UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Departament d'Enginyeria Electrònica

"Load Forecasting on the User-side by means of Computational Intelligence Algorithms"

Tesi doctoral presentada per a l'obtenció del títol de Doctor per la Universitat Politècnica de Catalunya, dins el Programa de Doctorat en Enginyeria Electrònica

Juan José Cárdenas Araujo

Directores:

Dr. Jose Luis Romeral Martínez
Dr. Antoni Garcia Espinosa

Terrassa, Julio de 2013

Para mis padres, Lida y Edilberto.

Agradecimientos

Sería casi imposible, bueno no, seguro que es posible, pero sería muy largo y difícil nombrar a cada una de las personas que han intervenido para que yo esté aquí en este punto final de mi doctorado. Tal vez una de las más transcendentales es aquella que me cedió su puesto de trabajo, trabajo en el que me enteré de la convocatoria doctoral que me ha permitido proseguir con mis estudios hasta llegar a este punto. Cosas de la vida.

A los que no puedo dejar de mencionar, en cambio, son en primer lugar a mis directores de tesis, Jose Luis Romeral y Antoni García.

Jose Luis Romeral me ha guiado en el camino de la investigación, donde creo haber aprendido que es y cómo se hace investigación. Las principales ideas de este trabajo han sido construidas gracias a su guía y han funcionado gracias a sus consejos y asesoramiento.

Antoni García, mi codirector, también ha aportado mucho en mi formación como investigador. Debo agradecer además su asesoramiento en lo referente a cómo escribir con estilo y cómo difundir mis ideas de una manera más simple y efectiva.

Siguiendo en el ámbito laboral-académico, tengo que dar las gracias también a mis compañeros de trabajo y estudios del MCIA que han tenido que soportarme, la mayoría por varios años, que me han puesto a prueba y me han enseñado lo importante que es la flexibilidad, tolerancia y empatía para disfrutar del día a día.

Por otra parte, y ya en el ámbito familiar, tengo que agradecer a mis padres, Lida y Edilberto, y hermanos, Marco, Claudia y Mauricio, por ese soporte emocional y económico que me han brindado durante el transcurso de este trabajo. Ellos son un pilar fundamental que ha hecho que las cosas sean más fáciles y llevaderas, aún en los momentos más difíciles.

Si la familia ha sido un soporte emocional, principalmente en los malos momentos, los amigos lo han sido tanto en los malos como los buenos. Algunos han llegado en el momento más indicado para enseñarme nuevas cosas y puntos de vista. Un especial reconocimiento a Tomasz Michalski, Julio Cesar Urresty y Fabio Andrade.

Por último y no menos importante, tengo que agradecer a dos grandes mujeres, Fernanda, mi actual compañera sentimental y su hija, Brigitte. Ustedes dos han entrado en mi vida para mostrarme el significado de la felicidad. Ustedes dos han sido esa cuña que se pone a la mesa coja para que esté estable y sea una base firme para pararte en ella y alcanzar tus metas.

Gracias a todos.

Abstract

Nowadays, it would be very difficult to deny the need to prioritize sustainable development through energy efficiency at all consumption levels. In this context, an energy management system (EMS) is a suitable option for continuously improving energy efficiency, particularly on the user side. An EMS is a set of technological tools that manages energy consumption information and allows its analysis. EMS, in combination with information technologies, has given rise to intelligent EMS (iEMS), which, aside from lending support to monitoring and reporting functions as an EMS does, it has the ability to model, forecast, control and diagnose energy consumption in a predictive way. The main objective of an iEMS is to continuously improve energy efficiency (on-line) as automatically as possible.

The core of an iEMS is its load modeling forecasting system (LMFS). It takes advantage of historical information on energy consumption and energy-related variables in order to model and forecast load profiles and, if available, generator profiles. These models and forecasts are the main information used for iEMS applications for control and diagnosis. That is why in this thesis we have focused on the study, analysis and development of LMFS on the user side.

The fact that the LMFS is applied on the user side to support an iEMS means that specific characteristics are required that in other areas of load forecasting they are not. First of all, the user-side load profiles (LPs) have a higher random behavior than others, as for example, in power system distribution or generation. This makes the modeling and forecasting process more difficult. Second, on the user side --for example an industrial user-- there is a high number and variety of places that can be monitored, modeled and forecasted, as well as their precedence or nature. Thus, on the one hand, an LMFS requires a high degree of autonomy to automatically or autonomously generate the demanded models. And on the other hand, it needs a high level of adaptability in order to be able to model and forecast different types of loads and different types of energies.

Therefore, the addressed LMFS are those that do not look only for accuracy, but also adaptability and autonomy. Seeking to achieve these objectives, in this thesis work we have proposed three novel LMFS schemes based on hybrid algorithms from computational intelligence, signal processing and statistical theory.

The first of them looked to improve adaptability, keeping in mind the importance of accuracy and autonomy. It was called an evolutionary training algorithm (ETA) and is based on adaptive-network-based-fuzzy-inference system (ANFIS) that is trained by a multi-objective genetic algorithm instead of its traditional training algorithm. As a result of this hybrid, the generalization

capacity was improved (avoiding overfitting) and an easily adaptable training algorithm for new adaptive networks based on traditional ANFIS was obtained.

The second scheme deals with LMFS autonomy in order to build models from multiple loads automatically. Similar to the previous proposal, an ANFIS and a MOGA were used. In this case, the MOGA was used to find a near-optimal configuration for the ANFIS instead of training it. The LMFS relies on this configuration to work properly, as well as to maintain accuracy and generalization capabilities. Real data from an industrial scenario were used to test the proposed scheme and the multi-site modeling and self-configuration results were satisfactory. Furthermore, other algorithms were satisfactorily designed and tested for processing raw data in outlier detection and gap padding.

The last of the proposed approaches sought to improve accuracy while keeping autonomy and adaptability. It took advantage of dominant patterns (DPs) that have lower time resolution than the target LP, so they are easier to model and forecast. The Hilbert-Huang transform and Hilbert-spectral analysis were used for detecting and selecting the DPs. Those selected were used in a proposed scheme of partial models (PM) based on parallel ANFIS or artificial neural networks (ANN) to extract the information and give it to the main PM. Therefore, LMFS accuracy improved and the user-side LP noising problem was reduced. Additionally, in order to compensate for the added complexity, versions of self-configured sub-LMFS for each PM were used. This point was fundamental since, the better the configuration, the better the accuracy of the model; and subsequently the information provided to the main partial model was that much better.

Finally, and to close this thesis, an outlook of trends regarding iEMS and an outline of several hybrid algorithms that are pending study and testing are presented.

Resumen

En el contexto energético actual y particularmente en el lado del usuario, el concepto de sistema de gestión energética (EMS) se presenta como una alternativa apropiada para mejorar continuamente la eficiencia energética. Los EMSs en combinación con las tecnologías informáticas dan origen al concepto de iEMS, que además de soportar las funciones de los EMS, tienen la capacidad de modelar, pronosticar, controlar y supervisar los consumos energéticos. Su principal objetivo es el de realizar una mejora continua, lo más autónoma posible y predictiva de la eficiencia energética.

Este tipo de sistemas tienen como núcleo fundamental el sistema de modelado y pronóstico de consumos (Load Modeling and Forecasting System, LMFS). El LMFS está habilitado para pronosticar el comportamiento futuro de cargas y, si es necesario, de generadores. Es sobre estos pronósticos sobre los cuales el iEMS puede realizar sus tareas automáticas y predictivas de optimización y supervisión. Los LMFS en el lado del usuario son el foco de esta tesis.

Un LMFS en el lado del usuario, diseñado para soportar un iEMS requiere o demanda ciertas características que en otros contextos no serían tan necesarias. En primera estancia, los perfiles de los usuarios tienen un alto grado de aleatoriedad que los hace más difíciles de pronosticar. Segundo, en el lado del usuario, por ejemplo en la industria, el gran número de puntos a modelar requiere que el LMFS tenga por un lado, un nivel elevado de autonomía para generar de la manera más desatendida posible los modelos. Por otro lado, necesita un nivel elevado de adaptabilidad para que, usando la misma estructura o metodología, pueda modelar diferentes tipos de cargas cuya procedencia pude variar significativamente.

Por lo tanto, los sistemas de modelado abordados en esta tesis son aquellos que no solo buscan mejorar la precisión, sino también la adaptabilidad y autonomía. En busca de estos objetivos y soportados principalmente por algoritmos de inteligencia computacional, procesamiento de señales y estadística, hemos propuesto tres algoritmos novedosos para el desarrollo de un LMFS en el lado del usuario.

El primero de ellos busca mejorar la adaptabilidad del LMFS manteniendo una buena precisión y capacidad de autonomía. Denominado ETA, consiste del uso de una estructura ANFIS que es entrenada por un algoritmo genético multi objetivo (MOGA). Como resultado de este híbrido, obtenemos un algoritmo con excelentes capacidades de generalización y fácil de adaptar para el entrenamiento y evaluación de nuevas estructuras adaptativas basadas en ANFIS.

El segundo de los algoritmos desarrollados aborda la autonomía del LMFS para así poder generar modelos de múltiples cargas. Al igual que en la anterior propuesta usamos un ANFIS y un MOGA, pero esta vez el MOGA en vez de entrenar el ANFIS, se utiliza para encontrar la

configuración cuasi-óptima del ANFIS. Encontrar la configuración apropiada de un ANFIS es muy importante para obtener un buen funcionamiento del LMFS en lo que a precisión y generalización respecta. El LMFS propuesto, además de configurar automáticamente el ANFIS, incluyó diversos algoritmos para procesar los datos puros que casi siempre estuvieron contaminados de datos espurios y gaps de información, operando satisfactoriamente en las condiciones de prueba en un escenario real.

El tercero y último de los algoritmos buscó mejorar la precisión manteniendo la autonomía y adaptabilidad, aprovechando para ello la existencia de patrones dominantes de más baja resolución temporal que el consumo objetivo, y que son más fáciles de modelar y pronosticar. La metodología desarrollada se basa en la transformada de Hilbert-Huang para detectar y seleccionar tales patrones dominantes. Además, esta metodología define el uso de modelos parciales de los patrones dominantes seleccionados, para mejorar la precisión del LMFS y mitigar el problema de aleatoriedad que afecta a los consumos en el lado del usuario. Adicionalmente, se incorporó el algoritmo de auto configuración que se presentó en la propuesta anterior para hallar la configuración cuasi-óptima de los modelos parciales. Este punto fue crucial puesto que a mejor configuración de los modelos parciales mayor es la mejora en precisión del pronóstico final.

Finalmente y para cerrar este trabajo de tesis, se realizó una prospección de las tendencias en cuanto al uso de iEMS y se esbozaron varias propuestas de algoritmos híbridos, cuyo estudio y comprobación se plantea en futuros estudios.

Keywords

Energy efficiency
Load profile
Load forecasting
User side
Energy management system
Intelligent energy management system
Load modeling and forecasting system
Adaptability
Autonomy
Computational intelligence
ANFIS
ANN
Genetic algorithm
Multi-objective genetic algorithm
Hilbert-Huang transform
Hilbert-spectrum analysis

CONTENTS

List of figures

List of tables

List of abbreviations by category

Energy management

EMS:	Energy Management System
iEMS:	intelligent EMS
HVAC:	Heating-Ventilation and Air Condition
LP:	Load Profile
LMF:	Load Modeling and Forecasting
LMFS:	LMF System
DP:	Dominant Patterns
MAPE:	Mean Absolute Percent Error
STLF:	Short-term Load Forecasting

Computational Intelligence

CI:	Computational Intelligence
ANFIS:	Adaptive-Network-Fuzzy Inference System
e-ANFIS:	exponential ANFIS
G-ANFIS:	Genetic ANFIS
ANN:	Artificial Neural Network
NF:	Neuro Fuzzy
ETA:	Evolutionary Training Algorithm
FS:	Fuzzy System
GA:	Genetic Algorithm
MOGA:	Multi-Objective GA
PM:	Partial Models

PM-DP: Partial Models of Dominant Patterns

RMSE: Root-Mean-Squared Error

Signal processing and other algorithms

WT: Wavelet Transform

HHT: Hilbert-Huang Transform

HSA: Hilbert-Spectrum Analysis

Main Equation and Graphic Nomenclature

t: Target

x: Possible inputs

y: Model output

N: Number of samples

Trn: training

Chk: checking

mf: membership function

in_mf: input membership function

out_mf: output membership function

1. Introduction

This first chapter is dedicated to introducing our motivations, starting hypothesis, the thesis and our objectives. Therefore, at the end of the Chapter the reader should have a general idea of the basis, targets, work performed and the structure of the thesis.

CONTENTS:

1.1. Introduction and motivations

Due to the current global energy scenario, ***energy efficiency*** is increasingly important at all levels of generation and consumption. On the one hand, oil reserves are depleting; political problems with oil supplying countries continuously destabilize oil prices; catastrophes such as 2011's Fukushima Daiichi's nuclear disaster are casting doubt on the safety of nuclear power plants; and renewable energies must still overcome energy storage problems due to their stochastic generation. On the other hand, climate change from greenhouse gas emissions is a serious problem to be resolved.

In that sense, according to the 450 scenario of the World Energy Outlook 2011 [1], more than 50% of CO_2 savings come from energy efficiency (Figure 1.1). In other words, the most important contribution to reaching energy security and climate goals actually comes from the energy that we do not consume. This fact highlights the importance of energy efficiency.

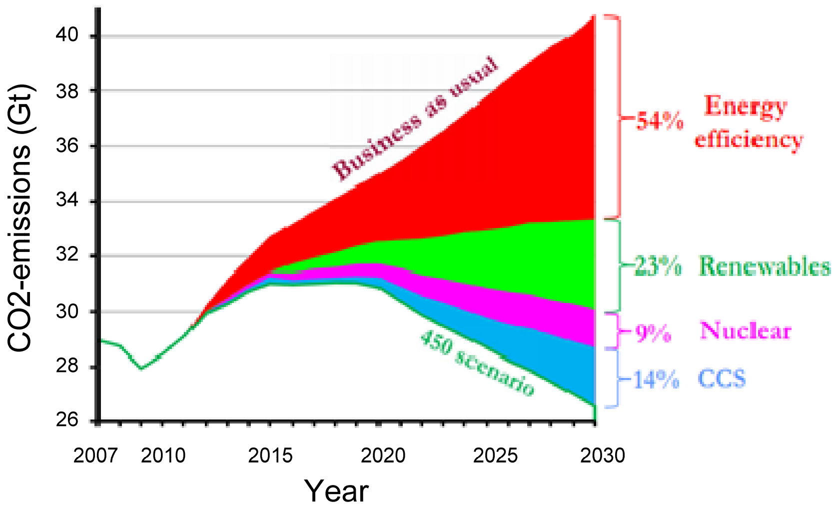


Figure 1.1. 450 scenario, World Energy Outlook 2011.

For this reason, it is increasingly important to use energy efficiently at every link in the energy chain, but even more on the user-side. It is in this field, especially for industrial users and tertiary buildings, where energy management systems (EMSs) are taking an important role by developing much more advanced and autonomous functions. They are moving from merely monitoring, reporting and supporting energy audits to having direct control over the loads, diagnostic functions and shape of the load demand, etc. As a result, the concept of ***intelligent EMS (iEMS)*** arises. Figure 1.2 depicts the concept of an iEMS.

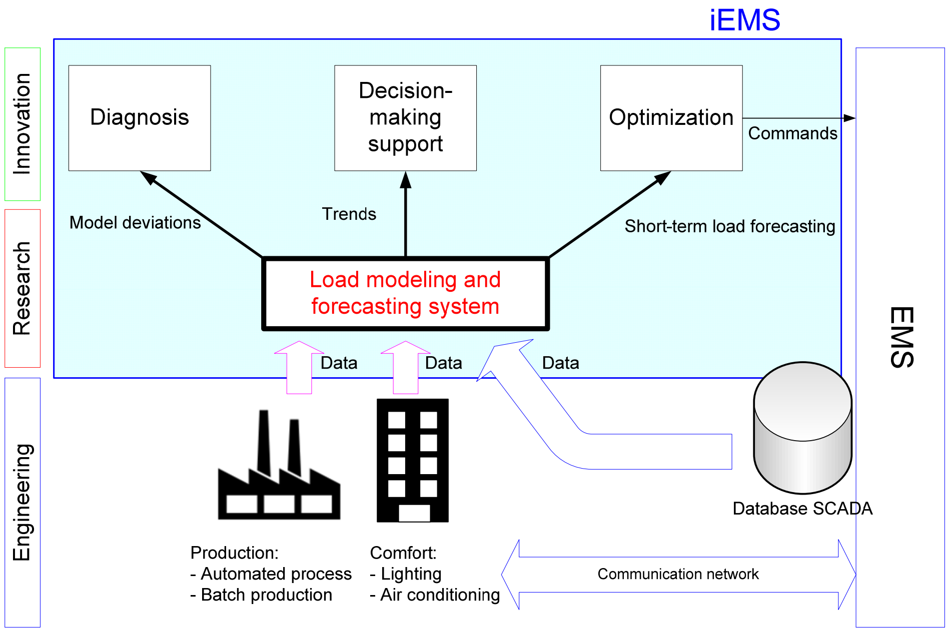


Figure 1.2. Block diagram of the proposed iEMS.

Furthermore, the new paradigms of smart-grid, micro-grids and distributed energy resources are changing the way of how the user consumes electric energy. Nowadays, the moment when the user uses energy and the resource from which it is taken affects energy prices and greenhouse emissions. In order to be able to optimally manage and control these key performance indicators, the user has to know how, where and when to use the energy. Furthermore, due to the high level of monitoring associated with these new paradigms, the availability of energy databases is rising. Thus, ***iEMSs, together with their advanced functions, are important technologies for supporting these new requirements and taking advantage of these huge energy databases for energy efficiency***.

1.2. Research area

In order to support these new advanced and autonomous functions in an iEMS, ***load modeling and forecasting (LMF)*** of energy consumption take a critical role. They take advantage of an energy database to build models of load demand at different time resolutions (from monthly to quarter-hourly), at different levels (overall consumption, specific department consumption, process consumption, etc.) and for forecasting at different time horizons (e.g., a day, week, month ahead).

Models are used to forecast consumption. The optimization modules can use these forecasts in order to detect demand peaks and take actions over the controllable loads to avoid them. Diagnosis modules based on the "healthy" patterns of consumption can track normal consumption under normal operation conditions and detect anomalous consumption. When this is detected, the diagnosis can initiate a specific algorithm for locating energy waste. Therefore, the iEMS can avoid energy waste in real time. Daily, weekly and monthly forecasts can support cost estimations and decision-making by administrative departments (e.g., scheduling workdays, on-off production lines, etc.). ***Thus, load modeling and forecasting (LMF) is the basis of an iEMS on the user side and the main topic of research for this thesis work.***

On the other hand, LMF has been a very important subject in energy management by utilities, mainly because of the necessity to match energy production with demand. Furthermore, this has been an interesting topic in the scientific literature for several years. Computational Intelligence (CI) algorithms have been commonly investigated as a tool for facing this problem, with successful results. Much of these algorithms are based on Artificial Neural Networks (ANN), Fuzzy Systems (FS) and different kinds of hybrid methods based on them.

However, only a few studies on LMF have been carried out on the user side. There is an exception regarding research in the field of heating, ventilation and air condition systems (HVAC), mainly for energy optimization in buildings. CI algorithms also have wide diffusion in this field.

The specific subject of LMF on the user side must still face some special issues that, so far, have not been investigated in depth. What's more, these issues have greater influence on the user side than on utilities or the energy market. They are:

- Random user behavior increase noisy load profile (LP) problem (the smaller the demand, the more noise the LP)
- Multiple and different types of places to be modeled, and
- Increasing energy databases to manage and model.

Hence, in this work, we study and research an LMF in the specific area of large energy users, such as factories and big buildings, with different time resolutions, supported by CI algorithms and signal processing functions.

1.3. Starting hypothesis and thesis

In order to tackle these challenging conditions in LMF on the user-side, in this work we define the next starting hypothesis:

- Intelligent control-optimization and diagnostic applications supported by the models are the key for an iEMS. Hence, one of ***the most important steps in improving energy consumption is to obtain a suitable LMF system***.
- In order for an iEMS to manage the huge available energy databases coming from, for example, smart-grids, micro-grids and other sources of energy databases, the LMF system has to be ***adaptive (multi-site modeling)*** and ***autonomous (self-configured)***.

- Taking into account the success of CI in areas like LMF for utilities, energy price forecasting and for time series modeling and forecasting in general, ***CI algorithms are currently the most appropriate for LMF on the user-side***. Neural networks (NN), adaptive-network-based fuzzy inference systems (ANFIS) and neuro-fuzzy (NF) systems are noteworthy algorithms for LMF. ***Hybrid algorithms are the best alternative for improving the current and classic structures.***
- ***Hybrid algorithms, that are hybrids between adaptive networks and evolutive algorithms,*** have the potential not only to increase accuracy, but also to add autonomy and adaptability. Autonomy means the capacity to auto-configure and adaptability refers to the capacity to model different sites using the same modeling scheme or structure.
- Given that we can find different ***patterns when analyzing demand load profiles*** on the user-side at ***different time resolutions*** and with ***different repetition frequencies***, such patterns can be used to give the LMF system ***extra information freely or with less degree of randomness (noisy LP problem) than the target LP.***

We can sum up the former hypothesis in the global thesis hypothesis:

"By combining CI tools with mathematical and statistical functions (hybrid algorithms), we can obtain an accurate, adaptive (multi-site) and autonomous (self-configured) LMF system, which is appropriate for iEMS implementation on the user-side."

This sentence depicts the thesis of the work here presented.

1.4. Objectives

Consistent with the thesis, the general and specific objectives below are defined in this work:

- ✓ ***Study and analysis of LMF process on the user-side.***
  - Study and analyze main troubles and challenging conditions for LMF on the user-side, and develop different approaches to solve them.
  - Analyze and pre-process energy databases. For example, detect and replace outliers and gaps in energy database to get reliable datasets from real raw data.
- ✓ ***Study and propose new structures and schemes for LMF on the user-side based on CI algorithms and mathematical and statistical functions.***
  - Study, design and test new hybrid structures based on CI algorithms for LF on the user side.
  - Study, design and test new hybrid self-configured structures based on CI algorithms.
  - Analyze dominant load profile patterns of energy demand as a basis for researching, developing and testing new schemes or methodologies for LMF.

In practice, these objectives must lead us to the main objective of obtaining an ***accurate***, ***adaptive*** (multi-site), ***autonomous*** (self-configured) ***LMF system (LMFS)***, appropriate for implementation in an iEMS.

1.5. Outline of the thesis

Chapter 1 includes the introduction, motivations, research area, hypothesis, thesis and objectives of the thesis. At the end of reading this chapter, the reader should have a clear global idea of the work performed in this thesis.

Chapter 2 presents the state of the art, starting with the evolution from EMS to intelligent EMS, and ending with the main research projects found regarding LMF during the development of this thesis.

Chapter 3 conducts a deep analysis of the main problems found in LMF and outlines the proposed solutions. This chapter is the basis and starting point for the subsequent chapters.

Chapter 4 proposes hybrid algorithms based on single modeling structures, oriented toward improving autonomy and the capacity of the LMFS to generalize for iEMS. A general Evolutionary Training Algorithm (ETA) for adaptive-network based fuzzy inference system (ANFIS) is proposed and tested with a modified ANFIS (e-ANFIS), which uses exponential membership functions for the output functions instead of polynomials.

Derived from these proposals, two notable congress papers were presented. One of them presented the ETA approach for genetic ANFIS training [2] and the other one, as a continuation of the work, presented the e-ANFIS trained by ETA [3].

Chapter 5 addresses the multi-site and self-configuration of LMFS by proposing a novel approach of a self-configured LMF framework based on a genetic algorithm (GA) and an ANFIS.

Related to the proposed self-configured LMF framework, a journal paper was written and published [4].

Chapter 6 introduces a new methodological scheme to take advantage of dominant patterns found in the LP, which is based on partial models of these dominant patterns. LMFS accuracy is the main subject addressed in this chapter, but adaptability and autonomy are inherited from previous proposals [5].

As a result of the studies and proposed approach to dominant pattern detection and partial model forecasting, a journal paper has been presented and is currently under revision.

Finally, **Chapter 7** summarizes the conclusions of the thesis, highlighting the main results obtained and, in **Chapter 8,** we present our outlook for future works on both iEMS trends and new LMFS approaches.

2. State of the Art

It is a fact load modeling and forecasting (LMF) on user-side are growing interest. Therefore, EMSs are including the capability of LMF in order to take autonomous decisions and to improve its energy optimization functions. Improving of this item could enhance the system performance and make more automatic the EMS without affect the production or comfort.

In this chapter, we first introduce the concept of EMS, intelligent EMS, user-side, load profile and load modeling and forecasting (LMF). Then we move to check the state of the art in LMF at different related areas as utilities, energy market, general time series modeling and finally on the user-side energy consumptions. We highlight the main trends, the most remarkable algorithms and problems found in LMF on the user-side. Next, we outline how we are going to tackle these problems. Finally, we outline the basic load forecasting algorithms and pre-processing tools used in the developing of this thesis work.

CONTENTS:

2.1. Energy management system (EMS) on the user-side

Due to the current energy problem, the interest in Energy Management (EM) is growing increasingly. Particularly in the field of industrial and building consumption, here called user-side[1], Energy Management Systems[2] (EMSs) are an excellent option to improve energy efficiency and company competitiveness. “Doing more with less” [6] it is the idea behind energy efficiency improvement by means of EMS.

When we talk about EMS on the user-side, we can refer to two fields of action. One is the ***industrial EMS***, also known as ***Enterprise Energy Management System (EEMS)*** [7], which is associated with SCADA and others industrial software systems. The other one is ***building EMS***, called ***Building Energy Management Systems (BEMS),*** which is associated with the concept of intelligent building [8-10].

So far in the industrial sector, EMSs have focused on the monitoring and “*passive*” management of energy, as outlined in [6] and [11]. Figure 2.1 shows the basic structure of these management systems. With “*passive*”, we mean that an EMS does not take autonomous actions for improvement of energy efficiency, as it will be explain further.

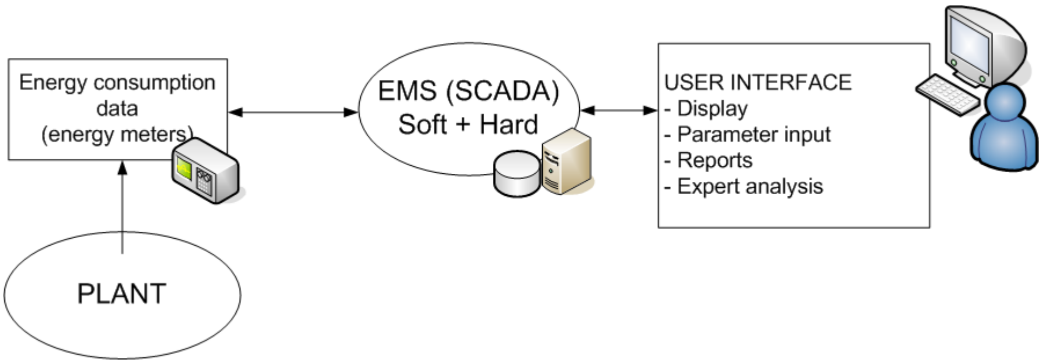


Figure 2.1. Basic structure of current EMS.

Typical EMS is based on the collection of information through energy meters (electricity, gas, water, etc.), which is taken by a SCADA system or another software for later information processing and management. For example, collecting, storing and presenting the data appropriately for the users. The software is also able to analyze data and generate reports to identify critical points of consumption. Based on these reports, expert users can propose actions in order to improve the energy efficiency.

[1] User-side makes reference all consumption in domestic and industrial users. However, we have focused in Industrial and big building users only. These kinds of users normally already have an EMS.
[2] In other contexts, EMS can refer to the set of policies taken by a company with the objective of improvement of energy efficiency. Here we refer only to the concept of EMS as a type of Energy SCADA.

The main advantage in these monitoring systems is to enable the user to control consumption and costs through energy audits that are supported by data collected continuously, improving the energy efficiency of the plant, its processes and devices.

In the case of BEMS, the current EMS is based mainly in controlling the variables related to the lighting system and the comfort of occupants in a building. Therefore, it focuses on the management and optimization of the electrical energy used by heating, ventilation and air conditioning systems (HVAC) [12, 13] and energy waste caused by old lighting systems.

In general, a strategic energy management approach that includes an EMS has the power to increase energy savings beyond the savings realized by traditional tactical practices alone. This has been demonstrated by an U.S. DOE paper [14]. It studied energy efficiency projects at more than 900 buildings and found that projects that implemented best practices in measurement and verification realized higher savings (both initially and over time) than comparable projects.

Besides, energy data could be used to build up different kinds of models to get consumption trends. So in some recent works models based on linear regressions of energy consumption versus production scheduling are proposed [14]. These models could be used to forecast energy consumption for a given production volume, which has a number of useful applications, ranging from consumption energy prediction for production scheduling to improving the efficiency by predicting the possible energy saving if some proposed energy efficiency measures are taken [10].

However, the models used are based on simple linear regressions, the mechanisms of control are normally off-line actions, and advanced analytical tools for prediction and control are not use. Improving of these items could enhance the system performance and make more automatic the EMS without affect the production or comfort. Therefore, the concept of ***intelligent EMS (iEMS)*** arises to improve the current EMS [15, 16].

2.2. Intelligent energy management system (iEMS)

In Figure 2.2 an iEMS is depicted [4]. As a typical EMS, it can access to information collected from the meters via field buses. In addition, it has capacity to interact with active systems in the building or factory plant to take automatic actions with the objective of optimizing the demanded power. The iEMS takes advantage of its ability to create advanced models of load profiles for load forecasting of consumptions from collected information (energy database) and information entered by users.

In order to carry out these advanced functions the iEMS is supported by the ***load modeling and forecasting system (LMFS)***. The system proposes to take advantage of available energy database to build up models of load demand at different time resolutions (ranging from quarter hourly to monthly scale) and at different levels (for overall plant consumption, specific department consumptions, particular process consumptions, etc.). Besides, the forecasting horizon can be from next day (short-term) to the next or more months (mid or long-term). This depends on the requirements of the final module that uses the LMFS.

These models are used to forecast consumptions. The optimization modules can use these forecasts to detect demand peaks and take actions over the controllable loads to avoid them [17, 18] and so to reduce the contracted electrical power [19, 20]. Diagnosis modules can make a tracking of normal consumptions (healthy patterns of consumptions) under normal operation conditions and detect anomalous consumptions [10]. Then, they can trigger specific algorithms for energy wasting localization. Daily and monthly forecasts can support estimation cost and making decisions in administrative departments (e.g. scheduling workdays, on-off production lines, etc.). ***Therefore, load modeling and forecasting (LMF) is the base for an iEMS.***

The objectives of an iEMS are to optimize and improve economical, environmental and electrical key performance indicators. The iEMS takes advantage of EMS framework and its entire energy database to implement automatic management and control and to save energy and improve the energy efficiency in "*active"* way.

It must be reminded and highlighted that intelligent control-optimization and diagnostic applications are the base core of the iEMS in order to get its objectives. Moreover, due to they are supported by the models of consumption patterns and their forecast, one of the most important steps to improve the energy consumptions is to get a suitable LMFS.

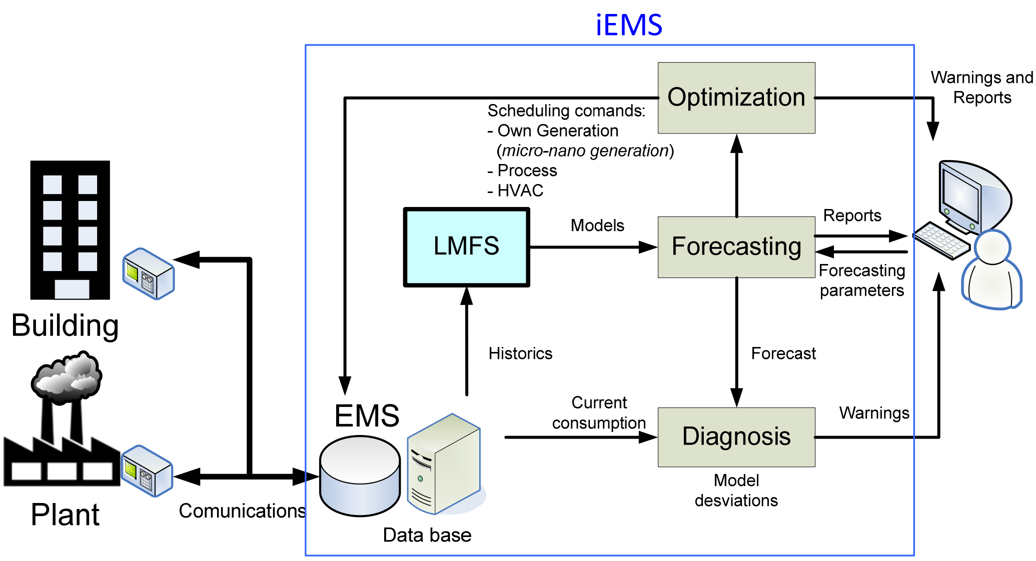


Figure 2.2 iEMS: general structure

Finally, in order to support iEMS advanced control and forecasting functions, the LMFS has to have some special features:

- ✓ Autonomy: ease to build up models in an automatic way;
- ✓ Adaptability: capacity to implement multi-site modeling;
- ✓ Accuracy: reliable forecast that is able to deal with high degree of randomness in loads and therefore to give a reliable base to the control system for implementing the optimization tasks.

2.3. Load Modeling and Forecasting Systems

In other areas of energy management, LMFSs have been widely used and investigated due to their importance in supporting energy management. However, on the user side, LMF is still an area rising in importance. We assume that this is mainly due to the lack of a strong need to perform energy management on the user side. Nowadays, due to the current ***energy and environmental crisis***, the need for energy management and smart consume on the user side is increasing more and more.

Owing to the importance of LMFSs in other areas, wide varieties of them have been proposed in the last two decades [21-27]. These are mainly based on a range of methods from pure statistic methods to complex structures of adaptive networks joined with advanced techniques of signal processing as wavelet transformations and integral filters. These methods are called hybrid methods.

Next, we are going to check the state of the art for LMFS in its main areas. At the end, we highlight some noteworthy approaches found in the recent scientific literature, mostly based on hybrid algorithms.

2.3.1. LMF: main areas in the energy field

LMF has had big importance at different areas of energy management, for example, for ***utilities*** in generation, transmission and distribution of electricity; in energy markets for ***price forecasting***; and in buildings for ***HVAC*** control and optimization. Furthermore, LMF as a ***time series*** modeling and forecasting problem has been investigated widely in the current scientific literature [28].

Utilities

In this area, LMF is important in supporting making decision process. For example, to face the problems of economic scheduling of generating capacity, scheduling of fuel purchases, security assessment, and planning for energy transactions. The energy security and stability rely on the right planning of these items. Therefore, LMF has taken an important role in utilities for long time ago (approx. since 1980s). Short-term load forecasting (STLF) is the most important type of forecasting in this field. Commonly, STLF is an energy forecast of 24 hours ahead, each quarter hour. More information and research was found by the beginning of this thesis work in this area [22, 24, 29-33]. Other important kind of forecasting it is peak power load forecasting of a few days ahead. It is used as an operation index for unit commitment and scheduling [34, 35].

Currently, there is a trend to use CI tools for LMF, mainly, hybrid algorithms. NN, NF and evolutionary algorithms are noteworthy of mention. This is mainly due to the capacities of CI algorithms to model no-lineal behaviors as electrical consumptions are.

Energy price forecasting

Other area where we have found important research on LMF has been in ***energy price forecasting*** [36-41]. For these applications, LMF takes high importance. This is because in the new model of energy market, where consumers can buy energy from different electric power suppliers that are called the pool, both kind of participants of the market need for LMF in order to maximize their profits [23].

In this area, possibly because market influence, statistical tools are the most used for LMF [37]. However, there is a trend now to make hybrid algorithms by joining statistical with CI tools and statistical with signal processing operators as wavelet transformations and integral filters [38, 39, 42].

BEMS

In ***BEMS*** LMF has taken high importance because of the need of control and optimization of ***HVAC systems***. Using the forecast of demand in, for example, a day ahead, BEMS can schedule the demand of HVAC taking advantage of slow thermal behavior of buildings [43-47]. Its importance has been such that there is a society founded in 1894 dedicate to HVAC systems for buildings [48]. This society, among its different lines of action, supports the research on LMF systems for energy consumption in buildings. In [43] it is also presented the potential of energy modeling in buildings, especially in HVAC systems forecasting. It is not mentioned that in the industrial sector the potential could be larger since it exists the possibility of process and machine control.

As in utilities area, in BEMS CI tools have good acceptation for LMF [44, 45] and also hybrid algorithms with CI and statistical algorithms [46]. Statistical LMF systems have been also proposed [49].

On the user side

Even though on the user-side, particularly for ***industrial users***, LMF of energy consumptions has not had the same importance as in the other areas, some works have been found [50, 51]. In these works, LMF is mainly aimed at supporting decision-making, looking for taking advantage of available energy smart-meters database by means of data mining. For example, in [51] is presented a fuzzy based energy forecasting system for clothing factory. This system can forecast the energy change based on some ***energy drivers (variables that affect the energy consumption)***, such as the daily total mass of finished garments produced, the total work hours of operators in the plant in one day and the total running time of equipment in the plant. However, no lower levels of modeling areas were carried out (especial areas or sections, process and machines), but only the overall consumption of the plant. Furthermore, the time resolution is large (daily forecasting).

Therefore, we can see that for industrial users LMF is a subject still rising in importance and CI technologies are the main approach being used to tackle the problem of LMF.

In conclusion, due to flexibility of CI algorithms, they are very interesting approaches for addressing LMF problem. The flexibility enables mixes of CI algorithms with other CI algorithms, statistical or signal processing functions. These mixes can be done at different levels and schemes. For example, we can mix different structures as NN and FL and get NF approaches [22]. Also, we can use evolutionary algorithms to train adaptive network (AN) structures as NN, NF, ANFIS and get evolutive trained AN structures [30, 42]. Furthermore, because of the cycling behavior of demand load profiles, statistical and signal processing functions or operators can be useful to highlight this kind of cyclic behavior and improve the results. Examples of these functions are wavelet transforms [42, 52], integral or Kalman filters [52, 53], correlation functions [33], etc. Therefore, proposals based on hybrid algorithms by joining CI with statistical, signal processing functions or operators have presented the best results in the resent years [26, 32, 52, 54].

2.3.2. Novel proposed approach for LMF

In [52], the authors propose two new LMF schemes for STLF in utilities. One of them is based on NF and wavelet transformation, particularly Daubechies wavelets, and it is called ***wavelet fuzzy neural network (WFNN)***. According to the authors, "*fuzzy rules and learning from neural networks can be inferred from multi-resolution property of the wavelets obtained by wavelet decomposition of the signal*". This action divides the complex input space into subspaces, appropriates to the fuzzification process that is done by the FL structure.

The second proposed of the same authors is based on integral filter, particularly the Choquet integral, combined with a NF structure. It is called ***fuzzy neural network based on Choquet integral (FNCI)***. In this case, the Choquet integral is used to compute the consequent part in order to convert the additive fuzzy system in a non-additive fuzzy system. This change in the NF structure makes more efficient the LMFS because avoids overlapping information coming from the rules of the premise part.

Other interesting propose is a LMFS based on ***wavelets, particle swarm optimization and ANFIS (Wavelet-PSO-ANFIS)*** [42]. In this case, the objective is to forecast the electrical energy price in the market, specifically into the pool. There are different suppliers, so the users can chose the best energy supplier according their interests. In order to do that, the users need to know the price forecasting for the next day ahead. This kind of forecasting has the challenge of the high randomness in the behaviour of the hourly energy price, as happened on the user-side energy consumption LP. The wavelet is used again to decompose the noising price series into a set of better-behaved constitutive series. These decomposed inputs are taking by the ANFIS structure, which is able to get the right relationship between output and decomposed inputs. PSO is used to tuning the membership functions of the ANFIS. The result is a more accurate forecasting of the energy price when compared with other LMF approach.

Apart from adaptive networks with CI tools, there are also others interesting proposal using data mining techniques in order to take advantage of energy databases. For example in [50] is proposed a ***data mining framework*** designed to sort, filter, and get relevant information from energy data. The information obtained is used for a LMF.

However, all these approach at different areas are looking for improving the accuracy of the LMFS. We are looking for improving the accuracy, but also taking into account the other features demanded by an iEMS. Therefore, our proposed approaches were thought to improve not only accuracy, but also autonomy and multi-site ability, getting the dominant patterns in an effective and simple way. In order to do this, ***we will take advantage of evolutionary algorithms to improve autonomy. We will use adaptive networks to obtain multi-site ability or adaptability. We will analyze user-side LP and based on this, we will propose a new scheme to take advantage of dominant patters. Thus, we will improve the accuracy and tackle the random behavior of user-side LP (noisy LP problem)***, ***both subjects supported by statistical functions.***

2.3.3. Basic structures for LMFS

It is important to point out the significance of basic structures as adaptive networks (AN), which plays currently an important role on LMFS. Two of the most important AN are Neural Networks (NN) and Adaptive-network-based fuzzy inference systems (ANFISs).

ANFIS and NN are excellent algorithms for LMF because of its potential to solve problems of time series modeling. Indeed, the availability of historical energy data on the iEMS databases and the fact that ANFIS and NN are data driven approaches capable of performing a non-linear mapping between sets of input and output variables make these modeling tools very attractive [4, 22, 45, 52]. ANFIS and NN do not need of explicit knowledge of the relationship between inputs and output to establish a map between them. Only historical data is needed. Once the AN structure is defined, training algorithms are used to extract the relevant information from the historical data and set up the model. One of the advantages of ANFIS over NN is that even though it does not need prior knowledge to be trained, it can be used if it is available, when the ANFIS's inference database (rules) are defined.

For deeper information on ANFIS and NN, Appendix A shows their foundations. Extra details can be found in [55] and [56] respectively.

2.3.4. Signal processing and statistical functions

On the other hand, a part from CI algorithms other mathematical algorithms have been used for hybrid algorithms, which belong mainly to signal processing and statistical functions. They are mainly aimed for pre-processing. The idea of using these pre-processing tools is to highlight the relationships between output and inputs to the LMFS. Besides, they can be used for outlier detection, input selection, noise reduction, etc. For example, wavelet transform (WT) is now being widely used in LMF for noise reduction and feature extraction. By means of decomposition of the time series being modeled, better-behaved signals can be obtained. These are easier to model and forecast than original target signal [38, 42, 52, 57].

Another interesting transform, perhaps less popular than WT, it is Hilbert-Huang Transform (HHT). It has high potential to analyze non-stationary and non-linear signals and to extract useful information and indicators from datasets. Chapter 6 will exploit these features to detect and select dominant patterns from energy LP.

Appendix B expands the related information with signal processing and statistical functions for hybrid algorithms.

2.4. Conclusions

On the user side, LMF becomes the core of the new iEMS. The main areas of energy management where we can find LMF are utilities; BEMS for HVAC control and optimization; energy market for energy price forecasting; and in a general approach, in energy- time series modeling.

The methods used for LMF range from pure statistical ones to advanced and complex schemes based on CI technologies mixed with signal processing, statistical or other mathematical functions. These last types of approaches, called hybrid algorithms, were reported as the most powerful tool to face LMF problem in the mentioned areas.

Even though there is some bibliography and results in load modeling and forecasting on the user side, there is still a lack of research in this area, especially for industrial users (EEMS). This is the reason for what it is necessary to adapt the LMF algorithms used in other areas as utility' EMS and come up with new ones to face the challenging conditions of the high degree of random in consumptions on the user side and other associated problems.

The approaches for LMF found on the user side were mainly oriented toward daily consumption, and overall consumptions of the buildings or factories. No lower levels of modeling areas were carried out, such as especial areas or sections, process and machines. This means that the current approaches for LMF have not been tested in fast industrial process of hour or quarter hour scheduling, such as it is proposed in this work and it is required in many demand-side management programs. In addition, the configuration of all these models is mainly based in human knowledge, which makes necessary the human intervention in the process of building up the models. These facts explain why it is necessary to implement a self-configured and multi-site modeling system, as it is required for advanced iEMS.

Therefore, for an iEMS as the depicted here, the schemes for LMF have to tackle other requirements apart from accuracy. These are autonomy and adaptability LMF. According to the checked literature, hybrid technologies are one of the best alternatives for providing the LMFS of these skills. Thus, we will take advantage of evolutionary algorithms to improve autonomy, use adaptive networks (NN, NF, ANFIS) to obtain adaptability or multi-site ability, and use signal processing or statistical functions to take advantage of dominant patters to improve the accuracy and tackle the random behavior of user-side load profiles.

3. Load modeling and forecasting on the user-side

In the previous chapters we introduce the problems on LMF on the user side and the trends on LMF at different areas related with energy. In this chapter we are going to go in depth in the analysis of the addressed problems in LMF on the user side. The aim of these analysis and studies is to give the bases for the subsequent chapters, where the proposed approaches are developed.

Therefore, in this chapter we ***analyze the load profiles*** (LPs) on the user side, the ***challenges of modeling*** them and the main variables that can affect the energy consumption. With this in mind, we present the particular methodology***,*** used ***to implement a LMFS***, which will be the base of each approach in the thesis. It describes the steps starting from the processing of raw energy data until the evaluation of the obtained models.

We finalize the chapter with a summary table where we present some examples of scenarios and proposals of the designed LMFSs, which are developed in the thesis work.

CONTENTS:

3.1. User-side load profile analysis

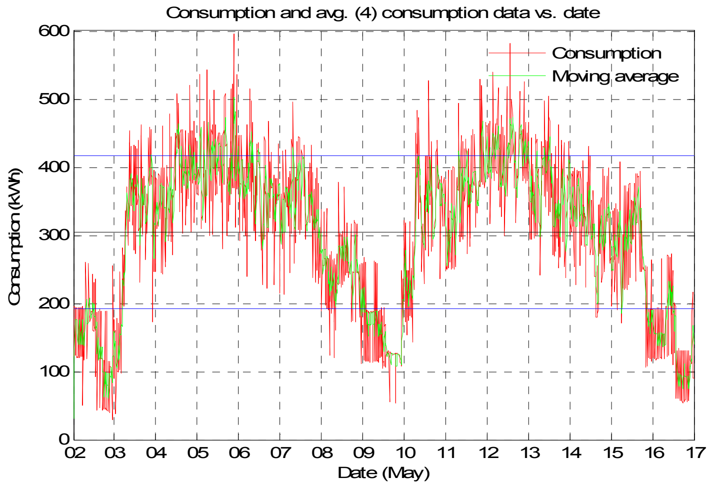
An iEMS [4] is based on an EMS. But the main difference between them is that the iEMS can take autonomous decisions in order to optimize one or several key parameters of energy efficiency. Thus, the iEMS needs accurate forecasting of the LP of the consumptions that we want to control. As presented previously, the main problems of LP forecasting on the user-side are the random user consumption that generates the ***noisy LP problem***, the ***large LP diversity*** and the ***quantity of possible LPs*** to be modeled. Hence, final configuration of the LMFS, i.e. selection of the inputs, pre-processing, filtering methods and the final modeling structure, are very important matters to get an accurate forecasting in a scenario with multiple types and number of loads to be modeled. In order to do that, a comprehensive analysis of the nature of the load and possible correlated variables is needed.

3.1.1. Differences between load forecasting for utilities and on the user-side (noisy LP problem)

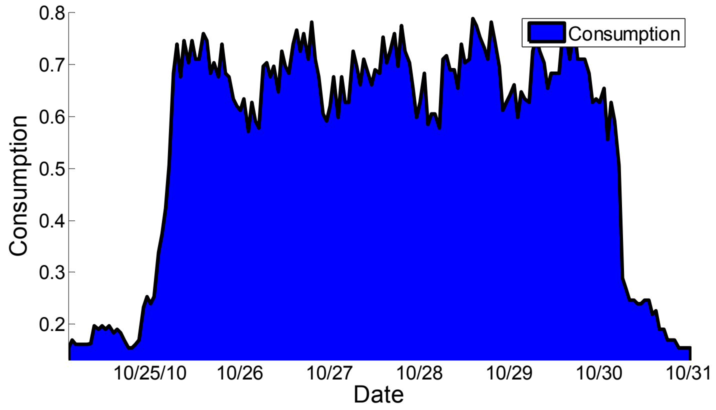
LMFS development on the user side has to deal with LPs with high degree of Randomness, which is here called noisy LP problem. In general, ***the lower the consumption is, the more randomness the LP has***. For that reason, consumptions LPs closer to the final user have higher noisy LP problem. The similar principle applies to the different levels of consumption into the user side: the lower the level of the LP is, the more randomness the LP has.

The ***time resolution*** of the LP also affects the randomness. ***Higher time resolutions will have more randomness***. The forecasting time resolution and ***forecasting horizon*** are defined depending on the final purpose of the forecast.

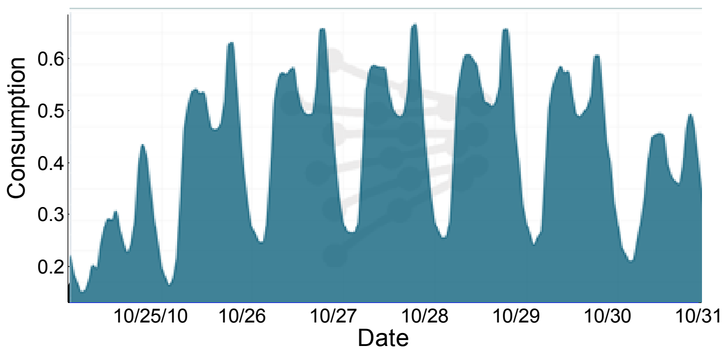
In order to show the effect of level of consumption and the time resolution, in Figure 3.1 we have the LP of three different places at different levels of consumption. Figure 3.1 (a) shows the LP of consumption of a machine process into an automotive factory (consumption into the user side); Figure 3.1 (b) depicts the LP of a full workhouse into the same factory; and finally, in Figure 3.1 (c), it is shown the LP of Spain. It is evident the difference in the randomness of the LPs and how the LP is getting smoother when the level of consumption is increasing. Besides, we can note how the time resolution affects the randomness in the LP. Figure 3.1 (b) and (c) have one hour time resolution and (a) has quarter hour time resolution. The first figure has the LP with highest randomness for lowest level of consumption and for highest time resolution.



(a)



(b)



(c)

Figure 3.1 Different levels of LPs. (a), two weeks LP on the user side of a machine process with a quarter hour time resolution. (b), a week LP on the user side of a workhouse into the same factory as (a). (c), Spanish demand LP for the same week (*source: www.ree.es*). (b) and (c) LPs are normalized between 0 and 1 and have a time resolution of one hour.

The criteria to choose the appropriate time resolution and forecasting horizon for a specific LMFS depend on the final application, where the LMFS is going to be used. Thus, for example if we want to control the power load peaks, it is needed a higher time resolution and short-term forecasting of the LP than if we want to use the LMFS to support the daily production schedule in a factory. In the first case, the time resolution can be quarter hour or even less, with a forecasting horizon of some hours until one day ahead, depending on the possible control strategies that can be taken. In the second case, with a eight (8) or twenty four (24) hours time resolution and forecasting horizon of one day, one week or one month ahead would be enough to carry out a proper daily production schedule.

In the literature the first example is useful to define short-term load forecasting and the second one for mid-term load forecasting. Other possible forecasting horizon is long-term load forecasting and it is used when we want to forecast consumptions some weeks, months of years ahead with similar time resolution to the forecasting horizon (by day, week, month, year).

3.1.2. Diversity of loads and energies on the user side

Other two of the biggest problems found on the user side are the diversity and quantity of loads to be modeled. For example, in an industrial user where energy meters are distributed around the facilities, it is important to analyze the used energy by departments, process or machines. If we add the need for modeling and forecasting not only of basic energies as electricity and gas and we desire to model also other resources as water, hot water, compressed air, etc. the problem of load modeling and forecasting increases and gets each time more complex.

Therefore, it is clear the existence of challenging conditions that have to be tackle when we want to implement a LMFS on the user side. Conditions that in other load forecasting scenarios are not found, as for example in load forecasting in power distribution systems by utilities.

3.1.3. Energy Drivers: consumption correlated variables

In order to be able to start a modeling process, one of the first steps is to choose the possible variables that affect the energy consumption LP. These are called energy drivers, the variables that are correlated in some way with the target LP.

In LMFS applied to energy consumption the most common energy drivers are:

- ✓ Weather variables as temperature, humidity, dew point;
- ✓ Working or process variables as scheduled production, working turn, type of the day (holiday or working day);
- ✓ Time variables such as hour or day of the week;
- ✓ Delayed variables of the target consumption.

Next we are going to analyze each one of these energy drivers.

Weather variables

Figure 3.2 shows the relationship between daily energy consumption for 2007 of overall consumption of an automotive factory *vs.* the daily minimum temperature. We can notice that for working days (upper data over the discontinuous red line in the figure) there is a trend to increase the consumption for minimum temperatures higher than 15ºC. For no working days (lower data under the discontinuous red line) we can identify two trends. One of these trends is around consumptions of $5x10^5$ kWh and other around 3x10^5 kWh ones. The first one correspond to Saturday consumptions and the second one to Sundays consumptions. Considering the workings days, we can notice the consumption dependence with temperature variables. Besides, at weekends we can see its dependence with the type of the days. Depending on the type of the weekend or holiday, i.e. starting holiday (Saturday), middle holiday or finishing holiday (Sunday), the daily consumption changes. The working and no working day variables are analyzed later.

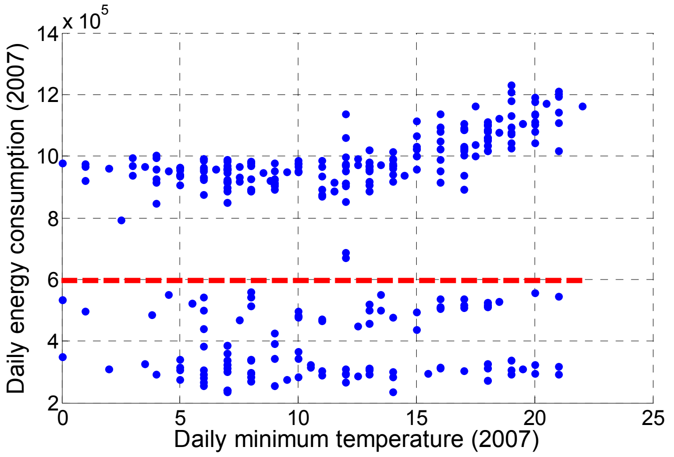


Figure 3.2. 2007 daily consumption vs. daily minimum temperature. The red line divides the tow kind of consumptions: working days (higher consumption) and weekend and holiday consumption.

Working or process variables

Figure 3.3 depicts the daily energy consumption of the same automotive factory *vs.* daily production for the year 2007. There we can observe that almost all production is near 2000 units for day and there is not significant variation among consumption in working days. When the production is zero (weekends and holydays) the consumption is much less than in working days, as expected. Then, the production variable can help to the system to determinate if a day is working or not working one, as it has been done in this work.

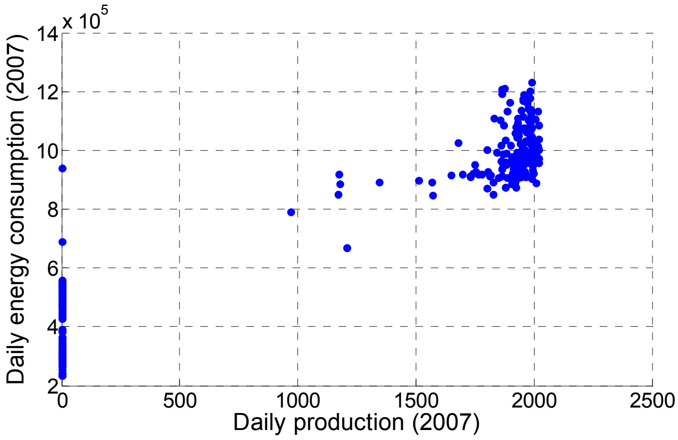


Figure 3.3. 2007 consumption vs. daily production. Production can define if a day is working or not working day.

For example, the variable ***working day*** indicates if a day is a working day or holiday. However, it was found that the use a binary variable for this input it is not the best alternative, mainly because of the sudden change that such a binary variable can put into the LMFS. This sudden change propagates into the LMFS and generates a no desired sudden change over the forecasted output. Furthermore, if a continuous version of this input is used instead of binary value, information of the transition from one state (for example no working day) to another (working day) can be given to the system. This soft transition fits better with fuzzy logic bases and improves the results avoiding sudden changes in the output.

Time variables

Time variables as hour or day of the week/month/year help the system to find cycle or periodical behaviours related with natural trends of the human behaviour, working time tables, seasonal patterns, etc. For example, in the "day of the year" variable is implicit the weather changes, because the weather change with the day of the year.

Other example of time variables impact can be found on the different consumption profiles in starting in working days as Mondays or in a day after holidays, intermediate days of the week, as Wednesday to Thursday, and ending days as Friday or last day before holidays. These profiles clearly distinguish the different days that have been considered in obtaining.

Sometimes these time variables have more useful information than the expected ones, for example weather variables. Besides, they have the great advantage that they are easy to get. Therefore, they are more appropriate to be chosen as possible inputs to the LMFS.

Delayed variables

For the delayed variables, also known as ***lagged variables***, obtained from the historical consumptions, ***one day*** and ***seven days*** delayed signals are mainly the most useful [52]. This is due to there is a daily trend of consumption repetition, easy to find in consumptions, as can be noted in LPs showed before in Figure 3.1. For type of day, for example Sundays, Mondays, etc., we can notice a repetition trend also. In other words, we can say there is a relationship between consumptions at the same time of the current day and one day before; and consumptions at the same time but one week ago. Furthermore, as the weather variables normally trend to change softly between days, delayed consumption variables also carry information of the weather of the week or season. For these reasons, these variables are powerful energy drivers that have to be considered in modeling the consumptions.

Holidays problem: However, in the process of obtaining one day delayed variable we have to be careful with holidays, Sundays and Fridays: i.e. the information of Sunday for Monday forecasting cannot be used, nor Friday information for Saturday forecasting. The same case happens when occurs a holiday. The holiday information only can be used for forecasting of holidays.

In order to solve this problem, the information of last Monday for forecasting the next Monday is used. The same solution can be done for Saturday forecasting and Sunday forecasting. When holidays occur similar algorithm can be applied. So, the delayed vector of one day before, actually is a mix of real one day delayed data and the nearest same type of day and type of working or no working day, in regards of consumption information.

Building up the seven day delayed consumption vector has fewer problems than one day delayed consumption vector. In this case only the information from holidays needs to be corrected.

Taking into account these specific changes in the delayed vector, there will be consistency between target data (the data to be modeled) and input data (energy drivers). Therefore, the AN structure will be able to model working days and nonworking days properly. However, if we want to get a more specialized and more accurate LMF for holidays or working days, the use of independent structures could be a better option [46, 58].

3.2. Proposed LMF methodology

After analyzing the LP being modeled and its energy drivers, we can start with the construction of the LMFS. Therefore, in this section the proposed modeling process that is followed to implement a LMFS is presented.

As it was mentioned in the introduction chapter, one of the most important parts of an iEMS is the LMFS. It could be said that this is the heart of the system. In order to meet the iEMS requirements for a modeling system with high capabilities of accuracy, autonomy and adaptability, the framework of the Figure 3.4 is proposed, which is discussed next. This framework is the base for each LMFS implemented in the thesis. Furthermore, we have used this figure to outline the main addressed subjects of the next chapters.

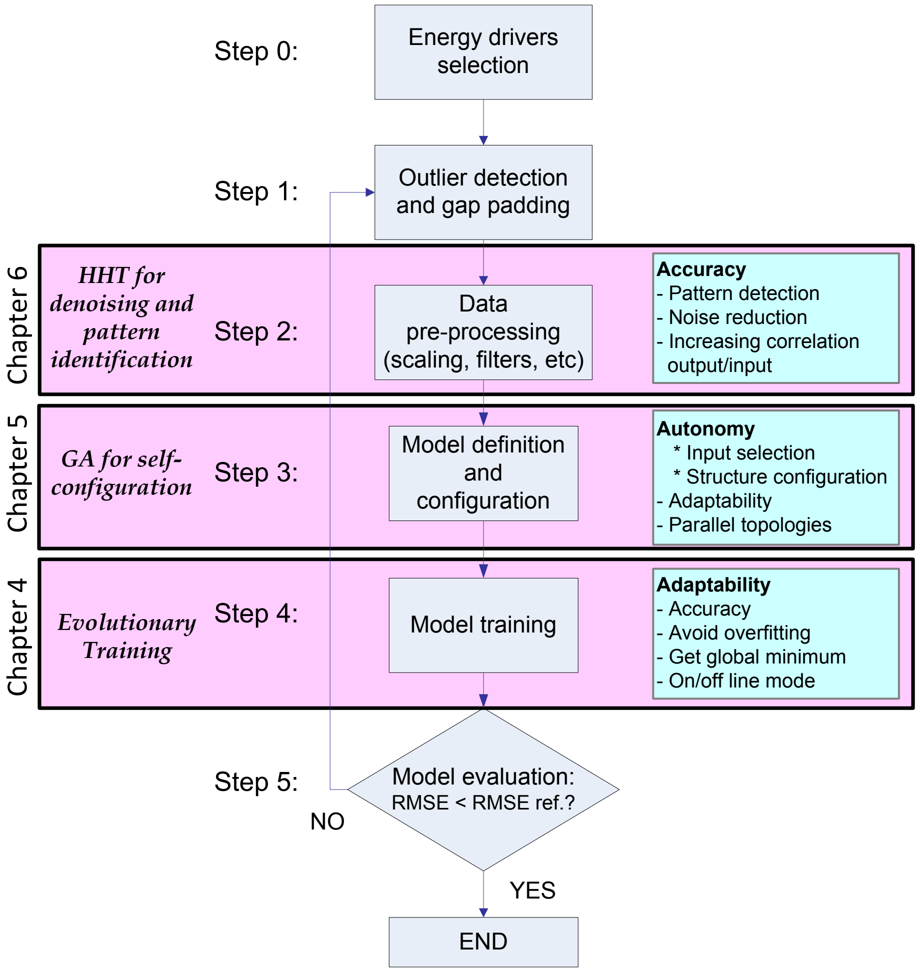


Figure 3.4 Dataflow of the system modeling and subject distribution for development of chapters of the thesis.

Step 0: energy drivers

As explained in the former section, energy drivers are the variable that can significantly affect the energy consumption LP. In this step the main energy drivers that can become candidates for inputs of the LMFS must be chosen. Of course in a real case, they are chosen from the available energy databases. In a later step, the final inputs of the LMFS are selected from these energy driver candidates.

Step 1: outlier detection and gap padding

The outlier detection and gap padding process is important to get a reliable dataset. This is due to outlier data and gaps introduce “noise” in the final model and the quality of the model will depend on the reliability of the data. With the aim of get reliable data, here the available data is analyzed to ***detect***, ***remove outlier data*** and ***pad the found and generated gaps*** in the raw data.

This is done by means of statistical parameter as standard deviation, correlation function, variance and linear regressions for padding, etc. Among these, outlier detection based on statistical parameters is one of the most effective and easy to implement methods.

For example, one of the criteria to indentify and remove outlier data is to remove energy consumption data equal or less to cero. Usually, in operation and also in no-operation days there are energy consumptions, so the zero or negative consumption data are taken as outlier. On the other side, huge spontaneous peaks also are taken as outlier data. In order to remove them, those data that are three times deviations bigger or smaller than the arithmetic mean are removed. These criteria are easily corroborated by visual analysis of the data profiles obtained from database. Expression (3.1) shows how outlier detection is done. There μ is the mean of the analyzed data, σ is the standard deviation of data, x represents analyzed data and $\{\mathrm{outlier\_data}\}$ is the set of outlier data. th is a threshold selected by means of try and error process, usually adjusted to three for the most of the evaluated cases in this work.

$$\text{if } |\mathrm{x} - \mu| \geq th \cdot \sigma \rightarrow \mathrm{x} \in \{outlier\_data\} \tag{3.1}$$

We want to highlight the importance of having reliable and continuous data for time series forecasting. Having reliable data is the base to get an appropriated and accurate LMFS. (3.1)

Step 2: data pre-processing

The second step is the ***data pre-processing***, where the data is scaling, filtered or converted by means of some mathematical transformation if it is needed. All these operations are done with the aim of:

- ✓ Improving or enhancing the existing relationships between the output and input data;
- ✓ Reducing noise (random peaks);
- ✓ Get dominant patterns useful for the LMFS.

As can be noticed, these operation results are oriented to improve accuracy for the LMFS and are helpful to tackle the randomness problem outlined in former sections. Therefore, as showed in Figure 3.4, this step will be object of specific research in this thesis work, oriented to take advantage of periodical patterns of the LP. We will propose to use an analysis based on Hilbert-Huang transform (HHT) to ***extract relevant information about periodical patterns*** as it is explained in Chapter 6, where a novel scheme of LMF based on these dominant patterns is proposed [5].

Step 3: model definition and configuration

In this step called model definition and configuration, different kind of settings must be defined. These depend largely on the type of adaptive network (AN) and the global structure of the LMFS. As showed in Figure 3.5, the first action in this step is the ***structure definition***. In general, we can find two types of structures: single and compound structures. From the first ones result LMFSs that use only one compact structure, for example a single ANFIS or NN. The second ones use more than one compact structure in parallel, serial or mixed topologies. Examples of this type of structures are parallel ANFIS, partial models [5], etc.

Once the structure is defined, the second action into this step is the ***structure configuration***. As we can see in Figure 3.5, this will depend on the type of selected structure. For example, for a single ANFIS structure the final input data, the number of membership functions and the type of membership functions of the inputs and the output have to be chosen. In the case of an NN, the number of layers, the number of neurons and type of internal functions has to be defined. In the case of compound structures, apart from interconnection between singles structures, the type of basic structure used (ANFIS, NN, NF, etc.) has to be chosen as well as the number of basic structures. Thus, when using compound structure, a basic or standard configuration should be used for each single structure in order to simplify the global structure configuration.

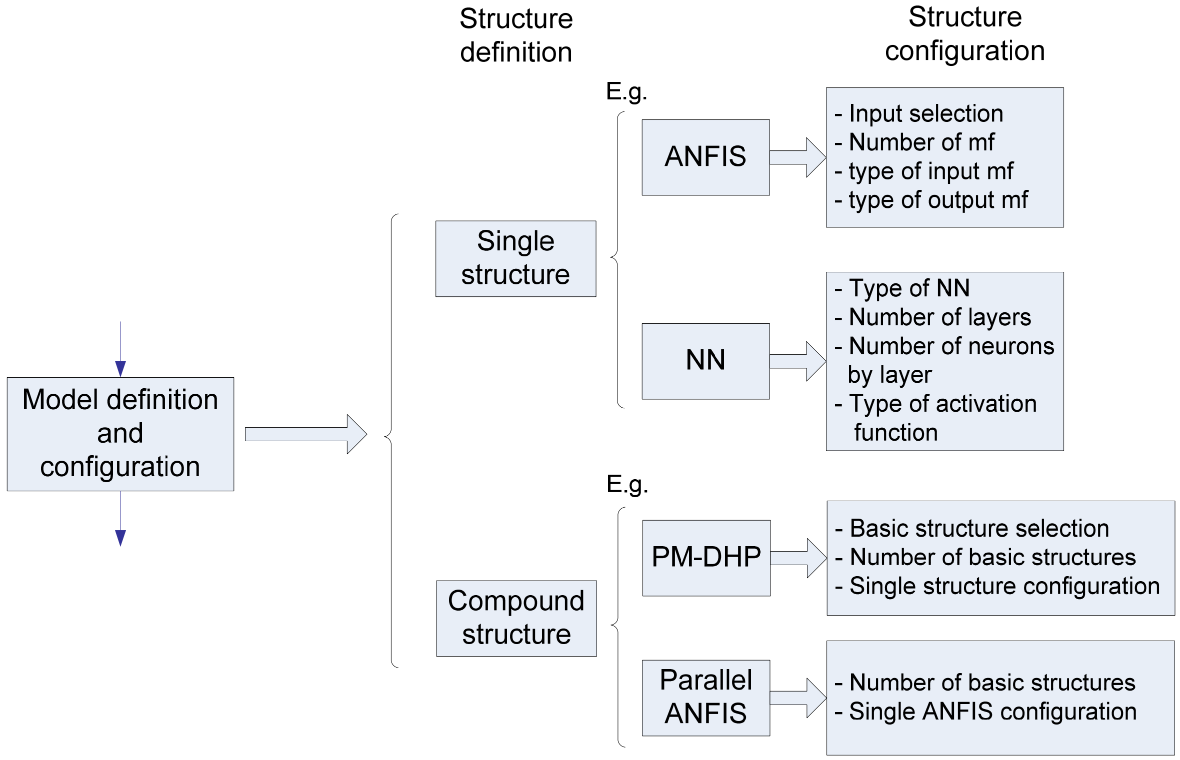


Figure 3.5. Model definition and configuration process.

As can be noticed, the large amount of parameters to be defined to get a suitable model for a particular place could be a problem if the number of places to be modeled is high. This is a drawback when working with flexible and adaptive networks as ANFIS and NN [59]. The problem increases when the used structure is a compound instead of a single one.

However, as highlighted in the state of the art, the fusion of adaptive networks with optimization algorithms as evolutionary ones could solve this problem. Therefore, with the aim of get a suitable LMFS with high degree of autonomy (self-configuration) and multi-site capacity, in this work genetic algorithm are proposed to solve this issue in the LMFS on the user side. As showed in Figure 3.4, in Chapter 5 a novel approach of a self-configured LMF framework based on an ANFIS and genetic algorithm (GA) is proposed and developed.

Step 4: training

In this step the training of the full LMFS is carried out. It means the tuning up of each internal parameter of each AN that belongs to the global structure of the LMFS. Before starting this process, it is important to select properly the training and the checking data, the training algorithm and the stop criterion. Usually, the stop criterion is the number of iterations, and it has been selected in this work.

The final training algorithm depends on the AN structure and the possible training algorithms that can be applied to it. For single structures the process is shorter than for compound ones. For compound structures there are more basic structures to train and the data for training and checking have to be selected according to the target signal of each basic structure. This will depend on the final design of the global structure.

The training process can be done in to modes: on-line or off-line mode. In the first one, before to make a forecasting, the training data is updated with the last available data immediately before to the period being forecasted. The quantity of the used data will depend on designer criteria. Then, the global structure is re-trained with the new training data and the forecasting is generated. Using on-line training, seasonal, weather and sudden changes are taking into account [58]. In off-line mode, the model is not updated before the forecasting. However, it is recommended to make an update when significant changes occur in the modeled place. Off-line mode is appropriated to be used for diagnosing modules. These modules need for reference models, which keep the information of healthy process and machines.

On the other hand, as outlined in Figure 3.4, good design of the training process is important for getting generalization capacity and accuracy in an LMFS. In order to get these important features, overfitting and local minimums must be avoided.

Furthermore, when we want to test new designs of adaptive network, special mathematical algorithms have to be defined to train the new or modified structures. Therefore, in chapter 4 we will test an Evolutionary Training Algorithm (ETA) with the aim of build up a generic training algorithm, able to train modified ANFIS with just simple changes. ETA uses a multi-objective genetic algorithm (MOGA) that enables it to take into account overfitting problem by means of simultaneous calculation of training and checking error. Besides, in order to test the ETA a modified ANFIS with exponential output membership functions is proposed, which could improve the generalization capacity of the normal ANFIS structure.

Step 5: model assessment

Finally, the obtained model is checked with the checking data selected in the previous step. Typically, the manner to quantify the fitness of the model is through the root mean square error (RMSE) calculation. The equation (3.2) shows the mathematical formula. N is the number of samples, $\mathrm{t_i}$ is the target output, in this case the actual consumption, and y_i is the output of the model, the forecasting of the consumption.

$$RMSE = \sqrt{\frac{1}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}}(\mathrm{t_i} - y_\mathrm{i})^2}$$

(3.2)

Since the aim is to get an automatic system modeling with continuous learning capabilities, the RMSE of the new model is compared with the RMSE of the model of reference. The RMSE of previous or old models could be used as reference, so if the new model's RMSE is better that the old one, the new model will replace it. If not, the cycle starts again and the results of each step are checked and corrected.

Other measurement of accuracy is carried out by means of the mean absolute percentage error (MAPE). It is more common for STLF focused in energy, whereas RMSE is a type of measurement of error more general and common in the field of computational intelligence. In this thesis work, RMSE is used in order to choose the final model. However, both indicators are used in the final assessment and comparison among the results of the different kind of models.

$$MAPE = \frac{100}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}}\left|\frac{\mathrm{t_i} - y_\mathrm{i}}{\mathrm{t_i}}\right|$$

(3.3)

3.3. Possible scenarios for LMF

With the aim of closing the presented studies and analysis, Table 3.1 presents examples of final configurations for each of the presented steps of the modeling methodology. We orient these configurations to two types of scenarios: power peak control and daily forecasting for making-decision support applications. As it can be seen, depending on the final application, the LMFS parameters change, as for example, time resolution, energy drivers and model definition and configurations. Furthermore, a summary of the proposed approaches to develop in this thesis work are presented in the same Table.

As it can be noticed, power peak control application needs higher time resolution than making-decision support application. In general, control applications will need more time resolution and their forecasting will be for short-term. These applications are the more demanding in terms of accuracy and reduction of LP randomness.

Going back to Figure 3.4 and reviewing the last column of the Table 3.1, we can review the different approaches proposed in this work.

Table 3.1. Configurations for two scenarios and thesis proposals

Steps and general configurations	Scenery 1: power peak control application	Scenery 2: daily forecasting application	Proposals
0: energy drivers and initial configurations	• 1 and 7 days delayed consumptions. • Temperature. • Type of day. • Quarter hour time resolution	• Production • Temperature • Daily time resolution	NA
1: outlier detection and gap padding	• Zero, negative and spontaneous peak elimination. • Padding by interpolation and copy of 1 or 7 days before data.		NA
2: data pre-processing	• Average filter, average jumping operator, correlation, HHT analysis	• NA	DP detection based on HHT analysis
3: model definition and configuration	• Single ANFIS, Partial models using ANFIS.	• Single ANFIS, Single e-ANFIS	Self-configured ANFIS, e-ANFIS, Partial models
4: training	• Hybrid algorithm, off-line training mode. • Number of iterations as stop criterion.	• Hybrid algorithm, ETA, on-line and off-line training mode. • Number of iterations as stop criterion.	ETA
5: evaluation	RMSE and MAPE		NA

3.4. Conclusions

In this chapter has been outlined the main studies and analysis that are necessary to design and develop a useful iEMS for the user side. Conclusions of this study will be used during the development of this thesis and its derived works.

By means of the study of the load profiles (LPs) on the user side we pointed out the challenges of LMF on the user side:

- High degree of randomness in LPs associated with random behavior of the user (noisy LP problem)

- Wide variety or diversity of loads and
- High number of points to model at the same user.

Besides, we analyzed the dependence of randomness of LP with the level of consumption and time resolution.

Our proposed methodology to implement a LMFS, from processing of raw energy data to evaluation of the obtained models, was presented and the steps that are going to be addressed were highlighted:

- Pre-processing algorithms and functions (Chapter 6).
- Model configuration (Chapter 5).
- Model training (Chapter 4).

Furthermore, we analyzed the possible problems that can be found in each step. For the addressed problems the proposed solution was introduced.

Finally, we close the Chapter with the definition of the more used configurations of the LMFS that are going to be used in the next chapters. Each of these configurations depends on the application that is going to use it. In general, the application will define time resolution, forecasting horizon and energy drivers. For example, control applications as power peak control will need higher time resolution than making-decision support applications.

4. Evolutionary training for new adaptive networks

In chapter 2 we studied the state of the art of LMFS applied to energy fields. One of the most important conclusions of this study was the rising importance that hybrid algorithms of CI are getting in this area, mainly due to the improvements achieved in forecast accuracy. Furthermore, in this thesis has been highlighted the importance of adaptability and autonomy for LMFS on the user side.

Taking into account these precedents, in this chapter we present the first proposal of hybrid algorithms, looking for improving the ***adaptability*** of the LMFS in a challenging area as LMF on the user side is, but keeping in mind the importance of accuracy and autonomy. The first objective of this approach is to improve adaptability of the LMFS creating a generic and flexible training algorithm, easy to adjust to new adaptive networks as for example modified ANFIS or ANN. The second objective is to exploit the multi-objective optimization possibilities that evolutionary algorithms offer in order to improve generalization capabilities of the LMFS.

CONTENTS:

4.1. Introduction and proposal

As outlined in the chapter 3, the training process is important for getting generalization capability (adaptability) and accuracy in an LMFS. In order to get these important features, overfitting and local minimums must be avoided. Furthermore, when we want to test new adaptive network, special mathematical algorithms have to be designed to train the new or modified structures. This fact is a drawback when we want to implement an *autonomous adaptive network designer*.

Therefore, in this chapter we propose an evolutionary training algorithm (ETA) for ANFIS structures. Its main purpose is to avoid overfitting problem and to define a path for a generic training algorithm of adaptive networks, mainly for those based on ANFIS. Multi objective feature of evolutionary algorithms, as genetic algorithms (GA), could help to avoid overfitting problem. Flexibility of evolutionary algorithm can support generic training because the designer does not have to define a deterministic mathematical training algorithm for any structural change done in a typical ANFIS.

The basic idea, as Figure 4.1 depicts, is concentrate the adaptive parameters of the AN and encode them to enable to some GAs to find the near-optimal solution. Using this scheme, any change can be done in the AN parameters and with little adjust in the GA codification, the designer (a person or machine) will be able to test their new structure. Multi-objective GA (MOGA) can considerer both training and checking error during training process and thus, test generalization capacity and avoid overfitting.

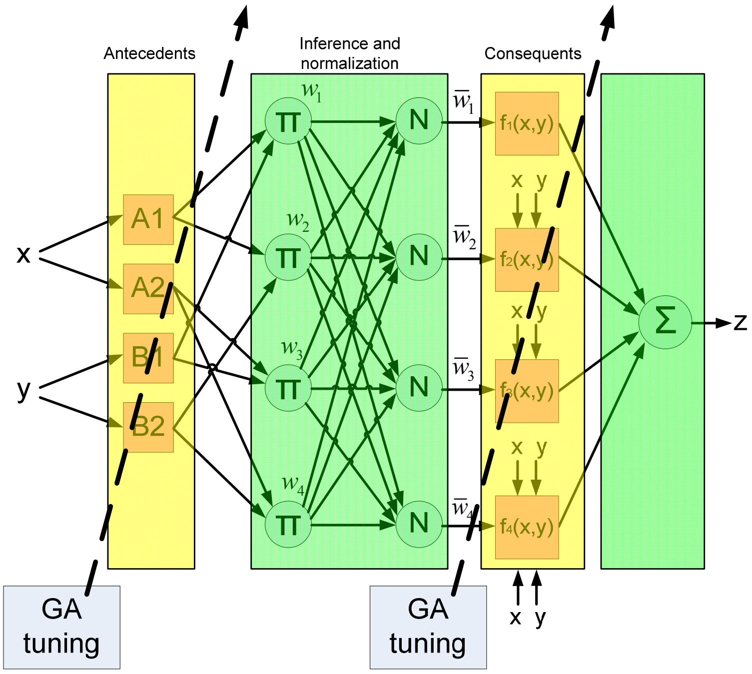


Figure 4.1 Evolutionary training algorithm (ETA) for ANFIS.

In order to test the ETA, we propose a new modified ANFIS, which is based on typical ANFIS with a change in its output membership functions. Instead of use normal polynomial output membership functions, it use exponential membership functions, and it is called e-ANFIS. Typical ANFIS' training algorithms are designed to get the coefficients of polynomials. However, if we want to use other type of functions, equations of training algorithm have to be redefined.

Finally, with the aim of counteracting the added burden by the evolutionary algorithm and taking advantage of the growing availability of parallel computing technologies, a parallel genetic algorithm is used for the implementation of the evolutionary training algorithm.

4.2. ANFIS training by means of GA

As Figure 4.1 shows, the evolutionary algorithm chosen for implementation of ETA was a genetic algorithm (GA). It was chosen because of its capacity of searching in complex spaces, simplicity, convergence capabilities, mathematical abstraction, multi-objective and parallel execution possibilities. ANFIS training based on GA provides flexibility and adaptability to this process.

Implement a GA has two main steps: chromosome definition and fitness-function definition. Joining these steps with the required steps to implement and training an ANFIS, the final procedure to carry out the evolutionary training is:

- ANFIS definition
- GA codification
- Fitness function definition
- Execution of training process
- Final validation

In the following sub-sections we are going to explain each of them.

4.2.1. ANFIS structure definition

This process involves choosing the inputs (variables that affect the consumption) and depending on their relationship with the output, the type and number of membership functions for each input. Additionally, other parameter that can be changed is the type of the output membership function. Frequently, the same type of input membership function is implemented for all the inputs.

Independently of real configuration of final ANFIS, which is to be decided along the thesis, for the subsequent explanation of the proposed algorithms, we are going to use the following example ANFIS configuration:

- Two energy drivers (inputs): daily production and minimum or maximum temperature of the day.

- Two membership functions for each input.
- Gauss membership function type for each input.
- Polynomial (ANFIS) or exponential (e-ANFIS) membership functions for the output.

This final configuration fits the ANFIS structure depicted in Figure 4.1, where *x* could be the forecasted maximum temperature and *y* the scheduled production.

4.2.2. GA codification

As it can be noticed on Figure 4.1 and according to first ANFIS proposal [55], there are two sets of parameters to get from training in the ANFIS architecture. These are the ***antecedents*** and ***consequents***. Typically, they are found by means of least squares (LS) and back-propagation (BP) algorithms respectively, and the training algorithm is called hybrid algorithm. However, in this approach, we are going to use GAs to find both parameter sets.

In order to implement the GA, first we have to build up the chromosome, also called genome or individual of the GA. This process is called codification or encoding. The chromosome or individual of a GA is composed by the encoded version of the variables that we want to find. Each encoded variable is called gene.

In this case, in order to carry out the GA codification for the ANFIS training, the first step is to identify the variables in each parameter set. For antecedents, because the selected membership-function (MF) was a Gaussian function (4.1), the variables or parameters to be found are the centre c_{Ai} and the standard deviation σ_{Ai} that define the Gaussian function for each MF.

$$\mu_{Ai}(x) = e^{-\frac{(x-c_{Ai})^2}{2\cdot\sigma_{Ai}{}^2}} \tag{4.1}$$

Where c_{Ai} is the center of the gauss bell and σ_{Ai} is the standard deviation for the membership function $\mu_{Ai}(x)$ that defines the degree of membership of the input *x* to the fuzzy set *Ai*.

In the consequent part, the parameters to be found are the coefficients of each polynomial, i.e. $p_i, q_i,\ r_i$, where *i=1,2,3,4.* There is a polynomial for each rule of the four rules in the defined ANFIS [55].

Therefore, for an ANFIS structure with two inputs (*x* and *y*), two membership functions for each input and four rules, the resulting chromosomes for antecedents and consequents are presented on Figure 4.2 and Figure 4.3 respectively.

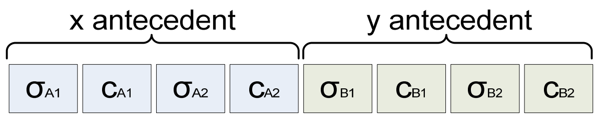


Figure 4.2. String code or chromosome for the antecedent GA.

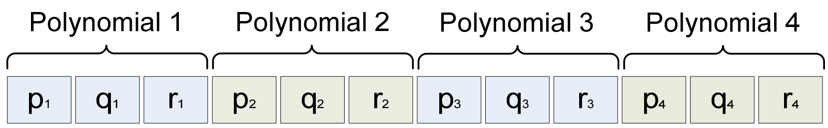


Figure 4.3. String code or chromosome for the consequent GA.

We have to highlight the fact that the content of the chromosome for the GA is as a black box. In other words, the GA does not care about what is the meaning of each gene into the chromosome. Instead, it uses the fitness function to get its feedback. This feature gives to the evolutionary training algorithm its flexibility and potential to train any kind of modified ANFIS that follows the same or similar codification with just few adjusts.

Due to the GA gets its feedback by means of the fitness function and not by the chromosome, the meaning of the chromosome is decoded into the fitness function. But it does not mean that the evolutionary algorithm loses its flexibility. It means that the differences among the possible modified ANFIS are relegated punctually to the fitness function and the decoded functions used in the implementation of the GA. Therefore, the adjustments for use the ETA in new ANFIS are reduced and simplified to be done in the fitness function.

4.2.3. GA configuration

We have said that GA provides flexibility to the training algorithm. However, this flexibility has its opposing party into the GA configuration. Even though a big part of the GA configuration relies on trial and error methodology, the taking advantage on the knowledge about the problem and how the different GA parameters affect its performance can help drastically in the objective of getting the right GA configuration. Next, we are going to outline how the main parameters of the GA for the evolutionary training were configured using the knowledge of the problem and the knowledge on GA.

Data normalization and GA constraints: Once we have defined the chromosome, we have to define the constraints and bounds for the GA. In this point is very important to normalize the data, so the magnitude of the parameters to be found remind near to plus and minus one and so restricts the search space. In order to carry out this normalization, each input and the output data have to be divided for the maximum of its type. The use of these constraints it is a way of introducing human knowledge to the GA in order to improve its convergence speed and reducing the search space.

Initial chromosome: Another important thing is the initialization of the population, so the GA can find the near-optimal solution faster. For the antecedents, it was taken as initial values symmetrical distribution of the membership functions for each input. Thus, the centre (c_{Ai} or c_{Bi}) of each membership function was: *0* (low range) or *1* (high range). For the consequents, the coefficients of a single polynomial model obtained by means of linear regression using of LS algorithm, were given to each polynomial of the output functions and encoded into the initial chromosome.

Population size: the population size is recommended to be at least equal to twice the number of variables. Therefore the individuals in each population span the search space. For the antecedents the population size was set up to three times the number of variables. For the consequent, it was followed the same criterion.

Initial range: this parameter affects the diversity of population. The diversity is one of the most important factors that determine the performance of GA. It cannot be too high or too low; otherwise the genetic algorithm could not perform well. For the calculation of antecedents and consequents the initial range was set up between *[-1 2]*, which was obtained by trial and error

Selection: among the different ways for implementing the selection function (stochastic uniform, remainder, uniform, roulette, tournament, etc.) the roulette selection function was chosen. This function simulates a roulette wheel with the area of each segment proportional to its expectation. Then, the algorithm uses a random number to select one of the sections with a probability equal to its area. At the end of the execution of this function we obtained the parents for the next generation.

Reproduction: here it is defined the number of elite, crossover and mutation children for the next generation. The value of elite children was set to 10% of the size of population, so we guarantee that for iteration the best fitness-function value decreases. However, we have to avoid higher number of elite children, because it increases the probability that the fittest individuals dominate the population, which can make the search to finish in a local minimum.

The rest of children are obtained by crossing and mutation. The proportion ratio between these kinds of children was chosen of 0.5. This value means that there is a high quantity of mutation children, thereby diversity is added to each generation and it is increased the likelihood that the algorithm will generate individuals with better fitness values and will avoid local minimums.

Mutation: this operator was implemented as an adaptative feasible mutation function, which randomly generates directions that are adaptive with respect to the last successful or unsuccessful generation. A step length is chosen along each direction so that linear constraints and bounds are satisfied.

Crossover: the function chosen for this operator was the scattered one, which is explained in Appendix C. In that appendix basic concept and definitions about GA and MOGA are given.

4.2.4. Genetic ANFIS training process

Because we have to get two sets of parameters, antecedents and consequents, the ETA training process is carried out in two main blocks of steps that are executed at each epoch. Figure 4.4 shows a flowchart of ETA training process. The obtained trained ANFIS using ETA training is called here genetic ANFIS (G-ANFIS).

In steps 1 to 3 initial configurations are executed according to instructions explained in the former sections. Once the initial configurations are done, the next step is to train the consequents (Figure 4.4, step 4) using the GA according to the flow chart explained in Appendix C. In this step a thick tuning is done. The antecedents remain unchanged. At the end of the step the current ANFIS consequents are updated (Figure 4.4, step 5). This set of steps are similar to forward pass in the hybrid algorithm [55].

The next step is to train the antecedents using again GA (Figure 4.4, step 6) and using the current ANFIS updated in the step before. At the end of the seventh step, the antecedents are updated (Figure 4.4, step 7) and one epoch is done. This set of steps is similar to backward pass in the hybrid algorithm [55]. The training finishes when the maximum number of epochs is reached (Figure 4.4, step 8).

The ***fitness function*** in both steps is the ***RMSE function***:

$$RMSE = \sqrt{\frac{1}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}}(d_i - o_i)^2}$$

(4.2)

Where d_i is the desired output and o_i is the ANFIS output for the $i\text{-}th$ sample from training data. N is the number of training samples. This function evaluates each individual for every iteration.

During the execution of the full flow chart, the result of each epoch is stored in a vector containing training evolution and results. Therefore, the best individual or chromosome is chosen at the end of the process.

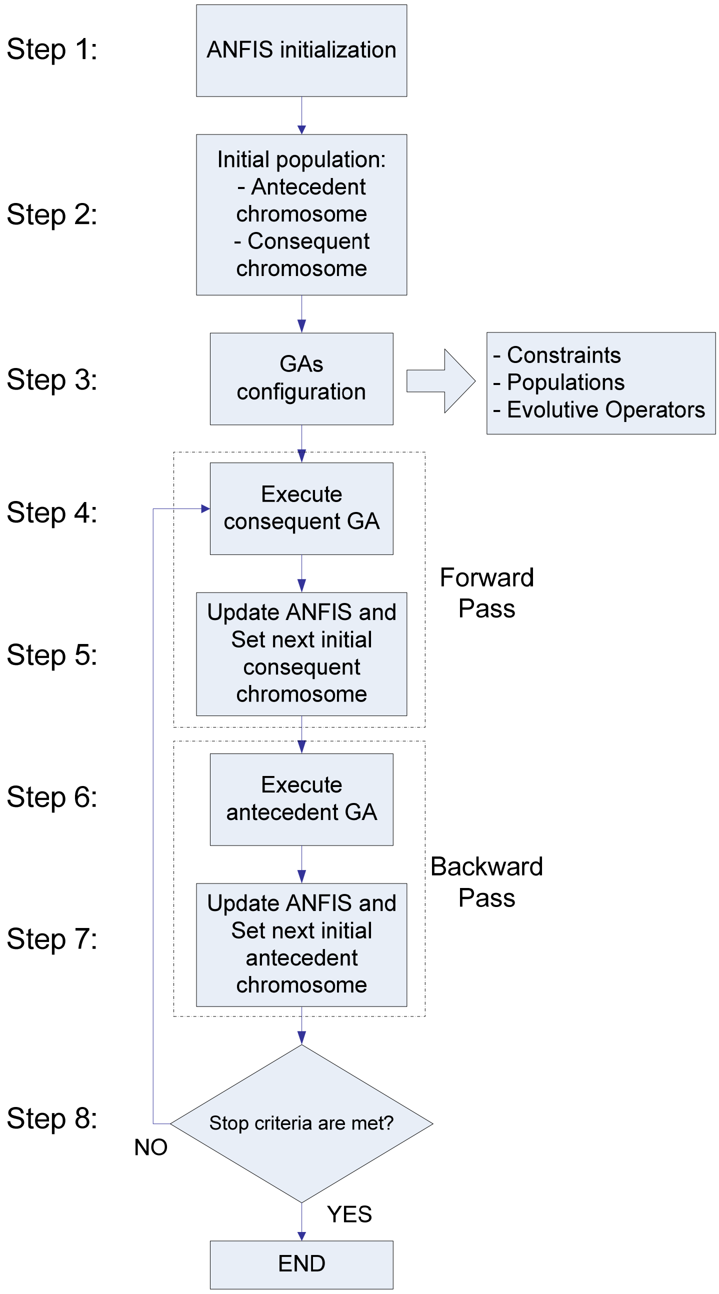


Figure 4.4. Flowchart of the G-ANFIS.

4.3. Parallel computing

Due to the computational burden that ETA demands, we have decide to take advantage of GA parallel structure in order to tackle this drawback. Furthermore, recently improvement and availability of ***multicore computers*** offer the conditions for ***parallel GA*** implementation. However, other strategies could be used to face this subject. For example, increase the computational power, to program in low level the algorithm to avoid no efficient routines, etc.

The fine-gained algorithm was the scheme used to implement the parallel GA in this proposal. In order to do that, Parallel Computing Toolbox by Matlab was used [60]. GA's coarse-grained or fine-grained grained parallel algorithms belong to the distributed applications concept. A distributed application runs independently on several nodes, possibly with different input parameters. There is no communication, shared data, or synchronization points between the nodes.

Run time of fine-grained application is significantly less than the time needed to coarse-grained application. For that reason Matlab implements Multiple-deme GA's based in fine-grained method.

"***ClusterSize***" indicates the number of workers available to the scheduler for running your jobs. This is the number of all threads in processor. A thread (a sequence of steps to be executed) is constructed in a "pipeline" and then "scheduled" for execution by a CPU core. Once a thread is scheduled, the CPU core is executing the pipelined instructions. Intel CPUs support multithreading, but only two threads per core.

A worker object represents the MATLAB worker session that evaluates tasks in a job scheduled by a job manager. Only worker sessions started with the start worker script can be represented by a worker object.

Finally, the algorithm was implemented on a PC with: 8GB DDR3 Ram, processor Intel i7 2600K, Clock Speed 3.4 GHz, 4 cores and two threads per core. The training time of model using the parallel structure was approx. 2 hours; meanwhile the required time using a single core was approx. 8 hours. ***This represents an improvement in speed of four times faster than sequential GA implementation.***

Even though two hours is still long time for a modeling system, this time is still low in comparison to the time resolution (1 day by sample) and horizon of the forecasting (1 week ahead).

4.4. e-ANFIS

The ideas driving the use of an exponential function instead of polynomial one are:

1. Test the ability of the evolutive training implemented in the G-ANFIS to train new ANFIS structures.
2. Take advantage of the versatility of ANFIS structure using new output membership function as is already done with input membership functions.
3. Open the path to implement an autonomous-evolutive AN designer, which is able to look for near-optimal AN structures for LMF.

For an iEMS, the capacity of autonomous modeling is important because the multiple kind of load of different nature that can be found on the user-side. Furthermore, the idea of "intelligent" EMS is associated with the autonomy of the system to infer models to support the making decisions tasks. Therefore, the idea of using modified ANFIS, which can improve accuracy and keep the adaptability capacity inherited from ANFIS, fulfils the requirements demanded by an iEMS.

4.4.1. e-ANFIS implementation

According to [55], the fuzzification layer realizes a fine tuning of the relationship output-inputs and the defuzzification layer a thick tuning. Normally as was proposed in [55], the output membership function is a constant or a polynomial as it is presented in (4.3). However in this work we want to test new alternatives to this output function using an exponential function instead of a polynomial one. In (4.4) is presented the proposed function.

$$f_i(x, y) = p_i x + q_i y + r_i \tag{4.3}$$

$$f_i(x, y) = p_i e^{\propto_i x} + q_i e^{\beta_i y} + r_i \tag{4.4}$$

The exponential output adds to the output layer the capacity of fine tuning and not only of the thick tuning. This is due to the exponential shape as showed in Figure 4.5. This output fine tuning improves the generalization feature of the structure and avoid the overfitting problem as it is showed in the next section.

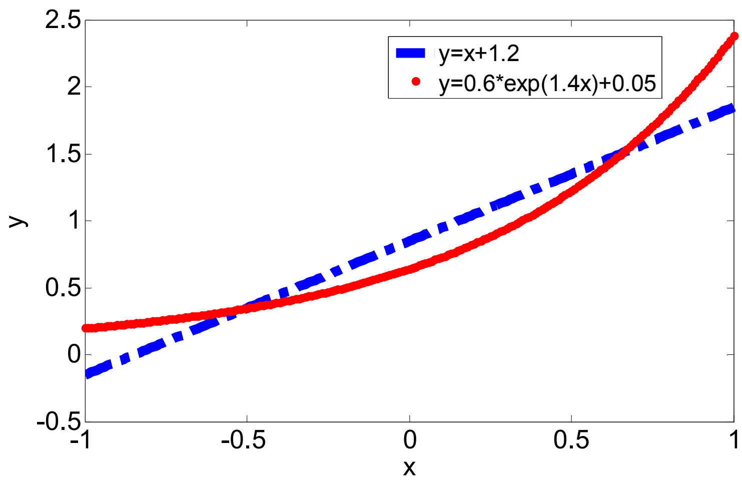


Figure 4.5. Comparison between lineal and exponential output MF.

However, we must point out that now we have to find two parameters more for each input and we have to adapt the training algorithm to this new expression.

Finally, because we use the ETA algorithm to train the ANFIS structure, few changes have to be done to introduce these new parameter in the evolutive training algorithm as it will be showed in the next sub-section. As it was mentioned in “GA codification” section, we have just to change the coding and decoding functions.

e-ANFIS consequent codification

Figure 4.3 shows how the consequent chromosome codification was when training the normal ANFIS. In Figure 4.6 we can see the updated chromosome in order to train the e-ANFIS. We can note that slight changes have to be done and the rest of the algorithm remains unchanged.

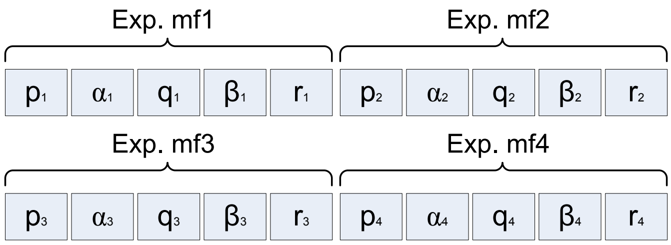


Figure 4.6. String code for the consequent GA.

4.5. Implementation and results

4.5.1. Case of study

In order to implement and test the proposed algorithms, real data from an automotive factory were used. They have been obtained thanks to a technology transfer agreement signed with the automobile company SEAT, in Martorell, Spain. They have provided information of its SCADA to develop a LMFS, with daily resolution. It is aimed to support the decision-making process. For example, it can support decisions about scheduling of daily production, shifts, working lines, maintenance tasks, etc.

The consumption data to be modeled belongs to the daily overall consumption of the factory. Apart of consumption data, daily temperatures (minimum and maximum) and daily production were available. The data correspond to the years 2007 and 2008; the first year was used for training and the second one for validation of the implemented models.

4.5.2. Models

Following two types of training schemes (off-line and on-line training) five models were trained and checked in order to test the proposed forecasting scheme.

- 2xANFIS: ANFIS structure using hybrid algorithm for training and using off/on-line training scheme.
- 1xG-ANFIS: ANFIS structure using the proposed ETA and using only off-line training scheme.
- 2xe-ANFIS: modified exponential ANFIS using the proposed ETA and using off/on-line training scheme.

The first two models were used as referenced models. They are typical ANFIS structures, trained with hybrid algorithm. The first was trained using off-line training and the second one on-line training. The third, the G-ANFIS, was used to test the ETA method and its evolution. The others two models are based on e-ANFIS structures and they use ETA for training. Again, one of them was trained off-line and the other one on-line.

The on-line training was implemented by updating the training data with the last year data immediately before to the week being forecasted. Then the ANFIS and e-ANFIS on-line models were re-trained with the new training data. The idea of using on-line training is to update the models according to weather and seasonal changes [58].

4.5.3. Training evolution of the ETA (off-line training)

Figure 4.7 depicts the off-line training evolution in each epoch for the hybrid training algorithm (denoted as ANFIS) and genetic based training algorithm (showed as G-ANFIS). Both, training and checking evolution are shown. We can see that the training rates in terms of epochs are similar for both algorithms. However, the G-ANFIS reaches before the minimum checking error, about epoch 12, while typical ANFIS training algorithm reaches the minimum about epoch 20. Note also that the G-ANFIS gets a better checking result than ANFIS. This result shows a better performance of G-ANFIS than the hybrid trained ANFIS. It is well known that the checking error is more important than the training one, because it shows the grade of generalization of the model.

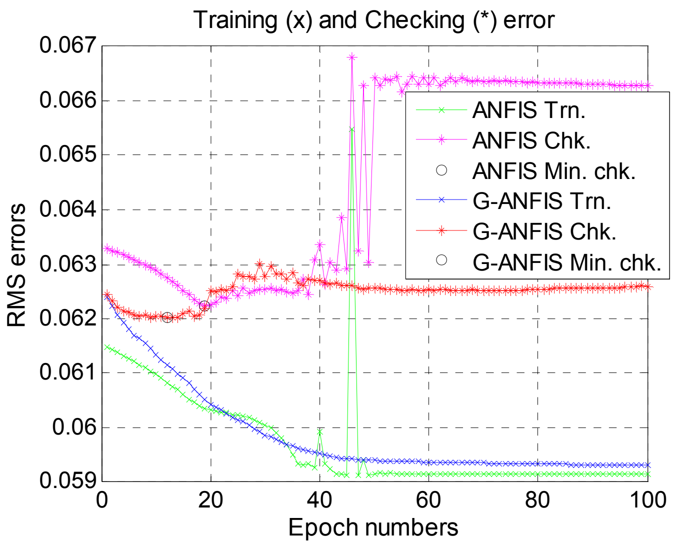


Figure 4.7. Training evolution for ANFIS and G-ANFIS.

4.5.4. ETA validation (off-line training)

Figure 4.8 shows the forecast for entire year 2008 (a) and for months of April to June 2008.(b) Both forecasts show a good prognosis for this year, which belongs to the checking data. This proves that the genetic algorithm is able to find near-optimal solutions for training an ANFIS, even for antecedent and consequent parameters (fine and thick tuning respectively)

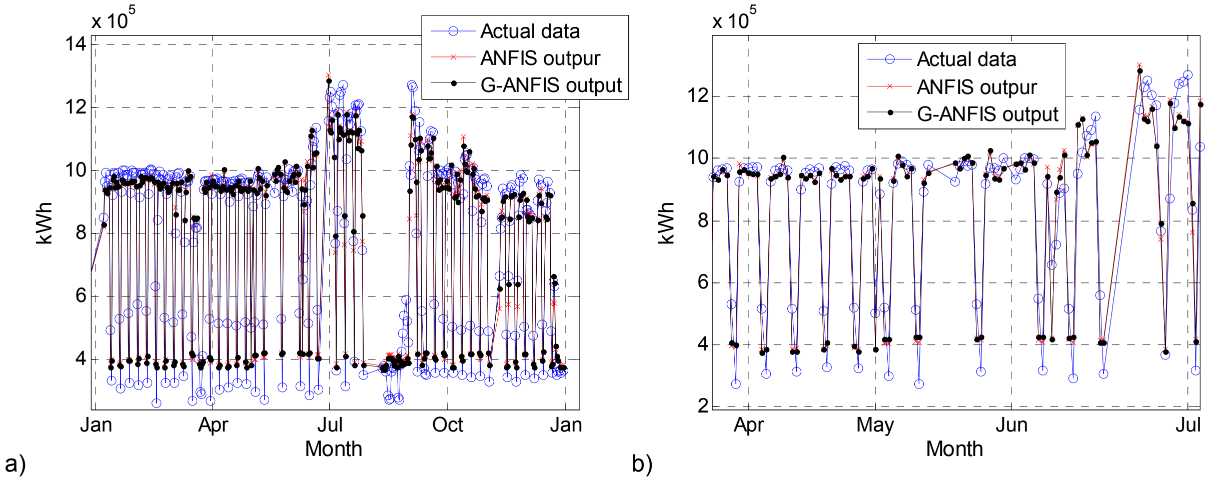


Figure 4.8. 2008 daily consumption forecast (a) and three months forecast (b) by ANFIS and G-ANFIS.

4.5.5. Weekly comparison results for all models (off-line and on-line)

Figure 4.9 represents the results for a week of each month of the eleven months of the 2008 year (because of outlier detection some data was deleted of the full year). From the picture, we can see that all models get quiet similar results except ANFIS on-line, which presents bigger RMSEs for weeks 26, 30 and 38 (summer and autumn). Furthermore, we can notice that among all models, the best and most stable results were getting with e-ANFIS on-line model. Week 22 is an exception. There all models got a big RMSE. Probably this is due to outlier data not detected in the pre-processing step.

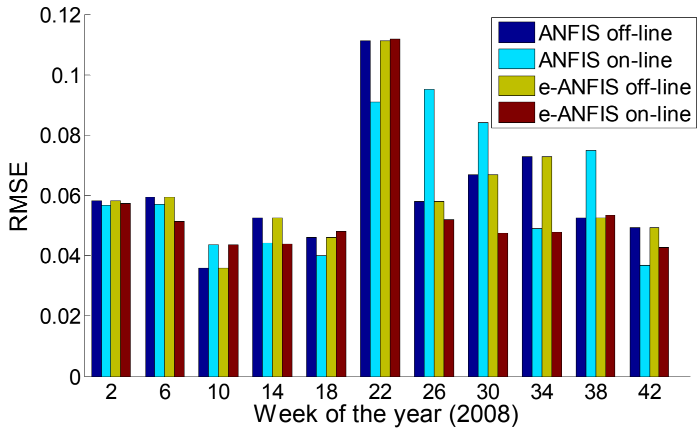


Figure 4.9. RMSE comparison among the four models for one week of each month of 2008 year.

In order to finish the validation and comparison results among the evaluated models, in Figure 4.10, we have the results for a week of each season in the 2008. The seven days of the week were tested, i.e. working days and weekend days. The results for the four models were similar for winter and spring weeks. However, for summer and autumn the results obtained with e-ANFIS on-line forecasting scheme are better than those obtained with the others schemes.

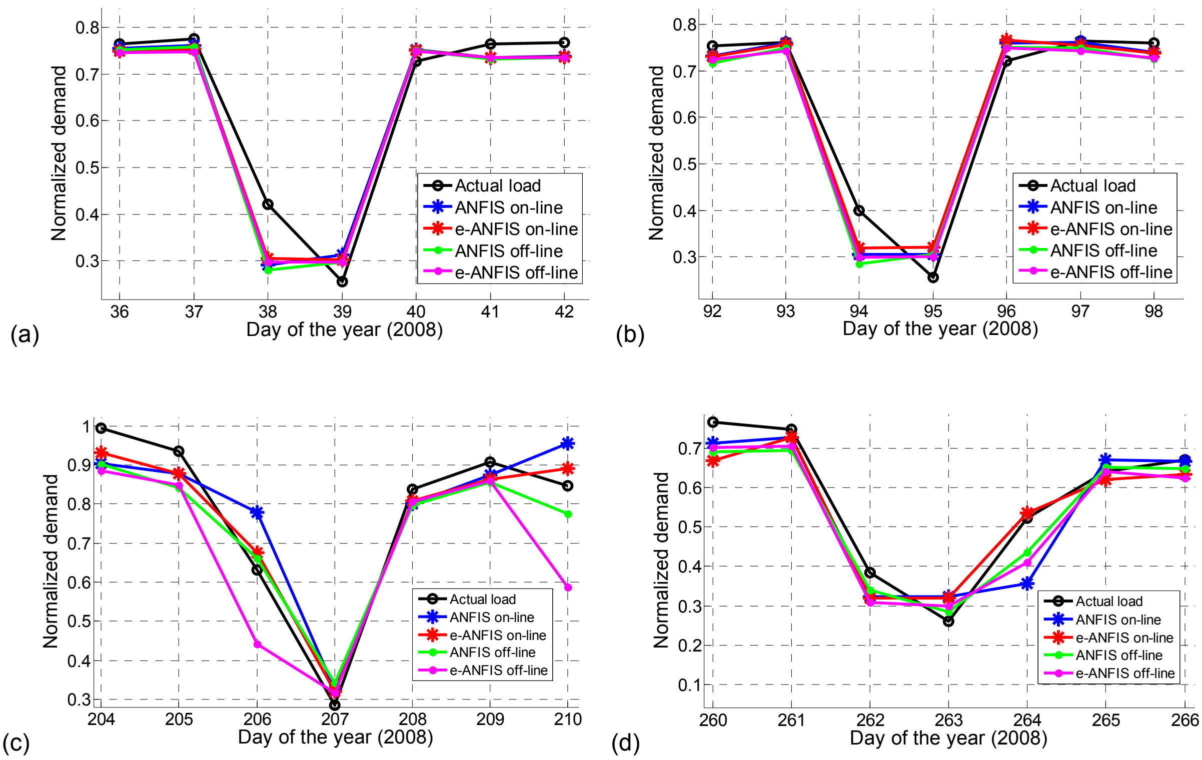


Figure 4.10. Actual and forecasting daily load profile for winter week (a), spring week (b), summer week (c), autumn week (d).

4.6. Conclusions

We have proposed new hybrid algorithms that look for improving the accuracy, autonomy and adaptability of the LMFS. In this case we have addressed the problem of proposing structural modifications in the way how the adaptive networks are trained and making modifications into the used adaptive structure. Particularly, the base algorithms were an ANFIS and a genetic algorithm.

Using them, we have developed an evolutive training algorithm (ETA) based on a GA to train an ANFIS and a modified ANFIS. This algorithm is the key to train and test any kind of new adaptive networks for load forecasting (LF) on the user-side. ETA helps us to avoid the problem of build up a deterministic and/or mathematically complex training algorithm for new adaptive networks.

Furthermore, multi-objective optimization capacity of GA was used to avoid overfitting problem and so, increase the generalization and adaptability of LMFS. MOGA can considerer both training and checking error during training process and thus, test generalization capacity and avoid overfitting.

On the other hand, off/on-line schemes were used to train the proposed and reference models used for comparison. The modified ANFIS was an ANFIS with exponential output functions called e-ANFIS. This new structure presented excellent generalization capability. Therefore, the forecasting scheme based on the ETA and e-ANFIS with on-line training obtained the best results for load forecasting on the user-side taking into account error and generalization for the four evaluated weather seasons. The results were better than the obtained using typical structure as ANFIS with or without on-line training. All the tests were done using real data from the EMS of an automotive factory in Spain.

In order to solve the problem of high computational cost that requires ETA, a parallel genetic algorithm was used. The parallel GA takes advantage of multi-core CPUs and reduces considerably the elapsed time for training.

Taking into account the obtained results with the ETA and e-ANFIS algorithms, we conclude that the proposed forecasting scheme is suitable to be used in an iEMS and can be used in other applications, as for example in power-system forecasting in utilities.

5. Autonomous configuration and multi-site modeling

This chapter discusses two of the three features that we define as important for a LMFS being used in an iEMS. These are *autonomy* and *adaptability*. When energy databases grow and get huge quantities of data from different places, how to get efficiently appropriated consumption models with the less possible human intervention is a challenging task. It is in this context where the mentioned features take high importance.

In order to improve these features in a LMFS and considering chapter 4 experience, we propose a new hybrid algorithm based on an ANFIS and GA. In this case, the GA is used to ***configure*** (no for training) the ANFIS providing it of *autonomy* and improving the adaptability inherent to ANFIS modeling structures.

CONTENTS:

5.1. Introduction and proposal

One of the most serious problems to face a LMFS design in an iEMS is to deal with the big number of data-available to be modeled, which are really high in a factory or smart building. The collected data by smart-meters generate huge energy databases, which correctly processed make possible the creation of energy models of different places in different levels of consumption. In order to take advantage of energy databases (consumptions and energy drivers) the LMFS has to be able to get automatically or with low human attendance accurate energy models. Therefore, this chapter is aimed to study and propose ***self-configured (autonomy)*** and ***multi-site (adaptability) LMFSs***. These features are two of the most important capabilities that LMFSs have to have in order to fulfil the iEMS requirements.

In this chapter, we present a ***self-configured electricity consumption-forecasting framework for multi-site LMFS*** (Figure 5.1). It is based on an ***Adaptative Neural Network Inference System (ANFIS)*** fused with a ***multi-objective genetic algorithm (MOGA)***. ANFIS gives to the LMFS adaptability and flexibility, which are needed for multi-site modeling. MOGA provides a power search heuristic algorithm to find the near optimal configuration of the ANFIS when it is trained in different places and levels of load demand on the user side. This structure gives to the LMFS autonomy to self-configuration for different kind of places and consumptions levels.

This framework is aimed to be implemented in industrial plants, such as automotive factories, with the objective of giving support to an iEMS. The forecasting purpose is to ***support the decision-making*** (i.e. scheduling workdays, on-off production lines, shift power loads to avoid load peaks, etc.) to ***optimize and improve economical, environmental and electrical key performance indicators***. Thus, quarter hour resolution of the load profiles (LPs) is needed. This is one of the highest resolutions found in LMFS on the user side and one of the most challenging because the associated randomness of the load profiles. In order to test the proposed framework, ***the LMFS was implemented in an independent section of an automotive factory***, which was selected for the high randomness of its main loads.

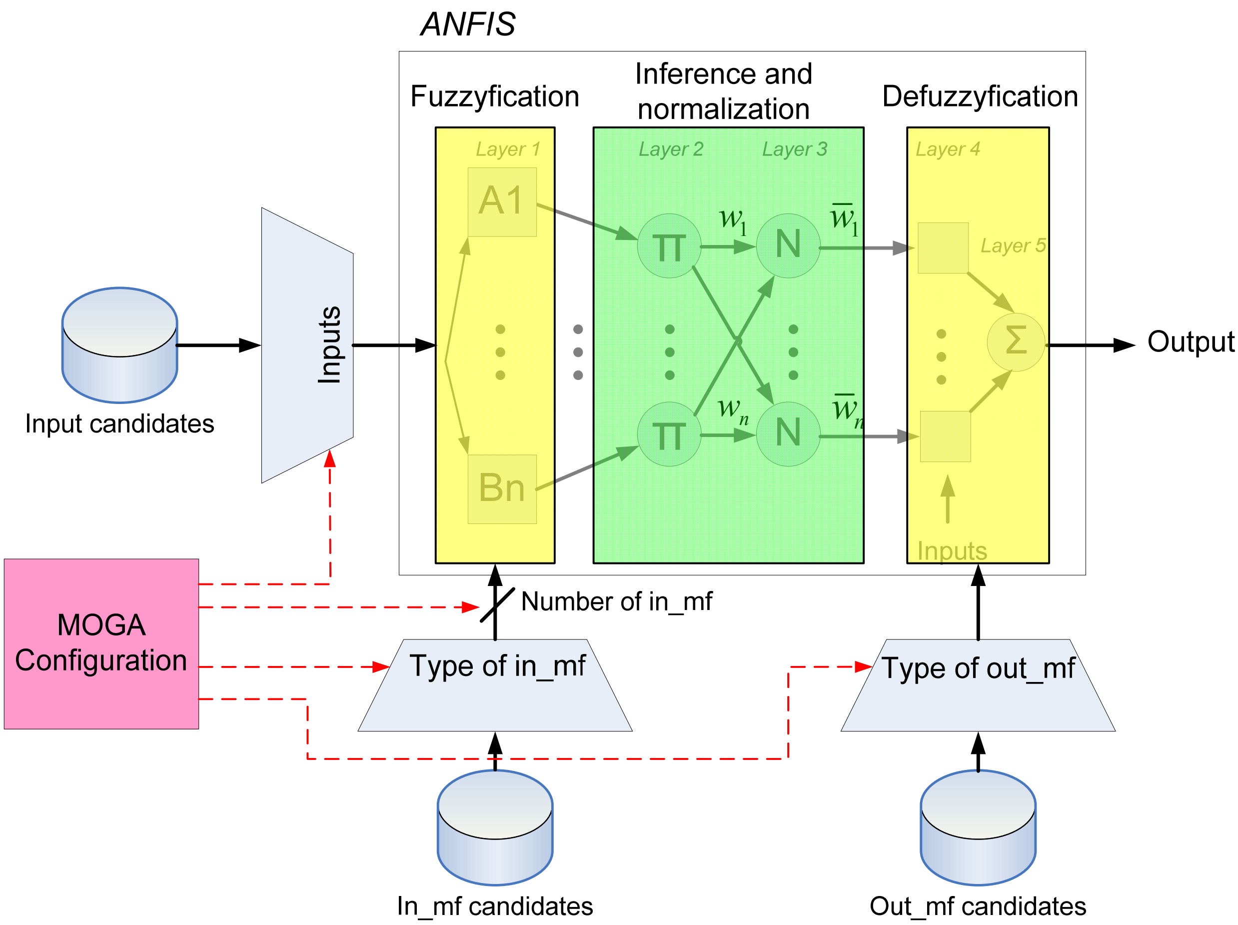


Figure 5.1. Self-configured electricity consumption-forecasting framework for multi-site LMFS.

5.2. Challenges in multi-side LMFS on the user side

The main challenge to face in multi-site modeling on the user side is the ***wide variety of load profiles (LPs) that we can find in the different levels of load demand*** in a factory or building. For example, the degree of randomness increases in lower levels of consumption. Thus, it will be easier to model and forecast the overall consumption of the factory or a workshop of the factory than the consumption of a specific process or machine.

Other factor to tackle derived of the LP variety is the ***selection and availability of the energy drivers***. For example, possibly the weather affects the consumption of the overall factory or building but it does not imply that it significantly affects the consumption or LP of a process or machine. On the other hand, the availability of energy drivers changes for different places. For example, it is possibly easier to get the production of the overall factory but it could be harder to get the specific production of a workshop or a manufacturing line into the same factory. Furthermore, the ***linearity or nonlinearity of the load demand changes from a place to another***.

All these variations make hard to find the optimal or near-optimal model (main structure) and model configuration. For example, if we are using as model an ANFIS, the complexity of this model will depend on the degree of linearity or nonlinearity between of target LP and its energy drivers. Thus, it will be necessary to use more membership functions (MFs), more rules, and more inputs in a target LP with a high degree of nonlinearity, randomness, high time resolution, etc. Besides, if we use a complex model when it is not necessary, then the probability of overfitting will increase. Therefore, the configuration of an ANFIS for different places, with a wide variety of LP, is not a trivial task and demands advanced algorithms to face the problem of self-configuration and multi-site modeling.

5.3. Adaptive network and Evolutionary algorithm

In order to tackle all these different kind of LPs using the same LMFS in an automatic way, we have appealed to universal approximation features of adaptive networks as ANFIS and the power heuristic search algorithm provides by evolutionary algorithms as multi-objective GA.

The problem of self-configuration feature of modeling systems using other CI structures or statistical algorithms has been addressed in published works and scientific literature. For example, in [26, 32], chaotic particle swarm optimization is used for seeking the optimal unknown parameters of support vector machine; in [24] a Bayesian neural network is proposed to implement an adaptative network with features of auto tuning.

ANFIS and MOGA are quite similar algorithms from CI tools and here we want to exploit their adaptive and optimization features. We use ANFIS and MOGA because ANFIS gives to the LMFS adaptability and flexibility, which are needed for multi-site modeling. Meanwhile, MOGA provides a power search heuristic algorithm to find the near optimal configuration of the ANFIS.

Furthermore, by means of its multi-objective property, MOGA can help us to solve overfitting problem of ANFIS. It can take into account both training and checking RMSE (root mean square error) in the search algorithm. This is other advantage for using MOGA in a self-configured and multi-site LMFS.

In Appendix A, the background about ANFIS and MOGA are outlined. For that reason, we will refer to the theory presented in that Appendix when necessary along the chapter.

5.4. Design of the LMFS

As it was mentioned in the introduction chapter, one of the most important parts of the iEMS is the LMFS. It could be said that this is at the heart of the system. Getting a LMFS with high capabilities of autonomy and adaptability means:

- ✓ Automatic selection of the best input variables,
- ✓ Automatic near-optimal tuning of parameters (number and type of MFs, rules, output functions, etc.),
- ✓ Derivation of an uncertainty interval on the model output,
- ✓ The possibility to perform a comparison of different models and, as a result, do the selection of the optimal model.

In order to achieve these targets, we based our current proposal of autonomous and adaptive LMFS on the framework called *"Load modeling and forecasting methodology"*, which was presented in chapter 3. Figure 5.2 represents the cited methodology and there it is emphasized the step where the current approach is focused in.

Because each one of those steps were reviewed and analysed former, in this chapter we are only going to check the main idea of each step and define the specific algorithm used for the real case presented here.

Furthermore, it is important to say that this chapter is mainly focused on the step three (3) "model definition and configuration", where the ANFIS configuration is made. This is the most complex step when we want to improve the multi-site and self-configuration features of the LMFS.

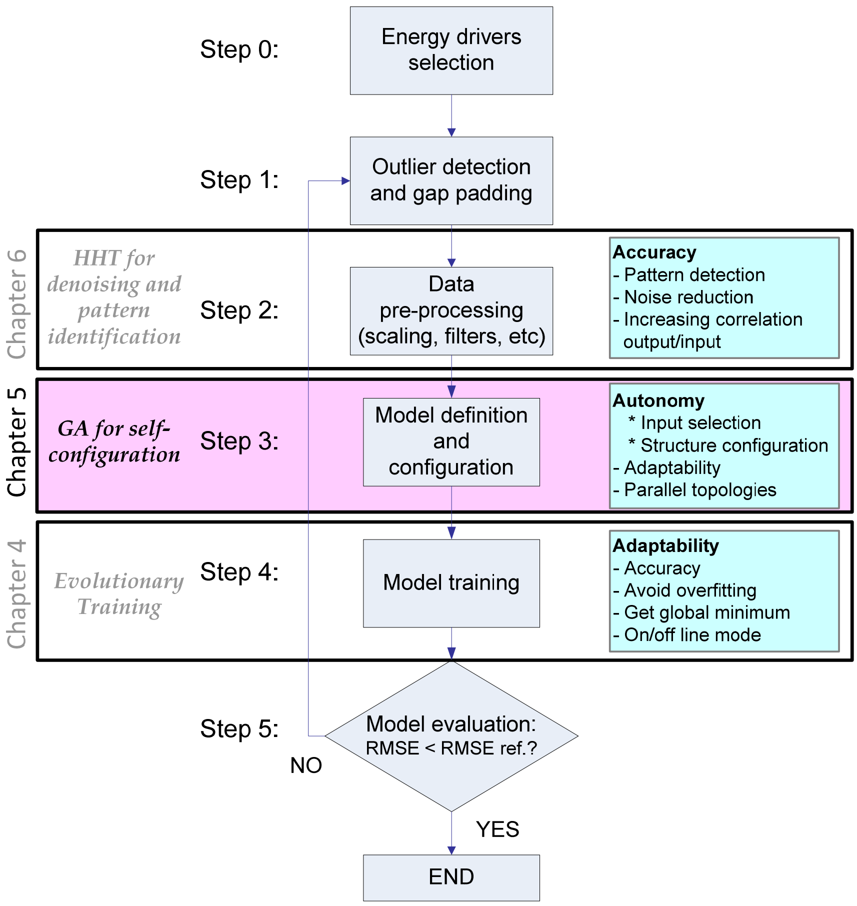


Figure 5.2. Dataflow of the system modeling methodology.

5.4.1. Outlier detection and gap padding

Collected from the sensors system we have the raw data of production, temperature, time data, work shift data, and energy data. The first step is to apply a statistical analysis to identify and remove the outlier data. This process is important to get successful in the automatic modeling because outlier data introduce “noise” in the final model.

For the case study of this chapter, the criteria to indentify and remove outlier data are to remove energy-consumption data equal or less to zero. Usually both in operation and in no-operation days there are energy consumptions, so the zero or negative consumption data is taken as outlier.

For energy consumption and typical energy drivers as temperature and production, big spontaneous peaks are taken as outlier data. In order to remove them, those data that are more than three standard deviations bigger or smaller than the arithmetic mean are removed. These criteria are even easily noted by visual analysis of the data profiles obtained from database. Expression (5.1) shows how outlier detection is done. There μ is the mean of the analyzed data, σ is the standard deviation of data, x represents analyzed data and $\{\mathrm{outlier\_data}\}$ is the set of outlier data.

$$\text{if } |\mathrm{x}-\mu| \geq 3 \cdot \sigma \rightarrow \mathrm{x} \in \{outlier\_data\} \tag{5.1}$$

This expression was selected as the best way to detected outlier data because its simplicity and effectiveness. These features make it suitable for multi-site modeling.

On another hand, when we remove the outlier data we have to replace the deleted data in order to keep a continuous stream of data. Therefore, to carry out this padding task we propose the next algorithm:

1. Detect gaps by means of date variable.
2. If the gap is shorter than four samples, use simple interpolation. If not, look for nearest day of the same type (working day or holiday) with data of the same hour.
3. Replace the gaps with the obtained information.

This simple algorithm worked appropriately in more than four different places, as it will be shown in the implementation section later.

5.4.2. Data pre-processing

In multi-site modeling on the user side, this step takes high importance due to there are a wide range of levels of consumption with their associated randomness. Due to this randomness, mainly found in lower levels, a moving window filter is applied to the energy consumption data. This filter reduces the high variation of the load profile.

In the equation (5.2), the mathematical expression of the filter can be found. There $\mathrm{y(n)}$ represents the filtered energy data, $\mathrm{x(n), x(n-1), x(n-2)}$ and $\mathrm{x(n-3)}$ represent the current and delayed energy data, which occurred one, two and three quarters of an hour before the current time respectively.

$$\mathrm{y(n)} = \frac{1}{4}\mathrm{x(n)} + \frac{1}{4}\mathrm{x(n-1)} + \frac{1}{4}\mathrm{x(n-2)} + \frac{1}{4}\mathrm{x(n-3)} \tag{5.2}$$

Other data transformations can be done in this step. These transformations can help to get new variables with higher correlation with the output than the original variable. For example, getting the day-of-the week (called *week_day* in this work) is proposed and done in this step from the date recorded in the database. Besides, data scaling could be considered. In the real implementation presented later in this chapter, different kinds of scaling were tested without considerable improvements, so the raw data were used. Using the raw data avoids scaling calculations and makes the analysis more direct.

On the other hand, with the objective of getting better-behaved signals (less noising) and extra additional features, time-frequency transforms could be applied in this step. For example, wavelets or Hilbert-Huang transforms are good candidates for this purpose. Chapter 6 will present our proposal addressing this hypothesis.

5.4.3. ANFIS parameter configuration

As can be noticed in Figure 5.3, single structure and ANFIS have been chosen as the final LMFS structure. Therefore, the problem of model definition and configuration is reduced to an ANFIS parameter configuration. The reasons for choosing ANFIS have been depicted in previous sections.

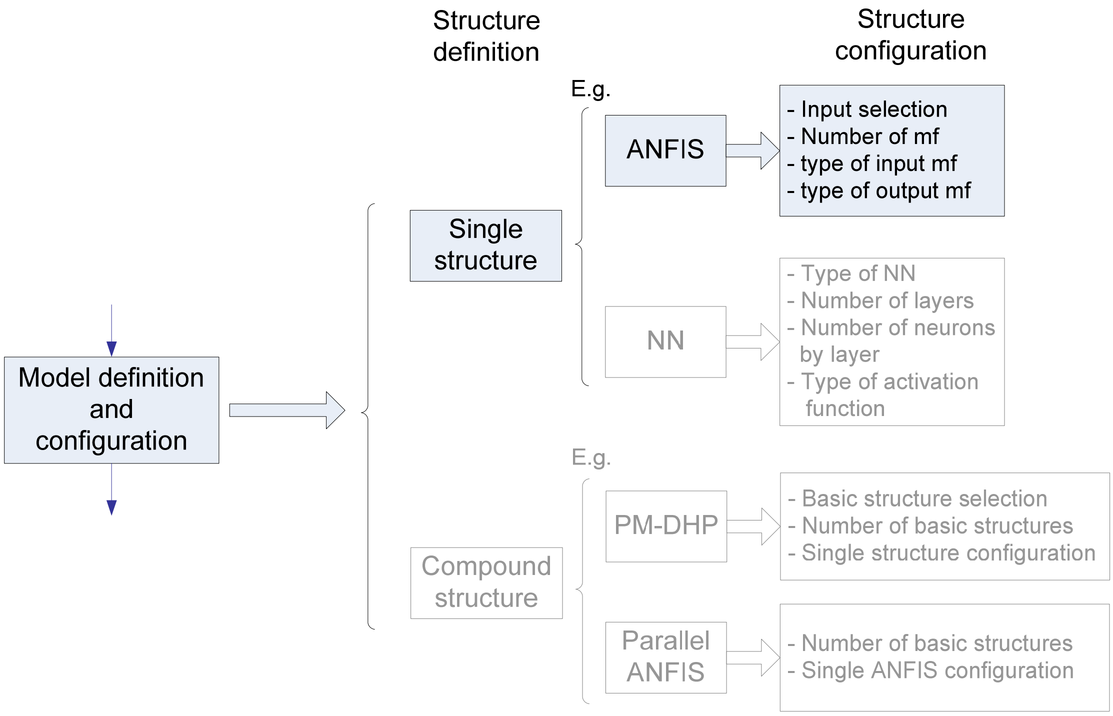


Figure 5.3. Model definition and configuration process focused on single ANFIS topology.

In ANFIS parameter configuration, external and internal ANFIS parameters have to be configured. These are different from those got in the training process (consequents and antecedents). External parameters are the inputs; internal are the number of membership functions for each input, the type of the membership function and the type of function of the output (the last one could be a linear equation or a constant function). A summary of all ANFIS parameters and methods for both training and configuration are showed in Table 5.1.

Table 5.1. ANFIS training, parameter configuration and methods.

ANFIS Parameter	Method	Forward Pass	Backward Pass
Input selection	Human knowledge / MOGA[a]	--	--
Number of MF[b] by input	Human knowledge / MOGA[a]	--	--
Type of MF[b]	Human knowledge / MOGA[a]	--	--
MF[b] parameters (antecedents)	Human knowledge / BP algorithm	Fixed	Gradient Descent
Rules	Human knowledge / BP-LS algorithm	LS Estimate	Gradient Descent
Coefficients (consequents)	LS algorithm	LS Estimate	Fixed

[a] MOGA is the acronym of Multi Objective Genetic Algorithm.

[b] MF is the acronym of membership function.

In the zero step on Figure 5.2, some pre-input selection methodologies can be applied (PCA and exhaustive ANFIS searching) to get pre-selected inputs (energy drivers). In the current step, these pre-selected inputs could be used as starting point for the final input selection. This will depend on the final selection algorithm.

Once the inputs are selected, the next step is to choose the number of the membership functions (MF) for each of them. In other words, choose the number of fuzzy sets for each input. The human knowledge could be used at this point, considering as basic criterion that: the simpler the model is, the more capacity of inference the system modeling will have. However, because the aim of our proposal is get a self-configured LMFS, human knowledge will not be used for carry out this step.

For the case of the membership function selection, some of the more used are triangular, trapezoidal, Gaussian, Gaussian bell, sigmoid and 'Z' functions, among others. Figure 5.4 shows the graphic representation of them.

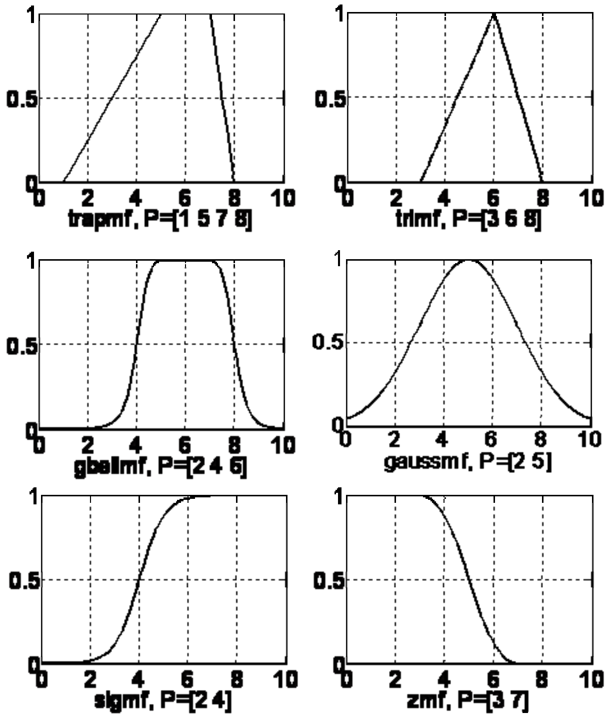


Figure 5.4. Typical membership functions for fuzzy sets.

As it can be noticed, the number of external parameters and their possible combinations are considerably high. These combinations define a complex search space, which is composed of different kind of variables (integers and binary). Getting the appropriated configuration for multiple and different behaviour places, increases the complexity of the search problem.

Therefore, it is necessary to come up with some kind of algorithm or methodology to get the optimal or near-optimal ANFIS configuration. This is a drawback to deal with when using these kinds of adaptive networks.

In addition, overfitting problem in ANFIS also has to be tackled in some way. Overfitting can be detected because when we are looking for a good ANFIS configuration, the training error is reduced but the checking error is increased.

Then, the use of a multi objective GA (MOGA) [61] can help to solve both problems. MOGA can take into account both training and checking RMSE in the search algorithm. Hence, it was decided to use a MOGA in order to find a near-optimal configuration, avoiding overfitting problem and solving the self-configuration problem. Figure 5.1 presents the proposed framework.

As for a GA, in order to implement a MOGA algorithm, two main parts have to be defined: the chromosome or individual codification and the fitness functions that have to be minimized. Next, the explanations of how these steps are done for the self-configured and multi-site LMFS are presented.

Chromosome codification

In summary, the ANFIS configuration algorithm has to define:

- Input variables (week day, time in minutes, maximum temperature and production by work shift)
- The number of membership functions or fuzzy sets for each input (minimum 2 and maximum 5)
- The type of membership function (triangular, trapezoidal, Gaussian, Gaussian bell, sigmoid and 'Z' functions)
- The type of the output function (linear or constant)

In this way, there are four external parameters to be configured before the final training is carried out. The final codification of the used chromosome for the MOGA algorithm is shown in Figure 5.5.

X1	X2	X3	X4
Input_code	Number_mf	Type_mf_in	Type_mf_o

Figure 5.5. Codification of the MOGA chromosome used for the ANFIS configuration.

Fitness function definition

Since we want to avoid the overfitting and this can be done getting acceptable RMSE for training and checking data, the two chosen fitness functions for the MOGA algorithm are those that calculate the RMSE. In (5.3) and (5.4) are presented the final fitness functions $\mathrm{f_1(x)}$ and $\mathrm{f_2(x)}$. There, N is the number of samples for training, $\mathrm{d_i}$ is the desired output and $\mathrm{trn\_output_i}$ is the output for training data of the evaluated model. M is the number of samples for checking and $\mathrm{chk\_output_i}$ is the output for checking data of the evaluated model. The parameters of the evaluated model are defined by the chromosome x = [x1, x2, x3, x4] (see Figure 5.5).

$$f_1(x) = RMSE_{trn} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(d_i - trn\_output_i)^2} \tag{5.3}$$

$$f_2(x) = RMSE_{chk} = \sqrt{\frac{1}{N}\sum_{i=1}^{M}(d_i - chk\_output_i)^2} \tag{5.4}$$

5.4.4. ANFIS training and model validation

As presented in chapter 4, ANFIS training can be done by different algorithms. Among them, the most popular algorithm is the hybrid algorithm, also presented in chapter 4. For the implementation of the framework here proposed we have used this algorithm, mainly because its speed, convergence and effectiveness in terms of accurate results.

On the other hand, the validation of the model is carried out by means of the calculation of RMSE for both training and checking data, as it was done for the evaluation of the fitness functions. Old models for the same site are used as reference model to check the improving of the new models. If the new calculated model is better than the old one, the latter is replaced.

5.5. Implementation and results

In this section a real implementation example of the proposed framework and the obtained results are presented. First, we start with a description of the studied real case; second, we present the global diagram of the real implementation; next, we show the MOGA results for self-configuration in different places; and finally, we present and analyze the obtained results.

5.5.1. Case of study

Places

In order to test the self-configured and multi-site LMFS, the proposed consumption-forecasting framework was implemented in a car factory. The places modeled were ***two main workshops*** and ***three press machines***.

One of the workshops was a ***car body workshop (now identified by OS1)***, where the parts of the car body are made from coils of steel, by cutting and pressing of the sheet steel. The other one was the ***assembling workshop (OS2)***, where the car body is assembled. This has less random behaviour and bigger power demand if it is compared with the OS1. The overall consumption of both workshops were analysed and modeled.

The other places, the ***three press machines***, belong to the OS1. These are indentified by ***PM1, PM2*** and ***PM3***. This workshop and its press machines were selected for the high randomness of their LPs. The sum of the three press machines is equivalent approximately to 60% of the overall consumption of the workshop.

Energy drivers

The considered energy drivers (possible inputs) were the ***production (p)*** of the selected section, the ***external temperature (T)***, the ***type of day (w)*** (working day or holiday), the ***day of the week (d)*** (Monday to Sunday), the ***time (h)*** in minutes and the ***work shift (ws)***. A forecast for the next 24 hours is obtained, with a time resolution of 15 minutes.

It must be considered that for a full day (24 hours) and forecasting with hour or quarter hour time resolution, the immediately delayed load sample cannot be used. If immediately delayed load sample is used, only the next sample could be forecasted using real data. Therefore, because we were looking for a LMFS of the next 24x4 samples, finally the delayed data were not taken into account as energy driver.

Data

The available energy database in the experimental plant starts from May 2 to 17 of 2010, i.e., two weeks of data time stamped of quarter hour for training and checking. The data from May 2 to 9 (first week) were used as training data and the rest data (second week) as checking data to validate the algorithms. Figure 5.6shows the quarter hour load profile of the workshop OS1. There, we can notice the randomness of the loads. The figure shows the real quarter hour energy consumption data (red) and the filtered data (green). The main challenges for modeling this data are the randomness of the loads and the time resolution that was as short as a quarter hour.

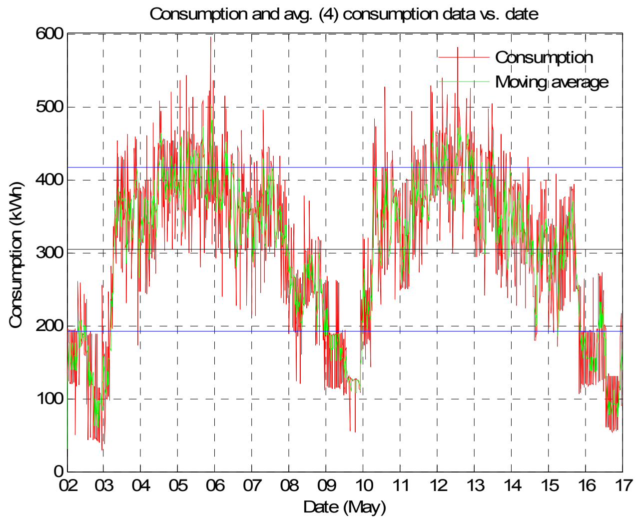


Figure 5.6. Quarter hour load profile of selected workshop (OS1).

5.5.2. Real implementation

The system modeling presented in Figure 5.2 is a subsystem of the final system modeling used in the real application. Figure 5.7 shows the diagram block architecture of how this general system was implemented and where the subsystem modeling was located. As a result, the system modeling is able to build up automatically the consumption models of different sites, getting the data from IEMS database.

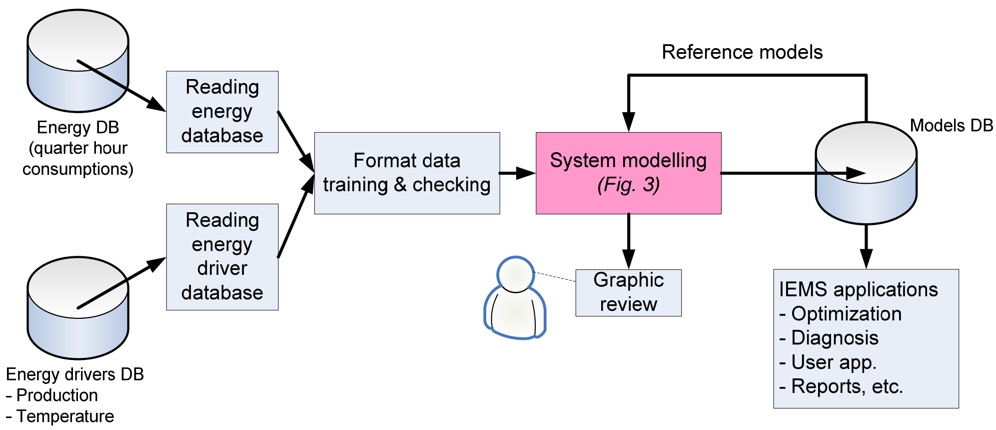


Figure 5.7. General system modeling implemented for the proposed IEMS.

5.5.3. Obtained chromosomes: final configurations

After running the LMFS, the following chromosomes were obtained:

Places: Workshop 1 (OS1) and press machines

The ANFIS configuration chromosome obtained by the MOGA was ***X1= 14, X2=3, X3=4 and X4=1.*** This means:

- ✓ Inputs: week day, time (in minutes) and work shift production;
- ✓ Three membership functions by input;
- ✓ Type of membership function is Gaussian bell function;
- ✓ The output function is lineal.

Place: Workshop 2 (OS2)

The ANFIS configuration chromosome obtained by the MOGA was ***X1= 13, X2=3, X3=4 and X4=1.*** This means:

- ✓ Inputs: week day, time (in minutes) and temperature;
- ✓ Three membership functions by input;
- ✓ Type of membership function is Gaussian bell function;
- ✓ The output function is lineal.

The results for the points of workshop1 (OS1, PM1, PM2 and PM3) got the same configuration. This can be explained because of press machines execute the same type of process, they are into the workshop 1 and they represent about 60% of its overall load. Therefore, it is hope that their models were similar or even equal, as it was the result.

However, for the case of the workshop 2 (OS2), the final input selection was different. One of the inputs was different in comparison with the other models. This difference can be explained because the process carry out into workshop 2 is not the same as in workshop 1. While the parts of the car body are built in workshop 1, in the workshop 2 these parts are welded to the chassis and so to build up the whole car body. Besides, the workshop 2 it is bigger than workshop 1 and its heating, ventilation and air conditioning system (HVAC) demands more power than the HVAC belonging to workshop 1. This can explain why the temperature is a better input for the overall consumption model in workshop 2.

5.5.4. Multi-site model validation

Figure 5.8 to 5.11 depict the obtained results for the four models. Circles and dots mark the real data and prognosis data respectively. It should be remembered that the first week was used for training and the second one for checking.

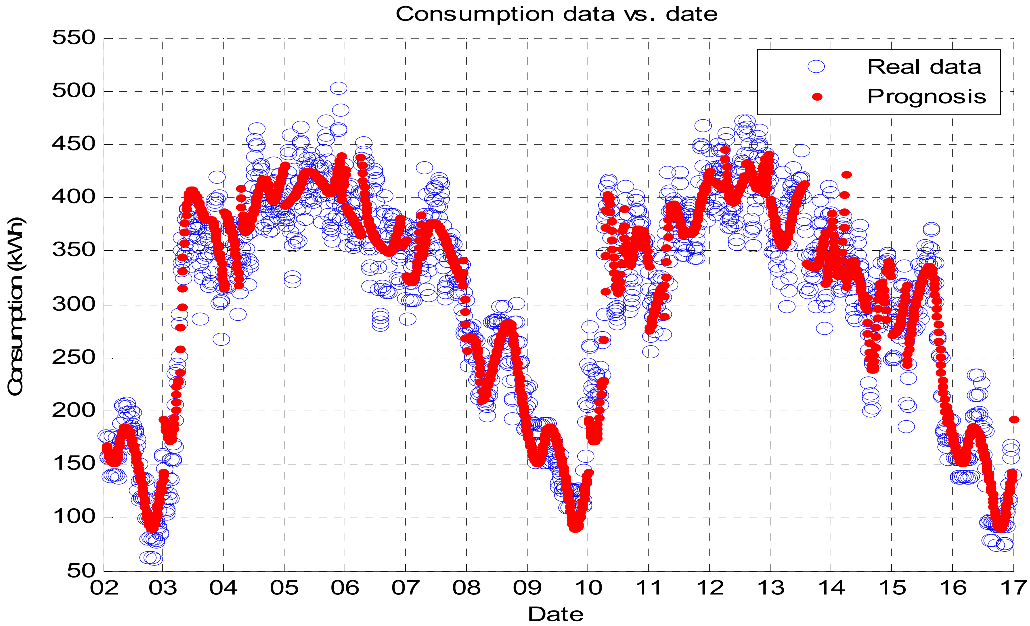


Figure 5.8. ANFIS model validation of overall energy consumption of the section OS1. Inputs: week day, hour and work shift production.

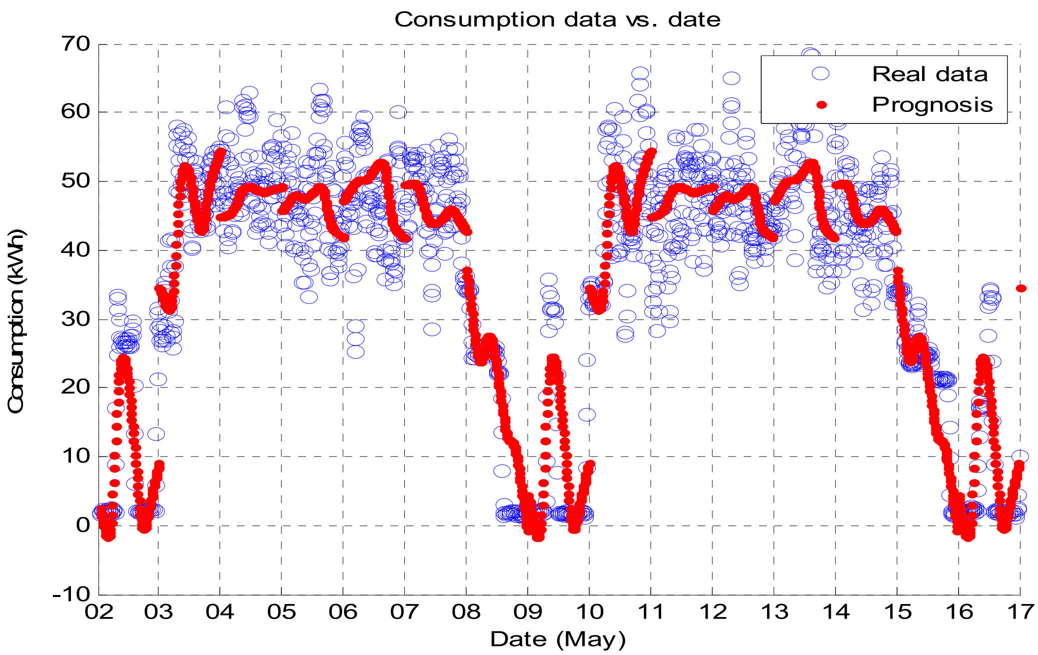


Figure 5.9. ANFIS model validation of energy consumption of press machine PM1. Inputs: weekday, hour and work shift production.

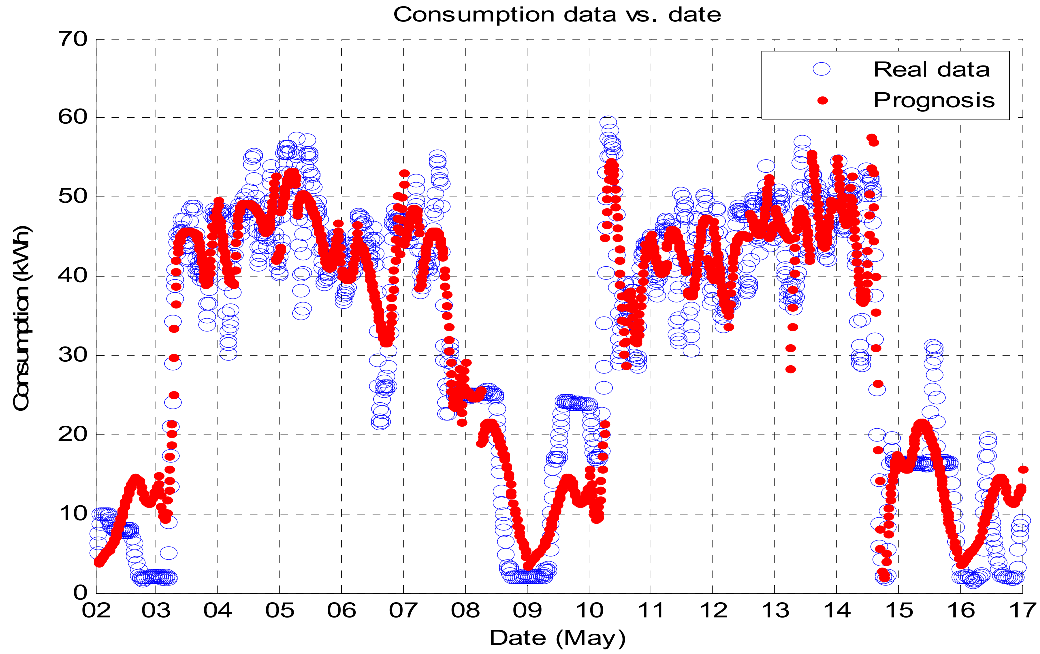


Figure 5.10. ANFIS model validation of energy consumption of press machine PM2. Inputs: weekday, hour and work shift production.

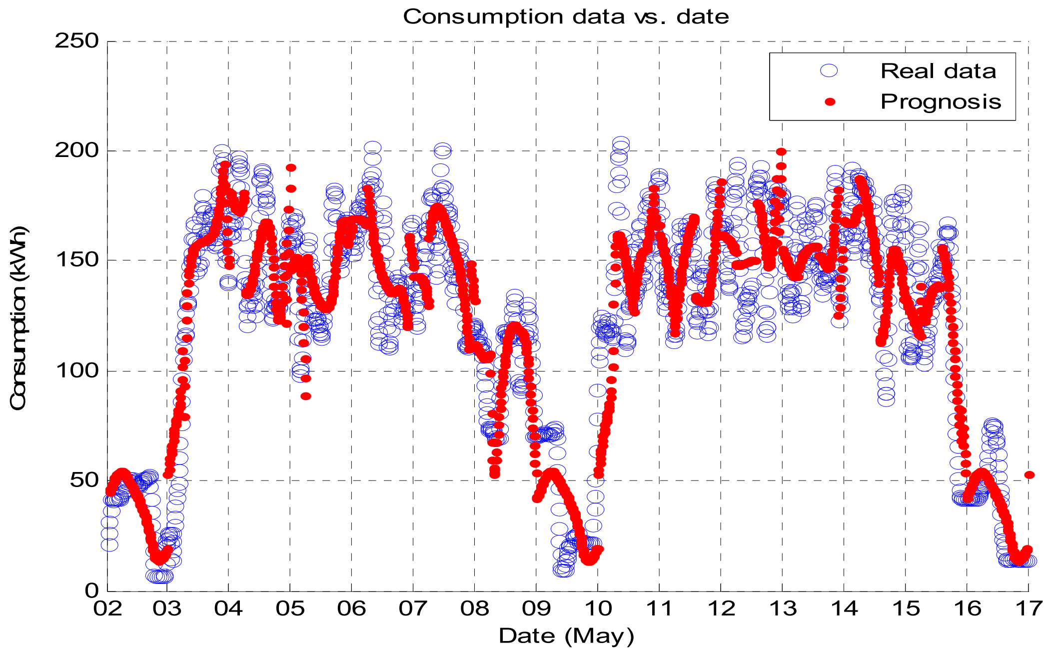


Figure 5.11. ANFIS model validation of energy consumption of press machine PM3. Inputs: weekday, hour and work shift production.

It can be seen that for the four models the forecasted load profiles fit quite well the real load profiles in both the training and checking week, despite the high variability of the load. In Table 2 is shown a summary of the obtained results. The result of OS2 was used as reference because the load profile of OS2 has less randomness and bigger consumption. For this case, the LMFS was able to get better load profile prognosis (Figure 5.12).

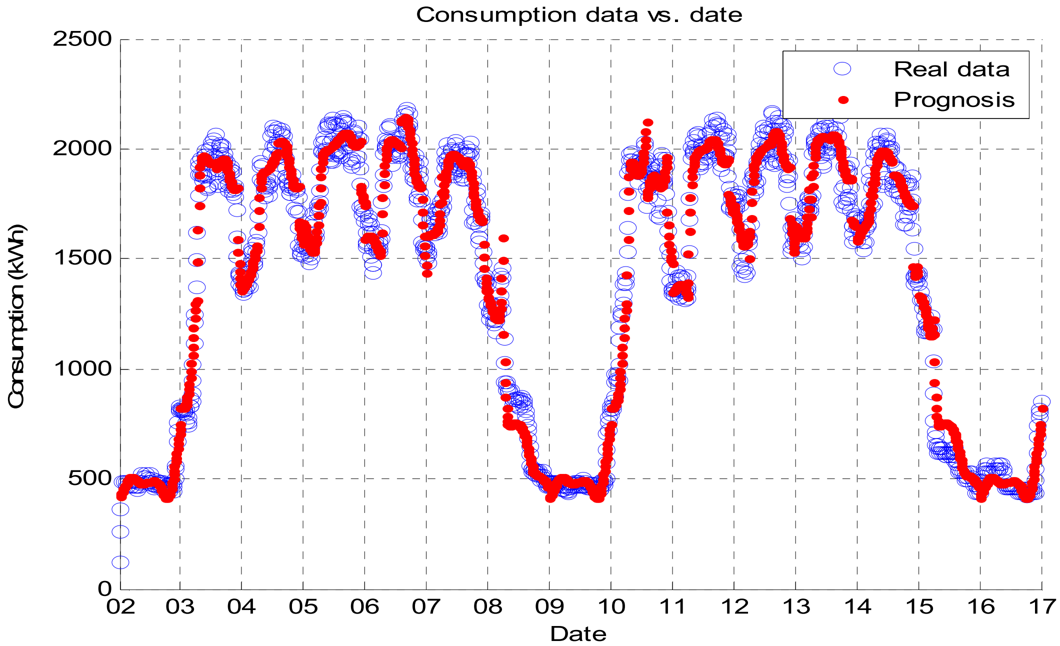


Figure 5.12. ANFIS model validation of energy consumption of other section (OS2). Inputs: weekday, hour and temperature.

Table 5.2. Results summary of multi-site load profile modeling.

Site	Min (kWh)	Max (kWh)	Mean (kWh)	Std. Desv. (kWh)	RMSE	RMSE/Max
PM1	0,8	68,3	35,3	18,5	6,1	9,0%
PM2	1,3	60,8	33,0	16,7	6,3	10,4%
PM3	8,8	212,3	122,9	51,9	21,0	9,9%
OS1	30,4	472,1	308,6	94,5	26,6	5,6%
OS2	120,8	2220,0	1397,3	618,6	90,9	4,1%

The relative RMSE parameter (RMSE/Max) shows how the accuracy of the LMFS is better when it is used with a load profile with less randomness and higher load demand (compare the Figure 5.12 with Figures Figure 5.8 to Figure 5.11). In fact, this relationship continues being true if higher levels of energy consumption are analyzed, as for example the overall plant consumption. This feature of load profiles was analyzed in chapter 3 in "User-side load profile analysis" section. Therefore, the proposed LMFS could be implemented not only for indoor installations, but also in power energy distribution forecasting.

The last observation should be considered when comparing the current obtained RMSE results with the obtained in other works [24, 45, 51]. Possibly, they get better RMSE results but they tested their systems always using high load profiles as substations for total consumptions for cities, which have not as high load variation as user side consumptions.

The obtained results show the adaptability and autonomy capacities of the proposed LMFS, which support the multi-site and self-configuration searched features.

5.6. Conclusions

We have studied and analyzed the LMFS process and proposed an algorithm to carry out each of its steps in order to get a self-configured and multi-site LMFS. We focused on the configuration step due to its complexity and importance in order to get an autonomous, adaptive and accurate LMFS.

The proposed modeling framework has been validated in a real user-side context. Sites with high variation or high randomness load profiles were selected in an industrial plant in order to test in deep the proposed LMFS. Under these challenging conditions, the autonomous configuration, adaptability, system learning and prognosis were tested successfully.

The system modeling was used to model the different sites and therefore to test the self-configuration and multi-site modeling capabilities. The results show that the proposal of using ANFIS algorithm together with the genetic algorithm is a good approach to autonomous and multi-site modeling of energy consumptions in industrial systems.

Supported by previous analysis presented in chapter three and the obtained results of this chapter, we can say that the proposed framework could be also suitable for load forecasting in utilities and in other types of load profiles of higher consumption level, which generally have less associated randomness.

6. Load modeling and forecasting based on dominant patterns

Following with the guide-diagram of LMFS presented in Chapter 3, in the current chapter is discussed the proposal focused in the pre-processing step. The objective of the work is to improve the *accuracy* of the final LMF and reduce the effect of the noisy LP problem.

Therefore, based in our former proposals and results on modeling real data from industry, here we present a LMF system (LMFS) based on detectable patterns, here called *dominant patterns*. These are indentified and selected by means of Hilbert-Huang transform and Hilbert-spectral analysis. As in the former chapter, GA is used to give self-configuration capabilities to the whole LMFS.

This approach, being the last of the thesis, besides using the dominant patterns to improve forecast accuracy, collects our experience in previous proposals and so to achieve the main objective of improving accuracy, adaptability and autonomy of LMFSs for iEMS on the user side.

CONTENTS:

6.1. Introduction and proposal

Currently, simple LMFS are mostly based on single structures of ANFIS, ANN or other adaptive network (AN) or hybrid algorithm. Despite of the wide use of these technologies on load forecasting for utilities and other time series modeling applications, LMFSs on user side demand new approaches that must be able to deal with noisy load profiles, which are generated for stochastic user behavior. For example, LMFSs based on complex structures as parallel ones.

In this context, this chapter presents a novel LMFS based on parallel AN, which has obtained improved accuracy and has been able to deal with noisy load profiles. Besides, it improves self-configuration capabilities and keeps the adaptability supported by adaptive networks. Figure 6.1 guides the steps where the proposed algorithms are focused in.

As presented in Figure 6.1, the proposed algorithm is focused on pre-processing step, where pattern detection, noise reduction and input-output correlation are faced. In order to deal with noisy LP problem on the user side, the proposed LMFS takes advantage of detectable patterns, here called *dominant patterns (DPs)*, present on user-side LPs. They are identified by means of a Hilbert Huang transform (HHT) and Hilbert-spectral analysis (HSA) and the more relevant ones are used to build up models of lower time resolution and better behavior to add useful information to the main model.

In comparison with wavelet and other decomposition techniques [52, 62, 63], HHT is found to be a powerful method for analyzing nonlinear and nonstationary data; it is not limited by the uncertainty principle and it presents the results in time-frequency-energy space for feature extraction [64, 65]. This last feature is the one we want to exploit to extract the main dominant patterns, which are found on the user-side LPs [66].

Furthermore, the derived empirical mode decomposition of Hilbert Huang transform can be used for denoising the consumption load profile and therefore to reduce the effect of noisy LP problem on the accuracy of the forecasting.

Taking advantage of the experience on the self-configuration algorithm presented in chapter 5, we use again GAs to support self-configuration of the LMFS [4].

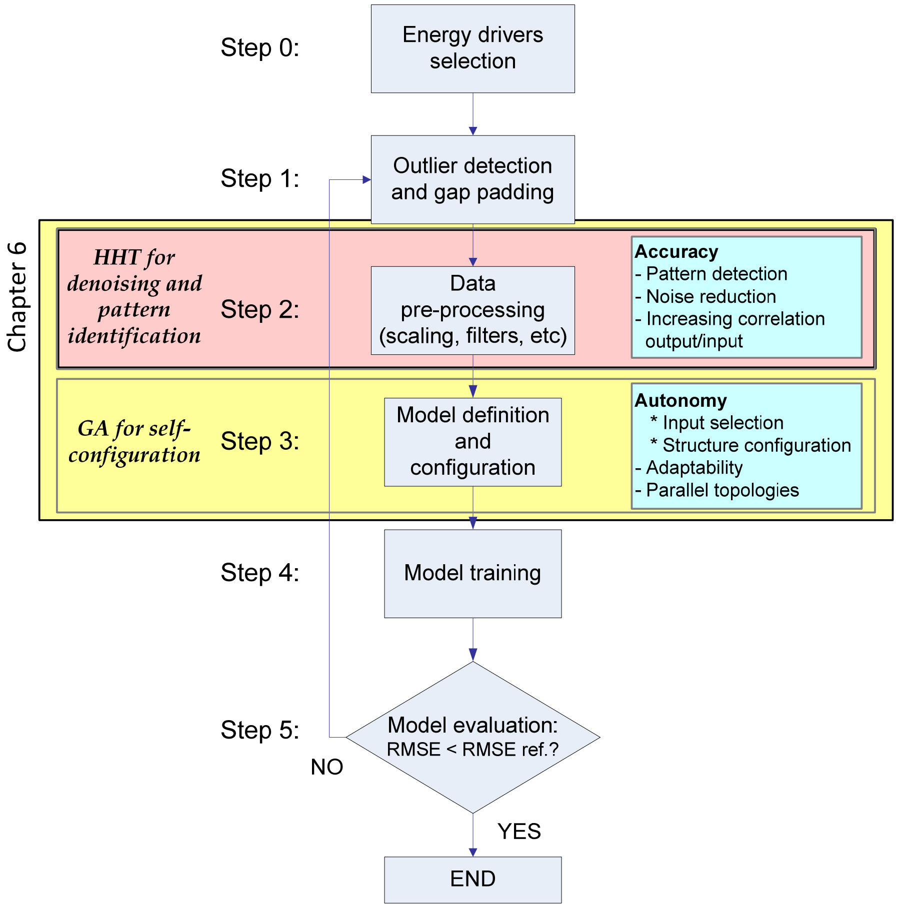


Figure 6.1. Proposal for a LMFS based on HHT and PM-DP analysis: pre-processing and model configuration.

Next section presents an analysis of user-side LP, which is aimed to detect visually DPs. Section 6.3 presents how using HHT analysis DPs can be identified. Then, in section 6.4, a model for the DPs is developed to support and supply useful information for the main modeling structure, improving its forecasting accuracy. The proposed method presents accurate forecasting and easiness to build up automatic models by using only the historical data, as it is shown by the obtained results, which are presented in section 6.5.

6.2. Load profile analysis: dominant patterns

6.2.1. User-side noisy LP problem

One of the main problems when dealing with modeling of LPs is when the target LP is full of random peaks, even though when a pattern can be visible (see Figure 6.2). Here we called this problem “noisy LP problem” and when happen on the user side we called it “user-side noisy LP problem”. Even though into the user side can be found a wide variety of LPs, in general can be stated a relationship between the magnitude, the time scale or time resolution of energy consumption and the noisy LP problem.

In order to show this relationship, in Figure 6.2 we have the LP of four different places at different levels of consumption. All of them have two weeks of hourly consumptions. The first plot (top-left plot in Figure 6.2) shows the LP of consumption of a press machine workshop of a car factory, where bodywork parts are made from sheet steels (consumption into the user side). The next plot in Figure 6.2 (top-right plot) depicts the LP of another workshop into the same factory, this time of the bodywork building process. Notice that the overall consumption of this last is about ten times bigger than the press-machine workshop and the LP has considerably less noisy LP behavior than the first one. On the other hand, the third plot in Figure 6.2 shows the overall consumption of school and even though the consumption is smaller than the two previous LPs, it has a similar noisy LP behavior to the bodywork workshop LP. Finally, the last plot in Figure 6.2 (bottom-right plot) presents the LP of consumption of a utility. Observe that this time the consumption is one or two orders bigger than the previous LPs and the noisy LP problem is much less noticeable in this LP. Therefore, in general, we can conclude that keeping the same time resolution, lower consumption levels will have bigger noisy LP problems. In the next section, the analysis of the noisy LP problem when the time resolution is changed is carried out.

For obvious reasons, on the user side, this noisy LP problem makes harder to obtain an accurate LF. This issue is the one we want to address in our proposal based on dominant patterns supported by Hilbert Huang analysis.

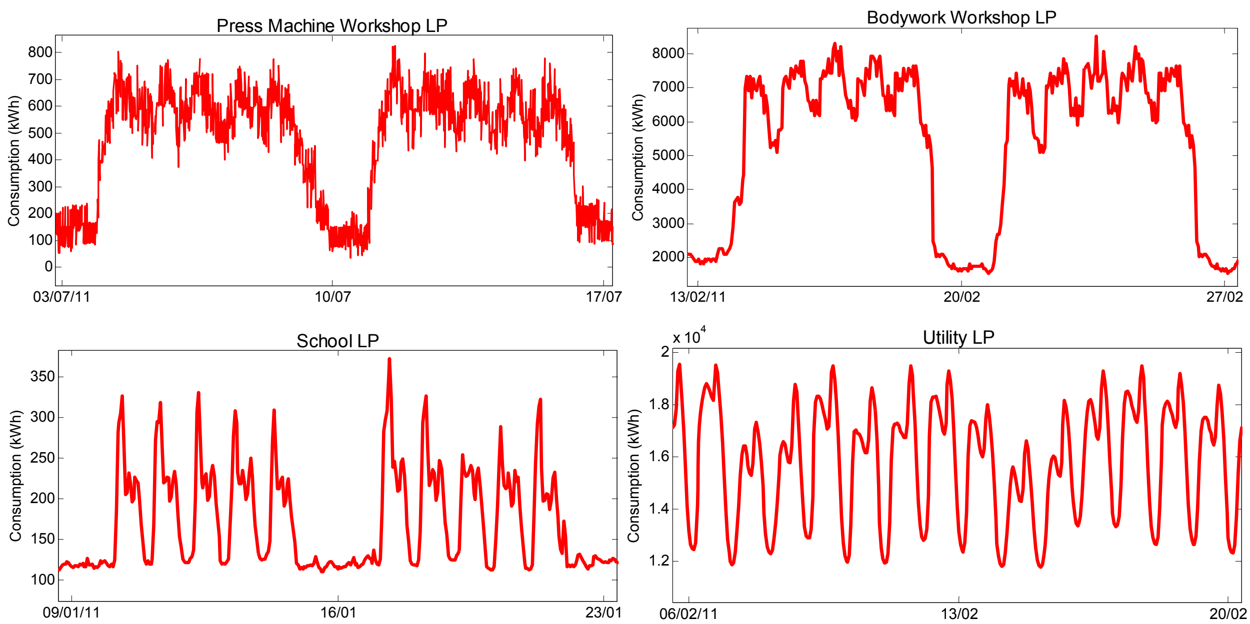


Figure 6.2. Different user-side LPs vs. Utility LP (bottom-right plot)

6.2.2. Visual pattern identification

Despite of the noisy LP problem, we can notice a quite similar pattern in the LPs in Figure 6.2. For example, analyzing the car bodywork workshop LP, in Figure 6.3, changing the scale and the time resolution of the LP and analyzing a month with eight (8) hours time resolution, some weekly patterns result evident. Checking the days where the lowest consumptions occur, it is evident that these patterns are due to production stops at weekends. Changing the scale, keeping the time resolution of the load profile, and analyzing the load profile for a week, one can notice daily patterns (Figure 6.4). They are stronger from Tuesday to Thursday. Monday and Friday have particular patterns and at weekends, the flat pattern is easily detectable. Observe also that with 1-hour time resolution (red line) the patterns are harder to visualize. However, for eight hours time resolution is clear the stepped pattern for almost every day, with exception at weekend. This stepped pattern could have relation with eight hours periodic work shift (three each day), daily temperature changes, scheduled production, etc. Besides, we can notice how at night work shifts, consumptions are lower than during sunlight hours.

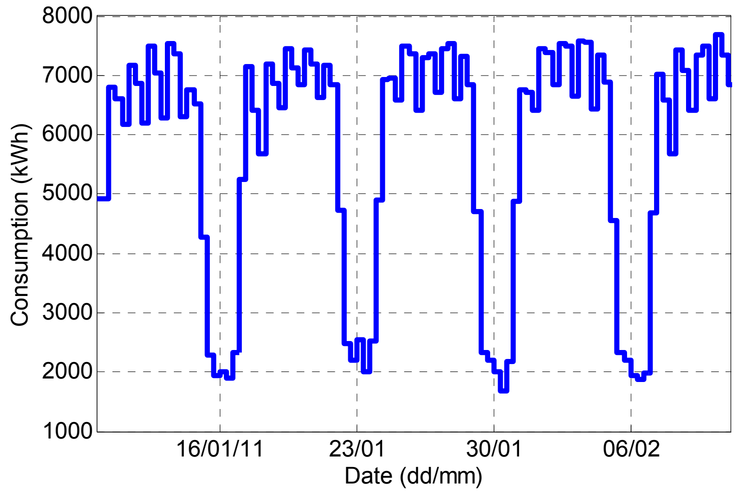


Figure 6.3. Analysis of eight hours load profile in one month.

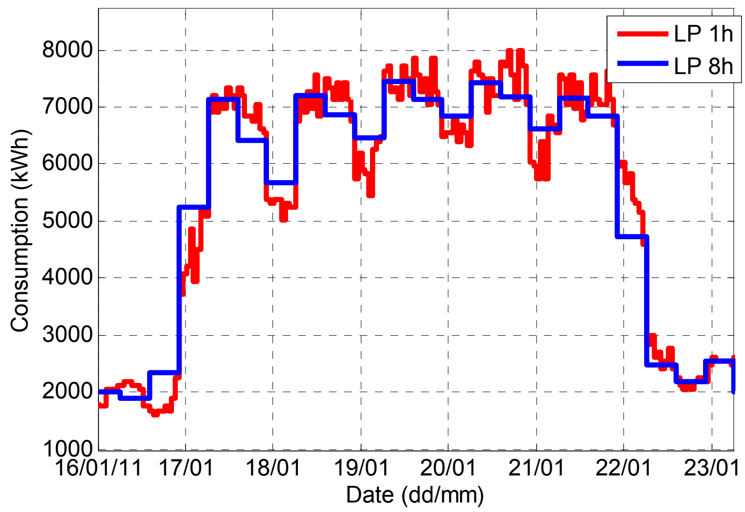


Figure 6.4. Hourly load profile and 8 hour load profile comparison in one week.

These graphical examples show us the visual patterns in load profiles and how changing the time resolution they can be easier to visualize and identify. These type either visible or no visible patterns are called here dominant patterns, DPs.

Because DPs have a reduced noisy LP problem, they are supposed to be easier to forecast. Therefore, the main idea of our approach is to take advantage of those dominant patterns that provide the most valuable information to the LMFS.

Next, Hilbert Huang transform (HHT) is used to detect and determinate these most stable and strongest DPs. Thus, the best DP(s) can be chosen to improve the accuracy of the final LMFS meanwhile the added complexity of the system keeps as reduced as possible.

6.3. Hilbert huang transform for DP detection

One of the main steps of Hilbert-Huang transform (HHT) is the empirical mode decomposition (EMD). It is based on a decomposition of the main signal in its simple intrinsic modes of oscillation. The idea of simple intrinsic modes and oscillation, which can be understood as periodicity or repetition, fits the basic concept of our proposal. Dominant patterns have a direct link with the intrinsic modes of oscillation. Therefore, HHT could be a useful tool to detect and select the main DPs to be used in the LMFS.

Among the main advantages of HHT analysis, the following can be stated: it permits a better physical analysis of the signal under test; it is appropriate for nonlinear and nonstationary signals, whose main frequencies depend on time; and it permits a time-frequency-energy analysis simultaneously, which is very useful for feature extraction. This last characteristic is the one we want to take advantage to track the main mentioned DPs.

As it has been pointed earlier, the LP on the user-side is subjected to random behaviors. Sudden changes, makes the user-side LP signal to be nonstationary. Besides, the relationship energy consumption with energy-drivers mainly is nonlinear. Time frequency analysis enables us to visualize the effect on frequency of the regular and no-regular changes on time, i.e. weekend and holiday's full stops and maintenance partial stops.

6.3.1. Hilbert Huang Transform

HHT is an empirically based data-analysis method, proposed by Huang et al. (1996, 1998, 1999) [64]. The procedure to implement the HHT is composed of two main parts: first, the decomposition of the signal in its intrinsic mode functions (IMFs), which is called Empirical Mode Decomposition (EMD), empirical Huang's approach; and second, the Hilbert spectral analysis (HSA) of each empirical mode, based on Hilbert transform. For the sake of simplicity, the basis and equations that define the HHT and the HAS have been moved to Appendix B.

There are presents the equations and algorithm to get the IMFs and the equations to get the Hilbert spectrum.

6.3.2. HSA for DP detection

In order to carry out the HHT and HSA, we analyze the weekly pattern obtained from 25 normal weeks. Normal weeks mean full working weeks, without holidays, but including weekends. The needed time resolution is one hour, so the pattern has this time resolution. Figure 6.5 illustrates the 25 LPs of each week and the weekly LP pattern (t_{mean}) obtained by means of an average operator applied to each hour:

$$t_{mean}(n) = \frac{1}{N}\sum_{j=1}^{N} WeekLP_j(n) \tag{6.1}$$

Where $t_{mean}(n)$ and $WeekLP_j(n)$ are respectively the values of the LP pattern and the week LP of the week jth for the hour n and N is the number of considered normal weeks, in this case of 25 normal weeks.

The details of the energy dataset used will be given later in the results section. Now, these normal weeks are used to explain the DP detection using HSA.

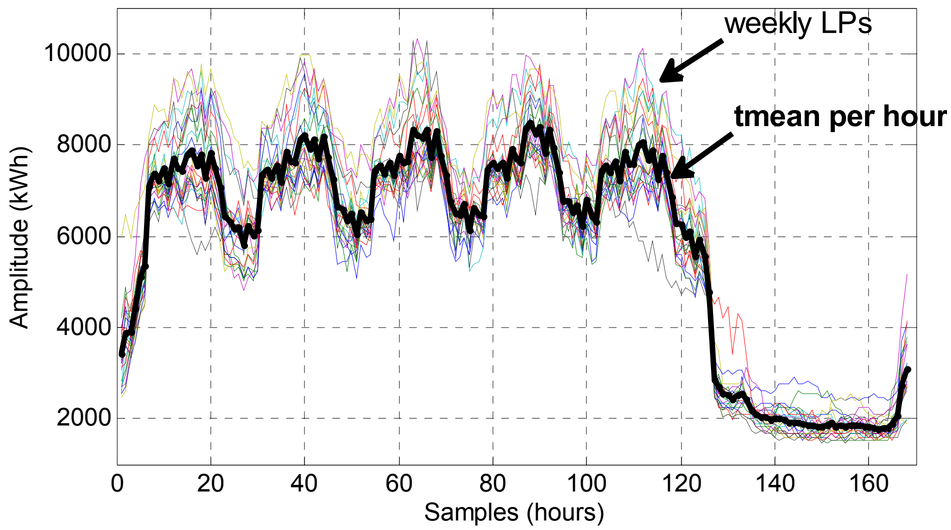


Figure 6.5. Weekly LP pattern, with hourly time resolution.

After apply the EMD to the weekly pattern, t_{mean}, we get the seven IMFs that are showed in Figure 6.6 (the number of IMFs obtained is determined by the EMD process and the stopping criterion). There we can see that the first IMFs correspond to high frequency modes and the last ones to low frequency modes. We can intuitively say that the first and second IMFs gather the consequence of random user behavior in the load profile and the other IMFs the most stable patterns of load profile. Furthermore, we can see that the most stable and strongest (highest amplitude) IMF is the fourth one. The main period of this IMF correspond to 24 hours. This type of strong and stable modes and their periods are the key parameters to define later the main DPs.

In order to confirm these observations we apply the HSA and we get the Hilbert spectral graphic plotted in Figure 6.7. This time, the highly variable frequency values in the spectral of the first two IMFs, c_1 and c_2, confirm our first intuition.

On the other hand, when we analyze the lower frequency IMFs, i.e. c_3 to c_6, we find more stable frequencies as it was noticed in the previous analysis. As well, we confirm our second observation about IMF c_4: it is one of the most stable and part of the energy of the signal relies in this IMF. Its frequency stability and energy makes it an excellent candidate to be used as part of the final LMFS, because, according to this analysis, it is predictable and it keeps important information of the original target.

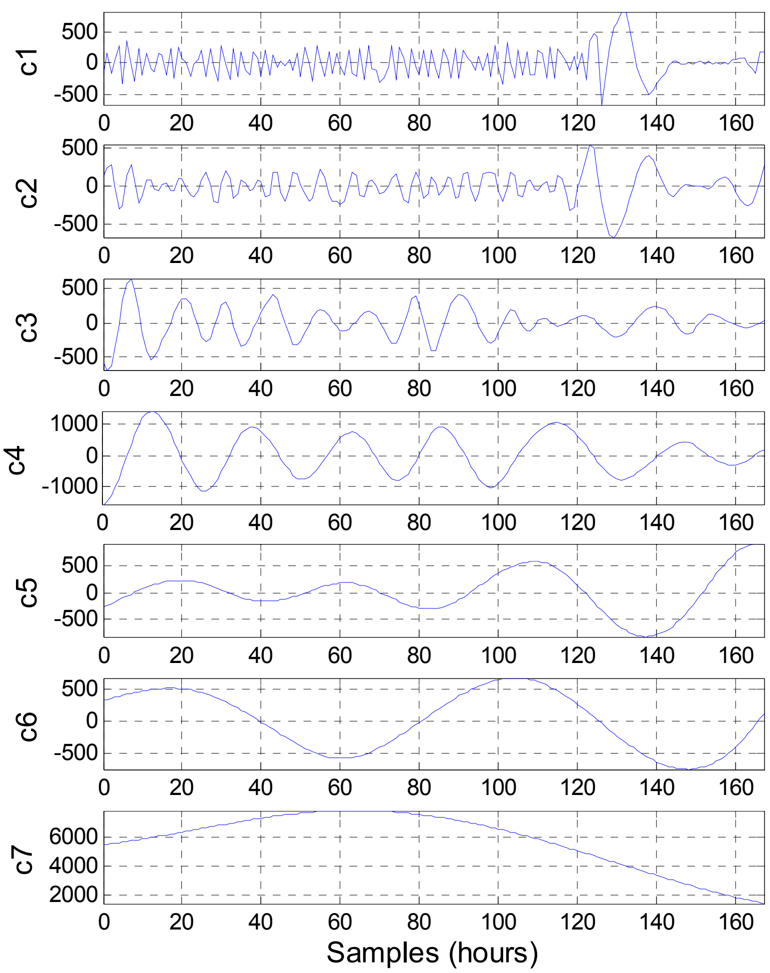


Figure 6.6. IMFs of the signal under analysis.

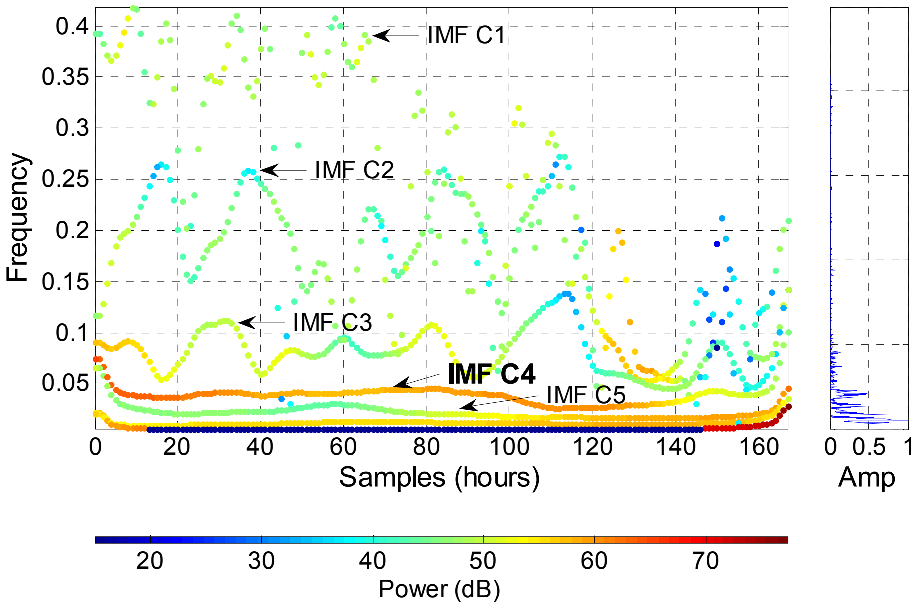


Figure 6.7. HSA of the signal under analysis.

6.3.3. New input candidates based on the obtained patterns

One of the methods to exploit information from patterns is using them as inputs. Therefore, we have create two new input candidates based on the full weekly pattern, t_{mean} (6.1), and a reduced version t_p, which is defined in (6.2) that omits the first three IMFs that were found the most unstable in the time-frequency spaces. There N is the number of obtained IMFs, $c_j(n)$ is the jth-IMF for the sample n and $r_N(n)$ is the residue of the EMD process.

$$t_p(n) = \sum_{j=4}^{N} c_j(n) + r_N(n) \tag{6.2}$$

Thereby, we give to the LMFS information of the full weekly pattern and of the most stable IMFs, from C_4 until C_7. Figure 6.8 presents the two new input candidates.

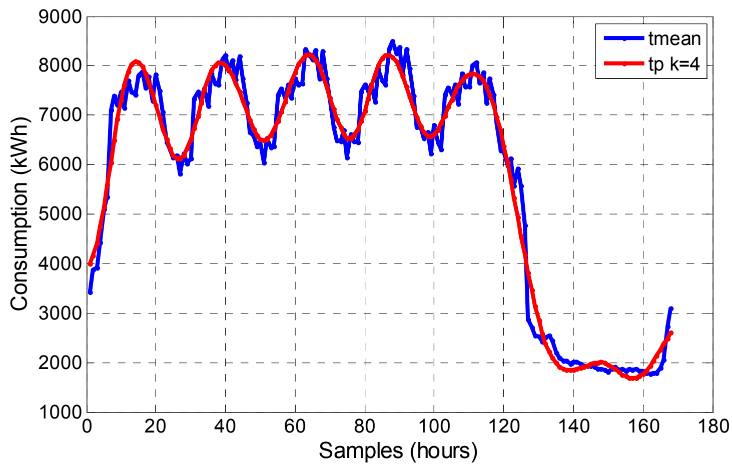


Figure 6.8. Weekly pattern with 1-hour time resolution (blue line) and reduced weekly pattern (red line).

6.4. Partial models of dominant patterns (PM-DP)

As it was pointed out, by means of time resolution reduction, patterns become more visible and with les noisy LP problem and therefore, they become easier to be forecasted. Therefore, a modular approach is suitable to take advantage of these DPs and improve forecasting accuracy. Here this modular approach is called "Partial models of dominant patterns (PM-DP)". Partial models (PMs) are local or expert models dedicated to model and forecast a particular version of the target LP, which has less time resolution than the original target LP. PMs can be based on any type of adaptive network or other modeling technique. However, here we used ANFIS and ANN to implement the PMs because they were found to be two of the most successful data driven approaches in the current literature.

The information obtained from PMs is added to the main PM by means of an adaptive addition, which is implemented using an ANFIS or ANN according to the selected adaptive network. Figure 6.10 depicts our proposal. It has to be considered that Figure 6.10 sketches the process of pre-processing and configuration of a LMFS and not the forecasting process itself. Once the LMFS has been configured (modeling process), the forecasting process demands a less complex structure.

Taking as reference the flow diagram presented in Figure 6.1, HHT analysis (HHTA) is carry out on step 2, i.e. data preprocessing step. After the HHTA is applied to the vector $\mathrm{T_{mean}}$, which corresponds to the dataset obtained from the mean target signal $\mathrm{t_{mean}(n)}$ (6.1), we got the parameter r, which contents the time resolution of the partial models (PMs). From the same analysis, the denoised vector $\mathrm{T_p}$ is obtained. This is the dataset of the reduced version $\mathrm{t_p(n)}$ (6.2). In a parallel process, the LMFS is able to build up the lagged variables, Tdi, from historical data and to change the time resolution of the candidate input variables (Xi) of the selected partial models. The selection of the PMs is a decision taken by the user based on the criteria that were explained in the former section.

The step 3, model definition and configuration, is supported by GAs, as is sketched in Figure 6.9. The GAs get the optimal final inputs for each PM and the number of membership functions when an ANFIS is used or the number of hidden neurons when an ANN is used instead. In [4] it is explained how the GA-based self-configuration process is carried out.

Finally, the output of each partial model, Yj, goes to an adaptive adder, which can be implemented by an ANFIS, ANN, a lineal combination of the Yj or any other adaptive addition. The output (Y) of this last unit is the forecast of the target signal.

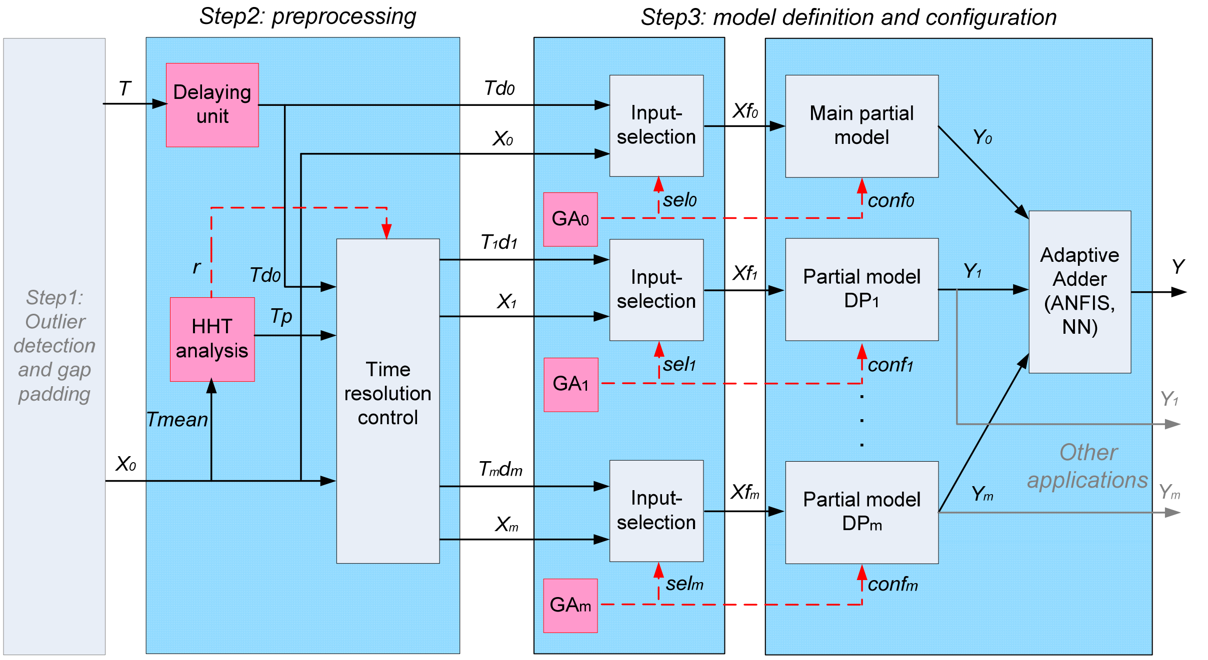


Figure 6.9. Proposal for a LMFS based on HHT and PM-DP analysis: preprocessing and model configuration.

Furthermore, using a parallel topology as the proposed one, the partial results (Y_1 to Y_m) can be useful for other upper applications, which do not need a high resolution forecasting. For example, for daily consumption estimation and control, energy wasting detection based on normal consumptions, etc.

Once the LMFS has been configured, the red lines and pink blocks in Fig. 10 are not more used and the LMFS now can be used for forecasting tasks putting the right inputs in its input layer. Notice that $\mathrm{T_{mean}}$ and $\mathrm{T_p}$ after configuration process become static patterns that are used as inputs. Beside, the delaying unit is not used anymore because in the forecasting mode it is supposed that the inputs are given to the LMFS in the right time order.

6.5. Implementation and Results

6.5.1. Energy database

In order to test and evaluate the proposed method, real energy consumption and outdoor temperature data from a car factory were used. The data were taken from an EMS database and it corresponds to one of the biggest workshops of the car factory, in which the process of building up the bodywork is done. This place was chosen because the variety and high noisy LP problem of the loads, so the method could be tested with a challenging LP.

The raw energy database starts from June 5 of 2010 to March 25 of 2011, with one-hour time resolution. However, as mentioned on HHT analysis section 4, the database was reduced to keep only the “normal weeks”, this is 25 full working weeks, without holidays, but with weekends

(Figure 6.5). The reason is holidays introduce considerable outlier behavior in the weekly pattern. In this stage, this problem is not considered and so it is avoided deleting no normal weeks.

6.5.2. Scenarios

We have tested the proposed scheme under two scenarios:

1. Scenario 1 uses directly the full available energy database, 25 normal weeks. The first 15 normal weeks (60%) are used for pre-processing, configuration and training (Fig. 2, steps 2 to 4). The last 10 weeks are used for checking and validation (step 5). This scenario emulates the case when a LMFS is implemented and there is a previous EMS database available.
2. Scenario 2 emulates the behavior of an increasing energy-database (eDB). When a LMFS is implemented in a new iEMS or EMS normally, there is not available an eDB from the beginning, so the LMFS has to be periodically updated with the incoming new data. In this scenario, the available database for training and checking starts with six normal weeks and the models keep updating with each "new" normal week until reach 25 weeks as in the scenario one. The percent of data for training always was 60% and 40% for checking.

Notice that the percent of training and checking data was always the same and the percent of checking data is considerably equal to percent of training data. This was chosen so in order to detect overfitting problems.

On the other hand, ANFIS and ANN were the selected adaptive networks for basic structures of the full LMFS. The idea of each test is to compare simple structures versus distributed and modular structures based on partial models of dominant patterns.

Finally, the forecasting target for both scenarios is one day ahead with one-hour time resolution (next 24 samples). As explained in the introduction section, this type of short-term LF is useful for predictive peak control, predictive energy source optimization (choose, in the future, the best combination of supplies from a pool of them), energy waste supervision, predictive energy supervision, etc.

6.5.3. Energy drivers or input candidates (step 0)

Using the available datasets (temperature and energy consumption), the input candidates for main and partial models were:

- the week day (WED),
- time in minutes (h),

- temperature (temp),
- continuous working day (CWD),
- one week lagged consumption ($t'(n-7days)$),
- one day lagged consumption data ($t'(n-1day)$),
- one week pattern based on the averaged consumption of training normal weeks, t_{mean},
- one week pattern based on the low frequency IMFs, t_p.

For one day lagged consumption, corrections were carried out in order to avoid using wrong information. For example, it is not correct using Sunday information to forecast Monday consumptions or Saturday information to forecast Sunday consumptions. One week lagged consumption was used to replace the wrong information.

6.5.4. Outlier detection and gap padding (step 1)

The datasets to be processed in this step are the energy consumption and temperature datasets. The other variables are built up from recorded time and date or they are derived directly from the consumption target, so they do not need to be checked.

This step was carried out following the statistical threshold-based detection presented on equation (3.1) in Chapter 3 for outlier detection. Gap padding was implemented using interpolation technique for short gaps. For long gaps, data of one or seven days before were used to pad the gaps.

Furthermore, the selection of normal weeks from the full energy consumption database was carried out in this step. At the end of this process, the raw energy database was reduced from 268 days with one-hour time resolution (6432 samples) to 175 days (25 normal weeks) with one-hour time resolution (4200 samples).

6.5.5. Data pre-processing (step 3)

In this step, using only training data, first we normalized the variable datasets dividing every variable for the maximum value of the respective full dataset. This maximum value is only a reference for scaling properly the data. It has to be considered when converting back the scaled forecast to the real units.

Then, with a reliable database, we applied the HHT analysis to the target signal to get the dominant patterns as explained in section 4.2. The result of this analysis is the detection of the dominant patterns and their respective time resolution. Remember that to select a specific DP the criteria are; it has to be relatively frequency steady and its main oscillation frequency has to be congruent with the time resolution of the target LP. In the analyzed datasets for scenario 1 and 2, we got that the strongest and most stable IMF or intrinsic oscillation mode was the fourth one (see Figure 6.7). This IMF has a main periodicity of 24 hours and we use this value directly to define the time resolution of the selected partial model.

For the implementation, we believed that it would be desirable to use only the related information of one IMF because either the other IMFs were unstable (C_1 to C_3, higher frequencies) or they had low information for only one day ahead forecasting (C_5 to C_7, lower frequencies). It has to be stated the lower frequencies could be useful for long-term forecasting or longer horizon forecasting, e.g. one week forecasting with one-day time resolution.

Notice that the chosen IMF could be related by daily periodical behavior of natural day cycle and the factory timetable, production schedule and normal life cycle of 24 hours. In the specific case of the factory under test, the factory is working continuously for 24 hours during working days (Monday to Friday).

6.5.6. Model definition and configuration (step 4)

For model definition and configuration, we applied the GA-based self-tuning algorithm [4] to choose:

- Final inputs for each PM (i.e. main PM (Y_{1h}) and dominant PM (Y_{24h})).
- Number of membership functions when ANFIS was the basic modeling structure or
- Number of hidden layers when ANN was the basic modeling structure.

Results are presented on Table 6.1. The data used to carry out this step was the training data defined for each scenario. One interesting result was that for both scenarios the obtained configuration remained without significant changes in all the executions of the LMFS (scenario 2 involves multiple executions of the proposed LMFS, one for each new introduced normal week).

Table 6.1. GA self-configuration decoded results. Input candidates, number of membership functions (ANFIS) and number of hidden neurons (ANN).

Input candidates	ANFIS		ANN	
	Main PM	24 h PM	Main PM	24 h PM
WED: week day	Yes, 2 mf	Yes, 5 mf	Yes	Yes
h: time of the day in minutes	Yes, 2 mf	No	No	No
temp: outdoor temperature	No	Yes, 4 mf	No	No
WD: working day	Yes, 2 mf	Yes, 2 mf	Yes	Yes
t'(n-168): denoised and 7 days delayed consumption	No	No	No	No
t'(n-24): denoised and 1 day delayed consumption	Yes, 2 mf	No	Yes	Yes
tm: weekly pattern	No	No	Yes	Yes
tp: reduced tm, using the IMFs C4 till C7	No	No	Yes	Yes
			9 hidden neurons	5 hidden neurons

An unexpected result after carry out the self-tuning GA was that the inputs for the ANFIS-based LMFS were different from the inputs for the ANN-based LMFS. This could be explained because the differences between the ways each structure maps the output and input data and the possible redundancy of information on the chosen input candidates.

6.5.7. Training and assessment

Scenario 1 (full eDB available)

Once each basic structure was configured, the full training process for the whole system was executed. The results are presented on Table 6.2 for scenery one. Improvement is calculated using: *%Improve.=(RMSE_simple_model-RMSE_PM-DP)/ RMSE_simple_model*100%*.

Table 6.2. RMSE and MAPE for checking validation (40% of full data)

	Simple model (1h)	PM-DP (1h+24h)		Simple model (1h)	PM-DP (1h+24h)	
Basic AN	RMSE	RMSE	% Improve.	MAPE	MAPE	% Improve.
ANN	0,037	0,029	22%	5,9	4,8	19%
ANFIS	0,036	0,030	17%	6,1	4,9	20%

For a graphic assessment of the proposed model, Figure 6.10 presents a sample of the obtained results for one week of forecasting consumptions using checking data. It can be seen how the PM-DP effectively corrects the error due to wrong offset forecasting (lack of lower resolution information).

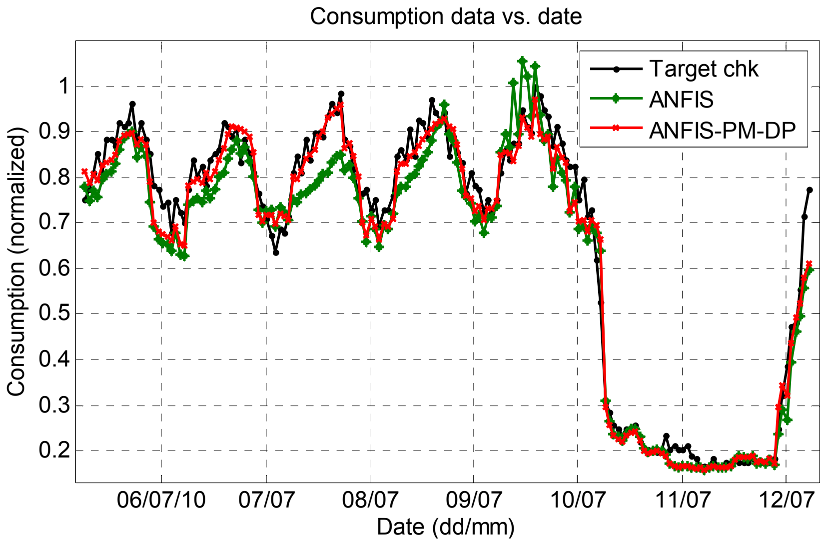


Figure 6.10. One week load forecasting using checking data.

Scenario 2 (eDB growth and weekly model update)

Table 6.3, Table 6.4 and Figure 6.11 and Figure 6.12 presents results for scenery two. Table 6.3 and Figure 6.11 correspond to the assessment of ANN-based LMFS and Table 6.4 and Figure 6.12 for ANFIS-based LMFS.

Table 6.3. RMSE for ANN-LMFS, using checking data (40% of available data), emulating energy database growth.

Season	eDB size (weeks)	ANN RMSE	PM-DP ANN RMSE	%Improve.
Summer	6	0,0593	0,0638	-8%
Summer	7	0,0608	0,0541	11%
Summer	8	0,0586	0,0526	10%
Summer	9	0,0601	0,0561	7%
Fall	10	0,0622	0,0563	9%
Fall	11	0,0603	0,0508	16%
Fall	12	0,0514	0,0430	16%
Fall	13	0,0491	0,0410	17%
Fall	14	0,0487	0,0435	11%
Fall	15	0,0613	0,0529	14%
Winter	16	0,0526	0,0461	12%
Winter	17	0,0532	0,0463	13%
Winter	18	0,0534	0,0453	15%
Winter	19	0,0533	0,0427	20%
Winter	20	0,0572	0,0491	14%
Winter	21	0,0525	0,0424	19%
Spring	22	0,0471	0,0391	17%
Spring	23	0,0404	0,0322	20%
Spring	24	0,0377	0,0314	17%
Spring	25	0,0370	0,0290	22%

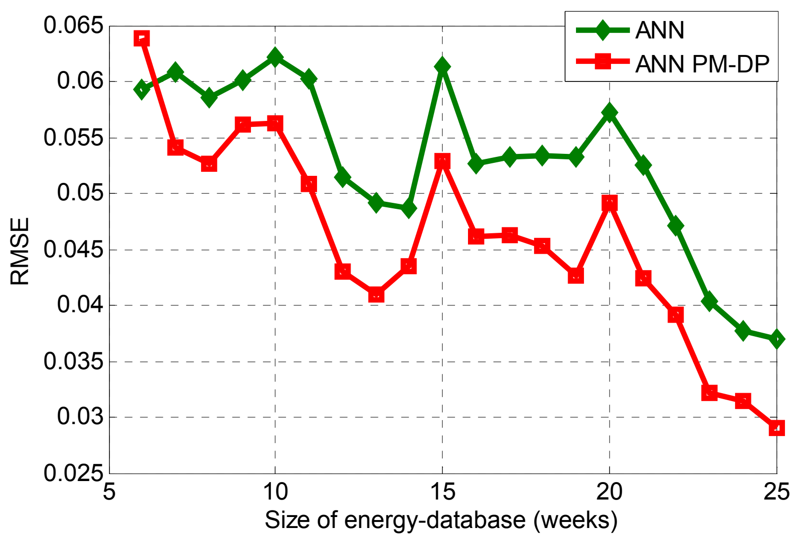


Figure 6.11. RMSE evolution according to eDB growth. ANN test. Models updated each week. Scenario 2.

Table 6.4. RMSE for ANFIS-LMFS, using checking data (40% of available data), emulating energy database growth.

Season	eDB size (weeks)	ANFIS	PM-DP ANFIS	%Improve.
Summer	6	0,062	0,053	15%
Summer	7	0,068	0,060	11%
Summer	8	0,072	0,065	10%
Summer	9	0,058	0,050	13%
Fall	10	0,059	0,052	11%
Fall	11	0,053	0,047	13%
Fall	12	0,051	0,043	16%
Fall	13	0,045	0,038	14%
Fall	14	0,057	0,050	12%
Fall	15	0,058	0,050	14%
Winter	16	0,054	0,046	15%
Winter	17	0,054	0,043	21%
Winter	18	0,055	0,047	16%
Winter	19	0,054	0,046	15%
Winter	20	0,053	0,044	17%
Winter	21	0,051	0,045	10%
Winter	22	0,048	0,042	13%
Spring	23	0,041	0,033	19%
Spring	24	0,038	0,030	21%
Spring	25	0,036	0,030	17%

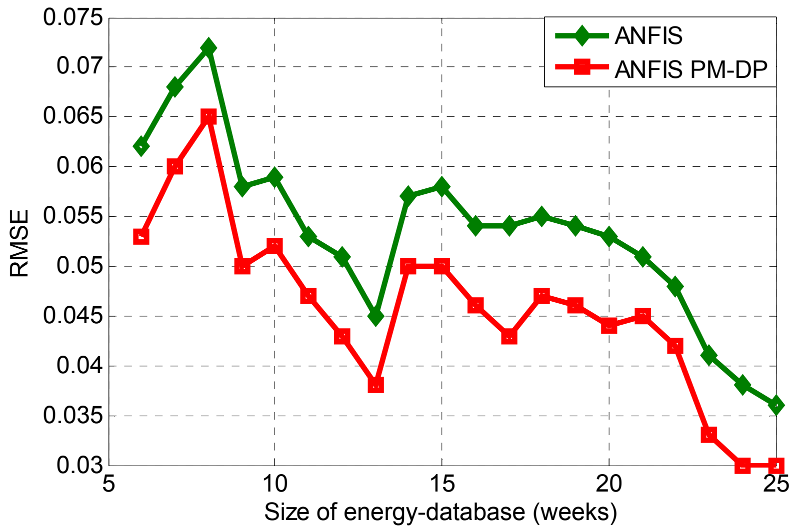


Figure 6.12. RMSE evolution according to eDB growth. ANFIS test. Models updated each week. Scenario 2.

6.5.8. Observations

According to the results obtained in both scenarios, we can say that effectively the information supplied for dominant patterns of lower time resolution improves the final forecasting result. We have defined this effect as an offset correction due to the effect of movement up or down of the forecasted LP (Figure 6.10).

Moreover, the degree of improvement will depend on the quality of the supplied information. This can be quantified by the RMSE of the used PMs, in this case the 24 hours PM. For this reason, the optimal or near-optimal configuration of each sub-model is key point to get the best performance of the LMFS. Therefore, the use of self-configuration algorithms is important to get the desired accuracy. Besides, this implementation solves other problem, which is the configuration of various models instead of only one main model.

Second scenery results shows that the propose LMFS work quiet well with low information and the improvement percent keeps always upper the 7% and in average was 14% for both cases (ANN and ANFIS based LMFS). The only exception was for the LMFS based on ANN when the eDB was 6 weeks only. In this case, the information to train the lower resolution model possibly was not enough to get a reliable forecast.

Finally, the proposed LMFS presents the advantage of generating lower resolution forecasts that can be used on other upper applications, which do not need high-resolution forecasts. Taking into account all these results and observations, we can conclude that the proposed LMFS outperforms the modeling simple structures usually used.

6.6. Conclusions

From the hypothesis of existence of dominant patterns (DPs), we have demonstrated their veracity by means of graphical analysis and mathematical transform. It was found that they occur at different time resolutions and their stability and modeling easiness depend on this parameter. The two last features qualitatively determined how useful for the LMFS can be the information obtained.

In order to detect and select the most useful DPs, we proposed Hilbert-Huang transform (HHT) and Hilbert-spectral analysis (HSA). The results shown that the HSA of IMFs, allowed us to visualize stability and strength of the IMFs and base on these parameters, we can choose properly the main DPs. Nevertheless, the method is open to use other time-frequency transforms or statistical operator to find the DPs.

Then, a new scheme to take profit of these dominant patterns to improve forecasting accuracy was presented and tested satisfactorily. We use a parallel topology of partial models of dominant patterns (PM-DP). Each PM is dedicated to model and forecast the associated DP and one of them the target LP. This is called main PM. Each PM is based on a single adaptive network as for example an ANFIS or ANN. However, the method can be applied to any other type of modeling structure.

The obtained results were satisfactory as long as PMs of main and dominant patterns were properly configured. The lower the RMSE of PMs was, the more improvement in the final output of the LMFS we got. Because of this proposition and the increment of number of basic models to be configured, a self-configuration algorithm via GA was proposed to guarantee a properly configuration of each PM.

This whole scheme demonstrates to be a very powerful and promising tool in the energy forecasting area. The methodology and the focus given to pattern repetitions on load profiles on user side, represent a breakthrough in applied modeling techniques.

7. Conclusion

The main conclusions of this thesis work, as well as our outlook of related trends and future work are presented in this chapter.

CONTENTS:

7.1. Conclusions

The starting idea of this thesis has been to study, design and propose a user-side load modeling forecasting system (LMFS) aimed at supporting an intelligent energy management system (iEMS). As explained at the beginning of the thesis, for an iEMS the accuracy of its LMFS is as important as its adaptability and autonomy.

We started with a study of the load profile (LP) on the user side, pointing out the main troubles and defining a suitable methodology for building up an LMFS. In arriving at the main conclusions of these studies we found three major problems, to which our proposals were oriented toward solving:

- High degree of randomness in load profiles associated with random behavior of the user (here called noisy LP problem)
- Wide variety of loads
- Great number of loads for modeling for the same LMFS.

The above problems highlight the need for an LMFS which has: **autonomy**, so that it can build multiple models with very little or no human intervention; **adaptability**, so the LMFS can deal with different types of LPs associated with different types of loads; and **accuracy**, in order to reduce the errors generated from the noisy LP problem and to meet the precision, time resolution and forecast horizon conditions required by the iEMS.

Taking into account the state of the art on load forecasting in the field of energy, we found that hybrid algorithms from computational intelligence, statistical and signal processing functions were the best tool candidates for designing the LMFS with the required features. Thus, three proposals were presented, based on such algorithms:

1. Evolutionary training for new adaptive networks.
2. Autonomous configuration and multi-site modeling.
3. Load modeling and forecasting based on dominant patterns.

The first of them tried to exploit the flexibility of genetic algorithms (GA) and its ability to solve multi-objective optimization problems. Therefore, an evolutionary training algorithm (ETA) was presented for training adaptive-network-based-fuzzy inference systems (ANFIS). The employed GA takes into account both training and checking errors simultaneously, so overfitting can be avoided; thus, excellent generalization capabilities were obtained.

An additional objective was that ETA should be easily adaptable for training new adaptive networks (AN), which were based on classic ANFIS. Thus, a modified ANFIS was proposed. The modification consists of changing its output membership functions, which were polynomials, to exponential functions. Using them as output functions provides a more flexible and non-linear function which can deliver finer adjustments in the ANFIS output layer; and this also has a positive effect on the generalization ability of the LMFS. ETA successfully fulfilled its purpose. With just a few modifications in the decoding functions of GA, an appropriate training algorithm was obtained for the new structure.

On the other hand, the added computational burden and the corresponding elapsed time was a drawback of the proposed system. This was reduced by the use of parallel GA, which exploits multiple CPU-processors throughout the computer simulation.

The second proposal, aimed at autonomous configuration based on a hybrid algorithm, was based on the experiences with GA and ANFIS; their potential was evident in the previous approach to modeling energy consumption on the user side. The hybrid algorithm was focused on improving autonomy and adaptability features. Thus, a self-configured and multi-site LMFS was obtained. Once again, the flexibility and adaptability of GA and ANFIS was exploited successfully.

The proposed LMFS achieved excellent results in managing consumption data from five different sources. Three of them were large machines belonging to same workshop. Another one was the selfsame workshop where the machines were. And the last one was another workshop whose overall consumption was considerably higher than the previous loads.

Moreover, by managing and modeling different types of loads and with different levels of consumption, we validated how the level of consumption was directly related with the randomness of the LP and how this affects the accuracy of the final forecast. This was clear when we compared: the results of the largest workshop with the results of the other loads; the result of the overall consumption of the smallest workshop with its machines; and the result of the largest machine with the other two machines. The absolute root mean squared error (RMSE) was always higher for higher consumptions, but relative RMSE (*RMSE/max. consumption * 100%*) was lower for higher levels of consumption.

Finally, and based also on one of the starting hypotheses of the thesis, our third and last hybrid LMFS sought to take advantage of the existence of dominant patterns (DPs) in user-side LPs of consumption. The objective at this stage was to improve the accuracy of final forecasting and to maintain the autonomy and adaptability achieved with the previous proposals.

In order to achieve these objectives, we took advantage of time-frequency transforms. This allowed us not only to identify the DPs, but also to visualize and analyze their stability over time. The chosen transforms were the Hilbert-Huang transform (HHT), which is based on the empirical mode decomposition (EMD) of the target LP, and the Hilbert spectral analysis (HSA), which obtained intrinsic mode functions (IMFs) in time-frequency space. This methodology allowed us to detect and quantify the information that a given dominant pattern would have supplied to the main forecasting.

In order to leverage the extra information from DPs, our proposal was to use a parallel topology of partial models of dominant patterns (PM-DP). Each PM is dedicated to modeling and forecasting the associated DP, one of them being the target LP. This is called main PM. The obtained results were satisfactory as long as the PMs of the main and dominant patterns were properly configured. The lower the RMSE of the PMs was, the more improvement in the final output of the LMFS we obtained.

Since our proposed PM-DP technique involves configuring a higher number of models, we solved this problem by leveraging our previous proposal for self-configuration via GA. Besides, having models and forecasts of the same LP with different time resolutions can be useful in supporting other applications that demand lower resolution forecasting of LPs.

In general, the obtained results were satisfactory and in the end an autonomous, adaptable and accurate LMFS was obtained, which was the focus of this thesis from its definition. These results fulfill iEMS requirements for implementing applications that control, diagnose and support decisions. Even though the target iEMS was for industrial users and large buildings, the presented architecture, the obtained LMFS and their functionalities can be extrapolated to other types of users. Moving down the level of consumption, we can recommend iEMS for home users; and moving up the level, iEMS is perfectly acceptable for utilities, cities, etc.

7.2. Trends and future work

This thesis has been motivated and oriented toward supporting iEMS' requirements of load modeling and forecasting on the user side. These come from iEMS' upper level applications, which are thought to generate energy savings and need a reliably LMFS to work properly.

Therefore, trends and future work related to these subjects have been divided in two sections, one dedicated to iEMS trends and other for depicting the next steps for exploring and improving the current LMFS approaches.

7.2.1. iEMS trends

iEMS for home users

The growth of energy consumption databases at the domestic user side, mainly associated with implementation of smart-grid paradigm, is creating a new field of research, development and, why not to say, business. The availability of such amount of energy consumption data, which years ago did not exist, creates new possibilities for the development of advanced applications oriented toward improving energy efficiency. Thus, it is expected that iEMS, as described in this work, will be common not only in large industrial or building consumers, but also at the home user that is increasingly in contact with intelligent applications through high-tech devices with high computing and communication capabilities such as smart-phones and tablets. Figure 7.1 depicts our outlook of one of possible home user iEMS.

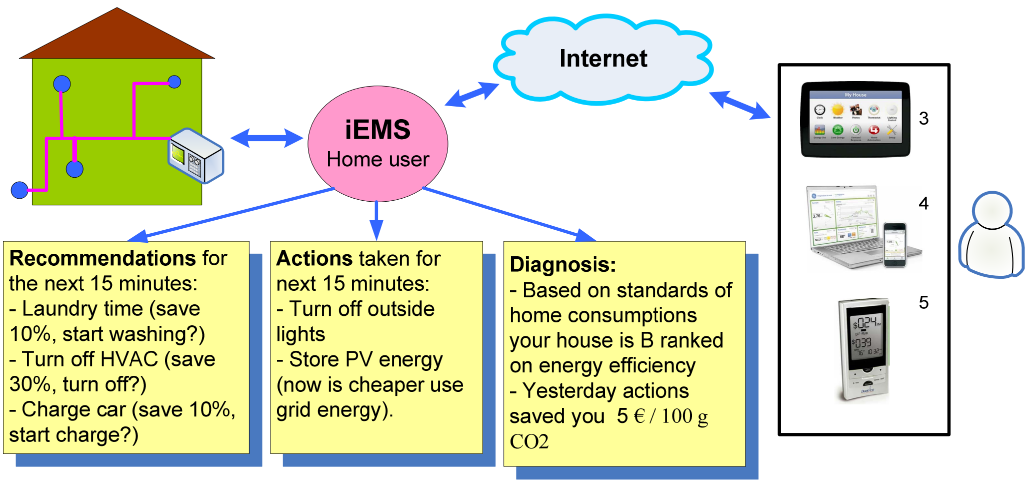


Figure 7.1. Home user iEMS[3 4 5].

The load modeling and forecasting systems (LMFS) with autonomy, adaptability and accuracy, as the one depicted in the thesis, is the first step for applications aimed towards the goal of improvement of energy efficiency for home users. Furthermore, they constitute the basis for

[3] Tablet photo source: http://www.mydigitallife.info
[4] Laptop and mobile phone source: http://blog.ce.org/index.php
[5] Home display photo source: http://buildaroo.com/es/

future applications, which use their forecast as fundamental information to carry out their purpose of supervision, control and diagnosis for energy use optimization.

Therefore, the main research lines in this direction are:

- LMFS for home users: challenges and novel approaches.
- Smart Home Energy Management, with full integration with home equipments and meters, sensors and protocols of communications for home user's iEMS.
- New developments on upper level applications and strategies, mainly those that can save energy and operation and maintenance costs in an automatic way and facilitate and increase the use of renewable energies.

They are expected to be a new field of research and development that can help us to achieve the goal of energy efficiency and sustainability, both objectives of current policies of worldwide governments and communities, as the European Union.

iEMS, upper level applications based on LMFS forecast

Continuing the research developed in the thesis, the next step in order to fulfil iEMS requirements is to study, analyze and develop upper level applications that can use the models and forecasts from LMFS for energy consumption savings, energy costs and/or greenhouse gases reductions. For example, they can maximize the use of energy from renewable sources, follow a specific load profile of consumption to avoid power peaks, detect energy waste, etc. Applications are supposed to generate direct savings, preferably in an automatic way or by means of suggestions as a human expert does.

In the thesis has been considered mainly two types of upper level applications. One of them is aimed for **load and energy source control**, which can decide or support the making decision process of when and how a given load starts to work and/or, in the case of micro-generation, from which energy source it is more efficient to use the demanded energy. Therefore it is possible to shape the load profile of energy consumption to a desired load profile, where, for example, power consumption peaks coincide with generation power peaks from renewable sources.

A second type of upper level applications that can exploit LMFS' models and forecasts are those that follow real time consumptions looking for over peaks or anomalous consumptions. They can use historic models of process or places as healthy reference to detect the mentioned anomalies. Furthermore, they can use extra information from user databases as scheduled production, weather variables, fail and maintenance reports, etc. in order to improve its diagnosis. Of course, this diagnosis could be linked to specialized maintenance software, to improve the detection of equipment faults through energy consumption (or production) analysis. Figure 7.2 sketches the former idea.

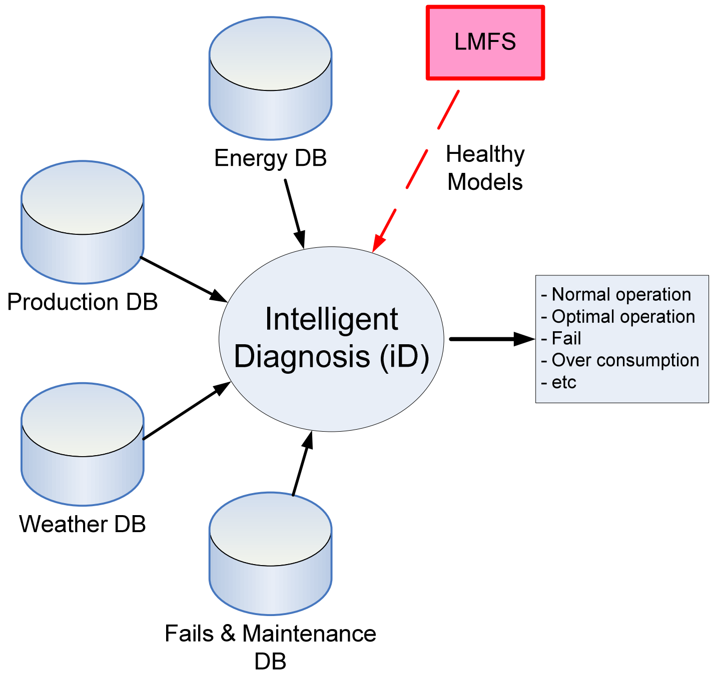


Figure 7.2. Intelligent diagnostic system.

Therefore, **optimal timing for consumptions and local generators to avoid energy over peaks on the mains** and **energy consumption diagnosis** for iEMS are other two interesting research lines needed to give iEMS its capacity of autonomous energy saver.

Modeling and forecasting of generators

The features of above applications are evidence of the need of modeling and forecasting not only of consumers but also of generators. On the user side, mainly for large users, are often found micro-generation systems such as cogeneration, micro-wind, photovoltaic or geothermal. Therefore, the **modeling and forecasting of micro-generation on the user side** is another field of research and development growing in interest and importance in energy management area right now and even more in the next years.

7.2.2. Future work for new approaches of LMFS on the user side

During the development of the thesis some CI tools, which promised a good performance for the pursuit of autonomy, adaptability and accuracy in LMFS on the user side, remained without being studied and tested. Some of them are:

- Genetic programming for getting an ***autonomous modeling system designer (AMSD)***: The main idea behind this approach is to use genetic programming together with adaptive networks, signal processing functions and other relevant operators for pre-processing, for getting new self-designed structures, which can be trained and evaluated with a generic training algorithm as the proposed in the thesis, the ETA. The ambition of this idea is to automate the process of design new adaptive networks using CI algorithms: the AMSD could start from and typical known adaptive network, as for example an ANN or ANFIS, and finish with a new hybrid adaptive network completely different and with improved features.

- ***ANFIS configuration by means of pruning algorithm:*** The technique consists of over sizing an ANFIS, using all the possible input candidates, a considerably high number of membership functions by input and training the obtained ANFIS. Then, to use a pruning algorithm to reduce the size of the ANFIS. This algorithm can be based on evolutionary algorithm as GA or particle swarm optimization (PSO) and binary codification of ANFIS' links and weights. The optimization algorithm would find the right configuration code or genome that minimizes the fitness function, which could be the RMSE of the training and checking data, using a multi-objective optimization process.
- ***Study, analysis and development of new methods for dominant patter detection and selection:*** As stated on this work, Hilbert-Huang transform is not the only efficient way to detect and select the dominant patterns of load profiles of energy consumption, but also a powerful tool to automate forecasting modeling. However, other strategies based on wavelet transforms or statistical functions, as correlation, are interesting approaches that could be also studied and analyzed in order to get lower resolution patterns, which are useful for accuracy improvement in a LMFS. Other topologies instead of parallel one also can be considered.

8. Thesis dissemination

This work has played an important role both in academy and applied research. The dissemination in the first of these fields is reflected by the international journal and conference publications; and by collaboration in technologic transfer projects for the second one. Both types of contributions are listed in this chapter.

CONTENTS:

8.1. Related journal and conference publications

8.1.1. Journals

January/2013 J. J. Cardenas, J. L. Romeral and A. Garcia "User-side Load Modeling and Forecasting System based on Dominant Patterns," Journal *Energy and Buildings*, Impact Factor: 2.386. Under review.

April/2012 J. J. Cárdenas, L. Romeral, A. Garcia, and F. Andrade, "Load forecasting framework of electricity consumptions for an Intelligent Energy Management System in the user-side," Journal *Expert Systems with Applications*, Impact Factor: 2.203. vol. 39, pp. 5557-5565, April 2012.

July/2010 J. J. Cardenas, A. Garcia, J. L. Romeral, and J. Urresty, "A Multi-Objective GA to Demand-side Management in an Automated Warehouse," *International Journal of Electrical Energy Systems*, vol. 2, Jul-Dec 2010.

8.1.2. Conferences

Sept./2012 J. J. Cardenas, F. Giacometto, A. Garcia, and J. L. Romeral, "STLF in the User-Side for an iEMS based on Evolutionary training of Adaptive Networks," in *ETFA2012, the IEEE 2012 Conference on Emerging Technologies and Factory Automation*, Kraków, Poland., 2012.

Sept./2011 J. J. Cardenas, A. Garcia, J. L. Romeral, and K. Kampouropoulos, "Evolutive ANFIS training for energy load profile forecast for an IEMS in an automated factory," in *Emerging Technologies & Factory Automation (ETFA), 2011 IEEE 16th Conference on*, 2011, pp. 1-8.

October/2010 J. J. Cardenas, J. L. Romeral, and A. Garcia, "Energy Management Systems by Means of Computational Intelligence Algorithms," in *2nd Barcelona Forum on Ph.D. Research in Communications, Electronics and Signal Processing*, Universitat Politècnica de Catalunya, 2010.

October/2009 J. J. Cardenas, A. Garcia, J. L. Romeral, and F. Andrade, "A genetic algorithm approach to optimization of power peaks in an automated warehouse," in *Industrial Electronics, 2009. IECON '09. 35th Annual Conference of IEEE*, 2009, pp. 3297-3302.

8.1.3. Collaborative work

May/2013 K. Kabouropoulos, J. J. Cardenas, F. Giacometto, and J. L. Romeral, "An Energy Prediction Method using Adaptive Neuro-Fuzzy Inference System and Genetic Algorithms," The 22nd IEEE International Symposium on Industrial Electronics, ISIE 2013, Taipei, Taiwan on May 28-31, 2013.

October/2012 F. Giacometto, J. J. Cardenas, K. Kampouropoulos, and J. L. Romeral, "Load forecasting in the user side using wavelet-ANFIS," in IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society, 2012, pp. 1049-1054.

July/2012 K. Kampouropoulos, F. Andrade, J. J. Cárdenas, J.L. Romerar, "A Methodology for Energy Prediction and Optimization of a System based on the Energy Hub Concept using Particle Swarms," The Annual Seminar on Automation, Industrial Electronics and Instrumentation, Guimaraes, Portugal, July 2012

January/2012 F. Andrade, J. J. Cardenas, L. Romeral, and J. Cusido, "Modeling and studying of power flow in a parking lot with plug-in vehicles and the impact in the public utility," in *Innovative Smart Grid Technologies (ISGT), 2012 IEEE PES*, 2012, pp. 1-7.

June/2011 M. Delgado, A. Garcia, J. A. Ortega, J. J. Cardenas, and L. Romeral, "Multidimensional intelligent diagnosis system based on Support Vector Machine Classifier," in *Industrial Electronics (ISIE), 2011 IEEE International Symposium on*, 2011, pp. 2124-2131.

October/2009 V. M. Sala, L. Romeral, J. J. Cardenas, and G. Ruiz, "Study and evaluation of DC-Link perturbations models for three-level DCI-NPC power amplifiers," in *Industrial Electronics, 2009. IECON '09. 35th Annual Conference of IEEE*, 2009, pp. 895-900.

Sept./2009 J. Urresty, J. Riba, J. Ortega, and J. Cardenas, "Stator short circuits detection in PMSM by means of Zhao-Atlas-Marks distribution and energy calculation," in *Power Electronics and Applications, 2009. EPE '09. 13th European Conference on*, 2009, pp. 1-9.

8.2. Technology transfer and innovation projects

8.2.1. European projects

Name of the project: Increase of automotive car industry competitiveness through an integral and artificial intelligence driven energy management system.

Funding body: Commission of European Communities

Code from funding body: FP7-288102-EUROENERGEST

Start date: 01/10/2011, 3 years

Role in the project: Researcher.

Task description: Study and analysis of energy consumptions in a car factory. Definition of critic energies and variables for modeling and forecasting. Definition of load modeling and forecasting parameters according to control and optimization needs. Design, implementation and tests of the load modeling and forecasting system.

8.2.2. National projects

Name of the project: Estudi i execució de projectes d' r+d+i en l'àmbit de l'eficiencia energètica i les renovables.

Head researcher: Jose Luis Romeral Martinez

Funding body: FUNDACIÓ CTM CENTRE TECNOLÒGIC

Start date: 01/04/2009 , 3 years - 1 month - 9 days

Role in the project: Researcher

Task description: Study and analysis of energy consumptions on the user side oriented to data driven modeling. Conceptualization and design of predictive and energy efficiency applications based on consumption forecasts.

Name of the project:	**C***oncepto, diseño y desarrollo de un sistema inteligente de gestión de energía en plantas industriales, proyecto colaborativo SEAT-PR*
Head researcher:	Jose Luis Romeral Martinez
Funding body:	PROMAUT, S.L.
Start date:	12/02/2009 , 2 years - 1 day
Role in the project:	Researcher
Task description:	Software development related with automatic load modeling and forecasting of energy consumptions in a car factory. Besides, optimization and supervision applications were implemented.

Bibliography

[1] I. E. Agency, *World Energy Outlook 2011*, 2011.

[2] J. J. Cardenas, A. Garcia, J. L. Romeral, and K. Kampouropoulos, "Evolutive ANFIS training for energy load profile forecast for an IEMS in an automated factory," in *Emerging Technologies & Factory Automation (ETFA), 2011 IEEE 16th Conference on*, 2011, pp. 1-8.

[3] J. J. Cardenas, F. Giacometto, A. Garcia, and J. L. Romeral, "STLF in the User-Side for an iEMS based on Evolutionary training of Adaptive Networks," in *ETFA2012, the IEEE 2012 Conference on Emerging Technologies and Factory Automation, Kraków, Poland.*, 2012.

[4] J. J. Cárdenas, L. Romeral, A. Garcia, and F. Andrade, "Load forecasting framework of electricity consumptions for an Intelligent Energy Management System in the user-side," *Expert Systems with Applications,* vol. 39, pp. 5557-5565, April 2012.

[5] J. J. Cardenas, J. L. Romeral, A. Garcia, and F. Giacometto, "User-side Load Forecasting System based on Dominant Patterns," *Expert Systems with Applications,* Under review.

[6] J. Sheppard and A. Tisot, "Industrial Energy Management: Doing More with Less," *IETC - Industrial Energy Technology Conference,* 2006.

[7] J. C. Van Gorp, "Maximizing energy savings with enterprise energy management systems," in *Pulp and Paper Industry Technical Conference, 2004. Conference Record of the 2004 Annual*, 2004, pp. 175-181.

[8] H. Doukas, K. D. Patlitzianas, K. Iatropoulos, and J. Psarras, "Intelligent building energy management system using rule sets," *Building and Environment,* vol. 42, pp. 3562-3569, 2007.

[9] R. Alcala, J. Casillas, O. Cordon, A. Gonzalez, and F. Herrera, "A genetic rule weighting and selection process for fuzzy control of heating, ventilating and air conditioning systems," *Engineering Applications of Artificial Intelligence,* vol. 18, pp. 279-296, 2005.

[10] B. Swords, E. Coyle, and B. Norton, "An enterprise energy-information system," *Applied Energy,* vol. 85, pp. 61-69, 2008.

[11] B. Forth and T. Tobin, "Right power, right price [enterprise energy management systems]," *Computer Applications in Power, IEEE,* vol. 15, pp. 22-27, 2002.

[12] D. Kolokotsa, K. Niachou, V. Geros, K. Kalaitzakis, G. S. Stavrakakis, and M. Santamouris, "Implementation of an integrated indoor environment and energy management system," *Energy and Buildings,* vol. 37, pp. 93-99, 2005.

[13] L. Pedersen, J. Stang, and R. Ulseth, "Load prediction method for heat and electricity demand in buildings for the purpose of planning for mixed energy distribution systems," *Energy and Buildings,* vol. 40, pp. 1124-1134, 2008.

[14] J. C. Van Gorp, "Enterprising energy management," *Power and Energy Magazine, IEEE,* vol. 2, pp. 59-63, 2004.

[15] J. J. Bann, G. D. Irisarri, S. Mokhtari, D. S. A. K. D. S. Kirschen, and B. N. A. M. B. N. Miller, "Integrating AI applications in an energy management system," *IEEE Expert [see also IEEE Intelligent Systems and Their Applications],* vol. 12, pp. 53-59, 1997.

[16] J. J. Cardenas, J. L. Romeral, and A. Garcia, "Energy Management Systems by Means of Computational Intelligence Algorithms," in *2nd Barcelona Forum on Ph.D. Research in Communications, Electronics and Signal Processing* Barcelona: Universitat Politècnica de Catalunya, 2010.

[17] J. J. Cardenas, A. Garcia, J. L. Romeral, and J. Urresty, "A Multi-Objective GA to Demand-side Management in an Automated Warehouse," *International Journal of Electrical Energy Systems,* vol. 2, Jul-Dec 2010.

[18] J. J. Cardenas, A. Garcia, J. L. Romeral, and F. Andrade, "A genetic algorithm approach to optimization of power peaks in an automated warehouse," in *Industrial Electronics, 2009. IECON '09. 35th Annual Conference of IEEE*, 2009, pp. 3297-3302.

[19] B. Ramanathan and V. Vittal, "A Framework for Evaluation of Advanced Direct Load Control With Minimum Disruption," *Power Systems, IEEE Transactions on,* vol. 23, pp. 1681-1688, 2008.

[20] Y. K. Penya., "Last-generation Applied Artificial Intelligence for Energy Management in Building Automation," in *In the Proceedings of the 5th IFAC International Conference on Fieldbus Systems and their Applications, FeT 2003* Aveiro Portugal, 2003.

[21] A. Khotanzad, M. H. Davis, A. Abaye, and D. J. A. M. D. J. Maratukulam, "An artificial neural network hourly temperature forecaster with applications in load forecasting," *Power Systems, IEEE Transactions on,* vol. 11, pp. 870-876, 1996.

[22] A. Khotanzad, E. Zhou, and H. Elragal, "A Neuro-Fuzzy Approach to Short-Term Load Forecasting in a Price-Sensitive Environment," *Power Engineering Review, IEEE,* vol. 22, pp. 55-55, 2002.

[23] A. J. Conejo, J. Contreras, R. EspÃnola, and M. A. Plazas, "Forecasting electricity prices for a day-ahead pool-based electric energy market," *International Journal of Forecasting,* vol. 21, pp. 435-462, 2005/9// 2005.

[24] P. Lauret, E. Fock, R. N. Randrianarivony, and J.-F. Manicom-Ramsamy, "Bayesian neural network approach to short time load forecasting," *Energy Conversion and Management,* vol. 49, pp. 1156-1166, 2008.

[25] O. A. S. Carpinteiro, I. Lima, E. M. Moreira, C. A. M. Pinheiro, E. Seraphim, and J. V. L. Pinto, "A hierarchical hybrid neural model with time integrators in long-term load forecasting," *Neural Computing & Applications,* vol. 18, pp. 1057-1063, Nov 2009.

[26] Q. Wu, "A hybrid-forecasting model based on Gaussian support vector machine and chaotic particle swarm optimization," *Expert Systems with Applications,* vol. 37, pp. 2388-2394, Mar 2010.

[27] D. J. Pedregal and J. R. Trapero, "Mid-term hourly electricity forecasting based on a multi-rate approach," *Energy Conversion and Management,* vol. 51, pp. 105-111, Jan 2010.

[28] C. M. Lee and C. N. Ko, "Time series prediction using RBF neural networks with a nonlinear time-varying evolution PSO algorithm," *Neurocomputing,* vol. 73, pp. 449-460, Dec 2009.

[29] M. Afshin and A. Sadeghian, "PCA-based Least Squares Support Vector Machines in Week-Ahead Load Forecasting," in *Industrial & Commercial Power Systems Technical Conference, 2007. ICPS 2007. IEEE/IAS*, 2007, pp. 1-6.

[30] W. C. Hong, "Hybrid evolutionary algorithms in a SVR-based electric load forecasting model," *International Journal of Electrical Power & Energy Systems,* vol. 31, pp. 409-417, Sep 2009.

[31] Z. Xiao, S. J. Ye, B. Zhong, and C. X. Sun, "BP neural network with rough set for short term load forecasting," *Expert Systems with Applications,* vol. 36, pp. 273-279, Jan 2009.

[32] Q. Wu, "Power load forecasts based on hybrid PSO with Gaussian and adaptive mutation and Wv-SVM," *Expert Systems with Applications,* vol. 37, pp. 194-201, Jan 2010.

[33] F. Shu, W. Yuan-Kang, L. Wei-Jen, and L. Ching-Yin, "Comparative study on load forecasting technologies for different geographical distributed loads," in *Power and Energy Society General Meeting, 2011 IEEE*, 2011, pp. 1-8.

[34] H.-C. Huang, R.-C. Hwang, and J.-G. Hsieh, "A new artificial intelligent peak power load forecaster based on non-fixed neural networks," *International Journal of Electrical Power & Energy Systems,* vol. 24, pp. 245-250, 2002.

[35] P. E. McSharry, S. Bouwman, and G. Bloemhof, "Probabilistic forecasts of the magnitude and timing of peak electricity demand," *Power Systems, IEEE Transactions on,* vol. 20, pp. 1166-1172, 2005.

[36] F. J. Nogales, J. Contreras, A. J. Conejo, and R. Espinola, "Forecasting next-day electricity prices by time series models," *Power Systems, IEEE Transactions on,* vol. 17, pp. 342-348, 2002.

[37] J. Contreras, R. Espinola, F. J. Nogales, and A. J. Conejo, "ARIMA models to predict next-day electricity prices," *Power Systems, IEEE Transactions on,* vol. 18, pp. 1014-1020, 2003.

[38] A. J. Conejo, M. A. Plazas, R. Espinola, and A. B. Molina, "Day-ahead electricity price forecasting using the wavelet transform and ARIMA models," *Power Systems, IEEE Transactions on,* vol. 20, pp. 1035-1042, 2005.

[39] R. Gareta, L. M. Romeo, and A. Gil, "Forecasting of electricity prices with neural networks," *Energy Conversion and Management,* vol. 47, pp. 1770-1778, 2006.

[40] R. Bo and F. X. Li, "Probabilistic LMP Forecasting Considering Load Uncertainty," *Ieee Transactions on Power Systems,* vol. 24, pp. 1279-1289, Aug 2009.

[41] F. Lira, C. Munoz, F. Nunez, and A. Cipriano, "Short-term forecasting of electricity prices in the Colombian electricity market," *Iet Generation Transmission & Distribution,* vol. 3, pp. 980-986, Nov 2009.

[42] Catala, x, J. P. S. o, H. M. I. Pousinho, and V. M. F. Mendes, "Hybrid Wavelet-PSO-ANFIS Approach for Short-Term Electricity Prices Forecasting," *Power Systems, IEEE Transactions on,* vol. 26, pp. 137-144, 2011.

[43] D. L. Loveday, G. S. Virk, J. Y. M. Cheung, and D. Azzi, "Intelligence in buildings: the potential of advanced modelling," *Automation in Construction,* vol. 6, pp. 447-461, 1997.

[44] B. Dong, C. Cao, and S. E. Lee, "Applying support vector machines to predict building energy consumption in tropical region," *Energy and Buildings,* vol. 37, pp. 545-553, May 2005.

[45] P. A. Gonzalez and J. M. Zamarreno, "Prediction of hourly energy consumption in buildings based on a feedback artificial neural network," *Energy and Buildings,* vol. 37, pp. 595-601, 2005.

[46] Xiaoli Li, Chris P. Bowers, and T. Schnier, "Classification of Energy Consumption in Buildings with Outlier Detection," *IEEE Transactions on Industrial Electronics: Accepted for future publication,* 2009.

[47] A. Kusiak, M. Y. Li, and Z. J. Zhang, "A data-driven approach for steam load prediction in buildings," *Applied Energy,* vol. 87, pp. 925-933, Mar 2010.

[48] ASHRAE, "American Society of Heating, Refrigerating and Air-Conditioning Engineers," founded in 1894.

[49] L. Pedersen, "Use of different methodologies for thermal load and energy estimations in buildings including meteorological and sociological input parameters," *Renewable and Sustainable Energy Reviews,* vol. 11, pp. 998-1007, 2007.

[50] D. De Silva, Y. Xinghuo, D. Alahakoon, and G. Holmes, "A Data Mining Framework for Electricity Consumption Analysis From Meter Data," *Industrial Informatics, IEEE Transactions on,* vol. 7, pp. 399-407, 2011.

[51] H. C. W. Lau, E. N. M. Cheng, C. K. M. Lee, and G. T. S. Ho, "A fuzzy logic approach to forecast energy consumption change in a manufacturing system," *Expert Systems with Applications,* vol. 34, pp. 1813-1824, 2008.

[52] M. Hanmandlu and B. K. Chauhan, "Load Forecasting Using Hybrid Models," *Power Systems, IEEE Transactions on,* vol. 26, pp. 20-29, 2011.

[53] S. Sargunaraj, D. P. S. Gupta, and S. Devi, "Short-term load forecasting for demand side management," *Generation, Transmission and Distribution, IEE Proceedings-,* vol. 144, pp. 68-74, 1997.

[54] Q. Wu, "Hybrid forecasting model based on support vector machine and particle swarm optimization with adaptive and Cauchy mutation," *Expert Systems with Applications,* vol. 38, pp. 9070-9075, 2011.

[55] J. S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *Systems, Man and Cybernetics, IEEE Transactions on,* vol. 23, pp. 665-685, 1993.

[56] S. O. Haykin, *Neural Networks and Learning Machines*, 3 ed. vol. 10: Prentice Hall, 2009.

[57] F. Giacometto, J. J. Cardenas, K. Kampouropoulos, and J. L. Romeral, "Load Forecasting in the User Side Using Wavelet-ANFIS," in *Industrial Electronics, 2012. IECON '12. 38th Annual Conference of IEEE, Montréal, Canada*, 2012.

[58] A. Khotanzad, R. Afkhami-Rohani, and D. Maratukulam, "ANNSTLF-Artificial Neural Network Short-Term Load Forecaster generation three," *Power Systems, IEEE Transactions on,* vol. 13, pp. 1413-1422, 1998.

[59] M. Khashei and M. Bijari, "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting," *Applied Soft Computing,* vol. 11, pp. 2664-2675, 2011.

[60] A. O. h. w. m. c. Parallel Computing Toolbox™.

[61] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*: Addison-Wesley Longman, Inc., 1989.

[62] S. Fan and L. Chen, "Short-term load forecasting based on an adaptive hybrid method," *Power Systems, IEEE Transactions on,* vol. 21, pp. 392-401, 2006.

[63] D. X. Niu, Y. L. Wang, and D. D. Wu, "Power load forecasting using support vector machine and ant colony optimization," *Expert Systems with Applications,* vol. 37, pp. 2531-2539, Mar 2010.

[64] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences,* vol. 454, pp. 903-995, March 8, 1998.

[65] E. H. Norden, "INTRODUCTION TO THE HILBERT-HUANG TRANSFORM AND ITS RELATED MATHEMATICAL PROBLEMS," in *Hilbert-Huang Transform And Its Applications*, pp. 1-26.

[66] M. Chung and H.-C. Park, "Building energy demand patterns for department stores in Korea," *Applied Energy,* vol. 90, pp. 241-249, 2012.

APPENDICES

A. Main load forecasting algorithms

In chapter 2 we reviewed the state of the art in LMF related with energy management. One of the main conclusions was the importance of computational intelligence algorithms for LMF. Therefore, in this section we are going to outline two of the most important adaptive network structures found in the current literature of LMF in energy related areas. These are neural networks (NNs) and adaptive-network-based fuzzy inference systems (ANFISs).

ANFIS and NN are excellent algorithms for LMF because of its potential to solve problems of time series modeling. Indeed, the availability of historical energy data on the iEMS databases and the fact that ANFIS and NN are data driven approaches capable of performing a non-linear mapping between sets of input and output variables make these modeling tools very attractive. For deeper information on ANFIS and NN you can check (Jang 1993) and (Haykin 2009) respectively.

A.1. ANFIS

ANFIS is an Adaptative network based on Takagi-Sugeno fuzzy system. A fuzzy system is constructed of input and output variables, membership functions, fuzzy rules and inference method. In this case, the inputs are the energy drivers, which are thought to affect the consumption profile such as daily production, outdoor temperature, day of the week, etc. The membership functions are the functions that define the fuzzy sets. They are used to compute the degree of membership for a determinate value of its inputs. For instance, to determinate the degree of how high or low the production is. The fuzzy rules are if-then rules that define how the output must be for a specific value of membership of its inputs. Thus, by means of the inference method, the output values are obtained when the input values are known. In general, the fuzzy systems have different kind of inference methods. However, the ANFIS is based on a particular type of fuzzy system with Takagi-Sugeno rules as inference method. Figure A.1 shows ANFIS architecture of two inputs, four if-then rules and one output, *z*. *x* and *y* are the selected energy drivers that could be, for example, scheduled production and forecasted temperature.

ANFIS structure

The Figure A.1 shows the architecture of an ANFIS with two inputs, four rules and one output. This structure has a maximum of four rules and they are depicted in the equation (A.1)

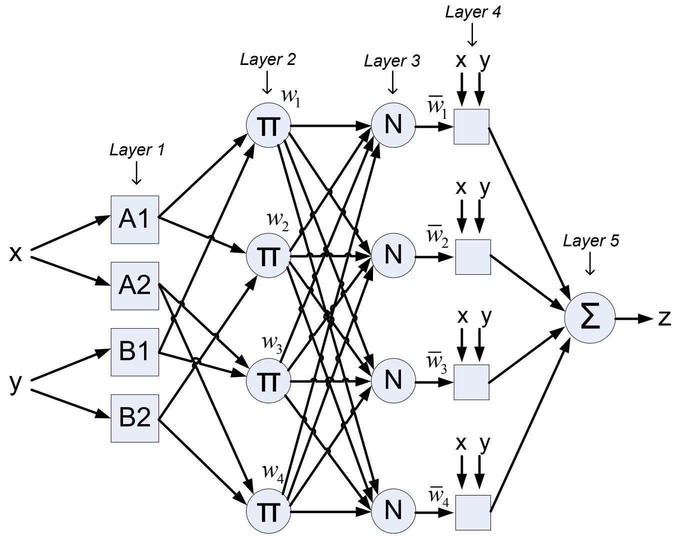


Figure A.1. ANFIS architecture: two inputs, four if-then rules and one output.

$$if\ x\ \in\ A_1\ \wedge y\ \in\ B_1\ \Rightarrow z_1 = p_1x + q_1y + r_1$$

$$if\ x\ \in\ A_1\ \wedge y\ \in\ B_2\ \Rightarrow z_2 = p_2x + q_2y + r_2$$

$$if\ x\ \in\ A_2\ \wedge y\ \in\ B_1\ \Rightarrow z_3 = p_3x + q_3y + r_3$$

$$if\ x\ \in\ A_2\ \wedge y\ \in\ B_2\ \Rightarrow z_4 = p_4x + q_4y + r_4$$

(A.1)

A linguistic interpretation of the rules applied to energy modeling problem could be:

$$if\ production\ (x) is\ low (A_1)\ \&\ time\ (y) is\ morning\ (B_1)$$

$$\Rightarrow consumption_1 = p_1x + q_1y + r_1$$

$$if\ production\ (x) is\ low (A_1)\ \&\ time\ (y) is\ night\ (B_2)$$

$$\Rightarrow consumption_2 = p_2x + q_2y + r_2$$

$$if\ production\ (x) is\ high\ (A_2)\ \&\ time\ (y) is\ morning\ (B_1)$$

$$\Rightarrow consumption_3 = p_3x + q_3y + r_3$$

$$if\ production\ (x) is\ high\ (A_2)\ \&\ time\ (y) is\ night\ (B_2)$$

$$\Rightarrow consumption_4 = p_4x + q_4y + r_4$$

The first part in (A.1) is related to antecedents and the second part to consequents. The ANFIS structure executes these rules and calculates the output through five layers (Figure A.1). In the layer 1, the membership values (or compatibility measures) are calculated by means of the membership functions that are identified by $\mu_{A_i}(x)$ and $\mu_{B_i}(y)$, *i=1, 2*. This step is called fuzzification.

In the second layer, the weight of each rule has to be computed by means of a fuzzy AND operation. The equation (A.2) shows how this is done. ω_j is the weight that belongs to the rule *jth*. This is the output of the layer 2 in Figure A.1.

$$\omega_j = \mu_{A_i}(x) \cdot \mu_{B_k}(y),\ \ j = 1, 2, 3, 4.\ i = 1, 2.\ k = 1, 2. \tag{A.2}$$

Next, in the layer 3, the strength, $\bar{\omega}_j$, is computed. It is the ratio of the *jth* weight to the sum of the all weights. (A.3) shows the related equation.

$$\bar{\omega}_j = \frac{\omega_j}{\sum_{i=1}^{N} \omega_i},\quad N = 4 \tag{A.3}$$

In the layer 4, $\bar{\omega}_j$ multiplies the related output function (linear equations of the consequent part in (A.1) , z_j). This is:

$$z_j \cdot \bar{\omega}_j = \bar{\omega}_j \cdot \left(p_j x + q_j y + r_j\right) \tag{A.4}$$

Finally, in the layer 5, the overall output is obtained, which is the sum of the former outputs, i.e.

$$z = \sum_j z_j \cdot \bar{\omega}_j \tag{A.5}$$

ANFIS training algorithm

Previously to the training process the rules have to be defined. Depending on the method to calculate the number of rules, it can be a third parameter to be configured. In an application of data-driven modeling, where none human knowledge is used to define the rules, the number of rules (r) are defined in function of the number of inputs (*n*) and number of membership functions by input (*m*). The maximum number of possible rules is calculated using (A.6). Therefore, this parameter does not change in the training process (Jang 1993).

$$r = m^n \tag{A.6}$$

Once the rules are defined the training process can be carry out. There are two kinds of parameters to find by training process in an ANFIS. The first ones to be tuned are the membership function parameters of each input, which are called *antecedents* (layer 1, Figure A.1, left side in (A.1)). Theses depend on the kind of selected membership function. The second parameters to be tuned are the ***consequents***, which are the coefficients of the output polynomials (layer 4, Figure A.1, right side in (A.1)).

The most common training algorithm is the hybrid algorithm, defined in (Jang 1993). The antecedents and consequents are obtained from historic data by means of ***back propagation (BP)*** and ***least squares (LS)***, respectively. This algorithm is carried out in two steps in each epoch (Table A.1). First, a forward pass and second, a backward pass. Once all the parameters are initialized, in the forward pass, with the antecedents fixed, input data and functional signals go forward to calculate each layer node output and the LS algorithm is used to compute the consequents. After identifying consequents, the functional signals keep going forward until the error measure is calculated. In the backward pass, the error rates propagate from the output end toward the input end, and the parameters in antecedent part are updated by the BP method.

Table A.1. Hybrid algorithm for ANFIS training.

ANFIS Parameter	Method	Forward Pass	Backward Pass
MF[b] parameters (premise part)	Human knowledge / BP algorithm	Fixed	Gradient Descent
Rules	Human knowledge / BP-LS algorithm	LS Estimate	Gradient Descent
Coefficients (consequent part)	LS algorithm	LS Estimate	Fixed

[a] MOGA is the acronym of Multi Objective Genetic Algorithm.

[b]MF is the acronym of membership function.

The function to be minimized in both steps is:

$$RMSE = \sqrt{\frac{1}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}}(d_i - o_i)^2} \tag{A.7}$$

Where d_i is the desired output and o_i is the ANFIS output for the *i-th* sample from training data. N is the number of training samples.

A.2. NN

Unlike ANFIS, NN have had more popularity in the research field and in real applications, including the field of load forecasting. Its theory is well known and for that reason only the main characteristics of the used NN and its parallelism with the ANFIS structure are here drawn.

As ANFIS does, NN do not need of explicit knowledge of the relationship between inputs and output to establish a map between them. Only historical data is needed. Once the NN structure is defined (kind of NN, inputs, outputs, hidden layers, neurons, etc.) training algorithms are used to extract the relevant information from the historical data and set up the model (weights of nodes).

There are different algorithms to train the NN. Among the most popular the ***error-back propagation (EBP)*** algorithm (Haykin 2009) can be found. This is also one of the algorithms used to train the full ANFIS or the antecedent part in hybrid algorithms. However this algorithm is slower than more advanced others and even sometimes produces no adequate results. For that reason, the algorithm here used was the ***Levenberg-Marquardt (LM)*** (Haykin 2009) algorithm. It is faster than EBP and able to deal with big NN (more than 500 weights in the network) Also the results are close-to-optimum NN (Wilamowski 2009).

The kind of NN chosen to modeling the LP was one of the most popular, the so called ***multi-layer perceptrón (MLP)***. The MLP consists of an input layer, one or several hidden layers and an output layer. In this case a hidden layer with only ten neurons was enough to model the load profile with a good capacity of interpolation. In Figure A.2 the general structure of the MLP can be seen (Wilamowski 2009).This architecture is quite similar to ANFIS architecture (see Figure A.1). In fact, the results here obtained with both structure were quite near. This will be showed in the results section.

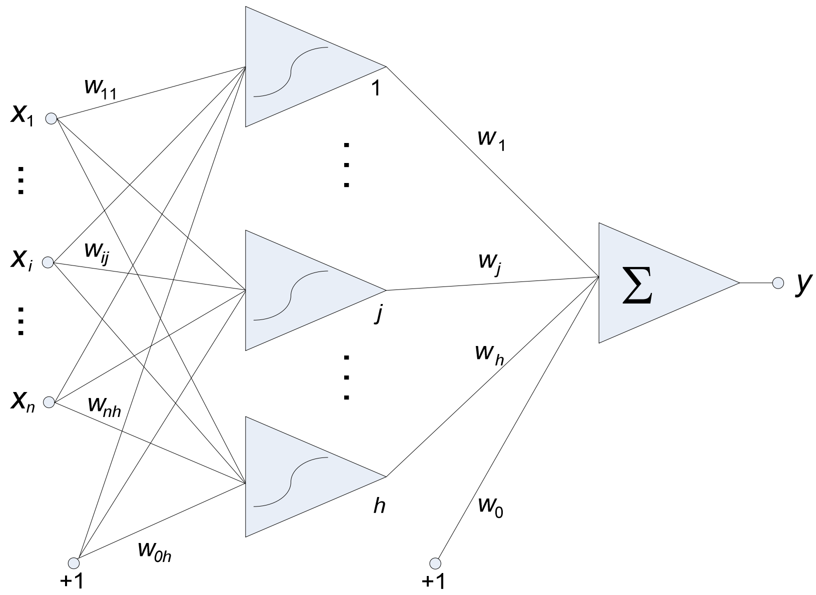


Figure A.2. The MLP-type n inputs, h hidden neurons and 1 output. Architecture n-h-1.

The input layer joins the model's inputs vector x while the output layer yields the model's output vector y. The hidden layer is characterized by several non-linear neurons. The non-linear-function (also called activation function) is usually the tangent hyperbolic function (A.8).

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{A.8}$$

This structure permit to NN build up a non-linear parameterized mapping from an input x to an output y given by the following relationship:

$$y = y(x\colon w) = \sum_{j=0}^{h}\left[w_j f\left(\sum_{i=0}^{n} w_{ji} \cdot x_i\right)\right] \tag{A.9}$$

The parameters of the NN model are given by the so called weights and biases that connect the layers between them. The NN parameters, denoted by the parameter vector w, govern the non-linear mapping.

The NN parameters *w* are estimated during a phase called the training or learning phase. During this phase, the NN is training using a dataset (called training set) of N input and output examples, pairs of the form $D = \{x_i, t_i\}_{i=1}^{N}$. The vector *x* contains samples of the input variables and *t* is the target signal or variable and it is the real measurement of energy consumption. This phase consists in adjusting the weights, *w*, in order to minimize the error function E_D, which is defined in (A.10).

$$E_D(w) = \frac{1}{2}\sum_{i=1}^{N}(y_i - t_i)^2$$

(A.10)

The next phase after the training is the generalization phase. It consists in to test the capacity of interpolation of the NN. This is done by giving pairs of data to the NN that were not used in the training phase. These data come from the data set called checking or test set. To evaluate the performance of the NN, RMSE and MAPE are used as described in Section 2.

B. Signal processing and statistical functions

In this Appendix, we present some functions and algorithms used to support the modeling process. They are mainly aimed for pre-processing. The idea of using these pre-processing tools is highlight the relationships between output and inputs to the LMFS. Besides, they can be used for outlier detection, input selection, noise reduction, etc.

Hilbert-Huang Transform is a useful tool in signal processing applications, less applied in LMFS but with high potential as it has been demonstrated in this thesis (Chapter 6).

B.1. Hilbert-Huang Transform

In order to indentify in an automatic way these visual and no visual patterns we propose here a Hilbert Huang transform (HHT) analysis (Huang, Shen et al. 1998). Among the main advantages of HHT analysis the following can be stated: it permits a better physical analysis of the signal under test; it is appropriate for nonlinear and nonstationary signals, whose main frequencies depend on time, and permit a time-frequency-energy analysis simultaneously, which is very useful for feature extraction. This last characteristic is the one we want to take advantage to track the main mentioned DPs. As it has been pointed, the LP on the user-side is subjected to random behaviors and sudden changes, making the LP signal to be nonstationary and the relationship with energy-drivers mainly become nonlinear. Time frequency analysis enable us to visualize the effect on frequency of the regular and no regular changes on time, i.e. weekend and holidays full stops and maintenance partial stops.

The procedure to implement the HHT is composed of two parts: first, the decomposition of the signal in its intrinsic mode functions (IMFs), which is called Empirical Mode Decomposition (EMD); and second, the Hilbert spectral analysis (HSA) of each empirical mode.

In the EMD process, the number of local maximums and minimums of each IMF must coincide with the number of zero crossings, with a maximum of difference of one. Furthermore, the mean of maximum and minimum envelope must be approx equal to zero.

Therefore, if $\mathrm{x(t)}$ is a signal and $\mathrm{m_1(t)}$ is its local maximum and minimum envelope, $\mathrm{h_1(t)}$ is:

$$h_1(t) = x(t) - m_1(t) \tag{B.1}$$

The signal $\mathrm{h_1(t)}$ will be the first IMF of the signal $\mathrm{x(t)}$ if it fulfills the former enunciated characteristics. In other case, the process of getting maximum and minimum envelope and mean with $\mathrm{h_1(t)}$ taking the place of $\mathrm{x(t)}$ continues till the conditions to get an empirical mode are achieved. This process is called sifting process.

When the sifting process finishes, the signal under analysis $\mathrm{x(t)}$ can be expressed as:

$$x(t) = \sum_{j=1}^{n} c_j(t) + r_n(t) \tag{B.2}$$

Where $\mathrm{c_j(t)}$ represent the jth-IMF derived of the EMD process and $\mathrm{r_n(t)}$ is the residue at the end of the overall process.

Now we can analyze independently each of the empirical modes and find out its physical meaning. In our case this can be the relationship with the dominant patterns of the load profile under analysis. To carry out this analysis we use the HSA, which is based on the Hilbert transform (HT) of the IMFs, $\mathrm{\hat{c}(t_0)}$, defined as:

$$\hat{c}(t_0) = \frac{1}{\pi} P \int_{-\infty}^{\infty} \frac{c(t)}{t - t_0} dt \tag{B.3}$$

where P is the Cauchy principal value.

The HT enables us to calculate the analytic signal defined as:

$$c_a(t) = c(t) + j\hat{c}(t) \tag{B.4}$$

Where $\mathrm{\hat{c}(t)}$ is the HT of the signal $\mathrm{c(t)}$, $\mathrm{c_a(t)}$ the corresponding analytic signal and j the imaginary unit. The module of the analytic signal is the instantaneous amplitude $\mathrm{A(t)}$ and its phase $\theta = \tan^{-1}\left[\frac{\hat{c}(t)}{c(t)}\right]$. The phase permits to calculate the instantaneous frequency:

$$f(t) = \frac{1}{2\pi}\frac{d\theta}{dt} \tag{B.5}$$

C. Genetic algorithm

In chapter 1 and 2 were highlighted the importance of getting an autonomous LMFS with self-configuration capabilities. Taking into account this objective and the revision of the literature, it was found that an excellent alternative to provide the LMFS of this capability are the evolutionary algorithms.

Here we present the outline of genetic algorithm (GA), mono objective and multi objective (MOGA). As it was seen in chapter 4 to 6, this kind of algorithm it is useful when we want to implement an algorithm of search in multidimensional spaces as it is the configuration process of an adaptive structure, for example. If it is configured properly, using a priori information, local minimums can be avoided and near optimal solutions can be found.

Genetic Algorithms have been developed by John Holland at the University of Michigan (Goldberg 1989). According to the proposed algorithm by Holland, GA consists on searching algorithms based on the mechanics of natural selection and natural genetics. They combine survival of the fittest among string structures (chromosomes) with a structured yet randomized information exchange to form a search algorithm with some of the innovative flair of human search.

In every generation, a new set of artificial chromosomes (strings) is created using pieces of the fittest of the old; an occasional new part is tried for good measure; these new chromosomes are gotten means functions or operators that mainly emulate the evolutionary processes of selection, mating and mutation.

While randomized, genetic algorithms are no simple random walk, they efficiently exploit historical information to speculate on new search points with expected improved performance. These algorithms are computationally simple yet powerful in their search for improvement.

Furthermore, they are not fundamentally limited by restrictive assumptions about the search space (assumptions concerning continuity, existence of derivatives, unimodality, and other matters).

The Figure C.1 shows the flowchart of a standard GA. Firstly, the initial population is obtained by means of random initialization process or it could be provided by the user, totally or partially, using information about where the solution could be lying. The number of individuals or chromosomes is an important parameter of the GA, one of which defines the diversity of the population. The diversity avoids not getting stuck in local minimum and ensures a broader exploration of the search space. Nevertheless, a very large population could make too slow the execution of the GA.

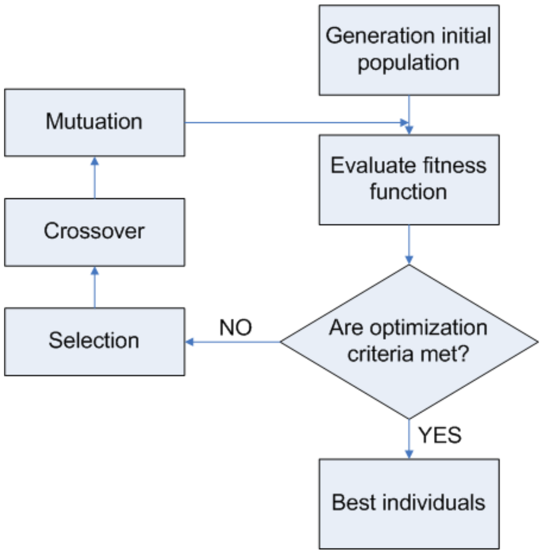


Figure C.1. Flowchart of a GA.

Once we have the initial population, the calculation of the fitness functions is executed. This is the objective function, which has to be minimized. Every individual or chromosome is evaluated by the fitness function.

Then, the optimization criteria are evaluated. If the criteria have been found, then the GA stops. If they are not, the evolutionary operators are being used in order to get the next population or next generation.

The typical operators are *selection*, *crossover* and *mutation* ones. As its name suggests, the selection is used to select the parents from the current population. There are many ways to execute this operator. Generally, the best individuals are selected as parents and some of them are directly put in the next generation (Goldberg 1989). This number of direct children is another GA parameter and it determinates the degree of opportunity for the reproduction of the best individuals. Generally is called as “the selective pressure”.

The crossover or mating function is the process of crossing over the genetic material between some parents, in order to create the genetic material of the children. The crossover can be done by different ways. For example, we can use the scattered function, which takes the genetic material from two parents and crosses them over following a generated random binary vector. Wherever the binary vector is 1, the gens are taken from parent 1 and in another case from parent 2. For example, considering as p_1 and p_2 the two parents:

$$p_1 = [a\ b\ c\ d\ e\ f\ g\ h]$$

$$p_2 = [1\ 2\ 3\ 4\ 5\ 6\ 7\ 8]$$

And if the binary vector is *[1 1 0 0 1 0 0 0]*, the function returns the following child:

$$child_1 = [a\ b\ 3\ 4\ e\ 6\ 7\ 8]$$

Mutation operation applies random changes to individual parents to form children. Mutation provides genetic diversity and enables the genetic algorithm to search a broader space. When the new population is obtained, the cycle is repeated until the optimization criteria are met.

C.1. Multi objective GA

Multi objective problem looks for the optimization of two or more fitness functions since it is normal that in optimization problem we want to minimize or maximize more than one variable. For example, we can seek to decrease the power peaks in a specific process, but without affect the performance of the whole system, for example the times of operation, comfort, etc. In order to get these objectives, a multi objective genetic algorithm can be used.

In the multi objective optimization, is normal that whereas one variable is being optimized the other one(s) is (are) being affected negatively. Then the search of an optimal solution has to be traded off in some way. So the concept of nondominated or noninferior variables and Pareto optimality appear (Goldberg 1989).

Nondominated or noninferior variables are those solutions to the optimization problem that are not dominated or inferior to any other solutions according to the fitness functions. For example, in Figure C.2 it is showed the result of evaluating the fitness functions of five possible solutions for an optimization problem of power peak reduction as main objective and operation time, as second objective. It could be observed that the best solutions are lower on the graphic and to the left. These are **A**, **B** and **C**. We can see that none of the three points is best along both dimensions. There are trade-offs from one of these three solutions to another.

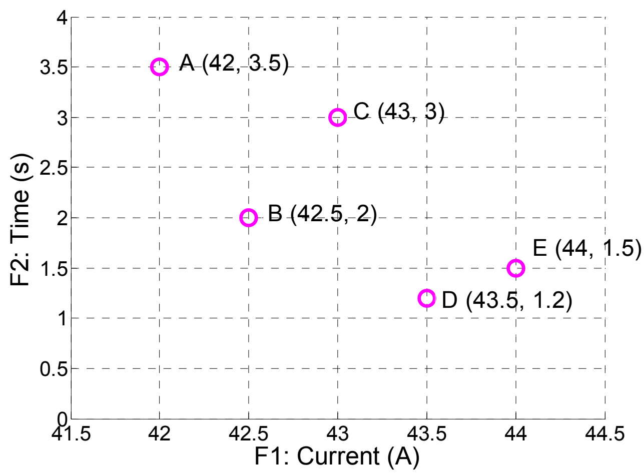


Figure C.2. Multi objective optimization. Five possible solutions for load scheduling

For instance, the solution **A** reduces the maximum load peak to 42 *A*, but introduces a delay extra in the total operation time of 3.5 *seconds*. **B** reduces the load peak to 42.5 *A*, 0.5 *A* less than **A** but introduces a delay shorter than **A**, 2 seconds only. Thus, these three points are nondominated because there are no points better than these. We can see that **D** and **E** are no good solutions and they are called dominated solutions. The set of the best solutions, in the example **A**, **B** and **C**, is called the Pareto optimal set.

An MOGA implementation is similar to a GA implementation. The main differences are: MOGA has more than one fitness function and the final result is not an alone individual, instead a set of optimal individuals are obtained at the end of the optimization process. These individuals below to the Pareto set or front. The selection of the final individual from the Pareto set depends on the main objective of the application and the importance of the other criteria to be minimized.

References

Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Longman, Inc.

Haykin, S. O. (2009). Neural Networks and Learning Machines, Prentice Hall.

Huang, N. E., Z. Shen, et al. (1998). "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis." Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences **454**(1971): 903-995.

Jang, J. S. R. (1993). "ANFIS: adaptive-network-based fuzzy inference system." Systems, Man and Cybernetics, IEEE Transactions on **23**(3): 665-685.

Wilamowski, B. M. (2009). "Neural network architectures and learning algorithms." Industrial Electronics Magazine, IEEE **3**(4): 56-63.