Università Politecnica delle Marche
Scuola di Dottorato di Ricerca in Scienze dell'Ingegneria
Curriculum in Ingegneria Biomedica, Elettronica e delle Telecomunicazioni

Energy and resources management in Micro Grid environments

Ph.D. Dissertation of:
Marco Severini

Advisor:
Prof. Stefano Squartini

Curriculum Supervisor:
Prof. Franco Chiaraluce

XV edition - new series

Università Politecnica delle Marche
Scuola di Dottorato di Ricerca in Scienze dell'Ingegneria
Curriculum in Ingegneria Biomedica, Elettronica e delle Telecomunicazioni

Energy and resources management in Micro Grid environments

Ph.D. Dissertation of:
Marco Severini

Advisor:
Prof. Stefano Squartini

Curriculum Supervisor:
Prof. Franco Chiaraluce

XV edition - new series

Università Politecnica delle Marche
Scuola di Dottorato di Ricerca in Scienze dell'Ingegneria
Facoltà di Ingegneria
Via Brecce Bianche – 60131 Ancona (AN), Italy

Abstract

Although Micro Grid technologies are still in the experimental phase, the potential improvement of efficiency robustness and flexibility is significant. The energy waste and the load swing can be greatly reduced, nonetheless an automated system that properly manages the resources is required to fully develop the potential of the available resources. On purpose, an energy management system approach, based on Mixed Integer Linear Programming technique has been investigated, implemented and proposed. The dissertation covers the theoretical aspects of the problem, such as the MILP management approach, the model of a Micro Grid for two of the most common scenarios, and the algorithms that support the management system. The experimentations have shown the effectiveness of the approach in terms of management efficiency and robustness. To improve the management, the modelling of the behaviour of a real life photovoltaic power plant has been deemed necessary . By taking into account the effect of partial shading, the actual performance of a plant can be evaluated and thus the accuracy of the forecast of solar energy production can be improved. Additionally, to feed the state of the system back to the manager, an algorithm that monitors the activity of each appliance within the system through the analysis of the aggregated energy consumption has been investigated. To support the management activity, also, a scheduling algorithm aimed at ultra low power micro controllers has been proposed and implemented, as a mean to develop sensor devices powered by renewable energy supply. This type of sensor can be effectively used in automated meter reading systems to provide the manager with the information relating water and gas consumption. Furthermore, a leakage detection algorithm has been developed and investigated to differentiate actual consumption from resource waste.

Sommario

Nonostante le tecnologie Micro Grid siano ancora in fase sperimentale, il potenziale miglioramento di efficienza, robustezza e flessibilità è significativo. Lo spreco di energia e le fluttuazioni del carico possono essere notevolmente ridotte, ciononostante un sistema automatico che gestisca correttamente le risorse risulta necessario per sviluppare completamente il potenziale delle risorse disponibili. Al riguardo, un approccio alla gestione dell'energia, basato su tecniche Mixed Integer Linear Programming è stato esaminato, implementato e proposto. La dissertazione copre gli aspetti teorici del problema, quali le tecniche di gestione MILP, il modello di Micro Grid per due degli scenari più comuni, e gli algoritmi a supporto del sistema di gestione. Le sperimentazioni hanno evidenziato l'efficacia del metodo in termini di efficienza e robustezza. Per migliorare la gestione, si è ritenuto necessario modellare il comportamento di un impianto fotovoltaico reale. Prendendo in considerazione l'effetto dell'ombreggiamento parziale, le performance dell'impianto possono essere valutate, e l'accuratezza nella predizione della produzione di energia solare migliorata. Inoltre, per fornire al gestore lo stato del sistema, un algoritmo capace di monitorare l'attività di ciascun carico a partire dall'analisi del consumo aggregato di energia è stato esaminato. A supporto dell'attività di gestione, inoltre, è stato implementato un algoritmo di schedulazione per micro controllori a consumo ridotto, per lo sviluppo di sensori alimentati da fonti rinnovabili impiegabili nei sistemi di lettura automatica dei contatori, così da fornire al manager le informazioni relative al consumo di acqua e gas. A complemento, un algoritmo per l'identificazione delle perdite, per distinguere il consumo effettivo dallo spreco di risorse, è stato investigato.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Nowadays society heavily depends on electrical energy, and the widespread of consumer electronics, air conditioning systems, and appliances has resulted in a steady rise of the energy demand. At the same time however, power distribution grids, based on old generation technology, have remained mostly unchanged and thus are reaching their limits. Because of that, to prevent grid failures, since they affect both availability and distribution of energy, the need to revise the approach towards the energy distribution has arisen.

In fact, with the integration of new power plants within pre-existing grids, even more so when renewable energy sources are used, the centralized topology, at the basis of the pre-existing power distribution generation technology, has begun to show its shortcomings, mostly pertaining flexibility, efficiency and robustness of the distribution system.

In order to overcome these limitations, a new generation of power distribution grids is being devised. At the core of the new design is the idea that a mesh topology may provide the means to overcome the limitation of nowadays power grids. Since in such a topology each energy production plant and each load would be a node in the grid, adding a new power plant would not alter the topology of the grid. At the same time the availability of many link among the nodes provides additional robustness and efficiency, given the availability of multiple paths that can be used to route the energy from the sources to the loads.

The implementation of this new generation of grids, however, presents many challenges. Among these two there is the need to coordinate the activity among the nodes efficiently, and the need to replace the old generation grid without hindrances or interruptions to the distribution services.

The first requirement implies that the nodes have to be smart, so that the a centralised coordination may be unnecessary. Thus the grid has to be smart. The second requirement, on the other hand, implies that the pre-existing grid has to be replaced a small portion at a time, which can be done by means of small scale Smart Grids, usually addressed to as Micro Grids, that can be connected at a later time to form a larger grid.

Given the complex nature of the problem, several topic are under research, addressing novel infrastructures and management challenges.

1.1 Infrastructural novelties

Regarding the structure of future power grids, when addressing the energy distribution problem, among the novel solutions, the most commonly considered are:

- Smart Grids;
- Smart Homes;
- Micro Grids.

1.1.1 Smart Grids

Smart Grids represent the future paradigm of large scale energy distribution grids [2, 3]. To ease the coordination among many power plants, Smart Grids are based on a mesh topology, whose main strong point redundancy.

Since the addition of new nodes does not alter the topology of the grid, it is easier to connect a new plant, with respect to a centralized grid. As such, even a distributed power plant, which can be regarded as a pool many separated nodes, each corresponding to a small scale independent plant, can be integrated without major issues. Since distributed power plants are more and more common with the widespread of renewable energy power plants, it is clear that a mesh topology greatly enhance the flexibility of a power distribution grid.

Although optimization is still under research [4, 5, 6], since low scale power plants are smaller, they can be located closer to the loads, resulting in a shorter distribution path and therefore in an improved efficiency.

Also, the redundant nature of mesh topologies provides the additional benefits of fault tolerance and self healing abilities. In fact, whenever a link, or a node, fails, the failing elements can be isolated and fixed, without halting the power supply toward the customers, since the availability of multiple links allows for many alternative routes to reach the customer.

With the availability of more advanced technologies, also, a diversity of solution can be integrated in a Smart Grid further improving the performance of the grids, varying from different energy sources such as fuel-cell and micro-combined heat and power plants, to energy storages, usually in the form of plug in electrical vehicles parking pools. Moreover, the smart nature of the nodes, can be used to implement proactive control strategies such as demand side management, to further lower the stress of the grid and its losses.

At present however, given the complex nature of the problems at hand, most of the solutions are currently under research, and experimental studies are based only on small scale scenarios.

1.1.2 Smart Homes

Smart Homes represent the new paradigm of residential environments, aimed to improve the comfort of the user, as well as the efficiency of the residential environment itself[7]. In that, a Smart Home is devised not only to be efficient, but also to improve the energy awareness of the user, thus guiding the user towards the efficient use of energy.

In that, Smart Homes, are envisioned as technologically advanced buildings that integrate monitoring systems and, depending on circumstances, some systems such local energy production from renewable sources, energy storages, heat ventilation and air conditioning systems, or even plug-in electrical vehicle charge systems, along with the means to allow the interaction among these.

Differently from home automation systems, that allows the user to remotely control the environment, however, a Smart Home usually also provides the means to actually manage the resources in order to improve the efficiency of their use. In fact, in most cases, a proper management of the resources, requires effort and time, thus an automated system that takes over the management in place of the user is required.

In turn, an automated energy and resource manager can provide support to further advanced technologies such as Demand Side management, which requires, however, to interact with a Smart Grid. In fact, if the utility can demote or promote the energy consumption exerted by the user, the burden of the grid can be adjusted to lower the stress of the distribution system and to improve its performance. In turn, the user can benefit from discounts by purchasing energy during off-peak hours.

Although Demand Side management can be achieved even without an automated system, it will require additional care from the user with sub-optimal performance, even more so when Dynamic pricing tariffs are used and the energy price changes hourly.

In other words, although Smart Homes can be regarded as a step closer toward the Net Zero Energy Building design [8, 9], their potential can grow further when they are integrated within a Smart Grid.

1.1.3 Micro Grids

Given the issues relating to the implementation of large scale Smart Grids, a simpler approach to the problem is represented by Micro Grids, which are small scale Smart Grids, that usually integrate a few Smart Homes, and additional resources such as small scale centralised plant for energy production, energy storage or even thermal systems[10, 11].

Thanks to its small scale, a Micro Grid can be connected to traditional power grids through a point of common coupling, thus appearing as a single load, and since a Micro Grid can spread across a small residential area, its aggregated energy consumption shows less swing, and thus is less prone to stress the main grid, with respect to the many housed directly connected to the main grid.

Moreover, a Micro Grid has less redundant resources and a lower number of subsystems, with respect to a full size Smart Grid, thus it is much easier to implement, and therefore it is the most common experimental case study in Smart Grid research. At the same time, it represents an intermediate step towards a full fledged Smart Grid, in that several Micro Grids can be interconnected into a Smart Grid at a later time.

Nonetheless, a Micro Grid, similarly to a Smart Home, also allows for centralized energy management, depending on the integrated subsystems. In addition, since larger plants can be used in a Micro Grid, with respect to a single home, such has micro-CHP, Vehicle-to-grid solutions for Plug-In Electrical Vehicles (PHEV), the efficiency can be improved further with respect to the single Smart Home.

1.2 Management challenges

Although Smart Homes and Micro Grid may possess the resources to improve the overall efficiency of the structure, to achieve a proper energy management it is mandatory to coordinate the use of energy and resources, while taking into account many aspects of the environment. Thus the development of an energy management system is required to automate the process [7, 12, 13], in order to improve the efficiency of the grid or the residential environment.

Nonetheless, although energy efficiency is important, from the end user perspective the cost savings are most often considered, whereas to the utility service a lower burden towards the grid is also important. Because of that, the improvement of the energy efficiency should not disregard other aspects of the problems such as the maintenance costs or the energy costs of the system. For instance, if the optimization process strongly depends on energy storage, the intense use may shorten the lifespan of the storage, leading to a more frequent replacement of the battery and to an increase of the maintenance costs.

From this perspective, the most efficient management schedule may not necessarily result in the lowest overall energy consumption, since it depend on the structure of grid, its topology, the resources to be managed and so on.

1.2.1 Energy Management

The aim of the energy management is the planning of the flow of energy from the sources to the loads in order to improve the efficiency of the grid. As such, not only the overall energy flow must not exceed the maximum capacity of the system, but also, unforeseen events that may lead in that direction should be addressed properly. In other words the manager has to detect external events, such as unplanned user activity, in order to prevent the overload of the system. On purpose, non invasive load monitoring techniques are mandatory to correctly identify the appliances in use.

Generally speaking, to improve the efficiency of a device or a system implies the reduction of the energy losses of the system, so that a given task can be executed with less energy. Of course, in the case of a Micro Grid, to achieve such a result would require to improve the efficiency of every device or system integrated in the grid, which is not actually related to management.

From the standpoint of energy management, in fact, to improve the efficiency of the system implies a reduction of the load that the Micro Grid exert towards the main grid through the point of common coupling. In this perspective, if the Micro Grid includes local energy production from renewable sources, the grid activity can be managed to exploit the local production thus lowering amount of energy that has to be purchased. On the other hand, if a dynamic pricing tariff is available, so that a proper energy manager allows to purchase energy when its price is at the lowest, although the energy amount may not change, both the burden towards the grid and the overall cost of the energy are lowered. Although not directly related to efficiency, these are also valuable benefits.

For instance, when demand side management is applied and the energy demand is promoted, it means that the grid burden is far below the peak level, thus a rise in the demand can be beneficial to the stability of the grid.

Clearly, energy management is most beneficial whenever it is possible to use local energy production from renewable sources, in place of fuel powered energy plant or the energy provided by the main grid. In most cases however, the energy availability from renewable sources cannot be controlled, therefore the amount of energy being purchased can be lowered effectively only by planning the user activity, so that tasks are executed when energy is available or by storing the energy and supplying it at a later time.

In a sense, then energy management is strongly bound to the management of the resources. Even considering combined heat and power (CHP) micro plant, whenever the heat production is not negligible, the waste of resources may be not tolerable, thus the activity of a micro CHP may be bound to the actual need of heat, and therefore the plant may fall in the more broad category of resources, rather than in the one of the energy plants.

1.2.2 Resources Management

Along with the management of energy, the management of resources is required to properly improve the efficiency of the environment.

Other than dealing with the limits of the environment, the manager may be required to properly prioritize the working cycles of the appliances, for instance it may need to interrupt the working cycle of an appliance at the right time, and to restart the appliance at a later time to let it complete its working cycle.

When dealing with the storage, the manager has to take into account its losses due to power conversion, and the stress of the battery system in order to avoid the need of an early replacement due to wear.

If the user comfort is being managed, the amount of energy required by the HVAC system has to be computed. Thus a proper model of the thermal behaviour of the building is required to evaluate the thermal needs. Moreover, if the HVAC system provides alternative heat or chill sources, such as heat pumps and gas boiler, the manager has to evaluate and use the most convenient one for each given time frame.

Also, if water and gas resources are to be managed, real time monitoring is required. In this case, also, novelty detection technologies to differentiate leakages from actual use are required to act accordingly.

1.2.3 Forecasting

From a broader perspective, the management of energy and use of resources focuses on planning the future activity of the Micro Grid environment. In order to achieve an efficient management, thus, the knowledge about the future conditions of the environment are required. In order to plan the use of the storage, for instance, the knowledge regarding the availability of energy from renewable source is required. If the tasks to be scheduled are not assigned, the manager need the information concerning the electrical load. Similarly, to properly plan the purchase, or the sale, of energy, the manager need to know the future prices of energy, whereas, to evaluate the thermal needs of the user, the future values of the outdoor temperature are needed.

Since each and every one quantity has its own characteristics and peculiarity, many different forecasting solutions are required. Although most of these types of information can be acquired, the acquisition process may be not straightforward and therefore at least some fall back solutions are needed nonetheless.

1.2.4 Additional models and algorithms

In addition to the issues relating to the management problem, also, the need to correctly model the resources and to correctly analyse the behaviour of the system is to be considered.

For instance, with the widespread of local Photovoltaic Power (PV) plants, a local power source is usually the most common target of an energy management system. However, in a real life scenario dealing with PV plants, the partial shading is a common occurrence that affects the performance of the local source.

Similarly, with regard to the management of the system, external factors can interfere with the management planned by the system. An appliance may fail, the user may turn on unmanaged appliances. In these circumstances, external events have to be identified, so that the proper response can be issued. With regard to this, the monitoring of the energy consumption and a Non Intrusive Load Monitoring (NILM) support may be required, so that unexpected loads can be identified on short notice.

1.3 Aims and outline

The effectiveness and efficiency of Micro Grid environments mostly depends on the resources availability, nonetheless, an automated system that properly manages the resources is required to fully develop the potential of the available resources. On purpose, to address the problem of energy and resource management in Micro Grid environments, an energy management system approach based on Mixed Integer Linear Programming (MILP) technique has been investigated, implemented and proposed. Further enhancements of the model, aimed to photovoltaic power plant modelling, and non invasive monitoring of the appliances, have been also proposed and investigated. Moreover, to integrate the management of resources such as water and gas, along with electrical energy and devices, a task scheduling algorithm aimed at ultra low powered devices, and a leakage identification algorithm required to identify potential issues based on the water and gas metering records has been proposed. The proposed research, described in the current dissertation, is the groundwork that has been carried out in order to develop an energy and resources management system that can be implemented and deployed in real life Micro Grid environment.

The state of the art, presented in Chapter 2, regards the main aspects of the management problem, that is energy management, thermal modelling of the environment, and data forecasting. Chapter 3 discusses the theoretical aspects of the problem, such as the theoretical basis of the MILP management approach, the theoretical model of the managed environment for two of the most common scenarios, and the additional algorithms that have been used to complement the management problem. The experimentations are discussed in Chapter 4 by presenting the experimental details of the case study scenarios and the results of the simulation based on the proposed scenarios. Chapter 5 focuses on the enhancement of two of the main aspects management problem, that is the partial shading effect on photovoltaic power plant and the NILM algorithms. The first topic is aimed on the modelling of the behaviour of a real life photovoltaic power plant to improve the forecast of solar energy production by taking into account the effect of partial shading and the actual performance of a real life photovoltaic power plant under different working condition. The second one focuses on the ability to monitor the activity of each appliance of the system through the analysis of the aggregated energy consumption, which can be used to feed the state of the system back to the manager. In Chapter 6 additional contributions are proposed. A scheduling algorithm aimed at ultra low power devices is proposed and implemented as a mean to develop sensor devices powered by renewable energy supply, that can be used in automated meter reading system to provide the manager with the information relating water and gas consumption. A leakage detection algorithm is also proposed to differentiate actual consumption from resource waste. Chapter 7 draws the conclusion of this work.

Chapter 2

State of the art

The management of Micro Grid or Smart Home environments, aimed at optimizing the use of energy through the planning of local tasks and device activities, is a process that requires to model the energy flow within the environment. Since the energy flow is bound to the devices, integrated within the system, each and every device has to be modelled accordingly. Moreover, if the user activity is to be planned along with the use of the energy, the tasks and their priorities are to be included in the model.

Although in most cases only a limited number of devices is integrated in the system, and thus the number of required models is limited, a general approach to the problem, able to operate in different scenarios, is complex, due to the diversity of potential subsystems to be included in the pool of models. The management of tasks further increase the diversity of the models.

2.1 Energy management

Although several optimization approaches have been proposed to address the problem of the energy management [14], most of them, such as Particle Swarm Optimization (PSO) [15], Artificial Neural Networks [16], Fuzzy Logic [17], Adaptive Dynamic Programming [18, 19, 20], but also Linear Programming [21], are not very well suited to deal with the high degree of complexities in the model that describes the flow of energy, which usually take the form of linear equations or inequalities.

On the other hand, although Mixed Integer Non Linear Programming (MINLP) [22, 23] proved to effective and rather general, since it can even use non linear equations to describe the systems, the inclusion of a non linear model can greatly increase the complexity of the problem to be solved, thus preventing a timely search of the optimal solution.

Therefore, in order to deal with a high degree of complexity, that is an high number of variables and equations, MILP [24] has been deemed the most effective approach, in that the solution can be computed in a short time, so that the energy flow within the system can be modelled and the solution to the optimization problem can be used to plan an efficient use of the energy. Although the main shortcoming of the MILP technique is that only linear models can be used within the problem, non linear models can still be used separately, to compute the input data of the problem. Moreover, linear approximation of non linear equations is still a viable solution whenever a model based on non linear equations must be used within a MILP problem.

2.2 Thermal modelling

One of the most important additions relating to the energy management issue is the management of the user comfort which usually, from the user perspective, can be regarded as the most important priority relating to any residential environment.

Since the thermal needs that satisfy the user comfort requirements depend on the thermal behaviour of the environment, the thermal flow within the building are to be modelled so that the thermal needs can be computed. Since the equation of the flow of the thermal energy are non linear, a non linear model is usually required.

Even though the model can be directly integrated in the problem, which requires a NILM programming approach [23], which is usually quite effective, a linear approximation can be also used [25], so that the computational load can be lowered, without significant accuracy loss. Nonetheless, since the temperature target is given by the user as an input to the system, the thermal comfort can be given higher priority with respect to other requirements, therefore the thermal needs can be computed separately altogether, with respect to the MILP problem, thus lowering the size of the problem.

By computing the thermal needs separately in fact, not only the main problem of the energy management results comparatively smaller since the thermal model of the environment is not present, but also the thermal need can be computed much more accurately by means of dedicated tools or, depending on circumstances, even obtained by monitoring the environment.

2.3 Forecasting

Since energy management focuses on planning the future activity of the system, in addition to model the way energy flow within the system or its thermal behaviour to compute the thermal needs, data forecasting is also of the utmost importance.

In fact, to plan the best course of action in the management of the environment, knowledge about the future energy availability, energy price, external temperature, or weather conditions, it is necessary to identify the right moment to carry out specific activities. Moreover, depending on the configuration of the system and on the complexity of the model, different types of information may be necessary.

Depending on the local energy source solar irradiation, wind direction and strength, or other quantities may be required. The forecast of the expected energy yield could be used as well. In the case that task management is not required, the future energy demand may be used.

Similarly, information on the weather conditions is required if the thermal needs of the environment are to be computed. Nonetheless, if historical data is available, the thermal needs can be forecast explicitly.

Regarding the demand side management, depending on circumstances, the forecast may be focused on the demotion and promotion factors over time. If a dynamic pricing approach is used, selling and purchase price may be forecast if they are not disclosed to the user beforehand by the service utility.

Presently, the yield of the solar power plant has been computed, based on the solar radiation that has been forecast by means of multilayer perceptrons (MLP). Also, the yield of the solar power plant has been forecast directly by means a direct forecast attempt by means of Radial Basis Function Networks (RBFNs) [26] and an on-line learning algorithm based on the Extended Minimal Resource Allocating Network (EMRAN) technique supported by an Adaptive Extended Kalman Filter (AEKF) [27, 28].

Price forecasting has been based on Extreme Learning Machine technology [29, 30], proven to be particularly effective and robust as a price classifier [31]. In fact, although many forecasting techniques are available [32, 33, 34] in literature, some of them may be not suited for residential users [35, 36], due to data input requirements, i.e. the forecaster requires data not available to the user, or because they may not target the needs of residential user, i.e. the forecaster is able to predict the market prices while not accounting additional costs and taxes.

Chapter 3

Energy and resource management: algorithms

The management of energy and resources, in a Micro Grid environment, aims to improve the efficiency of the system. Therefore, in order to plan the course of action that guarantee the most efficient use of both energy and resources, it is mandatory to model flow of the energy within the system that results from the activity of the system itself.

If a discrete time scale is used, the overall energy production, over a time frame, can be described as the sum of the energy produced by each source in each time slot within the time frame. The same consideration holds for the overall energy consumption. As such, the energy balance within a Micro Grid environment system can be described through an equation.

If the efficiency is regarded as an index, that describes the amount of energy to be purchased from the main grid to execute a given activity, its value E_{ff} can be described as a function $f(\cdot)$ of the difference between the local energy demand and the local energy production, that is:

$$E_{ff} = f(\sum_{t=1}^{slots} [\sum_{i=1}^{loads} E_{dem}(i,t) - \sum_{j=1}^{sources} E_{prod}(j,t)]), \tag{3.1}$$

where *slots* is the number of time slots within the time frame under evaluation, *loads* is the number of devices being supplied, *sources* is the number of the devices providing energy, $E_{dem}(i,t)$ is the energy demand of the i-th device at the t-th time slot, whereas $E_{prod}(j,t)$ is the energy production of the j-th device at the t-th time slot.

With regard to this, if the argument of the function $f(\cdot)$ is positive, the local energy demand exceeds the local energy production, thus additional energy is to be purchased, whereas if the argument of the function $f(\cdot)$ is negative, the local energy production exceeds the local energy demand, thus there is a surplus of energy to be sold. It is possible to conclude that the lower the argument of the function, the more the grid is efficient.

Henceforth, if the objective of the problem is to improve the efficiency of the Micro Grid, rather than compute its actual value, the actual form of the function $f(\cdot)$ can be disregarded, since the argument of the function $f(\cdot)$ is sufficient to evaluate the degree of efficiency of the Micro Grid.

Since the argument of the function $f(\cdot)$ is the balance of the energy flow within the Micro Grid, it can be described in the form of linear equations such as:

$$\begin{cases} \mathbf{c}^T \cdot \mathbf{x} \\ \mathbf{lhs} \leq \mathbf{Ax} \leq \mathbf{rhs} \end{cases} \tag{3.2}$$

where $x = [x_1, \ldots, x_n]$ is a vector whose each component represents a variable of the Micro Grid model for a single time slot within the time frame of the planning. The vector $\mathbf{c} = [c_1, \ldots, c_n]$, on the other hand, is the vector of the coefficient of the function being the object of optimization. Similarly, A is the matrix of coefficients, that implement the model of the Micro Grid in the form of linear constraints, whereas **lhs** and **rhs** are the left hand side and the right and side of the inequalities that describe the linear constraints represented by the matrix A.

3.1 MILP based approach

By means of a linear model of the Micro Grid energy flow, it is possible to find the combination of values, of the component of the vector x, that corresponds to the minimum amount of purchased energy, thus to the maximum theoretical efficiency of the system. The optimal values of the vector x can be computed by solving the set of linear inequalities by means of the simplex method and the branch and bound algorithm.

3.1.1 The simplex method

The simplex algorithm [37] is a solving method meant to handle linear programming problems. The main idea at the base of the algorithm is that, by exploiting the linear system properties, it is possible to efficiently traverse the solution space operating on a variable at a time. Every selection is carried out so that the objective function decreases, or at least does not increase, at each step.

It is possible to demonstrate that a linear programming problem can be represented in *standard form* as follows:

$$\begin{aligned} \text{Minimize} \quad & \mathbf{c}^T \cdot \mathbf{x} \\ \text{Subject to} \quad & \mathbf{Ax} = \mathbf{b}, \quad \mathbf{x} \geq 0 \end{aligned} \tag{3.3}$$

where $\mathbf{c}$ and $\mathbf{x}$ are n-dimensional vectors, whereas $\mathbf{b}$ is a m-dimensional vector and $\mathbf{A}$ is a $m \times n$ matrix.

Given the problem in standard form, a vector $\hat{\mathbf{x}}$ is a solution of the problem if and only if the columns of $\mathbf{A}$, that multiply the positive components of $\hat{\mathbf{x}}$, are linearly independent. In this case $\hat{\mathbf{x}}$ is given the name of basic feasible solution (BFS).

Since without loss of generality it is possible to assume that the rank of A matches the number of rows, it can be noticed that the constraints $\mathbf{Ax} = \mathbf{b}$ of the problem depend only on the m components of the $\hat{\mathbf{x}}$ vector, which are thus called basic variables. The remaining $n - m$ variables do not belong to the BFS and are thus given the name of nonbasic variables.

If the basic variables and non-basic variables are separated, it is possible to define an m-dimensional vector $\mathbf{x_B}$ composed of the basic variables and a $(n-m)$-dimensional vector $\mathbf{x_N}$ composed of the nonbasic variables. In a similar fashion the $\mathbf{A}$ matrix can be decomposed in a $m \times m$ matrix $\mathbf{B}$, whose columns are related to the basic variables, and a $(n - m) \times m$ matrix $\mathbf{N}$ whose columns are related to the nonbasic variables.

In this case it is possible to write:

$$\mathbf{x_B} = \mathbf{B}^{-1} \cdot \mathbf{b} - \mathbf{B}^{-1} \cdot \mathbf{N} \cdot \mathbf{x_N}. \tag{3.4}$$

meaning that the $\mathbf{x_B}$ depends on $\mathbf{x_N}$.

Therefore by replacing $\mathbf{x_B}$ in the objective function, the problem can also be rewritten as:

$$\begin{aligned} \text{Minimize} \qquad & \mathbf{c_B}^T \cdot \mathbf{B}^{-1} \cdot \mathbf{b} + \rho^T \cdot \mathbf{x_N} \qquad (3.5) \\ \text{Subject to} \qquad & \mathbf{x_B} = \mathbf{B}^{-1} \cdot \mathbf{b} - \mathbf{B}^{-1} \cdot \mathbf{N} \cdot \mathbf{x_N}, \quad \mathbf{x_B} \geq 0, \quad \mathbf{x_N} \geq 0 \end{aligned}$$

where ρ is called reduced costs vector, and is computed as:

$$\rho = \mathbf{c_N} - (\mathbf{B}^{-1} \cdot \mathbf{N})^T \cdot \mathbf{c_B} \tag{3.6}$$

The reduced cost vector is particularly meaningful because it is possible to prove that if $\rho \geq 0$ then the BFS is optimal whereas if $\rho > 0$ then the BFS is the only optimal solution. Also if there exists a negative element of ρ, and also the corresponding column of the $\mathbf{B}^{-1} \cdot \mathbf{N}$ is entirely non positive, it is possible to conclude that the problem is unbounded below.

If, for a given solution $\hat{\mathbf{x}}$, none of the two conditions are met, there exists an index i so that $\rho_i \leq 0$. Therefore given a scalar value ν and the (n-m)-

dimensional unit vector $\mathbf{e_i}$, the vector

$$\mathbf{x} = \begin{bmatrix} \mathbf{B}^{-1} \cdot \mathbf{b} - \nu \cdot \mathbf{B}^{-1} \cdot \mathbf{N} \cdot \mathbf{e_i}. \\ \nu \cdot \mathbf{e_i} \end{bmatrix} \tag{3.7}$$

is also a BFS of the problem, and also $\mathbf{c}^T \cdot \mathbf{x} \leq \mathbf{c}^T \cdot \hat{\mathbf{x}}$. In addition if the condition $\rho_i < 0$ is met then also $\mathbf{c}^T \cdot \mathbf{x} < \mathbf{c}^T \cdot \hat{\mathbf{x}}$ is true.

While the latter condition provides that the optimal solution can be found in a finite number of steps, the former is due to the fact that the same BFS can be encountered multiple times. This event occurs if the objective function assumes the same value for different BSF. In this case the algorithm stalls indefinitely in a cycle and therefore an index selection rule is necessary to prevent cycling. By dividing this process in two parts the simplex method is able to find the optimal solution of the problem (if available) in a finite step number. In the first part the feasibility of the problem is evaluated, a first BFS is found and the $\mathbf{B}^{-1}$ matrix is computed. If the problem is feasible and the BFS is not optimal the second part of the algorithm is executed. By properly selecting the value of ν and the index of the non basic vector element to replace, a new BFS is computed and evaluated. This process is repeated till a solution is found or the infeasibility condition is met.

3.1.2 Branch and Bound

The branch and bound algorithm [38] is meant to handle the combinatorial and the integer part of a MILP problem. In this case the solving procedure is based on the idea that, by dividing the feasible region in complementary not overlapping subregions, the main problem is replaced by multiple smaller sub problems, each of them with reduced complexity in respect of the original one.

This procedure, called branching, is applied recursively to each subregion as well, every time the related sub problem is too complex to be solved as it is. As a result a search tree is generated as the feasible region is inspected. From a theoretical point of view, the process may result in the inspection of every feasible solution of the problem however, since in real life scenarios a feasible space may be large, the inspection may be time consuming.

Clearly, since only one section of the feasible region contains the optimal solution, the inspection of the remaining areas is often unnecessary, thus the performance of the solution search may be improved. In fact if a meaningful area can be located early in time, the branching process in the remaining areas can be discarded (i.e., tree pruning). As a result only a limited part of the original feasible region is thoroughly inspected and the search time is severely reduced.

To sort out the unnecessary branches from the search tree, for each node generated by the branching procedure, an estimate of the best and the worst performance of the objective function is made. By comparing the best and worst results among the nodes, it is possible to select only the most promising nodes while discarding those leading to suboptimal results.

For instance when a node is generated, if the best performance of the objective function on that node is worse than the current best solution, the node can be discarded. Similarly, if the set being evaluated is dominated by a discarded set, the corresponding node can be discarded as well. Also if the node is proven not to lead to any feasible solution, or if the best solution related to the node has been found, further inspections may be avoided and the node is discarded.

In conclusion, the branch an bound algorithm can be described in four steps as follows:

- define a bound for the given problem.
- select a partition of the given set and the corresponding branching variable.
- evaluate the obtained subset
- discard unnecessary subsets

3.2 MILP problems

From a theoretical standpoint, the balance between local energy production and local energy consumption can be used to identify the Micro Grid activity schedule that corresponds to the maximum efficiency of the system. In order to implement a energy management system that can be deployed in a real life environment however, a more pragmatical approach may be required. Although energy efficiency is an important achievement, to the utility service a lower swing in the load may be a much desired goal, since it correspond to a lower burden to the grid in the peak hours. On the other hand, the end user may desire a lower energy bill.

To that end, the balance between energy cost and energy revenue can be used in place of the balance between energy consumption and energy consumption. In this way, efficiency can still be achieved, although the energy is weighted by the energy price, thus providing also more regard towards the interest of both end user and utility services.

On purpose, the MILP problem has been devised by means of a model that relies on a set of blocks and elements. The blocks model the structures within the main energy management subsystem. The elements model the single devices that compose each block. Elements can be arranged together to model a block, that results in a set of constraints that bind the variables of chosen elements together. Depending on the arrangement of elements and blocks, different system topologies, and even different operating policies, can be evaluated through simulation.

In order to investigate the modelling problem, two topologies have been evaluated: a scenario including both task scheduling and thermal model, which is used to evaluate the effect of data uncertainty, and a scenario based on a multi apartment micro grid, where electrical and thermal loads have been assigned directly as an input, which is used to investigate the effects of the grid topology and the management time frame.

For instance, a real life microgrid is used as a reference, namely the Leaf House, one of the six international case studies selected by the IEA Task 40/ECBCS Annex 52: "Towards Net Zero Energy Solar Buildings" [39, 40].

Based on the reference, several devices have been modelled, to be arranged as a MILP problem. Through this approach it has been possible to describe different topologies, by changing the arrangement of the models.

3.2.1 Task scheduling and thermal model

The first MILP problem that has been devised, has been aimed at the investigation of data uncertainty, and its effect on the management process. In order to investigate the effect of data uncertainty on different resources, the overall topology has been simplified, whereas a set of task to be scheduled and the management of the thermal needs have been included.

Notation

The list of used notation is reported as follows.
Indices:

t	time slot index
i	task index
j	house index

Parameters:

Δ	time slot duration (in hours)
$slots$	number of temporal slots
$houses$	number of houses
$tasks$	number of tasks
$tb_{j,i,t}$	task activity bit: equal to 1 when task is on, 0 otherwise
$\tau_{j,i}$	task execution time (in time slots)

$P_{j,i}$	task power demand (W)
$E_{j,i} = \Delta \cdot P_{j,i}$	task energy consumption over a single time slot (Wh)
Chl_t	electrical energy storage charge level at the t-th time slot (Wh)
Chl_t^{MIN}	electrical energy storage minimum charge level at the t-th time slot (Wh)
Chl_t^{MAX}	electrical energy storage maximum charge level at the t-th time slot (Wh)
η_c	electrical storage charge efficiency
η_d	electrical storage discharge efficiency
P_t^c	electrical storage charge rate at the t-th time slot (W)
P_t^d	electrical storage discharge rate at the t-th time slot(W)
P_{MAX}^c	electrical storage maximum charge rate (W)
P_{MAX}^d	electrical storage maximum discharge rate (W)
P_{MIN}^c	electrical storage minimum charge rate (W)
P_{MIN}^d	electrical storage minimum discharge rate (W)
$E_t^c = \Delta \cdot P_t^c$	electrical storage energy input at the t-th time slot (Wh)
$E_t^d = \Delta \cdot P_t^d$	electrical storage energy output at the t-th time slot(Wh)
E_t^{re}	renewable electrical energy input at the t-th time slot(Wh)
E_t^{he}	electrical energy required by the heater at the t-th time slot(Wh)
E_t^{so}	sold electrical energy at the t-th time slot(Wh)
C_t	electrical energy cost at the t-th time slot $(\$cent/kWh)$
S_t^{price}	electrical energy selling price at the t-th time slot(Wh)

Objective Function

To define the monetary balance, the amounts of energy respectively acquired from, and sold to the grid, are accounted in order to evaluate the energy management. To achieve the thermal comfort of the user, the thermal management of the building served by the smart grid is also included by means of a thermal model of the building.

The monetary balance of the energy management can be expressed as follows:

$$Q = \sum_{t=1}^{slots} \Bigg\{ \Bigg[\sum_{j=1}^{houses} \sum_{i=1}^{tasks} \big(E_{j,i} tb_{j,i,t}\big) + \big(E_t^c - E_t^d\big) - E_t^{re} + E_t^{so} + E_t^{he} \Bigg] C_t + \\ - E_t^{so} S_t^{price} \Bigg\} \quad (3.8)$$

where $tb_{j,i,t}$ is the task binary variable that defines the activity state (ON/OFF) of the i-th task of j-th building during the t-th time slot. The variable $E_{j,i}$ represents the energy demand of the i-th task of j-th building in each time slot. Therefore the sum over i and j index returns the total energy demand of the assigned tasks at the t-th time slot. The amount $\left(E_t^c - E_t^d\right)$ represents the energy transferred to or from the storage at the t-th time slot, while E_t^{re} is the renewable energy production, E_t^{so} accounts the sold energy amount and E_t^{he} describes the heater energy demand. Therefore the quantity within brackets accounts the net energy demand at each time slot. On the other hand, since C_t represents the energy purchase price at the t-th time slot, the amount within braces describes the total energy cost minus the total energy income at each time slot, that is the energy balance at the t-th time slot.

Electrical Task Constraints

The binary variable assignment per task and per time slot maps the activity of each task in each time slot. Based on this map, given the $\hat{\imath}$-th task of the $\hat{\jmath}$-th building at the $\hat{t}$-th time slot, the following conditions are available:

$$tb_{\hat{\jmath},\hat{\imath}}(\hat{t}) = 1 \quad \text{the task is executed at the } \hat{t}\text{-th time slot} \tag{3.9}$$

$$tb_{\hat{\jmath},\hat{\imath}}(\hat{t}) = 0 \quad \text{the task will never be executed at the } \hat{t}\text{-th time slot} \tag{3.10}$$

$$tb_{\hat{\jmath},\hat{\imath}}(\hat{t}) \leq 1 \quad \text{the task may be executed at the } \hat{t}\text{-th time slot} \tag{3.11}$$

For each task, the value of the binary bariable is left to the system to assign however, by means of the proposed conditions, the user can mark each time slot either as available, mandatory or forbidden to manipulate the task allocation partially or entirely.

The duration of tasks may span over multiple time slots. If $\tau_{j,i}$ represents the execution time length (in time slots) of the i-th task of the j-th house, the following constraints are assigned:

- task uniqueness: the active time slots assigned to a task shall match the length of the task

$$\sum_{t=1}^{slots} tb_{j,i,t} = \tau_{j,i} \quad \forall j, i \tag{3.12}$$

- task continuity: the time slots assigned to a task cannot be apart more than the task length

$$\begin{gathered}\forall j, i, \hat{t}, t: \\ \hat{t} = 1, 2, \ldots, slots - \tau_{j,i}, \\ t = \hat{t} + \tau_{j,i}, \hat{t} + \tau_{j,i} + 1, \ldots, slots \\ tb_{j,i}(\hat{t}) + tb_{j,i}(t) \leq 1 \end{gathered} \tag{3.13}$$

- in-order execution (optional): no time slot is to be assigned to the low priority task if the high priority task has not been completed yet

$$\begin{gathered}\forall \hat{t}, t: \\ \hat{t} = 1,\, 2, \ldots, slots \\ t = \hat{t}, \hat{t} + 1, \ldots, slots \\ tb_{j,high}(t) + tb_{j,low}(\hat{t}) \leq 1 \end{gathered} \tag{3.14}$$

- power constraint: in each building and in each time slot the power consumption cannot exceed the building wiring power rating

$$\begin{gathered}\forall t\colon\ t = 1,\, 2,\, \ldots,\, time\_slots \\ \forall j\colon\ t = 1,\, 2,\, \ldots,\, building \\ \sum_{i=1}^{tasks} \left[P_{j,i} \cdot tb_{j,i}(t) \right] \leq pr_j. \end{gathered} \tag{3.15}$$

where pr_j is the maximum power rating of the wiring of the j-th building.

Storage system constraints

- the charge level of the energy storage, at any given time slot, must not exceed the assigned boundaries

$$Chl_t^{MIN} \leq Chl_t \leq Chl_t^{MAX} \quad \forall t\colon\ 1 \leq t \leq slots \tag{3.16}$$

- the charger power must not exceed its own boundaries

$$0 \leq P_t^c \leq P_{MAX}^c \quad \forall t\colon\ 1 \leq t \leq slots \tag{3.17}$$

- the inverter power must not exceed its own boundaries

$$0 \leq P_t^d \leq P_{MAX}^d \quad \forall t\colon\ 1 \leq t \leq slots \tag{3.18}$$

- the energy stored, at any given time slot, depends on the energy level of the previous time slot plus the net energy income

$$Chl_{t+1} = Chl_t + \Delta P_t^c \eta_c - \frac{\Delta P_t^d}{\eta_d}$$
$$\forall t:\ 1 \leq t \leq slots - 1 \tag{3.19}$$

Energy constraints

- At each time slot the energy input to the system shall not fall below zero nor exceed the maximum available amount.

$$0 \leq \sum_{j=1}^{houses} \sum_{i=1}^{tasks} \left(E_{j,i} t b_{j,i,t}\right) + \left(E_t^c - E_t^d\right) - E_t^{re} + E_t^{so} + E_t^{he} \leq P_t^{MAX}$$
$$t:\ 1 \leq t \leq slots \tag{3.20}$$

Thermal Optimization Sub-Problem

The thermal model is presented apart since it is used separately to compute the thermal needs which are then included in the main problem.

Notation

Indices:

t	time slot index
j	house index
k	room index

Parameters

$surf$	number of surfaces of the house thermal model
E_t^{he}	electrical energy demand at the t-th time slot
$P_{j,k,t}^{th}$	thermal power at the t-th time slot to the k-th room of the j-th building
P_t^{th}	thermal power at the t-th time slot (single room case)
Δ	time slot duration (in hours)
θ_{hp}	heat pump output temperature ($°C$)
θ_i	indoor temperature ($°C$)
θ_o	outdoor temperature ($°C$)
$\theta_t(t)$	target temperature at the t-th time slot ($°C$)
$\theta_i^* = \frac{\theta_t(t)+\theta_i(0)}{2}$	mean value of the indoor temperature ($°C$)

ε	temperature tolerance (0.5) ($°C$)
c_p	air heat capacity at NTP
M_{air}	air mass in the house (Kg)
k_l	house heat loss factor ($W/°C$)

Heat pump constraints

An heat pump is used for heating/cooling. At each time slot, the required electrical energy can be computed as the sum of the ratio among the thermal energy required by each room and the Coefficient of Performance (COP) of the heat pump when heating. In place of the COP the Energy Efficiency Ratio (EER) of the heat pump is used when cooling:

$$E_t^{he} = \frac{1}{COP} \sum_{j=1}^{houses} \sum_{k=1}^{rooms} \left(\Delta P_{j,k,t}^{th}\right) \quad \forall t : t = 1, \ldots, slots. \tag{3.21}$$

The thermal power $P_{j,k,t}^{th}$ of the k-th room of the j-th building at the t-th time slot is computed by the solving algorithm in order to satisfy the thermal constraints. The thermal constraints are defined with the heat balance of the buildings as a base [41, 42, 43]. The overdot notation can be used to represent the time derivative, thus the heat fluxes can be defined as:

$$\dot{Q}_{hp} = P^{th} c_p (\theta_{hp} - \theta_i) \tag{3.22}$$

$$\dot{Q}_{loss} = k_l (\theta_i - \theta_o) \tag{3.23}$$

where (3.22) expresses the thermal energy provided by the heat pump while (3.23) represents the thermal energy escaping through the walls. On the other hand the indoor temperature variation over time can be calculated as the net heat flux on the thermal capacity of the air mass inside the room, which leads to the following:

$$\dot{\theta}_i = \frac{1}{M_{air} c_p} \left(\dot{Q}_{hp} - \dot{Q}_{loss}\right). \tag{3.24}$$

Since the management process is based on a discrete time domain formulation, from (3.24) a difference equation can be obtained, and thus the indoor temperature can be defined as:

$$\begin{aligned} \theta_i(t+1) = \theta_i(t) + \\ + \frac{\Delta}{M_{air} c_p} \cdot \left[P_t^{th} c_p(\theta_{hp} - \theta_i(t)) - k_l(\theta_i(t) - \theta_o(t))\right] \end{aligned} \tag{3.25}$$

being t the discrete time variable. By algebraic manipulation the temperature can then be expressed as:

$$\begin{aligned} \theta_i(t+1) = & \theta_i(t)\left[1 - \frac{P_t^{th}\Delta}{M_{air}} - \frac{k_l \Delta}{M_{air} c_p}\right] + \\ & + \frac{P_t^{th}\theta_{hp}}{M_{air}} + \frac{k_l \theta_o(t)}{M_{air} c_p} \end{aligned} \tag{3.26}$$

leading to a nonlinear constraint. On the other hand, by means of the approximation

$$\dot{Q}_{hp} = P^{th} c_p(\theta_{hp} - \theta_i^*) \tag{3.27}$$

the equation (3.25) could be written as

$$\begin{aligned} \theta_i(t+1) = & \theta_i(t)\left[1 - \frac{k_l \Delta}{M_{air} c_p}\right] + \\ & + \frac{P_t^{th}(\theta_{hp} - \theta_i^*)}{M_{air}} + \frac{k_l \theta_o(t)}{M_{air} c_p} \end{aligned} \tag{3.28}$$

and thus the linear constraint is obtained.

In either case, for each room of each building, the constraints set is obtained by requiring that

$$\begin{gathered} \forall t:\ t = 1, \ldots, slots \\ \theta_t(t) - \varepsilon \leq \theta_i(t) \leq \theta_t(t) + \varepsilon. \end{gathered} \tag{3.29}$$

The coefficients M_{air} and k_l that appear in the equations represent, respectively, the air mass within each room and the heat loss factor of each room. Their value are obtained based on a simple building geometry accounting several rooms. Regarding k_l is worth to mention that its value is estimated in accordance to the EN 12831, EN ISO 13370, EN ISO 13789 standards [44, 45, 46].

3.2.2 Multi apartment revamping model

The other MILP problem to be devised, models a multi apartment residential environment. The proposed model aims to evaluate the performance of the reference environment, and the effect of the topology as well as the management time frame. In this case, task scheduling and thermal modelling have not been included to simplify the evaluation process.

The objective function to be minimized has been revised as the sum of the costs of the energy to be purchased and the energy surplus. The aim is to avoid either unnecessary purchase and, at times, unnecessary sale of electricity, thus promoting storage, since the sale price is usually much lower than the purchase price. The variable modelling the energy fluxes within the environment can be addressed as following:

- EP_a: amount of energy purchased to supply the entire structure;
- ES_a: amount of energy surplus resulting from the entire structure;
- E_p: the energy purchase price;
- EP_i: energy required by the $i-th$ electrical block with $i = 1, 2, 3$;
- Eg_i: energy required by the $i-th$ electrical storage (to guard against over discharge) with $i = 1, 2$;
- EP_{hw}: electricity consumption of the water boiler;
- EP_{he}: electricity consumption of the heat pump during heating phases;
- EP_{re}: electricity consumption of the heat pump during cooling phases;
- ES_i: energy production surplus provided by the $i-th$ electrical block with $i = 1, 2, 3$.
- EP_{th}: electricity demand of the thermal blocks not covered by the available electricity surplus;
- $Cap_i(t+1)$: residual capacity level (of the $i-th$ electrical storage) at the end of the t-th time slot;
- $Cap_i(t)$: residual capacity level (of the $i-th$ electrical storage) at the beginning of the t-th time slot;
- $Ch_i(t)$: charged energy amount (of the $i-th$ electrical storage) during the t-th time slot;
- $Di_i(t)$: discharged energy amount (of the $i-th$ electrical storage) during the t-th time slot;

- De_i: demanded energy amount (of the $i-th$ electrical block);
- PV_i: energy production of the photovoltaic panel (of the $i-th$ electrical block);
- Ch_{eff}: charge efficiency of each storage device;
- Di_{eff}: discharge efficiency of each storage device;
- S_{di}: self discharge of each storage device;
- Ch_g: hourly energy amount, provided by the main grid, to compensate the self discharge;
- bg_i: binary guard that equals zero if $Cap_i(t)$ is higher than 30%;
- Ep_{hw}: heat production of the water boiler;
- Ch_{hw}: heat collected by the water tank;
- Di_{hw}: heat lost as the water leave the tank;
- S_{hw}: heat provided by the thermal solar panel;
- De_{hw}: heat demand corresponding to the hot water being used;
- $Cap_{hw}(t+1)$: energy level of the water tank at the end of the t-th time slot;
- $Cap_{hw}(t)$: energy level of the water tank at the beginning of the t-th time slot;
- Ep_{he}: heat production of the heat pump;
- Ch_{he}: heat collected by the heater storage;
- Di_{he}: heat output of the heater storage;
- De_{he}: heat demand of the environment;
- $Cap_{he}(t+1)$: energy level of the heat storage at the end of the t-th time slot;
- $Cap_{he}(t)$: energy level of the heat storage at the beginning of the t-th time slot;
- Ep_{re}: chill production of the heat pump;
- Ch_{re}: chill collected by the chill storage;
- Di_{re}: chill output of the chill storage;

- De_{re}: chill demand of the environment;
- $Cap_{re}(t+1)$: energy level of the chill storage at the end of the t-th time slot;
- $Cap_{re}(t)$: energy level of the chill storage at the beginning of the t-th time slot.

Therefore the objective function can be written as:

$$Q = (EP_a + ES_a) \cdot E_p \tag{3.30}$$

meaning that either energy purchase and the sale of energy surplus are not desirable, and thus they are accounted as a cost. In other words the objective function is used to model a policy rather than the actual energy cost. Of course, depending on the policy of choice, the objective function can be restricted not to account the energy surplus as a cost.

Almost every configuration adopted in the current scenario is composed of three electrical blocks and three thermal blocks. Since the thermal blocks require electrical energy to provide the thermal management two alternatives are possible.

On the one hand, if the model is configured not to supply the electricity surplus to the thermal blocks, the energy amount to be purchased can be computed as:

$$EP_a = EP_1 + EP_2 + EP_3 + Eg_1 + Eg_2 + EP_{hw} + EP_{he} + EP_{re} \tag{3.31}$$

which results in the energy surplus, computed as:

$$ES_a = ES_1 + ES_2 + ES_3 \tag{3.32}$$

being either stored or discarded.

On the other hand, if the model is configured to route the electricity surplus to the thermal blocks, Eq. (3.31) and (3.32) can be revised in the following manner:

$$EP_a = EP_1 + EP_2 + EP_3 + Eg_1 + Eg_2 + EP_{th}. \tag{3.33}$$

$$EP_{th} - ES_a = EP_{hw} + EP_{he} + EP_{re} - (ES_1 + ES_2 + ES_3). \tag{3.34}$$

In this case, the variable EP_{th} has been introduced to account the electricity demand of the thermal blocks that is not covered by the available electricity surplus. Conversely, ES_a represents the electricity surplus that still exceeds the needs of the whole structure.

Concerning the electrical blocks, the amount of electricity to be purchased, and the electricity surplus, are bound to production and demand according to the following:

$$EP_i - ES_i - Ch_i + Di_i = De_i - PV_i \tag{3.35}$$

meaning that purchase or sale, while accounting the energy managed by the storage, must match the difference among production and demand.

The electrical storage devices are modelled, each, by computing its residual capacity level as:

$$\begin{aligned} Cap_i(t+1) = Cap_i(t) + Ch_i(t) \cdot Ch_{eff} - Di_i(t) \cdot Di_{eff} + \\ - S_{di} + Ch_g \cdot Ch_{eff} \cdot bg_i(t) \end{aligned} \tag{3.36}$$

thus accounting charge and discharge efficiency as well as self discharge. Also, the model includes a charge level guard. If the battery residual energy level drops to 30%, a charger, that compensates for self discharge, is enabled so that the battery depletion due to self discharge is prevented.

A few constraints have also been devised to model the original policy used to manage the storage devices:

$$Ch_i(t) = \begin{cases} 0, & \text{if } PV_i(t) \leq De_i(t) \\ PV_i(t) - De_i(t), & \text{if } PV_i(t) > De_i(t) \end{cases} \tag{3.37}$$

$$Di_i(t) = \begin{cases} 0, & \text{if } Cap_i(t) < 65 \\ \\ 0.9 \cdot (De_i(t) - PV_i(t)), & \text{if } \begin{cases} PV_i(t) < De_i(t) \\ Cap_i(t) \geq 65\% \,. \end{cases} \end{cases} \tag{3.38}$$

For instance, Eq. (3.37) prevents purchased electricity from being stored, whereas Eq. (3.38) prevents storage device from being deeply discharged. These constraints are intended to achieve a conservative management of the storage, and they are paired with an explicit avoidance of energy sale. Thus they are complemented with the objective function presented in Eq. (3.30). Conversely, to adopt a more aggressive or, which is the same, a less conservative policy, said constrains are not applied. In this case, also, Eq. (3.30) does not account the energy surplus amount.

Similarly to the electrical blocks, with regard to the hot water management, the balance between consumption and production has been modelled:

$$Ep_{hw} - Ch_{hw} + Di_{hw} = De_{hw} - S_{hw}, \tag{3.39}$$

thus accounting hot water production and demand, and the management of a hot water tank.

The energy capacity of the water tank can be computed as:

$$Cap_{hw}(t+1) = Cap_{hw}(t) + Ch_{hw}(t) - Di_{hw}(t) \tag{3.40}$$

assuming that heat loss over time from the storage is negligible.

In the case of the heating block and the refrigeration block, the model is almost identical, the only exception being the lack of energy production from renewable sources, and the use of the heat pump in place of the water boiler. For instance, being the heat balance in the form $Ep_{he} - Ch_{he} + Di_{he} = De_{he}$, the integration of hot water and heat production has been modelled as a linear combination of said balance with Eq.(3.39), in place of two separate equations.

3.3 Additional algorithms

In addition to the MILP problems described in the scenarios presented in Sections 3.2.1 and 3.2.2, additional algorithm have been implemented in order to evaluate different aspects of the management problem.

In fact, since the thermal model has been devised as an Non Linear Programming problem, a suitable solver is required to compute the necessary input data.

Moreover, although a MILP problem can be used to model the energy flow within the system, to plan the activity schedule that regulate the system in the near future, much data is required beforehand. For instance in order to decide if a task can be delayed or not, the manager needs to know if delaying the execution of the task can lower the amount of energy to be purchase, which may depend on the energy availability of the local power plant.

In other words, different kinds of information have to be forecast depending on the resources to be managed. In addition, the forecasting process is prone to error thus introducing uncertainty in the data.

3.3.1 Non Linear Programming - Genetic Algorithm

To solve the NLP problem resulting from the nonlinear thermal model described in Section 3.2.1, a Genetic Algorithm solver, which is part of the MatLab Global Optimization ToolBox[1], is used. In this case the objective function and the

[1] http://www.mathworks.it/it/help/gads/index.html

constraints are assigned in the form of function handles.

Differently from deterministic approaches, a Genetic Algorithm (GA) [47, 48] searches for the optimal solution of the problem by emulating the evolutionary process that characterizes living organisms. As evolutionary processes combine diversification and selection, in a similar way a genetic algorithm generates feasible solutions and selects the most fitting ones among the generated pool. The solution space is represented in terms of genes, or chromosomes, while the objective function, provided to the solver, is used to select the best candidates.

The first step of the algorithm consists in the characterization of the set of variables as set of chromosomes. The second step purpose is the initialization of the population of candidates. In the third step a selection of the parent candidates is made so that, in the fourth step, the genetic procedures can be applied in order to produce new candidates. In the fifth step the old population is replaced by the new candidates. At this point the stop conditions are evaluated. If none of them is met, the process returns to step three.

Concerning the feasible solution space, usually the solution candidates contain information regarding the solution in the form of a binary string. However different criteria can be applied and thus different representations may be possible, such as integer values, permutations or even floating point values.

The initialization procedure provides the first generation of the solution candidates. The candidates are often obtained through a pseudo random number generation routine but, in this case as well, variations on the process can be made. Usually the generation interval can be provided by the user so that a promising area can be targeted. In other circumstances the population can be assigned altogether, rather than generated randomly, in order to improve the results of a previous optimization attempt.

The parent candidates selection is generally based on a random process against the fitness function. The most common approach sees the generation of two random numbers, that are used to discriminate the candidates, depending on their fitness value. Depending on the problem and its fitness function, this selection method may select only average performing solutions, thus removing the best performing ones, or even select the most promising ones and affecting the variety of the chromosome set. Since in either cases the space solution inspection may be affected, different corrective measures can be applied. The normalization of the population against the fitness function may help in balancing the chances of propagation of the chromosomes. On the other hand, the elitism can be used to preserve the best performing candidates found at each step.

The generation of new candidates is achieved by means of a crossover procedure and a mutation procedure. In the crossover procedure the chromosome set of the parents is mixed to generate the offspring. Among the different techniques, the single point crossover selects a random point within the chromosome. The part preceding the crossover point is provided by one parent, the part following it is selected from the second parent. In a similar fashion multiple crossover point can be selected. As an additional method, single elements of the chromosome encoding can be randomly selected.

The mutation procedure, differently from the crossover approach, does not involve the parents chromosomes but the offspring ones only. In this case randomly selected bits, or values, of the chromosome can be altered to increase the chromosome variety within the population and to better cover the space of feasible solutions.

Since the generation of new candidates has stochastic nature, in case a constrained problem is evaluated, infeasible solutions may be generated. In this case a corrective measure can be used to guide the evolution process. The infeasible solutions may be further impaired so that their chromosomes will not propagate towards the next generation of candidates. As an alternate measure the chromosomes can be mutated on purpose so that the constraints of the problem are satisfied.

At the end of the generation process the old population may be replaced completely, or partially, depending on the characteristics of the problem. At this point the iteration can start anew with a selection of additional parent candidates. Clearly, after consecutive iterations, the chromosome variety decreases, and therefore improvements in the fitness value of the candidates occur less frequently. While this event is usually one of the stop conditions, since the candidates can converge towards local minima, care is often required to avoid the premature interruption of the process. To avoid this occurrence a corrective measure to increase the chromosome variety can be applied. For instance the stopping conditions may be used to check the number of candidate generations reached, the number of consecutive candidate generations produced without any improvement, the improvement of the average fitness value with respect of a given reference value.

3.3.2 Forecasting algorithms

In order to investigate different aspect of the energy and resource management process, different working conditions have been devised. On purpose, different forecasting algorithms have been deemed necessary, and thus implemented, to provide the required sets of input data.

Multi Layer Perceptron forecasting

To evaluate how the forecasting process affects the task scheduling and energy resource management, based on the model proposed in Section 3.2.1, three forecasters have been devised. Since absolute prediction accuracy is paramount to evaluate the management error, the implementation has been designed with forecaster simplicity in mind.

In particular, by means of the Neural Network toolbox, provided by the MathWorks MatLab[2] 2012a framework, three multi-layered perceptrons (MLP) have been used as forecaster. The neural networks training is carried out over a data set covering two years, namely 2009 and 2010. The National Climatic Data Center[3] (NCDC) has been selected for the meteorological records, whereas the ISO of New England[4] (ISO-NE) provided information about the energy price.

The three forecaster share the same MLP structure: 49 neurons are used in the input layer, 48 in the hidden layer and 24 in the output layer. For every neuron the hyperbolic tangent is used as activation function. A day ahead prediction approach is used, thus 24 samples, one per hour, are generated at a time from the samples of the previous day. The data normalization is left to the toolbox and it maps the input to the interval [-1, 1]. The performance function of choice is the Mean Absolute Error (MAE). To reduce the training time, given the number of neurons, the gradient descent with momentum and adaptive learning rate is used. To avoid a premature completion of the training process, the number of validation check is fixed equal to 500, and the number of epoch has been increased accordingly.

Concerning temperature and irradiation forecasting, for each predicted set, the input data is represented by the 24 temperature hourly samples, the 24 solar irradiation hourly samples, and the day index within the year, which is coded through the minus cosine function. Regarding the price forecast, on the other hand, the energy price hourly samples are used as input in place of the solar irradiation hourly samples. The temperature samples are still used as input in this case. Clearly, since the output data is composed by 24 samples representing the hourly data of the next day, recurring prediction and error propagation have been avoided.

Given the simple structure of the network and its input set, the prediction accuracy is usually high if little to no changes are recorded from a day to the next. On the other hand, when highly variable conditions are encountered, the prediction accuracy drops. In this regard, even though a deep analysis of the results has not been carried out, a simple evaluation reveals that a proper data preprocessing would greatly improve the performance of the forecaster. Additionally, the nature of the forecast error appears to be systemic, and probably due to the lack of input data. As a result, the performance is not on par

[2]http://www.mathworks.com
[3]http://www1.ncdc.noaa.gov
http://www1.ncdc.noaa.gov/pub/data/nsrdb-solar/station-data-2010/
[4]http://www.iso-ne.com/aboutiso/index.html
http://www.iso-ne.com/markets/hstdata/hourly/index.html

with the performance of the state of the art. Nonetheless the obtained forecast provides an ample set of cases, that are more than adequate to test the error propagation through the scheduler. In a real life scenario, an accurate predictor would be mandatory. As such, the improvement of the forecaster issues will be addressed in future works.

Solar Power Forecasting

To evaluate the effect of the topology on the energy management process, based on the scenario modelled in Section 3.2.2, an on-line learning procedure is proposed, to predict the PV output and the Solar Thermal power output. On purpose, Radial Basis Function Networks (RBFNs) have been used. These networks have been widely used for nonlinear system identification [26] because they have the ability both to approximate complex nonlinear mappings, directly from input-output data, with a simple topological structure that avoids lengthy calculations, and to reveal how learning proceeds in an explicit manner [49]. The proposed on-line learning algorithm is based on the Extended Minimal Resource Allocating Network (EMRAN) technique, that adds hidden neurons to the network, based on the innovation of each new RBFN input pattern which arrives sequentially. As stated in [26], to obtain a more parsimonious network topology, a pruning strategy is introduced. This strategy detects and removes, as learning progresses, those hidden neurons which provide little contribution to the network output. If an observation has no novelty then, the existing parameters of the network are adjusted by an Extended Kalman Filter (EKF). In this paper the performance of the filter is improved by an on-line adjustment of the noise statistics, obtained by a suitably defined estimation algorithm; the proposed Adaptive Extended Kalman Filter (AEKF) is able to adaptively estimate the unknown statistical parameters [27, 28]. To minimize the computational effort, in real-time implementation, a "winner neuron" strategy is incorporate in the learning algorithm [26, 28]. In this work the proposed RBFN-based prediction algorithm is used, because it has the capability to adapt on-line as operating conditions vary (i.e. night and day and season succeed). Also these Networks do not need a training dataset and they can be used in different case studies without a training stage. The EMRAN estimation algorithm enhanced by the AEKF is called EMRANAEKF algorithm [27], and is shown in Fig. 3.1. In this work the input data consists of a tapped delay line of 10 samples of the past data, each sample corresponding to a one-hour time interval. The forecast is aimed to predict external temperature, solar irradiation and photovoltaic power data, so the input dimension is $10 \cdot 3$. The output data is a "one-hour ahead" forecast of solar irradiation and photovoltaic power production.

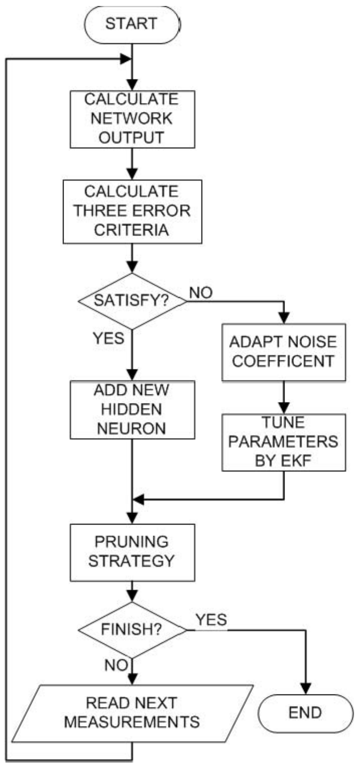


Figure 3.1: Flow chart of the EMRANAEKF algorithm.

Price Forecasting: Extreme Learning Machines

To evaluate the effect of the management time frame, in dynamic pricing scenarios, based on the MILP problem described in Section 3.2.2, a day-ahead forecaster has been implemented as Single Layer FeedForward network (SLFN) with kernel based ELM training. In ELM, the input weights of SLFNs do not need tuning and they can be randomly generated, whereas the output weights are analytically determined using the least-square method, reducing the training time.

Consider a set of N labelled training samples $\{(\mathbf{x}_1, t_1), \ldots, (\mathbf{x}_N, t_N)\}$, where $\mathbf{x}_i \in \mathbb{R}^I$ and $t_i \in \{-1, 1\}$, and a SLFN with I input neurons and L hidden neurons. The ELM decision function is the following:

$$f_L(\mathbf{x}) = \sum_{i=1}^{L} \beta_i h_i(\mathbf{x}) = \mathbf{h}(\mathbf{x})\boldsymbol{\beta}. \tag{3.41}$$

In the equation, the vector $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_L]^T$ contains the weights connect-ing hidden neurons and output ones, while $\mathbf{h}(\mathbf{x}) = [h_1(\mathbf{x}), \ldots, h_L(\mathbf{x})]$ is the output of the hidden layer with respect to the input $\mathbf{x}$. In general, $\mathbf{h}(\mathbf{x}) = [G(\mathbf{a}_1, b_1, \mathbf{x}), \ldots, G(\mathbf{a}_L, b_L, \mathbf{x})]$, $G(\mathbf{a}, b, \mathbf{x})$ is a non linear piecewise continuous function, that satisfies ELM universal approximation capability theorems, and $\{\mathbf{a}_i, b_i\}_{i=1}^L$ are randomly generated.

Defining the hidden-layer output matrix $\mathbf{H}$ as

$$\mathbf{H} = \begin{bmatrix} h_1(\mathbf{x}_1) & \cdots & h_L(\mathbf{x}_1) \\ \vdots & \vdots & \vdots \\ h_1(\mathbf{x}_N) & \cdots & h_L(\mathbf{x}_N) \end{bmatrix}, \tag{3.42}$$

training the ELM consists in minimizing $\|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\|$ and $\|\boldsymbol{\beta}\|$, where $\mathbf{T} = [t_1, t_2, \ldots, t_N]^T$. The solution to the problem can be calculated as the minimum norm least-square solution of the linear system:

$$\hat{\boldsymbol{\beta}} = \mathbf{H}^\dagger \mathbf{T}, \tag{3.43}$$

where $\mathbf{H}^\dagger$ is the Moore-Penrose generalized inverse of matrix $\mathbf{H}$. By computing output weights analytically, ELM allows good generalization performance with speedy training phase.

In kernel-based ELM, $\mathbf{h}(\mathbf{x})$ is unknown, and the output function can be written as:

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} K(\mathbf{x}, \mathbf{x}_1) \\ \vdots \\ K(\mathbf{x}, \mathbf{x}_N) \end{bmatrix}^T \left(\frac{\mathbf{I}}{C} + \boldsymbol{\Omega} \right)^{-1} \mathbf{T}, \tag{3.44}$$

where $\boldsymbol{\Omega}$ is defined so that each element $\Omega_{i,j} = h(\mathbf{x}_i) \cdot h(\mathbf{x}_j) = K(\mathbf{x}_i, \mathbf{x}_j)$. $K(\cdot, \cdot)$ is a kernel function as in SVM.

The Kernel based ELM solution has been used for price forecasting in our experimentation.

Chapter 4

Energy and resource management: set-up configuration and test

Energy and resource management techniques are being devised to improve the efficiency of an existing Micro Grid environment, nonetheless, given the complex nature of the problem, many factors can affect the performance of the system.

Among the many factors, the prediction error that affects the forecast data may be responsible for a poor management. Also, the availability of resources such as local energy sources or energy storages, or even the degree of integration among the subsystems, may enhance or impair the performance of the micro grid.

In order to investigate these issues, the scenarios proposed in Sections 3.2.1 and 3.2.2 have been implemented in and evaluated.

4.1 Task scheduling and thermal model

The most simple scenario, whose model is proposed in Sections 3.2.1 has been used to focus on the effect of data uncertainty due to the forecast input. On purpose, the MLP based forecasters presented in Section 3.3.2 have been used in this case.

4.1.1 Solar production

The current simulation scenario assumes a solar power plant as part of the domestic environment, its panel area being 20 squared meters, and its efficiency index being equal to 20%. As previously stated, the energy production is computed using the hourly solar irradiance as an input.

4.1.2 Data forecasting

To evaluate the scheduling framework performance, a forecast data set is used as an input to the scheduler. Although the forecast time frame spans over almost two years, which allowed several tests to be carried out, only the forecasts corresponding to a few days are presented as a sample.

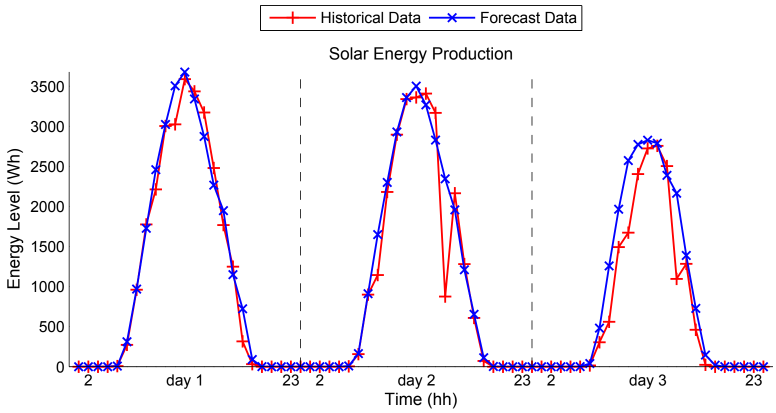


Figure 4.1: Solar Energy Production: forecast against historical data

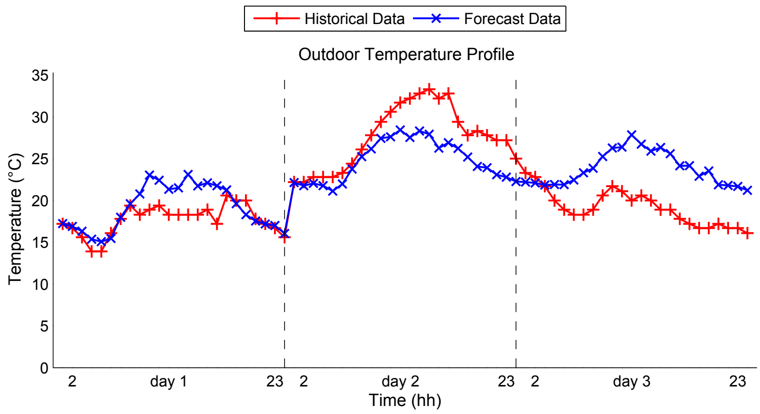


Figure 4.2: Temperature Profile: forecast against historical data

Also, to evaluate the forecast accuracy in a normalized fashion, for each day, the MAE of the forecast, against the corresponding historical data, is divided by the absolute mean value of the historical data. This approach has been devised to avoid the issues, due to negative and null values, that usually affect the Mean Absolute Percentage Error (MAPE).

This characterization was used to select three forecast set of samples. The

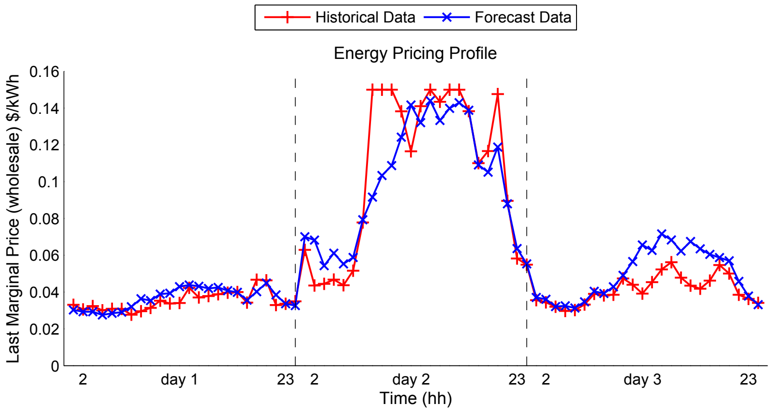


Figure 4.3: Energy Pricing Profile: forecast against historical data

set are referred to as "day 1" (02/05/2010), "day 2" (10/08/2010) and "day 3" (24/09/2010). The normalized MAE for these forecasts lies, respectively in the intervals [0, 0.1], (0.1, 0.2] and (0.2, 0.3].

In Figures 4.1, 4.2 and 4.3, the input data profiles corresponding, respectively, to solar energy production, outdoor temperature and energy prices are presented, comparing the forecast data against the historical counterpart for each of the selected days.

4.1.3 Task scheduling and energy cost accounting

The management interval spans over a time frame of 24 hours, and requires the scheduling of a set of appliances including a washing machine, a drying machine, a oven and a dish washer. A TV, also, has been considered in the scenario. The tasks are described in Table 4.1.

Table 4.1: Assigned tasks and their respective timings

	Allowed time window		Forbidden time window		Tasks Parts	Energy Demand	ID	Master ID	Duration (Hrs)
	Begin	End	Begin	End					
Washing machine	8:00	22:00	14:00	16:00	3	800	task 1	-	1
						1000	task 2	task 1	1
						900	task 3	task 2	1
Drying Machine	8:00	22:00	14:00	16:00	1	2500	task 4	task 3	2
Oven	10:00	13:00			1	2000	task 5	-	2
Dishwasher	14:00	22:00			3	800	task 6	-	1
						1000	task 7	task 6	1
						500	task 8	task 7	1
TV	8:00	18:00			1	100	task 9	-	10

Regarding the storage system, a single set of batteries is considered. The parameters are reported in Table 4.2.

Table 4.2: Storage system parameters

η_c	η_d	Chl^{MIN}	Chl^{MAX}	P^c_{MAX}	P^d_{MAX}
		kWh		kW	
0.85	0.85	1	5	2	2

Concerning the energy rates, a dynamic pricing scheme is emulated by mean of either historical or forecast market prices. On this regards, since the values refers to the wholesale locational marginal price, whereas the retail price includes taxes and ancillary costs, to obtain a realistic retail price, a multiplicative factor of ten is also accounted. As a result, in the current scenario, the retail energy price is assumed to be ten times the wholesale price. In the same scenario, the energy selling price has been assumed equal to 1 $cent per kWh.

4.1.4 Thermal model characterization

To properly evaluate the performance of the thermal regulation, a simple building structure has been employed as a target. While the structure is simplified enough to reduce the complexity of the model, the characterization is based on the European Standards EN 12831:2003, EN 13370:2007 and EN 13789:2007, so that all the most important thermal loss contributions are included.

A schematic representation of the building is reported in Fig. 4.4.

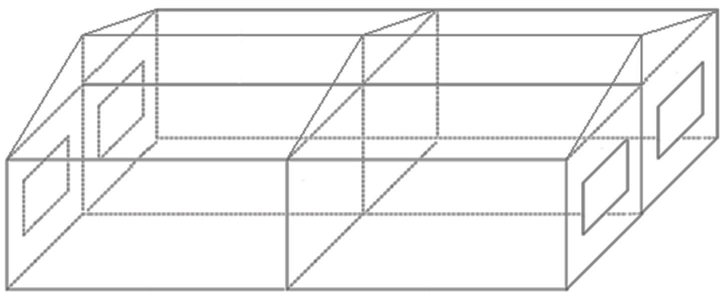


Figure 4.4: Building model schematic representation

The building parameters are shown in Table 4.3.

The set of parameters and the air density at NTP (1.204 kg/m^3) are used to compute the air mass within each room, referred as M_{air} in the model description in Section 3.2.1. The parameter referred as k_l that is, the heat loss

Table 4.3: Building geometry and parameters

Parameters	Value	Unit
Building's length	25	m
Building's width	10	m
Rooms' length	12.2	m
Rooms' width	4.7	m
Building's height	4	m
Roof's pitch	35	°
Windows' count	4	-
Windows' height	1	m
Windows' width	1	m
Windows' thickness	0.005	m
Walls' thickness	0.2	m
Slab on grade thickness	0.3	m
Windows' thermal conductivity	0.78	$W/(m^2 \cdot^\circ C)$
Walls' thermal conductivity	0.30	$W/(m^2 \cdot^\circ C)$
Slab on grade thermal conductivity	0.82	$W/(m^2 \cdot^\circ C)$

factor, is computed accordingly to the procedure stated in the aforementioned European Standards.

Concerning the heat pump, a 6 kW device is used. The COP factor is equal to 3.4 whereas the EER factor is chosen equal to 3. In the current scenario the outdoor temperature spans from about 15 $°C$ to about 35 $°C$, thus no additional adjustment to either the COP or the EER is deemed necessary. The output temperature of the heat pump is assumed equal to 50 $°C$ when heating, and equal to 10 $°C$ when cooling. The heat pump is assumed able to switch from heating to cooling and vice versa depending on the outdoor temperature.

In order to compute the energy demand of the thermal regulation process, the MatLab GA solver setup assumes a population size limited to 20 candidates, a generation stall limited to 10, whereas the tolerance is set to $1E^{-}10$, the mutation function is set to *@mutationdadaptfeasible* and the hybrid function is set to *@fmincon*. The genetic algorithm is therefore used to locate the area around the global minimum, whereas the *fmincon* operator locates the actual minimum within that area, improving the convergence time of the algorithm.

4.2 Multi apartment revamping model

The more complex scenario, whose model is described in Section 3.2.2, has been used to evaluate the effects resulting from the structure of the system as well as those resulting from the management time frame, and from dynamic pricing.

The resulting framework is intended as a design aid, meant to evaluate the performance of different alternative Micro Grid configurations. It follows that, given the needs and habits of the user, the designer can identify the best performing solution early in time, without the need of a detailed design of each candidate solution. To evaluate the performance of the framework, a real life Micro Grid environment has been analysed in order to identify its building blocks. Thereafter the identified blocks have been modelled and thus 8 different configurations, with growing complexity, have been derived from the original structure.

In the first configuration, only production and demand of electricity, hot water, heating and cooling has been evaluated. In this case, therefore, no storage device has been included. The set-up is composed of three electrical blocks, a hot water block, a heating block and a cooling one, but none of them is able to store energy. The second configuration differs from the first one, by the addition of three thermal energy storage devices, which pertain hot water, heating and cooling respectively. The third configuration, on the other hand, differs from the first one by the addition of two electrical energy storage devices. Thus, two electrical blocks are able to store energy, whereas the remaining electrical block, and the three thermal ones do not provide any form of storage. The fourth configuration includes two electrical energy storage devices and three thermal energy storage devices. The four configurations already described have, as common traits, that the building blocks are independent from each other, meaning that the energy surplus from each block is not routed to the others. Also, when the electrical storage is used, a management policy, based on the constraints presented in Eq. (3.37) and (3.38), is used. As said, this policy is meant to prevent the storage from being recharged from the main grid, and from being discharged if the residual capacity is below 65%. Also, the fourth configuration presented matches the actual set up of the real environment used as a reference.

To evaluate additional improvements, few incremental changes have been applied. From configuration 5 onwards, for instance, the constraints presented in Eq. (3.37) and (3.38) have been removed. Thus, configuration 5 differs from configuration 4 in the electrical storage management policy. In configuration 6, in addition, the management of hot water and heating has been integrated. Then, in configuration 7, the integration of electrical and thermal management is included, as presented in Eq. (3.34). Notably, the three electrical blocks remain independent in each configuration. As such, the surplus from an electrical block is never routed to another electrical block, therefore, even in configuration 7 only the heat pump and the water boiler can take advantage of the electricity surplus. Since in the configuration 7 each electrical block remain independent from the others, as a mean to evaluate a further integrated design, the configuration 8 is also proposed. In the case of configuration 8 the electrical subsystem has been modelled as a single phase circuit, meaning that the solar energy production is the sum of the production of the three panels, the energy demand is the demand of the 6 apartments, and the electrical storages, combined, support the whole demand and production of the building. As such, in configuration 8, electricity demand and production are managed at building level rather than at electrical block level.

The proposed configurations are briefly listed as follows.

Configurations under test

1. storage devices not used.
2. thermal storage
3. electrical storage with a conservative electrical storage management policy
4. thermal and electrical storage with a conservative electrical storage management policy
5. thermal and electrical storage
6. thermal and electrical storage with the integrated production of hot water and heat
7. thermal and electrical storage with the integration of thermal and electrical management
8. thermal and electrical storage with the integration of thermal and electrical management at building level

The first test is based on historical data. The input is composed of the electrical and thermal demand, the latter involving hot water, heating and cooling needs. Additionally, photovoltaic energy production, thermal solar energy production and residual stored energy are assigned to the system. For instance the simulation covers an entire year spanning from 1 November 2012 to 31 October 2013. For each hour a simulation step is carried out.

Test 1 algorithm

1. the simulation routine receives the input data for the t-th slot and simulates the energy management;
2. the simulation routine computes the energy fluxes at the t-th time slot, and the residual stored energy at the end of the same time slot;
3. the energy fluxes are recorded along with their costs and the residual stored energy;
4. the input set of the $(t+1)$-th time slot is composed and a new simulation step is carried out.

Also, at the beginning of the first step, the thermal storage devices are fully charged, the electrical storage devices are at 50% of their nominal capacitance level.

After the described simulation, to evaluate if the suggested framework is also able to plan the activity of the storage early in time, an additional test has been carried out. In this case the historical data, as regards photovoltaic energy production and thermal solar energy production, has been replaced with its forecast counterpart. For instance, since the forecast is "hour ahead" , the test requires two simulations per hour.

According to what stated above, the same framework can be of use, not only to evaluate the system performance ahead of time, but also to operate an actual resource management. In the forecasting case study then, since an offline planning is carried out, it is also possible to search for an optimal scheduling scheme.

Test 2 algorithm

---

1. at the t-th time slot, the energy management, based on forecast data, is carried out;
2. the planned storage activity (charge and discharge), at the t-th time slot, is then recorded, along with the expected costs;
3. the recorded storage activity is assigned, as an input, along with historical data, to execute the planned management;
4. the resulting energy cost, at the t-th time slot, is computed, and recorded along with the residual stored energy;
5. the process is repeated for the next time slot.

Notably, the distinction of the planning phase and the execution phase it is not simply limited to the way the framework is used. In order to separate properly the two phases, even the simulator configuration shall differ from the manager set-up, so that planning and execution may not overlap. In regard of this, it is possible to conclude that a policy is an essential element of a decisional process, and thus its role, within the management process, is clear. Conversely, however, its role during the execution process is dubious, since the execution of a plan shall be carried out without question. If that is not the case, in fact, the execution of the plan, and its management, overlap, leading to management conflicts. For these reasons, in order to distinguish clearly the management phase and the execution phase, the constraints presented in Eqs. (3.37) and (3.38), are applied to the configurations 3 and 4 during the management process, but they are removed during the execution phase.

In both tests, concerning the details of electrical configuration, three electrical blocks have been modelled, each of them featuring a photovoltaic panel of about 6 kW of peak power, and two of them including an electrical storage device of 5.8 kWh each. Concerning the thermal configuration, a hot water block has been modelled including a solar thermal panel, an electrical water boiler of about 15 kW and a water tank of 1300 litres. The water tank has been modelled assuming a ΔT of about 34°C between the temperature of the output water and the temperature of the input water, thus 50 kWh of maximum stored energy has been assumed for the water tank. Relating to the heating and cooling blocks, a heat pump of about 16.6 kW has been used. The water storage, of about 400 litres, has been modelled assuming a ΔT of about 10°C, leading to roughly 7 kWh of maximum stored energy. For simplicity sake, two separated storage devices and heat pumps have been assumed, one set for heating and the other for cooling, thus separating the heating management from the cooling management. As regards the electricity price, a two tiered tariff has been assumed, distinguishing peak hours from 8 a.m. to 7 p.m., from off peak hours. In particular the peak hours tariff amounts to 0.138 € , whereas the off peak hours tariff amounts to 0.129 € .

4.2.1 Dynamic pricing

In the present work, along with the flat rate pricing scheme presented above, a dynamic tariff is proposed to investigate the effect of dynamic pricing along with the different management time frame.

Due to the lack of dynamic pricing offers from Italian service utilities, a dynamic profile has been computed based on the Italian energy market prices[1].

[1] https://www.mercatoelettrico.org/EN/Download/DownloadDati.aspx

Dispatch costs, ancillary costs and taxes have been accounted based on the information provided by the Italian Regulatory Authority for Electricity Gas and Water[2].

In particular, since the residential environment used as a reference has a single point of common coupling (PCC), despite the six apartments, dispatch costs, ancillary costs and taxes applied to business users have been used. This choice has been also deemed consistent with the dynamic pricing scenario, since to business users single tiered costs are applied, whereas to residential user three tiered costs are applied.

Over the reference time frame the average of the two tiered tariff amounts to 0.00013354 € per Wh, whereas the average of the dynamic price amounts to 0.00012988 € per Wh.

To generate the dynamic price profile forecast, a day-ahead forecast has been used. For instance, the value corresponding to the current hour of the current day, has been forecast based on the values corresponding to the same hour of the previous seven days in order to better catch the weekly periodicity. Specifically, the price profile forecast, over the entire reference set, has been performed on hourly basis.

The forecaster of choice is a Single Layer FeedForward network (SLFN). Its training has been carried out by mean the kernel based ELM fast learning algorithm, with a radial basis function (RBF) kernel. The number of hidden neurons does not need to be known in advance. The implementation is based on the code[3] provided by Hong-Ming Zhou and Guang-Bin Huang[4].

The two day ahead forecasters used as reference have been implemented as Multilayer Perceptron (MLP) and NARX neural network respectively. Both have been implemented, by means of the MatLab Neural Network Toolbox, as a 2 layer networks, the hyperbolic tangent has been adopted as activation function of the neurons, and the Levenberg-Marquardt algorithm has been used in the training phase.

To set-up the forecasters, several test runs have been made, with each technique, in order to find the most performing configuration. For instance, since the prepared data sets date back to 2010, a price profile from 1st November 2010 to 31st October 2011 has been used to train the neural networks, whereas a price profile from 1st November 2011 to 31st October 2012 has been used to validate the neural networks. Then the price profile from 1st November 2012 to 31st October 2013 has been used to generate the price forecasts and to evaluate performance of the neural networks.

In the case of the SLFN forecaster, the best performance has been achieved

[2]http://www.autorita.energia.it/it/inglese/index.htm
[3]http://www.ntu.edu.sg/home/egbhuang/elm_kernel.html
[4]http://www.ntu.edu.sg/eee/icis/cv/egbhuang.htm

assuming 0.082 as the value of the kernel parameter, which is referred to as *kernel parameter* σ, in [29]. In regards to the MLP forecaster, in the best performing configuration the neurons have amounted to 15 in the hidden layer and to 1 in the output layer. In the case of the NARX neural network, the best performing configuration has been obtained with 10 neurons in the hidden layer and 1 neuron in the output layer. In this case, also, the delay time steps amount to 0 in the case of the input, and to 24 in the case of the feedback.

4.3 Results

Based on the proposed scenarios and set-up, the evaluation of the management process has been carried out by executing the management process to compute the activity schedule of the Micro Grid environment and, then, using the activity schedule as an additional input to the MILP problem, the activity of the Micro Grid environment has been simulate to evaluate the Micro Grid performance.

4.3.1 The task scheduling and thermal model

By means of the data forecast, a task time table is computed. Under these circumstances, the expected energy demand of the heat pump is the amount computed using the outdoor temperature forecast, whereas the expected energy production is computed with the solar irradiance forecast as a basis. After the time table is retrieved, the framework is used as a simulator. The returned energy monetary balance, which results from the execution of the task, accounts the actual energy demand of the heat pump, the actual energy production and the actual prices, thus it provides the actual energy cost, rather than the expected one. Since the forecast always include a prediction error, the result is almost certainly sub optimal, thus said result will be addressed as "Sub Optimal" in the reported plots and figures.

The performance reference is obtained from an ideal case that assumes that an exact prediction is possible. Whenever the prediction error can be considered equal to zero, the data forecast will match the historical counterpart. In this scenario, then, it is possible to search for the task time table using the historical data rather than the forecast counterpart. As a result, the actual energy monetary balance obtained through simulation will match the expected cost, computed by the framework during the search of the task time table. As such, since the algorithm will provide the time table that minimize the energy cost, the optimal result is obtained. For this reason, the obtained result will be referred to as "Optimal" in the reported plot and figures.

The comparison of optimal and sub optimal results will provide the means to evaluate the relationship, between prediction error and scheduling performance, in term of energy net cost.

An additional reference case, addressed as "baseline scenario", consists of a domestic environment, without any energy management feature, nor energy storage facility and without task scheduling abilities. In this environment the energy produced by the solar power plant is sold directly to the main grid, whereas the tasks are executed in the first available time slot. A time slot is assumed to be available if it falls within the time frame given to the task and if the task can be executed without exceeding the maximum power allowed by the building wirings. For this scenario no plot will be presented.

Concerning the realism of the scenario two aspects shall be accounted. The first one is related to the assumption that, within the environment, the thermal regulation depends on a thermostat. The thermal model, provided by the framework, is only meant to compute the energy demand. If historical data is used an input, the actual energy demand will be obtained. If forecast data is used instead, the expected energy demand is obtained.

The second aspect pertains the realism of the thermal regulation. In order to evaluate the framework performance, a strict thermal regulation is used to enforce the heat pump activity. In real life however, thermal regulation is usually not used all year around, thus a certain degree of realism is discarded in the current work.

The first test to be carried out is based on the data set referred to as "day 1". The error prediction of the data forecast, within this set, is less then 10%. As said, first the energy demand of the heat pump is computed. This routine depends on the outdoor temperature, the thermal constraints and the building thermal behaviour. For the optimal case, the temperature profile is reported in Fig. 4.5, whereas for the sub-optimal case the resulting indoor profile is reported in Fig. 4.6. By comparing the two, the effect of the prediction error is clear. The difference becomes even more apparent if the energy demand, reported in Fig. 4.7, is evaluated.

The comparison of the profiles highlights that the effects of the prediction error highly depend on the temperature constraints. Since no thermal regulation is required without thermal constraints, the prediction error is unable to propagate during the time slots where temperature requirements are not assigned. Also, the indoor temperature is bound to follow the outdoor temperature, and the thermal regulation is only required to maintain the indoor temperature within the given range. Therefore, if both the outdoor temperature and the corresponding forecast profile fall into the assigned range, no regulation will be necessary, thus the prediction error will not propagate. In the current scenario, a 1 °C temperature range was chosen, thus the error propagation occurs, if a wider range where to be chosen, however a different situation may present

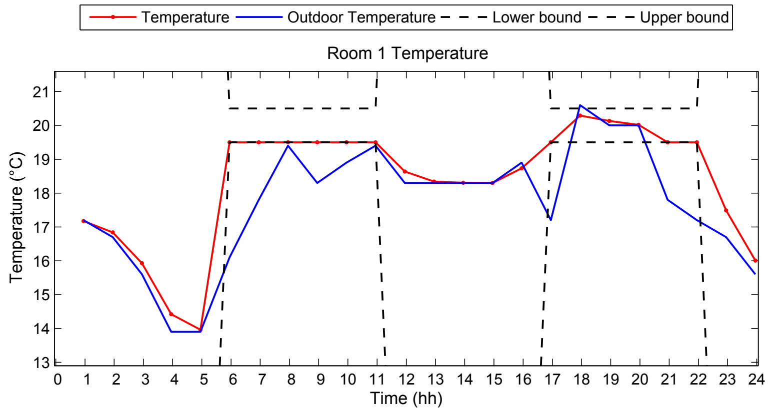


Figure 4.5: Data set "day 1": Room temperature profile against outdoor temperature and constraints

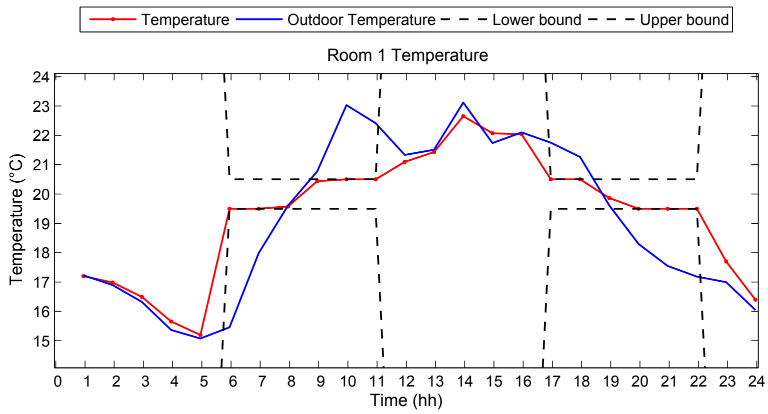


Figure 4.6: Data set "day 1": Room temperature profile against forecast outdoor temperature and constraints

itself.

Recalling that the task scheduling process accounts the heat pump activity, the energy price and local production, it is also possible to conclude that the prediction error, originating from the outdoor temperature forecast, affects the time table indirectly, whereas the prediction errors originating from both the solar irradiation and electricity price forecasts operate directly. On this subject, while a few educated guesses can be made in very specific cases, a general model of the interaction among the prediction errors does not exists. Due to their stochastic nature, in fact, it is not possible to assess beforehand the error amount, moreover in each time slot. In turn then the interaction among error, that is if the errors sum up or cancel each other out, is not known.

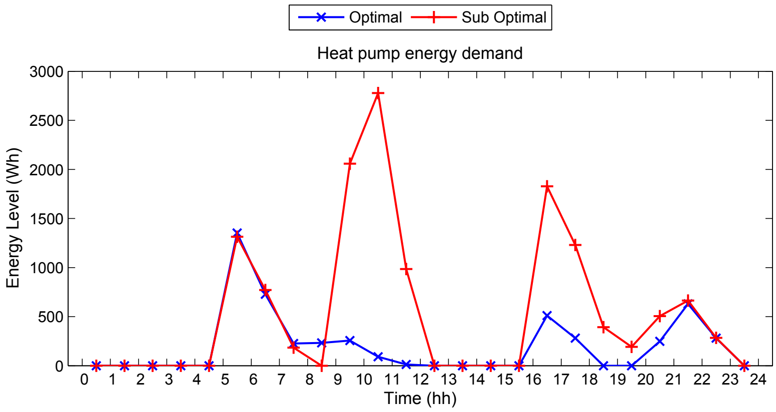


Figure 4.7: Data set "day 1": Hourly energy amount consumed by the heat pump

Although, from a general point of view, the way the task time table is distorted by the prediction errors is not predictable, it should also be noted that the time slot allocation is also subject to the system constraints. The maximum power allowed by the house wirings shall be accounted, and thus the fact that the number of tasks that can be executed at once is limited. The sequential nature of the task is also to be considered, and thus the fact that the tasks cannot be executed in any order. In addition, it shall be noted that each task has a given time frame. In other words, the robustness of the scheduling process, with respect to prediction errors, may increase the more tasks are to be scheduled, since the degrees of freedom of the allocation process are reduced. Nonetheless, it shall also be observed that, when energy demanding tasks are involved, even slightly distorted time schedule may lead to significant performance drop.

For instance, if the optimal time schedule, reported in Fig. 4.8, is to be compared to the sub optimal time schedule, reported in Fig. 4.9, it is possible to observe that only the entries marked as "Task 3" and "Task 8" are actually affected by the forecast error.

By simulating the task execution using the computed task schedule, the actual energy management profiles can be evaluated. Concerning the reference case, being optimal, the energy allocation produce the minimum energy cost. When the sub optimal case is considered, obviously, a less efficient result is achieved. It may also worth to mention, in this case, that since the optimal case, being optimal, guarantees the best achievable performance, the sub optimal setup can, at most, achieve the same result. That is, under no circumstances better performances can be achieved.

Concerning the energy costs, reported in Fig. 4.10, the "Optimal" entry

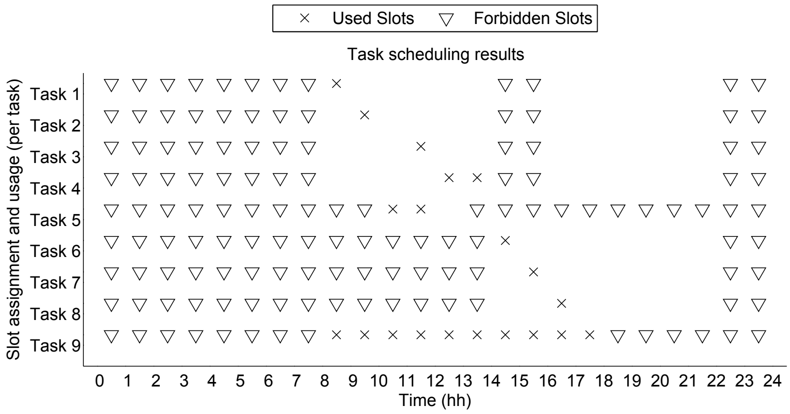


Figure 4.8: Data set "day 1": Optimal task scheduling activity

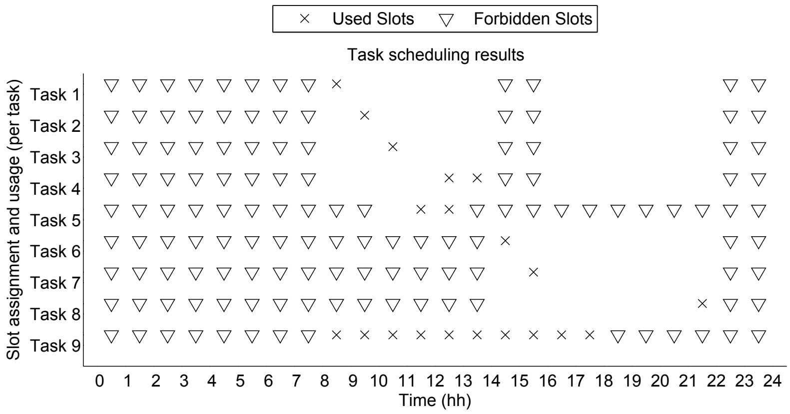


Figure 4.9: Data set "day 1": Sub Optimal task scheduling activity

produces a total energy cost of 0.21$, whereas the "Sub Optimal" counterpart leads to a total energy cost of 0.29$. Concerning the energy income, reported in Fig. 4.11, the total income of 0.1$ is obtained in both cases. The energy minimum net cost (reference case) is thus 0.106$, the expected net cost for the forecast case is 0.69$, whereas the actual net cost for the forecast case is 0.19$.

Also the stored energy level can be evaluated (Fig. 4.12), and a better insight of the scheduling process can be gained. In particular, after 20.00 the battery is fully charged, meaning that the different slope in the energy cost plots, after 20.00, only depends on the task allocation. Also, at 8.00 the optimal time schedule allows the battery to recharge to full level, whereas the sub optimal time table prevent the recharging phase till 10.00 and requires an additional discharge at 13.00, thus increasing the battery stress.

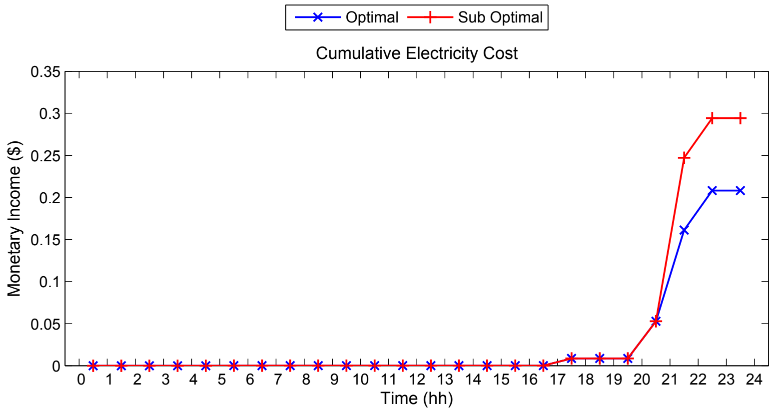


Figure 4.10: Data set "day 1": Energy cost over time

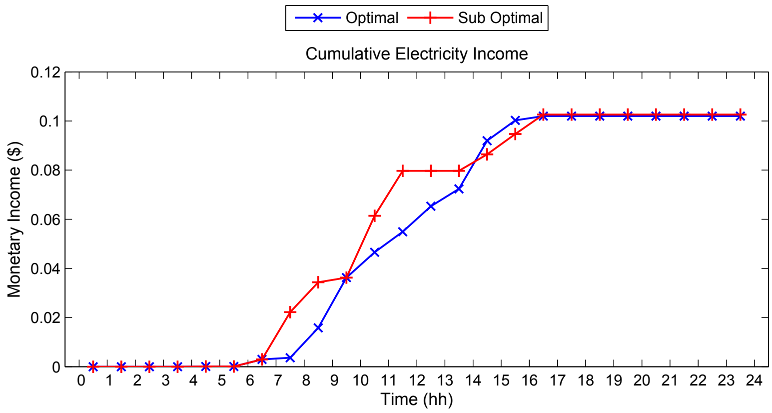


Figure 4.11: Data set "day 1": Energy production income over time

As a further mean of comparison, for a baseline scenario, the energy cost will amount to 5.31$, whereas the energy income would amount to 0.34$. The net energy cost will amount to 4.97$ meaning that the optimal case net cost will amount to 2.1% of the baseline net cost, whereas the sub-optimal net energy cost will amount to 3.8% of the baseline net cost.

By evaluating the scheduling process by means of the "day 3" set-up, a prediction error higher than 20% and lower than 30% will be accounted. Similarly to the previous case, concerning the heat pump expected energy demand, the forecast case will differ from the reference case (Fig. 4.13). The difference will be much more remarkable, since the prediction error is greater.

Pertaining the computed task schedules (Figs. 4.14 and 4.15), since the prediction errors originating from both the irradiance and price forecast are

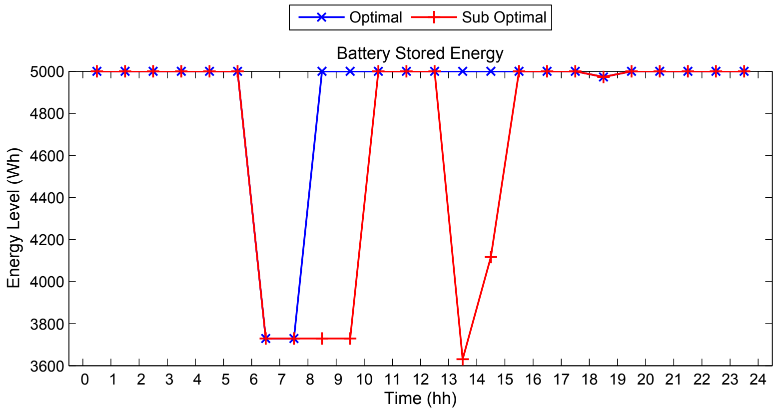


Figure 4.12: Data set "day 1": Battery energy level over time

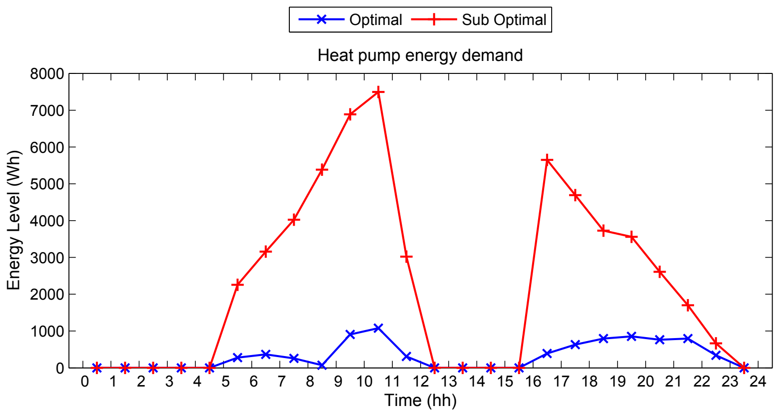


Figure 4.13: Data set "day 3": Hourly energy amount consumed by the heat pump

also greater than the previous case, the difference between the optimal case and the sub optimal counter part are much more evident. This result, also, suggests that the maximum and minimum errors, thus accounted with their sign, may be a set of indexes much more meaningful, towards the distortion of the task schedule, with respect to the MAE.

By simulating the task execution, based on each of the task schedule, the cumulative energy costs, depicted in Fig. 4.16 are obtained. The energy income amounts, on the other hand, are reported in Fig. 4.17, whereas the energy storage level over time is presented in Fig. 4.18.

Although the involved values may differ from the previous case, the same conclusion still holds. For instance, the net energy cost of the optimal case

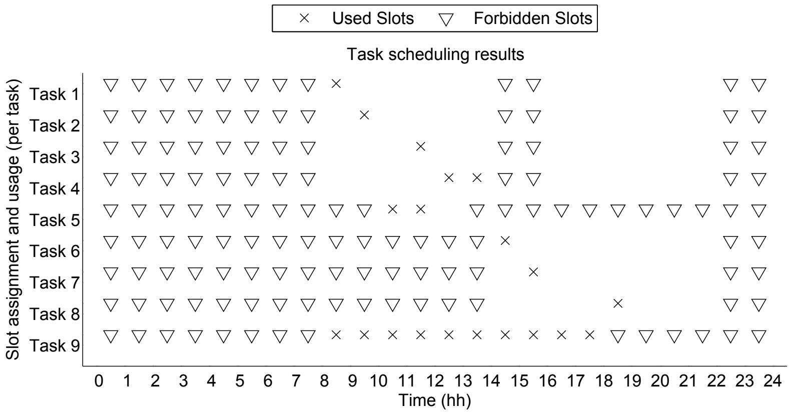


Figure 4.14: Data set “day 3”: Optimal task scheduling activity

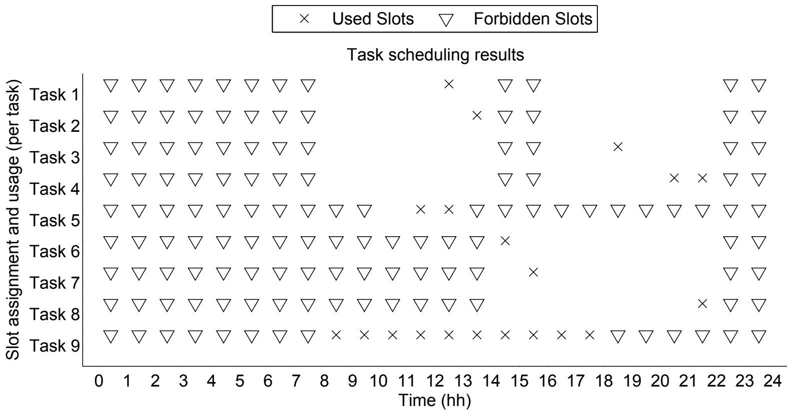


Figure 4.15: Data set “day 3”: Sub Optimal task scheduling activity

amounts to 1.07$, the expected net cost of the sub optimal case amounts, on the other hand, to 11.718$, while the actual net cost of the sub optimal scheduling amounts to 1.73$.

If the baseline scenario is taken into account, it is possible to observe that the total energy cost amounts to 8.11$ and the total energy income amounts to 0.21$. Since the net energy cost of the baseline scenario amounts to 7.9$, the net cost of the optimal case amounts to 13.5% of the baseline net energy cost. The net energy cost of the sub optimal case, on the other hand, equals the 21% of the baseline net cost.

Based on the proposed evaluations, it appears that, while the prediction errors may affect the scheduling process, thus impairing the allocation of the tasks and the energy management, the performance loss is not directly related

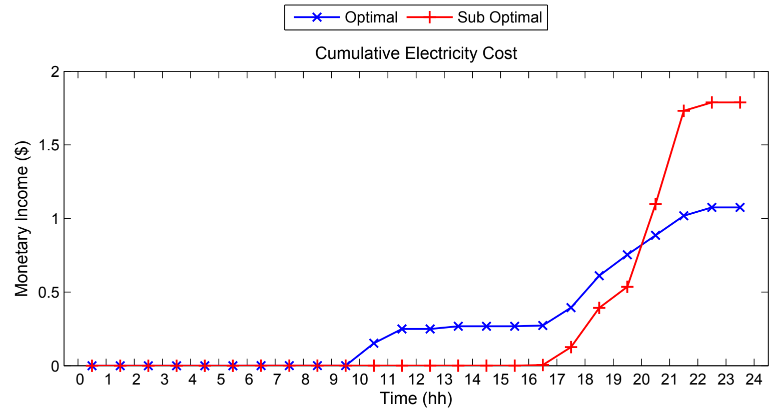


Figure 4.16: Data set "day 3": Energy cost over time

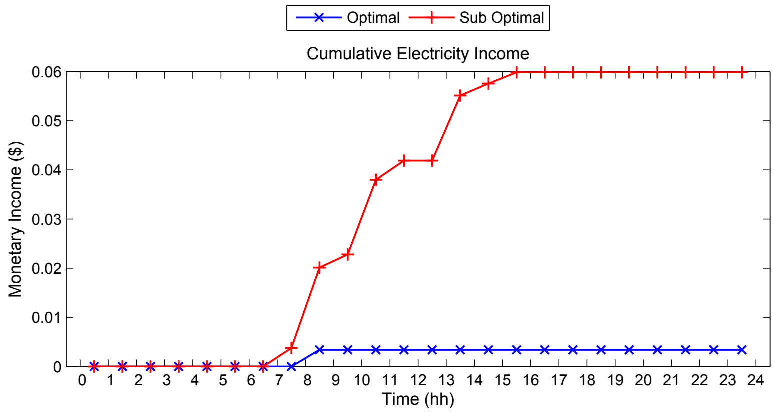


Figure 4.17: Data set "day 3": Energy production income over time

to the amount of said errors nor their interaction. Clearly, the forecast values used by the scheduler may promote or prevent the allocation of each time slot to the given tasks.

In this perspective, of course, the prediction errors may alter the time slots allocation in any way. However, if the optimal time slot for a given task can be addressed as the i-th time slot, and if a sub-optimal time slot assigned to the said task can be addressed as j-th time slot, it can be concluded that the performance loss actually depends on the difference between the i-th price and j-th price, that is the prices of the i-th time slot and the j-th time slot respectively.

Also, since the sub-optimal task allocation depends on both constraints and forecasts, due to the lack of correlation among the involved entities, it may be

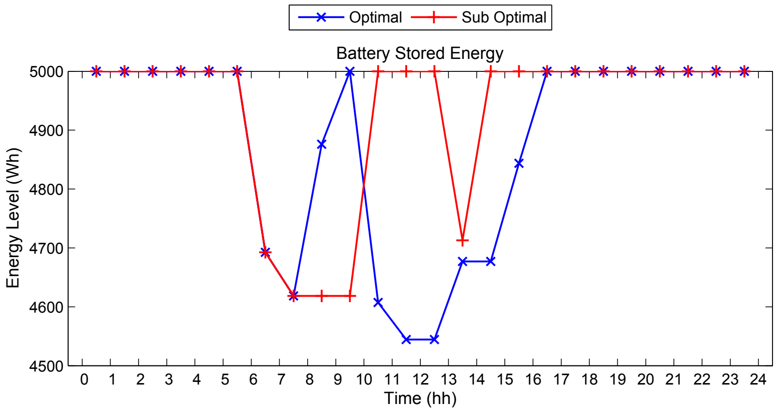


Figure 4.18: Data set "day 3": Battery energy level over time

safe to assume that the sub-optimal allocation distortion is actually a random process, at least if a prediction error exists. On this subject, the "day 2" case, in which the prediction error is higher than 10% and lower than 20%, appears to be particularly meaningful.

In fact, if the optimal task schedule (Fig. 4.19) is compared against the sub optimal task schedule (Fig. 4.20), it is possible to notice that the "day 2" case is quite similar to the "day 3" one. However, in the "day 2" case, the net energy cost amounts to 32.218$ for the optimal and 32.235$ for the sub-optimal task schedule, thus there no meaningful difference in terms of energy cost between the optimal and the sub-optimal case.

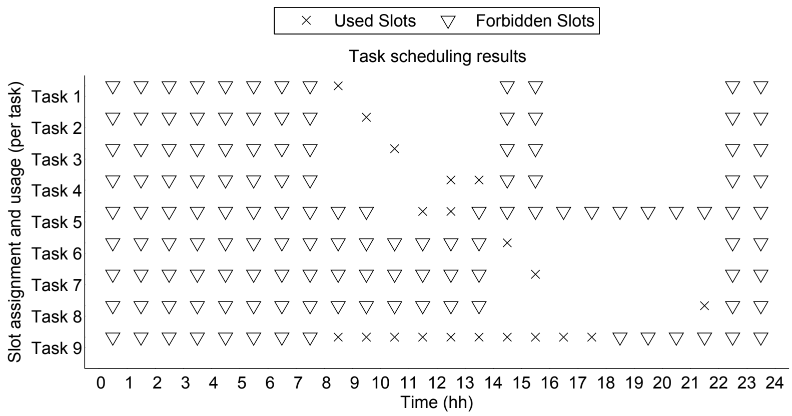


Figure 4.19: Data set "day 2": Optimal task scheduling activity

This aspect may also be seen as the reason why the scheduling process seems to be fairly robust against prediction error, in the sense that the entity of the

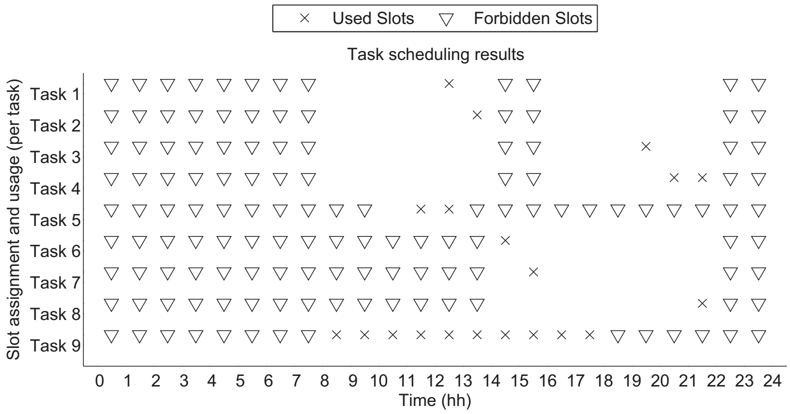


Figure 4.20: Data set "day 2": Sub Optimal task scheduling activity

performance loss does not seem directly bounded to the entity of the prediction error. Moreover, even if the performance loss is to be accounted, the improvement over a baseline scheme still remains remarkable.

4.3.2 Multi apartment revamping model with one hour ahead management

By means of the multi apartment model, the performance of the energy manager with a one-hour ahead management scheme has been investigated.

Table 4.4: Energy management, based on historical data, for all addressed system configurations. The identification number is based on the entries listed in Section 4.2.

		Configurations							
		1	2	3	4	5	6	7	8
Energy cost due to electrical blocks	(€)	618.85	618.85	488.50	488.50	493.42	493.42	493.03	441.08
Energy cost due to thermal blocks	(€)	1356.69	1318.76	1356.69	1318.76	1318.76	830.59	585.25	594.30
Energy overall cost	(€)	1975.54	1937.61	1845,15	1807,26	1812.18	1324.01	1078.28	1035.38
Purchased energy amount	(kWh)	14663	14682	13964	13681	13715	9874	7426	7926
Surplus energy amount	(kWh)	20630	20630	17778	17778	17396	14936	13666	11181
Residual electrical energy	(Wh)	0	0	3334	3334	3396	3396	3396	6267
Residual thermal energy	(Wh)	0	1750	0	1750	1750	1750	1750	1750
Residual refrigerating energy	(Wh)	0	1750	0	1750	1750	1750	1750	1750
Residual hot water energy	(Wh)	0	12500	0	12500	12500	12500	12500	12500
Electrical storage invest cost	(€)	0	0	30740	30740	30740	30740	30740	30740
Thermal storage invest cost	(€)	0	2000	0	2000	2000	2000	2000	2000
Total invest cost	(€)	0	2000	30740	32740	32740	32740	32740	32740
Electrical storage maintenance yearly fee	(€)	0	0	500	500	500	500	500	500
Thermal storage maintenance yearly fee	(€)	0	100	0	100	100	100	100	100
Storage maintenance total yearly fee	(€)	0	100	500	600	600	600	600	600

In the first test, the energy management is simulated, based on historical data, for each of the proposed configurations. The performance is summarized in Table 4.4, which reports, in each column, the results of the corresponding configurations.

The reported amounts present, in order, the cost of the energy purchased to supply the electrical blocks and the cost of the energy purchased to supply the thermal blocks. Next, the overall energy purchased amount, and surplus are reported. At last, the residual energy stored within each storage device is presented.

The comparison of the results, within the synoptic table is quite straightforward. In the 1st configuration, the storage devices are not included, therefore the energy surplus is discarded immediately. As a consequence, the system is forced to purchase additional energy in order to match the demand, either electrical or thermal, each time the local production is lacking.

In the 2nd configuration, on the other hand, the availability of the thermal storage grants the system the ability to preserve the thermal energy surplus thus lowering the electrical needs of the thermal system. In this case, however, since the solar collector only support the hot water production, the savings are quite limited and amount to 38€ per year.

In the 3rd configuration, the storage devices support the activity of two, out of the three electrical blocks. Due to the higher energy demand and production, therefore, an enhanced performance is possible, with respect to the 2nd configuration. In particular 130€ can be saved with respect to the 1st configuration.

Also, since each block operates independently from the others, when both thermal and electrical storage devices are used, the saving from both the hot water block and the electrical blocks concurs to reduce the overall energy bill, as shown from the results obtained by the 4th configuration.

The 5th configuration proves to be a little less performing than the 4th one. In this case, in fact, the savings provided by the addition of the electrical storage amount to 125 €, with respect to the reference case, represented by the 1st configuration. This is likely due to the fact that, without accounting the energy surplus in the objective function (Eq. (3.30)), the manager is not forced to store the energy surplus, thus achieving a sub optimal management of the energy surplus. On the other hand however, the absence of the constraints, described in Eq. (3.37) and (3.38), also grants the energy manager with two additional degree of freedom, thus improving the responsiveness of the storage management. Additional tests, not shown here for the sake of conciseness, reveal in fact that, by accounting the energy surplus within the objective function, an additional saving of about 20 € can be achieved.

The 6th configuration shows that the integration of hot water production and heating can greatly enhance the efficiency of the system. In fact, while the solar collector may support the heat production, thus reducing the energy required to heat the building, the heat pump can provide the extra heat required to support hot water production in place of the water boiler. In results, roughly 488 € can be saved with respect to the 5$^{\text{th}}$ configuration due to a more efficient management of thermal resources. The overall savings, therefore, rise up to about 650 € if compared against the performance of the 1$^{\text{st}}$ configuration.

In the same perspective, the integration of electrical and thermal management, with the ability to route the electricity surplus to the heat pump and the water boiler, further improve the ability of the system to exploit the renewable resources. For instance, the 7$^{\text{th}}$ configuration, achieves an additional reduction of the overall bill, of about 245 €. If compared against the results of the 1$^{\text{st}}$ configuration, the purchased energy amount is practically halved, whereas the energy surplus is lowered to roughly the 60% of its original amount.Similar results are achieved if the 8$^{\text{th}}$ configuration is used. When the electrical energy is managed from the building standpoint, a different allocation of the resources may be possible. As such, with respect to the 7$^{\text{th}}$ configuration, a further reduction of the energy cost due to the electrical block is achieved, saving about 52 €. At the same time however, since the energy surplus routed to the heat pump is also lowered, the energy cost due to the thermal blocks increase slightly, requiring an extra of about 9.05 €. The overall saving, therefore, amounts to 43 €. In this case, then, the algorithm is able to promote the energy purchase during the off peak hours, thus achieving a lower energy bill in spite of an increased amount of purchased energy. This management strategy entails an enhanced self-consumption of the renewable energy produced by the PV system, as confirmed by the decrease of the surplus energy amount.

While the results proposed in Table 4.4 are collected at the end of a simulation that cover an entire year time span, partial results have been collected during the entire simulation process. In Figs. 4.21 and 4.22 respectively, the energy cost due to the electrical blocks and the energy cost due to the thermal blocks are proposed.

At a glance, although the plots depict the same situation presented within Table 4.4, they also reveal that the actual results are consistent through the entire simulation. In addition, it is possible to assess in which way the costs increase through the year, and thus the ability of each configuration to save energy. As such, in Fig. 4.21 it is possible to observe that the energy demand of the electrical blocks remains roughly the same through the year, and therefore the cost increase, over time, is almost linear.

With respect to the energy demand of the thermal blocks in Fig. 4.22, in the other hand, it is possible to observe four seasonal changes, since the simulated time interval begins in November. In the first quarter of the plot, which roughly correspond to the winter season, the energy cost and demand increase with a sustained rate, due to the heat demand of the building. In the second quarter of the plot, that is the spring season, nor heating nor cooling are required, thus the cost remains almost constant. In summer, again, the cost over time shows a fast rate of growth that decreases in the last quarter. Also, the less efficient the configuration, the higher the cost rise over time.

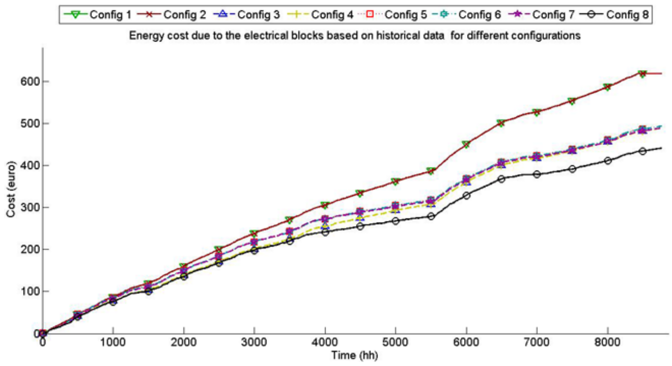


Figure 4.21: Energy cost due to the electrical blocks. The identification number is based on the entries listed in Section 4.2.

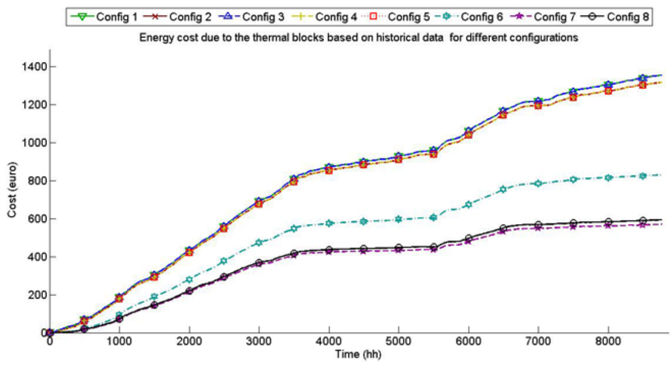


Figure 4.22: Energy cost due to the thermal blocks. The identification number is based on the entries listed in Section 4.2.

In relation to the case of the scheduling approach, the robustness to data uncertainty has been also evaluated. In this case, in fact, the storage activity is planned by means of the forecast data. This means that the prediction error may affect the performance of the system, in the different addressed configurations.

The performed tests, in particular, have shown two distinct trends in connection with the management of electrical and thermal storage devices. In relation to the energy demand due to the thermal blocks, in particular, only minor changes, with respect to the previous test, can be highlighted. The

Table 4.5: Energy management, based on forecast data, for all addressed system configurations. The identification number is based on the entries listed in Section 4.2.

		Configurations							
		1	2	3	4	5	6	7	8
Energy cost due to electrical blocks	(€)	618.85	618.85	623.99	623.99	620.38	620.38	655.20	614.93
Energy cost due to thermal blocks	(€)	1356.69	1349.76	1356.69	1349.76	1349.76	830.59	588.25	616.87
Energy overall cost	(€)	1975.54	1968.61	1980.68	1973,75	1970.14	1450.97	1243.45	1231.8
Purchased energy amount	(kWh)	14946	14895	14979	14928	14891	10976	9470	9384
Surplus energy amount	(kWh)	20630	20630	18791	18791	18382	18382	13453	11164
Residual electrical energy	(Wh)	0	0	10455	10455	10455	10455	10441	13340
Residual thermal energy	(Wh)	0	1750	0	1750	1750	1750	1750	1750
Residual refrigerating energy	(Wh)	0	1750	0	1750	1750	1750	1750	1750
Residual hot water energy	(Wh)	0	12500	0	12500	12500	12500	12500	12500
Electrical storage invest cost	(€)	0	0	30740	30740	30740	30740	30740	30740
Thermal storage invest cost	(€)	0	2000	0	2000	2000	2000	2000	2000
Total invest cost	(€)	0	2000	30740	32740	32740	32740	32740	32740
Electrical storage maintenance yearly fee	(€)	0	0	500	500	500	500	500	500
Thermal storage maintenance yearly fee	(€)	0	100	0	100	100	100	100	100
Storage maintenance total yearly fee	(€)	0	100	500	600	600	600	600	600

electrical management however has shown a significant difference.

The 1st configuration, since it lacks any form of storage, has achieved the same results shown in the previous test. The 2nd configuration, although to a smaller degree with respect to the previous test, is still able to improve its performance with respect to the 1st one. The 3rd configuration, conversely not only has not provided the expected benefits, but also worsened the situation with respect to the reference case, although to a small degree.

As a result, in the 4th configuration, by integrating both electrical and thermal storage, the overall performance remain almost unchanged with respect to the reference case. In the 5th configuration, the adoption of a less conservative management policy, partially ease the problems originating from the prediction error.

In the 6th configuration, the cost, due to the energy demand of the thermal block, remains almost unchanged with respect to the previous test, thus confirming that the thermal storage subsystem is fairly robust against prediction errors. This result is likely connected to the fact that a negligible thermal loss has been assumed to model the thermal storage, whereas a quite significant self discharge has been applied to the electrical storage. As such, the prediction uncertainty affects the activity time of the storage and therefore influences the storage management strategy.

From this perspective, the results achieved by the 7th configuration appear quite clear. The thermal blocks, in fact, remain almost unaffected by the prediction error, and no significant difference, with respect to the previous test, is present in relation to the energy cost due to the thermal blocks. The energy cost due to the electrical blocks, on the other hand, shows a further increase with respect to the 6th configuration. Since the energy surplus is to be routed towards the thermal blocks, the management of the electrical storage requires an increased accuracy, and thus the process becomes less robust against prediction errors.

On the other hand, the 8th configuration seems to be the most robust among the others under test. Indeed, differently from the 7th configuration, the energy cost saving due to the thermal blocks is not paired with an increased energy cost due to the electrical blocks. Rather, although limited, even the energy cost due to the electrical blocks is reduced. In the light of this, 8th configuration allows achieving the best performance within the proposed set of tests.

In conclusion, from the first test set it is possible to observe that the ability to store both the thermal and the electrical energy usually improves the performance of the energy management, even more so if the inclusion of storage devices is complemented with the integration of the different blocks within the system. In particular, the comparison of the 7th and 8th configuration against the 4th configuration, which is characterized by the same constraints for storage activity actually adopted in the Leaf House, suggests that there is room for new design choices within the energy management subsystem, and that these choices are likely to improve the related efficiency. The second test set results highlight that while the thermal energy management shows a fair amount of robustness against the error prediction, the management of the electrical storage can be improved based on this perspective. Therefore, future efforts will be targeted on this direction. For instance, the comparison of the 5th configuration against the 4th one, suggests that an appropriate choice of the management policy, in dependence on the expected liability of the forecaster on hourly basis, may solve the issue. Nevertheless, the last two configurations, proposing an integrated management of the electrical and thermal storage capabilities, still allow to achieve a significant reduction of the annual energy cost with respect to other approaches.

4.3.3 Multi apartment revamping model with one day ahead management

The multi apartment model, also, has been used to carry out an additional evaluation test, based on an 1 day ahead management scheme. In this case, also, a dynamic pricing scheme has been included.

In the current section the results of both forecasting techniques and energy and resource scheduling are proposed.

Table 4.6: Forecasting indices: back to back comparison

		MLP model	NARX model	Kernel based ELM model
MAPE	Mean:	5.57	5.57	5.45
	Variance:	0.002	0.01	0.00
RMSE	Mean:	9.33	9.49	8.71
	Variance:	0.02	0.04	0.00

Forecasting techniques

To compare the forecasters performance, 10 forecasts per technique have been generated. Being the most common indices, the percentage MAPE (Mean Absolute Percentage Error) and percentage RMSE (Root Mean Squared Error) have been computed and their mean value and variance are proposed in Table 4.6.

For instance, percentage MAPE and percentage RMSE have been computed as follows:

$$\text{MAPE} = 100 \cdot \frac{1}{n} \sum_{t=1}^{n} \left| \frac{P_t - F_t}{P} \right|, \quad \text{RMSE} = 100 \cdot \frac{1}{n} \sqrt{\sum_{t=1}^{n} \left| \frac{P_t - F_t}{P_t} \right|^2},$$

where, within a year, n is the n-th hour, P_t is the $t-th$ historical hourly price value, F_t is the $t-th$ forecast hourly price value and P is the average of the historical hourly price.

The MLP and NARX forecasters show similar results, although the lower variance of the MLP based forecasts reveals an increased robustness of this method with respect to the NARX network. Most likely, the NARX model has been affected by the lack of information about the weekly and seasonal correlation of the prices. It must be remarked that, omitting the day of the week and the day of the year, to adopt a simpler input set, may have resulted in sub-optimal working conditions for the NARX model. The results of the ELM based forecaster show that, with respect to MLP and NARX based forecasters,

Table 4.7: Energy cost for the 1-hour ahead management scheme, based on historical data and two tiered tariff, for all addressed system configurations. The identification number is based on the entries listed in Section 4.2.

			Configurations							
			1	2	3	4	5	6	7	8
Hourly management	Cost due to electrical blocks	(€)	618.85	618.85	488.50	488.50	493.42	493.42	493.03	441.08
	Cost due to thermal blocks	(€)	1356.69	1318.76	1356.69	1318.76	1318.76	830.59	585.25	594.30
	Overall cost	(€)	1975.54	1937.61	1845.15	1807.26	1812.18	1324.01	1078.28	1035.38

the percentage MAPE index mean value is 2.15% lower, the percentage RMSE index mean value is, respectively, 6.64% and 9.22% lower. The variability, also, is null. Thus, the ELM based forecaster appears to be more accurate, and more robust against the network random initialization.

Hourly management improvements

The first evaluation compares the hourly energy management results against the performance of the previous implementation of the framework, based on the overall cost of the purchased energy. Therefore a lower energy cost represents a better performance.

The scheduling results of the 1 hour ahead management are reported in Table 4.7, whereas the results of the the 1 day ahead management are proposed in Table 4.8. The set-up and parameters of the both implementation remain unchanged.

Regarding the hourly manager, the comparison of the results, reported in Tables 4.7 and 4.8 and labelled: *Overall cost*, reveals that with the exception of configuration 2 and 4, the revised cost function improves the efficiency of the energy management. If the thermal and the electrical contributions are accounted, respectively shown by the entries labelled: *Cost due to the thermal blocks* and *Cost due to the electrical blocks*, it can be observed that the revised cost function does not interfere with the policy of the electrical storage (configurations 3 and 4), it lowers the electrical needs when electrical and thermal subsystems are apart (configurations 5 and 6), whereas it promotes the use of the electrical storage to support the thermal subsystem. It should be reminded that only the electrical storages of configuration 3 and 4 have a management policy applied.

In the case of configuration 2 and 4, as said, the performance has worsened. However, the performance drop affects only the thermal management, whereas the electrical management remains unaffected. This behaviour is due to the

Table 4.8: Energy cost based on historical data and two tiered tariff (ideal scenario with no prediction error), for all addressed system configurations. The identification number is based on the entries listed in Section 4.2.

			Configurations							
			1	2	3	4	5	6	7	8
Hourly management	Cost due to electrical blocks	(€)	618.85	618.85	488.50	488.50	429.27	429.27	599.02	562.96
	Cost due to thermal blocks	(€)	1356.69	1349.79	1356.69	1349.79	1349.79	830.59	430.56	443.86
	Overall cost	(€)	1975.54	1968.64	1845.19	1838.29	1779.06	1259.86	1029.58	1006.82
Daily management	Cost due to electrical blocks	(€)	618.85	618.85	474.38	474.41	426.51	426.51	479.04	427.89
	Cost due to thermal blocks	(€)	1356.69	1354.73	1356.69	1306.28	1306.28	804.05	377.04	390.41
	Overall cost	(€)	1975.54	1973.58	1830.57	1780.69	1732.79	1230.56	856.08	818.30
	Overall cost difference	(€)	-	-4.94	14.62	57.60	46.27	29.30	173.5	188.52

Table 4.9: Energy cost based on historical data and dynamic tariff (ideal scenario with no prediction error), for all addressed system configurations. The identification number is based on the entries listed in Section 4.2.

			Configurations							
			1	2	3	4	5	6	7	8
Hourly management	Cost due to electrical blocks	(€)	631.08	631.08	486.05	486.05	424.13	424.13	596.73	557.45
	Cost due to thermal blocks	(€)	1367.62	1361.32	1367.62	1361.32	1361.32	824.26	425.30	438.78
	Overall cost	(€)	1998.70	1992.40	1853.67	1847.37	1785.45	1248.39	1022.03	996.23
Daily management	Cost due to electrical blocks	(€)	631.08	631.08	470.35	470.42	403.41	403.41	449.53	388.59
	Cost due to thermal blocks	(€)	1367.62	1161.62	1367.62	1160.71	1160.71	717.23	335.58	350.63
	Overall cost	(€)	1998.70	1792.70	1837.97	1631.13	1564.12	1120.64	785.11	739.22
	Overall cost difference	(€)	-	199.70	15.70	216.24	221.33	127.75	236.92	257.01

value assigned to the scaling factors, which is close to the difference between the off peak price and the peak price. Due to that, whenever the heat pump is used, due to the COP, shifting the thermal load is regarded as "not convenient" by the energy manager.

In fact, by shifting the electrical load from peak hours to off-peak hours, a saving can be achieved. The saving per electrical Wh amounts to the difference between the peak-hour tariff and the off-peak tariff (about 9.73e-6 €). Because of the COP of the heat pump, however, the saving per thermal Wh is roughly one third of the saving per electrical Wh. As such, since the storage activity has a "value" per Wh, represented by the scaling factor, higher than the saving, the storage activity is regarded as unnecessary and avoided by the manager. That is to say that, within the framework, the user can define the threshold below which the storage activity is "less convenient" than the savings it produces.

Table 4.10: Energy cost based on forecast data and dynamic tariff (real life scenario with prediction error), for all addressed system configurations. The identification number is based on the entries listed in Section 4.2.

			Configurations							
			1	2	3	4	5	6	7	8
Hourly management	Cost due to electrical blocks	(€)	631.08	631.08	623.65	622.57	576.55	576.55	653.97	605.57
	Cost due to thermal blocks	(€)	1367.59	1361.14	1367.59	1361.14	1361.14	824.22	505.12	522.23
	Overall cost	(€)	1998.67	1992.22	1991.14	1983.71	1937.69	1400.77	1159.09	1127.80
Daily management	Cost due to electrical blocks	(€)	631.08	631.08	553.10	550.30	544.01	543.80	572.24	365.04
	Cost due to thermal blocks	(€)	1367.59	1174.83	1367.59	1189.34	1189.34	726.64	410.74	431.43
	Overall cost	(€)	1998.67	1805.91	1920.69	1739.64	1733.35	1270.44	982.98	796.47
	Overall cost difference	(€)	-	186.31	70.45	244.07	204.34	130.33	176.11	331.33

Since a different choice of the scaling factors would change the presented behaviour, the framework can be used to evaluate different policies, depending on the user needs.

Daily management and hourly management

Regarding the daily energy management, comparing its performance, based on the results presented in Table 4.8, against the performance of the previous implementation (Table 4.7), shows that the configuration 1 does not provide the means to improve the system efficiency. In the case of configuration 2 the unexpected results are even worse than the revised hourly manager (shown above in Table 4.8). On the other hand, in the case of configuration 4, the daily manager achieves a lower cost even in the case of the thermal management, which leads to the conclusion that a daily management allows a better coordination of both thermal and electrical management. In fact, in every configuration from 3 onwards, the daily manager outperforms both the old and the revised hourly managers.

While the already mentioned Table 4.8 refers to a scenario based on historical data and a two tiered tariff, Table 4.9 presents the results corresponding to a scenario based on historical data and the historical dynamic pricing profile. An evaluation of the results reported in Table 4.9, for both configuration 2 and 4, reveals that, since the price changes every hour, and also the price difference is usually higher with respect to the two tiered tariff, the assigned policy does not lead to an unexpected result. Hence, it is possible to conclude that the daily manager outperforms the hourly one in every configurations, and that the improvement grows with the increased degree of integration among the subsystems.

Data uncertainty

Last but not least, the effect of data uncertainty over the energy management has been investigated. In particular, Table 4.10 presents the results based on a scenario that accounts forecast data for both solar production and dynamic pricing. In this case, for each time step, the forecast data is used to compute the schedule of the system activity, the activity of the residential environment is then simulated by means of the same framework, and the environment per-

Table 4.11: Difference of energy cost between historical data based management (Table 4.9) and forecast data based management (Table 4.10), for all addressed system configurations. The identification number is based on the entries listed in Section 4.2.

		Configurations							
		1	2	3	4	5	6	7	8
Hourly management	(€)	0.03	0.18	-137.47	-136.34	-152.24	-152.38	-137.07	-131.57
Daily management	(€)	0.03	-13.21	-82.72	-108.51	-169.23	-149.80	-197.87	-57.25

formance is computed. The simulation uses the historical data and the schedule previously computed as inputs. In each Table the difference between the performance of the hourly and the daily manager, for each configuration, is reported as well.

In this case, the results presented in Table 4.10 are to be compared against the ones presented in Table 4.9. The case of the hourly manager and the case of the daily manager have been addressed separately. The difference between the overall energy costs, for both cases, has been presented in Table 4.11.

Because of data uncertainty, the performance of the manager is sub optimal, i.e. while it is possible to improve the efficiency of the residential environment, it is usually not possible to achieve the minimum operating cost.

At a glance, with respect to the hourly management, the results proposed in Table 4.11 reveal that the losses due to the forecasting errors are consistent, spanning from about 131 € to about 152 €. Also, the fact that the losses affect configuration 3, but not configuration 2, leads to the conclusion that the electrical energy management is more exposed to the forecasting errors. In this scenario, configuration 8 is the only non trivial set-up that suffers the least drop in performance. In term of percentage with respect to the optimal scheduling, the drop in saving ranges from 7.38%, of configuration 4, to 13.41%, corresponding to configuration 7, suggesting that the high variability comes from the increased saving of the configuration, rather than the variability of the drop in performance.

Similar considerations, also, hold true in the case of the daily management, although the saving losses have a wider spread, from about 57 € to about 198 €. In term of percentage the value spans from 6.65% to 25.20%. Configuration 8 appears to be the most robust non trivial set-up, since the loss in performance, is about 57 €, which is half the drop with respect to configuration 4. That is to say that highly integrated systems appear to be less affected by the forecasting errors.

From a general point of view, in the case of the hourly management, the storage activity is planned depending on the expected local production and the local demand in the next hour. Once the activity is planned, the amount of energy routed to or from the storage cannot be adjusted. Hence, if the energy production exceeds the forecast value, the extra energy is not collected, ending wasted. If the production is lower than expected, the energy to be stored will be purchased.

In the case of the daily manager, on the other hand, the storage activity is planned over a time window of 24 hours. Thus local demand and local expected production are accounted for every hour within the time frame. Differently from the hourly manager, however, the expected energy prices are also accounted. In both cases, the cost function reaches its minimum when the storage activity is absent and, at the same time, the energy cost is at its minimum. However, the hourly manager does no plan the energy purchase accounting the prices. On the other hand, the daily manager, by means of the price forecast, can compare the prices, and plan the purchase as to when the price is expected hits its minimum as to lower the cost function. This behaviour, also, has been observed in our earlier work [50].

In fact, the daily manager is affected by the position of the local minima of the price forecasts, which in turn are affected by the forecasting error. In addition, since the relative position of the minima does not depend on the absolute value of the forecasting error, it is not possible to relate the scheduling performance to the forecast accuracy. For instance, the results reported in Table 4.11 do not reveal any obvious relationship.

Clearly, the results shown in Table 4.11 prove that, although sub-optimal, the improvement of the efficiency of the environment is consistent. Nonetheless, in order to assess the impact of the price forecasting error over the scheduling, an index that takes into account the displacement of local minima may be more effective than the widespread MAPE or RMSE.

4.4 Remarks

Based on the proposed evaluation tests, a MILP based energy management approach shows effective results not only as an energy manager but also as a Micro Grid design tool. The linear nature of the optimization problem can support the management of both small and large environment, while the modular nature of the implementation ease the arrangement of the model elements so that different topologies can be simulated and compared.

Although the approach shows an high degree of flexibility, the constrained nature of the problem render the model robust against data uncertainty proving that the proposed approach can be effectively used in a real life environment.

4.5 Future system developments

Although the management approach has proven to be effective in term of optimization abilities and flexibility, it is based on the MatLab environment and therefore can be considered a research tool rather than a real life solution. In order to manage a real life Micro Grid, in fact, several features are still required to overcome the shortcomings of proposed implementation.

On purpose, the proposed approach is used as a basis to develop a framework based the Python programming language. The new implementation is required to integrate additional features such as the ability to automatically arrange the elements of the problem through the selection of the device models from a model database. Thus the correct MILP problem can be generated automatically, for a given topology, without the need to program the model arrangement. Also the ability to collect the input data relating to the environment topology will be included.

Another additional feature is the ability to manage external events, that are generated by the feedback from the system. The model enhancements presented in Chapter 5 are to be included to improve the accuracy of the management.

Chapter 5

Energy and resource management: additional models and algorithms

The ability to manage both the energy and the resources of a Micro Grid environment provides a significant advantage in term of lower energy consumption and overall energy costs, with respect to an unmanaged Micro Grid environment. However, the effectiveness of the manager depends on several factors.

Indeed, in addition to the resources, such as energy storage systems or local power plants, and the integration among the subsystems, required to route the energy among the devices, also the ability to correctly forecast the energy availability and evaluate the system energy demand is required.

In fact, nowadays local energy production, based on photovoltaic panels, is quite common in many residential buildings but, to exploit this resource fully, the ability to forecast the energy availability is mandatory. With regard to this, however, although the hourly energy yield can be forecast directly, the accuracy of such a forecaster may be lacking.

On the other hand, the forecast of the solar irradiation can be more accurate, even more so if the information is collected from a specialized service utility. Based on the solar irradiation then, the energy yield must be computed.

Although the process may seem straightforward, the modelling of the photovoltaic plant has to be accurate to avoid significant computation errors. On purpose, the model of a photovoltaic panel has been implemented to compute the actual energy yield while accounting the common limitation of real life power plants such as partial shading.

Regarding the energy demand of a Micro Grid environment, it should be noted that, to correctly manage the electrical load, electrical faults and overloads should be prevented, therefore the actual consumption of the grid has to be monitored. Nonetheless, monitoring the aggregate consumption level has limited value. With the support of a NILM framework, in fact, the manager can identify the critical device thus, having a better insight on the problem, can respond correctly to the issue.

On purpose, a photovoltaic power plant model with support to partial shading and a NILM algorithm to support the implementation of a energy management system that can operate in a real life Micro Grid environment.

5.1 Photovoltaic power plant model with support to partial shading

Small size photovoltaic power plants are commonly adopted as local energy sources in Smart Homes and Micro Grid environments. Although these plants can provide energy to supply the devices within the system, their performance depends on several factors the intensity of solar radiation and, moreover, they can be severely affected from partial shading.

In fact, in a plant the panels are usually connected in series, therefore, whenever a panel is shadowed not only its current output drops, causing a drop of the current output of the whole series of panels. Although this shortcoming can be partially prevented by means of bypass diodes, the controller responsible for operating the panel series, which adjusts the power output and maximize the energy yield, is not able to identify the optimal working conditions.

To overcome this issue, new algorithms are required to improve the abilities of the controller and the performance of the plant. To investigate the problem and devise a solution, a model of the plant has been proposed.

5.1.1 Circuital model

To simulate the energy production of each PV module, the mathematical model of the I-V characteristic of P-N cells has been used, and it is reported in Eq.(5.1):

$$\begin{aligned} I =& I_{PV} - I_{01} \cdot [e^{\frac{q(V+I\cdot R_S)}{k\cdot T_C \cdot m_1}} - 1] + \\ & - I_{02} \cdot [e^{\frac{q(V+IR_S)}{k\cdot T_C\cdot m_2}} - 1] - \frac{(V + I \cdot R_S)}{R_P}, \end{aligned} \tag{5.1}$$

which is an evolution of the single diode PV cell model [51, 52], since the current loss due to the recombination of charge carriers is modelled through two diodes [53, 54, 55].

In particular I is the output current, I_{PV} is the current generated by the incident light, whereas I_{01} and I_{02} are the reverse saturation currents of the diodes. The quantity kT/q is the thermal voltage of the P-N junction, where q is the charge of the electron, k is the Bolzmann's factor and T the temperature of the junction (in Kelvin). Also, m_1 and m_2 are the ideality factors of the two diodes, R_S is the series resistance and R_P is the parallel resistance. Thus the quantity $(V + I \cdot R_S)/R_P$ is the shunt current loss.

The parallel resistance has been modelled, according to [53, 54, 55], by means of Eq.(5.2):

$$R_P = \frac{(V_{mp} + I_{mp} \cdot R_S)}{I_{PV} - I_0[e^{\frac{V_{mp}+I_{mp}R_S}{V_T}} + e^{\frac{V_{mp}+I_{mp}R_S}{(p-1)V_T}} - 2] - \frac{P_{mp}}{V_{mp}}}, \tag{5.2}$$

where the quantities V_{mp} and I_{mp} are, respectively, the voltage and the current at the maximum power point, and V_T is the thermal voltage of the P-N junction. In this study, for instance, the computation of the resistances values Rp and Rs is carried out iteratively [55], and it is assumed that the ideality factor of the first diode is equal to 1 [51].

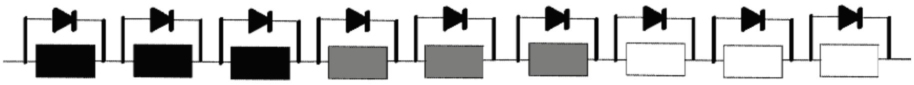
Based on this representation, each PV module has been modelled accounting the actual number of the PV cells of the panel. In turn, the PV arrays have been obtained as string of PV modules, each module having a bypass diode connected in anti parallel.

Indeed, since the open circuit voltage of a single cell is usually too low for actual power production, N cells are put in series forming a PV module, which presents an open circuit voltage that is N time the output voltage of a single cell, whereas the short circuit current is the same for every cell [55]. Because of that, under partial shading conditions, the cell with the lowest output limits the short circuit current of the entire module. To circumvent this issue a bypass diode is paired to every cell. As such, under partial shading conditions, the current is not limited anymore, but the P-V characteristic of the module now can exhibit multiple local maxima.

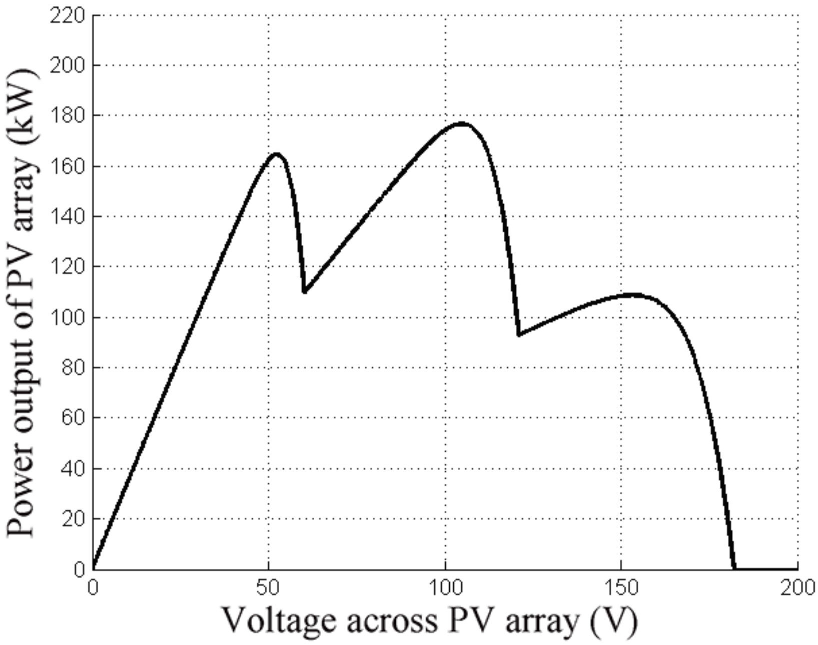
In a real life power plant, in most cases, several PV modules are connected in series, as well, thus forming a module string with the desired voltage output. Several strings are then connected in parallel to achieved the desired current output. In conditions of partial shading, every string can be considered as an independent subsystem, whereas the current output of the array is the sum of the currents of the strings [56].

For the sake of simplicity, it has been chosen to ignore the power losses due to the bypass diodes, and thus their effect has been approximated by simply disconnecting the modules affected by shadowing. Therefore, the characteristic curves of the plant have been computed as the sum of the curves of the strings. The curves of the strings, in turn, have been computed as the sum of the curves of as many panel sub sets as many irradiation levels are present over a string.

In fact, if we assume 3 irradiation levels over a single string of PV model (Fig.



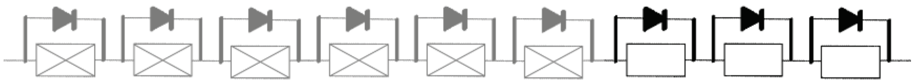
(a) Heterogeneus irradiation of a PV panel string



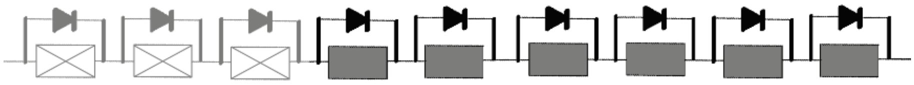
(b) Resulting PV characteristic curve

Figure 5.1: Panel string and its PV curve in heterogeneous irradiation conditions

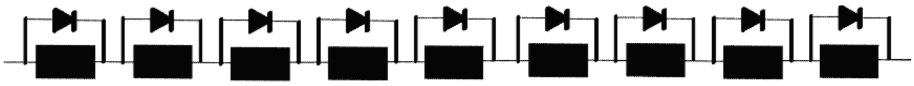
5.1(a)), the characteristic curve can be assumed as the result of three curves (Fig. 5.1(b)). If the drawn power level is below or equal to the output power of the least irradiated panels, all the module are able to supply energy (Fig. 5.2(c)). If the drawn power level rises above the output power of the least irradiated panels, the output voltage of the most shadowed modules drops, and thus these modules are disconnected from the string by the bypass diodes (Fig. 5.2(b)). In this case only the remaining modules can supply energy. If the drawn power level rise again, only the most irradiated modules can still contribute to energy production (Fig. 5.2(a)). In other words, as the output voltage rises from 0 to its maximum, the point representing the overall power output moves from one curve to the other (Fig. 5.2(d)) depending on how many panels can supply energy tracing the plot reported in Fig. 5.1(b).



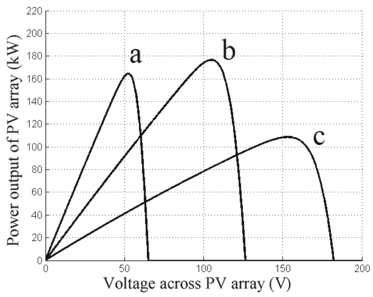
(a) Most irradiated sub set



(b) Intermediate irradiated sub set



(c) Least irradiated sub set



(d) Each PV characteristic

Figure 5.2: Homogenously irradiated panel substrings and their contribution to the overall PV curve

5.1.2 Maximum power point tracking

The output voltage of a PV module depends not only on the incident solar irradiation and the working temperature, but also on the amount of current being drawn. As such, in order to extract the maximum amount of energy from the module, the load has to be controlled so that the working point of the module matches the MPP of its P-V curve. Since the MPP tends to drift depending on irradiation and temperature, a MPP tracking algorithm is required to properly drive the power draw.

In partial shading conditions, however, due to the bypass diodes, the P-V curve of a string of module can show multiple maxima, thus hindering the work of the MPPT algorithms. To assess how this phenomenon affects the performance of the plant, two MPPT algorithms, the Perturb and Observe (P & O) [57, 58, 59] and the Incremental Conductance (IC) [60, 61, 62, 63, 64] have been implemented.

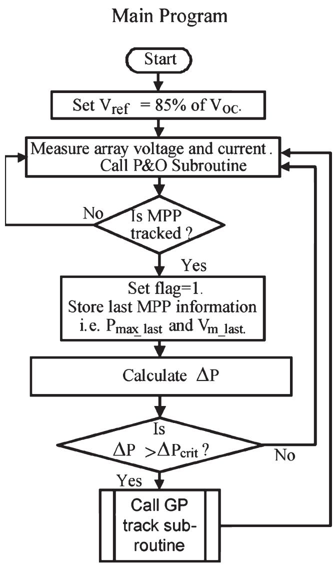


Figure 5.3: Implementation of the MPP with support to the Global Peak Tracking approach proposed in [1]

The P&O relies on the application of small disturbances to the output voltage, in the neighbourhood of the work point, in order to detect changes in the power output and, consequently, adjust the system to work in the point that provides greater power output. On the othe hand, the IC method is based on the conductance, computed as the derivative of the current, supplied by the system, with respect to the output voltage. This method relies on the fact that the slope of the curve P-V is null at the MPP, positive to its left and negative to its right.

As mentioned above, since those P&O and IC methods are not tailored to manage multiple maxima, which appear in the P-V curve under partial shading conditions, [65, 66], the Global Peak Tracking approach proposed in [1] has been implemented (Figs. 5.3 and 5.4).

In fact, on the P-V curve, the value of a local maximum is the smaller, the further it is from the global one. Hence, the GPT algorithm can move the working point to one local maximum to the global one, detecting the closest maximum by mean of a large perturbation.

Hence, by applying a large perturbation, either on a period-based scheme or on a event-based scheme, every local maximum, rather than the whole P-V curve, is evaluated.

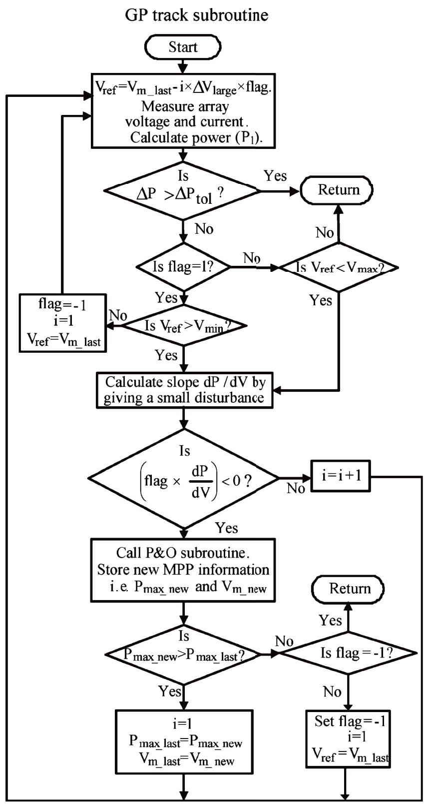


Figure 5.4: Implementation of the Global Peak Tracking subroutine based on [1]

5.1.3 Simulation setup

The model has been complemented with data obtained from real life PV modules. In particular a plant based on 4160 CanadianSolar PV panels (model CS6P-240P[1]) has been used as a reference. The overall surface of the panels amounts to 6691.42 square meters, while the plant is able to provide a nominal output power of 998.4 kWp. The panels are organized in strings made up of 20 modules each, for a total of 208 strings. Panel strings, in turn, are connected among themselves, thus the plant results in 13 groups each of them counting 16 strings of panels. All panels face south and present at 30° tilt angle with respect to the horizon. The distance among the panels is enough to prevent the panels from shadowing one another. A simplified sketch of the plant surface is presented in Fig.5.5.

Different scenarios have been evaluated based on different weather conditions:

a) single cloud shading;

[1]Online: http://www.solarelectricsupply.com/canadian-solar-cs6p-240-solar-panels-110

b) cloudy or rainy day;

c) sunny day (partial shading may occur at down and dusk, since the plant is located on sloped terrain).

The topology is investigated based on 3 configurations:

1) a centralized set-up has been modelled by assuming a dual inverter for the entire plant (2 in total) [67, 68, 69];

2) a macro block set-up has been modelled by assuming the adoption of one inverter for each subset of panel strings (13 in total);

3) a micro-block set-up has been modelled by assuming the adoption of one inverter per panel string (208 in total).

While the information regarding the PV power plant has been collected from a real life environment, the data on solar irradiation has been collected from the italian repository on solar irradiation [70]. In the current work, the time resolution of the simulation is of 1 minute, although its value can be adjusted in accordance to the MPPT device in use.

The SW framework, which is made available upon request, is based on MatLab and the Parallel Computing Toolbox. In our experimentation it has been running on a Windows 8.1 machine, based on an Intel Core i7-4700HQ CPU and with 8 GB of RAM.

The simulation of power production, with a time resolution of 1 minute, requires approximately 30 minutes for a single day, in the worst case. For the shading conditions and the MPPT techniques in the simulated scenario, this time resolution represents a good compromise between simulation's accuracy and processing time.

5.1.4 Results

Using a realistic model of solar irradiation (which varies in intensity in accordance to both the time of the day and the day of the year) and a model of shading, obtained on the basis of the three scenarios under analysis, the results plotted in Figs. 5.6 and 5.7 are obtained. The graphs refer to a single day; for the remaining days of the year it can expected a similar behavior although different average values of output power are achieved, depending on lighting conditions.

In the current experiments Configuration 3 is used as a reference. In fact, because of the small size of PV module strings, these are not affected by partial shading, thus operating in optimal conditions. Although it achieves a energy yield of 100 %, however, because of the great number of inverters, this topology is extremely costly, and thus unpractical.

In the proposed plots the blue line represents the maximum achievable power output, whereas the black line describes the actual power output of the devised power plant set-up.

Table 5.1: Yield loss of the proposed configurations with different MPPT strategies

		Configurations	
		1	2
P&O	(%)	33.5	20
IC	(%)	33.85	20.5
P&O with GPT	(%)	32.54	18
IC with GPT	(%)	33.97	20.7

In Fig. 5.6 a centralized power plant is subject to Scenario a during a Scenario c case. With respect to the case of Configuration 3, with Configuration 1 P&O algorithm is subject to a yield loss of about 33.5 %, whereas the IC algorithm loss reach 33.85 %. If the GPT subroutine [1] supports the MPPT tracking, P&O is able to perform better, with a yield loss of about 32.54 %. On the other hand the performance of the IC algorithm worsens, with a loss of 33.97 %. The results are summarized in Column 1 of Table 5.1 for each MPPT approach.

In Fig. 5.7 a string-based power plant is subject to the same scenario of Fig. 5.6. The smaller size of PV module block achieves a better performance. For instance while the standard P&O algorithm is affected by a yield loss of about 20 %, whereas in the case of the standard IC algorithm the yield loss is of about 20.5 %. If the GPT subroutine is integrated within the MPPT algorithm, P&O yield loss can be lowered down to 18%, whereas the IC algorithm performance still worsens with a loss of about 20.7 %. The results of Configuration 2 are summarized in Column 2 of Table 5.1.

On a yearly basis, the best performer is P&O with GPT support again for both Configuration 1 and 2 with a yield loss of about 10.8 % and 5.85 %, respectively. In absolute terms, the average daily energy yield of Configuration 2 is 225 kWh higher. Assuming a selling price of 38.9 €/MWh, this corresponds about 3200 € on a yearly basis.

Lastly, in Fig. 5.8 the micro-block power plant is subject to the same scenario of Figs. 5.6 and 5.7. In this case, clearly, the setup is the best performer, with the collection of the maximum power available. Consequently, there is no benefit from the GPT subroutine, since the efficiency is already at its highest, thus the corresponding plots are not shown.

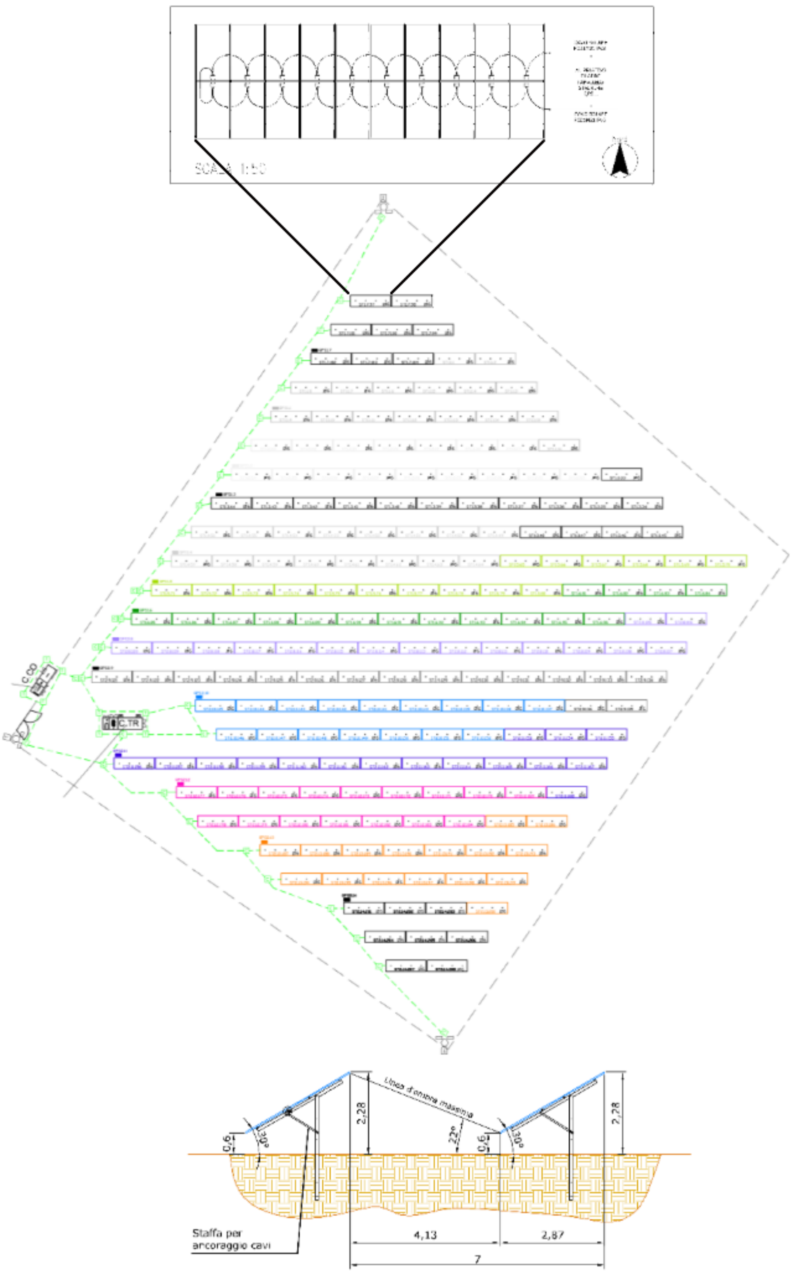


Figure 5.5: Power plant schematic configuration

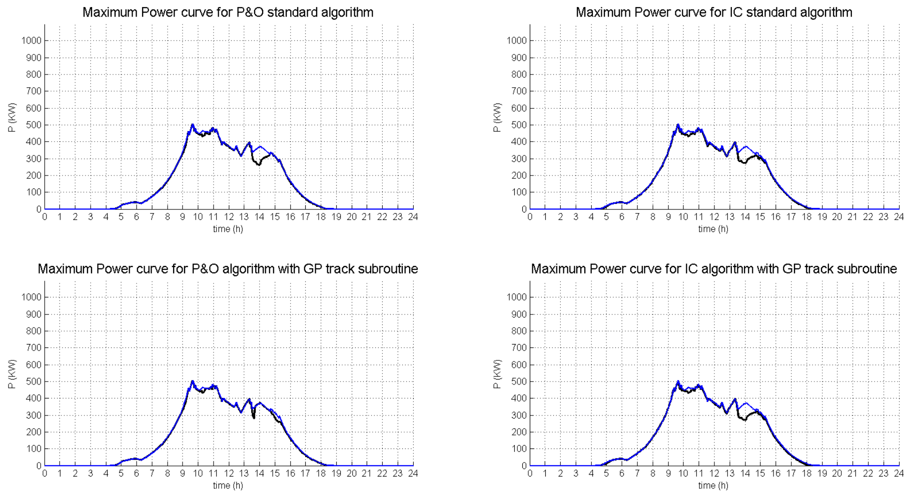


Figure 5.6: Simulation results for Configuration 1) with Scenarios a and c.

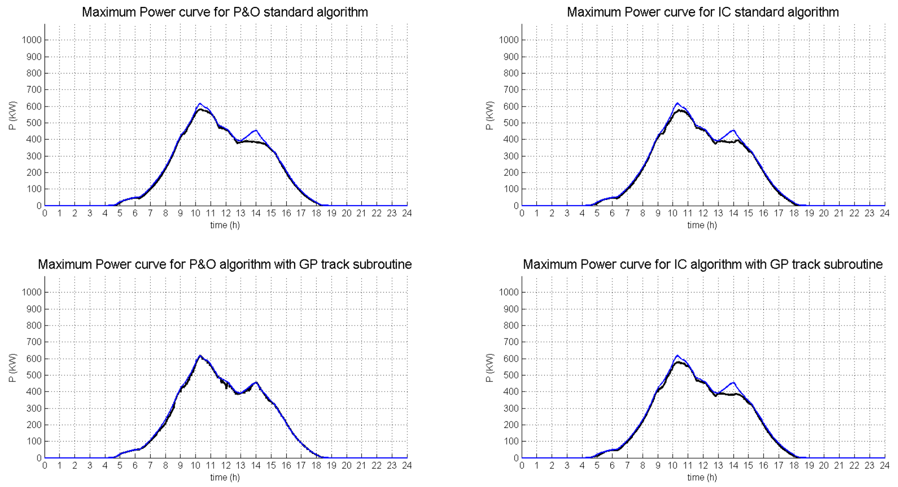


Figure 5.7: Simulation results for Configuration 2) with Scenarios a and c.

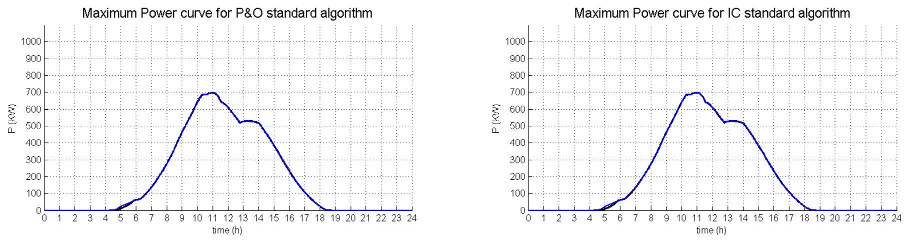


Figure 5.8: Simulation results for Configuration 3) with Scenarios a and c.

5.2 Non Intrusive Load Monitoring algorithm

Non Intrusive Load Monitoring techniques are being proposed, in literature, to identify the contribution of the domestic appliances within the aggregated energy demand of the residential environment. In fact, with respect to the monitoring of each appliance, that requires many sensor and therefore is potentially costly and invasive, NILM techniques can operate by means of a single sensor placed at the point of common coupling, providing a non invasive and low cost solution.

When integrate with an energy and resource management approach, then, NILM techniques represent the means to identify unplanned user activities, faults and events that may affect the energy and resource management.

On purpose, the Additive Factorial Approximate MAP (AFAMAP) algorithm has been extended, by means of a differential forward model, thus complementing the existing differential backward model. Furthermore, an aggregated data examination method has been employed, aimed to the detection of inadmissible working state combinations of appliances, as well as the constraints setting based on the reactive power disaggregation feedback.

5.2.1 The AFAMAP algorithm

The theoretical approach towards disaggregation is based on the work of Kolter and Jaakkola [71]. In this work the system is modelled relying on Additive Factorial Hidden Markov Model (AFHMM), for which the value of each aggregated power sample corresponds to a combination of working states of the appliances into the system.

Also, in this approach, the assumption that at most one HMM may change its state at any given time is made, which holds true if the sampling time is reasonably short. In this case, the transition on the aggregate power, when moving from a sample to the next, corresponds to the state change of a particular HMM.

Because of that, the differential signal, built from the aggregated power, can be modelled as the result of a Differential Factorial Hidden Markov Model (DFHMM), which relies on the same HMM models composing the AFHMM.

By combining the two models, the inference on the set of states of multiple HMMs can be computed through the Maximum A Posteriori (MAP) technique, which take the form of a Mixed Integer Quadratic Programming (MIQP) optimization problem.

One of the shortcomings of this approach is the non-convex nature of the problem, because of the integer nature of the variables: in this case, a relaxation towards real values is taken into account, leading to a convex Quadratic

Programming (QP) optimization problem. Thus, the Additive Factorial Approximate MAP (AFAMAP) approach is obtained.

In a real case scenario, the modelled output may not match with the observed aggregated signal, due to electrical noises, very small loads, or leakages. In that case, the issue is addressed by defining a robust mixture component in both AFHMM and DFHMM, named z_t and Δz_t, respectively. About this issue, in this work, a *denoised* environment [72] has been considered, i.e., all the contributions to the aggregated energy demand are known. Therefore, the robust mixture component is missing in this dissertation.

In the reference work [71], the parameter n defines the problem dimensionality: in this work, it is assumed $n = 1$, because the algorithm uses only the active power data to characterize the observed aggregated signal.

Specifically, the parameters of the problem follow:

- $N \in \mathbb{Z}_+$ is the number of HMMs in the system;
- $\overline{y}_\tau \in \mathbb{R}$ is the observed aggregated output (in denoised environments $\overline{y}_\tau = \sum_{i=1}^{N} y_\tau^{(i)}$, where $y_\tau^{(i)}$ corresponds to the true appliance output);
- $\sigma^2 \in \mathbb{R}$ is the observation variance.

The differential signal is referred to as $\Delta\overline{y}_{b\tau} = \overline{y}_\tau - \overline{y}_{\tau-1}$.

For the i-th HMM the parameters are:

- $m_i \in \mathbb{Z}_+$ is the number of states;
- $x_\tau^{(i)} \in \{1, \ldots, m_i\}$ is the HMM state at time instant τ ($x_\tau^{(i)} \equiv m_i$ corresponds to the OFF state);
- $\mu_j^{(i)} \in \mathbb{R}$ is the j-th state mean value;
- $\boldsymbol{\phi_b^{(i)}} \in [0,1]^{m_i}$ is the initial states distribution;
- $\boldsymbol{P_b^{(i)}} \in [0,1]^{m_i \times m_i}$ is the transition matrix.

The aggregated signal $\overline{y}_\tau$ is analysed using a windowing technique, where $\tau \in w_f = [(f-1)T + 1, \ldots, fT]$ for $f = 1, 2, \ldots, F$. The window w_f is the timebase for the algorithm and, for convenience, a new temporal variable is introduced by defining the relation $t = \tau - (f-1)T$, for $t = 1, 2, \ldots, T$, with $T \in \mathbb{Z}_+$. After the analysis of all the F windows, the disaggregated signals $\hat{y}_t^{(i)}$ are recomposed using the inverse relation $\tau = t + (f-1)T$.

In the optimization problem, the variables are defined as:

$$\mathcal{Q} = \left\{ \boldsymbol{Q(x_t^{(i)})} \in \mathbb{R}^{m_i}, \boldsymbol{Q(x_{t-1}^{(i)}, x_t^{(i)})} \in \mathbb{R}^{m_i \times m_i} \right\},$$

Input: $\overline{y}_{1:T}$ aggregated signal; $\left\{\boldsymbol{\mu}^{(1:N)}, \boldsymbol{P}_{\boldsymbol{b}}^{(1:N)}, \boldsymbol{\phi}_{\boldsymbol{b}}^{(1:N)}\right\}$ parameters for N HMMs; ${\sigma_1}^2, {\sigma_2}^2, \lambda$ covariance and regularization parameters.

Minimize over $\{\mathcal{Q} \in \mathcal{L} \cap \mathcal{O}\}$

$$\frac{1}{2{\sigma_1}^2}\sum_{t=1}^{T} E_t^{(a)} + \frac{1}{2{\sigma_2}^2}\sum_{t=2}^{T} E_t^{(bc)} + \frac{1}{2}\sum_{t=2}^{T} E_t^{(bnc)} + \\ + \sum_{t=2}^{T}\sum_{i=1}^{N}\sum_{\substack{j=1\\k=1}}^{m_i}\left\{Q(x_{t-1}^{(i)}, x_t^{(i)})_{j,k}\left(-\log {P_b}_{k,j}^{(i)}\right)\right\} + \\ + \sum_{i=1}^{N}\sum_{j=1}^{m_i}\left\{Q(x_1^{(i)})_j(-\log {\phi_b}_j^{(i)})\right\} \tag{5.3}$$

Output : $\hat{y}_{1:T}^{(1:N)}$, predicted individual HMM output

$$\hat{y}_t^{(i)} = \sum_{j=1}^{m_i} \mu_j^{(i)} Q(x_t^{(i)})_j \tag{5.4}$$

Figure 5.9: The AFAMAP algorithm.

for which the $Q(x_t^{(i)})_j$ variable is the indicator of the state assumed at time instant t, while the $Q(x_{t-1}^{(i)}, x_t^{(i)})_{j,k}$ variable is the indicator of the state transition from previous to actual time instant, for the i-th HMM.

The AFAMAP algorithm is shown in Figure 5.9.

In (5.3) the error terms are defined as:

$$E_t^{(a)} = \left(\overline{y}_t - \sum_{i=1}^{N}\sum_{j=1}^{m_i}\left\{\mu_j^{(i)} Q(x_t^{(i)})_j\right\}\right)^2, \tag{5.5}$$

$$E_t^{(bc)} = \sum_{i=1}^{N}\sum_{\substack{j=1\\k=1\\k\neq j}}^{m_i}\left\{\left(\Delta\overline{y}_{bt} - \Delta\mu_{k,j}^{(i)}\right)^2 Q(x_{t-1}^{(i)}, x_t^{(i)})_{j,k}\right\}, \tag{5.6}$$

$$E_t^{(bnc)} = D\left(\frac{\Delta\overline{y}_{bt}}{\sigma_2}, \lambda\right)\left(1 - \sum_{i=1}^{N}\sum_{\substack{j=1\\k=1\\k\neq j}}^{m_i} Q(x_{t-1}^{(i)}, x_t^{(i)})_{j,k}\right). \tag{5.7}$$

The QP optimization problem is defined in the form:

Minimize

$$\frac{1}{2}\boldsymbol{x}^T\boldsymbol{H}\boldsymbol{x} + \boldsymbol{f}^T\boldsymbol{x}, \tag{5.8}$$

subjet to the constraint:

$$\boldsymbol{A_{eq}}\boldsymbol{x} = \boldsymbol{b_{eq}}, \tag{5.9}$$

$$\boldsymbol{lb} \leq \boldsymbol{x} \leq \boldsymbol{ub}. \tag{5.10}$$

The variables of the problem are represented by the vector $\boldsymbol{x}$, which is composed of several subsets, based on the time instant t and the appliance index (i):

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{\Theta_1} \\ \vdots \\ \boldsymbol{\Theta_T} \end{bmatrix} \quad , \quad \boldsymbol{\Theta_t} = \begin{bmatrix} \boldsymbol{\Psi_t^{(1)}} \\ \vdots \\ \boldsymbol{\Psi_t^{(N)}} \end{bmatrix} \quad , \quad \boldsymbol{\Psi_t^{(i)}} = \begin{bmatrix} \boldsymbol{\xi_t^{(i)}} \\ \boldsymbol{\beta_t^{(i)}} \end{bmatrix},$$

$$\boldsymbol{\xi_t^{(i)}} = \begin{bmatrix} Q(x_t^{(i)})_1 \\ \vdots \\ Q(x_t^{(i)})_{m_i} \end{bmatrix} \quad , \quad \boldsymbol{\beta_t^{(i)}} = \begin{bmatrix} Q(x_{t-1}^{(i)}, x_t^{(i)})_{1,1} \\ \vdots \\ Q(x_{t-1}^{(i)}, x_t^{(i)})_{1,m_i} \\ \vdots \\ Q(x_{t-1}^{(i)}, x_t^{(i)})_{m_i,1} \\ \vdots \\ Q(x_{t-1}^{(i)}, x_t^{(i)})_{m_i,m_i} \end{bmatrix},$$

where the variables for the state are represented in $\boldsymbol{\xi_t^{(i)}}$, and the variables for the backward transition in $\boldsymbol{\beta_t^{(i)}}$.

The parameters of the problem fill up the elements of $\boldsymbol{H}$ and $\boldsymbol{f}$, according to the structure of the $\boldsymbol{x}$ vector, whereas $\boldsymbol{A_{eq}}$ and $\boldsymbol{b_{eq}}$ are used to represent the consistent constraints between the state and the transition variables. The vectors $\boldsymbol{lb}$ and $\boldsymbol{ub}$ define the lower and upper boundaries of the solution: because of the nature of the variables [71], the lower boundary is equal to 0, whereas the upper boundary to 1, for all the elements in $\boldsymbol{x}$.

5.2.2 Original contributions

In the reference approach, the DFHMMs are obtained as the difference, in term of power consumption, between the current and the previous sample (referred to as *backward transition*), so that a change in the state of an HMM can be evaluated against the change in the aggregated power consumption. Similarly, an additional evaluation, based on the next against the current sample (referred to as *forward transition*), is carried out. Furthermore, a smarter employment of the solver boundaries is evaluated, starting from a more accurate analysis

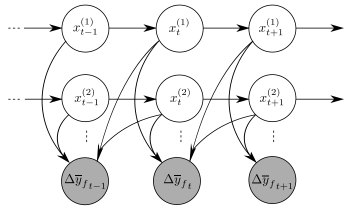


Figure 5.10: The Forward Differential FHMM.

of the aggregated power or using heterogeneous information, as the reactive power consumption of the electrical system.

Forward Differential Factorial Hidden Markov Model

Since the AFAMAP algorithm operates offline, it is possible to further extend the model by taking into account the transition from the current to the next state. The original DFHMM [71] is computed by looking backward from the current sample to the previous one, and thus it can be addressed to as Backward DFHMM. The new differential FHMM is computed by looking forward, as showed in Figure 5.10, and thus is referred to as Forward FHMM.

The formulation of the new model, also, differs from the original one, only in the index order. The new variables define the problem, as follow:

$$\mathcal{Q} = \left\{ \boldsymbol{Q(x_t^{(i)})} \in \mathbb{R}^{m_i}, \boldsymbol{Q(x_{t+1}^{(i)}, x_t^{(i)})} \in \mathbb{R}^{m_i \times m_i} \right\},$$

where the variables are indicators of the transition from the next to the current state: $Q(x_t^{(i)})_j = 1 \Leftrightarrow x_t^{(i)} = j$, and $Q(x_{t+1}^{(i)}, x_t^{(i)})_{j,k} = 1 \Leftrightarrow x_{t+1}^{(i)} = j, x_t^{(i)} = k$. The consistent constraints between the state variables and transition variables need to be satisfied:

$$\mathcal{L} = \left\{ \mathcal{Q} : \begin{array}{c} \sum_{j=1}^{m_i} Q(x_t^{(i)})_j = 1 \\ \sum_{k=1}^{m_i} Q(x_{t+1}^{(i)}, x_t^{(i)})_{j,k} = Q(x_{t+1}^{(i)})_j \\ \sum_{k=1}^{m_i} Q(x_{t+1}^{(i)}, x_t^{(i)})_{k,j} = Q(x_t^{(i)})_j \\ 0 \le Q(x_t^{(i)})_j, Q(x_{t+1}^{(i)}, x_t^{(i)})_{k,j} \le 1 \end{array} \right\}. \tag{5.11}$$

Therefore, the new cost function is derived for the Forward DFHMM, based on the forward differential aggregated signal $\Delta\overline{y}_{f\,t} = \overline{y}_t - \overline{y}_{t+1}$, as follow:

$$\begin{aligned}
&\frac{1}{2{\sigma_3}^2}\sum_{t=1}^{T-1} E_t^{(fc)} + \frac{1}{2}\sum_{t=1}^{T-1} E_t^{(fnc)} + \\
&+\sum_{t=1}^{T-1}\sum_{i=1}^{N}\sum_{\substack{j=1\\k=1}}^{m_i}\left\{Q(x_{t+1}^{(i)}, x_t^{(i)})_{j,k}\left(-\log {P_f}_{k,j}^{(i)}\right)\right\} + \\
&+\sum_{i=1}^{N}\sum_{j=1}^{m_i}\left\{Q(x_T^{(i)})_j(-\log {\phi_f}_j^{(i)})\right\},
\end{aligned} \tag{5.12}$$

where the error terms in (5.12) are defined as:

$$E_t^{(fc)} = \sum_{i=1}^{N}\sum_{\substack{j=1\\k=1\\k\neq j}}^{m_i}\left\{\left(\Delta\overline{y}_{f\,t} - \Delta\mu_{k,j}^{(i)}\right)^2 Q(x_{t+1}^{(i)}, x_t^{(i)})_{j,k}\right\}, \tag{5.13}$$

$$E_t^{(fnc)} = D\left(\frac{\Delta\overline{y}_{f\,t}}{\sigma_3}, \lambda\right)\left(1 - \sum_{i=1}^{N}\sum_{\substack{j=1\\k=1\\k\neq j}}^{m_i} Q(x_{t+1}^{(i)}, x_t^{(i)})_{j,k}\right). \tag{5.14}$$

The transition matrix $\boldsymbol{P_f^{(i)}}$ represents the probability of state change from the next to the current time instant: this parameter is equivalent to the typical representation of the transition matrix (i.e., the probability of state change from the previous time instant to the actual) evaluated after flipping the signal, thus it can be derived by using the available algorithm for HMM training. The parameter $\boldsymbol{\phi_f^{(i)}}$ represents the final state distribution, that is the initial state distribution starting from the end of the signal.

Since the duality in the forward and backward representation of the AFHMM (i.e., it is derived from the same observed signal, but in opposite directions), the problem definition using only one of the two versions of the DFHMM leads to the already known performance. Considering simultaneously both versions of DFHMM may lead to performance improvements: for this reason the forward differential function (5.12) is added to the reference formulation (5.3), thus leading to a new optimization problem. The variable vector $\boldsymbol{x}$ in the QP problem accounts for the new terms, following the same structure introduced in Section 5.2.1:

$$\boldsymbol{\Psi_t^{(i)}} = \begin{bmatrix} \boldsymbol{\xi_t^{(i)}} \\ \boldsymbol{\beta_t^{(i)}} \\ \boldsymbol{\phi_t^{(i)}} \end{bmatrix}, \qquad \boldsymbol{\phi_t^{(i)}} = \begin{bmatrix} Q(x_{t+1}^{(i)}, x_t^{(i)})_{1,1} \\ \vdots \\ Q(x_{t+1}^{(i)}, x_t^{(i)})_{1,m_i} \\ \vdots \\ Q(x_{t+1}^{(i)}, x_t^{(i)})_{m_i,1} \\ \vdots \\ Q(x_{t+1}^{(i)}, x_t^{(i)})_{m_i,m_i} \end{bmatrix},$$

where the new term $\boldsymbol{\phi_t^{(i)}}$ represents the variables for the forward transition.

The introduction of the new variables leads to an alteration of the problem constraints, represented by the parameters $\boldsymbol{A_{eq}}$ and $\boldsymbol{b_{eq}}$, and the variable boundaries $\boldsymbol{lb}$ and $\boldsymbol{ub}$.

Profile-based selection of the boundaries

In order to solve the optimization problem, different solutions, which satisfy the constraints, need to be evaluated before the solver finds the optimal one. As such, the values of $\boldsymbol{x}$ that are not compatible with the given set of samples can be discarded, to restrict the search domain and improve the search efficiency.

On purpose, the lower and upper boundaries of the variable $\boldsymbol{x}$ are selected beforehand in order to prevent that the solver investigates those combinations of states that do not match the value of the aggregated power consumption. The selection method is similar to the one proposed in [73].

If several runs of a single appliance are evaluated, although the same working states are identified, the signature tends to differ from a run to the other. For this reason, the appliance power consumption can be modelled as a stochastic process and, therefore, the output value $y_t^{(i)}$, relative to a working state $x_t^{(i)}$ of an appliance, can be modelled as a gaussian variable, described by a mean value and a variance value:

$$y_t^{(i)} | x_t^{(i)} \sim \mathcal{N}\left(\mu_{x_t^{(i)}}^{(i)}, {\sigma_{x_t^{(i)}}^{(i)}}^2\right). \tag{5.15}$$

Regard to this, the power signal is replaced by a simplified model that presents a constant power consumption, corresponding to the mean value of the working state power value, with a superimposed noisy contribution, described by the variance value in the working state.

Since the aggregated data $\overline{y}_t$ is assumed to correspond with the sum of the power consumption of each appliance, it can be modelled as a gaussian variable, described by a mean value and a variance value equivalent to the sum of the

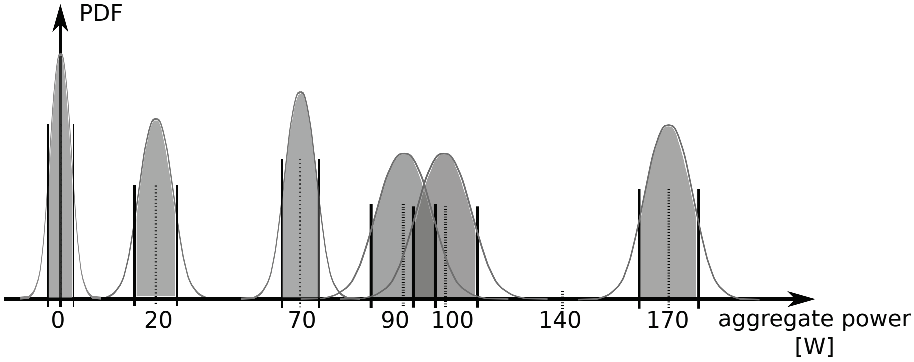


Figure 5.11: A sketch of the different probability density functions (PDF) for each aggregated power value produced by the combination of all appliances states power levels.

corresponding values of each appliance, under the assumption of statistical independence between the appliances:

$$\overline{y}_t | x_t^{(1:N)} \sim \mathcal{N}\left(\sum_{i=1}^{N} \mu^{(i)}_{x_t^{(i)}}, \sum_{i=1}^{N} {\sigma^{(i)}_{x_t^{(i)}}}^2 \right). \tag{5.16}$$

This simplified model results in a number of admissible combinations of working states equal to $\prod_{i=1}^{N} m_i$. It allows to evaluate which combination of working states fit the power value for each sample of the aggregated data, thus discarding the incompatible ones. The effectiveness interval for each combination is centred in mean value, and its width is twice the value of the standard deviation. For some combinations, which have similar mean value or great variance, the effectiveness intervals result overlapped: for those cases, if the power value falls in this region, both the combinations are considered valid.

Based on this observation, it is possible to manipulate the boundaries of the QP problem domain. For instance, if 2 HMMs are considered, $M1$ and $M2$, whose power levels are, $M1 = \{70, 0\}$ and $M2 = \{100, 20, 0\}$ respectively, the different combined power levels are $\{0, 20, 70, 90, 100, 170\}$, each one with its own variance value. This example is represented in Figure 5.11. Considering a few different values of aggregated power, e.g., $\overline{y}_t = \{20, 95, 140\}$, it can be observed that $\overline{y}_t = 20$ is obtained as the combination $(x_t^{(1)} = 2, x_t^{(2)} = 2)$, therefore the allowed constraints are defined as:

$$\begin{bmatrix} 0 \\ 1 \end{bmatrix} \leq \boldsymbol{\xi_t^{(1)}} \leq \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \leq \boldsymbol{\xi_t^{(2)}} \leq \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}.$$

If $\overline{y}_t = 95$, the value falls in an overlapped interval, belonging to the combinations ($x_t^{(1)} = 2, x_t^{(2)} = 1$) and ($x_t^{(1)} = 1, x_t^{(2)} = 2$), thus, the allowed constraints are defined as:

$$\begin{bmatrix}0\\0\end{bmatrix} \leq \boldsymbol{\xi}_t^{(1)} \leq \begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}0\\0\\0\end{bmatrix} \leq \boldsymbol{\xi}_t^{(2)} \leq \begin{bmatrix}1\\1\\0\end{bmatrix}.$$

Whereas, if $\overline{y}_t = 140$, no combination is corresponding, thus the boundaries remain as default.

Clearly, the same process can be applied to bound the $\boldsymbol{\beta}_t^{(i)}$ and $\boldsymbol{\phi}_t^{(i)}$. In regard to this, however, since transitions are related to the steady states, the evaluation of the steady states is enough to bound both kinds of variables.

Problem constraints through reactive power disaggregation

Even though disaggregation is aimed for the aggregated power consumption, in most cases the focus is centred on the active power alone. Nonetheless, given the generality of the AFAMAP algorithm, targeting the reactive aggregated power is also possible. In regard to this, in the present work, the application of the AFAMAP algorithm to the aggregated reactive power has been investigated as well, based on the fact that reactive power is a common trait of the power signature of a residential appliances subset.

In the current scenario, the disaggregation of the reactive power samples is carried out, in order to collect additional information about the activity states of the appliances. This information, in turn, is used to further define the lower and the upper boundaries of the states in the active power disaggregation. Similarly to the active power case, the HMMs are modelled for each appliances starting from the signature in the reactive power and the AFAMAP algorithm is run by using the aggregated reactive power signal as input.

Following the basic knowledge in circuit theory, an electrical load with a reactive component (i.e., an appliance) which has a reactive power consumption greater than 0 is necessary turned on, therefore the boundaries of the problem in active power disaggregation are assigned as follows:

$$\begin{bmatrix}0\\0\\0\end{bmatrix} \leq \boldsymbol{\xi}_t^{(i)} \leq \begin{bmatrix}1\\1\\0\end{bmatrix}.$$

Although, when the reactive power consumption is 0, the active component could be both null or greater than 0, depending on whether the appliance is turned off or only the load passive component is working. Therefore, the boundaries of the problem in active power disaggregation are setted as default.

5.2.3 Computer simulations

In this section different aspects of the experimentation are proposed. Among these are the metrics used in the performance evaluation of the proposed disaggregation approach. Also, the simulation scenario and the parameter settings used in the experiments are discussed, as well as the performance of the disaggregation algorithm.

Metrics

According to the study carried out by the authors [74], two metrics, that take into account different aspects of disaggregation, are chosen.

The first one, namely *state based*, considers the ability of the system to infer the exact state of evolution of each HMM in the model: for the i-th appliance, the multiclass confusion matrix is built by comparing, for each time instant $t = 1, 2, \dots, T$, the disaggregation variables $\boldsymbol{\xi_t^{(i)}}$ value assumed in the problem solution, with the exact evolution state $x_t^{(i)}$, defined as the *ground truth*. Each class corresponds to a state $j = 1, \dots, m_i$ of the i-th HMM. Since that the values in $\boldsymbol{\xi_t^{(i)}}$ are not-integral, the computed confusion matrix is soft weighted, similar to the fuzzy-logic [75]. For each class, the Precision $P_j^{(i)}$ and Recall $R_j^{(i)}$ are computed, then the average between the classes evaluates the medium performance for each HMM:

$$P_{S_f}^{(i)} = \frac{1}{m_i} \sum_{j=1}^{m_i} P_j^{(i)}, \quad R_{S_f}^{(i)} = \frac{1}{m_i} \sum_{j=1}^{m_i} R_j^{(i)}. \tag{5.17}$$

The second metric, namely *energy based*, deals with the comparison between the disaggregated profiles and the power consumption at appliance level: as defined by Kolter [71], Recall measures what part of the power consumption has been correctly classified. Precision, on the other hand, measures how much of the power assigned to an appliance truly belonged to that appliance. For the i-th appliance, the output of the algorithm $\hat{y}_{1:T}^{(i)}$, and the true output $y_{1:T}^{(i)}$, are used to compute the Precision and Recall as:

$$P_{E_f}^{(i)} = \frac{\sum_{t=1}^{T} \min\left(\hat{y}_t^{(i)}, y_t^{(i)}\right)}{\sum_{t=1}^{T} \hat{y}_t^{(i)}}, R_{E_f}^{(i)} = \frac{\sum_{t=1}^{T} \min\left(\hat{y}_t^{(i)}, y_t^{(i)}\right)}{\sum_{t=1}^{T} y_t^{(i)}}. \tag{5.18}$$

Both metrics are evaluated for a signal window w_f with $f = 1, 2, \dots, F$, therefore to evaluate the performance over the entire dataset the metrics are averaged over the windows:

$$P^{(i)}_{\{S,E\}} = \frac{1}{F}\sum_{f=1}^{F} P^{(i)}_{\{S_f,E_f\}}, R^{(i)}_{\{S,E\}} = \frac{1}{F}\sum_{f=1}^{F} R^{(i)}_{\{S_f,E_f\}}. \tag{5.19}$$

Finally, in order to consider the total performance of the disaggregation system, the average between the appliances is computed:

$$P_{\{S,E\}} = \frac{1}{N}\sum_{i=1}^{N} P^{(i)}_{\{S,E\}}, \quad R_{\{S,E\}} = \frac{1}{N}\sum_{i=1}^{N} R^{(i)}_{\{S,E\}}. \tag{5.20}$$

As unique evaluation metric, the F_1 is chosen and it is calculated as the geometric mean between Precision and Recall:

$$F_{1\{S,E\}} = 2\frac{P_{\{S,E\}}R_{\{S,E\}}}{P_{\{S,E\}}+R_{\{S,E\}}}. \tag{5.21}$$

Set up of the simulation scenario

The dataset used for the experiments is the Almanac of Minutely Power dataset (AMPds) [76]: it contains recordings of consumption profiles belonging to a single home in Canada for a period of two years at 1 minute sampling rate. It provides active and reactive power at appliance level, unlike most of the dataset in which only the active power is provided at appliance level [74]: this information is crucial to test the new approach based on the reactive power disaggregation as constraint.

Analysing the contents of the dataset, it can be noticed that the usage of the appliances is homogeneous throughout the entire period, therefore the experiments are evaluated on 6 months of data, which can be considered representative of the entire dataset.

To create the HMM models of the appliances, the training requires at least one signature per appliance, although multiple signatures lead to a more general model. In the proposed work, a subset of the data, spanning over 14 days, has been deemed sufficient to collect all the signatures required to train all the HMMs. The HMM are trained in accordance to the Baum-Welch algorithm, after determining the ground truth state over the time: those are obtained through a clustering procedure, in which every cluster represents a power consumption level of the appliance, thus a state of the HMM. This process is achieved using the k-means algorithm, in which the number of the cluster is imposed in a supervised manner, starting from the knowledge of the operating states of the appliance. The power level mean and the variance values are achieved by means of a gaussian variable fitting procedure over the samples belonging to each cluster.

To satisfy the condition of *denoised* system, the aggregated data is synthetically composed by summing the appliance level power signals. The experiments are conducted by using the appliances at higher contribution, therefore 6 appliances have been chosen: dryer, washing machine, dishwasher, fridge, oven, and heat pump.

The simulations are conducted in Matlab environment and the CPLEX solver is used to solve the QP problem.

The value of starting probability $\boldsymbol{\phi_b^{(i)}}$ of the i-th HMM is imposed to assume the certainty for the OFF state for $f = 1$, whereas for the consecutive windows, $1 < f \leq F$, it is imposed to assume the value of the last sample $\boldsymbol{\xi_T^{(i)}}$ of the previous window, in order to ensure the contiguity of the solution on the window border. The value of the ending probability $\boldsymbol{\phi_f^{(i)}}$, instead, is uniformly imposed in every state, since no information from the consecutive window is available.

Different experiments are conducted varying the size of the windows between the values $T \in \{10, 30, 60, 90, 120\}$ minutes, and the effectiveness of the innovative aspect is evaluated: the introduction of the forward term in the cost function, the selection of the boundaries related to the aggregated power level and to the disaggregation output of the reactive power.

The variance parameters are defined with ${\sigma_1}^2 = {\sigma_2}^2 = {\sigma_3}^2 = 0.01$ according to the variance of the experimental data and the regularization parameter $\lambda = 1$.

Results evaluation

The results of the experiments, based on the scenario described in in Section 5.2.3, are presented in the current section.

In Figure 5.12, the AFAMAP disaggregated power consumption profiles of the appliances are compared against the corresponding true outputs, provided by the dataset: in the figure a time span of 10 hours, corresponding to 600 samples, is considered. At the bottom, the energy distribution over the same period, expressed among the appliances in terms of percent value, is compared between the reconstructed and the true appliances consumption.

The signals reveal that the appliances which show an high steady power consumption are easily recognized, whereas the appliances with complex working cycles, or with several power levels, are more difficult to detect. Indeed, whenever several appliances present similar consumption levels, many combinations may satisfy the problem constraints, thus additional information is required to identify the active appliances. For instance, in Figure 5.12, the oven and the fridge are seldom recognised, whereas the detection of the dryer and the washing machine are partially more successful.

The evaluation of the algorithm performance is carried out by means of the metrics proposed in Section 5.2.3. Although the focus of the present work is the AFAMAP algorithm, the dataset being used and the proposed training method are different with respect to [71], therefore a direct comparison against the results proposed in the reference work is not possible. To overcome this shortcoming, the baseline has been created anew, by means of the AFAMAP algorithm, the AMPds dataset and the proposed training method.

The disaggregation results computed by means of the metrics are reported in Figure 5.13: in Figure 5.13(a) the state based metric is presented, whereas the energy based metric is proposed in Figure 5.13(b). The results are shown for different values of the time window length. Clearly, since all the results exceed 0.5, the plots have been drawn from 0.5 onwards. Both plots show that the best results are achieved using the shortest time window. On a different note, however, not every configuration improves in the same way.

Focusing on the state based metrics, it is possible to observe that the AFAMAP baseline shows a significant performance improvement with the decreasing of the window length, except when passing from the 30 to 10 minutes window size. On the contrary, the forward differential model gives an improvements at the shorter window size, resulting in the best performance in the unbounded problem solution, with a F_{1S} of 0.738 and an improvement of 1% respect to the baseline.

Fixing the boundaries of the problem, in every simulation case, gives the benefit on the disaggregation results: the profile based method gives a considerable performance improvements with every window size, but the highest relative improvement can be noted at the smallest size, resulting to a F_{1S} of 0.863 and a relative improvement of 18%.

Alternatively, the boundaries can be setted based on the reactive power disaggregation feedback: the results, showed in Table 5.2, demonstrate that the reactive power reaches high performance in disaggregation. This is due to the high difference in the reactive components of each appliance, which involves a strong distinction in the creation of the HMM, therefore allowing an highly reliable disaggregation. The usage of this information results in a performance improvement for every window size, more considerable at the smallest size: in general, the usage of the reactive power feedback gives benefits to the disaggregation, with a F_{1S} of 0.802 and a relative improvement of 9.7%, therefore less than the profile based constraints.

Clearly, the same trends presented about the state based metrics still hold true when evaluating with the energy based metrics. The most notably difference between the two plot, in fact, is that the rate of improvement of the algorithms when decreasing the time window length: indeed, the forward differential model introduction results to a F_{1E} of 0.771 and a relative improvement of 1.2% respect to the baseline, whereas the profile based setting of the boundaries results to a F_{1E} of 0.878 with a relative improvement of 15.2% and the reactive power based method to a F_{1E} of 0.832 with an improvement of 9.2%.

The forward differential model seems to be beneficial only with the shortest time window: it may be a direct consequence of the problem formulation alteration. Indeed, the introduction of additional variables increases the size of the problem, therefore the computational burden, for which the solver demonstrates worst performance, as it happens for the baseline approach with larger window size. Despite this, the improvements achieved adding the differential forward information to the model are restricted to the application scenario: since the algorithm operates on a per-sample basis, for each appliance behaviour two state changes unlikely happen across three contiguous samples, thus the forward difference cannot provide a substantial support to the inference of the actual working state.

The errors in the disaggregation phase are caused by the multiplicity of states combinations which can correspond to the same value of the aggregated data: for this reason the use of boundaries allows to exclude some solutions that are not eligible, therefore facilitates the solver to find the exact solution to the problem. Nevertheless, the variation over time of the power consumption associated to a specific appliance working state, causes an unwanted variability, i.e., a noise component, in the achieved solution.

Table 5.2: Disaggregation results on reactive power. The configuration used is: AFAMAP + Forward differential.

	window size				
Metric	10 min	30 min	60 min	90 min	120 min
State based: F_{1S}	0.922	0.877	0.869	0.867	0.865
Energy based: F_{1E}	0.935	0.883	0.877	0.875	0.874

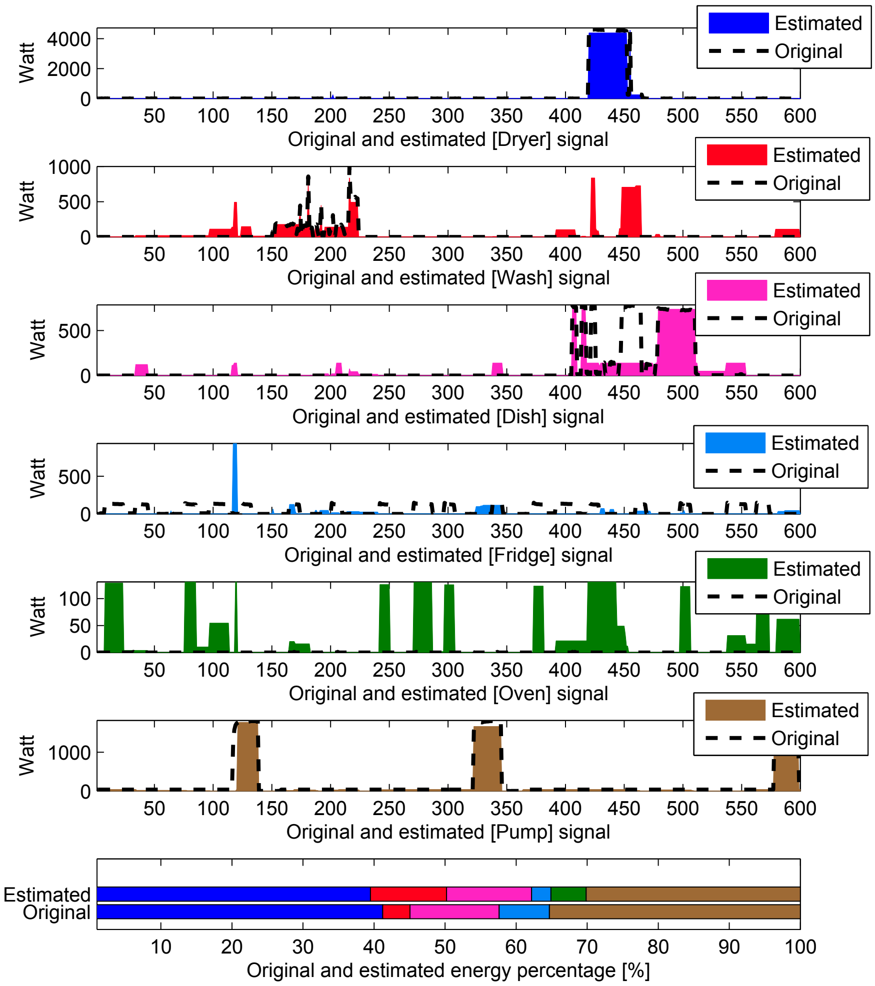
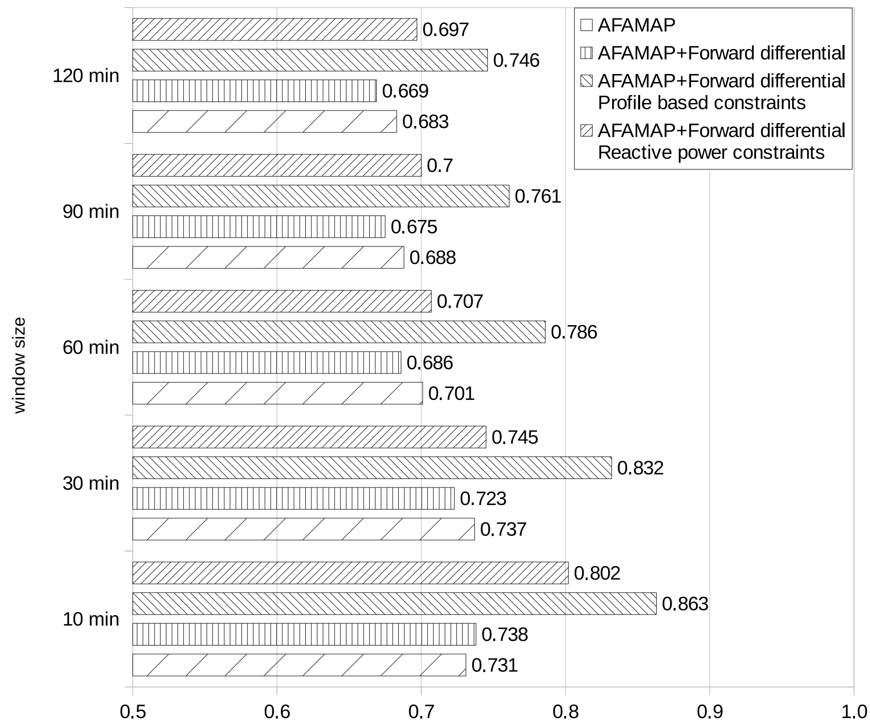
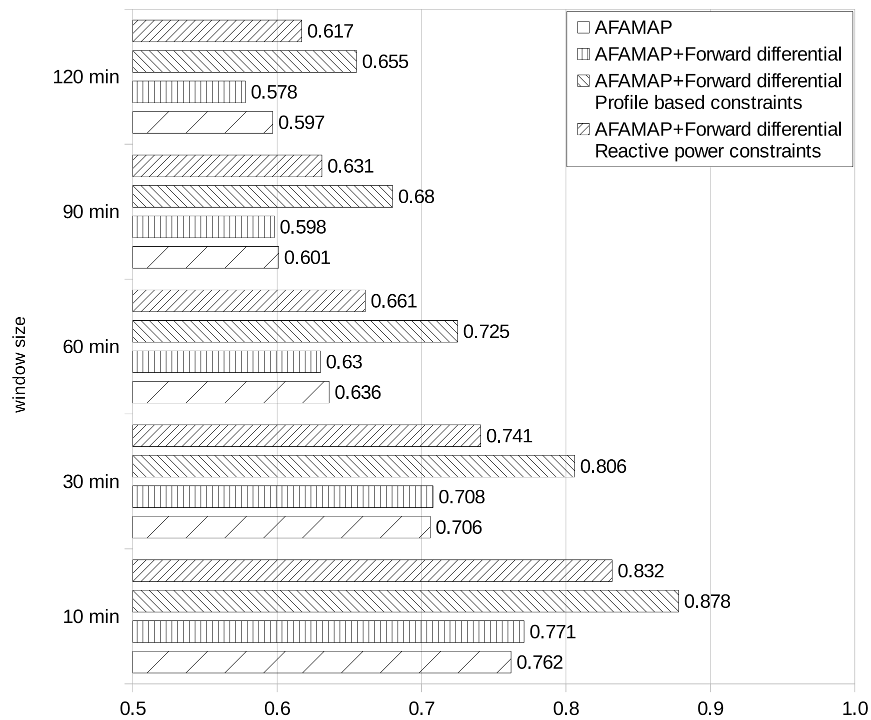


Figure 5.12: Appliances consumption: estimated AFAMAP disaggregation output against original signals.



(a) State based metric: $F1_S$



(b) Energy based metric: $F1_E$

Figure 5.13: Disaggregation performance on AMPds dataset using 6 appliances, with different algorithm configuration.

Chapter 6

Other contributions

The proposed management method focuses on improving the use of energy, and resources, from the standpoint of both end users and utility services, rather than on the basis of efficiency alone. However, the choice to improve the efficiency through the optimization of the overall energy cost broadens the potential and the generality of the method.

Indeed, since the residential activity lies on the use of electricity water and gas, the management method can be focused on the overall cost of the residential activity, thus optimizing the use of energy water and gas.

However, the implementation of such a management scheme in a real life Micro Grid environment requires the ability to monitor the water and gas consumption at real time. Therefore automatic meter reading support is required. In addition, since there is a chance that leakages can affect the water or gas distribution system, the ability to identify leakages is also required.

Regarding the automatic meter reading problem, since a renewable energy supply may lower the maintenance cost, an energy aware lazy scheduling algorithm, aimed at ultra low power consumption micro devices, has been devised. As such renewable energy power supply can effectively support an automatic meter reading device.

Regarding the need to identify a leakage, a novelty detection algorithm has been implemented to detect the occurrence of leakages through the analysis of the timely meter records.

6.1 Implementation of the Energy Aware Lazy Scheduling Algorithm

With the availability of ultra low power consumption micro devices, the development of wireless sensors has grown considerably. In most cases, however, the use of battery powered devices requires the periodic replacement of the power supply, thus rising the maintenance costs. To overcome this issue, renewable energy powered supply can be used, but the energy availability cannot be guaranteed.

On purpose, a task scheduling algorithm, namely Energy Aware Lazy Scheduling Algorithm (EA-LSA) [77, 78], has been developed so that the device can adjust its activity depending on the energy availability. In order to evaluate the effectiveness of the approach, the EA-LSA has been implemented in a ultra low power micro controller, and the performance of the device has been investigated.

6.1.1 Scheduler Theory

The proposed scheduling scheme is an improved implementation of the EA-LSA [77, 78]. It adds to the original design the ability to adapt the device workload to the actual harvested energy. EA-LSA also inherits the characteristics of LSA [79, 80] which, in turn, is derived from the Earliest Deadline First (EDF) scheduling algorithm [81, 82, 83].

The EDF operates by sorting the tasks depending on their deadlines, giving higher priority to the tasks whose deadline is closer to the current time, and executing the tasks immediately upon selection. In this scenario, the energy assignment is affected mostly by the arrival time and the time spent by the task in the waiting queue. Therefore a task whose arrival time fall close to the task deadline is more prone to starving than a task with the same deadline but an earlier arrival time. To prevent this behaviour and improve the energy assignment, the LSA suggests a lazy approach meant to delay the task execution and the energy consumption as much as possible. In this manner the energy assignment depends only on the deadlines and it is not affected by the arrival time.

By describing with the followings quantities the tasks of an assigned set:

- i: i-th task index;
- Q: index queue;
- J_i: i-th task;
- a_i: arrival time of the i-th task;
- d_i: deadline of the i-th task;
- s_i: start time of the i-th task (full power execution);
- f_i: completion time of the i-th task;
- $P_H(t)$: power income at time t;
- $P_S(t)$: CPU power consumption at time t;

- p_d: CPU maximum power consumption;
- $E_i(t_1, t_2)$: energy assigned to the i-th task over the $[t_1,\ t_2]$ interval;
- $E_C(t)$: residual energy at time t

the Lazy Scheduling Algorithm can be modelled as follows:

LSA Algorithm (Lazy Scheduling with p_d = const)

Require: maintain a set of indices i ∈ Q of all ready but not finished tasks J_i
$P_S(t) \Leftarrow 0$;
while(true)
  $d_j \Leftarrow$ min d_i : i ∈ Q ;
  calculate s_j;
  process task J_j with power $P_S(t)$;
  t ⇐ current time;
  if (t == a_k) **then** add index k to Q; **endif**;
  if (t == f_j) **then** remove index j from Q; **endif**;
  if ($E_C(t) == C$) **then** $P_S(t) \Leftarrow P_H(t)$; **endif**;
  if (t ≥ s_j) **then** $P_S(t) \Leftarrow p_d$; **endif**;
endwhile;

This algorithm is meant to achieve the energy assignment optimality (and thus the scheduling optimality) thanks to its lazy nature. Any task execution is postponed until the available energy is sufficient to operate the CPU, without interruption, from the start time to the corresponding deadline. The only exception appears when the reservoir is full: in this case the harvested energy is directly transferred to the task with the earliest deadline. The direct energy transfer is modulated to match the harvested power and to avoid any unnecessary storage depletion. When energy is transferred directly to the task, the start time, based on the deadline and the execution time at full power only, is further delayed to account the corresponding executed work.

The energy starving is avoided if the total energy demand of the assigned task set does not exceed, at each deadline, the available energy. The time starving is avoided if the total energy demand of the assigned task set does not exceed, at each deadline, the maximum workload that the CPU is capable of. Since those two conditions assess if a task set can be completed without violating any deadline, and since those depends on the energy amount over time only, the evaluation can be carried out when the task set is devised, without any burden for the device. In addition, given the constraints of both deadlines and energy availability, the result of the test does not depend on the scheduling algorithm. In other words, by proving that if LSA fails then these same conditions are not satisfied [80], it is proven that no algorithm can best LSA, that is the LSA optimality.

Due to its optimality LSA qualifies as an ideal task scheduling algorithm. Moreover, in [80], the admittance test is suggested as a tool also useful to select the minimum reservoir capacity, necessary to avoid any deadline violation. Indeed, from a theoretical point of view, LSA appears to be an interesting solution but, regretfully, it requires an exact prediction ability impossible to implement in a real life sensor node. To overcome this situation, rather than relegate LSA to the realm of the reference models, it was chosen to evaluate the algorithm, to point out the most relevant limits that may emerge in a real case scenario, and eventually overcome them.

Concerning the admittance tests, while extremely useful to seek any deadline violation, when used to tailor the task set energy demand may prevent the available energy from being used in full. In fact, by using the lower bound as the energy availability profile, the task set consumption never exceeds the minimum energy availability. While in adverse condition this configuration may help to avoid the deadline violations, it also limit the task set consumption in favourable conditions. In other words the harvested energy that exceed the minimum expected amount cannot be used, which means that a great amount of energy may be left unused [84], and also that the harvester will not be fully exploited.

From a different point of view, if the statistical nature of the energy profile is taken into account, it is also apparent that the admittance test may not guarantee the absolute deadline violation avoidance, and therefore a more restrictive profile may be used. As a result, the task energy consumption may be further constrained. Concerning the scheduling scheme itself, if task completion is not an absolute requirement and a few violations may be allowed, meaning that the LSA is operating in a non-ideal setup, it is possible to notice that LSA is prone to waste energy by executing tasks that cannot be completed, since it does not evaluate the tasks' energy demand at run-time.

Based on these evaluation, it was deemed that a run-time selection of the tasks to be executed, based on the available energy and the tasks' energy demand, might have improved the LSA performance, leading to the algorithm known as Energy Aware LSA that can be described as follows:

EA-LSA Algorithm (Lazy Scheduling with p_d = const)

Require: maintain a set of indices i $\in$ Q of all ready but not finished tasks J_i
$P_S(t) \Leftarrow 0$;
while(true)
  $d_j \Leftarrow \min d_i : i \in Q$;
  if $(e_j > (E_C(a_j) + E_H(a_j, d_j)))$ **then** remove index j from Q;

```
  else
   calculate s_j;
   process task J_j with power P_S(t);
   t ⇐ current time;
   if (t == a_k) then add index k to Q; endif;
   if (t == f_j) then remove index j from Q; endif;
   if (E_C(t) == C) then P_S(t) ⇐ P_H(t); endif;
   if (t ≥ s_j) then P_S(t) ⇐ p_d; endif;
  endif;
endwhile;
```

While maintaining the original approach of LSA, the new algorithm is able to improve the energy efficiency of a sensor node by the moment that the starving tasks are discarded upon selection and, as such, before they receive any energy. The evaluation of EA-LSA [77, 78] focused on this subject proved that, in case of deadline violations, the energy efficiency of LSA can be improved. At the same time it also hinted at a further development of the conditional execution concept. By evaluating the energy availability upon the tasks' selection, it is eventually possible to adapt the device workload to match the actual harvested energy. In this manner it may be possible to exploit any energy surplus in favourable condition, while still avoiding, or containing, the deadlines' violation problem in adverse situations.

In particular, in the test proposed in [77, 78], an energy shortage causes the starving task to be discarded tout court, avoiding any energy waste but not preventing the deadline violation itself. An improvement may be possible forfeiting the task accuracy. In this case it may be possible to lower the task energy demand to fit the energy amount, so that the task can be executed to completion nonetheless and the deadline violation is avoided. In a dual manner it is possible to increase the task accuracy exploiting any energy surplus.

Presently this suggestion has been integrated into the proposed implementation. Since the dynamic workload adaptation can be achieved with different schemes, based on the EA-LSA selective execution, this feature can be considered an application of EA-LSA rather than a brand new algorithm.

6.1.2 Scheduler Implementation

An energy harvester has a major impact on the final cost of a sensor node solution, while EA-LSA is meant to maximize the energy exploitation thus improving the sensor node cost effectiveness. As such tailoring the energy harvester output, while accounting the energy availability changes all over the year, should be less critical and, eventually, a lower final cost may be achieved. From this perspective a set up composed by the CBC-EVAL-09 universal energy harvester from Cymbet [85] and the Texas Instruments eZ430-RF2500T [86] represents an interesting low cost solution sample.

The mote board is based on the TI MSP430F2274 [87] micro-controller and hosts the CC2500 transceiver [88]. The CBC-EVAL-09, other than being compatible with the RF2500T board, is presently the sole harvester on the market providing the facilities to monitor the power income and the stored energy level, required by EA-LSA, through the CBC915EP controller [89].

By means of the Integrated Development Environment (IDE), provided by Texas Instrument, namely Code Composer Studio V5 (CCSv5)[1], the C++ is used as programming language. Nonetheless, C++ advanced features, such as the standard library, have been avoided to lower the memory needs.

The libraries provided as part of CCS are used to implement an hardware abstraction layer (HAL), that provides the interfaces and the primitives required by the scheduling routines. Since within the MSP430F2274 IC almost each peripheral device has its own set of control registers, the HAL is obtained as a set of classes. The scheduling routine, in a similar fashion, is implemented as a set of classes. The firmware structure is depicted in Fig. 6.1.

Hardware Abstraction Layer

For each peripheral device a class has been implemented to provide the interfaces, and the methods, required to manage the corresponding set of control registers according to the datasheet [87]. A common structure for each of the HAL modules has been adopted. Usually the member functions provide the control interfaces, a constructor, if required, implements the proper set-up routine, whereas a static member function, if necessary, manages the associated Interrupt Service Routine (ISR). To account different hardware configurations alternate implementation of the classes have been used.

The modules have been validated prior to the scheduler development, the correctness of the routines implementation and the absence of conflicts among the modules themselves have been also verified. The current implementation make use of five modules, within the source code, however, alternative versions of some of them are provided.

- Analog to Digital Converter module Concerning the ADC, the required facilities and registers are set-up upon initialization by the constructor. Alternative blocking methods are provided to manage the sampling of different channels. On this regard, since the internal voltage reference is affected by the changes in the power supply voltage, an external voltage reference has been used.

[1] http://www.ti.com/tool/ccstudio

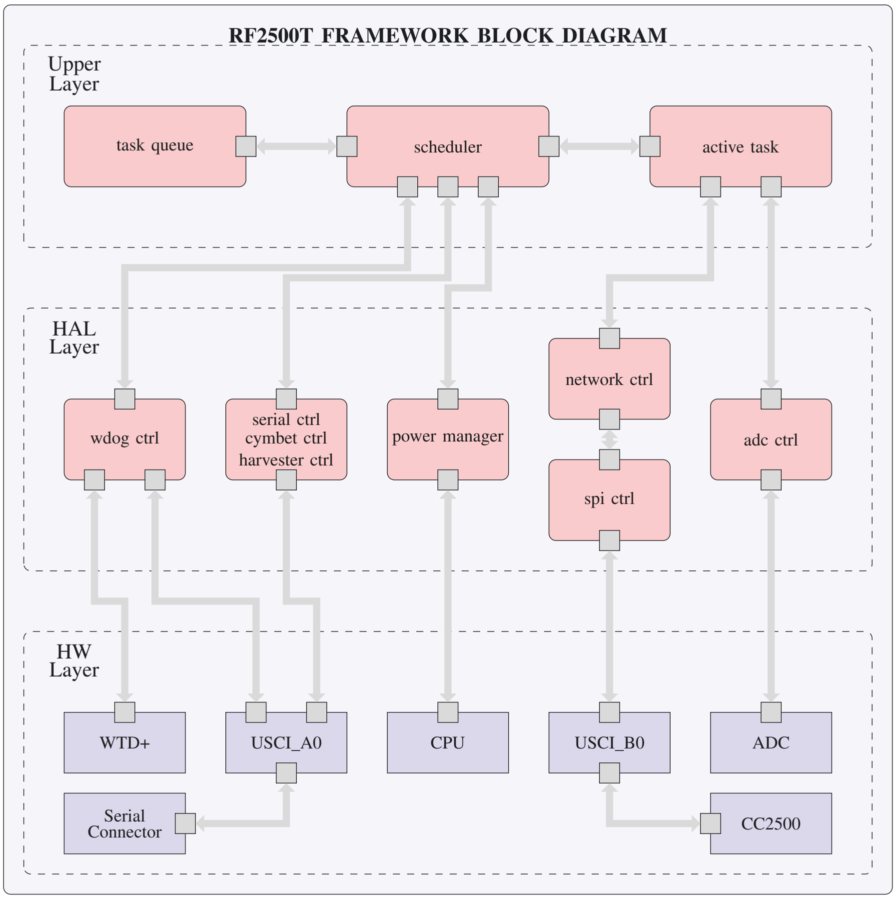


Figure 6.1: RF2500T hw/sw framework block diagram

- Serial Peripheral Interface Bus module An SPI module hosts the routines that implement the SPI protocol over the Universal Serial Communication Interface B (USCI_B) port and controls the communication toward the CC2500 transceiver. Bidirectional communication is provided by means of multiple routines due to the different communication schemes provided by the bus. The code is based on the material provided by TI such as the SLAA325a [90] and TI SimpliciTI libraries [91], the SWRA112B DN503 design note [92] and the SWRA141 NO49 application note [93] among others.

- Networking module A networking module, integrating the SPI one, has been developed although it has been actually used only in the preliminary test phase. In the evaluation of the scheduling algorithm performance, in fact, data transmission and acknowledgement (ACK) reception are employed as a mean to emulate a load, since no Access Point (AP) is present at the other hand of the wireless channel. The module is extensively based on the SWRS040C CC2500 user manual [88], the TI SimpliciTI libraries [91], the SLAA325A [90], the SLAA365 [94] among others.

The module has been validated by inspecting the data over the SPI bus during transmissions and receptions in a dual board setup. Regretfully, being the CC2500 transceiver an integral part of the board, it is incompatible with the SmartRF Studio tool, thus its set-up has yet to be optimized.

- Universal Asynchronous Receiver-Transmitter modules The communication over the Universal Serial Communication Interface A (USCI_A) has been used with different hardware set-ups. Multiple modules have been created from on a single model, thus they implements different primitives while sharing the same structure.

  A module provides the interface to interact with the Cymbet harvester. An additional module monitors the activity of the device through the USB dongle. A third one hosts both features to support an emulated harvester.

- Watchdog module A watchdog module is responsible of the regulation of the scheduling process. It relies on the watchdog interrupts to generate periodical events (local clock). Due to hardware clock limitations, the current implementation presents consistent overhead, heavily increasing the overall device consumption. These issues may be solved through by means of an autonomous local clock. The module reimplements the watchdog facility to integrate a serial port supervisor, required to detect serial communications time-outs. A block diagram of the routine is proposed in Fig. 6.2.

- Power Management module A power management module provides the facilities to manage the CPU activity according to the needs of EA-LSA that is, the primitives required to compute the start-time, and those to check the condition whether the task execution should be carried out or not, depending on the available energy and the current time. Since the task execution is carried out in steps, while the CPU is turned off altogether when not needed, the power manager is able to match energy production and energy consumption, as per the EA-LSA scheme, with a fixed CPU clock.

  Concerning the Dynamic Power Management (DPM), in idle phases, the device operates mostly in low power mode (LPM3), thus, although a Dynamic Frequency Scaling (DFS) scheme has been also programmed, its

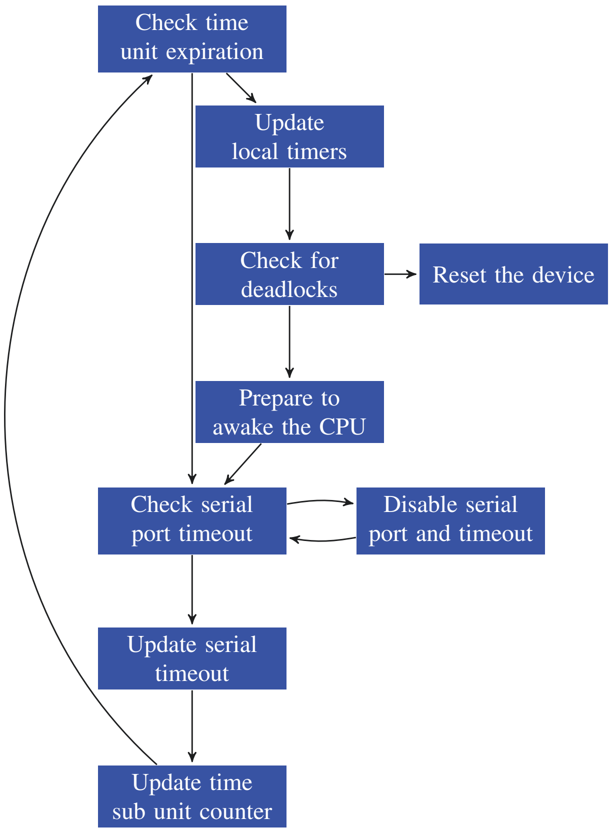


Figure 6.2: Watchdog ISR block diagram

actual use is deemed ineffective because of several hardware limitations. First and foremost, the RF2500T does not provide any Dynamic Voltage Scaling (DVS) facility, thus the device efficiency remain unchanged at best, as highlighted in [95]. Secondly, according to [87] and [86], in the current scenario the frequency of the Digitally Controller Oscillator (DCO), required to drive the USCI ports, is fixed, thus conflicting with the DFS.

As such, the current implementation of the DFS is based on the clock divider alone and, being said divider rather coarse, neither the enhancement of the energy saving ability of the algorithm nor the adaptation of the power consumption of the device to the power production of the harvester can be properly supported. In addition, if a finer DFS is to be implemented, the concurrent access to the DCO has to be managed, resulting in increased computational activity and increased energy consumption, therefore even an enhanced DFS would hardly provide any gain over the coarse version.

- Scheduling Routine Layer The completion and validation of the HAL, provided the tools to manage the hardware and implement the task scheduler. As said the purpose of the current work is the evaluation of the performance of EA-LSA. For this reason the implementation is quite close to a proof of concept, rather than a ready to market product. Thus, the guidelines of the development phase have been robustness and flexibility, whereas the optimization of the code and the coverage of long term scheduling have been discarded at present, and left to deal with at a later time. The scheduler in itself is quite simple, requiring a task queue, the scheduling routine itself, and the implementation tasks to be executed.

- Task queue To account the task management, the need of pre-emption and the resources constraints, a distinction between assigned tasks and active tasks has been made. The assigned tasks, whose execution is yet to begin, are managed through place holders so that the memory requirements per task are kept to a minimum. The active tasks, which are being executed, are implemented as state-machines entities. As such, the internal state of the active tasks is kept in RAM, whereas the actual code is stored in flash memory.

  To manage the scheduling, two separated ordered linked lists have been used. Insertion and removal policies have been designed so that the task with the earliest deadline is always at the top of the active tasks' queue, and the corresponding place holder is at the top of the assigned tasks' queue. This structure has been chosen so that the queues are traversed only once per task assignment, rather than once per time unit.

  Concerning the memory allocation policy, lacking a target application, flexibility has been chosen over robustness, thus a dynamic allocation scheme has been adopted. Memory leaks and saturation have been avoided implementing insertion and removal primitives.

- Scheduling Routine To implement a pre-emptive scheduling algorithm such as EA-LSA, to fit the limited RAM of the MSP430F2274, a cooperative multitasking scheme, based on a super-loop structure, has been chosen, in conjunction with state-machine like tasks. The proposed approach prevents the need of context-switches, since the state of each active task is kept in ram. To avoid concurrency, blocking primitives are used to access HAL and hardware.

  Each task is a compound of self contained subtasks, or steps, that are executed to completion. At the end of each step, the state of the task is

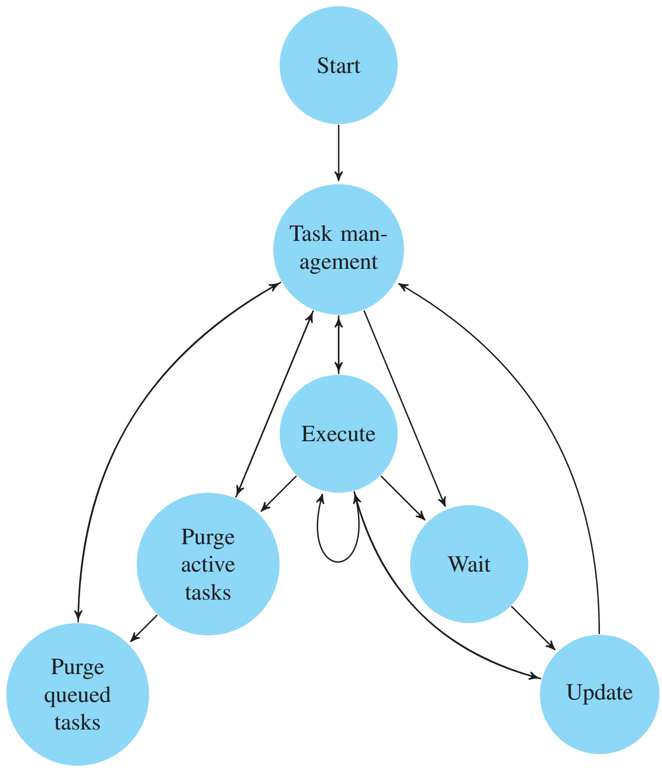


Figure 6.3: Scheduling routine state diagram

updated and the CPU control is returned to the scheduler. The scheduler checks if pending events are to be managed and, in case, it updates the scheduler state. If no pending events are present, and no task is ready to be executed, the Low Power Mode 3 (LPM3) is used as waiting condition till the next time unit. If a task ready to be executed is present, the CPU control returns to the task, that resume its execution from its last known state.

The communication needs between scheduler tasks are meet through the task main routine exit status. Presently interprocess communication is not used. The diagram state of the scheduling routine is represented in Fig. 6.3.

- Task implementation As anticipated, task implementation has been carried out as a state machine entity. Each task has only one main routine and several subroutines. The main routine selects a subtask depending on the internal state of the main task and a set of given conditions. The subroutines provide the facilities required by the subtasks. When the main routine is invoked a subtask is selected and it is executed to completion. Then the CPU control is returned to the scheduler along with an exit status providing information on the computational process occurred.

Although the tasks have been implemented solely to provide a load to the scheduler, they may be used as templates, implementing a sensing routine, a simple computation routine and the transmission routine.

All the tasks data types are obtained through inheritance from a common task data type. The common task data type provides the data member and the member function required to manage the tasks instances. Also the primitives required to access the underlying HAL are present. Each specific task data type provides the data members and function members required by the specific task alone, along with the conditions required to execute the assigned subtasks. Each task instance is instantiated upon execution and removed in consequence of either completion or starvation (deadline violation). Since the information needed to resume the task activity is contained within the instance itself, context switches are not required to suspend the task activity.

To evaluate the performance of the scheduler a set of five task types is created. Each task type collects and transmits a specific kind of information.

6.1.3 Energy harvesting

To properly evaluate the effectiveness of EA-LSA, an harvester providing information concerning the power income and the stored energy is required. However, collecting such information is rather difficult. Not only currents and voltages to be sensed are quite low in amplitude but also, depending on the energy availability and the sensing process consumption, the sensing process may be inaccurate or even incomplete. Furthermore, in most cases, sensing devices current demand amounts to tens of μA at the very least, thus being not negligible in the present scenario [96].

Since addressing such issues, and developing an harvester ad hoc, would fall outside the scope of the current work, a ready to market solution is searched. Among the available solutions the Cymbet Energy Harvesting Evaluation Kit 09 [85], being the only one able to provide the necessary information, is picked. Alternatives are evaluated as well, but none seems to provide the means to monitor both the power income and the stored energy.

Concerning the support to EA-LSA, the set-up is well suited, since the adopted harvester is designed to interact with the RF2500T. In fact, the Energy Processor CBC915EP [89] not only monitors the harvesting process, but also supports serial communication to provide the scheduler with information on both power income and residual energy. The main drawback, since the serial port is the only way to access the RF2500T controller, is the lack of control over the scheduler activity.

In particular, while be the wireless link can be used to retrieve information concerning the scheduling status, the additional data transmission would cause unaccounted consumption, affecting the scheduling activity. Additionally, packet losses would prevent the correct assessment of tasks' completion and starving. In a dual manner, direct access to the sensor node, requiring the EZ430 USB dongle, would prevent any additional data and power connection towards the RF2500T. To circumvent these issues, an emulated energy harvester has been devised, providing the scheduler with information concerning both power income and energy via a serial interface, while monitoring the scheduling process at the same time.

Therefore, while during the development phase and the preliminary test phase, the Cymbet harvester is used, to evaluate the performance of EA-LSA the implemented emulator, developed with the Cymbet Energy Harvesting Evaluation Kit 09 as a reference, is used. The emulator runs on the 64-bit version of the MathWorks MatLab 2011a [2] computing environment. The software was hosted on a notebook PC based on the Intel Core i7 CPU series, with 8GB of RAM and Microsoft Windows Seven 64-bit OS on board.

As a result, the evaluation process is deemed to be not dependent on the actual harvester measurement accuracy, nor on the test bench measurement accuracy. The former, in particular provides information concerning the stored energy in addition to the power income, as such an inaccurate measurement can affect the scheduling performance. The latter on the other hand, is used to measure the energy consumption of each subtask, information required to foresee if a task will starve or not. In the latter case, then, an inaccurate measurement can affect the scheduler performance as well.

By computing the residual storage energy, based on the measured energy demand of the subtasks, the measurement error is kept out of the equation thus improving the reliability of the evaluation test.

6.1.4 Tasks characterization

To program the sub task execution conditions, the energy demand of each subtask is required so that, at each task step, the residual energy demand to complete a task can be evaluated. The energy demand can also be used to compute the residual stored energy within the harvester emulator. To measure the device consumption a test bench has been developed based on the Texas Instruments SLAA378B application note [97]. An instrumentation amplifier is

[2] http://www.mathworks.com

used in accordance to the WisarLab reference [95].

The schematics of the measurement testbench is presented in Fig. 6.4.

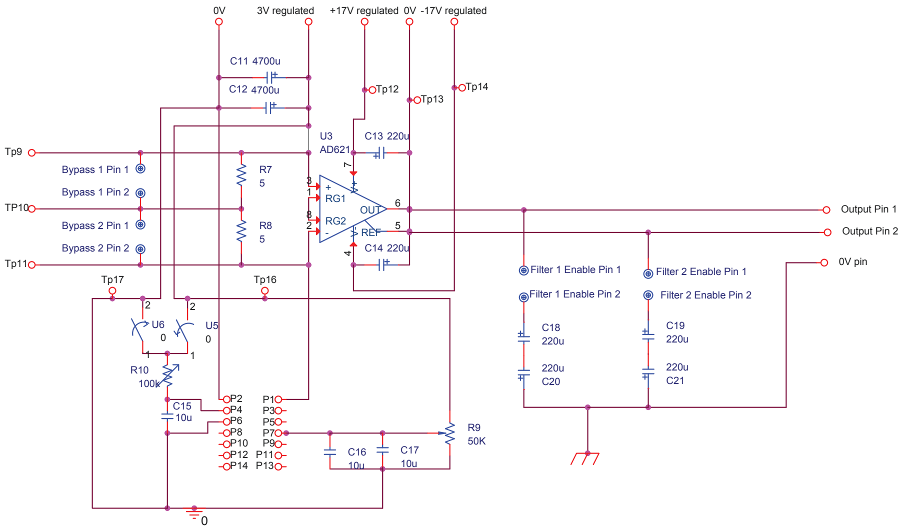


Figure 6.4: RF2500T testbench

The circuit hosts the AD621 instrumentation amplifier [98], which provides a 100 V/V gain, an external voltage reference, required by the ADC within the MSP430, the connection headers between the test bench and the RF2500T. Also a signal, driven by a switch, is used to toggle the firmware activity replacing the push button on the RF2500T. A series of two 5 Ω resistors, each paired with a bypass jumper, is used to sense the supplied current so that gains of 500 V/A and 1000 V/A are obtained. As a result, a dynamic range up to 30mA is available when the transceiver is used. At the same time the oscilloscope readings can be improved when low power configurations are inspected. A Tektronix TPS2024B oscilloscope[3] is used to read the output of the amplifier. For test purpose only, the filter capacitors, applied between the shield and the amplifier output, are enabled. The power supply of the AD621 is provided through a ±17 V linear stabilized regulator, while the RF2500T is powered by a 3V stabilized switching power supply. A preliminary test is conducted by replacing the RF2500T board with a 1MΩ resistor in order to draw a 3μA current. The 1000V/A gain is used and the filter capacitors are not enabled. A value of about 3.90 mV is registered at the output of the amplifier by the instrument as shown in Fig. 6.5.

The Low Power Modes (LPM) 4, 3 and 0 are evaluated using the 1000 V/A gain configuration, with and without the filter capacitors. During the LPM3

[3]http://www.tek.com/datasheet/oscilloscope/tps2000-digital-storage-oscilloscopes

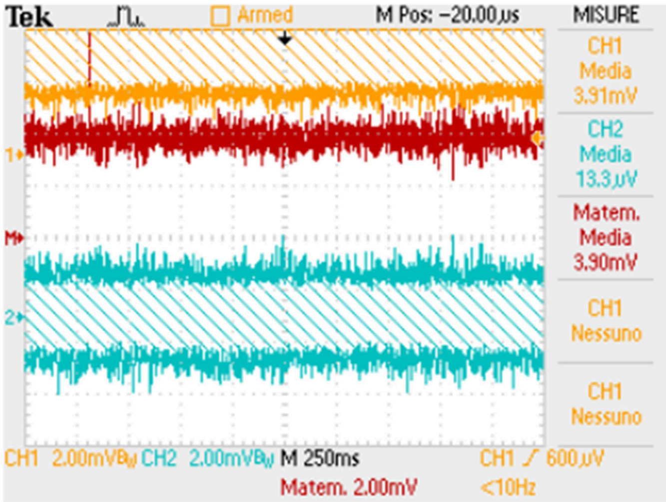


Figure 6.5: Testbench preliminary check

mode test, also, the watchdog ISR consumption is measured. On the other hand the CPU consumption is measured during the LPM 0. An endless loop executing a sum is used to evaluate the CPU demand. The results are reported in Table 6.1.

Table 6.1: CPU and watchdog power consumption

Routine type	Measured Voltage (mV)	Power (μW)
Cpu 1MHz (no filtercap)	426	1328.86
Cpu 1MHz (filtercap)	401	1250.08
Noise contribution	25	77.88
LPM3 (no filtercap)	9.69	30.27
LPM3 (filtercap)	1.46	4.56
LPM4 (no filtercap)	9.52	29.74
LPM4 (filtercap)	1.10	3.44

The profiling process is completed by the evaluation of the consumption of the hardware accessing routines. The RSSI measurement and the transmission require the 500 V/A gain configuration, while in the others cases the 1000 V/A gain configuration is used. The consumption profiling of the intended routines provides several traces, recorded by the oscilloscope, such the one in Fig. 6.6 that refers to the RSSI measurement (or sampling) routine.

For each trace, the intervals corresponding to a constant voltage level are measured and the values noted along with the corresponding voltage level. The pool of data is presented in Table 6.2. In the RSSI measurement case a trace is provided in Fig. 6.6, the different intervals are numbered accordingly to the Table 6.2 entries.

In the reception routine case, two implementations have been evaluated. The two differ only in regards to the fact that one does not receive any packet and thus ends after a timeout is reached. The other one is interrupted after the packet reception. Since the receiver power demand does not depend on actual data reception, only one profile is presented. The receiver consumption contribution is described in the “RX part 4” entry of Table 6.2.

At the end of the profiling process, the collected data are used to calculate the energy consumption of each task step. In fact, within a task step only two causes to energy consumption exist, that is the access to peripheral devices and the access to the CPU. The consumption and duration of the primitives used to access the peripheral devices are known, and since task steps are executed to completion, no concurrent access to the devices is possible. During the remaining time within the sub task, the CPU is used at full capacity. Then, measuring the duration of each task step, the CPU activity time during each sub task can be computed. To measure the duration of a task step, an internal timer that provides a μs time resolution is used. The resulting consumption values are presented in Table 6.3 on a per step basis.

A similar approach, with a few adjustment, is used to assess the energy consumption of the scheduler routine. To account changes of each state duration, that may occur depending on the required operations, the cyclic nature of the scheduling process is exploited. For each state of the scheduling routine, the average value of the state duration is measured over several scheduling periods. Then, by calculating the consumption and the number of occurrences of each

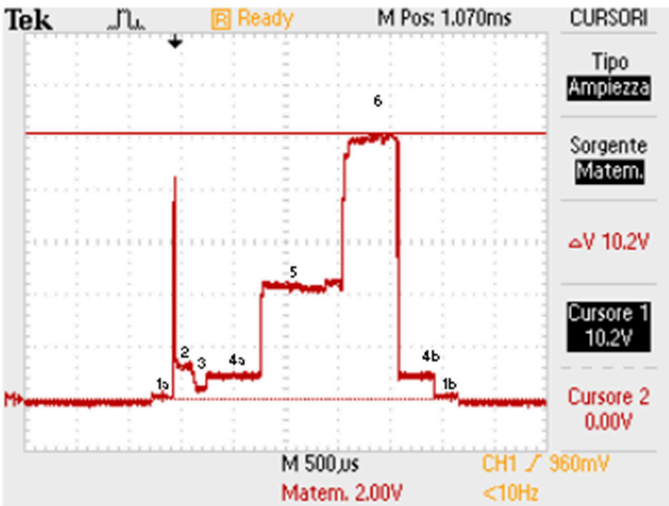


Figure 6.6: RSSI consumption profile

Table 6.2: Hardware access routines energy consumption

Routine type	Measured Voltage (mV)	Elapsed Time (μs)	Current (μA)	Power (μW)	Energy (μJ)
Temp part 1	640	2900	640	1994.92	5.78
Temp part 2	460	340	460	1434.75	0.49
Temp part 3	1410	100	1410	4383.30	0.44
Temp total		3340			6.71
Volt part 1	450	750	450	1403.61	1.05
Volt part 2	856	80	856	2666.21	0.21
Volt total		830			1.27
RSSI part 1	100	450	200	624.57	0.28
RSSI part 2	1200	150	2.4	75.00	1.12
RSSI part 3	400	100	800	2495.47	0.25
RSSI part 4	800	850	1600	4983.47	4.23
RSSI part 5	4300	825	8600	26434.97	21.81
RSSI part 6	10000	475	20000	60146.67	28.57
RSSI total		2850			56.26
TX part 1	200	500	400	1248.67	0.62
TX part 2	1500	250	3000	9319.50	2.33
TX part 3	1000	2500	2000	6224.67	15.56
TX part 4	4400	800	8800	27039.47	21.63
TX part 5	11900	750	23800	71046.97	53.28
TX total		4800			93.43
RX part 1	250	125	500	1560.75	0.19
RX part 2	500	1350	1000	3119.01	4.21
RX part 3	4000	750	8000	24672.64	10.50
RX part 4	9560	2125-10000	19120	57906.66	123.05-579.07
RX total		4350-11225			137.95-593.97

state, the average demand of the scheduling routine is computed. The results are reported in Table 6.4.

6.1.5 Task scheduling and execution

To evaluate the performance of EA-LSA, a test under simplified conditions is proposed. As a reference, the previously presented LSA and EDF are used. The tests are meant to compare the behaviour of the algorithms under optimal and sub optimal energy availability conditions, accounting that EDF and LSA are usually meant to manage a fixed workload, whereas EALSA is able to support the dynamic adaptation of the workload depending on energy availability.

On purpose a set of five fictional tasks has been created, based on the RF200T ability to retrieve data on temperature, supply voltage, RSSI, residual stored energy and power income. In the case of LSA and EDF, each task collects 50 samples of the mentioned quantities. In the case of EA-LSA, each task can collect a variable amount of samples, from a minimum of 20 to maximum of 80, depending on energy availability and power income. Each task, also, is

Table 6.3: Task steps energy consumption

Task step type	Task step duration (timer measurement) (μs)	Subroutine duration (μs)	CPU time (1 Mhz) (μs)	Subroutine energy consumption (μJ)	CPU energy consumption (μJ)	Task step consumption (μJ)
Volt measurement step (#04)	1025	830	195	1.27	0.26	1.53
Temp measurement step (#05)	3545	3340	205	6.71	0.27	6.98
RSSI measurement step (#06)	3164	2850	314	56.26	0.42	56.68
Power measurement step (#07)	630	0	630	0	0.84	0.84
Charge measurement step (#08)	328	0	328	0	0.43	0.43
Transmission and ACK reception step (#03) (best case)	10800	9150	1650	231.38	2.19	233.57
Transmission and ACK reception step (#03) (worst case)	17675	16025	1650	687.4	2.19	689.59

Table 6.4: Scheduler energy consumption

Scheduler Step Type	Duration	Consumption (μJ)
Step 1	51 ms	67.77
Step4	56 ms	74.42
Watchdog ISR	125 μs	0.1661
LPM3	~7 s	31.93
Scheduling routine (10 tasks)	256 time units	27443.24
Scheduling routine cycle (10 tasks)	~7 s	109.77
Scheduling routine mean (10 tasks)	1 s	15.68

required to send a data packet and wait for the transmission to be ACK by the AP. If the ACK is not received an optional retransmission may occur

The values in Table 6.3 have been rounded, with the exception of the exception of the best case of transmission and ACK reception, which has been increased to account the delay that the AP may introduce. The resulting energy demand of each step is reported in Table 6.5. Also the values of Table 6.5 are used by the harvester emulator, to account the energy consumption of the

Table 6.5: Energy consumption values (rounded) for each task step type (μJ)

Volt measurement step (#04)	Temp measurement step (#05)	RSSI measurement step (#06)	Power measurement step (#07)	Charge measurement step (#08)	Transmission step (#03) best case	Transmission step (#03) worst case
2	7	57	1	1	300	690

Table 6.6: Energy consumption values (rounded) for each task type (μJ)

Task type	**Minimum energy demand of tasks (EA-LSA)**	**Maximum energy demand of tasks (EA-LSA)**	**Energy demand of tasks (LSA and EDF)**
Volt measurements	730	850	790
Temp measurements	830	1250	1040
RSSI measurements	1830	5250	3540
Power measurements	710	770	740
Charge measurements	710	770	740

device at each subtask completion.

The scheduling cycle spans over an interval of 256 time units, each being 7 seconds long. At its beginning, two instances of the designated set are allocated to the scheduler. The deadlines fall within the cycle, they are 50 time units apart and depend on the task type. The scheduling process is restricted to two scheduling cycles to maintain plots intelligibility.

Regarding the energy management within the scheduling routines, a few assumptions have been made. Concerning EA-LSA and LSA, the prediction routine assumes that the power income remains constant in the near future, based on the fact that a deadline fall every about 350 s.

Also, in order to simplify the comparison of the results, the event of a device reset has been prevented, since it also reset the scheduling process. On purpose, LSA and EDF have been implemented so that the execution of a task step is postponed whenever the demand exceed the available energy. In particular while LSA relies on the expected energy production, EDF only relies on the stored energy. As a consequence, in the form EDF has been implemented, it may be addressed as earliest deadline as soon as possible [83].

Each task consumption is presented in Table 6.6 and it has been computed by accounting the data collection and a single transmission assuming the time out of the ACK. To address the case of an unforeseen retransmission request, an optional retransmission is allowed. In particular the scheduler expects the energy demand of the transmission worst case, whereas the actual demand is computed by the harvester emulator based on the best case.

The energy demand of the scheduling routine equals to 16 μJ per second, whereas about 24 μW are required to complete the fixed workload including the optional retransmissions. Therefore, in the first part of the test the energy production is assumed to be constant. In this case also, a power output of about 24 μW is suggested to evaluate the algorithm performance under ideal conditions, a power output of about 21 μW is used to emulate harvesting adverse conditions, and a power output of about 27 μW is used to evaluate the scheduling under favourable conditions. In the second part of the test, on the other hand, a variable production profile is used.

Test Results

The data collected during a scheduling instance is presented in two graphs. The first plot, in the upper side, reports the execution of each completed task step over time. Each different task step is coded with its own stem length. The retransmission on the other hand is marked by a crossed stem. The second plot, in the lower side, shows the amount of the energy stored on the capacitor over time.

The test routine spans over two scheduling cycles. In each cycle two instances of the assigned task set are scheduled. In the first test, the emulated harvester power output has been assumed equal to 24 μW. That amount is enough to complete all of the task, including the optional retransmission. The performance of EDF, LSA and EA-LSA is reported in Figs. 6.7, 6.8 and 6.9, respectively.

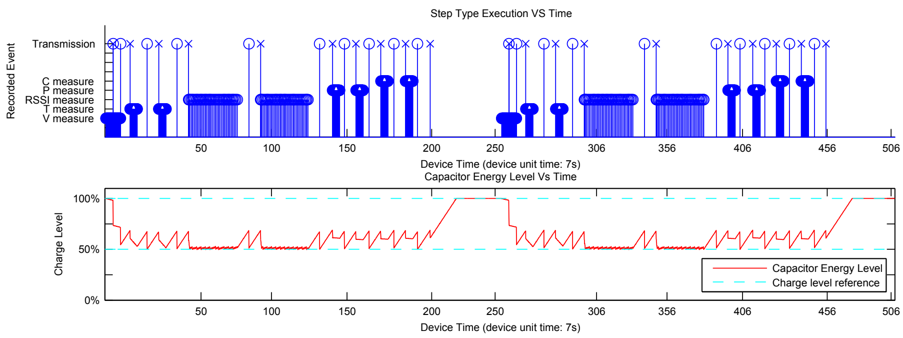


Figure 6.7: EDF algorithm. Energy harvester simulated power output: 24 μW

Every algorithm under test is able to collect fifty samples per task, and complete both transmission and also retransmission. In this particular case, in fact, the execution conditions implemented in EA-LSA are meant to maximize the number of completed task, rather than the number of collected samples, thus promoting the completion of multiple the task set instances. Different conditions, however can also be assigned depending on circumstances.

The main difference concerns the energy management carried out by EDF, with respect to the energy management of LSA and EA-LSA. While EDF

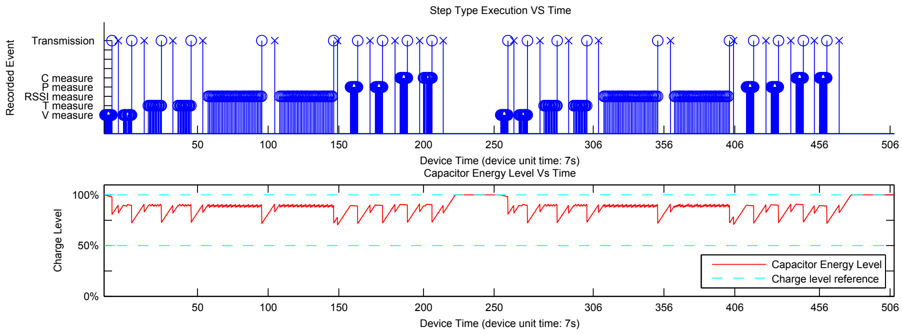


Figure 6.8: LSA algorithm. Energy harvester simulated power output: 24 μW

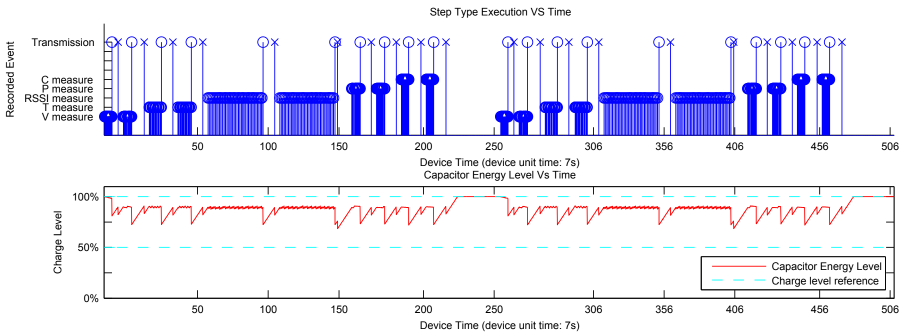


Figure 6.9: EA-LSA algorithm. Energy harvester simulated power output: 24 μW

executes a task step whenever the stored energy is sufficient to complete said step, which results in the stored energy being kept to its minimum, both LSA and EA-LSA are able to preserve as much energy as possible.

Adverse conditions are emulated assuming a power output equal to 21 μW. Again the performance of EDF, LSA and EA-LSA is reported in Figs. 6.10, 6.11 and 6.12 respectively.

In the current test, the lack of energy does not permit the completion of the intended workload. In fact, the RSSI sample collection task can be executed only once per scheduling cycle, rather than twice, notwithstanding the scheduling algorithm. Nonetheless different results are achieved with regards to transmissions and retransmission within said task. In particular, EDF and LSA are not able to transmit at all in the second scheduling cycle. On the other hand, EA-LSA completes transmission and retransmission in both scheduling cycles, since it lowers the number of collected samples in order to save the energy required to transmit.

As such, the dynamic workload adaptation granted by EA-LSA improves

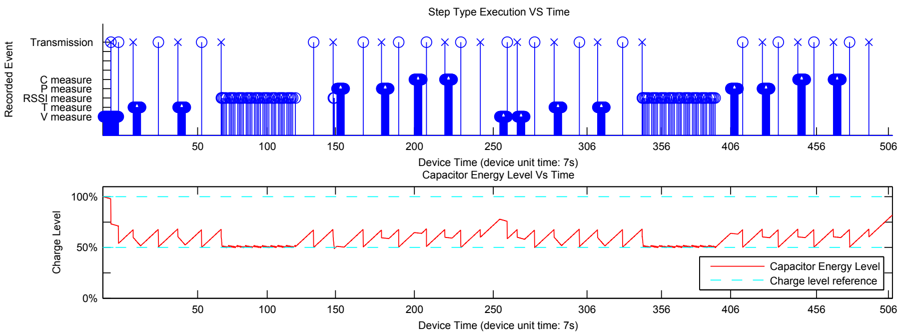


Figure 6.10: EDF algorithm. Energy harvester simulated power output: 21 μW

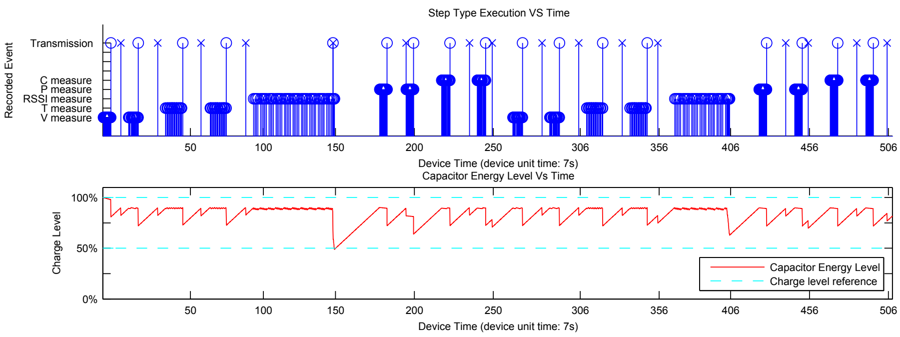


Figure 6.11: LSA algorithm. Energy harvester simulated power output: 21 μW

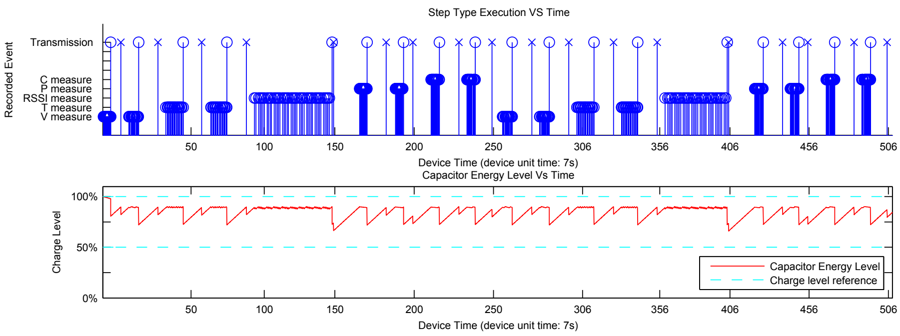


Figure 6.12: EA-LSA algorithm. Energy harvester simulated power output: 21 μW

the device efficiency and throughput, with respect to fixed workload scheduling algorithms. The ability to execute the assigned task to completion, although through a simplified routine, may prevents the waste of collected energy. If the assigned tasks do not depend on one another, is either executed to completion or not executed at all, thus no energy is usually wasted. If the tasks are interdependent, that is a task set is of use only if it is executed in its entirety, energy is still wasted if only a subset of tasks is completed. In this case, a proper design of tasks and their execution policy, which depend on the application, may be required to avoid such an event.

Favourable conditions are emulated assuming a power output equal to 27 μW. In this case the performance of EDF, LSA and EA-LSA is reported in Figs. 6.13, 6.14 and 6.15 respectively.

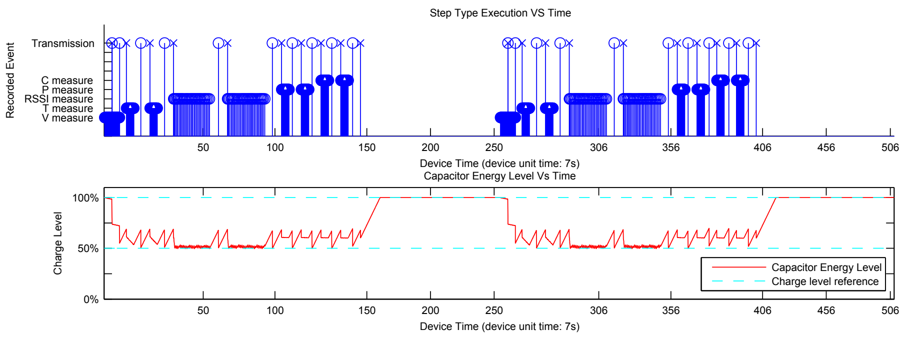


Figure 6.13: EDF algorithm. Energy harvester simulated power output: 27 μW

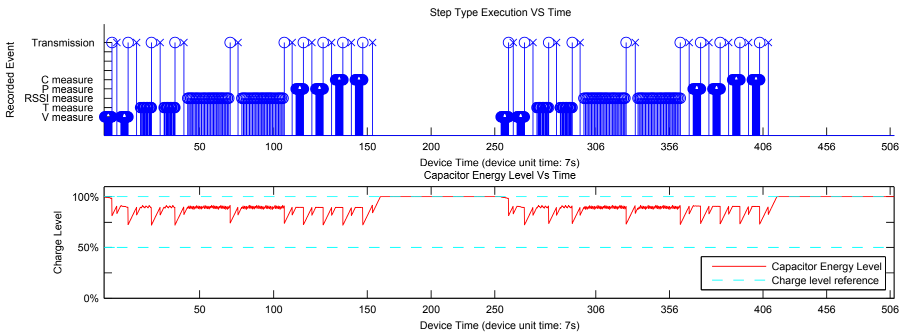


Figure 6.14: LSA algorithm. Energy harvester simulated power output: 27 μW

Similarly to the ideal case, the algorithms under test are able to complete all the assigned tasks. In this case, however, EA-LSA is able to increase the workload by collecting additional samples thus exploiting the energy surplus. At the end of each scheduling cycle, the energy storage remains at full charge, meaning that the renewable energy collected by the harvester cannot be stored, and thus is lost. On purpose, EA-LSA increases the device workload, so that

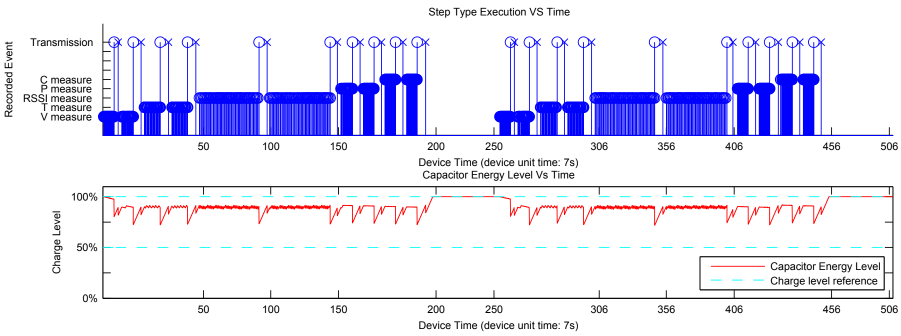


Figure 6.15: EA-LSA algorithm. Energy harvester simulated power output: 27 μW

the energy loss can be contained by completing the tasks with enhanced results.

To model a variable level power output, the profile presented in Fig. 6.16 has been used. As done before the performance of EDF, LSA and EA-LSA is reported in Figs. 6.17, 6.18 and 6.19 respectively.

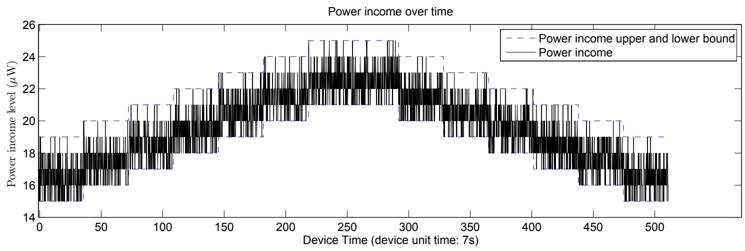


Figure 6.16: Energy harvester simulated power output profile

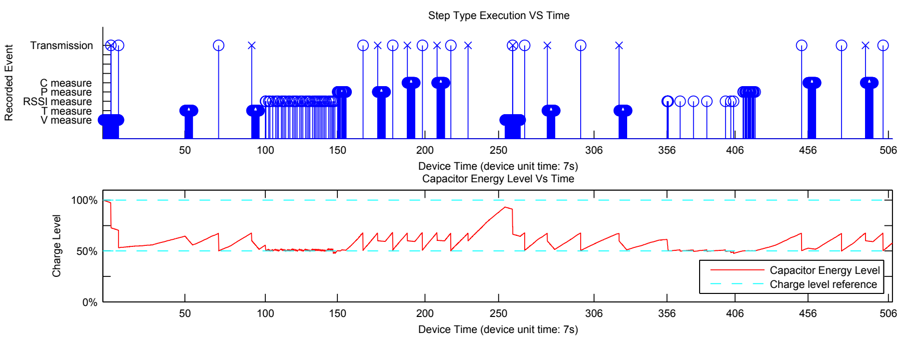


Figure 6.17: EDF algorithm. Energy harvester variable power production.

In the current test, the average power output spans from about 17 to about

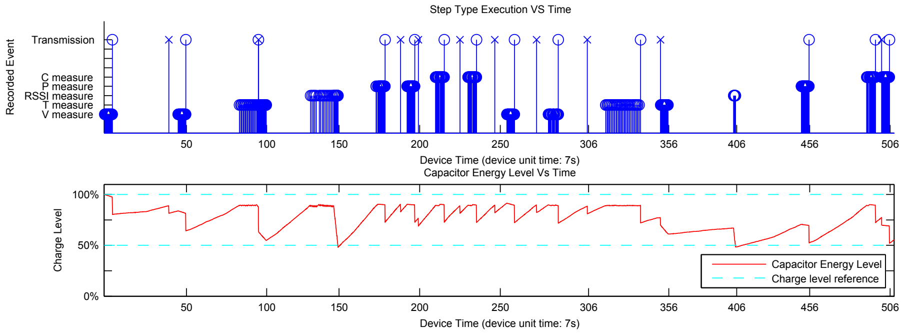


Figure 6.18: LSA algorithm. Energy harvester variable power production.

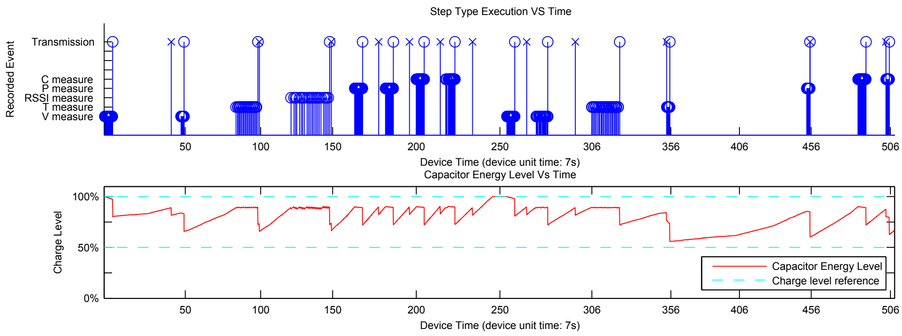


Figure 6.19: EA-LSA algorithm. Energy harvester variable power production.

23 μW, therefore the available energy amount is not enough to execute all the assigned tasks to completion. As a major consequence, the three algorithms fail to complete any set of tasks in the second scheduling cycle. A difference can be highlighted, however, in the behaviour of EA-LSA with respect to the behaviour of EDF and LSA. For instance, both EDF and LSA collect several temperature samples and RSSI samples. In both cases, however, the lack of energy prevent the device form transmitting the collected information. Additionally, in the second scheduling cycle, this behaviour prevent the device from retransmiting the information after the collection of power samples.

EA-LSA on the other hand, can lower the workload reducing the number of collected the temperature samples, thus completing the task. Regarding the RSSI samples, on the other hand, in the first scheduling cycle the available energy allows for the completion of the RSSI task, including both transmission and retransmission. In the second scheduling cycle, however, the lack of energy prevents the device from collecting the minimum number of samples and transmitting the related information, therefore the task is discarded altogether. The spare energy, however, can be used to complete the following task, retransmitting the data packet.

The results are briefly resumed in Table 6.7, in particular, the first three rows report the number of completed tasks, whereas the values between parentheses refer the completed set of tasks. In the last three rows, on the other hand, the number of collected samples has been reported. In this case, the amounts between parentheses account the collected samples that belong to incomplete tasks.

Table 6.7: Performance of scheduling algorithms

	21 μW	24 μW	27 μW	Variable power output
	Completed tasks (and sets)			
EDF	17 (1)	20 (2)	20 (2)	13 (0)
LSA	17(1)	20 (2)	20 (2)	13 (0)
EA-LSA	18 (2)	20 (2)	20 (2)	15 (1)
	Collected samples (and unused ones)			
EDF	908 (58)	1000 (0)	1000 (0)	808 (158)
LSA	908 (58)	1000 (0)	1000 (0)	807 (157)
EA-LSA	881 (0)	1000 (0)	1600 (0)	598 (0)

In the proposed tests EA-LSA has provided a more efficient use of the available energy with respect to either EDF and LSA. In particular we can observe that:

- in adverse conditions, no energy has been spent in infeasible tasks;
- in favourable conditions, part of the energy surplus has been used to increase the device workload.

6.2 Leakage detection

In water and gas distribution systems, leakages can cause of resource waste and damages, thus the ability to identify leakages can shorten the time required to repair the problem lowering the amount of wasted resources and the amount of damages.

However, although several leakage detection technique have been devised and implemented, most of them are invasive and can be applied effectively only to large distribution systems, in that the aggregated flow show a limited and slow paced swing over time.

On the other hand, in residential environments, the water flow shows a large and fast paced swing, and moreover the implementation of traditional monitoring systems are costly and invasive.

With the availability of automatic meter reading systems, real time water consumption monitoring has become possible, therefore a single point monitoring system can be used. By means of the collected information, then, Novelty Detection can be used to detect abnormalities in the trend of water consumptions, and therefore leakages.

6.2.1 Literature background and proposed approach

A recent application for leakage detection with household consumption has been addressed by Nasir *et al.* [99]. In their method, Artificial Neural Network (ANN) and Support Vector Regression (SVR) have been adopted to estimate the leakage size and location. The domestic network has been modelled and simulated with EPANET[4], acquiring the simulated data with 2 pressure sensors, 2 differential pressure sensors, and 2 flow sensors. Over $3,600$ scenarios have been realized (equally subdivided in training and test), producing leakages with different sizes and locations. The ANN has been realized assuming 6 input nodes, one hidden layer with 20 nodes, and two output nodes. The ϵ-regression with Radial Basis Function (RBF) kernel has been adopted for the SVM model. The prediction performance has been evaluated in terms of Mean Square Error (MSE) and Squared Correlation Error coefficient (R). The proposed quasi-static analysis confirmed the good behaviour of the SVM and its resilience to sensor measurement errors.
Leakage detection in domestic water systems has been also addressed by Oren and Stroch [100] and a mathematical model, based on the deviation from the average consumption, has been developed. Two thresholds, a minimum one and a maximum one, have been set for the deviation level. The leakage is detected when the following conditions are both true: $C(T_1) > MD_{T_1}$ and $C(T_2) > MD_{T_2}$, where MD_t denotes the threshold for the time instant t and $C(t)$ is the corresponding consumption. *Statistical hypotheses*, from the average domestic water usage of Israel, U.S., and Germany[5], have been extracted and used to compute the threshold values. Therefore, the statistical information has not been obtained from the sequence itself. To evaluate the approach, a consumption sequence with a minimum resolution of 15 minutes has been considered, and 2 leakages have been separately induced: a burst (cracked pipe) and a *steady consumption*. No further leakages have been tested and no evaluation criteria have been presented.
A novel change-detection test (CDT) has been developed by Boracchi and Roveri [101]. The approach has been developed in order to detect structural changes in time series and it has been tested with time series of flow measurements. The data was collected from the Barcelona water distribution network,

[4] `www.epa.gov/nrmrl/wswrd/dw/epanet.html`
[5] `www.water.gov.il` - `www.waterrf.org` - `www.umweltbundesamt.de`

over a time window of 82 days and with a resolution of 10 minutes. The series have been manipulated to simulate different types of alterations: leakage, sensor degradation, source change, and stack-at (constant measurements after a fixed time instant). Concerning the leakage, an offset proportional to the average flow during the normal phase is added. The detection performance has been evaluated in terms of False Positive Rate (FPR), False Negative Rate (FNR) and Detection Delay (DD). Two offset values have been tested, that is 25% and 50% of the mean flow, achieving a FPR of 0.1 and a FNR of 0.0. The detection of the greater leakage presents a lower delay than the smaller one, that is 156.4 and 914.2 samples respectively. The decision of the detection has been based on the whole test set.

In order to identify the onset of a leakage, Gamboa-Medina *et al.* [102] investigated a set of features extracted from a water network. Energy (ENE), entropy (ENT), zero crossing count (ZCC), and distribution of energy in the components of wavelet decomposition (WDE) are the adopted features. They have been computed from the pressure data collected by 15 sensors, that have been connected to an experimental circuit with a total length of $200\,m$. A total of 620 scenarios, 310 with losses and 310 without losses, have been collected with 1 minute recordings at 4 sample per second. The features distributions have been analyzed, and a patter recognition system, based on a binary classifier, has been employed for the detection. The best result has been achieved by the vector composed of all the features, and the WDE has been shown the best result among the single features.

A different approach, based on the Minimum Night Flow (MNF), has been widely used to detect leakages in monitored district metered areas (DMAs). In order to determinate the factors that affects the MNF, Jaber *et al.* [103] adopted a method based on multiple linear regressions. The study has been conducted with the data collected over 361 DMAs, monitoring flow and pressure for $24\,h$ every 15 minutes. The MNF was modelled in order to estimate the water loss in the DMAs, and linear correlations between loss and number of connections, total length of pipe, and weighted mean age pipe were explicated.

In order to detect and localize leakages in water network, Sanz *et al.* [104] proposed a Fuzzy Inductive Reasoning (FIR) approach. The method is composed of four stages: fuzzification, qualitative modelling, qualitative simulation, and defuzzification. The leakages-free data has been collected by two pressure sensors located in the District Metered Area, Nova Icària, in Barcelona and applied to the EPANET[1] model of the network. Over 30 days of data have been simulated, and split in training set (non-faulty) and test set (faulty). A loss of approximately $6.3\,L/s$ has been artificially introduced in the test set.

As seen so far, the literature thrives with applications developed to deal with the water leakage detection, specifically devised for district-sized networks or even larger ones. The bottleneck to the development of innovative approaches is surely due to the lack of suitable databases, as highlighted by Fagiani *et al.* [105], because of either the incoherent data in the database, or the unavailability of the database altogether. Furthermore, among the state-of-the-art approaches, many of them have been developed using different sets of measurements, such as flow and pressure data, or both data collected from multiple sensing point in the network [104, 103, 99, 102]. In this way, these methods result unsuitable for household scenarios, which present a single flow sensing point. In the end, only two studies addressed the leakage detection problem using single flow measurements, [100] and [101]. Both approaches are based on statistical characterization, but, as discussed above, in the former the detection thresholds have been sized based on unrelated references, that is, unrelated customers, resulting in the detection of only two leakages in the presented validation. The latter presents a CDT method suited for domestic consumption, unfortunately this method has been tested only against sequences collected in DMA, that present a smoother pace than the domestic consumption.

In this work the authors propose a leakage detection approach based on statistical modelling, meant for residential distribution of both water and natural gas. The method is inspired by the audio novelty detector presented by Ntalampiras *et al.* [106], and recently revised for a real-time implementation by Principi *et al.* [107]. The approach aims at the detection of leakages in household environments, and for this reason it has been evaluated using the Almanac of Minutely Power Dataset (AMPds) [108]. The dataset is a publicly available multi-year data record[6]. Currently, the data spans over 2 years and addresses a single house. Electricity, water, and natural gas consumption have been recorded, with a sampling rate of one sample for minute, and a total of $1,051,200$ readings for each resource. The tests have been conducted with 1 and 10 minutes resolution data. The sets have been divided in overlapped frames, and a collection of features, partially selected from the ones presented in [102], has been extracted. The features extracted form the training set, have been exploited to model the normality background, with both Gaussian Mixture Model (GMM) and Hidden Markov Model (HMM). During the evaluation phase, the models have been used to compute the likelihood of each frame and a threshold has established the presence of leakage. A variation of the Sequential Forward Selection (SFS) [109] has been executed in order to find the best features combination.

The sets have been manipulated assuming a leakage size between 25% and 50% of the average consumption in the train set, accordingly to Boracchi and Roveri [101]. In order to verify the realistic behaviour of the artificial leakage, the considerations reported by Britton *et al.* [110], have been taken into

[6] `ampds.org`

account. They collected the behaviour of over 22,000 residential households in Hervey Bay, between March and May 2009, and discovered that the most common leakage rate was $20\,L/h$ followed by $10\,L/h$. In general, over 49% of leakages are $20\,L/h$ or less. So, considering water consumption in the adopted dataset (AMPds), 25% and 50% of the average consumption correspond to $0.13\,L/m$ and $0.26\,L/m$, respectively. Therefore, they are equivalent to leakages in L/h of 7.76 and 15.53, respectively, confirming the reliability of the adopted sizes.

Up to the authors' knowledge, no other works have addressed the proposed novelty detection approach for leakage identification. In particular, it seems that a suitable collection of features has never been investigated on purpose. Therefore, a set of features has been studied for the present problem and proposed as a valid starting point for future developments. Furthermore, differently from the state-of-the-art methods, probabilistic models, i.e., GMM and HMM, have been employed and the proposed approach executes an on-line frame-based detection, or a frames sequence-based detection. So, being the network constantly tracked, both abnormal and normal states are detected, and the leakage start and end points are obtained, allowing a real-time monitoring, in agreement with [107].

The *true detection rate* (TDR) and the *false detection rate* (FDR) with a frame-based evaluation, for the GMM, and with a frames sequence evaluation, for the HMM, have been computed. Finally, executing the detection decision, leakage/no-leakage, for different threshold values, the *Receiver Operating Characteristic* (ROC) has been created, and the best features combinations have been evaluated in terms of *Area Under Curve* (AUC).

6.2.2 The novelty detection algorithm

The algorithm consists of two main stages: the creation of the normality model and the leakage detection. In the former, a normal scenario, that is leakage-free, is analyzed, and a representative background is created. In the latter, the decision process is accomplished, over a faulty set, evaluating the likelihood of the normality model.

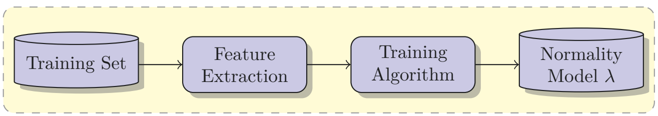


Figure 6.20: Block diagram of the normality model creation stage.

The background creation, as depicted in Figure 6.20, is characterized by two phases: the feature extraction and the training phases. In the former, the training dataset is opportunely elaborated, frame partitioning, and the selected features are extracted from each frame and arranged in a vector. In the latter, all feature vectors are processed in order to compute the statistical model of the normality behaviour. Two different statistical models have been evaluated in the presented approach: Gaussian Mixture Model (GMM) and Hidden Markov Model (HMM).

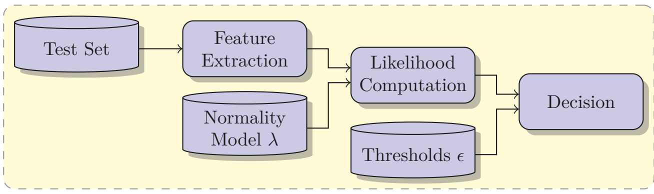


Figure 6.21: Block diagram of the leakage detection stage.

In the second stage, depicted in Figure 6.21, the features extraction process is replicated with the test set, that has been altered by the introduction of a leakage, that is, an abnormal event. Employing the features vectors into the normality models, likelihood values are computed, and a frame-based or a sequence-based decision is taken.

Features

Input data (DATA), moving average (MA), energy (ENE), distribution of energy in the components of wavelet decomposition (WDE), and logarithmic energy in the components of wavelet decomposition (LWE) are the adopted features. Specifically, in order to have a consistent evaluation, the data "as is" and the moving average of the set have been also taken into account. Furthermore, even the *first order positive differences* have been evaluated, for an overall total of 10 features.
The DATA feature is composed by the whole set of frame samples. On the other hand, the average value of the frame samples is addressed to as MA. The collection of the MA values, computed for the whole set, corresponds to the moving average of the set for a window equal to the frame length. The frame energy, ENE, is computed as:

$$ENE = \sum_{i=1}^{N} {x_i}^2 \ , \tag{6.1}$$

where N is the number of samples in the frame, and x_i denotes the value of the i-th sample. The WDE feature has been presented in [102], and it is computed using the wavelet decomposition of order 3 with the function

Daubechies 2 (*db2*). One approximation sequence (A_3) and three detail sequences (D_3,D_2,D_1) are obtained as result of the decomposition. The energy is computed for each sequence, as in equation (6.1), with N denoting the number of elements in the sequence. Then, the feature is composed by reporting the percentage value of the energy of each sequence respect to the total energy of the sequences. Therefore, for each frame a vector of 4 elements is obtained. Let C be a vector, given by the concatenation of the wavelet decomposition coefficients:

$$C = [A_3, D_3, D_2, D_1] \ , \tag{6.2}$$

then, the i-th element of the WDE feature is defined as:

$$WDE_i = 100 \cdot \frac{E_i}{E_T} \ , \tag{6.3}$$

where E_i denotes the energy of the corresponding i-th wavelet coefficient, and E_T is the sum of the energies of the sequences. Adopting the component energy E_i, the i-th element of the LWE feature is achieved by applying the following:

$$LWE_i = log(E_i + 1) \ . \tag{6.4}$$

The *first order positive differences* are indicated with the name of the feature preceded by Δ. They are computed, for $n = 1, 2, \ldots, N$ and $l = 1, 2, \ldots, L$, as:

$$\Delta F(n, l) = \begin{cases} F(n, l) - F(n-2, l), & \text{if } n \geq 2 \\ F(n, l). & \text{otherwise} \end{cases} \tag{6.5}$$

where F represents the features matrix, that is composed of N features vectors of length L, L is the number of the features elements in the vector, and N is the number of frames for the evaluated set.
Except for DATA and its *first order positive difference* (ΔDATA), which lengths depend on the number of frame samples, the remaining features have a fixed length that is independent from the dataset resolution and the frame length. Specifically, MA, ENE, WDE and LWE present a feature length of 1, 1, 4 and 4, respectively.

Normality Models: GMM and HMM

The features extracted from the training set represent the normality scenario, and its statistical behaviour, the background model, has been represented using both the GMM and the HMM. For the background modelled with the GMM, the multivariate normal distribution is obtained as:

$$g(\mathbf{x}|\mu, \Sigma) = \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma|^{\frac{1}{2}}} \exp^{\{-[(\mathbf{x}-\mu)^T \Sigma^{-1} (\mathbf{x}-\mu)]/[2]\}}, \tag{6.6}$$

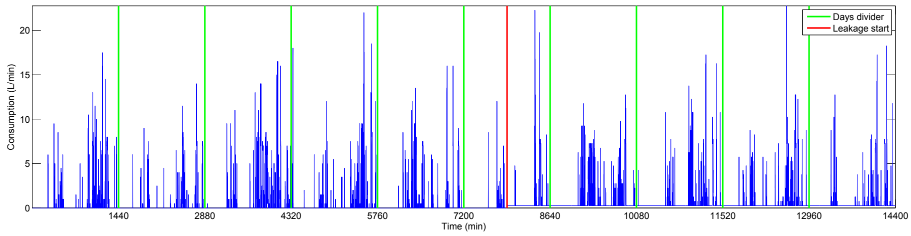


Figure 6.22: The figure reports 10 days of data recorded in the AMPds dataset, and their alteration due to the leakage introduction for the test phase.

where $\mathbf{x}$ denotes the features vector, μ is the mean values vector, and Σ represents the covariance matrix. Finally, the GMM is obtained by a weighted sum of (6.6), as:

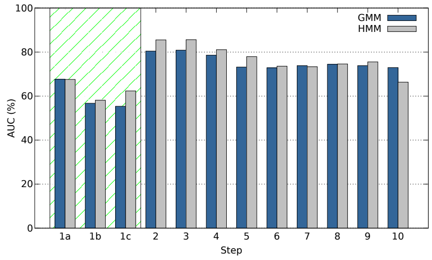
$$p(\mathbf{x}|\lambda) = \sum_{j=1}^{N_g} w_j g(\mathbf{x}|\mu_y, \Sigma_j), \tag{6.7}$$

where $w_j \in [0, 1]$, $\sum_{j=1}^{N_g} w_j = 1$, and $\lambda = \{\mu_j, \Sigma_j, w_j\}_{y=1}^{N_g}$. N_g denotes the number of Gaussians adopted for the GMM. During the training phase, the GMM parameters are initialized by the k-means algorithm, that splits the data in a number of clusters equal to the number of adopted Gaussian components. Then, a final model is achieved applying the Expectation-Maximisation (EM) algorithm [111], that estimates the parameters and the weight of the components guaranteeing a monotonous increase of the GMM likelihood. All GMMs have a diagonal covariance matrix.

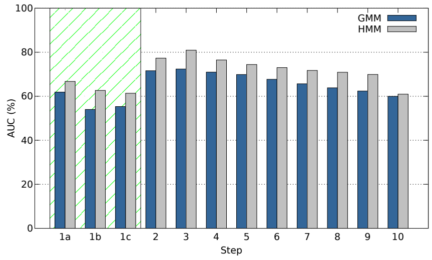
With regard to HMM modelling process, the left-right structure, commonly used in speech recognition problems [112], has been adopted. Considering a general structure, the i-th state is characterized by a transition probability from itself to the state j, denoted with the discrete probability a_{ij}. In addition, let $\mathbf{O} = \{o_1, o_2, \cdots, o_N\}$ be an observations sequence, the same i-th state is characterized by an emission probability of the observation o_n, at the instant n, given by the probability density $b_i(o_n)$. Therefore, the likelihood of the observations sequence, for the whole Markov chain, is given by:

$$p(\mathbf{O}, M) = \sum_{X} a_{x(0)(1)} \prod_{n=1}^{N} b_{x(n)}(o_n) a_{x(n)x(n+1)}. \tag{6.8}$$

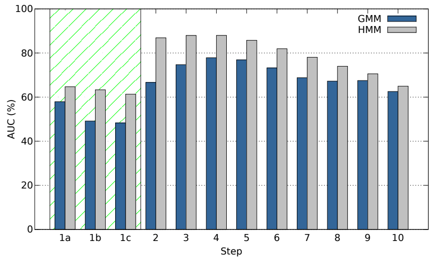
where M is the assumed Markov model, and X is the states sequence. The emitting states are modelled by a diagonal GMMs, and each relation between the states are trained by means of the *Baum-Welch* algorithm.



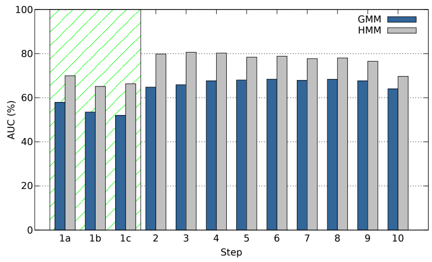
(a) Natural Gas, 1 minute resolution.



(b) Natural Gas, 10 minutes resolution.



(c) Water, 1 minute resolution.



(d) Water, 10 minutes resolution.

Figure 6.23: Features selection steps for all the evaluated cases.

Decision

In the decision phase, the likelihood of the extracted features vector is computed adopting the normality model. Then, the likelihood is compared against a threshold, in order to detect the leakage presence. Specifically, for the GMM background, a frame-by-frame decision is performed. For each input frame, the likelihood is obtained as result of the computation of the equation (6.7), assuming as input the corresponding features vector. If the obtained likelihood exceeds the threshold, the event represented by the frame is marked as leakage/abnormal event, otherwise it is considered a normal event. Being the HMM based on a sequence of observations, consecutive frames are dragged in the decision process. In particular, the number of frames involved is equal to the states number of the normality background. Each frames sequence is considered as a single event, and if the majority of the frames are detected as leakage occurrences, the event is marked accordingly.

6.2.3 Computer simulations and result analysis

Experiments have been conducted for both water and natural gas, using a fixed frame length, 5 hours, and two dataset resolutions: 1 minute and 10 minutes. Consequently, each frame is composed of 300 and 30 samples, respectively, with an overlap equal to 2/3 of the frame length. A label vector is created for the test set, the normal frames are marked as "0", conversely the abnormal frames are indexed as "1". For the HMM, the label vector has been adequately modified to accomplish a sequence-based evaluation.
In order to properly evaluate the slight changes due to the random initial conditions, 10 background models have been trained. Furthermore, each background model has been used to identify 10 random leakages, therefore, an overall of 100 losses have been evaluated in total. Specifically, the random parameters of the artificial leakage are the size, the starting sample, and the length. The sequences have been manipulated following the solution adopted in Boracchi and Roveri [101]. So, the test set $s(n)$ has been altered as:

$$s(n) = \begin{cases} s(n) + \alpha \cdot l_k, & \text{if } n \geq N^* \\ s(n), & \text{otherwise} \end{cases} \tag{6.9}$$

where $\alpha \in [0.25, 0.50]$, l_k is the average consumption computed from the training sequence, and N^* denotes the leakage start sample. As discussed in Section 6.2.1, the range of the flow loss, $\alpha \cdot l_k$, is conformed to a real-cases leakage [110]. Being the first sample and the length of the leakage randomly selected, the start and end sample may not match with the beginning and the end of a frame. Therefore, a border frame is marked as novelty occurrence, in the reference labels vector, if the majority of its samples result manipulated, according to the condition expressed in (6.9). The leakage length is randomly selected between 5 and 10 hours, that corresponds to 300 and 600 samples, respectively, for 1 minute resolution, and to 30 and 60 samples, respectively, for 10 minutes resolution. Moreover, the initial point of the leakage is randomly selected over a span between 10% and 90% of the set length. A partial sequence (10 days) extracted from the dataset, that shows the effect of the leakage introduction, is illustrated in Figure 6.22.

The GMMs have been tested varying the number of adopted Gaussian components as: $N_g = \{2, 4, 8, 16, 32, 64, 128, 256\}$. For the HMMs, in addition to the Gaussian components, the states number has been varied as well, from 1 to 4 (excluding the start and end states). The datasets have been split in training and test sets, composed of 70% and 30% of the data, respectively. The training sets have been used to create the background models, whereas the test sets have been adopted to evaluate the system detection performance. The data preparation, the feature extraction and the decision stages have been developed in MATLAB®. The GMM and HMM training algorithms, and the likelihood computation have been implemented in C++ using the Torch3[7] library.

The system performance has been evaluated in terms of *true detection rate* (TDR), *false detection rate* (FDR), and *Area Under Curve* (AUC). The *true detection rate* is defined as the conditional probability of correctly detect an anomaly. The *false detection rate* defines the probability of detect a normal event as anomaly. Specifically, they are computed from the detections achieved over the test set, as:

$$TDR = \frac{\text{no. of abnormal events detected as abnormal}}{\text{no. of abnormal events}}, \tag{6.10}$$

$$FDR = \frac{\text{no. of normal events detected as abnormal}}{\text{no. of normal events}}. \tag{6.11}$$

where, for the GMM, one event corresponds to one frame, whereas, for the HMM, one event corresponds to a sequence of frames, that is equal to the number of tested states. The number of occurrences considered as *true positive* (abnormal events detected as abnormal) or *false positive* (normal event detected as abnormal) are computed comparing the reference labels with the labels obtained after the decision process.

The Receiver Operating Characteristics have been obtained varying the detection threshold between the minimum and the maximum value of likelihood returned by the test data. Specifically, setting a threshold value lower than the minimum likelihood, the system is having a TDR and a FDR equal to 0%.

[7] `torch.ch/torch3/`

Each normal event is correctly detected, but all the abnormal events are classified as normal as well. On the contrary, setting the a threshold value higher than the maximum likelihood, both the TDR and the FDR reach a 100% rate. This means that each sequence, abnormal or not, is recognized always as abnormal. In the features selection experiments, the tested combinations have been evaluated in terms of AUCs, computed from the respective ROCs.

Selection of the best features combination

The pursuit of the best features combination is a time-consuming task, and a full-search is an unaffordable process. In fact, let m be the number of all the available features, and l the number of features for each vector combination, the overall number of tests is given by the binomial coefficient: $m!/l!\,(m-l)!$. For this reason, several suboptimal searching techniques have been proposed, and the adopted methods is based on the Sequential Forward Selection (SFS) [109]. In the SFS, the first step consists on the evaluation of each feature. After that, an *ex novo* set of features vectors is created coupling the "winner" feature, i.e. the one whose performance is the greatest, with each other feature. This procedure is repeated by selecting the "winner" vector, instead of the "winner" feature, but adopting the same criterion, and using it to create the new set of vectors. In this way, each new iteration reduces the number of the evaluated vectors by one, increasing the features in the vectors. So, applying this strategy, the required iterations are reduced to: $lm - l(l-1)/2$. Unfortunately, due to the low performance achieved for the first iteration, partially depicted in Figure 6.23, the chances of missing a good combination are considerably increased.
For these reasons, a slight variation has been introduced in the SFS. At the first step, instead of creating a new set of vectors from one winning feature, three new sets are created. The features that achieved the three better results are selected, and each of them is used to create the new combinations. The three best results achieved, at the first step, are reported with the labels 1a, 1b, and 1c in Table 6.23. The proposed selection remains a suboptimal search method, but clearly requires less computational resources than the full-search method.

In the selection experiments, the AUC averaged over all the tested backgrounds and leakages has been used as evaluation criterion. The best result of each step is reported in Figure 6.23. For the natural gas at 1 minute resolution, the best performance at the first step, for both GMM and HMM, has been achieved by DATA, ΔDATA, and MA features. At 10 minutes resolution, DATA, ΔDATA, and MA reached the best results for the GMM, whereas DATA, MA, and LWE for the HMM. For the water, DATA, ΔLWE, and LWE represented the best features at 1 minute resolution for the GMM, and MA,

Table 6.8: Best results and corresponding features combination achieved for each resource and resolution. The "Parm." column reports the number of Gaussians adopted for the GMM, and the states and Gaussians number for the HMM, respectively.

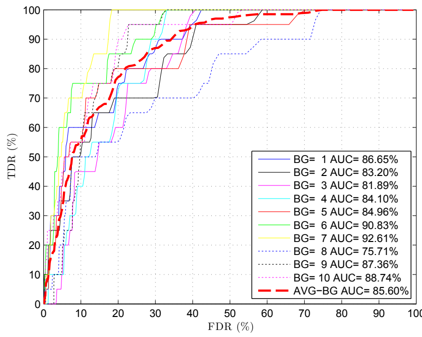
Res.	R.	Features comb.	AUC (%)			Model	Param.
			AVG	BEST BG	BEST		
G	1	MA+ENE+ΔMA	80.87	$87.52_{(7)}$	$97.43_{(10/8)}$	GMM	64
G	1	MA+ENE+LWE	85.60	$92.61_{(7)}$	$99.24_{(2/2)}$	HMM	3-256
G	10	MA+ENE+ΔMA	72.37	$79.96_{(7)}$	$96.13_{(8/2)}$	GMM	64
G	10	LWE+MA+ENE	80.93	$89.79_{(7)}$	$99.68_{(1/8)}$	HMM	4-32
W	1	LWE+MA+ ENE+ΔMA	77.85	$82.50_{(2)}$	$97.73_{(1/2)}$	GMM	256
W	1	MA+ENE+ LWE+ΔMA	87.97	$93.90_{(6)}$	$99.87_{(1/8)}$	HMM	4-64
W	10	DATA+ΔENE+ ΔLWE+LWE+ ENE+WDE	68.41	$73.27_{(8)}$	$93.75_{(2/5)}$	GMM	256
W	10	MA+LWE+ENE	80.63	$89.82_{(1)}$	$99.87_{(5/1)}$	HMM	4-256

DATA, and LWE for the HMM. At 10 minutes resolution, DATA, ΔDATA, and LWE are the best features for the GMM, and DATA, MA, and LWE for the HMM.

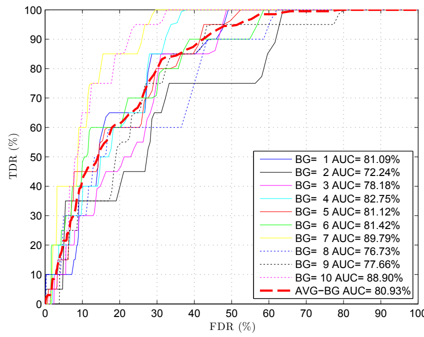
The best combinations for each tested scenario and the corresponding model parameters are reported in Table 6.8, together with average AUC in the column "AVG". For each resolution, the HMM approach has achieved better results than the GMM. In both water and natural gas cases, a slight worsening of the detection has been observed at lower resolution. Among the considered features, the best results have been reached, mainly, by the MA - ENE pair combined with LWE or ΔMA features. Finally, the selected combinations confirmed that using only the average consumption, MA, it is not possible to correctly detect leakages in the addressed scenario. Among the tested combinations, for both water and natural gas cases, the worst feature has been ΔDATA. Specifically, it has been always the feature added last to each vector in the selection process.

Additional considerations

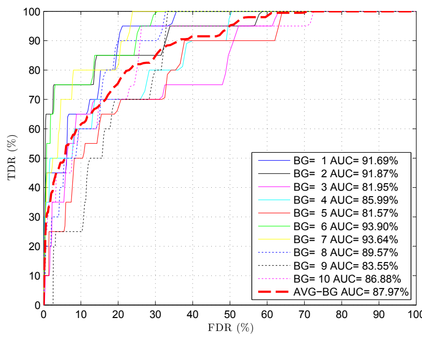
In order to have a better insight of the detection performance, in Table 6.8 are also reported the best AUCs achieved among the background models (column "BEST BG"), and the overall best AUC (column "BEST"). The former refers to the best result achieved as average of the AUCs, obtained for all the tested



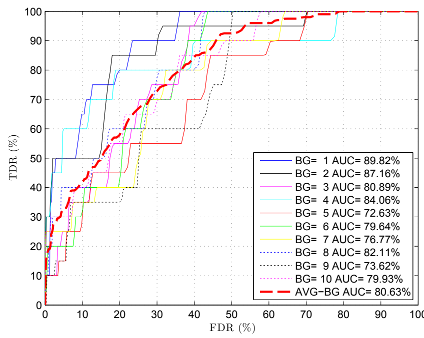
(a) Natural Gas, 1 minute resolution.



(b) Natural Gas, 10 minutes resolution.



(c) Water, 1 minute resolution.



(d) Water, 10 minutes resolution.

Figure 6.24: ROC for the best achieved result with each Background HMM model. The red dotted line represents the overall mean curve.

leakages within the same background model. The latter shows the best AUC achieved among the detected leakages, regardless of the adopted background model. The background number, for the “BEST BG”, and the background and the leakage number, for the “BEST”, that have achieved the reported result, are indicated in brackets. The ROCs concerning the average AUCs achieved for each HMM background model, among all the tested leakages and the overall averages, are depicted in Figure 6.24.

Furthermore, the better behaviour for the HMMs, with respect to GMMs, is confirmed by the FDR reported in Table 6.9. The values are obtained from the respective ROCs, whose AUCs are reported in Table 6.8. Each FDR has been computed for the lowest likelihood value that achieves a TDR of 100%. Note that, the FDRs reported in Table 6.9 can not to be compared to the points for TDRs of 100% in the curves depicted in Figure 6.24. Differently from the ROCs, where the average curves have been computed considering both TDR

Table 6.9: TDR and FDR values achieved with the best models.

Resource	Res.	TDR (%)	FDR (%)			Model
			AVG	BEST BG	BEST	
G	1	100	33.36	24.19	3.27	GMM
G	1	100	17.11	9.06	< 0.01	HMM
G	10	100	46.23	33.90	7.40	GMM
G	10	100	20.80	11.10	0.13	HMM
W	1	100	44.31	31.91	1.37	GMM
W	1	100	13.79	7.69	< 0.01	HMM
W	10	100	50.60	48.04	13.06	GMM
W	10	100	21.37	11.66	< 0.01	HMM

and FDR points, the results in Table 6.9 have been achieved performing the average of the FDR components only.

Table 6.10: Further features combinations.

Resource	Res.	Features comb.	AUC (%)	Model
W	1	MA+ENE+ΔLWE	80.55%	GMM
W	1	MA+LWE+ΔWDE+LWE	88.73%	HMM
W	10	MA+ENE+ΔLWE+ΔMA+LWE	69.49%	GMM
W	10	MA+LWE+ENE	80.63%	HMM

In order to evaluate the goodness of the revised SFS, further features combinations have been evaluated, among those features considered better-promising, which are: MA, ENE, LWE, ΔLWE, and ΔMA. For the natural gas case, none of the combination, at both resolutions, achieved better results than the ones already achieved. Conversely, the water results exhibited a perceptible gap from the original results in Table 6.8. The results achieved by the new combinations are reported in Table 6.10. Among these combinations, the one used for the water case, with 1 minute resolution and adopting the GMM, denotes a significant improvement. For this combination, in addition to the reported AUC, the BEST BG and BEST AUCs rose to 88.69% and 99.17%, respectively. For the HMM at 10 minutes resolution, the best features combinations remains the one selected during the original feature selection. Moreover, concerning the water case at 10 minutes resolution with GMM, the new combination does not shown a clear improvement, which is slightly greater than 1%, but its features number is lower than the original one. Specifically, even if the new vector has lost one component only, this is the heavier one, DATA, resulting in a drastic reduction in the number of components in the features vector. Finally, this last test confirmed the effectiveness of the adopted SFS procedure. Indeed, only for one condition, water at 1 minute resolution with GMM, a noteworthy improvement has been achieved for a combination not evaluated in the original features selection.

Chapter 7

Conclusions

In this work, the issues relating to the management of energy and resources in Micro Grid environments have been investigated, and the basis for the implementation of an energy and resource manager have been laid.

Through the analysis of the literature, the MILP technique has been deemed the most suitable approach to implement an energy and resource manager in a real life Micro Grid environment. By exploiting the linear nature of the technique, the objective function and the matrix of constraints, representing the main management problem, has been decomposed in blocks, each corresponding to the model of a device in the system. This result has allowed to implement the optimization problem as an arrangement of models, whose configuration has been defined in order to match the topology of a given Micro Grid environment. Once the general approach has been defined, two topologies have been chosen to investigate the main issues of the optimization problem.

The task scheduling and thermal model scenario presented in Section 3.2.1 has been used to evaluate the effect of data uncertainty on the management performance. On purpose, several feature have been implemented, such as task scheduling and thermal energy management, along with local energy production and storage. To limit the complexity of the study, a simplified Micro Grid topology has been used. On purpose, also, a Radial Basis Function Network based forecaster has been used to predict the solar irradiation and then the solar energy production. A genetic algorithm based solver, on the other hand, has been used to compute the thermal needs based on the thermal model of the environment.

The multi apartment revamping model scenario 3.2.2 has been proposed to evaluate how the availability of resources and the topology of the system affects the system performance. On purpose, with respect to the previous scenario, a more complex structure has been devised to model different Micro Grid configurations. In this case, a long term analysis has been carried out to limit the effect of seasonality. Nonetheless, to limit the complexity of the simulation, some features such as task scheduling and thermal energy management has been discarded, also a 1-hour ahead management approach has been used.

Furthermore, the multi apartment revamping model scenario 3.2.2 has been also used to investigate the performance of the management process for different time frame length. In this case both 1-hour ahead and 1-day ahead management have been evaluated and compared. Additionally, the effect of dynamic pricing has been evaluated as well.

The results of the simulations provided several insight on the management problem. The study based on the task scheduling and thermal model scenario revealed that data uncertainty affect the management process indirectly. The data uncertainty of the temperature forecast can affect the computation of the thermal needs through the expected internal temperature value, thus its effect appear only if a temperature target has been assigned by the user. The task scheduling and energy management, on the other hand, are affected by the data uncertainty on energy availability and price forecast. However, rather than the absolute value of the forecast error, what affects the management is the relative value of the forecast error. In fact the data uncertainty can alter the position of the local maxima of the energy availability and the local minima of the energy price, thus leading the manager to select the wrong time slot. On the other hand, a constant, systematic error, would not affect the management process. The task scheduling, however, is fairly robust against this issue, because the constraints on the task limits the width of the active time window in which the task execution take place.

The evaluation based on the multi apartment revamping model scenario with a 1-hour ahead management scheme, revealed that the the topology of the system can greatly affect the management performance. Although the availability of an energy storage system play a major role in the improvement of the Micro Grid efficiency, actually the integration of the subsystems is important as well. In fact, the ability to route the electrical energy surplus towards the thermal subsystem can lower the energy waste. Similarly, the ability to route the energy surplus from the solar thermal collector for hot water production toward the thermal system can greatly lower the energy waste.

With regard to the multi apartment revamping model scenario with a 1-day ahead management scheme, the analysis revealed that not only it can achieve greater saving with respect to the 1-hour ahead management scheme counterpart. It is also less affected by data uncertainty, even more so when the degree of integration among the subsystems is high.

To further improve the energy management process, a model of real life solar power plant, that can take into account the effects the maximum power point tracking algorithm and the effects of partial shading, has been devised and implemented. The model can provide a more accurate computation of the energy yield with respect to more simplified models, thus it can be used with solar irradiation forecast to provide a better estimation of the energy yield.

Moreover, the evaluation process has revealed that novel MPPT algorithm can be implemented, so that the performance of the plant can be improved with respect to nowadays technology.

To support the integration of the model in a real life Micro Grid environment, a NILM algorithm has been investigated and further developed, as a mean to provide a feedback from the system to the manager. Different configurations of the algorithm have been evaluated under different time window sizes, revealing improvement with respect to the state of the art, as well as room for further enhancement.

Lastly, to expand the resource management abilities so that water and gas can be subject to automated management, a scheduling algorithm aimed at ultra low power micro devices has been devised. The goal is to improve the performance of renewable energy supplied wireless sensors to be used in automated meter reading systems. The algorithm, namely EA-LSA has proven the ability to improve the reliability of the devices by enhancing the use of energy and adapting the device activity to the energy availability.

To the same end, a novelty detection algorithm has been investigated and improved to operate as a leakage detection algorithm. Combined with an automated meter reading system, the meter records can be investigated to identify the occurrence of leakages, so that the user can be notified of the problem, and the resource manager can process the information to correctly assess the actual resource consumption.

7.1 Future research topics

Smart and Micro Grids are regarded as the next generation power grids technology, and are currently being researched on to improve the efficiency of power distribution. Although the research has reached the experimental stage much effort is still required in order to achieved the necessary maturity level. To that end, the work described in this dissertation represents the groundwork towards an energy and resource manager that can effectively be implemented in a real life Micro Grid environment.

To proceed further in the implementation, however, a more suitable framework is required, and the technologies discussed in this dissertation must be further developed and enhanced.

On purpose, the approach described in this work is being used as a reference to develop an energy and resource manager by means of the Python programming language. The development of the new framework aims to enhance and integrate many of the feature proposed.

Although the management approach is based on a MILP problem obtained as the arrangement of the device models, the arrangement process is not automated yet, thus further research is required to overcome this limitation. Also, the availability of specialized tools aimed at thermal modelling, can be integrated to improve the management of the user comfort. Regarding the task scheduling, although supported by the energy and resource manager, it shows many limitations since the manager cannot interact with the controller of the appliances.

On the other hand, the photovoltaic power plant model with support to partial shading and the Non Invasive Load Monitoring approach, although not fully developed yet, can be integrated in the manager. Nonetheless, the photovoltaic power plant model still requires many refinement, whereas the NILM approach has to be extended to fully exploit the potential of the information regarding the reactive energy consumption.

Similarly, even though the EA-LSA and the leakage detection algorithms have proven their effectiveness, they are still in the experimental phase and thus additional refinements and testing are required.

List of Publications

- M. Severini, S. Squartini and F. Piazza "Hybrid Soft Computing algorithmic framework for Smart Home Energy Management", Soft Computing 17.11 (2013): 1983-2005.
- M. Severini, S. Squartini, e F. Piazza, "Energy-aware lazy scheduling algorithm for energy-harvesting sensor nodes", Neural Computing and Applications 23.7-8 (2013): 1899-1908.
- G. Comodi, A. Giantomassi, M. Severini, S. Squartini, F. Ferracuti, A. Fontia, D. N. Cesarini, M. Morodo, F. Polonara, "Multi-apartment residential microgrid with electrical and thermal storage devices: experimental analysis and simulation of energy management strategies", Applied Energy, 2015, 137: 854-866
- M. Severini, S. Squartini, F. Piazza e M. Conti "Energy-Aware Task Scheduler for Self-Powered Sensor Nodes: from Model to Firmware", Ad Hoc Networks 24 (2015): 73-91
- M. Fagiani, S. Squartini, L. Gabrielli, M. Severini, and F. Piazza, "A statistical framework for automatic leakage detection in smart water and gas grids," Energies, vol. 9, no. 9, p. 665, 2016.
- M. Severini, S. Squartini and F. Piazza, "An energy aware approach for task scheduling in energy-harvesting sensor nodes", ISNN'12 Proceedings of the 9th international conference on Advances in Neural Networks - Volume Part II, pp. 601-610, 2012.
- M. Severini, S. Squartini and F. Piazza, "Computational framework based on task and resource scheduling for micro grid design", in Proceedings of IJCNN 2014, Beijing, China, 2014
- M. Severini, S. Squartini, M. Fagiani and F. Piazza, "Energy management with the support of dynamic pricing strategies in real micro-grid scenarios," 2015 International Joint Conference on Neural Networks (IJCNN), Killarney, 2015, pp. 1-8.
- M. Fagiani, S. Squartini, M. Severini and F. Piazza, "A novelty detection approach to identify the occurrence of leakage in smart gas and water

grids," 2015 International Joint Conference on Neural Networks (IJCNN), Killarney, 2015, pp. 1-8.

- M. Severini, A. Scorrano, S. Squartini, M. Fagiani and F. Piazza, "SW Framework for simulation and evaluation of partial shading effects in configurable PV systems," 2016 IEEE International Conference on Environment and Electrical Engineering (IEEE EEEIC), Florence, 2016, pp. 1-6.

- M. Fagiani, S. Squartini, R. Bonfigli, M. Severini and F. Piazza, "Exploiting Temporal Features and Pressure Data for Automatic Leakage Detection in Smart Water Grids," 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver. To appear.

- . R. Bonfigli, M. Severini, S. Squartini, M. Fagiani and F. Piazza, "Improving the performance of the AFAMAP algorithm for Non-Intrusive Load Monitoring," 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver. To appear.

- M. Severini, S. Squartini, G.P. Surace and F. Piazza, "Smart Home Task and Energy Resource Scheduling based on Nonlinear Programming", Recent Advances of Neural Network Models and Applications. Springer International Publishing, 2014. 175-185

- M. Severini, S. Squartini, F. Piazza, "Energy Demand Management Through Uncertain Data Forecasting: An Hybrid Approach", Frontiers of Intelligent Control and Information Processing, World Scientific Publishing, August 2014

- S. Spinsante, S. Squartini, P. Russo, A. De Santis, M. Severini, M. Fagiani, V. Di Mattia, and R. Minerva, "Enabling Technologies for IoT-Oriented Sustainable Smart Gas and Water Grids: Communication Protocols and Data Analysis," in *Internet of Things Concepts, Technologies, Applications, and Implementations*, CRC Press, Florida, USA. Minor revision.

Bibliography

[1] Hiren Patel and Vivek Agarwal, “Maximum power point tracking scheme for pv systems operating under partially shaded conditions,” *Industrial Electronics, IEEE Transactions on*, vol. 55, no. 4, pp. 1689–1698, 2008.

[2] C. Cecati, G. Mokryani, A. Piccolo, and P. Siano, “An overview on the smart grid concept,” in *IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society*, nov. 2010, pp. 3322 –3327.

[3] A. Ipakchi and F. Albuyeh, “Grid of the future,” *Power and Energy Magazine, IEEE*, vol. 7, no. 2, pp. 52 –62, march-april 2009.

[4] P. Khajavi, A. B. Arani, and H. Monsef, “Identification of appropriate buses for implementation of smart grid in order to improve system efficiency,” in *Environment and Electrical Engineering (EEEIC), 2011 10th International Conference on*, 2011, pp. 1–4.

[5] M. Vaziri, S. Vadhva, T. Oneal, and M. Johnson, “Distributed generation issues, and standards,” in *Information Reuse and Integration (IRI), 2011 IEEE International Conference on*, 2011, pp. 439–443.

[6] C.J. Mozina, “Impact of green power generation on distribution systems in a smart grid,” in *Power Systems Conference and Exposition (PSCE), 2011 IEEE/PES*, 2011, pp. 1–8.

[7] Di Zhang, Lazaros G. Papageorgiou, Nouri J. Samsatli, and Nilay Shah, “Optimal scheduling of smart homes energy consumption with microgrid,” in *ENERGY 2011, The First International Conference on Smart Grids, Green Communications and IT Energy-aware Technologies*, 2011, pp. 70 –75.

[8] Eleonora Annunziata, Marco Frey, and Francesco Rizzi, “Towards nearly zero-energy buildings: The state-of-art of national regulations in europe,” *Energy*, vol. 57, no. 0, pp. 125 – 133, 2013.

[9] Umberto Desideri, Livia Arcioni, Daniela Leonardi, Luca Cesaretti, Perla Perugini, Elena Agabitini, and Nicola Evangelisti, “Design of a multipurpose "zero energy consumption" building according to european directive

2010/31/eu: Architectural and technical plants solutions," *Energy*, vol. 58, no. 0, pp. 157 – 167, 2013.

[10] M.V. Kirthiga, S.A. Daniel, and S. Gurunathan, "A methodology for transforming an existing distribution network into a sustainable autonomous micro-grid," *Sustainable Energy, IEEE Transactions on*, vol. 4, no. 1, pp. 31 –41, jan. 2013.

[11] A.M. Zein Alabedin, E.F. El-Saadany, and M.M.A. Salama, "Generation scheduling in microgrids under uncertainties in power generation," in *Electrical Power and Energy Conference (EPEC), 2012 IEEE*, 2012, pp. 133–138.

[12] Shi You, F. Marra, and C. Traeholt, "Integration of fuel cell micro-chps on low voltage grid: A danish case study," in *Power and Energy Engineering Conference (APPEEC), 2012 Asia-Pacific*, march 2012, pp. 1 –4.

[13] Se-Hwan Jang, Jong-Bae Park, Jae Hyung Roh, Sung-Yong Son, and K.Y. Lee, "Short-term resource scheduling for power systems with energy storage systems," in *Power and Energy Society General Meeting, 2012 IEEE*, 2012, pp. 1–7.

[14] F. De Angelis, M. Boaro, D. Fuselli, S. Squartini, and F. Piazza, "A comparison between different optimization techniques for energy scheduling in smart home environment," in *Neural Nets and Surroundings*, pp. 311–320. Springer Berlin Heidelberg, 2013.

[15] J. Soares, H. Morais, T. Sousa, Z. Vale, and P. Faria, "Day-ahead resource scheduling including demand response for electric vehicles," *Smart Grid, IEEE Transactions on*, in press.

[16] L. Hernandez, C. Baladron, J.M. Aguiar, B. Carro, A. Sanchez-Esguevillas, J. Lloret, D. Chinarro, J.J. Gomez-Sanz, and D. Cook, "A multi-agent system architecture for smart grid management and forecasting of energy demand in virtual power plants," *Communications Magazine, IEEE*, vol. 51, no. 1, pp. 106 –113, january 2013.

[17] Hong Cui and Wenliang Dai, "Multi-objective optimal allocation of distributed generation in smart grid," in *Electrical and Control Engineering (ICECE), 2011 International Conference on*, sept. 2011, pp. 713 –717.

[18] D. Fuselli, F. De Angelis, M. Boaro, S. Squartini, Q. Wei, D. Liu, and F. Piazza, "Action dependent heuristic dynamic programming for home energy resource scheduling," *International Journal of Electrical Power & Energy Systems*, vol. 48, pp. 148–160, 2013.

[19] D. Fuselli, F. De Angelis, M. Boaro, D. Liu, Q. Wei, S. Squartini, and F. Piazza, "Optimal battery management with adhdp in smart home environments," *Advances in Neural Networks–ISNN 2012*, pp. 355–364, 2012.

[20] Matteo Boaro, Danilo Fuselli, Francesco De Angelis, Derong Liu, Qinglai Wei, and Francesco Piazza, "Adaptive dynamic programming algorithm for renewable energy scheduling and battery management," *Cognitive Computation*, in press.

[21] C.-K. Tham and T. Luo, "Sensing-driven energy purchasing in smart grid cyber-physical system," *Systems, Man, and Cybernetics: Systems, IEEE Transactions on*, in press.

[22] K. Nagasaka, K. Ando, Y.B. Xu, H. Takamori, J. Wang, A. Mitsuta, O. Saito, and E. Go, "A research on operation planning of multi smart micro grid," in *Advanced Mechatronic Systems (ICAMechS), 2012 International Conference on*, sept. 2012, pp. 351 –356.

[23] Severini Marco, Stefano Squartini, Gian Piero Surace, and Francesco Piazza, *Smart Home Task and Energy Resource Scheduling Based on Nonlinear Programming*, pp. 175–185, Springer International Publishing, Cham, 2014.

[24] M.C. Bozchalui and R. Sharma, "Analysis of electric vehicles as mobile energy storage in commercial buildings: Economic and environmental impacts," in *Power and Energy Society General Meeting, 2012 IEEE*, july 2012, pp. 1 –8.

[25] F. De Angelis, M. Boaro, D. Fuselli, S. Squartini, F. Piazza, and Q. Wei, "Optimal home energy management under dynamic electrical and thermal constraints," *Industrial Informatics, IEEE Transactions on*, in press.

[26] N. Sundararajan, P. Saratchandran, and Y. Li, *Fully tuned radial basis function neural networks for flight control*, Kluver Academic, London, Great Britain, 2002.

[27] A. Giantomassi, G. Ippoliti, S. Longhi, I. Bertini, and S. Pizzuti, "Online steam production prediction for a municipal solid waste incinerator by fully tuned minimal rbf neural networks," *Journal of Process Control*, vol. 21, no. 1, pp. 164 – 172, 2011.

[28] M.L. Corradini, V. Fossi, A. Giantomassi, G. Ippoliti, S. Longhi, and G. Orlando, "Minimal resource allocating networks for discrete time sliding mode control of robotic manipulators," *Industrial Informatics, IEEE Transactions on*, vol. 8, no. 4, pp. 733–745, Nov 2012.

[29] Guang-Bin Huang, Hongming Zhou, Xiaojian Ding, and Rui Zhang, "Extreme learning machine for regression and multiclass classification," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 42, no. 2, pp. 513–529, April 2012.

[30] Guang-Bin Huang, "An insight into extreme learning machines: Random neurons, random features and kernels," *Cognitive Computation*, vol. 6, no. 3, pp. 376–390, 2014.

[31] NitinAnand Shrivastava, BijayaKetan Panigrahi, and Meng-Hiot Lim, "Electricity price classification using extreme learning machines," *Neural Computing and Applications*, pp. 1–10, 2014.

[32] RafaÅ Weron, "Electricity price forecasting: A review of the state-of-the-art with a look into the future," *International Journal of Forecasting*, vol. 30, no. 4, pp. 1030 – 1081, 2014.

[33] Francesco Lisi and Fany Nan, "Component estimation for electricity prices: Procedures and comparisons," *Energy Economics*, vol. 44, pp. 143 – 159, 2014.

[34] Gao Huang, Guang-Bin Huang, Shiji Song, and Keyou You, "Trends in extreme learning machines: A review," *Neural Networks*, vol. 61, pp. 32 – 48, 2015.

[35] F. Ziel, R. Steinert, and S. Husmann, "Efficient modeling and forecasting of electricity spot prices," *Energy Economics*, vol. 47, pp. 98–111, 2015.

[36] V. Kekatos, Y. Zhang, and G.B. Giannakis, "Electricity market forecasting via low-rank multi-kernel learning," *IEEE Journal on Selected Topics in Signal Processing*, vol. 8, no. 6, pp. 1182–1193, 2014.

[37] George Dantzig, *Linear Programming and Extensions*, Princeton University Press, 1963.

[38] A. H. Land and A. G Doig, "An automatic method of solving discrete programming problems," *Econometrica*, vol. 28, no. 3, pp. 497–520, 1960.

[39] Maurizio Cellura, Lucia Campanella, Giuseppina Ciulla, Francesco Guarino, D Nardi Cesarini, V Lo Brano, and Aldo Orioli, "A net zero energy building in italy: design studies to reach the net zero energy target," in *Proceedings of building simulation*, 2011.

[40] Maurizio Cellura, Giuseppina Ciulla, Valerio Lo Brano, Aldo Orioli, Lucia Campanella, Francesco Guarino, and Davide Nardi Cesarini, "The redesign of an italian building to reach net zero energy performances: A

case study of the shc task 40-ecbcs annex 52.," *ASHRAE transactions*, vol. 117, no. 2, 2011.

[41] Egidijus Kazanavičius, Antanas Mikuckas, Irena Mikuckienė, and Jonas Čeponis, "The heat balance model of residential house," in *Information Technology and Control, ISNN*, nov. 2006, vol. 35, pp. 391–396.

[42] B. Qela and H. Mouftah, "Simulation of a house heating system using C# – an energy conservation perspective," in *Electrical and Computer Engineering (CCECE), 2010 23rd Canadian Conference on*, may 2010, pp. 1 –5.

[43] Samuel F. Fux, Araz Ashouri, Michael J. Benz, and Lino Guzzella, "EKF based self-adaptive thermal model for a passive house," *Energy and Buildings*, 2012.

[44] "EN 12831:2003 heating systems in buildings - method for calculation of the design heat load," 2003.

[45] "EN ISO 13370:2007 thermal performance of buildings - heat transfer via the ground - calculation methods ISO 13370:2007," 2007.

[46] "EN ISO 13789:2007 thermal performance of buildings - transmission and ventilation heat transfer coefficients - calculation method ISO 13789:2007," 2007.

[47] SN Sivanandam and SN Deepa, *Introduction to genetic algorithms*, Springer Publishing Company, Incorporated, 2007.

[48] A. Arabali, M. Ghofrani, M. Etezadi-Amoli, M. S. Fadali, and Y. Baghzouz, "Genetic-algorithm-based optimization approach for energy management," *Power Delivery, IEEE Transactions on*, vol. 28, no. 1, pp. 162–170, 2013.

[49] L. Yingwei, N. Sundararajan, and P. Saratchandran, "Performance evaluation of a sequential minimal radial basis function (RBF) neural network learning algorithm," *IEEE Trans. Neural Networks*, vol. 9, no. 2, pp. 308–318, 1998.

[50] Marco Severini, Stefano Squartini, and Francesco Piazza, "Hybrid soft computing algorithmic framework for smart home energy management," *Soft Computing*, vol. 17, no. 11, pp. 1983–2005, 2013.

[51] Chih-Tang Sah, Robert Noyce, and William Shockley, "Carrier generation and recombination in pn junctions and pn junction characteristics," *Proceedings of the IRE*, vol. 45, no. 9, pp. 1228–1243, 1957.

[52] Huan-Liang Tsai, Ci-Siang Tu, Yi-Jie Su, et al., "Development of generalized photovoltaic model using matlab/simulink," in *WCECS 2008*, 2008, vol. 2008, pp. 1–6.

[53] Volker Quaschning and Rolf Hanitsch, "Numerical simulation of current-voltage characteristics of photovoltaic systems with shaded solar cells," *Solar Energy*, vol. 56, no. 6, pp. 513–520, 1996.

[54] Kashif Ishaque, Zainal Salam, et al., "A comprehensive matlab simulink pv system simulator with partial shading capability based on two-diode model," *Solar Energy*, vol. 85, no. 9, pp. 2217–2227, 2011.

[55] Kashif Ishaque, Zainal Salam, and Hamed Taheri, "Simple, fast and accurate two-diode model for photovoltaic modules," *Solar Energy Materials and Solar Cells*, vol. 95, no. 2, pp. 586–594, 2011.

[56] Hiren Patel and Vivek Agarwal, "Matlab-based modeling to study the effects of partial shading on pv array characteristics," *Energy Conversion, IEEE Transactions on*, vol. 23, no. 1, pp. 302–310, 2008.

[57] Chihchiang Hua, Jongrong Lin, and Chihming Shen, "Implementation of a dsp-controlled photovoltaic system with peak power tracking," *Industrial Electronics, IEEE Transactions on*, vol. 45, no. 1, pp. 99–107, 1998.

[58] FAO Aashoor and FVP Robinson, "A variable step size perturb and observe algorithm for photovoltaic maximum power point tracking," in *Universities Power Engineering Conference (UPEC), 2012 47th International*. IEEE, 2012, pp. 1–6.

[59] Dezso Sera, Laszlo Mathe, Tamas Kerekes, Sergiu Viorel Spataru, and Remus Teodorescu, "On the perturb-and-observe and incremental conductance mppt methods for pv systems," *IEEE J. Photovolt.*, vol. 3, no. 3, pp. 1070–1078, 2013.

[60] O Wasynezuk, "Dynamic behavior of a class of photovoltaic power systems," *IEEE Trans. Power App. Syst.*, , no. 9, pp. 3031–3037, 1983.

[61] Farah Kazan, Sami Karaki, Rabih A Jabr, and Moussa Mansour, "Maximum power point tracking using ripple correlation and incremental conductance," in *Universities Power Engineering Conference (UPEC), 2012 47th International*. IEEE, 2012, pp. 1–6.

[62] Gregory Joseph Kish, John Jaehwan Lee, and Peter Lehn, "Modelling and control of photovoltaic panels utilising the incremental conductance

method for maximum power point tracking," *Renewable Power Generation, IET*, vol. 6, no. 4, pp. 259–266, 2012.

[63] Yuansheng Xiong, Suxiang Qian, and Jianming Xu, "Research on constant voltage with incremental conductance mppt method," in *Power and Energy Engineering Conference (APPEEC), 2012 Asia-Pacific*. IEEE, 2012, pp. 1–4.

[64] Kaumil B Shah and Lokin P Joshi, "Comparative analysis of incremental conductance base mppt for multi-string photovoltaic system," in *Engineering (NUiCONE), 2013 Nirma University International Conference on*. IEEE, 2013, pp. 1–6.

[65] Marcelo Gradella Villalva, Jonas Rafael Gazoli, et al., "Comprehensive approach to modeling and simulation of photovoltaic arrays," *Power Electronics, IEEE Transactions on*, vol. 24, no. 5, pp. 1198–1208, 2009.

[66] Hajime Kawamura, Kazuhito Naka, Norihiro Yonekura, Sanshiro Yamanaka, Hideaki Kawamura, Hideyuki Ohno, and Katsuhiko Naito, "Simulation of i–v characteristics of a pv module with shaded pv cells," *Solar Energy Materials and Solar Cells*, vol. 75, no. 3, pp. 613–621, 2003.

[67] Huiying Zheng, Shuhui Li, Rajab Challoo, and Julio Proano, "Shading and bypass diode impacts to energy extraction of pv arrays under different converter configurations," *Renewable Energy*, vol. 68, pp. 58–66, 2014.

[68] E Diaz-Dorado, A Suarez-Garcia, C Carrillo, and J Cidras, "Influence of the shadows in photovoltaic systems with different configurations of bypass diodes," *IEEE Speedam 2010*, pp. 134–139, 2010.

[69] R Ramaprabha and BL Mathur, "A comprehensive review and analysis of solar photovoltaic array configurations under partial shaded conditions," *International Journal of Photoenergy*, vol. 2012, 2012.

[70] ENEA Fonti Rinnovabili, "Atlante italiano della radiazione solare," 2015.

[71] J. Zico Kolter and Tommi Jaakkola, "Approximate inference in additive factorial HMMs with application to energy disaggregation.," in *AISTATS*, Neil D. Lawrence and Mark Girolami, Eds. 2012, vol. 22 of *JMLR Proceedings*, pp. 1472–1482, JMLR.org.

[72] Stephen Makonin and Fred Popowich, "Nonintrusive load monitoring (NILM) performance evaluation," *Energy Efficiency*, vol. 8, no. 4, pp. 809–814, 2014.

[73] Dominik Egarter, Manfred Pöchacker, and Wilfried Elmenreich, "Complexity of power draws for load disaggregation," *CoRR*, 2015.

[74] R. Bonfigli, S. Squartini, M. Fagiani, and F. Piazza, "Unsupervised algorithms for non-intrusive load monitoring: An up-to-date overview," in *Environment and Electrical Engineering (EEEIC), 2015 IEEE 15th International Conference on*, June 2015, pp. 1175–1180.

[75] Claudia Beleites, Reiner Salzer, and Valter Sergo, "Validation of soft classification models using partial class memberships: An extended concept of sensitivity and co. applied to grading of astrocytoma tissues," *Chemometrics and Intelligent Laboratory Systems*, vol. 122, pp. 12 – 22, 2013.

[76] Stephen Makonin, Fred Popowich, Lyn Bartram, Bob Gill, and Ivan V. Bajic, "AMPds: A public dataset for load disaggregation and eco-feedback research," in *Proceedings of the 2013 IEEE Electrical Power and Energy Conference (EPEC)*, 2013.

[77] Marco Severini, Stefano Squartini, and Francesco Piazza, "An energy aware approach for task scheduling in energy-harvesting sensor nodes," *ISNN'12 Proceedings of the 9th international conference on Advances in Neural Networks - Volume Part II*, pp. 601–610, 2012.

[78] Marco Severini, Stefano Squartini, and Francesco Piazza, "Energy-aware lazy scheduling algorithm for energy-harvesting sensor nodes," *Neural Computing and Applications*, pp. 1–10, 2012.

[79] Clemens Moser, Davide Brunelli, Lothar Thiele, and Luca Benini, "Lazy scheduling for energy-harvesting sensor nodes," *Fifth Working Conference on Distributed and Parallel Embedded Systems, DIPES 2006*, pp. 144–149, 2006.

[80] Clemens Moser, Davide Brunelli, Lothar Thiele, and Luca Benini, "Real-time scheduling for energy harvesting sensor nodes," *Real-Time Systems Journal*, vol. 3, pp. 233–260, 2007.

[81] Hongya Wang, Jie Jin, Zhijun Wang, and LihChyun Shu, "On a novel property of the earliest deadline first algorithm," in *Fuzzy Systems and Knowledge Discovery (FSKD), 2011 Eighth International Conference on*, july 2011, vol. 1, pp. 197 –201.

[82] E. Okuyan and B. Kayayurt, "Earliest deadline first scheduling algorithm and its use in ANKA UAV," in *Digital Avionics Systems Conference (DASC), 2012 IEEE/AIAA 31st*, oct. 2012, pp. 8B1–1 –8B1–8.

[83] M. Chetto and A. Queudet, "A note on EDF scheduling for real-time energy harvesting systems," *Computers, IEEE Transactions on*, vol. 63, no. 4, pp. 1037–1040, April 2014.

[84] Daniela Krüger, Carsten Buschmann, and Stefan Fischer, "Solar powered sensor network design and experimentation," *Wireless Communication Systems, 2009. ISWCS 2009. 6th International Symposium on*, pp. 11–15.

[85] Cymbet Corporation, *CBC-EVAL-09 EnerChip EP Universal EH Eval Kit*, 2011.

[86] *eZ430-RF2500 Development Tool*, 2009.

[87] Texas Instruments, Inc., *MSP430x22x2, MSP430x22x4 Mixed Signal Microcontroller Data Sheet (SLAS 504F)*, 2011.

[88] Texas Instruments, Inc., *Low-Cost Low-Power 2.4 GHz RF Transceiver (SWRS 040C)*, 2009.

[89] Cymbet Corporation, *CBC915 EnerChip Energy Processor for Energy Harvesting Applications*, 2011.

[90] Keith Quiring and William Goh, "Msp430 interface to cc1100/2500 code library," Tech. Rep., Texas Instruments, Inc., 2010.

[91] *SimpliciTI Overview (Rev.B)*, 2008.

[92] Siri Namtvedt, *Design Note DN503 - SPI Access*, 2007.

[93] Magnus Wines, *Application Note NO49 - Software for CC1100/CC2500 and MSP430*, 2007.

[94] Randy Wu, "Clear channel assessment using ti msp430 and cc2500 radio," Tech. Rep., Texas Instruments, Inc., 2007.

[95] Fabio Di Franco, Christos Tachtatzis, Ben Graham, Marek Bykowski, David C. Tracey, Nick F. Timmons, and Jim Morrison, "Current characterisation for ultra low power wireless body area networks," in Intelligent Solutions in Embedded Systems (WISES), 2010 8th Workshop on, 2010.

[96] Mirko Carloni, Rocco d'Aparo, Pierpaolo Scorrano, Berardo Naticchia, and Massimo Conti, "A micropower supervisor for wireless nodes with a digital pulse frequency modulator battery monitor," *Proc. SPIE*, vol. 8764, pp. 87640P–87640P–12, 2013.

[97] Miguel Morales, "Wireless sensor monitor using the ez430-rf2500 (slaa 378b)," Tech. Rep., Texas Instruments, Inc., 2008.

[98] Analog Devices, Inc., *Low Drift, Low Power Instrumentation Amplifier AD621*, 2001.

[99] M.T. Nasir, M. Mysorewala, L. Cheded, B. Siddiqui, and M. Sabih, “Measurement Error Sensitivity Analysis for Detecting and Locating Leak in Pipeline using ANN and SVM,” in *Multi-Conference on Systems, Signals Devices (SSD), 2014 11th International*, Feb 2014, pp. 1–4.

[100] Gal Oren and Nerya Y. Stroh, “Mathematical Model for Detection of Leakage in Domestic Water Supply Systems by Reading Consumption from an Analogue Water Meter,” *International Journal of Environmental Science and Development*, vol. 4, no. 4, pp. 386–389, 2013.

[101] G. Boracchi and M. Roveri, “Exploiting Self-Similarity for Change Detection,” in *Neural Networks (IJCNN), 2014 International Joint Conference on*, July 2014, pp. 3339–3346.

[102] M.M. Gamboa-Medina, L.F. Ribeiro Reis, and R. Capobianco Guido, “Feature Extraction in Pressure Signals for Leak Detection in Water Networks,” *Procedia Engineering*, vol. 70, no. 0, pp. 688–697, 2014.

[103] JaberM.A. Alkasseh, MohdNordin Adlan, Ismail Abustan, HamidiAbdul Aziz, and AbuBakarMohamad Hanif, “Applying Minimum Night Flow to Estimate Water Loss Using Statistical Modeling: A Case Study in Kinta Valley, Malaysia,” *Water Resources Management*, vol. 27, no. 5, pp. 1439–1455, 2013.

[104] G. Sanz, R. Perez, and A. Escobet, “Leakage Localization in Water Networks using Fuzzy Logic,” in *Control Automation (MED), 20th Mediterranean Conference on*, 2012, pp. 646–651.

[105] Marco Fagiani, Stefano Squartini, Leonardo Gabrielli, Susanna Spinsante, and Francesco Piazza, “A Review of Datasets and Load Forecasting Techniques for Smart Natural Gas and Water Grids: Analysis and Experiments,” *Neurocomputing, Special Issue*, 2015, to appear.

[106] S. Ntalampiras, I. Potamitis, and N. Fakotakis, “Probabilistic Novelty Detection for Acoustic Surveillance Under Real-World Conditions,” *Multimedia, IEEE Transactions on*, vol. 13, no. 4, pp. 713–719, Aug 2011.

[107] Emanuele Principi, Stefano Squartini, Roberto Bonfigli, Giacomo Ferroni, and Francesco Piazza, “An Integrated System for Voice Command Recognition and Emergency Detection Based on Audio Signals,” *Expert Systems with Applications*, vol. 42, no. 13, pp. 5668–5683, 2015.

[108] S. Makonin, F. Popowich, L. Bartram, B. Gill, and I.V. Bajic, “AMPds: A Public Dataset for Load Disaggregation and Eco-Feedback Research,” in *Electrical Power Energy Conference (EPEC), 2013 IEEE*, Aug 2013, pp. 1–6.

[109] Sergios Theodoridis and Konstantinos Koutroumbas, *Pattern Recognition, Fourth Edition*, Academic Press, Burlington, 4th edition, 2008.

[110] Tracy C. Britton, Rodney A. Stewart, and Kelvin R. O'Halloran, "Smart Metering: Enabler for Rapid and Effective Post Meter Leakage Identification and Water Loss Management," *Journal of Cleaner Production*, vol. 54, no. 0, pp. 166–176, 2013.

[111] Douglas A. Reynolds, Thomas F. Quatieri, and Robert B. Dunn, "Speaker Verification Using Adapted Gaussian Mixture Models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[112] L. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, Feb 1989.