A fog-based hybrid intelligent system for energy saving in smart buildings

Article

Accepted version

A. De Paola, P. Ferraro, G. Lo Re, M. Morana, M. Ortolani

A Fog-based Hybrid Intelligent System for Energy Saving in Smart Buildings

**Alessandra De Paola · Pierluca Ferraro ·
Giuseppe Lo Re · Marco Morana ·
Marco Ortolani**

Abstract In recent years, the widespread diffusion of pervasive sensing devices and the increasing need for reducing energy consumption have encouraged research in the energy-aware management of smart environments. Following this direction, this paper proposes a hybrid intelligent system which exploits a fog-based architecture to achieve energy efficiency in smart buildings. Our proposal combines reactive intelligence, for quick adaptation to the ever-changing environment, and deliberative intelligence, for performing complex learning and optimization. Such hybrid nature allows our system to be adaptive, by reacting in real time to relevant events occurring in the environment and, at the same time, to constantly improve its performance by learning users' needs. The effectiveness of our approach is validated in the application scenario of a smart home by extensive experiments on real sensor traces. Experimental results show that our system achieves substantial energy savings in the management of a smart environment, whilst satisfying users' needs and preferences.

Keywords Ambient Intelligence · Fuzzy Systems · Fog Computing · Energy Efficiency

1 Introduction

The growing need for environmentally friendly ICT solutions aimed to support users of a smart environment in their everyday activities has inspired scientific research and has increasingly gained attention from many governments. Particular emphasis has been reserved to the issue of energy efficiency where shortcomings may depend on several complex and intertwined factors, including those related to the aging urban infrastructures. A recent report of the European Commission states that buildings are responsible for 40% of energy consumption in the EU (European Commission 2017), and this percentage sounds even more alarming if we consider that about 35% of the EU

A. De Paola, P. Ferraro, G. Lo Re and M. Morana are with the Department of Engineering, University of Palermo, Viale delle Scienze, ed. 6 - 90128 Palermo, Italy; E-mail: alessandra.depaola@unipa.it, pierluca.ferraro@unipa.it, giuseppe.lore@unipa.it, marco.morana@unipa.it
M. Ortolani is with the Keele University, Keele, Staffordshire, UK, E-mail: m.ortolani@keele.ac.uk

buildings are over 50 years old, and older buildings consume 7 times more than newer ones; a reduction in the energy consumption of pre-existing buildings would thus likely produce a noticeable positive effect on local economy, besides improving the citizens' quality of life. It is clear that strong actions must be devised to stimulate virtuous behaviors at various scale, from urban level down to the individual smart homes.

As the activities of the users have considerable influence on the energy footprint of residential, office, or commercial buildings, the European Union has consequently issued directives to promote a change in consumer behavior, and to foster better energy consumption practices. At the same time, the design of Building Energy and Comfort Management (BECM) systems (Nguyen and Aiello 2013) has grown into an independent research area, as part of the greater field of Ambient Intelligence (AmI) (Cook and Das 2007; Cook et al. 2009). Many efforts have been made in the past years to design systems for fine-tuning the deployed devices, or appliances, according to users' actual needs in order to reduce potential energy waste. Even though such systems are bound to gather data about the surrounding environmental conditions, typically through a network of pervasively deployed devices, they usually include a component of *artificial reasoning* in order to devise a plan toward the final goal of saving energy.

Most AmI systems attempt to maintain a sharp separation between low-level functionalities, and high-level ones by exploiting a distributed architecture where low-power devices are used as pure sensing platforms, whereas the intelligent algorithms reside in remote servers that can be reached through the Internet. In this context, the cloud computing paradigm is frequently adopted as a feasible solution to confine heavy computation in the cloud. However, all the benefits brought by this approach could be negligible in real-time applications, where data are continuously transferred from/to the cloud. To overcome this limitation, in 2012, the fog computing paradigm was introduced by Cisco as an extension of the cloud computing at the edge of the network. Nowadays, when dealing with large amounts of data which need to be processed locally, and timely, the fog has been widely accepted as the most proper solution to distribute data collection, analysis, and storage tasks among different devices located at distinct logic levels.

In our proposal, a fog-based architecture is adopted to enable a *hybrid intelligence* solution for the control of a smart environment in the context of an AmI system addressing energy efficiency in buildings (De Paola et al. 2015). The goal of the whole framework is to make the environment responsive to the users' needs by exploiting information gathered through pervasive monitoring equipment, while also meeting energy saving requirements. We address this issue by proposing a hybrid approach that combines a *reactive* component, hosted by edge devices and capable of quickly reacting to changes, with a *deliberative* component designed to perform more complex tasks (e.g., optimization) on the cloud.

Reactive intelligence is provided by a fuzzy controller which acts in response to current conditions by automatically selecting the actions the actuators should perform. This reactive behavior is not statically coded, but it is dynamically driven by the deliberative component according both to the current measurements and the overall system goals.

The remainder of the paper is organized as follows. Section 2 briefly discusses relevant work presented in the literature. The architecture of the Ambient Intelligence system is presented in Section 3. Section 4 describes the AmI algorithms hosted by the middleware layer which aim to model the users and the environment, while Section 5 presents the hybrid intelligent algorithms implemented for controlling the environment.

Section 6 describes the experimental setup and results. Conclusions are discussed in Section 7.

2 Related Work

The separation of the functionalities of an intelligent system between different logic levels can be achieved by making appropriate architectural choices which rely on the distributed model.

Fog computing, in particular, is becoming one of the most promising approaches when planning sustainable development of urban areas (Perera et al. 2017). This paradigm can be, for instance, successfully adopted in application scenarios involving energy management (Stojkoska and Trivodaliev 2017b; Yan and Su 2016; Stojkoska and Trivodaliev 2017a), Ambient Assisted Living (AAL) (Bhargava et al. 2017) and health-care (Stantchev et al. 2015; Rahmani et al. 2018).

Fog computing promotes a shift in information management; in particular, it is claimed that storage and processing should not be strictly confided to cloud platforms, but rather be partially moved to devices located on the edge of the network, such as terminal devices or intermediate gateways. Such shift, however, needs not be absolute; on the contrary the cloud generally maintains most of the computing-demanding functionalities, but lets the computational capabilities of edge devices to be fully exploited (Perera et al. 2017). One of the main advantages of combining cloud and fog computing is lower latency, which translates to real-time reaction in response to observed data. Such feature is useful in many smart buildings scenarios that typically need to support prompt responses to changes of the environment and of the context (Perera et al. 2017). An additional desirable feature that can be so achieved is a higher system availability. A purely cloud-based approach may suffer from several connectivity issues (Stojmenovic and Wen 2014), while a fog-based approach allows edge devices to independently perform their tasks even when the connection with the cloud is lost (Perera et al. 2017). Finally, fog computing addresses one of the main issues of pervasive environments, i.e., the heterogeneity of sensor and actuator devices. Intermediate devices can interact with physical devices and provide the cloud with a homogeneous point of access (Amadeo et al. 2017). For instance, a three-tier architecture aiming to provide an abstract view over heterogenous sensory devices in order to implement a number of higher-level smart services is presented in (De Paola et al. 2012).

Some researchers have proposed to exploit edge devices to perform a more efficient data storage, as the authors of (Yan and Su 2016) that describe a system for monitoring the energy consumption, where the smart meters act as edge devices and are responsible for locally storing consumption information. The smart meters thus constitute a distributed database that may be accessed on-demand by the cloud server. Edge devices can be also exploited to perform data compression in order to reduce the information volume to send toward cloud servers, by adopting moving average, autoregressive models, and least mean square (Stojkoska and Trivodaliev 2017a; Santini and Romer 2006).

One of the most relevant fog-based strategies which allows to achieve the aforementioned goals is to split the computations required to provide the system functionalities across different layers (e.g., IoT devices, edge gateways, cloud servers), by exploiting the respective device capabilities (Bhargava et al. 2017; Sahni et al. 2017). In some simple scenarios, data processing occurs completely on the IoT devices, as proposed

in (Bhargava et al. 2017) which describes an AAL system performing user localization through wearable sensors. Such devices perform activity recognition entirely on-board, and rely on the cloud infrastructure only to make easier accessing to recognition result. More often, the solutions described in the literature uses simple sensors, whose computational capabilities are very limited, and rely on intermediate gateways, deployed near the physical infrastructure, to perform a relevant part of the whole processing. In the healthcare system proposed in (Rahmani et al. 2018), the edge gateways, besides managing the connection with the heterogeneous set of devices composing the body area network, perform a local pre-filtering in order to reduce the amount of data to be transferred to the cloud. Moreover, the edge gateways locally analyse the gathered data in order to launch timely alarms for local operators, if anomalous events occur. Such solution allows to minimize latency and guarantees the correct functioning of the system even with a poor connectivity with the cloud. Similarly, the authors of (Verma and Sood 2018) leverage fog devices to classify data collected by a body area network through a Bayesian network, and use the cloud to analyze the trend over time of biometrics parameters, in order to provide valuable information to a decision support system.

In an Ambient Intelligent system, splitting the reasoning capabilities across different levels of the architecture poses a challenge regarding how to ensure unity and consistency of the reasoning. To this end, the use of a distributed hierarchical task network (D-HTN) has been proposed by (Amigoni et al. 2005); however, the *context* in which the system operates also needs to be taken into account and HTNs, together with an ontology-based occupant activity recognition technique, have been suggested as a tool to gather contextual information for energy saving in smart offices (Georgievski et al. 2013). A comprehensive survey of intelligent management systems for energy efficiency in buildings is proposed in (De Paola et al. 2014), which describes the architectural solutions and the intelligent functionalities used to perform energy management, as well as how such functionalities are split over system levels.

Clearly, neither the collected measurements, nor the information about user actions can be considered completely reliable, so great effort has been devoted to devise systems able to cope with uncertainty, either in the form of noisy measurements about the environmental quantities, or in variability of actions across different groups of users, or over time for the same user. Hagras *et al.* (Hagras et al. 2015) propose the use of fuzzy controllers for handling uncertainties in AmI systems. Both the fuzzy rules, and the membership functions may be inferred by using specifically designed meta-heuristics, typically through evolutionary methods. The work in (Herrera 2008) contains a taxonomy of genetic fuzzy systems with learning and adaptation capabilities, while methods for extracting an optimal set of fuzzy rules from an initial set, by applying multiobjective evolutionary algorithms are described in (Fazzolari et al. 2013; Hinojosa and Camargo 2013).

3 The Fog Infrastructure supporting the AmI System

This section describes the fog-based architecture adopted to manage heterogeneous IoT devices in a smart environment, to process data coming from different sources in an efficient way, and to infer high-level knowledge of the user's behavior according to the environmental conditions.
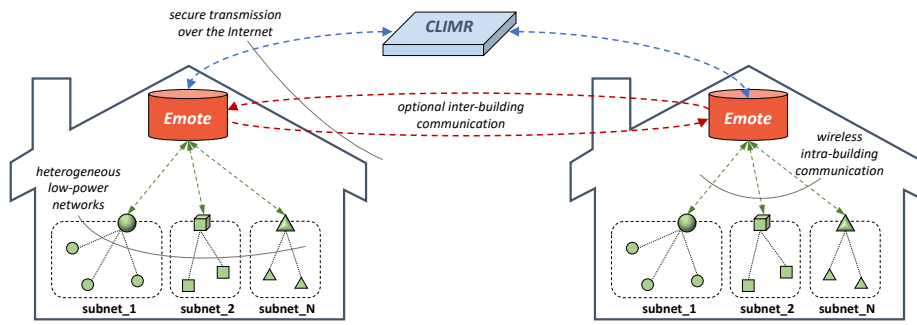
Fig. 1: System Architecture. Edge devices (green) cooperate within a low-power network, and share data about the user/environment with the fog units (orange). Data processed here are sent to the cloud data center (blue) to learn the overall system models.

The fog computing paradigm allows to allocate the computing tasks among several heterogeneous devices logically distributed over three layers: the boundary sensing infrastructure (*edge*) , the collecting intermediate computing devices (*fog*), and the remote storage and processing unit (*cloud*). Determining which activites are most suitable to be placed at the *edge*, *fog*, and *cloud* layers is critical to optimize the performances of the system. According to the model we propose, the *edge* layer consists of low-power Sensor and Actuator Networks (SANs) responsible for observing the environment and changing its state. The core functionalities of the AmI system are provided by a computing device that is logically located at an intermediate (*fog*) layer, so as to quickly process data coming from the SAN, and implement the **reactive** intelligence behaviors. Finally, the *cloud* layer hosts the software components that provide the **deliberative** intelligence functionalities required to maintain an overall representation of the smart home.

The sensor and actuator network is made up of a number of heterogeneous devices, both wireless and wired, communicating to each other by means of various network protocols (e.g., Bluetooth, Zigbee, Z-Wave, NFC). In order to guarantee system scalability and compatibility with as many different devices as possible, homogeneous subnetworks are managed through specific gateways which provide the proper interfaces to enable a bidirectional communication between the edge and the upper levels. At this *edge* level, since the system processes raw — non sensitive — data, communications are not encrypted, also meeting the computing constraints of the devices.

As showed in Fig. 1, the *Emote* (Energy Monitor) is the *fog* device that hosts several software components, named *AmI modules*, aimed at processing data captured by SANs at the *edge* of the network. In particular, the *AmI modules* allow to (i) create mathematical models of temperature and humidity trends over time, (ii) model the impact of the actuators on the overall energy consumption, (iii) recognize the activities performed by the users on the basis of multi-sensor information, and (iv) learn users' preferences. The *Emote* includes also additional modules for the communication with the lower level, which also provide the network management functionalities for each subnet, i.e., sensor registration, command dispatch, event/fault management and data delivery. In our architecture, a single smart home can be monitored by means of one or more *Emote*s, depending on the size of the area covered by the AmI system.

Fog devices are not only responsible for forwarding information from/to sensor and cloud layers, but can also communicate with each other to allow the overall AmI system to achieve complex goals. For instance, several *Emote*s controlling a single smart home should share with each other information about the energy consumption of the building and the users' behavior, even though each *Emote* maintains also specific models of the environment in which it is installed. Moreover, the *Emote*s responsible for different buildings within the same area (such as a hospital) could share their inner models to allow a single administrator to apply the most proper policy for all the buildings. At this level, information is exchanged using reliable wireless (e.g., WiFi, GSM/3G/4G) and wired networks. Here, data transmission is protected through encryption and authentication techniques, which guarantee both data integrity and user's privacy.

Information generated at the fog layer is sent to the cloud, where a component named *CLIMR* (Cloud Intelligent Modeler and Reasoner) provides the deliberative intelligence required to maintain an overall representation of the smart home, finding the best trade-off between the satisfaction of user's preferences and the minimization of energy consumption. The overall models of the AmI system are stored in the *CLIMR*, and sent back to the *Emote*s at the fog layer in order to update their behaviors, making the whole system synchronized.

Our system adopts a straightforward scheduling strategy according to which the general tasks that must executed by the *edge*, *fog*, and *cloud* devices are identified during the design of the AmI system. For instance, environmental monitoring and controlling is always performed at the *edge*, reactive intelligence tasks are always demanded to the *fog*, and deliberative intelligence routines are always executed by the *cloud*. This static strategy guarantees that tasks are always assigned to devices that have enough computing resources and power capabilities to complete them.

4 Modeling the Users and the Environment

As briefly discussed in the previous Section, the methods necessary to model the user behavior and the environment reside at the *fog* layer, i.e., in the *Emote*, where the following intelligent modules are provided: the *Actuation Module*, the *User Profiler*, the *Activity Recognition Module*, and the *Environmental Modeler*.

In the context of energy optimization, the aim of the *Actuation Module* is to learn the energy consumption of the actuators according to different settings. For instance, it might exploit sensory data gathered by specialized sensors and characterized by a low level of noise, such as those reported in (Corucci et al. 2011); it might also be manually configured by the system administrator according to the technical specifications of the actuators. The *User Profiler* module collects the preferences about environment conditions explicitly stated by the user, and can be easily extended to learn the user preferences exploiting also implicit feedbacks obtained by observing the interaction between the user and the actuators. In order to highlight the interplay between the reactive and the deliberative components, in the following, we provide more details about the remaining two modules, i.e. the *Activity Recognition Module* and the *Environmental Modeler*.

The *Activity Recognition Module* aims at analyzing heterogenous sensor data in order to detect and recognize the activities performed by the users. Activities are modeled as a Markov Chain in which the set of sensory readings $\boldsymbol{S_t} = (S_t^1, \ldots, S_t^n)$,

gathered by all the n sensors at time t, can be considered as the observable manifestation of the state X_t. The validity of adopting Markovian models to analyze users' behavior by exploiting pervasive sensory information is confirmed by several works in literature (Atallah and Yang 2009).

The link between the current state and the sensory manifestation observed by the i-th sensor is given by the probabilistic sensor model $P(S_t^i|X_t)$. Our model includes also a limited set of context information, $\boldsymbol{C_t} = (C_t^1, \ldots, C_t^k)$, which affects the current state of the system. Thanks to the Markov assumption, the current state belief depends only on the past state, the context information and the current observations, as expressed by the following equation:

$$Bel(X_t) = P(X_t|\boldsymbol{C}_{1:k}, \boldsymbol{S}_{1:n}) = P(X_t|X_{t-1}, \boldsymbol{C}_t, \boldsymbol{S}_t). \tag{1}$$

In order to reduce the complexity of the activity recognition procedure, only a reduced set of context information about the scheduled activities is considered. Sensory data are provided by pervasive inexpensive sensors; in particular, motion and door have been considered, whereas we chose not to monitor temperature and humidity since such quantities are not correlated with the activity performed by the users (De Paola et al. 2017).

Finally, the goal of the *Environmental Modeler*, is to learn mathematical models of the physical phenomena affecting the environmental conditions (i.e., temperature, humidity), possibly in response to the actuators' settings. Such models are built by exploiting a given number of past sensory observations, and their parameters are continuously updated in order to minimize the prediction error. Our model is based on the linear predictive controller proposed in (Gruber et al. 2001) and is trained using indoor sensors and an external weather forecast service. A rough 24-hour prediction of the outdoor temperature is generated according to data provided by the weather forecast service in the past 24 hours; then, as new data become available, the prediction is updated by making a linear correction over the next 6 hours. The indoor temperature is predicted according to an analogous linear model based on indoor and outdoor temperature captured in the past 24 hours, while the indoor humidity model depends on the humidity values measured in the past 24 hours and on the predicted indoor temperature. The linear dynamic relations for outdoor temperature T_e, indoor temperature T_i and indoor humidity percentage H_i are shown in eq.(2), (3) and (4) respectively. The indoor phenomena are also affected by the actuators' settings, and such influence is modeled through a factor $\mu_t(a_t)$ that is dynamically learned on the basis of the past observations.

$$T_e(t) = \alpha_t T_e(t-1) + \beta_t; \tag{2}$$

$$T_i(t) = \alpha_t T_i(t-1) + \beta_t T_i(t-2) + \gamma_t T_e(t-1) + \varphi_t T_e(t-2) + \mu_t(a_t); \tag{3}$$

$$H_i(t) = \alpha_t H_i(t-1) + \beta_t T_i(t) + \mu_t(a_t). \tag{4}$$

The service provided by the *Environmental Modeler* is requested by the *CLIMR* to foresee the effect of a given actuator setting on the environmental conditions, in order to evaluate which action would produce the desired conditions, and to estimate the time required to reach them.
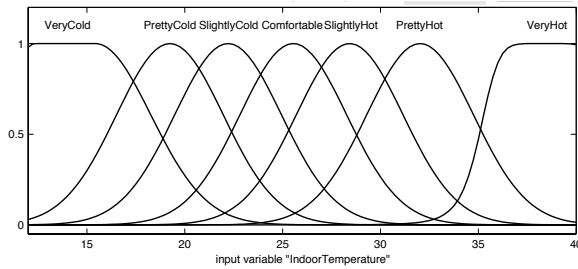
Fig. 2: Membership functions for the input variable associated to the indoor temperature.

5 Hybrid Intelligence for Environmental Control

In order to learn from past experiences and to adapt the behavior of the AmI system to variable conditions as well as users' needs, we propose a hybrid approach that combines two different types of intelligence: *reactive* to quickly respond to unexpected changes, and *deliberative* to perform complex reasoning.

5.1 Reactive Intelligence

The reactive intelligence of the AmI system, provided by the computing devices deployed at the *fog* layer, is based on fuzzy logic (Zadeh 1975). Fuzzy approaches have been frequently, and successfully, applied in the field of Ambient Intelligence (Doctor et al. 2005) in order to deal with the unavoidable subjectivity of the evaluations provided by the users. As an example, let us consider the indoor temperature of the controlled environment, which is a typical concept considered by AmI systems. It is not convenient to perform a rigid threshold-based classification in order to establish whether the current temperature is *hot* or *cold*; on the contrary, a fuzzy system allows a given temperature to be considered at the same time as *hot* and *cold*, but with different *membership values*. This concept is exemplified in Fig. 2, which shows the Gaussian *membership functions* (MFs) of the fuzzy sets used by our system to perform the temperature classification.

In order to reason on the various concepts related to the environment management, the reactive intelligence module is organized into independent components, each addressing a different set of independent environmental features. In particular, we propose two distinct fuzzy controllers for reasoning on (i) temperature and humidity, and (ii) light level, which are separately optimized by the deliberative component.

Regardless of the specific environment where the AmI system is deployed, the input variables to the temperature controller are known in advance: the indoor/outdoor temperature, the indoor humidity percentage, and information about the activity performed by the user. The indoor temperature and humidity are computed as an average of the measurements taken by the sensors deployed in the target room, while the outdoor temperature is measured on the building facade. Finally, the activity performed by the user is detected by the *Activity Recognition Module*.

In order to define the output variables is necessary to know the set of actuators installed in the target room, and for each of them, the set of available commands
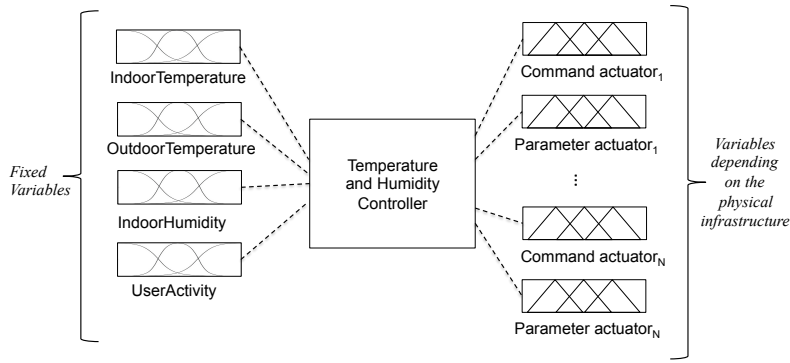
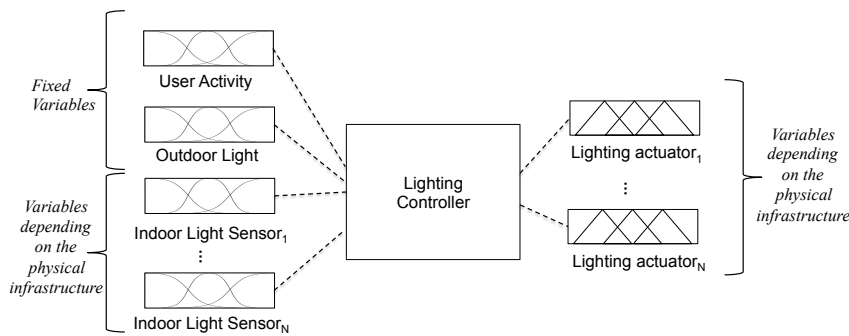Fig. 3: Temperature and humidity fuzzy controller.

Fig. 4: Lighting level fuzzy controller.

and their possible parameters. A typical scenario might consist of a room containing one HVAC actuator with four possible operating modes, i.e., Dry, Cold, Hot and Off, where Cold and Hot commands allow the user to specify the desired temperature. Consequently, the fuzzy controller will be characterized by two output variables, one with values mapping the four operating modes, and the other corresponding to the desired temperature, with integer values from 18 to 32. The fuzzy controller for temperature and humidity is shown in Fig. 3.

Unlike the spatial and temporal smoothness of temperature and humidity, lighting is characterized by a discontinuous behavior; thus, the average lighting level of a room is not a significative information. On the contrary, it is convenient to maintain fine-grained information about the lighting level in different locations of the considered room. Consequently, the structure of the lighting controller (see Fig. 4) is more complex than the one for temperature-humidity, because both its input and output depend on the physical deployment of sensors and actuators. To be more specific, there are as many input variables as the number of light sensors, one input variable related to the activity performed by the user, and one representing the external light level; each of the output variables is associated with a specific lighting controller. The number of light sensors and controllers, together with their commands and parameters, are defined by the physical infrastructure ontology.

The input variables of the controllers are modeled by Gaussian MFs, that are commonly adopted, as well as triangular and trapezoidal ones, to represent environmental physical features such as temperature or humidity (Baghban et al. 2016; Doctor et al. 2005). The identification of the proper MFs for each input variable has been performed through a simplified version of the double-clustering approach proposed in (Doctor et al. 2005; Castellano et al. 2002). Firstly, environmental data are clustered in a n-dimensional space, e.g., temperature and humidity are mapped in a two dimensional space as shown in Fig. 5. Each cluster is then projected along each dimension and represented by a *cluster prototype*, obtained by computing the mean value and the variance of data belonging to that cluster. If the distance between two projected prototypes is too low, the two clusters are merged and the resulting prototype represents a MF for a single variable.

As last step, given the range of admissible values $[v_i, v_f]$ for a certain variable, the boundary membership functions MF_i and MF_f, whose centroids C_i and C_f are the closest to v_i and v_f respectively, are manually modified in order to assign a membership value equal to 1 to any input point $v : v_i \leq v \leq C_i$ and $C_f \leq v \leq v_f$.

Output variables, which are intrinsically discrete, are represented through triangular MFs.

It is worth noting that the membership functions are computed only once, and are never updated. The fuzzy rules, on the other hand, are periodically learned by the deliberative intelligence, as it will be discussed in next Section.

Both the temperature-humidity and the lighting controllers are based on simple fuzzy rules in the form: «**if** *antecedent* **then** *consequent*», where the antecedent and the consequent are conjunctions of conditions on input and output variables respectively. As an example, a simple rule of the temperature controller may look like: «**if** Activity **is** Working **and** IndoorTemperature **is** Hot **then** Mode **is** Cold **and** DesiredTemperature **is** 24».

The adopted rules are of Mamdani type, i.e., they are characterized by the adoption of linguistic terms both for the antecedent and the consequent (Hosseini et al. 2012). Differently from Sugeno-type rules, Mamdani controllers allow to obtain a non-continuous output surface, that is more suitable for representing the discrete set of actuators' commands. Moreover, Mamdani rules are easy to understand for humans and this aspect is crucial to guarantee the proper management of the AmI system. The final defuzzification step is performed through the Centroid of Gravity (CoG) approach.

Tuning the set of fuzzy rules requires the selection of the number of rules to be adopted, the antecedent, and the consequent of each rule. This process is performed by the deliberative intelligence by combining a clustering step with a meta-heuristic algorithm.

5.2 Deliberative Intelligence

The *deliberative intelligence*, provided by the high performance computing devices available on the *cloud*, aims to optimize the behavior of the reactive intelligence in order to find the best trade-off between meeting the user's preferences and minimizing the energy consumption.

The optimization of a fuzzy system could be focused on different elements chosen among the set of linguistic variables, the fuzzy MFs, or the fuzzy rules (Herrera 2008). In our proposal, the set of linguistic variables and the MFs are fixed and represent the

a-priori knowledge for the fuzzy controller. Therefore, the *deliberative intelligence* will focus on optimizing the set of fuzzy rules and, with respect to the taxonomy proposed in (Herrera 2008), our approach can be classified as a "rule learning and selection" method.

Several works in the literature aim to learn fuzzy rules by exploiting different approaches, such as neural networks, reinforcement learning, evolutionary and heuristic search algorithms. In (Nauck and Kruse 1993) a neural network, whose nodes represent the preconditions and the results, is used to simulate the behavior of the fuzzy rules. All nodes of the network are connected through weighted links updated by means of a back propagation procedure. The main issue of this model is that the huge number of nodes required by large fuzzy systems makes it impractical due to its excessive computational load. As described in (Hosoya and Umano 2012), rules can be learned through a reinforcement algorithm such as Q-Learning. This technique creates a table where each configuration is rewarded or penalized according to the fitness function. The configurations that achieve the highest scores will be selected as the best set of rules for the system. In order to reduce the memory consumption, authors of (Lhotska et al. 2004; Navara and Peri 2004) adopt a hierarchical approach to learn fuzzy rules. In our AmI system, the best set of fuzzy rules is obtained by using the TabuSearch heuristic method (Denna et al. 1999), which explores the space of all the possible solutions for a given problem, while prohibiting the repetition of previous moves that would produce a loop in the search path (tabu moves).

We adopt TabuSearch in order to minimize computational costs, while maintaining adequate performance for our system. Recently, several works have successfully used TabuSearch to learn fuzzy systems rules, as shown in (Nguyen et al. 2015) and (Talbi and Belarbi 2013). Other approaches, such as genetic algorithms, often exhibit drawbacks that conflict with our stated goals. The TabuSearch approach has some elements in common with evolutionary techniques, but it clearly differs from genetic algorithms since the *evolution* is achieved by using a single individual, instead of a population of individuals (Denna et al. 1999). The main advantage is that the computational resources and, in particular, the amount of memory required, are much lower in the case of the TabuSearch. Moreover, genetic algorithms continue their search until a previously defined convergence criterion is reached. Thus, several parameters must be carefully adjusted to avoid problems of premature or delayed convergence, which could lead to a poor exploration of the search space. In contrast, TabuSearch is able to continue its search until an adequate solution is found. The cycle prevention mechanism of TabuSearch also ensures that the search space is explored efficiently (Bagis 2008).

We determine the optimal number of fuzzy rules by clustering the sensory data aggregated by the *Environmental Modeler* and the activities detected by the *Activity Recognition Module*, and thus by selecting cluster centroids as rule antecedents. Such selection aims to define at least one rule for each relevant group of environment values and context conditions, by aggregating similar configurations in a single rule and neglecting configurations of input variables which never occur. After such initial process, a different instance of the TabuSearch optimization algorithm is performed to learn the consequent part of each rule. Fig. 5 summarizes the behavior of the *deliberative intelligence* by showing only two sensory dimensions for the sake of simplicity.

A single execution of the TabuSearch algorithm is driven by a fitness function F which evaluates the effect of a given action on the environment. This is obtained by simulating the energy consumption of the actuator in response to the execution of a given rule, and the level of user satisfaction associated with the resulting environment
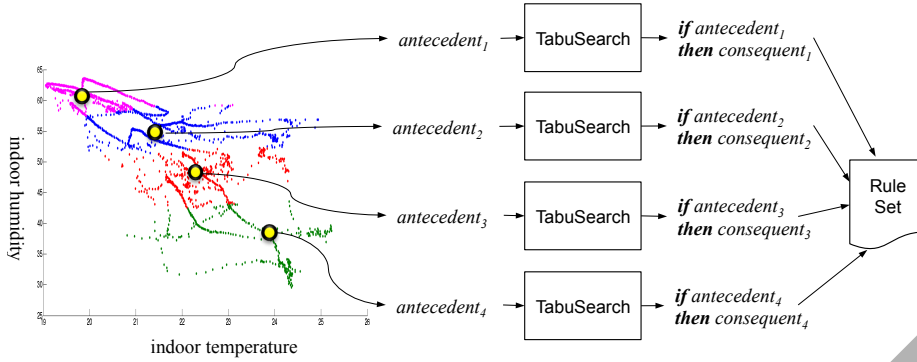
Fig. 5: Logical flow for the Deliberative Intelligence: (i) sensory data are clustered, (ii) cluster centroids identify rule antecedents, (iii) TabuSearch executions find the optimal consequent for each antecedent.

conditions. These results are provided by the *Actuation Module*, the *Environmental Modeler*, and the *User Profiler*. The right balance between fulfilling the user's preferences or the energy saving is established by the system administrator, according to the following equation:

$$F = \alpha * energy\_saving + (1 - \alpha) * user\_satisfaction; \tag{5}$$

where *energy_saving* is a value in the $[0, 1]$ interval, defined as the normalized difference between the upper limit of tolerable energy consumption and the current energy utilization. If α is close to 0, the system gives priority to user's preferences; on the other hand, values of α close to 1 increase the importance of energy saving.

In order to support the exploration of the solution space, the Gray code is adopted to represent the output MFs, as proposed in (Denna et al. 1999). Rule consequents are represented as binary strings and a move corresponds to the complementation of a single bit. A move is admissible if it does not contain repetitions within the last T steps, where the period T varies during the search process and indicates the number of tabu moves. The algorithm adopted to learn a single fuzzy rule accepts as input the rule antecedent and performs N independent searches starting from N random solutions, i.e., consequents.

The deliberative intelligence is able to adapt the fuzzy controllers to specific buildings, user preferences and seasonal trends of environmental phenomena. By periodically repeating the optimization algorithm it is possible to guarantee that fuzzy controllers ever represent the current optimal solution. As shown in the experimental section, rules learned by TabuSearch have proved to be effective for the purposes of our system, and have led to satisfactory performance.

6 Case Study

To evaluate the performance of the proposed system, we present in this section two sets of experiments addressing the scenario of a smart home in which a user carries

out various daily activities. The challenge of our system is to infer the context and ito plan the necessary direct and indirect modifications to the surrounding environment by physically acting on the actuators.

The system takes into account the fact that the user's environmental preferences are not static, and may change depending on the context, e.g., on the current activity. These preferences can cover all aspects of the smart home that are automatically controlled by actuators, including heating, lighting, opening and closing windows and shutters. For example, the system can change the lighting setting and ensure ideal indoor temperature and humidity if the user is doing relaxing activities, such as watching television or reading a book, or vice versa, modify these settings if the user needs to focus on working or studying.

The first set of experiments we performed aims to test the system in a simulated smart home, to analyze its behavior as parameters change. Once the best values for the system parameters have been identified, another set of experiments was carried out in a real smart home. In this case, the system was tested with the help of a volunteer who lived in the smart home for a period of 5 months, leaving it to our system to proactively modify the environment, acting on different ambient properties. In Section 6.2 we will present in detail the setting and the experimental results obtained from this practical experiment.

6.1 Experimental Analysis – Simulated smart home

The first set of experiments seeks to experimentally identify the best parameter values to use in order to maximize system performance. To do this, we analyzed real data which reflects the environment conditions and the user behavior. Sensory traces we used come from the Aruba dataset of the CASAS Smart Home Project (Washington State University) (Cook 2010). We have chosen this publicly available dataset because it is one of the most used in the field of smart environments, and is therefore a guarantee of reproducibility of the tests performed.

The Aruba dataset contains annotated data collected in a smart home with a single resident over a period of 7 months. Sensor events are generated by 31 motion sensors, 5 temperature sensors, and 3 door sensors. The smart home is divided in 8 rooms, with 5 sensors per room on average. The dataset is annotated with 11 activities of daily living (ADLs) performed by the user, such as *sleeping*, *eating*, *working*, and *relax*. Additionally, hourly external temperature data for the entire duration of the experiment come from a real weather station close to the smart home, provided by the AgWeatherNet project of Washington State University (Rasul et al. 2015). Moreover, since the CASAS Project does not provide information about actuators, we enriched this dataset by making use of the *Environmental Modeler* to estimate the impact of the actions performed by the intelligent system (or the user) to change the environmental conditions.

As a preprocessing step, we grouped the original data into time windows of 30 seconds. We also added two more ADLs, i.e., *outside*, which is active when the user is not at home, and *other*, which groups all unlabeled time windows. Four main rooms (bedroom, living room, kitchen, and office) have been studied during experiments. In the following, we assume that each room is equipped with an air conditioner, which can be automatically controlled by the system.

As metric to assess the effectiveness of the proposed system, we introduce the notion of *user satisfaction*. The user is considered *satisfied* if environmental conditions are within his preferred range. Otherwise, if the environmental conditions are outside this range for a certain period of time, the user becomes *unsatisfied* and manually acts on the actuators. As an example, let's assume the user's preferred temperature when working is 22 °C, his tolerance range is ± 2 °C and his tolerance time is 5 minutes. In such a case, if the user is working and the temperature of the room is in the range 20–24 °C, the user is *satisfied*. On the other hand, if the temperature of the room is outside this range for more than 10 minutes, he becomes *unsatisfied*, and directly acts to correct the temperature, by manually turning on (or off) the air conditioner.

At each timestep, the system exploits sensory data to infer the user's activity and position (with room-level granularity), and plan the best sequence of actions to keep the user satisfied, while ensuring energy savings. If the user is unsatisfied with the system's decisions, he can manually override them. Every 24 hours, the heuristic search algorithm is run again to find a new set of fuzzy rules which will be exploited by the reactive subsystem during the following day. Tests have been performed using a cross-validation method, by dividing the dataset into ten parts. For each experiment, nine out of these ten parts have been used to train the activity recognition subsystem, and the last one has been used for evaluating the system. Each experiment has been repeated ten times, changing the test set and averaging the results.

In the following, we will compare our hybrid intelligent system, which automatically plans the sequence of actuator actions to perform, against a baseline *manual* system where the user manually controls the actuators when he is unsatisfied. Then, we will focus on the behavior of the smart system during a given week, by carefully analyzing what happens in a single room when the user is present or absent. Finally, we will study the performance of the system when varying the trade-off between user preferences and energy saving, i.e., parameter α in Eq. 5.

The first set of experiments is run by setting α to 0, so as to prioritize user preferences. Table 1 summarizes the comparison of the *smart* and *manual* systems, reporting results for each test of the cross-validation, along with the averages. For each period, we consider the fraction of timesteps in which the user is satisfied, the *satisfaction ratio*, which is the ratio of the user's satisfaction with the smart system over the satisfaction with the manual one, and the energy saving obtained by the smart system, calculated as $1 - (c_s/c_m)$, where c_s and c_m are the power consumptions of the *smart* and *manual* system, respectively.

Since the smart system tries to save energy by turning off air conditioners whenever possible, we would expect a small reduction of the user's satisfaction and a big reduction of energy consumption, with respect to the manual system. Results reported in Table 1 confirm our insights. On average, the user is satisfied 84.6% of the time with the manual system, and 77.6% of the time with the smart one. The satisfaction ratio is, thus, 0.917. On the other hand, the smart system achieves 35.1% energy savings compared to the manual system. In other words, less than 10% decrease in user's satisfaction allows the system to reduce energy consumption by more than 35%, on average. Moreover, as will be shown in the last set of experiments, energy savings can be pushed as high as 55.3% when setting α to 1, with a satisfaction ratio of 0.736.

Table 1 provides useful insights as well, by reporting detailed results on single tests. The satisfaction ratio reaches its minimum in the third test (17 December – 8 January) and, in general, it is lower during winter months. On the other hands, it reaches its maximum value in the last test (21 May – 11 June) and, in general, it is

Table 1: User satisfaction (*Sat*) of the *manual* and *smart* systems, along with satisfaction ratio and energy savings of the *smart* system, in each test of the cross-validation experiment.

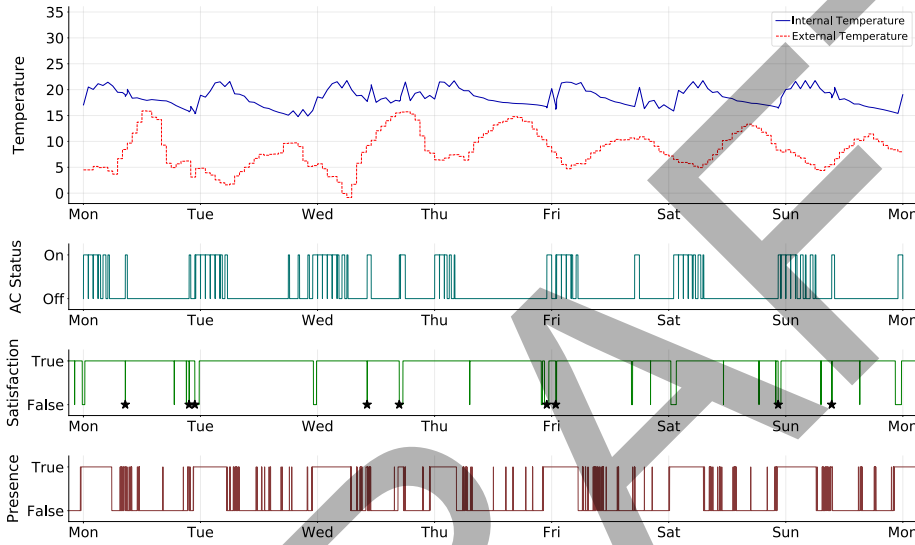| Test dates	Sat (manual)	Sat (smart)	Sat ratio	Energy savings
4 Nov - 25 Nov	85.5%	82.8%	0.968	20.0%
25 Nov - 17 Dec	83.4%	66.2%	0.794	37.5%
17 Dec - 8 Jan	86.2%	66.5%	0.771	27.1%
8 Jan - 30 Jan	83.9%	69.2%	0.825	49.5%
30 Jan - 21 Feb	83.1%	69.7%	0.839	32.4%
21 Feb - 15 Mar	82.0%	62.7%	0.765	41.3%
15 Mar - 7 Apr	84.6%	76.0%	0.898	45.3%
7 Apr - 29 Apr	83.1%	89.8%	1.081	23.0%
29 Apr - 21 May	87.2%	95.1%	1.091	33.8%
21 May - 11 Jun	87.0%	97.8%	1.124	41.0%
Average	**84.6%**	**77.6%**	**0.917**	**35.1%**

Fig. 6: Temperature (internal and external), air conditioner status, user satisfaction and presence in the bedroom during a given week. The stars indicate when the user performed a manual action.

higher during spring months. It is worth noting that the experiment took place in a rather cold environment, thus the different satisfaction values are mainly dependent on the high variability of the external temperature during the year. Indeed, the minimum external temperature registered during the experiment was $-25.17\,^{\circ}$C, while the user's preferred temperature was $22\,^{\circ}$C on average. It is thus understandable that trying to save energy in such conditions is challenging and may result in reduced satisfaction ratio. Surprisingly, the satisfaction ratio is greater than 1 in some of the tests, which means that the smart system managed to satisfy the user better than the manual one, while ensuring remarkable energy savings. Indeed, the user satisfaction in the last three tests is 94.2% (1.098 satisfaction ratio), and the energy savings are 32.6% on average. These results confirm the advantage of the smart system over the manual one, since it
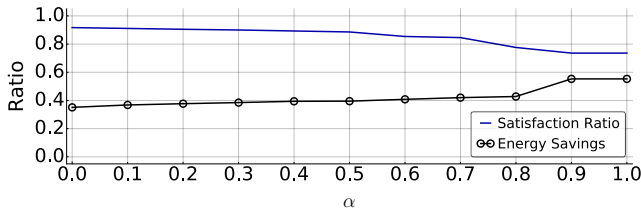
Fig. 7: User satisfaction ratio and energy savings when varying the parameter α.

achieves substantial energy savings in all considered tests, regardless of the extremely low external temperature, while maintaining a very high satisfaction ratio.

In order to better understand how the system achieves such performances, we also conducted extensive experiments to evaluate its behavior in a single room of the house, i.e., the bedroom, focusing on temperature controlling. The topmost graph in Fig. 6 shows the internal and external temperature trend during a week. The other three graphs illustrate, respectively, the air conditioner status, the user satisfaction, and the user presence in the room during the same period of time. Additionally, the stars included in the third graph indicate the moment when the user performs manual actions. The main activity performed in the bedroom is sleeping, which is, unsurprisingly, the most regular activity annotated in the dataset, and it is also the one with the longest average duration. Every day, the user spends the majority of the night in this room, and he seldom visits it during the rest of the day. This regularity makes the bedroom ideal to analyze with ease the behavior of the smart system.

As we can see, the system turns on the air conditioner during night, so as to maintain an ideal temperature while the user is sleeping. Whenever it is possible, however, the system temporarily turns the air conditioner off to save energy. Looking at the point when the user manually turns on (or off) the air conditioner, disagreeing with the system, it can be noticed that these interventions are remarkably infrequent and spaced over during the week, which means that the system was able to maintain the user satisfied while saving energy. Fig. 6 also shows that the air conditioner is automatically turned off in the mornings, and it generally remains off throughout the day, with occasional exceptions when the user spent time in the bedroom during the day (e.g., to do housework).

We will now discuss an example of how fuzzy rules and fuzzy sets are used for decision-making by the system. For the sake of clarity, let's just consider the following rules, which are a subset of those learned by the system during the simulation.

«**if** Activity **is** Sleeping **and** IndoorTemperature **is** VeryCold **and** OutdoorTemperature **is** PrettyCold **then** Mode **is** Hot **and** DesiredTemperature **is** 23».

«**if** Activity **is** Sleeping **and** IndoorTemperature **is** PrettyCold **and** OutdoorTemperature **is** PrettyCold **then** Mode **is** Hot **and** DesiredTemperature **is** 24».

«**if** Activity **is** Sleeping **and** IndoorTemperature **is** PrettyHot **and** OutdoorTemperature **is** SlightlyHot **then** Mode **is** Cold **and** DesiredTemperature **is** 22».

Let's assume that the current user's activity, as inferred from the Activity Recognition system, is *Sleeping*, the internal temperature is 20 °C, and the external temperature is 18 °C. Sensory data and user activity, are fed into the fuzzy reasoning system. The

input variables are fuzzified, depending on the degree of their respective membership functions. Then, the firing strength of each rule is determined, by combining the fuzzified inputs. With the example data, the firing strength of the first two rules is high, and they strongly contribute to the result, while the third rule does not come into play, since its firing strength is practically zero, and it is therefore irrelevant. After the defuzzification, which occurs as described in Section 5, with the CoG method, the system chooses the mode and temperature for the air conditioner, according to the rules with the highest degree of activation. In our example, the mode chosen will be *Hot*, and the desired temperature will be 24 °C.

Finally, we repeated the cross-validation experiment while varying the value of α, so as to fine-tune the ideal value of such parameter. Fig. 7 shows the satisfaction ratio and energy savings obtained by the smart system compared to the manual one, with different values of α. As expected, the satisfaction ratio decreases when α increases, registering its maximum value of 0.917 when α is 0, and its minimum (0.736) when α is 1. On the contrary, energy savings increase when raising α, from a minimum of 0.351 to an impressive maximum of 0.553, which means that the smart system cuts energy consumption in half while maintaining the user satisfied. The ideal value of α heavily depends on the application scenario, and on the relative importance assigned by system administrators to user satisfaction and energy savings.

6.2 Experimental Analysis – Real smart home

The second set of experiments consists of a system run performed on a smart home that was inhabited for 5 months by a single volunteer, from July 2016 to November 2016. To this end, multiple programmable wireless devices are pervasively deployed in the environment, providing heterogeneous information about environmental properties and user's whereabouts. In particular, temperature (indoor and outdoor), humidity, pressure, light, CO_2, motion and noise sensors were deployed in the smart home in a non-intrusive manner. In addition, actuators to control lighting and air conditioners were also installed, and all appliances were monitored by smart meters. Table 2 summarizes the sensors deployed in the smart home during experiments.

The smart home is divided in 5 rooms, with an average of 5 sensors per room. Sensors were placed at strategic locations, such as near windows or on the user's desk, to minimize intrusiveness while maximizing their effectiveness. Each room was equipped with a 9000 BTU air conditioner.

The activities identified by the activity recognition module included *sleeping*, *eating*, *studying*, *watching television*, *reading*, *listening to music*, and *working on the computer*.

As for the metrics used to evaluate the system, user satisfaction has been assessed on the basis of the actions performed (for example, the user is unsatisfied if he takes an action to contradict what the system does).

Based on the previous set of experiments, the best parameter values were selected to guarantee user comfort and compliance with his preferences, while keeping energy consumption low.

Fig. 8 shows the system's behavior in a single room, i.e., the living room, during a particular week (10 October 2016 - 16 October 2016). We decided to show this particular time frame to demonstrate the resilience of the system even in extreme climatic conditions.

Table 2: List of sensors used in the smart home.

Sensor Type	Characteristics
Temperature and relative humidity	Temp. range: $-40\,°C$ to $+123.8\,°C$ Temp. accuracy: $\pm0.5\,°C$ at $25\,°C$ Humidity range: 0 to 100% RH Absolute RH accuracy: $\pm3.0\,\%$ RH
Temperature and pressure	Temp. range: $-10\,°C$ to $+60\,°C$ Temp. accuracy: $\pm0.8\,°C$ at $25\,°C$ Press. range: 400 to 1100 mbar Press. accuracy: $\pm1.5\,\%$ at $25\,°C$
Temperature	Range: $-40\,°C$ to $+125\,°C$ Temp. accuracy: $\pm2\,\%$
Light	Range: 0 to 1847 lx Spectral responsivity: 400–1000 nm
Passive Infrared (PIR) Motion	Sensitivity range: up to 6 m, $110°$ x $70°$ detection range
Noise	Microphone sensitivity: 12.7 mV/Pa Range: 50 dBA to 100 dBA Accuracy: ±0.5 dBA (1 kHz) Frequency range: 20 Hz–20 kHz
CO_2	CO_2 range: 0 to 500 ppm CO_2 accuracy: ±30 ppm
Voltage, current, power factor, active power, reactive power, active energy, reactive energy	Accuracy: 0.5 %

Analyzing the first graph in Fig. 8, we can see that external temperatures on Thursday and Friday are much higher than in the previous days, with a sudden and unexpected growth. In fact, the maximum temperature on Wednesday was $22\,°C$, while on Thursday and Friday it was $31\,°C$ and $34.3\,°C$, respectively. This corresponds to an increase of almost $10\,°C$ in the maximum temperature from one day to the next, which can be very challenging for a system that plan its actions taking into consideration historic temperature data.

Nevertheless, the system has proved to be capable of handling unusual situations (the third graph in Fig. 8 shows that the user is satisfied even on these days of exceptional heat). Moreover, the effectiveness of the system is also demonstrated by the fact that, over the course of a whole week, the user performed only 9 manual actions in the living room, with an average of 1.29 actions per day, and 32 total actions in the entire smart home, with an average of 4.57 actions per day. This average remains constant throughout the tests we have carried out, with an overall average of 4.3 actions per day across the entire smart home, and 1.52 actions in the living room.

For the rest of the time, the system was able to meet the user's needs in a transparent and automatic way, saving as much energy as possible. The second graph in the figure shows that the use of air conditioners has been very limited, at strategic times during the day, maximizing user satisfaction while maintaining low energy consumption. This was made possible by the system's continuous preventive planning, which allowed the system to identify the right moments and ways to dynamically change the surrounding environment, while respecting the user's preferences.
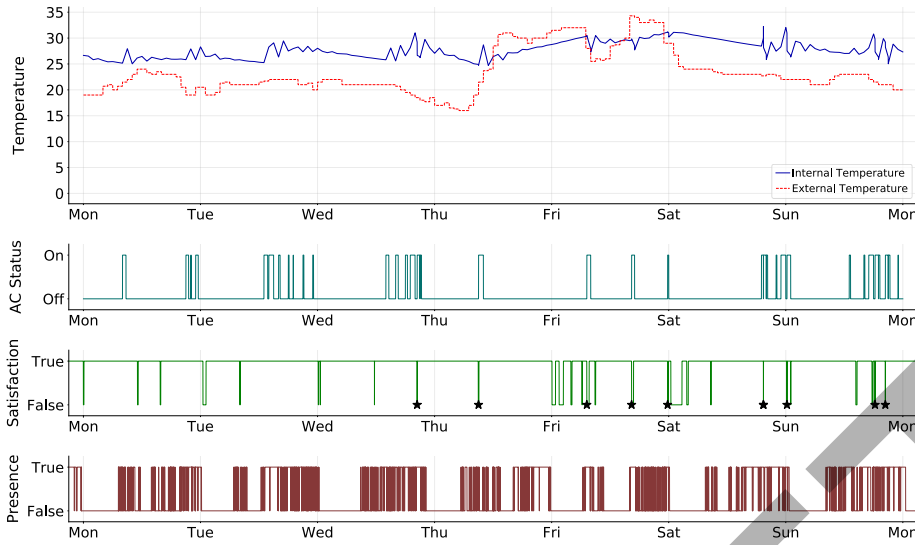
Fig. 8: Average temperature (internal and external), air conditioner status, user satisfaction and presence in the living room of the smart home during a given week. The stars indicate when the user performed a manual action.

7 Conclusion

In this paper, we presented a novel hybrid intelligent architecture which exploits heterogeneous devices pervasively deployed in the environment to reduce the energy consumption of buildings, making them *smarter*, while satisfying users' needs. Our hybrid approach combines reactive and deliberative components to dynamically learn and update a set of fuzzy rules which allow the system to effectively modify the environment according to users' preferences, while ensuring remarkable energy savings.

The effectiveness of the architecture we propose has been demonstrated with extensive experiments that exploit data coming from real sensors located in a smart home to predict the effect of the intelligent system on energy consumption and environment conditions, which directly affect the user's satisfaction. Results confirmed that our system is able to obtain substantial energy savings while keeping the user satisfied.

The main limitation of the current system is its unsuitability for managing multi-user scenarios, where users can influence each other, and individual needs and preferences might be conflicting. This issue could be addressed by explicitly modeling interactions among users, while also taking into account their uncertainty (Bilgin et al. 2012). Another way to solve conflicts in a multi-user environments could be to adopt of a set of priorities which define a hierarchy between users, as proposed in (Kulkarni 2002).

Nevertheless, it is worth noticing that there are many single-user scenarios in which the proposed approach could provide a valuable support for managing smart environments. For instance, the staff of a nursing home could exploit our AmI system to easily monitor elderly people in a non-intrusive way, according to principles of Ambient assisted living systems (Calvaresi et al. 2017). In such a scenario, data coming from the

sensors on the *edge* would reveal the conditions of the physical environment; similarly, ad-hoc activity/medical sensors could provide relevant information to detect alert situations. The availability of fog devices close to the final users would allow to analyze data locally, and act on the environment promptly, even without any human intervention. Conversely, the management of the whole nursing home would take advantage of the long-term analysis performed by the cloud in order to define the energy saving policies, supervise the behavior of the AmI modules, and be aware of the effectiveness of the chosen strategies.

A further limitation of the current system consists in its necessity of collecting initial training data in order to setup fuzzy controllers. Such data are mainly used to determine the fuzzy membership functions of the reactive intelligence, whereas the deliberative intelligence simply relies on data captured during the last 24 hours. In order to overcome this *cold-start* problem, we plan to enrich our system with a further online learning algorithm capable of a dynamic tuning of the membership functions.

Moreover, as future work we will investigate multi-buildings scenarios, trying to understand possible benefits of dynamically sharing and updating system models among different buildings, so as to further improve energy savings.

References

Amadeo M, Molinaro A, Paratore SY, Altomare A, Giordano A, Mastroianni C (2017) A cloud of things framework for smart home services based on information centric networking. In: Proc. of the 2017 IEEE 14th International Conference on Networking, Sensing and Control (ICNSC), IEEE, pp 245–250

Amigoni F, Gatti N, Pinciroli C, Roveri M (2005) What planner for ambient intelligence applications? IEEE Trans on Systems, Man and Cybernetics, Part A: Systems and Humans 35(1):7–21

Atallah L, Yang G (2009) The use of pervasive sensing for behaviour profiling - a survey. Pervasive and Mobile Computing 5:447–464

Baghban A, Bahadori M, Rozyn J, Lee M, Abbas A, Bahadori A, Rahimali A (2016) Estimation of air dew point temperature using computational intelligence schemes. Applied thermal engineering 93:1043–1052

Bagis A (2008) Fuzzy rule base design using tabu search algorithm for nonlinear system modeling. ISA transactions 47(1):32–44

Bhargava K, McManus G, Ivanov S (2017) Fog-centric localization for ambient assisted living. In: Proc. of the 2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC), IEEE, pp 1424–1430

Bilgin A, Dooley J, Whittington L, Hagras H, Henson M, Wagner C, Malibari A, Al-Ghamdi A, Alhaddad MJ, Alghazzawi D (2012) Dynamic profile-selection for zslices based type-2 fuzzy agents controlling multi-user ambient intelligent environments. In: Proc. of the 2012 IEEE International Conference on Fuzzy Systems, pp 1–8

Calvaresi D, Cesarini D, Sernani P, Marinoni M, Dragoni AF, Sturm A (2017) Exploring the ambient assisted living domain: a systematic review. Journal of Ambient Intelligence and Humanized Computing 8(2):239–257

Castellano G, Fanelli AM, Mencar C (2002) Generation of interpretable fuzzy granules by a double-clustering technique. Archives of Control Science 12(4):397–410

Cook DJ (2010) Learning setting-generalized activity models for smart spaces. IEEE Intelligent Systems 2010(99):1

Cook DJ, Das SK (2007) How smart are our environments? An updated look at the state of the art. Pervasive and Mobile Computing 3(2):53–73

Cook DJ, Augusto JC, Jakkula VR (2009) Ambient Intelligence: Technologies, applications, and opportunities. Pervasive and Mobile Computing 5(4):277–298

Corucci F, Anastasi G, Marcelloni F (2011) A WSN-based testbed for energy efficiency in buildings. In: Proc. of the 2011 IEEE Symposium on Computers and Communications (ISCC), pp 990–993

De Paola A, Gaglio S, Lo Re G, Ortolani M (2012) Sensor9k: A testbed for designing and experimenting with WSN-based ambient intelligence applications. Pervasive and Mobile Computing 8(3):448 – 466

De Paola A, Ortolani M, Lo Re G, Anastasi G, Das SK (2014) Intelligent management systems for energy efficiency in buildings: A survey. ACM Comput Surv 47(1):13:1–13:38

De Paola A, Lo Re G, Morana M, Ortolani M (2015) Smartbuildings: an ami system for energy efficiency. In: Sustainable Internet and ICT for Sustainability (SustainIT), 2015, pp 1–7

De Paola A, Ferraro P, Gaglio S, Lo Re G, Das SK (2017) An adaptive bayesian system for context-aware data fusion in smart environments. IEEE Trans on Mobile Computing 16(6):1502 – 1515

Denna M, Mauri G, Zanaboni AM (1999) Learning fuzzy rules with tabu search-an application to control. IEEE Trans on Fuzzy Systems 7(3):295–318

Doctor F, Hagras H, Callaghan V (2005) A fuzzy embedded agent-based approach for realizing ambient intelligence in intelligent inhabited environments. IEEE Trans on Systems, Man and Cybernetics, Part A: Systems and Humans 35(1):55–65

European Commission (2017) Energy efficiency in buildings. URL http://ec.europa.eu/energy/en/topics/energy-efficiency/buildings

Fazzolari M, Alcala R, Nojima Y, Ishibuchi H, Herrera F (2013) Improving a fuzzy association rule-based classification model by granularity learning based on heuristic measures over multiple granularities. In: Proc. of the 2013 IEEE Int. Workshop on Genetic and Evolutionary Fuzzy Systems (GEFS), pp 44–51

Georgievski I, Nguyen TA, Aiello M (2013) Combining activity recognition and ai planning for energy-saving offices. In: Proc. of the 2013 IEEE 10th Int. Conf. on Ubiquitous Intelligence and Computing and 10th Int. Conf. on Autonomic and Trusted Computing (UIC/ATC), IEEE, pp 238–245

Gruber P, Gwerder M, J T (2001) Predictive control for heating applications. In: Proc. of the 7th REHVA World Congress, Clima 2000. Naples, Italy, pp 1–15

Hagras H, Alghazzawi D, Aldabbagh G (2015) Employing type-2 fuzzy logic systems in the efforts to realize ambient intelligent environments [application notes]. IEEE Computational Intelligence Magazine 10(1):44–51

Herrera F (2008) Genetic fuzzy systems: taxonomy, current research trends and prospects. Evolutionary Intelligence 1(1):27–46

Hinojosa CE, Camargo HA (2013) Multi-objective iterative genetic approach for learning fuzzy classification rules with semantic-based selection of the best rule. In: IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), 2013 Joint, pp 292–297

Hosoya Y, Umano M (2012) Dynamic fuzzy q-learning with facility of tuning and removing fuzzy rules. In: Proc. of the 2012 IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE),, IEEE, pp 1–8

Hosseini R, Qanadli SD, Barman S, Mazinani M, Ellis T, Dehmeshki J (2012) An automatic approach for learning and tuning gaussian interval type-2 fuzzy membership functions applied to lung cad classification system. IEEE Transactions on Fuzzy Systems 20(2):224–234

Kulkarni AA (2002) A reactive behavioral system for the intelligent room. PhD thesis, Massachusetts Institute of Technology

Lhotska L, Macek J, Peri D (2004) Evaluation of ecg: comparison of decision tree and fuzzy rules induction. In: European Meetings on Cybernetics and Systems Research (EMCSR), pp 713–718

Nauck D, Kruse R (1993) A fuzzy neural network learning fuzzy control rules and membership functions by fuzzy error backpropagation. In: Proc. of the IEEE Int. Conf. on Neural Networks, IEEE, pp 1022–1027

Navara M, Peri D (2004) Automatic generation of fuzzy rules and its applications in medical diagnosis. In: Proc. of the 10th Int. Conf. on Information Processing and Management of Uncertainty, pp 657–663

Nguyen T, Khosravi A, Creighton D, Nahavandi S (2015) Fuzzy system with tabu search learning for classification of motor imagery data. Biomedical Signal Processing and Control 20:61–70

Nguyen TA, Aiello M (2013) Energy intelligent buildings based on user activity: A survey. Energy and buildings 56:244–257

Perera C, Qin Y, Estrella JC, Reiff-Marganiec S, Vasilakos AV (2017) Fog computing for sustainable smart cities: A survey. ACM Computing Surveys (CSUR) 50(3):32

Rahmani AM, Gia TN, Negash B, Anzanpour A, Azimi I, Jiang M, Liljeberg P (2018) Exploiting smart e-health gateways at the edge of healthcare internet-of-things: a fog computing approach. Future Generation Computer Systems 78:641–658

Rasul F, Rahman M, Shelia V, Hill SE, Islam A, Iosiliani O, Loyd N, Hoogenboom G (2015) Agricultural Weather Network (AgWeatherNet) Science for Society

Sahni Y, Cao J, Zhang S, Yang L (2017) Edge mesh: A new paradigm to enable distributed intelligence in internet of things. IEEE Access 5:16441–16458

Santini S, Romer K (2006) An adaptive strategy for quality-based data reduction in wireless sensor networks. In: Proc. of the 3rd international conference on networked sensing systems (INSS 2006), pp 29–36

Stantchev V, Barnawi A, Ghulam S, Schubert J, Tamm G (2015) Smart items, fog and cloud computing as enablers of servitization in healthcare. Sensors & Transducers 185(2):121

Stojkoska BLR, Trivodaliev KV (2017a) A review of internet of things for smart home: Challenges and solutions. Journal of Cleaner Production 140:1454–1464

Stojkoska BR, Trivodaliev K (2017b) Enabling internet of things for smart homes through fog computing. In: Proc. of the 2017 25th Telecommunication Forum (TELFOR), IEEE, pp 1–4

Stojmenovic I, Wen S (2014) The fog computing paradigm: Scenarios and security issues. In: Proc. of the 2014 Federated Conference on Computer Science and Information Systems (FedCSIS), IEEE, pp 1–8

Talbi N, Belarbi K (2013) Designing fuzzy rule base using hybrid elite genetic algorithm and tabu search: Application for control and modeling. International Journal of Hybrid Intelligent Systems 10(4):205–214

Verma P, Sood SK (2018) Fog assisted-iot enabled patient health monitoring in smart homes. IEEE Internet of Things Journal PP(99):1–1

Yan Y, Su W (2016) A fog computing solution for advanced metering infrastructure. In: Proc. of the 2016 IEEE/PES Transmission and Distribution Conference and Exposition (T&D), IEEE, pp 1–4

Zadeh LA (1975) The concept of a linguistic variable and its application to approximate reasoning. Information sciences 8(3):199–249