Chapter 12
Supporting Decision Making for Large-Scale IoTs: Trading Accuracy with Computational Complexity

Kostas Siozios, Panayiotis Danassis, Nikolaos Zompakis, Christos Korkas, Elias Kosmatopoulos, and Dimitrios Soudris

12.1 Introduction

During the last years, there are tremendous improvements in the domain of embedded devices. To begin with, the new process technologies enable the underline hardware to become smaller, cheaper, and more powerful. This trend in conjunction to the improvements at networking infrastructure enables the majority of the devices to be extended with communication capabilities. Hence, embedded devices nowadays are able to connect, interact, and cooperate with their surrounding environment. This new platform paradigm, also known as “Internet of Things” (IoT), is as a network of objects (or things) capable of detecting and communicating information between each other.

Defining things and recognizing what a particular thing is and what it represents in the context of IoT requires a careful analysis of what philosophers, such as the Aristotle and Philoponus have to say and how their philosophical thoughts can transcend into the near future. Specifically, in the work “The Categories” Aristotle gives strikingly general and exhaustive account of things that are beings. According to this opinion, beings include substance, quantity, quality, as well as relation among others. Hence, from the philosophical point of view, the word “things” is

K. Siozios (✉)
School of Physics, Aristotle University of Thessaloniki, Thessaloniki, Greece
e-mail: ksiop@microlab.ntua.gr

P. Danassis • N. Zompakis • D. Soudris
School of ECE, National Technical University of Athens, Athens, Greece
e-mail: panosd@microlab.ntua.gr; nzompakis@microlab.ntua.gr; dsoudris@microlab.ntua.gr

C. Korkas • E. Kosmatopoulos
Department of ECE, Democritus University of Thrace, Xanthi, Greece
e-mail: chriskorkas@iti.gr; kosmatop@iti.gr

G. Keramidas et al. (eds.), *Components and Services for IoT Platforms*,
DOI 10.1007/978-3-319-42304-3_12

not restricted to material things but can apply also to virtual things and the events that are connected to "things." In the context of IoT, a "thing" might be defined as a real (physical) or digital (virtual) entity, which is capable of being uniquely identified, and that exists and moves in space and time.

The challenge of integrating embedded computing and physical processes with feedback loops, where physical processes affect computations and vice versa has been recognized for some time, allowing applications with enormous societal impact and economic benefit to be developed by harnessing these capabilities across both space and time domains. Such a pooling of system's resources and capabilities together to create a new, more complex system which offers more functionality and performance than simply the sum of the constituent sub-systems. In the IoT paradigm, numerous objects that surround us will be on the network in one form or another. This trend is inline to the projection that in the near future it is expected that computing and communication capabilities will be embedded in all types of objects and structures in the physical environment [1]. To do so, flexible yet efficient protocols, architectures, and technologies are absolutely necessary, in which information and communication systems are transparently embedded in the enrolment around us.

While technology and platform capabilities ought to be equally important in order to design next-generation of smart systems, it is also crucial to manage the system's complexity under the constraint posed by the large amounts of data that these systems will produce. According to a study from Cisco Systems, there will be as many as 50 billion embedded systems and other portable devices connected to the internet by 2020. Because these systems are so inexpensive and the networks so pervasive, they could provide a wealth of data that industry could use to monitor and improve operations. For instance, by 2020, the digital universe will reach 44 zettabytes—a tenfold increase from 2013 [2]. On top of this, analysts will need deep knowledge of the specific target application domains to ensure they incorporate the right data to generate useful insights. Although new methodologies, services, and design technologies are absolutely necessary to address this challenge, the trend nowadays is to enable smart things and their services to be fully integrated in the developed systems by reusing and adapting technologies and patterns commonly used for traditional local-/wide-area networks. Specifically, the IoT is developing over time by a way of co-evolution, with technology, applications, and stakeholders' understanding of the implications driving each other.

The integration of physical processes and computing devices is not new. Embedded systems have been in place for a long time and these systems often combine physical processes (e.g., through digital/analog sensors) with computing. However, the core differentiator between an IoT and either a typical control system, or an embedded system, is the communication feature among system's components, which adds (re-)configurability and scalability, allowing instrumenting the physical world with pervasive networks of sensor-rich embedded computation [3]. The goal of an IoT architecture is to get maximum value out of a distributed large system by understanding how each of the individual (sub-)systems work, interface and are used. This trend is also supported by the continuation of Moore's law, which imposes that the cost of a single embedded computer equipped with sensing, processing, and communication capabilities drops towards zero [4]. Thus, it will be economically feasible to densely deploy networks with very large quantities of sensor readings from the physical world, compute quantities, and take decisions out of them. Such a very dense networks offer a better resolution of the physical world and therefore a better capability of detecting the occurrence of an event; this is of paramount importance for a number of foreseeable applications.

Apart from the technology-oriented parameters that affect the efficiency and/or the flexibility of an IoT-based system, the supporting tools are also crucial for deriving an optimum solution. The current solutions targeting to support the development of these platforms rely mainly on stand-alone tools that tackle distinct aspects of the system's development. Consequently, the constraint propagation among the employed tool-sets is the current viable way to enable the design of more complex platforms. Although this concept seems straightforward and promising, it relies on the fundamental premise that models are freely interchangeable amongst tool vendors and have interoperability amongst them. In other words, this imposes that models can be written, or obtained from other vendors, while it is known a priori that they will be accepted by any vendor tool for performing different steps of system's prototyping (e.g., architecture/topology analysis, simulation, implementation, etc.). On contrast to this "ideal" approach, the existing flows rarely support either model interoperability or independence between model and software tools. Also, due to the problem's complexity, the existing "constraint propagation" design technique will not be a viable approach for designing large-scale IoT platforms. Towards this direction, and as research and industry pushes for efficient designs, novel frameworks that tackle the entire system's complexity are of high importance [5].

In accordance with this trend, throughout this chapter we introduce a low-complexity decision-making mechanism targeting to IoT-based systems. For evaluation purposes, the efficiency of introduced solution will be demonstrated with the usage of a smart thermostat usecase that supports the building's cooling/heating control.

12.2 The Smart Thermostat Usecase

The studied usecase concerns the climate control of buildings equipped with their own renewable energy generation elements (e.g., photovoltaic arrays, wind turbines, geothermal energy, etc.) along with a collection of automatic control elements for affecting the building thermal characteristics e.g., automatically control the heating, ventilation, and air conditioning (HVAC) system set-points. Such a solution is part of home automation—a field that refers to the use of computer and information technology to control home appliances and features (such as heating and cooling). Depending on the building's size, the developed systems can range from simple remote control to complex computer/micro-controller-based networks with varying degrees of intelligence and automation, similar to the case depicted in Fig. 12.1. The popularity of these solutions has been increasing greatly in recent years due to much

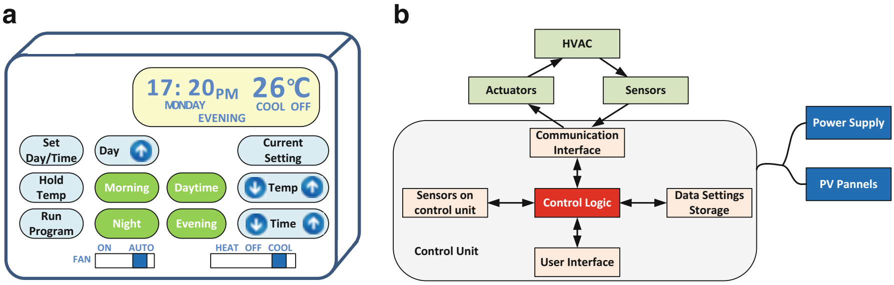

Fig. 12.1 The employed smart thermostat **(a)** component's view and **(b)** the functional blocks

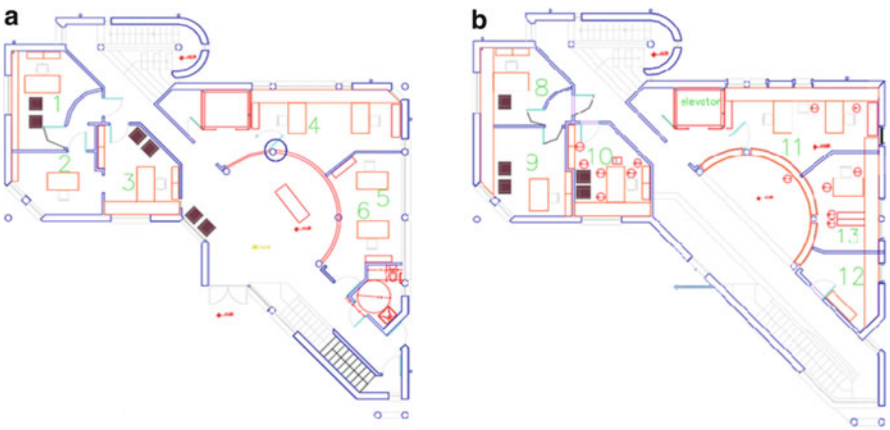

Fig. 12.2 Floorplans for the (**a**) ground and (**b**) first floor of our smart building

higher affordability and simplicity through smartphone and tablet connectivity, while the concept of the IoT has tied in closely with the popularization of home automation.

To test and evaluate the introduced decision-making framework, a building block located in Chania, Greece, will serve as the employed usecase. The floorplan of this building is depicted schematically in Fig. 12.2. The main reason for choosing this building is the fact that the building is equipped literally with multiple renewable energy generation elements and energy/thermal influencing elements that may found in a real-life building. Moreover, the overall building control infrastructure comprises a quite complex hierarchical system with different decision-making elements affecting the building thermal and energy-consuming performance at different levels of abstraction and in a quite complex manner. Furthermore, it has to be emphasized the fact that the building contains a large number of rooms and offices with totally different characteristics and purposes (laboratories, office rooms, conference spaces, data centre rooms, etc.). Finally, by studying the efficiency of this building with real weather data affecting a whole year (as they were provided by National Renewable Energy Laboratory [6]), depicts that our solution is subject to severe and abrupt weather as well as occupant behavior changes.

In order to guide the aggressiveness of introduced framework, the employed building uses a number of sensors for monitoring temperature values (both inside and outside the building), as well as the variation of sunshine and humidity. For our experimentation we assume (without affecting the generality of introduced solution) that these values are acquired once per 10 min, as the weather data is not expected to modified considerably within this time period. The analysis discussed at the rest of this manuscript considers the problem of operating air-conditioners during the summer period (June, July, and August), in order to cool-climate the rooms. The objective is to define a solution that takes into consideration both the energy efficiency and the user comfort satisfying level with the minimal computational complexity, as compared to existing state-of-the-art control solvers for similar problem. Furthermore, instead of relevant approaches which mainly rely on statistical data, the proposed solution does not consider weather forecasts; thus, the efficient addressing of decision-making problem becomes far more challenging.

12.2.1 System Modeling

In order to describe in more thoroughly the system's architecture, Fig. 12.3 presents in UML form the main functionalities performed by the smart thermostats regarding the general form of the employed usecase. Specifically, starting by collecting a number of weather-related data (e.g., temperature, humidity, and radiation), as they were acquired by the weather station, it is possible to analyze the efficiency of different thermostat configurations. The results are fed as input to the scenario analysis in order to compute the optimum scenario set in a season basis (winter, spring, summer, and autumn). Then, the employed controller implements the desired policies in order to maximize user's comfort with the minimum energy cost.

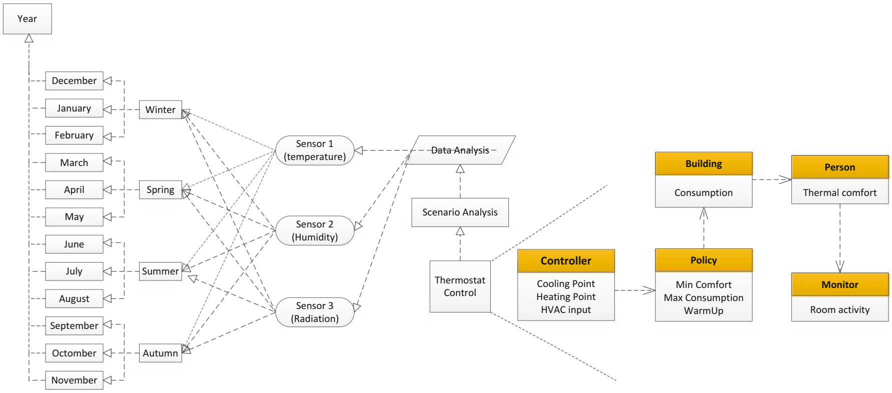

Fig. 12.3 UML encoding for the employed system

During this phase, alternative techniques for further energy savings (e.g., building warm-up phase, switch-off heating/cooling in case there is no motion detection) are also applicable. However, these techniques are beyond the scopes of the analysis discussed throughout this chapter, as we focus solely at the decision making.

12.3 Challenges and Motivation

This chapter describes in detail the proposed efficient control system that is easy- and inexpensive-to-deploy in everyday buildings in order to support the task of decision making for HVAC systems: no expensive infrastructure or modeling tools and effort are required. Instead of the typical adopted procedure for developing decision-making systems for building climate control, which relies on modeling initially the building dynamics using one of the existing building modeling tools (e.g., EnergyPlus [7], Modelica [8], etc.) and then developing a model-based control using the model for the building dynamics, our approach relies on a different strategy. The objective of our solution is to provide thermal comfort and acceptable indoor air quality with the minimum possible energy cost. More thoroughly, the problem at hand is a quite challenging problem where the control system attempts to exploit "as much as it can" the renewable energy so as to reduce the demand for non-renewable energy (coming from the grid) or during time-slots of low-cost tariffs, while maintaining user comfort (i.e., making sure that the building occupants are *satisfied* with the in-building temperature and other thermal conditions). This will be done without requiring the deployment of an "expensive," elaborate, and complete sensor infrastructure, a prerequisite for the deployment of state-of-the-art building climate control systems—each of the constituent systems is provided only with information about its own state and energy costs.

Despite the significant progress made in optimal nonlinear control theory [9, 10] the existing methods are not, in general, applicable to large-scale systems because of the computational difficulties associated with the solution of the Hamilton–Jacobi partial differential equations. Similarly, model predictive control for nonlinear systems, a control approach which has been extensively analyzed and successfully applied in industrial plants during the latest decades [11, 12], faces also dimensionality issues: in most cases, predictive control computations for nonlinear systems amount to numerically solving on line a non-convex high-dimensional mathematical programming problem, whose solution may require a quite formidable computational burden if on line solutions are required. Another family of approaches employ optimization-based schemes to calculate the controller parameters [13]. These approaches require analytical calculation of Jacobian and Hessian matrices, which in large-scale systems is a very time-consuming process. Existing simulation-based approaches are not able to efficiently handle systems of large-scale nature as this requires solving a complex optimization problem with hundreds or thousands of states and parameters [14, 15].

The previously mentioned problem imposed that novel techniques able to support efficient decision making with almost negligible computational and/or storage complexity are of high importance. These lightweight solutions would be ideal to support tasks related to control of IoT-based systems, as these systems usually rely on battery-based (low-performance) embedded processors. Additionally, the feature of hierarchical decision-making is of similar importance because such an approach further reduced the problem's complexity. Note that the absence of developing lightweight solutions (able to be executed onto embedded platforms) for supporting large-scale system's decision making is not due to neglect, but rather due to its difficulty. Also, as we have already highlighted, this problem becomes far more challenging in case the decision making has to be made under run-time (or real-time) constraints. In such a case, usually a compromise between the desired accuracy and the processing overhead is performed.

12.4 Support Decision Making with the Fuzzy Inference Systems

The task of determining the temperature set-points for each thermal zone can be accomplished by a mechanism that relies on two competing fuzzy inference systems (FIS). The first one (*availability FIS*) assesses the "availability" of energy ($A^E(t)$), while the second one (*demand FIS*) determines the predicted "demand" for energy consumption ($D^E(t)$).

The "availability" of energy ($A^E(t)$) is derived based on the weather data, the available funds for purchasing energy (AF) and the current market's energy trading rate ($P(t)$). Intuitively it is a metric that determines how affordable is to purchase energy from the grid. On the other hand, the "demand" for energy consumption ($D^E(t)$) is determined according to the occupants' thermal comfort. Intuitively it reflects the occupants' need to spent more energy depending on their satisfaction. The aforementioned FIS compete with each other until they reach an approximate equilibrium $A^E(t) \simeq D^E(t)$, i.e., being able to afford to spend the required energy (through the appropriate modification at the thermostat's set-point) while taking into account restrictions posed by the occupants' thermal comfort violation. To that end we start by determining an initial temperature set-point[1] and use it to compute the predicted occupants' thermal comfort based on a model for general thermal satisfaction called predicted mean vote (PMV).[2] By modifying the thermostat's set-point we change the PMV per room and consequently the demand for energy consumption ($D^E(t)$). Normally, improving the PMV comes

[1]The initial set-point can be a static value for the whole year (e.g., 23 °C), a plethora of predetermined values depending on the season, or a set-point derived by another engine (e.g., by using an artificial neural network).

[2]The PMV model is an index that provides the average thermal sensation according to the ASHRAE thermal sensation scale [−3, +3]. It was developed by Fanger in the 1970s and stands among the most recognized thermal comfort models.

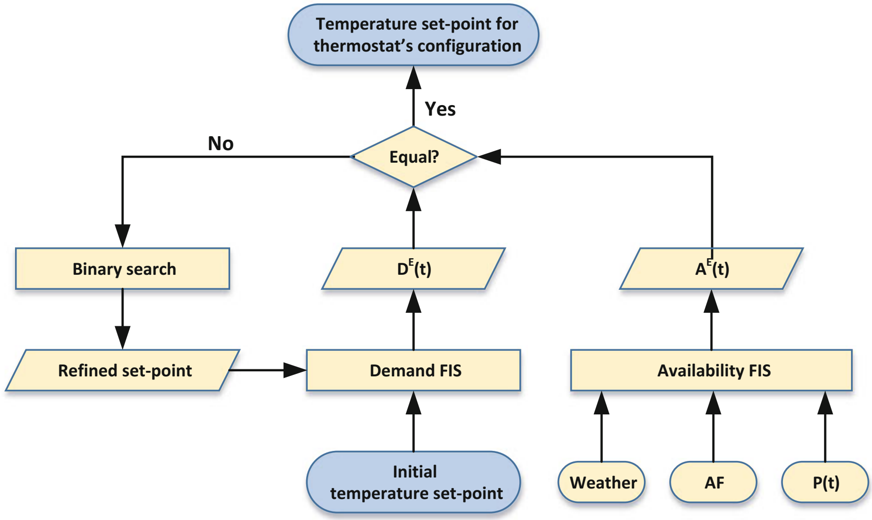

Fig. 12.4 Proposed methodology regarding the competing FIS

with a higher cost in energy consumption and that trade-off is the basis at which our two FIS compete on. For this purpose, the proposed mechanism performs a binary search[3] within the accepted temperature range [18–28 °C] and appropriately configures the thermostat's operational set-point to the temperature that results to the closest equilibrium. The aforementioned process is depicted in Fig. 12.4. Note that the "availability" of energy $A^E(t)$ remains constant during a specific time-step, since this value is independent of the examined set-point.

In the subsequent section we provide additional details on the architecture of the employed FIS.

12.4.1 Fuzzy Inference System's Architecture

An FIS uses *fuzzy* logic and sets to formulate a mapping between an input space to an output space. The term *fuzzy* refers to the fact that the logic involved extends the classical boolean logic in order to handle the concept of partial truth. Classical logic assumes the "Principle of Bivalence" which states that every sentence can be either true or false. In contrast, humans think using "degrees of truth" and linguistic terms

[3]Note that the binary search algorithm takes a monotonic function as input. In our case, we ensure that our function is monotonic by limiting our search space to a specific range [18–28 °C]. That way we guarantee that increasing the temperature during the winter and decreasing it during the summer result in an improvement in the occupants' thermal comfort.

such as “a little” or “too much” instead of absolute values. In order to deal with these “issues of vagueness,” fuzzy logic assigns any real number between 0 and 1 as the truth value of a statement. In essence, in fuzzy logic the truth of any statement becomes a matter of degree, making the FIS a convenient way to create a reasonable model for a complex system that is tolerant to imprecise data.

An FIS consists of a rule (or knowledge) base and a reasoning mechanism called fuzzy inference engine. The rule base is a collection of *if-then* rules provided by experts that use fuzzy sets in the hypothesis and conclusion part. The inference engine combines these rules using fuzzy reasoning and produces a mapping from the input to the output space. Additionally, there is a fuzzification mechanism that transforms the input space into fuzzy sets and a defuzzification mechanism that performs the reverse procedure, i.e., transforms the fuzzy set obtained by the inference engine into crisp values on the range of the output space. The aforementioned process is depicted in Fig. 12.5.

A fuzzy set is defined by a function that assigns to each entity in its domain a value between 0 and 1, representing the entity’s degree of membership in the set. Such a function is called membership function (MF). The membership functions associated with the PMV input variable for the *demand FIS* are depicted in Fig. 12.6. More thoroughly, if, for example, the computed PMV is -0.15, then the temperature is mostly excellent but there will be some people that will feel a bit cold. In essence these functions represent the vagueness which is the norm rather than the exception in real life and constitute the basis for converting crisp values to fuzzy sets.

The principal component of each FIS, though, is its rule base, which is used to formulate the expert’s knowledge. Evaluating a fuzzy rule involves the fuzzification of the premise, the application of any fuzzy operators and finally the implementation of the implication operator that gives the consequent of the rule. The fundamental difference compared to classical boolean rules is that fuzzy rules can be partially applied, depending on the degree of membership of each premise of the rule. Aggregating the consequent part of all the rules creates the final output fuzzy

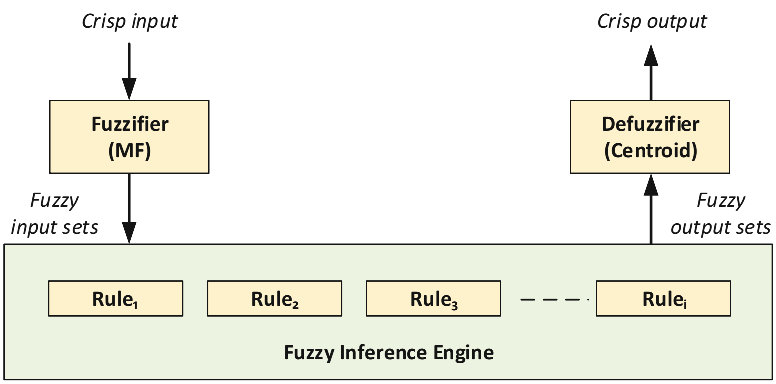

Fig. 12.5 Architectural organization of the employed FIS

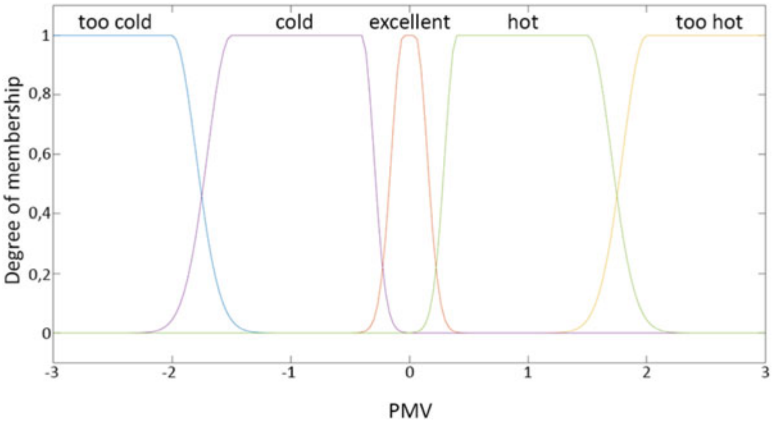

Fig. 12.6 Membership functions regarding the PMV input of the *demand FIS*

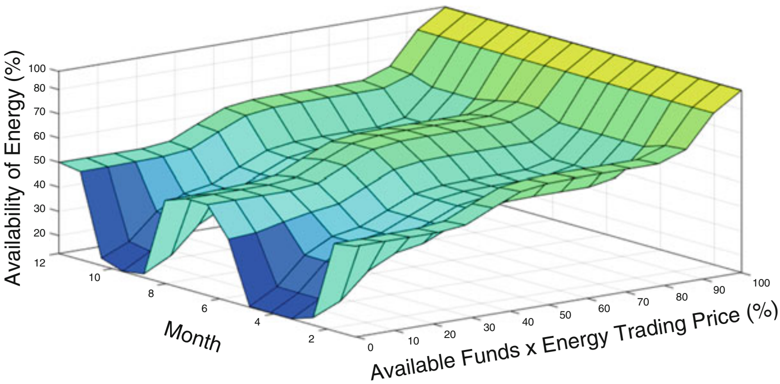

Fig. 12.7 Graphical representation of the input/output mapping regarding the *availability FIS*

set which, after the defuzzification process, provides the final desired output. The benefit of such approach is its ability to encode apparently complex problems using just a few simple fuzzy rules, in contrast to exact approaches which usually require much greater effort.

To help visualize the aforementioned process, Fig. 12.7 depicts the mapping from the input to the output space regarding the *availability FIS*. The horizontal axes represent the inputs of the FIS while the vertical axes represent the output (availability of energy $A^E(t)$ (%)). The depicted surface incorporates the entire rule base of the *availability FIS*. The true elegance of an FIS is its capacity to encode intuitive linguistic terms into applicable information. For instance, as you can see from the plot, during the colder (Dec.–Feb.) and hotter (Jun.–Aug.) months of the year the availability of energy $A^E(t)$ is high in order to cope with the extreme weather conditions, while when we are low on funds (AF) or when the trading rate for purchasing energy ($P(t)$) is high then the availability of energy $A^E(t)$ is low in order to reflect our inability of purchasing energy from the grid.

An analogous process is performed by the *demand FIS*. The concept in this case is that the more dissatisfied the occupants are, the more "funds" they should have in their disposal in order to be able to afford to purchase energy. As you can see, in contrast to alternative approaches, ours is a human-centric one, since it configures its aggressiveness according to the occupants' requirements.

To summarize, the *availability FIS* assesses how affordable is to purchase energy according to the current rate, weather, and the available funds, while the *demand FIS* determines the amount of energy needed according to the occupants' thermal comfort. When these two FIS come to an agreement ($A^E(t) \simeq D^E(t)$), then we modify the thermostat's set-point to the agreed upon temperature. Hence, using intuitive and linguistic terms, we manage to implementing a flexible yet efficient decision-making mechanism for smart thermostats.

12.5 Communication Links

The communication infrastructure plays a key role at the IoT platform, since it provides the necessary information transfer gathered by the sender nodes and processed by local embedded processing nodes to the destination (e.g., actuators or next level of processing cores). Since different application domains impose variations in terms of data transferring problem, various protocols have been proposed at the context of IoT. Table 12.1 summarizes some representative approaches and provides a qualitative comparison in terms of various supported features.

Typically, such communication links can be classified either as constrained, or unconstrained networks. Specifically, an unconstrained network is characterized by high-speed communication links (e.g., wired network). On the other hand, constrained networks support relatively low transfer rates (typically smaller than 1 Mbps) and large latencies, as offered by, e.g., IEEE 802.15.4 (ZigBee) protocol. The differences between these two concern the hardware capabilities: in the former category the objective is to provide the better services by exploiting all the hardware capabilities offered; in the latter approach the devices are limited by low energy availability, since they are usually battery powered, bandwidth as low as 300 Kbps and computational power as low as few MHz. Due to the power saving constraints posed by the employed usecase, the communication link studied throughout this chapter (similar to the majority relevant implementations) relies on unconstrained network infrastructure.

Another crucial parameter for selecting the optimum communication infrastructure relies on the topology of the links (e.g., mesh, star, and point-to-point), as well as the maximum distance between nodes. Various constraints that are posed by the target application domain usually introduce guidelines about the selection of optimum communication infrastructure. Regarding the network, it is realized

Table 12.1 Overview of available communication protocols

	NFC	RFID	Bluetooth	Bluetooth LE	WiFi	Zigbee
Standard	ECMA-340 and ISO/IEC 18092	ISO 15693 and ISO 14443	IEEE 802.15.1	IEEE 802.15.6	IEEE 802.11a/b/g/n	IEEE 802.15.4
Network	PAN	PAN	PAN	PAN	LAN	LAN
Topology	P2P	P2P	Star	Star	Star	Mesh, star, tree
RF frequency	13.56 MHz	13.56 MHz	2.4 GHz	2.4 GHz	2.4/5.8 GHz	868/915 MHz 2.4 GHz
Power	Very low	Very low	Low	Very low	High	Very low
Throughput	106–424 Kbps	400 Kbps	723 Kbps	1 Mbps	11–105 MBps	250 Kbps
Range	<10 cm	<3 m	10 m	5–10 m	10–100 m	10–300 m
Max. nodes	2	2	8	Application specific	32	65,000
Cost	Very low	Low	Low	Low	Medium	Medium

with a two-level hierarchical approach, where the sensors located inside each room are supported through a local router, while the second level of abstraction provides packet routing through a central gateway (e.g., a gateway might collect information from the entire building). The smart thermostat industry historically has generated small amounts of data, as they were captured from the corresponding weather and activity sensors, respectively.

Apart from the topology, the desired functionality for data transfer is also affected by the employed communication protocol. For the scopes of our usecase, a new application-specific (fully customizable) communication protocol has been implemented. The functionality of this protocol is completely reactive, as it waits for the arrival of any packet to be processed. For every packet which is received, its type has to be analyzed. Our approach supports five packet types, as they are summarized in Table 12.2. Specifically, in case the packet type is *HELLO*, then the corresponding acknowledge (*ACK*) packet is created, which contains information about the employed DATA packet size (depending on the link's implementation), as well as its origin and destination nodes. The *START* and *STOP* packets denote the starting and finishing of data transfer between source and destination nodes. This is especially crucial since many of the available communication protocols (as they were discussed at Table 12.1) support only P2P links; thus an established link cannot be shared with other nodes. The weather/occupant's information is transmitted in *DATA* packets, while control messages (e.g., link configuration, desired link speed, selected encryption scheme, etc.) are sent in *CONTROL* packets. As a response to these packets, the node confirms their proper receive with an *ACK* packet.

Throughout this chapter, we applied the previously mentioned communication scheme as part of the underline IEEE 802.15.1 protocol in order to support the data transfers between distributed sensors and low-performance processing nodes.

Table 12.2 Description of packets at the employed communication protocol

Type of packet	Description
P_HELLO	This packet discovers the network
P_START	The P_START packet is used for establishing the connection
P_STOP	It is used for terminating the connection
P_CONTROL	Includes control messages for operating device and link
P_DATA	It contains data about the patient's heart rate, as well as all the necessary redundant data for error correction. Note that P_DATA packets can be adapted to carry even more information
P_ACK	This packet acknowledges the successful receipt of packet

12.6 Evaluation

This section provides a number of experimental results that quantify the efficiency of the introduced framework. By appropriately computing the temperature set-points per thermal zone it is feasible to considerably improve the energy consumption and the occupants' thermal comfort. The experimental setup for our analysis consists of five buildings, as it was presented in Sect. 12.2. Without loss of generality both the proposed implementation and the reference control solutions adopt a 10-min time-step, i.e., each thermostat is configured once per 10 min. Additionally, we consider that people occupy the buildings only during the operating hours, while the distribution of people per room varies during the day as it is summarized in Table 12.3.

Each building is equipped with a number of weather sensors that monitor indoor/outdoor air temperature, humidity, and radiant temperature values. Furthermore photovoltaic (PV) panels are also employed in favor of minimizing the energy cost. Depending on the energy requirements the buildings can interact with the main grid in order to purchase or sell energy. The problem cannot be considered trivial due to the intermittent behavior of the solar energy, the uncertain dynamics of the buildings, and the need to meet the thermal comfort constraints for the micro-grid occupants. More thoroughly, the optimal operation for HVAC systems pre-assumes a mechanism that is able to handle in a closely coupled manner the increased number of thermal zones and cooling/heating strategies. This is a typical multi-objective problem, where the reduction of energy cost and the maximization of thermal comfort are in conflict with each other. Therefore, no single optimal solution can be found in these problems. Instead, a set of trade-offs that represent the best possible compromises among the objectives have to be computed. Regarding our experimentation, we consider that each of these cost metrics is of equal importance.

The target buildings are located in Crete (Greece), whereas our experimentation relies on real weather data [6]. Figure 12.8 plots the variation of the external air temperature and humidity values, as they have been acquired by the building's sensors. According to this figure, the variation of the external air temperature ranges between 14.6 and 37.2 °C, while the corresponding range for the external relative humidity and radiant temperature are 7.9–38.8 % and 0–47 KW/m^2, respectively. Therefore, proper selections of temperature set-points per thermal zone are absolutely necessary in order to minimize overall cost.

Table 12.3 Summary of building properties

Building	Surface area	Thermal zones	Operating hours	Warming-up pre-cooling	Random occupancy
#1	350 m^2	8	6:00am–9:00pm	No	Yes
#2	525 m^2	10	8:00am–9:00pm	Yes	Yes
#3	420 m^2	10	8:00am–5:00pm	Yes	Yes
#4	280 m^2	6	7:00am–8:00pm	Yes	Yes
#5	228 m^2	4	6:00am–6:00pm	No	Yes

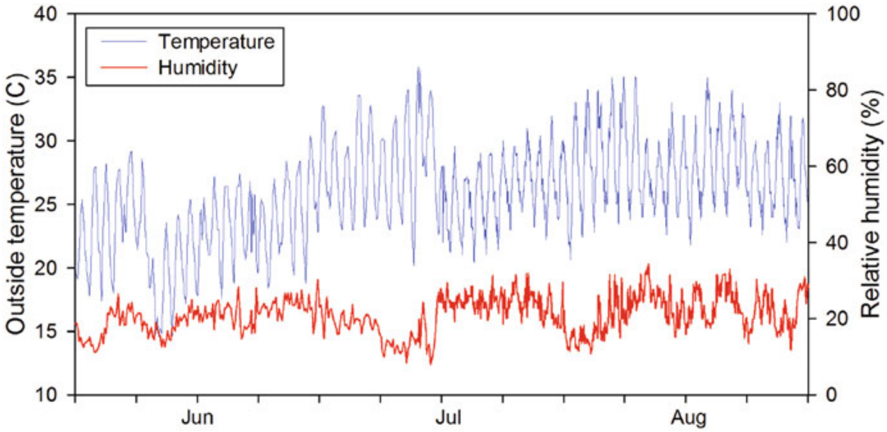

Fig. 12.8 Variation of temperature and humidity values for the summer period

Next, we quantify the efficiency of proposed decision-making mechanism in term of computing accurately the desired temperature set-points. For demonstration purposes, two well-established control mechanisms are employed as reference solutions for this analysis. More specifically, we employ the *Pattern Search* [16] and the *Fmincon* [17] solves. Both of these techniques are implemented in Matlab's Optimization Toolbox as an open-loop optimization procedure. Note that throughout this analysis only the cooling operation of HVACs is considered. Due to the enormous computational complexity imposed by the aforementioned existing solvers, each month is studied as a separate sub-problem and is addressed individually.

A well-established metric for quantifying the efficiency of HVAC control mechanisms is the improvement of thermal comfort. For this purpose, we study this metric in terms of the predicted percentage of dissatisfied (PPD) people, which portrays the percentage of the buildings' occupants that are dissatisfied with the current thermal conditions. The results of this analysis regarding the two existing solvers, as well as the introduced decision making, are summarized in Fig. 12.9. For demonstration purposes, these results correspond to the hottest summer day (middle of August). According to this figure, we can conclude that the introduced decision-making mechanism results to the minimum PPD among the alternative solutions. We have to mention that early at the morning, all the solvers exhibit increased PPD values, since people enter into a "closed" building. However, even in this case, the proposed framework leads to the minimum overhead.

Along with the occupants' thermal comfort, we are equally interested in reducing the total energy consumption and thus the energy expenses as well. Figure 12.10 depicts the variation of energy cost regarding the alternative decision-making mechanisms for the same August's day. To be consistent, this analysis does not take into account the power saving from PV panels. This analysis indicates that our framework achieves to reduce energy demand compared to Fmincon solver (this solver exhibits similar performance in term of PPD). Although the pattern

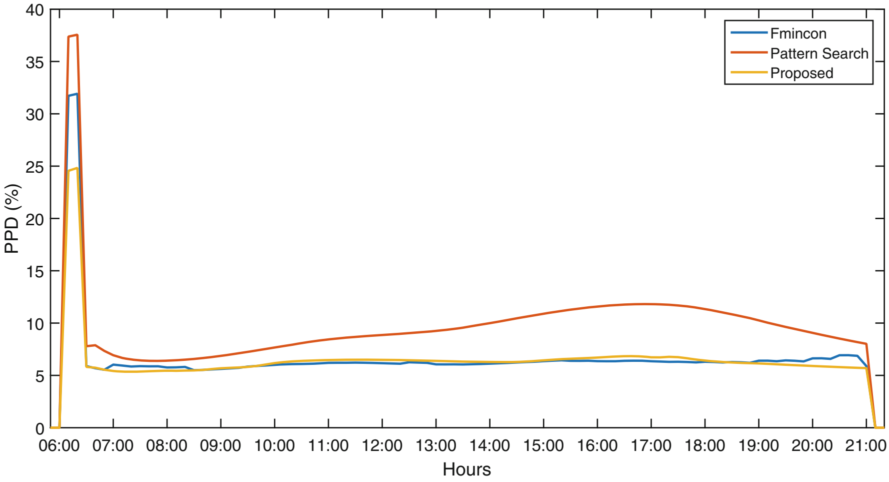

Fig. 12.9 Variation of thermal comfort during an August's day

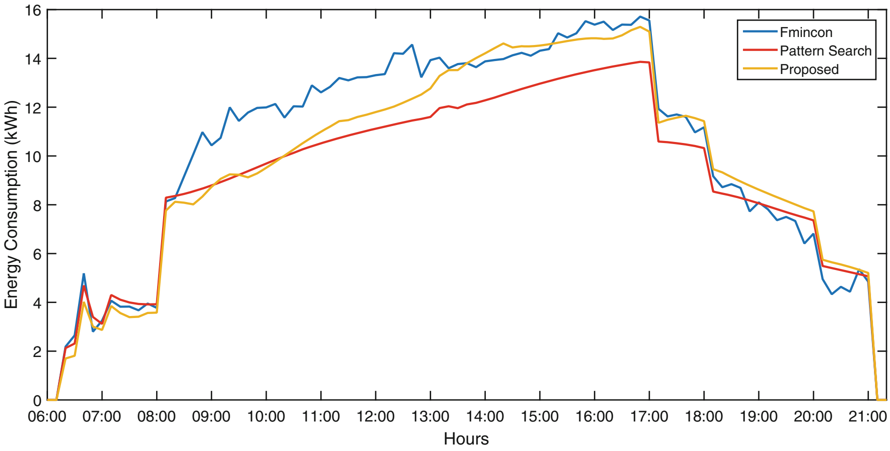

Fig. 12.10 Variation of energy consumption during an August's day

search solution seems to improve further the energy consumption, it also leads to higher PPD values; thus, it could not be though as an overall good approach (by taking into consideration both of the cost metrics). Note that similar results for energy consumption and thermal comfort are also retrieved for the rest days of our experiment.

Finally, in order to discuss the overall efficiency of our decision-making framework, as compared to existing approaches, we quantify the total cost for the entire 3 month experiment (summer period). As we highlighted previously, the selected experiment duration (summer period) exhibits increased temperature and humidity

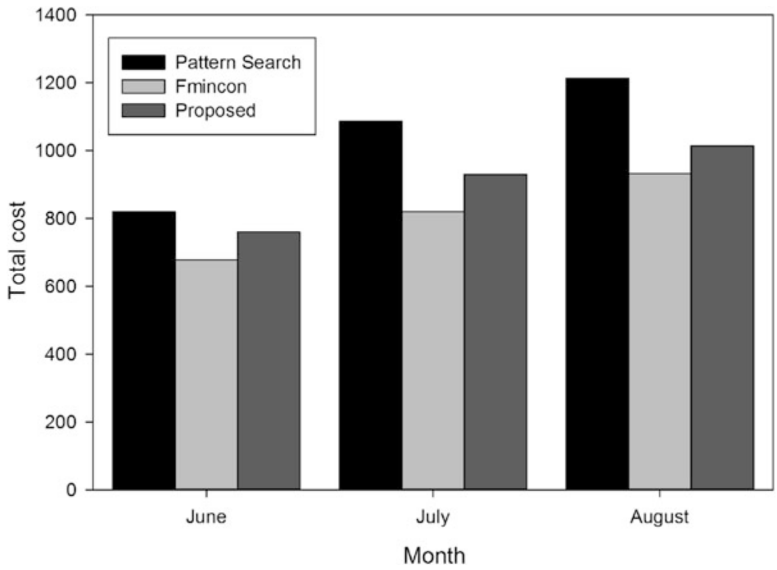

Fig. 12.11 Overall cost of alternative decision-making mechanism for the summer period

values, which should be considered during the building cooling procedure. For this analysis, both occupant's thermal comfort and the total energy cost are equally considered. The results from this experiment are summarized in Fig. 12.11. For demonstration purposes, these results are plotted in normalized manner over the results derived from pattern search solver. Based on this analysis, we might conclude that our solution is by far more efficient as compared to pattern search solver (average improvement by 8–16 %), while the Fmincon solver achieves an additional average enhancement by 8 %. Apart from the efficiency of alternative solvers to compute the optimum (or optimal) temperature set-points, the computational complexity of these approaches is also of high importance. We have to notice that the proposed solution can compute the output much faster than the existing approaches.

12.7 Conclusions

A novel framework for implementing a flexible yet efficient decision-making mechanism for large-scale IoT platforms was introduced. To support this functionality, a number of connected smart thermostats that have the capability to monitor their own performance, to classify, to learn, and to take proper actions are employed in a building environment. The proposed solution evolves computational intelligence in order to maximize the occupant's thermal comfort without affecting the overall energy cost. Experimental results highlight the superior of the introduced framework compared to existing approaches. Also, it is important to highlight the considerable lower computational complexity imposed by the proposed solution, proving it ideal for employment as part of a smart thermostat.

Acknowledgements The research leading to these results has been partially funded by the European Commission FP7-ICT-2013.3.4, Advanced computing, Embedded Control Systems, under the contract #611538 (Local4Global, http://www.local4global-fp7.eu/). Also, this work presented is partially supported by the FP7-2013612069-2013-HARPA EU project.

References

1. HiPEAC, HiPEAC Vision 2015 (2015) (available online: https://www.hipeac.net/publications/vision/)
2. J. Gantz, D. Reinsel, The digital universe in 2020: big data, bigger digital shadows, and biggest growth in the far east - United States, in *IDC iView: IDC Analyze the Future*, vol. 2007, pp. 1–16 (2012)
3. D. Estrin, D. Culler, K. Pister, G. Sukhatme, Connecting the physical world with pervasive networks. IEEE Pervasive Comput. **1**(1), 59–69 (2002)
4. ITRS, International Technology Roadmap for Semiconductors (2013) (available online: http://http://www.itrs.net/)
5. C. Bertin, C. Guillon, K. De Bosschere, Compilation and virtualization in the HiPEAC vision, in *2010 47th ACM/IEEE Design Automation Conference (DAC)*, pp. 96–101 (2010)
6. National Renewable Energy Laboratory, Weather Data Sources (2016) (available online: https://energyplus.net/weather)
7. National Renewable Energy Laboratory, EnergyPlus™ (2016) (available online: https://energyplus.net/)
8. Modelica Association, Modelica (2016) (available online: https://www.modelica.org/)
9. L. Wei-Min, J. Doyle, $\mathbb{H}\infty$ control of nonlinear systems: a convex characterization. IEEE Trans. Autom. Control **40**(9), 1968–1675 (1995)
10. T. Baúar, P. Bernard, $\mathbb{H}\infty$-*Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach* (Birkhäuser, Boston, 1995)
11. D. Mayne, J. Rawlings, C. Rao, P. Scokaert, Constrained model predictive control: stability and optimality. Automatica **36**(6), 789–814 (2000)
12. L. Magni, G. Nicola, L. Magnani, R. Scattolini, A stabilizing model-based predictive control algorithm for nonlinear systems. Automatica **37**(9), 1351–1362 (2001)
13. D. Bertsekas, *Nonlinear Programming*, 2nd edn. (Athena Scientific, Belmont, MA, 2004)
14. M. Trčka, J. Hensen, M. Wetter, Co-simulation of innovative integrated HVAC systems in buildings. J. Build. Perform. Simul. **2**(3), 209–230 (2009)
15. T. Nghiem, G. Pappas, Receding-horizon supervisory control of green buildings, in *American Control Conference (ACC)*, pp. 4416–4421 (2011)
16. C. Audet, J.E. Dennis Jr., Analysis of generalized pattern searches. SIAM J. Optim. **13**(3), 889–903 (2002)
17. R.H. Byrd, J.C. Gilbert, J. Nocedal, A trust region method based on interior point techniques for nonlinear programming. Math. Program. **89**(1), 149–185 (2000)