods can develop the energy demand functions, thereby offloading optimisation computation to achieve faster convergence at runtime.
7. Identified Eclipse VOLTTRON™, an open-source transactive control platform comparing various available MAS platforms.
8. Developed a functional prototype iSPACE system using an open-source MAS platform, deployed, and tested. Furthermore, designed and developed three PECS, SmartHub, RadiantCubicle, and SmartStrip, to deploy a proof-of-concept implementation of the framework to evaluate the system in the lab.
9. Highlighted the effectiveness of transactive controls at both task, zone, and building levels. Demonstrated the iSPACE system's capability to seamlessly integrate diverse PECS within task environments and with ambient control systems.
10. Demonstrated a comprehensive understanding of the system's strengths, areas for potential improvement, and its impact on energy efficiency through quantitative and qualitative analyses. The metrics indicate a 26% energy demand flexibility

compared to the benchmark, highlighting the substantial flexibility it offers for demand response management. Additionally, the study has showcased the system's potential for considerable energy efficiency enhancements, all while maintaining stability and fairness during periods of demand reduction.

11. Demonstrated that 100% convergence is possible by simulation study using the ground truth data derived from the experiment, which consisted of 1M test runs each at various levels (Building, Zone, and Task levels) on random system states and various parameter changes.
12. Provided recommendations from an in-depth examination of critical insights regarding the system's performance, advantages and challenges, including scalability and limitations.
13. An online codebase and the system's hardware design have been made available to the broader community, promoting further development and collaboration.
14. Established a functional testbed with the iSPACE prototype at IIITH in the FDD lab, providing a foundation for future research. The iSPACE prototype offers a platform for the research community, facilitating further experimentation and analytical model developments.

6.4 Application Of this Research Work

The key application of this research work include:

1. **Granularity in Energy Management:** The capability to manage energy usage at the task level, allowing real-time adjustments and end-user engagement with demand response.

2. **Engaging the End-User:** The system encourages end-user involvement in demand response management. The choice of comfort parameter adjustments during

demand reduction events is given to the occupant.

3. **Potential for Increased Occupant Satisfaction:** As users willingly adjust their comfort levels, there is a likelihood of improved satisfaction and maintained productivity.
4. **Innovative Control Applications:** With the data from individual SmartHubs, innovative applications can be developed.

6.5 Summary

The journey through this research unveiled a critical facet of the future of energy management – personal environmental control systems integrated with the ambient systems. As our world becomes increasingly intertwined with the cascade of technologies and as our energy sources diversify, managing that energy becomes not just a matter of convenience but of necessity.

The developed iSPACE system, central to this research, represents a significant leap in this direction and represent a blend of technological innovation and user-centric design. It underscores that effective energy management in buildings hinges on integrating the occupants - the end-users - into the system. By giving the end-user agency the ability to choose and modify their comfort parameters, iSPACE manages not just energy; it manages satisfaction and well-being.

The capability to manage energy at the task level introduces a new degree of precision, enabling swift adjustments in real-time demand scenarios and harmonising with the more extensive, evolving smart grid systems. This granularity is especially pertinent when considering the intermittent nature of renewable energy sources. As buildings increasingly adopt solar, wind, and other renewable sources, the need for dynamic demand-response management will grow. iSPACE addresses this emerging need head-on.

Beyond energy metrics, iSPACE fosters a transformative behavioural ethos. Empowering occupants to choose which comfort parameter to adjust during demand reduction events initiates a behavioural change. This active participation could lead to a ripple effect, fostering a culture of energy consciousness.

In essence, the strides made in this research offer more than just a technological solution. It presents a vision of future, where buildings are not just static entities but dynamic, responsive, and interactive spaces that prioritize energy efficiency and occupant well-being. iSPACE is a precursor to this vision, opening avenues for continued innovation in creating truly intelligent "*Grid-Responsive*" buildings.

Related Publications

Publications

Godithi, Sam Babu, Enna Sachdeva, Vishal Garg, Richard Brown, Christian Kohler, and Rajan Rawal. 2019. "A Review of Advances for Thermal and Visual Comfort Controls in Personal Environmental Control (PEC) Systems." Intelligent Buildings International, November, 1–30. Doi: 10.1080/17508975.2018.1543179. (Scopus cite score 4.6 (2023))

Patent

US: V. Garg, N. Reddy, **S. B. Godithi**, R. E. Brown, C. Kohler, and R. Singh, "System and apparatus for and methods of control of localized energy use in a building using price set points." Patent No.: US10496066B2, Grant Date: 03-Dec-2019. (**US Patent Granted**)

IN: V. Garg, N. Reddy, **S. B. Godithi**, R. E. Brown, C. Kohler, and R. Singh, "System and method for monitoring and controlling power consumption in a local environment" Application No.: IN 201744029921, Pub Date: 02-Mar-2018. (**Patent pending - Filed FER response**)

Bibliography

[1] M. F. Akorede, H. Hizam, and E. Pouresmaeil, ‘Distributed energy resources and benefits to the environment’, *Renewable and Sustainable Energy Reviews*, vol. 14, no. 2, pp. 724–734, Feb. 2010, doi: 10.1016/j.rser.2009.10.025.

[2] DNV GL, ‘A Review of Distributed Energy Resources’, Sep. 2014. [Online]. Available: https://documents.dps.ny.gov/public/Common/ViewDoc.aspx?DocRefId=%7B52BC75E8-BBDD-4E2C-9D7D-34C160E0753A%7D

[3] EIA, ‘International Energy Outlook 2023’, U.S. Energy Information Administration, 2023. [Online]. Available: https://www.eia.gov/outlooks/ieo/pdf/IEO2023_Narrative.pdf

[4] H. Jiayi, J. Chuanwen, and X. Rong, ‘A review on distributed energy resources and MicroGrid’, *Renewable and Sustainable Energy Reviews*, vol. 12, no. 9, pp. 2472–2483, Dec. 2008, doi: 10.1016/j.rser.2007.06.004.

[5] S. Li, J. Lian, A. J. Conejo, and W. Zhang, ‘Transactive Energy Systems: The Market-Based Coordination of Distributed Energy Resources’, *IEEE Control Systems Magazine*, vol. 40, no. 4, pp. 26–52, Aug. 2020, doi: 10.1109/MCS.2020.2990514.

[6] M. Wissner, ‘The Smart Grid – A Saucerful of secrets?’, *Applied Energy*, vol. 88, no. 7, pp. 2509–2518, Jul. 2011, doi: 10.1016/j.apenergy.2011.01.042.

[7] P. Siano, ‘Demand response and smart grids—A survey’, *Renewable and Sustainable Energy Reviews*, vol. 30, no. Supplement C, pp. 461–478, Feb. 2014, doi: 10.1016/j.rser.2013.10.022.

[8] Z. Liu, Q. Wu, S. Huang, and H. Zhao, ‘Transactive energy: A review of state of the art and implementation’, in *2017 IEEE Manchester PowerTech*, Jun. 2017, pp. 1–6. doi: 10.1109/PTC.2017.7980892.

[9] H. Lund, P. A. Østergaard, D. Connolly, and B. V. Mathiesen, ‘Smart energy and smart energy systems’, *Energy*, May 2017, doi: 10.1016/j.energy.2017.05.123.

[10] T. M. Lawrence *et al.*, ‘Ten questions concerning integrating smart buildings into the smart grid’, *Building and Environment*, vol. 108, pp. 273–283, Nov. 2016, doi: 10.1016/j.buildenv.2016.08.022.

[11] K. Kok and S. Widergren, ‘A Society of Devices: Integrating Intelligent Distributed Resources with Transactive Energy’, *IEEE Power and Energy Magazine*, vol. 14, no. 3, pp. 34–45, May 2016, doi: 10.1109/MPE.2016.2524962.

[12] S. B. Godithi, E. Sachdeva, V. Garg, R. Brown, C. Kohler, and R. Rawal, ‘A review of advances for thermal and visual comfort controls in personal environmental control

(PEC) systems', *Intelligent Buildings International*, vol. 11, no. 2, pp. 75–104, Apr. 2019, doi: 10.1080/17508975.2018.1543179.

[13] EIA, 'International Energy Outlook 2017', U.S. Energy Information Administration, Sep. 2017. [Online]. Available: https://www.eia.gov/outlooks/ieo/pdf/0484(2017).pdf

[14] IEA, 'World Energy Outlook 2017 - Executive Summary', International Energy Agency, 2017. [Online]. Available: https://www.iea.org/Textbase/npsum/weo2017SUM.pdf

[15] IEA, 'World Energy Outlook 2023', International Energy Agency, 2023. [Online]. Available: https://iea.blob.core.windows.net/assets/24b94acb-5ae6-451d-b79a-68a875d773d1/Executivesummary-WorldEnergyOutlook2023.pdf

[16] M. Iyer, S. Kumar, P. Mathew, S. Mathew, H. Stratton, and M. Singh, 'Commercial buildings energy data framework for India: an exploratory study | Energy Efficiency', vol. 14, no. 67, 2021, doi: https://doi.org/10.1007/s12053-021-09971-y.

[17] S. Kumar *et al.*, 'Energy Conservation and Commercialization (ECO-III)', *Developing an Energy Conservation Building Code, Implementation Strategy in India*, 2010, Accessed: Sep. 13, 2017. [Online]. Available: http://www.5dstudios.com/clients/gcca/wp-content/uploads/2012/05/Developing-an-ECBC-Implementation-Strategy-in-India-Report-No.1028.pdf

[18] S. Kumar, N. Yadav, M. Singh, and S. Kachhawa, 'Estimating India's commercial building stock to address the energy data challenge', *Building Research & Information*, vol. 47, no. 1, pp. 24–37, Jan. 2019, doi: 10.1080/09613218.2018.1515304.

[19] International Energy Agency, *India Energy Outlook 2021*. OECD, 2021. doi: 10.1787/ec2fd78d-en.

[20] M. Wissner, 'The Smart Grid – A Saucerful of secrets?', *Applied Energy*, vol. 88, no. 7, pp. 2509–2518, Jul. 2011, doi: 10.1016/j.apenergy.2011.01.042.

[21] V. M. Balijepalli, V. Pradhan, S. A. Khaparde, and R. M. Shereef, 'Review of demand response under smart grid paradigm', in *Innovative Smart Grid Technologies-India (ISGT India), 2011 IEEE PES*, IEEE, 2011, pp. 236–243.

[22] H. Lund, A. N. Andersen, P. A. Østergaard, B. V. Mathiesen, and D. Connolly, 'From electricity smart grids to smart energy systems - A market operation based approach and understanding', *Energy*, vol. 42, no. 1, pp. 96–102, Jun. 2012, doi: 10.1016/j.energy.2012.04.003.

[23] R. Lasseter *et al.*, 'The CERTS microgrid concept', *White paper for Transmission Reliability Program, Office of Power Technologies, US Department of Energy*, vol. 2, no. 3, p. 30, 2002, Accessed: Sep. 14, 2017. [Online]. Available: http://www.westernsunsystems.comorwww.gosolarcalifornia.org/research/notices/2002-05-02_WORKSHOP_SUPP.PDF

[24] GridWise Architecture Council, 'GridWise Transactive Energy Framework Version

1.0', GridWise Architecture Council (GWAC), 2015. Accessed: Sep. 15, 2017. [Online]. Available: http://www.gridwiseac.org/pdfs/te_framework_report_pnnl-22946.pdf

[25] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, 'Rate control for communication networks: shadow prices, proportional fairness and stability', *Journal of the Operational Research Society*, vol. 49, no. 3, pp. 237–252, Mar. 1998, doi: 10.1057/palgrave.jors.2600523.

[26] P. Samadi, A.-H. Mohsenian-Rad, R. Schober, V. W. S. Wong, and J. Jatskevich, 'Optimal Real-Time Pricing Algorithm Based on Utility Maximization for Smart Grid', in *2010 First IEEE International Conference on Smart Grid Communications*, Gaithersburg, MD, USA: IEEE, Oct. 2010, pp. 415–420. doi: 10.1109/SMARTGRID.2010.5622077.

[27] H. Akkermans, J. Schreinemakers, and K. Kok, 'Microeconomic Distributed Control: Theory and Application of Multi-Agent Electronic Markets', *Proceedings of CRIS 2004 - 2nd International Conference on Critical Infrastructures*, pp. 106–114, Oct. 2004, [Online]. Available: http://www.crisp.ecn.nl/deliverables/D4.3.pdf

[28] S. H. Clearwater, *Market-based Control: A Paradigm for Distributed Resource Allocation*. World Scientific, 1996.

[29] B. W. Boehm, 'Software Engineering Economics', *IEEE Transactions on Software Engineering*, vol. SE-10, no. 1, pp. 4–21, Jan. 1984, doi: 10.1109/TSE.1984.5010193.

[30] J. M. Perloff, *Microeconomics, Global Edition*, 8 edition. Harlow, UK: Pearson, 2018.

[31] K. Kok, 'The PowerMatcher: Smart Coordination for the Smart Electricity Grid', Technical University of Denmark, 2013. Accessed: Dec. 20, 2018. [Online]. Available: https://www.researchgate.net/publication/261286701_The_PowerMatcher_Smart_Coordination_for_the_Smart_Electricity_Grid

[32] Z. Liu, Q. Wu, S. Huang, and H. Zhao, 'Transactive energy: A review of state of the art and implementation', in *2017 IEEE Manchester PowerTech*, Manchester, United Kingdom: IEEE, Jun. 2017, pp. 1–6. doi: 10.1109/PTC.2017.7980892.

[33] S. Katipamula, R. Lutes, G. Hernandez, J. Haack, and B. Akyol, 'Transactional network: Improving efficiency and enabling grid services for buildings', *Science and Technology for the Built Environment*, vol. 22, no. 6, pp. 643–654, Aug. 2016, doi: 10.1080/23744731.2016.1171628.

[34] A. Mercurio, A. D. Giorgio, and A. Quaresima, 'Distributed control approach for Community Energy Management Systems', in *2012 20th Mediterranean Conference on Control Automation (MED)*, Jul. 2012, pp. 1265–1271. doi: 10.1109/MED.2012.6265813.

[35] A. Mercurio, A. D. Giorgio, and A. Quaresima, 'Distributed control approach for community energy management systems in presence of storage', in *2012 20th Mediterranean Conference on Control Automation (MED)*, Jul. 2012, pp. 1303–1308. doi: 10.1109/MED.2012.6265819.

[36] A. Mirakhorli and B. Dong, 'An Open Source Smart Building Energy Management

Platform through VOLTTRON’, in *Proceedings of the 15th IBPSA Conference 2017*, Aug. 2017.

[37] S. Huang, J. Lian, H. Hao, and S. Katipamula, ‘Transactive Control Design for Commercial Buildings to Provide Demand Response’, *IFAC-PapersOnLine*, vol. 51, no. 34, pp. 151–156, 2019, doi: 10.1016/j.ifacol.2019.01.058.

[38] B. Liu, M. Akcakaya, and T. E. Mcdermott, ‘Automated Control of Transactive HVACs in Energy Distribution Systems’, *IEEE Transactions on Smart Grid*, vol. 12, no. 3, pp. 2462–2471, May 2021, doi: 10.1109/TSG.2020.3042498.

[39] F. Bauman and E. Arens, ‘Task/ambient conditioning systems: engineering and application guidelines’, *Center for Environmental Design Research, University of California, Berkeley*, 1996, Accessed: Sep. 11, 2017. [Online]. Available: http://escholarship.org/uc/item/0r36z48d.pdf

[40] M. P. Andersen *et al.*, ‘Well-connected microzones for increased building efficiency and occupant comfort’, *Proceedings of ACEEE Summer Study on Energy Efficiency in Buildings*, p. 17, Aug. 2016, [Online]. Available: https://escholarship.org/uc/item/7710g5cb

[41] F. H. Rohles, ‘Temperature. & Temperament. A Psychologist Looks at Comfort.’, *ASHRAE Journal*, Feb. 2007.

[42] S. Tom, ‘Managing Energy And Comfort: Don’t Sacrifice Comfort When Managing Energy’, *ASHRAE Journal*, vol. 50, no. 6, p. 18, 2008, Accessed: Sep. 09, 2017. [Online]. Available: http://search.proquest.com/openview/7b2f413298cf1bc1e1b5b67405b09123/1?pq-origsite=gscholar&cbl=41118

[43] C. Marnay and R. Firestone, ‘Microgrids: An emerging paradigm for meeting building electricity and heat requirements efficiently and with appropriate energy quality’, *Lawrence Berkeley National Laboratory*, 2007, Accessed: Sep. 14, 2017. [Online]. Available: http://escholarship.org/uc/item/54b7r66j.pdf

[44] J. S. Vardakas, N. Zorba, and C. V. Verikoukis, ‘A Survey on Demand Response Programs in Smart Grids: Pricing Methods and Optimization Algorithms’, *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, pp. 152–178, Firstquarter 2015, doi: 10.1109/COMST.2014.2341586.

[45] F. Sehar, M. Pipattanasomporn, and S. Rahman, ‘Integrated automation for optimal demand management in commercial buildings considering occupant comfort’, *Sustainable Cities and Society*, vol. 28, pp. 16–29, Jan. 2017, doi: 10.1016/j.scs.2016.08.016.

[46] J. Hu, G. Yang, K. Kok, Y. Xue, and H. W. Bindner, ‘Transactive control: a framework for operating power systems characterized by high penetration of distributed energy resources’, *J. Mod. Power Syst. Clean Energy*, vol. 5, no. 3, pp. 451–464, May 2017, doi: 10.1007/s40565-016-0228-1.

[47] P. Carlsson, F. Ygge, and A. Andersson, ‘Extending equilibrium markets’, *IEEE Intelligent Systems*, vol. 16, no. 4, pp. 18–26, Jul. 2001, doi: 10.1109/5254.941354.

[48] F. Bauman, H. Zhang, E. Arens, and C. Benton, ‘Localized comfort control with a desktop task conditioning system: laboratory and field measurements’, *ASHRAE Transactions*, vol. 99, 1993, Accessed: Sep. 13, 2017. [Online]. Available: http://escholarship.org/uc/item/9377c6xm.pdf

[49] F. Bauman and E. Arens, ‘Task/ambient conditioning systems: engineering and application guidelines’, *Center for Environmental Design Research, University of California, Berkeley*, 1996, Accessed: Sep. 11, 2017. [Online]. Available: http://escholarship.org/uc/item/0r36z48d.pdf

[50] F. Bauman, T. Carter, A. Baughman, and E. Arens, ‘Field study of the impact of a desktop task/ambient conditioning system in office buildings’, *ASHRAE Transactions*, vol. 104, p. 1153, 1998, Accessed: Nov. 02, 2016. [Online]. Available: http://search.proquest.com/openview/9a2c1f02c3042cc97b4709c560a6540e/1?pq-origsite=gscholar

[51] R. Rawal, M. Schweiker, O. B. Kazanci, V. Vardhan, Q. Jin, and L. Duanmu, ‘Personal comfort systems: A review on comfort, energy, and economics’, *Energy and Buildings*, vol. 214, p. 109858, May 2020, doi: 10.1016/j.enbuild.2020.109858.

[52] H. Zhang, E. Arens, and Y. Zhai, ‘A review of the corrective power of personal comfort systems in non-neutral ambient environments’, *Building and Environment*, vol. 91, pp. 15–41, Sep. 2015, doi: 10.1016/j.buildenv.2015.03.013.

[53] A. K. Melikov and G. L. Knudsen, ‘Human Response to an Individually Controlled Microenvironment’, *HVAC&R Research*, vol. 13, no. 4, pp. 645–660, Jul. 2007, doi: 10.1080/10789669.2007.10390977.

[54] A. K. Melikov, T. Ivanova, and G. Stefanova, ‘Seat headrest-incorporated personalized ventilation: Thermal comfort and inhaled air quality’, *Building and Environment*, vol. 47, pp. 100–108, Jan. 2012, doi: 10.1016/j.buildenv.2011.07.013.

[55] A. K. Melikov, B. Krejciríková, J. Kaczmarczyk, M. Duszyk, and T. Sakoi, ‘Human response to local convective and radiant cooling in a warm environment’, *HVAC&R Research*, vol. 19, no. 8, pp. 1023–1032, Nov. 2013, doi: 10.1080/10789669.2013.842734.

[56] M. Veselý and W. Zeiler, ‘Personalized conditioning and its impact on thermal comfort and energy performance – A review’, *Renewable and Sustainable Energy Reviews*, vol. 34, pp. 401–408, Jun. 2014, doi: 10.1016/j.rser.2014.03.024.

[57] F. Bauman, H. Zhang, E. Arens, and C. Benton, ‘Localized comfort control with a desktop task conditioning system: laboratory and field measurements’, *ASHRAE Transactions*, vol. 99, 1993, Accessed: Sep. 13, 2017. [Online]. Available: http://escholarship.org/uc/item/9377c6xm.pdf

[58] W. Song, Z. Zhang, Z. Chen, F. Wang, and B. Yang, ‘Thermal comfort and energy performance of personal comfort systems (PCS): A systematic review and meta-

analysis', *Energy and Buildings*, vol. 256, p. 111747, Feb. 2022, doi: 10.1016/j.enbuild.2021.111747.

[59] S. Aghniaey and T. M. Lawrence, 'The impact of increased cooling setpoint temperature during demand response events on occupant thermal comfort in commercial buildings: A review', *Energy and Buildings*, vol. 173, pp. 19–27, Aug. 2018, doi: 10.1016/j.enbuild.2018.04.068.

[60] T. Hoyt, E. Arens, and H. Zhang, 'Extending air temperature setpoints: Simulated energy savings and design considerations for new and retrofit buildings', *Building and Environment*, vol. 88, pp. 89–96, Jun. 2015, doi: 10.1016/j.buildenv.2014.09.010.

[61] W. van M. Lichtenbelt, M. Hanssen, H. Pallubinsky, B. Kingma, and L. Schellen, 'Healthy excursions outside the thermal comfort zone', *Building Research & Information*, vol. 45, no. 7, pp. 819–827, Oct. 2017, doi: 10.1080/09613218.2017.1307647.

[62] D. Enescu, 'A review of thermal comfort models and indicators for indoor environments', *Renewable and Sustainable Energy Reviews*, vol. 79, pp. 1353–1379, Nov. 2017, doi: 10.1016/j.rser.2017.05.175.

[63] S. Watanabe, T. Shimomura, and H. Miyazaki, 'Thermal evaluation of a chair with fans as an individually controlled system', *Building and Environment*, vol. 44, no. 7, pp. 1392–1398, Jul. 2009, doi: 10.1016/j.buildenv.2008.05.016.

[64] A. K. Melikov and G. L. Knudsen, 'Human Response to an Individually Controlled Microenvironment', *HVAC&R Research*, vol. 13, no. 4, pp. 645–660, Jul. 2007, doi: 10.1080/10789669.2007.10390977.

[65] A. K. Melikov, T. Ivanova, and G. Stefanova, 'Seat headrest-incorporated personalized ventilation: Thermal comfort and inhaled air quality', *Building and Environment*, vol. 47, pp. 100–108, Jan. 2012, doi: 10.1016/j.buildenv.2011.07.013.

[66] A. K. Melikov, B. Krejciríková, J. Kaczmarczyk, M. Duszyk, and T. Sakoi, 'Human response to local convective and radiant cooling in a warm environment', *HVAC&R Research*, vol. 19, no. 8, pp. 1023–1032, Nov. 2013, doi: 10.1080/10789669.2013.842734.

[67] A. K. Melikov, 'Human body micro-environment: The benefits of controlling airflow interaction', *Building and Environment*, vol. 91, pp. 70–77, Sep. 2015, doi: 10.1016/j.buildenv.2015.04.010.

[68] J. F. Nicol and S. Roaf, 'Rethinking thermal comfort', *Building Research & Information*, vol. 45, no. 7, pp. 711–716, Oct. 2017, doi: 10.1080/09613218.2017.1301698.

[69] Y. Al Horr, M. Arif, A. Kaushik, A. Mazroei, M. Katafygiotou, and E. Elsarrag, 'Occupant productivity and office indoor environment quality: A review of the literature', *Building and Environment*, vol. 105, pp. 369–389, Aug. 2016, doi: 10.1016/j.buildenv.2016.06.001.

[70] H. Zhang, E. Arens, and Y. Zhai, 'A review of the corrective power of personal comfort systems in non-neutral ambient environments', *Building and Environment*, vol. 91, pp.

15–41, Sep. 2015, doi: 10.1016/j.buildenv.2015.03.013.

[71] M. Frontczak and P. Wargocki, ‘Literature survey on how different factors influence human comfort in indoor environments’, *Building and Environment*, vol. 46, no. 4, pp. 922–937, Apr. 2011, doi: 10.1016/j.buildenv.2010.10.021.

[72] G. Schiller, E. A. Arens, F. Bauman, C. Benton, M. Fountain, and T. Doherty, ‘A field study of thermal environments and comfort in office buildings’, *ASHRAE Transactions*, vol. 94 Part 2, Jan. 1988, Accessed: Sep. 13, 2017. [Online]. Available: http://escholarship.org/uc/item/4km240x7

[73] F. Bauman, T. Carter, A. Baughman, and E. Arens, ‘Field study of the impact of a desktop task/ambient conditioning system in office buildings’, *ASHRAE Transactions*, vol. 104, p. 1153, 1998, Accessed: Nov. 02, 2016. [Online]. Available: http://search.proquest.com/openview/9a2c1f02c3042cc97b4709c560a6540e/1?pq-origsite=gscholar

[74] A. Leaman and B. Bordass, ‘Productivity in buildings: the “killer” variables’, *Building Research & Information*, vol. 27, no. 1, pp. 4–19, Oct. 2010, doi: 10.1080/096132199369615.

[75] W. J. Fisk, ‘Health and Productivity Gains from Better Indoor Environments and Their Relationship with Building Energy Efficiency’, *Annual Review of Energy and the Environment*, vol. 25, no. 1, pp. 537–566, 2000, doi: 10.1146/annurev.energy.25.1.537.

[76] J. F. Nicol and M. A. Humphreys, ‘Adaptive thermal comfort and sustainable thermal standards for buildings’, *Energy and Buildings*, vol. 34, no. 6, pp. 563–572, Jul. 2002, doi: 10.1016/S0378-7788(02)00006-3.

[77] G. Brager, G. Paliaga, and R. De Dear, ‘Operable windows, personal control and occupant comfort.’, *ASHRAE Journal*, 2004, Accessed: Sep. 11, 2017. [Online]. Available: http://escholarship.org/uc/item/4x57v1pf.pdf

[78] S. Y. Lee and J. L. Brand, ‘Effects of control over office workspace on perceptions of the work environment and work outcomes’, *Journal of Environmental Psychology*, vol. 25, no. 3, pp. 323–333, Sep. 2005, doi: 10.1016/j.jenvp.2005.08.001.

[79] B. P. Haynes, ‘The impact of office comfort on productivity’, *Journal of Facilities Management*, vol. 6, no. 1, pp. 37–51, 2008, Accessed: Sep. 15, 2017. [Online]. Available: http://www.emeraldinsight.com/doi/abs/10.1108/14725960810847459

[80] M. Veselý and W. Zeiler, ‘Personalized conditioning and its impact on thermal comfort and energy performance – A review’, *Renewable and Sustainable Energy Reviews*, vol. 34, pp. 401–408, Jun. 2014, doi: 10.1016/j.rser.2014.03.024.

[81] G. Brager, H. Zhang, and E. Arens, ‘Evolving opportunities for providing thermal comfort’, *Building Research & Information*, vol. 43, no. 3, Jan. 2015, doi: 10.1080/09613218.2015.993536.

[82] S. Tanabe, M. Haneda, and N. Nishihara, ‘Workplace productivity and individual

thermal satisfaction', *Building and Environment*, vol. 91, pp. 42–50, Sep. 2015, doi: 10.1016/j.buildenv.2015.02.032.

[83] F. Bauman *et al.*, 'Changing the Rules: Innovative Low-Energy Occupant-Responsive HVAC Controls and Systems', *eScholarship*, Mar. 2017, Accessed: Sep. 13, 2017. [Online]. Available: http://escholarship.org/uc/item/23t9k6rm

[84] V. Garg, N. Reddy, S. B. Godithi, R. E. Brown, C. Kohler, and R. Singh, 'System and apparatus for and methods of control of localized energy use in a building using price set points', US10496066B2, Dec. 03, 2019 Accessed: Jan. 22, 2020. [Online]. Available: https://patents.google.com/patent/US10496066B2/en

[85] J. M. Perloff, *Microeconomics: Theory and Applications with Calculus, Global Edition*, 4 edition. Harlow London New York Boston: Pearson, 2017.

[86] S. Maharjan, Q. Zhu, Y. Zhang, S. Gjessing, and T. Basar, 'Dependable Demand Response Management in the Smart Grid: A Stackelberg Game Approach', *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 120–132, Mar. 2013, doi: 10.1109/TSG.2012.2223766.

[87] J. S. Vardakas, N. Zorba, and C. V. Verikoukis, 'A Survey on Demand Response Programs in Smart Grids: Pricing Methods and Optimization Algorithms', *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, pp. 152–178, Firstquarter 2015, doi: 10.1109/COMST.2014.2341586.

[88] S. P. Boyd and L. Vandenberghe, *Convex optimization*, Version 29. Cambridge New York Melbourne New Delhi Singapore: Cambridge University Press, 2023.

[89] M. P. Wellman, 'A Market-Oriented Programming Environment and its Application to Distributed Multicommodity Flow Problems', *arXiv:cs/9308102*, Jul. 1993, Accessed: Oct. 19, 2021. [Online]. Available: http://arxiv.org/abs/cs/9308102

[90] Y. He, N. Li, M. He, and D. He, 'Using radiant cooling desk for maintaining comfort in hot environment', *Energy and Buildings*, vol. 145, pp. 144–154, Jun. 2017, doi: 10.1016/j.enbuild.2017.04.013.

[91] V. R. Khare, R. Garg, J. Mathur, and V. Garg, 'Thermal Comfort Analysis of Personalized Conditioning System and Performance Assessment with Different Radiant Cooling Systems', *Energy and Built Environment*, Oct. 2021, doi: 10.1016/j.enbenv.2021.09.001.

[92] ASHRAE-Std-55, 'Thermal Environmental Conditions for Human Occupancy'. ASHRAE, 2010. [Online]. Available: www.ashrae.org

[93] G. Chitalia, M. Pipattanasomporn, V. Garg, and S. Rahman, 'Robust short-term electrical load forecasting framework for commercial buildings using deep recurrent neural networks', *Applied Energy*, vol. 278, p. 115410, Nov. 2020, doi: 10.1016/j.apenergy.2020.115410.

[94] E. Mocanu, P. H. Nguyen, M. Gibescu, and W. L. Kling, 'Deep learning for estimating

building energy consumption', *Sustainable Energy, Grids and Networks*, vol. 6, pp. 91–99, Jun. 2016, doi: 10.1016/j.segan.2016.02.005.

[95] F. Bellifemine, A. Poggi, and G. Rimassa, 'JADE- A FIPA compliant agent framework', *Proceedings of PAAM*, vol. 99, no. 33, pp. 97–108, 1999, [Online]. Available: http://jmvidal.cse.sc.edu/library/jade.pdf

[96] B. A. Akyol, J. N. Haack, S. Ciraci, B. J. Carpenter, M. Vlachopoulou, and C. W. Tews, 'VOLTTRON: An Agent Execution Platform for the Electric Power System', International Foundation for Autonomous Agents and Multiagent Systems, N/A, United States(US)., PNNL-SA-85853, Jun. 2012. Accessed: Aug. 24, 2023. [Online]. Available: https://www.osti.gov/biblio/1092023

[97] Craig H. Allwardt, Chandrika Sivaramakrishanan, Shwetha Niddodi, and Jereme Haack, 'VOLTTRON Modular Framework: Enabling flexible and scalable deployment solutions | Report | PNNL'. Accessed: Aug. 24, 2023. [Online]. Available: https://www.pnnl.gov/publications/volttron-modular-framework-enabling-flexible-and-scalable-deployment-solutions

[98] J. Haack, B. Akyol, N. Tenney, B. Carpenter, R. Pratt, and T. Carroll, 'VOLTTRON™: An agent platform for integrating electric vehicles and Smart Grid', in *Connected Vehicles and Expo (ICCVE), 2013 International Conference on*, IEEE, 2013, pp. 81–86. Accessed: Sep. 11, 2017. [Online]. Available: http://ieeexplore.ieee.org/abstract/document/6799774/

[99] J. A. Weems, 'Systems for workstation-mounted radiant panels', US10962242B1, Mar. 30, 2021 Accessed: Mar. 25, 2024. [Online]. Available: https://patents.google.com/patent/US10962242B1/en

[100] N. Ismail and D. Ouahrani, 'Modelling of cooling radiant cubicle for an office room to test cooling performance, thermal comfort and energy savings in hot climates', *Energy*, vol. 244, p. 123185, Apr. 2022, doi: 10.1016/j.energy.2022.123185.

[101] V. R. Khare, A. Mathur, J. Mathur, and M. Bhandari, 'Development of Personalized Radiant Cooling System for an Office Room', in *Proceedings of BS2015*, Hyderabad, India, Dec. 2015, p. 8. [Online]. Available: https://www.osti.gov/servlets/purl/1236589

[102] R. Garg, V. R. Khare, J. Mathur, and V. Garg, 'Performance Evaluation of Personalized Radiant Conditioning System for Cooling Mode', p. 8, 2017.

[103] Y. Chen, P. Xu, J. Gu, F. Schmidt, and W. Li, 'Measures to improve energy demand flexibility in buildings for demand response (DR): A review', *Energy and Buildings*, vol. 177, pp. 125–139, Oct. 2018, doi: 10.1016/j.enbuild.2018.08.003.

Appendices

A-1 Patent claims

Claims (12)

What is claimed is:

1) A method comprising:

   a) providing a plurality of apparatus, each apparatus of the plurality of apparatus comprising:

      a controller, the controller in communication with a control system, the control system comprising a building automation and control system (BACS) operable to control energy use of a building, the controller operable to control energy use of the apparatus;

      an input device to accept user input, the input device in communication with the controller;

      environmental sensors in communication with the controller;

      a light;

      a heating/cooling device; and

      a housing, the controller, the environmental sensors, the input device, the light, and the heating/cooling device being mounted to the housing, the housing being able to be placed on a table that is positioned within the building, the controller operable to actuate the light and the heating/cooling device;

b) receiving a power price from the control system at a first apparatus of the plurality of apparatus;

c) comparing the power price to a set point power price; and

d) adjusting power supplied each of the light of the first apparatus, the heating/cooling device of the first apparatus, a lighting device in a region proximate the first apparatus, and a heating/cooling device in the region proximate the first apparatus based on comparing the power price to the set point power price.

2) The method of claim 1, further comprising:

e) measuring or estimating a power use by the first apparatus of the plurality of apparatus;

f) sending the power use to the control system; and

g) receiving a new power price from the control system at the first apparatus of the plurality of apparatus.

3) The method of claim 1, wherein the power supplied is adjusted to at least one of the light of the first apparatus, the heating/cooling device of the first apparatus, a lighting device in the region proximate the first apparatus, and a heating/cooling device in the region proximate the first apparatus is based on input from the environmental sensors.

4) The method of claim 1, wherein the first apparatus further comprises an electrical output interface, and wherein the method further comprises adjusting power supplied to the electrical output interface in operation (d).

5) The apparatus of claim 1, wherein the first apparatus further comprises a wireless network interface, and wherein the first apparatus receives the power price from the control system through the wireless network interface.

6) The method of claim 1, wherein operation (d) includes reducing power supplied to at least one of the light of the first apparatus, the heating/cooling device of the first apparatus, the lighting device in the region proximate the first apparatus, and the heating/cooling device in the region proximate the first apparatus when the power price is greater than the set point power price.

7) A method comprising:

   a) providing a control system comprising a building automation and control system (BACS) operable to control energy use of a building and a plurality of apparatus, each apparatus of the plurality of apparatus comprising:

      a controller, the controller in communication with the control system

      an input device to accept user input, the input device in communication with the controller;

      environmental sensors in communication with the controller;

      a light;

      a heating/cooling device; and

      a housing, the controller, the environmental sensors, the input device, the light, and the heating/cooling device being mounted to the housing, the housing being able to be placed on a table that is positioned within the building, the controller operable to actuate the light and the heating/cooling device;

   b) sending a power price from the control system to a first apparatus of the plurality of apparatus;

c) comparing the power price to a set point power price at the first apparatus; and

d) adjusting power supplied to each of the light of the first apparatus, the heating/cooling device of the first apparatus, a lighting device in a region proximate the first apparatus, and a heating/cooling device in the region proximate the first apparatus based on comparing the power price to the set point power price.

8) The method of claim 7, further comprising:

e) measuring or estimating a power use by the first apparatus of the plurality of apparatus at the first apparatus;

f) receiving the power use at the control system; and

g) determining a new power price.

9) The method of claim 7, wherein the power supplied is adjusted to at least one of the light of the first apparatus, the heating/cooling device of the first apparatus, a lighting device in the region proximate the first apparatus, and a heating/cooling device in the region proximate the first apparatus is based on input from the environmental sensors.

10) The method of claim 7, wherein the first apparatus further comprises an electrical output interface, and wherein the method further comprises adjusting power supplied to the electrical output interface in operation (d).

11) The apparatus of claim 7, wherein the first apparatus further comprises a wireless network interface, and wherein the control system sends the power price to the first apparatus via the wireless network interface.

12) The method of claim 7, wherein operation (d) includes reducing power supplied to at least one of the light of the first apparatus, the heating/cooling device of the first apparatus, the lighting device in the region proximate the first apparatus, and the heating/cooling device in the region proximate the first apparatus when the power price is greater than the set point power price.

A-2 iSPACE Message parameters

Sl. No.	Parameter	Description
1	MSG_TYPE	An integer enumerate parameter defines the message type (vis-e-vis 0 - Price Message, 1 - Budget Message, 2 - Energy Message, 3 - Active Power Message).
2	VALUE	This parameter contains the value. It would be a price, budget, power, energy, or data of any complex structure.
3	VALUE_DATA_TYPE	This parameter contains the data type of the value parameter.
4	UNITS	This parameter contains the units of the value parameter, if any.
5	PRICE_ID	This parameter contains the price id of the message if the message type is Price Message, or it contains a price id corresponding to the current message.
6	DURATION	This parameter contains the duration in seconds for which this message is applicable.
7	ISOPTIMAL	This parameter is a Boolean value indicating whether the value corresponds to an optimal condition or not.
8	ONE_TO_ONE	This parameter is a Boolean value indicating whether this message is intended for a particular device or all devices.
9	SRC_IP	This parameter contains the IP address of the message origination. Note:

Sl. No.	Parameter	Description
		1. If SRC_IP is the same as the local IP, the message originated from the local device's agent. This check is helpful to identify whether a message originated locally.
10	SRC_DEVICE_ID	This parameter contains the Device ID of the message origination.
11	DST_IP	This parameter contains the IP address of the destination device if the ONE_TO_ONE parameter is True else None.
12	DST_DEVICE_ID	This parameter contains the Device ID of the destination device if the ONE_TO_ONE parameter is True else None.
13	TTL	This parameter contains the value of the Time-To-Live in seconds. The duration of the message lives in the communication channels w.r.t. to the timestamp. Note: 1. At each level, the TTL is decremented accordingly. 2. The routing agents do not forward if the TTL expires, i.e., the message is not valid if the current time – timestamp > TTL. 3. Also, the routing agents use this parameter to keep retrying till TTL expires or the max retries limit.

Sl. No.	Parameter	Description
		4. The intended recipients shall act upon the message only if the message arrives earlier than TTL. 5. In the bidding process, this parameter can indicate the bid should be responded to within what period.
14	TS	This parameter contains the timestamp of the message.
15	TZ	This parameter indicates the time zone of the TS parameter.
16	Energy Category	An integer enumerates parameter to indicate the energy category the message corresponds to (vis-e-vis 0 - Cooling, 1 - Lighting, 2 - Plug Load, and 9 - Mixed Load).

A-3 PCA's key configuration parameters

Sl. No.	Configuration Parameter	Description
1	mode_pass_on_params	
1.1	bid_timeout	The period within which the associated energy-consuming devices should respond with their bids
1.2	weight_factors	This parameter is a list of the locally associated energy-consuming device's weightage. Example: { 'SmartHub': 0.3, 'RadiantCubicle': 0.5, 'SmartStrip': 0.2}
2	mode_default_opt_params	
2.1	publish_optimal	This Boolean parameter checks if the current PCA can conclude the local auctioning, i.e., publish optimal price. Generally, Building Level PCA is configured to conclude the auctioning.
2.2	us_bid_timeout	This parameter is the maximum time for the bidding process to be completed.
2.3	lc_bid_timeout	This parameter is the max time for each bidding iteration. This parameter is the max time the downstream PCA should conclude its local bidding and respond with its energy demand bid.

Sl. No.	Configuration Parameter	Description
2.4	max_iterations	This parameter is the allowed number of maximum iterations.
2.5	max_repeats	This parameter is the allowed maximum number of consecutive iterations that result in no change in price.
2.5	deadbands	This parameter is a list of the locally associated energy-consuming device's deadbands. Example: {'SmartHub': 10, 'RadiantCubicle': 10, 'SmartStrip': 5}
2.7	Gammas	This parameter lists the locally associated energy-consuming device's step sizes. Generally computed as: $\delta = \frac{max_{price} - min_{price}}{max_{energy} - min_{energy}}$ Example: {'SmartHub': 0.0794, 'RadiantCubicle': 0.0040, 'SmartStrip': 0.0030}
2.8	Alphas	This parameter lists the locally associated energy-consuming device's learning rates. It can be tuned based on a statistical analysis of the system.
2.9	weight_factors	Same as weight_factors for mode_pass_on_params

A-4 SmartHub – PCB design

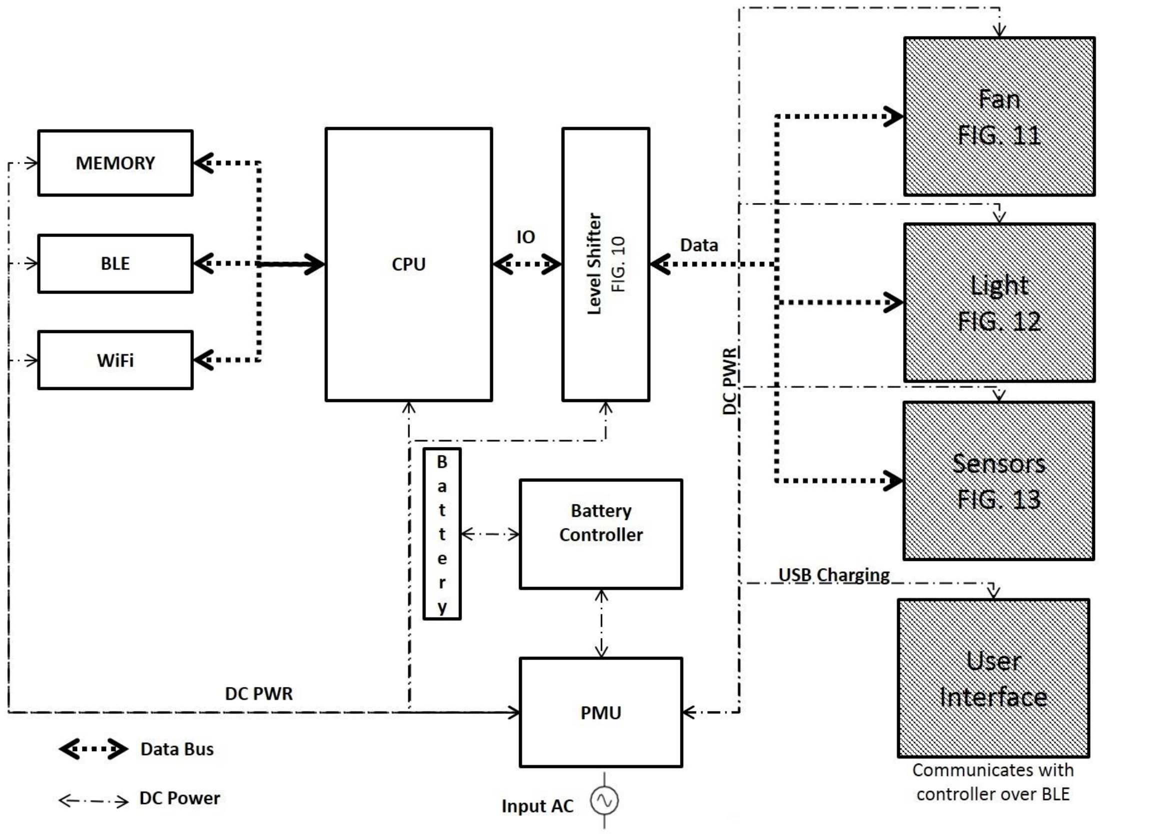


Figure A-4 1: SmartHub electrical functional block

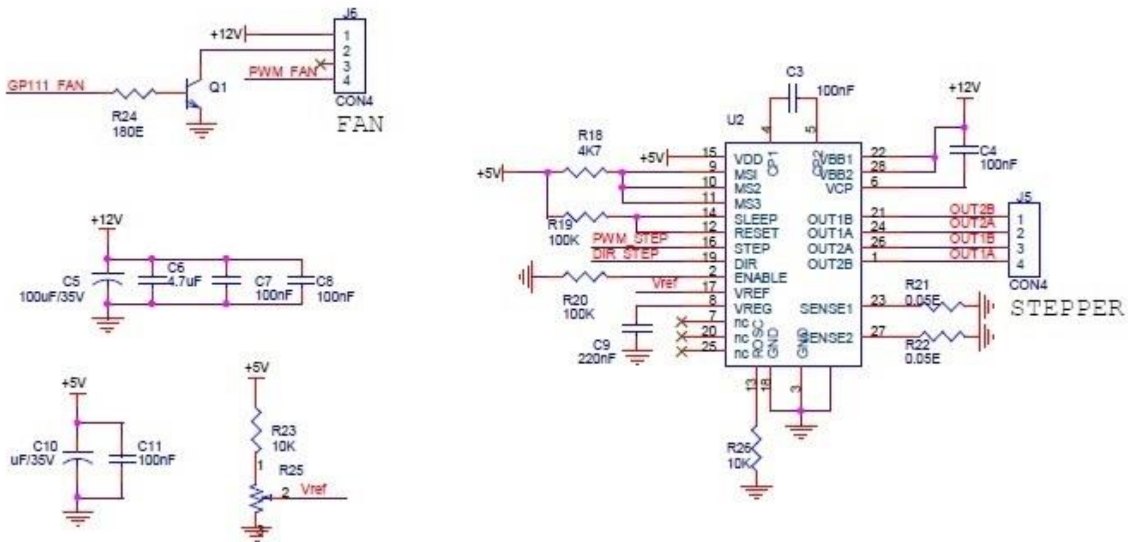


Figure A-4 2: Smarthub electrical line diagram-1

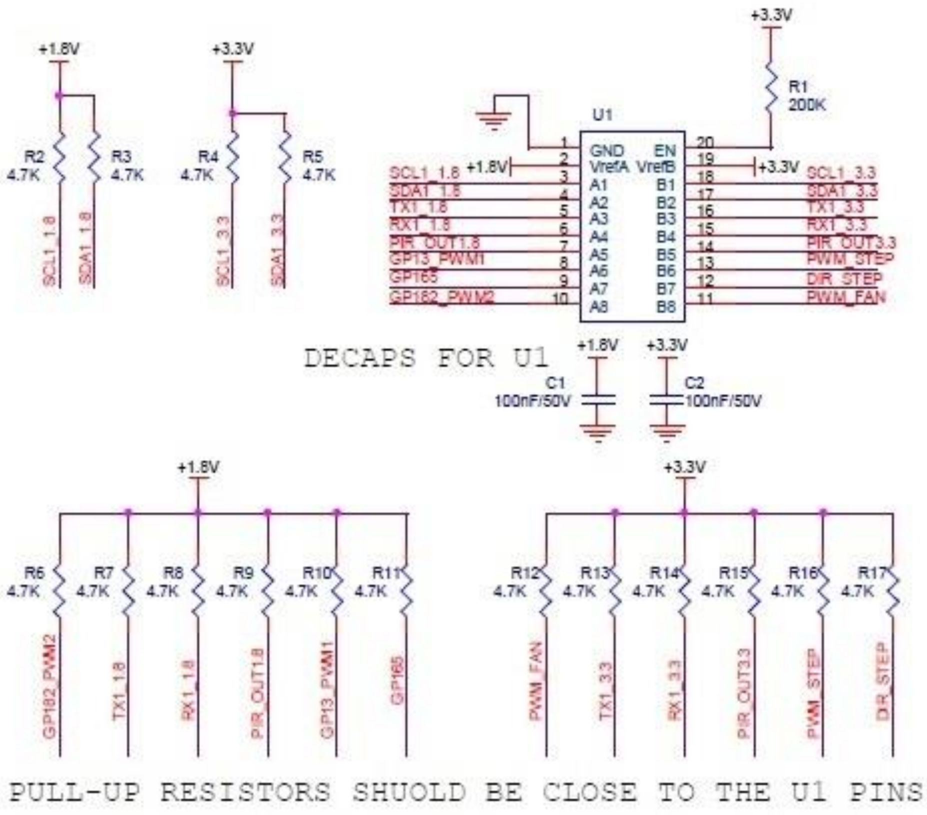


Figure A-4 3: Smarthub electrical line diagram-2

A-5 SmartStrip PCB design

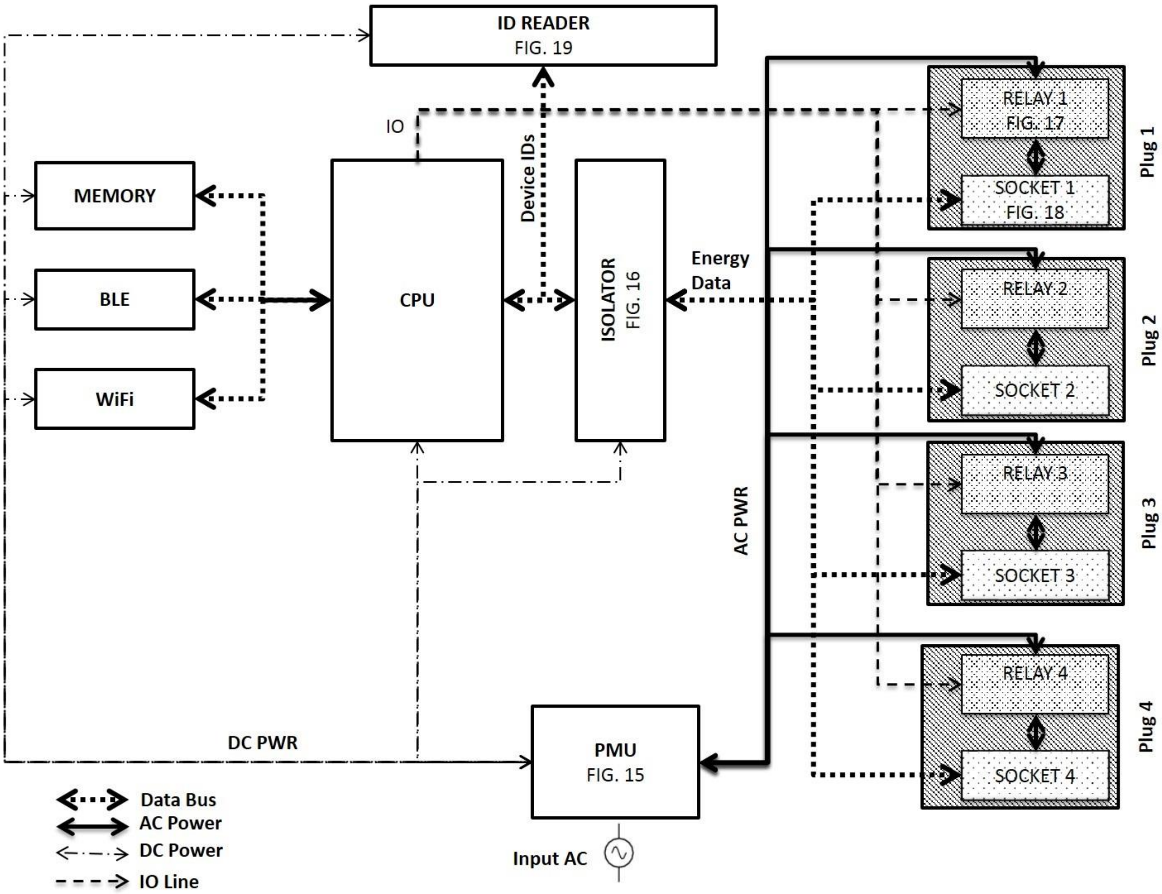


Figure A-5 1: Smartstrip functional block

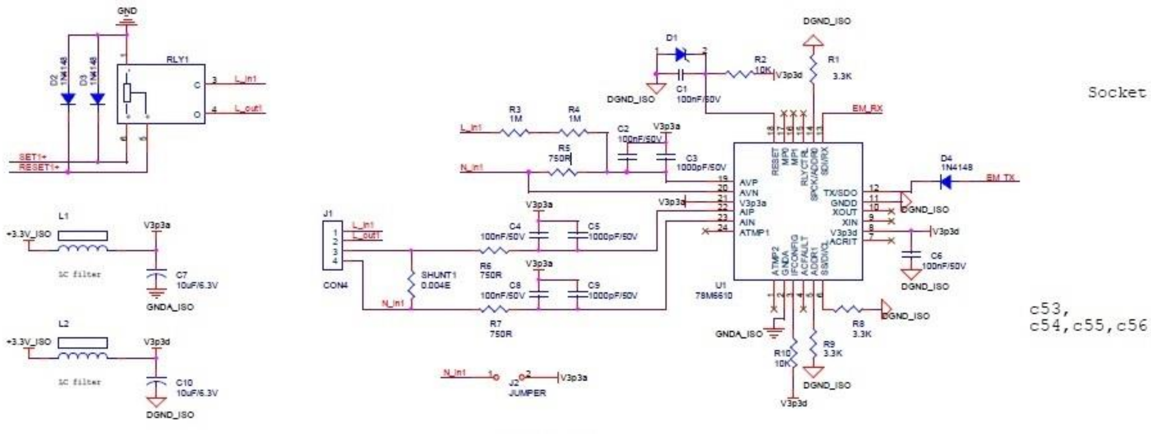


Figure A-5 2: Smartstrip electrical line diagram-1

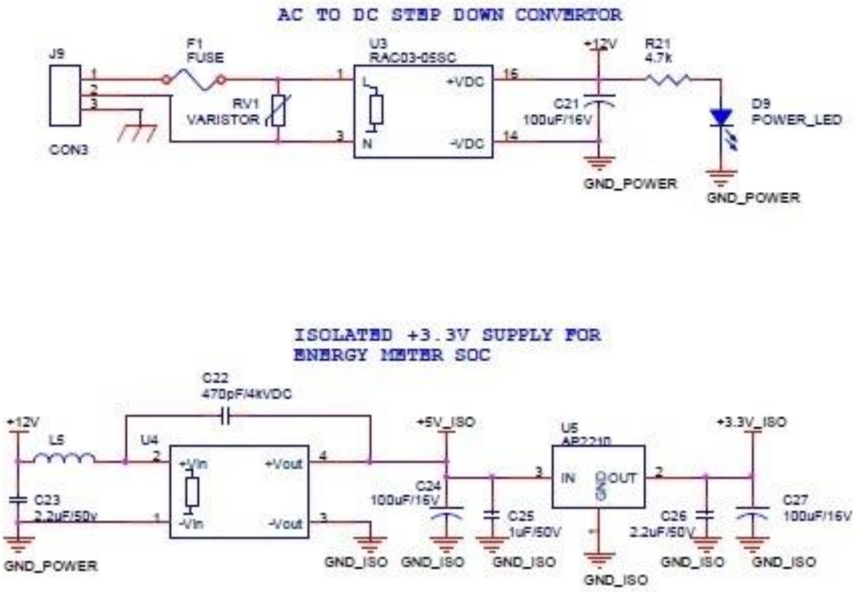


Figure A-5 3: : Smartstrip electrical line diagram-2

A-6 Source Code

The links to source code for the various modules available on GitHub is as follows:

a. **SmartHub**

- **Transactive Platform Code (Python Program)**: https://github.com/cbs-iiith/volttron/tree/dev-sam-phase4/applications/iiit
- **BACnet Server Code (C Program):** https://github.com/cbs-iiith/iSPACE/tree/dev-cbs-sam/edison/BACNETSmartHubSrv
- **UI Gateway (Node.js Program):** https://github.com/cbs-iiith/iSPACE/tree/dev-cbs-sam/edison/BLESmartHubSrv

b. **SmartStrip**

- **BACnet Server Code (C Program)**: https://github.com/cbs-iiith/iSPACE/tree/dev-cbs-sam/edison/BACNETSmartStripSrv

c. **UI mobile app (Android program)**

- https://github.com/cbs-iiith/iSPACE/tree/dev-cbs-sam/edison/SSAndroidApp

d. **Data extraction scripts**

- https://github.com/cbs-iiith/iSPACE/tree/dev-cbs-sam/data_extration

A-7 iSPACE BLE GATT Profiles

Table A-7-1: Basic information Bluetooth GATT profile

Type	**ID**	**UUID**	**Description**
Service	SERVICE_INFO	00001800-0000-1000-8000-00805f9b34fb	Device Information Service
	SERVICE_GENERIC_ATTRIBUTE	00001801-0000-1000-8000-00805f9b34fb	Generic Attribute Service
	CLIENT_CHARACTERISTIC_CONFIG	00002902-0000-1000-8000-00805f9b34fb	Client Char. Configuration Descriptor
Characteristic	DEVICE_NAME	00002a00-0000-1000-8000-00805f9b34fb	Device Name
	APPEARANCE	00002a01-0000-1000-8000-00805f9b34fb	Appearance
	SERVICE_CHANGED	00002a05-0000-1000-8000-00805f9b34fb	Service Changed
	MFG_NAME	00002a29-0000-1000-8000-00805f9b34fb	Manufacturer Name String
	MODEL_NO	00002a24-0000-1000-8000-00805f9b34fb	Model Number String
	SERIAL_NO	00002a25-0000-1000-8000-00805f9b34fb	Serial Number String

Table A-7-2: SmartHub Bluetooth GATT profile

Type	**ID**	**UUID**	**Description**
Service	SH_SERVICE_COMMONDATA	0000fd00-0000-1000-8000-00805f9b34fb	SmartHub Common Data Service
	SH_SERVICE_LED	0000fd01-0000-1000-8000-00805f9b34fb	SmartHub Led Service
	SH_SERVICE_FAN	0000fd02-0000-1000-8000-00805f9b34fb	SmartHub Fan Service
	SH_SERVICE_SENSORS	0000fd03-0000-1000-8000-00805f9b34fb	SmartHub Sensors Data Service

Type	ID	UUID	Description
Characteristic	SH_CHAR_SHNAME	0000fda0-0000-1000-8000-00805f9b34fb	SmartHub Name
	SH_CHAR_CURRENT_PP	0000fda1-0000-1000-8000-00805f9b34fb	Current Price Point
	SH_CHAR_LED_STATUS	0000fdb1-0000-1000-8000-00805f9b34fb	Led Status
	SH_CHAR_LED_LEVEL	0000fdb2-0000-1000-8000-00805f9b34fb	Led Level
	SH_CHAR_LED_THPP	0000fdb3-0000-1000-8000-00805f9b34fb	Led Threshold Price Point
	SH_CHAR_FAN_STATUS	0000fdc1-0000-1000-8000-00805f9b34fb	Fan Status
	SH_CHAR_FAN_LEVEL	0000fdc2-0000-1000-8000-00805f9b34fb	Fan Level
	SH_CHAR_FAN_THPP	0000fdc3-0000-1000-8000-00805f9b34fb	Fan Threshold Price Point
	SH_CHAR_FAN_SWING_STATUS	0000fdc4-0000-1000-8000-00805f9b34fb	Fan Swing Status
	SH_CHAR_SENSOR_LUX	0000fdd1-0000-1000-8000-00805f9b34fb	Lux
	SH_CHAR_SENSOR_TEMP	0000fdd2-0000-1000-8000-00805f9b34fb	Temperature
	SH_CHAR_SENSOR_RH	0000fdd3-0000-1000-8000-00805f9b34fb	Relative Humidity
	SH_CHAR_SENSOR_PIR	0000fdd4-0000-1000-8000-00805f9b34fb	PIR
	SH_CHAR_SENSOR_CO2	0000fdd5-0000-1000-8000-00805f9b34fb	CO2

Table A-7-3: SmartStrip Bluetooth GATT profile

Type	**ID**	**UUID**	**Description**
Service	SS_SERVICE_COMMONDATA	0000fc00-0000-1000-8000-00805f9b34fb	SmartStrip Common Data Service
	SS_SERVICE_PLUG1	0000fc01-0000-1000-8000-00805f9b34fb	SmartStrip Plug 1 Data Service
	SS_SERVICE_PLUG2	0000fc02-0000-1000-8000-00805f9b34fb	SmartStrip Plug 2 Data Service
	SS_SERVICE_PLUG3	0000fc03-0000-1000-8000-00805f9b34fb	SmartStrip Plug 3 Data Service
	SS_SERVICE_PLUG4	0000fc04-0000-1000-8000-00805f9b34fb	SmartStrip Plug 4 Data Service
Characteristic	SS_CHAR_SSNAME	0000fca0-0000-1000-8000-00805f9b34fb	SmartStrip Name
	SS_CHAR_CURRENT_PP	0000fca1-0000-1000-8000-00805f9b34fb	Current Price Point
	SS_CHAR_TAGID	0000fcb0-0000-1000-8000-00805f9b34fb	Tag ID
	SS_CHAR_THPP	0000fcb1-0000-1000-8000-00805f9b34fb	Threshold Price Point
	SS_CHAR_RELAY_STATE	0000fcb2-0000-1000-8000-00805f9b34fb	Relay State
	SS_CHAR_METERDATA_VOLT	0000fcb3-0000-1000-8000-00805f9b34fb	Voltage
	SS_CHAR_METERDATA_CURR	0000fcb4-0000-1000-8000-00805f9b34fb	Current
	SS_CHAR_METERDATA_APWR	0000fcb5-0000-1000-8000-00805f9b34fb	Active Power

A-8 List of Energy-Consuming Devices

Device	Total No. of Loads	Min Energy Demand (Wh)	Max Energy Demand (Wh)	Load Type	Price Function Type	Remarks
Ambient						
Central AC	2	800	1950	PID	Monotonic	
Light	2	80	140	Constant	Monotonic	The power consumed is a linear function of the setpoint, however a nominal power, 80 W, is consumed below 30% setpoint
SmartHub						
Fan	4	3	8	Constant	Monotonic	The power consumed is a linear function of the setpoint, however a nominal power, 3 W, is consumed below 30% setpoint
Light	4	3	10	Constant	Threshold	
Radiant Cubicle	4	50	300	PID	Monotonic	
SmartStrip						
Plug1	4	0	30	Variable	Threshold	Mobile Phone Charging
Plug2	4	0	150	Variable	Threshold	Secondary LED Monitor
Plug3	4	0	150	Variable	Threshold	Laptop
Plug4	4	0	50	Variable	Threshold	SmartHub
Total	**32**					

A-9 Sensor's calibration

The following tables provide sensors calibration for the pt500 temperature sensors used for the radiant cubicle.

Date: 19.05.2018 & 21.05.2018

Sensor details | pt 500 | 2WR

S.no.	Set points	Start time .	stabilized time	1st Reading Time	R91			B91			R92			B92		
					1	2	3	1	2	3	1	2	3	1	2	3
1	100 deg.	11:30	12:20	13:00	99.86	99.86	99.86	99.95	99.95	99.95	100.72	100.70	100.68	100.06	100.06	100.05
2	50 deg.	13:40	14:00	14:00	50.19	50.18	50.18	50.21	50.20	50.21	50.43	50.42	50.41	50.18	50.18	50.18
3	35 deg.	14:35	14:50	14:50	35.10	35.10	35.10	35.20	35.19	35.19	35.20	35.20	35.20	35.37	35.37	35.35
4	30 deg.	15:25	15:50	15:40	30.13	30.12	30.12	30.15	30.15	30.16	30.21	30.21	30.21	30.22	30.22	30.22
5	25 deg.	16:10	16:30	16:30	25.12	25.12	25.13	25.24	25.24	25.24	25.19	25.19	25.19	25.23	25.23	25.24
6	20 deg.	11:00	11:40	11:45	20.19	20.18	20.18	20.22	20.22	20.22	20.26	20.26	20.25	20.28	20.29	20.29
7	15 deg.	12:20	12:40	12:42	15.20	15.20	15.20	15.21	15.21	15.20	15.27	15.27	15.27	15.25	15.25	15.25
8	0 deg.	13:20	14:05	14:07	0.26	0.26	0.26	0.28	0.28	0.28	0.31	0.31	0.32	0.33	0.33	0.33

S.no.	Set points	Start time .	stabilized time		R93			B93			R94			B94		
					1	2	3	1	2	3	1	2	3	1	2	3
1	100 deg.				99.98	99.98	99.98	100.13	100.13	100.13	100.07	100.04	100.03	100.54	100.39	100.41
2	50 deg.				50.16	50.17	50.17	50.37	50.37	50.37	50.31	50.31	50.30	50.21	50.21	50.21
3	35 deg.				35.42	35.42	35.43	35.20	35.21	35.20	35.20	35.20	35.20	35.22	35.21	35.21
4	30 deg.				30.19	30.18	30.19	30.22	30.22	30.22	30.22	30.23	30.22	30.30	30.29	30.29
5	25 deg.				25.20	25.20	25.19	25.22	25.22	25.22	25.19	25.19	25.19	25.22	25.22	25.21
6	20 deg.				20.24	20.25	20.25	20.24	20.24	20.23	20.24	20.24	20.24	20.26	20.26	20.26
7	15 deg.				15.26	15.25	15.26	15.34	15.36	15.36	15.23	15.23	15.23	15.30	15.29	15.30
8	0 deg.				0.31	0.31	0.30	0.34	0.33	0.34	0.29	0.29	0.29	0.35	0.34	0.35

	Tb	T_{11}	T_{11}	T_{11}	T_{1a}	STDEV	$T_b - T_{1a}$				T_{21}	T_{21}	T_{21}	T_{2a}	STDEV	$T_b - T_{1a}$			example diff btwn T1 35 & T2 20	
100	100	99.86	99.86	99.86	99.86	0.000	0.140		100	100	99.95	99.95	99.95	99.95	0.0000	0.050	0.09			
50	50	50.19	50.18	50.18	50.18	0.006	-0.183		50	50	50.21	50.20	50.21	50.21	0.0058	-0.207	0.02		$T_b(35)-T_b(20)$	$T_{1a}(35)-T_{2a}(20)$
35	35	35.10	35.10	35.10	35.10	0.000	-0.100		35	35	35.20	35.19	35.19	35.19	0.0058	-0.193	0.09		15	14.88
30	30	30.13	30.12	30.12	30.12	0.006	-0.123		30	30	30.15	30.15	30.16	30.15	0.0058	-0.153	0.03			0.12
25	25	25.12	25.12	25.13	25.12	0.006	-0.123		25	25	25.24	25.24	25.24	25.24	0.0000	-0.240	0.12			
20	20	20.19	20.18	20.18	20.18	0.006	-0.183		20	20	20.22	20.22	20.22	20.22	0.0000	-0.220	0.04			
15	15	15.20	15.20	15.20	15.20	0.000	-0.200	-0.146	15	15	15.21	15.21	15.20	15.21	0.0058	-0.207	0.01	0.057		
0	0	0.26	0.26	0.26	0.26	0.000	-0.260		0	0	0.28	0.28	0.28	0.28	0.0000	-0.280	0.02			
100	100	100.72	100.70	100.68	100.70	0.020	-0.700		100	100	100.06	100.06	100.05	100.06	0.0058	-0.057	-0.64			
50	50	50.43	50.42	50.41	50.42	0.010	-0.420		50	50	50.18	50.18	50.18	50.18	0.0000	-0.180	-0.24		$T_b(35)-T_b(20)$	$T_{1a}(35)-T_{2a}(20)$
35	35	35.20	35.20	35.20	35.20	0.000	-0.200		35	35	35.37	35.37	35.35	35.36	0.0115	-0.363	0.16		15	14.91
30	30	30.21	30.21	30.21	30.21	0.000	-0.210		30	30	30.22	30.22	30.22	30.22	0.0000	-0.220	0.01			0.09
25	25	25.19	25.19	25.19	25.19	0.000	-0.190		25	25	25.23	25.23	25.24	25.23	0.0058	-0.233	0.04			
20	20	20.26	20.26	20.25	20.26	0.006	-0.257		20	20	20.28	20.29	20.29	20.29	0.0058	-0.287	0.03			
15	15	15.27	15.27	15.27	15.27	0.000	-0.270	-0.225	15	15	15.25	15.25	15.25	15.25	0.0000	-0.250	-0.02	0.045		
0	0	0.31	0.31	0.32	0.31	0.006	-0.313		0	0	0.33	0.33	0.33	0.33	0.0000	-0.330	0.02			
100	100	99.98	99.98	99.98	99.98	0.000	0.020		100	100	100.13	100.13	100.13	100.13	0.0000	-0.130	0.15			
50	50	50.16	50.17	50.17	50.17	0.006	-0.167		50	50	50.37	50.37	50.37	50.37	0.0000	-0.370	0.20		$T_b(35)-T_b(20)$	$T_{1a}(35)-T_{2a}(20)$
35	35	35.42	35.42	35.43	35.42	0.006	-0.423		35	35	35.20	35.21	35.20	35.20	0.0058	-0.203	-0.22		15	15.19
30	30	30.19	30.18	30.19	30.19	0.006	-0.187		30	30	30.22	30.22	30.22	30.22	0.0000	-0.220	0.03			-0.19
25	25	25.20	25.20	25.19	25.20	0.006	-0.197		25	25	25.22	25.22	25.22	25.22	0.0000	-0.220	0.02			
20	20	20.24	20.25	20.25	20.25	0.006	-0.247		20	20	20.24	20.24	20.23	20.24	0.0058	-0.237	-0.01			
15	15	15.26	15.25	15.26	15.26	0.006	-0.257	-0.262	15	15	15.34	15.36	15.36	15.35	0.0115	-0.353	0.10	-0.015		
0	0	0.31	0.31	0.30	0.31	0.006	-0.307		0	0	0.34	0.33	0.34	0.34	0.0058	-0.337	0.03			
100	100	100.07	100.04	100.03	100.05	0.021	-0.047		100	100	100.54	100.39	100.41	100.45	0.0814	-0.447	0.40			
50	50	50.31	50.31	50.30	50.31	0.006	-0.307		50	50	50.21	50.21	50.21	50.21	0.0000	-0.210	-0.10		$T_b(35)-T_b(20)$	$T_{1a}(35)-T_{2a}(20)$
35	35	35.20	35.20	35.20	35.20	0.000	-0.200		35	35	35.22	35.21	35.21	35.21	0.0058	-0.213	0.01		15	14.94
30	30	30.22	30.23	30.22	30.22	0.006	-0.223		30	30	30.30	30.29	30.29	30.29	0.0058	-0.293	0.07			0.06
25	25	25.19	25.19	25.19	25.19	0.000	-0.190		25	25	25.22	25.22	25.21	25.22	0.0058	-0.217	0.03			
20	20	20.24	20.24	20.24	20.24	0.000	-0.240		20	20	20.26	20.26	20.26	20.26	0.0000	-0.260	0.02			
15	15	15.23	15.23	15.23	15.23	0.000	-0.230	-0.217	15	15	15.30	15.29	15.30	15.30	0.0058	-0.297	0.07	0.039		
0	0	0.29	0.29	0.29	0.29	0.000	-0.290		0	0	0.35	0.34	0.35	0.35	0.0058	-0.347	0.06			