DISSERTATION

Smart Home Control Based on Behavioural Profiles

Submitted at the Faculty of Electrical Engineering, Vienna University of Technology
in partial fulfillment of the requirements for the degree of
Doctor of Technical Sciences

under supervision of

ao. Univ. Prof. Dipl.-Ing. Dr. techn. Wolfgang Kastner
Institute of Automation
Vienna University of Technology, Austria

and

O. Univ. Prof. Dipl.-Ing. Dr. techn. Dietmar Dietrich
Institute of Computer Technology
Vienna University of Technology, Austria

by

Félix Iglesias Vázquez
Matr.Nr. 1126854
Grundsteingasse 7/1/8-9, 1160 Wien

Date ______________________

Kurzfassung

Systeme der Heim- und Gebäudeautomation sind durch eine hohe Interaktion mit Verbrauchern charakterisiert, wobei den Begriffen Ubiquität und Kooperation ein wesentlicher Stellenwert zukommt. Die vorliegende Arbeit postuliert aus einer holistischen Top-Down-Perspektive eine Methodologie für den Entwurf neuer adaptiver Smart Homes, in denen die Verbrauchergewohnheiten als wesentliche Einflussgrößen für Dienste der Heimautomation verwendet werden, um eine generelle Verbesserung der Gesamtperformance zu erreichen. Der Fokus auf Verbrauchergewohnheiten ermöglicht zudem, pro-aktive Kontrollstrategien zu entwerfen, die eine bestmögliche Anpassung von Heimautomationssystemen an den Verbraucher erlauben.

Die Arbeit stützt sich eingangs auf vorhandene Literatur, um eine Taxonomie von Verbraucherprofilen herzustellen, die grundlegend für den nachfolgenden Gesamtentwurf des „Smart Home Control"-Konzepts ist. Im nächsten Schritt werden potentielle Integrationsszenearien in existierende Heimautomationstechnologien beschrieben und Synergieeffekte mit größeren Anwendungsbereichen im Kontext von Smart Cities gezeigt. Der gewählte Ansatz wird durch die Beschreibung einer Menge von profilbasierter Anwendungen untermauert, die die Hauptfunktionen eines automatisierten Heims abdecken. Davon werden Luftqualität und thermische Behaglichkeit für eine genauere Analyse herangezogen, Algorithmen und Strukturen entwickelt und im Detail diskutiert.

Der Fokus des zweiten Teils der Arbeit liegt auf der Ausarbeitung und Analyse von profilbasierten Kontrollstrategien. Unterschiedliche Modelle werden vorgestellt, die die Herleitung des Verbraucherverhaltens, eine Abstrahierung von Gewohnheiten und letztlich deren Kontext-spezifische Interpretation ermöglichen sollen. Dieser Herausforderung wird mit Hilfe von Clustering-Prozessen begegnet, die sich auf Techniken des Soft Computings stützen, jedoch ein grundlegendes Verständnis von und eine enge Anbindung an die dahinterliegenden Anwendungen erfordern.

Die Evaluierung der vorgestellten Konzepte erfolgt mittels mathematischer Analyse und unter Zuhilfenahme von Simulationen. Zugleich werden Anwendungsregeln abgeleitet. Als wesentlicher Bestandteil der Arbeit wurde ein neuartiger Algorithmus, genannt Exclusive Self-Organizing Map (XSOM), als Suchtechnik zur Mustererkennung und zur Kontextinferenz entwickelt. Die Ergebnisse zeigen, dass dieser Ansatz für den gegebenen Anwendungsbereich hervorragend geeignet ist.

Abstract

Automation of homes and buildings is characterized by a high casuistry and a strong interaction with users. Here, concepts like ubiquity and cooperativeness acquire an undeniable importance. From a top-down, holistic perspective, this thesis states a methodology for the design of adaptive Smart Homes where users' habits are considered as context elements able to join home applications, services and technologies in a sound overall performance. Furthermore, the focus on users' behaviours allows the exploration of persuasive, proactive designs oriented to improve system adaptiveness.

The work relies on the literature as a starting point for the establishment of definitions and taxonomy for behavioural profiles. It yields to the design of an overall smart home control concept based on habit profiles, describing the integration within current home automation networks, and also considering synergies with regard to wider scopes on the way to smart cities (e.g. building calculations, urban planning, etc.). The proposal gains soundness by the description of a set of profile-based applications that cover main functionalities of automated homes. Among them, the use cases "air quality" and "thermal comfort" are selected for a deep analysis, and algorithms and structures are developed and described in detail.

The second part of the study focuses on the accurate elaboration of profile-based controllers. It requires dealing with complexities related to users' behaviour modeling, habit abstraction and context interpretation. Such demanding tasks are faced by clustering processes grounded by Soft Computing techniques, advanced solutions that entail a careful design and parameterisation in a close relationship with the peculiarities of the intended applications.

The diverse proposals and discussions offered throughout the dissertation are evaluated by means of mathematical analysis and comparisons carried out in a test-bed simulated environment, culminating in a set of generalisations and rules of application. In addition, the Exclusive Self-Organising Map algorithm (XSOM) has been developed as a searching technique for pattern discovery and context inference that perfectly fit the requirements of the field under study.

In this dissertation, the holistic habit-based smart home consolidates as a promising approach mainly due to the following features: it enhances the control of isolated applications; it grounds global, coordinated management, improving thereby the overall performance; finally, it allows the development of high user-sensitive environments.

Acknowledgements

I am really glad to have the opportunity to express my sincere gratitude and respect to the persons who have significantly contributed to the accomplishment of the present work.

Firstly, I would like to say thanks to Prof. Jordina Vidal Moreno. She is a model from whom I learnt the meaning of being a true professional, far from any appearance or superficiality.

Prof. Dietmar Dietrich is to be mentioned. His expert advice has provided an injection of soundness to this dissertation.

Very special thanks to my friend Prof. Sergio Cantos. We enjoyed working together a lot. Some of the main ideas developed here originally belong to him.

Finally, I owe the highest appreciation to Prof. Wolfgang Kastner. His guidance and contribution have been exceptional. Beyond that, I have no words to express how important his kindness, trust and support have been during my long winters in Vienna.

I would also like to thank Laura Macdonald for her valuable help improving the wording and quality of my work.

Table of Contents

Abbreviations

AAL	Ambient Assisted Living
AmI	Ambient Intelligence
AQTC	Air Quality and Thermal Comfort
DSM	Demand Side Management
DHW	Domestic Hot Water
GUI, UI	(Graphical) User Interface
HBA	Home and Building Automation
HAS, BAS	Home, Building Automation System
HVAC	Heating, Ventilation and Air-Conditioning
IT, ICT	Information (and Communications) Technology
KB	Knowledge Base
MAS	Multi-Agent System
PMV	Predicted Mean Vote
PFG	Profile Generator
PTG	Pattern Generator
TC	Technical Committe
SC	Soft Computing

Standards Bodies :

AENOR	Spanish Association for Standardization and Certification
ANSI	American National Standards Institute
ASHRAE	American Society of Heating, Refrigerating and Air Conditioning Engineers
CEN	European Committee for Standardization
CENELEC	European Committee for Electrotechnical Standardization
ETSI	European Telecommunications Standards Institute
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
UIT	International Telecommunication Union

1 Introduction. Why *smart home control based on behavioural profiles*?

The complex nature of home-related research is progressively turning domestic technologies into a field of study of its own right. Within this scope, the *smart home* becomes a core concept that entails an overall control perspective in keeping with the paradigms of Pervasive Computing. The focus on users' behaviours offers an answer to the detected lack of design with a greater awareness of usability and the real context where families live, but also to find underlying and common elements that help to build bridges between applications. In short, the final aim is the optimisation of the automated house as a whole by improving the cooperative performance of services. In addition to enhance comfort and energy savings at home, the benefits of profile-based designs spread out in the scopes of building calculations and community services.

In this chapter, the scope of the current dissertation is introduced, describing the motivations, the planned objectives and presenting the deployed methodologies.

1.1 Background

By the end of the 1990s, the *smart home* concept timidly appeared to identify a heterogeneous group of technological applications and services embraced by the home environment. There are some other terms that refer to the same ideas with apparent changes of focus or nuances; e.g., smart house, home automation, integrated home systems, home technologies or domotics. The fact that even experts are not completely clear about these nuances or the definition of the terms themselves is striking (such questions are usually smartly dodged or surrounded in the related literature).

In part, this confusion comes from – and also causes – the classic lack of personality that the smart home issue has shown in the technological and scientific world so far. It also reflects the heterogeneous nature of the field, certain existing disorder and the necessity of unicity and normalisation that would pave the way to transform it from a multidisciplinar field of technology application and research [Hin99] into an optimised inter- or even transdisciplinar area.

From the origins of the first attempts or ideas to merge houses and technology to now, some lacks, difficulties and constraints have been holding back the logical evolution of the application of technology at home, in a way that the current status does not fulfil the expected level and quality of integration and development. In the pertinent book *Inside the Smart Home* [Har03], the causes are justified as follows:

> They (smart homes) have not been a hit because they have been too expensive, the housing stock is old, there has been a tendency for little networked connectivity, and finally, there has been too much technology push, and little attention given to users or usability.

Afterwards, the neglect regarding usability is explained as the result of no motivation to increase productivity at home, little involvement of users in designs, the view of domestic technology as unexciting and a continued focus on stand-alone appliances. In that respect, social scientists did not show much interest until recently. Nowadays they contribute with some criticism regarding the exclusively technical focusing, as well as remark upon the commonly forgotten attention to users, usability and the social circumstances in which home services occur [DLY+06].

Hence, a first and quick answer to the question that heads this chapter could be as follows: because the current design of smart homes is not leading us to an optimal reality. As far as the smart home application is concerned, we consider that this *optimal reality* would be reached in an extended situation where smart home systems are fairly integrated in the domestic environment and contribute to the improvement of sustainability and users' quality of life. This is in keeping with the technological resources of the society where users live. Nowadays, it is even difficult to guarantee that we are approaching to this ideal scenario.

Ambient Intelligence (AmI) principles are set to pursue the exposed goals and establish the frame where future smart homes must be developed. The AmI paradigm is supported by pervasive computing and profiling practices, and focuses on human-centric computer interaction design. It results in systems and technologies that aspire to be identified with the next common features: integration, context awareness, personalization, adaptation and anticipation [AE06].

It seems to draw a promising scenario for enhancements in comfort, security or safety, and also to fulfil sustainability aims (good examples can be seen in [CTGMM+09]). Indeed, the issue regarding energy efficiency is quite urgent. Not in vain, the European Commission adopted measures that should put Europe on the path to reduce its global primary energy use by 20% until 2020. These measures include on the one hand, rapidly improving the energy performance of existing buildings and on the other hand, taking the lead to develop a strategy for very low energy houses or passive houses [Com06].

For our field of interest, such concern demands not only applications oriented to minimise the energy consumption, but also to boost a quick spread of home technologies (which even in the most basic solutions usually entail benefits compared with old-standard, non-sustainable dwellings). Smart home designs must consider – at least at the same level of importance that the characteristics of AmI systems – aspects like affordability and simplicity. It means that they must be scalable and flexible, being acceptable, as much as possible, for the real residential sector nowadays.

In this respect, it is worth noticing that most of the current smart home designs and researches is mainly focused on new and high standard dwellings, involving expensive technologies and applications, and retaining experimental and futuristic connotations [BCC08; SMD+08].

These aspects introduce the necessity of a sensible design methodology for smart homes, not bound to any specific technology, application or system, which faces the problem from an overall perspective. Thus the smart system as a concept, regardless of the pieces that made it up, can guarantee that such pieces will cooperate to reach an optimised performance.

1.2 Holistic Control and Behavioural Profiles

The requirement of a coherent focus from top-down and holistic perspectives contrasts with the habitual scenario nowadays, where countless, sometimes glued, bottom-up solutions coexist. In any case, top-down designs are already seen as important steps towards the creation of intelligent systems in complex fields, specifically like Home and Building Automation (HBA) [DFZB09]. To that end, the *user behaviour-based* approach faces the high heterogeneity stating design guidelines on the basis of a priority attention to the relationship between users and the home environment as a whole. Unlike other technology applications where the main aim is the optimisation of a process, here it consists of the optimisation of a global experience. Therefore, the attention is primarily focused on the user, and the home – or the smart home – is considered as the context where such experience must be improved. In line with this conception, the user *behaviour* is seen as "the way in which *one* acts or conducts oneself [in the domestic environment]", becoming the object of study – and objective at the same time – for the automated control. Thus we face a technological challenge where psychological and sociological aspects acquire an unquestionable relevance.

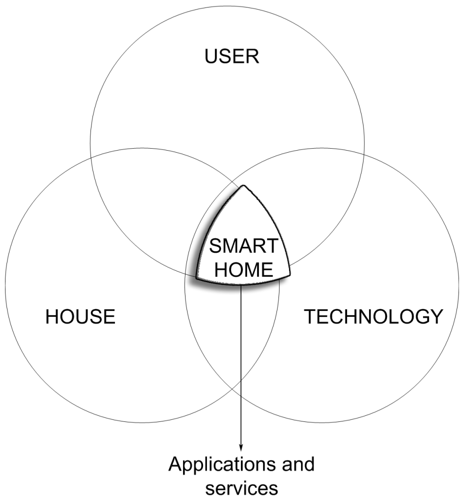


Figure 1.1: The Smart Home concept.

It is worth establishing in this point what the 'smart home' concept will mean for us from now on. Among others, we prefer this denomination due to the subjective connotations related to users' impressions and the supposed overall running. So, a *smart home* is considered as the conceptual abstraction of a domestic technological system that integrates and manages in a holistic manner the interaction between users, the dwelling and technology through services and applications (Figure 1.1). Additional discussions concerning smart home terminology are introduced later in Section 3.3.

Obviously, common smart home applications react to users' actions and commands, but rarely *realise* or *think* about the context in which interactions happen. If the system merely executes a programmed routine after punctual stimulation (caused by users or other agents), a big amount of potentially useful information is being obviated. To be aware of relationships, dependences, overlaps and transients among services, appliances and users is not an easy task. It usually overcomes the competences of specific applications, so it must be in hands of high-hierarchy management modules or agents. These agents represent the smart home overall concept and rule upon specific routines, devices and machinery in what we define as a pre-control phase or preparation management.

Considering the house itself as a static layer, users – or more specifically users' behaviours – draw the dynamic, continuous and common thread where smart control is executed. Beyond isolated or specific users' requirements, the way of being aware of users' behaviours is by means of the abstraction and understanding of their behaviours and the use they give to home elements, spaces and devices. When behaviours are repeated throughout time, we speak about *habits*; and a way of capturing and storing habits in a digital context is carried out by means of *patterns* or *profiles*.

Therefore, *habit profiles* become objects of global reasonings for overall top-down approaches and elements shared by different applications (Figure 1.2), unlike concrete inputs for the solution of specific services (they remain in a second operation layer). In short, smart systems deploy habit profiles to get context awareness regarding behaviours and use and, by means of context awareness, they pursue to capture the essence that makes the home to be experienced as a fluid and balanced environment.

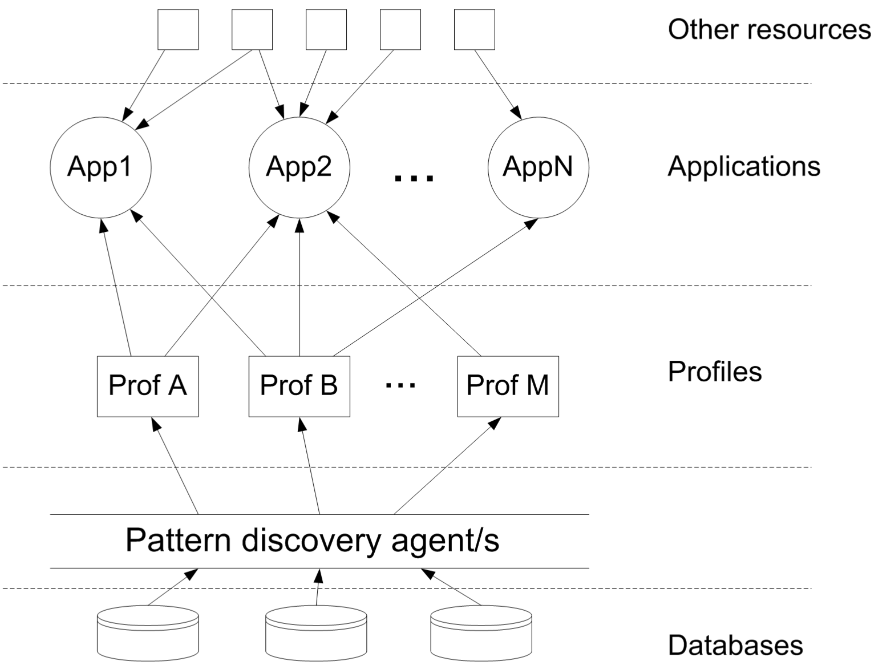


Figure 1.2: In smart home control, habit profiles can be designed as shared resources at disposal of the multiple home applications and intended to capture the context concerning users' behaviour.

Here, it is possible to sketch a sporty analogy for the sake of clarity. If specific home services and appliances are seen as football players, the smart home system as the coach and users as the rival team, irrespective of the fact that each player knows perfectly well how to fulfil its own tasks, the coach is necessary to optimise the global performance and *make them play together* in the best way according to the current match. In addition, if the way the rival team acts is well-known by the coach and the players, the next victory is at least closer than otherwise.

However, the abstraction of this *way of acting* is not a trivial matter. Modelling human behaviour entails several sources of uncertainties and to avoid some level of subjectivity is not possible

[FMMCM05]. To face this situation the utilisation of Soft Computing (SC) techniques is required, as they are intended to deal with imprecision, uncertainty and partial truth to achieve tractability, robustness and low solution cost [DO01]. Among these techniques, the introduced approach needs algorithms able to discover patterns and representatives hidden by high amounts of data collected in the home environment. Some clustering techniques are suitable to manage such imprecise situations, obtaining valid representatives and, in addition, evaluations about the level of validity and soundness of the obtained solutions.

On the other hand, beyond *understanding* users' habits and behaviours to improve systems' efficiency, the other side of the proposed approach explores the capabilities of technology to influence in habits; i. e. the possibility to *cause* changes in behaviours and thus contributing to reach home goals, mainly sustainability and comfort. Therefore, if in the first case *behaviours are seen as basis/inputs* for the system, now *behaviours are seen as goals*. This complementary point of view leads the smart system to consider users as collaborative elements, so *user knowledge* is taken as an aspect to evaluate and indirectly improve publishing information at the correct place and moment [Int02]. The end objective of such design concept is helping users to ameliorate their own behaviour (e.g., energy, health, lifestyle) and control the environment by themselves. This supporting perspective is also necessary to compensate or counteract the lacks and conflicts derived from the context-awareness inference limitations, as there are human aspects in the context that cannot be sensed or even inferred by technological means [BE01].

Answering again the starting question, we conclude that the focus on user behaviours satisfies necessities observed in current home automation designs. The profile-based orientation looks for a better integration of home technologies in daily life, to optimise the efficiency of the system as well as to remove control complexities and facilitate the achievement of sustainability and other goals assumed in the home environment. In addition, the approach introduced in this work sustain designs, schemes and networks set to obtain performance optimisation in urban and social infrastructures, leading to the realisation of smart cities.

1.3 Goals and Methodology

The main **objective** of the dissertation is *to study the suitability of behavioural profiles for smart home control*. This is achieved by the accomplishment of several subgoals listed as follows:

1. To propose formal definitions and a taxonomy for behavioural profiles in HBA.

2. To analyse the requirements of smart homes in order to achieve designs that look for a better integration and adaptation of home technologies.

3. To design conceptual structures to build up holistic, profile-based approaches for smart home control and management.

4. To propose a set of profile-based services which cover common functionalities expected in smart homes.

5. To deeply depict the implementation of a set of selected profile-based applications within cooperative structures intended for overall control.

6. To check the benefits of profile-based applications in terms of energy and comfort performances.

7. To establish guidelines and precedents for the design of profile-based controllers able to satisfactorily read the home, behavioural context.

8. To study the characteristics of pattern discovery methods (clustering) for context interpretation that fit the requirements of the proposed profile-based methods and applications.

The conducted **methodology** includes an exhaustive revision of the related literature concerning the different topics covered by the thesis. The study of the state of the art yields to the proposal of conceptual designs of methods and applications. Later on, proposals are checked by means of mathematical analysis and performance calculations executed under a test-bed environment for home and building simulation.

1.4 Arrangement of the Thesis

Chapter by chapter, the work is arranged as follows:

– Chapter 1 –
The current chapter, where the topics of the dissertation are introduced, and goals and methodologies are presented.

– Chapter 2 –
A revision of the literature with regard to profiles in the domestic environment is the starting point of the chapter, describing uses and applications up to now. Later on, the usefulness of home profiles beyond the scope of smart homes is explored; additional deployments and synergies are found for building designs, regional benchmarks, community applications, service providers, etc.

The lack of dominant methodologies for profile-based approaches requires the proposal of some definitions and taxonomies. They establish a simple basis to build sound applications around such concepts. Before going further, we find it necessary to justify and validate the underlying principle that gives sense to the whole work; i.e. the fact that *people keep habits at home*. It is carried out by the mathematical analysis of use databases collected from real buildings. The last part of the chapter shows a detailed description regarding the integration of profile techniques in habitual HBA networks. This is a practical requirement for a deeper understanding of the approach, moving us from a conceptual to an application framework.

– Chapter 3 –
Chapter 3 merges habit profiles and holistic control. We start studying current regulations, standards ans also previous works which face the control of smart homes from global perspectives. Such exploration culminates in the detection and explanation of the main objectives and foci aspired by smart home designers; the fact that helps us to state the boundaries and goals of our research. The next step is to sketch an abstract model to describe the interaction between the users and the smart home, and to identify the role that habit profiles play inside this model afterwards. Later on, a possible, representative system architecture is depicted, yielding to the presentation of a set of profile-based services that covers the classic requirements of smart houses. Finally, the case of a single apartment is given as an example to assess how some of the introduced services are embedded and work together.

– Chapter 4 –
Among the set of profile-based applications already introduced, services concerning air quality and thermal comfort management are selected to be explored in depth. Algorithms, agents and structures are widely depicted, and controllers based on profiles are described and tested in simulated environments where they are compared with classic control strategies.

– Chapter 5 –
Control strategies based on profiles depend on the effectiveness of advanced searching tools in the task of discovering habit patterns from collected data. The chapter deals with this problem, and clustering methods are presented as the right technique to discover habit patterns and provide context inference. However, in order to reach satisfactory results, clustering tools have to face multiple sources of uncertainty related to the clustering process itself but also to the intended application, i. e. modelling human behaviour in the home environment. The influence of configurations and methods – which must be selected during the design of the diverse phases of the clustering task – are thoroughly discussed, looking for the best options and solutions for our application scope.

– Chapter 6 –
As a continuation of Chapter 5, this chapter checks diverse aspects – introduced previously – that are submitted to uncertainty. In addition, it carries out tests and comparisons among different clustering techniques. The techniques are evaluated by means of clustering validity methods, representativeness analysis and assessing their performances in simulated environments. To discover and consolidate which features make clustering tools appropriate for home automation applications is the main goal of this part of the dissertation. Therefore, a set of findings derived from the executed analysis are listed at the end of the chapter.

– Chapter 7 –
Finally, the conclusion chapter picks up the subgoals which appear throughout the work, evaluates their accomplishment and exposes them in a common outlook.

– Appendix A –
Appendix A describes the XSOM (eXclussive SOM), a clustering technique based on Self-Organizing Maps that incorporates outlier detection and removal. It is specially designed for habit profile discovering and shows excellent results in the developed tests and simulations exposed in Chapters 5 and 6.

– Appendix B –
Appendix B depicts in detail the simulated test-bed environment utilised throughout the dissertation for comparisons and evaluations. Methods, strategies and performance indexes are also described.

2 Behaviour Abstraction by Means of Profiles

In this chapter, the concept of *habit profile* is introduced. Some definitions are presented, focusing on aspects related to the home and building environments. Within this scope, we comment on some projects, works and researches that utilise behavioural patterns or profiles. Later on, the existence of home trends, i. e. habits, is checked by mathematical analysis. The last part covers the integration of profile-based systems under common HBA networks.

2.1 Home and Building Profiles – Related Work

The relationship between people and the places where they live and carry out their normal activities has been the object of many works and studies. From the technological point of view, the aim is to optimise this relationship by means of technological applications.

From now on we differentiate two fields according to the deployment given to profiles. On one hand, profiles are utilised in *design phase* analysis. It normally includes calculations and optimisation in the design of buildings, systems and machinery. In this kind of application, profiles usually represent the behaviour of communities or a large population; i. e. how most people behave or use an object or space. Within this group, a representative application is the use of profiles as inputs for *building energy performance simulations and calculations.*

On the other hand, profiles are useful at *runtime or use phases* in order to obtain dynamic capabilities that guarantee a suitable adaptation to changeable scenarios. Therefore, profiles make management and control systems flexible and predict particular situations. In this case, profiles are not intended to abstract what most people usually do, but what *a specific user* (a single, a family or a group of persons that live or use the same facility) usually does. This dissertation mainly focuses on this kind of profiles, specifically with regard to *smart home* systems.

Boundaries between both fields are sometimes vague, and the same profiles can be used in design phases and control phases likewise. Despite the fact that we deal mainly with habit profiles for control purposes, we must also keep in mind the use of profiles in design phases, e. g. facility and equipment sizing, simultaneity coefficients, services based on community habits, elaboration of models and benchmarks, calculations for buildings (Section 2.1.2).

2.1.1 Smart Homes

Similarly to the concept of smart home (see e. g. [Har03] or [NAA10]), behavioural profiles have been utilised by home and building technologies without adapting to any canonical definition, being the features of each specific profile built over the requirements of the intended application.

In any case, in smart homes the deployment of profile techniques are mostly dedicated to reduce energy consumption and enhance comfort conditions. In the daily-life scenario of a house or a building, if the smart system is aware of inhabitants' habits, it is able to predict future events and act ahead [BBCM09]; so the response of controllers is optimised.

Focusing on control purposes, profiles or patterns have been utilised to directly abstract home activities or lifestyle – e. g. eating, sleeping, etc. – [MTIS04]. By means of activity abstraction, being aware about unexpected changes in repeated habits, it seems to be possible to infer user feelings [TTIH06], predict future services in intelligent homes [KS09], or detect possible accidents or situations that can require special attention [MFS+07].

Simpler patterns have also been deployed to monitor the occupancy and functional health status of elderly people [KC07]; or discover actions that a smart home agent should automate based on the repetition of device usage [HC03]. Combining occupancy profiles and room energy usage profiles, the energy consumption of the house can be optimised and reduced [BBCM09].

Indeed, occupancy profiles are the most required type of profile *de facto* for home control purposes. For instance, together with hot water usage profiles and profiles that store the likelihood that a zone will be entered in the next few seconds, they have been employed to control temperature, light, ventilation and water heating [Moz98]. Moreover, occupancy patterns improve Heating, Ventilation and Air-Conditioning (HVAC) control and energy optimisation [TRF07] and, together with lighting usage profiles, they can result in behavioural models for the automated adjustment of the lighting level [BRM06].

Complex structures and derived patterns for the user behaviour analysis and future service optimisation can be appreciated in [HJO06], where users' behaviour and home context are related based on monitored tasks and activities. Also in [DCB+02], complex patterns store movement history and actions, thus patterns participate in later predictions for keeping inhabitant comfort with operation cost minimization.

The introduced works present several ways to *understand* home profiles, and the list is extensive. To sum up, habit profiles have been found appropriate to improve applications related to the main pillars of HBA, i. e. comfort, energy consumption, HVAC, security, safety, informativeness, etc. Sometimes profiles are used to accomplish specific goals, but they are also seen as objects shared by different services, or elements that support global reasonings.

Note that the purpose of the presented habit profile applications is mainly dedicated to predictive control for smart homes, i. e. to anticipate a near future phenomenon and automatically act ahead. Still inside the smart home scope, habit profiles can also be seen as behaviour summaries to be offered by proactive applications whose goal is to inform and make users aware of their own trends [Int02]. In this regard, persuasive applications are discussed in Section 3.6.

2.1.2 Smart Buildings, Communities and Cities

From the building perspective, profiling techniques are mostly related with energy usage optimisation and sustainability. The exploitation of such technological solutions responds to environmental concerns that have pushed governments around the world to demand innovative solutions to improve the energy performance of future but also existing buildings [Com06].

As a general rule, in the built environment, energy profiling usually refers to the analysis of the actual or predicted energy performance of buildings. It habitually involves assessments of energy consumption and related carbon dioxide (CO_2) emissions [JFB97] [MC97]. Note that, from the perspective held in this work, both are types of habit profiles as they are the indirect result of uses carried out by users.

The value of energy profiling techniques to optimise the whole life-cycle of buildings and to contribute to the assessment of sustainable urban development is studied in [CDD10]. Indeed, a noteworthy aspect is that research and interest given to energy profiles covers different ranges, from individual buildings [DPIP07], to building types [GBM+07] [RRK08] and also entire localities [YS05]. This characteristic highlights the inherent scalability of profile applications, a fact that also starts to reveal the potential, synergistic benefits that profiles can entail for multiple stakeholders at different levels.

Just considering only building energy profiles would be a regrettable mistake. Collected energy profiles are useful benchmarks but, usually, what building energy performance tools actually do is to obtain energy profiles as outputs from their calculations. Some determinant inputs of such calculations are aspects more directly related with the users' behaviour, e. g. occupancy, equipment usage, lighting habits, etc. For example, in [NS07], where the feasibility of building simulation tools for predicting the energy demand of an office building is studied, authors find that the major sources of uncertainties have to do with the evaluation of lighting, equipments and occupancy schedules, concluding that the users' behaviour can significantly affect the energy consumption profile, making its forecasting more difficult or inaccurate (in case we do not have such information). Similar conclusions appear in [HHL+09], with authors' words: "The simple approach used nowadays for design assessments applying numerical tools are found inadequate for buildings that have a known close interaction of the user with the building". Also in [SP07], authors complain about the habitual lack of input behavioural data for simulations (that most of the time must be riskily presupposed), a fact that detracts from the credibility and accuracy of the outcomes. In this respect, in [CM07], diverse analysis corroborate that behaviour data is the dominant parameter adding uncertainties in building calculations.

Indeed, habit profiles are the basis for the elaboration of benchmarks and surveys for later building analysis, comparisons and modelling [HE09]. As we have seen, they become really important for architects and building designers, otherwise they are forced to make decisions just according to their experienced knowledge or trusting in generic constants, tables or values that often do not correctly match the real scenario that they have to design. Thus databases with realistic and representative habit profiles are highly relevant to obtain accuracy in building energy calculations [Mil04], and that is obviously reachable in a future where the collection of habit data in homes and buildings be a common trend. In the outstanding work by Crawley et al., where the features and capabilities of twenty major building energy simulation programs are compared, the convenience of feedbacks between users and program developers as well as available simulation inputs is earnestly stressed [CHKG08].

In the same line, there are several works that can be cited. For example, in [BRM06], in addition to using predicted behavioural models for control, the impact of user habits on the energy performance is estimated, emphasizing their importance. Furthermore, building occupancy is detected as one of the key issues to assess inhabitants' trends in [PRMS08], stating the direct consequences for the energy consumption. In [MP09], authors conclude that behavioural trends and patterns can be extracted from long-term observational data, and affirm that generalisation and models of people habits are important inputs to predict how the energy performance of a future building will be. Their own words are worth quoting: "The reliability of results obtained from building performance simulation applications depends not only on the validity of computational algorithms, but also on the soundness of input assumptions".

Widening the scope, we can see how habit profiles are highly suitable in other fields, e. g. for control processes, but also for the design of components, systems and subsystems. A good example is found in energy distribution and supply services, i. e. smart grids, where a good knowledge of the users' behaviour entails benefits and optimisation, paving the way for customised services [KSP+10] and fairer billing [ZASZ+10]. The advantages of behavioural profiles for smart grid simulations are also remarkable [LVK10]. Furthermore, usage profiles allow the accurate design and calculation of simultaneity factors in multiple domains; for example, to improve the Demand Side Management (DSM) on the planning and operation of distribution energy grids [GSR10] [SNMN10], or to ameliorate the design of Domestic Hot Water (DHW) distribution systems [Buj10]. Also the evaluation of users' behaviour is capital for the design and control of systems intended to manage the thermal comfort of buildings [BD98].

Finally, behavioural profiles can also be profited for urban design. As it is pointed out in [HBEK02], "the link between the built environment and human behaviour has long been of interest to the field of urban planning", however they also add later: "but direct assessments of the links between the built environment and physical activity (...) are still rare in the field."

In short, the introduced works are some of the many examples that highlight the multiple way of exploitation of users' habits information. From here, it is not difficult to imagine future Information and Communication Technology (ICT) frameworks where habit information is extensively shared, interchanged and profited. Indeed, comprehensive, highly dynamic ICT frameworks are indispensable components envisioned for future smart cities [KH09]. Figure 2.1 sets out a future scenario with different business/service relationships supported by the profile-based smart home. Note that, within such schema, smart homes acquire new roles as information prosumers (consumers and providers).

For instance, in order to sketch some quick examples of such relationships, it is possible to imagine smart houses facilitating information about consumption and use profiles, whereas energy providers use this information to optimise their predictions as well as to offer customised billing to the connected clients. In addition, energy providers, based on the precise knowledge obtained concerning their clients, can offer additional proactive control in the form of advices or warnings, or even directly if consumers agree (e. g. coordinating secondary or deferrable loads). At the same time, other service providers (e. g. home and building information managers) after collecting data from large amounts of linked houses and buildings, can perform summaries, assessments and benchmarks (with different scopes: neighbourhood, community, region, country, etc.) and return them to smart homes for control and informative purposes. On the other hand, as we referred to before, architects, building designers or construction companies are interested in summaries and benchmarks as long as they are useful in order to develop more accurate designs, improve Building Information Models (BIM) and get realistic building performance calculations tailored to the specific location and characteristics of the intended building.

The development of such ICT or IT frameworks is completely realizable nowadays, the maturity achieved by factory and building automation systems completely meet the demanded requirements without entailing high complexities or efforts. A recent survey of factory and building automation focused on integration and communication capabilities, trends and issues can be seen in [SSKD11].

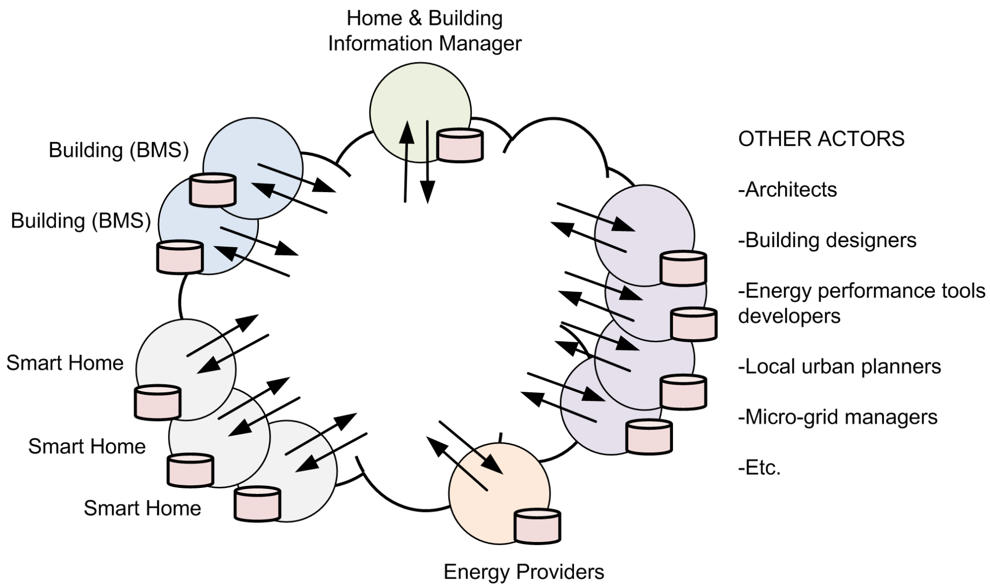


Figure 2.1: Potential stakeholders of information related to home habits and uses.

Furthermore, the plug and play integration and connection of intelligent home automation systems to the Internet is discussed in [KL01] or [Kas03].

2.2 Definitions and Taxonomy

Habit profiles are normally designed thinking of specific applications, or keeping in mind a particular group of devices and even an underlying system or hardware. This is one of the reasons why habit profiles or patterns show multiple ways of implementation in the related literature, and rarely coincide (unless a single approach has been taken as a basis for next developments).

In this section, we try to establish some fundamentals and give guidance for an open design of habit profiles, which pursues compatibility and transportability; i. e. they can be used by different applications and services within the same, cooperative, linked or even different frameworks and systems.

2.2.1 Definition of Profile

Within the scope of home and building technology, we define a *profile* as the collection of time-related data (i. e. a *univariate time series*) that corresponds to a certain phenomenon and represents its performance or behaviour. Profiles can be understood as *models*, or a special type of model that relates values with times in order to explain or characterise the phenomenon.

When profiles address to users' behaviours, they are called *habit, behavioural, usage or use profiles* (Figure 2.2). These terms can usually be considered synonymous in a superficial way of speaking, though in closer context the different nuances between *habit profile* and *behavioural profile* must be remarked (Section 2.2.4).

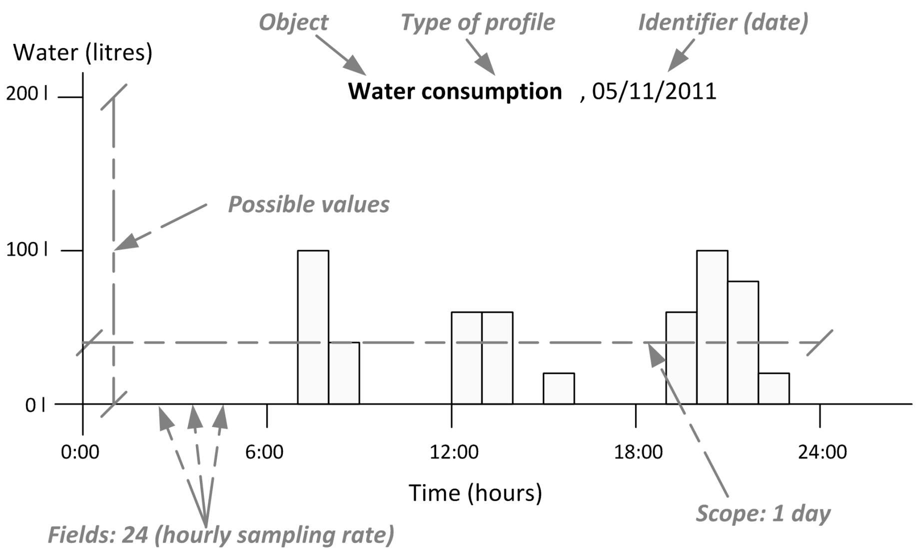


Figure 2.2: Example of a behavioural profile for the water consumption of a dwelling.

The fact that profiles must be referred to time is coherent with the purpose of abstracting habits, as habits entail facts or situations that are repeated throughout time. On the other hand, profiles can address different scale types (numeral, ordinal, interval, ratio), type of values (e. g. binary, integer, real), time scopes (daily, weekly, etc.) and sampling rates (n minutes, hours, etc.).

We consider the *profile scope* as the total length/duration of an individual profile, constituted by all the *profile fields* together. Profile fields are usually determined by the sampling rate, but not necessarily.

2.2.2 Habits Triggered by Events and Preference Profiles

Habits can also be seen as repeated events not necessarily related to times but to other events, i. e. "whenever *something* happens users usually expect or do *something*". Some of these situations are normally covered by specific applications, so it takes the form of an automated response triggered by a defined event (e.g., rolling up the awning when it starts to rain, or switching on corridor lights when somebody rings at the door during the night). Hence, most of these event-to-event relationships are solved by dedicated agents specifically designed for the task. On the other hand, it is absolutely possible that it happens without an evident connection or based on the users' lifestyle, e. g. an abnormal increment of the daily water consumption could be linked to a long absence (inhabitants have a shower and water the plants before leaving the flat for some days); being aware of that would result in better energy management. The level of awareness required for such cases entails high-complexity algorithms and systems not covered in this dissertation. As a general rule, we consider that the effectiveness reached by such systems so far is not properly probed or does not justify the required complexity. In any case, related-events' approaches can be seen in some works, e. g. [HJO06] or [DCB+02].

Another kind of object commonly mentioned as profiles are the so-called *preference profiles*, e. g. [Lee10] or [AT01]. They store static information concerning users' behaviours and device usage (without being necessarily related to time). We integrate them into our definitions and classifications (Section 2.2.6 and 2.2.7), considered as *abstract* profiles with only one time field (constant). In any case, it is worth keeping them apart due to the different management or level of operation that they are usually intended for. From a distinct viewpoint, users' preference profiles are seen as summarisations of low-level normal-defined profiles.

2.2.3 Ambivalent Approach for Profile Objects: Users or Devices/Zones

From the control perspective, behaviours can be related to users, but also to the use given to a device or group of devices, rooms or parts of the house, etc. Indeed, there is usually an ambivalent way of observing the interaction between users and the diverse elements and realities of their home (about conflicts among user, activity and object profiles, see [Lee10]). Therefore, for the same phenomenon, diverse designs are possible. For example, if we want the adjustment of an HVAC system in an office to be controlled by people's presence, *occupancy profiles* can address either users or the use of the space under control. To ensure the correct approach is not easy, although user profiles are prone to lead to more complex scenarios. In any case, it will depend on further aspects, starting on the architecture of the space, but also considering additional applications, available equipment and based on the general philosophy of the whole system design.

2.2.4 Addressing behaviours or Habits

In Section 2.2.1 we have commented on the fact that profiles can be *habit abstraction objects* or *behaviour abstraction objects*, existing differently in each case. Indeed, a profile does not necessarily represent a habit, but always a behaviour. This behaviour can be isolated (not representative), but also can be a repeated behaviour (so it involves a stable habit), or also an approximate representative of repeated facts with minor differences (so it represents or models them in the form of a habit). Figure 2.3 illustrates the point. In the example, the system is aware of the number of daily showers divided in four six-hour periods per day. Starting on the basis that we only have five profiles so far, we can see that three of them have the same shape, so they probably summarise a habit, whereas Profiles 2 and 5 are considered as singular behaviours.

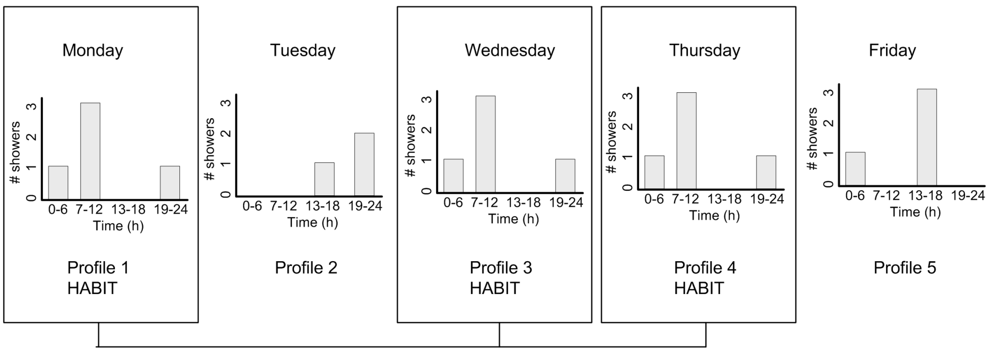


Figure 2.3: Profiles for punctual behaviours and habits.

2.2.5 Design Recommendations

In order to facilitate the sound and cooperative management of habit profiles for multiple applications, the following guidelines should be considered:

- Simplicity and independence.
  Profiles should be as simple as possible, it is preferable to have many simple types of profiles than a few complex ones. It follows the good practice known as *divide and conquer*, here intended to pave the way for controllers and tools that must abstract patterns. In the same line, a specific profile should not deal with mixed information, keep any reasoning embedded or entail mixed scale types.

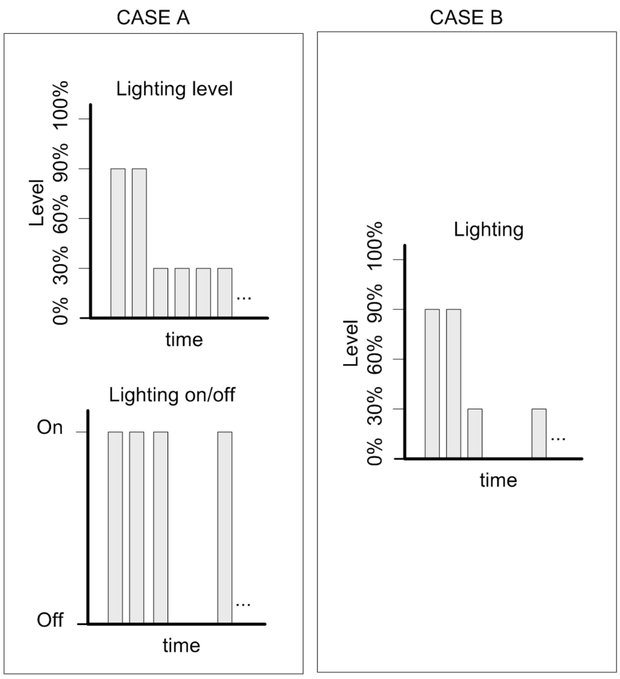


Figure 2.4: Two different ways of abstract lighting by means of profiles.

  For instance, in the Figure 2.4 two different ways of abstracting pieces of lighting are shown. In principle, we are inclined to select *CASE A* so it divides two different information existing in lighting. Thereby the reasoning by automated processes is simpler, which can freely compare collected profiles and speculate independently *on/off habits* and *lighting level* habits.

  In contrast, splitting data too much can imply considerable amounts of information that are stored and processed without being necessary. A correct design will state a commitment with the correct number and size of profiles based on the expected, global performance.

- Equivalence and invariance.
  Profile fields (Figure 2.2) should be equivalent in size/length and their values equally weighted; i. e. all the fields have the same importance before subsequent analysis.

- Compatibility and scalability.
  Scalability allows easy summarisation, i. e. to condense information and create new profiles and pieces of information with wider scopes. It makes profiles flexible and useful for multiple purposes. On the other hand, different profiles should allow an easy superimposition in order to facilitate controllers to make quick reasonings.

2.2.6 Objects of Home Profiles

Profiles refer to different objects or actors of the home environment. We propose the following classification for profile objects (examples are listed for every type of object):

- Device or house element (not necessarily physical), e. g. *lighting level of a lamp, status of a socket*, *status of a window*, etc.
- Group or line (logical or physical), e. g. *lamp group X status, ring circuit consumption,* etc.
- Room (usually the minimal space portion considered by a home system), e. g. *occupancy of a room, comfort temperature of a room,* etc.
- Zones and sub-zones, i. e. a group of rooms or/and corridors joined by their physical location (e. g. *first floor*) or their purpose (e. g. *sleeping rooms*). These profiles are usually – but not necessarily – formed by superimposed room profiles. Some examples: *water consumption of baths, lighting usage of corridors*, etc.
- Dwelling (house, flat, facility, etc.), i. e. the habitable space as a whole. Some examples of profiles: *number of people at home, flat electricity consumption*, etc.
- Individual users. Direct user profiles are appropriate to satisfy cases when a high-dedicated control is expected (Ambient Assisted Living, AAL) or when high-customised performance is desired. Some profiles: *comfort temperature for user X, expected location for user X* .
- Family or group of users. For instance, *adults, kids*, etc.
- External object, which is not strictly included in the smart home environment. Some examples of external profiles: *electricity spot prices, outdoor temperature*, etc.

2.2.7 Types of Home Profiles

According to the nature of the captured phenomenon, we establish a classification for profiles and distinguish the next types:

- Occupancy.
  Users' occupancy is probably the most important object of habit abstraction as many application utilise it to automatically adjust the system configuration, manage the energy performance or advance comfort requirements. It is also the best example of general resource profile as practically all home applications require the user presence for their running at some point.
- Consumption.
  Consumption profiles are directly related to measurable consumed magnitudes or costs – usually energy, like electricity, water, DHW or gas –, but it is open to other applications that do not necessarily concern energy; for example, they can also refer to money, time to fulfil specific tasks, etc. In any case, the evaluation and control of the energy consumption is the most common purpose of this type of profile. Beyond direct control applications, consumption profiles are very useful to provide feedback to users in order to allow the improvement of their energy behaviour, but also to develop benchmarks for further studies and designs [Fis08].

- Usage (or status).
  Usage profiles address specifically whether one object is *switched on* or *being used* at a certain moment. It usually differentiates between two states: active/passive, on/off, open/close, etc.[1] Nevertheless, a usage profile can entail more than two states, but note that in that case maybe it should be defined as a *setpoint or level profile*. Whenever one of the status/modes can be separated from the rest as a passive status (off, not-connected, etc.), we will feel preferably inclined to split the phenomenon into a *usage profile* and a *level profile*.

- Setpoint or level (or status).
  Setpoint or level profiles collect desired or detected input status (unlike consumption profiles that represent resulting output phenomena). *Comfort temperatures, relative humidity levels* or *lighting levels* are some examples. Again, if the measure belongs to a ratio scale (for the application scenario), we will be prone to separate it into *usage* and *level profiles*, becoming the ratio scale into an interval scale for the profile management.

- Abstract aim.
  Profiles that manage objects with some complexity or resulting from previous processing and reasoning phases are considered as abstract aim profiles. For instance, profiles that store users' activities [MTIS04], e. g. sleeping, having a meal, working, spare time, etc.

- Specific aim.
  Profiles that are not easily classified in the previous points fit here. This label is devoted to those profiles that are too specific and can hardly be useful for other applications or services beyond the one for which they have been designed. An example of a specific aim profile is *a profile that stores the time that the fridge door is open* [Int02].

2.3 Do People Keep Habits at Home?

Most of the related literature that pleads for the utilisation of habit abstraction in HBA applications consider the question regarding whether people keep habits at home as a matter of fact. It means that they do not justify it [HC03], or the justification is assumed as it works in the respective case study [MFS+07]. Indeed, thinking like this is rather logical. For instance, it is not surprising that a singular inhabitant wakes up at approximately the same time every day, or some people prefer to have a shower at night instead of in the morning; maybe users keep extra-activities with a weekly cadence, or it is quite possible that the times they leave or come back home usually coincide. These are some examples of home habits; in fact, everything that people do in a repeated way at home – and even outside – can be somehow translated into effects on the energy consumption or comfort performance of the dwelling.

A further interesting question is to know whether different people show similar habits, and if the facts that determine correspondences and differences between people's habits are due to, for instance, climatic conditions, distinct lifestyles, size of families, education, standards of living, type of dwelling, etc., whether the similarities have a very located character or they can be generalised to other populations.

[1] An *occupancy profile* is a special kind of 'usage profile', which has been differentiated due to its remarkable role.

2.3.1 Clustering Tools for Habit Analysis

Therefore, before starting to develop profile-based applications and algorithms, it is worth obtaining proof regarding the existence of domestic habits. In Chapter 5 we will see how *clustering analysis*, as a common data mining technique, is usually employed to discover patterns of habits within myriads of data. Indeed, the outcomes of the clustering tool can be used as a validation test in this respect, but it is sensible to keep them a priori in doubt for the following reasons:

- Clustering methods are intended to find out patterns, so they are prone to force data to adapt to superimposed structures that perhaps do not exist by themselves [TK03].
- The validity of clustering methods depends on data characteristics [QZ02], which are usually unknown at the very starting point. The best method and the best parametrisation to use are complex fields of discussion to carefully deal with.

For the time being, we can say that clustering tools are good methods to discover patterns once you are certain that a pattern exist. For that reason, as a first step, a more prudent option is to confirm the habit existence by means of mathematical methods.

2.3.2 Searched Habits in Real Databases

According to the two main deployments of habit profiles, i. e. building design and smart home control optimisation (Section 2.1), the points to check in this section are as follows:

1. People keep habits, i. e. there are daily repeated behaviours in singular users or families.
2. Habits of some people are similar, i. e. different inhabitants living in distinct dwellings show behavioural similarities.

To check the previous statements, we analyse a database provided by the Leako company (Section B.2.1). The variable selected to undergo the analysis is *water consumption*. We choose water because it is the available collected phenomenon least dependent on boundary conditions, e. g. seasonality or weather conditions. In other words, water consumption is the freest variable, or the most clear user-dependent. Because of boundary conditions, other variables (DHW consumption, comfort temperature, heating consumption) are more prone to show correlations.

We take hourly data from more than 2500 days for almost 700 dwellings. A first look at the developed tests is shown in Figure 2.5. There are two different types of test in keeping with the points shown in Section 2.3.2. The first set of tests focuses on individual users/families, by dividing the whole data into daily samples; the second set of tests compares different users/families each other.

Similarity is checked using Pearson's correlation distance (defined in Section 5.2.3).

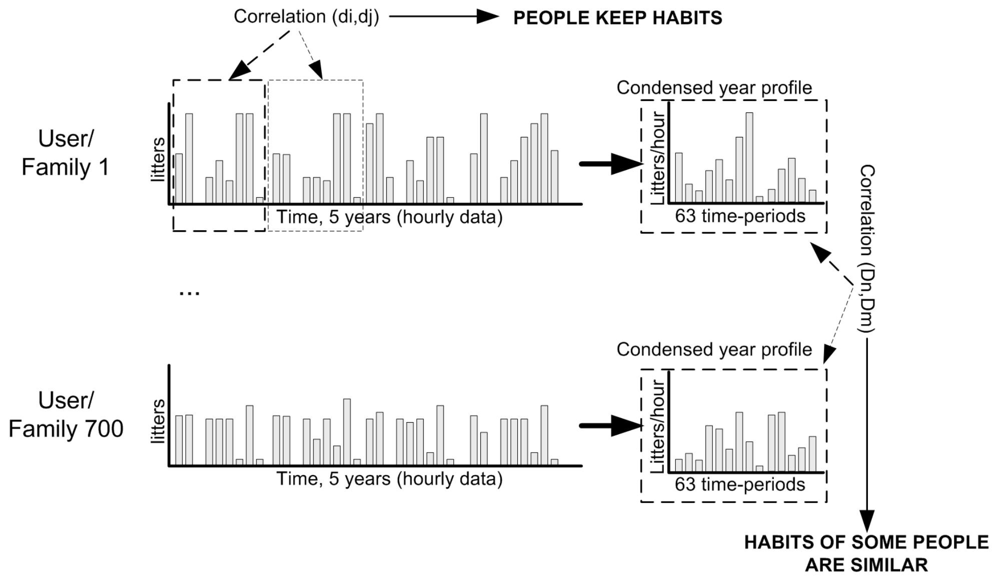


Figure 2.5: Developed correlation tests with Leako Database.

2.3.3 Daily Habits for a Single User/Family

Firstly, we independently analyse every dwelling as a temporal series. Every data array (corresponding to a single dwelling) is cut up into daily arrays, so we obtain set of consecutive samples within a duration of 24 hours. In this case, stationary process criteria are not usually achieved, i. e. time series analysis and model estimation methods cannot be applied.

Nevertheless, selecting a singular dwelling and analysing the correlation between its days, after filtering absent days (no water consumption); results usually show a high number of correlated days.

Figure 2.6 shows the probability density function (pdf) of the number of days with a correlation higher than a certain level ($\rho \geq 0.6$) for a dwelling selected at random. The shown pdf appears quite often in the database. In the figure, there is a high number of apparently uncorrelated samples or without linear dependence (left side), but there are also a lot of days that hold a high correlation among them (right side). This right side points up, showing that there is a significant number of days with certain resemblances (at least with regard to linear dependence). A rough possible abstraction of this scenario is shown in the same figure, illustrated by the hypothetical representation in a 2D solution space; the points symbolise days and the distance among them symbolises similarity (in the representation, for the sake of understanding, it is supposed that there is only a group of similar days). The *pdf* (probability density function) states that, at a fixed certain level of similarity ($\rho \geq 0.6$), if we choose a day at random, it will tend to be either alike the members of a set of similar days or significantly different to the rest of the whole population.

It is worth noticing that Pearson's correlation only shows linear dependences, as well as the fact that the meaning of correlation values depends on the size and characteristics of data samples. Correlation indexes must be related to some kind of similarity appraisal (for instance, visual observation). In this case, Figure 2.7 and Figure 2.8 are shown to give an impression regarding what correlation values of 0.6 and 0.9 are in our experiments. In any case, the correlation

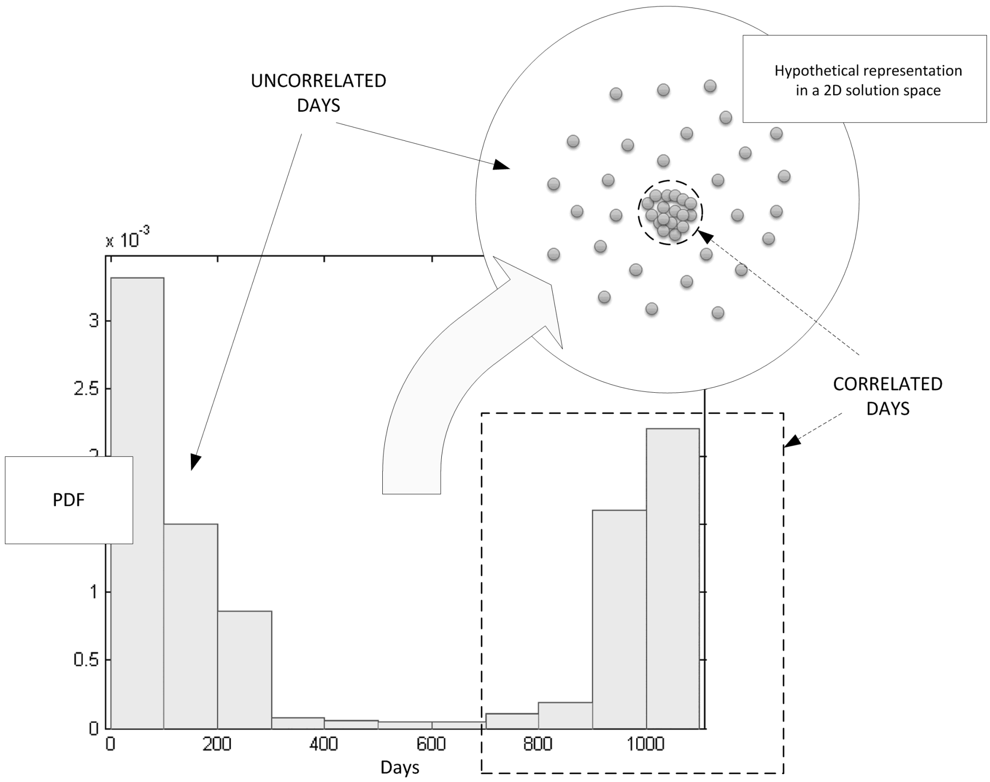


Figure 2.6: Pdf of the number of days with a correlation higher than 0.6 in a random selected dwelling and hypothetical representation in the solution space.

coefficient are not free of wrong interpretations and cannot replace the individual examination of data [RN88].

The tests concerning correlated days have been undertaken for different dwellings selected at random, Table 2.1 shows some results. Note that to correctly interpret the table can be complicated; for instance, the analysis in "dwelling 1" reveals that, removing days without consumption, about 24% of the days have a high similarity (Figure 2.7) with more than 100 other days, and about 42% have an acceptable similarity (Figure 2.8) with more than another 500 different days. In short, these results reveal existing trends and point to validate the asseveration regarding the fact that people keep daily habits (at least in water consumption behaviour).

	Dwelling 1	Dwelling 2	Dwelling 3
Total days	2641	2634	1494
After filtering 0-days ($af0$)	2384	1527	1274
$\rho_{0.6}(500)_{af0}{}^{*}$	1001	199	526
$\rho_{0.9}(100)_{af0}{}^{**}$	567	142	179

**: days that have a $\rho \geq 0.6$ with more than 500 other days.*
****: days that have a $\rho \geq 0.9$ with more than 100 other days.*

Table 2.1: Correlation (ρ) between days for three dwellings selected at random.

It's worthwhile recalling the fact that data has been collected for a period between five and seven

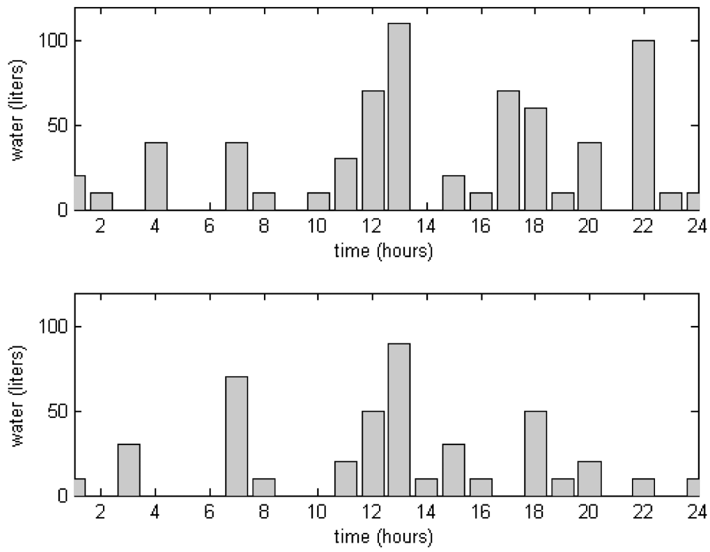


Figure 2.7: Two day samples with a correlation value of 0.6

years. For the analysis, we are not aware about possible changes of owners or tenants. Obviously, different inhabitants entail variations in habits which can disturb or decrease the expected values in correlation indexes.

2.3.4 Habits Shared by Communities

In order to validate the second point regarding whether *different people show similar habits* (Figure 2.5), data condensation has been carried out to facilitate resemblance comparisons (some loss of information is assumed). The performed summarisation matches the data frame utilised by Spanish official building energy performance tools that use behavioural data for calculations, [IDA07]. For these tests, samples represent dwellings, and the information of a whole year is condensed in an array of 63 elements by means of statistical procedures. Later on, the analysis is analogous to the *people keep habits* test case. Results are collected in Table 2.2. They find that about 11% of families show a high similarity of behaviour with at least 5 other families (compared throughout a year), and about 64% of families show acceptable similarity with more than another 50 families.

Again, we need to know what correlation indexes indicate, because the variation of magnitudes and the number of variables are different in this second case. Figure 2.9 is useful to gain an impression about what a $\rho = 0.6$ means for this experiment.

Both sets of analysis corroborate that representative patterns can be obtained/discovered within the available data. The direct observation and the experience of technicians and experts managing the database give extra support to the assumptions checked in this section.

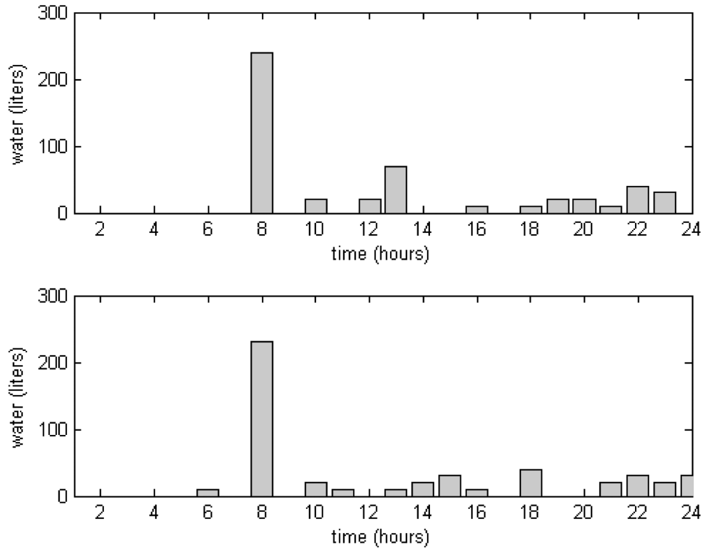


Figure 2.8: Two day samples with a correlation value of 0.9

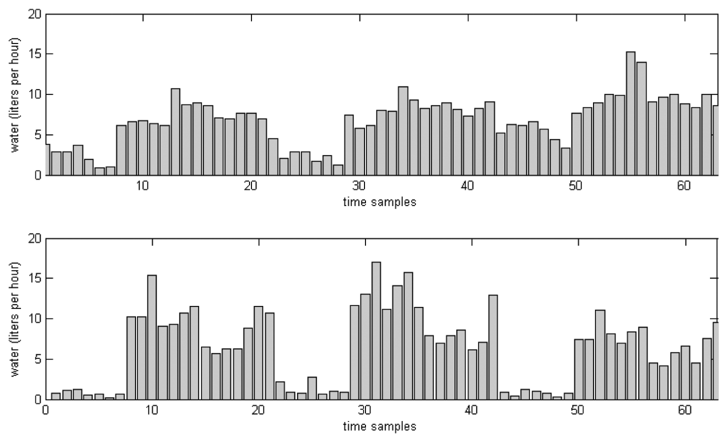


Figure 2.9: Two dwelling samples with a correlation value of 0.6

2.4 Integration in HBA Networks

Habit profiles are created by dynamic data present in houses and buildings. Therefore, HBA systems are in charge of capturing and managing the information that will turn into behavioural and habit profiles. This section deals with the integration of profiling techniques within home and building technologies and systems, how they are created, stored and are available for controllers and services that may use them.

2.4.1 Profile Generation Process

For the profile generation the usual process involves a sensor or group of sensors that transform physical phenomena into information understandable by the smart home system, as well as a

Total dwellings	685
After filtering 0-dwellings ($af0$)	668
$\rho_{0.6}(50)_{af0}{}^{*}$	426
$\rho_{0.9}(5)_{af0}{}^{**}$	74

**: dwellings that have a $\rho \geq 0.6$ with more than 50 other ones.*
****: dwellings that have a $\rho \geq 0.9$ with more than 5 other ones.*

Table 2.2: Correlation (ρ) between dwellings

module that takes data from sensors and creates a new profile. From now on, this module is mentioned as Profile Generator (PFG). A schema of the process can be seen in Figure 2.10.

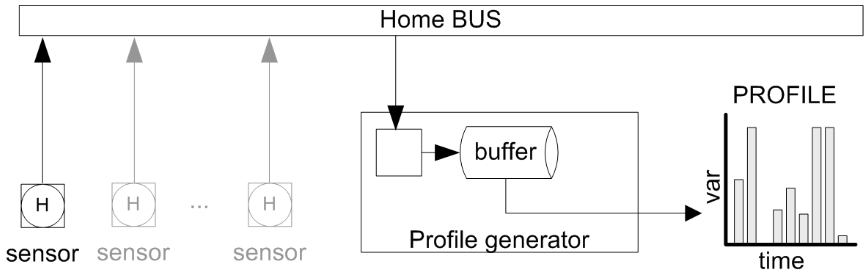


Figure 2.10: Process of profile generation.

Profiles are not always forced to be formed with data collected from the physical environment through sensors, they can also represent non-physical events or situations related directly or indirectly to habits (e. g. user commands received by e-mail or phone, or *user activities* deduced by additional smart modules). On the other hand, depending on the phenomenon to be abstracted as a profile or the employed methodology, the generation process can be simple or complex, i. e. requiring a stage for the adaptation of the data coming from sensors. For example, a profile of the dwelling occupancy could be built using a special lock in the main door, but also through the cooperative work of a set of presence sensors. Whereas the smart lock can directly give binary readings (0 absence, 1 presence) to the PFG, the set of sensors needs a previous logic that prepares the data.

2.4.2 Profile Management within HBA Networks

The core of the profile generation, storage and control are mainly operated in a *management level*, and they should reduce or even bypass some logics that sometimes are embedded into the *automation level* (Figure 2.11). Some case-adaptability and flexibility in the network architecture is commonly accepted [KNSN05], but a coherent and hierarchical design between control tasks of both layers in order to provide an efficient management is expected. This differentiation between control and management levels (Automation BUS and Management BUS) can also be considered in a symbolic way as the separation between specific control and global control. High-level interfaces (gateways or routers) able to provide abstractions of the field area network are set to facilitate such migration and arrangement, allowing the natural and smooth creation of complex applications [KN04]. Such gateways or routers are protocol converters and allow the interconnection of different or independent systems and architectures at any communication level.

In this specific case, they establish a natural separation between two domains where usually prevail different types of communication: commands (automation BUS) and information (management BUS).

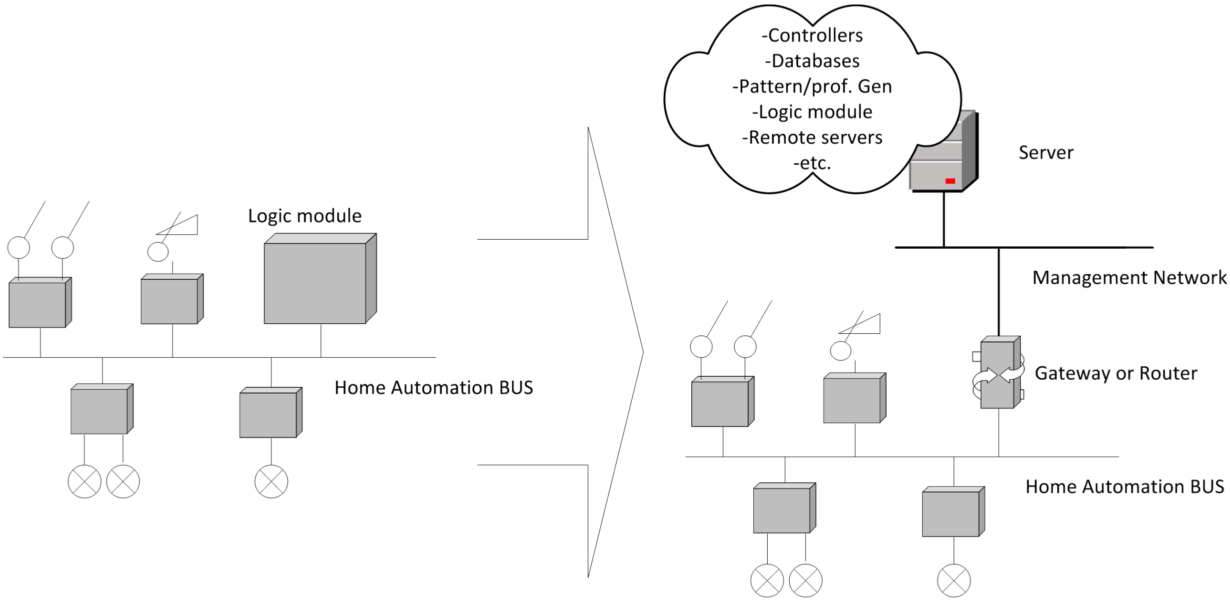


Figure 2.11: Some logics in the *automation level* must migrate to the *management level* to keep the coherence and allow an accurate control in keeping with the information generated by habit profiles.

Upgrading the schema drawn by Figure 2.10, Figure 2.12 widely depicts the process of profile generation, storage, management and use (for direct control purposes). The information collected by sensors is sent with a certain cadence to the PFG module. It is not necessary that the sampling rate deployed to build profiles exactly matches the sampling frequency of the monitored sensors; intermediate stages for data transformation can be acceptable in both network levels. When the PFG has enough information to generate a complete profile, it stores the newborn profile in the corresponding Profile Database (a *behaviour profile* has been created and stored). Later on, in a pre-fixed time according to the control requirements or based on the time scope of the intended profiles, the Pattern Generator (PTG) takes a set of profiles from the database and executes an advanced algorithm in order to discover patterns among the set. The discovered pattern or patterns and extra information concerning their characteristics are saved in the respective Pattern Databases (one *habit profile* has been created and stored). This new pattern or patterns replace old ones as far as they represent more clearly the current users' habits. Controllers will take them to perform control actions and decisions for the next day.

Unlike *controllers* placed at a management level, the rest of the elements at the same level do not operate in the forefront of the demanded actuation and should not be stressed by instantaneous requirements. Therefore, latency periods for PFGs and PTGs, as well as database managements, do not involve relevant constraints or critical points since controllers can continue to execute their routines fairly. They would use previous or old patterns and profiles, whilst waiting for new patterns to be available.

In the profile management the amount of information is usually higher than the amount of commands, ensuring that time is not a critical aspect of the process. These characteristics

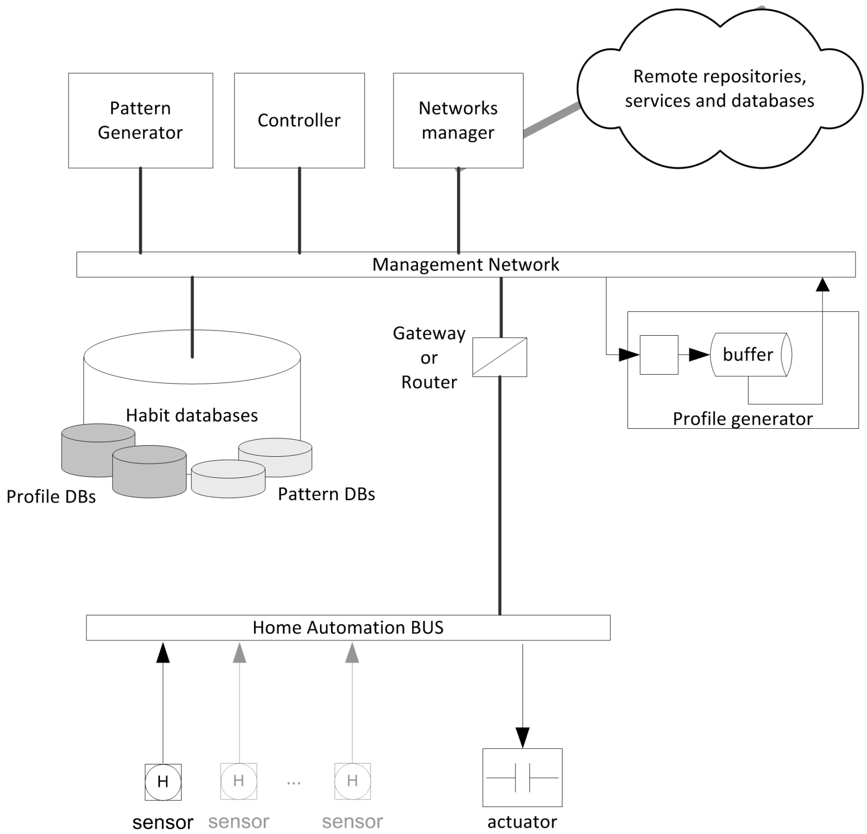


Figure 2.12: Profile management structure.

make IT (or IP) networks very suitable to accommodate profile databases, PFGs and PTGs – discussion about HBA network architectures and functional hierarchies are offered in related publications, e. g. [KNSN05]. Here, the most critical point concerns the controllers, which can require a quick response depending on the specific application. Network communication from controllers at management level to controllers at field level (or end equipment) must be carefully considered.

In any case, the requirements of global control based on habit profiles is completely fulfilled by most of the current automation standards, e. g. LonWorks [Ech09], EIB/KNX [KNX04], BACnet [Ash10], Zigbee [ZB 05], whether or not being specifically devised for HBA (standards are commented in Section 3.2). In order to keep the desired level of compatibility, cooperation and flexibility, the utilisation of open systems is always preferred or, in any case, networks and systems that allow an easy integration of multiple and common solutions (by means of gateways and/or servers). Dynamic application frameworks like OSGi [The11] perfectly fit global control approaches. For the integration at management level, open standards are also desirable, e. g. OPC [OPC09], XML/SOAP [ML07], oBIX [Fra06], BACnet/WS [KS11].

In short, there are no special requirements that challenge the capacity of the referred solutions. Global profile-based control remains in a higher, conceptual level of design and it is devised taking into account the possibilities, capabilities and limitations of common technologies for the HBA environment.

3 Habit-based Smart Homes

In this chapter the concept of overall smart home control is studied and related to the existing literature and previous projects. The state of the art and standards review culminates in the analysis of the main foci faced up by global approaches, and the later localisation of the principal objectives aimed by the current thesis. Before the definition of profile-based applications, it is necessary to have a model for the system-user interaction that explains the different ways of feedback between the users and the system. Later on, following this perspective, habit profiles are introduced twofold: as shared objects for global reasoning and context abstraction, but also as objectives of persuasive, proactive designs. To offer a horizontal view of the approach, a set of profile-based applications that cover expected functionalities of automated houses is depicted. Finally, an example case of a profile-based apartment is described, showing the joined implementation of some of the proposed services.

3.1 Overall Smart Home Systems – Related Work

In smart homes, applications fulfilling isolated requirements by routines that follow independent goals is a common reality. Behind this classic scenario, research and application efforts have been mostly dedicated to optimise specific solutions and to facilitate communication, coexistence and exchange of information among different devices, technologies and systems [Har03].

For that reason, we do not err when we say that most of the research for smart homes is faced from a *bottom-up* perspective. It seems to be logical if we consider the broad application field covered by the term *smart home* or *home automation*, in a high correspondence and similarity with the requirements of the *building automation* area. Here, experts dedicate their efforts to scenarios as accurate, precise and isolated as possible, in order for them to improve the detail.

However, the existing multidisciplinarity demands also consider interactions from *top-down* viewpoints and overall managements. Using the words of Borggaard et al. [BBSZ09]:

> Recent results have shown that by considering the whole building as an integrated system and applying modern estimation and control techniques to this system, one can achieve greater efficiencies than obtained by optimizing individual building components such as lighting and HVAC.

In keeping with this idea, there are several works that exploit the advantages of overall management for smart homes. Michael H. Coen proposes in [Coe98] design criteria for creating highly embedded, interactive spaces to be named *Intelligent Environments*. Their experience in a test-bed room leads them to emphasise the careful selection of "modalities" (i.e appliances or applications), but also the necessity of dynamical adjustment to the place's activity, systems that can train themselves, and avoiding extensive manual calibration.

Covering such requirements and presented as a home that programs itself, Mozer's Adaptive House in Colorado [Moz98] is a good example of an intelligent environment. It deploys neural networks for the control phase intelligence and, similarly to our approach, it infers usage patterns from the inhabitants' lifestyle and uses them for prediction-based actions.

Some integral designs, attempting to cover the required cooperativeness but keeping the appliance independence, propose structures based on *Multi-Agent Systems* (MAS)[1]. MavHome [CYH+03] is a good example. Here, groundings are defined with a hierarchical framework of rational agents where each agent is configured following a structure of four layers (physical, communication, information and decision). MavHome pays attention to context awareness, perceiving the state of the house and learning the inhabitants' interaction. For prediction purposes, MavHome uses a backpropagation neural network to establish confidence values for different prediction algorithms based on historical data. The final prognosis is performed by considering the predictions of all the algorithms and filtering them by a voting scheme.

Furthermore, in view of the high heterogeneity of the field, some approaches use *ontologies* for the knowledge representation[2]. For example, the DomoML project [SPF05] proposes a taxonomy which does not deal with the building structure but emphasises household appliances and their location. Also an interesting project is the Sydney Opera House [SMA+], where the ontology representation is mainly focused on aspects related to the facility management. Semantic and intelligent knowledge representation of energy-related information for future smart homes are introduced and widely explained in [KRK12] and [KRK11].

Another example of global control can be observed in the inHaus project in Duisburg, Germany [Nar07]. The IT infrastructure combines different technologies (ZigBee, WLAN network, RSSI based people tracking system, UHF RFID gate, mobile LF- and UHF-Reader units, etc) with the help of a middle-ware layer. In the context of inHaus the authors of [DPN07] also propose a generic probabilistic reasoning framework for networked homes based on ontologies.

On the other hand, the heterogeneous reality of information and systems present in the dwelling environment is a common concern considered in many works. For example, the university of Essex has developed iDorm [HCC+04], an intelligent dormitory that operates with multiple and heterogeneous systems and networks. It consists of an adaptive agent that uses fuzzy algorithms to work in a lifelong learning mode, just assimilating users' needs and preferences. Works like HomePort [Mad09] simply try to propose a way to connect different home control systems through an intelligent gateway, just by attending the broad variety of existing technological solutions focused on the dwelling environment. By means of subsystem communication drivers, a virtual communication layer and a dedicated high level controller, the global control of a smart house can be faced.

Putting focus again on multi-agent concepts, the OSGi (formerly Open Services Gateway initiative) platform also proposes to implement an agent based framework [ZWA05]. This project

[1] For a broad introduction to MAS: [Fer99].
[2] A good introduction to ontologies can be found in [Gru95].

pursues the integration of different domotic devices allowing remote control and fault diagnosis. UPnP (Universal Plug and Play) in combination with an agent framework are used for device discovery, registry, and management.

Moreover, multi-agent architectures have also been proposed specifically for wireless sensor networks [DL08]. Here, the effort focuses on developing a cooperative and distributed control system with conflict resolution and users' behaviour identification capabilities under a wireless infrastructure (ZigBee). Moreover, the multi-agent concept is an open reality that can be understood in different ways. For example, beyond agents representing applications or services, Cheng and Tseng [CT07] define space agents, which are distinguished by house zoning and commanded by a governor agent. UPnP and Microsoft's SCP (Simple Control Protocol) are used to communicate and manage the whole system. UPnP connectivity is very important in the world of smart homes as it is called to facilitate the interconnection of devices and equipments in a coordinated and intelligent manner without requiring any set-up work or administrative effort by the users. An overview of UPnP technologies for HBA is shown in [KS04].

Finally, overall concepts supporting the development of profile-based smart home control are introduced in [RKK10]. The groundings of this approach are also the establishment of a cooperative community of agents connected to an ontology-based knowledge representation. Such underlying structure allows profile-based designs, boosting the optimised and coherent application of artificial intelligence (AI) methods and controllers in a sound interaction. The integration of AI methods in this system proposal are introduced and discussed in [KKR10].

These are some interesting works; without proceeding to compare them, we can just see that they face similar problems or scenarios setting out overall intelligences based on different structures and algorithms. As far as holistic perspectives are concerned, note the importance of commented concepts like *top-down perspectives, MASs, intelligent frameworks, ontologies, heterogeneity of systems, equipments and designs*. Such ideas draw together the current reality of research and proposals which deal with the high complexity of the intended issue.

Hence the conclusion is that the smart home matter needs something beyond specific solutions or application examples, it requires the definition of a conceptual underlying basis capable of drawing a paradigm for the design of use cases as well as the development of open, flexible, compatible and complete technologies. This assessment does not mean that the current world of smart homes is a complete mess. There are diverse regulations and attempts of standardisation that try to establish a sound basis for the integration of HBA systems and the building of smart homes. Albeit their success must be considered mainly in a technical layer, reviewing the present directives and standards related to HBA is worthwhile.

3.2 Existing Regulations and Standards

HBA installations must be carried out according to regulations that control the correct and safe application of technologies and electrical equipment. Within the European scope, directives and regulations, periodically published by the Official Journal of the European Union (OJEU), are compulsory laws and their compliance is guaranteed in products by the CE marking. As far as HBA is concerned, we can mention the next directives as examples:

- Low Voltage Directive (LVD) 2006/95/EC, which ensures that low voltage electrical equipment meets health and safety requirements as well as enjoys a single market in the European Union.

- Electromagnetic Compatibility (EMC) Directive 2004/108/EC, which limits the electromagnetic emissions of equipment (according to its intended use and placement) and also demands certain level of immunity to interference.

Note that CE marking does not involve any assessment or guarantee concerning the quality of the product, i. e. the level of success to accomplish what the equipment is intended to. Aiming in that direction, in addition to laws, there are initiatives of standardisation related to HBA that are developed by European organizations as well as international agencies. The validity of the published standards is supported by the international recognition of such organisations. It is worth highlighting some of them in keeping with their relevancy for the discussed area. ISO (International Organization for Standardisation), CEN (European Committee for Standardisation) and ANSI (American National Standards Institute) are the international, European and American main entities to develop general purpose standards. Within the European scope, CENELEC (European Committee for Electrotechnical standardisation) and ETSI (European Telecommunications Standards Institute) have been specially created to be in charge of the areas of electrical engineering and telecommunications, respectively. The equivalent international organisations are IEC (International Electrotechnical Commission) and UIT (International Telecommunication Union). ASHRAE (American Society of Heating, Refrigerating and Air Conditioning Engineers) is also a very important society which has published a considerable amount of standards with respect to the application of building technologies.

The usage of standards assumes the presumption of compliance with laws, but note that, in comparison to compulsory regulations, the fulfillment of standards is always voluntary. In any case, their adoption is being more and more required by the smart consumer, as there is an increased necessity of open systems and technologies. This is specially justified in HBA, where equipment is expected to last for many years, fields of application are diverse and heterogeneous, and flexibility, compatibility and system-interoperability are mandatory features. For the drawing up and acceptance of standards, the introduced national and international organisations, in addition to other companies for standardisation, establish competent technical committees (TC). In short, these committees take care of guaranteeing the coupling or connection among independently manufactured devices, also their technical quality, safety, replacement and designs according to social responsibility. Some of them, in connection to HBA issues, are listed as follows:

Technical Committees and Main Standards

ISO TC 205, called “Building Environmental Design”, looks after the preparation of standards and recommendations that cover the integration of electronic systems in the scope of homes and buildings. The aim is to guarantee a correct design of building indoor environments in terms of comfort and energy efficiency, dealing with air quality, thermal, acoustic and visual factors. This TC is composed of 25 participating countries and is responsible of 16 ISO standards so far.

Similarly, with 24 standards already published and some other under development, **CEN TC 247** is in charge of performing European standards for “Building Automation, Controls and Building Management”, dealing with definitions, requirements, functionality and test methods of HBA products and systems.

Both ISO TC 205 and CEN TC 247 together are responsible of EN/ISO 16484 Series, entitled: “Building Automation and Control Systems (BACS)”. As the name indicates, it defines the principles that must allow the integration of different subsystems within the BACS. It is composed by different parts: (1) Project specification and implementation, (2) Hardware, (3) Functions, (5) Data communication protocol, and (6) Data communication conformance testing. Parts 5 and 6 (ISO 16484-5:2012 and ISO 16484-6:2009) are focused on the BACnet protocol, consolidating it as a European and international communication system.

The CEN TC 247 is also responsible of the European EN 13321 and EN 14908 Series. The second part of the former series, EN 13321-2:2006 and entitled: "Open data communication in building automation, controls and building management. Home and building electronic systems. KNXnet/IP Communication", defines the communication of KNX systems by tunnelling over IP networks. This standard should be considered as built on the basis of the previous EN 50090 Series, published by CENELEC TC 205. On the other hand, EN 14908 Series entails the European acceptance of the LonWorks protocol.

CENELEC TC 205 is a European committee specially focused on ensuring the integration of control and management applications regarding home and buildings, including gateways and interconnection of different media, and also taking into account issues concerning EMC, and electrical and functional safety. The relevant EN 50090 is a CENELEC standard that covers all technical rules for Home and Building Electronic Systems (HBES). It is strongly bound to the KNX protocol, and assures safety aspects, conformity assessment of products, as well as specifications and descriptions to fulfil every communication requirement. It is composed of the next parts: (1) Standardisation structure, (2) System overview, (3) Aspects of application, (4) Media independent layers, (5) Media and media dependent layers, (6) Interfaces, (7) System management, (8) Conformity assessment, and (9) Installation requirements.

Finally, noteworthy is the ISO/IEC Joint Technical Committee JTC 1, Subcommittee SC 25, Working Group WG 1 – all together: **ISO/IEC JTC1 SC25 WG1** – and entitled: "Information Technology, Home Electronic System (HES)". It is an international committee focused on the standardisation of control communication within homes. Therefore, efforts here are mainly concentrated on gateways called to perform a fluid information exchange between the internal HBA network and external wide-area networks (e. g. Internet). This committee approved KNX as an international standard (ISO/IEC 14543-3) and also LonWorks (ISO/IEC 14908 Series). ISO/IEC 14908 Series consist of four parts that cover: (1) Protocol stack, (2) Twisted pair communication, (3) Power line channel specification, and (4) IP communication. Moreover, ISO/IEC JTC1 SC25 WG1 has drawn up other relevant standards for HBA. For example: ISO/IEC 15045 Series (Information technology – HES – gateway), consisting of: (1) A residential gateway model for HES, and (2) Modularity and protocol. Also ISO/IEC 18012 Series (Information technology – HES – Guidelines for product interoperability), with: (1) Introduction, and (2) Taxonomy and application interoperability model.

After reviewing such official documents, we can conclude that, nowadays, if we look for open system solutions able to cope entirely with HBA applications, BACnet, KNX and LonWorks are maybe the main or preferred options fully recognised by standards bodies.

Open System Solutions: BACnet, KNX and LonWorks

As a communication protocol, **BACnet** was originally developed in 1987, getting the American certification ASHRAE/ANSI Standard 135 in 1995. Finally, it becomes ISO 16484-5 in 2003. It was devised as an open solution, free of any licenses or fees, to allow communication and a fluent integration of different domestic and building technologies, equipments, services and applications, in a flexible manner and irrespective of the specific scenario to face. The BACnet protocol establishes services and objects used to exchange information within the building device framework, as well as it is defined to operate using diverse existing data link and physical layers. According to surveys in the United States, Europe and Japan in 2003, more than 28000 installations have been carried out in 82 different countries all over the world [KNSN05]. For additional information of the BACnet protocol, the interested reader is addressed to [Ash10] or the official website "www.bacnet.org".

On the other hand, with more than 100 member companies, about 7000 sets certified devices by different manufacturers, and more than 21000 installation companies in about 70 countries, **KNX** has been usually considered the main European network communication protocol for HBA. It was built up on three previous standards: the European Installation Bus (EIB or Instabus), BatiBUS and the European Systems Protocol (EHS). In short, KNX keeps to a large extent the basis inherited from EIB but adding the physical layers and configuration modes of BatiBUS and EHS. Being administered by the KNX Association, the KNX trademark logo guarantees the interworking and interoperability of all KNX devices irrespective of the manufacturer. KNX has been approved as European Standard (CENELEC EN 50090 and CEN EN 13321-1) and Chinese Standard (GB/Z 20965). In November 2006, KNX also achieves the international acceptance and becomes ISO/IEC 14543-3, including all the transmission media: TP (Twisted Pair), PL (Power Line), RF (Radio Frequency), IP (Internet Protocol). Characteristics of the protocol can be consulted in the official KNX handbook [KNX04], moreover additional knowledge about the system can be obtained through the official website of the KNX Association: "www.knx.org".

Finally, **LonWorks** is an American networking technology created by Echelon Corporation and specially addressed for ubiquitous, advanced control-networking systems. Unlike BACnet and KNX, LonWorks was originally devised to fulfil the necessities of a wider scope than solely HBA (for example, it has also been often utilised for industry and transport). Interoperability among LonWorks systems is not guaranteed by default, devices must also accomplish with the LonMark certification to ensure a fluent coexistence. The communication protocol LonTalk was firstly recognised as a standard in 1999, ANSI/CEA 709 Series. Later on, in 2005, it was defined by the EN 14908 Series developed by the CENELEC TC 205, officially becoming an ISO/IEC standard (ISO/IEC 14908 Series) in 2008 after being accepted by the ISO/IEC JTC 1. An introduction to LonWorks is provided in [Ech09], whereas broad information is available in the Echelon Corporation website: "www.echelon.com".

Smart House Code of Practice. A Spanish Example

Beyond the definition of standards and the consolidation of existing open HBA systems and protocols, a good and encouraging attempt to establish some principles for smart homes was carried out by the European Union by means of the Smart House European Project, where CEN, CENELEC and ETSI played a fundamental role. It concluded with the publication of the "CWA 50487 Smart House Code of Practice" in 2005. It is a reference document – non-normative and non-binding – for the implementation of a "smart house" based on accepted rules and specifications. It is intended as a practical guide of system design, installing and maintenance, addressed to all the involved actors: users, instalators, prescribers, service providers, system providers, etc.

Derived from this initiative, the Spanish standardisation Organization AENOR launched the EA0026 specification (November, 2006), with the underlying proposal to boost the market development, clarify the existing confusion and establish the minimum requirements. In short, it allows users to ask for an official certification about the features, performance and guarantee of good practices for home automation projects. The first AENOR Domotic Installation Certificate in Spain was conceded in 2008, the project was carried out by the company ADR Engineering deploying mainly a KNX system.

These exposed attempts of standardisation are obviously suitable and useful but non-conclusive. For example, the AENOR specification – considered as a test developed with the Smart House Code of Practice basis – evaluates if the HBA project is well-installed and finished focusing in each of the individual applications. Furthermore, it assesses the level of automation (minimum, medium or high). The global design evaluation results from checking if the classic functionalities – or objectives, i. e. comfort, energy savings, security and safety, and communications – are covered by individual services. It is missed a better global appraisal that evaluates the project as a whole, usability issues are forgotten as well as appraisals about the quality, balance and convenience of the applications. In short, we can conclude that a certificated house probably will work properly (technically speaking) and some important requirements will be accomplished, but it does not mean and does not give guarantee that the interaction with users will be satisfactory or they really appreciate or even find useful some of the installed services.

As an illustrative example, we can mention a real – representative and common enough – installation conducted in Spain under AENOR specification requirements. One year after the finalization of the project, clients revealed that they hardly use most of the included applications and even they have disconnected some of the initial functions. For example, the automated control of blinds and shutters based on schedule and lighting levels resulted annoying as well as other *smart* applications. They ended up regretting the investment on home automation and considering it as a superfluous whim.

It must not be understood as a criticism to the AENOR test (or the underlying introduced standards) that, in the worst scenario, are significantly helping to reach the desired regularisation. Obviously, the specification is developed as a technical exam, easy to check and trying to avoid subjective evaluations. The conclusion is that the core of smart home designs continues missing a robust methodology, sensitive to usability and psychological aspects, that establishes the ground of next sound proposals and clearly fixes what the future home automation must pursue as global guidelines. Under this outlook, holistic approaches have been usually incomplete or unsuitable for the real necessities. The most promising proposals rarely go beyond solutions of compatibility, cooperation, coexistence, connectivity and/or conflict resolution in a very technical layer, but still tied up to follow and fulfill the requirement of specific goals, whereas global guidelines remain nonexistent or under a blur definition.

3.3 Foci and Scopes of Global Approaches

After the revision of overall approaches and existing standards, we can summarise the main questions concerned by designers as follows (Figure 3.1):

a) How to deal with the high heterogeneity of the field?

b) How to be aware about the context?

c) What are the goals and how to fulfil them?

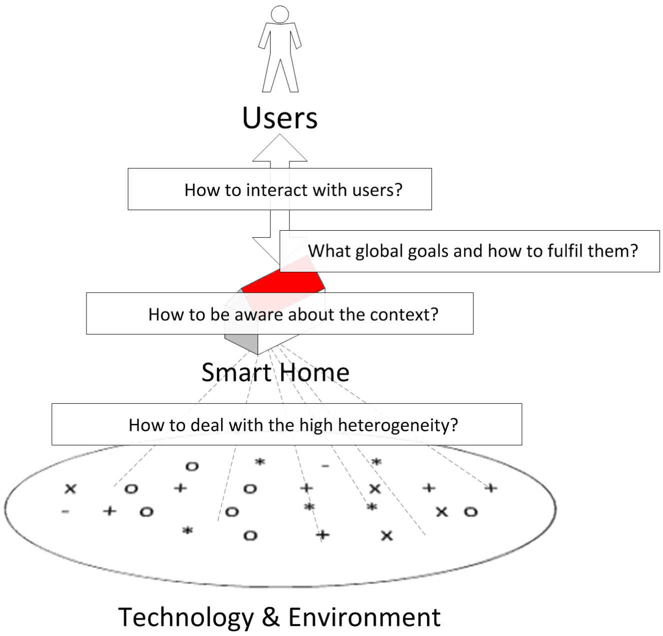


Figure 3.1: Questions considered for the design of smart home global approaches.

d) How to interact with users?

Research and projects usually face the smart home issue focusing on one of these aspects (if not more or all of them). In any case, the four points are strongly connected. For example, the first question involves how to deal with a changeable environment, as well as thinking about communication capabilities and compatibilities among the high number of existing technologies and equipment, allowing cooperativeness whenever possible. It is to be noted that the heterogeneity of smart homes does not only point to technologies; it also spreads in other dimensions, e. g. to consider different kinds of buildings, dwellings and facilities, as well as goals and applications, and even different types of users, families and their particular requirements. The strong interconnection with the other three questions is therefore obvious.

On the other hand, the concern regarding *context awareness* is the cornerstone of smart home design, and where the sense of global intelligence for a such heterogeneous environment takes form. Here, the potential of pervasive (ubiquitous) computing is highly required. It allows the correct integration of sensors in dwellings to obtain the necessary data to understand the context. Assuming that the proper arrangement of sensors and equipment is known, the heart of the question finally lies in *which data and how this data is arranged, managed and reasoned to capture the context*. So, moving away from low-level problematics – framework, technology and system heterogeneity, *question a)* –, the main matter or objective of most of the current thesis is focused on context-awareness, but also dealing to a great extent with the other concerns (questions) about the selection of goals and applications, and the interaction with users. With regard to this issue, as an example, we can refer the reader to [DKM+04], where different technologies focused on HBA work together as a whole to perceive and recognize a global situation, they require extensive information about the environment to end up summarizing the context by means of abstract symbols.

In this point, to clarify some concepts covered by the holistic smart home becomes necessary. We started to do that in the introduction of the dissertation (Chapter 1), where we commented on the term confusion in the related literature due to the high heterogeneity of the field under discussion, also characterised by frequent overlaps and fuzzy boundaries among related concepts, fields, aspects, approaches and technologies. We try to clarify the matter with the help of the tower sketched in Figure 3.2.

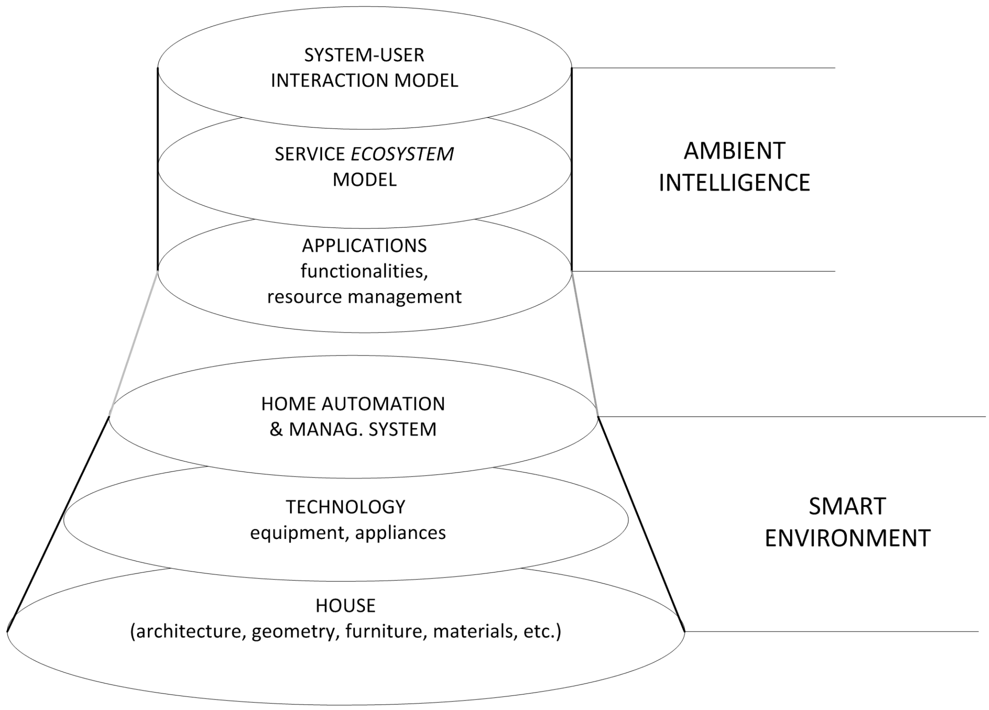


Figure 3.2: Smart Home Tower.

The first noticeable aspect of the tower is that we separate "Ambient Intelligence" and "Smart Environment" in our dissection of a "Smart Home" (the definition is provided in Chapter 1) . From now on we will follow the terminology proposed in [NAA10] and we will denote as *smart environment* the physical structure (sensor, actuators and network) that supports the system, whereas the term *AmI* is used for the mechanisms that control the behaviour of the environment. In [NAA10], authors compare AmI with the behaviour of a trained assistant (e. g. a nurse):

> The assistant will help when needed but will restrain to intervene except when necessary. Being sensible demands recognizing the user, learning or knowing her/his preferences, and the capability to exhibit empathy with or react to the user's mood and the prevailing situation, i. e. it implicitly requires the system to be sensitive.

However note that the term "smart" (in "Smart Environment") emphasises the desire of an adaptive, friendly and emotionally intelligent environment, beyond just problem resolution capabilities, i. e. "a digital environment that proactively, but sensibly, supports people in their daily lives" [Aug07]. It gives an impression of consciousness and behaviour, and leads us back to the AmI principles and hence to the semantic confusion.

This is because of the fact that *to separate the system behaviour and purpose of its body components* is difficult or not possible. We can even say that the implementation of the system's behaviour – AmI in our case – could be identified as the *psyche* of the system. For the definition of complex technical systems, information engineers distinguish between three models or layers: hardware, software and application. In this respect, the authors of [DFZB09] point out the question: "But what is the psyche in this scenario? The software? The application?" They conclude that this differentiation has not yet been defined, which partially explains the difficulties and ambiguities related to the terminology (at least in case of smart homes). For us, restricting to our field, the psyche of the system remains in the applications as they define *what the system does for the user*, but also in hardware and software as they build the foundation for *what the system is able to do*. In this respects, efforts to map psychoanalytical models that embrace intelligence, feelings and emotions together into intelligent machines or systems are faced in the interesting papers by Dietrich et al. [DFKU07] and [BDK+04].

In any case, once we are carefully aware of this strong connection, we can focus on the upper-part of the tower, i. e. the one that defines the system behaviour. The approach proposed in this thesis finds the way to optimise AmI in Smart Homes by the use of behavioural and habit profiles. To do that, the first step is to understand how users interact with the home environment through the smart system. A model describing the existent paths of feedback between the system and the users is depicted in Section 3.4. The model will be used as a frame to design human-centered applications to improve the system adaptiveness and acceptance. Later on, habits as input – i. e. objects of global reasoning to improve context awareness – is dealt with in Section 3.5. The complementary perspective, habits as goals, is explained in Section 3.6, giving special attention to social and psychological aspects of the home environment. Once defined the conceptual groundings, a system architecture built on the previous points is proposed in Section 3.7, i. e. a *service* ecosystem *model* for smart homes. Thus, the model is oriented to allow the coexistence of applications and the joined development of context awareness and global reasoning capabilities as well as pursuing adaptiveness and being sensitive to psychological aspects. Section 3.8 proposes a set of significant profile-based applications coherent with the presented models for holistic control based on habits. In Section 3.9, the mechanisms that allow profile-based services to check and adjust themselves are explained. Finally, Section 3.10 shows a case example where the introduced ideas and developments are implemented and can be assessed together.

3.4 System-User Interaction Model

Figure 3.3 shows a model to describe interaction paths between the home control system and the users.

As illustrated in the figure, the system obtains knowledge concerning users' existence and activities in two ways: directly and indirectly. The direct way covers actions deliberately carried out upon the system by users. Here, they are arranged forming three groups:

- *General adjustments.* Actions that affect different functions and services at the same time, or change the general character of the whole system, e. g. prioritizing energy consumption; switching on a *running mode* (e. g. 'winter', 'holiday', 'sleepings', 'relax'); activation or deactivation of services, parts or subsystems, etc.

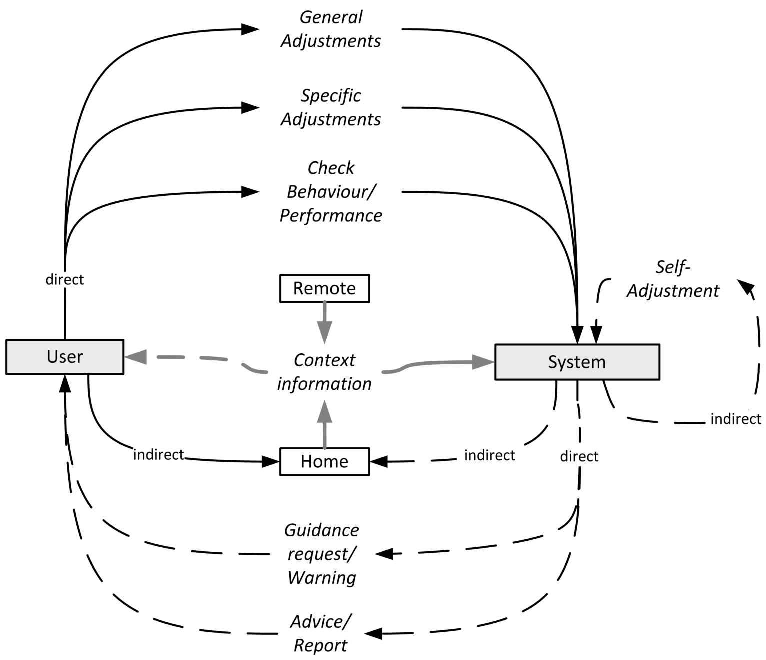


Figure 3.3: User-System Adaptation Model (feedbacks)

- *Specific adjustments.* Simple user actions with a specific purpose, e. g. switching on a lamp, adjusting the setpoint temperature of the room, connecting the dishwasher, etc.
- *Consult or check behaviour/performance.* Whenever users utilise the system to obtain information about their own behaviour or the house performance, e. g. readings of consumption levels (e. g. water, electricity, heating) and comparisons, but also any kind of consultation, as well as possible failures, predictions, health services, etc.

The indirect way represents information abstracted from users' behaviours and activities without the users being directly aware of it, e. g. occupancy habits, times that users have a shower per week, habitual schedule of kitchen appliances. This information is usually collected by sensors and, together with other home related information (e. g. room temperature, status of windows, consumption of devices, date and time, etc.) and information coming from remote actors (e. g. weather forecast, electricity spot prices), it gives shape to the context that the system is able to understand. The design introduced through this work uses behavioural and habit profiles to store and manage this kind of information.

Beyond specific control actions and commands (for instance switching on a light), the system must deploy the arriving direct and indirect information to trigger and perform the adaptive processes and routines that will ultimately lead it to fit users and their desires. These *adaptive actions* are also divided into three groups:

- *Self-adjustment.* Unsupervised tasks that the system executes according to its programming and the interpretation of the context.
- *Guidance request/warning*, which happens whenever a *relevant* conflict or a counterproductive situation is detected and the system can not solve it or requires guidance or additional information.

- *Advice/report*, including, for instance, the notification of minor conflicts, reports of energy consumption, advice concerning habits that could be improved, etc.

Self-adjustments are actions that are not oriented directly to users, but to the system itself, as well as expected, normal controller operations that perform programmed changes in the home environment (i. e. the indirect way that points at "Home" in Figure 3.3).

Therefore, direct actions performed by the system are specifically addressed to users. Between them, *warnings* and *advice* are also called 'active feedback' as they try to draw users' attention by text messages, pop-ups in User Interfaces (UIs), led indicators, etc. On the other side, *guidance requests* and *reports* are 'passive feedback', whenever the system generates information that remains at users' disposal waiting to be taken into account.

The introduced user-system adaptation model becomes relevant for the design of home services, systems and frameworks displayed in the next sections.

3.5 Habits as Inputs

In Figure 3.3, habits are firstly understood as *indirect context builders*. For us, habits are main constituents of the context, in a way that a lack of them identifies a *blind* environment. Beyond solving concrete control situations, an intelligent management system must be able to undertake global reasoning based on information abstracted from the context. Furthermore, it must develop logics that glue, manage and operate upon all applications in parallel as well as being conscious of how they interact and construct the common environment in which they are embraced. In this respect, repeated behaviours are common elements that build bridges between home applications, functionalities and user experiences.

By means of habit abstraction the system in charge of managing the operation of appliances and technology equipment, running as an overall intelligence, obtains a better knowledge regarding its inhabitants and their relationship with the dwelling. *Habit profiles* play this role, i. e. they become *shared* objects of global reasoning and, consequently, the main elements that store the habit abstraction in the HBA context. The profile quality of being *shared resources* is fundamental as it allows the fluid and coherent connection and coordination among different parts, services and applications.

Figure 3.4 shows the connection between different services and a shared profile, as the control of lighting, blinds, heating, entertainment devices and UIs can require *occupancy profiles* in the decision making processes. A singular service can also utilise other habit profiles (e. g. comfort temperature profile for heating control, lighting level profile for lighting control, or room consumption profile for the generation of *user reports*) as well as other habitual inputs not recorded as profiles.

Advanced control based on profiles can be independent of classic specific runnings, work in parallel, be superimposed or can also require a redefinition or revision of the application control and design. It is not rare that certain logic that usually operates in an automation level must be delegated to the management layer, into the hands of agents that are able to understand the implications comprised by profiles. We are reminded here that definitions, classifications and requirements of profile-based services to be integrated into HBA systems are explained in Section 2.

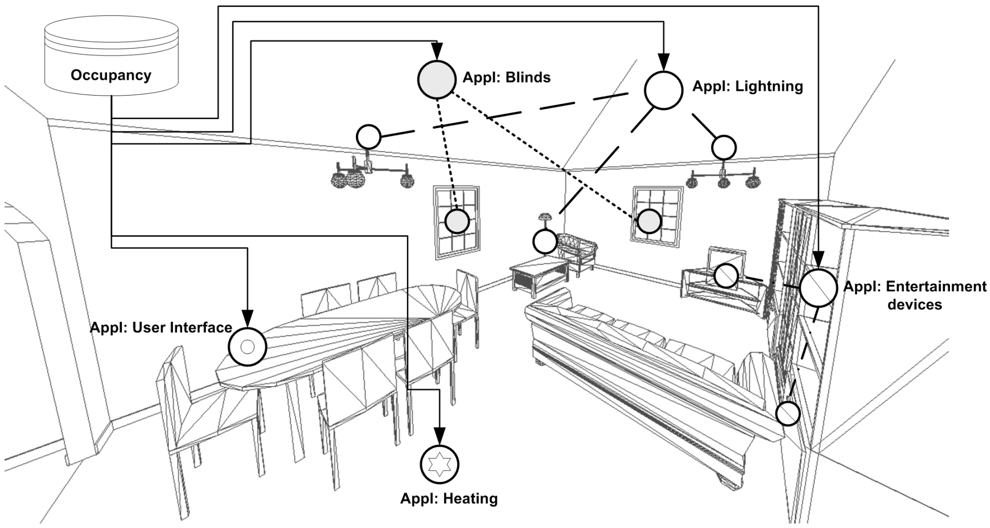


Figure 3.4: Different applications take advantage of occupancy profiles (shared resources) to perform control adjustments and decisions.

User habits contain information about how users want things to be at home – how home is expected to work – and how the relationship among applications, services and usages must be; in other words, they link the diverse home experiences in a common, fluid, consistent layer. In front of complicated situations where conflicts between services happen, the system can simply make decisions consulting how users managed the same or similar situations in the past, what they usually do, or latest user adjustments after previous automated actuations.

In short, habit profiles are intended to support the next functionalities:

a) Abstraction of the smart home expected performance.

b) To merge the operation of applications in a common context.

c) Feed-forward and predictive control.

d) Supply of feedbacks based on past experiences.

These four points are found mandatory to provide to the system flexibility, self-checking skills and the capability to gradually adapt to users.

Table 3.1 shows a simple, quick example where the use of habits culminates in outcomes for the four introduced functionalities. Note that habit profiles generate in advance an amount of context information that would be missed otherwise. The system works with suppositions and predictions, that can obviously fail or sometimes be inaccurate. Nevertheless, profile-based approaches can evaluate the reliability and likelihood of the employed patterns, and also carry out alternative routines or strategies in case predictions are not good or habit patterns produce lax or unreliable results (see Section 3.9).

Profiles are emphasised as shared resources due to the fact that they present improved performances when they are used in a cooperative, conscious and coordinated manner. Since instantaneous variables are often used simultaneously by different services, they can also be seen as

Variables: ▹ Present Time = 17.45h. ▹ Instantaneous Occupancy = 0
Patterns: ▹ Occupancy profile → Next occupancy at 19:00h. ▹ DHW profile → Next demand at 19:30h. ▹ Electricity consumption → low from 18:00 to 19:00h; very high from 19:00 to 21:00h.
Abstraction of the smart home expected performance · Comfort heating levels are expected as of 19:00h · Hot water will be demanded between 19:00 and 20:00h · A high electricity consumption is expected between 19:00 and 21:00h · Dishwasher must be activated by 9:00h of tomorrow
To merge the operation of applications in a common context · Home services are in a *Preparation* period. · Postpone dishwasher operation to a time slot with lower electricity consumption. · Next occupancy is expected for 2h, hot water recirculation can be switched off at 21:00h. · After 21:00 home services will be in a *Short Absence* period.
Feed-forward and predictive control · Switch on heating at comfort levels when *preparation.* · Switch on hot water recirculation 10 minutes after *preparation.*
Supply of feedbacks based on past experiences · "Electricity consumption is unbalanced, a sharp peak is habitual during evenings". · "The usual demand of DHW fits sustainable managements".

Table 3.1: Example of functionalities accomplished by the use of habit profiles.

shared resources, but note that they are usually independently utilised without being aware of the use given to the same variables by other services. In the end, the arrangement of a set of home habit profiles generates a richer context definition to be explored by global reasoning or a control pre-phase.

In profile-based design, global reasoning is embedded into an intermediate, conscious layer placed between users and the equipment (or end, specific controllers). Considering a system built on a MAS structure and an ontology, a conceptual overview of the intermediate layer can be seen in Figure 3.5. The objective of this layer is to facilitate the work for end controllers, optimise the combined operation of appliances, guarantee the inclusion and compatibility of different mechanisms and manage possible conflicts that appear. The pre-control phase thus optimises the operation of the whole system regardless of the installed equipment and deals with the different features of the available technologies in a flexible manner.

As it has been shown, profile-based control and holistic management perfectly fit implementations on MASs and ontologies, but they are not strictly bound to such approaches. A deeper development of some profile-based applications on such structures are shown in Chapter 4.

3.6 Habits as Goals

We have introduced the capability of smart homes to improve system's context awareness by means of users' habits. In order to close the loop, we will now discuss the capabilities of smart

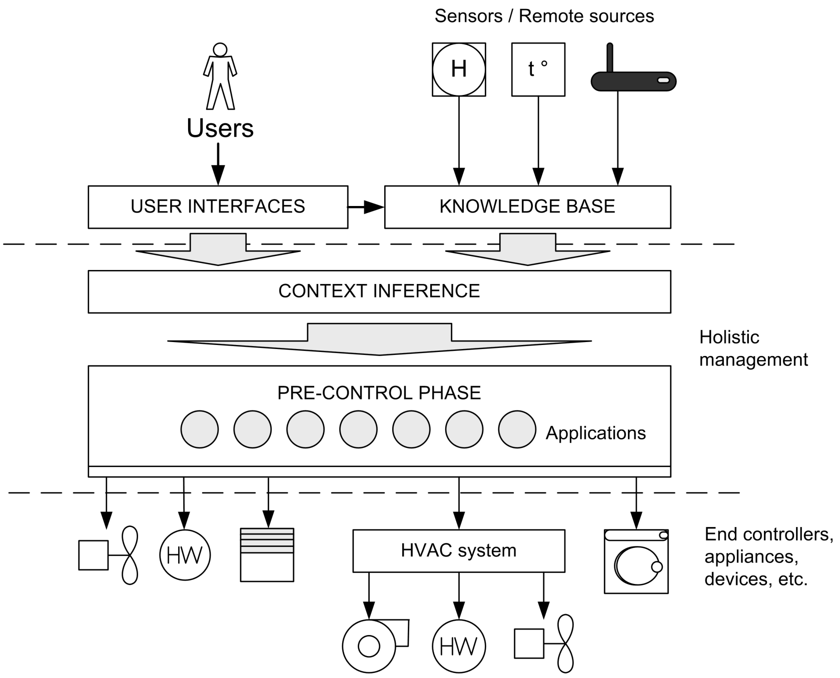


Figure 3.5: The holistic control adapts to the underlying equipment and takes care of the optimisation in a pre-control or high-level control phase. The reasoning behind the pre-control phase is based on a global evaluation of the context realised by the cooperative running of all home applications.

homes to cause behavioural change (Figure 3.6). We interpret users as actors who eventually follow negative trends or ignore information that can significantly improve their quality of life.

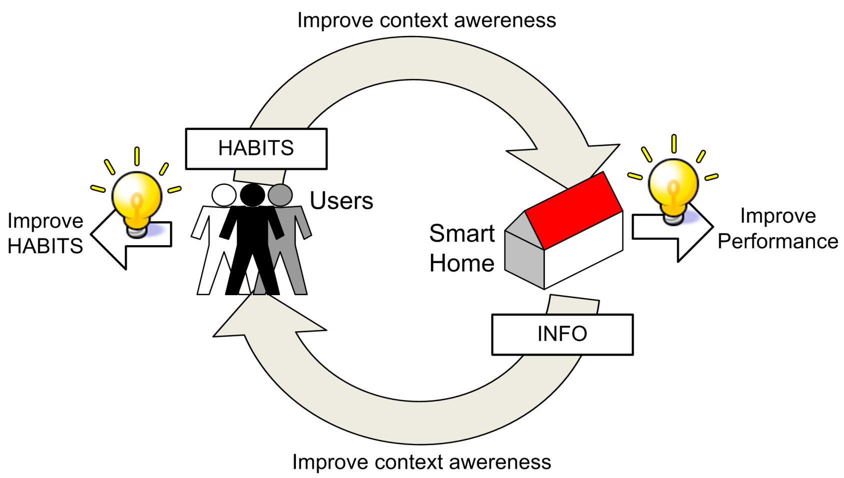


Figure 3.6: In the diagram, habits are seen as inputs, but also as goals, in a recursive loop that dynamically tends to reinforce the achievement of benefits in the performance.

From the perspective followed in this thesis, improving users' behaviours is considered as a relevant, promising approach – although often being underestimated –, not difficult or exhausting for users, and even pleasant or motivating provided there are suitable designs. Assuming that the system must not bother or overwhelm users, it should keep available to services oriented to empower behavioural change. These assessments can be obvious, but also require being grounded by psychological studies.

3.6.1 Psychological Background

If we look at the scientific literature, in [VA99] authors begin pointing out the classic low attention given to habits, habitually considered as "a construct of marginal interest". They are critical with this opinion and stress the importance of habits as study objects to understand the principles of automaticity in social psychology. With their words:

> A habit seems to be accompanied by an enduring cognitive orientation, which we refer to as "habitual mind-set", that makes an individual less attentive to new information and courses of action, and thus contributes to the maintenance of habitual behaviour. Focusing on habitual mind-sets and automatic cue-response links, rather than on statistical associations between past and future behaviour, makes habit an interesting construct for future research.

In the cited paragraph we find the two natures of habits practiced in this thesis. On one hand, "the statistical associations between past and future behaviour", which, although being criticised in the quote, corresponds to the classic way of studying habits. It becomes a valid, ongoing approach to make an automated system able to learn and interpret human behaviours. On the other hand, "focusing on habitual mind-sets" brings to view our current, second perspective. It sees habits as aims to ameliorate, according to the fact that they are motivated by potentially educable "mind-sets".

The way of improving habits by technology is providing persuasive information that is useful and important for users. The association between habits and users' goals has been shown, being the automaticity in habits conditional on the presence of an active goal [AD00]. It extends the relationship of habits with "users' goals" besides "user's mind-sets". The conclusion is that habits can be changed or improved as long as users are concerned about the pursued goals (i. e. sustainability, health, comfort) and want to actively do something about it. A good example can be seen in [GB01], where conducted surveys show that environmental values create a predisposition to change purchase habits, specifically concerning eco-labelled products.

In this respect, in [Boy06] the author tackles the issue of *intentional change* and explains how sustainable desired change can occur at all levels of social interaction, from individual to wider scopes (teams, organisations, communities, etc.). Here, intentional change is seen as a *self-directed learning* that can be explained by means of concepts from complexity theory. Boyatzis offers key aspects to analyse for the optimisation of designs that must empower sustainable intentional change. They are based on studies that confirm that "adults learn what they want to learn", so most sustainable behavioural change is intentional.

Within the domestic scope, only an environmental predisposition does not guarantee the acceptance of services oriented to support behavioural change, additional aspects must be also considered in the technology design. For example, under this perspective, the concept of *self-efficacy* plays an important role in the design of persuasive systems. Psychologist Bandura defined *self-efficacy* as *one's belief in one's ability to succeed in specific situations.* Therefore, future, supportive ambient intelligent environments must not make the development of users' self-efficacy difficult, as well-being, accomplishments and autonomy are intimately related. As far as habit management is concerned, it is hypothesised that expectations of personal efficacy determine whether coping behaviour will be initiated [Ban77]. Also the credibility and confidence in the source of information affects the effect of persuasion, since the recipient's initial opinion about the source is determinant [SL78].

In short, we have scientific evidence to think that if users *control* an environment that is motivating, reliable, pleasant and tailored to them, in all probability they will be prone to accept and follow advice coming from the environment and concerning the issues they care about (we mainly consider: sustainability, health and comfort).

3.6.2 Impact of Information in Users. Persuasive Technologies

For the reasons previously exposed, the perspective adopted in the current thesis considers *informativeness* as one of the main aspects to face for the design of future smart homes. Within the technology context, 'informativeness' must be understood as the "system's capability of being informative to users". This approach integrates users as end elements of the control loop, able to carry out the better, desired performances by themselves. A wrong interpretation of the concept is to guess systems that overwhelm users by means of data bombardment. The correct perspective remains in systems that have meaningful, qualitative information at users' disposal, providing them in the correct moment and according to their requirements. To do that, informativeness must go along with other relevant system features like usability and adaptiveness.

We opt for informativeness instead of the related, far synonymous, term *persuasiveness*. Indeed, the former has softer, low-intrusive connotations and retains the positive nuance of awareness in users actively and willingly accessing or controlling home services. *Persuasive technologies* covers a wider scope, but in any case they also ultimately pursue the education, redirection and amelioration of users' habits. The groundings of persuasive technologies as well as deep discussions and multiple applications concerning this field are offered by B.J. Fogg in his book, *Persuasive Technologies: Using Computers to Change What We Think and Do (Interactive Technologies)* [Fog03]. Here, B.J. Fogg does not avoid the most controversial and thorny aspects of persuasive technologies, but discusses them, explores ethic implications and even deals with how to measure and evaluate them.

As the title of the book states, since the goals of persuasive computing are to change what we think and do (behaviour and habits), they are unequivocally attached to profile-based or habit-based applications. Therefore, smart systems must be aware of habits, and collect and evaluate them, in order to perform informative actions oriented to their change or reinforcement.

One of the best examples of smart home designs that follows such ideas is presented in [Int02]. The team headed by S.S. Intille envisions the home of the future as a digital environment that "should require human effort in ways that keep life as mentally and physically challenging as possible as people age". Instead of systems that serve users (and somehow can even spoil them), the goal is to develop homes that empower users' control and self-sufficiency. With the author's words:

> Based on discussions with medical professionals, patients, educators, and homeowners, we believe that the home of most value in the future will not use technology primarily to automatically control the environment but instead will help its occupants *learn how to control the environment on their own.*

In order to achieve such motivating objectives, in the same article different information- and habit-based home applications are shown. Let us see some examples or cases where just the proper publication of information has resulted in important benefits in terms of sustainability (energy savings) and users' health, safety, quality of life and productivity.

Enhancements in Energy Efficiency

One of the most promising areas to achieve improvements by means of feedback to users is electricity use. Nowadays, most electricity markets do not consider consumers as active elements capable of adopting optimised strategies and decisions but simply as loads to be continuously supplied [Kir03]. Persuasive and informative applications are called to improve the present low elasticity of short run prices [YD02], contributing with substantial benefits and savings. Actually, the collection of strategies oriented to improve the use of energy given by users is usually embraced by the term DSM [Kir03].

Under this scenario, home control systems are expected to work in three levels:

a) Reducing energy consumption.

b) Displacing consumption from peak to valley hours, i.e., tending to flatten the consumption [KSCPM00].

c) Giving real-time information to users (advice and instructions) about energy prices, energy resources and demand evolution [ZASZ+10].

In this respect, the interesting survey carried out in [Fis08] explores whether feedback on household electricity consumption is a tool for saving energy, analysing which features must present feedbacks and feedback-supplier systems to be successful.

Moreover, there are several works that publish achieved saving rates by providing instantaneous feedback on household electrical demand. For instance, 22% is advanced in [LEMM09]. An average 7% is obtained during the second year of application of a feedback system installed in 20 case study homes in Florida [Par08]. In [FSS10], the reduction in consumption is also estimated in 7% when prepayment of electricity is not involved, being twice this amount when users obtain benefits from a tailored prepayment system. 21% savings are obtained by feedback for clothes washing tasks in [MM02]. Finally, qualitative analysis of how householders interact with feedback from smart energy monitors in UK can be seen in [HNB10].

Note that such reductions should not be considered only in a linear way, i. e. as a proportional estimation of energy and economic savings. As we have commented above, energy feedback is not only oriented to motivate reduction in consumption but also displacement. It means that, even in case there would not be a net decrease of total consumption, the benefits obtained by peak shaving are actually considerable [FHNP07].

On the other hand, although people gradually adopt new, energy-efficient technologies, a common habit detected in energy appliance consumers is that they usually do not get rid of the old machines, but put them in another place or give them to relatives or friends. In order to avoid that, some country and regional governments have launched renovation plans for electrical appliances, encouraging users to recycle and to not keep using old equipment, i. e. [IDA08]. In that regard, note that feedback and energy comparisons can support such initiatives as they are able to warn concerning failures, devices that are consuming too much or malfunctioning.

Finally, in spite of the fact that we have stressed the electricity case, feedback to users can also be very effective in connection with other energy sources, e. g. water [ABS05], gas [MMWZ83], [VHVR89], etc.

Enhancements in Quality of Life

The effect of information must not only be evaluated by the capability of obtaining energy reduction and a better use of energy, but also to improve users' quality of life. For example, pervasive technology designs that support motivation are found promising to empower behaviour change in health applications [AFB+11].

In this respect, there are a vast number of publications related to AAL, i. e. smart environments designed to improve the quality of life of people that require special care, e. g. disabled or elderly. For example, the edition of October, 2011 (number 87) of the ERCIM (European Consortium for Informatics and Mathematics) journal is fully dedicated to AAL, presenting an interesting collection of related articles, and headed by a keynote entitled: *Ambient Assisted Living and Ambient Intelligence: Improving the Quality of Life for European Citizens* (by Constantine Stephanidis), where a specific stress is set on the user-system interaction. More recently, the *Handbook of Ambient Assisted Living* [AHK+12] collects an extensive compilation of articles, where the relationship with the smart home is specially considered, and feedbacks to users provided by smart environments are carefully dealt with and depicted.

In [KBR+07], authors present a laboratory for AAL solutions which is intended to train elderly people to handle modern interfaces and get used to smart systems. User acceptance and support by various user-interfaces are found as aspects of absolute necessity, hence the evaluation of the suitability and usability of home technologies becomes mandatory. Again, usability and adaptiveness are relevant aspects to insist in future research. Regarding this point, some works emphasises the good reception of older adults, for example: [DRA+04]; whereas other works express the existing disagreements of distinct stakeholders for the evaluation of smart home technologies, for instance: [SSBW08].

Moving away from special care scenarios, we should remind ourselves that informative systems are also useful to improve quality of life in all kind of homes, offices and buildings. As an example, for the HVAC scope, diverse researches have shown that productivity can be further enhanced if occupants are in direct control of their indoor climate [KC04], provided users are properly motivated and informed [Wyo00]. Moreover, safety is also a mandatory field to solve and correctly develop in buildings, efficiently fulfilling requirements in terms of usability and informativeness [KN09].

In short, the introduced works point to establish feedback-to-users, informative systems and habit-oriented services as mandatory aspects or features to consider in the design of future smart homes. As we have referred above, assumed this fact, the challenge now is to design services that show meaningful information in the properly, comprehensible and effective way and in the correct places and times.

3.7 System Architecture

From the holistic perspective, the smart home *system* is understood as a set of connected technological services that share a common domestic space and work in a cooperative manner. These services are formed by processes designed to accomplish specific tasks, but also to keep the coordinated service and achieve global goals pursued by the system as a whole.

This section proposes an example of an underlying architecture that supports holistic profile-based smart home designs. The main required components are: a) self-checking shadow processes and b) a multi-agent environment.

3.7.1 Shadow Processes

Among the less-visible processes intended to coordinate the overall execution, we place parallel or *shadow processes*, whose objective is the validation of the performance of a normal service and the subsequent decision to improve it (in case it is necessary). The criteria for validation is established in terms of users' agreement, level of satisfaction, as well as considering accuracy, effectiveness and efficiency (whenever appropriate and depending on the specific application). Figure 3.7 shows the schema of a generic shadow process.

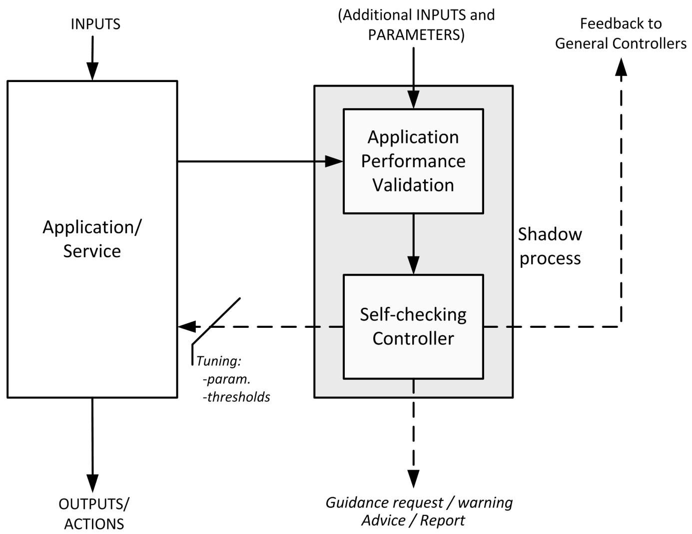


Figure 3.7: Scheme of a shadow process.

In the figure, the shadow application shows two differentiated blocks:

- *The application performance validation* block, which is in charge of validating the performance according to rules, thresholds or context information. Input arrows represent context data, which are collected from the original application but can also require other additional sources (e. g. user preferences).

- *The self-checking controller* block. The evaluation provided by the validation block is used by the controller to decide if it carries out an adaptation-oriented action. According to the model depicted in Figure 3.3, these actions can be: 1) automatic adjustment of the source application (e. g. parameters, variables, switching algorithms, etc.); 2) actions addressed to users, i. e. reports, advices, warnings or asking users for guidance; or also 3) feedbacks to general controllers.

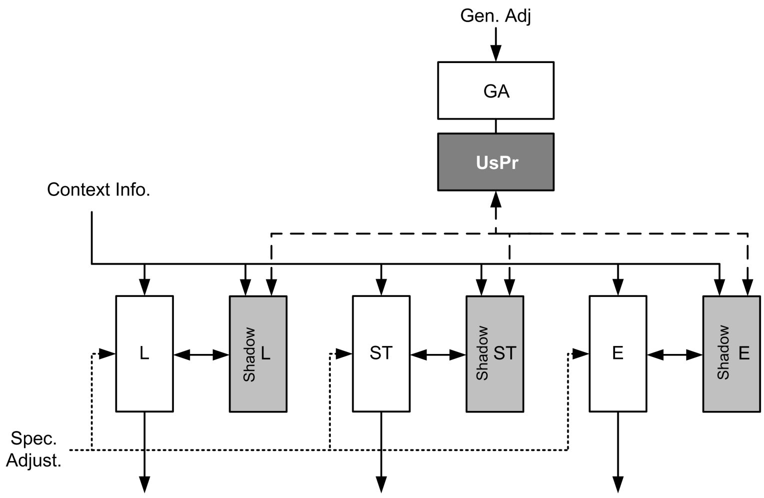


Figure 3.8: Example of scheme of services and blocks for a set of random applications.

Shadow processes require parameters and are susceptible to tuning or adjustment. They add extra responsibilities to general purpose controllers (e. g. a *User Preferences* module, UsPr) that will be in charge of the adjustment of shadow processes. This module collects the customised *modes* and *selections* configured by users (or by default instead) that we have addressed above as *general adjustments*. Based on this configuration the system is able to reason how the parametrisation of shadow processes must be tuned.

Figure 3.8 shows an example of relationships among normal services, shadow processes and the User Preference module. In the figure, L stands for the *Room Lighting Control* service, ST for the *Setpoint Temperature Control* service and E for the *Control of Energy Consumption* service; GA is intended to allow users executing *general adjustments* using the different, available UIs. The mentioned applications are illustrative, but have been taken from the set of profile-based applications explained later in Section 3.8.

3.7.2 Development in a Multi-Agent Structure

Figure 3.9 sketches a multi-agent framework for a smart home where profile-based services, holistic reasoning and shadow processes are implemented together.

In the figure, the central white circles with the thick border form the potential system functionality. They are decision makers and controller agents and represent the specific applications and services that are developed for the smart home. Note that they are usually supported by agents in charge of shadow processes (dark circles), which have the task of controlling the performance and adjust controllers.

The KB (represented by an ontology or even by simple databases) stores dynamic data (instantaneous variables), and also patterns and profiles. PTGs and PFGs are implemented inside dedicated agents. In addition, distinct intermediate agents exist to carry out transformations or profile-based interpretations of the context; they usually serve controller agents or general purpose agents. Finally, global reasoning is undertaken by general purpose agents that act as referees as they establish priorities, switch control algorithms and adjust general parameters.

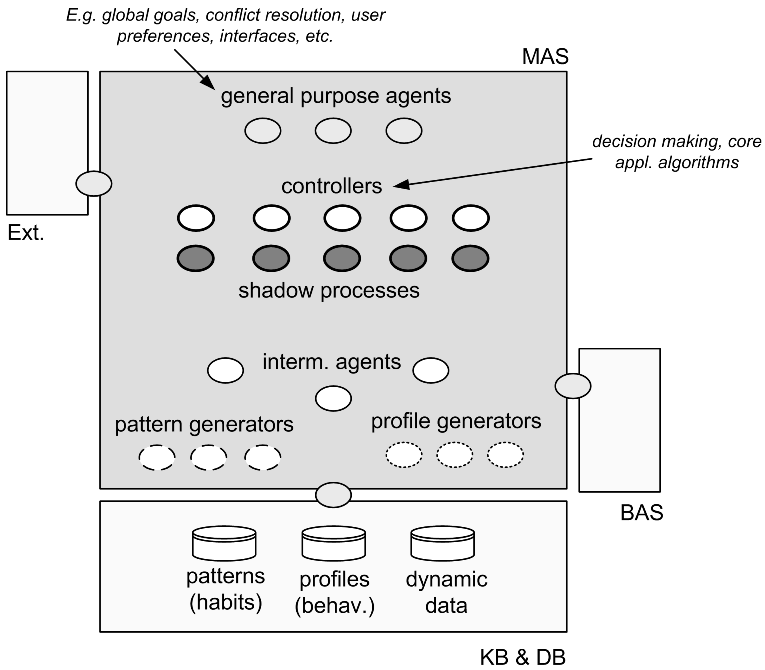


Figure 3.9: Schema of a profile-based MAS. Ext. represents remote sources of information.

Some examples or general purpose agents are: *global goals* agents, *conflict resolution* agents, *user preferences* agents, but also agents that act as interfaces and gateways with other networks and physical environments (e. g. KB, remote servers, UIs, HAS or BAS networks, etc).

Agent based control for smart homes is dealt in [RK11]. Moreover, a detailed example of the introduced, specific architecture is widely explained in Chapter 4 with an example that covers the fields of air quality and thermal comfort.

3.8 Applications Based on Habit Profiles

This section introduces a set of profile-based applications for smart homes in keeping with the concepts previously presented. The applications cover representative areas and services that are expected to be carried out by HAS. On the other hand, the integration of classic applications and services not-based on profiles within the same framework is assumed.

A noteworthy feature of the presented applications is that they do not demand either a high-sophisticated equipment or unreachable or futuristic technologies. They are feasible and deploy usual equipment and systems at our disposal in the current market. As a requirement for the profile generation, the applications demand intelligent field devices able to communicate their status to the smart control by themselves or by means of intermediate interfaces. In any case, note that not every home appliance and device has to be represented in the smart system by profiles, the final design is flexible and open, and the characteristics of each specific case will state which devices are suitable to be integrated with profiles.

Similarly, the design of the applications is representative but obviously not unique, being the number of profiles, the selected ones, their features, the control design and the control aims open to discussion and alternative approaches. Table 3.2 shows a selection of possible, habitual profiles that can be collected in the home environment. The services explained in this section together with the profiles they require are listed in Table 3.3.

Profile	Object	Type	Value	Scope	Sample rate
– Light status	device	usage	on/off	daily	30'
– Light level	device	level	%	daily	30'
– Lighting user readjustment	group	specific	$\mathbb{N}$	daily	1h
– General lighting command	group	usage	$\mathbb{N}$	daily	1h
– Dwelling occupancy	dwelling	occupacy	yes/no	daily	30'
– Zone occupancy	dwelling	occupacy	yes/no	daily	30'
– Blinds level	device	level	%	daily	30'
– Blinds user readjustment	group	specific	$\mathbb{N}$	daily	1h
– General blinds command	group	usage	$\mathbb{N}$	daily	1h
– Alarm status	dwelling	status	on/off	daily	1h
– Comfort relative humidity	dwelling	level	%	daily	30'
– Comfort temperature	dwelling	level	$\mathbb{R}$	daily	30'
– Window status	device	usage	open/close	daily	30'
– Rel. humidity user readjustment	group	specific	$\mathbb{N}$	daily	1h
– Temperature user readjustment	group	specific	$\mathbb{N}$	daily	1h
– Hot water demand	dwelling	level	%	daily	30'
– Device consumption	device	consumption	$\mathbb{R}$	daily	30'
– Device status	device	usage	on/off	daily	30'
– Pow. line consumption	line	consumption	$\mathbb{R}$	daily	1h
– Pow. line status	line	usage	on/off	daily	1h
– Dwelling elec. consumption	dwelling	consumption	$\mathbb{R}$	daily	1h
– Dwelling gas consumption	dwelling	consumption	$\mathbb{R}$	daily	1h
– Dwelling water consumption	dwelling	consumption	$\mathbb{R}$	daily	1h
– Dwelling DHW consumption	dwelling	consumption	$\mathbb{R}$	daily	1h
– Elect. spot prices	dwelling	consumption	$\mathbb{R}$	daily	1h

Table 3.2: Some common habit profiles.

3.8.1 Room/Zone Lighting Control

According to [DiL05], lighting control can perform seven discrete functions: on/off, occupancy recognition, scheduling, task tuning, daylight harvesting, lumen depreciation compensation, and demand control. The integration of lighting by means of profiles improves the performance of such functions as they become more adaptive (Figure 3.10). Some profile-based services related to lighting control are introduced as follows:

– *General Command for Lighting Groups*
Given a structure of groups for the home lighting where lights are physically or logically associated (i. e. living-room lights), a profile-based service can automatically adjust the status and levels of the group. A *general command* triggers the service and rules over the group, whereas specific lamp commands are deployed by users to refine the desired customisation whenever it is necessary. Thus, the system progressively learns how it must be the lighting combination of the group for every moment of the day according to the information stored in profiles.

– *Self-checking: Lighting Management*
Additionally, general and specific commands are monitored and related to usage profiles in order

Service	**Required profiles**
– General Command for Lighting Groups	Light status (each light) Light level (each dimmable light)
– Self-checking, Lighting Management	Lighting user readjustment General lighting command
– Efficient Lighting	Light status (each light) Light level (each dimmable light) Zone (or Dwelling) occupancy
– General Command for Blinds Groups	Blinds level (each blind)
– Unsupervised Control for Blinds Groups	Blinds level (each blind)
– Thermal Comfort Support	Occupancy (room, dwelling, zone) Comfort temperature Comfort relative humidity
– Self-checking, Blinds Management	Blind user readjustment General blinds command
– Unusual Presence Detection	Dwelling occupancy
– Oversight Detection	Alarm status Dwelling occupancy
– Setpoint Temperature Control	Comfort temperature Dwelling occupancy
– Setpoint Relative Humidity Control	Comfort relative humidity Dwelling occupancy
– Self-checking, Temperature Management	Temperature user readjustment
– Self-checking, Rel. humidity Management	Rel. humidity user readjustment
– DHW Recirculation	Hot water demand
– Customized Billing	Dwelling demand
– Elec. Peak Shaving	Device demand (each device) Device status (each device) Dwelling electricity demand Elect. spot prices
– Check Energy Behaviours – Smart Home Customisation	Device demand (each device) Device status (each device) Line consumption (each line) Line status (each line) Dwelling electricity demand Occupancy (house, room, zone) Comfort temperature Comfort rel. humidity General lighting command General lighting command Lighting user readjustment Blinds user readjustment Window status (each windows) Rel. Humidity user readjustment Temperature user readjustment Others...

Table 3.3: Profile-based services and required profiles.

to perform evaluations of the system-user interaction. This application measures the level of user satisfaction based on the rate of re-adjustments after the activation of a general command. A tendency to decrease is expected as the system gets adapted to users' habits and desires (if there are not manual re-adjustments, old habits are reinforced); otherwise, the system would ask for assistance (preferences, customisation, disconnection of services) by means of *warning messages* in ambient indicators and home GUIs. Self-checking capabilities are explained in Section 3.9.

– *Efficient Lighting*
Designed as an AAL service, lighting activation and regulation can also be directly triggered by occupancy and habitual scheduling. This is implemented by the *Efficient Lighting* service, which is aware of occupancy and lighting usage habits and performs a smart, unsupervised management of house lighting (becoming aware of preferences, possible oversights, unexpected absences, etc.).

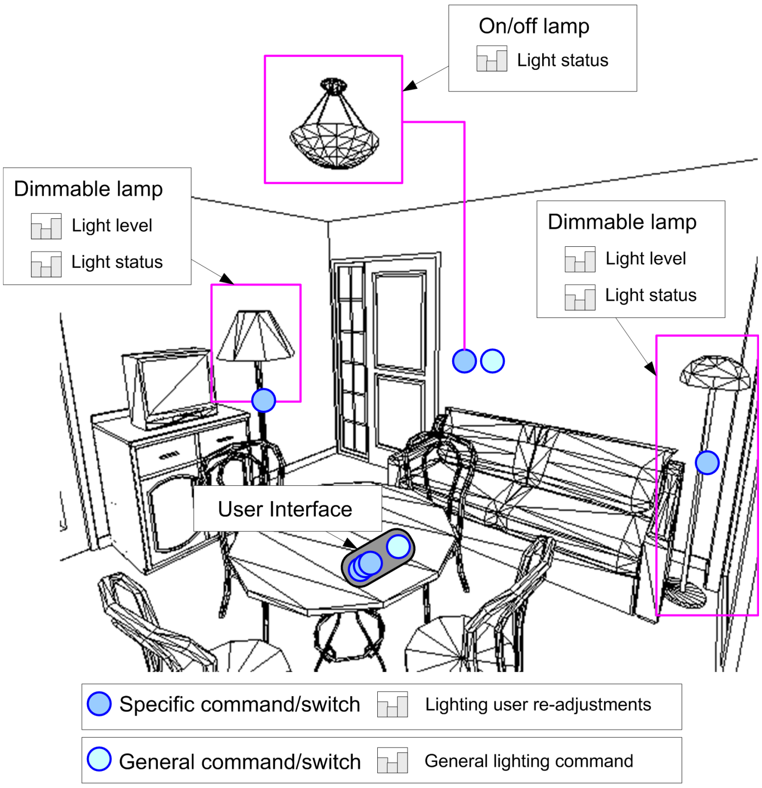


Figure 3.10: Case example for lighting control. The *general command switch* loads the status pointed by the profiles. *Lighting user re-adjustments* and *general lighting command* profiles are mainly utilised to evaluate the level of users' agreement.

3.8.2 Automatic Control of Shading Devices

In automated dwellings, blinds and shutters are usually controlled by specific commands that state the desired position (rolled, unrolled or partially unrolled); in addition, there are some shading devices whose slats can spin to regulate the sunlight filtration (an example of possible operation for blinds can be seen in Figure 3.11). The status of blinds and shutters have an influence on the visual and thermal comfort conditions, in a way that the control of heating, cooling, ventilation and lighting is partially affected. Profile-based services are set to optimise the overall running and avoid conflicts.

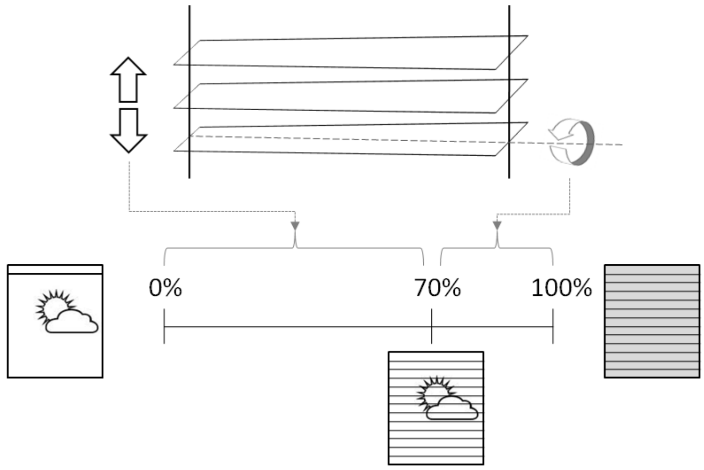


Figure 3.11: Example of operation for automatic blinds. Given this scenario, only one profile is necessary to save the blinds level.

– *General Command for Blinds Groups* and *Self-checking, Blinds Management*
General Command for Blinds Groups and *Self-checking, Blinds Management* have a similar running to the analogous services introduced in Section 3.8.1. It is worth pointing out that, whereas in lighting the simpler general command for groups distinguishes two states: 'on' and 'off', here it should contemplate three states: 'open', 'normal' and 'close' (where 'normal' is fixed by profiles).

– *Unsupervised Control for Blind Groups*
Some users appreciate or make the most of an unsupervised management of blinds and shutters. Whether this service is active, it operates shading devices according to profiles without user supervision. In this case, a *rate of actuation* parameter establishes how often the system consults profiles and executes changes.

– *Thermal Comfort Support*
In unoccupied periods where next user presence is approaching in time, advanced thermal comfort strategies can opt for managing blinds, curtains or automated windows, so as to balance indoor and outdoor climate conditions or improve indoor insulation; therefore, energy savings are achieved as cooling, heating or forced ventilation are not unnecessarily switched on. This is implemented by the *Thermal Comfort Support* service. This service is active in unoccupied periods in order not to bother users or cause conflicts with their desires, in occupied periods it elaborates their outputs as recommendations or advice published in the home GUIs.

3.8.3 Supporting Security System

Alarm systems must be activated and deactivated from specific commands, usually requiring keys or codes. A supporting security system based on profiles can be developed for dwellings without contracted alarm systems as well as to offer extra capabilities to the normal installation.

– *Unusual Presence Detection*
When it is active, it deploys occupancy profiles to warn users about unusual presence (through sms or email), regardless of whether the alarm system is active or not.

– *Oversight Detection*
This application stores alarm status in profiles. Therefore it warns users when they forget to activate the alarm system, provided that it detects some incongruence in alarm activation habits. In addition, the system offers the possibility of automatically activating the alarm system if a long absence is predicted.

Advanced security systems offer by default similar services to the introduced ones. In any case, to incorporate such services into the holistic control is advisable as the rest of the system becomes aware of the expected security performance and possible simultaneities, conflicts and overlaps are better dealt with.

3.8.4 Air Quality and Thermal Comfort

Both *air quality* and *thermal comfort* (AQTC) are phenomena that depend on objective, but also subjective, context evaluations, and both issues are usually covered in houses and buildings by HVAC equipment. There exist specific norms and directives in each country that establish requirements and sizing for the installation of such systems. Basically, AQTC control depends on temperature, humidity, wind (and also diverse gas and other pollutant) sensors, as well as on users' subjectiveness. Therefore, HVAC performances are fundamentally dependent on user setpoints in addition to levels fixed by pre-defined calculations and rates.

Profiles can be very useful to improve the performance of advanced HVAC techniques as they predict the desired setpoints in keeping with the expected habits. Here remains one of the most favorable aspects of the profile-based control, the fact that the subjectiveness of *thermal comfort* is overcome by the *particular feeling of comfort* stored in habit profiles. It is worth remarking that the next applications are superimposed to the habitual system controllers with minimal variations (since they operate in a previous stage to the genuine HVAC system). The services introduced here are deeply developed in the next chapter (Chapter 4).

– *Setpoint Temperature Control* and *Setpoint Relative Humidity Control*.
The system switches to comfort (comfort profiles), off, safety, setback or setforward values depending on instantaneous occupancy and occupancy profiles. Occupancy profiles are deployed to guess the next occupancy and reach anticipated comfort conditions before people arrive. They also know if the next absences will be long or short and, based on this evaluation, switch to the appropriate control status. The *Setpoint Relative Humidity Control* has an analogous running but deals with relative humidity levels.

Both applications fix thermal comfort conditions having learned the subjective taste of users. Over this basis, additional information/variables help to improve the final control/response (i. e. values can be finally adjusted depending on the number of people present, type of room, schedule, outdoor temperature, etc.).

– *Self-checking, Temperature Management* and *Self-checking, Relative Humidity Management*
Both service work like other self-checking services introduced before (e.g. Section 3.8.1).

3.8.5 Domestic Hot Water

DHW distribution systems usually present low-rates of energy efficiency [Hil05]. Centralised systems are designed to provide district heating to buildings (e. g. hotels, hospitals, sports centers, social facilities, or also multi-family residential or apartment buildings). In such scenarios, they have to deal with the fact that distances covered by water from boilers to consumption points are usually long. In order to avoid unnecessary waste of water and user discomfort, such systems normally have recirculation loops intended to keep hot water close enough to fixtures. It entails severe sources of inefficiency, e. g. heater loss, recirculation loop loss, branch loss or wasted water [Bon10]. Here, profile-based operational strategies can be applied in order to achieve an optimised management of resources.

– *DHW Recirculation*
Using profiles that store the probability of hot water demand of a flat or building zone, boilers and recirculation pumps can be adjusted to prepare comfort conditions in advance, preempting the requirements and making the most of demand simultaneities. Combined with *demand recirculation* strategies [LKSH02], the obtained performances can reach excellent rates of users' satisfaction, water savings and energy use.

3.8.6 Energy Consumption

Smart metering applications usually record electricity consumption profiles for monitoring and billing purposes. In addition, consumption profiles are useful to prepare reports for users in order to inform them about device consumption, energy behaviour and failure detection. The convenience of energy feedback for users can be consulted in [Fis08]. Consumption profiles can be employed for a more customised and profitable relationship between service providers and consumers [ZASZ+10], as well as to calculate realistic *simultaneity factors* for power, water and gas, allowing the sizing of equipments, predictive configurations, or to optimise the distribution if there is some kind of connection with the grid and communal systems [GSR10]. In short, energy profiling is not only important for the internal control of the smart home, but it can be utilised for external actors in a way that the smart home is transformed into a new kind of service provider. Remote entities, after managing the data from multiple linked houses and buildings, will be able to perform useful benchmarks and assessments addressed to the final home optimisation (see Section 2.1.2).

Services derived from the usage of energy consumption are multiple. Energy reports, profiles and summaries can refer to specific devices, lines or the whole dwelling, covering daily, weekly, monthly, seasonal or even yearly scopes. Here we introduce two services based on habit profiles, whereas habit profiles for user reports are dealt in the next section (Section 3.9).

– *Customised Billing*
A well-known problem of energy generation and distribution is the low-balance curve that energy demands present. For example, electricity night tariffs exist to empower people to displace consumption to valley hours in order to smooth the global demand curve. Commentaries and references related to the low flexibility of the energy market and the necessity of proactive relationships between energy suppliers and consumers can be seen in Section 3.6.2.

A customised, tailored billing can be carried out using profiles. For example, the energy behaviour of a flat, dwelling or building can be represented with two profiles: a daily profile (that characterises the normal hourly evolution of demand) and a weekly/monthly profile (which shows the different consumption rates with a daily scope). Such profiles offer valuable information for energy suppliers and retailers, who can improve their predictions, match demand profiles with expected prices and even give tailored advice to users based on forecasted future scenarios. Therefore, by means of profiles, energy providers can offer a fairer system (hourly sized and customised) that rewards sustainable behaviours and even develops preempted strategies based on feedback to users. The process is illustrated in Figure 3.12.

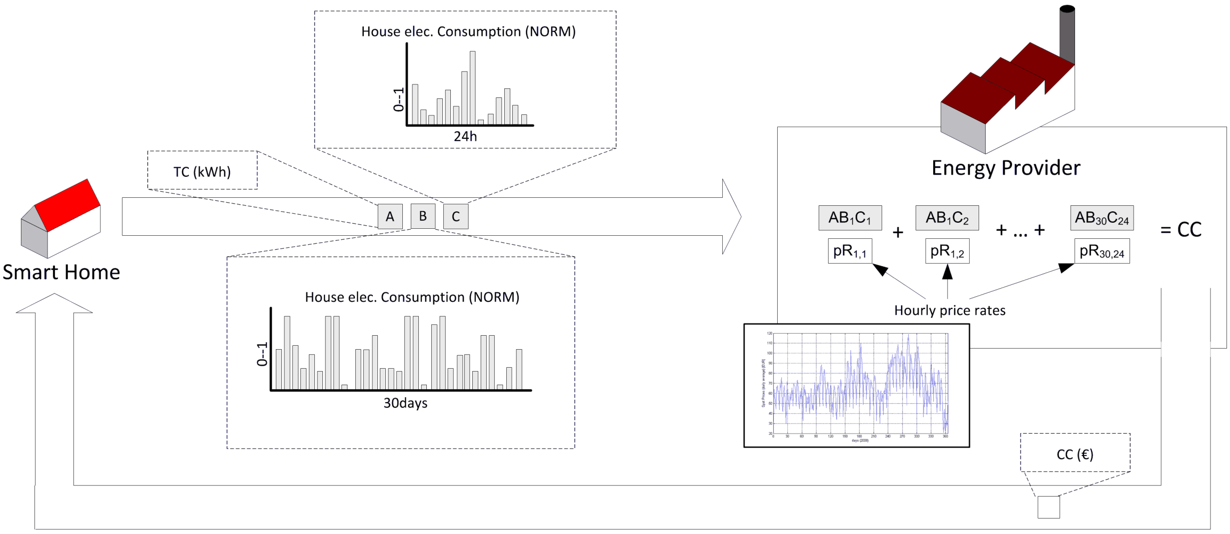


Figure 3.12: Example of customised billing process.

– *Control based on Load Definition*
In a smart home where electricity loads are identified according to a *load definition*, profile-based strategies can support the correct management of DSM strategies. Load definition involves *classification* and *description*. A possible and basic load classification (labels are non-exclusive) is given in Table 3.4, which can be more extensive depending on the control applications and requirements (e.g., if supporting control for distribution generation systems or smart grids is considered, new load types might be necessary). The description entails load types but is also open to other parameters (if applicable) like status (on/off), nominal power, supply time, etc.

Profiles are necessary to perform the *control of shiftable loads*. There are some electricity loads that users do not need to supply immediately and can be shifted to periods when electricity is cheaper and less demanding. In those cases, smart controllers find the best moments to supply shiftable loads according to context data that include profiles (e.g., load definitions, predicted demands, predicted/available electricity prices, strategies, additional constraints). The aim of shifting DSM strategies is to flatten the consumption curve or to counteract negative trends.

3.8.7 User Reports

Home reports are devised to inform inhabitants concerning the house performance, as well as to give information to improve user control upon the system and the dwelling.

To elaborate reports the total amount of home profiles are at the system's disposal, hence it is difficult to stress some profiles over others as it strongly depends on the purpose of the report

Type	Definition	Devices	Control action
Stand-by	Devices that show consumption in standby mode and remain connected when people are absent.	Cooker, oven, white goods, office equipment, entertainment (TV, DVD, etc.).	Open/close electrical supply depending on occupancy (or in sleeping periods).
Permanent	Devices that are continuously switched on with a quite stable energy consumption.	Fridge, freezer.	No control (green devices or specific solutions).
Shiftable (or movable, deferrable)	Loads that can be shifted in time.	Washing machine, dishwasher, storage heater and water heater, pumps, etc.	Move the load activation to a best moment for the energy system.
Priority (or arbitrary, mandatory)	Normal loads that must be supplied when it is required for their normal running.	Lighting, communication devices, cooker, oven, dryer, white goods, office equipment, entertainment electronics, battery chargers, ventilation, cooling devices, etc.	No control (green devices or specific solutions).

Table 3.4: Load definition table.

service (the options are multiple). For instance, energy reports can simply show the energy consumption evolution of the house (only the *dwelling consumption* profile is needed), but they can also elaborate more accurate reports relating to consumption and occupancy (*occupancy* profiles would be necessary as well). In any case, the next services are presented as guide examples:

– *Check Energy Behaviours*
With *Check Energy Behaviours* applications it is possible to check how sustainable the users' behaviour is or how energy is being used at home. Inhabitants obtain feedback to improve their energy behaviour, learn how to optimise the use of appliances, obtain economical benefits by energy savings and even detect failures or counterproductive habits.

The applications utilise habit profiles and also benchmarks obtained from remote repositories to develop reports with energy evaluations, useful information and advice for users. To give some examples, some of the possible assessments and advice can be based on the following appraisals:

- Evolution of the whole energy consumption.
- Energy consumption compared with benchmarks.
- Index of electricity flattening.
- Energy consumption related to instantaneous energy prices (best/worst periods for the consumption).
- Separated analysis of energy consumption (devices, lines, energy sources).
- Detection of failures, lines and devices that are consuming too much.
- Evaluation of the relationships: occupancy-consumption, lighting-presence, sunlight-lighting, thermal comfort-energy demand, etc.

– *Smart Home Customisation*
This application shows common points with the already seen self-checking applications. Now,

users occasional run this service in any of the home UIs to supervise the overall operation, force self-checking routines, review warnings and advice and maximize the customisation. Since users voluntarily start the process, the system here offers the possibility to display low-level warnings that are usually hidden in order to not disturb inhabitants in their daily life.

A possible structure for the menu tree of the *Smart Home Customisation* is shown in Figure 3.13.

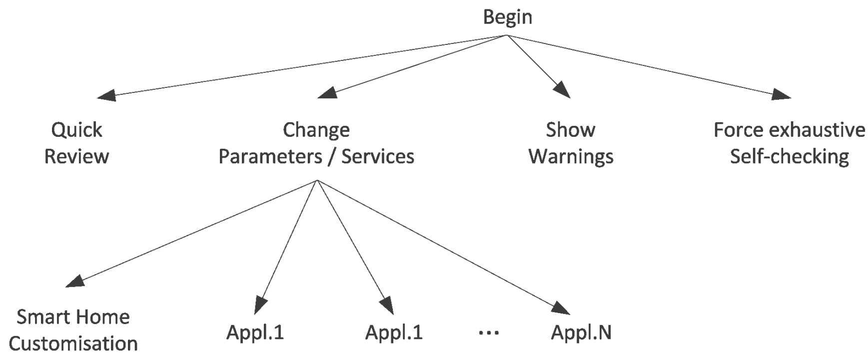


Figure 3.13: Example of *Smart Home Customisation* menu tree

In the example sketched in Figure 3.13, the *Quick Review* is designed to show a quick general checking, reviewing the most general aspects submitted to users' preferences and showing the main detected problems. With *Change Parameters/Services* users can go directly to the application they have in mind to connect/disconnect or modify specific parameters. *Show Warnings* would offer recorded warnings, errors, problems, advice and recommendations arranged by appearance. The *Force exhaustive Self-checking* option would carry out complete checking actions and list the results by priority order.

3.9 System Self-checking Capabilities

All the self-checking applications introduced in previous points have a similar operation but with variations within input variables and defined output actions. They are embedded into Shadow Processes, operate behind normal services and are intended to find out whether performances fit users' expectations. By means of these services the system is able to change strategies, routines, as well as to ask for user guidance in case it faces conflictive situations that can not be solved without supervision.

Therefore, all shadow processes are ruled by general purpose agents or modules, e. g. user preferences, conflict resolution or global goal agents. These general purpose referees establish some parameters that are common to all shadow processes, i. e. all the self-checking services. For example, Table 3.5 shows a simple set of rules to manage the behaviour of self-checking applications. In the table, *setting levels of unsupervised actuation* sets the freedom of the system to make adaptive decisions automatically, and *setting levels of tolerances* stands for the sensitivity of the system dealing with drifts and signs of disagreement and bad performances.

Thanks to the introduced general parametrisation, shadow processes are able to deal properly (i. e. as expected by users) with bad performances and conflicts, reaching a smooth and friendly adaptation to users.

Param.	Operation	Possible values/modes
O1	Prioritising comfort or energy savings	'energy savings', 'balanced' and 'comfort'
O2	Setting levels of system feedback	'none', 'only passive feedback', 'important warnings' or 'advisor mode'
O3	Setting levels of unsupervised actuation	'low', 'normal' and 'high'
O4	Setting levels of tolerances	'low', 'normal' and 'high'

Table 3.5: Example of parameters common in shadow processes.

Detection of Undesirable Performances

As time passes, it is expected that the system gradually adapts to the users' subjectiveness and comfort estimations. Some possible reasons that can make this adaptive process difficult are listed as follows:

- The habit abstraction algorithm is not discovering suitable patterns.
- The dwelling/facility is being used by many different people without following stable habit patterns.
- Users have a chaotic lifestyle.
- The application is badly designed for the specific case or is not aware of determinant factors.
- There have been a change of users (maybe temporal, maybe permanent).
- There is a system or device failure/breakdown/malfunctioning.

Considering unsupervised context-awareness capabilities, the system is able to detect such situations, i. e. user discomfort or undesirable performances, by means of two indicators:

- *Habit profiles are lax or unreliable.* The collected habits do not present enough of a degree of stability required to achieve a good performance (the associated value of reliability is under a prefixed threshold). In other words, the use related to the habit profile tends to be chaotic in time, i. e. unpredictable. Explanations concerning how to evaluate the reliability of habit profiles are widely explained in Section 5.4, and shown by examples and tests in Section 6.4.
- *High number of user re-adjustments.* The system detects an abnormal number of commands by users after the normal automatic adjustment is carried out by specific service controllers. A good adaptation is bound to be reached as long as the number of user re-adjustments (within a fixed time after the programmed adjustment) decreases as well as the quotient between these two values. Otherwise, the system can infer that the service is not fitting users' expectations.

Since the issue concerning habit reliability is dealt with in posterior chapters, we will now concentrate on the cases of user re-adjustment. To clarify the idea that underlies this kind of indicator, we sketch a simple, illustrative example. A system switches setpoint temperatures from 21° to 23° according to reasonings in a control pre-phase and comfort profile. However, a few minutes later, users manually switch to 21° again. This contradictory, undesired situation could mean that users are not satisfied with the last automated change. If everything runs properly, the number of manual adjustments must decrease or keep in certain low limits as time passes by. Otherwise, the phenomenon represents an unsatisfactory operation that should be corrected.

The described situation can occur similarly in any profile-based application. For example, let us check it in the case of the *Self-checking: Lighting Management* service (Figure 3.10). We know that in profile-based lighting a general command modifies status and levels of all the lamps defined in the same group, whereas each specific command only affects its respective light. The values of the *general lighting command profile* and the *lighting user re-adjustment profile* are established as shown in Figure 3.14, i.e. the system considers a *re-adjustment* whenever there is a change of level or status caused by any specific command *after* a rising edge in the general command (the length of "*after*" is fixed by a *threshold*).

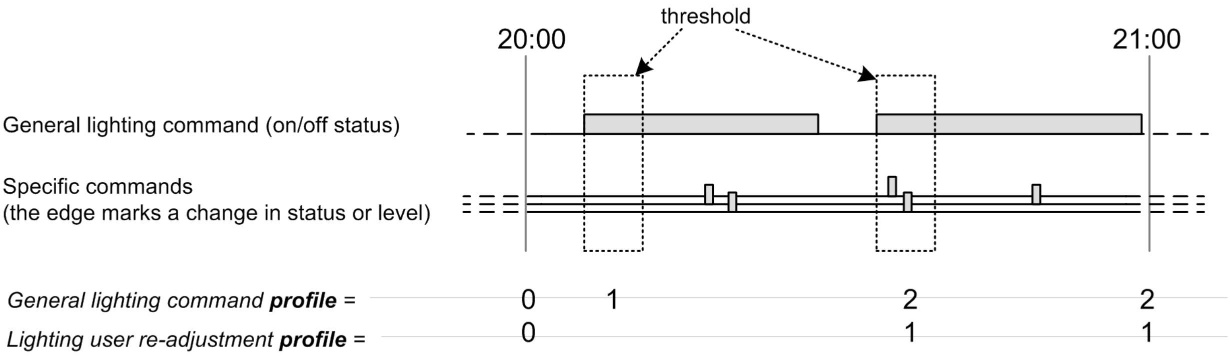


Figure 3.14: Detection of a high number of user re-adjustments using *general lighting command profiles* and *lighting user re-adjustment profiles*

Now, we can imagine hypothetical users with the profile values shown in Figure 3.15. Here, we show monthly profiles (summaries) formed by means of daily profiles in order to have a scope big enough to evaluate situations where self-checking assessments are justified.

It is worth discussing the example case by case:

1. 'Case 1' is the expected for a good performance of the lighting application. The number of re-adjustments is kept under certain level or is prone to gradually decrease as time passes by and the system learns users' habits and preferences.

2. 'Case 2' shows a situation where the number or re-adjustments is continuously very high. In this case, the system must change the control strategy in order to avoid unsatisfactory performances. Moreover, compared with 'case 1', the number of activations of the general command is quite low. This can be indicating that inhabitants do not use the general command because they are not pleased with the service; in such a case, lighting status profiles are checked together to know the level of usage given to the intended lights and carry out corrective routines in case it is necessary.

3. 'Case 3' shows an erratic situation where the quotient between the two profile values alternates between high and low. It obviously points to the necessity of some corrective actions. Unlike in 'case 2', a look on the daily profiles can reveal that disagreements are only happening in a specific time or period slot (maybe specific days of the week).

CASE 1

	Day1	Day2	Day3	Day4	Day5	Day6	Day7	Day8	Day9	Day10
Gen. ligh. com. ***profile*** (a) =	7	10	8	6	2	7	10	10	9	7
Ligh. user re-adj. ***profile*** (b) =	4	5	3	2	0	1	2	1	0	0
b/a =	0.6	0.5	0.4	0.3	0	0.1	0.2	0.1	0	0

CASE 2

	Day1	Day2	Day3	Day4	Day5	Day6	Day7	Day8	Day9	Day10
Gen. ligh. com. ***profile*** (a) =	4	5	3	2	3	4	5	4	3	3
Ligh. user re-adj. ***profile*** (b) =	6	5	2	3	3	3	4	4	6	3
b/a =	1.5	1	0.7	1.5	1	0.8	0.8	1	2	1

CASE 3

	Day1	Day2	Day3	Day4	Day5	Day6	Day7	Day8	Day9	Day10
Gen. ligh. com. ***profile*** (a) =	7	11	8	2	0	10	5	8	8	8
Ligh. user re-adj. ***profile*** (b) =	4	6	0	2	0	6	5	0	5	5
b/a =	0.6	0.5	0	1	0	0.6	1	0	0.6	0.6

Figure 3.15: Three possible situations of self-checking assessments for the lighting services case

Corrective, Adaptive Actions

The solution to face undesirable cases varies depending on the specific shadow process or self-checking service. The system can relax the parametrisation of the habit discovery tool or the response of the profile-based controllers, moving towards either extreme or middle values. On the other hand, the system can disconnect the service, or ask users to obtain guidance in order to know how to behave from now on (e.g., disconnection of s specific controller based on profiles, to load a pre-fixed scenario, apply bias on the profile interpretation made by controllers, change priorities, change programming, etc.). A linked smart home can even make the most of other linked smart homes and receive feedback by means of remote servers.

Coming back to the *Self-checking: Lighting Management* service example, a possible *self-checking controller* could execute one or more of the following actions as a response to low levels of habit reliability or high levels of user re-adjustment:

1. *Self-adjustment.* a) Keep the same configuration. b) Apply smooth filters to the available profiles so as to receive a more relaxed profile. c) Replace one or more profiles with fixed lighting levels. d) Load a pre-fixed scenario instead of using profiles while habit-indicators are out of desired levels.

2. *Active feedback.* a) Issue a conflict warning (light indicator). b) Conflict warning (sound indicator).

3. *Pasive feedback.* a) Report poor service performance, causes and made decisions. b) Request action to be conducted. c) Request actions to be conducted in future conflicts.

The set of actions finally executed will depend on the general parametrisation of shadow processes. For instance, given a system that operates using Table 3.5, in a case where O4 is set to 'low', the demanded levels of reliability find the current occupancy profile as 'unreliable'. If O1 is 'energy savings', O2 is 'only passive feedback' and O3 is 'high', the shadow block (after an analysis focused on the involved profiles) would execute by default: 1.d) and 3.a), and would remain the same until the occupancy profile is reliable again or users change the configuration. However, in the same case in a different smart home with O3 fixed to 'low' and O2 to 'important warnings', the shadow block would execute by default: 1.a) and 3.a). If the conflict persists (the forbearance of the system is stated by O4), it would add 2.a), 3.b) and 3.c) to the actions by default for the next time.

3.10 Case Example: A Smart Medium Standard Apartment

This section presents a small medium standard apartment for singles or couples where a profile-based smart home system has been implemented. The apartment dimensions are between 30m^2 and 40m^2, plus a balcony of about 10m^2 (Figure 3.16).

In the figure, diverse house elements are emphasised and associated to profiles directly related to use or behaviour. Note that we are not marking inputs or outputs of the HAS, and we are obviating habitual non-profile-based applications or services. It is not necessary for each automated point to be represented by a profile, in the same way that it is not necessary for every possible automated point to be under the control of the HAS.

The applied profile-based services are as follows:

- General Command for Lighting Groups.
  It is applied for the lights of the dining room, three common on/off lights – L1, L2 & L3 – and a dimmable light – L4. The required profiles represent *status* of L1, L2, L3 and L4, and *level* of the dimmable light L4. Note that, although the application needs to be triggered by a general command, the *general lighting command* profile is not required to perform any action within this service.

- General Command for Blinds Groups.
  The running is analogous to the previous one, but here the system controls blinds B1, B2, B3 & B4, collecting *level* profiles.

- Thermal Comfort Support & Setpoint Temperature Control.
  In the example, these services are implemented without considering relative humidity. A smart lock in the main door (CG) is responsible for generating *dwelling occupancy* profiles. *Comfort temperature* profiles are built by a wall mounted command together with menus displayed in UIs (both interfaces are represented in the figure by the "User Interface & Commands" block).
  In the case example, the Thermal Comfort Support application only manages blinds in unoccupied periods and performs advice in occupied periods considering blinds, contact sensors in windows and the HVAC system status. The Setpoint Temperature Control automatically adjusts setpoint temperatures based on the experience collected in profiles.

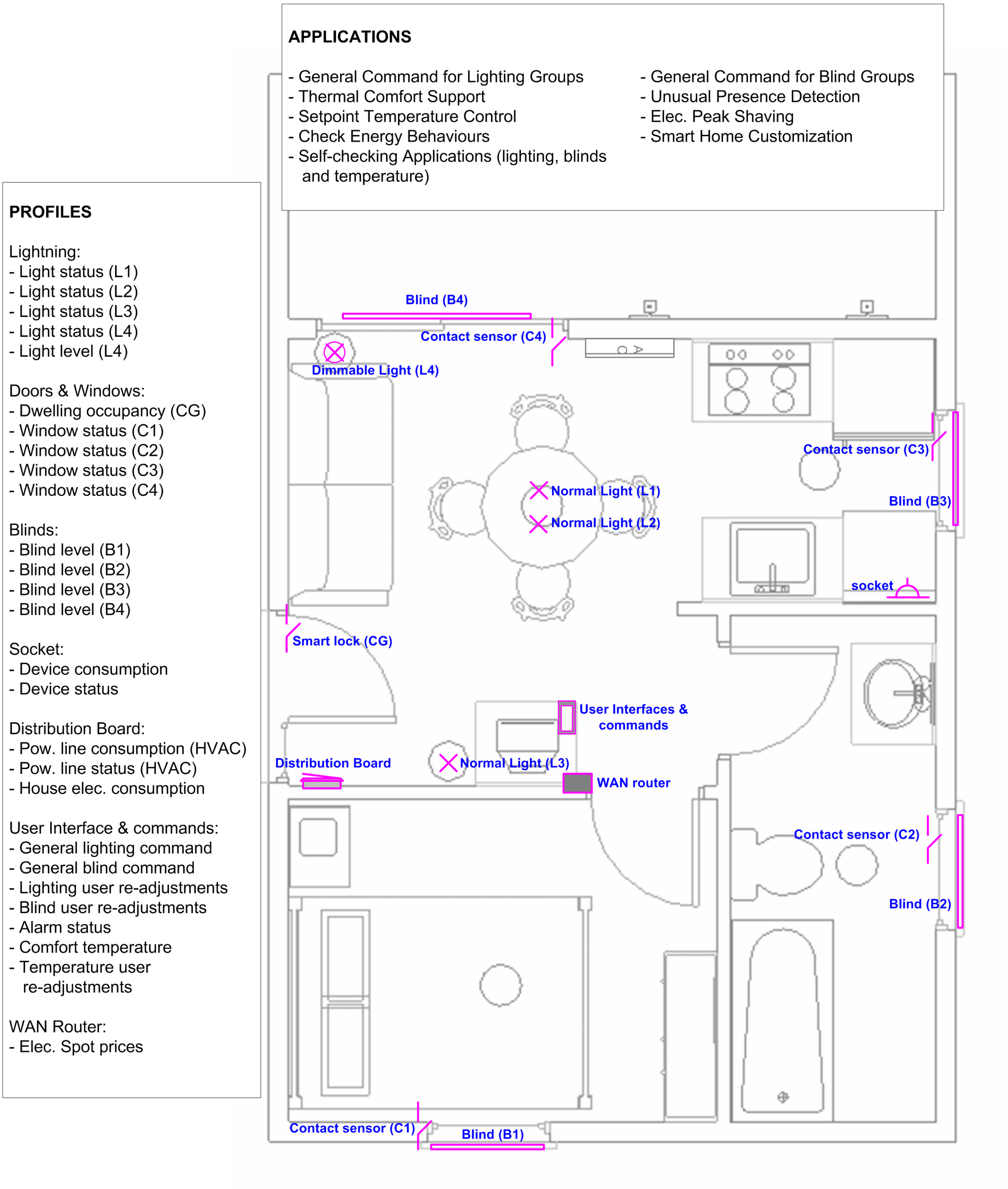


Figure 3.16: Profiles and applications included in smart medium standard apartment case example.

- Unusual Presence Detection.
  Messages concerning unusual presence detection are elaborated considering *dwelling occupancy* profiles as well as monitoring changes in C1, C2, C3 or C4 during unoccupied periods.

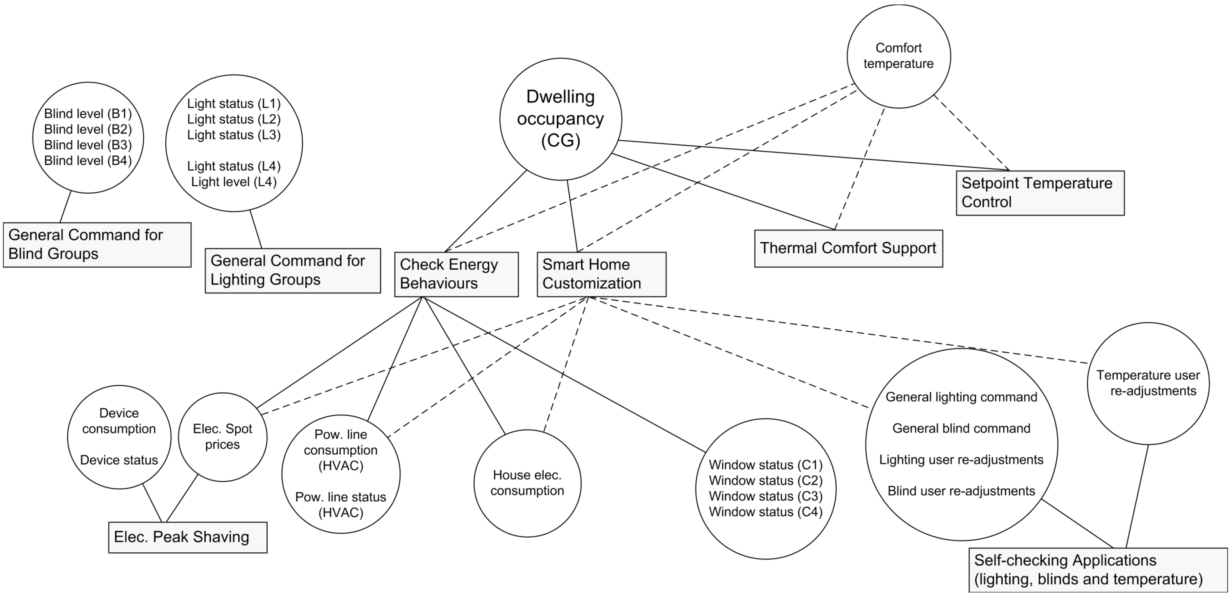


Figure 3.17: Possible relationship between profiles (circles) and applications (light grey boxes) in the smart apartment example.

- Elec. Peak Shaving.
  The service optimises the management of the washing-machine connected to the socket shown in the figure. Here, the used profiles are *device consumption*, *device status*, *dwelling elec. consumption* and *elec. spot prices*, the last one obtained by remote repositories. The service deploys profiles to predict the expected consumption for the following day, relates them with the electricity prices and, therefore, places the connection of the washing-machine in the optimum time slot.

- Check Energy behaviours.
  The included reports are as follows:

  - Evolution of the house overall consumption (required profile: *dwelling elec. consumption*).
  - Evolution of real electricity costs (required profiles: *dwelling elec. consumption* and *elec. spot prices*).
  - Evaluation of HVAC use (required profiles: *comfort temperature*, *pow. line consumption – HVAC*, *pow. line status – HVAC*, *dwelling occupancy* and *windows status – C1, C2, C3, C4*.

Figure 3.17 draws the relationships between applications and profiles. Note that profiles are shared resources, so it is common that one application deploys diverse profiles and one profile is utilised by different applications.

4 Application Example: AQTC Control Based on Usage Habits

A vertical approach to profile-based control for smart homes is developed throughout this chapter, selecting the AQTC field as focus. Designs of structures and applications are widely explained, depicting smart agents and how profiles are generated, transmitted and deployed. The last part of the chapter presents tests and comparisons in a simulated scenario in order to assess the validity of some of the profile-based control solutions ahead other classic strategies for the same scenario.

4.1 A Top-Control Phase

In Section 3.5, we introduced profile-base applications forming a pre-control phase previous to the end controllers. The set of applications depicted as follows – named *AQTC Control Applications* – follow this schema, they do not try to replace usual HVAC controllers, but to integrate AQTC in the context of holistic smart home management by means of the referred top-control or pre-control stage. In other words, these applications are embedded into an intermediate layer placed between the users, the home environment and the equipment (the description of profile-based design is explained in Chapter 3). Thus, every application is integrated into a conscious coexistence with other applications, users' habits and the usual use of home devices and appliances.

The added pre-control phase is intended to operate regardless of the installed equipment and is flexible dealing with the variable features of every available system. Figure 4.1 shows a *System-Equipment Interface* agent/module in charge of settling the relationship between the implemented applications and the specific end equipment (and their controllers). It receives support from the *Conflict Resolution* agent/module, which solves possible conflicts among applications intended for the same or very close functionalities, and it is even aware of simultaneities and overlapping also with further or non-directly related applications. Concerning the selected scenario, it is worth noticing that HVAC equipment obviously has an essential role in the optimisation of AQTC, but also other equipment and home elements contributing to the final performance.

In keeping with such conceptual scheme, some of the applications depicted in Chapter 3 are developed in this chapter specifically for the AQTC management. Following on from descriptions of the system, examples and tests are also described in order to check the validity of the proposals.

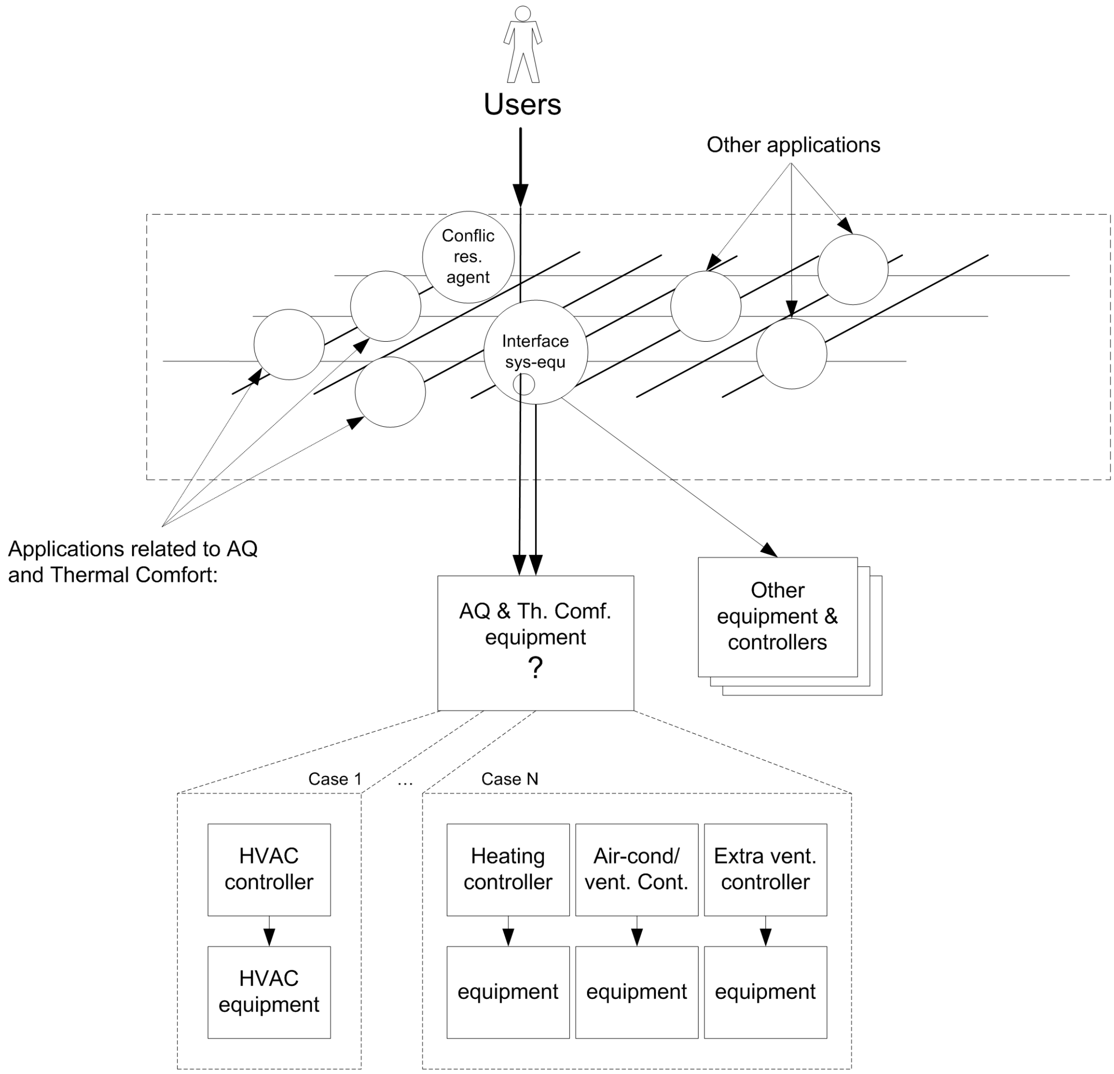


Figure 4.1: Holistic management for AQTC. The control approach is devised regardless of the used equipment and is able to adapt to it.

4.2 AQTC Backgrounds

Air quality indices stand for assessments of the atmospheric pollution, i. e. a measure of *how healthy the air is* based on well-defined air pollutant levels that state acceptable ranges to develop people's activities and daily life. Additionally, the definition established by ASHRAE [Ash01] also considers people's feelings of *satisfaction* as a subjective component in the evaluation of air quality. 'Thermal comfort' is defined as a subjective evaluation from users, and it changes depending on metabolic rate, clothing insulation, air temperature, radiant temperature, air speed and humidity [Ash04]. Commonly forgotten, thermal comfort is not only a matter of comfort or pleasure, but also a matter of health [Org07]. In fact it does not only affect physical health, but merely low levels of both air quality and thermal comfort have been detected as determinant factors for people's decrease of productivity, mood disorders as well as psychological well-being, e. g. [Par93; Lun96; LB99].

The previous paragraph highlights two aspects: on one hand, the importance of an appropriate management of AQTC in houses and buildings due to health reasons (besides comfort). On the other hand, the inevitable influence of users' subjectivity has to be considered. As far as the automation of habitable spaces is concerned, these two factors draw a scenario in which stating well-defined boundaries differentiating between harmful and hazardous ranges is relatively easy, but identifying good, acceptable or unpleasant conditions is a more difficult, case-dependent task.

From the technological perspective, the achievement of AQTC requirements is usually undertaken by HVAC systems. Within this field, for about the last 40 years HVAC practitioners have mostly used the Fanger's PMV (Predicted Mean Vote) model [Fan70] to represent people's opinion concerning thermal comfort. Therefore, Fanger's model and specific systems, improvements and evolutions based on this model, e. g. [LD05], remain in the basis of controllers, system designs and building engineering calculations.

Despite its widespread usage, Fanger's PMV model has been criticised several times due to the fact that its predictions are sometimes biased and do not represent people's requirements satisfactorily in the variable conditions of real buildings and spaces [HN02; San03]. In [DD04], the nature of the critic is even deeper when the author asks "why engineers, notably Fanger, have become to dominate a research topic that falls so clearly within the scope of psychology?". The offered answer points to a certain lack of interdisciplinarity in the habitual approaches.

In any case, the variability in people's thermal comfort appraisals is so large that trusting in models or standards can lead to unpleasant performances. In addition to this, such models are completely unnecessary where users are provided with individual control on the thermal climate in a way that they can adjust their preferences [Fan01]. In the same line of thought, in [VH08] the author concludes stating that "thermal comfort for all can only be achieved when occupants have effective control over their own thermal environment".

The methodology described here is developed keeping these last ideas in mind, so it moves away from Fanger's model and automates the system response based on the collection of previous users' adjustments and the abstraction of habits. In addition, control tasks are primarily faced from a top-down perspective that considers all the applications and elements of the home environment at the same time. This focus avoids some drawbacks related to bottom-up approaches: these approaches suffer not being able to see the implications that emerge from the integration into bigger structures. Indeed, HVAC engineers usually trust standards and recommendations in their designs without necessarily considering requirements and the realities faced in a building with regard to other building appliances. Hence, the HVAC installation is often experienced to be quite an isolated part, resulting in a restriction of smart home control, and posing an impediment for an efficient global performance. Synergies and feedbacks with other applications are easily overseen, and therefore installations are prone to be oversized, conflicts tend to happen more frequently and in consequence unnecessary extra economical and energy costs appear.

The developed applications to counteract these effects perform the fine adjustment of AQTC variables by means of the cooperative usage of habit profiles, while healthy boundaries are fixed by global standards (cf. Fig. 4.2). Habit profiles store time-related user adjustments, preferences and decisions in a way that to abstract what users *found comfortable* in the past is possible. Thereby the system *learns* what thermal comfort exactly means for its own users. A profile KB (Knowledge Base) is managed in order to discover implicit routines based on the inhabitants' habits, dynamically differentiating between seasons, days of the week, etc. Thus, the *personal factors* considered in the definition of thermal comfort can also be partially covered by profiles, as activities and clothing of people usually recur in time and follow daily, weekly and seasonal

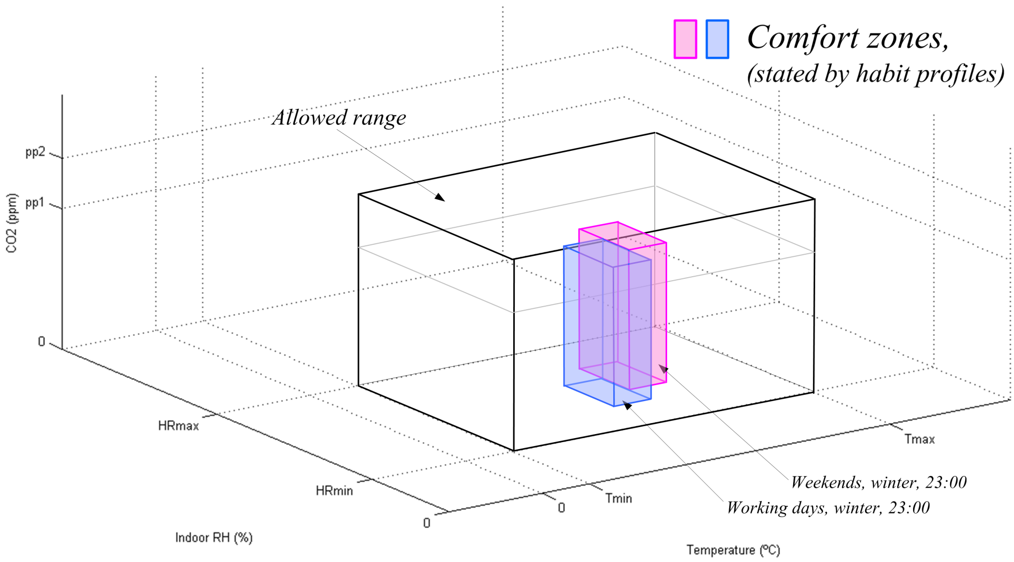


Figure 4.2: In the figure, AQTC setpoints are established by assessments of indoor temperature, indoor relative humidity and indoor CO_2 concentration levels. The allowed range is fixed by competent standards and directives, whereas the comfort zones are adjusted using time-related habit profiles.

periods. In short, the overall smart home control supports the HVAC system with a better knowledge of the context, contains the coordinated management of all home appliances and is able to recognise the environment status.

Such an approach is coherent with findings in HVAC application development and research. They usually stress the importance of systems with high context awareness, as well as the benefits obtained by the smart actuation on other house and building elements beyond the classic HVAC equipment. To give some examples, in [ECPC11] authors consider *occupancy* for the HVAC management, and assure that it is possible to achieve 42% annual energy savings whilst still maintaining ASHRAE standards. Also in [Bra99], demand-controlled ventilation strategies imply as much as 20% savings of electrical energy needed for cooling. Further, in [HP10] authors point out that strategies making use of natural ventilation not only seem to be more appreciated by occupants, but can also be very effective, providing acceptable indoor AQTC in addition to energy savings. Similarly, some works have developed models to face the control of operable windows depending on indoor and outdoor parameters, e. g. [SN09].

HVAC systems that consider occupancy and also control the status of windows lead us to consider sophisticated devices. For instance, in facilities where natural and forced ventilation are to be combined, HVAC designers have to face scenarios complex to solve and rather unpleasant [MP08]. Moving part of the system intelligence to the overall smart home management – at least the part related to context awareness and in connection with other appliances – seems to provide additional benefits. For example, leaving the concern about occupancy and the management of windows' status in the hands of the global smart home control results in an economical performance, where these elements can also be utilised for the optimisation of other applications (lighting, security and safety, energy consumption, DHW, etc.). Moreover, it is open and adaptable to individual HVAC equipment and levels of complexity. Finally, the top-level view has a better perspective to successfully manage conflicts and overlaps.

4.3 Design of Application Control Agents/Modules

The considered AQTC Control Applications are the following:

- *Setpoint Temperature Control.*
  It evaluates the thermal requirements of the space and fixes the setpoint temperature based on instantaneous occupancy, as well as on habit profiles concerning occupancy and comfort temperatures.

- *Setpoint Relative Humidity Control.*
  Analogous to the Setpoint Temperature Control case but for indoor relative humidity.

- *Thermal Comfort Support.*
  During unoccupied periods, this service manages blind devices, air dumpers and natural ventilation elements in order to keep the space in the best conditions with the lowest energy consumption possible. In occupied periods, it executes its actions in the form of users' advice and recommendations.

- *Ventilation Demand.*
  Based on occupancy and air quality assessments, the smart home enables or disables the ventilation system.

Implementation in a Multi-Agent Framework

These four applications coexist with other home applications within the smart home domain. They are formed by modules (*agents*) according to the identification of well-delimited tasks and purposes (MASs are referred in Section 3.1, whereas the foundations of the used MAS is explained in [RKK10]). Applications do not depend on one another, but share agents and profiles as long as they also have their representative processes. Fig. 4.3 offers an overview of the defined applications together, implemented in a framework of agents. It is worth considering that some agents are available for other applications and do not solely exist for the exclusive deployment of HVAC-oriented services. In any case, the agents that realise parts of the introduced applications are as follows:

- Control Agents.
  Setpoint Temperature Control Agent, *Setpoint RH Control Agent*, *Thermal Comfort Support Agent* and *Ventilation Demand Agent* are in charge of making decisions in their respective and specific control fields and therefore rules, routines and algorithms for direct control are implemented in these agents. To accurately do that, they receive information about the context from the rest of the agents. Their decisions are expressed as changes in defined output variables that they send to the *Interface System-Equipment Agent*, e. g. setpoint values for end controllers or message codes for informative services.
  The *Conflict Resolution Agent* is a special type of general purpose control agent that looks

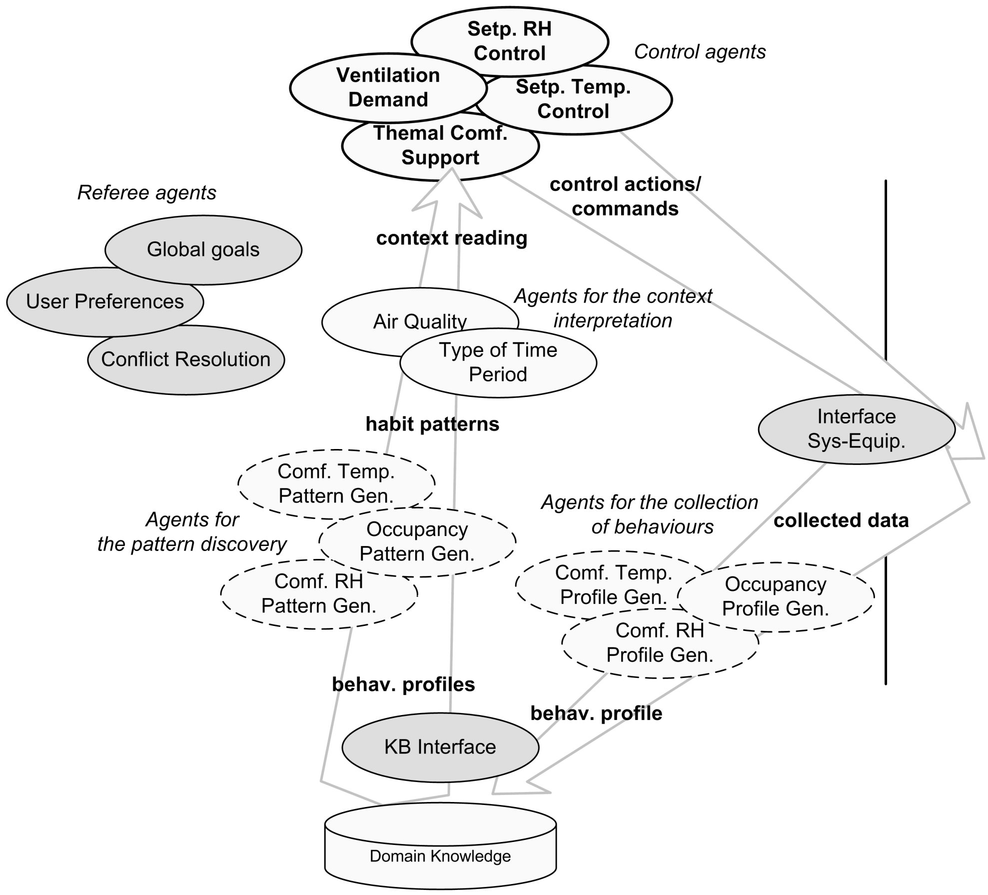


Figure 4.3: Schematic diagram of relationships and connections among agents in AQTC Applications. Bold agents are representatives for the specific decision making of the application tasks; ovals with dark gray background are general purpose agents; the ones embraced by dashed lines stand for agents that manage profiling processes; the rest are agents for data interpretation and transformation.

for the harmonious global communication and collaboration among all control agents. It acts as a referee, establishing priorities for agents and end devices. Further, it detects conflicts and incoherences in simultaneous services with the help of information in the KB, suggesting final solutions to avoid unpleasant overall performances.

- User Preferences Agent.
  The *User Preferences Agent* keeps track of parameters and modes directly established by users, for example the activation or deactivation of available services, the selection of whether to give priority to energy savings or comfort, or the adjustment of information provided by the system to the users. This agent usually acts upon control agents, adjusting parameters considered as constants or switching alternative algorithms implemented inside them.

- Global Goals Agent.
  The *Global Goals Agent* operates in combination with the User Preferences Agent to establish a coherent global operation. It analyses the context information and the user preferences in the KB together, checking for inconsistencies and performing actions and strategies in

keeping with the defined global goals (mainly energy efficiency and comfort) but also respecting user desires and adjustments.

- Context Inference Agent.
  Comfort Temperature PFG Agent, *Comfort RH PFG Agent*, *Occupancy PFG Agent*, *Comfort Temperature PTG Agent*, *Comfort RH PTG Agent*, *Occupancy PTG Agent*, *Air Quality Agent* and *Type of Time Period Agent* are collected in this group. They do not make decisions that directly affect the end equipment but operate with data to transform or abstract information about the context. They can be considered as *input or intermediate agents* in comparison with control agents, which would be *output agents*.

- KB Interface Agent.
  The *KB Interface Agent* connects the agents with the state of the domain represented in the KB. This agent is in charge of storing static data (preferences, profiles, patterns), but also manages instantaneous and short-term data (equipment states, building conditions) in a dynamic knowledge store, keeping an active coordination link with the Interface System-Equipment Agent.

- Interface System-Equipment Agent.
  Finally, the *Interface System-Equipment Agent* builds a bridge between the available equipment and the rest of agents. In this component, features of field devices, sensors, controllers and mechanisms are carefully identified, as well as *how to deal with input data* and *how to transmit output data* are accurately defined. In short, this agent can be considered as an *advanced gateway* that encompasses multiple protocols and architectures. Furthermore, it has capabilities to warn the Conflict Resolution Agent about detected conflicts and overlaps between control agents.

In the prototype, for sharing of information between agents, a dedicated KB is considered. The advantage of such a sophisticated knowledge representation instead of a classical database scheme is that it holds information about the dependencies of different profiles in a logically structured fashion. This way, certain inferences can already be drawn in the KB itself, leading to a more comprehensive model of the world for agents' operation. In the example case, an ontology in the Web Ontology Language (OWL)[1] is considered. The logical foundation of such a KB permits the inference of new information already in the knowledge store according to a set of specified rules. The most expressive and yet decidable form of OWL is based on Description Logics (DL) and as such allows the representation of complex dependencies with the help of logical axioms. An ontology as knowledge representation for a smart home system can therefore be used to automatically classify profiles, taking away complexity from the system operating on it. Hereby, for a time representation in an OWL ontology, the W3C OWL-time recommendation [Tim06] can be used. This way a standardised time-division facilitates the classification of profiles. For a more thorough discussion of the OWL KB in the context of the smart home system and its individual parts, the interested reader is referred to [KRK12].

The following sections explain the main tasks carried out by agents and the intelligent mechanisms that control and organise the operation of the AQTC Control Applications.

[1] `http://www.w3.org/TR/owl2-overview/`

4.3.1 Time Period

Occupancy (by extension also occupancy profiles) are common elements in the selected applications, as well as to a large extent of profile-based services. Indeed, as we mentioned in Section 2.1, occupancy is one of the most required variables *de facto* for home control purposes.

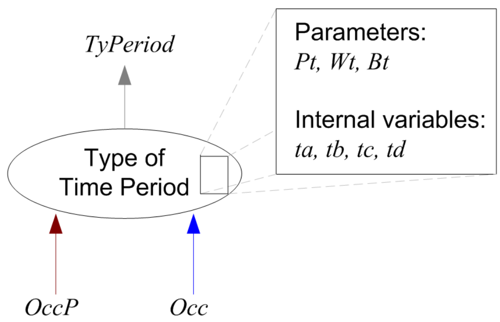


Figure 4.4: Inputs, outputs and parameters of the Type of Time Period Module.

Based on instantaneous occupancy and occupancy profiles, we model an agent (Type of Time Period Agent) that defines *time periods types*, elements that play an important role in the execution of different services. The method introduced in this thesis states: *Long Absence, Short Absence*[2], *Preparation and Presence*. Table 4.1 shows these common periods and the relationship with the defined applications. In the table, different states appear: *Active (Act.), Inactive (Inact.), Safety* and *Setback/Setforward (Set.)*; the meaning of each status is explained above with connection to the respective control agent. Moreover, in the developed approach, design periods coincidently have the same length and trigger moments for all applications, however this is not mandatory: in an alternative case, every single application could have its own Type of Time Period Agent.

	Long Abs.	Preparation	Pres.	Short Abs.
Setpoint Temperature Control	Safety	Act.	Act.	Set.
Setpoint Relative Humidity Control	Safety	Act.	Act.	Set.
Thermal Comfort Support	Idle	Act.	Advisor	Act.
Ventilation Demand	Inact.	Safety	Act.	Safety

Table 4.1: Time periods and status based on occupancy profiles.

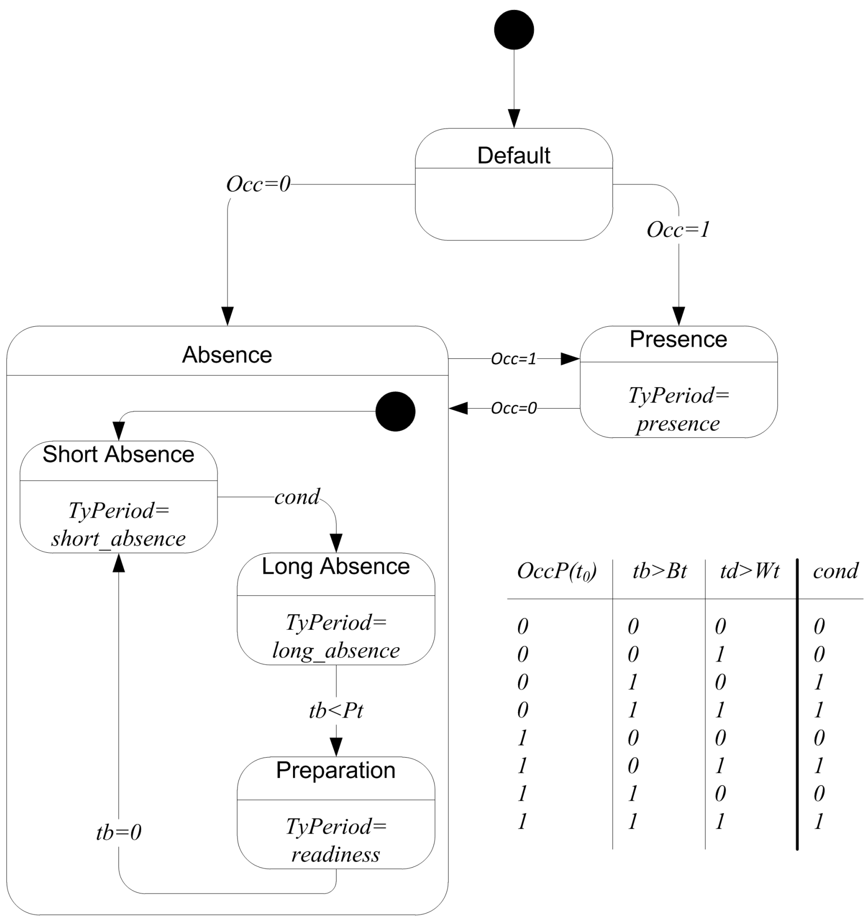
Table 4.2 shows the required input, output, internal variables and parameters for the Type of Time Period Agent. The parameter values of the building are represented in the global KB and are integrated into the system in the set-up calibration phase by domain experts, knowledge engineers or an automated transformation mechanism in order to represent the geometric characteristics of the building, the building materials, the resulting inertia of the system, or specific HVAC equipment. In any case, they are open to modifications by users, by means of the User Preferences Agent and also by the Global Goals Agent.

The value taken by the *TyPeriod* variable depends on instantaneous occupancy, the occupancy profile in use and also on their own past values. It is displayed in the state diagram in Fig. 4.5. The agent's algorithm can be explained as follows:

[2] Differentiating between short and long absences is suitable for HVAC control, as experts agree on the benefit of switching to setback/setforward temperatures during short absences, e. g. [IH85], [MH11].

Object	Type	Description
Occ	input	Instantaneous occupancy
$OccP$	input	Current occupancy pattern
$TyPeriod$	output	Type of time period: Long Absence, Short Absence, Preparation, Presence
ta	int. var.	Time after the last rising edge in the occupancy profile
tb	int. var.	Time until the next rising edge in occupancy profile
tc	int. var.	Time after the last real occupancy
td	int. var.	$td = min(ta, tc)$
Pt	parameter	Preparation Period, which defines how long it takes a specific room or space to be in fine climatic conditions
Wt	parameter	Waiting Period, which establishes the time that the system allows a continuous inconsistency in a positive occupancy prediction ($OccP(t) = 1$, but $Occ = 0$)
Bt	parameter	Break Period, which fixes the maximum duration of a short absence

Table 4.2: Type of Time Period Agent inputs, outputs, internal variables and parameters.



$OccP(t_0)$	$tb>Bt$	$td>Wt$	$cond$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Figure 4.5: State Diagram of the Type of Time Period Agent.

1. When the application is active, the system goes to the default state (*Default*). From here it can transcend to Absence or Presence depending on the instantaneous occupancy ($Occ = 0$ or $Occ = 1$).

2. Irrespective of any other variable, if there are people in the facility ($Occ = 1$) the agent goes to Presence state.

3. When people leave the room/space (occupancy drops, $Occ = 0$) the systems changes to the Short Absence state. Here it faces two possibilities distinguished by the current value of the occupancy profile $OccP(t_0)$.

   (a) On one hand, the system correctly expected the absence, $OccP(t_0) = 0$. This will result in a short absence if the time until the next predicted occupancy (tb) is lower than the Break Period (Bt). Otherwise, it jumps to the Long Absence state ($tb > Bt$).

   (b) On the other hand, the system expected occupancy at this moment, $OccP(t_0) = 1$. It can happen if users leave before the expected time, but also if users do not arrive when it was predicted. Both situations present the same values ($Occ = 0$, $OccP(t_0) = 1$), but are different situations. In the first case, the system must consider how long the unexpected absence takes related to the last actual occupancy (tc), in the second case to the predicted occupancy (ta) and both have to be compared with the Waiting Period (Wt). In both situations, the systems jumps to the Long Absence state if $td > Wt$, since $td = min(ta, tc)$.

   In short, the conditional jump from Short Absence to Long Absence ($cond$) happens when Equation 4.1 is true.

4. Once in the Long Absence state, the system will remain until there is an unexpected presence or when the remaining time for the next predicted presence (tb) is shorter than the Preparation Period ($tb < Pt$). Then, it jumps to the Preparation state. Note that it is not possible to hop from the Long Absence state directly to the Short Absence state.

5. The Preparation state lasts until the remaining time for the next predicted occupancy arrival is over ($bt = 0$). Then, if there was not any presence, it again goes to the Short Absence state.

$$cond = \overline{OccP(t_0)} \cdot (tb > Bt) + OccP(t_0) \cdot (td > Wt) \tag{4.1}$$

4.3.2 PFGs and PTGs

PFGs and PTGs (Profile Generators and Pattern Generators) have been already introduced in Section 2.4. It is worth commenting again on the differences between both types of processes. PFGs are in charge of elaborating daily profiles and address them to the suitable databases, whereas PTGs take a set of profiles from databases and perform the representative profile or pattern that will be deployed in control decisions for the next day. Figure 4.6 graphically sketches such deployment from the data collection to the control decision making, passing for the generation of profiles and patterns.

Note that a PFG spends most of its time collecting data from the environment and preparing the daily profile. When the profile is full or complete, it stores the *newborn profile* in the respective database. Later on, a PTG – which is in an idle state most of the time – takes a set of profiles and abstracts representative patterns and additional information for the context. The outputs of the PTG are stored in databases as it must be available for other controllers (control agents)

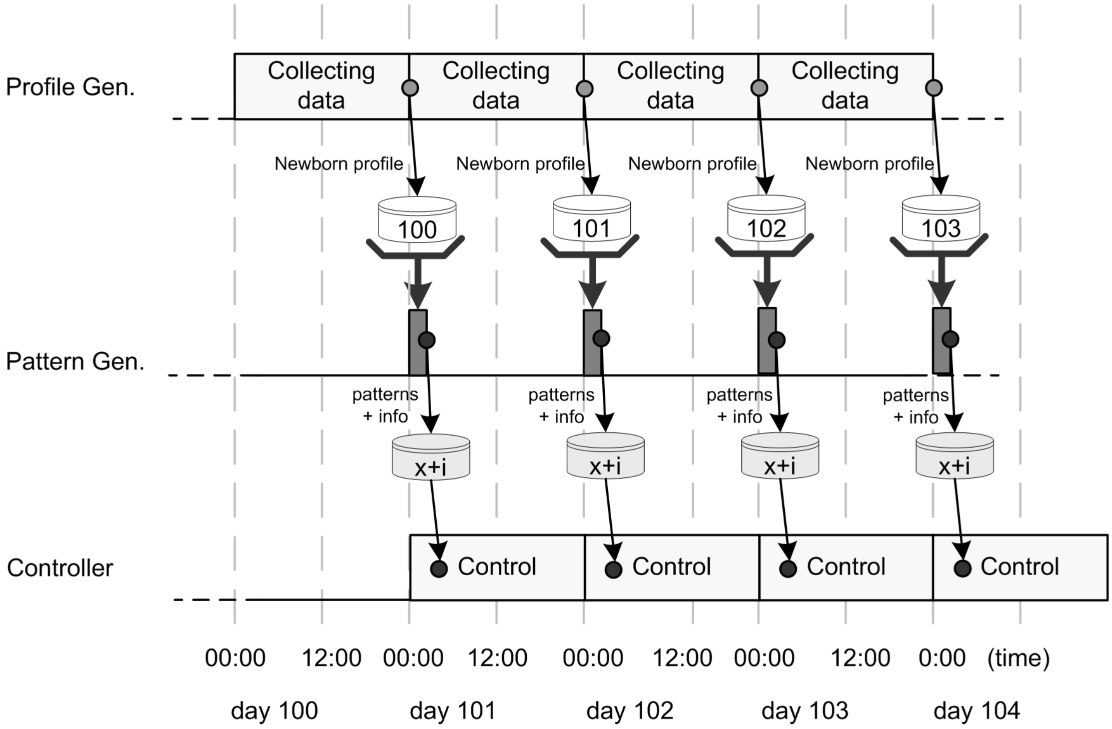


Figure 4.6: Schema of how PTGs, PFGs and Controllers cooperate and work together. ‘x’ stands for an undetermined number of patterns and ‘i’ for the extra information related to the ‘x’ patterns.

and applications. Finally, a specific controller fetches the patterns for today and performs the decision making based on them. The time required for storing the newborn profile, executing the clustering process and intermediate analysis, and fetching the patterns does not entail any annoying inconvenience, controllers can utilise previous patterns until new patterns are ready to use.

This explanation is equally valid for the three kind of profiles presented for AQTC Applications, they are: *comfort temperature profiles*, *comfort relative humidity profiles* and *occupancy profiles*.

To produce these profiles, the data collected by PFGs come from habitual and common sensors in HBA systems. For the selected services:

- *Temperatures*, by means of thermostats or electronic thermometer.
- *Relative humidity values*, collected using indoor/outdoor humidity sensors and hygrometers.
- *Occupancy*, provided by subsystems that offer a reliable reading of the occupancy of a room/space/dwelling. They are usually triggered by smart locks, buttons or contacts, and also proximity, occupancy, presence or movement sensors.

PTGs

PTGs are more complex than PFGs as they integrate artificial intelligence algorithms in order to yield summaries of habits and context appraisals. How advanced methods are able to abstract/discover patterns and offer context readings is widely dealt with in Chapter 5.

PTGs can have distinct implementation depending on the variable to abstract. In any case, the basic version of a PTG can be noticed in Figure 4.7. Here, four differentiated blocks appear:

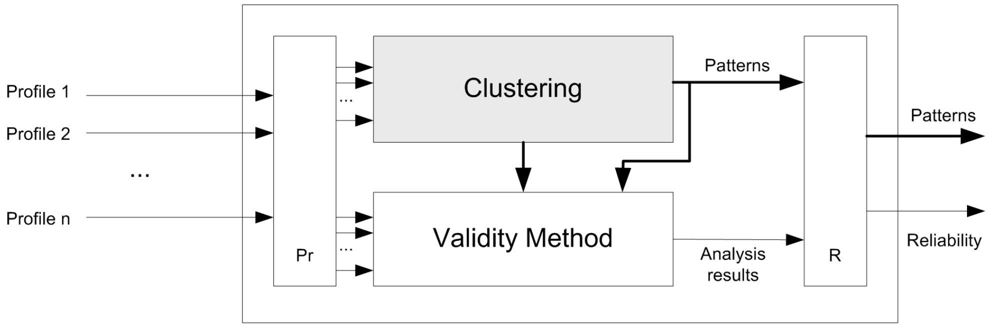


Figure 4.7: Schema of a basic PTG module.

- Preparation block, *Pr*
  It is a first input block for the data reception, filtering and normalisation. Profiles are taken from the appropriate database or from a set included in a more global database. Data normalisation issues for clustering is dealt with in Section 5.2.2. In addition, let us suppose that profiles stored in databases have already overcome a first filter of wrong data, out-of-range values, incoherences, etc. Beyond these pre-processing steps, it is possible to implement additional filtering phases, as, for example, *empty profiles* detection (i. e. profiles whose fields only have values by default, normally '0') whenever it makes sense (e. g. in case of occupancy profiles, empty profiles indicate absent days). The amount or percentage of empty profiles would be transmitted to the Validity Method block.

- *Clustering* block
  The tool in charge of obtaining/discovering representative patterns from input profiles (Chapter 5).

- *Validity Method* block.
  This block checks the validity of representatives based on extra information generated during the clustering process, e. g. number of discovered patterns, external distances (to measure how similar are the obtained representatives one another), internal distances or density (to evaluate the level of similarity of cluster members) and membership levels (to know how much population is included in the respective cluster). A wider analysis of these indexes is shown in Section 5.4.

- Output Rules block, *R*
  By means of a rule table or dedicated algorithms, patterns and analysis data are examined. Final outcomes are the representative patterns and also additional values related to the fitness of every discovered pattern. These values are usually as follows:

  - Membership level (directly taken from the Validity Method block), and expressed by a percentage or a value between '0' and '1'.
  - *Domain* or *level of dominance*. It is derived from a first hypothesis concerning clustering results. It expresses whether the respective pattern clearly dominates a specific period of time subset covered by input samples. For example, given a subset of profiles

corresponding to past Mondays, if the profiles of the subset are identified by patterns as follows: P1, P1, P3, P1, P1, P2, P1, P1; it is obvious that P1 is the dominant pattern of this specific subset (the value of dominance would be 5/8). In the case of a draw, the *most crowded* pattern (highest membership level) is selected as the dominant. If there is no dominant pattern, the selected period/subset is usually considered as *unpredictable*.

– *Reliability* is stated as a function of *membership level*, *density* and *level of dominance*, i. e. the representativeness inside the respective subset of the current database. Although closely related, reliability should not be mistaken for likelihood, the former just shows stability and soundness of past behaviours.

Unpredictable domains or unreliable patterns indicate erratic behaviours or unstable, lax users' habits. The isolated existence of such cases does not entail any critical situation or a severe degradation of the system's performance; in such an undesired case, profile-based controllers are able to detect high and widespread levels of unpredictability and therefore switch to alternative strategies, as habit-based prediction would make no sense in the given situation.

The joined running of the four blocks can be more clearly understood with an example. Given a database with daily profiles concerning 'water consumption', let us suppose that the PTG must work with a set of 120 profiles corresponding to the last four months. Block 'Pr' filters 15 'empty profiles' (days without water consumption), later on 'Clustering' finds 4 different patterns. The 'Validity Method' block analyses distances and membership levels and gives all the information to the final 'R' block. Detected close distances between some representatives causes it to reduce the number of final representatives from 4 to 2. This PTG also performs analysis of the 'day of the week' that the patterns dominate, here it decides to include a 'empty profile' as a representative due to the considerable repetition and concentration. Finally, 'R' states (outputs):

- Pattern 1, size: 56%, density: 0.6 (moderate), domain: Monday, Tuesday, Thursday.
- Pattern 2, size: 23%, density: 0.8 (high), domain: Friday, Saturday.
- Pattern 3 ('empty profile'), size: 13%, density: 0.9 (high), domain: Sunday.
- Reliability of... Mondays: 0.5, Tuesdays: 0.6, Wednesdays: unpred., Thursdays: 0.6, Fridays: 0.6, Saturdays: 0.7, Sundays: 0.8.

The percentages missed in *size* evaluations correspond to skipped outliers or minor clusters. In addition, note that we do not have a pattern for Wednesdays, so each respective controller that uses the outputs of this PTG will have to decide what it does on Wednesdays according to its specific parametrisation.

4.3.3 Setpoint Temperature Control & Setpoint Relative Humidity Control

The potential of these two control applications remains in the intelligent adjustment of setpoint temperatures according to instantaneous occupancy and occupancy predictions. Comfort is guaranteed as the system learns what the desired comfort is from the habits abstracted in patterns. At the same time, the system predicts occupied periods and turns on the heating/cooling to comfort conditions some minutes before the users' arrival (preparation periods). The energy performance is optimised as long as the system switches to reduced setpoint levels in unoccupied periods (setback, setforward and safety rates). Safety levels are stated for long absence periods to avoid damages related to extreme conditions. Setback and setforward values exist for short absences and nighttimes in order to maintain acceptable conditions and allow quick thermal comfort recoveries, both under a good energy performance. In fact, for temperature adjustments, improving energy savings has been found to be very suitable, mainly in extreme climates [MS05].

Since Setpoint Temperature Application and Setpoint Relative Humidity Application show a very similar design, we will explain here the case of temperature only and assume the case regarding relative humidity as to be analogously faced. Most HVAC systems automatically adjust relative humidity without considering relative humidity setpoint values as inputs that can be externally tuned. As we referred to above, the purpose of the proposed control is not bypassing any function of the specific end equipment, but to cover the diverse possibilities that exist for HVAC performances from the pre-control phase.

Table 4.3 and Figure 4.8 show inputs, outputs and parameters considered for the Setpoint Temperature Control Agent.

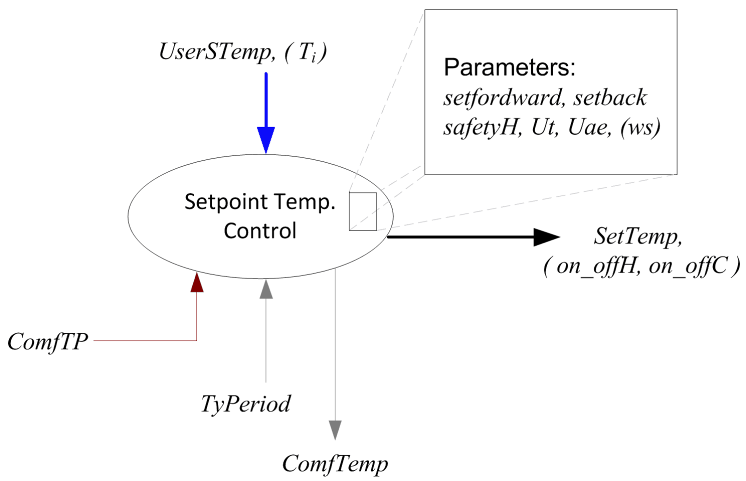


Figure 4.8: Inputs, outputs and parameters of the Setpoint Temperature Control Module.

In these services, agents are not only in charge of fixing setpoint values and switching the equipment on/off in case it is required, but also to give comfort values to the Comfort PFG agent. This is the reason why there is an output called comfort temperature ($ComfTemp$) in addition to the normal setpoint temperature output (*SetTemp*). The difference between these two variables is explained as follows: as long as *SetTemp* is the real setpoint temperature, $ComfTemp$ shows the same value whenever occupancy is detected. In other respects, when there is no occupancy, whereas *SetTemp* usually switches to setback or safety, $ComfTemp$ keeps the values that *SetTemp* would have in case that there were people at home. In order to do that, $ComfTemp$ simply copies the values given by the current profile that the agent is using for control. Therefore, comfort temperature profiles inform us about the desired temperature values of users, supposing that they are always present. At the end, *SetTemp* is delivered to the heating and cooling controllers, and $ComfTemp$ is used to elaborate the daily comfort profile.

Object	Type	Description
$UserSTemp$	in.	Setpoint Temperature (manual)
$TyPeriod$	in.	Type of Time
$ComfTP$	in.	Comfort Temperature Profile
$SetTemp$	out.	Setpoint Temperature
$ComfTemp$	out.	Comfort Temperature
on_offH	out.	On/off Heating
on_offC	out.	On/off Cooling
ws	par./in.	Winter/summer mode
$setforward$	param.	Cooling Setforward Temp.
$setback$	param.	Heating Setback Temp.
$safetyH$	param.	Heating Safety Temp.
Ut	param.	User Period
Uae	param.	User Adjustment Effect

Table 4.3: Setpoint Temperature Control Agent inputs, outputs, internal variables and parameters.

Activation and User Adjustments

The activation of the Setpoint Temperature Control Application by users starts the process shown in Figure 4.9. After the activation (*STC Activation*), in the first step, the system decides whether it must run in winter mode or summer mode (or even mild climate season if it is defined). In spite of the fact that it does not appear as an input (it is considered as a parameter), this evaluation can be established manually by users, by feedbacks from HVAC equipment or according to additional variables, e. g. indoor and outdoor temperatures, time and date, etc. In short, the matter concerning season mode has a variable implementation (here, regardless of the origin, changes in the season mode are dealt with as interruptions). Therefore, the system starts one of the setpoint temperature control routines (see below), either heating or cooling, and remains there until there is a change of season mode or if there happens any adjustment by users.

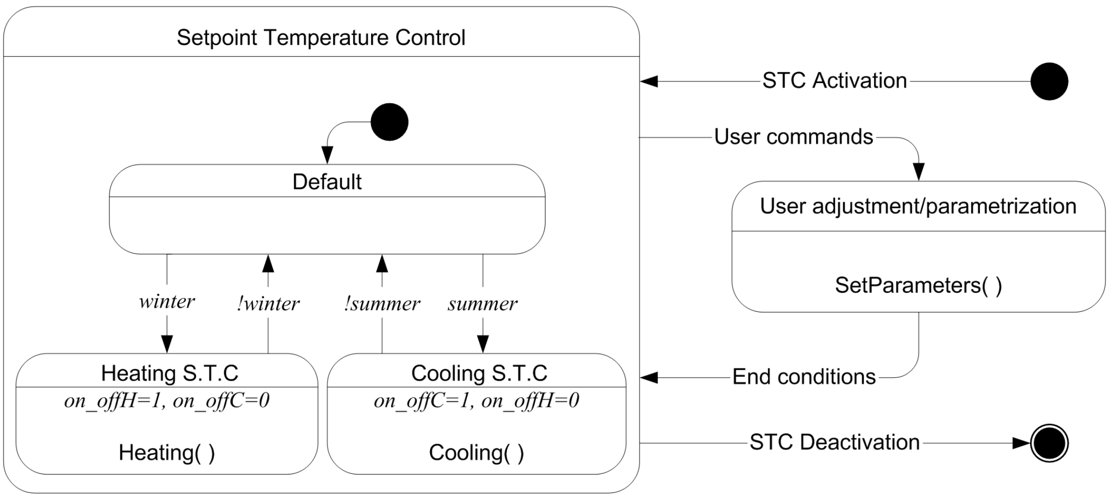


Figure 4.9: State machine for the Setpoint Temperature Control.

Control processes are interrupted whenever users intervene by means of commands (*User commands*). In such situations, the system executes algorithms related to users' adjustments (through the User Preferences Agent, e. g. *SetParameters*). The adjustments include changes in the setpoint value, but also modifications in other parameters or even the deactivation of the service. When the service is active, the setpoint value is automatically adjusted based on the comfort pattern. If users manually change the setpoint, the control agent replaces the values of the current comfort pattern by new values based on the user action/desire. How this modification is carried out depends on the *Uae* parameter (e.g. 'keep the new temperature value until the next absence', 'apply the difference of temperature ($dTemp = T_{previous} - T_{desired}$) to the rest of the whole current comfort temperature profile', etc).

In addition, the *Ut* parameter states the period of time that the service is allowed in an *idle state* after any user adjustment or modification in the setpoint. Therefore, *End conditions* in Figure 4.9 represents either when adjustment algorithms have finished or, if setpoint temperatures have been modified, the *User Period* time is just over. *Ut* and *Uae* parameters are usually under control of Shadow Processes (Section 3.7.1).

It is worth noticing here how the system deals with user changes concerning desires and habits. The *Uae* parameter executes a quick and active response to user modifications, whereas the performance of future habit profiles is a slow and passive reaction. A good management of the Comfort PTG will end up reaching a perfect adaptation to user desires and evolving habits.

Heating & Cooling Processes

Table 4.4 shows in more details actions related to time periods for heating processes, *Heating()*. We can consider cooling as an analogous, twin function that involves minor variations.

Time Period	Heating
Presence	$SetTemp = ComfTP(t_0)$
Short absence	$SetTemp = setback$
Long absence	$SetTemp = safetyH$
Preparation	$SetTemp = ComfTP(t_0)$

Table 4.4: Time periods and actions for the heating process

Thus, in each time state, the system applies a different value on the setpoint temperatures output ($SetTemp$).

- Presence → The setpoint temperature is fixed with values given by the current comfort temperature profile.
- Short Absence → The setpoint temperature is fixed with setback values (usually between 16 and 19 degrees).
- Long Absence → The setpoint temperature is fixed with safety values (usually between 8 and 10 degrees).
- Preparation → Again, the setpoint temperature is fixed with values given by the current comfort temperature profile.

On the other hand, the value of the output comfort temperature is defined as a function of:

$$ComfTemp = f\left[ComfTP(t_0), UserSTemp, Uae\right] \tag{4.2}$$

where $ComfTP(t_0)$ is applied by default, but submitted to the priority $UserSTemp$ instantaneous values, whose effect is defined by Uae.

Finally, values of on_offH have a high dependence on the equipment. Advanced heating (or HVAC) equipment will not necessarily be required to be switched on and off, but only fed with the setpoint temperature (hence they will manage the on/off status). In this case, the application does not need to have values concerning indoor temperatures (T_i). In other cases, on_offH can be defined as a function of:

$$on\_offH = f\left[ComfTP(t_0), T_i, ws, TyPeriod\right] \tag{4.3}$$

Switching Patterns and Strategies

In Section 4.3.2 we have seen that controllers usually have more than one pattern at their disposal, as well as information about their reliability and relevance (everything provided by PTGs). The combination of such data can make controllers switch patterns or even strategies depending on the case. Let us see how that is carried out for the current application by two different control modalities:

In a normal situation, for a given habit profile (e. g. *dwelling occupancy*) we have n available patterns ($patt_i$, $i = \{1, ..., n\}$), originally provided by the respective PTG, where n is the number of significant representatives.

$$ComfTP(t) = patt_i(t) \tag{4.4}$$

Depending on the controller/case, i is fixed according to one of the next options or combinations:

- *Pattern size.*
  The controller applies the pattern by default whose *size* is higher (i. e. the pattern represents the biggest amount of past daily profiles). As the day passes, *notable differences* between the applied pattern and the real day profile make the controller switch patterns for the next predictions (within the same day). Therefore, the controller applies the most representative pattern by default, but changes it if, when *notable differences* are detected, the trend of the real case is more similar to a secondary pattern. Figure 4.10 shows an example of when the controller decides to switch in $t = 6$ (hours) due to

  $$sim_{0\to6}(r\_case, patt_1) < sim_{0\to6}(r\_case, patt_2) \tag{4.5}$$

  sim denotes a *similarity* function, r_case refers to the current occupancy case, $patt1$ and $patt2$ are the two patterns found by the clustering tool deployed in the example.

  Drifts in patterns become *notable differences* if they exceed a certain threshold usually fixed by the duration of profile fields.

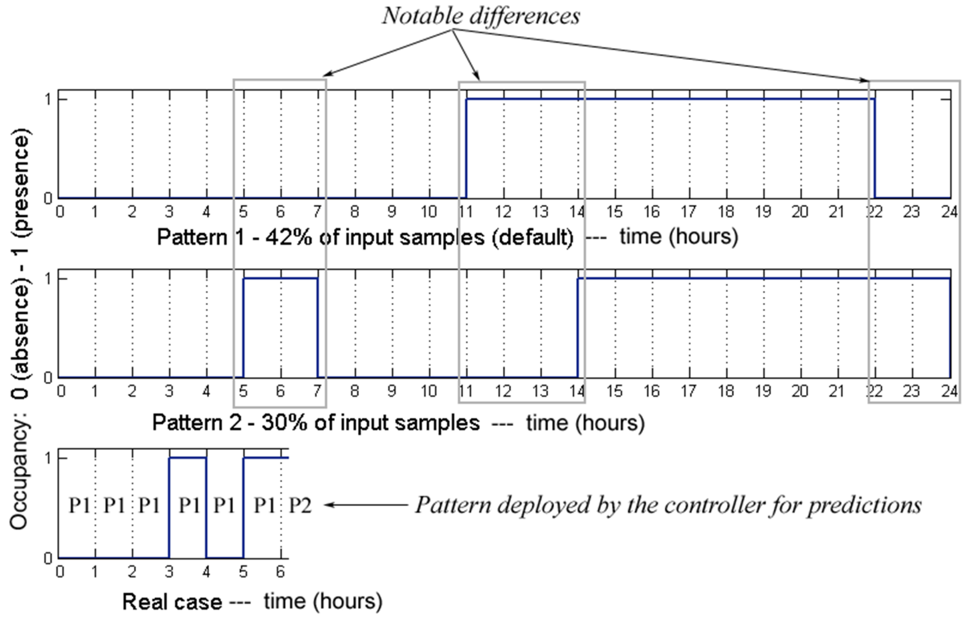


Figure 4.10: Change of pattern depending on the evolution of the present day. Two occupancy patterns and the real case are shown.

- *Pattern domain.*
  In this case, the controller applies the pattern whose dominion includes the present day. If the present day has not been clearly bound to any pattern, it is declared as a *non-predictable day*. Then the solutions for the controller are the same as if it had been covered by a pattern with a low reliability (see below).

In both cases, controllers *check the reliability* of the selected pattern for the present day. If the value does not overcome a certain pre-fixed threshold, the controller executes alternative routines according to its parametrisation. For the Setpoint Temperature Controls case, three options are considered:

- *Keep the most representative pattern (highest* size*).* Option by default.
- *Switch to the scheduled or combined strategy.* In case *comfort* is priorised by users (User Preferences Agent).
- *Switch to the on/off strategy.* In case *energy savings* is priorised by users (User Preferences Agent).

On/off, Scheduled and Combined strategies are explained in Appendix B.1.3. The introduced options are some basic examples to face critical scenarios, but not unique. Low habit-stability and low performance detection trigger automatic routines but also informative actions. Checking the application performance is the matter of Shadow Processes, it is described in Section 3.7.1.

4.3.4 Air Quality Module

The Air Quality agent performs evaluations of the current indoor and outdoor air quality. Although other options for the air quality assessments could be implemented instead, we propose an approach consistent with the deployment of habit profiles and the capabilities of smart home systems to embed advanced intelligence methods into independent agents.

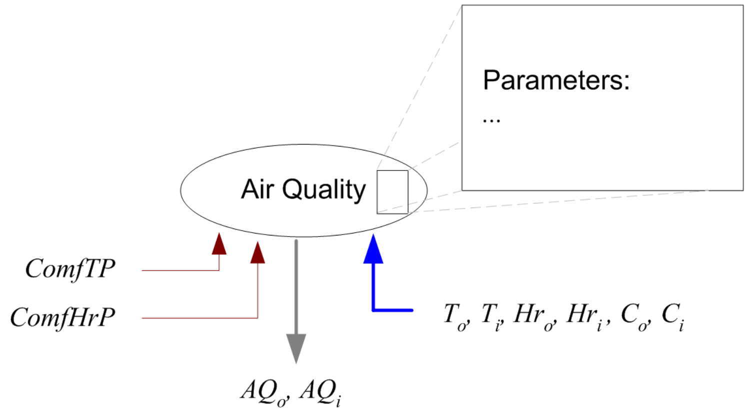


Figure 4.11: Inputs, outputs and parameters of the Air Quality module.

The inputs, outputs and required parameters are shown in Figure 4.11 and Table 4.5.

Object	Type	Description
$ComfTP$	in.	Comfort Temperature Profile
$ComfHrP$	in.	Comfort RH Profile
T_o	in.	Outdoor Temperature
T_i	in.	Indoor Temperature
Hr_o	in.	Outdoor Relative Humidity
Hr_i	in.	Indoor Relative Humidity
C_o	in.	Outdoor CO_2 Concentration
C_i	in.	Indoor CO_2 Concentration
$param$	param.	Set of parameters based on directives for the fuzzy rules adjustment
AQ_o	out.	Outdoor Air Quality
AQ_i	out.	Indoor Air Quality

Table 4.5: Air quality agent inputs, outputs, internal variables and parameters.

In the developed case, the Air Quality agent is implemented deploying data concerning indoor and outdoor temperature, relative humidity and CO_2 levels. A suitable way to obtain the qualitative output is by means of fuzzy inference systems for both indoor and outdoor data. In Figure 4.12, we can see an example of such a system.[3]

Instead of directly managing absolute temperature and humidity values, Air Quality modules compare instantaneous values with comfort values stored in profiles, calculating the difference.

[3] The suitability of fuzzy systems for air quality can be seen, for example, in [Fis03].

Thus, the system adjusts the ideal point dynamically for the later fuzzy reasoning according to the user habits for every moment of the day and for different seasons. It means that, for instance, an indoor temperature of 19°C can be seen as *acceptable* at night and *quite cold* at noon depending on the values stored by comfort profiles.

Therefore, the difference of comfort and real values in temperature and relative humidity, as well as CO_2 concentration, are normal inputs, whereas the air quality evaluation is the output.

$$dTemp(t) = ComfTP(t) - T(t) \tag{4.6}$$

$$dHr(t) = ComfHrP(t) - Hr(t) \tag{4.7}$$

For the assessments, some limits are settled from the comfort values given by the profiles until rates, which have been published by competent norms in order to guarantee healthy and habitable conditions. The shapes of the rules (slopes and rule weights) are adjusted according to the users' preferences (for example by means of the User Preference Agent). As an example, if users prioritise 'energy saving' modes, the slopes of *comfort sets* would result flattened, Figure 4.13.

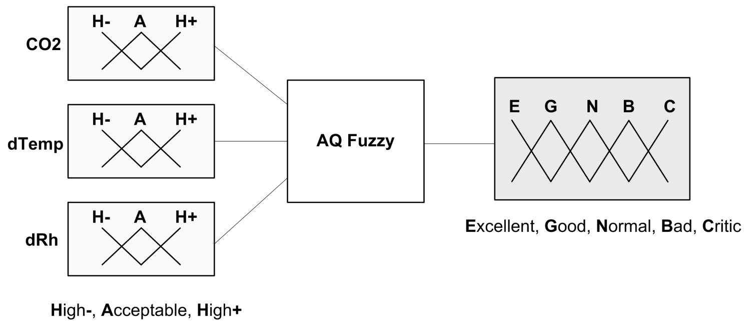


Figure 4.12: Fuzzy Inference System for Air Quality. The same agent includes two parallel modules, one for outdoor air quality and one for indoor air quality

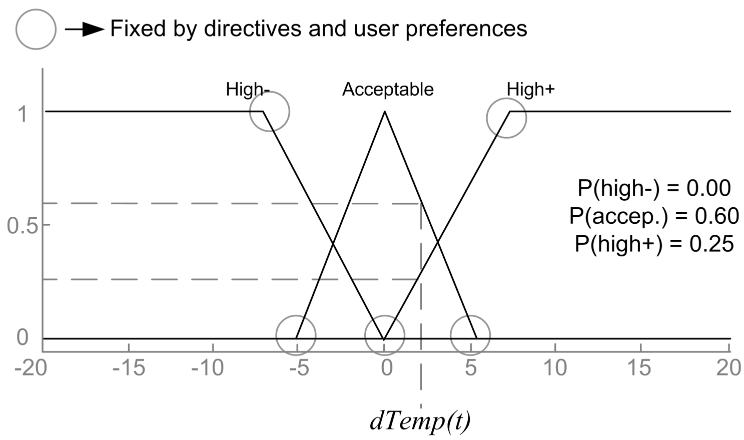


Figure 4.13: Rules of fuzzy sets are adjusted by means of directives and user preferences. Supposing that we are in an 'energy savings' mode, switching to a 'comfort' mode would shrink the base of the triangle that represents the 'Acceptable' set.

The assessments concerning indoor air quality are necessarily based on data provided by sensors installed in the Smart Environment and transmitted to the Air Quality agent by the Interface System-Equipment agent. However, data regarding outdoor conditions can also be provided by means of remote services (e.g. by a Auxiliary Data Agent, not included in the schema of Figure 4.3), as outdoor CO_2 levels are usually stable, and outdoor temperature and humidity can be facilitated by local weather stations.

4.3.5 Thermal Comfort Support & Ventilation Demand

Figure 4.14 depicts the schema of the *Thermal Comfort Support Agent*, inputs, outputs and parameters are also shown in Table 4.6 .

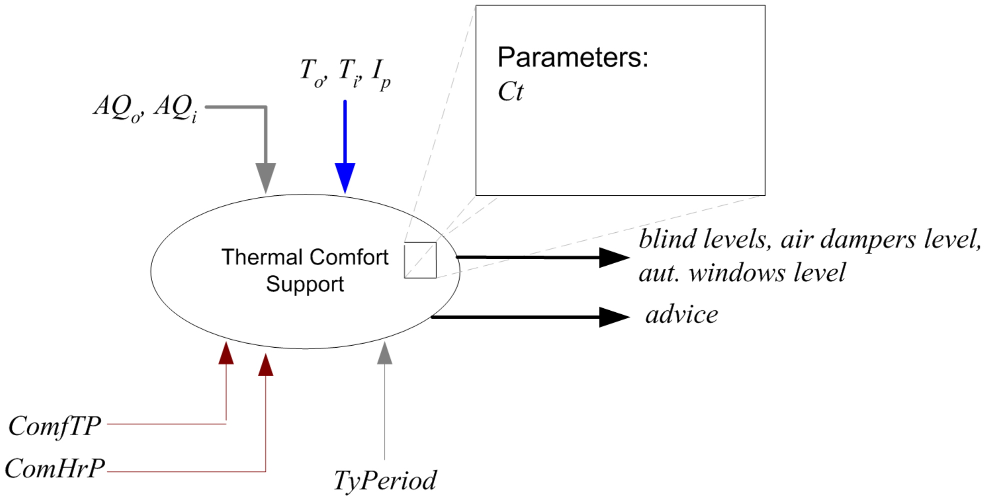


Figure 4.14: Connected agents, inputs, outputs, parameters and internal variables of the Thermal Comfort Support Module.

Object	Type	Description
$ComfTP$	in.	Comfort Temperature Profile
$ComfHrP$	in.	Comfort Relative Humidity Profile
T_o	in.	Outdoor Temperature
T_i	in.	Indoor Temperature
I_p	in.	Solar Gain through Panes
$TyPeriod$	in.	Type of Time
AQ_o	in.	Outdoor Air Quality
AQ_i	in.	Indoor Air Quality
Ls	param.	Load Status Mode
Ct	param.	Cadence Period
outputs	out.	Blinds level, Automated Windows Status, Air Dampers Status
advice	out.	Set of advice

Table 4.6: Air quality agent inputs, outputs, internal variables and parameters.

Regarding the application objectives, it requires two main roles depending on the house occupancy status:

- In **unoccupied periods**, it increases the thermal insulation or balance indoor and outdoor conditions aspiring to maximize indoor air quality and thermal comfort for the next users' arrival. It is carried out keeping the minimal energy consumption by means of acting on blinds, automated windows, shading devices and air dampers. For example, during summer, blinds can be closed to reflect incident radiation back out the windows and avoid an additional heating load in the room. Otherwise, in winter the process can be reversed to retain this extra heating.
- On the other hand, in **occupied periods**, it performs its outputs as advice and recommendations for users, in keeping with the idea of natural comfort with minimum energy costs.

Some previous studies support the management strategies shown by the Thermal Comfort Support Application. For instance, the energy cost differences between performances with and without shading devices (internal and external) have been measured for the Canadian housing in [GRSM05]. They conclude with about 15-20% of energy savings after an appropriate management of shading devices, both in winter and summer seasons.

Thermal Comfort Support Agent

The Thermal Comfort Support Agent also highlights dependence on the time periods settled by instantaneous occupancy and occupancy profiles ($TyPeriod$). Table 4.7 defines the action of the controller depending on the 'type of time'.

Time Period	Role	Comments	Process
Presence	Advisor	Messages for users	[$LoadStatus()$], $ShowAdvice()$
Short absence	Active	Management of devices	$SetDeviceStatus()$
Long absence	Idle	No action	—
Preparation	Active	Management of devices	$SetDeviceStatus()$

Table 4.7: Actuation of Thermal Comfort Support Application according to $Typeriod$.

Regardless of the fact that cooling and heating is used in short unoccupied times or not, the Thermal Comfort Support Application tries to approach to desired thermal conditions by means of low energy demanding block/unblock strategies. Hence it is equally valid, as heating and cooling will be required in the next occupied time in order to reach comfort conditions.

The running of the system is even simpler than in the setpoint control case. If there is presence, the application remains in an advisor mode. Otherwise, it is only active during short absences and preparation times. Then, the *SetDeviceStatus* process is executed.

'SetDeviceStatus' is a routine triggered according to a pre-defined polling time (Ct). Then, the diverse inputs of the module are evaluated and some decisions concerning the actuation over devices are undertaken. A simplified version of the decision making process can be appreciated in Table 4.8.

The meaning of the actions is as follows: when *idle*, the smart control does not execute any action; with *balance* it opens blinds, automated windows, air supply and exhaust dampers; whereas *insulate* means to close blinds, automated windows, air supply and exhaust dampers.

However, some controversial situations can occur. For instance, it is possible that ventilation insulation is required to keep good indoor conditions but solar gain through panes can even

Indoor AQ	Outdoor AQ	Action
good	good	idle
good	tolerable	insulate
good	bad	insulate
tolerable	good	balance
tolerable	tolerable	idle
tolerable	bad	insulate
bad	good	balance
bad	tolerable	balance
bad	bad	idle

Table 4.8: Air quality insulation table of rules. Summarised version from a fuzzy inference system.

improve the indoor air quality. In order to solve such situations, *solar gain through windows* (Q_{sg}) and *sensible transmission through glass* (Q_{tg}) are compared. Q_{sg} and Q_{tg} can be summarised [Ash97] in the following equations:

$$Q_{sg} = k_1 \cdot A_g \cdot SHGC \cdot SC \cdot I_p \tag{4.8}$$

$$Q_{tg} = k_2 \cdot A_g \cdot U_g(T_o - T_i) \tag{4.9}$$

Where A_g stands for the pane's surface, U_g is the U-Factor or U-Value and indicates how well the window conducts heat. On the other hand, $SHGC$ is the Solar Heat Gain Coefficient (dimensionless) and measures how well a product blocks heat from the sun. SC is the Shading Factor (dimensionless) and I_p offers the solar radiation over the window [W/m^2]. k_1 and k_2 represent engineering parameters.

The comparison between equations 4.8 and 4.9 provide us with an appraisal about thermal gains through the windows in a normal status (blinds[4] rolled), and help the system to know whether acting over the blinds can be suitable. For this purpose, Table 4.9 is defined for the operation of the blinds according to the thermal gain contribution (it has priority over Table 4.8, only conflictive cases are shown).

Desired effect	Q_{tg} sign	$\lvert Q_{tg}\rvert ... \lvert Q_{sg}\rvert$	Blinds/shutters
cool down	- (loss)	$>$	open
cool down	-	$<$	close
warm up	-	$>$	close
warm up	-	$<$	open

Table 4.9: Set Device Status Process Table for Blind Operation.

The *desired effect* is calculated comparing the indoor/room temperature (T_i) with the setpoint temperature offered by profiles for the specific moment, $ComfTP(t_0)$. For instance, if $T_i = 12°C$ and $ComfTP(t_0) = 20°C$, the control agent realises that *warming up* strategies are suitable at the moment. The Q_{tg} sign marks whether there are thermal conduct gains or losses through the window; since Q_{sg} cannot be negative, conflicts can happen when Q_{tg} is negative.

The simple approach given by equations 4.8 and 4.9 can be enough to have suitable assessments about heat gains and losses due to conduction-convection and solar radiation at the same time.

[4] From now on we will refer indistinctly to all kind of shading devices as blinds.

In this point, it is necessary to count on additional information about the house or building. For example, accurate SC and $SHGC$ assessments take different values depending on materials, type of fenestration, type of room floor, date and time, building location and geometry, glazing orientation, solar incidence angle, frame effects, shading devices, etc. By means of tables [Ash97] and a set of sensors, smart systems can perform adequate calculations.

Nevertheless, beyond the proposed solution, the usage of simulation tools to predict different future scenarios and make decisions ahead (building simulations for decision making of controllers) is an advanced option to consider; indeed, the agreement between measures and simulations for solar gain in windows has been already demonstrated [KWCB09]. In addition, due to the exposed dependence on the building characteristics, a smart home control supported by BIM [KK10] is empowered by this sort of application.

Advice Generation & Scenario Loading

The *advisor capabilities* of the smart system are intended to produce advice and recommendations to users in occupied times. Then, the direct actuation of the Thermal Comfort Support Application is substituted by the publication of its *reasonings* in the form of advice; it is the *ShowAdvice* process. It covers mainly:

- Recommending to open or close windows and hatches.
- Suggesting to apply or remove internal and external shading devices, but considering the users' lighting requirements.[5]

Advice is published in the home UIs or directly in the objective house elements; for example, dedicated colour LED lights on windows and blinds can inform users about the convenience of changing the status. This informative service can perfectly coexist and even empower other persuasive applications for the comfort and energy optimisation (Section 3.6).

The *LoadStatus* process decides how to instantly react in front of the next users' arrival concerning the status of the controlled devices. Depending on the Ls parameter, possible options are:

- Keep the current status.
- Switch to the status active at the end of the last (previous) presence.
- Load a pre-defined status.

Ventilation Demand

The Ventilation Demand Application is the only one of the introduced applications that does not strictly require profiles. It is intended to execute ventilation on-demand and to be more active when the HVAC equipment is basic, uses devices from different manufacturers or does not have context awareness capabilities. This agent mainly switches on/off equipment, zones, and adjusts fan levels. The considered implementation basically operates with $TyPeriod$, AQ_o and

[5]It demands the integration of indoor and outdoor light level sensors. Lighting level profiles can be also utilised for the final advice performance.

AQ_i, activating ventilation when there are people in the facility ($TyPeriod = Presence$) and air quality assessments require it.

Even if the HAS installation includes an advanced HVAC system, the integration within the overall smart home system and the pre-control represented by the Ventilation Demand Agent is suitable, as conflicts with other applications, users' habits and devices are managed.

4.4 Examples of the Depicted Applications

4.4.1 Setpoint Temperature Control

In order to evaluate the capability for decision making of control agents, Figure 4.15 shows an example where the Setpoint Temperature Control Agent (abbreviated as CA) operates using the information provided by the Type of Time Period Agent (TA). It is possible to see the cooperative running of both agents carefully following the timeline. Tables 4.1, 4.2 and 4.3 are required for the next explanations.

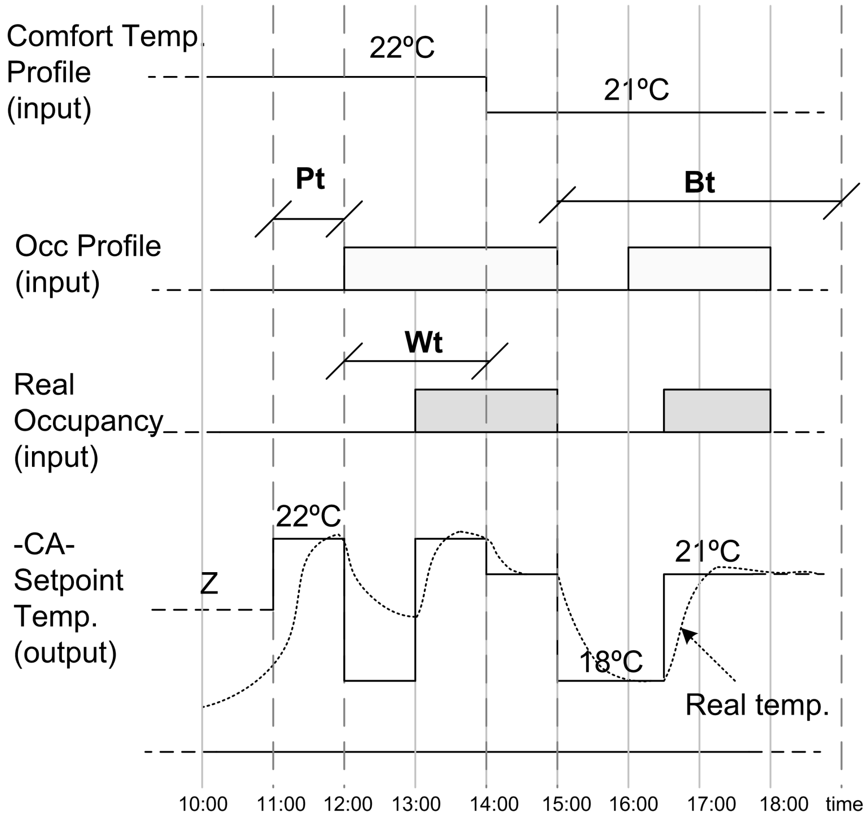


Figure 4.15: Example of Setpoint Temperature Control action with $Pt = 1h$, $Wt = 2h$ and $Bt = 4h$.

- In the example situation, at 10:00h. the control agent is running in an idle mode (Z). TA comes from a Long Absence and cannot leave the state while $Occ = 0$ and $tb > Pt$.

- 11:00h. $OccP$ predicts the next arrival at 12:00h. Since $Pt = 1h$, one hour before the predicted arrival TA enters in the Preparation state, as $tb < Pt$. Therefore, CA switches on the heating fixing the setpoint temperature by comfort profiles ($SetTemp = ComfTP(t)$).
- 12:00h. Users do not arrive at the predicted time for an unexpected reason. $tb = 0$, so TA jumps to the Short Absence state and the CA switches to setback values ($SetTemp = setback$). Since $ta < Wt$ (alike tc), TA continues to consider the current period as a Short Absence.
- 13:00h. Users arrive ($Occ = 1$), it jumps to the Presence state ($SetTemp = ComfTP(t)$).
- 14:00h. There is a change in the comfort profile, according to the usual habits.
- 15:00h. Now a well-predicted absence happens. In a real application both falling edges in instantaneous occupancy and occupancy profile will not occur simultaneously. Analysing both cases:
  a) Occupancy profile falls earlier. As it still happens that $Occ = 1$, TA maintains the Presence state. When people finally leave the place ($Occ = 0$), TA jumps to the Short Absence state and CA switches to setback ($SetTemp = setback$). Since $tb < Bt$, TA continues to consider the period as a Short Absence.
  b) Instantaneous occupancy falls earlier. TA automatically switches to the Short Absence state and CA switches to setback ($SetTemp = setback$). Now, $tc < Wt$ and (also $ta < Wt$), so TA continues to consider the moment as a Short Absence. When finally the occupancy profile falls $OccP(t) = 0$, the evaluation in the case *a)* takes place.
- 16:00h. TA predicts occupancy at this time, but users are late again. Since TA does not come from the Long Absence state, it cannot go through the Preparation state, so it holds the Short Absence state. The fact that ta and tc continue to remain shorter than Wt corroborates this decision.
- 16:30h. TA detects occupancy ($Occ = 1$) and hops to the Presence state. CA switches to comfort values ($SetTemp = ComfTP(t)$).

4.4.2 Thermal Comfort Support Example

The Thermal Comfort Support Application can be clearly assessed in the next example cases. Both cases can be considered consecutively in time, in order to understand how the system adapts to a sudden scenario change.

In Fig. 4.16, by means of Occ and $OccP$ the system knows that the present *type of time* is a *short absence*. The weather corresponds to a winter season, with low outdoor temperatures but in a sunny day. The fuzzy evaluations concerning air quality leads the system to take the *insulation* strategy; it is, to close the air damper (AD) and the blinds. However, solar gains through $W.1$ overcome thermal loses therefore the blinds in this window continue to be unrolled.

In Fig. 4.16, some users enter in the room and the application leaves the active role to switch to advisor mode. Air dampers are opened by the normal HVAC controllers. Depending on the customised configuration, the status of blinds can remain the same (according to the decisions taken in the absent period) or can be restored to the conditions saved during the last (previous)

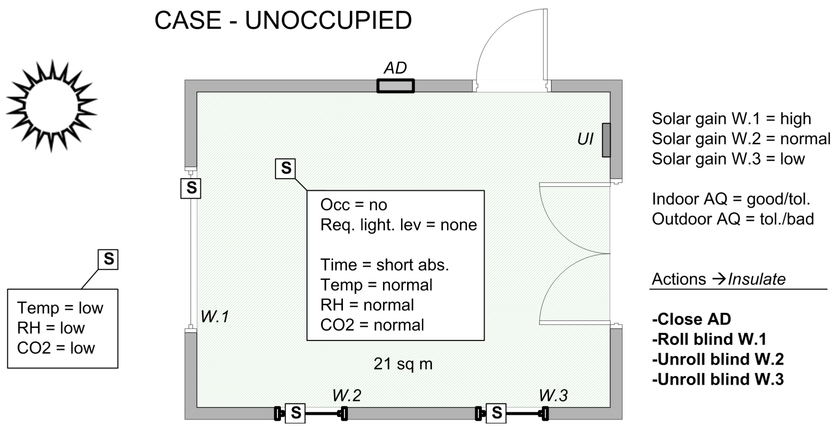


Figure 4.16: Example of the Thermal Comfort Support Application: unoccupied case.

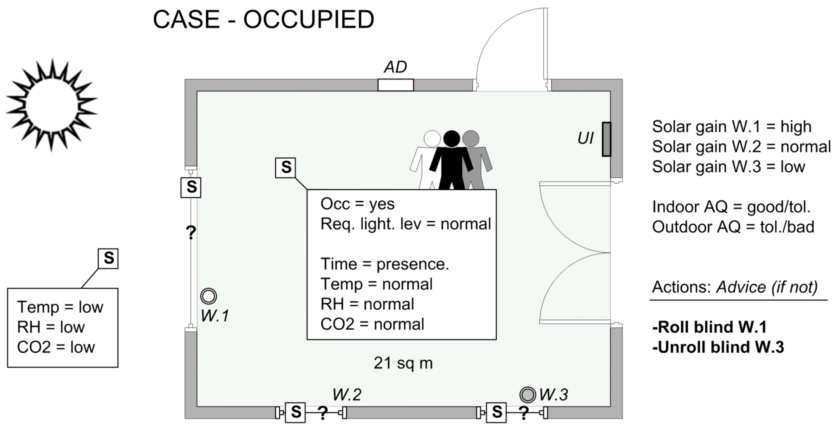


Figure 4.17: Example of the Thermal Comfort Support Application: occupied case.

occupancy. In any case, if the status does not coincide with the ideal scenario reasoned by the Thermal Comfort Support Application, it is informed in UIs and directly in device indicators. Note that, in the example, there is no advice concerning the window W.2 due to a certain level of admitted uncertainty as lighting requirements also come into play.

4.5 Comparison of Strategies for the Setpoint Temperature Control

The purpose of the following test is to compare profile-based control strategies for the Setpoint Temperature Control with classic strategies, establishing which one obtains the best energy and comfort performances.

4.5.1 Scenario and Test Description

Given a dwelling/office with a system able to automatically adjust the *setpoint temperature*, energy and comfort performances are checked by means of a simulated representation of the smart dwelling and its behaviour throughout time. The simulated environment is widely depicted in Appendix B.1.

Simulations cover repetitions of the same test, but change the setpoint temperature controller and the users (five different families or group of users). The duration of each simulation is 16 days. Therefore, each controller is checked five times for the same 16-day period. Figure 4.18 shows a schematic overview of the whole process.

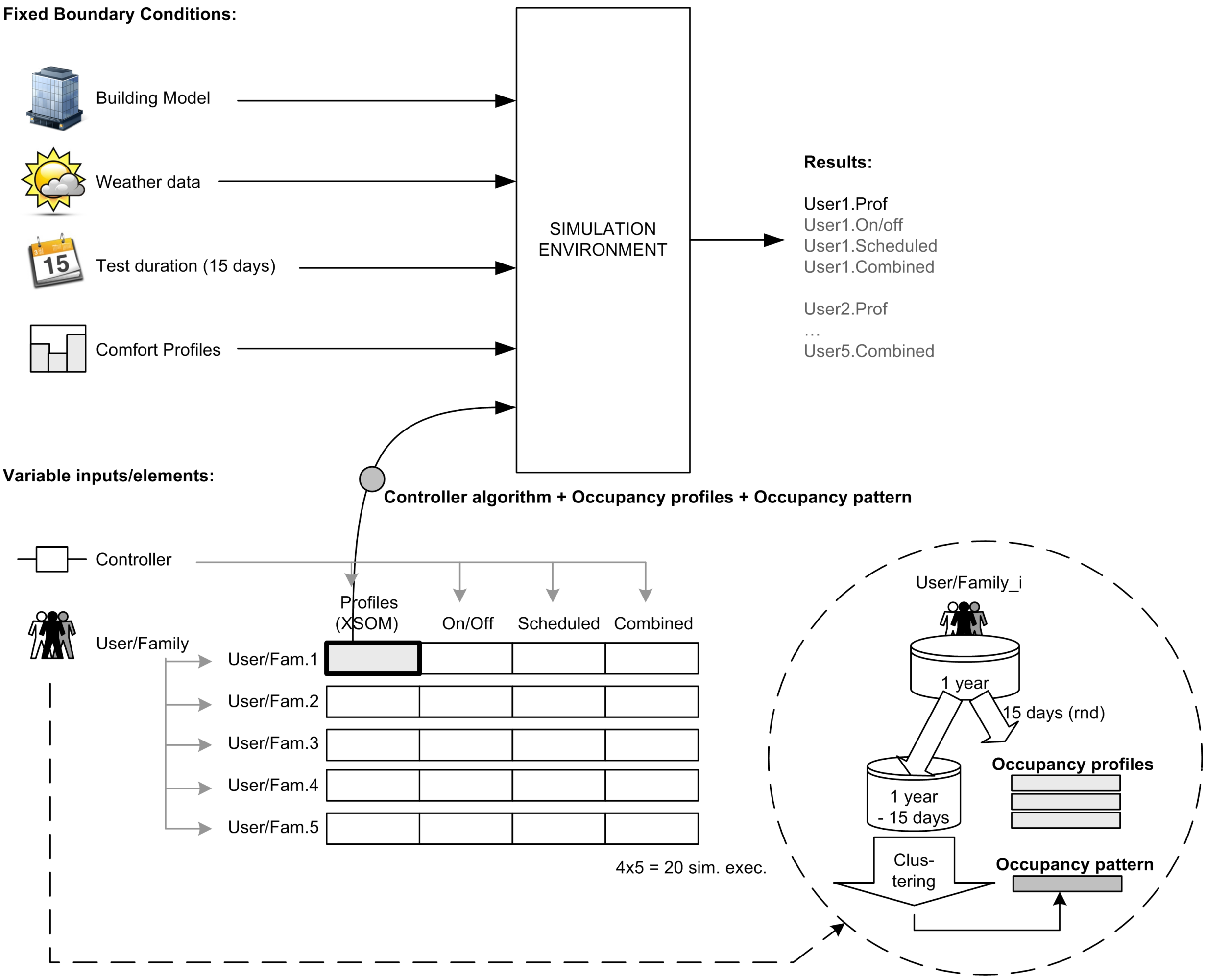


Figure 4.18: Comparison of Strategies for the Setpoint Temperature Control: test design.

The effect of each different family or group of users appears with regard to the occupancy data. For each family, the occupancy of every simulated day is different, as well as the occupancy pattern deployed by controllers.

On the other hand, the same comfort preferences (comfort profiles) for all families are assumed in order to fix boundary conditions and equalise all cases (it does not benefit nor penalise any case/strategy). Thus the evaluation is carried out under occupancy variations, leaving comfort differences for the next analysis.

4.5.2 Test Execution and Evaluation Method

The next specifications and features belong to the specific tests developed in this section, the rest is configured as shown in Appendix B, options by default.

Simulations are executed in the autumn/winter period, meaning that that only heating is applied. The applied weather data corresponds to De Bilt (Netherlands), 52.12°N 5.18°E; 11th to 27th, November, 1981.

Controllers are identified by the strategies they execute. Therefore, the strategies under test are: strategy based on profiles (XSOM), on/off control, schedule-based control, and a combined controller. The outputs of the simulations are: consumption, setpoint temperatures and real indoor temperatures.

To perform the tests, water usage data for one year from five families has been selected at random from different buildings (Leako database, Appendix B.2.1). This data is transformed by a smoothing filter, in a matter that occupancy is inferred according to the collected water consumption. Thus, an occupancy database is obtained for each family. 16 days are selected at random from each database to simulate the occupancy of each day. Figure 4.19 shows 9 occupancy days/samples for Family/Users 1.

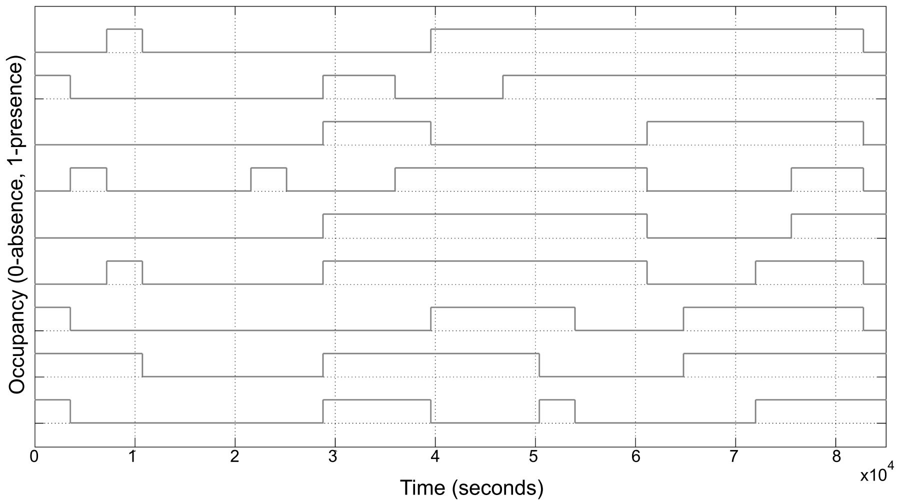


Figure 4.19: 9 Occupancy days for Family 1.

By means of simulation performance indexes, the methods are benchmarked. The utilised indexes are: Consumption (Q), Temperature difference (dT), Time in Comfort (TiC) and Time to Comfort (TtC); explanations concerning indexes are in Appendix B.1.5.

4.5.3 Results & Discussion

Tables 4.10, 4.11, 4.12, 4.13 and 4.14 present simulation results. They show that, as long as only energy savings are valued, the On/off strategy performs best, with the profile based strategy still being within reach. The explanation for the extreme energy savings is the fact that heating is only turned on if a presence is detected. Nevertheless, it can also be observed that comfort related indexes (dT, TiC_i and TtC_i) of the On/off strategy for the given test cases are quite bad and, thus, it must be concluded that users do not feel fully comfortable in these situations. In contrast to that, the strategy based on profiles exhibits the best performance when focusing on the comfort indexes.

	Prof.	On/off	Sched.	Comb.	Best
Q (Kwh)	1.38	**1.30**	1.50	1.35	**Low**
dT ($°C$)	**0.15**	0.25	0.56	0.19	**Low**
TiC (hours)	178.2	168.4	**178.5**	171.3	**High**
TtC (hours)	19.8	29.7	**19.5**	26.7	**Low**

Q ... consumption
dT ... difference between real and desired temperature
TiC ... time while the system keeps comfort conditions
TtC ... time needed to reach comfort conditions

Prof. ... strategy based on profiles
On/off ... on/off controller
Sched. ... scheduled controller
Comb ... combined controller

Table 4.10: Family 1 results.

Obviously, the On/off strategy will always receive the best indexes in energy savings (except maybe for very extreme weather conditions), being unsatisfactory for comfort. The strategy based on profiles shows the best commitment between energy costs and comfort. The combined strategy obtains acceptable performances with low complexity algorithms. The prevalence of strategies based on profiles over the combined strategy (and others) is additionally empowered by the compatibility with holistic control approaches and the developed context awareness capabilities. Additional simulations – for the clustering comparison (Section 6.3) and for the habit regularity analysis (Section 6.4) – show again how strategies based on profiles outperform classic options.

	Prof.	On/off	Sched.	Comb.	Best
Q (Kwh)	1.30	**1.22**	1.44	1.31	**Low**
dT ($°C$)	**0.13**	0.22	0.27	0.14	**Low**
TiC (hours)	**180.7**	169.1	178.6	176.7	**High**
TtC (hours)	**15.2**	26.9	17.4	19.3	**Low**

Table 4.11: Family 2 results.

	Prof.	On/off	Sched.	Comb.	Best
Q (Kwh)	1.30	**1.13**	1.47	1.25	**Low**
dT ($°C$)	**0.10**	0.25	0.32	0.17	**Low**
TiC (hours)	**127.8**	111.0	126.8	117.8	**High**
TtC (hours)	**11.3**	28.0	12.2	21.3	**Low**

Table 4.12: Family 3 results.

Details of Profile Based Control

Figure 4.20 presents some characteristics of the strategy based on profiles. Detail 1 shows a non-desirable (yet possible) situation, in which the smart system does not expect people coming home

	Prof.	On/off	Sched.	Comb.	Best
Q (Kwh)	1.35	**1.27**	1.47	1.32	**Low**
dT ($^{\circ}C$)	**0.11**	0.22	0.63	0.17	**Low**
TiC (hours)	**169.9**	156.8	161.4	161.7	**High**
TtC (hours)	**15.1**	28.2	23.6	23.3	**Low**

Table 4.13: Family 4 results.

	Prof.	On/off	Sched.	Comb.	Best
Q (Kwh)	1.22	**1.03**	1.46	1.21	**Low**
dT ($^{\circ}C$)	**0.13**	0.31	0.82	0.17	**Low**
TiC (hours)	**94.9**	76.9	88.3	88.8	**High**
TtC (hours)	**16.1**	34.1	22.8	22.2	**Low**

Table 4.14: Family 5 results.

for a long time and decides to switch off the heating. However, there is an unexpected occupancy and the system has to switch on again to reach comfort temperature as soon as possible. In this case, the strategy shows the same behaviour as the On/off Strategy. Details 2 and 3 mark the main advantage of the profile based strategy: the profile is a powerful tool to (correctly) predict the next occupancy; thus the system adjusts temperatures before people arrive to the space under control. Detail 3 shows this fact as well as how the comfort temperature drops due to the profile (or a manual change in the setpoint temperature). In addition, it changes to setback when the dwelling becomes unoccupied, but recovers comfort values as soon as users return.

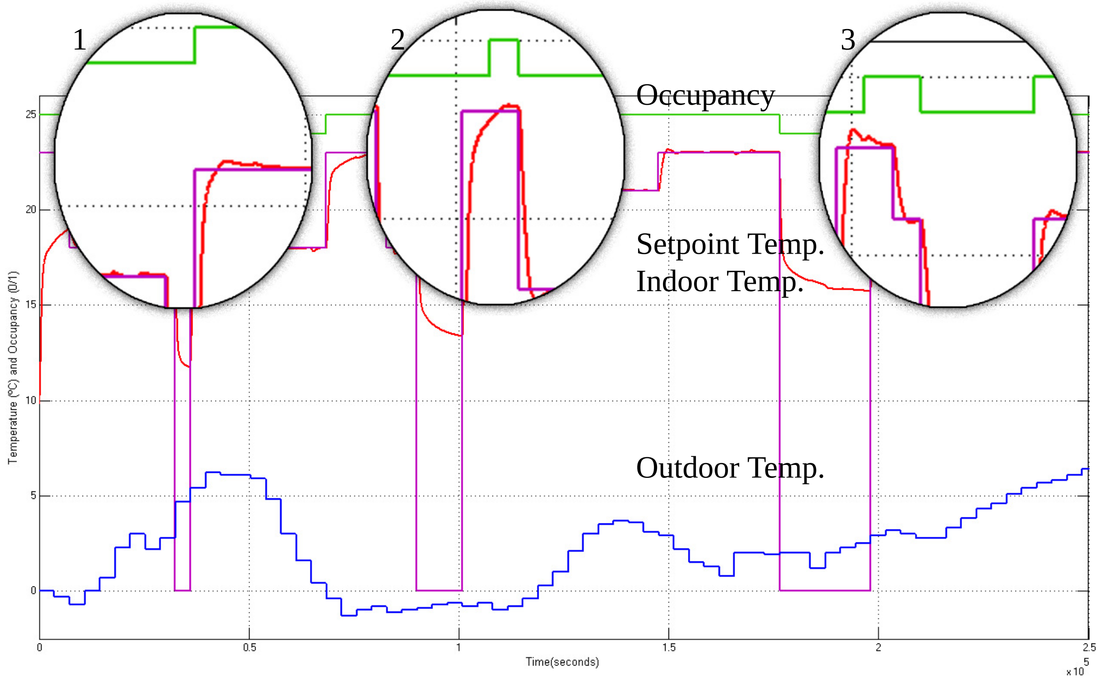


Figure 4.20: Details of the profile-based strategy performance.

5 Pattern Discovery in Smart Homes

This chapter deals with unsupervised algorithms and methodologies intended to establish groups and discover patterns within huge amounts of data (i. e. clustering), specifically analysing the implications derived from using them to obtain habit profiles for smart home applications. After an extensive discussion related to inherent uncertainties, selection of methods, parameters, derived problems and possible configurations, the deployment of clustering techniques to provide information about the context is studied.

5.1 Modeling Human Behaviour in the Home Environment

The research and applications concerning habit pattern discovery in the home and building environment manage some uncertainties which usually add difficulty to the achievement of optimised results. This section introduces some of the aspects related to the habit discovery as well as some uncertainties and difficulties.

The main goal of the whole chapter is to acquire knowledge for designing accurate applications and controllers based on habit profiles. In addition, we aim to understand the existing limitations, and to be able to anticipate them as well as to cope with them.

5.1.1 Dealing with Imprecision and Partial Truth

The first problem to be faced is related to the goals of the profiling obtaining techniques. Regardless of any specific object, in our case of interest, profiles aspire to capture *human behaviours*. This is not a trivial task. Concerning human behaviour, absolute statements can hardly be established and we are inevitably submitted to subjectivity. As it is observed by Frias-Martinez et al. in [FMMCM05],

> Traditional machine learning techniques have some limitations for modelling human behaviour, mainly the lack of any reference to the inherent uncertainty that human decision-making has.

Reviewing the literature, we can expect that uncertainties in modelling human behaviours can be solved by Soft Computing techniques (SC), a term that principally refers to fuzzy logic, artificial neural networks and evolutionary computation. In [DO01], where several industrial applications that use SC are shown, the authors define SC with the following words:

> Soft computing (SC) is an evolving collection of methodologies, which aims to exploit tolerance for imprecision, uncertainty, and partial truth to achieve robustness, tractability, and low cost. SC provides an attractive opportunity to represent the ambiguity in human thinking with real life uncertainty.

The usage of such techniques for classification and feature selection are well-known [Saa08], and their importance has also been emphasized for home appliance technologies [OG01].

5.1.2 Types of Aimed *Behaviours*

In Section 2.3 we saw that, for smart homes and buildings, modelling the human behaviour does not only refer to capturing repeated acts of one particular user, but also to achieve key information that models groups of users or even entire localities (districts, towns, cities, countries, etc.). At the same level of discussion, we aim representing the usage of objects and elements by a specific user or group of users throughout time, but also for a generic user that becomes a model for the whole population. These nuances must be carefully considered when habit discovery methodologies are planned and designed. *Behavioural models* stand for distinct purposes and the obtaining process has decisive implications in the final performance.

Therefore, we keep in mind the two cases we introduced in Section 2.3, they are:

1. Profiles that represent the normal behaviour or usage of a particular user or group of users. These profiles will be mainly deployed by smart home controllers.

2. Profiles that represent habits of populations or localities. The main purpose of such profiles is to be utilised out of (or in addition to) the scope of smart home control, i. e. building simulation and design, energy prediction, simulation tools, urban planning development, etc.

According to the topic of this dissertation, we mostly focus on the first case, without forgetting the second kind of application due to the obvious connections.

5.1.3 Overview of the Habit Abstraction Difficulties

Irrespective of which of the two cases we are dealing with, we require a machine or tool which automatically carries out pattern discovery with regard to human behaviours. It is an *unsupervised task*, and we find that there are even problems to define a priori any *identification criterion*. On one hand, it means that there is no class labelling for the training samples; in other words, we do not start with models that help us to identify the population. Indeed, we have huge amounts of samples described by features quantitatively measured and we hope that advanced tools will smartly find out relationships, similarities and labels; being able to group and differentiate samples, propose representatives and mark those features that are important and state *why* they are important. Furthermore, we are not even completely sure about how to define the criteria that our tool must use to group and differentiate.

Figure 5.1 is sketched to provide a quick and intuitive idea of the problem. If the mentioned tool works with 'water consumption profiles' (say like the ones on the figure), it should quantitatively measure and know if 'sample 2' is more similar to 'sample 1' than 'sample 3'. This task is even difficult for the human eye, and we could agree that is a subjective question with no *absolute* answer. However, the criteria adapted by the smart tool are bound to be decisive in the controller's decision making (i. e. it has an effect on the final energy performance and on the degree of users' satisfaction).

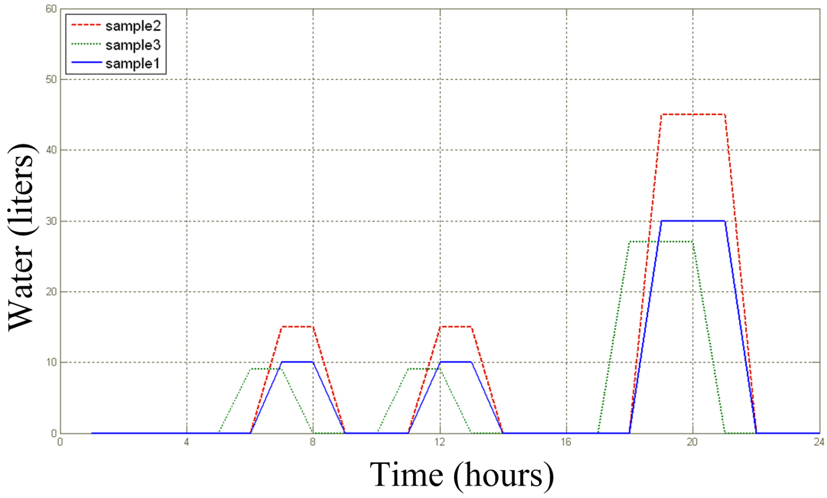


Figure 5.1: In this example with three profiles of water consumption, which sample is more similar to 'sample 1'? 'sample 2' or 'sample 3'? Note that 33% of 'sample 2' does not fit 'sample 1', whereas 38% of 'sample 3' does not match 'sample 1'. What is the correct metric? Is simultaneity or magnitude more important?

SC techniques do not solve these types of problems by the simple application of methods. The suitability of SC lies on the fact that they offer alternative approaches to problems that do not adjust to linear solutions, require non-restrictive assignments or demand more parallel processes and exploration in order to find valid solutions. But even SC tools need adequate parametrisation and a good problem description if we expect satisfactory results. It means that, for example in the case of Figure 5.1, SC also requires the previous and human assisted definition of *criteria* for the existing *objective functions*. In short, we can say that SC does not transform uncertainty into certainty; they are just not so limited or disqualified by uncertainties in their capabilities to achieve good solutions.

For the human behavioural modelling, the detection and understanding of accompanying uncertainties is mandatory if we want to reach acceptable solutions. It allows us besieging the problem until we correctly locate uncertainties, calculate and measure the effect, as well as to obtain a measure of the level of reliability of the obtained results.

5.1.4 The Application Frame: Requirements and Constraints

We link the word *application* to the special use or purpose to which abstracted habits – the outputs of habit discovery tools – are put. It covers why, for what, how, where and under which specific boundary conditions they are utilised. Chapters 3 and 4 deal with these issues for the current proposal of smart home global control. The application influences all the process from the beginning to the end, imposing annoying constraints and conditions, but also framing the requirements of the scenario. It means helping to fix parameters, the feature selection and also solving some uncertainties. In other words, the application gives a tangible sense to the goals of the deployed intermediate algorithms.

Looking back to the problem sketched in Figure 5.1, we can reasonably believe that comparing performances of simulations or directly focusing on the final application can help us to decide which *similarity* criteria must be applied. Of course, the final application offers a valuable guidance to retroactively select tools, methods and criteria, but it is not a completely trustworthy technique to validate such selections (note that we are moving away from *idealistic* interpretations of *similarity* and getting closer to more *pragmatic* perspectives). Here we must face new challenges that somehow can even delegitimise the achieved confidence. Some of them (staying with our fields of interest) are:

- *Keeping reproducibility and repeatability.* Using real applications to compare input parameters or models entails unavoidable problems to reproduce experiments with boundary conditions absolutely under control.

- *Inherent distortion of simulations.* Simulated scenarios involve differences compared with real cases that can notably affect the comparison. The effect of such differences is usually unknown or impossible to calculate.

- *Inherent distortion of applications.* Applications introduce bias and have their preferences because of their inherent features and data transformations. For instance, it is even possible that accurate models present worse performances than rough models due to the fact that the application does not react properly in front of such accuracy.

- *Specificity, or difficult generalisation.* The assessments obtained by the application/simulation tests are not necessarily valid outside the margins fixed by the data used for checking. On the other hand, if we trust in real future data to check models, it is possible that a 'bad model' in training matches the future more accurately than a 'good model'. Therefore, the transportability of the models is in question. Furthermore, if models are to be used in diverse applications, objections and doubts increase.

- *Relative performance: a valid solution, no guarantee of the optimum.* Different criteria and methods offer distinct solutions that are not always comparable or easy to compare, even for the final application. For example, it can be difficult to decide whether a performance is better: two models with high representativeness and normal reliability, or three models with a more reduced representativeness but with a high reliability.

- *Solution oriented, not knowledge oriented.* Only using this kind of test usually does not usually clarify why certain models, methods and criteria are better than others, thus acquiring new knowledge is difficult.

In part, these points are some of the characteristics inherent to the *trial and error* method of problem solving or knowledge obtaining [Hum95]. The introduced drawbacks empower and justify the study of the properties of methods and criteria irrespective of the application purposes. In any case, the application, the study of requirements and the nature of the problem, as well as tests, comparisons and simulations, will allow us to discriminate and find the features we specifically need, as well as to decide among methods, parametrisation and criteria. The nature of the whole problem demands the joint reading of both sides, namely *theoretical* vs *practical*, or *mathematical* vs *engineering* approaches.

5.2 Clustering for Pattern Discovery

The most suitable searching techniques for pattern discovery are represented by methods for *clustering analysis*. Deep down, what clustering algorithms really do is arrange objects into clusters according to the features of the input data, so they state significant groups that are inherent to the given community. From a mathematical point of view, every one of these objects takes up a place in a multidimensional space that is created by their common features. Therefore, a group or cluster can be defined as a dense region of objects or elements surrounded by a region of low density [PCCS07]. From here, a consequent step is to consider that any output group can be represented by a model individual (existent or nonexistent), which usually will correspond to the gravity center of the respective cluster, i.e. a *discovered pattern*.

Note that so far we have distinguished between *profiles* and *patterns*, adducing that *a pattern* is the representation of a group of similar *profiles*. The literature related to clustering usually names it as patterns to the input objects that must be clustered. In any case, we will continue with the terminology used up to now, which is related to our application models, i. e. we will differentiate between profiles (input objects or elements, samples, vectors, input data) and patterns (output models, centroids, representatives).

5.2.1 Applications and Steps of Clustering

We can summarise clustering as a useful data-mining technique to find data segmentation and pattern information. The habitual clustering applications cover the following aspects [TK03]:

- *Data reduction.* In situations where the amount of available data is too large for a successful processing and some compression or summarisation is required.
- *Hypothesis generation.* Clustering is used to suggest hypothesis and infer knowledge regarding the nature of the input data.
- *Hypothesis testing.* In this context, clustering is utilised to the validation or verification of hypothesis.
- *Prediction based on groups.* In some applications, the response or behaviour of a new element in front of a specific situation can be predicted according to the response previously given in the same situation by other members of the same cluster.

The smart home applications introduced in this thesis mainly use profiles for predictive control (see Section 5.1.1, case '1'). It means *prediction based on groups* (but also *data reduction*). Controllers try to guess the profile that is going to happen the next day based on the experience accumulated in the past.

For the applications referred to in Section 5.1.1, case '2', i. e. building and community calculations, we are dealing with a *data reduction* case, where it is expected to represent most of a certain population by means of a model profile (a pattern).

In both application cases, the *hypothesis generation* is an additional outcome of the clustering use. It is a relevant aspect in the investigation or design phase as it provides valuable knowledge (feedback) for the progressive tuning and correct adjustment in every one of the distinct steps. In addition, *hypothesis generation* is very important for smart home control from an overall perspective, not in vain, it is an alternative way of naming *context awareness*. In Section 5.4, how clustering is deployed to achieve *readings* of the home context is widely explained.

In [TK03] the basic steps that an expert must follow in order to develop a clustering task are listed. We mention them again in connection with services based on profiles for smart homes:

- *Feature selection*. Features must be carefully selected based on meaningfulness and avoiding redundancy. In our case, it is strongly determined by the application design. This issue is widely dealt with in Section 5.2.2.
- *Proximity measure*. It consists of the *similarity* or *resemblance* criteria utilised to compare two singular input vectors or samples (see Section 5.2.3).
- *Clustering criterion*. The expert must decide which kind of clusters (shape) he/she is looking for (compact, elongated, hyper-spherical, etc.), as well as how to interpret phenomena like density, or decide the *importance* of cluster features such as big, small or crowded; as well as how to deal with outliers, noise, etc (see Section 5.2.5). Furthermore, he/she must also know how many clusters the solution requires: few, many, a fixed number, or the optimal number within a range (Section 5.2.4).
- *Clustering algorithms*. The selection of the specific clustering algorithm schema, i. e. the way of clustering data (Section 5.2.6).
- *Validation of the results*. A good practice requires that experts verify the validity, correctness or suitability of clustering results (Section 5.2.7).
- *Interpretation of results*. It is the extension to the validation of results, but in keeping with other experimental evidence and analysis, i. e. tests in the application field, and the later collection of feedback and derived knowledge in order to ameliorate the decisions made in the whole clustering design process and even in the related phases and structures where the clustering process has been integrated (also Section 5.2.7 and Section 5.4).

5.2.2 Feature Selection, Data Transformation and Vector Shape

Feature selection and data transformation are aspects to carefully consider for modelling users' behaviours. They mainly answer the following questions: *what data must be used to represent users' behaviours?* and *what transformations must be carried out before raw data is presented to* *the clustering tools?* The objectives are the maximal exploitation of the clustering potential and the clearest and most significant collection of input information.

Scenario/Application Dependence

As we have referred to above, the feature selection strongly depends on the application scenario. Therefore, the pragmatic question is not *what features better represent human behaviour?*, but *in which manner do I need behavioural data to accomplish the requirements of the intended use case?* Hence, the dominant factor is the application objectives or purposes, which must form the basis in order to design the whole process where clustering tools appear for habit abstraction. The feature selection also depends on technological limitations, so we are forced to be realistic and use data that can be normally collected from the home environment without entailing high expenses or unpleasant implementations.

For example, in Chapter 4 we saw that the Setpoint Temperature Control Application requires behavioural information concerning occupancy habits and comfort temperature habits. Chapter 2 offers a broad explanation about how behavioural information can be stored and dealt with in order to use it for control applications as well as be easily processed by intelligent algorithms. The exposed way of managing behavioural information is obviously not unique, but based on concepts for smart home global control introduced in Chapter 3. Hence, within this context and for the commented case, we have devised profiles separately for habitual comfort temperatures and for occupancy, both related to time in a daily scope. This arrangement fits the application according to specific design foundations and guidelines, it does not mean that they have to be the best or unique way to represent behavioural habits.

In a more general scope, this aspect related to the data representation is smartly introduced in [JMF99]:

> Because of the difficulties surrounding pattern representation, it is conveniently assumed that the pattern representation is available prior to clustering. Nonetheless, a careful investigation of the available features and any available transformations (even simple ones) can yield significantly improved clustering results. A good pattern representation can often yield a simple and easily understood clustering; a poor pattern representation may yield a complex clustering whose true structure is difficult or impossible to discern.

As we have warned in previous sections, note that in the quote, a *pattern* is equivalent to what we call a *profile* (for us the *minimal amount of information that is able to represent a habit for the selected application*). Therefore, the *pattern representation* concerns itself with the *shape* or *structure* of these vectors, profiles or patterns, and their fields, scopes, elements and dimensions.

Aspects of the Feature Selection in a Specific Case

Once we have accepted the constraints imposed by the *application purposes* and the *technological limitations* as a part of the problem description, we can face the feature selection and the design of the profile structure. In Chapter 2 we introduced a methodology for the behavioural collection that is, to a large extent, justified by the idea of facilitating the work for the subsequent clustering algorithm that has to manage such information. We are not going to expose the same explanations again, but let us show some of the implications in the specific example of the Setpoint Temperature Control Application.

For this application the original design sets out the optimisation of thermal comfort by means of the smart management of setpoint temperatures and deploying habit information regarding occupancy and comfort temperatures (that is stated by the *application purposes*). For most tests and experiments, we have utilised the feature selection represented by 'Option A' in Figure 5.2, but here we can imagine other different sample representations, like 'Option B' and 'Option C'.

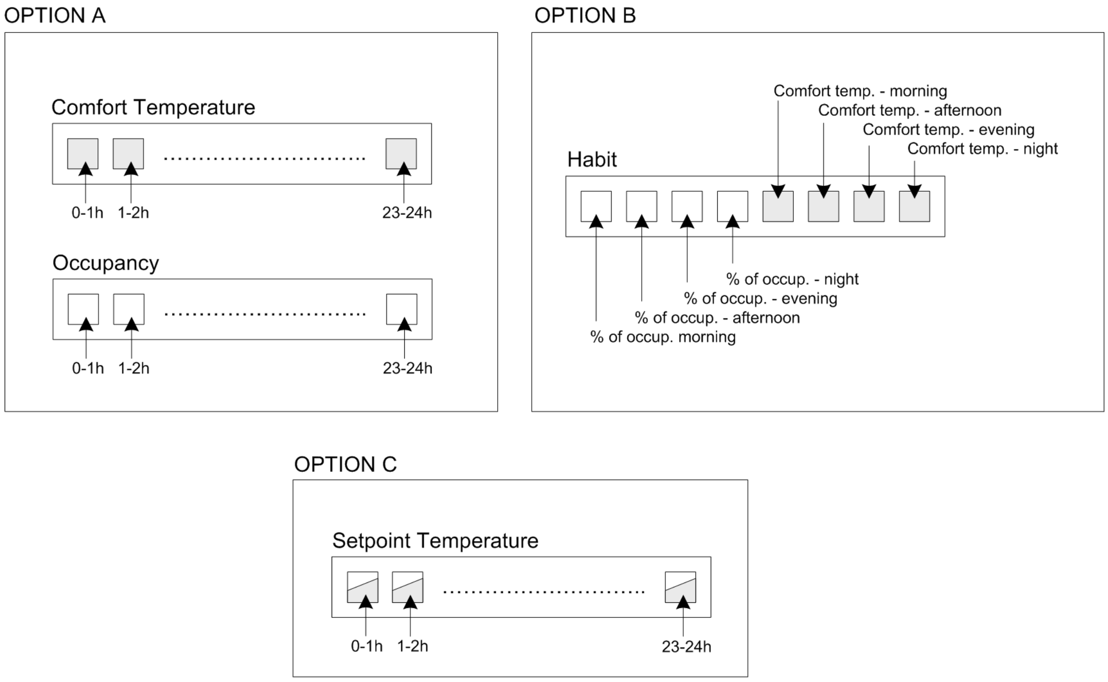


Figure 5.2: Different vector representations for the Setpoint Temperature Control Application habit profiles.

Maybe the first noticeable difference is the fact that with 'Option A' we are using two separate habit structures whereas in 'Option B' we are considering comfort and occupancy together in the same profile, or in 'Option C' we directly deal with setpoint temperature values. Let us suppose that a priori the three options are equally valid for the application and, if the habit is correctly abstracted, there would not be any important difference in the final performance irrespective of which option is used. Similarly, let us consider the differences in the time scope as irrelevant or negligible. Therefore, we focus on the impact that the three habit representations have for the subsequent clustering phase. It is worth remembering here that the application deploys clustering to find out representatives or models and not merely to define clusters or groups.

- *Mixing or separating phenomena in the same vector structure.*
  The main drawback of separating comfort temperature and occupancy is that we lose possible bonds and correlations between the two phenomena. Either we assume that such bonds are not relevant or that they are going to be dealt otherwise, beyond the concerns of the clustering tool. Indeed, in the present case, profiles have been designed in a way that phenomena are isolated as much as possible in order to facilitate the profile and context

interpretation by controllers. That is the reason for why we do not work directly with *setpoint temperature* profiles (like in 'Option C'), in which comfort and occupancy could become mixed depending on the final HVAC system. The more heterogeneous the profiles are (mixing different phenomena in the same field, 'Option C'; either in the data structure itself, 'Option B'), the more difficult the interpretation of clustering outputs will be, including knowing if the clustering algorithms are working properly with the data. On the other hand, some applications can force us to manage pattern discovery within individuals characterised by different phenomena and scale types.

- *Coherence and correlation among fields/features.*
On the other hand, fields in 'Option A' and 'Option C' are ordered depending on time. This is a relevant feature for the similarity assessments. In 'Option B' the arrangement depending on time is broken (or partially broken). In such a case, we are rejecting the possibility of measuring relationships/dependencies between near fields in comparison to far fields; in other words, in the 'Option B' case each field is a dimension equidistant to the rest. 'Option A' and 'Option C' are open to similarity definitions that consider the position of data features within the vector structure. Again, the convenience of such aspects depends on the final application; in our case, according to the designs introduced in Chapter 2, the capability of finding relationships among close fields seems to make sense for the developed smart home services.

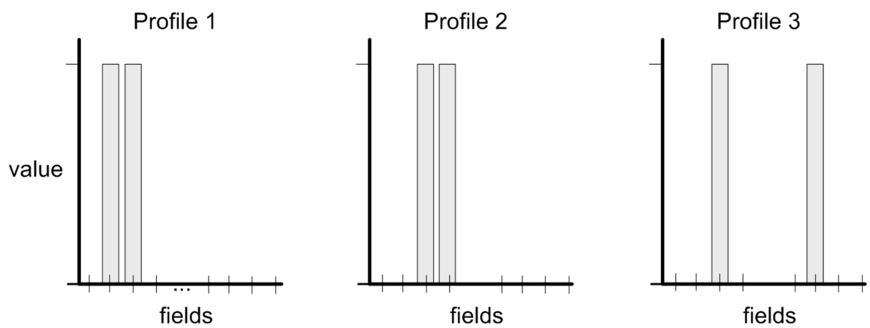


Figure 5.3: Some examples of generic binary profiles.

The previous explanation can be better understood with Figure 5.3. It is expected that an appropriate clustering tool find 'Profile 2' closer to 'Profile 1' than 'Profile 3' in the 'Option A' and 'Option C' cases. For the 'Option B' the position of the distinct fields is in principle irrelevant, so for the clustering tool, with no other information available, the three profiles are equidistant from one another.

- *Global normalisation or feature by feature.*
A mandatory step for clustering is the adaptation of raw collected data to subsequent phases, it means a pre-processing step that prepares data to be correctly dealt by abstraction algorithms. Some of these aspects consist of detecting and filtering wrong data, out-of-range values, incoherences and impossible combinations, missing values, etc. [Pyl99], but also the normalisation of input data required by clustering algorithms. Here, every field or feature of the profile represents a different dimension of a multi-dimensional solution space. In this space individuals are placed as points and the clustering machine states the cluster boundaries that will fix the classification. The normalisation process introduces a

deformation in the space that can be suitable and imperative to find a correct solution but, if it is not carefully carried out, it can also be counterproductive in some scenarios.

Quoting de Souto et al. in [SAC+08]:

> Also, in many practical situations a dataset could present patterns whose attributes or features values lie within different dynamic ranges. In this case, for proximity indexes such as Euclidean distance, features with large values will have a larger influence than those with small values. However, this not *necessarily* will reflect their importance for defining the clusters.

The *italics* is ours, just to stress that the opposite case – i.e. *features with large values will have a larger influence than those with small values* – could also be suitable for some cases.

A standardised process of normalisation is using *mean and standard deviation* for each field or feature of the pattern (for a method that supposes data following normal distributions, see Section 5.2.5). Also sigmoid or logarithmic functions can be used to emphasise data within specific intervals. Indeed, using a linear normalisation method like mean and standard deviation is stating a dynamic equivalence among features; i.e. the value variability of each field has the same relative importance for the final cluster solution. In other words, if our vectors have 10 features or fields, the normalisation method is assuming a priori that the 10 fields are equally relevant to the final classification. This can be suitable for a high amount of cases, but it can also entail undesirable or unnecessary distortions in situations where a previous bond exist or a relationship of proportionality among features that must be considered for the classification.

For example, in Figure 5.1, where profiles present 24 fields or features corresponding to water consumption (in liters) during consecutive hours, it can be preferable to apply the same normalisation for all the fields, considering the mean and standard deviation of water consumption per hour of all fields collected together. Thus the drawing superimposed on the features (say *silhouette* or *envelope*) is not distorted and can be considered in the clustering process.

Here we have seen another reason to prefer 'Option A' in Figure 5.2 before 'Option B', as we do not have to deal with the relationship of weights among different features. Therefore, features that show a variable dependence upon each other are not mixed and we are not forced to take unknown assumptions by default in this respect.

- *Feature reduction.*
  Finally, the time-scope reduction observed in 'Option B' in comparison with 'Option A' and 'Option C' makes sense, provided we ensure that relevant information is not being lost with such compression, and assuming that the possible distortion is derived from the data-reduction processes. It is worth remembering here that *data reduction* is one of the main purposes for the application of clustering. In addition to obvious computational and time-demanding reasons, *dimensionality reduction* is strongly recommended for clustering to avoid *curse of dimensionality* [ZR99] and *peaking phenomenon* [SD08] problems.

Time Series Clustering

The reasons exposed above and the design of services and controllers depicted in Chapters 3 and 4 determine together the final shape of the clustering vectors. Therefore, they are shaped as *time* *series profiles* (already introduced in Chapter 2). Within the vast world of clustering, they have their own field, named *raw-data-based univariate time series clustering.*

Thus, the difference between time series and normal clustering is that, in the time series case, the shape of input vectors entails features that are arranged in time, hence in the univariate time series an input vector is usually the succession of values that a certain variable takes throughout a specific time scope.

Clustering time series is usually carried out twofold: a) either *feature-based* or *model-based*, i. e. previously summarising or transforming raw data by means of feature extraction or parametric models (e. g. ARIMA), so the problem is moved to a space where clustering works more easily; b) or *raw-data-based*, where clustering is directly applied over time series vectors without any space-transformation previous to the clustering phase. Several works concerning each kind of time series clustering are referred to in detail in [War05].

Beyond the obvious loss of information due to *feature-based* or *model-based* techniques, they can also present additional drawbacks; for instance, the application-dependence of the feature selection, or problems associated to the parametric models [GP01]. On the other hand, *raw-data-based* can entail working with high-dimensional spaces (*curse of dimensionality*) and it is normally sensitive to noisy input data.

In addition, using the *raw-data-based* approach shows further advantages. For instance, outcomes are more transportable, i. e. conclusions and hypothesis can be more easily generalised for other behaviour modelling applications (e. g. individual or community profiles for energy, occupancy, comfort temperature). Moreover, it is the option that more clearly and directly allows the analysis of correlated data in clustering.

5.2.3 Concept of Similarity

We consider *similarity* as the measure that establishes an absolute value of resemblance between two vectors, in principle isolated from the rest of the vectors and without assessing the location inside the solution space.

Considering continuous features, the most common metric is the Euclidean distance

$$d_E(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})(\mathbf{x} - \mathbf{y})'} \tag{5.1}$$

which works well with data distributed in compact, hyper-spherical, isolated clusters. There are situations where Euclidean distances are not the most suitable; for instance, when we expect a different handling of the dependencies and relationships among features. Such situations can sometimes be faced from the normalisation phase, but not always [MJ96]. Note that Euclidean distance is invariant when dealing with changes in the order that time fields/features is presented; it means that it is (in principle) blind to capture vector or feature correlation. Figure 5.4 clarifies this discussion with an example.

For time series data comparison, where trends and evolutions are intended to be evaluated, or when the shape formed by the ordered succession of features (*silhouette* or *envelope*) is relevant, similarity measures based on Pearson's correlation

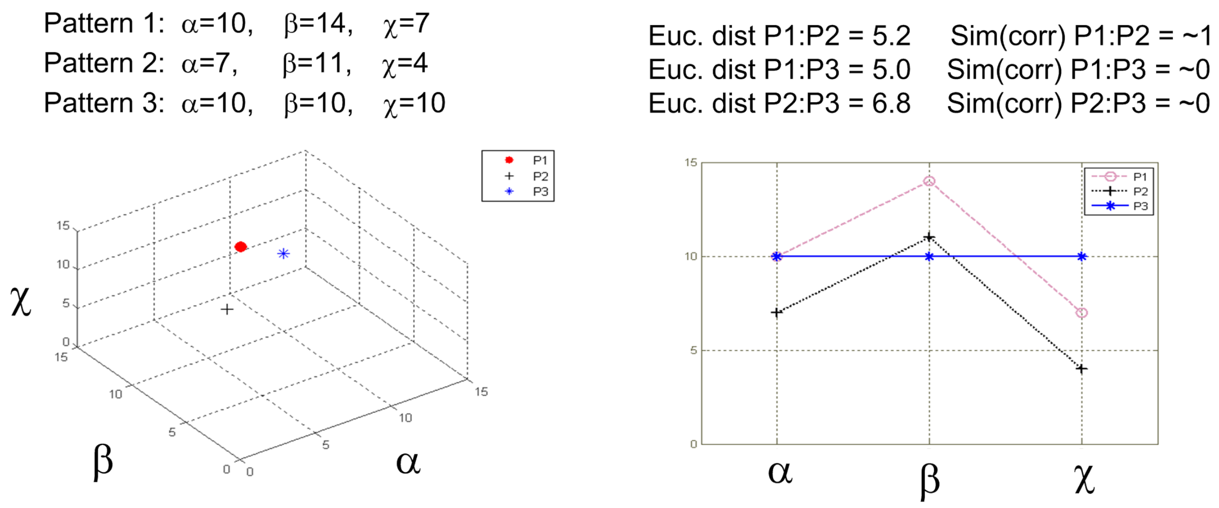


Figure 5.4: Example where Euclidean distance and correlation have been deployed as basis for similarity functions. α, β und γ are the three (time-arranged) features of the shown patterns. Note that P1 is more similar to P3 according to Euclidean distances, but more alike P2 according to correlation measures.

$$d_C(\mathbf{x},\mathbf{y}) = 1 - \frac{(\mathbf{x}-\bar{\mathbf{x}})(\mathbf{y}-\bar{\mathbf{y}})'}{\sqrt{(\mathbf{x}-\bar{\mathbf{x}})(\mathbf{x}-\bar{\mathbf{x}})'}\sqrt{(\mathbf{y}-\bar{\mathbf{y}})(\mathbf{y}-\bar{\mathbf{y}})'}} \tag{5.2}$$

have been also widely utilised, although it is not free of distortions or problems [RN88]. Mahalanobis distance

$$d_M(\mathbf{x},\mathbf{y}) = \sqrt{(\mathbf{x}-\mathbf{y})C^{-1}(\mathbf{x}-\mathbf{y})'} \tag{5.3}$$

can be seen as an evolution of the Euclidean distance that takes into account data correlation. It utilizes the covariance matrix of input vectors C for weighting the features. Mahalanobis distance usually performs successfully with large data sets wits reduced features, otherwise undesirable redundancies tend to distort the results [MJRM00].

An interesting measure specially addressed to time series comparison is the Dynamic Time Warping (DTW) distance [Keo02]. This measure allows a non-linear mapping of two vectors by minimising the distance between them. It can be used for vectors of different lengths: $\mathbf{x} = x_1, ..., x_i, ..., x_n$ and $\mathbf{y} = y_1, ..., y_j, ..., y_m$. The metric establishes an n-by-m *cost matrix* C which contains the distances (usually Euclidean) between two points x_i and y_j. A warping path $W = w_1, w_2, ..., w_K$, where $\max(m, n) \leq K < m + n - 1$, is formed by a set of matrix components, respecting the next rules:

- Boundary condition: $w_1 = C(1, 1)$ and $w_K = C(n, m)$
- Monotonicity condition: given $w_k = C(a, b)$ and $w_{k-1} = C(a', b')$, $a \geq a'$ and $b \geq b'$.
- Step size condition: given $w_k = C(a, b)$ and $w_{k-1} = C(a', b')$, $a - a' \leq 1$ and $b - b' \leq 1$.

There are many paths that accomplish the introduced conditions; among them, the one that minimises the warping cost is considered the DTW distance:

$$d_W(\mathbf{x}, \mathbf{y}) = \min\left(\sqrt{\sum_{k=1}^{K} w_k}\right) \tag{5.4}$$

The main drawback of the measure remains in the effort dedicated to the calculation of the path of minimal cost, in addition to the fact that, actually, it cannot be considered as a metric, i. e. it does not accomplish the triangular inequality.

Beyond these general-purpose, popular distances, there are many additional similarity measures. A survey of distance metrics for time series clustering can be found in [War05]. Other noteworthy options are: the *cosine measure* [Hua08], good for patterns with different or variable size or length; or Jaccard and Tanimoto similarity measures, that can also be intuitively understood as a combination of Euclidean distances and correlations assessed by means of the inner product [Lip99]; and also combinations based on Minkowski metrics for mixed-scale-type patterns [WM97]. For discrete features, Hamming distances and discrete correlations are usually utilised as metrics.

The performances of distinct similarity options to discover habit profiles in time series clustering is tested in Section 6.1. We will see there that experiments contradict what we could expect according to the plain theory.

5.2.4 Number of Clusters and Representatives

A very relevant uncertainty related to clustering has to do with the number of clusters to be found (for our application cases it also means the number of output patterns or representatives). Most techniques demand this number as an input parameter, a fact that usually requires previous knowledge of the internal structure of the data. As we have already referred to several times, this knowledge usually does not exist on the desired level (perhaps merely foreseen).

The ideal situation claims for a clustering method that explores the input data set and establishes the optimum number of clusters by itself. It can be carried out by running consecutive clustering executions changing parameters (i. e. the number of initial clusters) and comparing the performances afterwards. As we will see regarding validation techniques, objective measures for clustering optimality are not free of problems either. For instance, if we compare performances for finding out the right number of initial clusters, we must consider the following aspects:

- A high increase of computational cost and time cost.
- The quality of the clustering process should not vary due to different numbers of initial clusters (only because of the inherent nature of the scenario structure). Otherwise, for instance, we risk to compare a good performance executed with n initial clusters with a poor performance with m clusters. The system will state that n is the optimal number of a cluster when perhaps a better performance was obviated with m clusters.
- The capability of successfully comparing performances with different numbers of clusters is subdued to certain subjectivity and trade-off solutions.

Again, the final application can force us to determine parameters and validations, establishing an initial number of clusters according to the requirements. But, in any case, it does not mean that the clustering tool must necessarily execute clustering considering as many initial clusters as patterns are required for the application. A better solution could be reached considering more clusters and either removing or merging a posteriori the less significant ones (Figure 5.5).

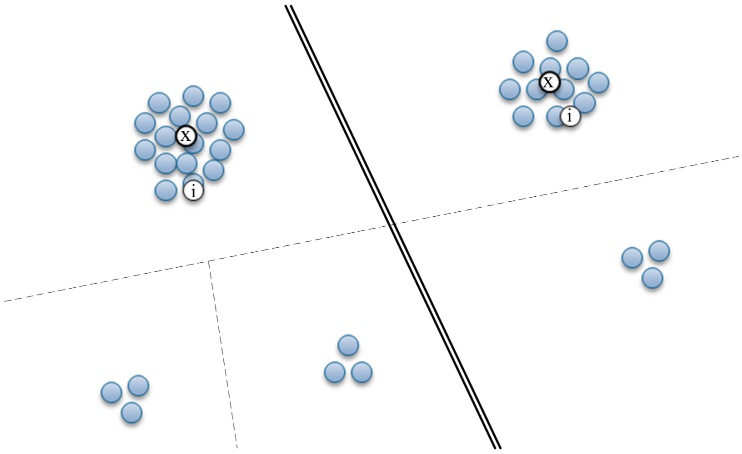


Figure 5.5: Comparison of two performances with a different initial number of clusters ($k = 2$, $k = 5$). Even if the application only requires two representatives (the two crowded clusters), it can be appropriate to execute the algorithm with $k = 5$ and reject the minor clusters afterwards. Otherwise, representatives can be astrayed by the inclusion of elements that clearly pertain to other clusters. 'x' stands for the centroid or representative with $k = 5$, whereas 'i' does it for $k = 2$.

In any case, not all clustering techniques react in the same way managing the number of initial clusters. For example, the typical k-means algorithm establishes a centroid or center for each cluster and initially places them at random or fairly spread in the multi-dimensional problem space [KA04]. Later on, the input samples, as long as they are being processed by the algorithm, move the centers and the boundaries of the clusters following proximity relationships. It is easy to see that solutions obtained are extremely biased by the initial pre-fixed number of clusters. Nevertheless, more advanced methods than direct (sometimes *brutal*) k-means usually carrying out a previous phase (before the direct partitioning) that obtains certain orientation concerning the data set structure. Although they are also dependent on the pre-defined initial number of clusters, they are more prone to form empty clusters if the initial number is excessive (e. g. SOM clustering) or require additional clusters if the initial number is insufficient (e.g. Graph clustering).

For example, standard SOM clustering methods compute a set of reference prototypes representing local means of the data before proceeding to the partitioning. The capabilities of rearranging multi-dimensional input data into a two-dimensional map with a spatial topological-based organisation confer more sensitivity to state empty clusters when they make sense (beyond better local minima resolution, better variable density adaptation, etc.). Methods for the definition of the initial number of clusters in SOM algorithms are depicted in [CB10].

Graph clustering [Sch07a] consists of finding out clusters in a data set that has been previously modelled by a graph. Hence, the algorithm looks for clusters within the graph. Again, the graph offers an early interpretation of the data set structure and is able to consider that the pre-configured number of clusters has been wrongly stated (Figure 5.6). In addition, this fact makes graphs as a resorted method to state the initial number of clusters before applying any clustering technique [SC04].

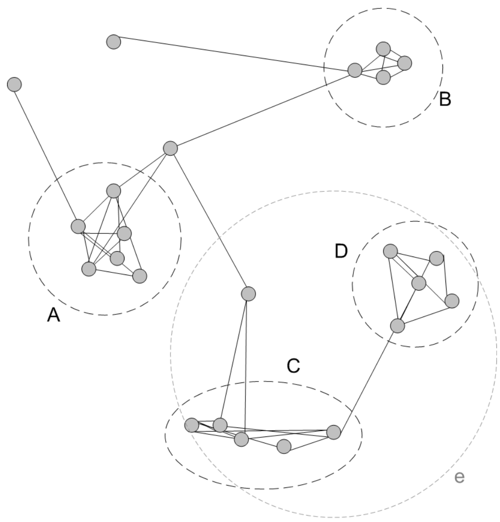


Figure 5.6: The graph representation prefers the clustering solution: 'A', 'B', 'C', 'D' according to the graph drawing logic, even in spite of the fact that the original parametrisation imposed $k = 3$.

In [MC85], 30 procedures for determining the number of clusters are evaluated. In spite of being useful, the study is not conclusive, emphasising again that criteria depend on the data nature. In [JPDD03], a measure of clustering optimality based on the squared error sum as a clustering algorithm process is proposed. In short, this method introduces *clustering gain*, which consists of an evaluation whose optimal performance is achieved when the intra-cluster similarity is maximum and the inter-cluster similarity is minimum. It also stands for sensitivity analysis of other parameters as well as for comparisons among different clustering techniques (see Section 5.2.7).

Clustering gain can be an excellent approach, but it is not free of discussion either. It can present considerable drawbacks (starting with the three points mentioned above), and bad assessments in complicated scenarios are also possible. In the end, the final application must prevail to determine what is and what is not suitable and justified. For instance, contrary to the case exposed in Figure 5.5, even in a case where clusters are so clear such as in Figure 5.7, 'Case A', the final application can work more comfortably with the centroids of a less accurate solution as they would be representing more population, e. g. 'Case B'. As we have referred to before, a good study of the application requirements ends up helping to develop correct designs supported by clustering tools correctly parametrised.

5.2.5 Data Distribution – Outliers

Another problem related to modelling behaviours appears with connection to the sample distribution. To suppose that human behaviour always follows normal distributions is at least premeditated, so it leads to a new aspect submitted to uncertainty: the outlier detection.

An outlying observation, or outlier, is one that appears to deviate markedly from other members of the sample in which it occurs [Gru69]. In statistics, the presence of outliers indicates some kind

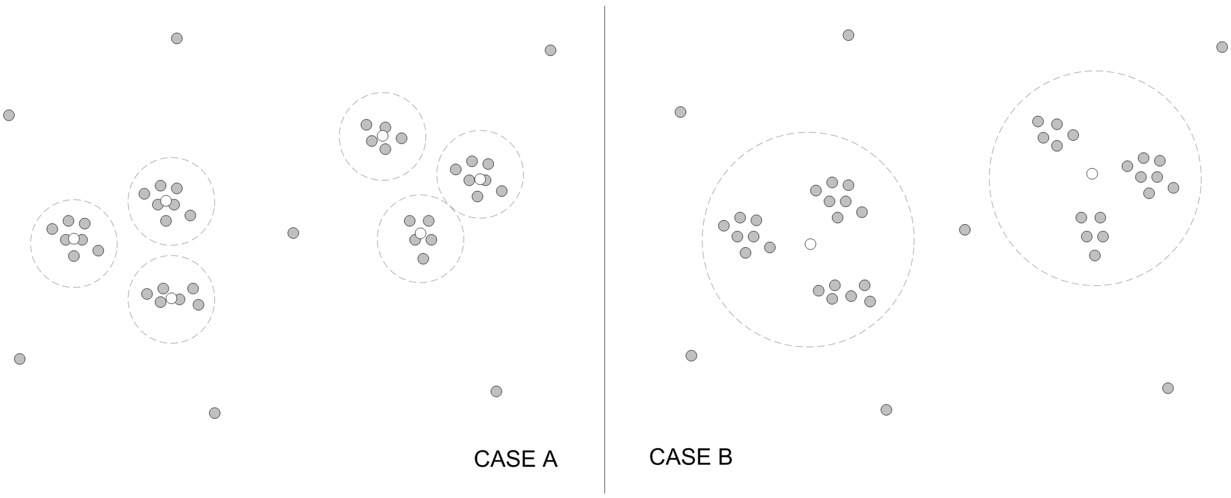


Figure 5.7: Solutions for $k = 2$ and $k = 6$ with outlier rejection. White circles stand for representatives (or centroids).

of problem; it usually refers to a sample which does not fit the model or an error in measurement. But the truth is that there is not an absolute mathematical definition or an ubiquitous method to state whether or not a measured sample is an outlier.

Thus, an outlier has a flexible performance that depends strongly on the scenario, the nature of the samples, the distribution, and what we expect from the classification. Depending on the application goals, it can be desirable to directly reject the outliers or detect them in order to study in depth the outlying cases. The more information we know about the scenario and the data, the better it is to establish whether or not it is an outlier in that particular case. Therefore, for instance, if the assumption that most of the samples follow a normal distribution can be assumed, there are some outlier detection methods that work successfully.

In [LBS10], two outlier detection methods are applied in an energy consumption classification in a case related to buildings. The first method works with the normal distribution assumption, the second one works well in a little set of samples. This is a good example because: 1) it shows the outlier problem in a human behaviour modelling case; 2) the case points out the necessity of modeling and classifying the energy behaviour in buildings; and 3) outliers are seen as phenomena to analyse themselves and not only to reject or filter.

Note that the existence of outliers can offer relevant information about the input data distribution. To say that a sample is an outlier is an assessment usually given after considering the location of individuals in the solution space. In any case, if outliers exist, they must be detected somehow and properly dealt with, otherwise they are prone to lead centroids astray, affect clustering algorithms and distort solutions (Figure 5.8). That is specially important when we use clustering for discovering representatives (i. e. most of the introduced applications).

In any case, a vector can be considered as an outlier when the combined values of its features make it an outlier, but also because of the fact that specifically any or some of its features present a strange distribution that causes the outlier existence. In short, it can reveal the special, unknown behaviour of some of the characteristics, and also discover new knowledge about the impact of specific features or their combinations. In this way of thinking, J. G. Cheng [Che00] emphasises the necessity of analysing and determining the outlier origins, in order to know if outliers are

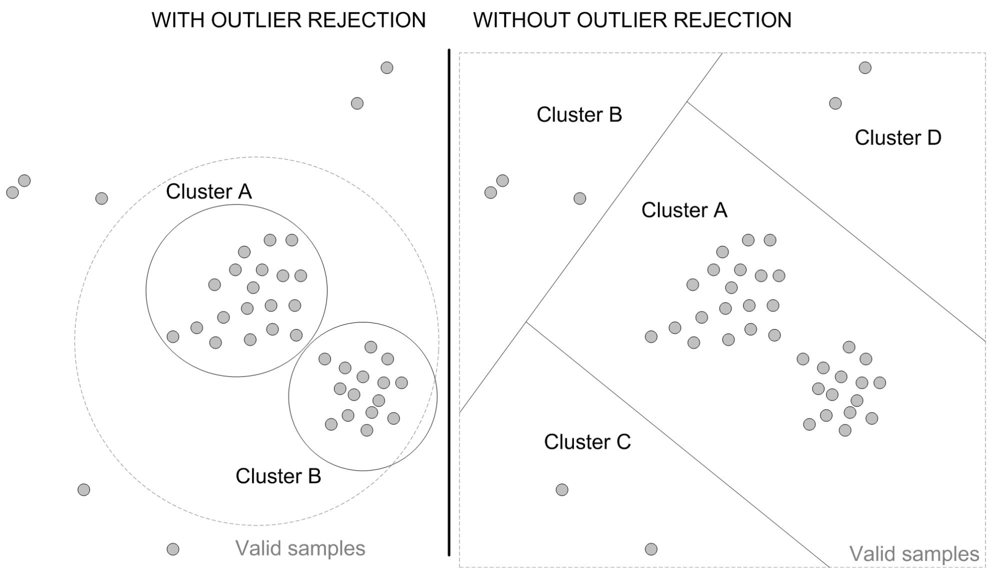


Figure 5.8: Examples of distortion introduced by outliers. Outliers distract the tool and do not allow the core and significant part of the input vectors to be appropriately clustered.

helping us to discover new knowledge ('good outliers') or they are just noise ('bad outliers'). With his words:

> It requires not only an understanding of the mathematical properties of data but also relevant knowledge in the domain context in which the outliers occur.

However, we usually require clustering for hypothesis generation, i. e. to obtain relevant knowledge in the domain context; hence we are suffering circularity again. Analysing the human behaviour in the built environment is a complicated field of research where previous assumptions can barely be done. Besides, many variables take part in the performance of the profiles and part of them are not usually collected, cannot be easily collected or abstracted, or are even hidden or unknown.

According to the profile designs proposed in Chapter 3 for smart homes, outliers existence is not usually a serious problem in cases where the behaviour of a single user or family is abstracted. There are not many features and samples, boundary conditions are fairly controlled, and detecting anomalous behaviours is not a difficult task. Experiments show that for these services the outlier detection (beyond prior filtering) receives a little improvement in the results (Section 6.3 and Section 6.2). However, when the objective is modelling a large group of users (e.g. models for energy building simulation calculations), outliers can easily exist, disturb the clustering tools, and make the results arbitrary or trivial (Section 6.5).

5.2.6 Clustering Techniques & Algorithms

There are many different clustering algorithms and every one of them has a different strategy or methodology to accomplish the task of grouping; it means that, even with the same input parameters and *similarity* criteria, they will probably obtain distinct clusters in the solution.

That is not rare if we consider for a moment the magnitude of the number of different ways to group n input elements in k clusters, it is a Starling number of the second kind [JPDD03]:

$$S_n^{(k)} = \frac{1}{k!}\sum_{i=1}^{k}(-1)^{k-i}\binom{n}{k}i^n \tag{5.5}$$

For instance, only containing 9 elements in the input data set ($n = 9$), there are 7770 ways of partitioning them into 4 ($k = 4$) non-empty subsets.

Considering only this fact and the unsupervised nature of the clustering task, it is easy to guess that trying to find out the best method for a given scenario is not a trivial matter. Different cluster solutions are usually possible and every method is biased or favours certain structures or phenomena related to the data inter- and intra-relationships.

As an example, imagine that in a specific experiment *similarity* is unequivocally fixed as ‘exact physical resemblance’. Later on, two different clustering techniques are forced to establish only two clusters in order to divide a certain population of people. The first method distinguishes two clusters where a smart observer can label the groups as ‘children’ and ‘adults’, whereas the second method states two dominant groups that are better labelled as ‘males’ and ‘females’. In the example, the second method was designed with the capability to consider paths of connection inside the samples (being ’old men’ and ’male babies’ two extremes of the same elongated cluster). It does not necessarily make the second method better than the first one, in this case both solutions seem to be equally valid to classify the population (it is worth stressing again that we are using the same similarity metric in both cases). As a general rule, the same characteristics that make a method useful for a specific scenario are drawbacks for the other situation where such capabilities are not required or do not exist.

Hence, the selection of the suitable clustering method is also an uncertainty to face. In addition, clustering methods usually impose a structure on the data even if it does not exist, so to expect that they provide us with a reliable description about *how data is structured* (in situations where classic statistical analysis are insufficient) is risky. In [TK03], it is smartly exposed:

> We must have an indication that the vectors of X form clusters before we apply a clustering algorithm. (...) Poor estimation of these parameters and inappropriate restrictions on the shape of the clusters (wherever such restrictions are required) may lead to incorrect conclusions about the clustering structure of X.

This is the reason why experts’ intuition and knowledge about the data play a fundamental role in order to select the method and the parametrisation.

It is worth knowing how clustering methods are also *clustered* or differentiated according to their capabilities, the clustering methodology or their features. There exists some definitions and taxonomies, e. g. [TK03], [JMF99], [JD88]; here we combine some of these works and revise some fundamental or principal aspects (always keeping in mind the requirements of the habit pattern discovery for smart homes):

- *Partitional vs. hierarchical algorithms.*
  Partitional or sequential algorithms are fast and straightforward schemes that tend to produce hyper-spherically or hyper-ellipsoidally shaped clusters. They work in a single layer,

hence all clusters are at the same level and there are not nested clusters. On the other hand, in hierarchical algorithms, clusters are organised in levels, so algorithms search for clusters inside clusters and so forth. Hierarchical clustering develops *dendograms*, tree-shaped graphical representation of a clustering solution where groups are progressively stated in pairs of elements and similarity levels are clearly seen.

Hierarchical clustering presents problems to compute dendograms for scenarios with large data sets. In addition, the clustering structure imposed by the dendogram step by step can entail wrong interpretations, sometimes due to the in-two-sets partitioning procedure, but in general, because of certain lack of flexibility to reach overall perspectives (Figure 5.9).

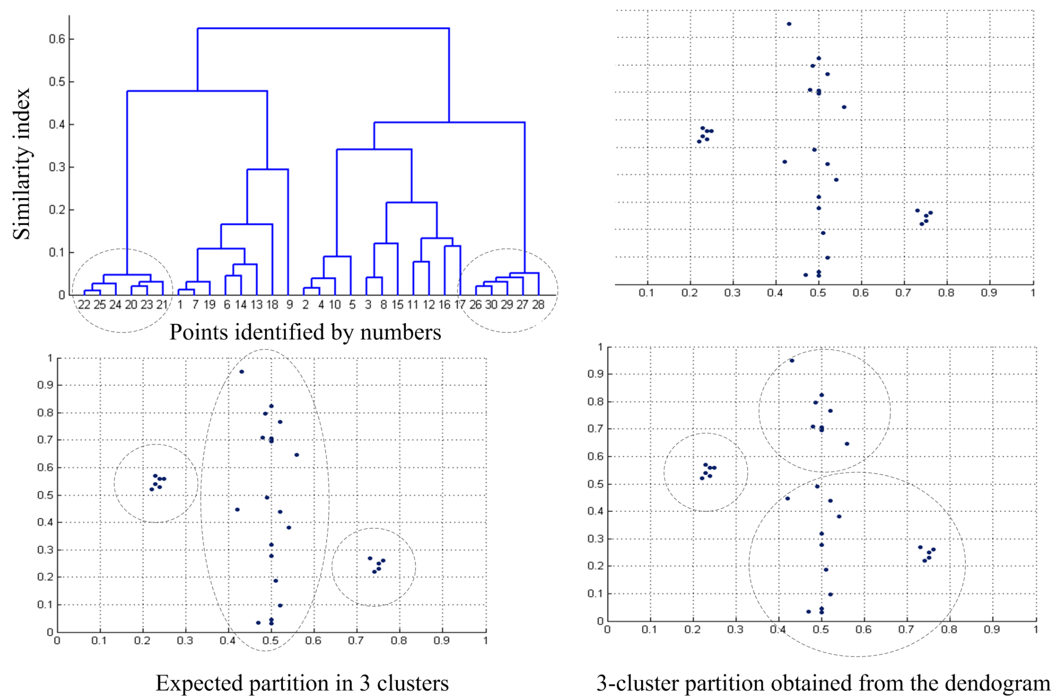


Figure 5.9: Dendogram of a set of 30 input samples, taking Euclidean distance as ‘similarity’ measure. The tool yields to a good performance with 4 clusters, but it fails if 3 clusters are demanded in the solution.

Partitional clustering can overcome these difficulties but, in comparison to the hierarchical option, it has more trouble to select the correct number of output clusters (it is the main accompanying problem of partitional clustering).

The experience with clustering in smart home applications leads to prefer partitional clustering instead of hierarchical. It is not due to the fact that the scenario manages a very large data set (usually), neither are we compelled by computational costs; in addition, the easier management of the number of output clusters makes hierarchical clustering very attractive according to the scenario requirements. In any case, the risk of a counterproductive data-structure imposition is highly unpleasant as very robust methods are required in the early overall division in few clusters (unlike in Figure 5.9), hence we have to move towards more flexible options that focus on a single layer partitioning (the depth and other advantages of the dendogram structure are not profited in the applications shown throughout the dissertation).

- *Monothetic vs. polythetic.*
  Polythetic algorithms use all features at the same time for the computation of distances between samples, whereas monothetic algorithms consider features sequentially to divide input samples into groups [Cha98].

  This point introduces the discussion concerning relationships among features, as a previous step to establish resemblances among elements. In our applications, we try to solve this issue previous to the clustering phase in the design of the structure of input elements (see Section 5.2.2). We propose structures where all features initially have the same meaning and importance, so we are prone to use polythetic methods.

- *Algorithms based on cost function optimization.*
  These methods look for the optimisation of a cost function J, which evaluates the performance of the clustering solution. Here we can differentiate among: *hard or crisp* algorithms, where a vector belongs exclusively to a specific cluster; *probabilistic* algorithms, where vectors are assigned to clusters based on likelihoods established by Bayesian classification arguments; *fuzzy* algorithms, where vectors show a degree of membership to every existing cluster; *boundary detection* algorithms, they are methods focused on searching cluster boundaries, i. e. low density areas that appear in the multi-dimensional solution space.

- *Algorithms based on a supporting transformation or analysis of the data set.*
  Some clustering methods make the most of other data-processing techniques in order to gain a previous transformation or analysis of the propierties and special features of the data set. In other words, we can say that they receive support from *mapping* procedures that help them to fix the number of clusters, centroids, density differences or even the distances among vectors; some examples are: Self-organising maps, Graph clustering, Kernel-based methods, etc. Note that as long as they perform a supporting representation, this representation or map entails a certain degree of distortion or manipulation inherent to the supporting methodology.

Concerning these two last points, at the moment it is difficult to make absolute statements of their running in connection to the intended smart home applications. Sections 6.2 and 6.3 offer comparisons and discussions about the utilisation of diverse clustering approaches. In any case, we have seen in Section 5.1.1 that SC methodologies are supposedly appropriate for dealing with uncertainties related to human behaviour abstraction. Therefore, it is expected that clustering schemes that deploy SC technologies result in suitable performances.

5.2.7 Validation Techniques and Benchmarks

We have already introduced these aspects in Section 5.1.4, here we deal more deeply with existing validation techniques and their implications. In short, we can say that the validation of the results obtained by a clustering algorithm tries to give us a measure about the level of success and correctness reached by the algorithm. Here, two ways of checking clustering solutions are differentiated:

- On one hand, we introduce the classic *cluster validity* tests, which try to evaluate results according to mathematical analysis and direct observation of solutions based on the inherent characteristics owned by the input data set. In a way of speaking, it consists of *idealistic* *or pure* analysis methods as they focus on the definition given to *a cluster* irrespective of the reason that lead us to deploy clustering.

- On the other hand, sometimes different clustering solutions can be benchmarked and checked directly by the application or an environment that simulates the final application. Therefore, it is a more *practical* (or engineering) approach, which covers application-based methods; it means that we are carrying corruption and deformations introduced by the application (or its simulated scenario), the boundary conditions and the specific data used for testing.

In both cases, the value of such quantitative measures is always relative. As it is commented in [TK03]:

> It must be emphasised that the results obtained by these methods are only tools at the disposal of the experts in order to evaluate the resulting clustering.

With regards to the first set of validity measures, three different kinds of criteria are usually considered [HBV02]:

- *External criteria.* It consists of an evaluation of how the solution matches a pre-defined structure based on a previous intuition concerning the data nature.
- *Internal criteria*, which evaluate the solution only considering the quantities and relationships of the vectors of the data set (e. g. proximity matrix).
- *Relative criteria.* Carried out comparing clustering solutions where one or more parameters have been modified.

In [HBV02] some of these evaluation methods are introduced, concluding that they usually work better when dealing with compact clusters. This reasoning yields to an interesting point that remarks uncertainty also with regard to cluster validity; i. e. as it happens with clustering that usually imposes a structure on the input data, cluster validation methods also impose a rigid definition of *what a good cluster is* and develop their assessments according to this particular, own definition.

Uncertainties, commitments and discussions also appears concerning the foundations of the cluster validity measures, as they must fix some essential concepts as, for example, how clusters must be represented (point, hyperplane or hyper-spherical representations), how to calculate the distance between clusters (*max proximity function*, *min proximity function*, *average proximity function*, *mean proximity function*, etc.) or how to calculate the distance between a point and a cluster (*max proximity function*, *min proximity function*, *average proximity function*)[1].

There are a lot of works that compare clustering methods by distinct approaches. To give some examples: for instance in [ZCL05], clustering methods are benchmarked utilising *Log Likelihood* and *classification accuracy* criteria. In [QZ02], popular algorithms are analysed from three different viewpoints: *clustering criteria* (or the definition of similarity), *cluster representation* and *algorithm framework* (which stands for the time complexity, the required parameters and the techniques of preprocessing). In [WHH05], the criteria are: "*stability* (Does the clustering change

[1]For a wide description of these concepts see [TK03]

only modestly as the system undergoes modest updating?), *authoritativeness* (Does the clustering reasonably approximate the structure an authority provides?) and *extremity of cluster distribution* (Does the clustering avoid huge clusters and many very small clusters?)".

A common fact in many works that comparing clustering methods or cluster validity methods is that authors usually have a good previous knowledge of the nature of the data set that they are using before the application of the distinct techniques. They develop comparisons in controlled scenarios or with friendly data that have the characteristics they want to find out (that is the reason behind why some of these papers barely discuss about the deployed data and its origin). There is a lack of demanding scenarios to check clustering methods and cluster validity methods, but it is also because of the fact that we usually do not have an appropriate method to validate the results when they are not already known or intuitively understandable.

Again, we suffer a suffocating circularity. We are in the starting point, smartly expressed in [JPDD03] by the authors when they speak about the difficulties to evaluate clustering performances:

> One reason is that clustering should be performed without a priori understanding of the internal structure of the data. In addition, it is impossible to determine which distribution of clusters is best given certain input patterns without an objective measure for clustering optimality.

And, again, it leads us to move towards the application field as a way to know what we require from clustering. But we can not forget the virtues of validation techniques which are not based on the application scenario, mainly due to the fact that they can identify useful general principles of application, e. g. *fuzzy clustering performs well for shell shaped clusters*.

In the dissertation, we mostly utilise clustering for *pattern or representative discovery*, in order to find suitable validity methods that focus on representativeness or give an important role to representatives. Therefore, we have developed a validity measure called *clustered-vector balance* based on the *clustering balance* measurement introduced in [JPDD03]. In addition, we propose new methods to compare clustering solutions based on their representativeness (Section 6.2).

5.3 A New Cluster Validity Method: Clustered-Vector Balance

The clustering balance measurement finds the ideal clustering solution when "intra-cluster similarity is maximised and inter-cluster similarity is minimised" [JPDD03]. In order to extend the comparison to partitioning clustering and include other parameters under test (in addition to the number of clusters), we introduce substantial modifications to the original equations. It concludes in the *clustered-vector balance* method, a new technique for cluster validity. Clustered-vector balance results in a perfect tool for the progressive refinement and validation of the profile-based application processes that constitute the scope of our research.

5.3.1 Clustered-vector Balance Description

In the clustered-vector balance validation technique, every solution is expressed by a *representative clustered-vector* which takes Λ_v and Γ_v (intra-cluster and inter-cluster average distance per vector) as component values (Figure 5.10). The expressions for Λ_v and Γ_v rest as follows:

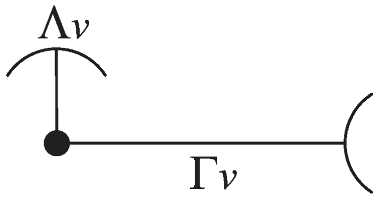


Figure 5.10: Symbol for a representative clustered-vector. The short segment with the concave arc stands for the average intra-cluster distance, the long segment with the convex arc for the average inter-cluster distance.

$$\Lambda_v = \frac{1}{n} \sum_{j=1}^{k} \sum_{i=1}^{n_j} e(p_i^{(j)}, p_0^{(j)}) \tag{5.6}$$

$$\Gamma_v = \frac{1}{k(k-1)} \sum_{j=1}^{k} \frac{n_j}{n} \sum_{l \neq j}^{k} e(p_0^{(j)}, p_0^{(l)}) \tag{5.7}$$

Where n is the total number of input vectors, n_j stands for the vectors embraced in cluster j and k is the number of clusters. $p_i^{(j)}$ refers to the input vector i that belongs to cluster j, whereas $p_0^{(j)}$ is the centroid or representative of cluster j. $e(\mathbf{x}, \mathbf{y})$ stands for the error function or distance between the vectors $\mathbf{x}$ and $\mathbf{y}$. Note that the subindex v denotes the postscript "per vector".

The main differences with respect to [JPDD03] remain in the definition of Γ, which now is not related to the distance to an hypothetical global centroid, but to the distances among centroids, individually weighted according to each cluster population. In addition, Λ and Γ are now expressed in connection with a single, representative vector for the whole solution, and this makes both magnitudes comparable. Therefore, Λ_v is the average distance between a clustered vector and its centroid, whereas Γ_v is the average distance between a clustered vector to other clusters (more specifically, to other centroids).

Directly relating Λ_v and Γ_v can lead to doubtful, meaningless absolute indexes. In [JPDD03], authors introduce an α weighting factor to achieve a commitment between Λ and Γ. The parameter seems to be arbitrarily defined just to relate to both indexes, being adjusted to $1/2$ by default without providing an appropriate discussion. In our case, we can obviously expect that the best solutions will tend to show lower Λ_v and higher Γ_v, but the relationships among both values, their possible increments and the performance evaluation are not linear and have a high scenario-dependence. Since we lack a priori additional knowledge, the final clustered-vector balance index is proposed to be obtained by relating Λ_v and Γ_v using a previous Z-score transformation (i. e. $z = \frac{x-\mu}{\sigma}$). Means and standard deviations of both Λ_v and Γ_v are obtained considering the total set of solutions to compare. Finally, the best solution maximises:

$$\mathcal{E}_v(\mathcal{X}) = \Gamma_v|_z - \Lambda_v|_z \tag{5.8}$$

We no longer require α. However, we can consciously add it again if we have a previous biased opinion with respect to *what a good clustering solution is* according to the final application, i. e. whether we want to favour solutions where clusters are compact or we prefer that they are as different/far as possible. Hence it would remain:

$$\mathcal{E}_v(\mathcal{X}) = \alpha\Gamma_v|_z - (1-\alpha)\Lambda_v|_z \tag{5.9}$$

5.4 Interpretation of the Context

In applications that deploy clustering for control processes, we have seen that clustering algorithms are not only intended to execute *prediction based on groups*, but also *hypothesis generation* (i. e. *context awareness*). Beyond providing clusters and representatives, clustering tools can offer relevant information concerning the analysed data set and the properties of the obtained patterns/clusters. Thereby the linked controller interprets the context following 'fixed rules' (devised for the specific case, see Section 4.3.2).

Thus, cluster validity methods and clustering outputs are utilised to provide additional information that allows the controller to receive context awareness concerning user habits. The capabilities of diverse clustering techniques to offer reliable information about the context is checked in Section 6.4.

Beyond the patterns (representatives), the clustering information considered to appraise the context includes:

- Number of discovered patterns.
- External distance (or distance among patterns), i. e. inter-cluster similarity.
- Density (coherence, internal distance or distance among vectors grouped in the same cluster), i. e. intra-cluster similarity.
- Membership level or size of clusters (in terms of number of members).

A priori, the number of discovered patterns points to indicate the number of existing habits, but the validity of each specific discovered pattern must be assessed. Let us see with some examples how representative patterns can be assessed.

5.4.1 Assessing Representative Patterns

Moving away from the representatives for a moment – i. e. the shape and the information included in vectors –, the following examples explain how the relevance of representatives (their strength or coherence) can be evaluated.

Note that figures are shown in order to allow us to visually corroborate if the automated reading of the context is coherent. In a real implementation the interpretation/summarisation of the context is carried out by a dedicated module or agent. This module (usually a PTG) adapts the context information in order to be ready for the specific control agent decision making. A generic model of the PTG is depicted in Section 4.3.2.

Example: case 1

Figure 5.11 shows a two-dimensional representation of an input vector data set. The clustering tool has discovered three clusters with the additional information provided in Table 5.1. Please note that, in the table, distances are normalised and linearised for a better understanding. For each cluster, $\overline{dist}$ and σ_d are defined as follows:

$$\overline{dist} = \frac{1}{N}\sum_{i=1}^{N} dist(v_i, \bar{v}) \tag{5.10}$$

$$\sigma_d = \sqrt{\frac{1}{N}\sum_{i=1}^{N} dist(v_i, \bar{v})^2} \tag{5.11}$$

Where N is the number of vectors that are members of the specific cluster under analysis, $\bar{v}$ is the representative (discovered pattern) and v_i stands for each vector of the explored cluster. $dist(\mathbf{x}, \mathbf{y})$ depends on the selected similarity measure.

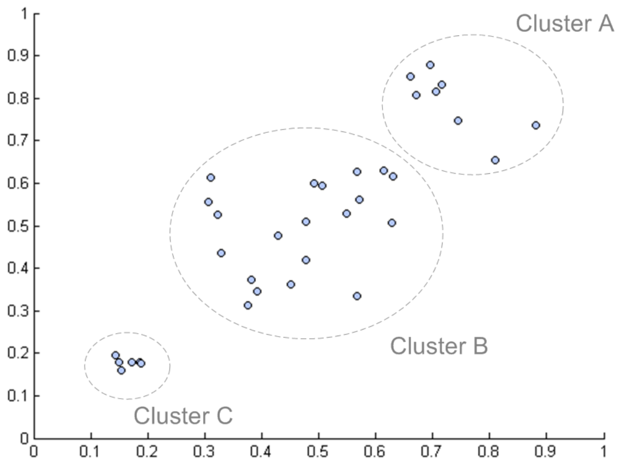


Figure 5.11: Example of clustering solution, case 1.

Cluster	members	$\overline{dist}$	σ_d
A	23%	0.087	0.047
B	60%	0.140	0.053
C	18%	0.019	0.007

dist(A,B) =	0.396
dist(A,C) =	0.836
dist(B,C) =	0.439

Table 5.1: Information obtained from clustering results in case 1.

With the information provided by the table, the context interpretation agent can make the next deductions:

- Distances among representatives are big enough, it means that they share the solution space with certain equidistance and, therefore, representative vectors are quite different from one another, being 'Cluster B' the central cluster.
- 'Cluster A' is small (few members) and a little dense (high $\overline{dist}$ and σ_d), i. e. it results in a barely reliable representative.
- 'Cluster B' is significant and crowded (big pattern) with most of the input vectors, but it also presents low density (due to high values in $\overline{dist}$ and σ_d), so it is a bit vague pattern.
- 'Cluster C' is small (few members) but dense (low $\overline{dist}$ and σ_d), therefore it presents an occasional, but reliable pattern.

Example: case 2

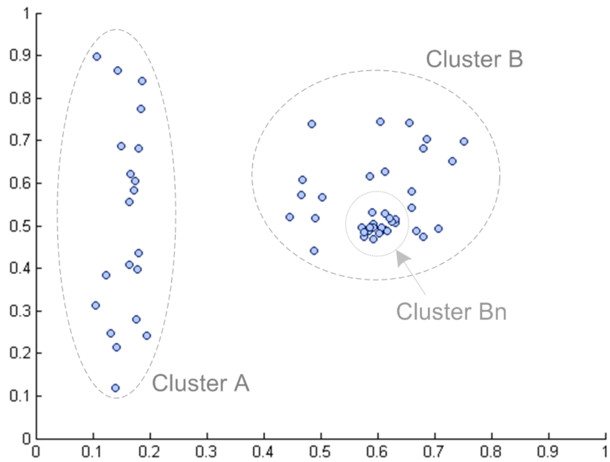


Figure 5.12: Example of clustering solution, case 2. 'Cluster Bn' is included in 'Cluster B'.

Cluster	members	$\overline{dist}$	σ_d
A	33%	0.205	0.111
B	67%	0.096	0.055
(Bn)	33%	0.023	0.010

dist(A,B) =	0.444
dist(A,Bn) =	0.442
dist(B,Bn) =	0.051

Table 5.2: Information obtained from clustering results in case 2.

For this second case let us suppose that we have two different tools, one of them only finds 'Cluster A' and 'Cluster B', whereas the second tool is able to find nested clusters and also discovers 'Cluster Bn'.

In a case where we only have 'Cluster A' and 'Cluster B', the reasoning is as follows:

- Distances among representatives are big enough again, it means that they share the solution space with certain equidistance and, therefore, representative vectors are quite different from each other.
- 'Cluster A' is crowded enough (33% members). The very high values in $\overline{dist}$ and σ_d indicates low density in principle. In any case, if we trust the clustering tool is performing correctly, we consider the magnitude of the discovered clusters and their distances, and we know that the tool is able to find an elongated cluster, such values in $\overline{dist}$ and σ_d lead us to recognise that we are dealing with an elongated cluster (in any case, this capability is not usually required in the applications explored in this work).
- 'Cluster B' is a big cluster (67% members). In comparison with 'Cluster B' in 'case 1', the reduction in $\overline{dist}$ means a higher density, but the fact that σ_d is even a bit increased now points to identify differences of density within the cluster.

Additionally, if the tool also detects 'Cluster Bn', the available information reveals that 'Bn' is a considerable cluster (33% members), dense (low $\overline{dist}$ and σ_d), nested and almost concentric to 'Cluster B' (very low $dist(B, Bn)$ in comparison to $\overline{dist_B}$).

With these two examples we have seen how the context can be read based on additional clustering output data. How to react to such context readings is concerned to the design of controllers.

5.4.2 Interpretation of User Habits in a Real Case

Now, let us check how this additional data together with the patterns can provide readings of the context in a real example of a smart home.

Scenario Description

The data for the test belongs to the *Viennese Flat Database* (Appendix B.2.2). The collected data embraces a set of 57 occupancy profiles (57 available day samples).

The employed clustering tool is a Self-Organising Map embedded in a PTG as it is described in Section 4.3.2. The PTG is the module in charge of discovering patterns and giving a correct interpretation of the context concerning occupancy habits.

Test Execution

The PTG takes the 57 input vectors and executes the clustering process. It finds out three main clusters – named 'Cluster 1', 'Cluster 2' and 'Cluster 3' –, they embrace 16%, 23% and 42% of the total population (input vectors) respectively. The remainder (19%) is formed by a 0-value or absent days group (5%) and some minor, distant groups that are directly rejected due to the low membership levels (5%, 5% and 4% respectively). Since we admit only one representative pattern per day of the week as maximum, clusters are considered by the PTG when they include at least $1/7 \approx 10\%$ of the input population.

The obtained clusters are represented by the representatives/patterns shown in Figure 5.13: 'Pattern 1', 'Pattern 2' and 'Pattern 3'.

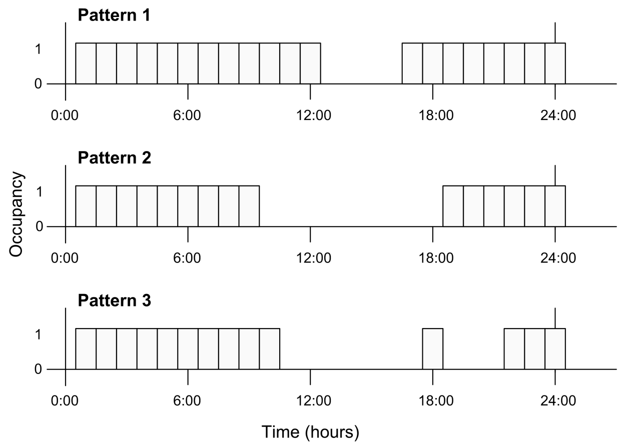


Figure 5.13: Found patterns for the example. $Occupancy = 1$ stands for presence and $Occupancy = 0$ for absence.

Since we are dealing with all the patterns of the database together, the context awareness process starts putting discovered patterns in connection with the days of the week, i. e. which pattern dominates every day of the week. The PTG checks this correspondence and states new *groups* for each pattern based on its findings. It discovers that 'Pattern 1' dominates Sundays (group 1), 'Pattern 2' represents Thursdays (group 2) and 'Pattern 3' stands for Mondays, Tuesdays, Wednesdays and Fridays (group 3). Since there is not a dominant pattern for Saturdays, it is evaluated as a *non-predictable* day, Figure 5.14. Depending on the intelligence embedded in the applications that make use of this occupancy information, specific control agents will apply alternative or preventive strategies for this day of the week whilst it does not have a stable pattern which represents it.

Following validity methods for the context interpretation, the Pattern Generator states that the three patterns are solid, being 'Pattern 2' the most respected or stable (very low $\overline{dist}$). Solid means dense enough (based on Euclidean similarity measures) and with a considerable membership. The distances among the three patterns are significant enough to be considered as distinct (it can be directly seen in Figure 5.13).

On the other hand, checking distances among profiles within groups, the differences detected in groups 2 and 3 (considerable σ_d in comparison with their respective $\overline{dist}$) point to consider that existing *outliers* or erratic elements are quite far from the representatives. This last detail is probably indicating the presence of 'days off' inside the groups that cover working days, a fact that would be avoided if users previously inform the system about expected holidays and periods of days off.

Test Evaluation

Now we have three patterns that theoretically represent the daily life (weekly occupancy cadences or habits) of a normal inhabitant. Since the database comes from a known user, we can check the validity of results and see if the interpretation of the context carried out by the PTG gives a good summarised description of the user behaviour.

The user under test works 30 hours per week (central 'absence' in 'Pattern 2' and 'Pattern 3') and does sport activities on Monday, Wednesday and Friday evenings (second gap in 'Pattern 3')

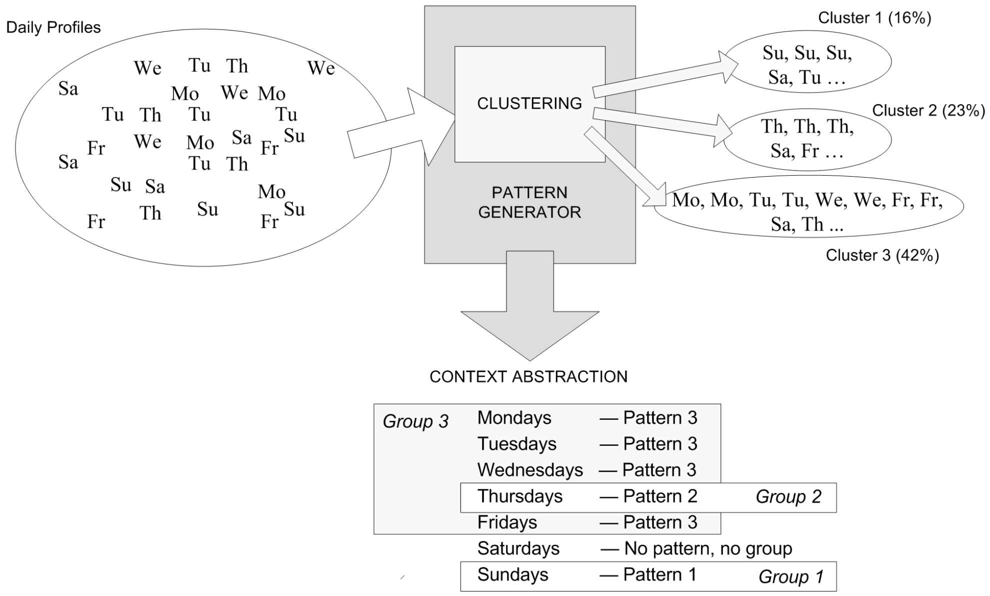


Figure 5.14: Representation of the process carried out in the example. Note that the *clusters* stated by the clustering tools and the *groups* established by the PTG are not the same. The *groups* appear when the PTG are forced to identify a *pattern* for each day of the week according to the clustering outputs.

as well as Thursday mornings (it corresponds to the earlier leaving-time in 'Pattern 2'). Tuesday evenings he usually goes out instead of doing sports. He wakes up later than usual on Sundays, when he normally goes for a walk with friends until the afternoon ('Pattern 1'). Saturdays are entertainment days without any habitual activity or repeated habit (no dominant pattern). Therefore, the user's direct testimony confirms absolutely the reading of the context carried out by the PTG (using SOM clustering).

Thus, we can conclude that clustering tools embedded in PTGs show outstanding capabilities to discover habits and abstract context information.

6 Tests for the Optimisation of Clustering Processes

This chapter covers analysis and comparisons among clustering methodologies, which are tested in order to check performance skills and the suitability for accomplishing application requirements. The goal of the tests is to acquire knowledge and a better understanding of the progressive refinement and improvement of the introduced smart home applications, designs, control algorithms, parametrisation and tools.

6.1 Comparison of Similarity Methods

The conducted experiments of this section have two main objectives:

1. To check *clustered-vector balance* as a clustering validity algorithm by means of comparisons with other relative clustering validity criteria. Clustered-vector balance has been introduced in Section 5.3 as a new approach to obtain improved evaluations and comparisons of clustering performances.
2. To obtain a precedent for the selection of the most appropriate similarity metric (Section 5.2.3) for the application case: energy consumption pattern discovery in buildings, which is a significant example of time series clustering.

These two aspects are relevant as (1) we require reliable tools for the continuous refinement of profile-based applications, and (2) our design methodologies are prone to building profiles as a time series in order to represent users' habits and behaviours.

6.1.1 Input Data

Data concerning the energy consumption of five Spanish university buildings have been utilised as input data to carry out the tests. Details of the database are commented in Appendix B.2.3.

For each building 124 days of information have been collected and transformed into input vectors afterwards. Thus, an input vectors is a time series with 24-fields of hourly information of energy consumption in kWh. They are divided into two subsets: 100 days taken for training and cluster validity analysis processes, and 24 days deployed in the evaluation.

Figure 6.1 shows an example of three consecutive profiles of one of the buildings.

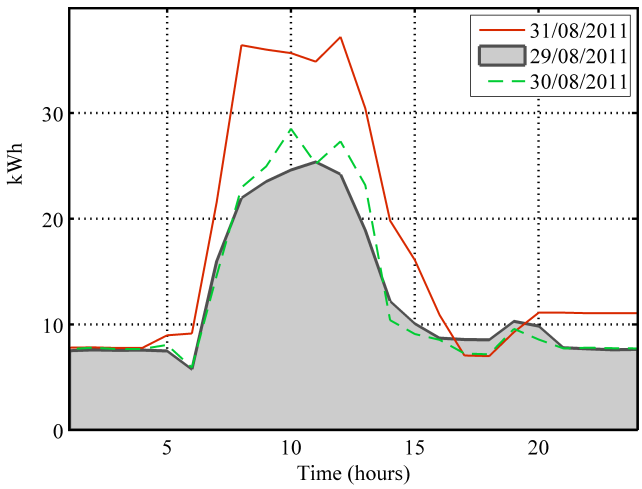


Figure 6.1: Example of three consecutive consumption days ("Rectorat").

6.1.2 Scenario and Test Description

Tests are described as follows:

Real cases are clustered using different similarity distances. Later on, each clustering solution is *validated* by means of different validation techniques (the similarity measure of the validation algorithm is switched as well). In addition, test vectors (selected at random and not processed by the clustering tool) are utilised to *evaluate* the representativeness of the main patterns of the clusters or centroids, measuring the average distance between the test vectors included in a cluster and the representative of the respective cluster. To do that, Equation 6.1 is defined as follows:

$$V = \frac{1}{m} \sum_{j=1}^{k} \sum_{i=1}^{m_j} e(q_i^{(j)}, p_0^{(j)}) \tag{6.1}$$

where m is the total number of vectors put aside for evaluation, m_j stands for the vectors embraced in cluster j. $q_i^{(j)}$ refers to the evaluation vector i that belongs to cluster j. The membership of the evaluation vectors is established according to the proximity to the found patterns p_0. e represents the distance used for evaluation. The evaluation also uses all of the diverse similarity distances under test.

In the trivial situation that all similarity measures affect the clustering solution in the same way, or in the hypothetical case that each distance is the most successful at finding a clustering solution with specific characteristics, we should expect that clustering carried out using a specific distance obtains the best results when the same distance has been used for validations or evaluations. Otherwise, we will have arguments to establish better and worse similarity measures for our specific application case.

Figure 6.2 shows a schematic overview of the experiment.

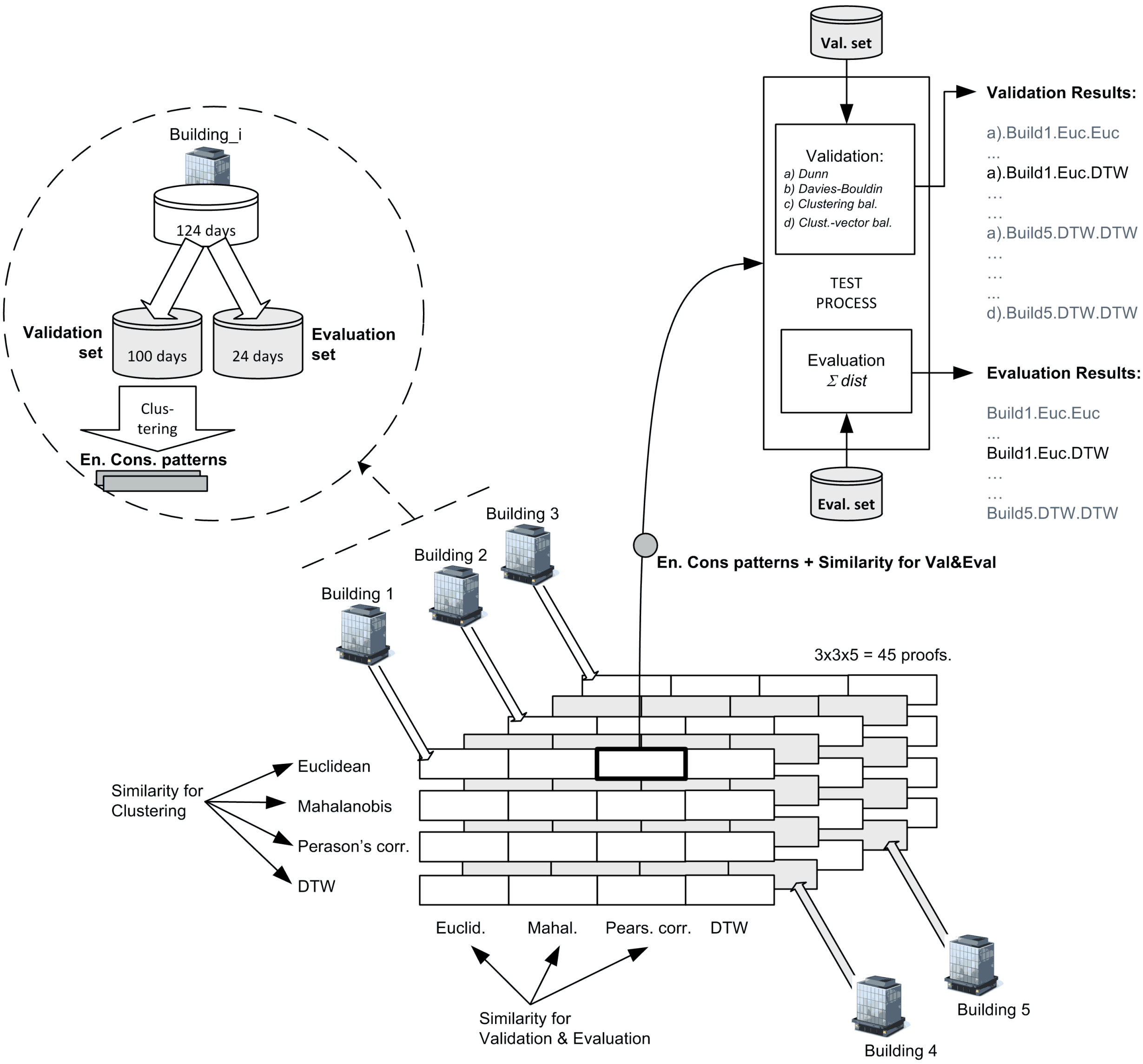


Figure 6.2: Comparison of Similarity Metrics: test design

6.1.3 Test Parameters, Execution and Comparisons

The similarity measures under test have been explained in Section 5.2.3, they are: a) Euclidean distance, b) Mahalanobis distance, c) distance based on Pearson's correlation and d) DTW distance.

In the first step, the training data is processed by a Fuzzy clustering module that uses the FCM algorithm to compute clusters. As referred to above, the FCM algorithm uses the four distance measures to state vector proximity. In each case, the initial number of clusters has been fixed according to *clustering balance* and Mountain Visualisation [RNK03], as well as maintaining the final scenario purposes (i. e. allowing a maximum of 8 energy consumption models).

Since all features correspond to the same phenomenon (electricity consumption), normalisation is not carried out feature by feature, but based on the mean μ and standard deviation σ of the whole dataset (i. e. a simple uniform scaling). Failing to ensure that all features move within similar ranges has been addressed as a problem for similarity measures like Euclidean distance, as "features with large values will have a larger influence than those with small values" [SAC+08]. In any case, for univariate time series we are confident that the multi-dimensional input space is not distorted and the relationship among features keep the same shape and proportionality.

The clustering solutions are *validated* using: a) clustering balance with $\alpha = 1/2$ [JPDD03], b) clustered-vector balance (Section 5.3), c) Dunn's index [Dun73] and d) Davies-Bouldin index [DB79]; and *evaluated* by means of e) Equation 6.1, which checks how representative discovered patterns are by means of data separated for testing.

Therefore, the test process results in: 5 builds. $\times$ 4 clust.(metrics) $\times$ 4 indices $\times$ 4 index(metrics) $=$ 320 validations/evaluations. With all the obtained outcomes the next comparisons are carried out: a) best clustering solution (best validation), b) best evaluation, c) soundness of validation algorithms, d) best independent clusters.

The last point refers to the capability of finding good clusters (i. e. dense, regular high similarity) irrespective of the global solution. The best clusters obtain minimum values in the next fitness function:

$$f_j = (1 - m_j) \times \Lambda_j \tag{6.2}$$

where m_j stands for the membership or amount of population embraced by cluster j (0: none, 1: all input samples) and Λ_j for the intra-similarity of cluster j. Clusters must overcome a membership threshold to be taken into account ($m_j \geq 0.08$, i. e. at least 8% of total population).

6.1.4 Results

The high number of generated indices lead us to condense results in a meaningful way in some figures and tables. We discuss the obtained findings in separated points.

Characteristics of the Scenario Under Test

Experiments face quite a demanding scenario where to identify clear clusters is not an easy task and the selection of the similarity measure effects the shape of obtained models. It is obvious when the solution patterns are compared, e. g. Figure 6.3 shows the representative pattern corresponding to a specific discovered cluster according to every one of the clustering solutions in the case of building "Edifici A1". Note that patterns are similar in shape, but different enough to have a relevant influence in subsequent applications. For instance, a control system that uses the predicted patterns to adjust the supply of energy sources in advance would perform differently in each case, resulting in distinct levels of costs and resource optimisation. Moreover, the patterns displayed in the figure represent a different percentage of the input population (Euclidean: 17%, Mahalanobis: 20%, Correlation: 13%, DTW: 24%).

In addition, the demanding nature of the problem is also noticeable in the disagreement detected by the validation techniques (see next point).

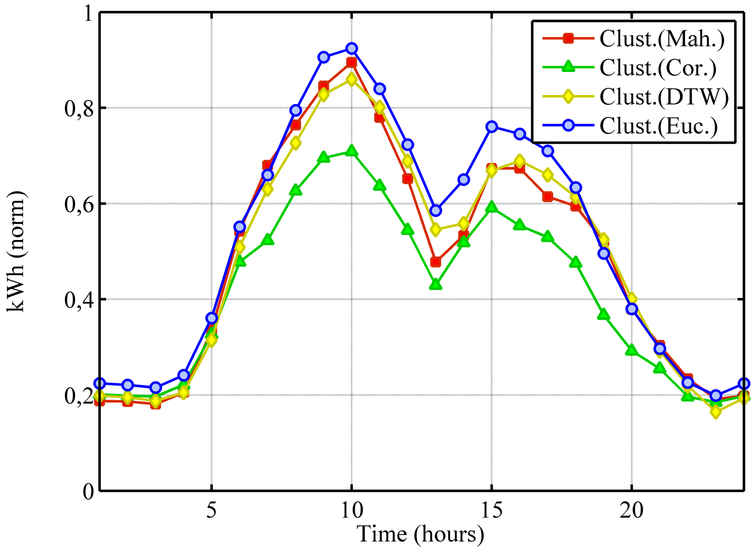


Figure 6.3: Representative pattern of a specific cluster for building "Edifici A1"according to every clustering solution: using Euclidean (blue circles), Mahalanobis (red squares), based on Pearson's Correlation (green triangles) and DTW (yellow diamonds) similarity metrics.

Best Validation and Best Evaluation

To establish which similarity measure involves the best clustering performances, we must check all the tests together, but by separating *validation* and *evaluation* due to the different nature with which they approach the assessment task (see above).

For the evaluation, we use four different validity methods. In order to gain an overall, joined perspective of the obtained results and indices, we ensure they (validity methods) assign points to the similarity distance that they consider the best for every conducted test. For example, in building "Rectorat", in the test where all validity methods deploy DTW distance for validation, Dunn's index and clustered-vector balance index find that the Euclidean metric is the best, whereas Davies-Bouldin's index bets for Mahalanobis metric, and finally, clustering balance index supports the solution based on the DTW distance. Hence, in this example the Euclidean metric gains $1/2 = 1/4 + 1/4$ points, Mahalanobis metric $1/4$, DTW distance also $1/4$ and 0 for Correlation. This way of summarising results leads to Figure 6.4.

What Figure 6.4 displays is that validity methods are prone to consider clustering solutions based on Euclidean metric as the best, irrespective of the measure used for validation. Moreover, note that the coincidence between the distance for clustering and the distance for validation has no significant influence in the assessments.

The case of validation is analogously checked, but here only Equation 6.1 is used for the assessments. Results are shown in Figure 6.5. Using data put aside for testing, evaluation reveals that DTW and Euclidean distances compete similarly for the best scoring measure for clustering, whereas Mahalanobis and Correlation metrics always perform worse. Curiously enough, DTW distance obtains the worst records in the validation analysis; this issue is dealt with later when validity methods are compared.

In short, as far as distances for clustering are compared, *validation* analysis set Euclidean as the

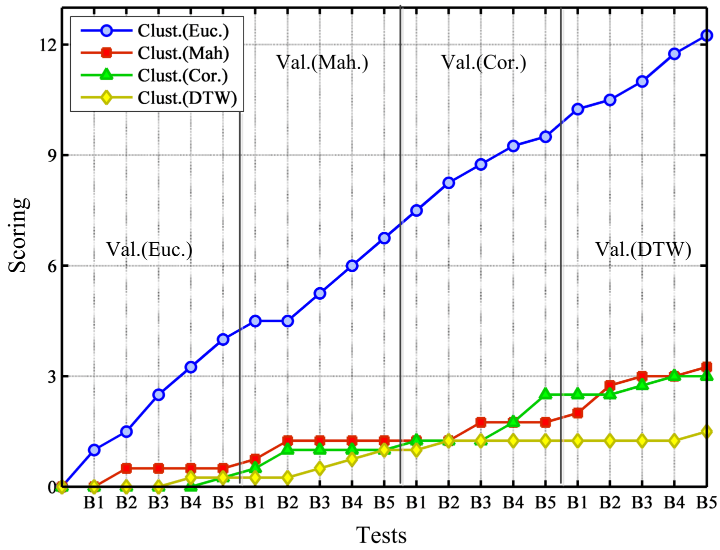


Figure 6.4: Joined assessment carried out by clustering validity methods.

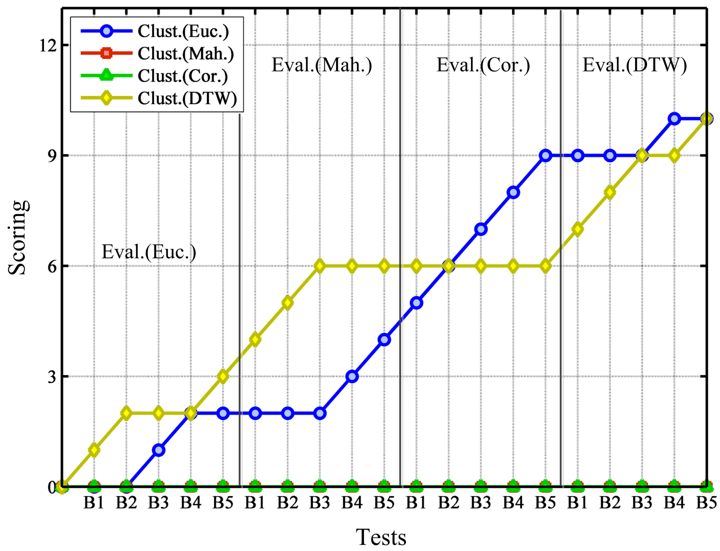


Figure 6.5: Assessment carried out using data saved for testing

best metric for time series clustering, whereas *evaluation* tests favour both DTW and Euclidean similarity distances.

Validation Algorithms

To review validation algorithms is not an easy task, note that the purpose here is to audit the performance of algorithms that are usually used for checking. In any case, we can reach some conclusions comparing their results to one another as well as looking at the evaluation outcomes. Table 6.1 displays the trends that validity techniques show when comparing clustering solutions that use different similarity measures. Considering all the tests together, the Mode represents the most typical position taken by the clustering solution that uses the marked distance, standing "1st" for the best evaluation and "4th" for the worst. The Mean contributes to the assessment and gives an impression about how stable the typical scoring is. Therefore, the following points can be reasoned from Table 6.1:

	Dunn	D-Boul.	Clust.b.	Vect.b.
	Mode – Mean	Mode – Mean	Mode – Mean	Mode – Mean
Clustering (Euclidean)	1st – 1.4	1st – 1.6	1st – 1.8	1st – 1.3
Clustering (Mahalanobis)	2nd – 2.3	3st – 2.1	4th – 2.9	4th – 3.2
Clustering (Correlation)	3rd – 2.5	3rd – 2.5	4th – 3.0	4th – 3.2
Clustering (DTW)	4th – 3.9	4th – 3.9	3rd – 2.4	2nd – 2.4

Table 6.1: Validity techniques evaluations, statistical Mode and Mean.

- All methods are in agreement over the measure that achieves the best clustering in general terms, i. e. Euclidean metric.
- Later on, two groups appear:
  - Group 1: Dunn's and David-Bouldin's indices usually scorn solutions based on DTW distance and put it in the worst place, finding that Mahalanobis and Correlation metrics are more suitable for the intended clustering.
  - Group 2: Otherwise, clustering balance and cluster-vector balance give credibility to the DTW distance, placing it before Mahalanobis and Correlation.

The validation tests favours the assessments given by group 2, so we have arguments to believe that clustering balance and clustered-vector balance are techniques more appropriate to evaluate time series clustering solutions, at least for the current application case. If we look again at Table 6.1 and compare these two techniques with each other, vector balance seems to be more stable judging distance measures, whereas result comparisons of clustering balance are more variable and case-dependent. In short, there are three factors that opt for clustered-vector balance instead of clustering balance: clustered-vector balance 1) shows higher coincidence with the rest of validity methods, 2) is more stable in the assessments and 3) matches the validation test outcomes better.

Now it is possible to clarify why DTW distance received such a low score in validation tests, in part due to the rejection of Dunn's and David-Bouldin indices, but also because of the fact that, although usually showing a very little difference in the evaluations, clustered-vector balance rarely places DTW-based clustering before Euclidean-based (note that in Figures 6.4 and 6.5 only the 1st solution obtains points; the 2nd, 3rd and 4th solutions receive no points).

Best Independent Clusters

Grouping all the clusters generated by the diverse clustering solutions together, the three best clusters according to Equation 6.2 are highlighted. Again a competition among similarity measures is carried out, and results are displayed in Figure 6.6. Here, results show no evidence to say that a specific distance measure obtains better compact clusters as a general rule. Again, the type of distance for validation does not significantly affect this measure (except for perhaps the case of Euclidean clustering); instead, the specific case (building) exerts a decisive influence for the selection of the measure to discover compact clusters (low internal dissimilarity). In any case, although results are not discriminative, it is at least worth considering the advantage of DTW and Correlation distances, and the fact that Euclidean metrics gains the lowest results in this aspect.

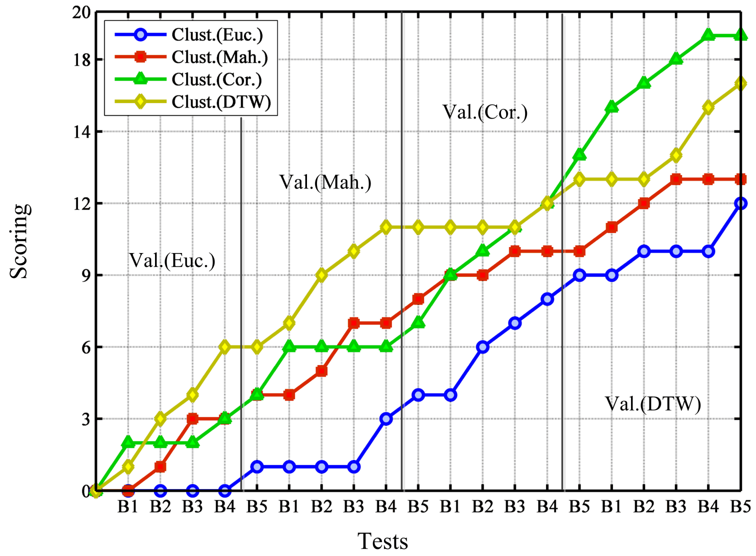


Figure 6.6: Comparison of the capability to discover *the best individual clusters*

6.1.5 Discussion

In short, the developed experiments place Euclidean distance as the best similarity metric to receive good general solutions in *raw-data-based* time series clustering. In other words, using Euclidean distance as a similarity metric, the best trade-off, balance solutions are obtained, as it is the most appropriate option to deal with the input space as a whole. Therefore, we hypothesise that Euclidean distance actually considers data correlation in an indirect and fair enough way, suitable for the general clustering solution.

The weights that Mahalanobis distance provides in the measures in order to favour the appraisal of correlations also introduces a questionable distortion in the input space that causes loss of information or structure and can be even seen as an unnecessary redundancy. On the other hand, distances based on Pearson's correlation, intended to indicate the strength of linear relationships, have trouble correctly interpreting the distribution and relationship of vectors that present low similarity, in addition to being more sensible facing outliers, whether they are vectors or feature values. In the end, Mahalanobis and normal correlation seem to perform well the detection of certain nuclei, but have more problems dealing with intermediate vectors, i. e. the background clouds of vectors with low, variable density. In short, we can consider that these two metrics are biased to find a specific sort of relationship, losing capabilities to manage the space as a whole.

DTW distance deserves a special mention as it has been the most successful in the evaluation test and in finding the best clusters. We can expect that in related prediction applications it performs as good as the Euclidean distance and sometimes even better. If both similarity measures are compared based on the conducted test, the reasons for the different performances can be inferred

(Figure 6.7 shows discovered patterns and embraced input clusters for both clustering solutions, using DTW and Euclidean similarity measures).

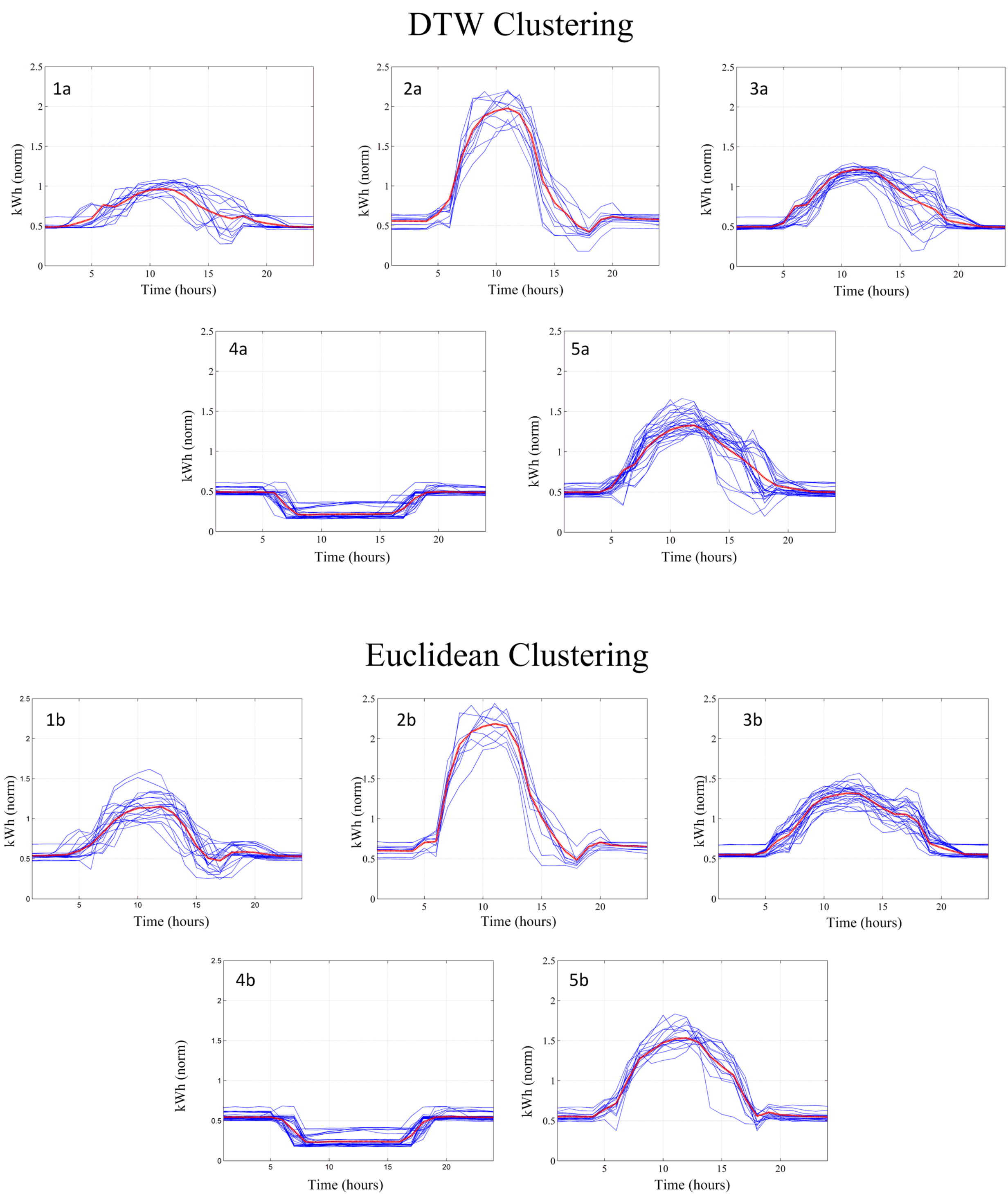


Figure 6.7: Patterns discovered using DTW in 'Rectorat' building and embraced samples.

On one hand, using DTW distance for clustering also entails a deformation of the input space in order to better capture the representative nuclei. It ensures that the clusters' gravity centers move toward areas where high-correlated samples (or parts of the samples) are better represented, sacrificing capabilities to represent or embrace samples that do not show such high-correlation or coincidence. But, compared to Mahalanobis or Correlation distances, the induced deformation is more respectful with the overall shape or structure that forms the input samples all together. Figure 6.7 is a good example to check Euclidean clustering, not only compared to DTW, but to all the considered measures that somehow estimate correlation (where DTW has proved to be the most suitable). At first sight, to visually compare between the two clustering solutions is not easy, both seem to capture the essential patterns with minimum variation. DTW distance favours samples that show parts that really match one another, being more lax if the rest of the curve does not fit such coincidence. In Figure 6.7, this is noticeable due to the fact that, as a general rule, the DTW solution shows more dark zones (curves are closer) as a result of the *obsession* to find correlated parts. The two equivalent patterns labeled '3a' and '3b' are a good example to assess this phenomenon. Here, in the DTW case, the effort made to fix the high-correlated first part of the profile is significantly spoiled by the less-coincidental last part of the profile. Otherwise, the group found by the Euclidean solution perhaps displays a better trade-off, balanced solution.

In short, and according to the test results, the DTW distance usually better defines the important clusters, losing representativeness in the less correlated ones; otherwise, Euclidean metric can result in main cluster representatives which are not so good, but better in order to define the lower-density ones and to summarize the input space as a whole.

Finally, even though computer resources are not a priori a limiting factor in the introduced application case, the time required by the clustering process in every one of the tested configurations is worthy of consideration. Only by changing the similarity measure, the required time by the clustering task shows a different order of magnitude: Euclidean similarity takes hundredths of seconds (0.0Xs); Mahalanobis metric, tenths (0.Xs); Correlation needs seconds (Xs); and DTW distance, tens of seconds (X0s).[1]

[1] These values must not be taken as absolute measures, but only to compare clustering performances with one another. Please, note that the processing time depends on the machine used for computation.

6.2 Comparison of Clustering Techniques: Representativeness Analysis

The *representativeness analysis* evaluates and compares how output patterns discovered by different clustering techniques represent the samples embraced in their respective clusters.

The main goals of this analysis are as follows:

1. On one hand, to know which clustering approaches/techniques show the best capabilities to discover representative patterns in cases where human behaviours are modelled as time series profiles.
2. On the other hand, to discern which features of clustering techniques are owned by the analysed methods and what influence they have in the representative-discovery task.

Thus, we want to know which tools are the most interesting for us (1) and how to progressively refine them (2). Such objectives are pursued by analytical techniques that do not directly take into account the application scenario.

6.2.1 Scenario and Test Description

The vector inputs for the clustering tools are occupancy profiles obtained using Leako databases (Appendix B.2.1). Three **cases**, i.e. three types of occupancy behaviours (or levels) related to three different dwellings, are studied:

- *High/medium daytime (DO)*, corresponding to a user that presents a high/medium occupancy during daytime hours.
- *High/medium at night (NO)*, corresponding to a user that usually presents a high/medium occupancy at night.
- *Low presence (LO)*, corresponding to a user that presents a low occupancy irrespective of the moment of the day.

The number of samples for each case covers a period of five years, i.e. about 1800 daily profiles (input vectors).

The tests consist of the comparison of diverse clustering techniques. It is done by forcing the tools to discover a few numbers of clusters among the set of input vectors for each considered case (DO, NO and LO). Therefore, the set of cluster centroids discovered by each different technique will represent the whole clustering solution in the analysis. Two *credibility indices* that evaluate the representativeness of centroids will establish which is the best technique in the comparison. Figure 6.8 shows a schematic overview of the experiment.

The clustering methods under test are the following: Self-organizing Maps, Exclusive Self-organising Maps, Fuzzy C-means, K-means, K-means with Repeated Bisection, Graph Clustering and Support Vector Clustering.

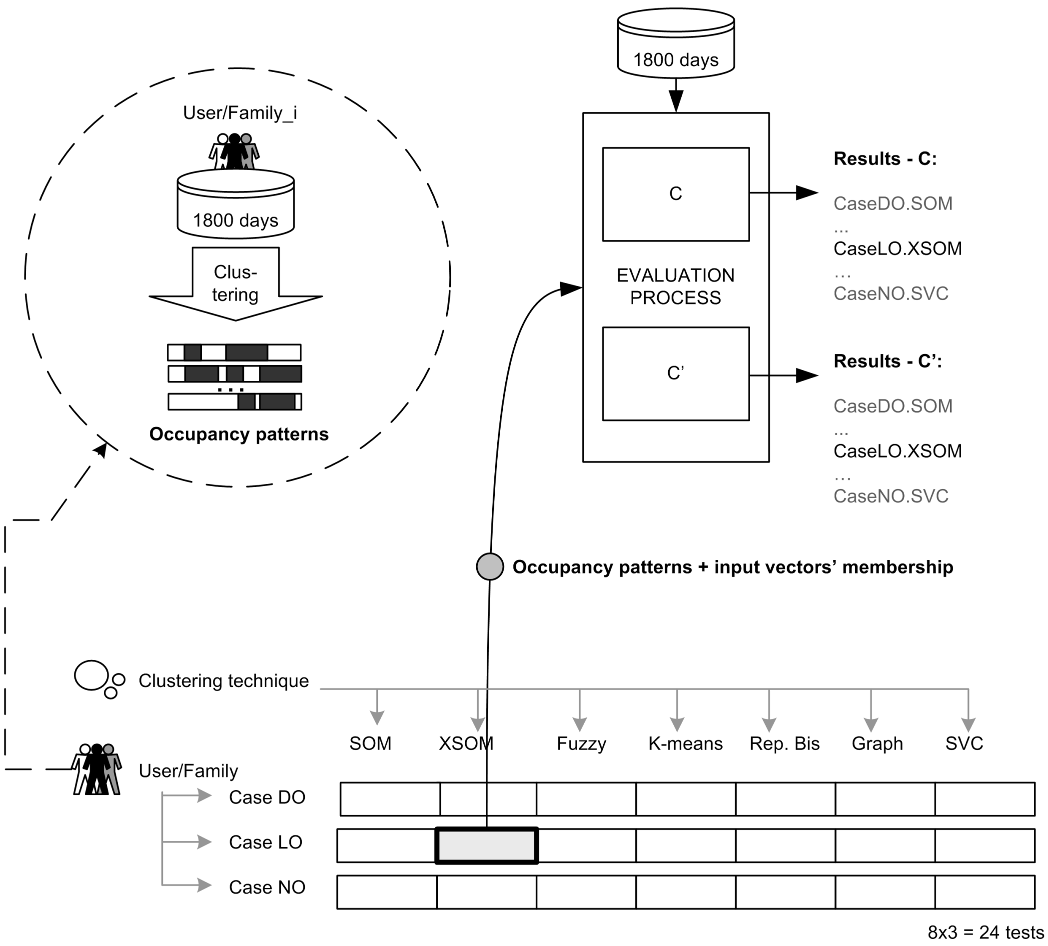


Figure 6.8: Comparison of Clustering Techniques by means of Representativeness Analysis: test design

6.2.2 Test Execution

Explanation of the clustering techniques and the utilised parametrisation by default are offered in Appendix B.1.4. In addition, all the methods utilise the *mean center* to establish the point for the cluster centroid or representative. For every clustering technique, different configurations have been checked in order to obtain the optimal parametrisation. Among this first set of possible configurations, the one with the best performance is selected to compete against the other clustering techniques.

Tables 6.2, 6.5, 6.6, 6.3, 6.7, 6.4 and 6.8 stand for the distribution of input samples in the discovered output clusters for each behaviour case and for each clustering technique. For all methods, a pattern is considered as negligible if it embraces a percentage of population less than 8.0%. Note that every discovered cluster also means a discovered representative pattern.

The first technique under test is the SOM algorithm, whose maximum number of clusters is initially fixed to 10 (a maximum established by the a priori expert knowledge). Unlike other clustering tools, SOM-based methods are not so forced by the initial number of clusters and are able to generate empty clusters. Hence, according to the results obtained by SOM clustering, we

Cluster	DO Case	NO Case	LO Case
1st	30.6%	41.4%	58.6%
2nd	24.4%	28.5%	23.5%
3rd	24.1%	17.2%	15.7%
4th	18.9%	12.7%	–
Neglig.	2.0%	0.2%	2.2%

Cluster	DO Case	NO Case	LO Case
Outliers	10.6%	6.5%	9.2%
1st	26.9%	39.5%	53.0%
2nd	21.9%	25.8%	24.0%
3rd	22.1%	17.2%	12.7%
4th	18.2%	10.8%	–
Neglig.	0.3%	0.2%	1.1%

Cluster	DO Case	NO Case	LO Case
1st	39.1%	30.7%	44.6%
2nd	23.5%	22.9%	30.2%
3rd	17.3%	18.9%	25.2%
4th	10.5%	14.4%	–
5th	9.6%	13.1%	–

Cluster	DO Case	NO Case	LO Case
1st	51.0%	63.3%	53.2%
2nd	41.1%	36.5%	46.8%
Neglig.	7.9%	0.2%	0.0%

Cluster	DO Case	NO Case	LO Case
1st	39.7%	30.1%	47.0%
2nd	21.5%	24.1%	28.0%
3rd	10.7%	21.0%	21.0%
4th	9.7%	14.5%	–
5th	9.6%	10.3%	–
Neglig.	8.8%	0.0%	4.0%

Cluster	DO Case	NO Case	LO Case
1st	46.4%	25.9%	44.6%
2nd	25.6%	21.2%	29.8%
3rd	10.6%	19.7%	21.5%
4th	8.8%	19.5%	–
5th	8.5%	13.7%	–
Neglig.	0.1%	0.0%	4.1%

Table 6.7: K-means with Repeated Bisection sample distribution

Cluster	DO Case	NO Case	LO Case
Outliers	22.7%	8.4%	12.7%
1st	77.3%	91.6%	87.3%

Table 6.8: SVC sample distribution

obtain a first approximation to adjust boundaries for the number of initial, existing clusters for all of the techniques. In any case, for every specific clustering methodology, the most suitable number of clusters is finally fixed analysing graphic representations of different performances around the number facilitated by the SOM tool. These graphic representations are based on validation techniques that allow identifying averages and deviations of similarity between centroids and input samples (see Appendix B.1.4, Fig. B.6).

It is worth commenting on the fact that some significant aspects are shown by the rest of clustering methods. For example, note that Fuzzy C-means clustering always identifies only two main clusters (other clusters have insignificant representation) regardless of the studied case. As far as K-means-based partitioning is concerned, the most notable aspect with regard to the adjustment and configuration is that they obtain the best results when measures of similarity based on Pearson's correlation coefficient are used. In this respect, Graph clustering also finds the best performances when using a similarity function based on correlation coefficients, in addition to an asymmetric graph model. Unlike the other approaches, Graph Clustering finds 11 clusters in the DO case (however, only the 5 most representative ones are shown in the table). In the NO case, it discovers 5 clusters, and 3 in the LO case (Table 6.4).

Finally, independent of the configuration, SVC identifies only one pattern in all cases and rejects a certain, variable amount of samples as outliers.

6.2.3 Evaluation Methods

For the evaluation and comparison of the different clustering techniques, we have developed two different indexes based on the representativeness of the discovered patterns: the *Pattern Credibility Index* (C) and the *Method Credibility Index* (C'). Both are based on a *matching measure* (i. e. similarity metric) stated by the Hamming distance, Equation 6.3.

$$H(\mathbf{x}, \mathbf{y}) = \sum_i x_i \oplus y_i \tag{6.3}$$

Pattern Credibility Index (C)

The Pattern Credibility Index calculates how many input samples match with a specific discovered representative. Later on, it compares the value corresponding to this specific representative with the values obtained by other output representatives found by all methods included in the comparison. For the index, we consider a perfect correspondence between the selected pattern and the input samples, i.e. $H(\mathbf{x}, \mathbf{y}) = 0$, but there are also some deviations: one, two, three or even four bits (hours), i.e. $H(\mathbf{x}, \mathbf{y}) \leq 1$, ..., $H(\mathbf{x}, \mathbf{y}) \leq 4$. The following equation allows comparing the patterns and gives an assessment of credibility for each one:

$$C_i = \frac{f_{E_i}}{f_{E_M}} + \alpha \frac{f_{1h_i}}{f_{1h_M}} + \beta \frac{f_{2h_i}}{f_{2h_M}} + \gamma \frac{f_{3h_i}}{f_{3h_M}} + \delta \frac{f_{4h_i}}{f_{4h_M}} \tag{6.4}$$

where C_i stands for the *credibility of pattern* i. f_{E_i} corresponds to the number of input samples that match pattern i in the sub-test with exact matching; $f_{1h_i}, f_{2h_i}, f_{3h_i}, f_{4h_i}$ are the same for sub-tests with allowed tolerance. $f_{E_M}, f_{1h_M}, f_{2h_M}$, f_{3h_M}, f_{4h_M} are the maximum numbers obtained by any pattern in each sub-test. The sub-test is marked by the tolerance sub-indexes: $E, 1h, 2h, 3h, 4h$. Finally, the coefficients $\alpha = 0.96; \beta = 0.92; \gamma = 0.86; \delta = 0.83$ are used to penalise the tolerance of each sub-test in accordance with the allowed similarity loss.

The process can be best explained by an example (Figure 6.9). In the LO case, the Fuzzy method discovers two patterns $p1$ and $p2$; both SOM and XSOM obtain the same patterns, namely $p1$, $p2$ and $p3$; K-means detects $p3$, $p4$ and $p8$, while K-means with Repeated Bisection identifies $p4$, $p5$ and $p8$; Graph method detects the patterns $p1$, $p6$ and $p7$, and, finally, SVC receives only $p7$ as a pattern. There are 11 input samples that exactly match $p1$, thus $f_{E_1} = 11$. The pattern with the maximum number of coincidences is $p7$, with $f_{E_M} = f_{E_4} = 27$. Next, the same operation is repeated allowing 1, 2, 3, and 4 bits of difference between patterns and samples in the matching assessment. The penalty coefficients exist to represent that as long as differences are permitted, the credibility of the pattern (and, thus, its reliability or usefulness) decreases.

This way of comparing patterns is not entirely fair because central patterns (patterns that exist inside a crowded area) take advantage of their central position. The same applies to crowded patterns (e. g. big clusters). In any case, if clustering methods are evaluated by this index, it can be expected that a good method obtains high values in all its patterns (mainly in the biggest ones), taking into account that, as long as a cluster has less population, it is normal that the credibility value also decreases.

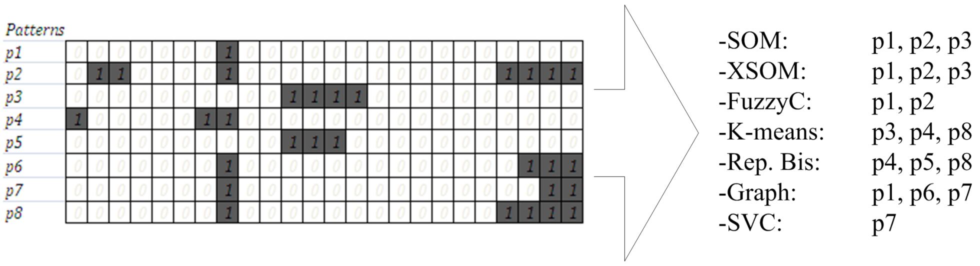


Figure 6.9: Patterns and clustering techniques in the LO case.

Method Credibility Index (C')

Unlike the first credibility index which is only dedicated to patterns, the Method Credibility Index addresses a chosen method. It assesses the number of samples that match each pattern divided by the number of expected samples grouped by the respective cluster. It follows the equation

$$C'_j = \sum_k coef_k \frac{1}{N_j} \sum_i^{N_j} \frac{F_{kji}}{X_{ji}} \tag{6.5}$$

where C'_j stands for the *credibility of method* j. k marks the kind of sub-test regarding the tolerance or number of admitted errors (0, 1, 2, 3 or 4). Therefore, $coef_k$ corresponds to $1, \alpha, \beta, \gamma$ or δ, again proportionally penalising the tolerance of each sub-test. N_j is the number of patterns of the current method. F_{kji} is the number of obtained samples that match the pattern i of the method j in the sub-test k ($F_{ki} = [f_{E_i}, f_{1h_i}, f_{2h_i}, f_{3h_i}, f_{4h_i}]$). Finally, X_{ji} is the number of samples that pertain to the cluster represented by pattern i in the method j.

An example shall further clarify the meaning of the equation. Taking the previous case (LO), the SVC method is identified by the ordinal number 7. It only finds one pattern, so $N_7 = 1$. Therefore, for this method, equation 6.5 can be expressed as

$$C'_7 = \sum_k coef_k \frac{F_{k71}}{X_{71}} = \frac{f_{E_1} + \alpha f_{1h_1} + \beta f_{2h_1} + \gamma f_{3h_1} + \delta f_{4h_1}}{X_{71}},$$

where $X_{71} = 1117$ (the total number of input samples embraced by cluster 1 of method 7). In an ideal situation, all the samples included into a cluster should be similar to the pattern that embraces them (i. e. their representative).

The test is executed twice. In the second evaluation (set 2 in Table 6.10), the periods of time (vector fields) that are the same in all patterns (from all methods) are not considered. In addition, only exact matching and a one bit difference is allowed.

This alternative way to evaluate credibility of patterns and methods is also unfair because it tends to swell methods that find a higher number of clusters, and methods whose patterns are close to one another. This is due to the fact that as long as *bit differences* are admitted, it is possible that some samples simultaneously add credibility to diverse patterns in the calculations. Furthermore, C' also favours dense and compact clusters and penalises global clusters.

In any case, part of the biases in both credibility methods point to opposite directions, so they are compensated when both indices are assessed together. The other part favours patterns that suit smart home applications, e. g. central clusters (see Section 6.3).

6.2.4 Results and Discussion

On account of the characteristics in the cases explained above, clustering methods are facing three scenarios where clusters exist within a dense cloud of samples, that could be even considered as quite dominant background noise.

	SOM	XSOM	Fuzzy	K-m	RB	Graph	SVC
1st p.	3.67	4.20	3.67	2.94	3.67	2.94	3.55
2nd p.	4.20	3.67	2.80	2.80	2.80	3.08	
3rd p.	3.16	2.84		3.55	1.14	2.23	
4th p.	1.95	3.25		1.81	1.31	1.90	
5th p.				1.05	1.04	4.20	

High/Medium Daytime Occupancy Case – **DO**

	SOM	XSOM	Fuzzy	K-m	RB	Graph	SVC
1st p.	2.40	2.40	2.93	1.86	2.40	0.70	2.84
2nd p.	2.80	2.80	2.17	2.93	0.66	2.84	
3rd p.	4.17	4.17		1.50	0.80	0.52	
4th p.	1.73	1.73		0.64	2.48	2.14	
5th p.				0.08	0.64	0.96	

High/Medium Night Occupancy Case – **NO**

	SOM	XSOM	Fuzzy	K-m	RB	Graph	SVC
1st p.	3.99	3.99	3.99	1.86	1.86	1.88	2.28
2nd p.	2.52	2.52	2.52	4.27	4.27	2.28	
3rd p.	0.60	0.60		0.60	0.80	2.72	

Low Occupancy Case – **LO**

Table 6.9: Credibility of patterns (C)

	SOM	XSOM	Fuzzy	K-m	RB	Graph	SVC
Set 1	1.61	1.92	0.85	2.03	1.80	**2.63**	0.58
Set 2	0.40	**0.48**	0.22	0.37	0.31	0.41	0.16

High/Medium Daytime Occupancy Case – **DO**

	SOM	XSOM	Fuzzy	K-m	RB	Graph	SVC
Set 1	1.46	**1.57**	0.71	0.95	1.01	1.26	0.44
Set 2	0.18	**0.19**	0.05	0.05	0.05	0.05	0.02

High/Medium Night Occupancy Case – **NO**

	SOM	XSOM	Fuzzy	K-m	RB	Graph	SVC
Set 1	0.86	0.92	0.99	0.67	0.70	**1.09**	0.54
Set 2	0.25	0.26	**0.29**	0.16	0.16	0.22	0.07

Low Occupancy Case – **LO**

Table 6.10: Credibility of methods (C')

For the first credibility index C, SOM and XSOM obtain the best partitions, but also Fuzzy C-means shows good evaluations, however, with less clusters identified (see Table 6.9). SVC has trouble going beyond the dense cloud of central points. For the chosen scenario, it is negligible if there is a dominant pattern, and sub-patterns present high similarity with it. Nevertheless, it can lead to unsatisfactory solutions when two or three significant patterns that reside far-off exist. Normal and with Repeated Bisection K-mean methods find valid clusters but they have trouble fixing suitable boundaries. Thus, final pattern shapes of these two methods show signs of randomness. Graph Clustering delivers the most arbitrary results. This is due to the fact that big clusters do not show a fine graph representation. However, for finding out small and dense clusters, Graph Clustering is a fine approach.

The second credibility index C' remarks on this fact and rewards the small, dense clusters discovered by the Graph Clustering method (see Table 6.10). The superiority of ANNs-based methods (SOM and XSOM) over K-means points to a better cluster boundary formulation taking into account local minima. SVC and Fuzzy methods are strongly penalised due to the significant dispersion of samples inside big clusters (i. e. low density). Thus, they are grouping samples with a similar core but with important differences.

Hence, it is possible to discard SVC because of its inability to immerse itself into the problem beyond a superficial approach. Likewise, Graph Clustering does not work well when the case implies clusters embraced in a mesh with ambiguous points. Methods based on K-means partitioning are rough at dealing with ambiguous points. This is due to the lack of global awareness in the algorithm and the high and unsatisfactory dependence to the initial number of clusters. SOM and XSOM perform better classifications. Finally, in spite of finding only two groups, Fuzzy C-means clustering shows the highest robustness. Its patterns always reach the relevant bisection in comparison to patterns found by other methods.

Comparing SOM and XSOM, XSOM's ability to reject outliers lead to better results in cases with 'annoying' outliers. Otherwise, both perform in a similar fashion.

6.3 Comparison of Clustering Techniques: Performance in a Simulated Scenario

This test has the same objectives as the test previously depicted in Section 6.2. Now, the same clustering techniques are checked, compared and evaluated but using a simulated environment that reproduces the conditions expected for a smart home application. This application or service – Setpoint Temperature Control Application – can work using profile-based controllers.

6.3.1 Scenario and Test Description

The 'Scenario and Test Description' presented in Section 6.2 is equally valid in this section. In addition, the selected simulated scenario develops the Setpoint Temperature Control Application introduced in Chapter 4, and is widely described in Appendix B.1. Simulations utilize the building model and weather data introduced as options by default in the appendix.

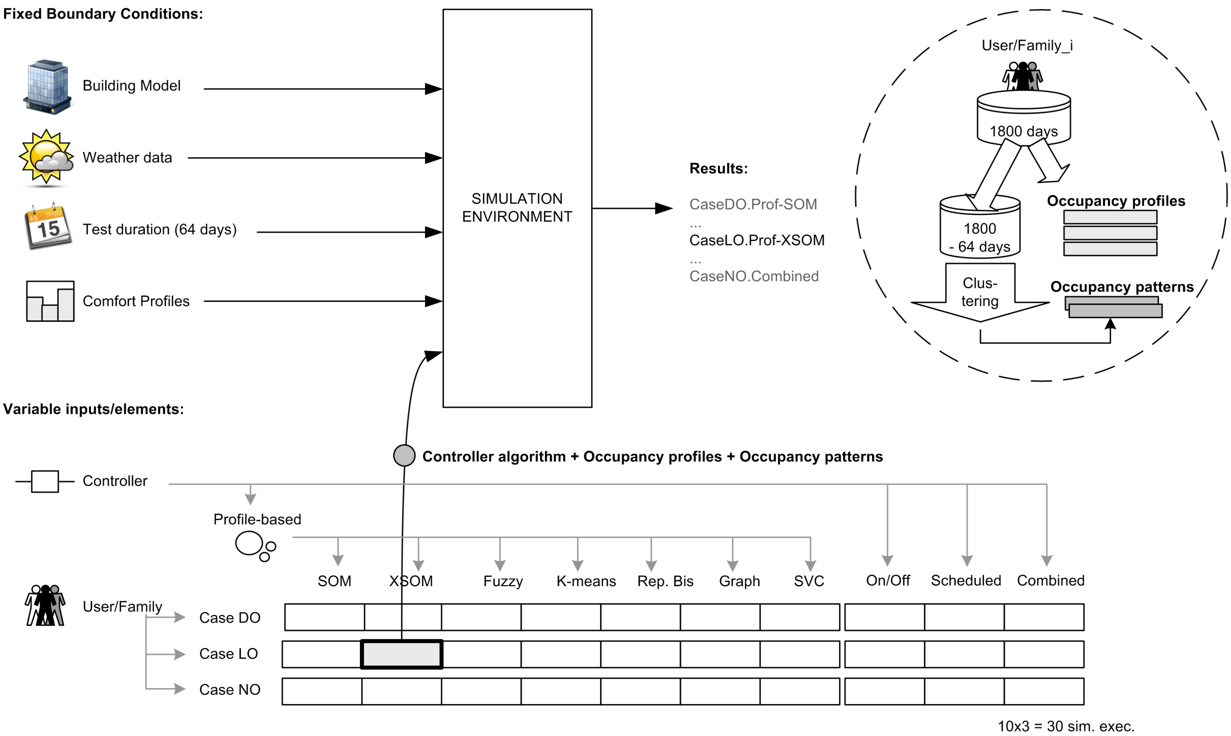


Figure 6.10: Comparison of Clustering Techniques for the Setpoint Temperature Control: test design

The profiles selected to simulate occupancy are chosen at random from the whole database. Thus, for each behaviour case, 64 occupancy daily profiles are randomly selected to simulate an instantaneous occupancy. Figure 6.10 shows a schematic overview of the undertaken simulations.

6.3.2 Test Execution and Evaluation Method

The clustering methods under test are the following: Self-organising Maps, Exclusive Self-organising Maps, Fuzzy C-means, K-means, K-means with Repeated Bisection, Graph Clustering and Support Vector Clustering.

In order to have a good reference frame based on the performances of controllers not based on profiles, three additional strategies are also checked: on/off, schedule-based control, and a mixture of both (Appendix B.1.3). Thereby, the complete experiment consists of 30 simulations. The outputs of the simulations are consumption, setpoint temperatures, real indoor temperatures and predicted occupancy. By means of simulation performance indices the methods are benchmarked. The utilised indices are: Consumption (Q), Temperature difference (dT), Time in Comfort (TiC), Time to Comfort (TtC) and Percentage of Correct Occupancy Prediction (POk). Explanations concerning indices are provided in Appendix B.1.5.

As a general rule, the three cases are less friendly than expected in a real situation. This is due to the fact that we are considering all the days for five years together in the same flat, we ignore them if they all correspond to the same family/users, and we assume that the same habits are approximately kept during the whole period. Furthermore, as we have referred to in Chapter 3, in a real implementation, databases are divided into seasonal and daily sets (since people usually hold trends with a weekly cadence). In any case, we have kept this performance so that clustering methods face a more difficult scenario where comparison results can be discriminant enough.

6.3.3 Results and Discussion

Simulations results are shown in Table 6.11.

As far as classic strategies are concerned, the simple *on/off* strategy fixes the minimal possible energy consumption and also informs us about the expected comfort values for this minimum. Nevertheless, even in the LO case (which represents the most 'delicate' one for profile-based strategies according to the outcomes), profiles are correctly predicting about 75% of time, obtaining more than a 35% increment of comfort time with much lower *difference of temperature* values (dT), paying an increment of consumption of around 20%. Thus, compared to simple strategies, all different clustering methods enable controllers to reach better performances in the three cases.

Focusing on the comparison among profile-based controllers, it is important to notice that the algorithm designed to control as well as the control scenario itself smooth the influence of patterns, sometimes even filtering defects in the clustering process. This fact tends to equalise the obtained solutions irrespective of the underlying clustering methodology. That is one of the reasons why all methods seem to reach acceptable, similar performances. In this respect, the worst case is shown by SVC clustering, which moves to be inadvisable in some polarised situations (e. g. two compact and distant clusters).

If we demand high accuracy, we can see that the most robust and optimised response is offered by the Fuzzy C-means method, closely followed by XSOM and SOM.

It is possible to reason that the percentage of time that the system correctly predicts occupancy (POk) should be the convincing index to benchmark the best method in simulations. However, not all the vector fields (periods of time) in the profile shape have the same importance. For instance, to correctly predict trigger-times after long absences is more decisive than to guess occupancy in times that are usually not transitory. Thus, a higher POk does not necessarily mean better performances. Indeed, the better sensitivity detecting rising and falling edges is one of the reasons that sets Fuzzy, SOM and XSOM clustering apart from the rest.

	Q (Wh)	dT (°C)	TiC (h)	TtC (h)	
On/off	1338.0	0.28	561.0	118.6	
Scheduling	1549.0	0.56	619.0	61.4	
Comb.	1423.0	0.18	582.0	97.7	
	Q (Wh)	dT (°C)	TiC (h)	TtC (h)	POk
SVC	1422.5	0.18	591.6	88.4	77.1%
Graph	1432.4	0.16	600.3	79.8	75.3%
Rep. bi.	1434.7	0.15	605.7	74.3	78.4%
Direct k-m	1438.4	0.15	605.6	74.4	77.1%
Fuzzy	1434.7	0.15	605.7	74.3	78.4%
XSOM	1432.1	0.15	601.7	78.3	79.1%
SOM	1432.4	0.16	600.2	79.8	79.2%
Best value	*lowest*	*lowest*	*highest*	*lowest*	*highest*

High/Medium Daytime Occupancy Case – **DO**

	Q (Wh)	dT (°C)	TiC (h)	TtC (h)	
On/off	1314.0	0.22	625.0	94.3	
Scheduling	1554.0	0.39	639.0	80.3	
Comb.	1422.0	0.14	650.0	69.7	
	Q (Wh)	dT (°C)	TiC (h)	TtC (h)	POk
SVC	1396.9	0.12	663.8	56.3	68.8%
Graph	1395.2	0.12	663.6	56.4	71.9%
Rep. bi.	1398.0	0.12	665.3	54.8	70.8%
Direct k-m	1398.0	0.12	665.3	54.8	71.1%
Fuzzy	1400.8	0.11	667.2	52.9	75.5%
XSOM	1398.6	0.11	666.8	53.2	74.7%
SOM	1398.6	0.11	666.8	53.2	74.7%
Best value	*lowest*	*lowest*	*highest*	*lowest*	*highest*

High/Medium Night Occupancy Case – **NO**

	Q (Wh)	dT (°C)	TiC (h)	TtC (h)	
On/off	952.0	0.36	232.0	159.9	
Scheduling	965.0	2.07	94.0	298.4	
Comb.	1139.0	0.21	298.0	93.6	
	Q (Wh)	dT (°C)	TiC (h)	TtC (h)	POk
SVC	1209.8	0.06	359.9	32.1	72.9%
Graph	1206.3	0.07	358.0	34.0	72.9%
Rep. bi.	1201.2	0.08	353.7	38.3	77.1%
Direct k-m	1202.4	0.08	352.3	39.8	76.0%
Fuzzy	1193.2	0.06	362.5	29.5	76.6%
XSOM	1195.7	0.06	360.1	31.9	74.0%
SOM	1195.7	0.06	360.1	31.9	74.0%
Best value	*lowest*	*lowest*	*highest*	*lowest*	*highest*

Low Occupancy Case – **LO**

Table 6.11: Simulation results

6.4 Habit Sensitivity and Context Interpretation

The next test explores the effect of users' habit regularity comparing performances of different user models, from one that shows strict stability to the other extreme, whose behaviour can be designed as chaotic. As far as the clustering phase is concerned, input vector data sets cover from high-density, well-defined clusters (stable user) to low-density, bad-defined clusters (chaotic user).

The analysis pursues the following aims:

1 To obtain a better appraisal of the effect of user behaviours in the comfort and energy performance of a representative home application.

2 To study how well clustering methods can represent different models of behaviour and offer correct interpretations of such behaviours. In addition, to look for clustering features that are suitable for home control applications.

In short, both objectives will help to refine and improve the conceptual approach (1) and the implementation design (2) of habit-based applications and services.

6.4.1 Scenario and Test Description

For the analysis, 10 different users (habit-regularity models) have been developed, covering from high-regularity levels to chaotic behaviours. Again, tests are run using the Setpoint Temperature Control Application introduced in Chapter 4, under the simulated environment described in Appendix B.1. In short, simulations check the performance of the building submitted to changes in the controller and the users. Therefore, each habit-regularity model is tested with five different controllers, keeping the other conditions, variables and parameters the same. Again, the chosen phenomena to represent users' behaviour is occupancy.

Each habit-regularity model entails 30 occupancy profiles (30 days). The first 25 days are utilised as past days for training and the adjustment of controllers, whereas the last 5 days are intended for testing and evaluation. Figure 6.11 shows a schematic overview of the test.

6.4.2 Test Execution and Evaluation Methods

Specific features for the tests developed in this section are as follows. Setpoint temperatures are set to 22°C for normal comfort, 19°C in sleeping times and 17°C is set as the setback temperature. Controllers use the next strategies: on/off, scheduled, combined and based on profiles, as depicted in Appendix B.1.3. The strategies based on profiles deploy the following clustering techniques: XSOM and Fuzzy (see Appendix B.1.4). We select these two clustering methods due to the fact that they obtained the best performances in previous clustering comparison tests, Sections 6.2 and 6.3. On the other hand, the first simulated day has been removed in the results calculation so as to avoid transitory states (Figure 6.12). Finally, the indices used to compare performances are Percentage of pleasant time (Pt) and Energy-cost of pleasant time (Ec), described in Appendix B.1.5.

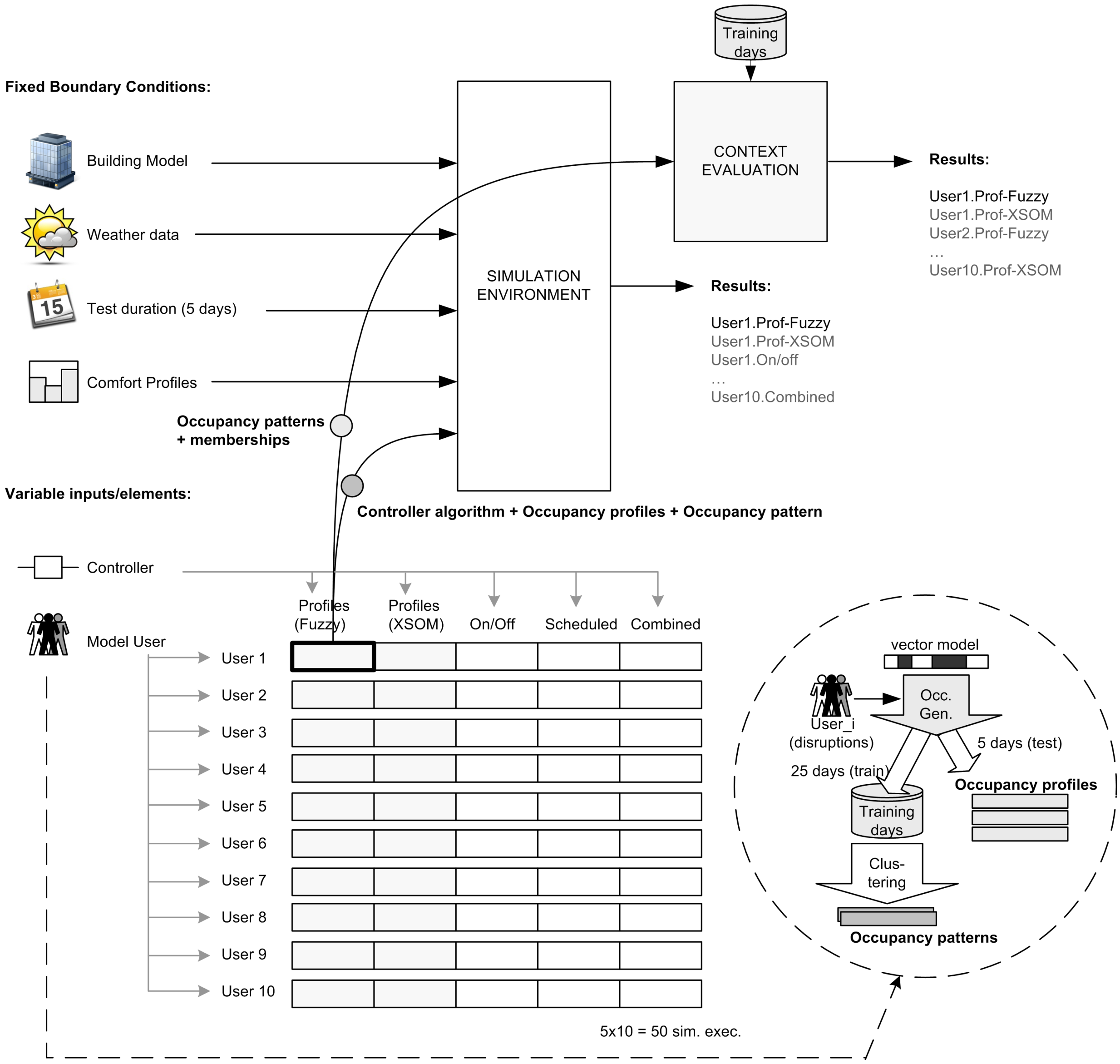


Figure 6.11: Habit Sensitivity and Context Interpretation: test design.

Normalised Euclidean distances are used for assessments of similarity. Figure 6.13 shows the correspondence between differences in two occupancy profile vectors and their normalised distance.

Unlike other experiments shown throughout the dissertation, occupancy is not obtained from real databases, but modelled (see Appendix B.1.2). Figure 6.14 shows, in dark gray, the *original occupancy vector* used to design most of the habit-regularity models. It represents a possible occupancy of a flat for a single user or couple. The flat becomes empty after 7am and people arrive home around 6pm.

The addition of diverse types of alterations with variable frequency into the basis model generates a wide amount of cases. Among them, the cases shown in Table 6.12 have been selected to illustrate the main conclusions concerning the clustering techniques comparison and the sensitivity analysis.

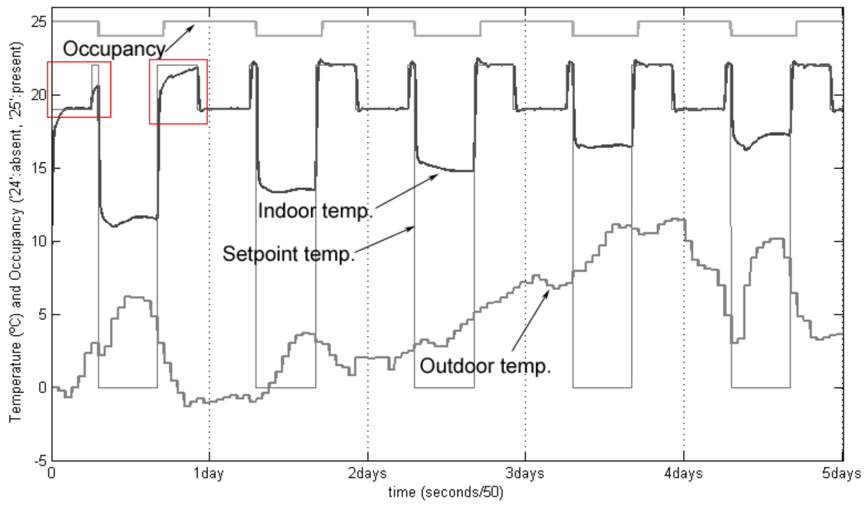


Figure 6.12: Temperatures (setpoint, indoor, outdoor) and occupancy in "case 1". Note the transitory states of the first day of simulation, emphasised by boxes.

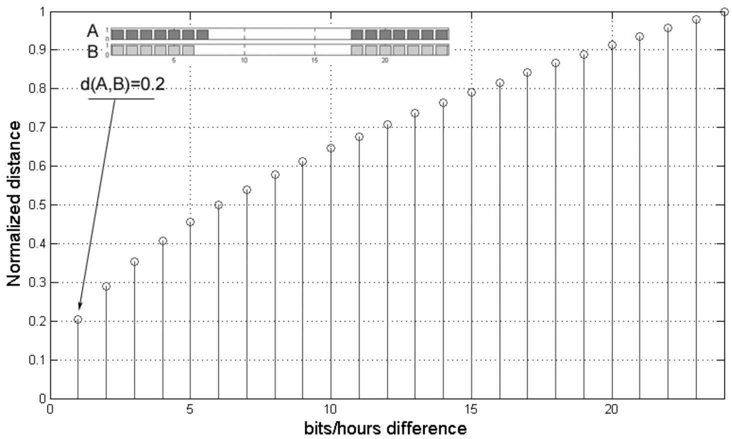


Figure 6.13: Normalised distance as a function of bits/hours. A and B represent two aleatory profiles.

To gain a first impression of the level of dissimilarity between the original vector and the generated samples for each case (k), a *percentage difference* based on the Hamming distance is stated as follows:

$$d = 100 \left(\frac{\sum_{i=1}^{H} \sum_{j=1}^{N} \overline{case_k(i)} \oplus case_k(i,j)}{HN} \right) \tag{6.6}$$

where H represents the hours of the day (24), N is the amount of simulated days (30). $\overline{case_k(i)}$ is the average vector (day) of the case k. It usually coincides with the main clustering representative pattern, but not always (in cases with very high disruptions it can be different). The XOR ($\oplus$) results in 1 whenever an hour/bit of a simulated day does not fit the average or expected model day.

Case	E and/or L	Short UPA	Medium UPA	Long UPA	Comments
1	–	–	–	–	Perfect regularity model, 0.0% difference.
2	x	–	–	–	5.1% difference.
3	–	x	x	x	38.7% difference.
4	–	x	–	–	11.0% difference.
5	–	–	x	–	26.5% difference.
6	–	–	–	x	29.0% difference.
7	x	x	x	x	42.6% difference.
8	not specif.	n.s.	n.s.	n.s.	At random hour by hour, 43.5% difference.
9	x	x	x	x	Additional trend, 30.0% difference.
10	–	–	–	–	Perfect regularity model 2, 0.0% difference.

E and/or L stands for earliness or lateness in leaving and arrival times.
UPA are unexpected presences and absences (Appendix B.1.2).

Table 6.12: Occupancy model cases.

Unlike other cases, 'case 8' is not modeled using the *original occupancy vector*. Instead, uniformly distributed pseudo-random numbers are used to generate hourly occupancy.

In 'case 10', a new model is used as a template. This second model shows more alternated periods of presences and absences (every two or three hours there is a change in the occupancy).

In order to avoid confusing interpretations of percentage values, it is worth remarking that, for example, the 38.7% of difference caused by randomness in 'case 3' indicates a case with very frequent alterations, in a way that it is very difficult to detect the original pattern in the samples even for the human eye (Figure 6.14, picture on the right side). As a comparison, an absolute random case – 'case 8' – shows 43.5% of percentage difference.

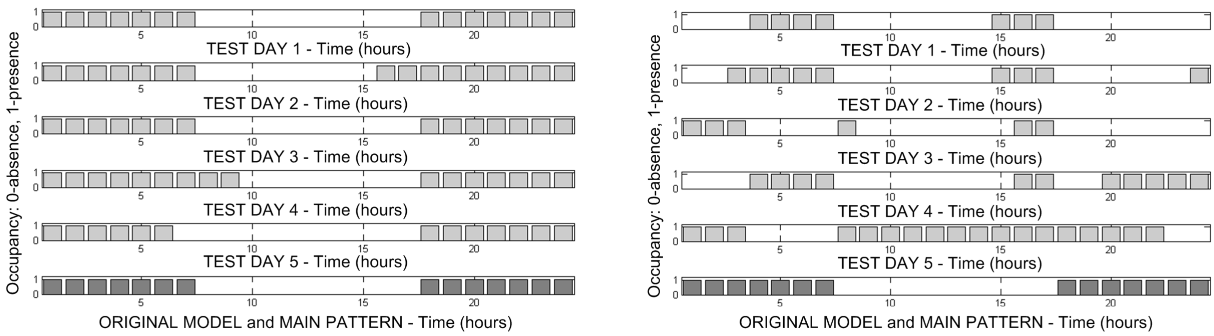


Figure 6.14: Original models, main patterns and five test days for case 2 (left) and case 3 (right).

6.4.3 Results and Discussion

Despite chaotic behaviours, Pt values seem to be acceptable in all cases and regardless of the underlying controller. This is due to the application's nature and boundary conditions (e. g. thermal inertia of the buildings) as well as design constraints (e. g. the minimum time considered for presences/absences is 1 hour). Nevertheless, a $Pt = 90\%$ means that people feel discomfort in one of 10 hours at home. Similarly, the same factors can also make it difficult for controllers to achieve the best ratings even if they worked perfectly. For instance, temperature changes from sleeping and daytime modes (22°C and 19°C respectively) entail a transient time that causes inevitable penalisations in the performances as *time in comfort* conditions are not fulfilled during the transient time.

For that reason, interpretation of indices is not free of uncertainty or delusion. Indices are best used through comparisons between controllers where boundary factors draw the same influences for every strategy. Therefore, comparisons between cases must be carefully carried out.

Tables 6.13 and 6.14 show the results of simulations. As long as the objective of the controllers is to obtain a Pt as high as possible with an Ec as low as possible, it is easy to realise that controllers based on habit profiles always perform best. More interesting situations are 'case 7' and 'case 8' where the scheduled strategy receives a best comfort time percentage, however at a higher energy cost. The lack of regularity in these cases makes flimsy prediction capabilities. In any case, it is difficult to establish a best method in both situations, and it depends on how the commitment between energy and comfort is considered.

Controllers based on profiles show good flexibility in front of the announced noise: earliness and lateness in arrival and/or leaving times; short, medium and long presences and absences; and combinations. The existence of additional patterns or sub-patterns is well solved by controllers based on profiles, whereas the scheduled strategy proves to be vulnerable against long-term perturbations ('case 6' and 'case 9'). As a general rule, the scheduled strategy is not flexible against lacks of regularity. In those cases, it can show good comfort performances if the schedule is benevolently fixed with regard to low, unknown or unpredictable occupancy times, accepting the consequent extra costs in energy as inevitable.

As far as only the performance indexes are concerned, the comparison between the controller based on XSOM clustering and the controller based on Fuzzy clustering is not easy to solve. In general, both obtain good results, often the same. XSOM clustering seems to be more aware or sensitive about sub-patterns or non-main patterns, but is also prone to be astray by them. This discussion can be clarified, bearing in mind how each tool abstracts the habit-regularity context (see below).

Compared to the other methods, the on/off controller improves as long as user behaviours are more and more unpredictable. It is down to the fact that its performance is not directly affected by habit-regularity but by the number and length of daily presences and absences.

Finally, among the classic options, the combined controller always obtains satisfactory performances, close enough to profile-based controllers.

Assessments of the Context

Data shown in Table 6.15 offers the main indicators that allow the controller to know what is happening concerning occupancy habits. A first look at Table 6.15 can suggest that the Fuzzy controller receives better ratings, it seems to be more reliable (i. e. main patterns embrace more input samples and the average distances take more extreme values than in the XSOM method). However, a closer look reveals that XSOM provides more stable and coherent information. Moving away from trivial situations ('case 1' and 'case 10'), it is worth mentioning the results case by case.

'Case 2' is a situation that is easy to solve for the controllers as it represents well-defined trends. Both Fuzzy and XSOM obtain the same patterns, but they do not receive the same clustering. The fuzzy method seems to obtain a best main cluster here, where XSOM notices the influence of a second (in that case irrelevant) pattern more strongly.

Case 1 ($Occ = 58\%$).

	XSOM	Fuzzy	On/off	Sched	Comb
$Pt\|_{\%}$	95.0	95.0	91.5	95.0	92.0
$Ec\|_{KWh}$	1.49	1.49	1.51	1.49	1.50

Case 2 ($Occ = 61\%$).

	XSOM	Fuzzy	On/off	Sched	Comb
$Pt\|_{\%}$	95.9	95.9	92.8	89.2	93.0
$Ec\|_{KWh}$	1.44	1.44	1.45	1.50	1.44

Case 3 ($Occ = 42\%$).

	XSOM	Fuzzy	On/off	Sched	Comb
$Pt\|_{\%}$	90.5	90.7	79.6	39.4	89.2
$Ec\|_{KWh}$	2.09	2.07	2.21	2.91	2.08

Case 4 ($Occ = 52\%$).

	XSOM	Fuzzy	On/off	Sched	Comb
$Pt\|_{\%}$	90.0	90.0	85.0	89.7	86.6
$Ec\|_{KWh}$	1.66	1.66	1.74	1.72	1.67

Case 5 ($Occ = 58\%$).

	XSOM	Fuzzy	On/off	Sched	Comb
$Pt\|_{\%}$	93.1	93.8	90.0	83.3	90.8
$Ec\|_{KWh}$	1.45	1.48	1.61	1.63	1.48

Case 6 ($Occ = 49\%$).

	XSOM	Fuzzy	On/off	Sched	Comb
$Pt\|_{\%}$	94.1	94.1	85.8	45.7	90.7
$Ec\|_{KWh}$	1.68	1.68	1.80	2.51	1.73

Case 7 ($Occ = 43\%$).

	XSOM	Fuzzy	On/off	Sched	Comb
$Pt\|_{\%}$	90.3	86.6	77.2	97.4	86.0
$Ec\|_{KWh}$	2.14	2.25	2.30	2.30	2.25

Case 8 ($Occ = 53\%$).

	XSOM	Fuzzy	On/off	Sched	Comb
$Pt\|_{\%}$	82.7	84.2	73.4	87.2	80.8
$Ec\|_{KWh}$	1.76	1.78	2.08	1.81	1.77

Case 9 ($Occ = 64\%$).

	XSOM	Fuzzy	On/off	Sched	Comb
$Pt\|_{\%}$	95.4	94.1	92.0	48.9	93.7
$Ec\|_{KWh}$	1.42	1.43	1.50	2.00	1.47

Case 10 ($Occ = 63\%$).

	XSOM	Fuzzy	On/off	Sched	Comb
$Pt\|_{\%}$	89.6	89.6	88.7	87.0	89.6
$Ec\|_{KWh}$	1.43	1.43	1.49	1.58	1.46

Table 6.14: Simulation Results, cases 6 to 10.

			XSOM	Fuzzy
	d	cl	$mem - \overline{dist}$	$mem - \overline{dist}$
Case 1	00.0%	1st	100% – 0.00	100% – 0.00
Case 2	05.1%	1st	50% – 0.20	56% – 0.11
Case 3	38.7%	1st	55% – 0.45	64% – 0.20
Case 4	11.0%	1st	67% – 0.20	100% – 0.30
Case 5	26.5%	1st	45% – 0.31	100% – 0.50
Case 6	29.0%	1st	59% – 0.47	44% – 0.11
Case 7	42.6%	1st	43% – 0.40	100% – 0.64
Case 8	43.5%	1st	41% – 0.54	56% – 0.63
Case 9a	30.0%	1st	56% – 0.12	56% – 0.15
Case 9b	30.0%	2nd	36% – 0.26	36% – 0.70
Case 10	00.0%	1st	100% – 0.00	100% – 0.00

d: percentage difference
cl: pattern order based on the cluster size
mem: % of input samples embraced within the cluster (membership)
$\overline{dist}$: average distance of the cluster members regarding their pattern, Equation 5.10

Table 6.15: Clustering, reading about the context.

In 'Case 4' (not very demanding), both controllers obtain the same main pattern again. XSOM finds additional useless patterns but, on the other hand, it establishes the boundaries of the main cluster better. Here, the differences are not important for the application (it feels comfortable with big, medium-density clusters), but they are providing different – but both reliable – interpretations of the context. Fuzzy states that all past days are around the pattern with an average difference of three hours, whereas XSOM establishes that about 70% of past days show an average difference of two hours. XSOM seems to provide a more accurate reading.

'Case 9' is a case with good regularity and is easy to solve, except for the drawback of the additional pattern. XSOM draws the situation perfectly and recognises both patterns accurately. The fuzzy controller has trouble with additional patterns as a general rule, it finds them, but is usually distorted or confused by the fuzzy boundaries of the clusters and by equivocally accepted outliers. This has an effect on the context evaluation: in Table 6.15, the value regarding the average distance of the second pattern cluster is not realistic.

Therefore, the Fuzzy approach has difficulty correctly reading scenarios with density differences and dealing with outliers (i. e. cases where habit-regularity is very low). In such situations, it usually discovers main average patterns that fit the application requirements, but it seems to be confused by randomness in the context appraisal.

On the other hand, XSOM clustering shows stability and coherence when it assesses the level of reliability of its patterns. If we consider that in 'case 9' the value d would be between 5.0% and 10.0% if we separate both existing patterns, there is always a correlation between d and $\overline{dist}$ that does not exist in the Fuzzy method with such robustness. In other words, the XSOM controller can deploy confidently $\overline{dist}$ values to know the habit-regularity level of users, even better than considering d values, used to build the input samples. In order to value this higher reliability of XSOM's $\overline{dist}$ instead of d, note the ratings of XSOM in the worst situations ('case 7' and 'case 8'). Both have similar percentage differences (d), but 'case 7' is built over the basis of an almost lost pattern, whereas the other one is absolutely generated at random. XSOM detects it (lower $\overline{dist}$ for 'case 7'), whereas Fuzzy $\overline{dist}$ is clearly confused and the percentage difference index (d) is not useful to be aware of this difference in such demanding cases.

Impact of habits in the performances

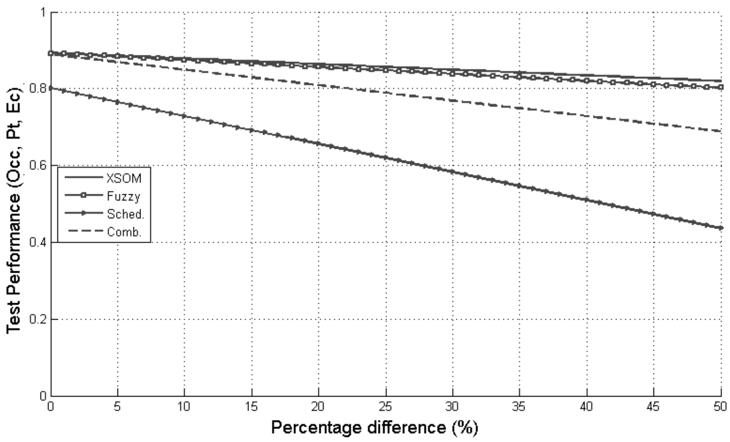


Figure 6.15: Controllers performance as a function of percentage difference.

Figure 6.15 shows linear regressions that relate to the percentage difference with a normalised performance index that weights equal comfort time variations (Pt) and energy cost variations (Ec). The contribution of occupancy ratings is also considered in the calculation, because as long as the average occupancy decreases, Ec tends to increase. Percentage difference has been stated above as a rough way to assess the level of habit-regularity: the more the expected behaviour is lost (i. e. unexpected) the more the percentage difference increases. Apart from comparing the controllers again, Figure 6.15 clearly shows a degenerative trend caused by chaotic behaviour. Noteworthy, keeping a good habit-regularity allows the improvement of comfort while keeping energy consumption low.

'Case 10' is given to compare perfect habit-regularity situations where the differences are in the patterns themselves. Expected short absences and presences have effect on comfort assessments without noticeable penalisation in the energy cost of comfort time. Therefore, in the current application, transition times mainly affect the comfort achievement.

It is very difficult or even impossible to establish an accurate relationship between habit-regularity loss and performance degradation, as the effect of unexpected alterations depends strongly on the specific features of the scenario and when they happen. As a rough notion and forcing linear relationships, a loss of 5% habit-regularity causes a decrease between 0.5% and 2.5% of comfort, and an energy efficiency degradation between 0.2% and 3.0% of the energy usage. Obviously, these assessments depend on the employed control strategy (they are not meaningful for the on/off controller as it is not based on any kind of prediction).

6.5 Impact of Outliers (Profiles for Building Calculations)

The next analysis checks the convenience of XSOM clustering for pattern discovery in a behavioural profiling case where outliers are existent. The XSOM performance is compared with the results obtained by SOM and K-means clustering. In addition, this section intends to study the possible effect (disruptions) of outliers in the clustering task.

Unlike the previous analysis of this chapter, addressed to cases where profiles are intended to represent a particular user or group of users (mainly for control applications), the present test consists of a case where profiles are used to discover community models for an application oriented to energy building calculations (Section 5.1.2).

6.5.1 Scenario and Test Description

The conducted test requires the application of clustering in two different phases.

1. The input vectors for the first clustering phase are water consumption profiles obtained from Leako databases (Appendix B.2.1), and SOM is the clustering technique utilised in this step (Appendix B.1.4). In this step, every dwelling of the database is processed separately by the clustering algorithm. Therefore, it is used to summarise the available information collected for 5 years and generate profiles that represent the yearly behaviour of every specific dwelling (user/family). Obviously, clustering processes discover more than one main pattern in each analysed dwelling, but only the main, crowded pattern of each dwelling is considered for the next phase, and provided it covers over 40% of input samples. The rest is discarded.
2. Later on, all the output patterns or centroids obtained from the first phase are taken as input vectors by a second clustering process. The objective of the second phase (the process under test) is to find clusters and representatives within he different dwellings (i. e. super-patterns or community representatives). The clustering tools deployed and compared in the second phase are: SOM, XSOM and K-means.

Note that we consider that outliers have a negligible influence in the first clustering phase. The experience acquired with the test developed in previous sections points out to corroborate this assumption most of the time.

6.5.2 Test Execution and Evaluation Methods

As far as the second clustering phase is concerned, different configurations of XSOM are tested executing a sweep of *tolerances* and assessing the obtained performances (the tolerance parameter is explained in Appendix B.1.4). In short, *tolerance* is a parameter that determines the sensitivity of the clustering tool to consider a sample as an outlier and to reject it. XSOM performances are compared with SOM and K-means clustering solutions later on. Except for the tolerance variation, the rest of the parameters have been equivalently adjusted for SOM and XSOM. The application demands large spherical and globular clusters with high representativeness (marginal dense clusters are not important). The initial number of clusters has been fixed to 5 according to the maximum desired for the application, and taking into account that, the SOM method does not present noticeable variations changing the number of initial clusters between 5 and 10. The similarity function is based on Euclidean distance.

With regard to K-means method (using CLUTO tools [Kar03]), the best performance is reached applying direct K-means methodology, with an initial number of 5 clusters, using Euclidean distances and the I_2 criterion for the optimisation function (Appendix B.1.4). Different parametrisations and other approaches, like Repeated Bisection or Graph arrangements, have also been tested with worse results.

For the evaluation (i.e. the validity of the clustering method), the following outputs in each performance have been studied:

- Number of significant patterns (Ps).
- Number of outliers (nO).
- Form of patterns.
- Number of samples embraced in each cluster (nP_j, where j is the pattern identifier in the respective experiment).
- External similarity, i. e. distances among centroids.
- Internal similarity, i. e. distances and statistical data between each centroid and their embraced samples.

6.5.3 Results and Discussion

Table 6.16 shows a number of results obtained by different tolerance factors. As long as tolerance decreases, the number of outliers grows (Fig. 6.16). In parallel, the number of significant patterns rises. This is due to the disappearance of the outlier distortion, that appears when they are classified and accepted inside clusters. Outliers move the gravity center of the group and, thus, decisive differences between close elements are ignored. While SOM only detects one pattern, XSOM configurations detect more significant groups. In addition, the appearance of new clusters is also due to the fact that tolerance adjustment redefines the meaning of 'outlier'. Tolerance fixes how far samples can remain from the cluster center, so outliers are not only errors or samples that distort normal distributions. Summarising, XSOM allows an outlier to be part of a new group of non-clustered members.

Paying attention to the distance values between the representative pattern and the samples in SOM, the existence of outliers can be confirmed (according to the common definition). Distances do not follow a normal distribution but their relationship is close to a logistic distribution, that resembles normal distribution in shape but has a higher kurtosis. Most of the variance is due to odd extreme deviations [Bal92]. A logistic distribution can be expressed as follows:

$$f(x, \sigma, \mu) = \frac{1}{2} + \frac{1}{2} tanh\left(\frac{x-\mu}{2\sigma}\right) \tag{6.7}$$

where μ stands for the mean and σ is the standard deviation. The usual definition of the (excess) kurtosis (γ_2) is shown in the following equation:

$$\gamma_2 = \frac{\mu_4}{\sigma^4} - 3 \tag{6.8}$$

tol_m	Ps	nO	$\%O$	nP_1	nP_2	nP_3	nP_4	nP_5
∞	1	0	0.0%	659	13	4	6	2
158.70	1	23	3.4%	648	4	4	3	2
79.37	2	31	4.5%	538	110	4	1	0
39.68	2	43	6.3%	519	122	0	0	0
15.87	2	60	8.8%	404	220	0	0	0
7.94	3	72	10.5%	352	222	38	0	0
3.97	3	93	13.6%	376	191	24	0	0
1.59	3	201	29.4%	223	192	68	0	0
0.79	3	307	44.9%	184	97	95	0	0
0.32	4	541	79.1%	64	30	26	23	1

Table 6.16: Results in the tolerance sweep test

where μ_4 is the fourth moment about the mean. Whereas in a normal distribution the excess kurtosis equals 0, in a logistical distribution it equals 1.2.

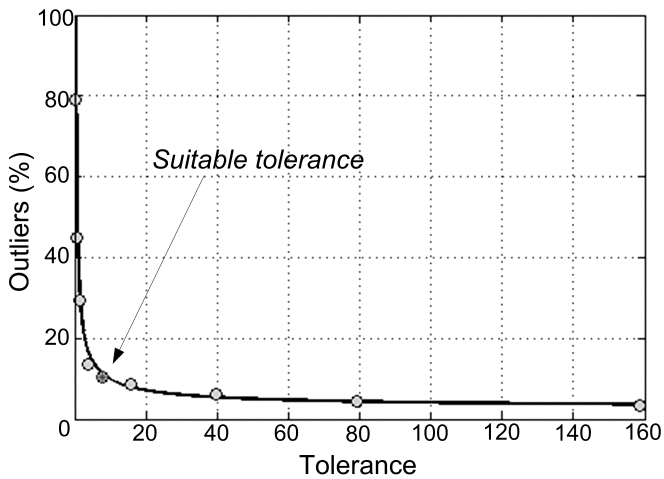


Figure 6.16: Outliers vs tolerance

It is desirable to reach a good tolerance value that rejects a suitable number of outliers. Outliers-tolerance relationship fits an inversely proportional function (Fig. 6.16). According to this relationship, the compromise is reached when the increment of outliers and the increment of tolerance are balanced ($\Delta out = \Delta tol$). The tolerance that matches the previous conditions is shown by the sixth experiment in Table 6.16, resulting in aprox. 10% of outliers in the entire population. In this case, XSOM identifies two patterns. The criterion applied to establish the selected tolerance can be widely discussed because it has been stated without a previous outlier definition and based on a commitment for the tolerance-outlier evolution. In any case, the tolerance sweep, in its medium values, always shows two main patterns that do not differ too much in shape and members.

The most remarkable question concerns the differences between SOM and XSOM comparison. In Fig. 6.17, SOM is compared against the most suitable XSOM performance. While SOM considers a great group (96.3% of input samples) the XSOM classification delivers two groups (with 59.1% and 32.2% elements, respectively). Thus SOM ignores a well-defined group that represents 32% of the population in XSOM classification. Instead of that, it absorbs these elements into the group represented by the big pattern. As it can be noticed in Fig. 6.17, the outlines of most significant patterns in SOM and XSOM experiments are similar, but SOM has higher values in the whole range.

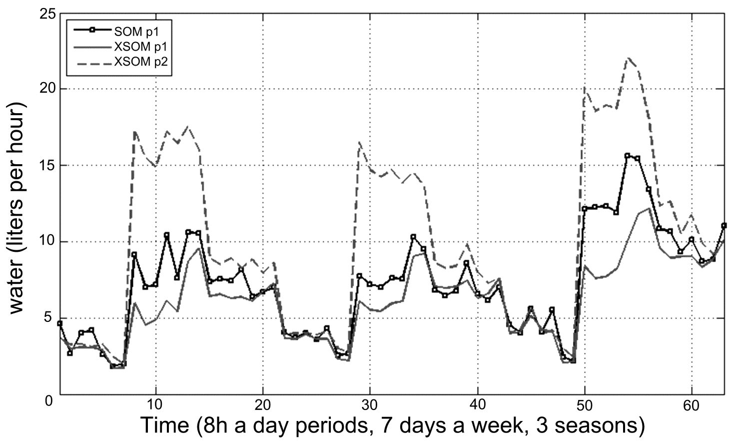


Figure 6.17: SOM and XSOM discovered patterns

In the K-means case, the two main patterns embrace 42.3% and 24.1% of the inputs, respectively. The different performances are assessed based on the average similarities and standard deviations between the obtained patterns and the patterns with their respective clustered samples. In any case, resulting representatives show distorted and even incoherent shapes. We conclude that approaches based on K-means methodology are too poor to properly deal with the current scenario.

Without reliable benchmarks, it is difficult to establish a best method, but it is possible to submit the results to statistical analysis in order to know how close input samples are to their representative patterns (i. e. internal cluster validity). Table 6.17 shows this evaluation. It discloses that elements classified in XSOM are very much closer to their own patterns than elements in SOM or K-means classifications. Therefore, the two patterns obtained in XSOM case are more representative than the others.

		SOM	XSOM	K-means
P1	samples	96.3%	59.1%	42.3%
	distance (mean)	0.61	0.06	0.63
	distance (σ)	5.59	0.07	0.87
P2	samples	–	32.2%	24.1%
	distance (mean)	–	0.12	2.39
	distance (σ)	–	0.17	0.44

Table 6.17: Internal cluster validity in the three cases.

So far, Leako's technical experts confirm the existence of two trends in users. Besides water consumption, this also concerns heating and cooling energy consumption (backed up by the billing). These trends have not yet been carefully studied – this could also be due to geographic matters, building or family sizes, orientation, different systems, etc.

6.6 Acquired Knowledge

The tests developed in this chapter represent a constant effort to refine the design of solutions that utilise clustering for habit abstraction. Thereby smart home control and building calculations are improved. The introduced tests culminate some valuable knowledge and produce findings useful to enhance real implementations and further studies. We can summarise them as follows:

- Considering univariate time series to build profiles, i. e. objects to represent habit and behaviours in smart home control, Euclidean distance is, as a general rule, the most suitable similarity metric (in terms of robustness, stability and reliability) to interpret and deal with the solution space as a whole.
- The intrinsic characteristics of SOM- and Fuzzy-based clustering methods make them preferable to other clustering techniques in the task of discovering optimum patterns within databases that store human behavioural data. Fuzzy techniques are prone to locating the imperative dissections, whereas SOM-base tools present a most accurate classification and an improved reading of the context. In other words, only a strong correlation in time series clustering does not justify the use of similarity measures that consider data correlation rather than the Euclidean distance.
- Clustering validation techniques based on membership, density evaluation, and inter- and intra-clustering distances offer valuable readings of the context. Therefore, discovered patterns (i. e. representative habits) can be soundly assessed in terms of reliability and stability. Application controllers make the most of such context awareness skills, acquiring flexibility to quickly adapt and react to changeable scenarios.
- Profile-based control for smart home applications (at least the ones that follow the proposed methodologies) is robust and preferable compared to other common, classic control strategies. Even in extreme cases (e. g. facing chaotic or unstable behaviours), profile-based control correctly manages the situation, being able to detect undesirable performances.
- Stability in domestic habits helps profile-based controllers to obtain optimised energy and comfort performances.
- Due to the intrinsic nature of smart homes, outlier detection is advisable in most of control applications, but not imperative. However, for applications where the behaviours of a considerable amount of different users must be modelled (e. g. building energy performance calculations), outlier detection and removal appear to be mandatory.
- As a general rule, an empirical criterion for the outlier removal in the conducted human habit modelling states a range within 7-10% of input vectors that must be discriminated as outliers.
- Finally, the previous assessments lead us to state XSOM clustering as the most preferable technique to be embedded in controllers of smart home profile-based services, accomplishing tasks of pattern discovery and context reading.

7 Conclusions & Outlook

7.1 Conclusions

The current dissertation began with the question: *Why smart home control based on behavioural profiles?* Throughout the work we have tried to check if there is a point in this assertion, as well as to discern how this concept can be developed in order to achieve potential benefits derived from the exploitation of behavioural information in the home environment. The main objective has been therefore *to study the suitability of behavioural profiles for smart home control*. Now, we have arguments to confirm such suitability, mainly due to the following reasons:

a) *Improvements in isolated applications*, i. e. applications whose controllers use behavioural profiles acquire enhanced context awareness and are able to overperfom classical strategies.

b) *Improvements in the overall performance*. Behavioural profiles become perfect objects to draw the common context where home applications and services coexist. Simultaneities, synchronies, conflicts and overlaps are better dealt by global managers.

c) *Designs are sensitive to psychological aspects*. Behavioural and habit profiles are key factors in order to understand user-system interactions and optimise systems in terms of adaptiveness, integration and usability. Furthermore, they enclose meaningful information to build up proactive designs and persuasive applications oriented to empower behavioural change and a better fulfillment of users' expectations.

d) *Potential benefits for the urban infrastructure*. Smart homes can participate as prosumers in comprehensive ICT frameworks, which constitute a fundamental part of future smart cities. Within this scope, habit profiles are high-value elements for diverse stakeholders.

These four aspects lean on a set of consecutive subgoals that have been progressively tackled throughout the chapters. We recall them from the introduction of the dissertation, Section 1.3, and check how they have been accomplished.

(1) As a starting point, profiles in smart homes are established as time series' that represent behaviours, uses and habits. The definition of a singular profile is not restrictive, but they are necessarily bound to a temporal description as the time-domain plays a fundamental role in the interpretation of behaviours, the abstraction of habits and the exploitation of profiles by controllers. Beyond this time-dependence, profiles are considered resources that address an *object*

and are identified by a *type* based on the nature of the collected phenomenon and how it is represented (values, scale type, etc.). *Objects* and *types* identified for the home environment are enclosed in a simple classification that makes the management within technological ecosystems easier. In this respect, taking into account subsequent calculations, control phases and shared deployments, the guidelines for the design of profiles are set in a basis of simplicity, scalability and equivalence.

(2) Therefore, the design of profiles is stated in keeping with the requirements of current and future smart homes. Previous to the fulfillment of habitual home functionalities (i. e. security and safety, health, comfort, energy savings and communications), smart homes must improve in terms of integration and adaptiveness. Further benefits can not be achieved without firstly having designs that empower the spreading of home technologies and fight against the current poor uptake. We propose a model that explains and identifies user-system interactions in order to correctly address subsequent smart home designs. A better adaptiveness makes *holistic approaches* mandatory, as they have the capability to understand the whole home context and the combined performance of applications; they are also able to impose global, adaptive models in a general, coordinated fashion. On the other hand, the focus on users forces us to pay special attention to psychological aspects. We follow proactive, AmI designs that consider habits, not only as objects for the context reading, but also as goals, i. e. objects to improve. Such perspective, together with evidences found in the related literature, lead us to conclude that *system informativeness* is one of the main pillars to reach acceptable levels of usability and integration. In addition to this feature, a better adaptiveness is finally achieved when the system is able to automatically check its own performance and adjust the desired level of system-user interaction (feedbacks) based on past experiences. To this purpose we devise *shadow processes*, parallel services in holistic designs that guarantee the satisfactory performance of home applications.

(3) The proposed concepts and models need to be settled on digital ecosystems that can manage the demanded complexity. Multi-agent structures perfectly fulfil the requirements. In this dissertation we build up our designs on a framework of agents where new types of agents, specially devised for profile-based holistic approaches, are stressed; they are: agents for the collection of behaviours (PFG), agents for the pattern discovery (PTG) and agents which embed *shadow processes*. They coexist with other agents (more common in habitual MAS) that adopt additional roles and are distinguished as: control agents, general purpose or referee agents, and agents for the context inference. The coordinated operation of the framework makes top-down perspectives possible and allows it to face the control of the smart home as a whole.

(4) All the developed work so far makes no sense if we do not define applications that make the most of such concepts, models and frameworks. Due to the broad dimensions of the intended field, this subgoal is dealt with twofold. Firstly, a horizontal approach is explored. Here we describe profile-based home applications and services that cope with functionalities related to lighting, control of shading devices, security, air quality, thermal comfort, DHW, energy consumption, self-checking capabilities and informative services. Some of these profile-based services are put together in a case example. Thereby the shared nature of profiles and the cooperative existence between profiles and services are seen.

(5) Secondly, in a vertical approach we take profile-based applications for the management of air quality and thermal comfort, and undertake a deep description. Modules, agents and algorithms are explained, specifying required parameters, inputs, outputs, state diagrams and time analysis.

(6) The evaluation of the vertical approach is carried out checking parts of the proposed subsystems – specifically applications concerning the adjustment of thermal comfort applications.

Profile-based applications compete in a simulated test-bed environment against controllers based on classical, common operative strategies. Simulations highlight the advantages of profile-based approaches, which outperform other options in terms of energy savings and comfort.

(7) Since profile-based control results in better performances, the design of profile-based controllers becomes a focus of study by itself. Their optimisation demands the analysis of the tools in charge of transforming a set of behavioural profiles into context readings. We discover that the context can be accurately inferred by means of habit patterns as reliable as possible, but also using parallel algorithms that evaluate the fitness and representativeness of the discovered patterns. *Clustering* is the methodology selected for the pattern discovery, and *cluster validity methods* are found strongly appropriate to enhance context appraisals and accompany habit patterns with assessments of their reliability (i. e. level of stability or chaos of the current habit), representativeness (i. e. how much of the past behaviour is modelled by the pattern) and domain (i. e. in which periods of time it is the dominant behaviour).

(8) Thus, cluster analysis is considered as the cornerstone to provide controllers with increased context awareness capabilities. The features, implications and uncertainties of clustering methods for human behaviour modelling are widely explored and explained. They are related to the introduced control requirements in order to progressively refine the design of the diverse profile-based applications, controllers and strategies. Mathematical analysis and simulations are used to perform sensitivity analysis and compare different clustering methods and parameters. Among other aspects, the disclosed knowledge stresses the use of Euclidean-based Self-Organising Maps with Outlier Exclusion (XSOM) as the most suitable option to support profile-based control. It is worth underlining that these findings are bound to the frame fixed by the design of the human behavioural containers (i.e. profiles), the control strategies and the applications. Generalisations must be carefully undertaken, considering these results as precedents for more general hypothesis.

7.2 Outlook

After examining the previous points, we can conclude that the conducted work significantly contributes to the fields of research in Smart Homes, AmI and HBA. This work establishes guidelines for the design of future domestic environments that aspire to be more adaptive and better accepted by users. It is not focused on how to optimise or accurately accomplish specific functionalities, but how to reach a fair, smooth, user-sensitive integration of home technologies as a holistic, sound system. The developed proposals do not contribute to the technology push or require futuristic technologies, they involve little extra costs in comparison to habitual HBA installations, being feasible, affordable and flexible, facing different levels of expected functionality.

Finally, given the wide application field explored in the dissertation, throughout the long way covered we have left unexplored or scarcely explored several aspects that are worthy of attention and future research. We specially mention three linked points that are seen as appropriate continuations or complements of the current work.

(1) The lack of comprehensive simulation tools for smart home applications should be solved in order to pave the way for future research. Among other features, new test-bed environments should allow: to simultaneously model multiple home applications (e.g. HVAC, DWH, lighting, electric devices); to easily apply and join advanced control strategies and embedded algorithms (e.g. profile-based controllers); to make simulations at appropriate scale times and levels of detail (e.g. to calculate the effects of a window open for ten minutes); to consider user occupancy effects,

number of users and human activities (e.g. rest, relax, work); to take into account the multiple effects of home devices and elements (e.g. energy consumption but also thermal contribution of appliances, effects of shading devices in thermal and lighting gains and losses); to simulate user sensitivity (e.g. dealing with high levels of CO_2, low humidity), etc.

(2) Such a complete tool would facilitate the simulation and checking of holistic control approaches for smart homes. Thus, the performance of a set of profile-based applications working together could be assessed. This is one of the main missing points of the current dissertation, that has not been accomplished due to the lack of competent simulation packages. As long as the previous point is not achieved, the checking of smart homes working as a whole can only be tackled in real installations, and assuming that, in such a case it would represent an isolated sample with limited possibilities for the analysis.

(3) Finally, another point that requires a thorough development and is addressed in future research is the description and mapping of conflicts, simultaneities and overlaps common in the home environment, as well as to define mechanisms for their detection, subscription and resolution. The combination of these functions with shadow processes point to provide the ability to adapt and react quickly and smartly to any problem or situation, making AmI systems more intelligent and satisfactory.

7.3 Derived Publications

The work displayed throughout the dissertation has been previously presented and published in diverse technical and scientific journals and conferences. The publications together with the aspects covered in them are commented on as follows:

We introduce the XSOM algorithm for the first time in [CIV09], aiming to study user profiles for building's energy performance simulations. The use of profiles for building calculations is also the scope of [ICKM11], but here the focus concentrates on how outliers alter the clustering task and the discovered representatives.

Habit patterns are presented and tested for the first time for smart home control in [IK10], with a wider description of XSOM and profile-based controllers. In [IK11], different clustering techniques are simulated and compared for the optimized implementation of profile-based setpoint temperature control systems for smart homes. In the same application case, the effect of habit pattern stability and reliability in the energy and comfort performance is explored in detail in [IKR11]. [RKIK11] introduces a comprehensive holistic architecture for the smart home control grounded by MAS, ontologies and profile-based control solutions. The vertical approach of such structure for the case of "air quality" and "thermal comfort" is faced in [IKK]. The Thermal Comfort Support Application is described in detail in [IK12b].

On the other hand, models for the system-user interaction, shadow processes and AmI environments that deal with habit information are introduced in [IK12a]. Profile-based and habit-oriented DSM applications that pursue energy optimization in common flats are developed in [IKGR11]. In a similar fashion, but for the case of houses that depend on Stand-alone Hybrid Power Systems, applications are proposed in [IPCK12].

In addition, two journal papers are under reviewing process at the date of the presentation of this dissertation. They cover the study of metrics (clustering refining) and clustering validation techniques for building's energy pattern discovery, and the implementation of profile-based control for central DHW systems.

Appendix

A Clustering Tool: eXclusive Self-Organising Map (XSOM)

There is a broad experience of using Self-Organising Maps (SOM) algorithms for pattern discovery, e. g. [Cho02], [FCNL01], [TZ02] and [LNH02]. It leads us to consider SOM algorithms as good alternatives to perform use and habit profile recognition in the built environment. Incongruous results in energy use scenarios suggested the mathematical analysis of available building data and pointed out the outlier presence. It caused the necessity of refining SOM tools in order to be able to reject outliers and allow the easy integration in HBA control and simulation. The final result is the development of eXclusive SOM or XSOM, presented in this section as an *advanced tool for pattern discovery in the home and building environment.*

A.1 XSOM Description

Given a set of samples (population), in which every sample is identified by means of a set of characteristics, SOM algorithms can identify and distribute the samples into groups according to two-dimensional mapping of a hypothetical solution space that is built on the basis of the resemblance among input vectors (or samples). Obviously, this relationship (or resemblance) is extracted from the global assessment of the sample characteristics. SOM algorithms usually work with Euclidean distances, but not always [JL04]. Other metrics can be suitable depending on the scenario nature, e. g. [KS98].

SOM networks use training samples to establish groups represented by neurons located in a competitive mesh. This is a dynamic process where each sample contributes and, at the same time, the SOM shapes patterns used as guidance for the classification of subsequent inputs. At the end of the process, diverse groups are stated and the training samples have been distributed into them; the pattern of each group (i. e. the winning neuron, centroid or gravity center) becomes the most representative member of its respective cluster, in spite of the fact that it is not an input sample and it is not necessary that a copy among the training input samples exist. Later, SOM networks classify test samples making assessment of their resemblance with previously fixed patterns.

Figure A.1 shows a possible classic SOM mapping of a set of training samples distributed in a bi-dimensional space, whereas Figure A.2 represents the classification performed by the XSOM

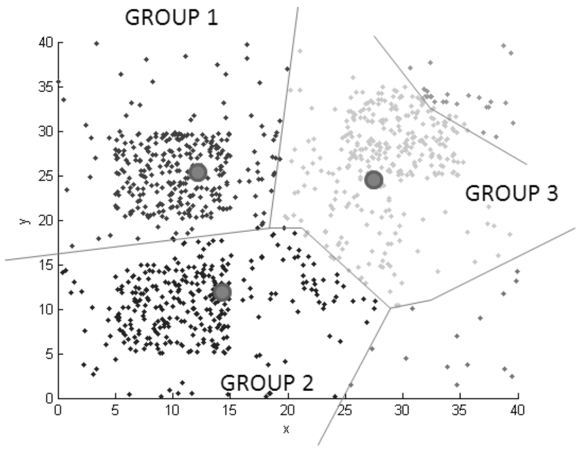


Figure A.1: Example of SOM classification in a 2D data representation

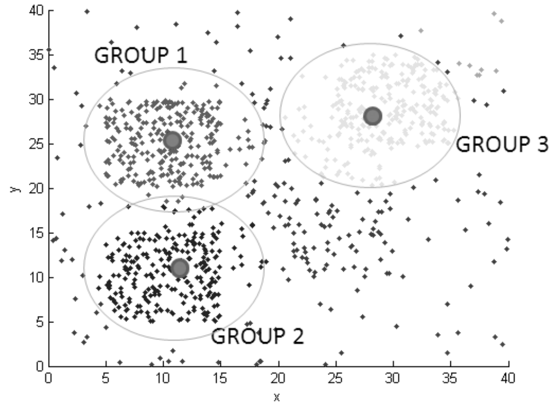


Figure A.2: Example of XSOM classification in a 2D data representation

in the same scenario. Classical SOM divides the space completely and each sample is absorbed by one of the found groups. The XSOM tool usually discovers the same number of groups as well as an extra-group, where the outliers elements are classified (in other words, this is the group of the samples that, because of their erratic and dissociated nature, should not pertain to any of the main groups). In both (Figure A.1 and Figure A.2) the boundaries of the groups are roughly marked, but in any case the main differences between SOM and XSOM classification methods can be clearly seen.

In the examples, let us focus on the three main groups. We can see how both tools have performed the centroids/patterns (big circles) assessing the contribution of each sample included into the same group. In the SOM example, remote samples (which are outliers in the XSOM case) influence the pattern generation. Therefore, the main patterns in the SOM case are significantly different to the ones in the XSOM case, which is due to a certain level of corruption introduced by the supposed outliers. In some cases, the outliers' presence can seriously affect the shape of the patterns and even the SOM's ability of finding out representative groups.

In order to achieve this filtering capability, XSOM algorithm introduces a new parameter, called *tolerance*, that fixes the admitted level of appropriateness. In Figure A.2, tolerance states the radius of the circle that surrounds each pattern. All the samples whose features place them out of the patterns' influence circles are considered outliers. And, in spite of the removal capability, XSOM tool informs about the nearest group to each outlying sample.

There is a lot of literature related to how SOM algorithms work; we can mention once again the well-known paper by Teuvo Kohonen [Koh90]. In short, the competitive learning algorithm for training is explained in the following points (XSOM enhancements are also introduced by steps 7 and 8):

1. Weights initialisation:
   $w_{ij} \in [-0.5, 0.5]$, with $1 \leq i \leq N$ and $1 \leq j \leq M$. N is the size of the input vector (variables or characteristics of the samples); M represents the number of output neurons (the maximum number of groups that the algorithm is allowed to find).

2. Data presentation:
   A normalised input vector (a sample) $I(t) = \{I_1, I_2, ..., I_N\}$ is introduced in the network.

3. Calculation.
   $$d_j = \sum_{i=1}^{N} (I_i(t) - w_{ij}(t))^2 \tag{A.1}$$
   is calculated, where $w_{ij}(t)$ is the node weight from the input node i to the output node j relevant in time t.

4. Selection:
   The output node $j*$ with the minimum distance d_j is selected (and the corresponding distance is saved). Here, the sample has been classified inside a group (the winning neuron). Each neuron represents a group.

5. Learning:
   For all $1 \leq i \leq N$ and $1 \leq j \leq M$ the weights of node $j*$ and its neighbours are adapted according to
   $$w_{ij}(t+1) = w_{ij}(t) + \Phi(j*, t)\eta(t)(I_i(t) - w_{ij}(t)). \tag{A.2}$$
   $\eta(t)$ is called the learning index, has a value between 0 and 1, and decreases progressively. The neighbourhood is defined by $\Phi(j*, t)$ and has a value between 0 and 1 depending on how the neighbourhood relationship is defined. The neighbourhood relationship between two given neurons can be '1', '0', as well as an intermediate value based on proximity functions: linear, Gaussian, etc. At this point, the sample classified during the selection step modifies the pattern that represents its group (changing the weights of the winning neuron) as well as the nearest neurons in a lower degree. Samples are classified and redistributed over the representation space at the same time.

6. Loop 1:
   The algorithm is repeated from step 2 until all data in the set have been processed.

7. (XSOM only) Reject samples:
   All samples where the distance to the nearest node exceeds the tolerance parameter $\frac{d_j}{N} > tol$ are rejected. The parameter *tol* has to be defined in advance.

8. (XSOM only) Loop 2:
   The algorithm is repeated from step 2 until there is no new excluded sample (outlier).

To sum up, XSOM enhances SOM by repeating the SOM procedure after rejecting the found outliers. Weights (pattern frames) are recalculated without the outliers influence. In the end, the output nodes of the SOM or XSOM algorithm (centroids) represent significant groups and become the most representative patterns of each group. Algorithms not only classify input vectors and assign them to groups (or reject the sample in case of XSOM), the distance between each sample and its pattern is also stored and can be used to derive how close samples are to their related patterns.

Let us see how XSOM adds new aspects to the normal SOM performance with two simple examples of pattern recognition and sample classification.

A.2 Pattern recognition simple case

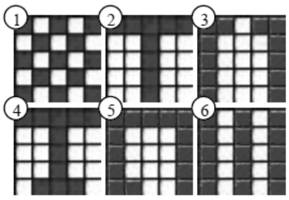


Figure A.3: Input samples in a simple example of pattern recognition

In this example, SOM and XSOM tools are used to abstract patterns from a given set of training samples (Figure A.3). Both tools are demanded to find only two patterns. SOM classifies the samples '1', '2' and '4' in a first group, and the samples '3', '5' and '6' in a second group. On the other hand, XSOM finds that samples '1' and '6' are not representative enough to pertain to any group either to contribute to the performance of patrons that will be used to classify new samples. Thus, the first XSOM group includes samples '2' and '4', and the second group is formed by samples '3' and '5'.

Figure A.4: SOM & XSOM patterns

Figure A.4 represents the two performed patterns in each case. It is visually perceptible that SOM patterns are more diffuse (corrupted by noise) than the XSOM ones, and it seems to be clear which are the best pair of patterns in this example. It is not completely free of discussion, as it depends on what we expect of the obtained patterns or the subsequent classification. In any case, so far we can see that XSOM introduces certain flexibility that does not appear in the SOM case.

Depending on the scenario, the tolerance enhancement is not always necessary. Changing tolerance makes the filtering effect more relaxed or harder, or even disappears (an 'infinite' value on tolerance leads to the normal SOM case; therefore, we can say that SOM becomes an special case of XSOM). In the given example, if tolerance is changed to a higher value, the sample '6' is assimilated inside the group of samples '3' and '5'.

A.3 Sample classification simple case

In the second example (Figure A.5), SOM and XSOM tools have already been trained with six patterns (six output neurons). Later on, both tools are required to classify the test samples shown in Figure A.6.

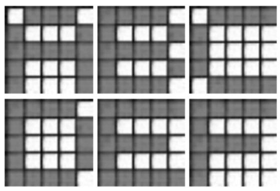


Figure A.5: Patterns in the classification example

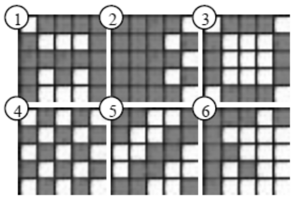


Figure A.6: Test samples in the classification example

Test sample	SOM pattern	XSOM pattern
1	A	A
2	B	B
3	D	D
4	B	Outlier(B)
5	B	Outlier(B)
6	F	F

Table A.1: SOM and XSOM tool classification.

SOM and XSOM tools perform the classification which is shown in Table A.1. Both tools essentially perform the same classification, but the XSOM tool identifies the samples '4' and '5' as outliers, regardless of the fact that it also recognises the nearest group that both samples pertain. XSOM classification seems to be closer to the human assessment.

B Testing & Evaluation Resources

Simulations are the main mean to evaluate proposals, tools and methods throughout the current dissertation. In this chapter, the simulated test-bed environment is widely described, explaining the features of available databases, models, controllers, clustering tools and performance indices.

B.1 Simulated Environment

Simulation is a suitable means for designing energy efficient buildings [JH02]. The environment applied in this work is built on top of the HAMBase model – a simulation model commonly used for investigating the heat and vapour flows in a building. With this model, the indoor temperature, the indoor air humidity and energy use for heating and cooling of a multi-zone building can be analysed. The physics of the HAMBase model is based on ELAN model implemented in MATLAB [Sch07b].

The simulation environment has been originally devised to check parameters and controllers of the Setpoint Temperature Control Application, introduced in Section 4.3.3. Controller agents have been embedded in MATLAB/Simulink and, finally, bound to the HAMBase model. Figure B.1 gives an overview of the simulation environment.

- **Area 1** marks the building model.
- **Area 2** determines the time step, selects the strategy and fixes the heating setback temperature.
- **Area 3** represents the *setpoint temperature control agent* of the smart system. It decides the next setpoint temperature based on the time, the selected strategy, the occupancy data and the current occupancy profile (if the strategy based on profiles is selected).
- **Area 4** shows the heating controller, a PID controller which takes the indoor temperature and the setpoint temperature as inputs. It also outputs the heating power for the building model's heating system. By default, the parameters are: $K_P = 2, K_I = 0.8$, and $K_D = 0.4$.
- Components not circled are used for visualisation, data management and storage.

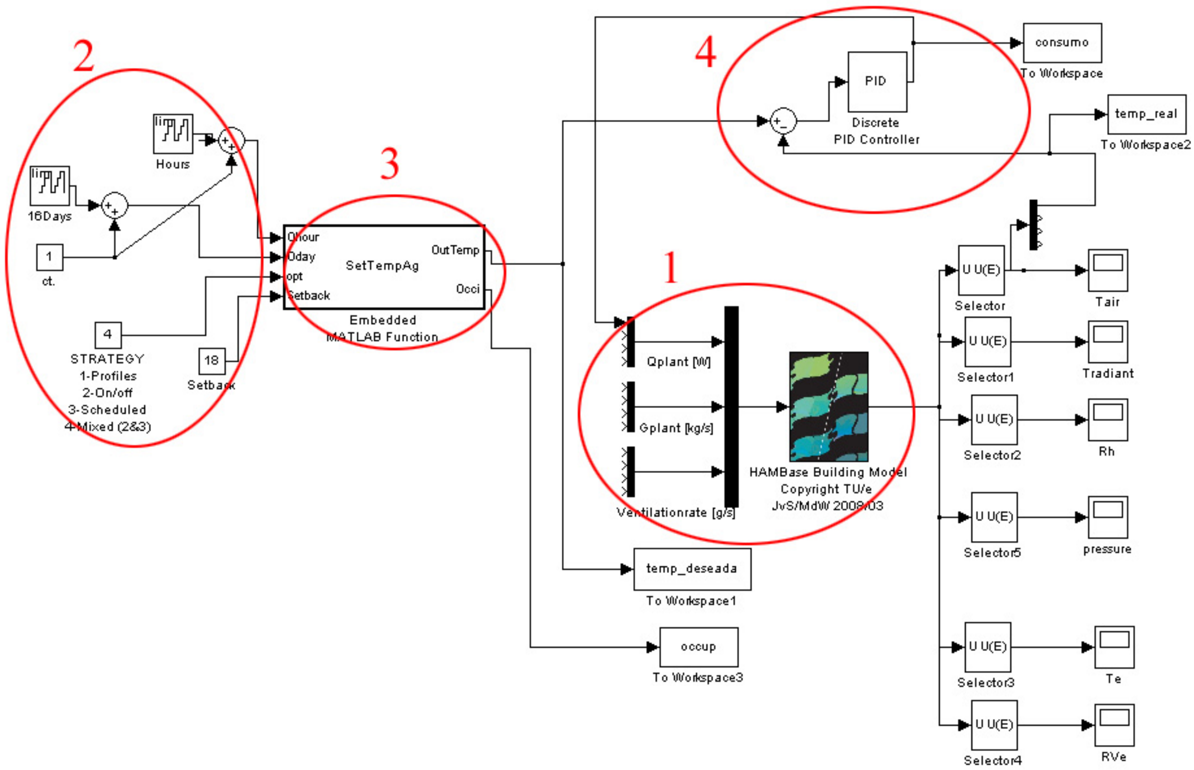


Figure B.1: Simulation environment for strategy comparisons

B.1.1 Building Model

The simulated facility consists of an office like building based on the Hamlab default model (http://sts.bwk.tue.nl/hamlab/). The building is separated into four zones; however, only in the first zone (120m^3) is heating control applied. This zone is also the area under test. The location is De Bilt (Netherlands), 52.12°N 5.18°E. Thus, weather data correspond to typical Central Europe autumn/winter time. Building and weather data belong to the HAMBase configuration by default.

As far as the building geometry and materials are concerned, in Zone 1 the external wall is northwest oriented and made up of $0.214m$ limestone (kalkzandsteen), $0.120m$ insulation (steenwol), $0.050m$ air gap and $0.100m$ brick (rood). The internal surface heat transfer of the whole wall is $Ri = 0.13[Km^2/W]$, whereas the surface heat transfer resistance at the opposite site is $Re = 0.04[Km^2/W]$. The external solar radiation absorption coefficient is $ab = 0.9$, and the external long-wave emissivity is also $eb = 0.9$. It has no heat loss because of thermal bridges. The external wall has $8m^2$ of glazing (saint-roch skn 165) without obstacles or shadings.

As far as weather data is concerned, it is obtained from real weather databases supplied by HAMLab.

Simulations are executed in the autumn/winter period, which means that only heating is applied. As an example, Table B.1 depicts key values to understand the simulated context concerning weather conditions for the test executed in Section 4.4.2 – collected in De Bilt (Netherlands), 52.12°N 5.18°E; 11th to 27th, November, 1981.

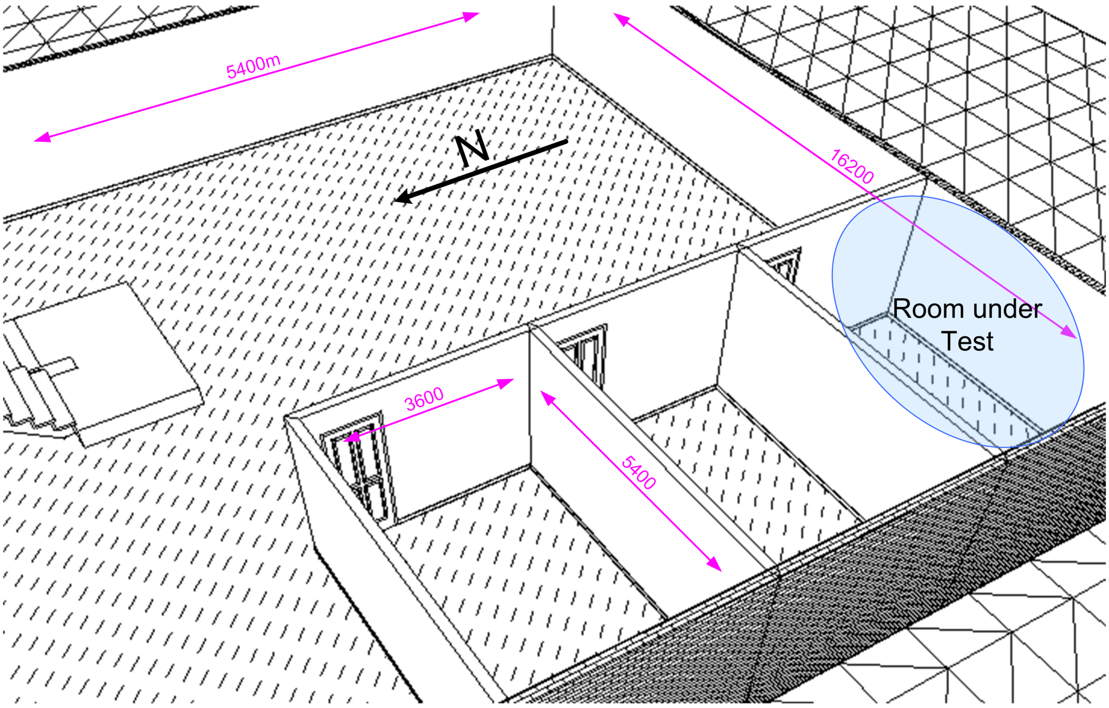


Figure B.2: Plan of the simulated building.

	DSR	AirT	DSRp	CC	RH	WV
Max	244.2	19.2	446.8	8	100	9.8
Min	0	-2.0	0	1	56	0
Mean	49.2	9.9	21.1	5.7	84.7	3.3
σ	72.1	3.7	62.5	2.4	10.6	2.0
Mode	–	–	–	8	–	–

DSR: Diffuse Solar Radiation [W/m^2].
AirT: Outdoor Air Temperature [°C].
DSRp: Diffuse Solar Radiation (plane normal to the direction) [W/m^2].
CC: Cloud Cover [1...8].
RH: Outdoor Relative Humidity [%]
WV: Wind Velocity [km/h].

Table B.1: Climate data summary.

B.1.2 Occupancy Simulation

Simulations work using instantaneous occupancy, which is either taken from real databases or modelled depending on the test. The real available databases are depicted in Appendix B.2, and the process to model occupancy is shown below in this section.

In case of coming from databases, a set occupancy vectors or profiles are initially processed so as to remove non-complete or wrongly-monitored days. Occupancy profiles are then time-series vectors with 24 fields that represent hours (Figure B.3). Hence, the sample rate is one hour, therefore the minimum considered presence or absence in simulations lasts one hour.

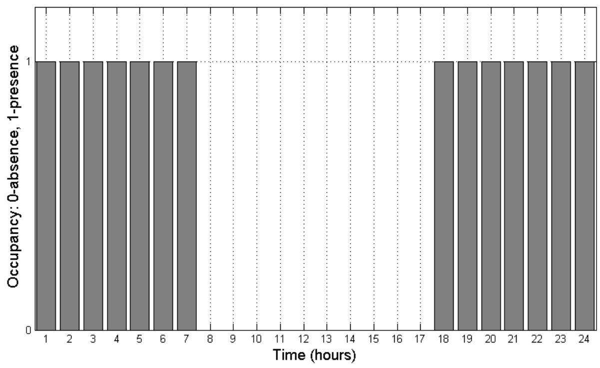


Figure B.3: Example of a daily occupancy profile.

Modelling Occupancy

For some tests, occupancy profiles are not taken from databases but modelled, e. g. Section 6.4. In order to model a set of occupancy profiles for simulation, we start designing a singular template as shown in Figure B.4.

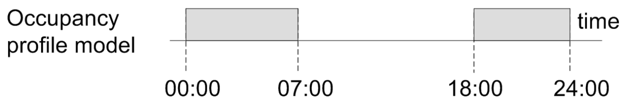


Figure B.4: Occupancy profile model of a flat for a single user. The flat usually becomes empty after 7am and the user arrives home around 6pm.

Later on, considering the basis of a steady user who always immutably keeps the same habit profile, the amount of daily profiles required by the test is generated adding one or some of the next variations upon the original model:

1. *Ahead of schedules or delays in arrival and leaving times* (α).
   The time when people leave or come back to the flat can be randomly shifted within two hours before or after the original expected occupancy.

2. *Unexpected short presences and absences* (β).
   Random presences and absences with a duration no longer than two hours.

3. *Unexpected medium presences and absences, UPA (γ).*
   Random presences and absences with a duration between two and five hours.

4. *Unexpected long presences and absences (δ).*
   Random presences and absences with a duration between five and twelve hours.

5. *Additional long period trends (ϵ).*
   Superimposed habits that are repeated with a lower frequency than expected (every second week, third week or every month). It supposes additional, well-shaped patterns within the database.

6. *Change of habits.*
   This aspect is not considered in simulations because it is assumed that smart tools will adapt gradually to new habits (or variations of current habits). At the beginning, new trends are necessarily considered as random noise before they turn into stable habits.

For every kind of disruption, a probability value is assigned in a way that a new occupancy profile ($occp_i$) is stated as follows:

$$occp_i = model \oplus f(\alpha, \beta, \gamma, \delta, \epsilon) \tag{B.1}$$

where $f(..)$ is a function that implements the explained perturbation sources, and $\alpha, \beta, \gamma, \delta, \epsilon \in [0, 1]$ express respectively the likelihood of appearance for the described types of perturbation 1, 2, 3, 4 and 5, respectively.

B.1.3 Controllers of the Setpoint Temperature Control Application

For the Setpoint Temperature Control Applications, controllers make decisions based on occupancy (instantaneous and profiles) and setpoint temperatures (comfort profiles and setback/setforward values). The differences among them can be also assessed in Figure B.5.

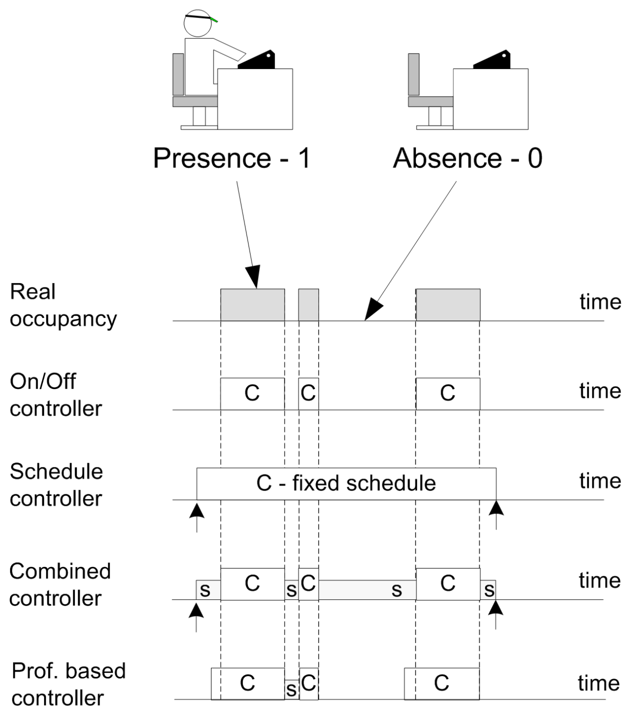


Figure B.5: Controllers' performance comparison. 'C' stands for 'Comfort', 's' for 'setback/setforward'.

- On/off controller (On/off).
  On/off controllers follow a simple strategy that consists of switching on devices when people arrive home (or the specific room) and switching them off when they leave the

dwelling/room. The controller may be triggered manually or automatically by means of occupancy detection.

- Scheduled controller (Sched.).
  The Scheduled controller establishes comfort temperatures during the expected or normal building/flat use schedule. It is common in office buildings, in the public service sector, or even in houses where inhabitants do not want to be bothered with the manual heating/cooling adjustment. The energy performance is usually rather poor, but comfort ratings are satisfactory provided people are at home during the scheduled time.

  In simulations, each schedule is calculated using averages of the used occupancy samples during the simulated days, beginning one hour prior to all the expected arrivals in order to fulfil thermal comfort requirements.

- Combined controller (Comb.).
  The combined controller sets the temperature to setback/setforward levels within the schedule, and goes up to comfort levels when occupancy is detected. It is a good evolution of both previous controllers with a better commitment between comfort and energy savings.

- Controller based on profiles (Prof.).
  The system is aware of users' habits and regulates the temperature on the basis of this knowledge. This kind of controller is supported by clustering methods for the pattern discovery. The clustering methods available for the simulations are explained in Appendix B.1.4

The controller parameters utilised by default in simulations are as follows:

- Setback temperature fixed to $18°C$.
- Comfort temperature set to $23°C$.
- Controller based on profiles: XSOM (with the parameters shown in Appendix B.1.4). The Preparation Time (Pt) is fixed to $Pt = 1$ hour, and the Waiting Time (Wt) is $Wt = 4$ hours.

B.1.4 Pattern Discovery Techniques – Clustering Tools

Clustering is the technique utilised throughout this dissertation to discover representatives (patterns) inside the available use database. As explained in Chapter 5, there are different approaches to face the task of clustering. The examples commented on as follows have been used and checked in our research. To do that, several parameterisations have been necessary; for every technique we show configurations or parameters by default.

- Self-Organising Maps (SOM) & Exclusive Self-Organising Maps (XSOM).

  Self-organising maps or Kohonen maps are artificial neural networks with unsupervised learning. An exhaustive explanation can be looked up in the works of Teuvo Kohonen, e. g. [Koh90]. Experiments with SOM and XSOM have been carried out by the MATLAB SOM toolbox 2.0 [VHAP00] and software applications developed under CVI/LabWindows.

  Some parameters commonly used in the simulations are as follows: the similarity metric is based on Euclidean distance; the maximum number of clusters is initially fixed to 10; the learning index $\eta(t)$ decreases in a linear fashion as a function of the number of iterations; the neighbourhood $\Phi(j*, t)$ is set to 1 for the adjacent neurons, and 0 for the rest. Explanations concerning parameters are depicted in Appendix A.

Exclusive SOM (XSOM) is a SOM evolution with outliers detection. The clustering algorithm is widely explained in Appendix A. The parameters remain the same like in the SOM case, but XSOM additionally requires the adjustment of the *tolerance*. By default, it is fixed to reject approximately 10% of input samples.

- Fuzzy C-means.

  In fuzzy clustering, every input sample belongs to every possible cluster in a certain degree, instead of just belonging to a single cluster. The objective of the Fuzzy C-means algorithm is to find the optimal partition that minimises the objective function – called *C-means Functional* [Bez81] –, probably the most commonly used function in fuzzy clustering. There are many works that extensively deal with fuzzy clustering, the interested reader can be directed to, for example, [SSJ02].

  For simulations, the deployed algorithm has been developed with the Fuzzy Clustering and Data Analysis Toolbox [BAF09]. The default parametrisation for the experiments based on Fuzzy C-means is fixed as follows: the exponent for the matrix U is set to $m = 2.0$. The termination tolerance is fixed to $\epsilon = 0.001$. Finally, the number of initial clusters has usually been set to $c = 5$ for most of the tests.

- Simple K-means & K-means with Repeated Bisection

  The K-means algorithm is a type of partitional clustering method that assigns each input sample to the cluster whose center (centroid) is nearest. The center is the average of all samples included in the cluster.

  The K-means method needs the number of clusters to be fixed in advance. For the developed experiments and tests, the final and most suitable initial number of clusters has usually been fixed analysing graphic representations of different K-means performances. This allows identifying averages and deviations of similarity between the found centroids and samples regarding their respective centroid (Fig. B.6).

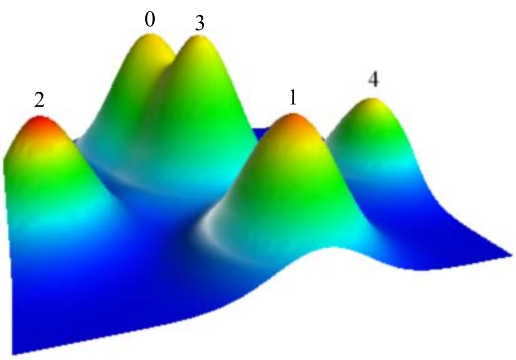


Figure B.6: Example of a clustering result by graphic assessment. Keys to interpret the figure are available in gCLUTO's manual [Kar03].

In addition, the optimisation function to obtain the clusters is fixed by the optimisation of the I_2 criterion, which is suitable to find *globular* clusters:

$$I_2 : maximise \longrightarrow \sum_{i=1}^{k} \sqrt{\sum_{v,u \in S_i} sim(v,u)} \tag{B.2}$$

where k is the total number of clusters and S is the total set of objects to be clustered. S_i stands for the set of objects assigned to the i^{th} cluster. v and u represent two objects, and $sim(v,u)$ is the similarity between two objects.

On the other hand, the K-means with Repeated Bisection algorithm first cuts the entire population into two clusters. Then, one of them is chosen to be further bisected, leading to a total of three clusters. This process continues until k clusters are obtained. Each of these divisions is executed in a way that the resulting solution optimises the criterion function. Unlike the K-means method, K-means with Repeated Bisection gives rise to hierarchical solutions and deals more successfully with density variations.

The parametrisation by default remains the same as in the normal K-means case; in addition, the method used to select which cluster to bisect next is usually *the largest one*.

For the experiments, both K-means techniques have been developed using CLUTO tools [Kar03]. K-means-based clustering is perhaps the most commonly used technique for clustering, an interesting, recent work can be looked up in [Wu12].

- Graph Clustering

  Graph Clustering starts with setting up a graph based on neighbourhood (each sample is a vertex). Next, the graph is split into clusters using a min-cut graph partitioning algorithm. The method is well-suited in applications and cases that can easily be well-represented by graphs, i. e. dense sub-graphs with relatively few connections towards the rest of the graph. A good, interesting overview of graph clustering is offered in [Sch07a].

  For our applications, graph clustering is also performed by CLUTO tools [Kar03]. The best results are usually obtained with similarity functions based on correlation coefficients, using asymmetric graph models, and selecting *the best one* as the method used to select the next cluster to split.

- Support Vector Clustering (SVC)

  Support Vector Clustering is based on the support vector approach. This method tries to fix cluster boundaries in regions of data space where there is little data. The input samples are mapped into a high dimensional feature space using a kernel function. Later on, the method searches for the smallest sphere that embraces the image of the data by means of the *Support Vector Domain Description* algorithm. This sphere draws boundaries in the data space that can become clusters to classify the entire population. A deeper understanding of SVC is given in [BHHSV01].

  SVC tests have been performed using the toolbox RapidMiner (http://http://rapid-i.com/). Parameters often used by default are as follows: the minimal number of samples in each cluster is set to $c = 100$. The kernel type taken is *radial*. The scale parameter is fixed to $q = 5$, and the precision is set to $\varepsilon = 0.0010$. A maximum of $i = 10000$ iterations is allowed. Finally, the number of virtual sample points to check for proximity is set to $nn = 20$.

B.1.5 Performance Indices

Outcomes of simulations and analysis are summarised by means of a set of indices devised to give an evaluation of the obtained, final performance. Data used by the indices as inputs are listed as follows:

- N [s], total simulated time.
- Q_i [W], instantaneous heating consumption.
- Occ_i (dimensionless), instantaneous occupancy. $Occ_i \in \{0, 1\}$.
- $stpT_i$ [K], the setpoint or desired temperature.
- $realT_i$ [K], real indoor temperature.
- Pcc_i (dimensionless), instantaneous predicted occupancy $Pcc_i \in \{0, 1\}$.

The performance indexes are:

- Consumption (Q).

  It is the average consumption of heating. For the sake of convenience, sometimes results in tables are converted into Wh.

$$Q\big|_W = \frac{\sum_{i=1}^{N} Q_i}{N} \tag{B.3}$$

- Temperature difference (dT).

  It is the average difference between real temperature and desired temperature during occupied periods.

$$dT_i = \begin{cases} 0, & \text{if } Occ_i = 0 \\ |stpT_i - realT_i|, & \text{if } Occ_i = 1 \end{cases} \tag{B.4}$$

$$dT\big|_{^\circ C} = \frac{\sum_{i=1}^{N} dT_i}{N} \tag{B.5}$$

- Time in Comfort (TiC).

  Whenever there are people at home, TiC is the time the system matches comfort temperatures.

$$TiC_i = \begin{cases} 1, & \text{if } (Occ_i = 1 \wedge dT_i < 0.5) \\ 0, & \text{otherwise} \end{cases} \tag{B.6}$$

$$TiC\big|_{hours} = \frac{\sum_{i=1}^{N} TiC_i}{3600} \tag{B.7}$$

- Time to Comfort (TtC).

  It is the accumulated necessary time to reach comfort temperatures.

$$TtC_i = \begin{cases} 1, & \text{if } (Occ_i = 1 \wedge dT_i \geq 0.5) \\ 0, & \text{otherwise} \end{cases} \tag{B.8}$$

$$TtC\big|_{hours} = \frac{\sum_{i=1}^{N} TtC_i}{3600} \tag{B.9}$$

- Occupancy rating (Occ, in %).

$$Occ\big|_{\%} = \frac{1}{N}\sum_{i=1}^{N} Occ_i \tag{B.10}$$

- Percentage of Correct Occupancy Prediction (POk).

  It is the percentage of time that the systems correctly predicts the occupancy.

$$POk\big|_{\%} = 100(1 - \frac{1}{N}\sum_{i=1}^{N} Occ_i \oplus Pcc_i) \tag{B.11}$$

- Percentage of pleasant time (Pt) in %.

  It evaluates which time percentage people are in comfort.

$$Pt = 100\frac{TiC - TtC}{TiC} \tag{B.12}$$

- Energy-cost of pleasant time (Ec) in Wh.

  It assigns a cost-value to the comfort time, sharing out the total consumption among comfort hours. It is important to note that the Ec index does not evaluate the total energy consumption, but a relationship between comfort time and energy (i. e. how effectively the energy is applied).

$$Ec = \frac{Q}{TiC} \tag{B.13}$$

B.2 Databases

Experiments demand real input data to simulate a home and building context as realistically as possible. To do that, the following databases have been utilised in the conducted tests:

B.2.1 Leako

Leako (www.leako.es) is an enterprise from the Basque Country which specialises in central heating, DHW and air conditioning installation, distribution, and metering. Leako System has been working and developing novel control systems since 1995 in order to improve conventional installations and obtain more security, efficiency and energy savings. The system is prepared for sets of apartments and office buildings and takes advantage of communication technology possibilities. It consists of a central installation which supplies heating, DHW and air conditioning to the entire set of apartments or buildings, and it incorporates subcentrals in every dwelling that provide an individual service for each customer (Figure B.7).

The Leako Database consists of energy data obtained each hour during seven years from more than 700 dwellings; specifically, the collected data include: heating KWh, DHW KWh, consumed water liters, and average indoor temperature.

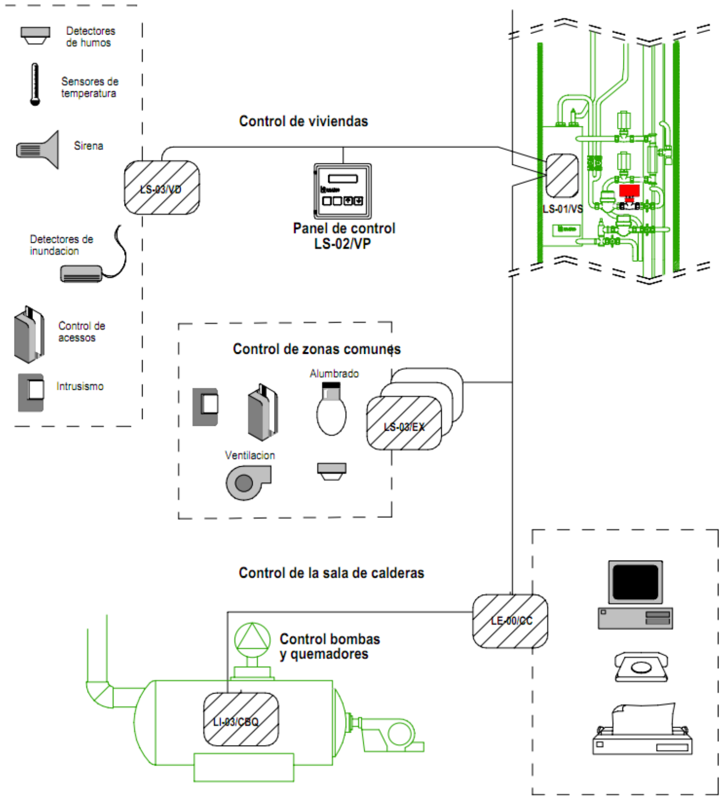


Figure B.7: Leako's domotic schema.

B.2.2 Viennese Flat

The Viennese Flat database stores occupancy data from a Viennese flat ($65m^2$) with a single inhabitant. Occupancy data is obtained by means of a data-logger which receives data from a sensor connected to the lock of the main door. The information has been collected from 21-09-2011 to 17-11-2011.

The team in charge of the management of the Viennese Flat database is part of the Automation Systems Group (Institute of Computer Aided Automation) of the Vienna University of Technology (https://www.auto.tuwien.ac.at/).

B.2.3 Spanish University Buildings

The Spanish University Buildings database contains information concerning energy consumption collected from five Spanish university buildings.

Buildings are located in Barcelona, Spain, and data cover hourly consumption from August, 29th 2011 to January 1st 2012. Data is publicly available at (http://www.upc.edu/sirena). The selected buildings belong to the "Campus Nord", they are: "Edifici A1", "Edifici A4" and "Edifici A5" (university classrooms and laboratories), "Biblioteca" (a library) and "Biblioteca" (an office building for administration and rectorship). The usable space of the buildings have the following dimensions: "Edifici A1", 3966.59 m^2; "Edifici A4", 3794.95 m^2; "Edifici A5", 3886.12 m^2; "Biblioteca", 6644.4 m^2; "Biblioteca", 5927.21 m^2.

Figure B.8: Snapshot of the SIRENA Project main site.

The collection of energy information has been carried out by the Spanish University UPC (*Universitat Politecnica de Catalunya*) from 2003 on [SIR11]. Energy data has been collected and is published annually, progressively improving the range of the project. Nowadays, the 99% area of the University Campus (417.456m^2) is monitored for electricity, 32% water and 65% gas. In this project, enhancements in energy savings are obtained as a consequence of different factors, but mainly due to improvements in the management of the monitored information and the existence of derived energy feedbacks (Figure B.8).

Literature

[ABS05] Arroyo, Ernesto ; Bonanni, Lonardo ; Selker, Ted: Waterbot: Exploring feedback and persuasive techniques at the sink. In: *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*. New York, USA : ACM, 2005 (CHI '05), S. 631–639

[AD00] Aarts, H. ; Dijksterhuis, A.: Habits as knowledge structures: Automaticity in goal-directed behavior. In: *Journal of Personality and Social Psychology* 78 (2000), Nr. 1, S. 53–63. – APA, American Psychological Association

[AE06] Aarts, Emile ; Encarnaçao, José: *True Visions: The Emergence of Ambient Intelligence*. Springer, 2006

[AFB+11] Axelrod, L. ; Fitzpatrick, G. ; Balaam, M. ; Mawson, S. ; Burridge, J. ; Ricketts, I. ; Smith, P.P. ; Rodden, T.: A toolkit to explore lived experience of motivation: When words are not enough. In: *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 5th International Conference on*, 2011, S. 32–39

[AHK+12] Augusto, Juan C. ; Huch, Michael ; Kameas, Achilles ; Maitland, Julie ; McCullagh, Paul ; Roberts, Jean ; Sixsmith, Andrew ; Wichert, Reiner: *Handbook of Ambient Assisted Living: Technology for Healthcare, Rehabilitation and Well-being (Ambient Intelligence and Smart Environments)*. IOS Press, 2012

[Ash97] *ASHRAE Handbook – Fundamentals*, 1997. – Chapters 27 to 29

[Ash01] *Ventilation for Acceptable Indoor Air Quality*. ANSI/ASHRAE Standard 62, 2001

[Ash04] *Thermal Environmental Conditions for Human Occupancy*. ANSI/ASHRAE Standard 55, 2004

[Ash10] *BACnet: A Data Communication Protocol for Building Automation and Control Networks*. ANSI/ASHRAE Standard 135, 2010

[AT01] Adomavicius, G. ; Tuzhilin, A.: Using data mining methods to build customer profiles. In: *Computer* 34 (2001), feb, Nr. 2, S. 74 –82

[Aug07] Augusto, Juan C.: Ambient Intelligence: The confluence of ubiquitous/pervasive computing and artificial intelligence. In: *Springer London. Intelligent Computing Everywhere* (2007), S. 213–234

[BAF09] Balasko, B. ; Abonyi, J. ; Feil, B.: *Fuzzy Clustering and Data Analysis Toolbox*. Veszprem, Hungary: Department of Process Engineering, University of Veszprem, 2009. – Available: `http://www.sunfinedata.com/wp-content/uploads/2009/10/FuzzyClusteringToolbox.pdf`

[Bal92] Balakrishnan, N.: *Handbook of the Logistic Distribution*. New York : Marcel Dekker, 1992

[Ban77] Bandura, Albert: Self-efficacy: Toward a unifying theory of behavioral change. In: *Psychology Review* 84 (1977), S. 191–215

[BBCM09] Barbato, A. ; Borsani, L. ; Capone, A. ; Melzi, S.: Home energy saving through a user profiling system based on wireless sensors. In: *Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*. Berkeley, California, 2009, S. 49–54

[BBSZ09] Borggaard, J. ; Burns, J.A. ; Surana, A. ; Zietsman, L.: Control, estimation and optimization of energy efficient buildings. In: *American Control Conference, ACC '09*, 2009, S. 837–841

[BCC08] Bonino, D. ; Castellina, E. ; Corno, F.: DOG: An ontology-powered OSGi domotic gateway. In: *Tools with Artificial Intelligence, ICTAI '08. 20th IEEE International Conference on* Bd. 1, 2008, S. 157–160

[BD98] Brager, Gail S. ; de Dear, Richard J.: Thermal adaptation in the built environment: A literature review. In: *Energy and Buildings* 27 (1998), Nr. 1, S. 83–96

[BDK+04] Brainin, Elisabeth ; Dietrich, Dietmar ; Kastner, Wolfgang ; Palensky, Peter ; Rösener, Charlotte: Neurobionic architecture of automation systems – Obstacles and challenges. In: *Proc. of 7th IEEE Africon Conference in Africa (AFRICON '04)* Bd. 2, 2004, S. 1219–1222

[BE01] Bellotti, V. ; Edwards, K.: Intelligibility and accountability: Human considerations in context-aware systems. In: *Human-Computer Interaction* 16 (2001), Nr. 2-4, S. 193–212

[Bez81] Bezdek, J. C.: *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York : Plenum Press, 1981

[BHHSV01] Ben-Hur, Asa ; Horn, David ; Siegelmann, Hava T. ; Vapnik, Vladimir: Support vector clustering. In: *Journal of Machine Learning Research* (2001), Nr. 2, S. 125–137

[Bon10] Bonneville, Charlotte: *Central Domestic Hot Water Systems in Multi-Family Buildings*. ACEEE Hot Water Forum. 2010. – Presentation available: `http://aceee.org/files/pdf/conferences/hwf/2010/3C_Charlotte_Bonneville.pdf`

[Boy06] Boyatzis, Richard E.: An overview of intentional change from a complexity perspective. In: *Journal of Management Development* 25 (2006), Nr. 7, S. 607 – 623

[Bra99] BRAUN, James E.: The impact of demand-controlled and economizer ventilation strategies on energy use in buildings. In: *ASHRAE Transactions* 105 (1999), Nr. 2

[BRM06] BOURGEOIS, Denis ; REINHART, Christoph ; MACDONALD, Iain: Adding advanced behavioural models in whole building energy simulation: A study on the total energy impact of manual and automated lighting control. In: *Energy and Buildings* 38 (2006), Nr. 7, S. 814 – 823. – Special Issue on Daylighting Buildings

[Buj10] BUJAK, Janusz: Heat consumption for preparing domestic hot water in hospitals. In: *Energy and Buildings* 42 (2010), Nr. 7, S. 1047 – 1055

[CB10] CABANES, Guenael ; BENNANIR, Younes: Learning the number of clusters in Self Organizing Map. In: MATSOPOULOS, George K. (Hrsg.): *Self-Organizing Maps*. InTech, 2010

[CDD10] CROSBIE, Tracey ; DAWOOD, Nashwan ; DEAN, John: Energy profiling in the life-cycle assessment of buildings. In: *Management of Environmental Quality: An International Journal* 21 (2010), Nr. 1, S. 20–31

[Cha98] CHAVENT, Marie: A monothetic clustering method. In: *Pattern Recognition Letters* 19 (1998), S. 989–996

[Che00] CHENG, J. G.: *Outlier Management in Intelligent Data Analysis*, University of London, Diss., 2000

[CHKG08] CRAWLEY, Drury B. ; HAND, Jon W. ; KUMMERT, Michael ; GRIFFITH, Brent T.: Contrasting the capabilities of building energy performance simulation programs. In: *Building and Environment* 43 (2008), Nr. 4, S. 661–673

[Cho02] CHO, Sung-Bae: Structure-adaptive SOM to classify 3-dimensinal point light actors' gender. In: *Proc. of the 9th international Conference on Neural information Processing (ICONIP 'O2)* Bd. 13, 2002, S. 949–953

[CIV09] CANTOS, Sergio ; IGLESIAS, Félix ; VIDAL, Jordina: Comparison of standard and case-based user profiles in building's energy performance simulation. In: *11th Conference of International Building Performance Simulation Association, Building Simulation*. Glasgow, Scotland, July 2009, S. 584–590

[CM07] CORRADO, Vincenzo ; MECHRI, Houcem E.: Effect of data uncertainty on energy performance assessment of buildings. In: *Climamed Proceedings*, 2007, S. 737–758

[Coe98] COEN, Michael H.: Design principles for intelligent environments. In: *Proceedings of the 15th National Conference on Artificial intelligence / 10th Conference on Innovative Applications of Artificial Intelligence*. Menlo Park, CA, USA : American Association for Artificial Intelligence, 1998 (AAAI '98/IAAI '98), S. 547–554

[Com06] COMMISSION OF THE EUROPEAN COMMUNITIES: *Action Plan for Energy Efficiency: Realising the Potential*. Brussels : COM(2006) 546, 2006

[CT07] CHEN, W. ; TSENG, W.: A novel multi-agent framework for the design of home automation. In: *4th International Conference on Information Technology (ITNG)*, 2007, S. 277–281

[CTGMM+09] Cardenas-Tamayo, Rolando A. ; García-Macías, J. A. ; Miller, Timothy M. ; Rich, Patrick ; Davis, Janet ; Albesa, Joan ; Gasulla, Manel ; Higuera, Jorge ; Penella, María T. ; Garcia, José ; Fernández-Montes, Alejandro ; Grado-Caffaro, Maria-Angeles ; Kappel, Karin ; Grechenig, Thomas ; Umut, lhan ; Uçar, Erdem ; Wall, Josh ; Ward, John: Pervasive computing approaches to environmental sustainability. In: *IEEE Pervasive Computing* 8 (2009), January, S. 54–57. – ISSN 1536–1268

[CYH+03] Cook, D.J. ; Youngblood, M. ; Heierman, III ; Gopalratnam, K. ; Rao, S. ; Litvin, A. ; Khawaja, F.: MavHome: An agent-based smart home. In: *Pervasive Computing and Communications, PerCom. Proceedings of the 1st IEEE International Conference on*, 2003, S. 521 – 524

[DB79] Davies, David L. ; Bouldin, Donald W.: A cluster separation measure. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* PAMI-1 (1979), april, Nr. 2, S. 224 –227

[DCB+02] Das, S. K. ; Cook, D. J. ; Battacharya, A. ; Heierman III, E. O. ; Lin, T.-Y.: The role of prediction algorithms in the MavHome smart home architecture. In: *Wireless Communications, IEEE* 9 (2002), Nr. 6, S. 77–84

[DD04] De Dear, R.: Thermal comfort in practice. In: *Indoor Air* 14 (2004), S. 32 – 39

[DFKU07] Dietrich, Dietmar ; Fodor, Georg ; Kastner, Wolfgang ; Ulieru, Mihaela: Considering a technical realization of a neuro-psychoanalytical model of the mind. In: *Preprints Proc. of Workshop 1st Intl. Engineering and Neuro-Psychoanalysis Forum*, 2007, S. 13–20

[DFZB09] Kap. 1 In: Dietrich, Dietmar ; Fodor, Georg ; Zucker, Gerhard ; Bruckner, Dietmar: *Simulating the Mind*. Springer Wien New York, 2009

[DiL05] DiLouie, Craig: *Advanced Lighting Controls: Energy Savings, Productivity, Technology and Applications*. The Fairmont Press, Inc., 2005

[DKM+04] Dietrich, Dietmar ; Kastner, Wolfgang ; Maly, Thomas ; Rösener, Charlotte ; Russ, Gerhard ; Schweinzer, Herbert: Situation modeling. In: *Proc. of 5th IEEE Intl. Workshop on Factory Communication Systems (WFCS '04)*, 2004, S. 93–102

[DL08] Duan, P. ; Li, H.: ZigBee wireless sensor network based multi-agent architecture in intelligent inhabited environments. In: *IET 4th International Conference on Intelligent Environments*, 2008, S. 1–6

[DLY+06] Davidoff, Scott ; Lee, Min K. ; Yiu, Charles ; Zimmerman, John ; Dey, Anind K.: Principles of smart home control. In: *Lecture Notes in Computer Science: UbiComp, Ubiquitous Computing* (2006), S. 19–34

[DO01] Dote, Y ; Ovaska, S J.: Industrial applications of soft computing: A review. In: *Proceedings of the IEEE* 89 (2001), Nr. 9, S. 1243–1265

[DPIP07] Doukas, Haris ; Patlitzianas, Konstantinos D. ; Iatropoulos, Konstantinos ; Psarras, John: Intelligent building energy management system using rule sets. In: *Building and Environment* 42 (2007), Nr. 10, S. 3562 – 3569

[DPN07] Dimitrov, Todor ; Pauli, Josef ; Naroska, Edwin: A probabilistic reasoning framework for smart homes. In: *Proceedings of the 5th international workshop on Middleware for pervasive and ad-hoc computing: held at the ACM/IFIP/USENIX 8th International Middleware Conference*, 2007, S. 1–6

[DRA+04] Demiris, George ; Rantz, Marilyn J. ; Aud, Myra A. ; Marek, Karen D. ; Tyrer, Harry W. ; Skubic, Marjorie ; Hussam, Ali A.: Older adults' attitudes towards and perceptions of 'smart home' technologies: A pilot study. In: *Informatics for Health and Social Care* 29 (2004), Nr. 2, S. 87–94

[Dun73] Dunn, J. C.: A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. In: *Journal of Cybernetics* 3 (1973), Nr. 3, S. 32–57

[Ech09] Echelon Corporation: *Introduction to the LonWorks Platform, Revision 2*. LonWorks general manual 078-0183-01B. 2009

[ECPC11] Erickson, V.L. ; Carreira-Perpinan, M.A. ; Cerpa, A.E.: OBSERVE: Occupancy-based system for efficient reduction of HVAC energy. In: *Information Processing in Sensor Networks (IPSN), 10th International Conference on*, 2011, S. 258 –269

[Fan70] Fanger, P. O.: *Thermal Comfort*. Copenhagen, Denmark : Danish Technical Press, 1970

[Fan01] Fanger, P. O.: Human requirements in future air-conditioned environments. In: *International Journal of Refrigeration* 24 (2001), Nr. 2, S. 148 – 153

[FCNL01] Fu, Tak-chung ; Chung, Fu-lai ; Ng, Vincent ; Luk, Robert: Pattern discovery from stock time series using self-organizing maps. In: *Workshop Notes of KDD Workshop on Temporal Data Mining*, 2001, S. 26–29

[Fer99] Ferber, Jacques: *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*. 1st. Boston, MA, USA : Addison-Wesley Longman Publishing Co., Inc., 1999

[FHNP07] Faruqui, Ahmad ; Hledik, Ryan ; Newell, Sam ; Pfeifenberger, Hannes: The power of 5 percent. In: *The Electricity Journal* 20 (2007), Nr. 8, S. 68 – 77

[Fis03] Fisher, Bernard: Fuzzy environmental decision-making:Applications to air pollution. In: *Atmospheric Environment* 37 (2003), Nr. 14, S. 1865–1877

[Fis08] Fischer, Corinna: Feedback on household electricity consumption: a tool for saving energy? In: *Energy Efficiency* 1 (2008), Nr. 1, S. 79–104

[FMMCM05] Frias-Martinez, E. ; Magoulas, G. ; Chen, S. ; Macredie, R.: Modeling human behavior in user-adaptive systems: Recent advances using soft computing techniques. In: *Expert Systems with Applications* 29 (2005), Nr. 2, S. 320 – 329

[Fog03] Fogg, B. J.: *Persuasive Technology: Using Computers to Change What We Think and Do (Interactive Technologies)*. Morgan Kaufmann Publishers, 2003

[Fra06] Frank, Brian: *oBIX Specification, obix-0.12.2.* 2006. – Available: `http://xml.coverpages.org/OBIX-CD02-19017.pdf`

[FSS10] Faruqui, Ahmad ; Sergici, Sanem ; Sharif, Ahmed: The impact of informational feedback on energy consumption: A survey of the experimental evidence. In: *Energy* 35 (2010), Nr. 4, S. 1598 – 1608. – Demand Response Resources: the US and International Experience

[GB01] Grankvist, Gunne ; Biel, Anders: The importantce of beliefs and purchase criteria in the choice of eco-labeled food products. In: *Journal of Environmental Psychology* 21 (2001), Nr. 4, S. 405–410

[GBM+07] Gaglia, Athina G. ; Balaras, Constantinos A. ; Mirasgedis, Sevastianos ; Georgopoulou, Elena ; Sarafidis, Yiannis ; Lalas, Dimitris P.: Empirical assessment of the Hellenic non-residential building stock, energy consumption, emissions and potential energy savings. In: *Energy Conversion and Management* 48 (2007), Nr. 4, S. 1160 – 1175

[GP01] Galeano, Pedro ; Pena, Daniel: Multivariate Analysis In Vector Time Series / Universidad Carlos III, Departamento de Estadística y Econometría. 2001. – Statistics and Econometrics Working Papers

[GRSM05] Galasiu, A.D. ; Reinhart, C.F. ; Swinton, M.C. ; Manning, M.M.: Assessment of Energy Performance of Window Shading Systems at the Canadian Centre for Housing Technology / NRC-CNRC, National Research Council Canada. 2005. – Forschungsbericht

[Gru69] Grubbs, F. E.: Procedures for detecting outlying observations in samples. In: *Technometrics* 11 (1969), Nr. 1, S. 1–21

[Gru95] Gruber, Thomas R.: Toward principles for the design of ontologies used for knowledge sharing. In: *The International Journal of Human-Computer Studies* 43 (1995), Dezember, Nr. 5–6, S. 907–928

[GSR10] Gwisdorf, B. ; Stepanescu, S. ; Rehtanz, C.: Effects of demand side management on the planning and operation of distribution grids. In: *Innovative Smart Grid Technologies Conference Europe (ISGT Europe), IEEE PES*, 2010, S. 1 –5

[Har03] Harper, Richard: *Inside the Smart Home.* London, England : Springer, 2003

[HBEK02] Handy, Susan L. ; Boarnet, Marlon G. ; Ewing, Reid ; Killingsworth, Richard E.: How the built environment affects physical activity: Views from urban planning. In: *American Journal of Preventive Medicine* 23 (2002), Nr. 2, Supplement 1, S. 64–73. – Innovative approaches understanding and influencing physical activity

[HBV02] Halkidi, Maria ; Batistakis, Yannis ; Vazirgiannis, Michalis: Cluster validity methods: Part I. In: *SIGMOD Rec.* 31 (2002), June, S. 40–45

[HC03] Heierman III, E.O. ; Cook, D.J.: Improving home automation by discovering regularly occurring device usage patterns. In: *Data Mining, ICDM. 3rd IEEE International Conference on*, 2003, S. 537 – 540

[HCC+04] Hagras, H. ; Callaghan, V. ; Colley, M. ; Clarke, G. ; Pounds-Cornish, A. ; Duman, H.: Creating an ambient-intelligence environment using embedded agents. In: *IEEE Intelligent Systems* 19 (2004), Nr. 6, S. 12–20

[HE09] Hendron, R. ; Engebrecht, C.: Building America research benchmark definition: Updated December 2009 / National Renewable Energy Laboratory. 2009. – Forschungsbericht

[HHL+09] Hoes, P. ; Hensen, J.L.M. ; Loomans, M.G.L.C. ; de Vries, B. ; Bourgeois, D.: User behavior in whole building simulation. In: *Energy and Buildings* 41 (2009), Nr. 3, S. 295 – 302

[Hil05] Hiller, Carl: Hot Water Distribution System Research – Phase I Final Report / California Energy Commission Report No. C-03-23A. 2005. – Forschungsbericht. Available: `http://www.cuwcc.org/WorkArea/downloadasset.aspx?id=2178`

[Hin99] Hindus, Debby: The importance of homes in technology research. In: *Proceedings of the Second International Workshop on Cooperative Buildings, Integrating Information, Organization, and Architecture*. London, UK : Springer-Verlag, 1999, S. 199–207

[HJO06] Ha, Seung ; Jung, Hong ; Oh, Yong: Method to analyze user behavior in home environment. In: *Personal Ubiquitous Comput.* 10 (2006), January, S. 110–121. – ISSN 1617–4909

[HN02] Humphreys, Michael A. ; Nicol, J. F.: The validity of ISO-PMV for predicting comfort votes in every-day thermal environments. In: *Energy and Buildings* 34 (2002), Nr. 6, S. 667 – 684

[HNB10] Hargreaves, Tom ; Nye, Michael ; Burgess, Jacquelin: Making energy visible: A qualitative field study of how householders interact with feedback from smart energy monitors. In: *Energy Policy* 38 (2010), Nr. 10, S. 6111–6119

[HP10] Heiselberg, P. ; Perino, M.: Short-term airing by natural ventilation implication on IAQ and thermal comfort. In: *Indoor Air* 20 (2010), Nr. 2

[Hua08] Huang, A.: Similarity measures for text document clustering. In: *Proceedings of the Sixth New Zealand Computer Science Research Student Conference NZCSRSC*. Christchurch New Zealand, 2008, S. 49–56

[Hum95] Humphreys, Paul: Computational science and scientific method. In: *Minds and Machines* 5 (1995), S. 499–512

[ICKM11] Iglesias, Félix ; Cantos, Sergio ; Kastner, Wolfgang ; Montero, José A.: Behavioral profiles for building energy performance using eXclusive SOM. In: Iliadis, Lazaros (Hrsg.) ; Jayne, Chrisina (Hrsg.): *Engineering Applications of Neural Networks* Bd. 363, Springer Boston, 2011, S. 31–40

[IDA07] IDAE: Documento de condiciones de aceptación de Programas Informáticos Alternativos, Anexos / The Institute for Energy Diversification and Saving, (Spanish Government). Spain, 2007. – Forschungsbericht

[IDA08] IDAE: Plan de Acción 2008-2012 / The Institute for Energy Diversification and Saving, (Spanish Government). Spain, 2008. – Forschungsbericht

[IH85] Ingersoll, J. ; Huang, J.: Heating energy use management in residential buildings by temperature control. In: *Energy and Buildings* 8 (1985), Nr. 1, S. 27 – 35

[IK10] Iglesias, Félix ; Kastner, Wolfgang: Usage profiles for sustainable buildings. In: *Proc. of the 15th IEEE Conference on Emerging Techonologies and Factory Automation (ETFA' 10)*, 2010, S. 1–8

[IK11] Iglesias, Félix ; Kastner, Wolfgang: Clustering methods for occupancy prediction in smart home control. In: *Industrial Electronics (ISIE), 2011 IEEE International Symposium on*, 2011, S. 1321–1328

[IK12a] Iglesias, Félix ; Kastner, Wolfgang: Detecting user dissatisfaction in ambient intelligence environments. In: *Emerging Technologies Factory Automation (ETFA), IEEE 17th Conference on*, 2012, S. "to be published"

[IK12b] Iglesias, Félix ; Kastner, Wolfgang: Thermal Comfort Support application for smart home control. In: Novais, Paulo (Hrsg.) ; Hallenborg, Kasper (Hrsg.) ; Tapia, Dante I. (Hrsg.) ; RodrÃguez, Juan M. C. (Hrsg.): *Ambient Intelligence - Software and Applications* Bd. 153. Springer Berlin / Heidelberg, 2012, S. 109–118

[IKGR11] Iglesias, Félix ; Kastner, Wolfgang ; Gaceo, Sergio C. ; Reinisch, Christian: Electricity load management in smart home control. In: *12th Conference of International Building Performance Simulation Association, Building Simulation*. Sydney, November 2011, S. 957–964

[IKK] Iglesias, Félix ; Kastner, Wolfgang ; Kofler, Mario: Holistic smart home models for air quality and thermal comfort management. In: *Intelligent Journal of Intelligent Decision Technologies. Special issue: Pervasive and Context Aware Decision Technologies* . – "to be published"

[IKR11] Iglesias, Félix ; Kastner, Wolfgang ; Reinisch, Christian: Impact of user habits in smart home control. In: *Emerging Technologies Factory Automation (ETFA), IEEE 16th Conference on*, 2011, S. 1–8

[Int02] Intille, S.S.: Designing a home of the future. In: *Pervasive Computing, IEEE* 1 (2002), Nr. 2, S. 76 – 82

[IPCK12] Iglesias, Félix ; Palensky, Peter ; Cantos, Sergio ; Kupzog, Friederich: Demand side management for stand-alone hybrid power systems based on load identification. In: *Energies* 5 (2012), Nr. 11, S. 4517–4532

[JD88] Jain, Anil K. ; Dubes, Richard C.: *Algorithms for clustering data*. Upper Saddle River, NJ, USA : Prentice-Hall, Inc., 1988

[JFB97] Jaccard, Mark ; Failing, Lee ; Berry, Trent: From equipment to infrastructure: community energy management and greenhouse gas emission reduction. In: *Energy Policy* 25 (1997), S. 1965–1074

[JH02] Jacobs, P. ; Henderson, H.: State-of-the-Art Review Whole Building, Building Envelope, and HVAC Component and System Simulation and Design Tools / Architectural Energy Corporation. 2002. – Forschungsbericht

[JL04] Jaiyen, S. ; Lursinsap, C.: Non-Euclidean self-organizing classification using natural manifold distance. In: *Neural Networks, 2004 IEEE International Joint Conference on*, 2004, S. 799–802

[JMF99] Jain, A. K. ; Murty, M. N. ; Flynn, P. J.: Data clustering: A review. In: *ACM Computing Surveys* 31 (1999), Nr. 3, S. 264–323

[JPDD03] Jung, Yunjae ; Park, Haesun ; Du, Ding-Zhu ; Drake, Barry: A decision criterion for the optimal number of clusters in hierarchical clustering. In: *Journal of Global Optimization* 25 (2003), S. 91–111

[KA04] Khan, Shehroz S. ; Ahmad, Amir: Cluster center initialization algorithm for K-means clustering. In: *Pattern Recognition Letters* 25 (2004), Nr. 11, S. 1293 – 1302

[Kar03] Karypis, G.: *CLUTO: A Clustering Toolkit.* Minneapolis, MN: University of Minnesota, Dept. of Computer Science, 2003

[Kas03] Kastner, Wolfgang: In-home networks and their connection to the outside world. In: *Proc. of 8th International Conference on Intelligence in Next Generation Networks (ICIN '03)*, 2003

[KBR+07] Kleinberger, Thomas ; Becker, Martin ; Ras, Eric ; Holzinger, Andreas ; MÃ¼ller, Paul: Ambient intelligence in assisted living: Enable elderly people to handle future interfaces. In: Stephanidis, Constantine (Hrsg.): *Universal Access in Human-Computer Interaction. Ambient Interaction* Bd. 4555. Springer Berlin / Heidelberg, 2007, S. 103–112

[KC04] Khoo, Ivan ; Cook, Mark: Web-based occupant feedback for building energy management systems using fuzzy logic. In: *ASME Conference Proceedings* 2004 (2004), Nr. 41758, S. 661–670

[KC07] Kaushik, Alka R. ; Celler, B. G.: Characterization of PIR detector for monitoring occupancy patterns and functional health status of elderly people living alone at home. In: *Technol. Health Care* 15 (2007), December, S. 273–288

[Keo02] Keogh, Eamonn: Exact indexing of dynamic time warping. In: *Proc. of the 28th International Conference on Very Large Data Bases*, VLDB Endowment, 2002 (VLDB '02), S. 406–417

[KH09] Karnouskos, Stamatis ; Holanda, Thiago Nass d.: Simulation of a smart grid city with software agents. In: *Proceedings of the 2009 Third UKSim European Symposium on Computer Modeling and Simulation.* Washington, DC, USA : IEEE Computer Society, 2009 (EMS '09), S. 424–429

[Kir03] Kirschen, D.S.: Demand-side view of electricity markets. In: *Power Systems, IEEE Transactions on* 18 (2003), Mai, Nr. 2, S. 520 – 527

[KK10] Kofler, M.J. ; Kastner, W.: A knowledge base for energy-efficient smart homes. In: *Energy Conference and Exhibition (EnergyCon), IEEE International*, 2010, S. 85 –90

[KKR10] Kastner, Wolfgang ; Kofler, Mario J. ; Reinisch, Christian: Using AI to realize energy efficient yet comfortable smart homes. In: *Proc. of of 8th IEEE International Workshop on Factory Communication Systems (WFCS '10)*, 2010, S. 169–172

[KL01] Kastner, Wolfgang ; Leupold, Markus: Discovering internet services: Integrating intelligent home automation systems to plug and play networks. In: *Proc. of Intl. Workshop on Innovative Internet Computing Systems (IICS '01)* Bd. 2060, Springer Verlag, 2001, S. 67–78

[KN04] Kastner, Wolfgang ; Neugschwandtner, Georg: Service interfaces for field-Level home and building automation. In: *Proc. of 5th IEEE International Workshop on Factory Communication Systems (WFCS '04)*, 2004, S. 103–112

[KN09] Kastner, Wolfgang ; Novak, Thomas: Functional safety in building automation. In: *Proc. of 14th IEEE Conference on Emerging Technologies and Factory Automation (ETFA '09)*, 2009, S. 1–8

[KNSN05] Kastner, W. ; Neugschwandtner, G. ; Soucek, S. ; Newmann, H.M.: Communication systems for building automation and control. In: *Proceedings of the IEEE* 93 (2005), june, Nr. 6, S. 1178 –1203

[KNX04] KNX Association: *KNX Handbook*. Version 1.1. 2004

[Koh90] Kohonen, T.: The self-organizing map. In: *Proceedings of the IEEE* 78 (1990), September, Nr. 9, S. 1464–1480

[KRK11] Kofler, Mario J. ; Reinisch, Christian ; Kastner, Wolfgang: An intelligent knowledge representation of smart home energy parameters. In: *Proceedings of the World Renewable Energy Congress (WREC '11)*. Linköping, Sweden, May 2011, S. 921–928

[KRK12] Kofler, Mario J. ; Reinisch, Christian ; Kastner, Wolfgang: A semantic representation of energy-related information in future smart homes. In: *Energy and Buildings* 47 (2012), Nr. 0, S. 169–179

[KS98] Kohonen, Teuvo ; Somervuo, Panu: Self-organizing maps of symbol strings. In: *Neurocomputing* 21 (1998), Nr. 1-3, S. 19–30

[KS04] Kastner, Wolfgang ; Scheichelbauer, Horst: UPnP connectivity for home and building automation. In: *Proc. of IASTED Intl. Conference on Parallel and Distributed Computing and Networks (PDCN '04)*, 2004, S. 619–624

[KS09] Kim, Hak S. ; Son, Jin H.: User-pattern analysis framework to predict future service in intelligent home network. In: *Intelligent Computing and Intelligent Systems, ICIS. IEEE International Conference on* Bd. 2, 2009, S. 818 –822

[KS11] Kastner, Wolfgang ; Szucsich, Stefan: Accessing KNX networks via BACnet/WS. In: *Industrial Electronics (ISIE), IEEE International Symposium on*, 2011, S. 1315 –1320

[KSCPM00] Kirschen, D.S. ; Strbac, G. ; Cumperayot, P. ; de Paiva Mendes, D.: Factoring the elasticity of demand in electricity prices. In: *Power Systems, IEEE Transactions on* 15 (2000), Mai, Nr. 2, S. 612 –617

[KSP+10] Ko, Jinhee ; Shin, In-Hye ; Park, Gyung-Leen ; Kwak, Ho-Yong ; Ahn, Khi-Jung: Design of a multi-agent system for personalized service in the smart grid. In: Kim, Tai-hoon (Hrsg.) ; Stoica, Adrian (Hrsg.) ; Chang, Ruay-Shiung (Hrsg.): *Security-Enriched Urban Computing and Smart Grid* Bd. 78. Springer Berlin Heidelberg, 2010, S. 267–273

[KWCB09] Kotey, N.A. ; Wright, J.L. ; Collins, M.R. ; Barnaby, C.S.: Solar gain through windows with shading devices: Simulation versus measurement. In: *ASHRAE Transactions* 115 (2009), Nr. 2

[LB99] Leaman, Adrian ; Bordass, Bill: Productivity in buildings: The 'killer' variables. In: *Building Research & Information* 27 (1999), Nr. 1, S. 4–19

[LBS10] Li, Xiaoli ; Bowers, C.P. ; Schnier, T.: Classification of energy consumption in buildings with outlier detection. In: *Industrial Electronics, IEEE Transactions on* 57 (2010), Nr. 11, S. 3639 –3644

[LD05] Liang, Jian ; Du, Ruxu: Thermal comfort control based on neural network for HVAC application. In: *Control Applications, CCA '05. Proceedings of IEEE Conference on*, 2005, S. 819 –824

[Lee10] Lee, Jaewook: Conflict resolution in multi-agent based intelligent environments. In: *Building and Environment* 45 (2010), Nr. 3, S. 574 – 585

[LEMM09] Laitner, J. A. ; Ehrhardt-Martinez, K. ; McKinney, V.: Examining the scale of the behavior related energy efficiency continuum. In: *Proceedings of the ECEEE Summer Study*, 2009, S. 217–223

[Lip99] Lipkus, Alan: A proof of the triangle inequality for the Tanimoto distance. In: *Journal of Mathematical Chemistry* 26 (1999), S. 263–265

[LKSH02] Lutz, James ; Klein, Gary ; Springer, David ; Howard, D. B.: Residential Hot Water Distribution Systems: Roundtable Session / Lawrence Berkeley National Laboratory. 2002. – Forschungsbericht

[LNH02] Lichodzijewski, Peter ; Nur ; Heywood, Malcolm I.: Host-based intrusion detection using self-organizing maps. In: *Neural Networks, IJCNN. Proceedings of the International Joint Conference on* Bd. 2, 2002, S. 1714–1719

[Lun96] Lundberg, Ante S.: Psychiatric aspects of air pollution. In: *Otolaryngology - Head and Neck Surgery* 114 (1996), Nr. 2, S. 227 – 231

[LVK10] Lampropoulos, I. ; Vanalme, G.M.A. ; Kling, W.L.: A methodology for modeling the behavior of electricity prosumers within the smart grid. In: *Innovative Smart Grid Technologies Conference Europe (ISGT Europe), IEEE PES*, 2010, S. 1 –8

[Mad09] MADSEN, P.P.: HomePort: An intelligent home control system. In: *4th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2009, S. 2519–2523

[MC85] MILLIGAN, Glenn ; COOPER, Martha: An examination of procedures for determining the number of clusters in a data set. In: *Psychometrika* 50 (1985), Nr. 2, S. 159 – 179

[MC97] MYER, A. ; CHAFFEE, C.: Life-cycle analysis for design of the Sydney Olympic Stadium. In: *World Renewable Energy Congress IV Renewable Energy, Energy Efficiency and the Environment* Bd. 10, 1997, S. 169–172

[MFS+07] MORI, T. ; FUJII, A. ; SHIMOSAKA, M. ; NOGUCHI, H. ; SATO, T.: Typical behavior patterns extraction and anomaly detection algorithm based on accumulated home sensor data. In: *Future Generation Communication and Networking (FGCN '07)* Bd. 2, 2007, S. 12 –18

[MH11] MOON, Jin W. ; HAN, Seung-Hoon: Thermostat strategies impact on energy consumption in residential buildings. In: *Energy and Buildings* 43 (2011), Nr. 2-3, S. 338 – 346

[Mil04] MILLS, Evan: Inter-comparison of North American residential energy analysis tools. In: *Energy and Buildings* 36 (2004), Nr. 9, S. 865–880

[MJ96] MAO, Jianchang ; JAIN, A.K.: A self-organizing network for hyperellipsoidal clustering (HEC). In: *Neural Networks, IEEE Transactions on* 7 (1996), jan, Nr. 1, S. 16 –29

[MJRM00] MAESSCHALCK, R. D. ; JOUAN-RIMBAUD, D. ; MASSART, D.L.: The Mahalanobis distance. In: *Chemometrics and Intelligent Laboratory Systems* 50 (2000), Nr. 1, S. 1 – 18

[ML07] MITRA, Nilo ; LAFON, Yves. *SOAP Version 1.2 Part 0: Primer.* 2007

[MM02] MCCALLEY, L.T. ; MIDDEN, Cees J.: Energy conservation through product-integrated feedback: The roles of goal-setting and social orientation. In: *Journal of Economic Psychology* 23 (2002), Nr. 5, S. 589–603

[MMWZ83] MIDDEN, Cees J. ; METER, Joanne F. ; WEENIG, Mieneke H. ; ZIEVERINK, Henk J.: Using feedback, reinforcement and information to reduce energy consumption in households: A field-experiment. In: *Journal of Economic Psychology* 3 (1983), Nr. 1, S. 65–86

[Moz98] MOZER, Michael C.: The Neural Network House: An environment that adapts to its inhabitants. In: *AAAI '98: Proceedings of the American Association for Artificial Intelligence Spring Symposium on Intelligent Environments*, ACM, 1998, S. 110–114

[MP08] MAHDAVI, Ardeshir ; PRÖGLHÖF, Claus: A model-based approach to natural ventilation. In: *Building and Environment* 43 (2008), Nr. 4, S. 620–627. – Part Special: Building Performance Simulation

[MP09] Mahdavi, Ardeshir ; Pröglhöf, Claus: User behaviour and energy performance in buildings. In: *IEWT, Int. Energy Economics Workshop TUV*. Vienna, Austria, 2009, S. 1–13

[MS05] Manning, Marianne ; Swinton, Mike: Effects of Thermostat Setting on Energy Consumption / Canada Mortgage and Housing Corporation. Ottawa, Canada, 2005. – Forschungsbericht

[MTIS04] Mori, T. ; Takada, A. ; Iwamura, Y. ; Sato, T.: Automatic human life summarization system in sensory living space. In: *Systems, Man and Cybernetics, IEEE International Conference on* Bd. 2, 2004, S. 1583 – 1588

[NAA10] Kap. 1 In: Nakashima, Hideyuki ; Aghajan, Hamid ; Augusto, Juan C.: *Handbook of Ambient Intelligence and Smart Environments*. Springer, 2010

[Nar07] Naroska, E.: AMIGO middleware-based ambient-intelligence user interaction. In: *KNX Scientific Conference*, 2007

[NS07] Neto, A. H. ; Sanzovo, F. A.: Use of simulation tools for manging buildings energy demand. In: *10th Conference of International Building Performance Simulation Association, Building Simulation* Bd. 29, 2007, S. 1883–1889

[OG01] Ovaska, S.J. ; Gao, Xiao-Zhi: Soft computing in industrial innovation: Case study on home appliance technology. In: *Systems, Man, and Cybernetics, IEEE International Conference on* Bd. 1, 2001, S. 70 –76

[OPC09] OPC Foundation: *OPC UA Part 1 - Overview and Concepts 1.01*. OPC specification document. 2009

[Org07] Organization, World H.: Housing, Energy and Thermal Comfort. A Review of 10 Countries Within the WHO European Region / WHO Regional Office for Europe. 2007. – Forschungsbericht

[Par93] Parsons, Kenneth C.: *Human Thermal Environments: The Effects of Hot, Moderate, and Cold Environments on Human Health, Comfort, and Performance*. Taylor & Francis, 1993

[Par08] Parker, Danny S.: Pilot evaluation of energy savings from residential energy demand feedback devices. In: *Solar Energy* (2008), Nr. January, S. 13

[PCCS07] Pamudurthy, S.R. ; Chandrakala, S. ; Chandra Sekhar, C.: Local density estimation based clustering. In: *Neural Networks, IJCNN. International Joint Conference on*, 2007, S. 1249 –1254

[PRMS08] Page, J. ; Robinson, D. ; Morel, N. ; Scartezzini, J.-L.: A generalised stochastic model for the simulation of occupant presence. In: *Energy and Buildings* 40 (2008), Nr. 2, S. 83–98

[Pyl99] Pyle, Dorian: *Data Preparation for Data Mining*. Morgan Kaufmann Publishers, Inc, 1999

[QZ02] Qian, W. ; Zhou, A.: Analyzing popular clustering algorithms from different viewpoints. In: *Journal of Software* 13 (2002), Nr. 8, S. 1382–1394

[RK11] Reinisch, Christian ; Kastner, Wolfgang: Agent based control in the smart home. In: *Proc. of 37th Annual Conference of the IEEE Industrial Electronics Society (IECON '11)*, 2011, S. 334 – 339

[RKIK11] Reinisch, Christian ; Kofler, Mario J. ; Iglesias, Félix ; Kastner, Wolfgang: ThinkHome: Energy efficiency in future smart homes. In: *EURASIP Journal on Embedded Systems* 2011 (2011), S. 18

[RKK10] Reinisch, Christian ; Kofler, Mario J. ; Kastner, Wolfgang: ThinkHome: A smart home as digital ecosystem. In: *Proceedings of 4th IEEE International Conference on Digital Ecosystems and Technologies (DEST '10)*, 2010, S. 256–261

[RN88] Rodgers, J. L. ; Nicewander, W. A.: Thirteen ways to look at the correlation coefficient. In: *The American Statistician* 42 (1988), Nr. 1, S. 59–66

[RNK03] Rasmussen, Matt ; Newman, Mark ; Karypis, George: *gCLUTO Documentation*. Minneapolis, MN: University of Minnesota, Dept. of Computer Science, 2003. – Version 1.0

[RRK08] Räsänen, Teemu ; Ruuskanen, Juhani ; Kolehmainen, Mikko: Reducing energy consumption by using self-organizing maps to create more personalized electricity use information. In: *Applied Energy* 85 (2008), Nr. 9, S. 830 – 840

[Saa08] Saad, A.: An overview of hybrid soft computing techniques for classifier design and feature selection. In: *Hybrid Intelligent Systems, HIS'08. 8th International Conference on*, 2008, S. 579 –583

[SAC+08] de Souto, M.C.P. ; de Araujo, D.S.A. ; Costa, I.G. ; Soares, R. ; Ludermir, T.B. ; Schliep, A.: Comparative study on normalization procedures for cluster analysis of gene expression datasets. In: *Neural Networks, IJCNN (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, 2008, S. 2792 –2798

[San03] Santamouris, M.: *Solar Thermal Technologies for Buildings: The State of the Art*. London, UK : James & James (Science Publishers), 2003

[SC04] Salvador, S. ; Chan, P.: Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In: *Tools with Artificial Intelligence, ICTAI. 16th IEEE International Conference on*, 2004, S. 576 – 584

[Sch07a] Schaeffer, S.: Graph clustering. In: *Computer Science Review* 1 (2007), August, Nr. 1, S. 27–64. – Survey of many graph clustering algorithms and techniques

[Sch07b] Schijndel, A.W.M. v.: *Integrated Heat Air and Moisture Modeling and Simulation*, Eindhoven, Technische Universiteit, Diss., 2007

[SD08] Sima, Chao ; Dougherty, Edward R.: The peaking phenomenon in the presence of feature-selection. In: *Pattern Recogn. Lett.* 29 (2008), August, S. 1667–1674

[SIR11] *Informe SIRENA 10. Anàlisi dels resultats d'estalvi energètic i d'aigua i propostes de futur* / Institut de Sostenibilitat – Universitat Politècnica de Catalunya. 2011. – Forschungsbericht

[SL78] STERNTHAL, Brian ; LEAVITT, Clark: The persuasive effect of source credibility : Tests of cognitive response. In: *Journal of Consumer Research* 4 (1978), Nr. 4, S. 252–260

[SMA+] SCHEVERS, Hans A. ; MITCHELL, John ; AKHURST, Paul ; MARCHANT, David M. ; BULL, Stuart ; MCDONALD, Kevin ; DROGEMULLER, Robin M. ; LINNING, Chris: Towards digital facility modelling for Sydney Opera House using IFC and semantic web technology. In: *Journal of IT in Construction* Bd. 12

[SMD+08] STAWITZ, C.C. ; MCGEHEE, H.S. ; DEVLIN, C.M. ; TAN, F. ; WONG, S.: Increasing awareness of residential energy consumption: Data analysis and presentation for ecoMOD, a sustainable housing initiative. In: *Systems and Information Engineering Design Symposium, SIEDS. IEEE*, 2008, S. 55 –59

[SN09] SPINDLER, Henry C. ; NORFORD, Leslie K.: Naturally ventilated and mixed-mode buildings – Part II: Optimal control. In: *Building and Environment* 44 (2009), Nr. 4, S. 750 – 761

[SNMN10] SUHARA, Y. ; NAKABE, T. ; MINE, G. ; NISHI, H.: Distributed demand side management system for home energy management. In: *IECON, 36th Annual Conference on IEEE Industrial Electronics Society*, 2010, S. 2430–2435

[SP07] SABATÉ, J. ; PETERS, Christoph: 50% CO2-reduction in Mediterranean social housing through detailed life-cycle analysis. In: *Climamed Proceedings*, 2007, S. 927–959

[SPF05] SOMMARUGA, Lorenzo ; PERRI, Antonio ; FURFARI, Francesco: DomoML-env: an ontology for human home interaction. In: *Proc. of 2nd Italian Semantic Web Workshop*. Trento, Italy, 2005

[SSBW08] SPONSELEE, A. ; SCHOUTEN, B. ; BOUWHUIS, D. ; WILLEMS, C.: Smart home technology for the elderly: Perceptions of multidisciplinary stakeholders. In: MÜHLHÄUSER, Max (Hrsg.) ; FERSCHA, Alois (Hrsg.) ; AITENBICHLER, Erwin (Hrsg.): *Constructing Ambient Intelligence* Bd. 11. Springer Berlin Heidelberg, 2008, S. 314–326

[SSJ02] SATO, Mika ; SATO, Yoshiharu ; JAIN, Lakhmi C.: *Fuzzy Clustering Models and Applications (Studies in Fuzziness and Soft Computing)*. Physica-Verlag HD, 2002

[SSKD11] SAUTER, Thilo ; SOUCEK, Stefan ; KASTNER, Wolfgang ; DIETRICH, Dietmar: The evolution of factory and building automation. In: *IEEE Industrial Electronics Magazine* 5 (2011), S. 35–48

[The11] The OSGi Alliance: *OSGi Service Platform*. Core Specification, Release 4, Version 4.3. 2011

[Tim06] HOBBS, Jerry E. (Hrsg.) ; PAN, Feng (Hrsg.): *Time Ontology in OWL*. 2006. – W3C Working Draft 27 September 2006. Available: `http://www.w3.org/TR/owl-time/`

[TK03] THEODORIDIS, Sergios ; KOUTROUMBAS, Konstantinos: *Pattern Recognition: Fourth Edition*. Academic Press (ELSEVIER), 2003

[TRF07] Tachwali, Y. ; Refai, H. ; Fagan, J.E.: Minimizing HVAC energy consumption using a wireless sensor network. In: *Industrial Electronics Society, IECON. 33rd Annual Conference of the IEEE*, 2007, S. 439 –444

[TTIH06] Takada, M. ; Tano, S. ; Iwata, M. ; Hashiyama, T.: A proposed home agent architecture to infer user feeling from user action pattern. In: *Systems, Man and Cybernetics, SMC '06. IEEE International Conference on* Bd. 6, 2006, S. 4818 –4824

[TZ02] Tang, Chun ; Zhang, Aidong: An iterative strategy for pattern discovery in high-dimensional data sets. In: *Proceedings of the 11th International Conference on Information and Knowledge Management*. New York, NY, USA : ACM, 2002 (CIKM '02), S. 10–17

[VA99] Verplanken, Bas ; Aarts, Henk: Habit, attitude, and planned behaviour: Is habit an empty construct or an interesting case of goal-directed automaticity? In: *European Review of Social Psychology* (1999), S. 101–134

[VH08] Van Hoof, J: Forty years of Fanger's model of thermal comfort: Comfort for all? In: *Indoor Air* 18 (2008), Nr. 3, S. 182 – 201

[VHAP00] Vesanto, Juha ; Himberg, Johan ; Alhoniemi, Esa ; Parhankangas, Juha: Self-organizing map in MATLAB: The SOM Toolbox. In: *In Proceedings of the MATLAB DSP Conference*, 2000, S. 35–40

[VHVR89] Van Houwelingen, Jeannet H. ; Van Raaij, W. F.: The effect of goal-setting and daily electronic feedback on In-Home energy use. In: *Journal of Consumer Research* 16 (1989), Nr. 1, S. 98–105

[War05] Warrenliao, T.: Clustering of time series data – A survey. In: *Pattern Recognition* 38 (2005), November, Nr. 11, S. 1857–1874

[WHH05] Wu, Jingwei ; Hassan, Ahmed E. ; Holt, Richard C.: Comparison of clustering algorithms in the context of software evolution. Los Alamitos, CA, USA : IEEE Computer Society, 2005, S. 525–535

[WM97] Wilson, Randall D. ; Martinez, Tony R.: Improved heterogeneous distance functions. In: *Journal of Artificial Intelligence Research* 6 (1997), S. 1–34

[Wu12] Wu, Junjie: *Advances in K-means Clustering: A Data Mining Thinking (Springer Theses)*. Springer, 2012

[Wyo00] Wyon, David: *Enhancing productivity while reducing energy use in buildings*. Rand, 2000, S. 233–257

[YD02] Yusta, José. M. ; Domínguez, José A.: Measuring and modelling of industrial demand response to alternative prices of the electricity. In: *Power Systems Computation Conference (PSCC)*. Seville (Spain), 2002

[YS05] Yao, Runming ; Steemers, Koen: A method of formulating energy load profile for domestic buildings in the UK. In: *Energy and Buildings* 37 (2005), Nr. 6, S. 663–671

[ZASZ+10] ZEDAN, F.M. ; AL-SHEHRI, A.M. ; ZAKHARY, S.Z. ; AL-ANAZI, M.H. ; AL-MOZAN, A.S. ; AL-ZAID, Z.R.: A nonzero sum approach to interactive electricity consumption. In: *Power Delivery, IEEE Transactions on* 25 (2010), Januar, Nr. 1, S. 66 –71

[ZB 05] ZB Alliance: *ZigBee specification*. ZigBee document 053474r06. 2005

[ZCL05] ZHENG, Xuezhi ; CAI, Zhihua ; LI, Qu: An experimental comparison of three kinds of clustering algorithms. In: *Neural Networks and Brain, ICNN&B '05. International Conference on* Bd. 2, 2005, S. 767–771

[ZR99] ZERVAS, G. ; RUGER, S.M.: The curse of dimensionality and document clustering. In: *Microengineering in Optics and Optoelectronics (Ref. No. 1999/187), IEE Colloquium on*, 1999, S. 19/1 –19/3

[ZWA05] ZHANG, H. ; WANG, F.-Y. ; AI, Y.: An OSGi and agent based control system architecture for smart home. In: *Networking, Sensing and Control. IEEE Proceedings*, 2005, S. 13–18

Dipl.-Ing. Félix Iglesias Vázquez

CV, last updated: December 1, 2012

Personal

- Born on July 31, 1980, Madrid (Spain).
- Address: Grundsteingasse 7-1-8/9, A-1160 Vienna (Austria)
- Passport / ID card: BF095607 (Spain) / 46352473K (Spain)
- Phone: +43 680 1347847
- Email: fiv@gmx.at

Education

- **Diploma of Advanced Studies in IT**, Ramon Llull University (La Salle), 2009
  – *Dissertation:* Obtaining of Energy Usage Profiles from Real Data.

- **Master's Degree in Electronics**, Ramon Llull University (La Salle), 2005
  – *Dissertation:* Energy Management of Hybrid Systems. Simulation and Control in FSA.
  – *Dist.:* graduated with special distinction.

- **Bachelor's Degree in Telecommunications**, Ramon Llull University (La Salle), 2002
  – *Dissertation:* Comparative Study of Control Algorithms in Chemical Processes.
  – *Dist.:* graduated with special distinction.

Research Interests / Specializations

- **Advanced Control and Simulation**.
- **Artificial Intelligence and Soft Computing**.
- **Data Mining, Cluster Analysis**.
- **Home and Building Automation**.

Languages

- Spanish: mother tongue.
- Catalan: mother tongue.
- English: well.
- German: B2 level.

Professional Activities

- **Junior Scientist**.
  Complex Energy Systems Group, Energy Department, AIT Austrian Institute of Technology, 2012.
  – *Area:* Fundamental research on Cyber-physical Energy Systems. Simulation of Communication Networks, Networked Control Systems and Co-simulation.

- **Project Assistant**.
  Automation Systems Group, Faculty of Informatics, Technische Universität Wien (TU), 2010–2012.
  – *Project:* ThinkHome, FFG Haus der Zukunft Plus – P822 170.

- **Researcher and Project Supervisor**. Ramon Llull University (La Salle), 2007–10.
  – *Project:* European Commission INTUBE (INTelligent Use of Building's Energy Information).

- **R&D on Electronics for Solar Photovoltaic**. Wattpic, 2007–08.

- **Domotics and Building Automation Projects**. Freelance, 2006–07.

- **Researcher**. Ramon Llull University (La Salle), 2004–05.
  – *Project:* GEINCO (Intellig. Consump. Manag.), supported by the Spanish Profit Program.

Teaching

- **Electronic Technology**, Ramon Llull University (La Salle), 2000-03 and 2008-09.
- **Electronics & Instrumentation**, IQS (Chemical Institute of Sarriá), 2003-04.
- **Physics: Electromagnetic Fields**, Ramon Llull University (La Salle), 2006-07.
- PDP degree: Building Control and Sustainable Architecture, URL (La Salle), 2007-10.
- Domotics II (Cedom II), Corporate Training and Cedom, 2008.
- Domotics I (Cedom), Corporate Training and Cedom, 2005-06 and 2008.
- EIB-KNX, Corporate Training and Cedom, 2008.
- Lonworks, Corporate Training and Cedom, 2005-06 and 2008.
- Control and Management of Building and Dwellings, Voltimum, 2005
- Master in Telecommunications Infrastructures: Domotics, URL (La Salle), 2004.
- Master in Communication Installations, UPC (Polytechnic University of Catalonia), 2004.
- Domotic Systems, Corporate Training and Asimelec, 2004-09.

Merits

- PAMMS 2012 Award of Scientific Excellence.
- EUROSOLAR 2008 Prize, section: *Initiatives for education/teaching in renewable energies.*
- Spanish Profit Program subvention for the GEINCO project, 2004.

Articles in Journals

u. r. Félix Iglesias and Wolfgang Kastner. Optimization of times series clustering for the discovery of building energy patterns. *Energies*, "under review".

. Félix Iglesias and Peter Palensky. Profile-based control for central domestic hot water distribution. In *IEEE Transactions on Industrial Informatics*, Special Section on "Building Automation, Smart Homes, and Communities", "under review".

2013 Félix Iglesias, Wolfgang Kastner and Mario Kofler. Holistic smart home models for air quality and thermal comfort management. In *Intelligent Journal of Intelligent Decision Technologies*, special issue: *Pervasive and Context Aware Decision Technologies* , "to be published".

2012 Félix Iglesias, Peter Palensky, Sergio Cantos and Friederich Kupzog. Demand side management for stand-alone hybrid power systems based on load identification. *Energies.* 2012; 5(11):4517–4532.

2011 Christian Reinisch, Mario J. Kofler, Félix Iglesias, and Wolfgang Kastner. ThinkHome: Energy efficiency in future smart homes. *EURASIP Journal on Embedded Systems*, 2011:18, 2011.

Conference Contributions

2012 - Félix Iglesias and Wolfgang Kastner. Detecting user dissatisfaction in ambient intelligence environments. In *Emerging Technologies Factory Automation (ETFA), IEEE 17th Conference on*, work-in-progress paper, "to be published". September 2012.

. Félix Iglesias and Wolfgang Kastner. Thermal comfort support application for smart home control. In Paulo Novais, Kasper Hallenborg, Dante I. Tapia, and Juan M. Corchado RodrÃguez, editors, *Ambient Intelligence - Software and Applications*, volume 153 of *Advances in Intelligent and Soft Computing*, pages 109-118. Springer Berlin / Heidelberg, 2012.

2011 Félix Iglesias, Wolfgang Kastner, Sergio Cantos and Christian Reinisch. Electricity load management in smart home control. In *12th Conference of International Building Performance Simulation Association, Building Simulation*, pages 957-964, Sydney, November 2011.

. Félix Iglesias, Wolfgang Kastner and Christian Reinisch. Impact of user habits in smart home control. In emphEmerging Technologies Factory Automation (ETFA), IEEE 16th Conference on, pages 1 -8, September 2011.

. Félix Iglesias, Sergio Cantos, Wolfgang Kastner and J. Antonio Montero. Behavioral Profiles for Building Energy Performance Using eXclusive SOM. In Lazaros Iliadis and Chrisina Jayne, editors, *Engineering Applications of Neural Networks*, volume 363 of *IFIP Advances in Information and Communication Technology*, pages 31-40. Springer Boston, 2011.

. Félix Iglesias and Wolfgang Kastner. Clustering methods for occupancy prediction in smart home control. In *Industrial Electronics (ISIE), IEEE International Symposium on*, pages 1321-1328, June 2011.

2010 Félix Iglesias Vazquez and Wolfgang Kastner. Usage profiles for sustainable buildings. In *Proc. of the 15th IEEE Conference on Emerging Techonologies and Factory Automation (ETFA' 10)*, pages 1-8, September 2010, Bilbao (Spain).

2009 Sergio Cantos, Félix Iglesias and Jordina Vidal. Comparison of standard and case-based user profiles in building's energy performance simulation. In *11th Conference of International Building Performance Simulation Association, Building Simulation*, pages 584-591, 2009, Glasgow (Scotland).

2008 Torsten Massek and Félix Iglesias. 'Solar Cube' ETSAV – A project for teaching, researching and monitoring on building integrated photovoltaic. *The Oxford Conference 2008*, Oxford (UK).

Non Scientific-Technical Publications

2008 Félix Iglesias. *Hipólito y Fedra.* Madrid: Éride, Nuevos Escritores. ISBN: 978-84-936845-4-9.

- *Special:* Finalist of the Éride First Literary Award (2009).